Timo M. R. Alho

Variable Voltage-based Power Management of DC Microgrids

With Focus on Shipboard Power Systems

ACTA WASAENSIA 574

University of Vaasa
VAASAN YLIOPISTO

Portions of the technical solutions described in this dissertation are subject to a pending patent application (No. 10 2025 103 705.1), filed on January 31, 2025, at the German Patent Office (DPMA). The patent application is unpublished at the time of publication of this work.

ISBN 978-952-395-236-2 (print)
978-952-395-237-9 (online)
ISSN 0355-2667 (Acta Wasaensia 574, print)
2323-9123 (Acta Wasaensia 574, online)
URN http://urn.fi/URN:ISBN:978-952-395-237-9

PunaMusta Oy, Joensuu, 2025.

ACADEMIC DISSERTATION

To be presented, with the permission of the Board of the School of Technology and Innovations of the University of Vaasa, for public examination on the 25th of November, 2025, at noon.

Monograph dissertation of the School of Technology and Innovations at the University of Vaasa in the field of electrical engineering.

Author
Timo M. R. Alho
orcid.org/0000-0001-7541-867X

Supervisors
Professor Hannu Laaksonen
University of Vaasa
School of Technology and Innovations
Electrical Engineering
orcid.org/0000-0001-9378-8500

Assistant Professor Mustafa Alrayah Hassan Ibraheem
University of Vaasa
School of Technology and Innovations
Electrical Engineering
orcid.org/0000-0003-3232-8206

Custos
Professor Hannu Laaksonen
University of Vaasa
School of Technology and Innovations
Electrical Engineering
orcid.org/0000-0001-9378-8500

Reviewers
Professor Jorma Kyyrä
Aalto University
School of Electrical Engineering
Department of Electrical Engineering and Automation
Power Systems and Energy Conversion
orcid.org/0000-0002-8178-0613

Professor Pasi Peltoniemi
LUT University
LUT School of Energy Systems
Electrical Engineering
Laboratory of Electrical Drives Technology
orcid.org/0000-0001-5818-0794

Opponent
Associate Professor Navid Bayati
University of Southern Denmark
Institute of Mechanical and Electrical Engineering
Centre for Industrial Electronics
orcid.org/0000-0001-9247-0840

TIIVISTELMÄ

Yli vuosisadan ajan AC-sähkönjakelu on ollut laivojen sähköjärjestelmien vakiintunut standardi. Kasvavat vaatimukset polttoainetehokkuudelle ja päästövähennyksille ohjaavat kuitenkin kehitystä kohti tehoelektroniikkaan perustuvaa DC-jakelua. DC-mikroverkot mahdollistavat säädettävänopeuksisten generaattoreiden, energian varastoinnin ja uusiutuvien energialähteiden tehokkaamman integroinnin, mutta perinteiset ohjausstrategiat pohjautuvat AC-jakelun periaatteisiin, keskittyen jäykkään jännitteen säätelyyn järjestelmän joustavuuden ja vikasietoisuuden kustannuksella.

Tämä väitöskirja esittää vaihtelevaan jännitteeseen perustuvan tehonhallintastrategian DC-mikroverkoille, parantaen vikasietoisuutta ja autonomista jännitteenhallintaa. Toisin kuin keskitetyt menetelmät, strategia hyödyntää DC-jännitteen luonnollista vaihtelua, mahdollistaen hajautetun ja itseohjautuvan toiminnan. Kuormainvertterit, jotka eivät perinteisesti osallistu jännitteensäätöön, tukevat verkon vakautta ali- ja ylijännitesäätimiensä avulla. Tämä mahdollistaa kuormien itsenäisen tuen jännitelähteille, estäen lähteiden ylikuormitukset ja sähkökatkot.

Tutkimuksen keskeiset kontribuutiot ovat: (1) matemaattinen mallinnusparadigma, joka kuvaa DC-mikroverkon ja kuormainvertterien jännitesäätäjien toimintaa; (2) uudet ali- ja ylijännitesäädinalgoritmit, jotka mahdollistavat sujuvan rinnakkaistoiminnan muiden jännitettä ylläpitävien laitteiden kanssa; (3) uudet varaustilan tasapainotusalgoritmit energian varastoinnille, jotka toimivat yhteen ohjausstrategian kanssa ja takaavat häiriöttömän jännitetason; (4) strategian suorituskyvyn arviointi laivakäytössä, sisältäen tehonsyötön priorisointiin ja sähkökatkojen estämiseen liittyviä sovelluksia.

Mallinnus- ja kenttätestitulokset osoittavat, että esitetty hallintastrategia parantaa selvästi DC-mikroverkkojen vikasietoisuutta, energiatehokkuutta ja luotettavuutta merisovelluksissa. Tulokset viittaavat siihen, että menetelmä voi toimia vahvana vaihtoehtona perinteisille keskitetyn hallinnan järjestelmille, tarjoten skaalautuvan ja autonomisen ratkaisun tulevaisuuden laivasähköjärjestelmille.

Avainsanat: DC-mikroverkot, vaihtelevaan jännitteeseen perustuva hallinta, laivojen sähköjärjestelmät, vikasietoisuus, kuormainvertterien jännitesäätö, sähkökatkojen estäminen, tehonsyötön priorisointi.

ABSTRACT

For over a century, AC power distribution as proven technology has been the standard in shipboard power systems. However, increasing demands for fuel efficiency and emissions reduction are driving a shift toward DC distribution, enabled by power electronics-based technology. DC microgrids allow more efficient integration of variable-speed generators, energy storage, and renewables, but conventional control strategies remain based on AC-distribution's principles, prioritizing rigid voltage regulation over system flexibility and resilience.

This dissertation proposes a variable voltage-based control strategy for DC microgrids, enhancing fault resilience and autonomous power management in shipboard power systems. Unlike conventional methods that rely on centralized control, the proposed strategy leverages natural voltage variations, allowing distributed, self-regulating operation. Modern load inverters, traditionally passive in voltage control, actively contribute to grid stabilization through their undervoltage and overvoltage regulators, enabling loads to support voltage sources when needed. This prevents overloading and blackouts without requiring a supervisory system.

Key contributions of the research include: (1) the development of a mathematical modeling paradigm accurately representing the behavior of DC microgrids, including load inverter voltage regulators; (2) new load inverter under- and overvoltage regulator algorithms enabling smooth parallel operation with other voltage-maintaining equipment; (3) new energy storage state-of-charge balancing algorithms compatible with the control strategy, ensuring stable voltage levels; and (4) a detailed performance evaluation of the strategy in shipboard scenarios, including case studies on power supply prioritization and blackout prevention using real-world data.

Modeling results and real-world validations demonstrate that the proposed control strategy significantly improves the resilience, efficiency, and fault tolerance of DC microgrids in marine applications. The findings suggest that this approach can serve as a robust alternative to conventional centralized power management systems, offering a scalable and autonomous solution for next-generation shipboard power systems.

Keywords: DC microgrids, variable voltage-based control, shipboard power systems, fault resilience, inverter voltage regulation, blackout prevention, power supply prioritization.

“Any sufficiently advanced technology
is indistinguishable from magic.”

— Sir Arthur C. Clarke

ACKNOWLEDGEMENTS

This dissertation brings together thirteen years of accumulated understanding of DC microgrids from my own perspective. The aspiration to pursue a doctoral degree has always been present, but I chose to delay the effort until I felt ready to ensure that the dissertation would not be a mere formal exercise serving only my own advancement, but instead a work that provides tangible benefits to those working in the field. To reach that point, I first needed to build up the necessary expertise and know-how so that I could confidently present the hypotheses and solutions developed here.

Altogether, the dissertation has required nearly two years of active—and at times less active—writing and modeling to reach its present form. I would like to thank my supervisor, Professor Hannu Laaksonen, for his support throughout this period. He, together with my second supervisor, Assistant Professor Mustafa Alrayah Hassan Ibraheem, has been instrumental in enabling this work and has provided valuable insights along the way.

From industry partners I extend my gratitude to WE Tech Solutions Oy and Danfoss Drives for their support during the doctoral studies, as well as to Wasaline for allowing me to utilize the data collected from Aurora Botnia in this work.

Over the past thirteen years I have also met numerous colleagues and business partners who have supported me on this journey. It would be impossible to name each and every one individually, so I can only extend my sincere thanks to all of them.

Finally, I am deeply grateful to my wife Huaying and our children for their unwavering support and patience throughout this journey, even during the periods when my work required me to be far away from home for long periods of time — 谢谢，我爱你。

Vaasa, October 2025

Timo M. R. Alho

Contents

List of Tables

List of Figures

List of Codes

Nomenclature

$\boldsymbol{\Gamma}_C$ Γ_C-matrix containing the measurement error induced offsets for diag($\boldsymbol{\Gamma}$)-matrix to compute circulating currents:

$$\begin{bmatrix} \Gamma_{CS_1} & \dots & \Gamma_{CS_1S_N} \\ \vdots & \ddots & \vdots \\ \Gamma_{CS_NS_1} & \dots & \Gamma_{CS_N} \end{bmatrix}$$

$\mathbf{B}_{runS}$ Boolean array for indicating which voltage source is running:

$$\begin{bmatrix} B_{runS_1} \\ \vdots \\ B_{runS_N} \end{bmatrix}$$

$\mathbf{B}_{\mathrm{VS}}$ Boolean array for indicating which voltage source is in voltage control mode: $\begin{bmatrix} \mathrm{B}_{\mathrm{VS}_1} \\ \vdots \\ \mathrm{B}_{\mathrm{VS}_N} \end{bmatrix}$

$\mathbf{G}_{\mathrm{nodes}}$ Nodal analysis conductance matrix [S]

$\mathbf{I}_{\mathrm{C}}$ Current matrix consisting of currents of every equipment in the system, including inter-equipment circulating currents [A]

$\mathbf{I}_{\mathrm{L}}$ Load current array: $\begin{bmatrix} I_{\mathrm{L}_1} \\ \vdots \\ I_{\mathrm{L}_N} \end{bmatrix}$

$\mathbf{I}_{\mathrm{nodes}}$ Nodal analysis current array [A]

$\mathbf{I}_{\mathrm{nomS}}$ Source nominal current array: $\begin{bmatrix} I_{\mathrm{nomS}_1} \\ \vdots \\ I_{\mathrm{nomS}_N} \end{bmatrix}$

$\mathbf{I}_{\mathrm{S}}$ Source current array: $\begin{bmatrix} I_{\mathrm{S}_1} \\ \vdots \\ I_{\mathrm{S}_N} \end{bmatrix}$

$\mathbf{I}_{\Sigma\mathrm{C}}$ System's total current array, where individual values are the sums of load supplying currents and circulating currents between other voltage regulating equipment [A]

$\mathbf{V}_{\mathrm{DC}}$ Voltage matrix containing potential differences from $\boldsymbol{\Upsilon}_{\mathrm{nodes}}$ [V]

$\boldsymbol{\Upsilon}_{\mathrm{nodes}}$ Node potential array from nodal analysis [V]

Γ_x^{OV} Proportional value of how much contribution is required by equipment x's overvoltage regulator

$\Gamma_{\mathrm{C}x}^{\mathrm{OV}}$ Offset for Γ_x^{OV} to take in to account the effect of circulating currents in x's output when running against overvoltage regulator

Γ_x^{UV} Proportional value of how much contribution is required by equipment x's undervoltage regulator

$\Gamma_{\mathrm{C}x}^{\mathrm{UV}}$ Offset for Γ_x^{UV} to take in to account the effect of circulating currents in x's output when running against undervoltage regulator

Γ_x Proportional value of how much contribution is required from the voltage source x to maintain DC-link voltage, defined as $\frac{V_x^* - V_{\mathrm{DC}}}{\zeta_x V_x^*}$

$\Gamma_{\mathrm{B}x}$ Offset for Γ_x to include DC voltage reference-based energy storage state-of-charge balancing functionality

$\Gamma_{\mathrm{C}xy}$ Offset for Γ_x to take in to account the effect of circulating currents in x's output feeding from equipment x to y

$\Gamma_{\mathrm{C}x}$ Offset for Γ_x to take in to account the effect of circulating currents in x's output running the load

$\mathrm{B}_{\mathrm{run}x}$ Boolean value: 1 if equipment x is considered running, 0 if it is considered stopped

$\mathrm{B}_{\mathrm{V}x}$ Boolean value: 1 if equipment x maintains DC voltage via reference and droop, 0 if it is in current control mode

$\mathrm{diag}(\mathbf{\Gamma}_{\mathrm{S}})$ Diagonal source Γ-matrix: $\begin{bmatrix} \Gamma_{\mathrm{S}_1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \Gamma_{\mathrm{S}_N} \end{bmatrix}$

$\mathrm{diag}(\mathbf{I})$ Diagonal Current Matrix: $\begin{bmatrix} I_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & I_{\mathrm{N}} \end{bmatrix}$

K_x^* Equipment x's normalized proportional current reference for internal calculations without voltage regulator or current limiters applied

$\mathrm{K}_{\mathrm{Reg}x}^*$ Equipment x's normalized proportional current reference for internal calculations with voltage regulators applied but not current limiters

K_x (Greek capital letter Kappa) Coefficient to indicate equipment x's final proportional load reference after voltage regulators and current limiters

$\mathrm{SoC}_{\mathrm{C}}$ Center of energy for multiple connected battery systems in the DC microgrid

SoC_x State-of-charge of a energy storage system for equipment x

SoH_x State-of-health of a energy storage system for equipment x

$\overline{I}_{\mathrm{nom}x_n}$ Average nominal current rating of equipment x_1, x_2, x_3 ... x_n

$\overline{Q}_{\mathrm{max}x_n}$ Average maximum energy storage capacity of equipment x_1, x_2, x_3 ... x_n

Υ_a Absolute electric potential of node a in the reference-free nodal analysis framework. Used internally to solve system voltages; physically meaningful voltages are always computed as potential differences between nodes

ζ_x^{OV} Overvoltage regulator's drooping coefficient for equipment x

ζ_x^{UV} Undervoltage regulator's drooping coefficient for equipment x

ζ_{eq} System equivalent droop coefficient

ζ_x Voltage source drooping coefficient for equipment x indicating proportional voltage reference drop at nominal output current

$\zeta_{\text{B}x}$ Energy storage state-of-charge balancing induced DC droop coefficient offset for equipment x

G_{D} Droop conductance [S]

I_x^* Equipment x's DC current reference from external source [A]

$I_x^{\text{p.u}}$ Harmonized current of equipment x: $I_x^{\text{p.u}} = \dfrac{I_x}{\sum_{n=1}^{N} I_{x_n}}$

I_x Current associated with equipment x [A]; $I_x > 0$ indicates current flowing into the DC-link's positive terminal (i.e., current being produced), and $I_x < 0$ indicates current flowing out toward equipment (i.e., current being consumed)

$I_{\text{C}xy}$ Circulating current flowing from equipment x, via the DC-link, to equipment y

$I_{\text{C}x}$ Equipment x's load-supplying current affected by DC voltage measurement error

$I_{\text{max}x}$ Equipment x's maximum current limit [A]

$I_{\text{min}x}$ Equipment x's minimum current limit [A]

$I_{\text{nom}x}$ Equipment x's nominal current [A]

$I_{\Sigma\text{C}x}$ Equipment x's total current, including load feeding current and circulating currents between other voltage sources [A]

n Modeling index

P_x^* Equipment x's DC power reference from external source [W]

$Q_x^{\text{p.u}}$ Harmonized energy capacity of equipment x: $Q_x^{\text{p.u}} = \dfrac{Q_x}{\sum_{n=1}^{N} Q_{x_n}}$

Q_{max} Maximum energy capacity for a energy storage system [Ah]

Q_{rated} Rated energy capacity of an energy storage system [Ah]

R_{L} Load's internal resistance [Ω]

R_{MG} System's total resistance [Ω]

$R_{\mathrm{C}_{i\pm}}$ Resistance of cable i's positive ($+$) or negative ($-$) conductor

$R_{\mathrm{D}x}$ Equipment x's droop resistance [Ω]

V^*_{eq} System equivalent DC voltage reference [V]

V^*_x DC voltage reference for equipment x [V]

$V^*_{\mathrm{B}x}$ Energy storage state-of-charge balancing induced DC voltage reference offset for equipment x [V]

$V^*_{\mathrm{DB}x}$ Drooped DC voltage reference, including energy storage state-of-charge balancing related offset, for equipment x [V]

$V^*_{\mathrm{D}x}$ Drooped DC voltage reference for equipment x [V]

$V_x^{\mathrm{OV}*\prime}$ Conditionally droop-compensated overvoltage regulator activation limit for equipment x [V]

$V_x^{\mathrm{OV}*}$ Desired overvoltage regulator activation limit for equipment x [V]

$V_x^{\mathrm{UV}*\prime}$ Conditionally droop-compensated undervoltage regulator activation limit for equipment x [V]

$V_x^{\mathrm{UV}*}$ Desired undervoltage regulator activation limit for equipment x [V]

V_{DC} Actual DC-link voltage [V]

V_x DC-link voltage measured by equipment x [V]

$V_{\mathrm{err}x}$ Equipment x's DC voltage measurement error [V]

RQx Research Question number x

Declaration of AI Usage

During the finalization phase of the dissertation the author used ChatGPT's GPT-4o model to prepare the abstracts and GPT-5 to generate Tables 1 and 14 based on the already existing dissertation content prepared by the author himself. The author reviewed the contents and modified them when deemed necessary and takes full responsibility for them.

The author used ChatGPT's GPT-4o model to give comments on possible contextual gaps in the dissertation during the work finalization phase. The

received comments were taken into account where applicable and the author modified the dissertation based on them.

The author used ChatGPT's GPT-5 model for language checking of his own produced texts and implemented the revised texts only after verifying the content and modifying it where applicable.

1 INTRODUCTION

Shipboard power systems are, in many respects, unique. They range from comparatively small systems of a few hundred kilowatts to large-scale systems reaching dozens of megawatts. Their power generation methods are continuously evolving, incorporating a wide range of fuel types for combustion engines—each with distinct capabilities and characteristics—and new technologies such as hydrogen fuel cells. Battery systems and other energy storage technologies, such as supercapacitors, are also being introduced to support power system operations through peak shaving, load scheduling, and other strategies that promote energy-efficient operation. These systems may also intermittently connect to shore-based power supplies. Propulsion architectures are gradually shifting from conventional mechanical main engines toward fully electric systems.

At the same time, shipboard power systems must rank among the most resilient in the world, capable of withstanding blackout-inducing fault conditions and maintaining continuous operation of critical functions such as propulsion, navigation, and essential services, to prevent potentially catastrophic accidents.

As shipboard power systems become more sophisticated and functionally capable, the enabling technologies must also evolve. This development is driving a fundamental shift in power distribution paradigms. Alternating current (AC) distribution has served as the de facto standard in shipboard power systems for over a century. However, due to increasing demands for fuel efficiency and reduced carbon emissions, as noted in Kim, Park, Roh, and Chun (2018), the adoption of power electronics-based inverters—enabling variable-speed combustion engines and integration of energy storage—is paving the way for direct current (DC) distribution systems, also known as DC microgrids.

1.1 Background and Motivation

The objective of this dissertation is to develop an advanced and adaptive control strategy for DC microgrids, which leverages the DC voltage level and

the capabilities of modern load inverters to actively respond to voltage deviations. This represents a fundamental shift from the conventional assumption that load inverters do not, or cannot, contribute to DC voltage regulation.

By enabling load inverters to autonomously react to grid events, a portion of the microgrid control functionality is distributed to the inverters themselves. As some of the fastest-reacting devices in the system, load inverters can enhance the robustness of microgrid behavior under a wide range of conditions. The principles presented in this dissertation allow for the design of shipboard power systems that are autonomous, highly resilient, and tolerant to fault conditions—capable of maintaining stability during exceptional events without requiring exhaustive pre-programming of all possible contingency scenarios.

1.1.1 DC Microgrid Control and Drooping

Microgrid control refers to the mechanisms and operational principles used to maintain the balance between power production and consumption. In shipboard power systems, this is often part of or closely linked with the broader power management system. In droop controlled DC microgrids, this balance is reflected by the DC-link voltage: if the voltage is rising, more energy is being produced than consumed, and vice versa—if the voltage is falling, consumption exceeds production. The rate of voltage change corresponds to the severity of the imbalance. Without such balance, the voltage will deviate outside acceptable bounds, causing protective tripping of connected equipment and loss of stable operation.

Different microgrid control strategies—centralized, decentralized, and distributed—make use of this voltage feedback in various ways. Each approach has merits depending on system requirements and operational context. A detailed comparison of these strategies is provided in Chapter 2, with Section 2.4 specifically discussing their structure and implications.

The one thing that the microgrid controller has very limited control over is the loads. While it is possible for the shipboard controller to forcibly disconnect non-critical loads, such action is typically reserved for exceptional

situations in which the active loads are in danger of exceeding the power production capacity. During normal steady-state operation, the loads operate independently, and the microgrid must adapt accordingly to maintain voltage stability—i.e., there must always be at least one voltage source in the microgrid that continuously measures and regulates the DC voltage.

A fundamental aspect of microgrid control in larger systems is enabling the parallel operation of multiple voltage sources. In a purely theoretical scenario, two voltage sources maintaining and measuring exactly the same voltage would operate in harmony. The real world, however, is not ideal: component tolerances, measurement errors, and control inaccuracies always exist. The result is that voltage sources may observe slightly different voltages—for example, one sees 749 V and another 751 V—even though both aim to maintain a nominal 750 V. This discrepancy leads to conflict: one source injects more current to raise the voltage, while the other absorbs current to reduce it, effectively fighting each other. Such conflicts persist until one or both sources exceed their current limits and trip offline.

Several methods exist to enable parallel operation of voltage sources, such as synchronizing inverter modulation. However, many such solutions rely on communication, which introduces potential failure points. To address this, the concept of DC voltage drooping was introduced—a method that allows independent sources to coordinate implicitly based on the measured voltage level, as described in Ito, Zhongqing, and Akagi (2004), and defined in Equation (1.1):

$$V_{\mathrm{DS}}^{*} = V_{\mathrm{S}}^{*} - R_{\mathrm{DS}} I_{\mathrm{S}} \tag{1.1}$$

The basic premise in this method is to create virtual resistance, R_{DS}, at the output of each voltage source. This then effectively creates a non-ideal voltage source (sometimes referred to as a practical voltage source), with the virtual resistance acting as the internal resistance, see Figure 1. The most prominent feature of a non-ideal voltage source is that its actual output voltage, V_{DS}^{*}, varies as a function of its output current, I_{S}. In other words, power electronic equipment, like inverters operating as voltage sources in droop control, emulate, through software-based control, the behavior of internal resistance on output voltage. The higher the load demand towards the DC microgrid, the lower the output voltage they aim to produce—and vice versa. This method enables multiple voltage sources to be added to the same micro-

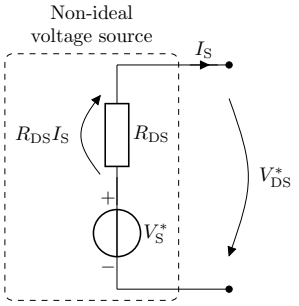


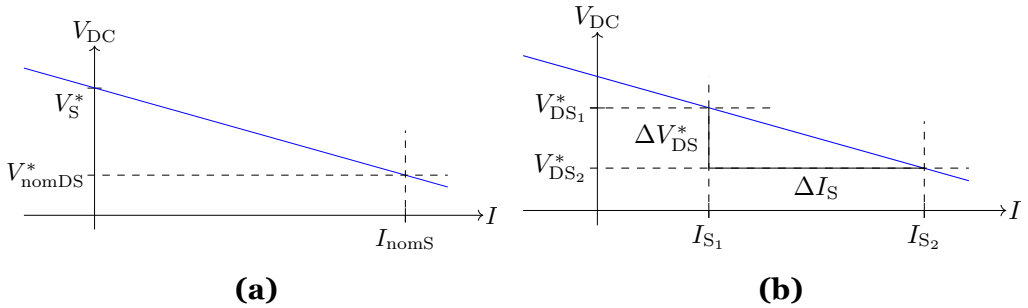
Figure 1. Non-ideal voltage source circuit, equivalent for droop principle.

grid and operated in parallel without active communication between them. The only "communication" is the shared DC voltage level of the microgrid.

Instead of a resistance value, it is often more intuitive to use a unitless coefficient that indicates the relative output voltage drop at the nominal current I_{nomS}. This makes the drooping proportional to the source's nominal current, see Figure 2a, which is conceptually clearer than using a resistance value that defines the slope of the droop curve, see Figure 2b. This coefficient for a voltage source is denoted ζ_S in this work,

$$V^*_{DS} = V^*_S \left(1 - \frac{I_S}{I_{\text{nomS}}}\zeta_S\right). \tag{1.2}$$

A key benefit of the proportional nature of ζ_S is that, when equal drooping behavior is desired across all voltage sources—even when the equipment have different nominal ratings—the same value of ζ_S can be applied. This contrasts with using R_{DS}, which must be calculated individually for each voltage source and inherently differs depending on the nominal current, as noted by Bidram, Reno, Abadi, Aparicio, and Bauer (2024).



(a) **(b)**

Figure 2. Visualization to derive (a) ζ_S and (b) droop resistance R_{DS}.

The virtual resistance represents the negative rate of change in output voltage over a range of output currents, as illustrated in Figure 2b:

$$R_{\mathrm{DS}} = \frac{-\Delta V^*_{\mathrm{DS}}}{\Delta I_{\mathrm{S}}} = \frac{-\left(V^*_{\mathrm{DS2}} - V^*_{\mathrm{DS1}}\right)}{I_{\mathrm{S2}} - I_{\mathrm{S1}}} = \frac{V^*_{\mathrm{DS1}} - V^*_{\mathrm{DS2}}}{I_{\mathrm{S2}} - I_{\mathrm{S1}}}. \tag{1.3}$$

The ζ_{S} coefficient, by contrast, directly expresses how much the DC voltage reference is permitted to drop from its nominal value when operating at the nominal current. It effectively encodes this as a fraction or percentage, making it more intuitive. This is shown in Figure 2a, and defined as:

$$\begin{aligned} &\text{if } V_{\mathrm{DC}}\left(I_{\mathrm{nomS}}\right) = V^*_{\mathrm{nomDS}}, \\ &\qquad \text{then } \zeta_{\mathrm{S}} = 1.0 - \frac{V^*_{\mathrm{nomDS}}}{V^*_{\mathrm{S}}}, \end{aligned} \tag{1.4}$$

where V^*_{nomDS} is the desired drooped voltage level at 100% load. In practical implementations, ζ_{S} is typically set to a small value, such as 2% (i.e., $\zeta_{\mathrm{S}} = 0.02$), as modern measurement systems offer sufficient voltage accuracy. This was the value used in the power supply prioritization case study by Alho and Laaksonen (2020).

1.1.1.1 DC Voltage as Balance Indication and Its Modeling

Fundamentally, the fact that DC voltage reflects the balance between production and consumption in droop-controlled DC microgrids follows directly from Kirchhoff's current law and Ohm's law. Any mismatch between generation and load manifests as a net current ΔI at the DC-link. With grid and droop resistances considered constant, the corresponding voltage deviation is

$$\Delta V = R \cdot \Delta I. \tag{1.5}$$

Hence, a larger power imbalance causes a proportionally larger deviation in the common DC voltage. In this way, the DC-link voltage serves as a global indicator of instantaneous power balance—analogous to frequency in AC microgrids. This principle forms the basis of the DC voltage modeling developed in Sections 3.1.3.1 and 3.1.10.

This principle holds regardless of whether capacitors or other energy storage

elements are explicitly modeled. Including them introduces time-dependent phenomena, such as natural delays and smoothing in the voltage response and transients, but they do not affect the steady-state values. These dynamic phenomena become relevant primarily at sub-second time scales. Omitting such elements results in instantaneous voltage deviations—a simplification that may appear unrealistic for transient studies, but is appropriate for the quasi-static modeling used in this work. As described in Sections 3.1.3.1 and 3.1.4.1, the studied control strategy operates on time scales of seconds to minutes, where the dominant observable effects of imbalance between power production and consumption are adequately captured without detailed modeling of transient dynamics.

1.1.2 Droop Compensation

Some researchers have proposed droop compensation as a foundational principle for DC microgrid control strategies. In this approach, while drooping is still used to enable parallel operation of multiple voltage sources, the DC voltage references of the sources are continuously adjusted by an upper-level controller to counteract the voltage deviation induced by droop. This results in a system where the DC voltage is maintained near a target level, but requires additional control mechanisms to compensate for the droop-induced deviation. Guerrero, Vasquez, Matas, de Vicuna, and Castilla (2011) incorporated this approach into a general microgrid control strategy applicable to both AC and DC systems.

In droop-controlled DC microgrids, voltage deviations can introduce several challenges. One of the most critical is the stability issue associated with constant power loads (CPLs): when a CPL draws a fixed power, a voltage drop causes its current demand to increase, which in turn further lowers the voltage, potentially leading to a voltage collapse. Additional concerns include reduced power transfer capability—as lower voltages require higher currents to deliver the same power—and decreased efficiency due to increased resistive losses.

These issues are particularly significant in large-scale DC distribution systems with substantial cable resistances. In contrast, the systems examined

in this work—mechanically compact yet power-dense shipboard power systems—are assumed to have negligible distribution losses. As demonstrated in the real-world case study Alho and Laaksonen (2020) analyzed in Section 5.4.1, modern inverters can operate reliably with droop coefficients in the range of 2–3 %, making the resulting voltage deviations acceptable for most connected loads in such contexts. Nevertheless, cable and busbar resistances are included in the nodal analysis framework in Section 3.1.10 to capture their effects when applicable.

A fundamental drawback of droop compensation is that it removes valuable system state information. In shipboard power systems the voltage sources are traditionally parameterized identically (i.e., same voltage references and droop coefficients), making the deviation in DC voltage directly proportional to the loading of the voltage sources. This deviation can be leveraged by load inverters to improve system robustness, fault tolerance, and autonomy. By observing the voltage level, inverters can infer the system load state and act accordingly—without centralized coordination.

Maintaining a constant DC voltage requires the capability to instantaneously and accurately match power generation to consumption at all times. In exceptional situations where this is not possible, the voltage will collapse. Systems that rely solely on centralized control are vulnerable in such cases unless all contingencies are pre-programmed and communication systems are both immediate and reliable. Even with accurate power availability forecasts and preassigned load limits, the complexity of determining which loads to shed and by how much increases rapidly.

Allowing the DC voltage to deviate within the few percentage points introduced by droop control enables inverters to deduce the system loading state and react autonomously. This enables system-level functionalities that were not previously achievable. Furthermore, voltage deviations caused by exceptional conditions can be similarly interpreted by the inverters, enabling them to respond without requiring explicit coordination.

Nonetheless, droop compensation remains a useful tool in specific cases. For example, auxiliary systems such as DC/DC converters connected to supercapacitors can use variable voltage-based control principles to function independently during fast transients. In normal operation, these devices should remain unloaded, with their internal voltage references tracking the

actual DC voltage.

Droop compensation is also used in Section 3.1.7 as a mechanism to force under- and overvoltage regulators in load inverters to activate when desired. Without such compensation, their activation thresholds would vary with inverter loading, leading to unintended behavior.

In conclusion, droop compensation can be beneficial when applied deliberately and in specific scenarios. However, if used indiscriminately, it risks discarding useful information about system load state, reducing the overall controllability and resilience of the microgrid.

1.1.3 Variable Voltage-based Control

Utilizing DC microgrid voltage variations for control purposes is not a new idea in itself; for instance, Bryan, Duke, and Round (2004); Chen, Xu, and Yao (2013); Papadimitriou, Kleftakis, Rigas, and Hatziargyriou (2014); Zhang, Wu, Xing, Sun, and Guerrero (2011) have applied a method called DC Bus Signaling. In such approaches, the DC voltage level acts as a reference signal for a centralized controller, which changes the system's operational mode accordingly.

However, relying solely on the controller to act based on the DC voltage level has the drawback that the fast-acting control is not distributed to the equipment most capable of reacting rapidly to system changes—namely, the inverters themselves. Instead, the system includes a centralized controller that supervises the DC voltage and makes decisions affecting inverter operation. While a higher-level supervisory controller is typical in power systems, using it as the sole actor based on DC voltage introduces two critical problems in the shipboard context:

1. **Fast reactions:** As noted earlier, the DC voltage level reflects system loading conditions, including exceptional events that may require reactions on the scale of milliseconds to tens of milliseconds—to prevent blackouts. If the centralized controller alone supervises the voltage and makes decisions, time-scales are too short to survive the situation.

2. **Manual control over operation modes:** In shipboard systems, the crew must maintain full authority. All operational mode changes must be crew-initiated or at least the automatic changes are designed with the ship owner. For safety and clarity, the system should not autonomously perform major operational changes. It must remain resilient and ride through exceptional conditions (e.g., sudden power shortfall) until manual corrective action is taken, then return to steady-state operation.

In the variable voltage-based control method, fast response is distributed to the lowest possible level—the inverters themselves. More details on this control strategy's capabilities are presented in Chapter 5. A minimal example of similar kind of control is given by Yu, Huang, Burgos, Li, and Du (2013), where inverters operate based on local DC voltage measurements without centralized coordination.

Inverters continuously monitor the DC voltage level and adjust their contributions based on their internal control parameters: control method (current or voltage source), droop coefficient, voltage reference, under- and overvoltage regulation thresholds and associated droop characteristics, and other similar settings—see Figure 3.

These parameters define operational boundaries—a control sandbox. Within this sandbox, each inverter's behavior emerges naturally from system con-

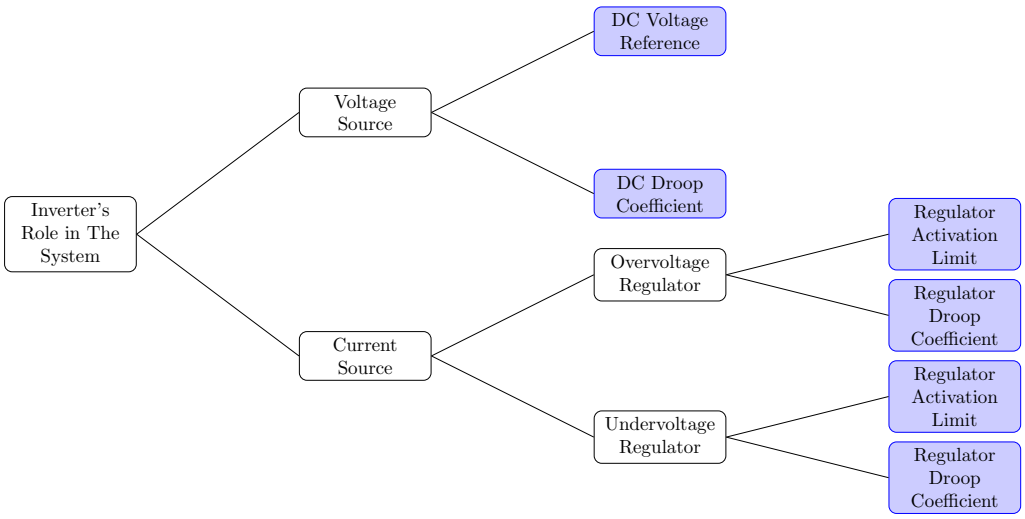


Figure 3. Parameters used in variable voltage-based control strategy.

ditions, rather than centralized commands for droop compensation or system-level mode changes. For instance, both a DC/DC converter (connected to a battery) and a main propulsion inverter observe the same voltage level and respond accordingly: the converter begins discharging when the voltage drops; if voltage continues to fall, the propulsion inverter supports the system via its undervoltage regulator by reducing its own load. Each unit acts autonomously within its configured limits.

1.1.4 Research Motivation

While shipboard power distribution systems are gradually transitioning toward more energy-efficient and robust DC architectures, the underlying control strategies found in the literature still largely reflect the principles traditionally used in AC systems. This includes, for example, requirements and functionalities aiming for absolute DC voltage stability—a concept that appears to be a remnant of the strict frequency stability requirements inherent to AC systems. However, enforcing constant voltage in DC systems can prevent—or at least limit—the full exploitation of the benefits offered by DC distribution and DC-based equipment.

As noted earlier, although the use of DC voltage as a communication mechanism has been proposed and implemented in the form of DC Bus Signaling, this methodology has not been developed to its full potential. In most implementations, the voltage level merely informs the upper-level controller, which then adapts the system's operation accordingly. This approach offers only marginal benefits and retains many of the drawbacks of centralized control: communication delays, limited fault tolerance, and vulnerability to single points of failure.

When the same concept is adopted as the foundation for a proper DC microgrid control strategy—particularly in systems equipped with modern inverters and energy storage solutions—a fundamentally different system behavior emerges. Such a system can achieve resilience and fault tolerance against blackouts, ensuring high reliability and availability of power supply without reliance on centralized upper-level control.

1.2 Objectives of This Dissertation

The objective of this dissertation is to develop a novel DC microgrid control strategy and map out its performance. The control strategy is based on utilizing the DC-link voltage as a communication method to disseminate information on the microgrid's voltage sources' loading to the load inverters. These are then capable of autonomously supporting the sources according to the DC-link voltage.

This voltage regulator functionality is an existing feature in modern inverters, but it has been so far mainly used as a protection function, instead of creating a more comprehensive system level functionality around it. This dissertation includes the development of an advanced voltage regulator functionality for the load inverters, providing capability to support the DC voltage in parallel with the voltage sources when DC voltage deviates to the regulator activation limit.

To accurately assess the performance of the control strategy, it is essential to model the microgrid and the behavior of its components with precision. Additionally, the control strategy for ships must be designed to be highly resilient and fault-tolerant, ensuring it can handle and ride through various exceptional situations and operational scenarios. The modeling approach must also incorporate the ability to simulate the voltage regulators of the load inverters and their transitions between different operating modes.

The core principle of the variable voltage-based control strategy is that loads can actively adjust their operation based on the DC voltage level. While they do not necessarily become true voltage sources, they can function as voltage-supporting units by transitioning from current sinks to providing support through their under- and overvoltage regulators. This is a novel concept in marine DC microgrid power management, fundamentally changing how these systems can be designed and operated.

- **Research Question 1**
  The majority of academic research on DC microgrid control strategies has focused on applying the same principles as in AC microgrids. Since AC systems require frequency stability, DC systems have similarly been controlled by keeping the voltage as close to the defined ref-

erence as possible. The first research question (RQ1) addresses how to model a DC microgrid control strategy that embraces both the sources' DC drooping induced voltage deviation and the full DC voltage regulating functionalities of modern load inverters.

Section 3.1 presents the mathematical foundation for the quasi-static modeling paradigm developed in this dissertation in order to model ideal and non-ideal DC microgrids and the inverters within it with their under- and overvoltage regulators.

- **Research Question 2**
  The second research question (RQ2) considers how the load inverter's voltage regulators could be optimized. To have built-in drooping with conditional compensation to enable parallel operation with other DC voltage maintaining equipment, but without the drooping interfering with the normal operation of the inverter.

  Section 3.1.7 covers the development of the regulators, where the operation is modeled in a system with two loads supporting a voltage source of limited maximum output in an ideal microgrid. Section 3.1.10 covers the regulator operation in non-ideal microgrids.

- **Research Question 3**
  While the microgrid voltage sources' drooping induced DC-link voltage deviation is expected and allowed by the studied control strategy, it is, however, expected to be the direct consequence of increasing load in the system, and not unoptimized equipment behavior. As modern shipboard power systems are expected to have a growing amount of energy storage, and it is expected to be utilized to its full potential by state-of-charge balancing algorithms, the inverters with these algorithms must not cause voltage deviations while working as voltage sources in the microgrid. The third research question (RQ3) defines how to develop an energy storage state-of-charge balancing algorithm compatible with the studied DC microgrid control strategy.

  Section 4.3 covers the development of state-of-charge balancing algorithm based on DC voltage reference offset, and in Section 4.4 a droop coefficient offset-based is presented. The algorithms are built following the same principles that were covered by the first research question (RQ1), to be compatible with the modeling paradigm.

- **Research Question 4**
  In order to assess the performance and usability of any DC microgrid control strategy for the marine segment, the strategy and the equipment behavior in the system must be developed and modeled to be as resilient as possible during both emergency and normal operation scenarios that shipboard power systems face in real-life. The fourth research question (RQ4) focuses on studying the control strategy's performance in selected and modeled shipboard power system's real-life scenarios.

  Chapter 5 includes system modeling in a number of different scenarios that ships in real-life can and will face. They range from mundane situations, like starting a voltage source automatically, to more critical ones, like ship blackout and voltage source overload preventions.

- **Research Question 5**
  The fifth research question (RQ5) studies how well the modeling paradigm proposed in this dissertation models the behaviors of a real-life system. For the modeling paradigm to be useful at all, its results must have predictive value and model the real-life situations accurately.

  Data from a real-life power supply prioritization use case is presented in Section 5.4.3. The situation is also modeled utilizing the developed modeling paradigm and the data from the real ship and the model are compared.

- **Research Question 6**
  Finally, the last research question (RQ6) studies how well the operational principles of the proposed DC microgrid control strategy work in real-life situations. For the control strategy to have any usefulness, its principles must be usable in real-life, i.e., they need to work in practice as modeled.

  This dissertation presents two real-life ships that utilize some of the variable voltage-based control strategy's principles: Section 5.4.1 uses variable voltage-based power supply prioritization, first presented by Alho and Laaksonen (2020), and Section 5.6.2 covers a real-life variable voltage-based blackout prevention use case. Both cases are presented with real-life data on the operation of the system.

1.2.1 Research Methodology

The effectiveness of the variable voltage-based DC microgrid control strategy is analyzed through both mathematical modeling and real-life implementation cases. The modeling is based on the author's self-developed quasi-static modeling paradigm for droop-controlled microgrids that include voltage sources and load inverters with under- and overvoltage regulation functionalities. The simulations are conducted in MATLAB, and the associated scripts are made available in this dissertation to support reproducibility and further research.

The development of a dedicated modeling paradigm is essential for accurately representing the behavior of inverter-based load regulators. These under- and overvoltage regulators are central to the control strategy's operational principles and require specific capabilities that are either not supported—or are prohibitively complex to implement—in existing commercial simulation tools known to the author.

The real-life validation is conducted through use cases involving actual ships, whose power systems the author has directly contributed to by developing, programming, and, in some cases, commissioning. These systems incorporate the control principles detailed in this dissertation.

1.3 Main Contributions

This dissertation extensively covers the author's self-developed DC microgrid quasi-static modeling paradigm. It includes capabilities to model ideal and non-ideal DC-links, as well as voltage sources and loads with under- and overvoltage regulators. As energy storage is becoming a vital part of shipboard power systems, the modeling also includes state-of-charge balancing by means of DC voltage reference and DC droop coefficient. The developed state-of-charge balancing algorithms are analyzed for their compatibility with the variable voltage-based DC microgrid control strategy.

The custom modeling paradigm was developed to avoid the limitations and complexities of existing simulation environments, such as Simulink or PSCAD,

particularly regarding the changing operational behavior of load inverters with under- and overvoltage regulators. These regulators enable the inverters to switch between voltage and current source operation. To the author's knowledge, such behavior is not natively supported in existing software, and implementing it manually would increase model complexity and error risk. The custom modeling paradigm allows regulator algorithms to be implemented in a way that is fully compatible with the control strategy. Additionally, voltage sources using droop control can be modeled proportionally without manually constructing equivalent circuits from ideal sources and individual resistors, which would require separate calculations and increase the likelihood of errors.

The contributions also include the author's own state-of-charge balancing algorithms, capable of handling multiple energy storage systems with arbitrary capacities and DC/DC converter current ratings, in a manner that is fully compatible with the control strategy. Since the control strategy relies on the DC voltage level to indicate system loading status, it is critical that the voltage variation reflects real loading changes—not fluctuations introduced by unoptimized control of subsystems. The developed balancing algorithm ensures such undisturbed operation.

Because the dissertation focuses on DC microgrid control strategy—which, in shipboard systems, is essentially equivalent to power management—the work does not include transfer function modeling of inverters. The inverters are instead modeled as ideal devices operating precisely within their defined control limits. While transfer function modeling could have been achieved in Simulink, it would have introduced unnecessary complexity for power flow studies, which are the primary concern of this research.

Using the developed modeling paradigm, several DC microgrid scenarios are modeled based on real-world requirements for shipboard systems. These are complemented by real-life case studies from ships whose power systems the author has helped develop, implement, or commission. Together, they demonstrate the versatility of the variable voltage-based DC microgrid control strategy.

The main operational scenarios contributing to a resilient shipboard power management system, as covered in this dissertation, include:

- **Ensured power availability**
  Modeled scenarios in which loads and renewable sources automatically limit their output to prevent voltage source overload, without external commands.

- **Automatic stand-by voltage source startup**
  Modeled scenario demonstrating automatic startup of standby voltage/power sources based solely on the DC voltage level.

- **Automatic limitation of excess power generation**
  Modeling of two mechanisms for limiting the output of current sources without external commands, maintaining system stability.

- **Power supply prioritization**
  Real-life implementation and corresponding model demonstrating prioritization among voltage sources.

- **Stopping non-critical loads**
  Modeling of non-critical load shedding based solely on the DC voltage level.

- **Blackout prevention**
  Real-life demonstration aboard Wasaline's *Aurora Botnia*, operating between Vaasa, Finland, and Umeå, Sweden, showing blackout prevention based on the variable voltage-based control strategy.

1.4 Outline of the Dissertation

In addition to the introduction, and the list of the research questions in Section 1.2, the work is divided into six chapters, covering the following aspects. Chapter 2 contains a brief introduction to different classifications and characteristics of microgrids and their control strategies.

Chapter 3 covers in detail the mathematical foundation for the author's own DC microgrid modeling paradigm, which is used in this dissertation to demonstrate the variable voltage-based control strategy's performance in different use cases. This chapter includes a detailed description of the fundamentals

of the modeling paradigms and load inverter voltage regulators, answering RQ1 and RQ2.

As energy storage is steadily increasing its presence in the DC microgrids, especially in shipboard power systems, it was deemed vital that its usage be covered more deeply. To this end, Chapter 4 covers the author's own developed state-of-charge balancing algorithms, capable of balancing multiple energy storages of arbitrary capacities and DC/DC converter nominal current ratings, and compatible with the covered DC microgrid control strategy, answering RQ3.

Chapter 5 dives more deeply into the performance of the variable voltage-based control strategy, and demonstrates its capabilities in different scenarios ships encounter in real life. Real-life use cases are also presented with data recorded from ships sailing in Australia and Finland. Together with the different scenarios and real-life data comparison, this chapter answers RQ5 and RQ6.

Chapters 6 and 7 are the discussion and conclusions chapters, respectively, covering the recap of the dissertation and drawing conclusions on the data from the mathematical models and real-life data of different scenarios. The recap also lists the various shortcomings and development areas of the modeling paradigm presented in this dissertation.

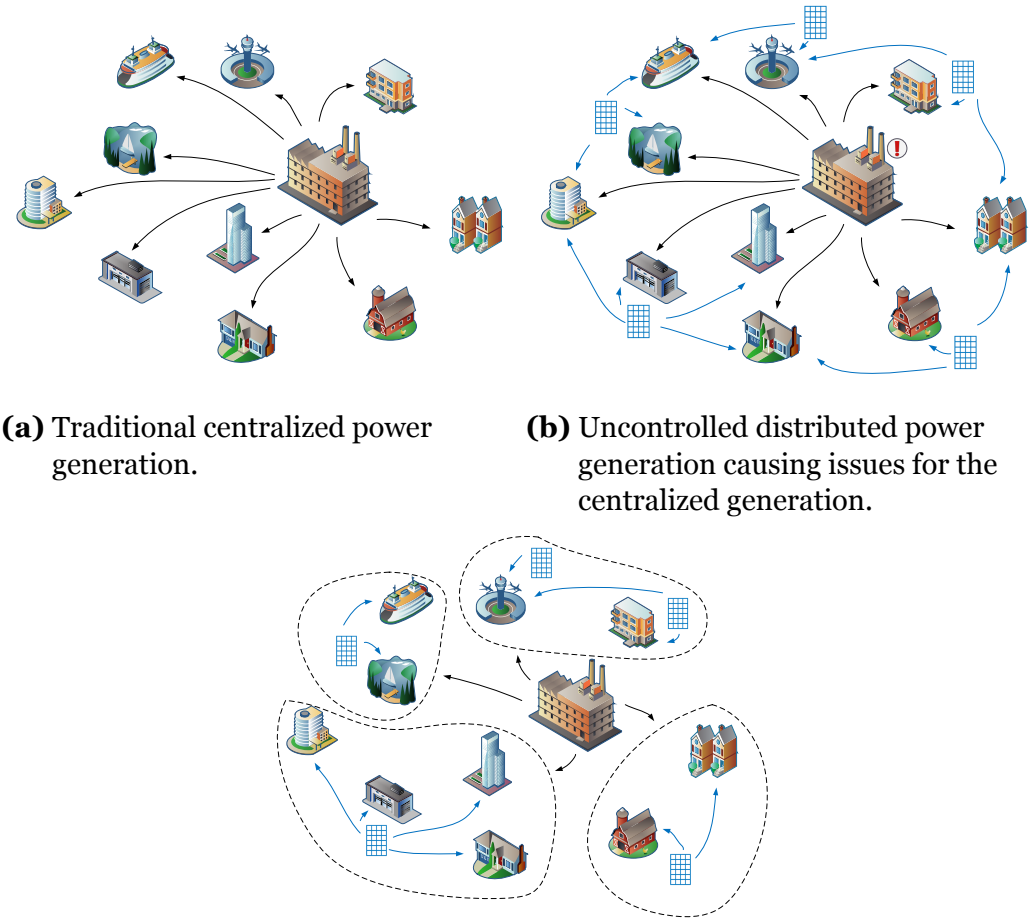
2 CLASSIFICATION AND CHARACTERISTICS OF DC MICROGRIDS AND THEIR CONTROL STRATEGIES

Isolated electrical grids have long existed in various forms, such as on ships, offshore platforms, islands, remote rural communities, and aircraft. However, the term “microgrid” only emerged in recent decades, driven by the proliferation of distributed energy resources (DERs) and the operational challenges they introduce. The microgrid concept provides a framework for improving power quality and system stability in grids where generation is no longer centrally controlled. It does so by partitioning larger electrical systems into smaller, autonomous units with local control capabilities Lasseter and Paigi (2004); see Figure 4.

Conventional terrestrial power systems were historically designed around large, centralized power plants, as illustrated in Figure 4a. Power flowed unidirectionally from these centralized units through transmission lines to end-user loads. Voltage and frequency control were maintained by a relatively small number of large generators.

With the increasing adoption of renewable energy technologies—such as rooftop photovoltaic systems and small-scale wind turbines—the grid began to incorporate a large number of geographically distributed, small-scale generators (Figure 4b). These units, while beneficial for sustainability, introduced challenges for traditional grid control schemes. In particular, the centralized generators were ill-equipped to respond to the rapid and intermittent fluctuations in net load caused by variable renewable output. Moreover, bidirectional power flows began to appear, as areas that historically acted solely as loads became net producers. This introduced complications for protection coordination, voltage regulation, and system stability.

To address these challenges, Lasseter and Paigi (2004) introduced the concept of the microgrid, wherein the larger electrical network is divided into smaller, semi-autonomous microgrids (Figure 4c). These microgrids are capable of operating both in parallel with the main grid and in islanded mode during disturbances. This capability improves system resilience by allowing localized control and operation even when the main grid becomes unstable. The concept also facilitates better utilization of distributed generation, as



(a) Traditional centralized power generation.

(b) Uncontrolled distributed power generation causing issues for the centralized generation.

(c) Utilization of the microgrid concept's principles stabilizing the power generation.

Figure 4. Simplified depiction on the birth of the microgrid concept.

local control can allow higher penetration and greater flexibility in power management.

2.1 AC vs. DC Microgrids

As AC remains the global standard for electrical distribution, it is no surprise that most microgrids are also AC-based. However, the increasing prevalence of power electronics-based converters—which are inherently DC devices—has accelerated interest in DC microgrids, particularly in sectors like maritime power systems Alho, Shahparasti, and Laaksonen (2024).

Modern power systems increasingly rely on components that are naturally DC in character. Examples include battery and supercapacitor energy storage systems, fuel cells, and photovoltaic arrays—all of which require interfacing through power converters or inverters. Additionally, efficient control of electric motors typically involves frequency converters, which also introduce DC stages into the system. As the number of such components grows, it raises a fundamental question: why maintain an intermediate AC grid at all?

DC microgrids offer several technical advantages over their AC counterparts: reduced conversion stages between AC and DC, absence of harmonic distortions and reactive currents, higher power density, and elimination of frequency control requirements (Manandhar, Ukil, and Kiat Jonathan (2015); Justo, Mwasilu, Lee, and Jung (2013); Clerici, Paris, and Danfors (1991)). These advantages are especially attractive in marine applications, where constraints on space, efficiency, and fault tolerance are critical. As a result, DC microgrids are gaining traction even in traditionally conservative domains such as the marine sector Shekhar, Ramírez-Elizondo, and Bauer (2017); F. Wang et al. (2015); Zohrabi, Shi, and Abdelwahed (2019).

AC systems offer several advantages, particularly in terms of voltage transformation and the minimal need for power electronics. Voltage levels in AC grids can be changed efficiently using passive transformers, which, although bulky and requiring cooling, are extremely reliable due to their passive nature. Historically, AC systems have operated entirely without power electronics, enabling straightforward generation, distribution, and usage.

However, modern power systems increasingly rely on power electronic converters, even in AC networks. These devices introduce harmonic distortions into the grid, necessitating additional filtering and mitigation strategies. As a result, there is growing interest in DC microgrids, which naturally avoid issues related to harmonics and can simplify system design when paired with inherently DC-based components such as batteries, photovoltaic systems, and variable-speed drives.

2.2 Marine vs. On-shore Microgrids

The marine sector is often conservative in adopting major new technological solutions—and for good reason. Ships serve as both living environments for crew and platforms for transporting people and cargo, frequently operating for extended periods in remote locations where external assistance is unavailable. They also navigate confined spaces such as canals and ports, where system malfunctions can lead to significant operational and safety hazards, including grounding or collisions.

Several key differences exist between marine and on-shore microgrids, particularly in terms of power production, consumption types, and relative sizes. On-shore microgrids typically feature a high penetration of renewable energy sources such as solar and wind, with generator sets primarily serving as backup. In contrast, solar and wind-based renewable energy has yet to achieve widespread adoption in the marine sector. Marine power systems are typically centered around large propulsion-related loads—such as main propulsors and thrusters—which often dominate the ship's total generation capacity. On-shore microgrids generally lack such high power intensity loads in relation to their production capacity.

Consequently, in isolated operation, marine and on-shore microgrids face distinct challenges. All-electric ships operate with large loads relative to their generation capability, creating challenges in balancing power under heavy load conditions. Conversely, isolated on-shore microgrids often consist of multiple, smaller, and potentially intermittent power sources, posing challenges in managing generation-side variability and ensuring fuel efficiency.

In off-grid on-shore systems intended for residential or remote use, control priorities often include optimizing generator set operation to minimize fuel consumption, possibly through solar and wind-based renewable integration and energy storage to shift excess daytime production to nighttime use. On the other hand, shipboard power systems prioritize blackout prevention above all else. Any power outage must be quickly recovered to maintain navigational and operational safety. While fuel efficiency and solar and wind-based renewable energy use are desirable, they remain secondary concerns at sea. If such technologies compromise reliability, operators are quick

to disable them.

2.2.1 Energy Storage Usage

There are notable similarities in the use of energy storage systems between on-shore and shipboard power systems. In on-shore systems, batteries are often used as uninterrupted power sources to ride through grid faults, serve as a blackout prevention mechanism, or allow for a controlled shutdown in case of power disturbances. Their primary role, however, is typically to store excess energy from intermittent generation sources for later use, enabling load scheduling and peak-shaving—for example, in electric vehicle fast-charging stations.

In shipboard power systems, energy storage usage is characterized by a different operational profile but also depend heavily on the ship design. During normal sailing—typically a stable operating condition—batteries can remain idle, as propulsion loads are steady and overall power dynamics are relatively constant and no peak-shaving is required. In such cases, batteries may be reserved for blackout prevention functions or used for system service tasks, such as periodic state-of-health diagnostics. Their most active use typically occurs when the vessel is at shore. For example, some ships operate with batteries only near port to reduce emissions and general cargo ships equipped with cranes or chemical tankers operating cargo pumps may utilize batteries for peak-shaving and load scheduling, thereby reducing the number of generator sets required and improving fuel efficiency.

While the fundamental roles of batteries are similar in both environments, their load profiles differ significantly. On-shore systems tend to use batteries continuously, though typically with moderate power levels. In contrast, shipboard systems may involve infrequent but highly intense battery usage, especially during shore operations. These differences have implications for energy storage system dimensioning—not only in terms of power and energy ratings, but also in selecting a voltage level high enough to limit current magnitudes in high-power applications.

2.3 Grid Connected vs. Islanded

Shipboard microgrids can operate both as grid/shore-connected systems and as islanded (stand-alone) grids when the ship is at sea. Traditionally, shore connection has been used to reduce the need for main engine operation while at port, thereby lowering emissions. In most vessels—excluding for example large cruise ships—the hotel loads supplied from shore are relatively small, typically on the order of a few hundred kilowatts, whereas total onboard power demand, including propulsion, can reach several megawatts and above.

Today, as energy storage systems are increasingly integrated into ships, the purpose of shore connections is expanding beyond hotel load supply to include battery charging. This shift poses significant challenges for port infrastructure. A ship that previously required a constant 200 kW of shore power may now demand an additional 2,000 kW for a few hours to charge its batteries before departure.

In on-shore microgrids, the role of the grid connection varies depending on the system's design and intended use. In some cases, the national grid is the primary power source, while in others it serves only as a backup to local generation when demand exceeds supply.

Regardless of the application, grid or shore connections require careful planning—both for intended and unintended disconnection scenarios. While generator sets can be synchronized to an existing microgrid before connection, the opposite is not true: the national grid cannot be synchronized to the microgrid. Instead, the microgrid must be synchronized to the grid before connection, and this is typically handled by the microgrid's power management system.

2.4 Microgrid Control Strategies

Microgrid control strategies refer to the mechanisms and operational principles designed to ensure uninterrupted operation of the power system. Foremost, this involves ensuring that power producers are not endangered by

overloading or forced backpower conditions, as their failure would lead to a blackout. Secondly, critical loads must be kept powered, and smooth system operation must be maintained across various routine and exceptional scenarios—such as voltage source startups or shutdowns, or preferential loading of specific power sources. Some of these functions require rapid reaction times, particularly in exceptional circumstances, while others—like the controlled shutdown of a voltage source—can tolerate slower response.

In the marine sector, this concept is also commonly referred to as power management (Edrington, Ozkan, Papari, and Perkins (2022), Ni, Hu, and Li (2017)), which is typically closely integrated with the ship automation system. It provides the ship's crew with the necessary system-level tools to operate the vessel across various operational modes such as maneuvering, sailing, or shore connection. Depending on the system architecture, ship size, and equipment portfolio, additional operational modes may exist, and these can translate into different mode sets for ship sub-systems.

As microgrids themselves are not a new concept, the same applies to their control strategies. A substantial body of academic research has emerged on this topic. The literature commonly classifies microgrid control strategies into three types: centralized, decentralized, and distributed (Khushoo, Sharma, and Kaur (2022), Kant, Singhal, Mahto, and Jain (2022), Abhishek et al. (2020), Zhang, Zhang, Zeng, and Yang (2018), Jin, Savaghebi, Vasquez, Meng, and Guerrero (2016), Garg, Joshi, and Oruganti (2016)). Some authors, such as Dhir, Garg, and Mahajan (2022) and Abhishek et al. (2020), consider the hierarchical control strategy introduced by Guerrero et al. (2011) as a separate category. However, in this work it is treated as a variation of centralized control, since it is based on an upper-level controller that continuously adjusts the DC voltage references of equipment to compensate for droop-induced voltage deviations.

Naturally, real-world systems rarely fit neatly into one category. A given implementation may contain aspects from multiple control strategies, influenced by the architecture of the system, the equipment used, and the vendors involved. The following sections briefly describe the main features of each control method.

2.4.1 Centralized

In a centralized microgrid control strategy, the control system for the microgrid is a single controller, see Figure 5. It may have backup redundancy hardware available and even implement a hot-swapping of the controller if the main one malfunctions, but from the control point of view that single controller controls everything. It has to rely on fast communication buses to receive data from the equipment and hardwired signals for controlling them.

Shipboard power systems could easily be categorized as this kind of system, where everything is centralized, and sometimes they are, but it is not a standard practice. Centralization is not a very good design principle for shipboard power systems as it creates a single failure point that can stop the power supply of the whole ship.

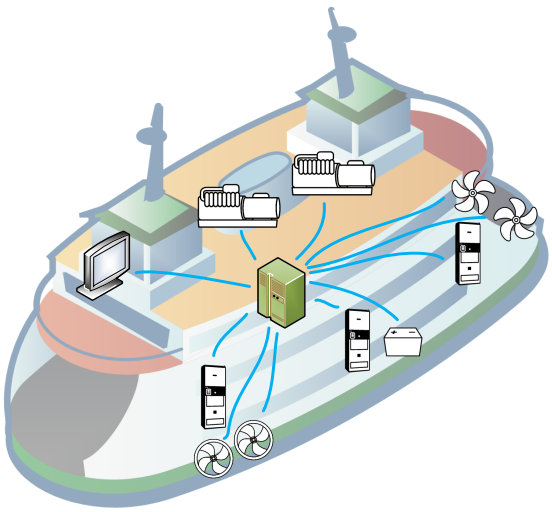


Figure 5. Centralized control.

2.4.2 Decentralized

In decentralized strategy the system is divided into smaller sections, which then have their own controllers, see Figure 6. This resembles the most how actual shipboard power systems are built: the ship is divided in to, e.g., the main power system, shaft-generator power system, and energy storage

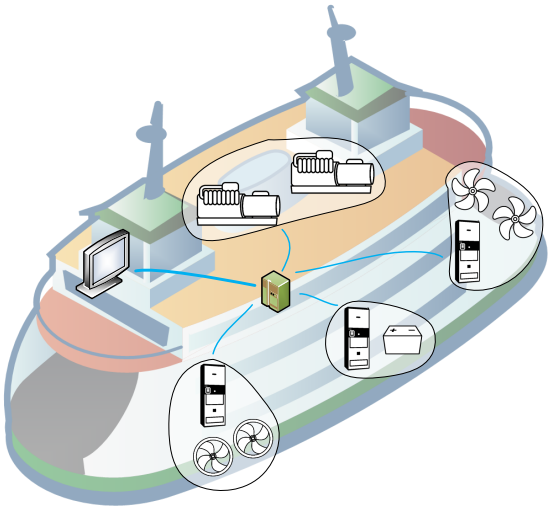


Figure 6. Decentralized control.

system. The main power system, i.e., microgrid controller or power management system, can then command the other systems to provide power or charge the batteries, etc.

In real-life ships the power management system can be divided, e.g., into two separate parts, one for port and one for the starboard side of the ship. From a user control point of view a single system is controlled, but due to the decentralized nature of the control strategy, the starboard side controller, for example, can be completely lost, and the ship can still be used with half of the functionalities.

This kind of division is in line with the basic ship design principles as well. Ships are divided into fire-zones that are designed so that even if a complete fire-zone with all of its equipment is lost, the ship will have at least 50% of main propulsion power available.

2.4.3 Distributed

The variable voltage-based control strategy presented in this work belongs to the category of distributed control strategies. The equipment in the microgrid works independently according to their own parameters, and their continuous operation is not dependent on any communication lines, see Fig-

ure 7. The communication line is needed between the equipment and control system for the ship's operational purposes, e.g., to make certain that at least the rules' specified minimum amount of power sources are running near the shore, and that the operational parameters are set correctly to the equipment in particular ship level operation modes.

The necessity of the ship-level control system stems from the ship operational requirements, rather than from the low-level control of ship equipment. The equipment in distributed control acts according to its non-time critical control parameters and references that the control system assigns according to the ship's operational requirements, and all time-critical decisions and actions are made independently by the equipment itself.

Thus, distributed control strategies have the advantage that the continuous operation of the system and the equipment therein are not relying on the control system performing low-level control. We will cover this independent operation from the variable voltage-based control strategy's perspective more closely in Chapter 5 of this work.

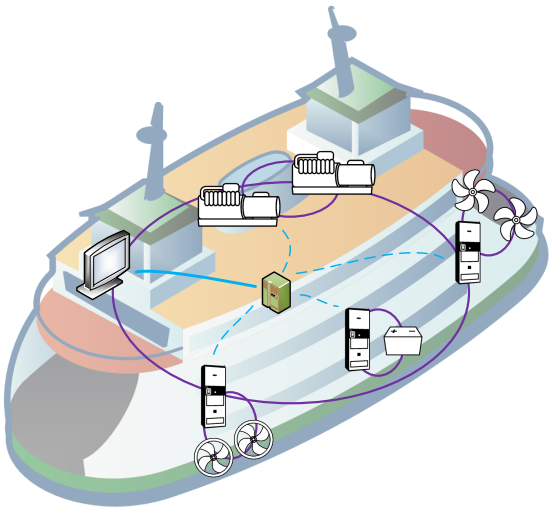


Figure 7. Distributed control.

2.4.4 On the Control Strategies

There is no single answer that states which control strategy is generally the best. Every control strategy has its own benefits and drawbacks, and analysis on what control strategy gives the best possible outcome in each situation is based on the system at hand, and how much each strategy's pros and cons weigh into the decision making process. Table 1 compares the strategies on a functional level and lists certain aspects that are beneficial to take into account when making the decision.

For shipboard power systems, however, there are certain design rules that are universal for all ships, e.g., no single fault must be able to cause loss of propulsion (DNV AS (2024), Bureau Veritas (2025)). This single rule already affects the system design quite significantly, as it raises multiple concerns related to the possible faults:

1. How to ensure the captain does not accidentally overload the generator sets just by increasing the electric propulsion's reference?
2. How to ensure a generator set suddenly tripping will not cause a chain reaction which will overload and trip the rest of the generator sets as well?
3. How to utilize the shipboard energy storage to perform blackout pre-

Table 1. DC microgrid control strategy comparison (OpenAI (2025a)).

Feature	Centralized	Decentralized	Distributed
Control Mechanism	Single microgrid controller (EMS/PMS) gathers data and issues commands to all equipment.	System partitioned into autonomous subsystems (e.g., port/starboard, shaft-generator, ESS), each with its own controller; limited coordination between subsystems.	Local controllers act on local measurements (e.g., droop / variable voltage – based); ship-level system mainly sets modes/parameters, not fast closed-loop control.
Communication Dependency	Very high: fast buses to receive data and hardwired/signaled outputs to command equipment.	Moderate: communication within each subsystem; minimal cross-subsystem dependence.	Low: communication chiefly for configuration/monitoring; closed-loop decisions are local.
Fault Tolerance	Low: inherent single point of failure at the central controller (even if hardware redundancy exists).	Good: failure of one subsystem does not collapse the others; fits ship fire-zone philosophy.	Very high: no single controlling node; units continue operating independently if comms/nodes fail.
Reaction Speed	Limited by communication and central processing latency for fast events.	Fast within each subsystem (local decisions); slower only for inter-subsystem coordination.	Very fast: decisions are local, avoiding communication and centralized decision delays.
Scalability	Poor: adding units increases central computation and communication burden.	Good: modular expansion by adding subsystems with their own controllers.	Excellent: adding units preserves local behavior with minimal coordination overhead.
Applicability to Shipboard Systems	Limited / not preferred: centralization creates a critical single failure point for a ship.	Good fit: mirrors common marine practice (redundant halves, ≥50% propulsion retained).	Very suitable: robust, responsive, and aligned with ship operational needs when parameters are set by ship-level logic.

vention in case the only running generator set is tripped?

As for items #1 and #2 from above, the traditional answer is for the system to constantly calculate the "power available"-value, and based on it assigning limits for the loads to prevent generator sets from being overloaded. The "power available" signal is calculated from the sum of the running power generation units' maximum allowed outputs and the sum of running loads' consumptions, which gives the amount of power that is still available for the ship loads to utilize. This "power available" -value is all that stands between the ship running smoothly and a blackout, which then raises a lot of new questions and concerns regarding the accuracy and possible faults affecting the calculation the value, etc. This value is not dependent on the microgrid control strategy directly, and it can be used or at least made available for the crew with any strategy. Its existence—while being completely logical, clearly usable, and intuitive value for the crew—is, however, the result of traditional microgrid control strategies utilized in shipboard power systems.

The reliance on a single value so critical to the ship's operation—such that operating without it while avoiding blackouts would be extremely difficult—creates a single point of failure in itself. As without it, the microgrid controller is unable to assign the limits for the loads and the loading of the generator sets would have to be controlled manually. While as mentioned before, system designers have developed redundancies and fail-safes for it, however, these only further increase the complexity of the system.

If the equipment running the loads, power generating units, and energy storage would inherently know the system loading situation due to their own measurement, the necessity to rely on the centrally calculated "power available" -value would disappear. This would effectively enable full utilization of the distributed microgrid control strategy in the shipboard power system. The distributed microgrid control strategy also has the inherent benefit of distributing the decision making power as low in the system architecture as possible—in this case, the inverters themselves, that are, for example, running the propulsion motors or running the energy storage units—which eliminates the need for communication and the resultant communication and decision making delays.

This speed of decision making in distributed microgrid control strategy is

one of the possible answers to all of the items #1, #2, and #3 above. In larger systems of multiple system architectural layers, the need for decision making speed reduces as we move further up from the ground level in the architecture. For example, a very large cruise ship or a super-tanker can have so large electrical system that it is divided into microgrids in itself, and thus there needs to be a multi-microgrid control strategy running those microgrids in tandem. These systems, however, are outside of this work's scope.

3 UNDERSTANDING DC MICROGRIDS

Before getting into the details of variable voltage-based DC microgrid control, the mechanisms behind the variable voltage behavior have to be understood. That is, how the inverters running in the DC microgrid on the one hand contribute to the voltage variations and on the other can take advantage of them. The different mechanisms are based on their active operation mode, which in turn is based on their role in the system. With this, it is possible to model the different real-life scenarios reliably and prove that the functionalities are working.

From the DC microgrid control point-of-view, one could see that there are only two roles for the inverters: either actively maintaining the DC-link voltage, i.e., a voltage source, or not, i.e., a current source or sink. These are basically the two possible roles inverters can have toward the DC-link itself, but they are not mutually exclusive. While an inverter working as a voltage source is not in most cases required to independently change its operation, the inverters working as current sources or sinks can, and in certain situations have to, change to voltage source mode through their under- and overvoltage regulators.

This capability for an inverter to change its operation mode from current source or sink to DC voltage source is not a new inverter feature in itself, but rather an overlooked one at least in academia as there is practically no research done utilizing these functionalities. The capability for load inverters to dynamically change the operation mode to support the DC microgrid voltage reveals completely new kinds of possibilities for building a microgrid control strategy. The challenge in modeling microgrids containing these said load inverters is that existing simulation software do not inherently support the operational mode change. There can be a number of reasons why, but one of the most obvious is that the algorithms associated with the mode changing are inverter supplier specific and not available to the public. And even if they did support it, there is no guarantee that the algorithms utilized would be usable for the research at hand, and it would still have to be programmed manually.

Thus, as it is imperative for the research that the inverter operation mode change works in a compatible way with the control strategy in question, a

modeling paradigm must be developed to support these types of inverters. Creation of a modeling paradigm gives complete freedom in developing the necessary algorithms to achieve the control mode change in a simulated environment, without limitations of existing simulation environments. The algorithms are then transferable to real-life inverters if further real-life study is made.

3.1 System Mathematical Modeling

Numerous modeling paradigms have been proposed for DC microgrids, often reflecting the specific focus or preferences of individual researchers. In many cases, the models incorporate detailed physical elements, such as cable inductances, as seen in Abdolmaleki, Babak and Bergna-Diaz, Gilbert (2022) and Ghanbari, Bhattacharya, and Mobarrez (2018). While such high-fidelity modeling is valuable for stability analysis and transient behavior studies, it may be excessive for research focused on control strategy development. Similarly, equipment such as inverters and converters are frequently modeled with internal capacitances and inductances Rashad, Ashraf, Bhatti, and Minhas (2018); Y. Wang, Wei, and Zuo (2021), which is justified in studies requiring detailed representation of device-level dynamics. However, from the perspective of DC microgrid control strategy research, such detail often introduces unnecessary complexity without improving the practical relevance of the results.

The study and development of power management strategies, however, revolve around time scales relevant to the control systems implementing them. In practical shipboard automation systems, the control logic typically operates on a one-second cycle, where measurements are read and commands issued once per second. In centralized control strategies, equipment reacts to fast events based on its internal design and parameterization, while the control system manages the overall behavior by responding to the consequences of those events at the slower control-system timescale.

Even in distributed control strategies—such as the one considered in this work—the transients themselves remain largely irrelevant from the perspective of DC microgrid control. Equipment still responds to fast disturbances

according to its configuration, but the decentralized control logic enables it to independently react at the power management level when, for example, a load is disconnected or a source is tripped due to a fault.

3.1.1 On the Modeling Paradigm

This section employs a matrix-based quasi-static model of the DC microgrid in which only the virtual/droop resistances of voltage-supporting equipment (e.g., voltage sources and load inverters when their under- and overvoltage regulators are active) and cable resistances are considered. This simplification omits capacitances and inductances and therefore excludes fast transient dynamics. The loss of fidelity affects only very fast, time-dependent simulations aimed at transient analysis, which are outside the scope of DC microgrid control strategy assessment and of this work.

In the model, the matrices represent the current injected into or drawn from the DC-link by each device, irrespective of its role (voltage source or load). Each device's contribution is computed in a uniform manner; what differs is the origin of the contribution (e.g., droop behavior versus regulator action). The DC-link voltage—common to all equipment—is then obtained from these contributions together with the modeled resistances.

The matrix-based approach scales well: devices can be added to or removed from the matrices with minimal structural changes. Inter-source currents are explicitly representable when the corresponding terms are included: both circulating currents caused by voltage measurement errors and imbalance currents resulting from differing voltage references or droop coefficients. The framework also extends to nodal analysis with a network resistance matrix; dynamic elements (capacitances and inductances) could be incorporated in future work, with the understanding that this introduces state variables and requires time integration beyond the present quasi-static scope.

For exposition, the system is first treated in an idealized setting with no inter-source currents (neither circulating nor imbalance), loads modeled as simple current sinks, and a simplified DC-link. Thereafter, the modeling is extended to include circulating currents and the conditional voltage-source behavior of loads when their under- and overvoltage regulators activate—a

core element of the variable voltage-based DC microgrid control strategy. Finally, non-ideal microgrids are addressed via nodal analysis with non-negligible cable and busbar resistances.

Currents are signed from the potential perspective: a current directed toward the higher-potential (DC+) is positive, and a current directed toward the lower-potential (DC-) is negative. To smoothly vary the load level in the examples, the smootherstep function Perlin, Ken (2002) is used to generate loading profiles, which also expose behavior under voltage measurement errors and create cases where loads' under- and overvoltage regulators activate and deactivate.

Alongside the equations, MATLAB code listings for the modeled scenarios are provided for transparency and reproducibility. The listings are formatted for direct copy-paste into MATLAB; some long lines may reduce readability.

3.1.2 Voltage Sources

The voltage source's operations purpose is quite straightforward: it maintains the DC voltage at its reference level, which, if drooping is in use, is allowed to deviate internally according to the inverter's loading, see Equations (1.1) and (1.2). The practical limiting factors being, e.g.: current limits to either direction of current flow and if it is actually in operation or has it tripped to fault or simply stopped. If any of these limiting factors are in an effect, like current limit is hit, either some other equipment compensates or the voltage goes to an uncontrolled increase or decrease. For the purpose of this dissertation, it is assumed that DC drooping is the paralleling mechanism in use for the voltage sources.

Traditionally the voltage sources in DC circuit analysis have been modeled simply as voltage sources, i.e., devices that take in voltage reference and forcefully keep the potential difference between their poles at the specified level, and let the current fluctuate according to load and/or generation variations. Adding internal resistance, which mathematically functions as droop resistance, does not changed this; it is then simply modeled as a non-ideal voltage source.

As mentioned before, the modeling paradigm is based on a current matrix that contains all the equipment's currents. Further, droop controlled voltage sources' current output is directly proportional to the difference between the reference used and the actual DC voltage. Thus, to calculate the current contribution of a voltage source, it can be assumed that the DC-link voltage V_{DC} follows exactly V_{DS}^{*} from Equation (1.2), then it is possible to derive how the source current I_{S} behaves in the function of DC-link voltage V_{DC}, see Appendix 1:

$$I_{\mathrm{S}} = \frac{V_{\mathrm{S}}^{*} - V_{\mathrm{DC}}}{\zeta_{\mathrm{S}} V_{\mathrm{S}}^{*}} I_{\mathrm{nomS}} \tag{3.1}$$

where subscript S indicates the variables are for voltage source named S, and I_{nomS} stands for the voltage source's nominal current. As the voltage reference V_{S}^{*} and drooping coefficient ζ_{S} are controllable parameters by an external controller or an user, DC voltage V_{DC} is the only free variable. Now the fraction in front of I_{nomS} can be presented as a separate function Γ_{S}:

$$\Gamma_{\mathrm{S}} = \frac{V_{\mathrm{S}}^{*} - V_{\mathrm{DC}}}{\zeta_{\mathrm{S}} V_{\mathrm{S}}^{*}}, \tag{3.2}$$

which gives us the proportional value of how much contribution is required from the voltage source S. The MATLAB code implementation for the Γ function is presented in Code 1.

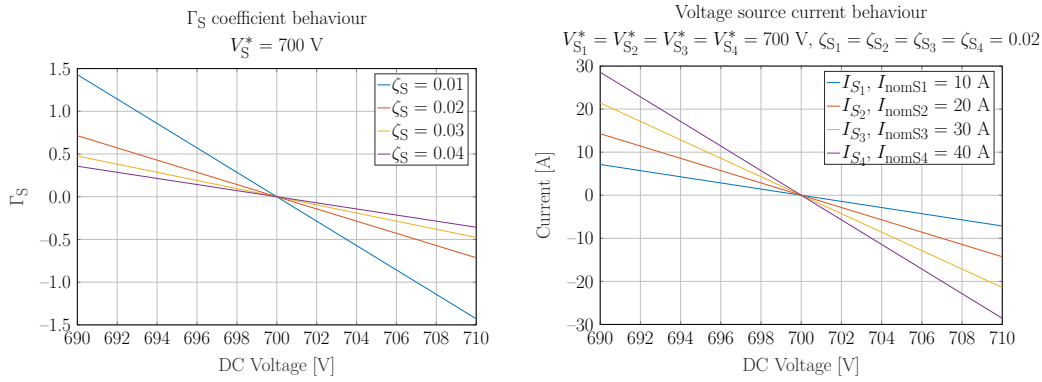
Figure 8a shows the Γ_{S}'s behavior in the voltage domain with different values for drooping coefficient ζ_{S}. Figure 8b shows differently sized voltage sources' current behavior in the voltage domain around the specified DC reference voltage V_{S}^{*} of 700 V, and with static drooping coefficient ζ_{S} of 0.02.

Now the equation for the current I_{S} can be simplified by using the Γ_{S} func-

```
%% Gamma Function
function output = Gamma(DCVoltage, DCVoltageReference, DCDroopingCoefficient)
    if DCDroopingCoefficient == 0
        error('Gamma:ZeroDroop', ...
            'Zero droop not supported, leads to division by zero.');
    elseif DCVoltageReference == 0
        error('Gamma:ZeroReference', ...
            'Zero reference not supported, leads to division by zero.');
    end

    output = (DCVoltageReference - DCVoltage)/(DCDroopingCoefficient * DCVoltageReference);
end
```

Code 1. MATLAB code for Γ function.



(a) Behavior of Γ with different ζ values across a voltage range of 690–710 V.

(b) Differently sized voltage sources' current behavior in the voltage domain with DC voltage references of 700 V and drooping coefficients of 0.02.

Figure 8. Behaviors of Γ and current in the voltage domain.

tion, and voltage source current equation now becomes,

$$I_S = \Gamma_S I_{\text{nomS}}. \tag{3.3}$$

from which we can deduce a more generalized formula for the DC microgrid voltage sources' loading distribution $\mathbf{I}_S$:

$$\mathbf{I}_S = \text{diag}(\boldsymbol{\Gamma}_S)\mathbf{I}_{\text{nomS}} = \begin{bmatrix} \Gamma_{S_1} & 0 & \dots & 0 \\ 0 & \Gamma_{S_2} & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \Gamma_{S_N} \end{bmatrix} \begin{bmatrix} I_{\text{nomS}_1} \\ I_{\text{nomS}_2} \\ \vdots \\ I_{\text{nomS}_N} \end{bmatrix} = \begin{bmatrix} I_{S_1} \\ I_{S_2} \\ \vdots \\ I_{S_N} \end{bmatrix}. \tag{3.4}$$

With this first mathematical tool, it becomes possible to perform basic modeling of the currents contributed by voltage sources in a system with a prescribed DC-link voltage profile. Consider a system consisting of four voltage sources with nominal current ratings of 10 A, 20 A, 30 A, and 40 A, as illustrated in Figure 9. The DC-link voltage is defined to follow two consecutive smootherstep transitions. The operational parameters of all voltage sources are set to have a DC voltage reference of $V^* = 700$ V and a droop coefficient of $\zeta = 0.02$.

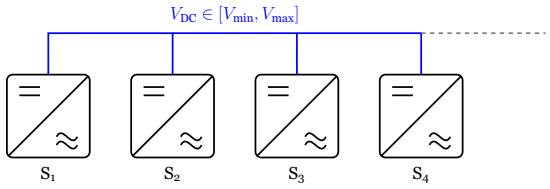


Figure 9. Modeled system where the DC voltage is following predetermined values and load is not considered.

Having the sources' voltage references and droop coefficients identical, corresponds with the systems in real-life, where the aim is to proportionally share the load between the sources. Having the voltage references and/or the droop coefficients differing from each other, would result in proportionally uneven generator load distribution which is generally an unwanted situation in ships. The modeling paradigm however can handle sources with different references and droop coefficients which are covered directly in sections 3.1.4, 3.1.5 and 3.1.8 and chapter 4, as well as indirectly in form of modeled cable resistances in section 3.1.10.

The resulting currents can now be calculated using the Γ function presented in Equation (3.4) and Figure 10 shows the results. The DC voltage is pre-

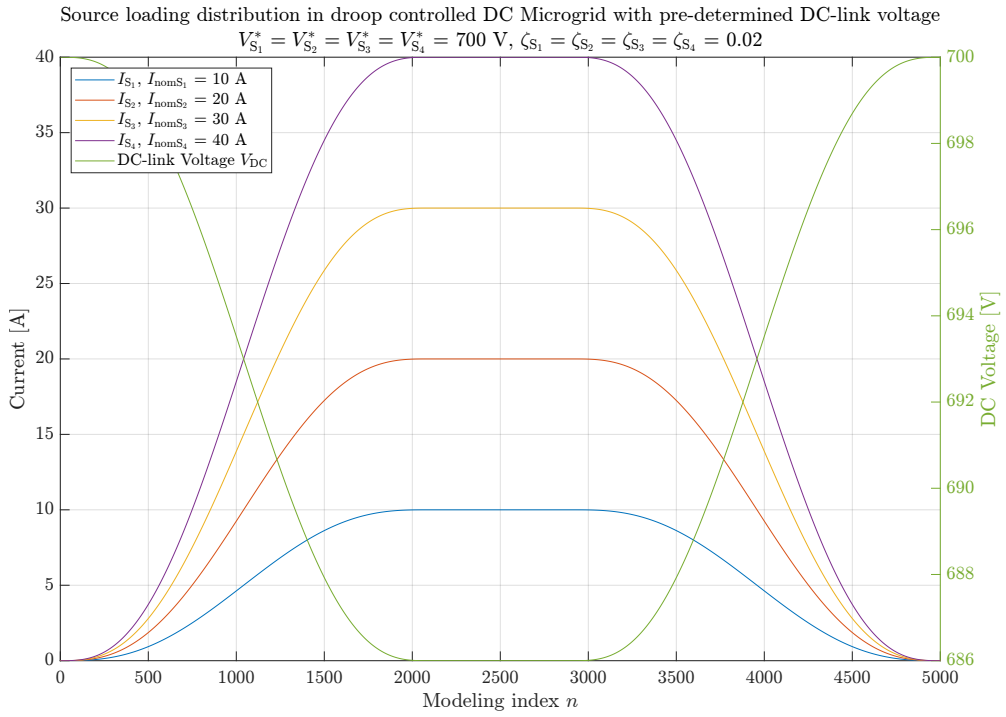


Figure 10. Load distribution calculated by using Γ of four ideal voltage sources in DC microgrid of predetermined voltage.

determined to be varying between 700 and 686 V. When the DC voltage is at the DC voltage reference of 700 V, the source currents are zero, as the difference between reference and actual voltage is zero. As the voltage deviates further away from the DC voltage reference, the outputs of the voltage sources increase, just as the drooping definitions in Equations (1.1) and (1.2) predict. 686 V was selected as the low peak for the DC voltage because this level corresponds with with 2% deviation from the nominal, resulting nominal loading of the sources due to drooping coefficient of 0.02.

MATLAB code for the modeling is presented in Code 2. First in the code variables needed in the modeling are initialized, next sources' nominal currents, DC voltage references and droop coefficients are defined, and the used DC voltage behavior is determined. The last part with the for-loop is the actual modeling where the modeling steps are ran through and at each modeling

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [10; 20; 30; 40];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('A_SourceOnlyModeling:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
sourceCurrentArrays = zeros(numberOfSources,length(steps));

%% Define the predetermined DC voltage behaviour
% Create smootherstep-based DC-link voltage profile
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
s = 6*x.^5 - 15*x.^4 + 10*x.^3;
DCVolt = DCRefs(1) * (1 - 0.02 * s);

%% Run through the calculations
for n=steps
    %% Source handling
    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        GammaMatrix(j,j) = Gamma(DCVolt(n), DCRefs(j), DCDroops(j));
    end

    % Calculate the source currents
    sourceCurrentArrays(:,n) = GammaMatrix * InomS;
end
```

Code 2. MATLAB code for modeling the source currents based on predetermined DC voltage behavior.

index n the $\mathbf{\Gamma}$ matrix is populated, which is then used to calculate the source current array.

3.1.3 Incorporating the Voltage Sources to DC-Link with Predefined Loads

The source modeling methodology presented earlier provides a way to derive source currents based on the DC-link voltage. That is, the sources do not intrinsically "know" how much current they must supply. Instead, each source independently determines its current output based on the common system variable: the DC-link voltage. This enables the development of a decentralized, sequential modeling approach, as illustrated in Figure 11, in which a change in load causes a change in the DC-link voltage, prompting each source to react autonomously by adjusting its current. The change in DC voltage depends on the characteristics of the voltage-controlling equipment currently influencing the system, while the change in an individual source's output depends on its own internal parameters.

The sequential modeling paradigm, illustrated in Figure 11, offers the advantage of treating sources and loads individually. Equipment behavior is modeled in a decentralized and modular way—each source's current is computed independently of other devices. Furthermore, this structure facilitates later integration of load behaviors into the modeling algorithm, including current and power references and over-/undervoltage regulator functionalities.

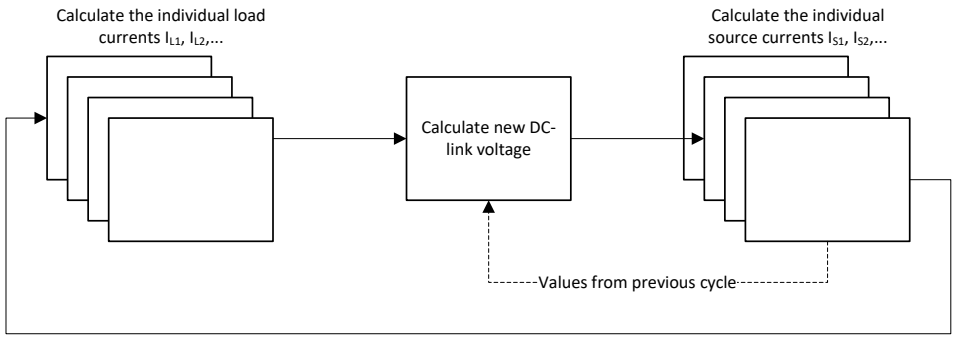


Figure 11. Modeling sequence.

3.1.3.1 Modeling the DC Voltage

With the source current contributions expressed as functions of the DC-link voltage, it becomes necessary to define how the DC-link voltage itself is modeled. Since the objective of this dissertation is to demonstrate the usability and operational principles of the variable voltage-based DC microgrid control strategy—and considering that power management strategies operate on timescales ranging from seconds to minutes, which are orders of magnitude slower than electrical transients—there is no need to model full circuit dynamics such as transient current responses. As a result, high-frequency phenomena, such as those originating from inverter switching or capacitor dynamics, fall outside the scope of this work. In real systems, such fast dynamics are handled locally by the internal control algorithms of individual devices, such as inverters, whereas the microgrid-level control strategy reacts only to the resulting control challenges, such as those caused by equipment tripping.

The modeling of inverter tripping events—which in the quasi-static context is equivalent to inverter stopping—is covered in Section 3.1.4.1.

To illustrate the relevant control timescale, Figure 12 presents measured propulsion load data from the ship *Aurora Botnia* during her sea trials. The data in this figure shows the main propulsion inverter's load behavior, initially operating in speed control mode, after which the propulsion load is gradually reduced and the control mode changes to power control. The observed load transitions occur over timescales ranging from several seconds to minutes, highlighting the temporal domain in which power management strategies are designed to operate.

In quasi-static modeling, each step is assumed to represent a steady state, with all transient phenomena considered to have settled. As a result, the modeled scenarios in this work do not include time on the figures' x-axes, since no time-dependent variables are involved. Although the real-world examples presented in later sections do include timestamps from the original data on the x-axis, time plays no role in the modeling in those cases either.

The quasi-static modeling approach not only provides a computationally efficient method for modeling the phenomena with sufficient accuracy for the

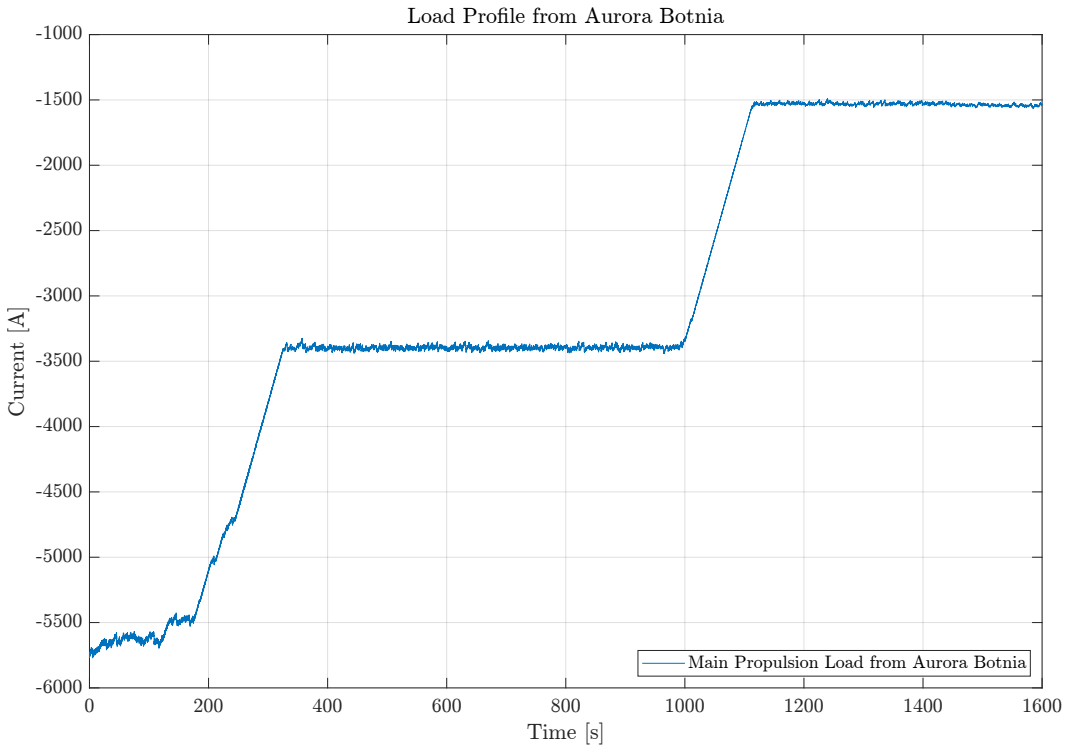


Figure 12. Loading of Aurora Botnia main propulsion during sea trials, graph presented with permission from Wasaline.

time scales involved, but also serves as both a foundation for verifying the proposed modeling paradigm and a basis for future development of dynamic models. If the verification shows that the paradigm fails under quasi-static conditions, it cannot be expected to function correctly in dynamic scenarios.

A straightforward method to compute the updated DC-link voltage can be derived from Equation (1.5), following the same principles that define how DC voltage behavior indicates the state of balance between power production and consumption. When applied in a step-by-step modeling context, the voltage at discrete time index n can be calculated as:

$$\begin{aligned} V(n) &= V(n-1) + \Delta V \\ &= V(n-1) + (I(n) - I(n-1)) \cdot R(n), \end{aligned} \tag{3.5}$$

where ΔI from Equation (1.5) is now represented as the change in current between modeling steps n and $n-1$.

Applying this formulation to the DC microgrid model, the system voltage V_{DC} is updated by adding the inter-step current imbalance—defined as the

sum of previously generated and currently consumed currents—multiplied by the system resistance value. Taking advantage of Kirchhoff's current law, which states that currents at the same modeling index must sum to zero, we note that if consumption and production currents are taken from different time steps, the resulting discrepancy can be used with Ohm's law to calculate the corresponding voltage difference. The system resistance, denoted by R_{MG}, represents the total effective resistance seen by the DC-link in the modeled system. As shown in Equation (3.4), the generated source currents are obtained from the array $\mathbf{I}_{\mathrm{S}}$, and an analogous array $\mathbf{I}_{\mathrm{L}}$ is defined for the load currents. The DC-link voltage update equation is then expressed as:

$$\begin{aligned} V_{\mathrm{DC}}(n) &= V_{\mathrm{DC}}(n-1) + \Delta V_{\mathrm{DC}} \\ &= V_{\mathrm{DC}}(n-1) + \Delta I \cdot R_{\mathrm{MG}} \\ &= V_{\mathrm{DC}}(n-1) + \left(\sum \mathbf{I}_{\mathrm{L}}(n) + \sum \mathbf{I}_{\mathrm{S}}(n-1)\right) \cdot R_{\mathrm{MG}}(n), \end{aligned} \tag{3.6}$$

where the summation notation implies summing over all elements of each current array. As previously defined, currents produced by the equipment are considered positive, and currents consumed are negative. Therefore, a direct summation of the current arrays yields the system's inter-step current imbalance.

In this formulation, if the load remains unchanged between time steps, the total current imbalance is zero and the DC-link voltage remains constant. Any change in load results in a proportional voltage adjustment, governed by the inter-step current imbalance between generation and consumption, and scaled by the system resistance—the combined resistance in the modeled circuit. The corresponding MATLAB implementation is provided in Code 3.

In the idealized case where cable and busbar resistances are neglected, R_{MG} is determined solely by the parallel combination of the voltage sources' droop resistances R_{DS}.

The droop resistance R_{DS} for each source is calculated from the nominal source current I_{nomS}, voltage reference V_{S}^{*}, and droop coefficient ζ_{S}, as follows:

$$R_{\mathrm{DS}} = \frac{-\Delta V_{\mathrm{DS}}^{*}}{\Delta I_{\mathrm{S}}} = \frac{-(V_{\mathrm{S}}^{*}(1-\zeta_{\mathrm{S}}) - V_{\mathrm{S}}^{*})}{I_{\mathrm{nomS}} - 0\,\mathrm{A}} = \frac{V_{\mathrm{S}}^{*}\zeta_{\mathrm{S}}}{I_{\mathrm{nomS}}}, \tag{3.7}$$

These parameters may vary during system operation. For example, changes

```
%% DC-link voltage calculation
function newDCVoltage = calculateDCVoltage(dcVoltage, generation, load, R_MG)
    tempVolt = dcVoltage + ((generation + load) * R_MG);

    % Handle situations where:
    % 1. No active voltage sources exist and R_MG = Inf
    % 2. Voltage has crashed to zero
    if isnan(tempVolt) || tempVolt <= 0
        newDCVoltage = 0;
    % Limit maximum voltage to not crash the modeling in case load with OV
    % regulator exists and "catches" the voltage
    elseif tempVolt == Inf
        newDCVoltage = 1000;
    else
        newDCVoltage = tempVolt;
    end
end
```

Code 3. MATLAB function for DC voltage calculation.

occur if a source reaches its output limit or is shut down—as discussed in Section 3.1.4—or during voltage source startup sequences where the voltage reference is ramped, as described in Section 5.2.2. Or when a load's under- or overvoltage regulator activates—as discussed in Section 3.1.7. These changes directly affect the resulting value of R_{MG} and are discussed in their respected sections.

As shown in Figure 11, the source currents in the algorithm are updated after the DC voltage calculation and are then used to compute the DC voltage in the next modeling step. This structure is reflected in Equation (3.6), where the $\mathbf{I}_{\text{S}}$ values correspond to modeling index $n-1$. The effects of any operational changes in the voltage sources at step $n-1$ are thus taken into account when computing the DC voltage at step n.

Consequently, if R_{MG} varies due to source operation at modeling index $n-1$, the resulting impact is reflected in the DC voltage level at index n.

To visually derive the terms used in the ΔV_{DC} calculation, consider the system in Figure 13, which consists of four droop-controlled voltage sources and a single current source as a load. The circuit assumes no cable or busbar resistances. The equivalent form used for the ΔV_{DC} derivation is shown in Figure 14.

Each voltage source – droop resistance pair may be interpreted as a Thevenin equivalent circuit, which can be transformed into a Norton equivalent with a known current value, as shown in Figure 15. At this point, the Norton cur-

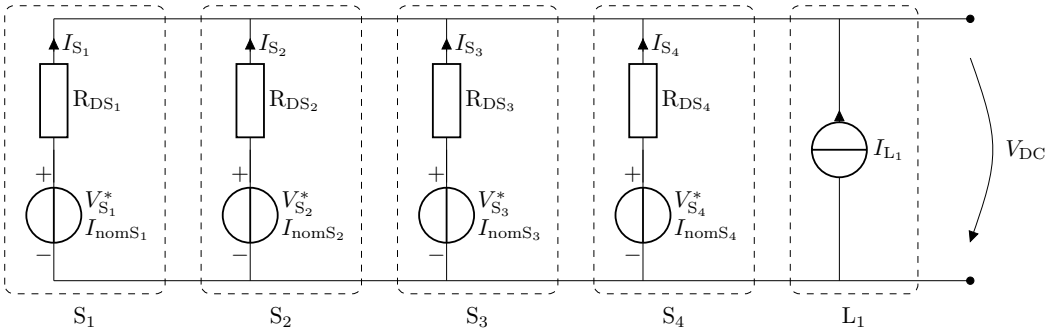


Figure 13. Circuit illustrating a system composed of a single load and four droop-controlled voltage sources, where the internal resistances represent droop behavior.

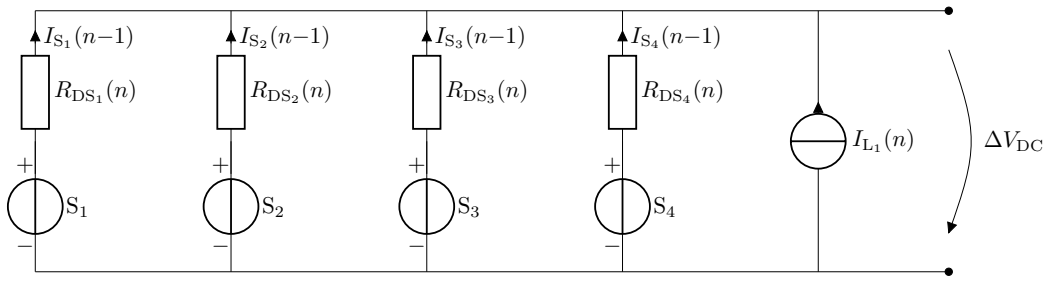


Figure 14. Equivalent circuit of the system based on Figure 13 used to derive the ΔV_{DC} calculation in Equation (3.6).

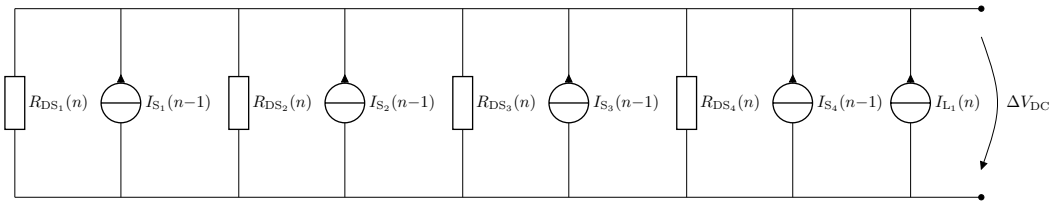


Figure 15. Circuit in Figure 14 after transforming each source to its Norton equivalent.

rent sources can be summed to obtain the total current imbalance between the modeling steps as in Figure 16, which is used in the ΔV_{DC} calculation in Equation (3.6).

It is now straightforward to compute the system resistance R_{MG}, illustrated in Figure 17, as the parallel combination of the individual droop resistances which are the only resistances in the system:

$$R_{\text{MG}}(n) = \left(\sum \frac{1}{R_{\text{DS}}(n)} \right)^{-1}. \tag{3.8}$$

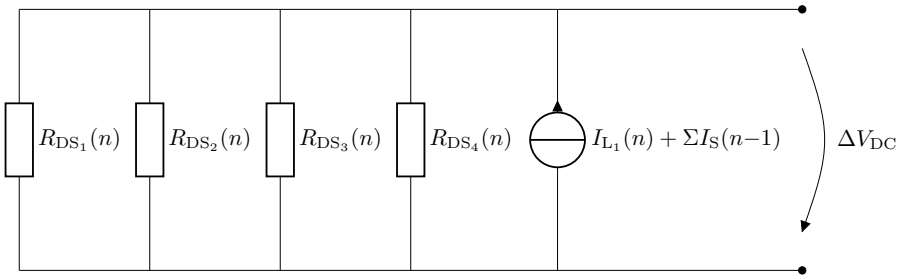


Figure 16. Circuit in Figure 15 after summing the current source contributions.

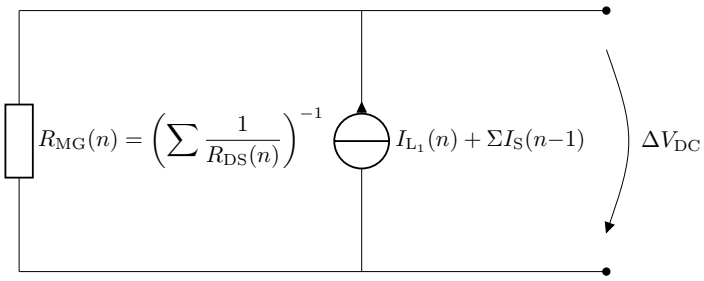


Figure 17. Equivalent system resistance obtained by combining the paralleled droop resistances.

The modeling sequence presented above assumes an ideal system where cable and busbar resistances are neglected. However, there is no restriction preventing these resistances from being included in this method. Modeling a system with cable and busbar resistances requires computing the terminal voltage individually for each voltage source and load. Consequently, both the current deviation ΔI and the system resistance R_{MG} must be evaluated from each source's perspective, since each source experiences a different local terminal voltage and line resistance. In the idealized system presented earlier, all voltage sources observe the same DC-link voltage due to the absence of cable and busbar voltage drops. Therefore, a single scalar value for V_{DC} is sufficient to describe the system behavior.

More efficient way to consider the cable and busbar resistances is to utilize nodal analysis, which enables to calculate the voltages over every equipment in the system in a single matrix operation. While this section covers the system modeling by omitting the cable and busbar resistances, the nodal analysis is covered in Section 3.1.10.

3.1.3.2 On the Quasi-static Nature of the Modeling

The quasi-static modeling paradigm employed in this work omits fast voltage transients and is therefore not intended for analyzing short-term dynamic phenomena. However, such high-speed dynamics are irrelevant at the seconds-to-minutes time scales at which the power management strategy operates. While sudden events—such as an abrupt voltage source shutdown—can be captured (see Section 3.1.4.1), the associated transient waveforms fall outside the scope of the intended modeling horizon.

From the control system's perspective, even cascading device trips would appear as a (near) simultaneous loss of multiple devices. The power management strategy then operates on the updated system state observed at the next modeling step, without requiring insight into the sub-second transient sequence that led to the loss.

This is not to say that phenomena such as voltage dips caused by constant power loads are ignored. Rather, while the transient initiating such behavior cannot be captured, and while the whole scenario is dynamic in nature, a quasi-static approximation of the resulting voltage depression could be modeled. In this sense, the control-related consequences of constant power loads—including the destabilizing feedback loop they may introduce—can still be explored using quasi-static methods. The modeling framework can represent inverter behavior that emulates constant power loads and examine the conditions under which instability would arise, along with the control strategy's response.

All test cases throughout this work are deliberately constructed to represent quasi-steady scenarios. These were selected to highlight the operational capabilities of the proposed control strategy and to reflect realistic time scales of power management actions. Even in the real-life blackout prevention case discussed in Section 5.6.2, where inverter monitoring data was recorded during the event, the system behavior was effectively quasi-static throughout the relevant duration. The control system's only concern was how inverters were parameterized to maintain DC-link voltage if and when such blackout-inducing scenario occurs.

Accordingly, the quasi-static modeling paradigm provides a valid and suffi-

ciently accurate approximation of system behavior at the relevant time scales for power management and control. It achieves a balance between model simplicity and fidelity required to support the proposed control framework.

A logical future extension would be to build upon the DC voltage calculation framework presented here and incorporate a more detailed voltage dynamic model capable of simulating transient behavior. Such an enhancement would be particularly useful for applications like system-level fuse sizing, where sub-second fault currents must be accurately captured.

3.1.3.3 Utilizing the DC Voltage Modeling

Figure 18 shows the behavior of a DC microgrid supplied by four voltage sources and loaded by a load with the predetermined current profile $\mathbf{I}_{\mathrm{L}}$ varying between 0 and 100 A. The modeled system's circuit is presented in Figure 13 and as a simplified single-line diagram in Figure 19. It is an extended version of the system presented in Figure 9, with a current $I_{\mathrm{L_1}}$ following a

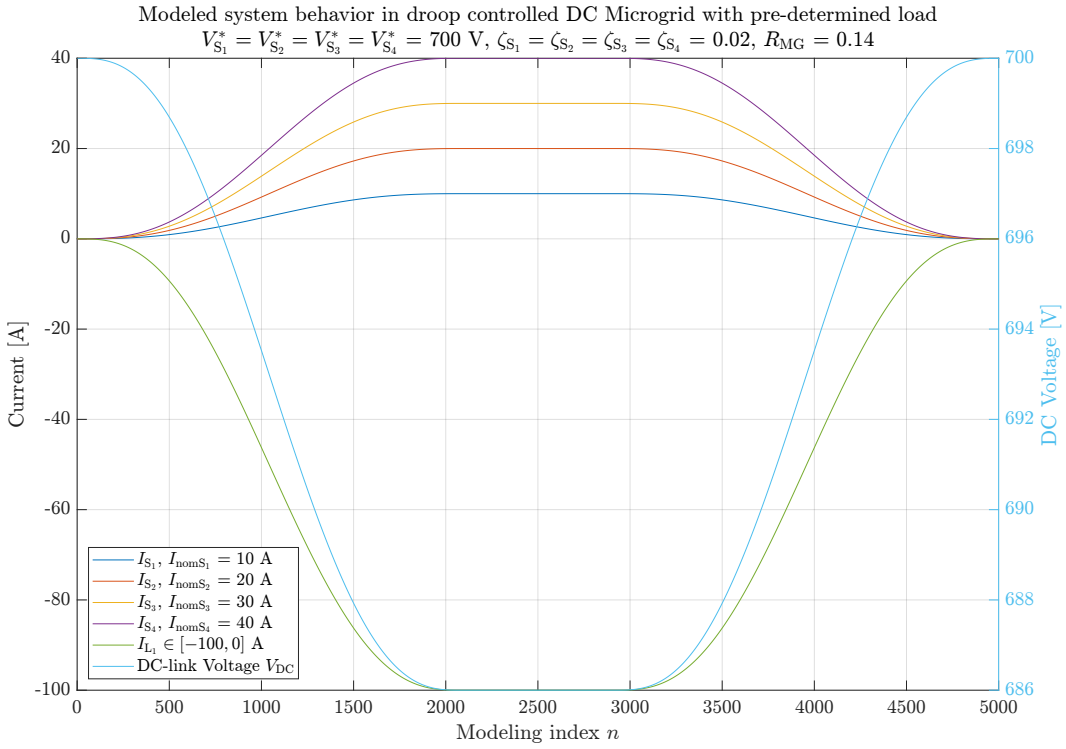


Figure 18. Load distribution of four ideal voltage sources in DC microgrid with predetermined load profile.

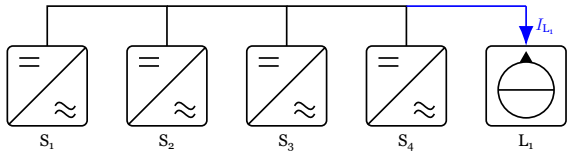


Figure 19. Modeled system with four voltage sources and ideal load.

load profile to produce similar DC voltage behavior as in the previous example. In the single-line diagram of Figure 19, ⊖ symbol represents a perfect current sink, whose load profile is predetermined before hand and is incapable of being adjusted from outside of the equation via references, and does not have any under- or overvoltage regulator functionalities.

Code 4 presents the MATLAB code for the modeling of the system, where a distinct section for DC voltage handling is added. The load profile is not written anymore directly as the DC voltage as was in Code 2 but as the actual load current, which is then used to calculate the DC voltage values.

As the total of the source nominal currents and the load peak equals 100 A, and all sources share identical voltage references of 700 V with 2% droop, the voltage dips to 686 V when nominal loading is reached. This behavior matches that of the previous example in Figure 10, where only the DC voltage was modeled. The difference lies in the voltage trajectory during load changes, discussed further in Section 3.1.3.4, which results from the DC voltage calculation method.

This level of DC microgrid mathematical modeling is sufficient for basic analysis of system-level behavior, where even complex current-based load profiles can be represented. However, this method lacks the ability to capture more advanced load behaviors, such as alternative reference types or the operation of under- and overvoltage regulators. Consequently, it is not adequate when considering a complete control strategy based on variable DC voltage. To support such strategies, the model must be extended to accommodate variable loading changes driven by external references and regulator responses. Further, incorporating circulating current behavior between voltage sources and voltage-supporting loads provides additional insight beneficial for system design.

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [10; 20; 30; 40];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('B_ModelingWithDCLink:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
sourceCurrentArrays = zeros(numberOfSources,length(steps));

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources, 1);
droopConduc = zeros(numberOfSources, 1);
R_MG = 0;

%% Create smootherstep-based load profile varying between -100 and 0 A
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
s = 6*x.^5 - 15*x.^4 + 10*x.^3;
load = -100 * s;

%% Run through the calculations
for n=steps
    %% DC voltage handling
    % Calculate R_MG
    for r=1:numberOfSources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = 1/droopResis(r);
    end
    R_MG = 1/sum(droopConduc);

    % Calculate the DC voltage
    if n==1 % If first step use default values
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, load(n), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), ...
            sum(sourceCurrentArrays(:,n-1)), load(n), R_MG);
    end

    %% Source handling
    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        GammaMatrix(j,j) = Gamma(DCVolt(n), DCRefs(j), DCDroops(j));
    end

    % Calculate the source currents
    sourceCurrentArrays(:,n) = GammaMatrix * InomS;
end
```

Code 4. MATLAB code for modeling system behavior with predetermined load.

3.1.3.4 Note on stepwise path-dependence of the DC-link model

Equations (3.5) and (3.6) update the DC voltage by comparing the sum of load currents at modeling index n to the sum of source currents from the previous index $n-1$. This discrete update rule (with no time-domain capacitance) introduces a small path-dependent offset at load transitions: the instantaneous V_{DC} at the same numerical load level depends on the preceding step.

For example as can be seen in Figure 18, when sweeping towards $100\,\mathrm{A}$, the load passes $50\,\mathrm{A}$ at about $691\,\mathrm{V}$. While calculating a standalone $50\,\mathrm{A}$ operating point via Equation (3.2), and modeling with $50\,\mathrm{A}$ maximum load in Figure 20, gives the V_{DC} of $693\,\mathrm{V}$.

This is not a physical transient but a modeling artifact arising from the n vs. $n-1$ current imbalance; it vanishes in steady segments and does not affect the methodology elsewhere.

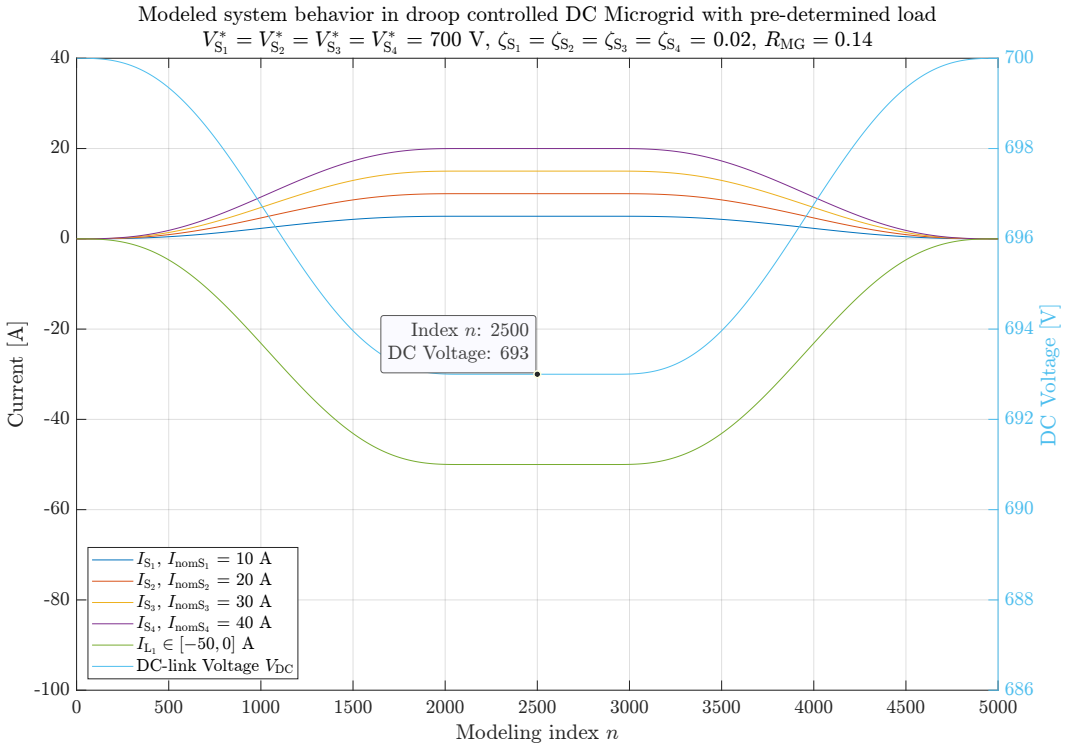


Figure 20. Load distribution of four ideal voltage sources in DC microgrid with predetermined load profile peaking at $50\,\mathrm{A}$.

The magnitude is bounded approximately by $|\Delta V_{\text{DC}}| \approx R_{\text{MG}} \, |\Delta I|$, where R_{MG} is the system resistance and ΔI is the current change between consecutive indices.

When the system resistance R_{MG} is changing between modeling steps, the resulting modeling artifact can be reduced by evaluating the equivalent system resistance as the average of two consecutive indices:

$$R_{\text{MG,avg}} = \frac{1}{2}\big(R_{\text{MG}}(n) + R_{\text{MG}}(n-1)\big), \tag{3.9}$$

which better approximates the continuous slope of the combined droop characteristic.

This modification would moderate the apparent voltage deviation at changing circumstances e.g. during voltage source startup; however, the path-dependence itself originates from the $n \; - \; n{-}1$ current imbalance and therefore cannot be entirely removed within the present discrete quasi-static formulation. The modeling in this work does not employ the averaging but rather uses $R_{\text{MG}}(n)$ only.

3.1.4 Modeling Verification in Source Exceptional Situations, Nonuniform Parameterization and Positive Loads

Traditionally in shipboard power systems, power generating units like combustion engine-based generator sets and inverter-based shaft-generator systems, are always parameterized uniformly. This means that their references and droop coefficients are the same, such that they share the loads proportionally equally. This work mirrors that design aspect in the models and equal DC voltage references and droop coefficients are used for each voltage source.

But this status quo can be broken, e.g. when an equipment producing current hits its limit, and the source no longer at that point works as voltage source but rather just a current source with fixed current reference. Or if it is stopped due to a fault, it creates then different system level scenario.

Also, an everyday scenario of a voltage source starting creates nonuniformity

in the source parameterization. This is due to the voltage reference initially being at level corresponding with the DC-link voltage and then ramping up to the intended value. Starting the voltage source with DC voltage reference corresponding with actual DC-link voltage enables starting the source with zero load.

In modern shipboard power systems, equipment working as positive loads, e.g. solar power and energy storage systems, are becoming more popular every day. The power management strategy, the power system it is used in, and the modeling paradigm must be able to handle positive loads as well, in order to cope with and prepare for possible backpower situations. To this end this section covers modeling with positive load reference as well.

3.1.4.1 Source Exceptional Situations

Exceptional source-side situations, such as current limitation or source shutdown, require appropriate handling when calculating R_{MG} and $\mathbf{I}_{\text{S}}$. Limiting a source's current to a predefined maximum can be achieved by constraining its corresponding Γ value. For example, if the Γ value of source 3, as presented in earlier models, is limited to 0.6, its output cannot exceed $0.6 \cdot I_{\text{nomS}_3} = 18\,\text{A}$. See Figure 21.

When a source is output-limited, it no longer behaves as a droop-controlled voltage source but instead operates as a current source. As a result, its droop resistance effectively becomes infinite and is excluded from the system resistance calculation.

This shift to current source behavior affects the overall system resistance R_{MG} used in the DC voltage calculation in Equation (3.6). This operational change affects also the circulating currents, as will be discussed in Section 3.1.5.1. Opposite operational changes are introduced for loads through under- and overvoltage regulators in Sections 3.1.7 and 3.1.8, where loads transition to voltage sources.

Similarly, source shutdown limits the source's output current to zero and also implies an infinite droop resistance. The two scenarios, current limitation and source stopping, are prudent to keep separate even when their effect

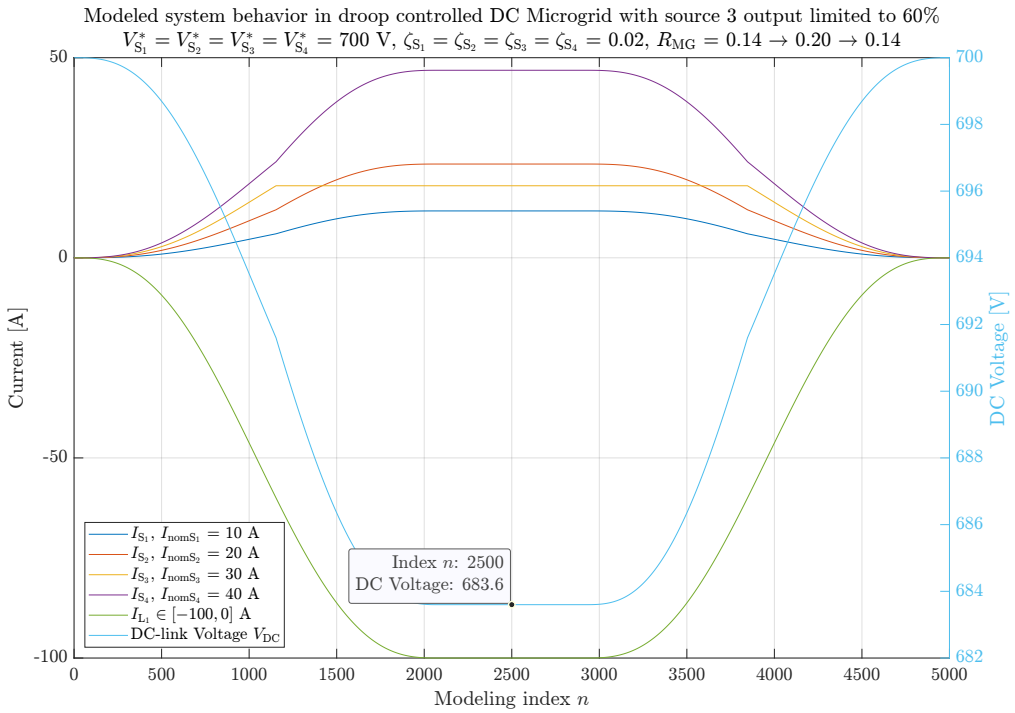


Figure 21. DC microgrid simulation with voltage source 3 output current limited to 60% by constraining $\Gamma_{S_3} \in]-\infty, 0.6]$.

in this simple modeling is similar. With separated statuses for limitation and stopping, and later model development including inverter modulation, it is easier to distinguish between running at zero current and full stop.

Rather than explicitly assigning an infinite value to the limited source's droop resistance R_{DS}, a clearer and more numerically robust approach is to define a multiplier that sets the inverse droop resistance—used in the system resistance calculation in Equation (3.8)—to zero when the source is limited or stopped.

A practical mathematical mechanism for the multiplier in modeling current limitations and source shutdowns involves the use of Boolean arrays, where each element represents the operational state of a corresponding voltage source. These arrays are defined as $\mathbf{B}_{VS}$ for voltage source operation and $\mathbf{B}_{runS}$ for running status:

$$\begin{aligned} \mathbf{B}_{\mathrm{VS}} &= \left[\mathrm{B}_{\mathrm{VS}_1}, \mathrm{B}_{\mathrm{VS}_2}, \ldots, \mathrm{B}_{\mathrm{VS}_N}\right]^{\mathrm{T}}, \\ \text{where } \mathrm{B}_{\mathrm{VS}_i} &= \left(I_{\mathrm{S}_i} > I_{\mathrm{minS}_i}\right) \wedge \left(I_{\mathrm{S}_i} < I_{\mathrm{maxS}_i}\right) \\ &= \left(\Gamma_{\mathrm{S}_i} > \frac{I_{\mathrm{minS}_i}}{I_{\mathrm{nomS}_i}}\right) \wedge \left(\Gamma_{\mathrm{S}_i} < \frac{I_{\mathrm{maxS}_i}}{I_{\mathrm{nomS}_i}}\right), \end{aligned} \tag{3.10}$$

with I_{minS_i} and I_{maxS_i} denoting the specified minimum and maximum allowable output currents for Source i.

$$\begin{aligned} \mathbf{B}_{\mathrm{runS}} &= \left[\mathrm{B}_{\mathrm{runS}_1}, \mathrm{B}_{\mathrm{runS}_2}, \ldots, \mathrm{B}_{\mathrm{runS}_N}\right]^{\mathrm{T}}, \\ \text{where } \mathrm{B}_{\mathrm{runS}_i} &= \begin{cases} 1, & \text{if Source } i \text{ is considered running}, \\ 0, & \text{otherwise}. \end{cases} \end{aligned} \tag{3.11}$$

These Boolean arrays are used throughout the modeling framework to selectively disable or include sources in matrix-based calculations depending on their state. The $\mathbf{B}_{\mathrm{VS}}$ array is typically updated at each modeling step based on the computed source current.

The Boolean array for running status, $\mathbf{B}_{\mathrm{runS}}$, can be defined based on the modeled system behavior—for example, based on user input or simulated fault conditions. Figure 22 presents this behavior by introducing a multiplier into the $\mathbf{I}_{\mathrm{S}}$ equation, which is set to 1 when the source is operational and 0 otherwise as described later in this section.

These Boolean arrays can be used in the system resistance calculation in Equation (3.8) by applying the logical AND operator between the operational state and running status of each voltage source:

$$R_{\mathrm{MG}} = \left(\sum_{i=1}^{N} \frac{\mathrm{B}_{\mathrm{VS}_i}(n) \wedge \mathrm{B}_{\mathrm{runS_i}}(n)}{R_{\mathrm{DS}_i}(n)} \right)^{-1}. \tag{3.12}$$

The logical AND returns 1 only when both the voltage source's operational and running states are true, resulting in a non-zero conductance for that source in the R_{MG} calculation. If either value is 0, the conductance becomes zero—which corresponds to an infinite droop resistance—and the source is

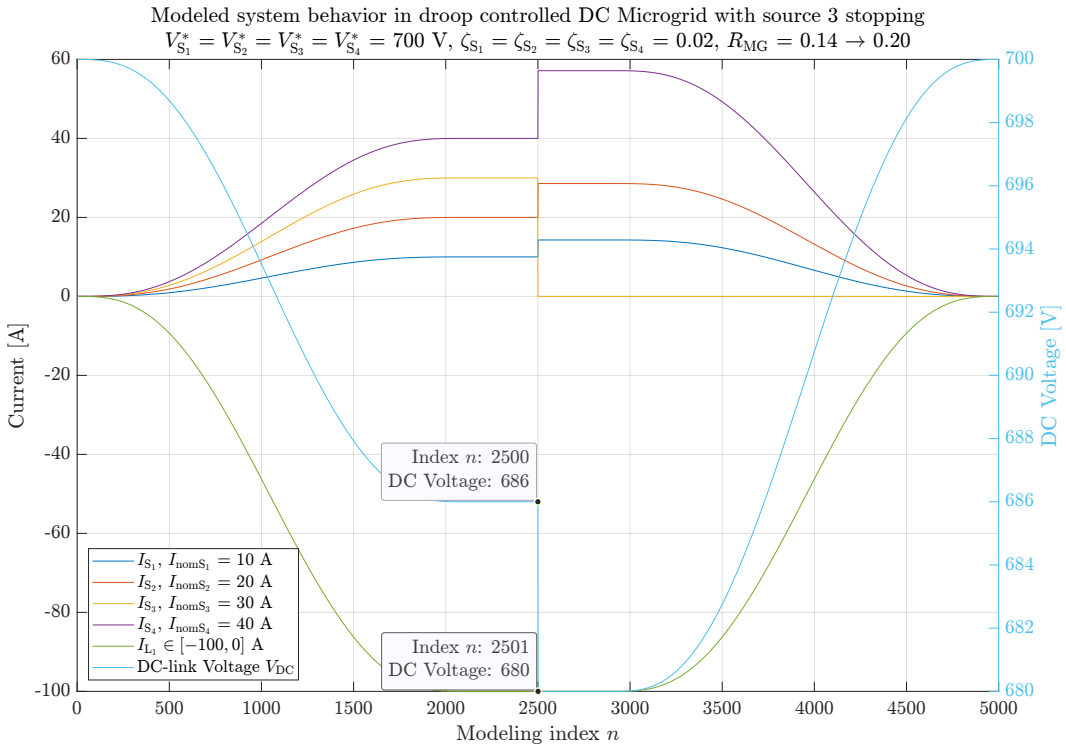


Figure 22. DC microgrid simulation with voltage source 3 suddenly stopping.

effectively excluded from the system resistance.

During modeling, the values of $\mathbf{B}_{VS}$ can be computed concurrently with Γ_S values, as they can be directly derived from them according to Equation (3.10):

$$\mathbf{B}_{VS} = \begin{bmatrix} \left(\Gamma_{S_1} > \frac{I_{minS_1}}{I_{nomS_1}}\right) \wedge \left(\Gamma_{S_1} < \frac{I_{maxS_1}}{I_{nomS_1}}\right) \\ \left(\Gamma_{S_2} > \frac{I_{minS_2}}{I_{nomS_2}}\right) \wedge \left(\Gamma_{S_2} < \frac{I_{maxS_2}}{I_{nomS_2}}\right) \\ \vdots \\ \left(\Gamma_{S_n} > \frac{I_{minS_n}}{I_{nomS_n}}\right) \wedge \left(\Gamma_{S_n} < \frac{I_{maxS_n}}{I_{nomS_n}}\right) \end{bmatrix}. \tag{3.13}$$

The effect of stopped sources is incorporated to the modeling by applying an element-wise product (Hadamard product) with the run status array $\mathbf{B}_{runS}$ when calculating the source current array $\mathbf{I}_S$, yielding the final array:

$$\mathbf{I}_S = (\Gamma_S \cdot \mathbf{I}_{nomS}) \circ \mathbf{B}_{runS}. \tag{3.14}$$

In MATLAB, this operation corresponds to the $.*$ operator. The use of a compact $N \times 1$ Boolean array avoids the need for a full $N \times N$ diagonal matrix, while preserving extensibility for future modeling needs.

Code 5 presents the MATLAB code for running modeling with mechanisms for limiting and shutting down specific sources. The code utilizes the $\mathbf{B}_{\mathrm{VS}}$ and $\mathbf{B}_{\mathrm{runS}}$ arrays, as well as arrays for positive and negative source output

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [10; 20; 30; 40];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('B2_ModelingWithDCLinkBooleanArrays:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
sourceCurrentArrays = zeros(numberOfSources,length(steps));
B_VS = ones(numberOfSources, 1);
B_runS = ones(numberOfSources, 1);
posLims = ones(numberOfSources, 1) * inf;
negLims = ones(numberOfSources, 1) * -inf;

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources, 1);
droopConduc = zeros(numberOfSources, 1);
R_MG = 0;

%% Create smootherstep-based load profile varying between -100 and 0 A
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
s = 6*x.^5 - 15*x.^4 + 10*x.^3;
load = -100 * s;

%% Run through the calculations
for n=steps
    %% DC voltage handling
    % Calculate R_MG
    for r=1:numberOfSources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = (B_VS(r) && B_runS(r))/droopResis(r);
    end
    R_MG = 1/sum(droopConduc);

    % Calculate the DC voltage
    if n==1 % If first step use default values
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, load(n), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), ...
            sum(sourceCurrentArrays(:,n-1)), load(n), R_MG);
    end

    %% Source handling
    % B_runS(3) = n < 2500; %<- Stop source 3 at modeling index 2500
    % posLims(3) = 0.6; %<- Limit source 3 positive output to 60%

    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        GammaMatrix(j,j) = min(max(Gamma(DCVolt(n), DCRefs(j), DCDroops(j)), negLims(j)), posLims(j));
        B_VS(j) = GammaMatrix(j,j) > negLims(j) && GammaMatrix(j,j) < posLims(j);
    end

    % Calculate the source currents
    sourceCurrentArrays(:,n) = (GammaMatrix * InomS) .* B_runS;
end
```

Code 5. MATLAB code with mechanisms implemented for source limitation and shutdown.

limits to control the source output limitation. Source stopping can be done simply by assigning its corresponding value in $\mathbf{B}_{\text{runS}}$ to zero.

To verify the accuracy of the modeling, the expected DC-link voltage can be analytically calculated for each scenario and compared to the simulated values. At peak load, the system is subjected to a 100 A load, which matches the sum of the nominal source currents (10 A + 20 A + 30 A + 40 A = 100 A). Under normal conditions, given that all sources have identical voltage references and droop coefficients, each source operates at its rated current, resulting in a 2% voltage drop.

In the first exceptional case, Source 3—with a nominal current of 30 A—is limited to 60% of its output capacity. This effectively turns it from a droop-controlled voltage source into a current source delivering a fixed 18 A. Subtracting this from the total load leaves 82 A to be supplied by the remaining sources. As the remaining sources collectively have a nominal capacity of 70 A, they must operate at $\frac{82}{70} \approx 1.17$ times their rated current. With a baseline 2% voltage drop at rated current, the new voltage drop becomes $\frac{82}{70} \cdot 2\% = \frac{164}{70}\% \approx 2.34\%$. This corresponds to a DC-link voltage of $700\,\text{V} \cdot (1 - \frac{164}{7000}) = 683.6\,\text{V}$, which matches the simulated result shown in Figure 21.

In the second case, Source 3 is completely stopped, contributing no current. The entire 100 A load must now be supplied by the remaining sources, which again have a combined nominal capacity of 70 A. This results in a current overload factor of $\frac{100}{70} = \frac{10}{7} \approx 1.43$, leading to a voltage drop of $\frac{10}{7} \cdot 2\% = \frac{20}{7}\% \approx 2.86\%$. The resulting DC-link voltage becomes $700\,\text{V} \cdot (1 - \frac{20}{700}) = 680\,\text{V}$, which aligns precisely with the model output shown in Figure 22.

The latter scenario shown in Figure 22 clearly demonstrates the quasi-static nature of the selected DC-link voltage modeling method. Both the voltage and current values shift instantaneously to their new steady-state levels, regardless of how dramatic the associated transients would have been in a real-life. If the modeling were focused on capturing fast phenomena—such as current spikes induced by a short-circuit during a sudden source disconnection—then the time-dependent dynamics would certainly need to be included. At minimum, such a model would require incorporating the inverter capacitances into the algorithm.

However, in this work the developed modeling paradigm is specifically used

in power management strategy, whose scope—both in this study and in real-life applications—does not extend to such fast phenomena. In the scenario shown in Figure 22, the role of the power management system is to coordinate power generation and consumption in such a way that the loss of one source does not trigger a cascading control failure that trips the remaining sources and causes a blackout. From this perspective, the quasi-static nature of the selected modeling method is entirely appropriate and sufficient for the intended purpose.

3.1.4.2 Source Nonuniform Parameterization

While during stable shipboard power system operation the sources' DC voltage reference and droop coefficient parameters would generally all be set uniformly, there are multiple scenarios where they are intentionally different. These scenarios range from energy storage state-of-charge balancing methods covered in Sections 4.3 and 4.4 to voltage source starting covered in Section 5.2.2.

For model verification purposes, let's have similar model as previously, but with Source 2 having DC voltage reference of 705 V and Source 3 DC droop coefficient of 2.5%, as seen in Code 6. The modeling results are presented in Figure 23.

Independent verification for the modeling accuracy in this scenario is more complicated process than in previous section due to the "in-fighting" between the sources. The sources are at the same time feeding the load but also feeding current between each other. In order to verify the accuracy of the model, the DC voltage values must be found at different loads, to which all sources concurrently agree/converge to.

```
%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [10; 20; 30; 40];             % Source nominal currents
DCRefs = [700; 705; 700; 700];        % Source DC voltage references
DCDroops = [0.02; 0.02; 0.025; 0.02]; % Source DC droop coefficients
```

Code 6. Source parameters in MATLAB code for modeling scenario with nonuniform source parameters.

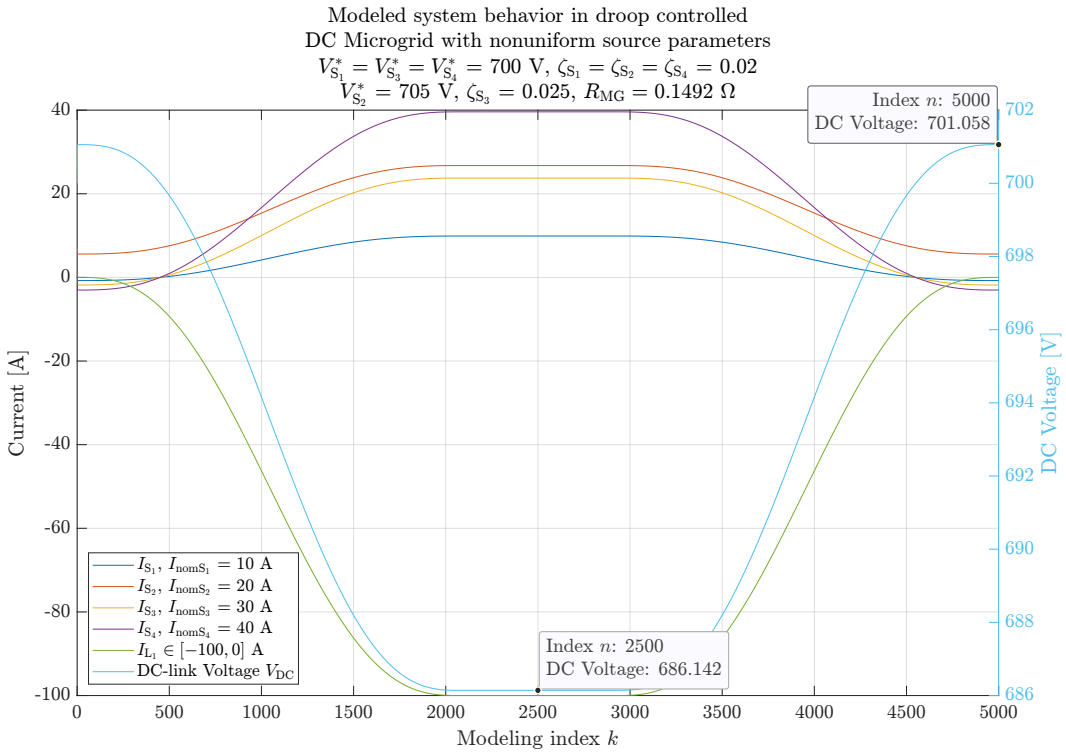


Figure 23. Modeling results of system with nonuniform source parameterization.

As shown in Equation (3.7) the droop resistance ties the DC voltage reference and droop coefficient together. In no-load situation the DC voltage must be exactly the same as equivalent system DC voltage reference, which can be calculated utilizing the droop resistances as droop-weighted average of the source voltage references:

$$V^*_{eq} = \frac{\sum \frac{V^*_S}{R_{DS}}}{\sum \frac{1}{R_{DS}}} = \frac{\sum G_{DS} V^*_S}{\sum G_{DS}}. \tag{3.15}$$

Calculating this gives $V^*_{eq} = 701.0579$ V, which is also the voltage at zero load, just as the model predicts in Figure 23.

By reapplying Equation (3.7) together with the equivalent voltage reference V^*_{eq} and the sum of the source nominal currents $\sum \boldsymbol{I}_{nomS}$, the equivalent droop coefficient of the system can be calculated as:

$$\zeta_{eq} = \frac{R_{MG} \sum \boldsymbol{I}_{nomS}}{V^*_{eq}}, \tag{3.16}$$

where R_{MG} is the system resistance as defined in Section 3.1.3.1. Substituting the values yields $\zeta_{\mathrm{eq}} = 0.0213$.

To calculate the expected DC voltage under a rated load of 100 A, the following expression is used:

$$V_{\mathrm{DC}} = V_{\mathrm{eq}}^{*}\,(1 - \zeta_{\mathrm{eq}}) = 686.1418\ \mathrm{V}, \tag{3.17}$$

which matches the modeling result shown in Figure 23.

3.1.4.3 Positive Loads

For the defined modeling methodology and the math utilized, the direction of the current is of no consequence. For testing purposes, the load can be defined to be varying between -100 and 100 A with Code 7. The modeled currents and DC voltage behave exactly the same way as before, just with opposite signs. From Figure 24 it can be seen how the voltage drops 2% at rated loading in motoring side, and then rises 2% in the generating side at rated load.

This confirms that the modeling paradigm correctly handles bidirectional power flow without requiring separate case handling or sign adjustments in the logic. Furthermore, nonlinearities or direction-dependent behaviors can also be implemented with minimal modifications if needed. This is particularly relevant for modeling equipment such as bidirectional converters, batteries, or regenerative drives, where the control strategy or current limits may depend on the direction of power flow.

```
%% Create smootherstep-based load profile varying between -100 and 100 A
x = 2*(steps)./max(steps);
x(x > 1) = 2 - x(x > 1);
s = 6*x.^5 - 15*x.^4 + 10*x.^3;
load = -200 * s + 100;
```

Code 7. MATLAB code for generating load profile varying between -100 and 100 A.

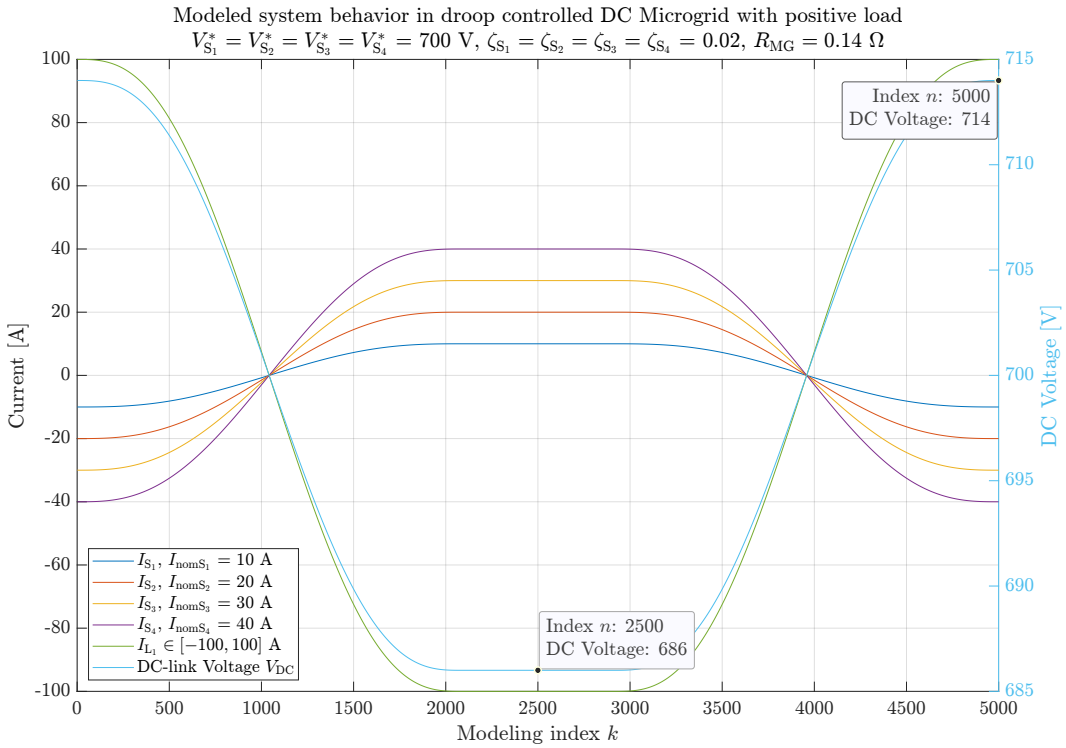


Figure 24. Modeling results of system with load sweeping also in positive side.

3.1.5 Voltage Source Circulating Currents from Measurement Offsets

Circulating currents as a phenomenon refer to currents that are not reaching the loads, i.e., the currents are wasted by flowing from voltage source to voltage source instead of contributing to maintaining the DC voltage. This is a result of the voltage sources' internal DC voltage measurement errors or imperfections in their operation. This results in the sources trying to generate a different voltage than supposed to, and they start "fighting" each other. This is not to be confused with current flow behavior resulting from voltage reference or drooping coefficient differences between the voltage sources, as the phenomenon they produce is source unbalancing, and is covered in Section 3.2 and utilized in Sections 4.3 and 4.4.

To keep the mathematical architecture in this work clear and modular, the expanding modeling capabilities are not embedded directly into the existing mathematical core definitions. This means that e.g. the existing Γ function is not treated as an ever-expanding entity, as that would soon lead to an un-

readable and unmaintainable formulation. Instead, and much like in software architecture, new features are added in parallel with the existing mathematical structures. This does not create computationally the most efficient math, but mathematics where unnecessary features can easily be excluded when not required.

Following this architectural principle, the circulating currents are modeled as an offset to voltage source Γ_{S} functions, let's call this offset Γ_{C}. When Γ_{C} is incorporated to the Equation (3.3) we get,

$$I_{\mathrm{CS}} = (\Gamma_{\mathrm{S}} + \Gamma_{\mathrm{C}}) I_{\mathrm{nomS}}, \tag{3.18}$$

where the source current including the circulating current element is called I_{CS}.

The function Γ_{C} must be defined separately for two current types: (1) the load-supplying currents affected by voltage measurement errors, denoted as $I_{\mathrm{CS}_1}, I_{\mathrm{CS}_2}, I_{\mathrm{CS}_3}, \ldots, I_{\mathrm{CS}_N}$, and (2) the circulating currents induced between voltage sources due to measurement discrepancies, such as $I_{\mathrm{CS}_1\mathrm{S}_2}$, $I_{\mathrm{CS}_1\mathrm{S}_3}$, $I_{\mathrm{CS}_1\mathrm{S}_4}, \ldots$. This separation is essential because voltage measurement errors directly affect a source's output both to the load and to other sources. For instance, if a voltage source overestimates the bus voltage by 1 V, it will supply proportionally less current to a consuming load. Likewise, a 1 V discrepancy between two sources' measured voltages will cause the one with the lower reading to inject current into the other, resulting in unintended circulating currents.

Equations (3.19) and (3.20) show how the Γ_{C} is calculated for the currents supplying the load and currents circulating between the voltage sources, Γ_{CS_a} and $\Gamma_{\mathrm{CS}_a\mathrm{S}_b}$, respectively. Variable subscripts a and b denote indexes for the voltage sources: $\mathrm{S}_1, \mathrm{S}_2, \mathrm{S}_3 \ldots \mathrm{S}_N$. For example, $\Gamma_{\mathrm{CS}_1\mathrm{S}_2}$ is the multiplier for the current flowing from source S_1 to source S_2, while $\Gamma_{\mathrm{CS}_4\mathrm{S}_3}$ applies to the current from source S_4 to source S_3. The term V_{S_x} refers to the sensed DC voltage of voltage source S_x, which includes its individual voltage measurement error. Due to the proportional nature of the Γ variables, the $\Gamma_{\mathrm{CS}_a\mathrm{S}_b}$ must be scaled according to the voltage source's normalized nominal current. Codes 8 and 9 provide the MATLAB code implementations for the Γ_{CS_a}

and $\Gamma_{\mathrm{CS}_a\mathrm{S}_b}$ equations respectively.

$$\Gamma_{\mathrm{CS}_a} = \frac{V_{\mathrm{DC}} - V_{\mathrm{S}_a}}{\zeta_{\mathrm{S}_a} V^*_{\mathrm{S}_a}} \tag{3.19}$$

$$\Gamma_{\mathrm{CS}_a\mathrm{S}_b} = \frac{V_{\mathrm{S}_b} - V_{\mathrm{S}_a}}{R_{\mathrm{DS}_a} + R_{\mathrm{DS}_b}} \cdot \frac{1}{I_{\mathrm{nomS}_b}} \tag{3.20}$$

P., Monica and M., Kowsalya and Guerrero, Josep M. (2021) employed an equation similar to Equation (3.20) to evaluate the quality of load sharing between voltage sources, using the resulting circulating current as a metric for assessing current-sharing accuracy. In the present work, a similar formulation is adopted, but with the explicit purpose of modeling the actual circulating current that arises between voltage sources sensing different DC-link voltages. In Equation (3.20), the computed current is scaled by the nominal current of the receiving source, to make it compatible with the principles of the Γ-based modeling framework. Droop resistances are used in place of droop coefficients to ensure a compact and elegant mathematical representation as using ζ_{S} values would result in more complex equation.

The order of the voltage subtraction in the Equation (3.19) numerator differ from that of the Γ_{S} function in Equation (3.2) due to a different perspective of the voltage difference. In the case of Γ_{S}, a lower actual voltage relative to the reference indicates that the voltage source must increase its current injection toward the DC-link. In contrast, under a voltage measurement error, if the source perceives the voltage to be higher than it actually is, it reduces its

```
%% Gamma_Ca Function
function output = Gamma_Ca(DCVoltage, MeasurementError, DCVoltageReference, ...
    DCDroopingCoefficient)

    if DCDroopingCoefficient == 0
        error('Gamma_Ca:ZeroDroop', ...
            'Zero droop not supported, leads to division by zero.');
    elseif DCVoltageReference == 0
        error('Gamma_Ca:ZeroReference', ...
            'Zero reference not supported, leads to division by zero.');
    else
        output = (DCVoltage - (DCVoltage + MeasurementError)) / ...
        (DCDroopingCoefficient * DCVoltageReference);
    end
end
```

Code 8. MATLAB function for calculating the measurement error's effect on the current supplying the load.

```
%% Gamma_Cab Function
function output = Gamma_Cab(DCVoltage, MeasErra, RDroopa, Inomb, MeasErrb, RDroopb)

    % Sensed voltages
    V_a = DCVoltage + MeasErra;
    V_b = DCVoltage + MeasErrb;

    % Sensed voltage mismatch
    deltaV = V_b - V_a;

    % Equivalent resistance between source a and b
    Reqab = RDroopa + RDroopb;

    % Circulating current from a to b (amperes)
    if Reqab == 0
        error('Gamma_Cab:ZeroR_eq', ...
            'Zero combined resistance not supported, leads to division by zero.');
    else
        I_ab = deltaV / Reqab;
    end

    % Return gamma coefficient such that I_ab = gamma * InomB
    if Inomb == 0
        error('Gamma_Cab:ZeroInom', ...
            'Zero nominal current not supported, leads to division by zero.');
    else
        output = I_ab * (1/Inomb);
    end
end
```

Code 9. MATLAB function for calculating the circulating current between two voltage sources.

current injection toward the DC-link compared to the non-erroneous case. In such situations, the resulting value of Γ_{CS_x} becomes negative.

Using Equations (3.19) and (3.20), we can define the $N \times N$ offset matrix $\mathbf{\Gamma}_{\mathrm{C}}$ as,

$$\mathbf{\Gamma}_{\mathrm{C}} = \begin{bmatrix} \Gamma_{\mathrm{CS}_1} & \Gamma_{\mathrm{CS}_1\mathrm{S}_2} & \dots & \Gamma_{\mathrm{CS}_1\mathrm{S}_N} \\ \Gamma_{\mathrm{CS}_2\mathrm{S}_1} & \Gamma_{\mathrm{CS}_2} & \dots & \Gamma_{\mathrm{CS}_2\mathrm{S}_N} \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_{\mathrm{CS}_N\mathrm{S}_1} & \Gamma_{\mathrm{CS}_N\mathrm{S}_2} & \dots & \Gamma_{\mathrm{CS}_N} \end{bmatrix}. \tag{3.21}$$

In the matrix above, the diagonal elements represent the measurement-error-induced coefficient offsets to the load currents. These are computed using Equation (3.19). The off-diagonal elements provide the coefficients for the circulating currents between the voltage sources, calculated using Equation (3.20).

To apply the $\mathbf{\Gamma}_{\mathrm{C}}$ matrix within the current calculation framework, the nominal current array must be expanded into a diagonal matrix, $\mathrm{diag}\,(\mathbf{I}_{\mathrm{nomS}})$,

to ensure dimensional compatibility. When source current limitations and shutdown conditions are not considered, the load-bearing and circulating currents in the system can be calculated as:

$$
\begin{aligned}
\mathbf{I}_{\mathrm{CS}} &= \left(\operatorname{diag}(\boldsymbol{\Gamma}_{\mathrm{S}}) + \boldsymbol{\Gamma}_{\mathrm{C}}\right) \cdot \operatorname{diag}\left(\mathbf{I}_{\mathrm{nomS}}\right) \\
&= \left(\begin{bmatrix} \Gamma_{\mathrm{S}_1} & 0 & \dots & 0 \\ 0 & \Gamma_{\mathrm{S}_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \Gamma_{\mathrm{S}_N} \end{bmatrix} + \begin{bmatrix} \Gamma_{\mathrm{CS}_1} & \Gamma_{\mathrm{CS}_1\mathrm{S}_2} & \dots & \Gamma_{\mathrm{CS}_1\mathrm{S}_N} \\ \Gamma_{\mathrm{CS}_2\mathrm{S}_1} & \Gamma_{\mathrm{CS}_2} & \dots & \Gamma_{\mathrm{CS}_2\mathrm{S}_N} \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_{\mathrm{CS}_N\mathrm{S}_1} & \Gamma_{\mathrm{CS}_N\mathrm{S}_2} & \dots & \Gamma_{\mathrm{CS}_N} \end{bmatrix} \right) \begin{bmatrix} I_{\mathrm{nomS}_1} & 0 & \dots & 0 \\ 0 & I_{\mathrm{nomS}_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I_{\mathrm{nomS}_N} \end{bmatrix} \\
&= \begin{bmatrix} \left(\Gamma_{\mathrm{S}_1} + \Gamma_{\mathrm{CS}_1}\right) I_{\mathrm{nomS}_1} & \Gamma_{\mathrm{CS}_1\mathrm{S}_2} I_{\mathrm{nomS}_2} & \dots & \Gamma_{\mathrm{CS}_1\mathrm{S}_N} I_{\mathrm{nomS}_N} \\ \Gamma_{\mathrm{CS}_2\mathrm{S}_1} I_{\mathrm{nomS}_1} & \left(\Gamma_{\mathrm{S}_2} + \Gamma_{\mathrm{CS}_2}\right) I_{\mathrm{nomS}_2} & \dots & \Gamma_{\mathrm{CS}_2\mathrm{S}_N} I_{\mathrm{nomS}_N} \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_{\mathrm{CS}_N\mathrm{S}_1} I_{\mathrm{nomS}_1} & \Gamma_{\mathrm{CS}_N\mathrm{S}_2} I_{\mathrm{nomS}_2} & \dots & \left(\Gamma_{\mathrm{S}_N} + \Gamma_{\mathrm{CS}_N}\right) I_{\mathrm{nomS}_N} \end{bmatrix} \\
&= \begin{bmatrix} I_{\mathrm{CS}_1} & I_{\mathrm{CS}_1\mathrm{S}_2} & \dots & I_{\mathrm{CS}_1\mathrm{S}_N} \\ I_{\mathrm{CS}_2\mathrm{S}_1} & I_{\mathrm{CS}_2} & \dots & I_{\mathrm{CS}_2\mathrm{S}_N} \\ \vdots & \vdots & \ddots & \vdots \\ I_{\mathrm{CS}_N\mathrm{S}_1} & I_{\mathrm{CS}_N\mathrm{S}_2} & \dots & I_{\mathrm{CS}_N} \end{bmatrix},
\end{aligned}
\tag{3.22}
$$

where $\mathbf{I}_{\mathrm{CS}}$ is an $N \times N$ matrix that includes all individual current contributions in the system, covering both load-supplying currents and circulating currents between voltage sources. From this matrix, the total output current array $\mathbf{I}_{\Sigma\mathrm{CS}}$ can be computed by summing each row:

$$
\mathbf{I}_{\Sigma\mathrm{CS}} = \mathbf{I}_{\mathrm{CS}} \cdot \mathbf{1} \in \mathbb{R}^{N \times 1}, \tag{3.23}
$$

where $\mathbf{I}_{\Sigma\mathrm{CS}}$ gives the total current provided by each voltage source. In a real system, these values correspond to the actual measured output currents at each source. Current limitation, as discussed later in Section 3.1.5.1, is based on this quantity.

The currents that affect the DC-link voltage are the load supplying currents $I_{\mathrm{CS}_1}, I_{\mathrm{CS}_2} \ldots I_{\mathrm{CS}_N}$ seen in Equation (3.22). This is due to the circulating currents flowing between any two sources cancel each other out, so their combined sum is zero, and so is their effect on the DC-link voltage. Thus the source current array $\mathbf{I}_{\mathrm{S}}$ used for DC-voltage modeling, must be constructed as

$$
\mathbf{I}_{\mathrm{S}} = \begin{bmatrix} I_{\mathrm{S}_1} \\ I_{\mathrm{S}_2} \\ \vdots \\ I_{\mathrm{S}_N} \end{bmatrix} = \begin{bmatrix} \left[\mathbf{I}_{\mathrm{CS}}\right]_{1,1} \\ \left[\mathbf{I}_{\mathrm{CS}}\right]_{2,2} \\ \vdots \\ \left[\mathbf{I}_{\mathrm{CS}}\right]_{N,N} \end{bmatrix}. \tag{3.24}
$$

To demonstrate this in a practical context, static DC voltage measurement errors from Table 2 are applied to a system with four voltage sources and a single load, similar to the configuration considered previously. The corresponding equivalent circuit is presented in Figure 25. For reference, the figure highlights the total current $I_{\Sigma CS_1}$ produced by source S_1, the circulating currents from source S_1 to the other sources ($I_{CS_1S_2}$, $I_{CS_1S_3}$, and $I_{CS_1S_4}$), and the current supplied to the load (I_{CS_1}). The internal DC voltage measurement associated with source S_1 is also indicated. The remaining voltage sources (S_2 to S_4) are structurally identical, each having the same internal measurement mechanism and current contributions.

Figure 26 shows the modeled total currents with a similar load profile as in Figure 18. In this model the the static voltage measurement errors are introduced to the voltage sources. The measurement errors cause offset to the system's equivalent DC voltage reference and the total currents of the sources. Code 10 presents the code to perform the modeling in MATLAB.

Verification of the modeling can be done with the system's equivalent DC voltage reference V_{eq}^* and drooping coefficient ζ_{eq}, following the same principles as in Section 3.1.4.2. The Equation (3.15) defines the V_{eq}^* to be calculated using the sources' voltage references V_S^* and droop resistances R_{DS}. The measurement errors can be taken into account by assigning them as offsets to the DC voltage references.

Table 2. Static DC voltage measurement errors used.

Voltage Source	Measurement Error
Source 1	$V_{errS_1} = +1.0\,V$
Source 2	$V_{errS_2} = +0.5\,V$
Source 3	$V_{errS_3} = -1.0\,V$
Source 4	$V_{errS_4} = \pm 0.0\,V$

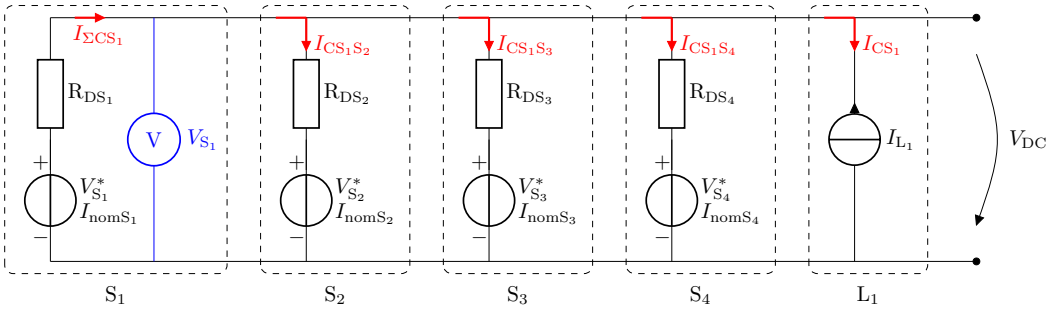


Figure 25. Modeled system with source S_1's currents and its sensed DC voltage value $V_{S_1} = V_{DC} + V_{errS_1}$.

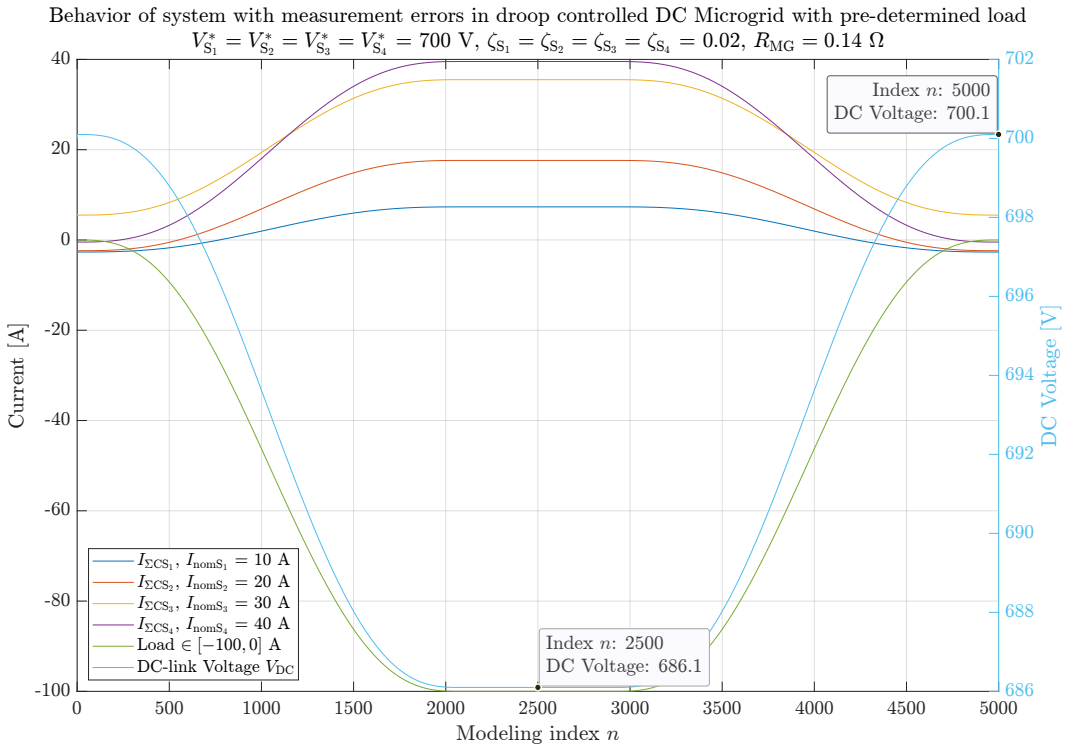


Figure 26. Load distribution of four voltage sources with circulating currents in DC microgrid with predetermined load.

Source's measurement error however, must not affect the source's droop resistance calculation, even though Equation (3.7) defines it is based on DC voltage reference. The droop resistance defines the slope of how steeply the source's drooped voltage reference V^*_{DS} is changing under load, as defined in Equation (1.1) and seen in Figure 2b. This slope is defined by the intended voltage reference V^*_S only, not by measurement error.

In this scenario the system equivalent voltage reference is calculated as

$$V^*_{eq} = \frac{\sum \frac{V^*_S - V_{errS}}{R_{DS}}}{\sum \frac{1}{R_{DS}}}, \tag{3.25}$$

where V_{errS} represents the sources' measurement errors as defined in Table 2. With this equation the $V^*_{eq} = 700.1$ V, which corresponds with the no-load DC-link voltage level at index 5000 in Figure 26.

As measurement error does not affect the droop resistance, it does not affect the sources' droop coefficients ζ_S either. The equivalent droop coefficient ζ_{eq}

```
%% Initialize variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [10; 20; 30; 40];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
SourceMeasErrors = [1; 0.5; -1; 0];  % Define equipment measurement errors
if ~isequal(length(InomS), length(DCRefs), length(DCDroops), length(SourceMeasErrors))
    error('C_ModelingSourceCircCurr:SourceParameterMismatch', 'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
GammaCMatrix = zeros(numberOfSources,numberOfSources);
SourceCurrentMatrices = zeros(numberOfSources,numberOfSources,length(steps));
SourceTotalCurrentArrays = zeros(numberOfSources,length(steps));
SourceLoadSupplyingCurrentArrays = zeros(numberOfSources,length(steps));

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources, 1);
droopConduc = zeros(numberOfSources, 1);
R_MG = 0;

%% Create smootherstep-based load profile varying between -100 and 0 A
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
s = 6*x.^5 - 15*x.^4 + 10*x.^3;
load = -100 * s;

%% Run through the calculations
for n=steps
    %% DC voltage handling
    % Calculate R_MG
    for r=1:numberOfSources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = 1/droopResis(r);
    end
    R_MG = 1/sum(droopConduc);

    % Calculate the DC voltage
    if n==1 % If first step use default DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, load(n), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), sum(SourceLoadSupplyingCurrentArrays(:,n-1)), load(n), R_MG);
    end

    %% Source handling
    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        GammaMatrix(j,j) = Gamma(DCVolt(n), DCRefs(j), DCDroops(j));
    end

    % Populate Gamma_C matrix
    for a = 1:numberOfSources
        for b = 1:numberOfSources
            if a==b
                % Diagonal
                GammaCMatrix(a,a) = [Gamma_Ca(DCVolt(n), SourceMeasErrors(a), DCRefs(a), DCDroops(a))];
            else
                % Off-diagonal (circulating current from source a to b)
                GammaCMatrix(a,b) = Gamma_Cab(DCVolt(n), SourceMeasErrors(a), droopResis(a), ...
                    InomS(b), SourceMeasErrors(b), droopResis(b));
            end
        end
    end

    % Calculate the source current matrix
    SourceCurrentMatrices(:,:,n) = (GammaMatrix + GammaCMatrix) * diag(InomS);

    % Calculate source total currents
    SourceTotalCurrentArrays(:,n) = SourceCurrentMatrices(:,:,n)*ones(numberOfSources,1);

    % Calculate load supplying currents
    SourceLoadSupplyingCurrentArrays(:,n) = diag(SourceCurrentMatrices(:,:,n));
end
```

Code 10. MATLAB code for modeling system of four voltage sources with measurement errors and predetermined load.

can be calculated with Equation (3.16) and it is 0.02. Calculating the voltage drop at sources' nominal load is done by $V_{\text{eq}}^{*}\,(1-\zeta_{\text{eq}}) = 686.1$ V. This corresponds with the voltage drop at index 2500 in Figure 26.

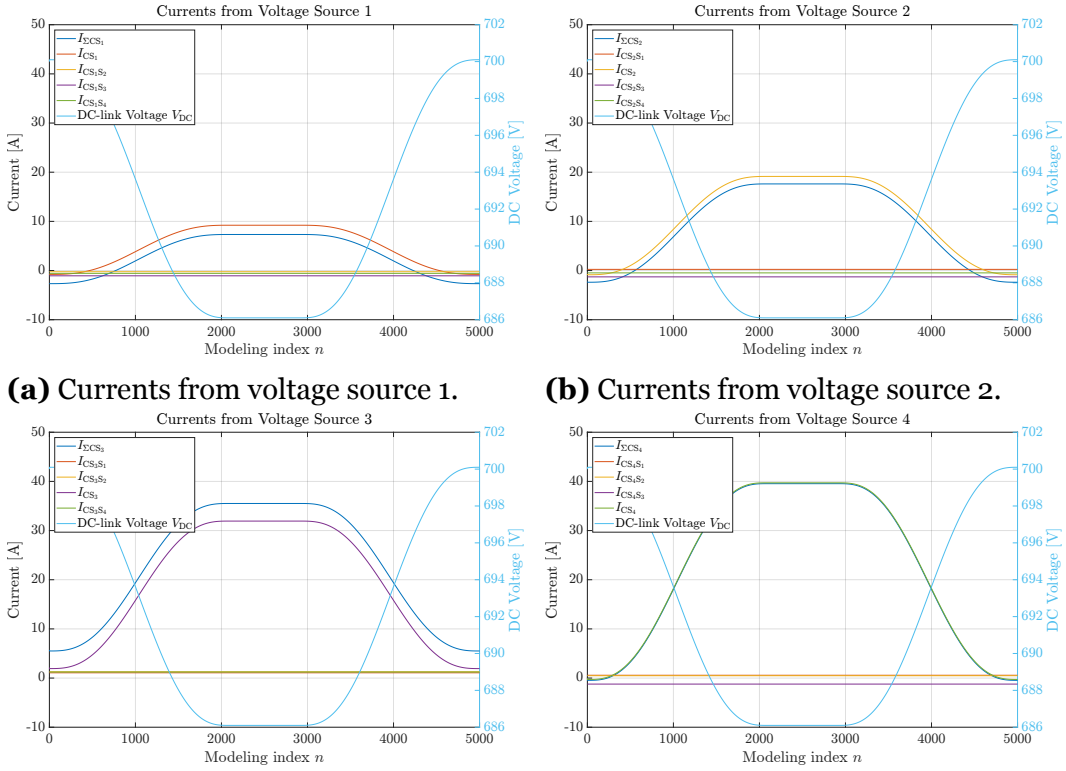
To verify that circulating currents indeed cancel each other out, and that the load-supplying currents accurately reflect the applied load, the components of the total current matrix and the load-supplying current array are summed at selected modeling steps. These sums are then compared against the known load to confirm current balance and validate Kirchhoff's Current Law compliance. Code 11 shows the code snippet to calculate the mentioned verification values after Code 10 is executed. At both selected modeling steps—modeling index 2500 corresponding to a 100 A load, and modeling index 5000 corresponding to a 0 A load—the summed current values matched the applied load in each case.

Each calculation point in this model of four voltage sources, returns a 4×4 size $\mathbf{I}_{\text{CS}}$ matrix containing all of the currents in the system associated with each of the voltage sources. From this matrix it is then possible deduct the total and the load supplying currents for each source. Figure 26 presents the total currents fed by the voltage sources, as well as the load and DC-link voltage. Figure 27 presents all of the currents associated with each individual voltage source. The numerical calculation results from modeling indexes 2500 and 5000 are shown in Table 3.

Due to the modeled measurement errors specified in Table 2, voltage source S_1 measures the highest DC voltage among all sources. Consequently, its proportional droop-based contribution to DC-link voltage support is the lowest. This behavior is also evident in the circulating current components presented in Table 3, where the entries $I_{\text{CS}_1\text{S}_2}$, $I_{\text{CS}_1\text{S}_3}$, and $I_{\text{CS}_1\text{S}_4}$ are all negative.

```
%% Calculation verification related variables
V_eq = sum((DCRefs-SourceMeasErrors)./droopResis)/sum(1./droopResis)
zeta_eq = (R_MG * sum(diag(InomS)))/V_eq
V_nomLoad = V_eq*(1-zeta_eq)
sum(sum(SourceCurrentMatrices(:,:,2500)))
sum(sum(SourceLoadSupplyingCurrentArrays(:,2500)))
sum(sum(SourceCurrentMatrices(:,:,5000)))
sum(sum(SourceLoadSupplyingCurrentArrays(:,5000)))
```

Code 11. MATLAB code snippet to calculate the source circulating current modeling verification related values.



(a) Currents from voltage source 1.

(b) Currents from voltage source 2.

(c) Currents from voltage source 3.

(d) Currents from voltage source 4.

Figure 27. Modeled currents including circulating currents, graphs (a), (b), (c), and (d) show currents associated with voltage sources 1, 2, 3, and 4, respectively.

Table 3. The modeled source currents at modeling indexes 2500 and 5000 in Figure 26.

Currents at Index 2500	**Currents at Index 5000**
$\mathbf{I}_{\Sigma CS} = \begin{bmatrix} 7.33\,\text{A} \\ 17.62\,\text{A} \\ 35.51\,\text{A} \\ 39.54\,\text{A} \end{bmatrix}$ $\mathbf{I}_{S} = \begin{bmatrix} 9.21\,\text{A} \\ 19.14\,\text{A} \\ 31.93\,\text{A} \\ 39.71\,\text{A} \end{bmatrix}$	$\mathbf{I}_{\Sigma CS} = \begin{bmatrix} -2.67\,\text{A} \\ -2.38\,\text{A} \\ 5.51\,\text{A} \\ -0.46\,\text{A} \end{bmatrix}$ $\mathbf{I}_{S} = \begin{bmatrix} -0.79\,\text{A} \\ -0.86\,\text{A} \\ 1.93\,\text{A} \\ -0.29\,\text{A} \end{bmatrix}$
$\mathbf{I}_{CS} = \begin{bmatrix} 9.21\,\text{A} & -0.24\,\text{A} & -1.07\,\text{A} & -0.57\,\text{A} \\ 0.24\,\text{A} & 19.14\,\text{A} & -1.29\,\text{A} & -0.48\,\text{A} \\ 1.07\,\text{A} & 1.29\,\text{A} & 31.93\,\text{A} & 1.22\,\text{A} \\ 0.57\,\text{A} & 0.48\,\text{A} & -1.22\,\text{A} & 39.71\,\text{A} \end{bmatrix}$	$\mathbf{I}_{CS} = \begin{bmatrix} -0.79\,\text{A} & -0.24\,\text{A} & -1.07\,\text{A} & -0.57\,\text{A} \\ 0.24\,\text{A} & -0.86\,\text{A} & -1.29\,\text{A} & -0.48\,\text{A} \\ 1.07\,\text{A} & 1.29\,\text{A} & 1.93\,\text{A} & 1.22\,\text{A} \\ 0.57\,\text{A} & 0.48\,\text{A} & -1.22\,\text{A} & -0.29\,\text{A} \end{bmatrix}$

These values indicate that current is flowing into S_1 from the direction of the other sources. This current direction reflects the relative overestimation of the DC voltage by S_1, which results in reduced current injection toward the DC-link in comparison to the other voltage sources.

The results in Table 3 illustrate, for example, that the load-supplying current of Source 1, I_{S_1}, is 9.21 A, while the total current at its terminal, including

circulating currents, $I_{\Sigma \mathrm{CS}_1}$, is only 7.33 A at modeling index 2500. Since the load-supplying current is the one that directly affects DC voltage, measurement errors that induce circulating currents can significantly misrepresent the true contribution of each source to the system.

These circulating currents also place disproportionate stress on the source experiencing the lowest sensed voltage—here, Source 3. Although its nominal current is 30 A, it must deliver a peak current of 35.51 A, resulting in an overload of more than 18%, despite the total load being 100 A, which exactly matches the sum of all source nominal currents. Of the 35.51 A provided by Source 3, only 31.93 A contributes to supplying the actual load; the remainder is lost to circulating currents.

Note that when modeling with circulating currents, in the exceptional situation of the voltage source suddenly stopping, it is not sufficient to just write Γ_{S_x} to zero, as this would not eliminate the corresponding circulating currents in $\mathbf{\Gamma}_{\mathrm{C}}$ as their existence is not dependent on the loading of the voltage source. The source stopping simulation requires either defining also the source's measurement error to be zero, on-the-fly manipulation of the equations by removing the source in question from the matrices, or considering it as a load with zero reference.

3.1.5.1 Circulating Currents With Current-Limited and Stopping Voltage Sources

Taking into account current-limited and stopped sources requires managing the corresponding values in the $\mathbf{I}_{\mathrm{CS}}$ matrix using the Boolean arrays $\mathbf{B}_{\mathrm{runS}}$ and $\mathbf{B}_{\mathrm{VS}}$.

When a voltage source becomes current-limited, it no longer behaves as a droop-controlled voltage source that adjusts its output in response to the measured DC voltage. Although the output current is originally derived from the voltage measurement, once the current limitation is active, the source's output current becomes fixed—effectively clamped—and no longer responds to voltage deviations. As a result, circulating currents involving that source are eliminated, since it ceases to participate in the voltage regulation.

A stopped source exhibits a similar decoupling from the DC voltage but differs operationally: its current output is forcibly set to zero, representing seizure of operation. In contrast, a current-limited source still contributes a fixed nonzero current. It is useful to distinguish these cases already at this stage of the modeling paradigm, as future developments outside the scope of this work—particularly those incorporating detailed inverter behavior such as modulation—will require separate treatment of these two operating conditions.

As discussed in Section 3.1.4.1, the Boolean array $\mathbf{B}_{\text{runS}}$ is assigned based on the modeled system behavior, for example, user commands or fault events. The actual current limitation and consequently the values for $\mathbf{B}_{\text{VS}}$ are now determined using the total current of each source, as calculated from Equation (3.23).

Outside of the system resistance calculation for DC voltage modeling, there are no mathematical interactions between sources in this modeling paradigm. While the off-diagonal elements in $\mathbf{I}_{\text{CS}}$ are computed based on the mutual droop resistances between source pairs (Equation (3.22)), zeroing the circulating currents for a limited or stopped source does not affect interactions between any other sources. Therefore, the Boolean arrays $\mathbf{B}_{\text{runS}}$ and $\mathbf{B}_{\text{VS}}$ can be used to directly zero the corresponding rows and columns of the affected source in $\mathbf{I}_{\text{CS}}$ to nullify its circulating currents.

The procedure for incorporating limited and stopped sources in the presence of circulating currents closely follows the approach described in Section 3.1.4.1, though some differences arise due to the handling of a current matrix instead of a simple array. Initially, the complete current matrix which includes the circulating currents is computed as per Equation (3.22) and stored in the auxiliary variable $\mathbf{I}_{\text{origCS}}$. From this, the total current array $\mathbf{I}_{\text{orig}\Sigma\text{CS}}$ is obtained by summing the rows of $\mathbf{I}_{\text{origCS}}$ according to Equation (3.23).

The Boolean array $\mathbf{B}_{\text{VS}}$ is determined from the total current array $\mathbf{I}_{\text{orig}\Sigma\text{CS}}$ as:

$$\mathbf{B}_{\text{VS}} = \begin{bmatrix} (I_{\text{orig}\Sigma\text{CS}_1} > I_{\text{minS}_1}) \wedge (I_{\text{orig}\Sigma\text{CS}_1} < I_{\text{maxS}_1}) \\ (I_{\text{orig}\Sigma\text{CS}_2} > I_{\text{minS}_2}) \wedge (I_{\text{orig}\Sigma\text{CS}_2} < I_{\text{maxS}_2}) \\ \vdots \\ (I_{\text{orig}\Sigma\text{CS}_N} > I_{\text{minS}_N}) \wedge (I_{\text{orig}\Sigma\text{CS}_N} < I_{\text{maxS}_N}) \end{bmatrix} \tag{3.26}$$

Using the Boolean arrays $\mathbf{B}_{\text{runS}}$ and $\mathbf{B}_{\text{VS}}$, the current matrix without the circulating currents of limited or stopped sources can be derived as:

$$\mathbf{I}_{\text{noCirCurrCS}} = \left(\text{diag}\left(\mathbf{B}_{\text{runS}} \wedge \mathbf{B}_{\text{VS}}\right)\right) \mathbf{I}_{\text{origCS}} \left(\text{diag}\left(\mathbf{B}_{\text{runS}} \wedge \mathbf{B}_{\text{VS}}\right)\right), \tag{3.27}$$

where both rows and columns corresponding to limited or stopped sources are zeroed.

Finally, the diagonal elements of the current matrix $\mathbf{I}_{\text{CS}}$ are assigned based on whether the corresponding source is limited but still running:

$$I_{\text{CS}_{i,j}} = \begin{cases} \min\left(\max\left(I_{\text{origΣCS}_{i,j}}, I_{\text{minS}_i}\right), I_{\text{maxS}_i}\right), & \text{if } i = j \wedge (\neg \text{B}_{\text{VS}_i} \wedge \text{B}_{\text{runS}_i}) \\ I_{\text{noCirCurrCS}_{i,j}}, & \text{otherwise,} \end{cases} \tag{3.28}$$

where i,$j = 1, \ldots, N$.

The droop resistances of limited and stopped sources are excluded from the system resistance R_{MG} calculation in Equation (3.12) and, by extension, from the DC voltage modeling in Equation (3.6). This approach ensures that these sources remain included in the current matrix while appropriately affecting the system resistance and voltage calculations.

To demonstrate the behavior of a limited and stopped source, MATLAB code for a system with four voltage sources and one load is provided in Appendix 2. In this model, Source 3 is limited to 60% of its nominal output (i.e., 18 A) and is configured to stop entirely at modeling index $n = 2500$. Figure 28 shows the total source currents, the load current, and the DC voltage. Figure 29 presents all current components per source, including individual circulating currents.

The modeling results reveal an unstable region, commonly referred to as chattering, occurring when Source 3 is transitioning from a voltage source to a current source around modeling index $n = 1000$. This phenomenon is expected; it arises from the interplay of voltage measurement errors and the circulating currents they induce. Because the current limit constrains the total current of Source 3, denoted $I_{\Sigma\text{CS}_3}$, but not all of its total output contributes to load feeding or DC voltage regulation. This discrepancy between the total current and the amount of voltage regulating current causes distur-

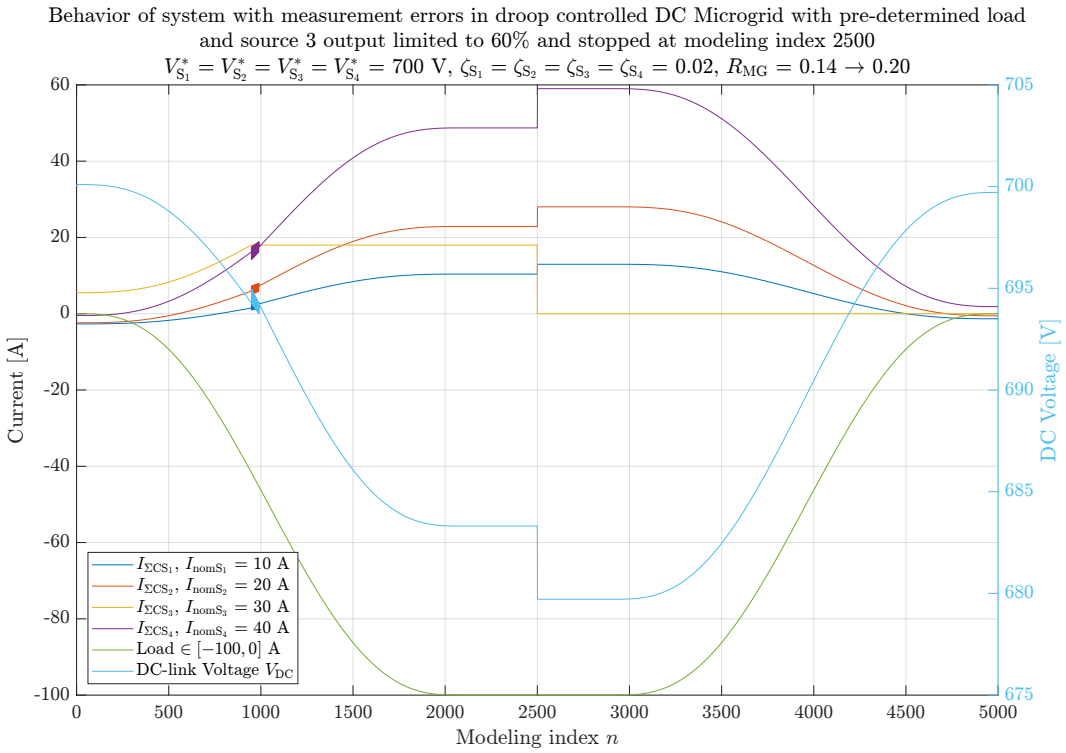


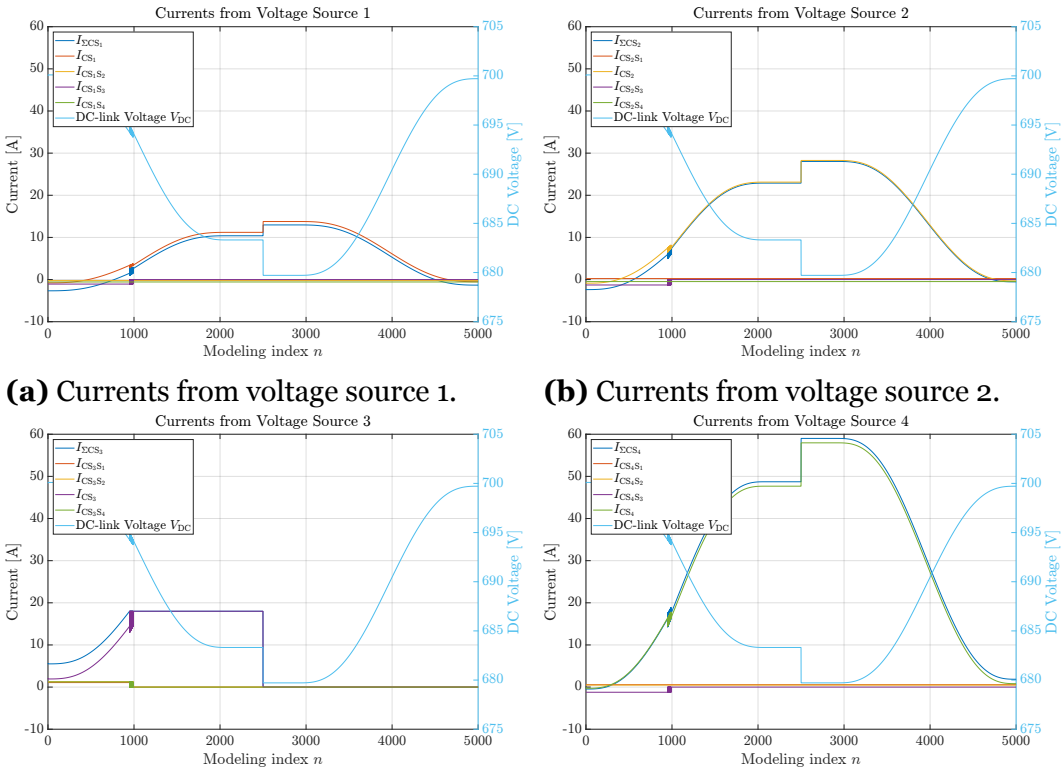
Figure 28. Load distribution of four voltage sources with circulating currents in DC microgrid with predetermined load and Source 3 output limited to 60% and stopped at modeling index 2500.

bances in the system. This same phenomenon would happen also when the source is transitioning back from a current source to a voltage source.

Table 4 presents the system currents at modeling index $n = 949$, which is the first step where the current limiter for Source 3 activates. The table compares the modeled case with the limiter, and a case what would happen if the limiter would not be activated. In the data, the total current $I_{\Sigma CS_3}$ is marked in blue, the load-feeding/voltage regulating current I_{S_3} in green, and the fixed current output of the current-limited source in red.

Table 4. The modeled source currents at modeling index $n = 949$.

With limitation	**Without limitation**
$\mathbf{I}_{\Sigma CS} = \begin{bmatrix} 2.58\,\text{A} \\ 7.25\,\text{A} \\ 18.00\,\text{A} \\ 17.44\,\text{A} \end{bmatrix}$ $\mathbf{I}_S = \begin{bmatrix} 3.38\,\text{A} \\ 7.48\,\text{A} \\ 18.00\,\text{A} \\ 16.40\,\text{A} \end{bmatrix}$	$\mathbf{I}_{\Sigma CS} = \begin{bmatrix} 1.50\,\text{A} \\ 5.96\,\text{A} \\ 18.02\,\text{A} \\ 16.22\,\text{A} \end{bmatrix}$ $\mathbf{I}_S = \begin{bmatrix} 3.38\,\text{A} \\ 7.48\,\text{A} \\ 14.44\,\text{A} \\ 16.40\,\text{A} \end{bmatrix}$
$\mathbf{I}_{CS} = \begin{bmatrix} 3.38\,\text{A} & -0.24\,\text{A} & 0.00\,\text{A} & -0.57\,\text{A} \\ 0.24\,\text{A} & 7.48\,\text{A} & 0.00\,\text{A} & -0.48\,\text{A} \\ 0.00\,\text{A} & 0.00\,\text{A} & 18.00\,\text{A} & 0.00\,\text{A} \\ 0.57\,\text{A} & 0.48\,\text{A} & 0.00\,\text{A} & 16.40\,\text{A} \end{bmatrix}$	$\mathbf{I}_{CS} = \begin{bmatrix} 3.38\,\text{A} & -0.24\,\text{A} & -1.07\,\text{A} & -0.57\,\text{A} \\ 0.24\,\text{A} & 7.48\,\text{A} & -1.29\,\text{A} & -0.48\,\text{A} \\ 1.07\,\text{A} & 1.29\,\text{A} & 14.44\,\text{A} & 1.22\,\text{A} \\ 0.57\,\text{A} & 0.48\,\text{A} & -1.22\,\text{A} & 16.40\,\text{A} \end{bmatrix}$



(a) Currents from voltage source 1.

(b) Currents from voltage source 2.

(c) Currents from voltage source 3.

(d) Currents from voltage source 4.

Figure 29. Modeled currents including circulating currents, graphs (a), (b), (c), and (d) show currents associated with voltage sources 1, 2, 3, and 4, respectively when Source 3 output is limited to 60% and stopped at modeling index 2500.

If Source 3 were still operating as a voltage source, it would contribute 14.44 A to support the load and regulate the DC voltage. However, once its current limiter activates, the source instead delivers a fixed 18.00 A, behaving as a current source. This change effectively removes 14.44 A of droop-based voltage regulation from the system, while simultaneously reducing the net system load by 18.00 A. As a result, the DC voltage rises in the following modeling step, as shown in Figure 30.

The increased voltage reduces the current required from the droop-controlled sources, including Source 3, causing its calculated output to fall below the 18.00 A threshold. This deactivates the limiter, and the source resumes operation as a voltage source. As a result, the system load increases by the same 18.00 A, leading to a voltage drop. In the next modeling step, this drop causes Source 3's output to again reach its current limit, reactivating the limiter.

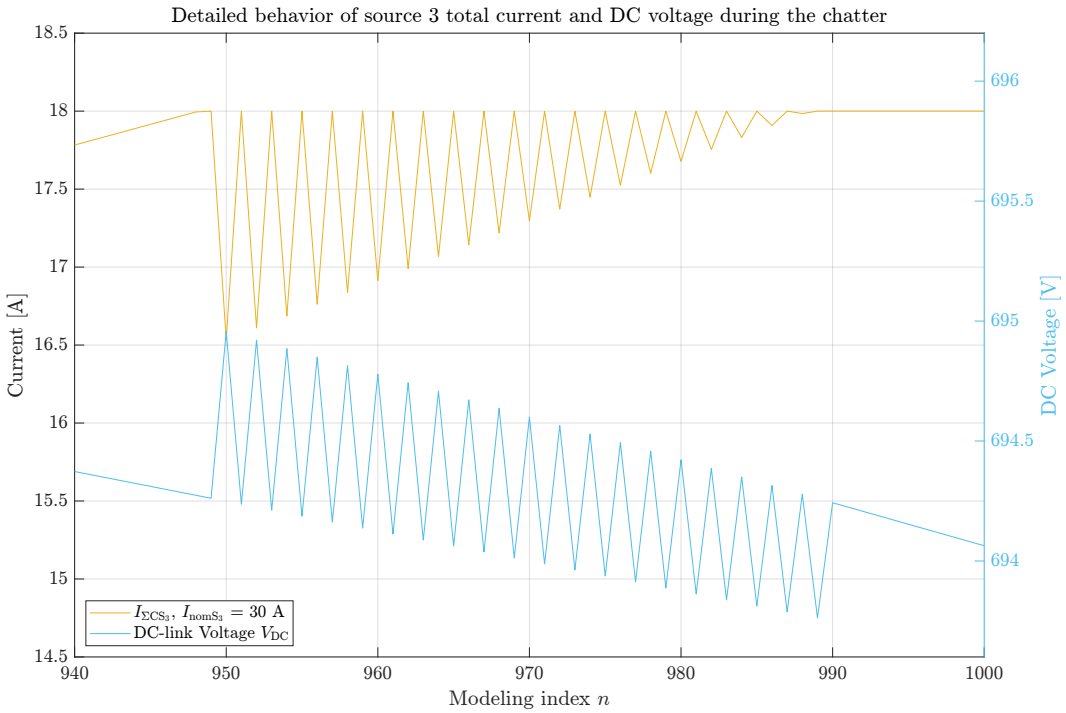


Figure 30. Detailed figure of the chattering during source 3 transitioning to a current source.

This alternating behavior results in chattering, which persists until the load becomes sufficiently large for the limiter to remain active without inducing a voltage rise that would otherwise deactivate it. In this idealized quasi-static model, where no time-dependent components are included, the transitions occur at every modeling step. Therefore, the chattering manifests as index-by-index alternation, limited solely by the discrete resolution of the quasi-static model—effectively instantaneous in the absence of time dependence.

3.1.6 Incorporating Loads in to the Modeling Paradigm

Earlier, the load was treated solely as an input argument $\mathbf{I}_{\mathrm{L}}$ in the DC-link voltage calculation in Equation (3.6). This approach is sufficient for systems comprising simple loads that do not actively participate in DC-link voltage regulation—even when the total load includes multiple components or behaves as a current source, such as in the case of photovoltaic inverters. However, this modeling approach becomes inadequate when loads are capable of altering their behavior and functioning as voltage sources to support the

DC-link voltage under certain conditions. To accommodate such functionality, the load models must be integrated more explicitly into the system's mathematical framework.

In Equation (3.3), the current of each voltage source is computed using a coefficient Γ, which reflects its proportional contribution to the load demand based on its own parameters and the DC-link voltage. To apply a similar modeling principle to the loads, a corresponding coefficient is required. However, since load behavior is externally defined and not directly governed by the system, the Γ-equivalent for loads is modeled as a proportional current reference, denoted by K:

$$I_{\mathrm{L}} = \mathrm{K}_{\mathrm{L}} I_{\mathrm{nomL}} \Rightarrow \mathbf{I}_{\mathrm{L}} = \mathrm{diag}(\mathbf{K}_{\mathrm{L}})\mathbf{I}_{\mathrm{nomL}}, \tag{3.29}$$

where $\mathbf{I}_{\mathrm{nomL}}$ is the nominal current array of the loads and $\mathbf{K}_{\mathrm{L}}$ array of the load current references.

The behavior of K depends on the operational characteristics of the loads, which may include, for example, motor drives operating under power control or batteries under voltage control. In real-world systems, K can also be derived from empirical data, which may be further analyzed and incorporated into simulation models. This modeling approach is demonstrated in Section 5.4.1 of this dissertation. Ultimately, K represents an application-specific reference that is transformed into a current reference by the inverter's control algorithms.

The previously presented mathematical modeling example, shown in Figure 18 with a predetermined load profile, is now extended by introducing a parameterized load representation. In the earlier approach, the load profile was directly inserted into the array $\mathbf{I}_{\mathrm{L}}$, and thus into Equation (3.6). In the revised formulation, the load profile is encoded into the proportional current reference $\mathrm{K}_{\mathrm{L}_1}$ behavior, which is multiplied by the load's nominal current, here defined as $I_{\mathrm{nomL}_1} = 100\,\mathrm{A}$. Figure 31 presents a simplified single-line di-

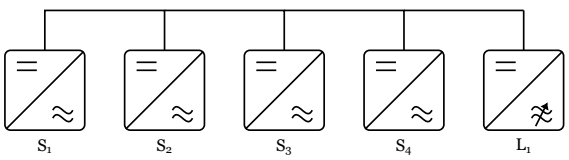


Figure 31. Modeled system showing inverter-based adjustable load.

agram of the modeled system, and Figure 32 shows the result of the system modeling using the new load representation.

Code 12 presents the MATLAB code for the modeling presented in Figure 32. In the code the modeling sequence shown in Figure 11 is clearly visible with distinctive sections for load, DC voltage and source handling.

The DC-link voltage V_{DC} and the source current array $\mathbf{I}_{\mathrm{S}}$ are considered the primary outputs of the DC microgrid model. The load currents $\mathbf{I}_{\mathrm{L}}$ are also included as part of the model output, particularly when the load under- and overvoltage regulators—tied to the operation of the variable voltage-based DC microgrid control strategy—are implemented, as described later in Section 3.1.7. In contrast, the array $\mathbf{K}_{\mathrm{L}}$ serves as the model's input interface, used to define externally modeled load scenarios and to analyze how different load behaviors influence the DC microgrid.

Figure 33 presents the reference signal flow in the modeling paradigm with two example input types: current references I_{L}^{*} and power references P_{L}^{*}. Additional forms may be derived as needed. The references are first nor-

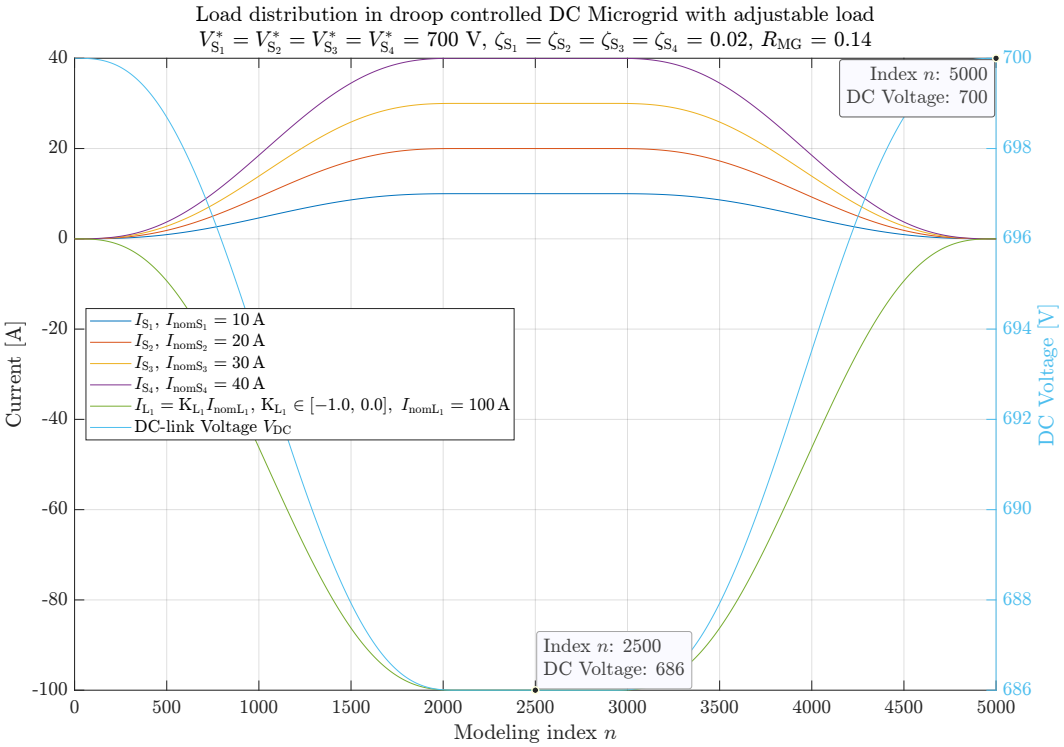


Figure 32. Load distribution of four voltage sources in DC microgrid with varying load.

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [10; 20; 30; 40];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('D1_ModelingLoad:SourceParameterMismatch', 'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
sourceCurrentArrays = zeros(numberOfSources,length(steps));

%% Initialize load modeling variables
% Order of the loads is as follows: [L1, L2...]
InomL = [100]; % Load nominal currents
numberOfLoads = length(InomL);
KappaMatrix = zeros(numberOfLoads,numberOfLoads);
loadCurrentArrays = zeros(numberOfLoads,length(steps));
loadProfiles = zeros(numberOfLoads,length(steps));

%% Create smootherstep-based load current reference profile for L1 varying between -1.0 and 0.0 A
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
KL1 = -(6*x.^5 - 15*x.^4 + 10*x.^3);
loadProfiles(1,:) = KL1;

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources, 1);
droopConduc = zeros(numberOfSources, 1);
R_MG = 0;

%% Run through the calculations
for n=steps
    %% Load handling
    % Populate the Kappa-matrix with new values according to
    % the load profiles
    for i=1:numberOfLoads
        KappaMatrix(i,i) = loadProfiles(i,n);
    end
    % Calculate the load currents
    loadCurrentArrays(:,n) = KappaMatrix * InomL;

    %% DC voltage handling
    % Calculate R_MG
    for r=1:numberOfSources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = 1/droopResis(r);
    end
    R_MG = 1/sum(droopConduc);

    % Calculate the DC voltage
    if n==1 % If first step use default DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, sum(loadCurrentArrays(:,n)), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), sum(sourceCurrentArrays(:,n-1)), sum(loadCurrentArrays(:,n)), R_MG);
    end

    %% Source handling
    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        GammaMatrix(j,j) = Gamma(DCVolt(n), DCRefs(j), DCDroops(j));
    end
    % Calculate the source currents
    sourceCurrentArrays(:,n) = GammaMatrix * InomS;
end
```

Code 12. MATLAB code for modeling DC microgrid behavior with a parameterized load.

malized into a proportional current reference K_L^*, which is not yet bounded in any way. It represents the proportional current the load would need to consume or produce in order to fulfill the given reference. The resulting K_L^* is then processed through the under- and overvoltage regulator logic to yield

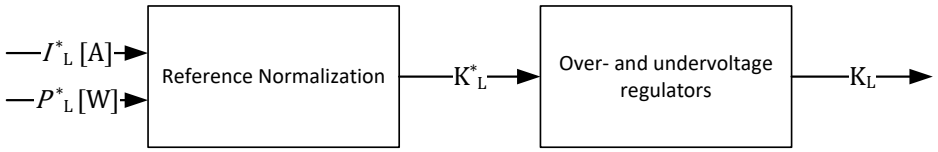


Figure 33. Signal flow of the reference chain, illustrating how the final load reference K_L is derived from input quantities.

the final value K_L. The regulator functions will be detailed in Section 3.1.7.

For a load that supplies or absorbs a specified amount of current to or from the DC microgrid, its current reference is denoted by I^*_L, and the reference chain—without considering voltage regulation—is defined as:

$$K_L = K^*_L = \frac{I^*_L}{I_{nomL}}. \tag{3.30}$$

For a load instead defined by a DC power reference P^*_L, the current reference becomes:

$$I^*_L = \frac{P^*_L}{V_{DC}}, \tag{3.31}$$

and the corresponding proportional reference is:

$$K_L = K^*_L = \frac{I^*_L}{I_{nomL}} = \frac{P^*_L}{V_{DC} I_{nomL}}. \tag{3.32}$$

This formulation serves as a foundation for incorporating more advanced load models into the system framework. In particular, the next section demonstrates how the final normalized load reference K_L can be constrained by regulator-defined bounds Γ^{UV}_L and Γ^{OV}_L, which depend on the droop-compensated voltage thresholds and the actual DC-link voltage. These bounds override the unconstrained normalized current reference K^*_L, thereby implementing the under- and overvoltage regulation behavior.

3.1.7 Including Load Under- and Overvoltage Regulators

At this stage, the model supports DC microgrid voltage sources with circulating currents and possible operational constraints, supplying adjustable loads that may also have their own limitations. As previously discussed, accurate modeling requires that loads be capable of switching to voltage-support mode when needed. To enable this behavior, the load model must include under- and overvoltage regulation. These regulators activate when the DC-link voltage reaches defined thresholds, allowing the loads to function similarly to voltage sources to help stabilize the system.

The under- and overvoltage regulators described here should not be confused with the inverter's under- and overvoltage protection functions. Protection functions are safety mechanisms that stop the inverter if the DC-link voltage exceeds critical thresholds—either too high to risk damage or too low for continued operation. These protections are always active and are not subject to application-specific configuration.

In contrast, the under- and overvoltage regulators are process-oriented functionalities that may be optionally enabled to allow the inverter to assist in stabilizing the DC-link voltage. Their activation is not mandatory, and in certain applications, their use may even be undesirable. For instance, in motor drive applications such as ship propulsion, the overvoltage regulator is typically disabled. If enabled, it would command the motor to increase power and speed to absorb excess energy and limit the voltage rise—an effect that could be inappropriate or unsafe in operational contexts like maritime navigation.

The ability to limit propulsion power is common, as propulsion systems are typically the largest consumers in shipboard power systems and have the greatest impact on overall power demand. If a situation arises where, for example, a power-generating unit is suddenly disconnected and the ship is at risk of a blackout, it is acceptable—and often necessary—to reduce propulsion power, including that of the main propulsion, since preserving power availability is more critical than maintaining full propulsion capability. The specific prioritization of which equipment is unloaded first is part of blackout prevention planning which is covered in Section 5.6 of this work.

The exact activation thresholds, deactivation behavior, and control principles of under- and overvoltage regulators vary across inverter manufacturers. This work presents a representative implementation of such regulators. Similar to the DC-link modeling approach discussed in Section 3.1.3, the regulators' behavior could serve as a valuable topic for future research and refinement.

The core operational principle of both under- and overvoltage regulators is that when the DC-link voltage reaches a predefined activation limit, the corresponding regulator becomes active. Once activated, the load begins to assist in stabilizing the DC-link voltage, treating the activation threshold as its voltage reference.

In the case of the undervoltage regulator, the load inverter reduces its power consumption and may even attempt to generate power, if its configuration permits. Conversely, when the overvoltage regulator is activated, the inverter increases its power consumption. In motor drive applications, this can result in an increase in motor speed, serving as a means to absorb excess energy from the DC-link.

A load running against its under- or overvoltage regulator maintaining the DC-link voltage typically runs also in parallel with other voltage maintaining equipment. To this end, the regulators presented here have drooping functionality built in to enable the parallel operation. Just like with normal voltage sources, which adjust their references according to their loading, the loads adjust their regulator activation limits as they are loaded.

To follow the principles of the presented modeling paradigm—in which a voltage source's current contribution to DC voltage maintenance is represented using Γ functions—the influence of the under- and overvoltage regulators on the load inverter can be modeled by imposing Γ-based bounds on the load's reference value K. This constraint is expressed as

$$K_L = \begin{cases} \Gamma_L^{OV}, & \text{if } K_L^* \geq \Gamma_L^{OV} \\ K_L^*, & \text{if } \Gamma_L^{UV} < K_L^* < \Gamma_L^{OV} \\ \Gamma_L^{UV}, & \text{if } K_L^* \leq \Gamma_L^{UV} \end{cases} \tag{3.33}$$

where K_L is the final bounded load reference, and K_L^* is the unconstrained

proportional current reference defined earlier in Figure 33. The regulators' coefficients $\Gamma_{\mathrm{L}}^{\mathrm{UV}}$ and $\Gamma_{\mathrm{L}}^{\mathrm{OV}}$ define the allowable range for $\mathrm{K_L}$ based on the activation limits of the under- and overvoltage regulation logic.

Analogous to how the Γ coefficient governs a voltage source's proportional current injection based on the sensed voltage's deviation from the voltage reference, the regulator Γs define the permitted range of the load inverter's behavior. When the DC-link voltage reaches one of the regulator activation thresholds, the load inverter's reference $\mathrm{K_L}$ is constrained by the corresponding Γ limit, effectively "overriding" its normal operation.

The equations for $\Gamma_{\mathrm{L}}^{\mathrm{UV}}$ and $\Gamma_{\mathrm{L}}^{\mathrm{OV}}$ can be expressed as:

$$\Gamma_{\mathrm{L}}^{\mathrm{UV}} = \frac{V_{\mathrm{L}}^{\mathrm{UV}*} - V_{\mathrm{DC}}}{\zeta_{\mathrm{L}}^{\mathrm{UV}} V_{\mathrm{L}}^{\mathrm{UV}*}} \tag{3.34}$$

and

$$\Gamma_{\mathrm{L}}^{\mathrm{OV}} = \frac{V_{\mathrm{L}}^{\mathrm{OV}*} - V_{\mathrm{DC}}}{\zeta_{\mathrm{L}}^{\mathrm{OV}} V_{\mathrm{L}}^{\mathrm{OV}*}} \tag{3.35}$$

where $V_{\mathrm{L}}^{\mathrm{UV}*}$ and $V_{\mathrm{L}}^{\mathrm{OV}*}$ are the voltage references at which the respective regulators activate, and $\zeta_{\mathrm{L}}^{\mathrm{UV}}$ and $\zeta_{\mathrm{L}}^{\mathrm{OV}}$ are the corresponding droop coefficients.

A notable drawback of this droop-based regulation is that variation in the DC voltage inherently modifies the regulator's Γ value. As a result, the bounds defined in Equation (3.33) for the load reference $\mathrm{K_L}$ are directly affected by V_{DC}, narrowing or shifting the operational window. This behavior is illustrated in Figure 34a under a varying load scenario.

As illustrated in the example shown in Figure 34a, drooping behavior causes the regulator activation limits to continuously shift according to the changing DC voltage. This effect introduces the drawback that the under- or overvoltage regulators may activate even when the actual DC-link voltage is still far from the originally defined threshold values.

In a single-load system, compensating for DC voltage deviations by adjusting the inverter's regulator activation limits during commissioning is relatively straightforward, as the voltage deviation is directly attributable to the behavior of that load. However, in systems with multiple loads, the DC-link voltage deviation results from the combined behavior of all loads. As a consequence, it is no longer feasible to embed compensation in the fixed

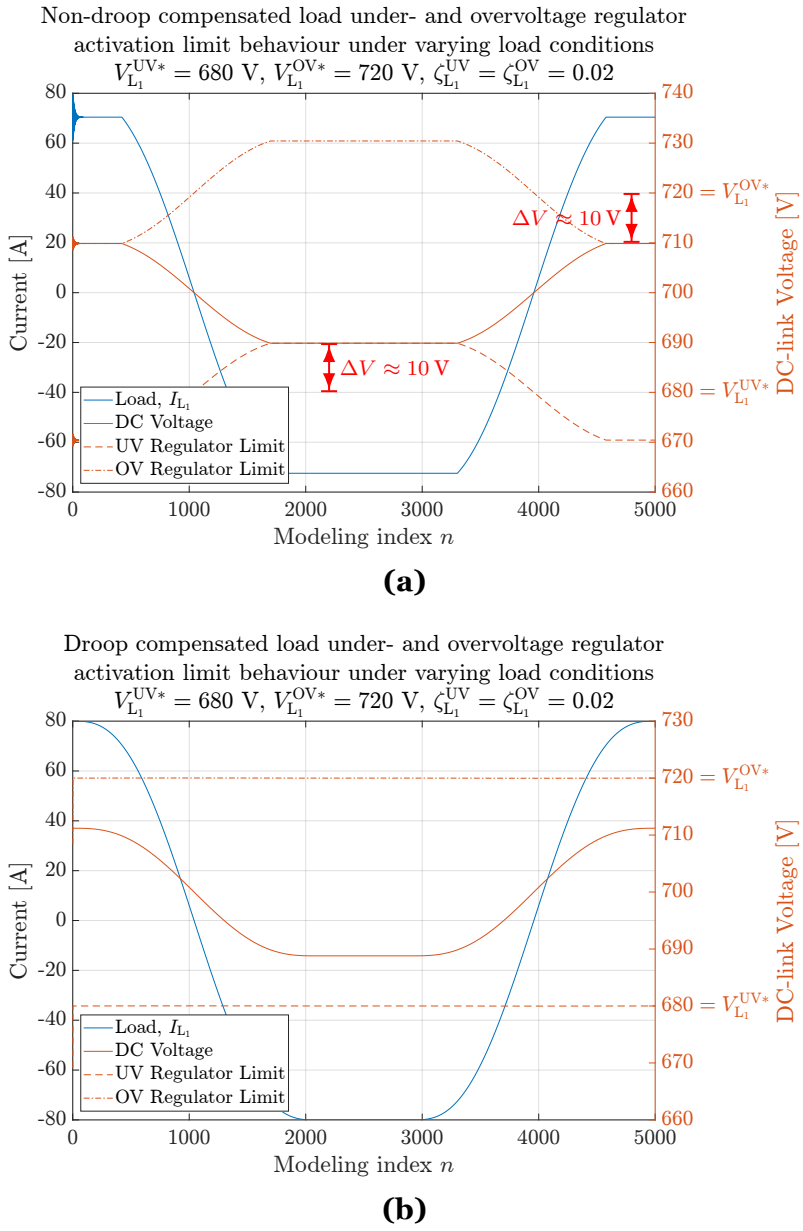


Figure 34. Demonstration of non-droop compensated (a) and droop compensated (b) load under- and overvoltage regulator activation limit behavior under varying load conditions.

parameters of any individual inverter.

To mitigate this issue, a selective droop compensation mechanism is implemented. This compensation ensures that any individual load's regulators activate only when the DC-link voltage genuinely reaches its intended activation limit, thereby restoring consistency in regulator behavior. The corrected regulator response is illustrated in Figure 34b.

Importantly, droop compensation is applied only when a regulator is not active. If compensation were applied while a regulator is active, the regulator would effectively suppress all DC-link voltage deviations, thereby eliminating the prioritization effect between loads with different activation thresholds.

As defined in Equation (3.33), regulator activation is controlled by limiting $\mathrm{K_L}$ between $\Gamma_{\mathrm{L}}^{\mathrm{UV}}$ and $\Gamma_{\mathrm{L}}^{\mathrm{OV}}$. To prevent these limits from deviating due to the inverter's own loading effect, droop compensation must cancel out the voltage deviation caused by the inverter itself in the $\Gamma_{\mathrm{L}}^{\mathrm{UV}}$ and $\Gamma_{\mathrm{L}}^{\mathrm{OV}}$ equations.

By introducing a multiplier to counter the droop-induced voltage deviation observed in Equation (1.2) when the regulator is inactive, the conditionally droop-compensated regulator activation limits, $V_{\mathrm{L}}^{\mathrm{UV}*'}$ and $V_{\mathrm{L}}^{\mathrm{OV}*'}$, are defined as follows:

$$V_{\mathrm{L}}^{\mathrm{UV}*'} = \begin{cases} V_{\mathrm{L}}^{\mathrm{UV}*'}, & \text{if } V_{\mathrm{DC}} < V_{\mathrm{L}}^{\mathrm{UV}*'} \text{ or } \mathrm{K}_{\mathrm{L}}^{*} < \Gamma_{\mathrm{L}}^{\mathrm{UV}} \\ V_{\mathrm{L}}^{\mathrm{UV}*}\left(1 + \dfrac{I_{\mathrm{L}}}{I_{\mathrm{nomL}}}\zeta_{\mathrm{L}}^{\mathrm{UV}}\right), & \text{otherwise} \end{cases} \tag{3.36}$$

$$V_{\mathrm{L}}^{\mathrm{OV}*'} = \begin{cases} V_{\mathrm{L}}^{\mathrm{OV}*'}, & \text{if } V_{\mathrm{DC}} > V_{\mathrm{L}}^{\mathrm{OV}*'} \text{ or } \mathrm{K}_{\mathrm{L}}^{*} > \Gamma_{\mathrm{L}}^{\mathrm{OV}} \\ V_{\mathrm{L}}^{\mathrm{OV}*}\left(1 + \dfrac{I_{\mathrm{L}}}{I_{\mathrm{nomL}}}\zeta_{\mathrm{L}}^{\mathrm{OV}}\right), & \text{otherwise.} \end{cases} \tag{3.37}$$

After incorporating these conditionally droop-compensated limits into the under- and overvoltage regulators' Γ functions, the updated regulator coefficients can be expressed as

$$\Gamma_{\mathrm{L}}^{\mathrm{UV}} = \frac{V_{\mathrm{L}}^{\mathrm{UV}*'} - V_{\mathrm{DC}}}{\zeta_{\mathrm{L}}^{\mathrm{UV}} V_{\mathrm{L}}^{\mathrm{UV}*}} \tag{3.38}$$

and

$$\Gamma_{\mathrm{L}}^{\mathrm{OV}} = \frac{V_{\mathrm{L}}^{\mathrm{OV}*'} - V_{\mathrm{DC}}}{\zeta_{\mathrm{L}}^{\mathrm{OV}} V_{\mathrm{L}}^{\mathrm{OV}*}} \tag{3.39}$$

In these Γ equations, the DC-link voltage is compared to the conditionally droop-compensated activation limit, which determines the actual point of regulator activation. The resulting deviation is scaled by the product of the regulator's drooping coefficient and the originally specified activation limit—both defined by the user to reflect the desired scaling behavior.

With the equations above, the operational principles for the under- and overvoltage regulators are established. These regulators maintain stable DC-voltage-based activation limits under changing load conditions when inactive, and enable parallel operation with other voltage sources when active. Since their operation is governed by the coefficients $\Gamma_{\mathrm{L}}^{\mathrm{UV}}$ and $\Gamma_{\mathrm{L}}^{\mathrm{OV}}$, which act as bounds for $\mathrm{K_L}$, the activation and deactivation behavior is inherently embedded in the algorithm, and the limits correspond to the specified DC voltage levels.

The load current matrix calculation shown in Equation (3.29) can now be explicitly formulated to include the constraints imposed by the under- and overvoltage regulators:

$$\mathbf{I}_{\mathrm{L}} = \begin{bmatrix} \mathrm{K}_{\mathrm{L}_1} & 0 & \cdots & 0 \\ 0 & \mathrm{K}_{\mathrm{L}_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathrm{K}_{\mathrm{L}_n} \end{bmatrix} \begin{bmatrix} I_{\mathrm{nomL}_1} \\ I_{\mathrm{nomL}_2} \\ \vdots \\ I_{\mathrm{nomL}_n} \end{bmatrix},$$
$$\text{where } \mathrm{K}_{\mathrm{L}_i} = \begin{cases} \Gamma_{\mathrm{L}_i}^{\mathrm{OV}}, & \text{if } \mathrm{K}_{\mathrm{L}_i}^{*} \geq \Gamma_{\mathrm{L}_i}^{\mathrm{OV}} \\ \mathrm{K}_{\mathrm{L}_i}^{*}, & \text{if } \Gamma_{\mathrm{L}_i}^{\mathrm{UV}} < \mathrm{K}_{\mathrm{L}_i}^{*} < \Gamma_{\mathrm{L}_i}^{\mathrm{OV}} \\ \Gamma_{\mathrm{L}_i}^{\mathrm{UV}}, & \text{if } \mathrm{K}_{\mathrm{L}_i}^{*} \leq \Gamma_{\mathrm{L}_i}^{\mathrm{UV}}. \end{cases} \tag{3.40}$$

In this formulation, the normalized current reference $\mathrm{K}_{\mathrm{L}_i}^{*}$ must be defined according to the operating principle of each individual load inverter. If, for example, Load 2 operates under a power reference, then $\mathrm{K}_{\mathrm{L}_2}^{*}$ is defined according to Equation 3.32, and $\mathrm{K}_{\mathrm{L}_2}$ becomes:

$$\mathrm{K}_{\mathrm{L}_2} = \begin{cases} \Gamma_{\mathrm{L}_2}^{\mathrm{OV}}, & \text{if } \mathrm{K}_{\mathrm{L}_2}^{*} \geq \Gamma_{\mathrm{L}_2}^{\mathrm{OV}} \\ \mathrm{K}_{\mathrm{L}_2}^{*} = \dfrac{P_{\mathrm{L}}^{*}}{V_{\mathrm{DC}} I_{\mathrm{nomL}}}, & \text{if } \Gamma_{\mathrm{L}_2}^{\mathrm{UV}} < \mathrm{K}_{\mathrm{L}_2}^{*} < \Gamma_{\mathrm{L}_2}^{\mathrm{OV}} \\ \Gamma_{\mathrm{L}_2}^{\mathrm{UV}}, & \text{if } \mathrm{K}_{\mathrm{L}_2}^{*} \leq \Gamma_{\mathrm{L}_2}^{\mathrm{UV}}. \end{cases} \tag{3.41}$$

To incorporate the load's under- and overvoltage regulators into the previously presented system model of four voltage sources and a single current load—excluding circulating current effects for the time being—the following regulator parameters are applied: $V_{\mathrm{L}_1}^{\mathrm{UV}*} = 688.8$ V, $V_{\mathrm{L}_1}^{\mathrm{OV}*} = 710$ V, and $\zeta_{\mathrm{L}_1}^{\mathrm{UV}} = \zeta_{\mathrm{L}_1}^{\mathrm{OV}} = 0.02$. Where for demonstration purposes $V_{\mathrm{L}_1}^{\mathrm{UV}*} = 688.8$ V corre-

sponds with 80% voltage drop from the 2% nominal defined by the source droop coefficient, meaning the sources are loaded 80% when the load's under voltage regulator activates.

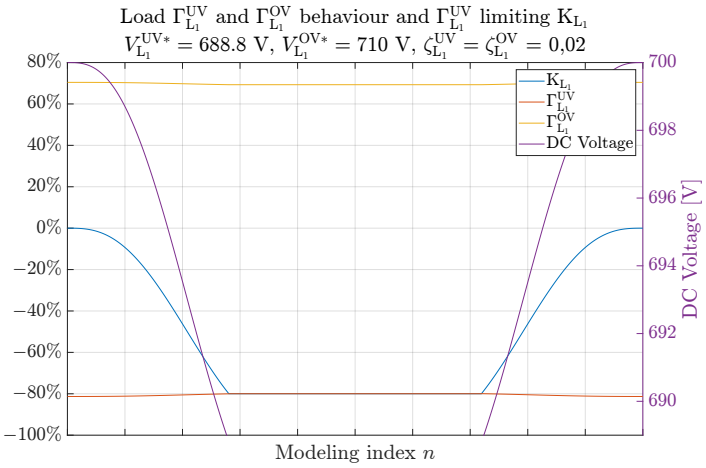
When loads operate against their under- or overvoltage regulators, they effectively act as voltage sources. In such cases, they also contribute to the system resistance R_{MG} used for DC-link voltage calculation. Accordingly, Equation (3.12) must be generalized to include both sources and regulator-engaged loads:

$$R_{\mathrm{MG}} = \left(\sum_{i=1}^{N} \frac{\mathrm{B}_{\mathrm{V}x_i}(n) \wedge \mathrm{B}_{\mathrm{run}x_i}(n)}{R_{\mathrm{D}x_i}(n)} \right)^{-1}, \tag{3.42}$$

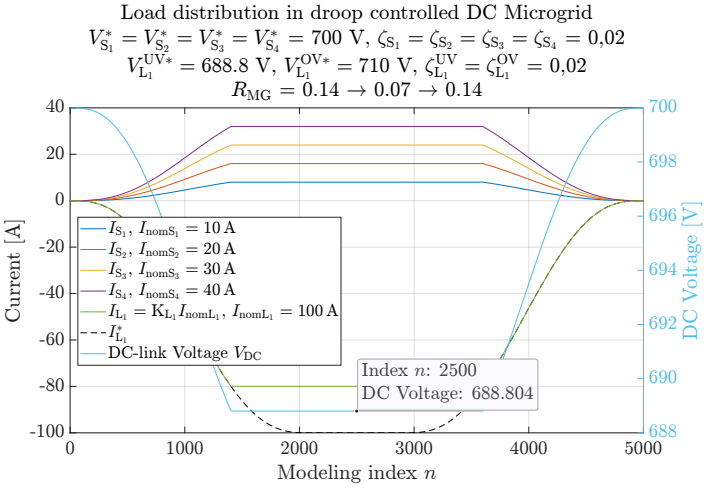
where x_i denotes each device in the system (e.g., "S_1" or "L_2"). Here $\mathrm{B}_{\mathrm{V}x_i}(n)$ indicates whether device x_i operates in voltage-control mode at step n, and $\mathrm{B}_{\mathrm{run}x_i}(n)$ denotes its running status, defined analogously to the source-specific terms in Equations (3.10) and (3.11). Load current limiting and stopping are handled separately in Section 3.1.8.2.

Figure 35 presents the modeling results in which the load inverter adjusts its current draw to prevent the DC-link voltage from falling below the undervoltage regulator activation threshold. Figure 35a illustrates the behavior of $\mathrm{K}_{\mathrm{L}_1}$ as it becomes limited by $\Gamma^{\mathrm{UV}}_{\mathrm{L}_1}$. The output currents of the modeled inverters are shown in Figure 35b, along with the load inverter's external current reference $I^{*}_{\mathrm{L}_1}$. The behavior of the DC-link voltage is presented in more detail in Figure 35c.

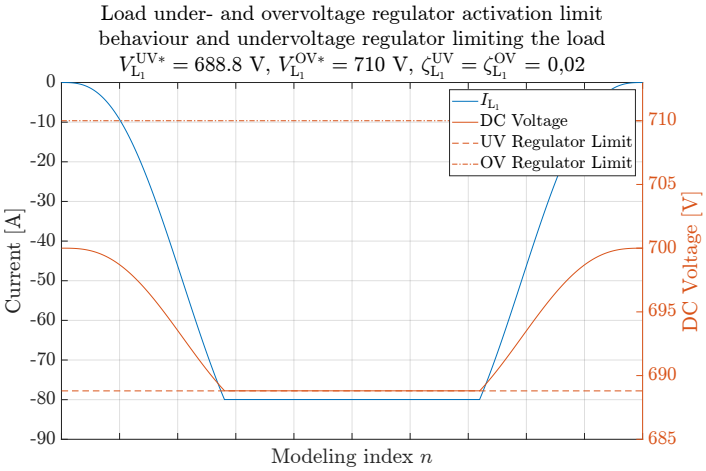
Figure 35c displays the behavior of the under- and overvoltage regulator activation limits with the implemented conditional droop compensation. When the undervoltage regulator is active, all devices in the system contribute to maintaining the DC-link voltage, indicating a stable operating point. As the current reference $I^{*}_{\mathrm{L}_1}$ lowers below the actual load current, the undervoltage regulator deactivates.



(a) Load's Γ_{L}^{UV} and Γ_{L}^{OV}, and effect of Γ_{L}^{UV} limiting the K.



(b) Inverter outputs and DC-link voltage.



(c) Load's under- and overvoltage regulator voltage limits.

Figure 35. Results of system mathematical modeling with load undervoltage regulator.

3.1.7.1 Voltage Regulation in Multi-Load Configurations

Shipboard power systems are rarely simple enough to be designed for a single load. Therefore, regulator functionality must be evaluated in the context of multiple loads operating in parallel, each equipped with its own voltage regulation. To demonstrate the operation of the under- and overvoltage regulators in a system with a more complex load configuration, the modeling scenarios in this section differ from the examples shown earlier in two principal ways (see also Figure 36):

1. The voltage source setup has been simplified to a single source, with its output limited to a maximum of 100 A, corresponding to its nominal current rating.
    - The focus in this section is on the behavior of the load-side regulators, rather than on power sharing among multiple voltage sources.
    - Depicting multiple loads and sources simultaneously would reduce figure readability.
    - The limited output prevents the source from overloading itself when supplying the loads. Without functioning voltage regulators, the DC-link voltage would collapse. This phenomenon is examined more closely in Section 5.2.
2. The system under study includes two loads, each following its own load profile and equipped with undervoltage regulator.
    - For simplicity, only undervoltage regulation is examined more deeply in this section. Overvoltage regulator functional demonstration is covered in Section 3.1.7.3
    - The undervoltage activation threshold for load L_1 is set equal to the source's nominal drooped voltage: $V^*_{S_1}(1 - \zeta_{S_1}) = 686$ V.
    - For load L_2, the undervoltage activation threshold is set 2 V lower than that of load L_1 to enable clear differentiation between regulator activation levels.

While the modeling results in Figure 35 confirm that the specified regulator functions as intended in a single-load scenario, applying the same regulator

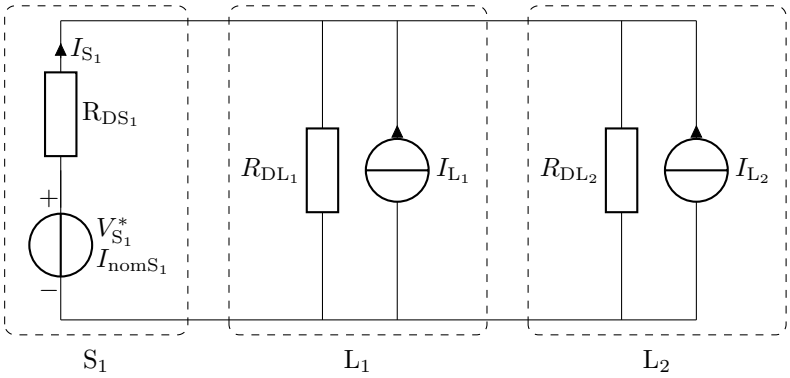


Figure 36. Circuit diagram of the single source and multiple load system for regulator development.

configuration in a system with two loads leads to unsatisfactory results, as illustrated in Figure 37.

In addition to the undervoltage regulator of load L_2 remaining active even when the DC-link voltage is well above any defined undervoltage activation thresholds, Figure 37 also illustrates how load L_1 fails to operate smoothly under these conditions. While L_1 is constrained by its undervoltage regula-

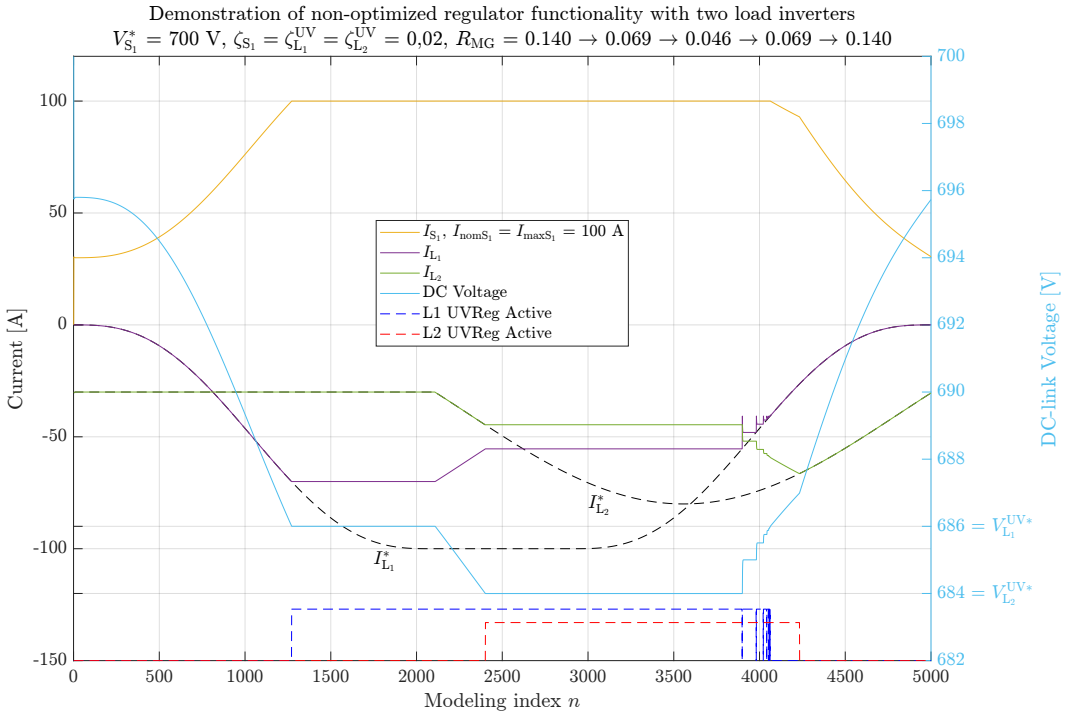


Figure 37. Demonstration of non-optimized undervoltage regulator operation with two load inverters.

tor and its reference $I^*_{\mathrm{L}_1}$ is rising towards zero, the reference's eventual crossing of the load's output current triggers regulator deactivation $\left(\mathrm{K}^*_{\mathrm{L}_1} < \Gamma^{\mathrm{UV}}_{\mathrm{L}_1}\right)$. At the same time, load inverter L_2 remains locked under its own undervoltage regulator. The deactivation of L_1's regulator leads to an internal update of its droop-compensated activation limit, which is accompanied by a sudden increase in the DC-link voltage. This rise is undesirable on its own, but more critically, the resulting voltage remains below the newly updated activation threshold, causing the regulator to re-activate immediately — resulting in rapid toggling or chattering of the regulation logic.

To eliminate this instability, the regulator must account for the inverter's normalized reference $\mathrm{K}^*_{\mathrm{L}}$ even while the regulator is active. This enables smooth and stable deactivation of the regulator once system conditions normalize.

As defined in Equations (3.36) and (3.37), the activation thresholds $V^{\mathrm{UV}*'}_{\mathrm{L}}$ and $V^{\mathrm{OV}*'}_{\mathrm{L}}$ are droop-compensated to offset voltage deviations caused by the load's own current I_{L} when the regulator is inactive. Once the regulator activates, these thresholds remain fixed. However, maintaining them strictly constant can cause discontinuities when $\mathrm{K}^*_{\mathrm{L}}$ reaches the activation limit $\Gamma^{\mathrm{UV}}_{\mathrm{L}}$ or $\Gamma^{\mathrm{OV}}_{\mathrm{L}}$, particularly if the background system dynamics change due to other loads. To avoid this, the threshold must remain bounded by the normalized current reference $\mathrm{K}^*_{\mathrm{L}}$ while the regulator is active, enabling smooth and stable deactivation.

It is assumed that inverters follow their references perfectly under normal conditions, such that

$$\frac{I_{\mathrm{L}}}{I_{\mathrm{nomL}}} = \mathrm{K}_{\mathrm{L}} = \mathrm{K}^*_{\mathrm{L}} \tag{3.43}$$

when the regulators are inactive.

If the activation thresholds $V^{\mathrm{UV}*'}_{\mathrm{L}}$ and $V^{\mathrm{OV}*'}_{\mathrm{L}}$ are constrained to the normalized reference-wise drooped values, the Equations 3.36 and 3.37 become:

$$V^{\mathrm{UV}*'}_{\mathrm{L}} = \begin{cases} V^{\mathrm{UV}*'}_{\mathrm{L}} \in \left[V^{\mathrm{UV}*}_{\mathrm{L}}\left(1 + \mathrm{K}^*_{\mathrm{L}}\zeta^{\mathrm{UV}}_{\mathrm{L}}\right), \infty\right[& \text{if } V_{\mathrm{DC}} < V^{\mathrm{UV}*'}_{\mathrm{L}} \text{ or } \mathrm{K}^*_{\mathrm{L}} < \Gamma^{\mathrm{UV}}_{\mathrm{L}}, \\ V^{\mathrm{UV}*}_{\mathrm{L}}\left(1 + \dfrac{I_{\mathrm{L}}}{I_{\mathrm{nomL}}}\zeta^{\mathrm{UV}}_{\mathrm{L}}\right) & \text{otherwise,} \end{cases} \tag{3.44}$$

$$V_{\mathrm{L}}^{\mathrm{OV}*\prime} = \begin{cases} V_{\mathrm{L}}^{\mathrm{OV}*\prime} \in \left]-\infty, V_{\mathrm{L}}^{\mathrm{OV}*}\left(1+\mathrm{K}_{\mathrm{L}}^{*}\zeta_{\mathrm{L}}^{\mathrm{OV}}\right)\right] & \text{if } V_{\mathrm{DC}} > V_{\mathrm{L}}^{\mathrm{OV}*\prime} \text{ or } \mathrm{K}_{\mathrm{L}}^{*} > \Gamma_{\mathrm{L}}^{\mathrm{OV}}, \\ V_{\mathrm{L}}^{\mathrm{OV}*}\left(1+\dfrac{I_{\mathrm{L}}}{I_{\mathrm{nomL}}}\zeta_{\mathrm{L}}^{\mathrm{OV}}\right) & \text{otherwise.} \end{cases} \tag{3.45}$$

These formulations eliminate the stepwise transitions at regulator deactivation points, as demonstrated in Figure 38.

This smooths the regulator deactivation, eliminating sudden voltage rises and unwanted regulator toggling. However, the undervoltage regulator of load L_2 still fails to deactivate when it should, remaining active even after load L_1's regulator has turned off.

It is evident that a situation occurs where $\mathrm{K}_{\mathrm{L}_2}^{*} < \Gamma_{\mathrm{L}_2}^{\mathrm{UV}}$, keeping the regulator active, even though the DC-link voltage has risen above the specified activation threshold, i.e., $V_{\mathrm{DC}} > V_{\mathrm{L}_2}^{\mathrm{UV}*}$. This exposes a flaw in the current logic used to determine whether droop compensation should be applied, as defined in Equations (3.44) and (3.45).

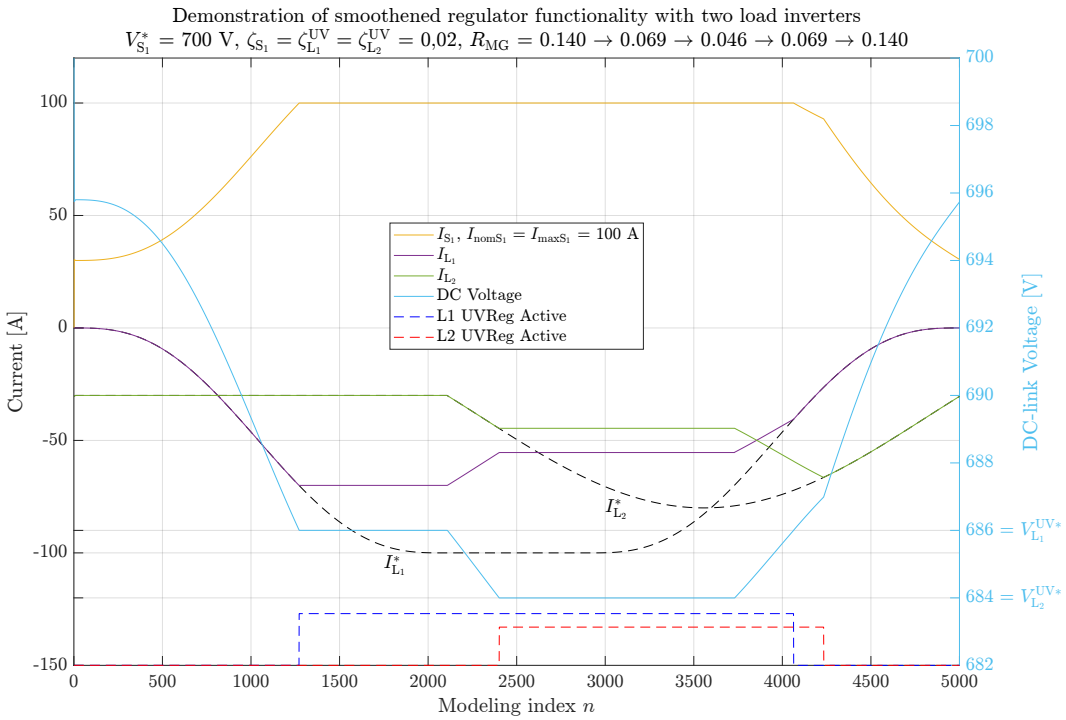


Figure 38. Demonstration of smoothened undervoltage regulator operation with two load inverters.

The underlying problem arises because the clamped normalized current $\mathrm{K_L}$ is based on $\mathrm{K_L^*}$, constrained by either $\Gamma_\mathrm{L}^\mathrm{UV}$ or $\Gamma_\mathrm{L}^\mathrm{OV}$ depending on which limit is reached first, as defined in Equation 3.33. These $\Gamma_\mathrm{L}^\mathrm{UV}$ and $\Gamma_\mathrm{L}^\mathrm{OV}$ themselves are based then on the $V_\mathrm{L}^{\mathrm{UV}*\prime}$ and $V_\mathrm{L}^{\mathrm{OV}*\prime}$ respectively, which are in turn dependent on $\mathrm{K_L^*}$ as seen in Equation 3.45, creating a circular logic.

To eliminate this feedback loop, the condition for applying droop compensation must be decoupled from the regulator limits $\Gamma_\mathrm{L}^\mathrm{UV}$ and $\Gamma_\mathrm{L}^\mathrm{OV}$ and instead rely directly on comparing the DC-link voltage V_DC to the desired regulator activation thresholds $V_\mathrm{L}^{\mathrm{UV}*}$ and $V_\mathrm{L}^{\mathrm{OV}*}$, as follows:

$$V_\mathrm{L}^{\mathrm{UV}*\prime} = \begin{cases} V_\mathrm{L}^{\mathrm{UV}*\prime} \in \left[V_\mathrm{L}^{\mathrm{UV}*} \left(1 + \mathrm{K_L}\zeta_\mathrm{L}^\mathrm{UV}\right), \infty \right[& \text{if } V_\mathrm{DC} \leq V_\mathrm{L}^{\mathrm{UV}*}, \\ V_\mathrm{L}^{\mathrm{UV}*} \left(1 + \dfrac{I_\mathrm{L}}{I_\mathrm{nomL}}\zeta_\mathrm{L}^\mathrm{UV}\right) & \text{otherwise,} \end{cases} \tag{3.46}$$

$$V_\mathrm{L}^{\mathrm{OV}*\prime} = \begin{cases} V_\mathrm{L}^{\mathrm{OV}*\prime} \in \left] -\infty, V_\mathrm{L}^{\mathrm{OV}*} \left(1 + \mathrm{K_L}\zeta_\mathrm{L}^\mathrm{OV}\right) \right] & \text{if } V_\mathrm{DC} \geq V_\mathrm{L}^{\mathrm{OV}*}, \\ V_\mathrm{L}^{\mathrm{OV}*} \left(1 + \dfrac{I_\mathrm{L}}{I_\mathrm{nomL}}\zeta_\mathrm{L}^\mathrm{OV}\right) & \text{otherwise.} \end{cases} \tag{3.47}$$

Now, with the modified equations, the voltage regulators smoothly respond to decreasing current references, compensate for droop-induced voltage deviations in their activation thresholds, and operate correctly in parallel with other loads under regulation. Figure 39 shows the improved regulator in the same scenario as earlier examples. It can be observed that the regulators activate and deactivate as expected at their respective voltage thresholds, with smooth transitions driven by changes in their current references.

As shown in Figure 39, both loads operate against their undervoltage regulators in a coordinated manner, keeping the system loading at a level that maintains the DC-link voltage at the regulator activation limits as needed. When the reference of load $\mathrm{L_1}$ decreases, reducing its power consumption, load $\mathrm{L_2}$ correspondingly increases its load toward its external reference. This behavior occurs because the reduced loading by $\mathrm{L_1}$ frees up capacity, allowing $\mathrm{L_2}$ to resume following its intended reference more closely. Consequently, the DC-link voltage remains near the undervoltage threshold until all load references have decreased sufficiently to allow normal operation without regulator intervention.

Figures 40 and 41 present more accurately the internal operations of the

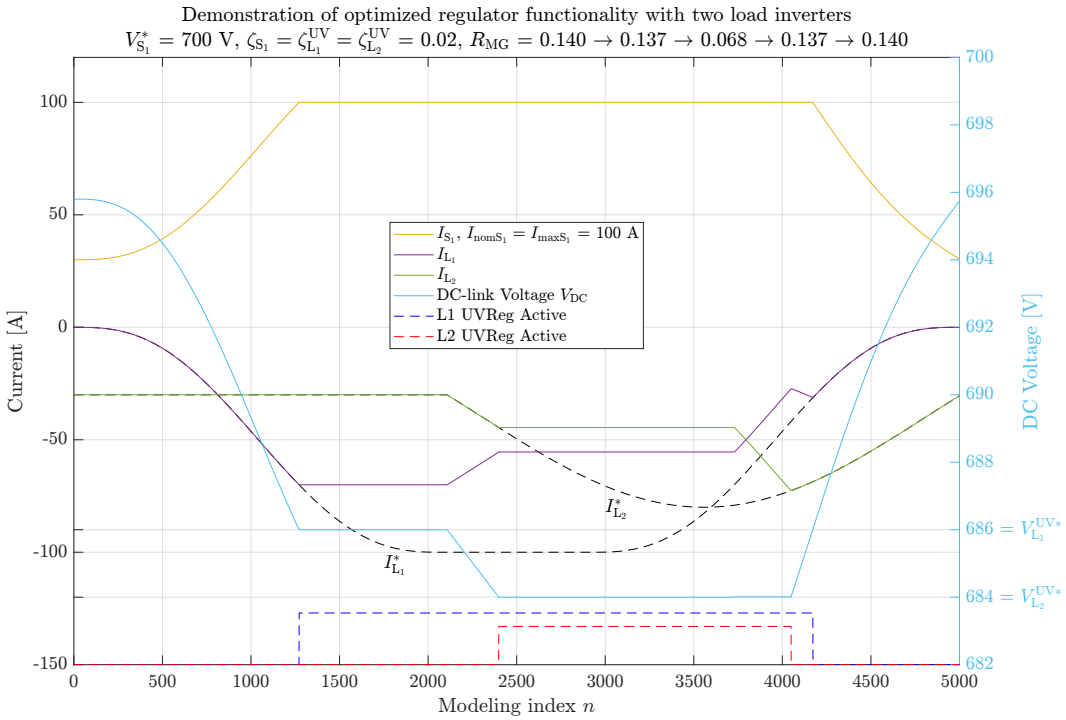


Figure 39. Demonstration of optimized undervoltage regulator operation with two load inverters.

undervoltage regulators. The figures show the conditionally droop-compensated regulator activation limits $V^{UV*'}_{L_1}$ and $V^{UV*'}_{L_2}$ calculated with Equation (3.46) and their limiting factors for when the regulator is active, i.e., $V^{UV*}_{L_1}\left(1 + K_{L_1}\zeta^{UV}_{L_1}\right)$ and $V^{UV*}_{L_2}\left(1 + K_{L_2}\zeta^{UV}_{L_2}\right)$.

After the normalized reference $K^*_{L_1}$ of load L_1 has peaked and is changing toward zero, the corresponding undervoltage regulator activation limit $V^{UV*'}_{L_1}$ rises. This rise reflects reduced output demand from L_1 and, consequently starting from modeling index $n \approx 3700$, reduces the pressure its operation places on the DC-link voltage. As a result, load L_2 is allowed to lower its own $V^{UV*'}_{L_2}$ threshold further while remaining active, enabling it to increase its output current and follow its reference more closely. This interaction illustrates how the variable voltage-based control allows multiple loads to operate simultaneously under regulation, while gradually releasing the regulators in a coordinated and stable manner as load demands shift.

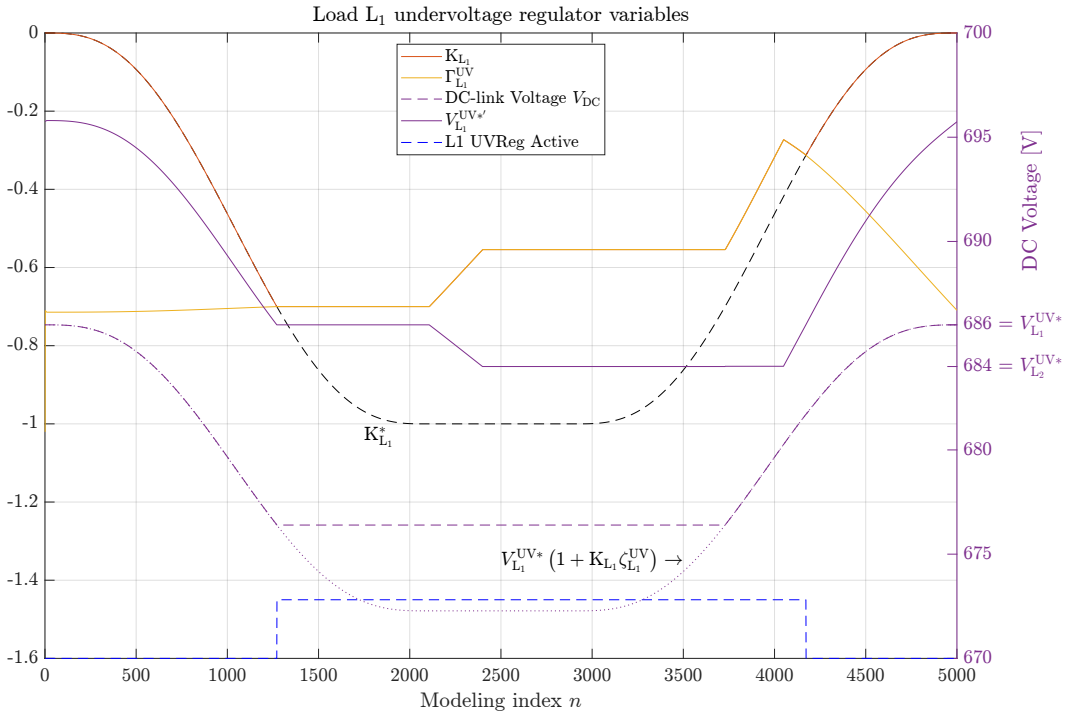


Figure 40. Undervoltage regulator operation in load L_1.

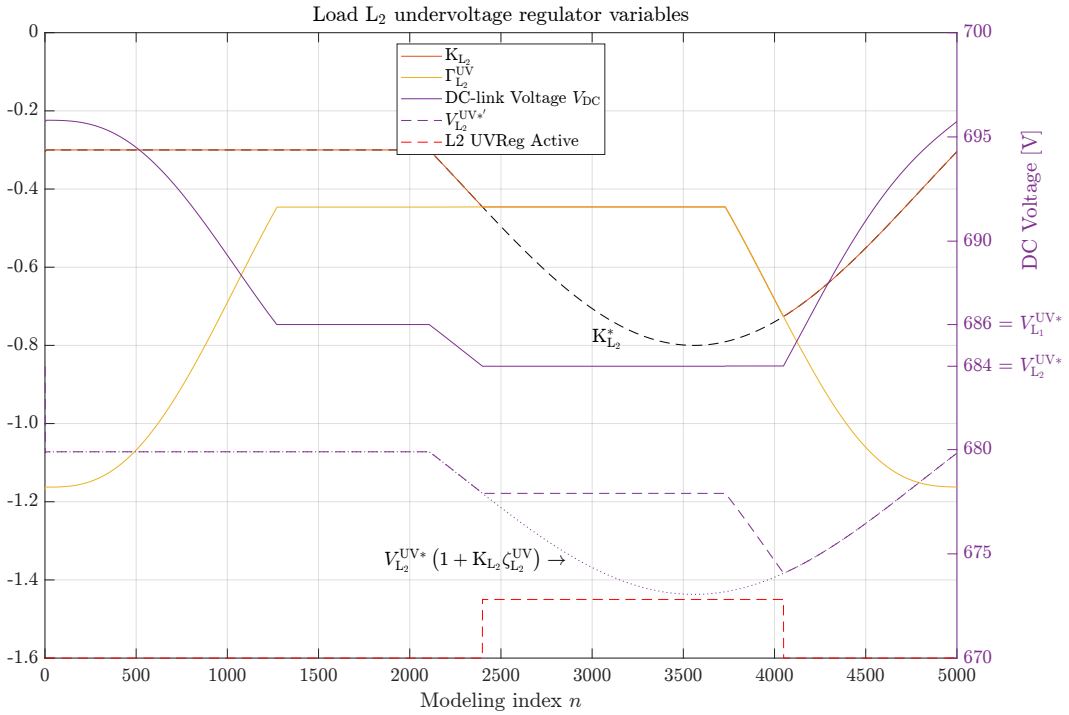


Figure 41. Undervoltage regulator operation in load L_1.

3.1.7.2 MATLAB Implementation of the Optimized Voltage Regulators

As the under- and overvoltage regulators are inverter-specific functionalities, the inverters were implemented as a reusable MATLAB class to streamline the modeling software architecture. The source code is presented in Appendix 3. This class-based implementation enables object-oriented modeling, allowing the same inverter class to be reused for as many inverter instances as needed.

The inverter class in Appendix 3 also includes support for using a power reference, which is applied in Section 3.1.10 during nodal analysis, and multiple other not yet covered features like current limiting and stopping. For comparative purposes, the class also provides the option to disable conditional droop compensation and operate the voltage regulators in a basic mode. These functionalities are configured via constructor parameters when creating the inverter objects; see Code 13 for details.

Appendix 4 presents the source code for modeling a system in which the loads are instantiated from the inverter class. Similar to Section 3.1.4.1, where a source is limited to 60% of its nominal output, this example limits the source to 100% and treats its droop resistance as infinite when the current limit is reached. Additionally, the loads' droop resistances are considered non-zero when operating under their voltage regulators.

```
loads(1,1) = inverterAdvReg(InomL(1), ...
                        UVRefs(1), ...
                        UVDroop(1), ...
                        OVRefs(1), ...
                        OVDroop(1), ...
                        1, ... % 0 = normal drooping regulator
                        ...    % 1 = drooping regulator with conditional droop compensation from the dissertation
                        0, ... % 0 = reference is considered current reference
                        ...    % 1 = reference is considered power reference
                        700); % Nominal voltage for power calculations
```

Code 13. MATLAB code snippet from Appendix 4 demonstrating inverter constructor.

3.1.7.3 Demonstration of Overvoltage Regulator Operation

The previous sections have focused on the undervoltage regulator functionality for clarity, even though the mathematical formulation of the overvoltage regulator differs from that of the undervoltage regulator (see Equation 3.47), and therefore requires separate verification. Utilizing the code from Appendix 4, it is straightforward to modify the system so that the load direction is reversed. In this configuration, the droop controlled source operates in negative supply mode, which raises the DC voltage. Code 14 lists the lines to add or modify in order to achieve this setup.

Figures 42 presents the modeled currents and DC voltage for the reversed loading case, while Figures 43 and 44 show the regulators' control variable

```
negLims(1) = -1.0;                      % Added code line
OVRefs = [700*1.02; 716];               % Modified code line
loadProfiles(1,:) = -KL1;               % Modified code line
loadProfiles(2,:) = -load2CurrentRef; % Modified code line
```

Code 14. Code snippet showing the modifications needed to reverse the system loading in Appendix 4 for overvoltage regulator testing.

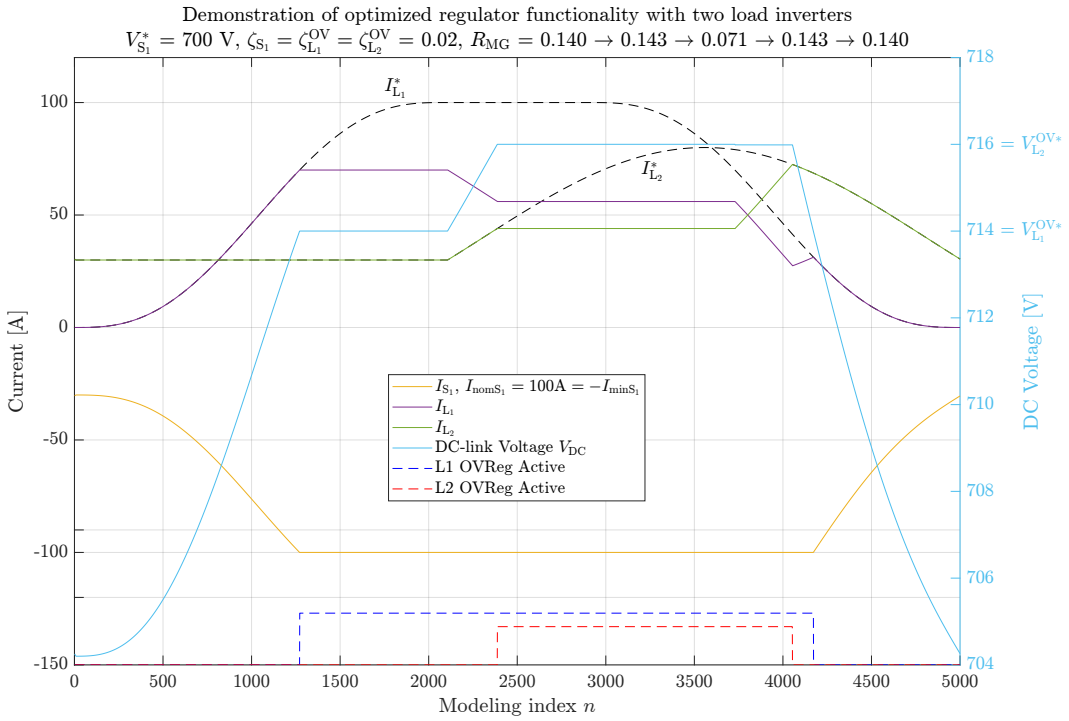


Figure 42. Demonstration of overvoltage regulator operation.

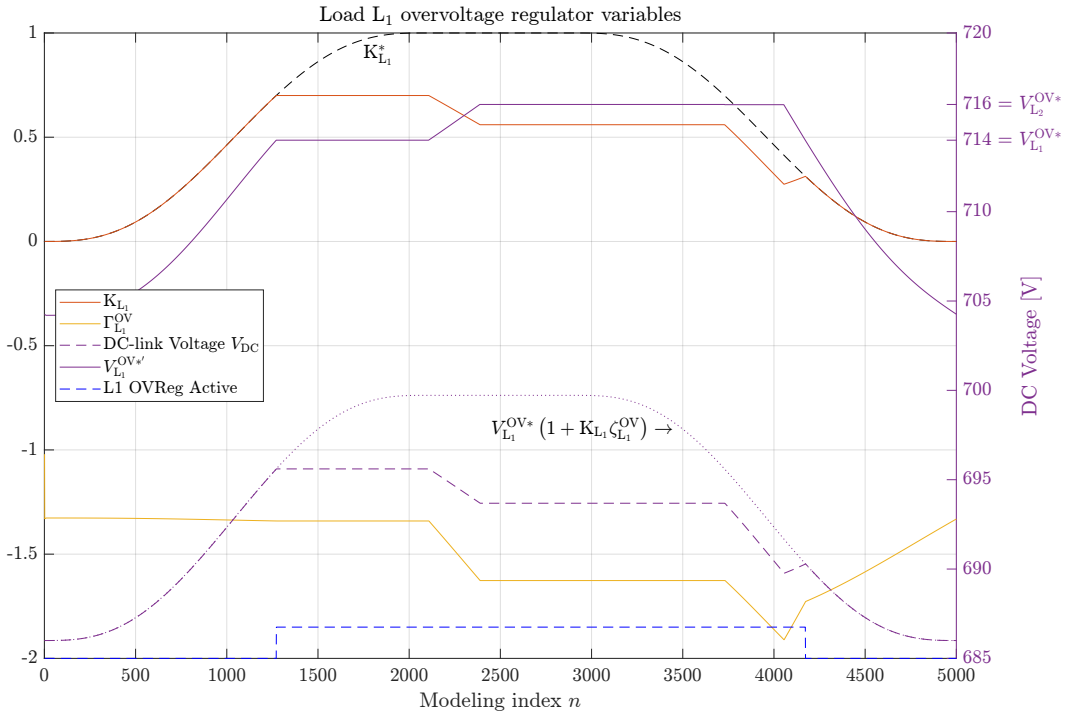


Figure 43. Overvoltage regulator operation in Load L_1.

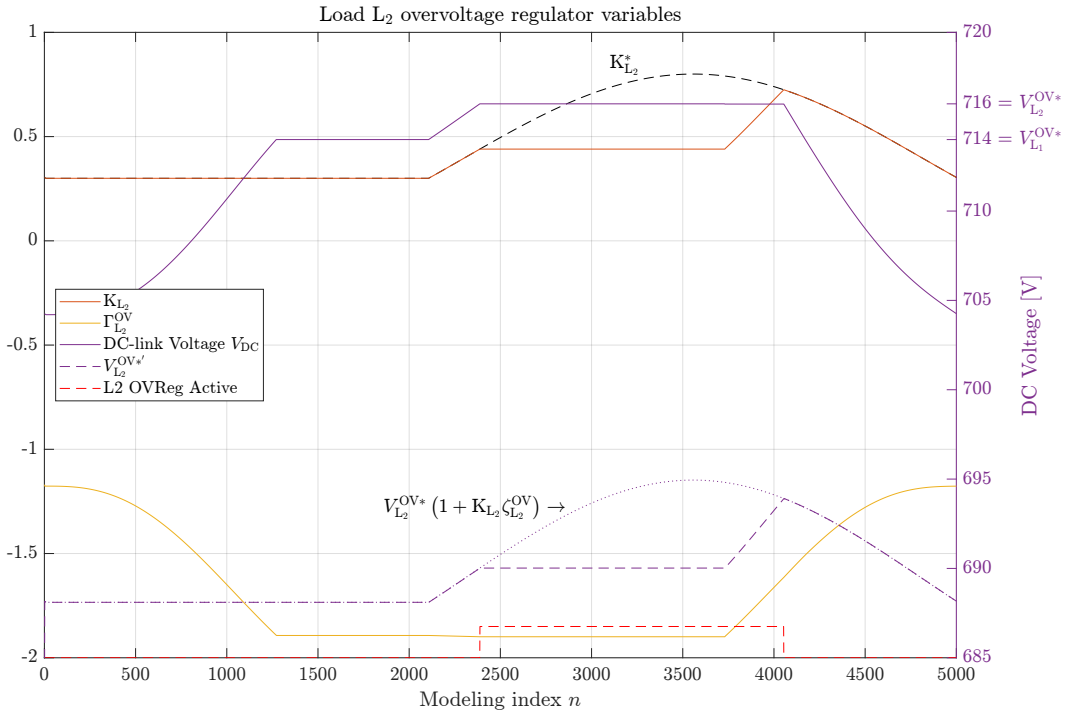


Figure 44. Overvoltage regulator operation in Load L_2.

behaviors during the operation, as in Section 3.1.7.1. In this scenario, Source S_1 is limited in the negative direction to its nominal current, and the activa-

tion level of Load L_1's overvoltage regulator is set to be 2% higher than the source's reference voltage. This creates a situation analogous to the undervoltage regulator case, where the activation level coincides with the drooped DC voltage under nominal load. The activation level of Load L_2's regulator is then set 2 V higher that of L_1. As expected, the results are practically identical to those obtained for the undervoltage regulator, except that the power directions are reversed.

3.1.8 Including Load DC Voltage Measurement Offset Induced Circulating Currents and Exceptional Situations

During normal current-controlled operation, the loads do not depend on the sensed DC voltage. Therefore, they do not introduce circulating currents caused by DC voltage measurement errors. However, circulating currents appear whenever a load regulates the DC voltage via its under- or overvoltage regulator, since in that moment the load effectively operates as a voltage source.

This behavior is analogous, but inverse, to that of actual voltage sources with current limits. A voltage source normally maintains the DC voltage, but once its output current reaches the specified limit, it switches to current-source behavior, and reverts to voltage-source behavior when the load decreases. Conversely, a load normally operates as a current source (or sink), but once the DC voltage reaches its specified limit, it begins voltage regulation and thus ceases to operate in current-controlled mode.

Consequently, when loads operate under their voltage regulators and effectively behave as voltage sources their output then depends on the measured DC voltage. This renders them susceptible to voltage measurement errors and thus, the loads in this situation can both induce and be influenced by circulating currents within the system.

Circulating currents affect all voltage-regulating devices in the system—both sources and loads. For voltage sources, this is reflected by $\mathbf{I}_{CS}$ in Equation (3.22), where currents circulate between sources. With loads now also capable of regulating voltage, a more comprehensive current matrix $\mathbf{I}_C$ is required. This matrix must represent the current contributions of all sources

and loads, including all circulating currents between them:

$$\mathbf{I}_{\mathrm{C}} = \begin{bmatrix} I_{\mathrm{CS}_1} & \cdots & I_{\mathrm{CS}_1\mathrm{S}_N} & I_{\mathrm{CS}_1\mathrm{L}_1} & \cdots & I_{\mathrm{CS}_1\mathrm{L}_N} \\ \vdots & \ddots & & & & \vdots \\ I_{\mathrm{CS}_N\mathrm{S}_1} & & I_{\mathrm{CS}_N} & I_{\mathrm{CS}_N\mathrm{L}_1} & \cdots & I_{\mathrm{CS}_N\mathrm{L}_N} \\ I_{\mathrm{CL}_1\mathrm{S}_1} & & I_{\mathrm{CL}_1\mathrm{S}_N} & I_{\mathrm{CL}_1} & & I_{\mathrm{CL}_1\mathrm{L}_N} \\ \vdots & & & & \ddots & \vdots \\ I_{\mathrm{CL}_N\mathrm{S}_1} & \cdots & I_{\mathrm{CL}_N\mathrm{S}_N} & I_{\mathrm{CL}_N\mathrm{L}_1} & \cdots & I_{\mathrm{CL}_N} \end{bmatrix} \tag{3.48}$$

Calculating the matrix can be done mathematically the same way as was with sources only. Except that now all variables must be expanded to involve loads as well.

From $\mathbf{I}_{\mathrm{C}}$ it is then possible to derive other relevant current arrays similarly to the case when only sources were considered: the array of equipment total currents $\mathbf{I}_{\Sigma\mathrm{C}}$ as the sum of the rows (Equation (3.23)), and the array containing the diagonal elements of $\mathbf{I}_{\mathrm{C}}$, which represent each equipment's net contribution to the DC-link $\mathbf{I} = [\mathbf{I_S} \quad \mathbf{I_L}]^{\mathrm{T}}$—i.e., the currents flowing directly between each device and the DC-link (Equation (3.24)).

Consequently, Boolean arrays for load's voltage source operation $\mathbf{B}_{\mathrm{VL}}$ and running statuses $\mathbf{B}_{\mathrm{runL}}$, analogous to sources' $\mathbf{B}_{\mathrm{VS}}$ and $\mathbf{B}_{\mathrm{runS}}$, have to defined and all of them be generalized under $\mathbf{B}_{\mathrm{V}}$ and $\mathbf{B}_{\mathrm{run}}$, thereby unifying the modeling framework for all equipment. To calculate the $\mathbf{I}_{\mathrm{C}}$ matrix containing every source and load, also the nominal current arrays must be combined as $\mathbf{I}_{\mathrm{nom}} = [\mathbf{I}_{\mathrm{nomS}} \quad \mathbf{I}_{\mathrm{nomL}}]^{\mathrm{T}}$, as well as their Γ and K arrays as $\boldsymbol{\chi} = [\boldsymbol{\Gamma}_{\mathrm{S}} \quad \mathbf{K}_{\mathrm{L}}]^{\mathrm{T}}$. All these generalizations are listed in Table 5.

The mathematics remain unchanged; previously, circulating currents and other exceptional behaviors (measurement imperfections, current limits, and the possibility to stop) were modeled only for sources. Now that loads are modeled with the same features, $\mathbf{I}_{\mathrm{C}}$ must include all equipment:

$$\mathbf{I}_{\mathrm{C}} = (\mathrm{diag}(\boldsymbol{\chi}) + \boldsymbol{\Gamma}_{\mathrm{C}})\, \mathbf{I}_{\mathrm{nom}}. \tag{3.49}$$

Assigning the $\mathbf{B}_{\mathrm{V}}$ values and zeroing the circulating-current components of $\mathbf{I}_{\mathrm{C}}$ are performed exactly the same way as in Section 3.1.5.1. The calculation of R_{MG} remains as specified in Equation (3.42). The symbol $\boldsymbol{\Gamma}_{\mathrm{C}}$ is retained: it denotes circulating-current offsets specific to voltage-source operation (con-

Table 5. Generalized equations for modeling sources and loads.

Equation	Description
$\mathbf{I}_{\mathrm{C}} = \begin{bmatrix} \mathbf{I}_{\mathrm{CS}} & \mathbf{I}_{\mathrm{CSL}} \\ \mathbf{I}_{\mathrm{CLS}} & \mathbf{I}_{\mathrm{CL}} \end{bmatrix}$	Individual current components between microgrid equipment
$\mathbf{I}_{\Sigma\mathrm{C}} = \mathbf{I}_{\mathrm{C}} \cdot 1 \in \mathbb{R}^{N \times 1}$	Sum of each $\mathbf{I}_{\mathrm{C}}$ matrix row; Equipment total current array
$\mathbf{I} = \mathrm{diag}\left(\mathbf{I}_{\mathrm{C}}\right) = \begin{bmatrix} \mathbf{I}_{\mathrm{S}} \\ \mathbf{I}_{\mathrm{L}} \end{bmatrix}$	Voltage regulating or load feeding currents
$\mathbf{I}_{\mathrm{nom}} = \begin{bmatrix} \mathbf{I}_{\mathrm{nomS}} \\ \mathbf{I}_{\mathrm{nomL}} \end{bmatrix}$	Equipment nominal currents
$\mathbf{B}_{\mathrm{V}} = \begin{bmatrix} \mathbf{B}_{\mathrm{VS}} \\ \mathbf{B}_{\mathrm{VL}} \end{bmatrix}$	Boolean array for equipment voltage regulation status
$\mathbf{B}_{\mathrm{run}} = \begin{bmatrix} \mathbf{B}_{\mathrm{runS}} \\ \mathbf{B}_{\mathrm{runL}} \end{bmatrix}$	Boolean array for equipment running status
$\boldsymbol{\chi} = \begin{bmatrix} \boldsymbol{\Gamma}_{\mathrm{S}} \\ \mathbf{K}_{\mathrm{L}} \end{bmatrix}$	Equipment current contribution coefficients

sistent with the use of Γ in this work). Its contents are now extended to cover the offsets arising from loads when they operate as voltage sources, computed using Equations (3.19) and (3.20).

The challenge in modeling circulating currents and current limitation for loads arises from the fact that, at the beginning of each modeling step, the overall system state, is not yet known (see Figure 11). While the loads' next current reference and the voltage regulator activation status—based on the DC voltage from the previous step—are known, the updated DC voltage level and, in particular, the source-side contributions regulating voltage and injecting circulating currents remain undetermined (sources' $\boldsymbol{\Gamma}_{\mathrm{S}}$ array and the whole $\boldsymbol{\Gamma}_{\mathrm{C}}$ matrix).

This poses a problem because, when a load operates under its voltage regulation mode, the total output current $I_{\Sigma\mathrm{CL}_i}$—which must be constrained by the load's current limit—includes not only the current used for regulating voltage I_{L_i} but also the circulating current components induced by all other

voltage-regulating devices. If the load hits its current limit in this mode, it transitions back into current-controlled operation, thereby altering the system's dynamics.

To model this behavior accurately, each load would require knowledge of the circulating currents it exchanges with all other voltage-regulating equipment at the start of the modeling step. These values would then be used in current limitation and voltage regulation calculations. However, this information only becomes available after the updated DC voltage and source currents have been calculated. Moreover, activating a current limit in one device could cause another device to reach its limit in response, leading to a cascade of interdependent mode transitions. Solving this accurately would require iterative-style recalculations of the system state within a single modeling step.

Although this quasi-static modeling paradigm omits time-based dynamics—such as system capacitances and inductances—these control-based dynamics remain present. From a control development standpoint, it would be beneficial to address and validate these control dynamics within this electrically simpler modeling environment before transitioning to time-variant models, where additional physical dynamics are introduced.

Appendix 5 presents the MATLAB code for modeling a system consisting of four voltage sources and a single load. The load is implemented using the inverter class from Appendix 3 with the generalized variables presented in Table 5, supporting voltage regulation, current limitation, operational stopping, and circulating currents induced by voltage measurement errors. In this implementation, the circulating currents of both sources and loads are computed at the end of each modeling step and applied to the load calculations in the subsequent step. This approach allows the model to determine whether the load's current limit is exceeded based on the total output current, including circulating currents from the previous modeling step.

For more accurate modeling, a more advanced approach would be required. This would involve sequential steady-state recalculations of the DC voltage and current matrix within a single modeling step, incrementally accounting for mode transitions, current limitations, and their consequences. Each recalculation would be based on fully updated system states rather than iterative refinement of previous estimates.

3.1.8.1 Modeling the Load's Measurement Error Induced Circulating Currents

Each load's DC voltage measurement error affects its regulator contribution calculations both directly and indirectly: directly through Equations (3.38) and (3.39), and indirectly via the droop-compensated regulator activation limits in Equations (3.46) and (3.47). As with sources, the sensed DC voltage for load L_i is given by

$$V_{\mathrm{L}_i} = V_{\mathrm{DC}} + V_{\mathrm{errL}_i}, \tag{3.50}$$

where V_{errL_i} is the measurement error of load L_i. Using V_{L_i}, the updated regulator contribution equations become:

$$\Gamma_{\mathrm{L}}^{\mathrm{UV}} = \frac{V_{\mathrm{L}}^{\mathrm{UV}*\prime} - V_{\mathrm{L}}}{\zeta_{\mathrm{L}}^{\mathrm{UV}} V_{\mathrm{L}}^{\mathrm{UV}*}} \tag{3.51}$$

and

$$\Gamma_{\mathrm{L}}^{\mathrm{OV}} = \frac{V_{\mathrm{L}}^{\mathrm{OV}*\prime} - V_{\mathrm{L}}}{\zeta_{\mathrm{L}}^{\mathrm{OV}} V_{\mathrm{L}}^{\mathrm{OV}*}}, \tag{3.52}$$

and the corresponding activation limits:

$$V_{\mathrm{L}}^{\mathrm{UV}*\prime} = \begin{cases} V_{\mathrm{L}}^{\mathrm{UV}*\prime} \in \left[V_{\mathrm{L}}^{\mathrm{UV}*} \left(1 + \mathrm{K}_{\mathrm{L}} \zeta_{\mathrm{L}}^{\mathrm{UV}}\right), \infty \right[& \text{if } V_{\mathrm{L}} \leq V_{\mathrm{L}}^{\mathrm{UV}*}, \\ V_{\mathrm{L}}^{\mathrm{UV}*} \left(1 + \dfrac{I_{\mathrm{L}}}{I_{\mathrm{nomL}}} \zeta_{\mathrm{L}}^{\mathrm{UV}}\right) & \text{otherwise,} \end{cases} \tag{3.53}$$

$$V_{\mathrm{L}}^{\mathrm{OV}*\prime} = \begin{cases} V_{\mathrm{L}}^{\mathrm{OV}*\prime} \in \left] -\infty, V_{\mathrm{L}}^{\mathrm{OV}*} \left(1 + \mathrm{K}_{\mathrm{L}} \zeta_{\mathrm{L}}^{\mathrm{OV}}\right) \right] & \text{if } V_{\mathrm{L}} \geq V_{\mathrm{L}}^{\mathrm{OV}*}, \\ V_{\mathrm{L}}^{\mathrm{OV}*} \left(1 + \dfrac{I_{\mathrm{L}}}{I_{\mathrm{nomL}}} \zeta_{\mathrm{L}}^{\mathrm{OV}}\right) & \text{otherwise.} \end{cases} \tag{3.54}$$

Using these DC voltage measurement error included equations it is possible to derive now $\mathrm{K}_{\mathrm{RegL}}^*$ value which includes the voltage regulator defined limitations, but not yet the actual current limitations:

$$\mathrm{K}_{\mathrm{RegL}_i}^*(n) = \begin{cases} \Gamma_{\mathrm{L}_i}^{\mathrm{OV}}(n), & \text{if } \mathrm{K}_{\mathrm{L}_i}^* \geq \Gamma_{\mathrm{L}_i}^{\mathrm{OV}}(n) \\ \mathrm{K}_{\mathrm{L}_i}^*, & \text{if } \Gamma_{\mathrm{L}_i}^{\mathrm{UV}}(n) < \mathrm{K}_{\mathrm{L}_i}^* < \Gamma_{\mathrm{L}_i}^{\mathrm{OV}}(n) \\ \Gamma_{\mathrm{L}_i}^{\mathrm{UV}}(n), & \text{if } \mathrm{K}_{\mathrm{L}_i}^* \leq \Gamma_{\mathrm{L}_i}^{\mathrm{UV}}(n). \end{cases} \tag{3.55}$$

The complete reference chain with the $\mathrm{K}_{\mathrm{RegL}}^*$ and current limiter is presented

in Figure 45.

To incorporate the effect of circulating currents into the load model, they must be considered when deriving the final current contribution factor $\mathrm{K_L}$ from the $\mathrm{K^*_{RegL}}$ value, as shown in Figure 45. However, only the circulating current components should be included—that is, the off-diagonal elements of $\mathbf{I}_\mathrm{C}$—while omitting the diagonal elements representing load feeding or voltage-regulating currents:

$$\mathbf{I}_{\Sigma\mathrm{CC}} = (\mathbf{I}_\mathrm{C} - \mathrm{diag}\,(\mathbf{I}_\mathrm{C})) \cdot \mathbf{1} \in \mathbb{R}^{N\times 1} \tag{3.56}$$

Since the actual circulating currents are unknown at the beginning of the modeling step—when these calculations must be performed—the implementation shown in Appendix 5 uses the circulating currents from the previous modeling step. The equation for $\mathrm{K_{L_i}}$ becomes now:

$$\mathrm{K_{L_i}}(n) = \begin{cases} \dfrac{I_{\mathrm{maxL}_i}}{I_{\mathrm{nom}_i}} & \text{if } \mathrm{K^*_{RegL_i}}(n) + \dfrac{I_{\Sigma\mathrm{CCL}_i}(n-1)}{I_{\mathrm{nomL}_i}} \geq \dfrac{I_{\mathrm{maxL}_i}}{I_{\mathrm{nomL}_i}}, \\ \dfrac{I_{\mathrm{minL}_i}}{I_{\mathrm{nom}_i}} & \text{if } \mathrm{K^*_{RegL_i}}(n) + \dfrac{I_{\Sigma\mathrm{CCL}_i}(n-1)}{I_{\mathrm{nomL}_i}} \leq \dfrac{I_{\mathrm{minL}_i}}{I_{\mathrm{nomL}_i}}, \\ \mathrm{K^*_{RegL_i}}(n) & \text{otherwise,} \end{cases} \tag{3.57}$$

where I_{maxL_i} and I_{minL_i} are the load's maximum and minimum allowed outputs current limits.

The code in Appendix 5 uses the DC voltage measurement errors from Table 6. Figure 46 presents modeling results consisting the load feeding or DC regulating currents, with the specified measurement errors but otherwise

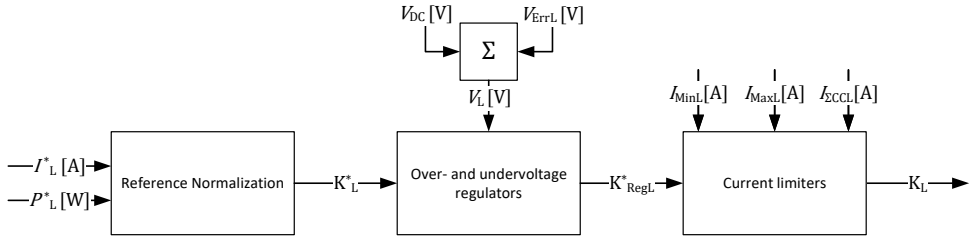


Figure 45. Load reference chain with current limiter taking into account the voltage measurement error and circulating currents.

Table 6. Static DC voltage measurement errors for modeled load and sources.

Voltage Source	Measurement Error
Source 1	$V_{\mathrm{errS}_1} = +1.0\,\mathrm{V}$
Source 2	$V_{\mathrm{errS}_2} = +0.5\,\mathrm{V}$
Source 3	$V_{\mathrm{errS}_3} = -1.0\,\mathrm{V}$
Source 4	$V_{\mathrm{errS}_4} = \pm 0.0\,\mathrm{V}$
Load 1	$V_{\mathrm{errL}_1} = -1.5\,\mathrm{V}$

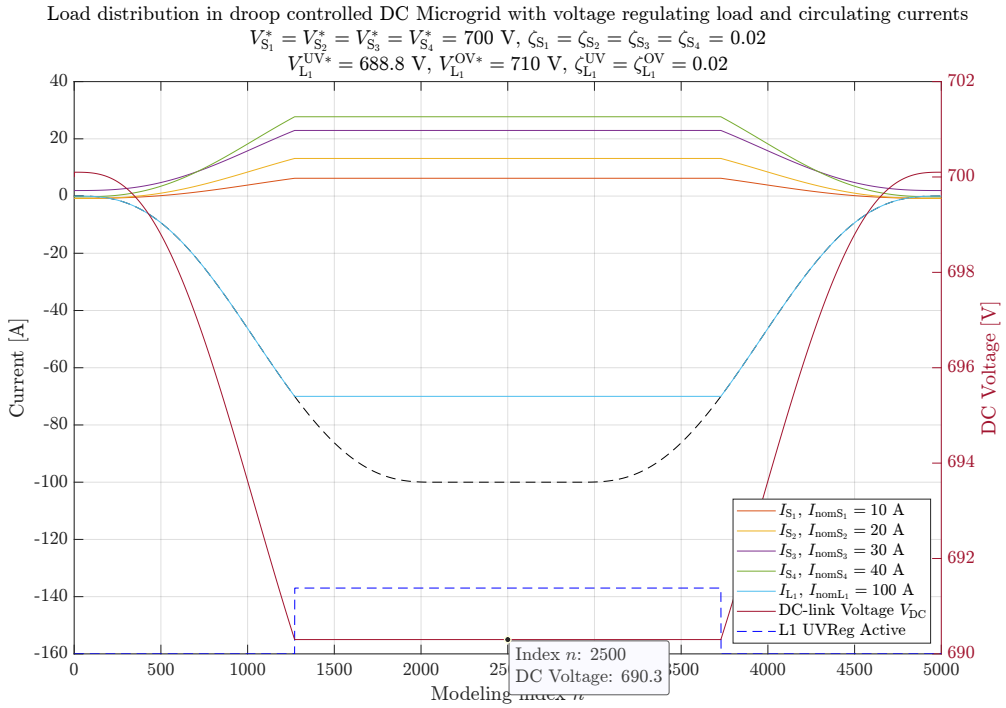


Figure 46. System load feeding currents from modeling with a load running against its undervoltage regulator with circulating currents.

similar equipment operational parameters as in Figure 35.

Due to the $-1.5\,\mathrm{V}$ DC voltage measurement error specified in Table 6 for the load, its control logic senses a DC voltage that is $1.5\,\mathrm{V}$ lower than the actual value. Consequently, the undervoltage regulator activates at an actual DC voltage of $690.3\,\mathrm{V}$ instead of the specified activation level of $688.8\,\mathrm{V}$, corresponding to an upward shift of $1.5\,\mathrm{V}$. The inverters' total currents shown in Figure 47 and listed in Table 7 illustrate how this measurement error influences the load's total output current. In the modeling step $n = 2500$, the load provides a voltage-regulating current of $-70.00\,\mathrm{A}$, while the measurable total current is only $-62.06\,\mathrm{A}$ (negative values indicating current flowing to the load from the DC link). This reduction in total current magnitude results

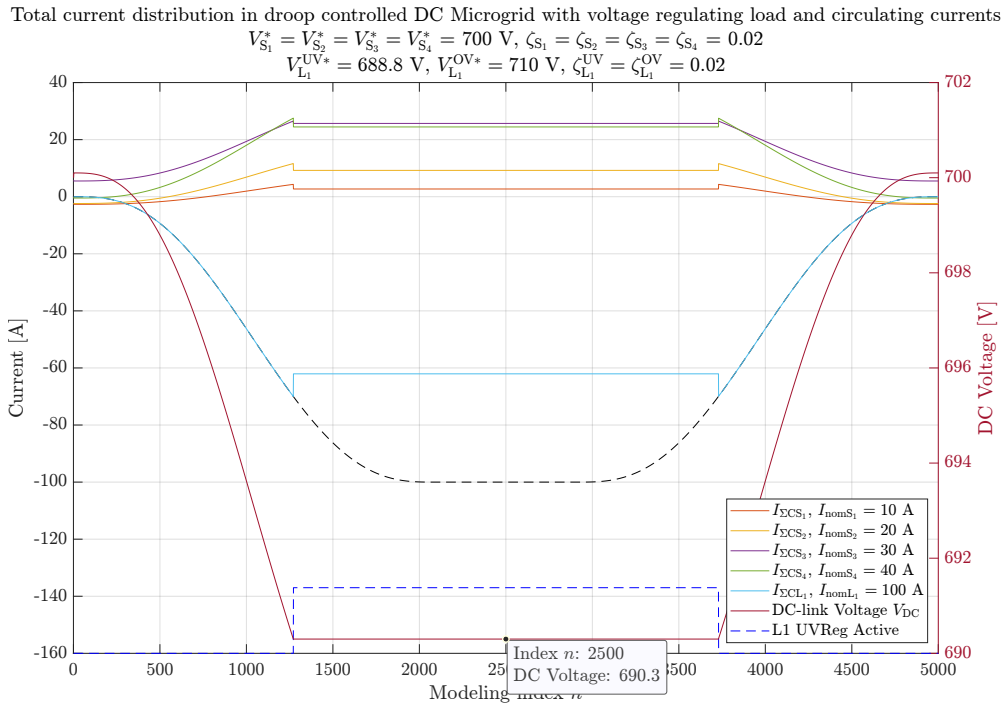


Figure 47. System total currents from modeling with a load running against its undervoltage regulator with circulating currents.

Table 7. The modeled total currents at modeling index 2500 in Figure 47.

Currents at Index 2500										
$\mathbf{I}_C =$	6.21 A	−0.24 A	−1.07 A	−0.57 A	−1.63 A	$\mathbf{I}_{\Sigma C} =$	2.71 A	$\mathbf{I} =$	6.21 A
	0.24 A	13.14 A	−1.29 A	−0.48 A	−2.39 A		9.23 A		13.14 A
	1.07 A	1.29 A	22.93 A	1.22 A	−0.83 A		25.68 A		22.93 A
	0.57 A	0.48 A	−1.22 A	27.71 A	−3.09 A		24.45 A		27.71 A
	1.63 A	2.39 A	0.83 A	3.09 A	−70.00 A		−62.06 A		−70.00 A

from the measurement error and requires the voltage sources to compensate in order to maintain the DC voltage. This can be seen as the positive circulating current values in the load's $\mathbf{I}_C$ row, indicating the currents are flowing towards the load. Because the load current is negative and its total current magnitude is smaller than its voltage-regulating current magnitude, it means that the voltage sources can subsequently reduce their total output current compared to before the voltage regulator was active.

3.1.8.2 Modeling Loads' Exceptional Situations

When a load's output reaches its current limiter or is stopped while operating in current-controlled mode, the behavior is straightforward. Only K_L is limited; the operation mode remains unchanged, the R_{MG} value is unaffected, no circulating currents are present for the load, and the total current consists purely of load-feeding or "DC regulating" current, i.e., $I_{\Sigma CL_1} = I_{L_1}$. Figure 48 presents a case modeled using the code from Appendix 5, with modifications where the load is first limited to 50% of its nominal current and then stopped at modeling index $n = 2500$.

Stopping a load while it is operating in voltage-control mode is equally uneventful, as shown in Figure 49. In this situation however the circulating currents and R_{MG} are affected at the moment of stopping. The circulating currents between the now stopped load and other equipment must be zeroed in the current matrix and B_{L_1} value updated accordingly in order for the droop resistance to be excluded from R_{MG} calculation.

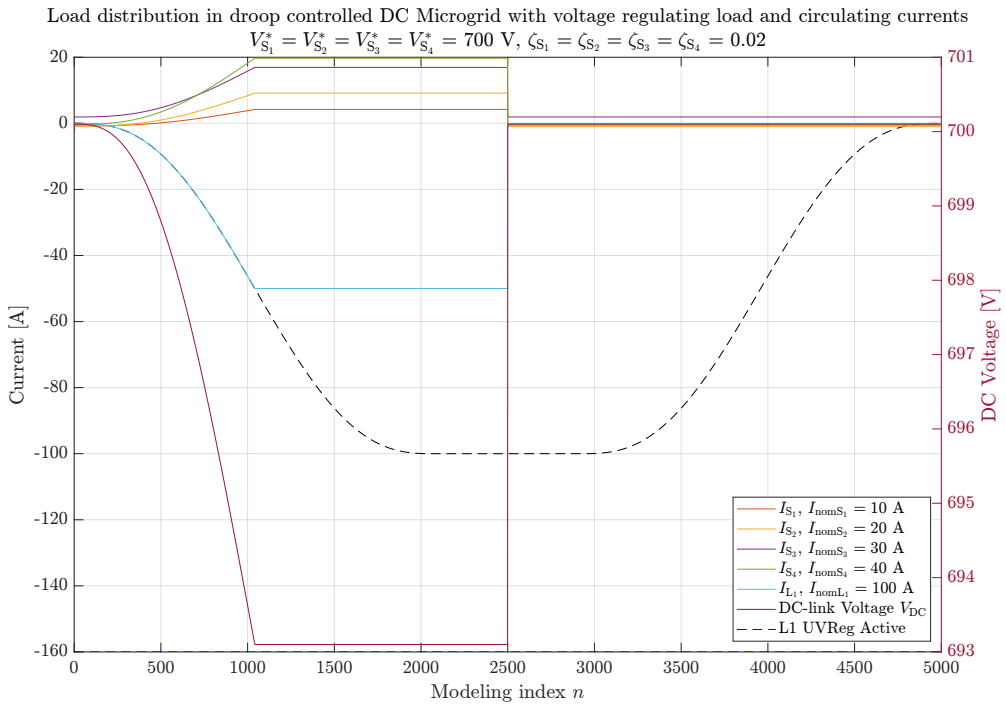


Figure 48. System behavior when current controlled load with measurement error induced circulating currents is limited and stopped.

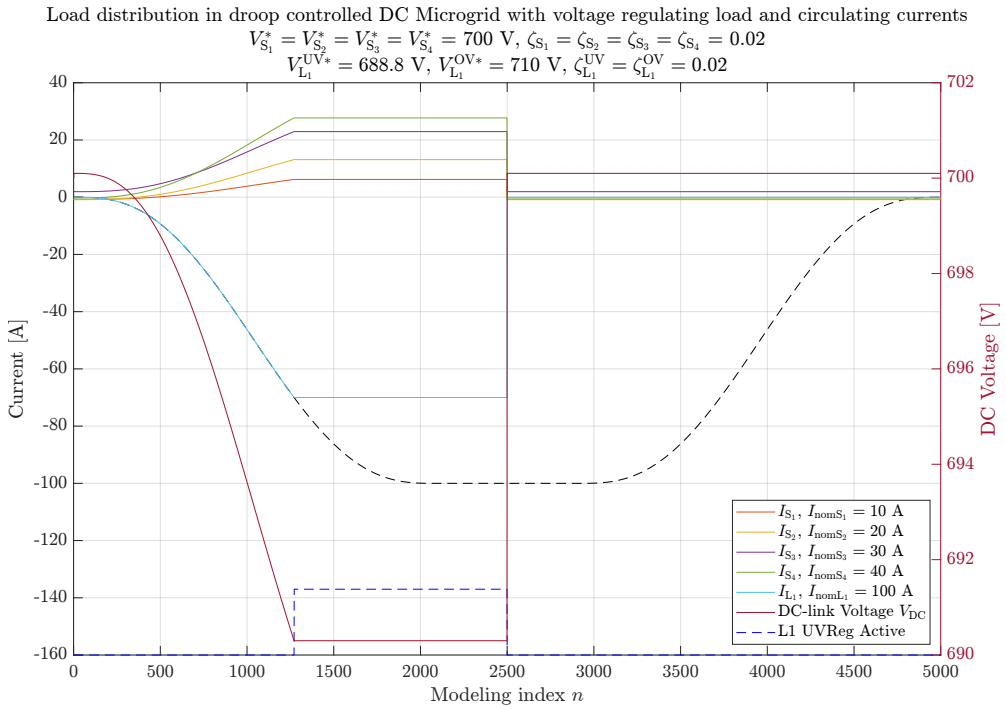


Figure 49. System behavior when voltage controlled load with measurement error induced circulating currents is stopped.

When a load is limited during voltage-control operation, the limiter only engages when the total current reaches the limit. Reaching the limit with only the DC voltage-regulating current does not trigger the limiter. For example, in the modeled case given in Table 7, the DC voltage-regulating current is $-70.00\,\text{A}$, while the total current is $-62.06\,\text{A}$. If the current limit is set to $-65.00\,\text{A}$ during voltage-control operation, the inverter continues operating normally until the voltage regulator deactivates, as illustrated in Figure 50. Conversely, if the limit is set before the undervoltage regulator activates, the voltage regulator never engages, as shown in Figure 51.

3.1.9 Rigid Voltage Sources

In some DC microgrid applications, rigid voltage sources—i.e., voltage sources without drooping—are advantageous. One such example is presented in Alho and Laaksonen (2020), where one power source is loaded to its full capacity before a second source begins supplying power. If the load continues to increase after the first source reaches 100% output, the second source supplies the remaining demand without sharing it with the first source. In other

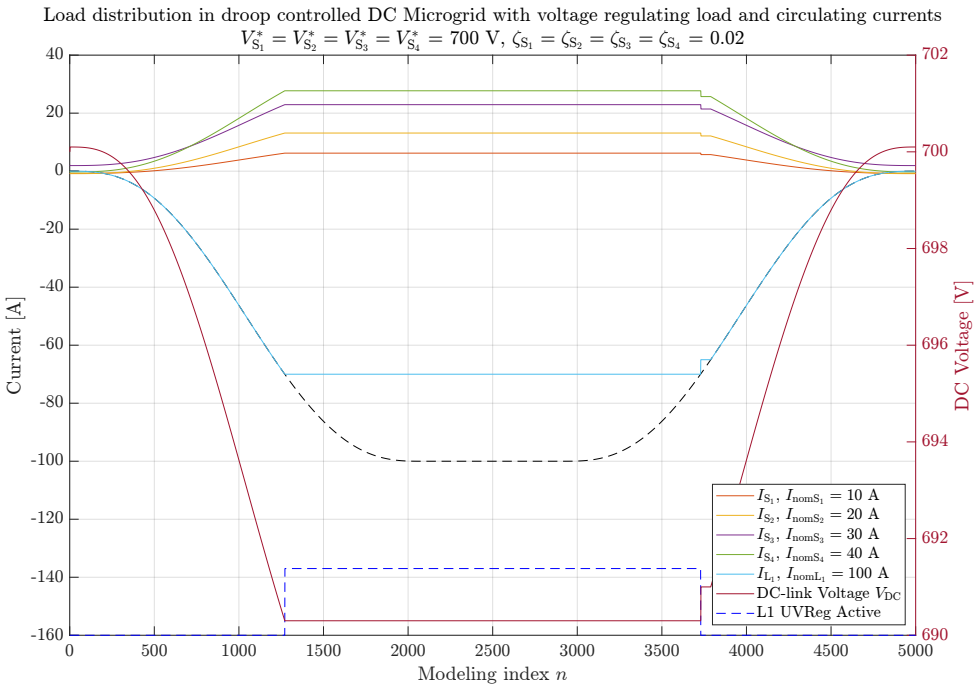


Figure 50. System behavior when voltage controlled load with measurement error induced circulating currents is limited to -65.00 A at modeling index $n = 2500$.

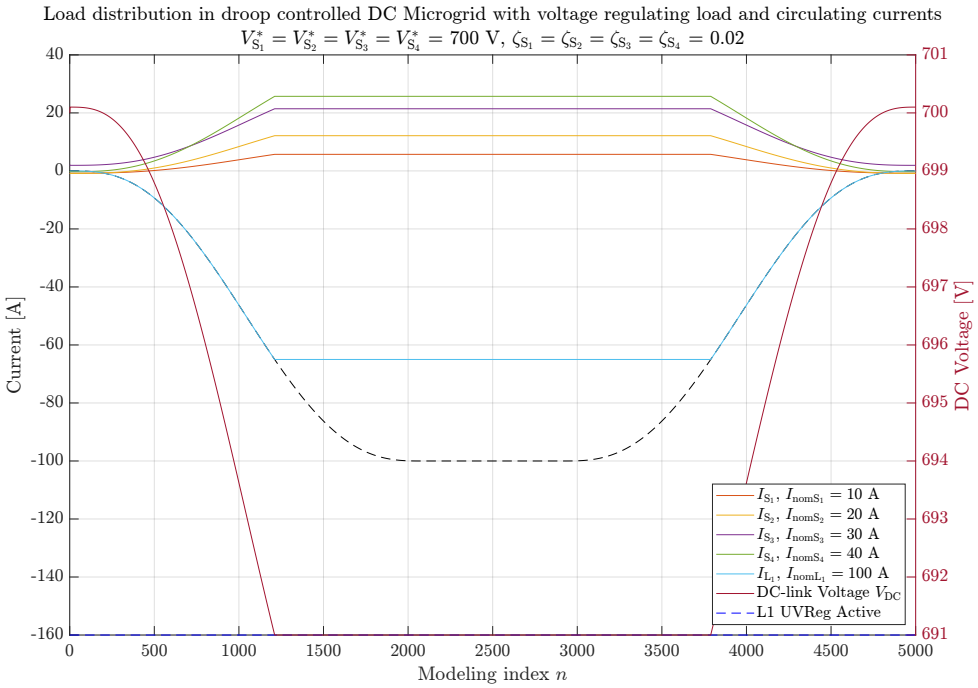


Figure 51. System behavior when voltage controlled load with measurement error induced circulating currents is limited to -65.00 A constantly.

words, the control strategy enforces prioritization rather than load sharing: the first source takes the load up to predetermined value, after which the second source handles any excess.

In the cited example, the second power source uses its undervoltage regulator to determine when to begin contributing. Because the regulator operates without droop, the second source rigidly maintains the DC-link voltage at its activation threshold. This arrangement has the benefit that no power or current limiters are required on the first source to control current flow.

Rigid voltage sources can be modeled in various ways, ranging from implementing PI controllers that replicate the equipment's internal control, to a simple direct assignment of the load current to the source in the mathematical model. For example, if the total load is $50\,\mathrm{A}$, the rigid voltage source is forcefully assigned to supply $50\,\mathrm{A}$. In this work, where the focus is on DC microgrid control strategy rather than the detailed internal control of inverters, the latter, simplified method is adopted.

3.1.10 DC Microgrid Nodal Analysis

In mechanically large systems, i.e., systems involving larger distances, the cable resistances between the equipment become non-negligible and will cause noticeable voltage drops between the devices. In this situation the single scalar value for DC voltage is no longer sufficient and voltages at each device DC terminals are needed. There are a number of different strategies utilized to model these large systems, like graph theory which, was employed by Abdolmaleki, Babak and Bergna-Diaz, Gilbert (2022). To continue with the matrix-based modeling that the paradigm covered in this work has been utilizing, DC microgrid voltages can be modeled by utilizing nodal analysis method.

In nodal analysis, the circuit under study is first transformed into an equivalent form containing only current sources, as illustrated earlier in Figure 15, but here with non-negligible cable resistances between each source and load. In the traditional formulation of nodal analysis, a reference node is selected and assigned a potential of zero volts. The unknowns are then the node potentials of all other nodes, expressed relative to this reference. These

reference-based node potentials are commonly referred to as "node voltages" in circuit analysis. If the voltage between two arbitrary nodes is required, it is obtained as the difference between their respective node voltages.

In this work, the modeling approach follows the principles of real-world shipboard power systems, which—except under special circumstances—operate without grounding the DC potentials. Accordingly, the DC microgrid is assumed to be floating, and all equipment terminal voltages are computed as differences between the node potentials at their positive and negative terminals. This avoids introducing an artificial reference potential not present in the physical system. Future development may include configurations where, for example, the negative DC bus is grounded at a specific location through capacitors; in such cases, the corresponding ground potential can be explicitly incorporated into the model.

While the letter V is traditionally used in electrical engineering to denote potentials, in this work the node potentials are denoted by Υ and collected into the potential array $\boldsymbol{\Upsilon}$, in order to clearly distinguish them from voltages. Here, Υ_i denotes the absolute (floating) potential of node i. Accordingly, the voltage across equipment x connected between nodes i and j is given by

$$V_x = \Upsilon_i - \Upsilon_j. \tag{3.58}$$

The basic principle in the matrix-based calculation remains the same as in Equation (3.6), just now the node potential array is calculated instead of a scalar value for DC voltage:

$$\boldsymbol{\Upsilon}(n) = \boldsymbol{\Upsilon}(n-1) + \mathbf{G}_{\text{nodes}}^{-1}(n)\mathbf{I}_{\text{nodes}}(n). \tag{3.59}$$

However, in order to keep the mathematical principle exactly the same as in Equation (3.6), where the source currents are from the previous time step, the $\mathbf{I}_{\text{nodes}}(n)$ needs to be rewritten to accommodate that. And due to no reference node being selected in the analysis, the conductance matrix is singular, thus instead of normal inverse the Moore–Penrose pseudoinverse of the conductance matrix $\mathbf{G}_{\text{nodes}}^{+}$ has to be calculated:

$$\boldsymbol{\Upsilon}(n) = \boldsymbol{\Upsilon}(n-1) + \mathbf{G}_{\text{nodes}}^{+}(n)\left(\mathbf{I}_{\text{nodes}}^{(L)}(n) + \mathbf{I}_{\text{nodes}}^{(S)}(n-1)\right). \tag{3.60}$$

Above, $\boldsymbol{\Upsilon}$ is the $1 \times N$ node potential array, containing the individual node potentials of the system $\boldsymbol{\Upsilon} = \begin{bmatrix} \Upsilon_a & \Upsilon_b & \Upsilon_c & \dots \end{bmatrix}^{\mathrm{T}}$, and $\mathbf{I}_{\text{nodes}}$ is a $1 \times N$ current array containing the nodes' currents $\mathbf{I}_{\text{nodes}} = \begin{bmatrix} I_a & I_b & I_c & \dots \end{bmatrix}^{\mathrm{T}}$. $\mathbf{G}^{+}_{\text{nodes}}$ is the $N \times N$ Moore–Penrose pseudoinverse of the conductance matrix containing the equipment's internal and cable conductances.

The use of $\mathbf{G}^{+}_{\text{nodes}}$ instead of $\mathbf{G}^{-1}_{\text{nodes}}$ is required because, in a floating DC microgrid without a defined reference node, the conductance matrix is rank-deficient and does not have a unique inverse. In a floating DC microgrid without a designated reference node, the conductance matrix $\mathbf{G}_{\text{nodes}}$ is singular because its rows sum to zero, making it non-invertible in the usual sense; thus, the Moore–Penrose pseudoinverse $\mathbf{G}^{+}_{\text{nodes}}$ must be used to obtain a unique least-squares solution for the node potentials. In MATLAB the Moore–Penrose pseudoinverse is calculated with the `pinv(...)` function.

The formulation in Equation (3.60) also makes it easier to replace, in the MATLAB code, the DC voltage modeling based on Equation (3.6). Since the basic building blocks used to calculate the DC voltages in Equation (3.60) are the same as in the single-scalar case of Equation (3.6), the software-architectural change from scalar to nodal analysis is straightforward: replace the scalar DC voltage calculation with the nodal-analysis-based implementation. Other required modifications are (i) to assign inverter-specific voltage values instead of the single scalar value used previously, and (ii) circulating currents cannot be handled the same way as with ideal cables and the Γ_{C} mus be omitted.

In nodal analysis with non-negligible cable resistances, the handling of DC voltage measurement error-induced circulating currents differs fundamentally from the ideal-cable case. In the ideal-cable formulation, the Γ_{C} matrix can be used because all voltage sources are connected to the same node potential, allowing direct computation of source-to-source circulating current contributions. When cable resistances are non-zero, each source terminal has its own node potential, and Γ_{C} can no longer be constructed without first solving the complete nodal system.

Mathematically, in the ideal case all source terminals share the same actual potential, so the source-to-source circulating current between sources i and

j can be expressed simply as

$$I_{\mathrm{CS}_i\mathrm{S}_j} = \frac{V^*_{\mathrm{S}_i} - V^*_{\mathrm{S}_j}}{R_{\mathrm{DS}_i} + R_{\mathrm{DS}_j}}, \tag{3.61}$$

as was done by P., Monica and M., Kowsalya and Guerrero, Josep M. (2021), allowing the entire set of circulating currents to be precomputed in matrix form in Equation (3.22). With non-ideal cables, each V_{S_i} is naturally different and depends on the solution of the nodal equations, so $I_{\mathrm{CS}_i\mathrm{S}_j}$ cannot be determined a priori.

In this case, the measurement error V_{errS_i} is incorporated directly into the per-source Γ-function as

$$\Gamma_{\mathrm{S}_i} = \frac{V^*_{\mathrm{S}_i} - (V_{\mathrm{S}_i} + V_{\mathrm{errS}_i})}{\zeta_{\mathrm{S}_i} V^*_{\mathrm{S}_i}}, \tag{3.62}$$

and the Γ_{C} matrix is omitted. The source output current array $\mathbf{I}_{\mathrm{S}}$ with maximum and minimum current limits $\mathbf{I}_{\mathrm{maxS}}$ and $\mathbf{I}_{\mathrm{minS}}$ and running status $\mathbf{B}_{\mathrm{runS}}$ then follows from Equation (3.4) as

$$\mathbf{I}_{\mathrm{S}} = \max\left(\min\left(\mathrm{diag}(\boldsymbol{\Gamma}_{\mathrm{S}})\mathbf{I}_{\mathrm{nomS}}, \mathbf{I}_{\mathrm{maxS}}\right), \mathbf{I}_{\mathrm{minS}}\right) \circ \mathbf{B}_{\mathrm{runS}}. \tag{3.63}$$

The local inclusion of the DC voltage measurement error in Equation (3.62) ensures that each source's output is directly affected by its own measurement deviation, and indirectly affects other sources through the coupling of node potentials in the conductance matrix. Thus each element in $\mathbf{I}_{\mathrm{S}}$ effectively corresponds with the total current value from the models with ideal cables. If exact source-to-source circulating currents are required, they must be calculated separately from the nodal analysis solution.

3.1.10.1 Implementing Nodal Analysis to System Modeling

To implement nodal analysis for a system similar to the previous case of four voltage sources and a single load, the cable resistances between the equipment must be defined. Figure 52 presents the system circuit diagram with $R_{\mathrm{C}_{i+}}$ and $R_{\mathrm{C}_{i-}}$ for cable resistances on positive and negative potentials re-

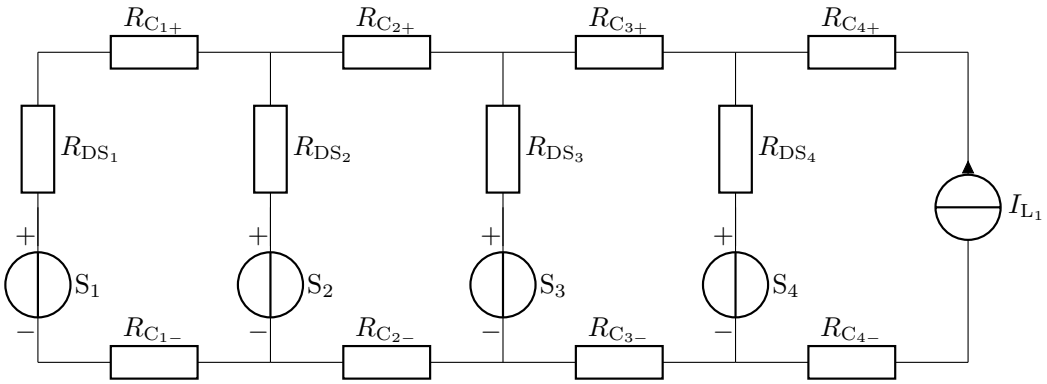


Figure 52. Circuit diagram of the system with cable resistances in between the equipment.

spectively. In this scenario all cables are assumed to have non-negligible resistances.

As noted earlier, a system analyzed using nodal analysis must be represented solely in terms of current sources. Therefore, the voltage sources are converted to their Norton equivalents with the currents I_{x_i} as their outputs, as shown in Figure 53. As the circulating currents cannot be calculated now with the Γ_C and measurement errors are taken into account locally in the current calculation for each source, the I_{x_i} values effectively are the total currents of the inverters, including also the circulating currents.

Also, since nodal analysis operates with conductances, the cable resistances are converted to conductances. Furthermore, as the load is capable of operating as droop-controlled voltage source through its voltage regulators, the load's internal droop conductance is also depicted to clarify the construction of the conductance matrix. The individual nodes are also marked in Figure 53.

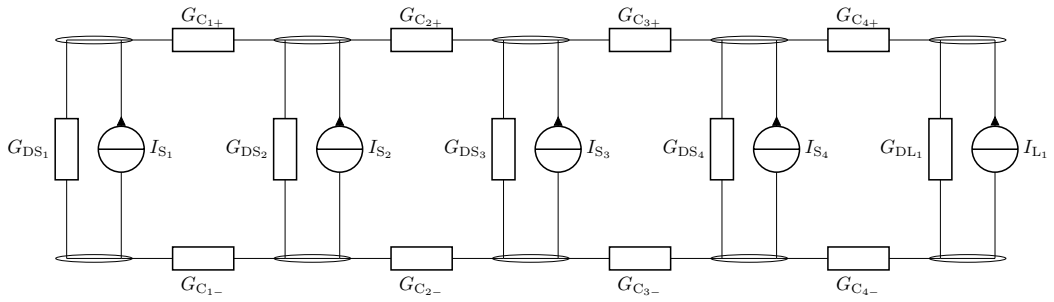


Figure 53. Circuit diagram with modeled cables and nodes marked for nodal analysis.

Here, the equipment effective droop conductances can be calculated with the same Boolean arrays $\mathbf{B}_{\mathrm{V}}$ and $\mathbf{B}_{\mathrm{run}}$ used with R_{MG} calculation in Equation (3.12):

$$G_{\mathrm{D}x_i} = \frac{\mathrm{B}_{\mathrm{V}x_i} \wedge \mathrm{B}_{\mathrm{run}x_i}}{R_{\mathrm{D}x_i}}. \tag{3.64}$$

Where the Boolean values allow automatic enabling and disabling of the droop conductances as required by the operating state, for example, when a source is stopped or current-limited. For sources the droop resistances are calculated as in Equation (3.7) and for loads as:

$$R_{\mathrm{DL}_i} = \begin{cases} \dfrac{\zeta_{\mathrm{L}_i}^{\mathrm{OV}} V_{\mathrm{L}_i}^{\mathrm{OV}*}}{I_{\mathrm{nomL}_i}}, & \text{when overvoltage regulator is active,} \\ \dfrac{\zeta_{\mathrm{L}_i}^{\mathrm{UV}} V_{\mathrm{L}_i}^{\mathrm{UV}*}}{I_{\mathrm{nomL}_i}}, & \text{when undervoltage regulator is active,} \\ *, & \text{otherwise.} \end{cases} \tag{3.65}$$

As the load inverter is not voltage regulating device when the regulator is deactivated, and thus its corresponding $\mathbf{B}_{\mathrm{V}}$ value will be zero in Equation (3.64), the non-voltage regulating value for R_{DL_i} in Equation (3.65) is irrelevant.

To construct the conductance matrix and current array from the circuit diagram in Figure 53, a consistent node labeling must first be defined. Figure 54 shows the node labeling scheme adopted in this work. This order of labels must be followed exactly when assembling both the conductance matrix and the current array; any mismatch would result in an incorrect node potential array Υ when applying Equation (3.60).

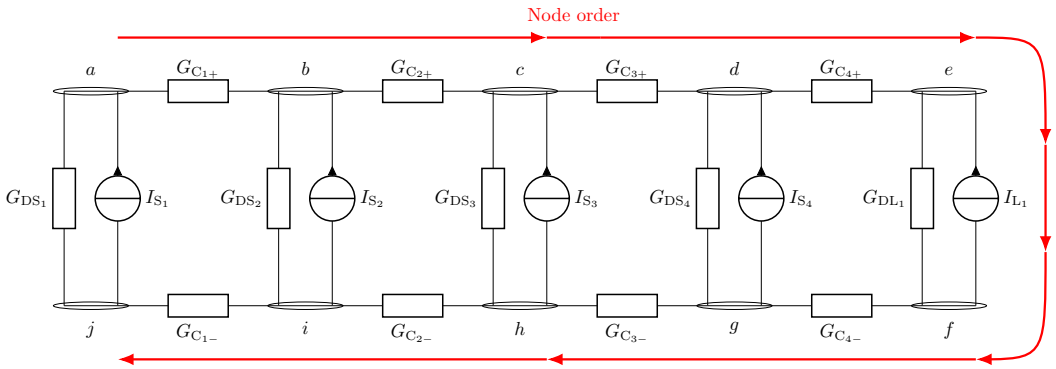


Figure 54. Circuit diagram with modeled cables and assigned nodes for nodal analysis.

The conductance matrix corresponding to the node numbering in Figure 54 is given in Appendix 6, and the node current array for modeling step n is:

$$\mathbf{I}_{\text{nodes}}(n) = \begin{bmatrix} I_a(n) & I_b(n) & I_c(n) & I_d(n) & I_e(n) & I_f(n) & I_g(n) & I_h(n) & I_i(n) & I_j(n) \end{bmatrix}^{\text{T}}$$
$$= \begin{bmatrix} I_{\text{S}_1}(n-1) + G_{\text{C}_{1+}} \left(\Upsilon_b(n-1) - \Upsilon_a(n-1)\right) \\ I_{\text{S}_2}(n-1) + G_{\text{C}_{2+}} \left(\Upsilon_c(n-1) - \Upsilon_b(n-1)\right) + G_{\text{C}_{1+}} \left(\Upsilon_a(n-1) - \Upsilon_b(n-1)\right) \\ I_{\text{S}_3}(n-1) + G_{\text{C}_{3+}} \left(\Upsilon_d(n-1) - \Upsilon_c(n-1)\right) + G_{\text{C}_{2+}} \left(\Upsilon_b(n-1) - \Upsilon_c(n-1)\right) \\ I_{\text{S}_4}(n-1) + G_{\text{C}_{4+}} \left(\Upsilon_e(n-1) - \Upsilon_d(n-1)\right) + G_{\text{C}_{3+}} \left(\Upsilon_c(n-1) - \Upsilon_d(n-1)\right) \\ I_{\text{L}_1}(n) + G_{\text{C}_{4+}} \left(\Upsilon_d(n-1) - \Upsilon_e(n-1)\right) \\ -I_{\text{L}_1}(n) + G_{\text{C}_{4-}} \left(\Upsilon_g(n-1) - \Upsilon_f(n-1)\right) \\ -I_{\text{S}_4}(n-1) + G_{\text{C}_{4-}} \left(\Upsilon_f(n-1) - \Upsilon_g(n-1)\right) + G_{\text{C}_{3-}} \left(\Upsilon_h(n-1) - \Upsilon_g(n-1)\right) \\ -I_{\text{S}_3}(n-1) + G_{\text{C}_{3-}} \left(\Upsilon_g(n-1) - \Upsilon_h(n-1)\right) + G_{\text{C}_{2-}} \left(\Upsilon_i(n-1) - \Upsilon_h(n-1)\right) \\ -I_{\text{S}_2}(n-1) + G_{\text{C}_{2-}} \left(\Upsilon_h(n-1) - \Upsilon_i(n-1)\right) + G_{\text{C}_{1-}} \left(\Upsilon_j(n-1) - \Upsilon_i(n-1)\right) \\ -I_{\text{S}_1}(n-1) + G_{\text{C}_{1-}} \left(\Upsilon_i(n-1) - \Upsilon_j(n-1)\right) \end{bmatrix}. \tag{3.66}$$

As stated in Equation (3.60), the voltage source currents are taken from the previous time step $(n-1)$, whereas the load currents are taken from the current time step n. This follows directly from the order of operations in the modeling algorithm: the load outputs are calculated first, then the DC node potentials are updated, and only after that are the source outputs computed.

The equipment droop conductances $G_{\text{D}x_i}$ influence the node potentials during voltage-controlled operation and must therefore be included in the conductance matrix. However, the corresponding branch currents through $G_{\text{D}x_i}$ are already represented in the I_{x_i} value of each device. Including them again in the node current array would double-count the same physical contribution. Consequently, the node current array contains only the inverter currents and the contributions from the cable conductances.

The node potential array $\boldsymbol{\Upsilon}$ is then calculated using Equation (3.60), with the conductance matrix from Appendix 6 and the current array from Equation (3.66). From $\boldsymbol{\Upsilon}$, the individual DC voltages across each piece of equipment can be obtained by taking the potential difference between its two connection nodes as shown in Figure 54:

$$\begin{aligned} \boldsymbol{\Upsilon} &= \begin{bmatrix} \Upsilon_a & \Upsilon_b & \Upsilon_c & \Upsilon_d & \Upsilon_e & \Upsilon_f & \Upsilon_g & \Upsilon_h & \Upsilon_i & \Upsilon_j \end{bmatrix}^{\mathrm{T}} \\ \Rightarrow \mathbf{V}_{\mathrm{DC}} &= \begin{bmatrix} V_{\mathrm{S}_1} \\ V_{\mathrm{S}_2} \\ V_{\mathrm{S}_3} \\ V_{\mathrm{S}_4} \\ V_{\mathrm{L}_1} \end{bmatrix} = \begin{bmatrix} \Upsilon_a - \Upsilon_j \\ \Upsilon_b - \Upsilon_i \\ \Upsilon_c - \Upsilon_h \\ \Upsilon_d - \Upsilon_g \\ \Upsilon_e - \Upsilon_f \end{bmatrix} \end{aligned} \tag{3.67}$$

3.1.10.2 Modeling System of Four Sources and a Single Load

To ensure stable operation and avoid divergence at modeling startup, the node potential array $\boldsymbol{\Upsilon}$ must be assigned reasonable initial values. If all node potentials were initialized to zero, the resulting equipment terminal voltages would also be zero, causing droop-controlled voltage sources to attempt injecting unrealistically large currents. This would lead to excessive node-voltage excursions in the subsequent step and numerical instability.

A practical initialization is:

1. Set each source's positive terminal to its voltage reference and its negative terminal to zero, so that the source initially sees its internal reference across its terminals

2. Set each load's negative terminal to zero, with the positive terminal initialized to a weighted average of the sources' references, this corresponds to a voltage level which would in ideal cable-case correspond with systems voltage at zero load.

Using this principle, for the example with four voltage sources and one load,

the initial values are

$$\boldsymbol{\Upsilon}(0) = \begin{bmatrix} \Upsilon_{a_0} \\ \Upsilon_{b_0} \\ \Upsilon_{c_0} \\ \Upsilon_{d_0} \\ \Upsilon_{e_0} \\ \Upsilon_{f_0} \\ \Upsilon_{g_0} \\ \Upsilon_{h_0} \\ \Upsilon_{i_0} \\ \Upsilon_{j_0} \end{bmatrix} = \begin{bmatrix} V^*_{\mathrm{S}_1} \\ V^*_{\mathrm{S}_2} \\ V^*_{\mathrm{S}_3} \\ V^*_{\mathrm{S}_4} \\ \dfrac{\sum (V^*_{\mathrm{S}} I_{\mathrm{nomS}})}{\sum I_{\mathrm{nomS}}} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 700\,\mathrm{V} \\ 700\,\mathrm{V} \\ 700\,\mathrm{V} \\ 700\,\mathrm{V} \\ 700\,\mathrm{V} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{3.68}$$

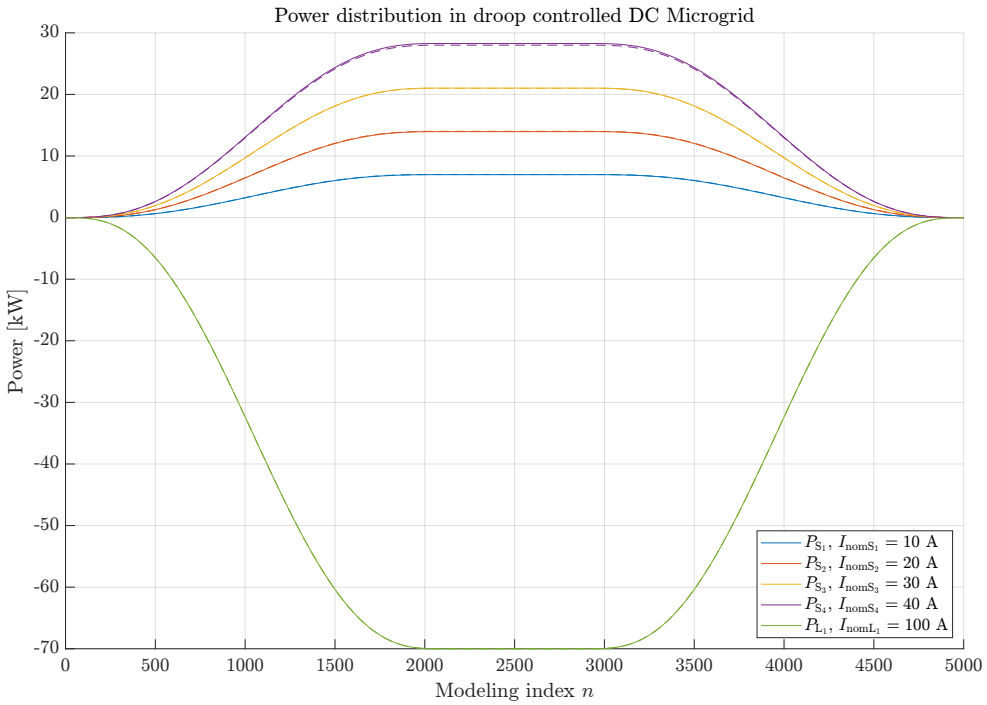
The modeled circuit is based on Figure 52, with cable resistances defined in Table 8. Figures 55a and 55b show the power distribution and DC voltages of the equipment, respectively, when the same varying proportional load profile as in the previous examples is applied. In this case, however, the load operates under a power reference rather than a current reference, in order to highlight the effects of cable resistance. In practice, loads are rarely purely current-controlled, and with non-ideal cables present, the voltage at each unit can be observed to drop more significantly as the load draws additional current to maintain its power reference.

In Figures 55a and 55b, the dashed lines represent the powers and DC voltage in the ideal-cable case. The powers differ only slightly, and the load's power is exactly the same as in power-control mode.

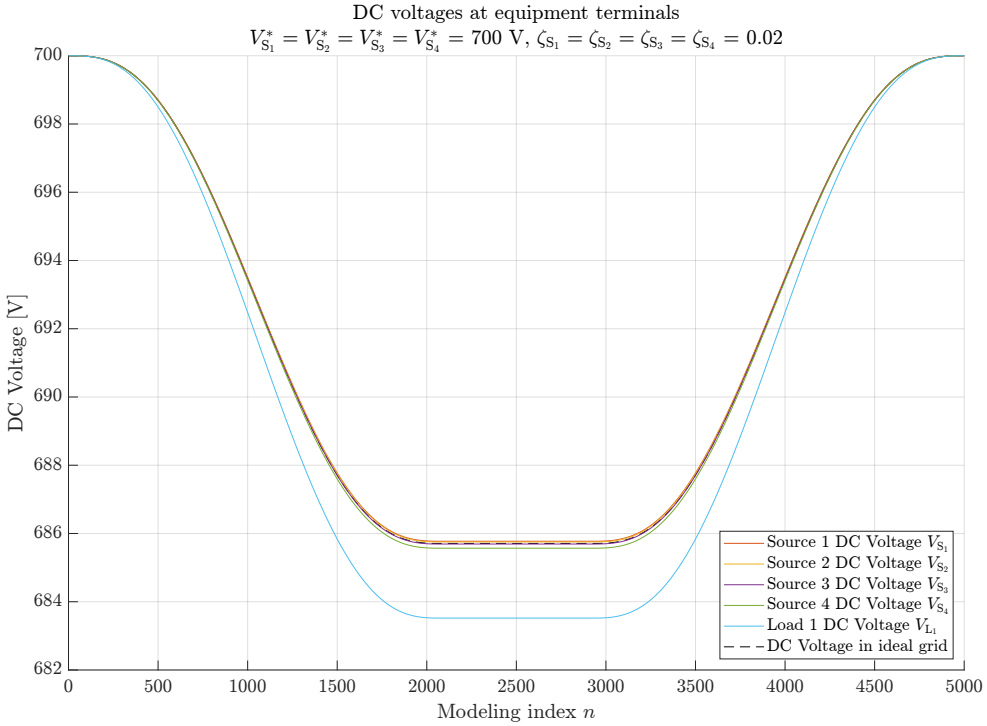
The DC voltage level for the load is considerably lower than that of the sources

Table 8. Modeled cable resistances and conductances.

Cable	Resistance	Conductance
C_{1+}	1 mΩ	1000 S
C_{2+}	1 mΩ	1000 S
C_{3+}	1 mΩ	1000 S
C_{4+}	10 mΩ	100 S
C_{1-}	1 mΩ	1000 S
C_{2-}	1 mΩ	1000 S
C_{3-}	1 mΩ	1000 S
C_{4-}	10 mΩ	100 S



(a) Power distribution in the modeled system with non-ideal cables, with dashed lines as reference from ideal model.



(b) DC voltages per inverter in the modeled system with non-ideal cables.

Figure 55. Results of nodal analysis made for system with four voltage sources and a single load.

due to the much larger cable resistance, as evident from Table 8. Figure 55b also shows that in the ideal-cable case the DC voltage closely follows the sources' voltage. This is expected: just as the load in power-control mode dictates the ideal power, the voltage sources in voltage-control mode dictate the ideal voltage.

Table 9 lists the powers and DC voltages at modeling index $n = 2500$, and also the sum of the equipment powers which is 219.31 W. As described earlier, positive device currents correspond to current flowing into the DC bus, while negative values correspond to current flowing out of it. The same sign convention applies to the powers: a positive device power denotes net export to the DC bus, and a negative value denotes net import. A positive sum of all device powers therefore indicates that the sources collectively produce more power than the load consumes. The surplus—219.31 W in this case—is dissipated as resistive losses in the cables.

Appendix 7 presents the MATLAB code to perform the nodal analysis for the system of four voltage sources and one load. The code supports the assignment of current limits, voltage regulators, starting and stopping of equipment and the load is parameterized to handle its proportional reference input as power reference.

Table 9. Powers and voltages at modeling index $n = 2500$ in Figures 55a and 55b.

Equipment	Powers	DC Voltages
S_1	6,969.20 W	685.77 V
S_2	13,957.90 W	685.75 V
S_3	21,024.66 W	685.69 V
S_4	28,267.55 W	685.57 V
L_1	−70,000.00 W	683.52 V
	$\sum P = 219.31$ W	

3.1.10.3 Including Load's Voltage Regulator

The cable resistance in front of the load directly affects the stability of the under- and overvoltage regulators, as well as the operation of the variable voltage-based control strategy as a whole, because it acts as additional droop resistance for the inverter. Nodal analysis with no regulators active in Figure 55b shows a considerable voltage drop at the terminals of load L_1 due

to the cable resistance, indicating a strong influence on the load's voltage regulator behavior.

In such cases, the voltage drop from long feeder cables must be taken into account when parameterizing the inverter to ensure the control strategy operates correctly. The drop shifts the output current or power at which the under- and overvoltage regulator activation limits, $V_{\mathrm{L}}^{\mathrm{UV}*}$ and $V_{\mathrm{L}}^{\mathrm{OV}*}$, are reached. In effect, the cable resistance adds to the load inverter's droop characteristic, modifying also its effective droop coefficient. Without including these effects in the inverter parameterization, the regulators may activate at unintended times, causing instability because the inverter operates with a substantially different droop characteristic than other voltage-supporting devices.

The presented modeling approach is therefore beneficial for predicting and preparing for such problems in systems with long cables, as it allows the combined effects of voltage drops and additional droop to be quantified and addressed already in the design phase.

Figures 56 and 57 show the power distribution and DC voltages for the case where the load operates with the undervoltage regulator active. The regulator parameters are identical to those in Figures 35 and 46, with an activation limit of $V_{\mathrm{L1}}^{\mathrm{UV}*} = 688.8\,\mathrm{V}$ and a droop coefficient $\zeta_{\mathrm{L1}}^{\mathrm{UV}} = 0.02$. For comparison, the figures also include dashed lines showing the corresponding results for an ideal grid with negligible cable resistance.

The undervoltage reference $V_{\mathrm{L1}}^{\mathrm{UV}*} = 688.8\,\mathrm{V}$ is set to correspond to the voltage level at which, in the ideal case, the sources would be loaded to 80% of their nominal capacity. The ideal case corresponding with the intended situation is visible in the Figure 56 as dashed lines. However, due to cable-induced voltage drops, the undervoltage regulator activates earlier than intended, thereby reducing the loading of the sources below the intended level.

Two simple adjustments are possible: (i) raise the sources' voltage references, or (ii) lower the load's activation limit. While these adjustments work for a single-load system, in a multi-load system the situation becomes more complex. Adjusting only the source references "calibrates" them to specific loading patterns, making each load's regulator activation dependent on the others.

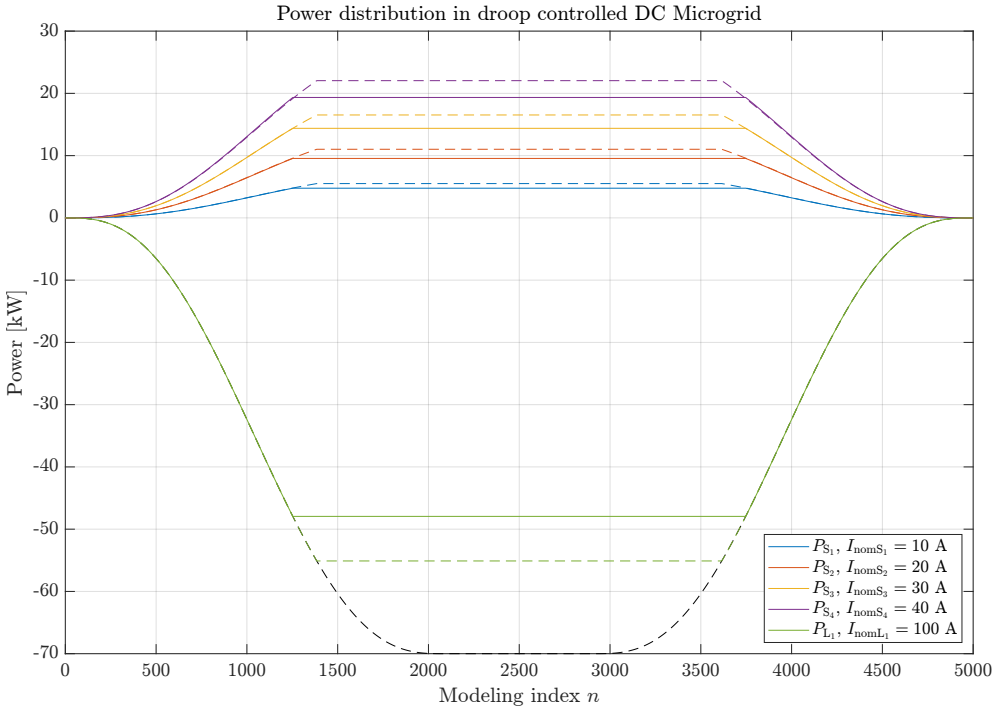


Figure 56. Power distribution in the modeled system with non-ideal cables and load undervoltage regulator.

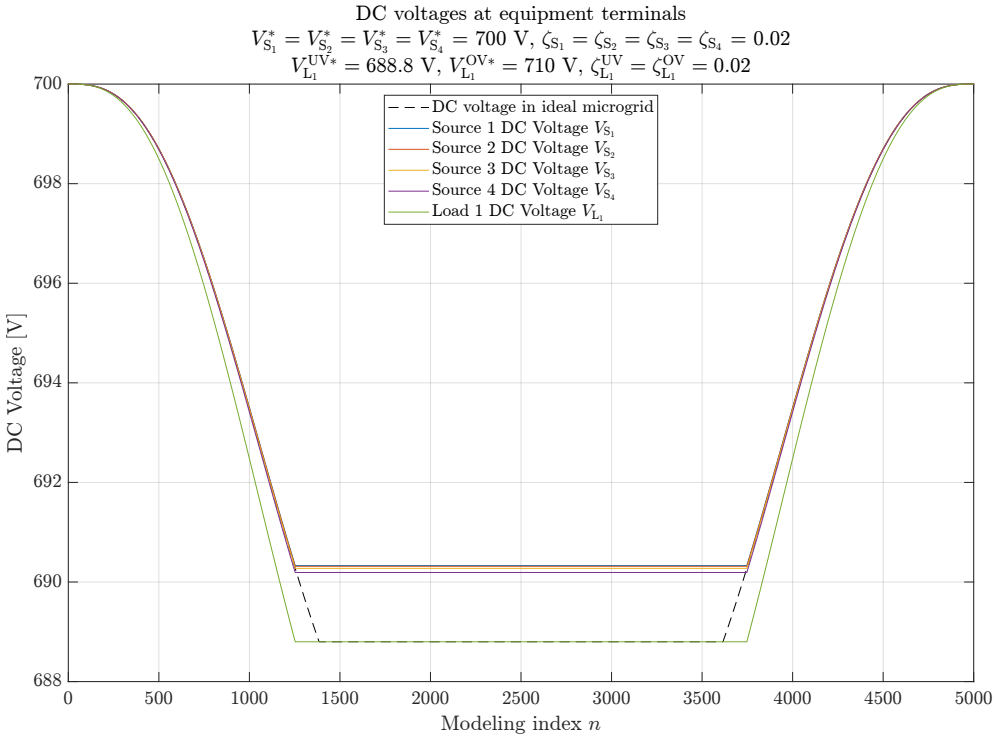


Figure 57. DC voltages per inverter in the modeled system with non-ideal cables and load undervoltage regulator.

A more robust approach is to calculate the cable voltage drops for each load and subtract these as offsets from the ideal-case activation limits. Apply corresponding offsets to both the source voltage references and the load activation limits so that the most distant load receives sufficient voltage in all scenarios. If the voltage difference between the sources and the most distant load remains too high, the load's limit must be increased above the ideal target, or the cable resistance must be reduced, for example by using thicker conductors.

This scenario can be reproduced with the MATLAB script in Appendix 7 by adjusting the undervoltage regulator activation limit $V_{\mathrm{L}_1}^{\mathrm{UV}*}$ to the desired value. Code 15 shows the two lines to modify in Appendix 7 to set the new activation limit.

```
UVRefs = [700*(1-0.02*0.8)];
% UVRefs = [100];
```

Code 15. MATLAB code snippet to change the load undervoltage regulator activation limit.

3.1.10.4 Modeling a System with Multiple Loads and Voltage Regulators

To study the effect of unequal cable lengths between loads, consider a system similar to that in Section 3.1.7.1: one source with a limited maximum output and two power-controlled loads equipped with undervoltage regulators; see Figure 58. The source output is explicitly limited to 100 A, and the load-side regulator parameters are identical to those used in Section 3.1.7.1. In this configuration the system is not at risk of blackout: cable-induced voltage drops trigger the regulators earlier than strictly necessary, which primarily reduces efficiency by curtailing source loading, rather than compromising security of supply. If the feeder resistances become excessively large, however, the operating point can become infeasible because the load terminal voltage remains too low.

To map the impact of cable length asymmetry, the same study as in Section 3.1.7.1 is repeated with the three sets of cable resistances given in Ta-

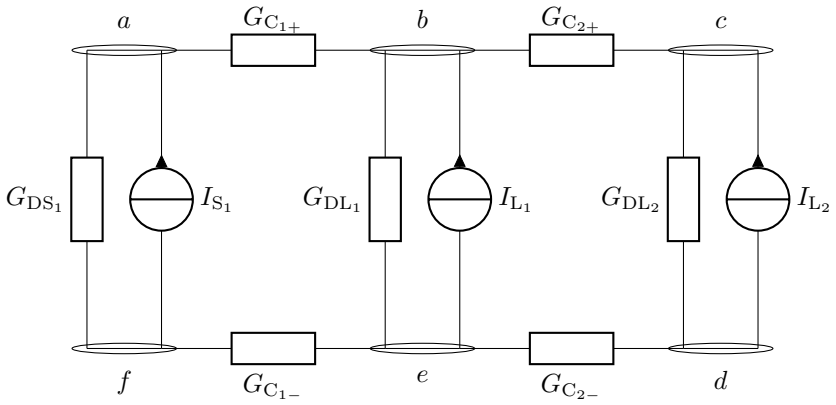


Figure 58. Circuit diagram with single source with limited output and two loads.

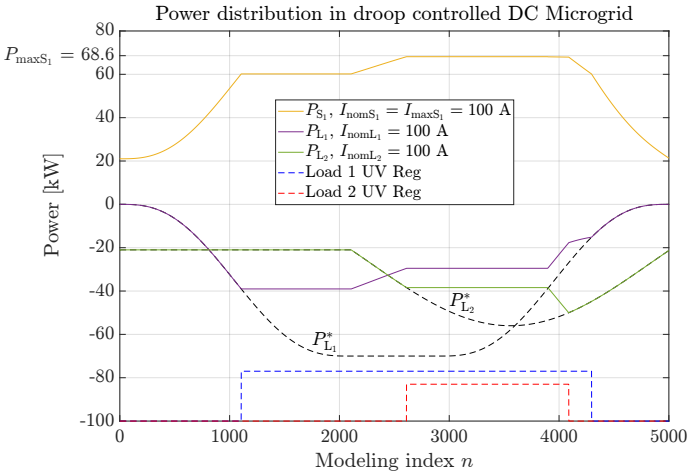
ble 10. Case 1 models loads that are close to each other but farther from the source. Case 2 places the source close to Load 1 while Load 2 is farther away. Case 3 sets all equipment at comparable electrical distance. The resulting powers and DC voltages are shown in Figures 59 and 60, where $P_{\mathrm{maxS_1}}$ corresponds with the source's practical maximum output power with assigned current limit and drooping:

$$\begin{aligned} P_{\mathrm{maxS_1}} &= I_{\mathrm{nomS_1}} V_{\mathrm{S_1}}^{*} \left(1 - \zeta_{\mathrm{S_1}}\right) \\ &= 100\,\mathrm{A} \cdot 700\,\mathrm{V} \cdot (1 - 0.02) \\ &= 68.60\,\mathrm{kW} \end{aligned} \tag{3.69}$$

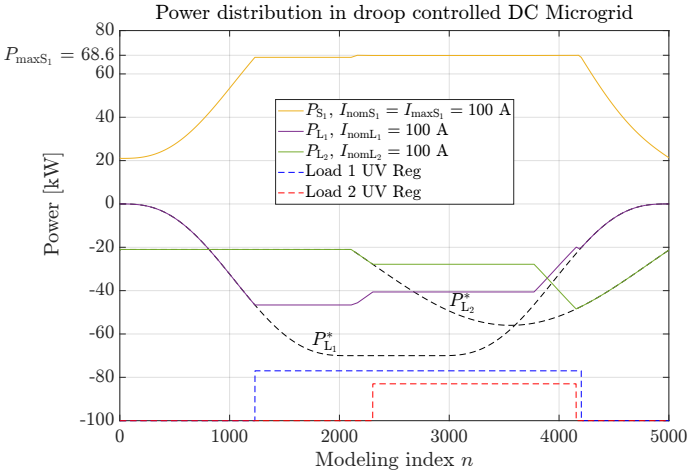
Appendix 8 presents the MATLAB code for performing the modeling with two loads and their undervoltage regulators active. The three cable resistance cases defined in Table 10 can be activated or deactivated by commenting or uncommenting the corresponding lines of code in the script. Code 16 shows these lines in more detail.

Table 10. Cable resistances and conductances used to study effects on voltage-regulator operation.

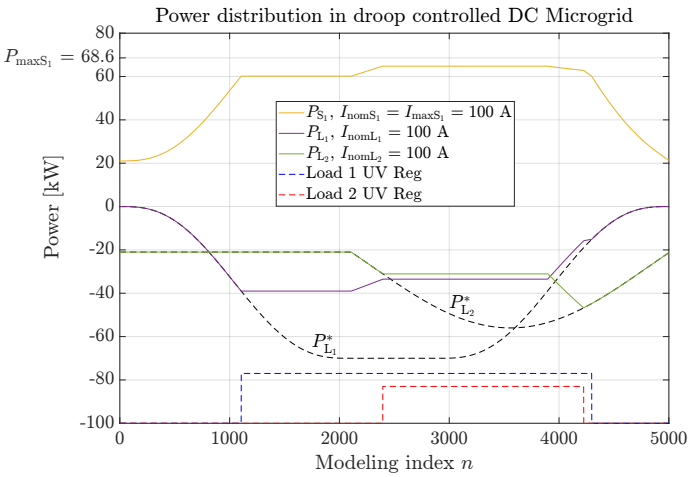
	Case 1		**Case 2**		**Case 3**	
Cable	R	G	R	G	R	G
$\mathrm{C_{1+}}$	10 mΩ	100 S	1 mΩ	1,000 S	10 mΩ	100 S
$\mathrm{C_{2+}}$	1 mΩ	1,000 S	10 mΩ	100 S	10 mΩ	100 S
$\mathrm{C_{1-}}$	10 mΩ	100 S	1 mΩ	1,000 S	10 mΩ	100 S
$\mathrm{C_{2-}}$	1 mΩ	1,000 S	10 mΩ	100 S	10 mΩ	100 S



(a) Case 1.

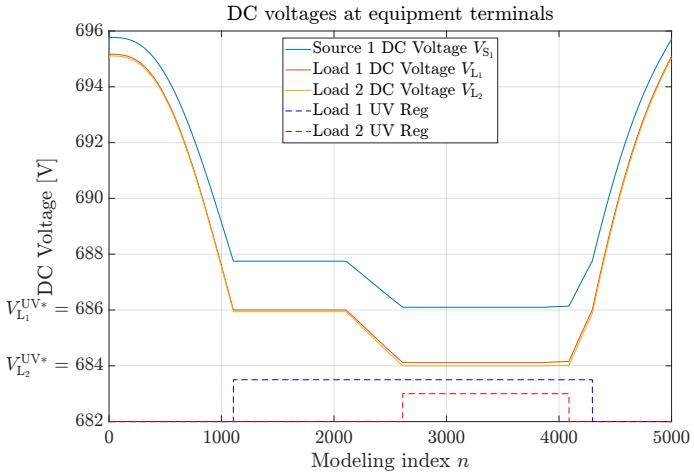


(b) Case 2.

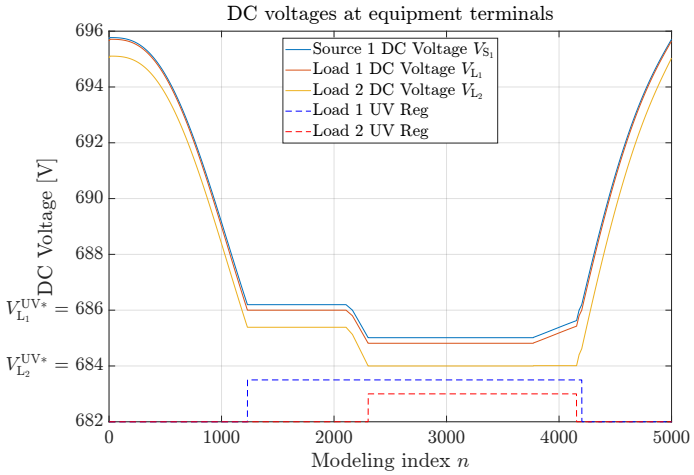


(c) Case 3.

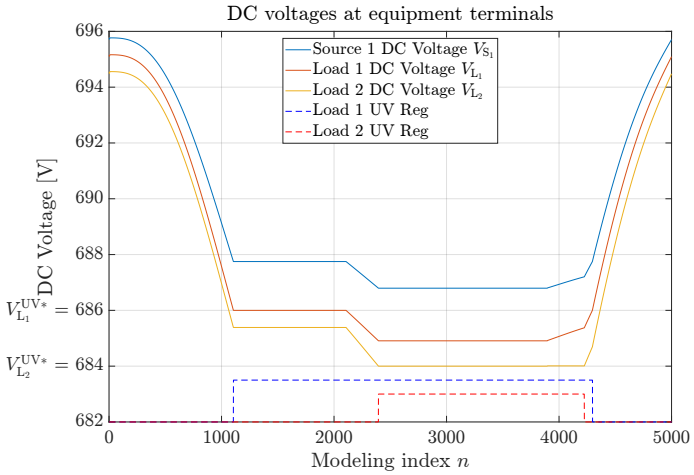
Figure 59. Power distribution in system of two loads with under voltage regulators with different cable resistances.



(a) Case 1.



(b) Case 2.



(c) Case 3.

Figure 60. DC voltages in system of two loads with under voltage regulators with different cable resistances.

```
%% Initialize cable resistances
% Case 1
RC1p = 0.010;
RC2p = 0.001;
RC1n = 0.010;
RC2n = 0.001;

% % Case 2
% RC1p = 0.001;
% RC2p = 0.010;
% RC1n = 0.001;
% RC2n = 0.010;
%
% % Case 3
% RC1p = 0.010;
% RC2p = 0.010;
% RC1n = 0.010;
% RC2n = 0.010;
```

Code 16. MATLAB code snippet presenting the cable resistance cases in Appendix 8.

In Case 1 (Figures 59a and 60a), the long shared feeder between the source and the loads produces a significant voltage drop at Load L_1, causing its undervoltage regulator to activate at a considerably lower power level than in the Case 2. As a result, the source never reaches full loading before the regulator activates, and the source and Load L_1 jointly droop the DC-link voltage as Load L_2's demand increases.

In Case 2 (Figures 59b and 60b), Load L_2 is farther from the source and Load L_1. The voltage drop across the high-resistance feeder is smaller than in Case 1, because it now carries only the current for Load L_2 rather than the combined current of both loads. Load L_1, being close to the source, has its regulator activate almost simultaneously with the source reaching its current limit, whereas Load L_2's regulator activates slightly higher source power.

In Case 3 (Figures 59c and 60c), all cables are long, so both loads experience substantial voltage drops. Here the source never reaches its maximum output, as even Load L_2's regulator activates well before that point.

3.2 Unbalanced Sources vs. Circulating Currents

As introduced in Section 3.1.5, the term "circulating currents" refers to currents that do not reach the load and are instead induced between voltage

sources due to imperfections in internal voltage measurement or regulation. These circulating currents result from deviations between the actual DC-link voltage and the sensed voltages, and they persist regardless of the load.

In contrast to that phenomenon, the following example demonstrates a different kind of current flow between sources. This example uses ideal conditions without measurement errors and shows that source-to-source current can still occur purely due to mismatches in internal voltage references. Such current should not be classified as circulating current.

Consider a system consisting of two ideal droop-controlled voltage sources and a current-based load connected to a common DC-link, as shown in Figure 61. Both sources are initially configured with a DC voltage reference of 700 V and a drooping coefficient of 0.02 with no load applied in the system. Under these symmetrical conditions, both sources maintain the DC-link voltage with no current flowing between them. This balanced operation is shown in Figure 62 between modeling indexes $n = 0$ and $n = 1000$.

At $n = 1000$, the DC voltage reference of Source 2 is increased by 1 V, creating an intentional mismatch. At this modeling index, Source 2 sees a 1 V deviation between its reference and the DC-link voltage and responds by injecting a large current to raise the voltage. Source 1 still operates with the original voltage reference and outputs zero current during that step as the voltage is at that step the same as its reference. This current spike from Source 2 causes the DC-link voltage to increase in the next modeling index ($n = 1001$). After this adjustment, the system stabilizes into a state where both sources output equal but opposite currents: Source 2 continuously sources current, trying to raise the voltage, while Source 1 sinks the same amount to pull it

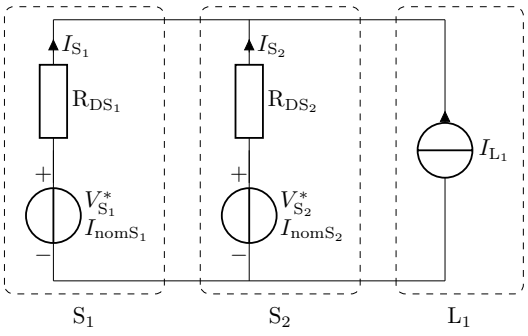


Figure 61. Circuit with two voltage sources and a single current-controlled load.

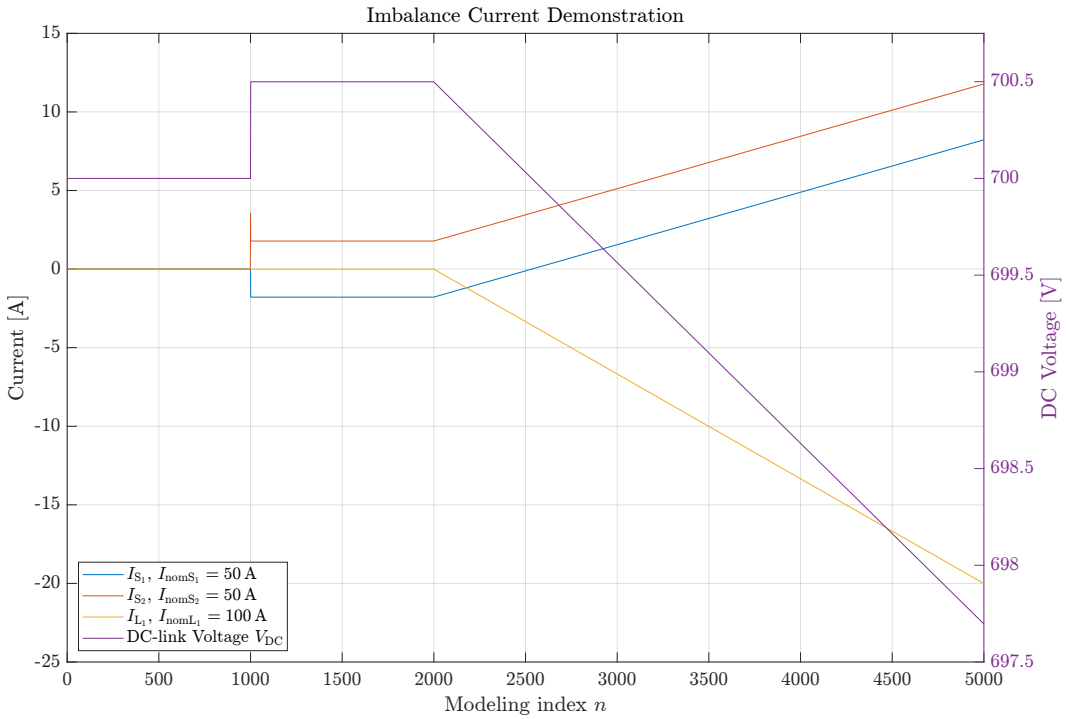


Figure 62. Current and voltage behavior when voltage reference of Source 2 is raised at $n = 1000$ and load applied at $n = 2000$.

back down. This imbalance between the sources is evident constantly.

At $n = 2000$, a current-controlled load begins to increase. As the load demand grows, more of the source current is redirected toward the load, and the current exchanged between the sources diminishes. Once the load current magnitude exceeds the total source output magnitude, i.e., $|I_{\text{Source 1}}| + |I_{\text{Source 2}}|$, the mutual current flow between the sources disappears entirely. This occurs at modeling index $n = 2534$. Code 17 presents the MATLAB code to perform the modeling.

If circulating current is defined strictly as current that does not reach the load, the formulation in Equation (3.20) already satisfies this, since it contains no load-current terms and therefore renders the loads irrelevant. This suggests that such a load-based definition is too vague. In the example, the inter-source currents observed after modeling index 1000 are not caused by voltage measurement imperfections but by intentional differences in voltage references. From a control perspective, the sources in this situation effectively act as loads for one another: any voltage source operating with a different reference or droop coefficient will appear as a power sink or source

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [50; 50];            % Source nominal currents
DCRefs = [700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('H1_ImbalanceCurrent:SourceParameterMismatch', 'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
sourceCurrentArrays = zeros(numberOfSources,length(steps));

%% Initialize load modeling variables
% Order of the loads is as follows: [L1, L2...]
InomL = [100]; % Load nominal currents
numberOfLoads = length(InomL);
KappaMatrix = zeros(numberOfLoads,numberOfLoads);
loadCurrentArrays = zeros(numberOfLoads,length(steps));
loadProfiles = zeros(numberOfLoads,length(steps));

%% Initialize the load to zero
loadProfiles(1,:) = zeros(length(steps), 1);

%% Create static reference profile for L2
% loadProfiles(2,:) = loadProfiles(2,:) + -0.3;

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources, 1);
droopConduc = zeros(numberOfSources, 1);
R_MG = 0;

%% Run through the calculations
for n=steps
    if n >= 1000
        DCRefs = [700, 701];
    end

    if n >= 2000
        loadProfiles(1,n) = loadProfiles(1,n-1) - (1/15000);
    end

    %% Load handling
    % Populate the Kappa-matrix with new values according to the load profiles
    for i=1:numberOfLoads
        KappaMatrix(i,i) = loadProfiles(i,n);
    end

    % Calculate the load currents
    loadCurrentArrays(:,n) = KappaMatrix * InomL;

    %% DC voltage handling
    % Calculate R_MG
    for r=1:numberOfSources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = 1/droopResis(r);
    end
    R_MG = 1/sum(droopConduc);

    % Calculate the DC voltage
    if n==1 % If first step use default DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, sum(loadCurrentArrays(:,n)), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), sum(sourceCurrentArrays(:,n-1)), sum(loadCurrentArrays(:,n)), R_MG);
    end

    %% Source handling
    % Populate the Gamma-matrix with new values according to the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        GammaMatrix(j,j) = Gamma(DCVolt(n), DCRefs(j), DCDroops(j));
    end

    % Calculate the source currents
    sourceCurrentArrays(:,n) = GammaMatrix * InomS;

    if n>2 && sourceCurrentArrays(1,n) >= 0 && sourceCurrentArrays(1,n-1) < 0
        source1CurrentZeroCrossing = n;
    end
end
```

Code 17. MATLAB code for imbalance current demonstration.

relative to its peers.

A more precise term for this type of current is imbalance current. It originates from differing internal control objectives, not from error. Such imbalance may be used intentionally, for example, in battery state-of-charge balancing (see Section 4.3) or controlled generator startup (see Section 5.2.2). In conclusion, while circulating currents are unwanted byproducts of imperfect measurement or operation and should be minimized, imbalance currents arise from differences in control settings and are often a necessary and purposeful tool in system-level control strategies.

4 STATE-OF-CHARGE BALANCING OF MULTIPLE ENERGY STORAGE SYSTEMS IN DC MICROGRIDS

The use of multiple energy storage systems is becoming increasingly common in shipboard power systems, supported by the adoption of DC-based power distribution architectures. However, they introduce new system operational requirements related to energy management and system serviceability. While service-related procedures—such as manufacturer-defined state-of-health tests—are typically carried out during port operations or maintenance periods, energy management functions must operate continuously, including during sailing.

Energy management in this context refers to supervisory functionalities that monitor and control the operation of energy storage systems to maintain overall system availability. Although power system blackouts are impossible to be entirely eliminated, maritime regulations aim to minimize their likelihood DNV AS (2024). A central part of this effort is ensuring timely responses to energy availability constraints, such as initiating auxiliary generator start-up if the system load becomes excessive or the state-of-charge of the energy storage systems drops below critical thresholds. Balancing the state-of-charge levels across multiple energy storage systems is a fundamental part of this supervisory logic.

Effective state-of-charge balancing improves not only the long-term load distribution across individual energy storage systems—thus extending their usable life—but also the system's ability to sustain high-power operation for as long as possible, both of which are also among the requirements set by ship owners. State-of-charge balancing is considered standard functionality for ship power systems involving multiple energy storage systems (P. Juppo, Chief Technology Officer at WE Tech Solutions Oy, personal communication, August 19, 2025).

Each energy storage system exhibits nonlinear charging and discharging current limits, determined by its local management system based on internal factors such as temperature, aging, and state of charge. This current-limit behavior is illustrated in Figure 63, which shows the curve of a real marine battery system as a function of state of charge. If one unit's state of

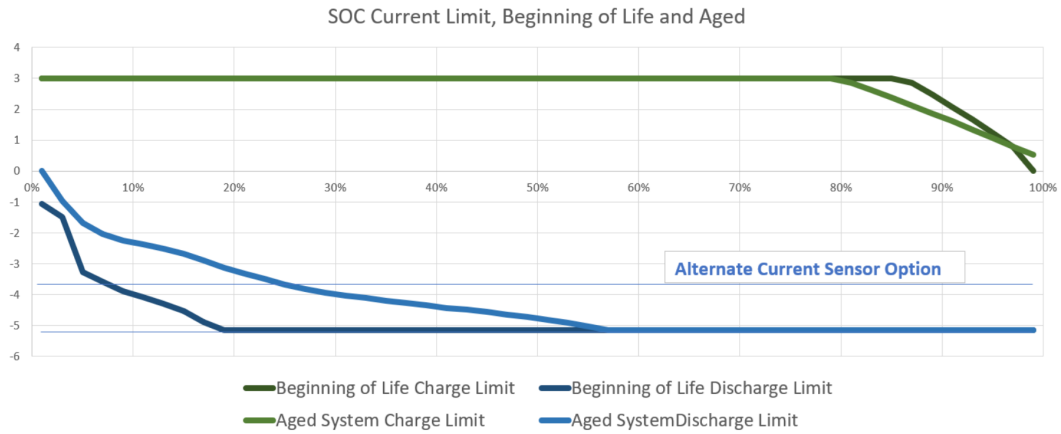


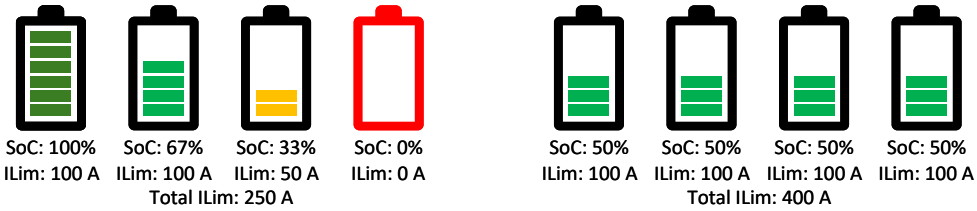
Figure 63. Battery current limit behavior as a function of state of charge (Battery system supplier, personal communication, August 2025).

charge falls sufficiently below that of the others, its discharging capability is restricted more than the others, thereby reducing the total available discharge current of the combined system. Figure 64a illustrates the principle of this effect, showing how variations in state of charge across multiple storage units constrain the overall discharging current limits.

A well-implemented state-of-charge balancing algorithm virtually combines the multiple energy storage systems into a single, unified storage entity. The composite parameters of this virtual energy storage system—such as total energy capacity [kWh], nominal current [A], and current limits [A]—are calculated as the sum of the corresponding quantities of the individual systems. The state-of-charge is treated as a synchronized quantity across all systems, such that the virtual energy storage system and each physical energy storage system maintain the same state-of-charge level at all times. Figure 64b presents the principle behind how the balanced state-of-charge values enable the full usage of the combined energy storage system, as full current is still available, while the total stored energy is the same as in Figure 64a.

The same principle in Figure 64 holds regardless of the sizes of the energy storage systems involved. Assuming the storages are of the same type but of different sizes, their current limits will proportionally behave in the same way, and balanced state-of-charge values ensure full utilization of the combined capacity of the systems.

When different types of energy storage systems are used—for example, bat-



(a) Example of unbalanced energy storages and their discharging current limits.

(b) Example of balanced energy storages and their discharging current limits.

Figure 64. Comparison of example energy storage systems with varying level of state-of-charge balancing.

teries based on different chemistries—their corresponding depth-of-discharges will differ. Depth-of-discharge represents the state-of-charge window within which the energy storage system should be operated in order to help ensure the manufacturer-specified lifetime. It can be viewed as specifying the "practical maximum and minimum state-of-charge values" between which normal operation should occur.

In such cases, the system should employ depth-of-discharge balancing, which is practically implemented by applying an additional scaling factor on top of the state-of-charge values. This work, however, concentrates only on state-of-charge balancing, as depth-of-discharge balancing is a direct derivation of it.

4.1 Operational Compatibility With Variable Voltage-based Control Strategy

For the balancing operation not to interfere with the variable voltage-based control strategy, it should not cause any extra deviations to the DC-link voltage. To this end, the balancing method and strategy must be selected carefully. Both the voltage sources' output voltages and currents must be in balance in order to achieve that.

In the most trivial case, the energy storage systems are current-controlled devices following an external reference, very much like the loads in Sec-

tion 3.1.6. Then the state-of-charge balancing is achieved simply by assigning references to the storages' DC/DC converters such that their offsets balance each other out, i.e., the sum of the balancing reference offsets is zero. But suppose the batteries are the voltage-supporting devices of the DC microgrid. In that case, the balancing must be performed while they actively maintain the DC-link voltage.

In this situation, for the balancing to happen, an intentional imbalance in the outputs of the DC/DC converters running the energy storage systems is required—effectively manipulation of the Γ functions. As shown in Equation (3.2), there are two variables that can be used to achieve this: DC voltage reference V_{S}^{*} or DC drooping coefficient ζ_{S}. Both of the variables have fundamentally different effects on the operation of the DC voltage-maintaining DC/DC converter, thus both references are handled separately in Sections 4.3 and 4.4 respectively.

The operation of balancing the states-of-charge between multiple energy storage systems requires knowledge of their charge states. This information is not inherently shared between the equipment in the DC microgrid, as it cannot be deduced from the DC-link voltage. Additionally, sharing that information via communication lines between the inverters would create an unnecessarily complicated system. This information, however, must be presented to the ship's crew and is therefore available to the microgrid controller, which can utilize it to perform the state-of-charge balancing.

4.2 Modeling the State-of-Charge Update

In the energy storage related mathematical models, the state-of-charge of each energy storage system E_i at modeling index n is calculated iteratively from its previous value at index $n-1$ and the current $I_i(n)$ that its corresponding DC/DC converter supplies to or draws from the DC-link. The storage capacity is allowed to vary between modeling steps to account for conditions such as battery string disconnections or reconnections. This variable maximum energy capacity is denoted as $Q_{\mathrm{maxE}_i}(n)$:

$$\mathrm{SoC}_{\mathrm{E}_i}(n) = \mathrm{SoC}_{\mathrm{E}_i}(n-1) - \frac{I_i(n)}{Q_{\mathrm{maxE}_i}(n)}, \tag{4.1}$$

where the current $I_i(n)$ follows the sign convention used throughout this work: a positive current indicates power flow toward the DC-link (discharging the storage), and a negative current indicates charging. Accordingly, the state-of-charge decreases when the converter current is positive and increases when it is negative.

Although the quasi-static modeling approach omits time-dependent dynamics such as those introduced by capacitances or inductances, the state-of-charge update inherently involves energy transfer over time. Nevertheless, the goal of the modeling in this work is not to predict time durations such as how long energy storage charging takes. Instead, it aims to determine optimal control strategies and to understand how different parameterizations and scenarios influence the system's behavior. To approximate the charging and discharging behavior, the storage current $I_i(n)$ above is interpreted as an ampere-hour equivalent per modeling step. It should be emphasized that this formulation does not introduce a time variable; the update is purely algebraic, and each modeling step represents a discrete change in stored charge rather than a time-integrated quantity. This enables tracking the change in stored energy without defining the actual time interval between steps.

While this approach does not allow for predicting how long an energy storage system can sustain a load or how much time state-of-charge balancing will require, it is entirely sufficient for modeling the control behavior of the balancing itself. Once this method is combined with the inverter models introduced in Section 3.1 and extended with proper modeling of energy storage current limit behaviors, it enables accurate simulation of the system's behavior under various conditions. In the future, when time dynamics are fully incorporated into the modeling paradigm, the charging and discharging can be modeled more accurately.

4.3 Utilizing Voltage Reference

The most straightforward way to implement state-of-charge balancing between multiple DC/DC converters operating as voltage sources is to induce an imbalance in the outputs of the converters via different DC voltage refer-

ences. The approach of balancing by utilizing the DC voltage reference has been used in the literature by, e.g., Hu and Weaver (2016) and Li, Dragicevic, Diaz, Vasquez, and Guerrero (2014). The effect of this method was also described in Section 3.2, where the voltage sources are forced to "fight" each other, while the DC voltage drooping keeps the system stable and controls the output currents of the voltage sources.

In the sources above, Hu and Weaver (2016) create an offset to the voltage reference directly based on the state-of-charge of the energy storage systems, multiplied by a selected coefficient. This creates balancing that is not dependent on any communication line and allows the DC-link voltage to deviate according to the states-of-charge of the storages. While this approach could have interesting applications in the variable voltage-based control strategy, this dissertation, however, concentrates on the principle that only the loads in the microgrid induce voltage deviation through the sources' drooping.

Li et al. (2014) propose an algorithm where, instead of directly basing the voltage reference offset on the storage's state-of-charge, as is done by Hu and Weaver (2016), it is based on a coefficient derived by dividing each storage's state-of-charge by the average of the state-of-charge differences between the storages, from which 1.0 is subtracted. This method requires communication in order for the equipment—or a centralized controller—to have knowledge of the state-of-charge levels of the different storages. The division, however, creates highly unstable coefficients that approach positive or negative infinity as the states-of-charge converge and their differences approach zero. Thus, it is uncertain how the algorithm manages to function, but according to the presented results, the state-of-charge balancing works without causing deviations to the DC-link voltage.

Mathematically, in the proposed modeling paradigm of this work, the state-of-charge balancing method in question adds an offset V_{B}^{*} to the source's droop-controlled DC voltage reference V_{DS}^{*} defined in Equation (1.2),

$$V_{\mathrm{DS}}^{*} + V_{\mathrm{B}}^{*} = V_{\mathrm{S}}^{*}\left(1 - \frac{I_{\mathrm{S}}}{I_{\mathrm{nomS}}}\zeta_{\mathrm{S}}\right) + V_{\mathrm{B}}^{*}, \tag{4.2}$$

and, by extension, also for the Γ-function used to calculate the voltage source's

current as presented in Equations (3.1) and (3.2),

$$\begin{aligned} I_{\mathrm{E}} &= \left(\frac{V_{\mathrm{S}}^{*} - V_{\mathrm{DC}}}{V_{\mathrm{S}}^{*}\zeta_{\mathrm{S}}} + \frac{V_{\mathrm{B}}^{*}}{V_{\mathrm{S}}^{*}\zeta_{\mathrm{S}}} \right) I_{\mathrm{nomE}} \\ &= \left(\Gamma_{\mathrm{S}} + \Gamma_{\mathrm{B}} \right) I_{\mathrm{nomE}} \\ &= \Gamma_{\mathrm{E}} I_{\mathrm{nomE}}. \end{aligned} \tag{4.3}$$

The derivation of the above equation is presented, and the naming convention is explained, in Appendix 9.

This creates a balancing functionality whose operation is independent of the system loading state, as the method simply adds an offset $V_{\mathrm{B}_i}^{*}$ to the equipment's voltage reference, thereby forcing the equipment to adjust its output current accordingly. It is not tied to the load-induced voltage deviation between the reference V_{S}^{*} and the DC-link voltage V_{DC}, and thus constitutes a separate mechanism from the normal droop-based operation. Additionally, as the voltage source's output current is directly proportional to the DC voltage reference offset V_{B}^{*}, as shown in Equation (4.3). To prevent DC-link voltage deviation, it is sufficient to keep the voltage reference offsets balanced—the output currents will then naturally follow.

4.3.1 Calculating The Reference Offset

For the state-of-charge balancing operation, the microgrid controller must calculate the offset variable V_{B}^{*} for each of the energy-storage-controlling DC/DC converters separately. One implementation is to utilize the drooping principle: as with DC voltage control, the output current is higher the greater the difference between the reference and the actual voltage. The same principle can be used here as well. In this context, the reference is the target state-of-charge for the balancing, denoted by $\mathrm{SoC}_{\mathrm{T}}$; the actual value is the energy storage system's current state-of-charge, denoted as $\mathrm{SoC}_{\mathrm{E}_i}$; and the output is the voltage reference offset $V_{\mathrm{B}_i}^{*}$:

$$V_{\mathrm{B}}^{*} = \frac{\mathrm{SoC}_{\mathrm{E}_i} - \mathrm{SoC}_{\mathrm{T}}}{\zeta_{\mathrm{B}_{scale}} V_{\mathrm{B}_{scale}}^{*}}, \tag{4.4}$$

where $\zeta_{\mathrm{B}_{scale}}$ is the drooping factor indicating how many percentage points the state-of-charge must deviate in order for the offset V_{B}^{*} to be equal to $V_{\mathrm{B}_{scale}}^{*}$. In this work, $V_{\mathrm{B}_{scale}}^{*}$ is always considered to be 1 V, and $\zeta_{\mathrm{B}_{scale}}$ is 0.02. In order for the balancing to work symmetrically between each energy storage system, the values of $\zeta_{\mathrm{B}_{scale}}$ and $V_{\mathrm{B}_{scale}}^{*}$ must be the same for all of them.

With Equation (4.4), it is now possible to calculate the balancing offset V_{B}^{*} for the DC/DC converter operating as a voltage source. As it is based on droop, it is self-regulating—producing a larger value the greater the difference between the actual state-of-charge and the target. To utilize this equation, the target state-of-charge $\mathrm{SoC}_{\mathrm{T}}$ must be defined.

4.3.2 Defining the Balancing Strategy

Defining the balancing strategy involves selecting a target state-of-charge, $\mathrm{SoC}_{\mathrm{T}}$, which serves as the reference toward which each energy storage system's state-of-charge is regulated using its respective V_{B}^{*} offset.

To ensure practical plausibility, $\mathrm{SoC}_{\mathrm{T}}$ must remain within the bounds defined by the minimum and maximum state-of-charge values currently present in the system. If $\mathrm{SoC}_{\mathrm{T}}$ falls outside this range, the balancing algorithm would effectively attempt to inject or absorb energy from sources beyond the participating energy storage systems. This would result in positive or negative loading of the system and DC voltage deviation.

Even when $\mathrm{SoC}_{\mathrm{T}}$ is within valid limits but numerically fixed—for example, at 70%—any converter that reaches this value would, by definition, attempt to counteract further deviations. This behavior would prevent further discharging of the storage unit to support load demands. Consequently, $\mathrm{SoC}_{\mathrm{T}}$ must be dynamic and adapt to the actual state-of-charge values of the energy storage systems.

Consider, for instance, a system consisting of four identical DC/DC converters operating as voltage sources, as illustrated in Figure 65. Each is connected to an energy storage unit of equal capacity, with initial state-of-charge values assigned at random, as shown in Table 11, where energy storages' states-of-charge associated with the individual sources are denoted as

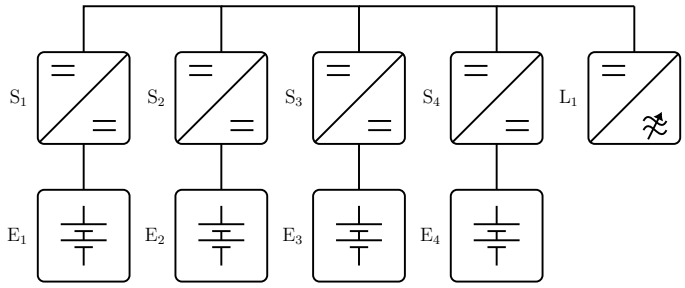


Figure 65. Simplified single-line diagram of the modeled system with four DC/DC converters working as the microgrid's voltage sources.

Table 11. The randomly selected initial state-of-charge values for the modeled energy storage units.

Initial Energy Storage States-of-Charge
$\mathrm{SoC}_{\mathrm{E}_1} = 76\%$
$\mathrm{SoC}_{\mathrm{E}_2} = 66\%$
$\mathrm{SoC}_{\mathrm{E}_3} = 100\%$
$\mathrm{SoC}_{\mathrm{E}_4} = 67\%$

$\mathrm{SoC}_{\mathrm{E}_i}$. Two intuitive options for selecting $\mathrm{SoC}_{\mathrm{T}}$ in such a system are the minimum or maximum state-of-charge values.

Selecting $\mathrm{SoC}_{\mathrm{T}}$ to follow the lowest state-of-charge among the storages in the system and activating the balancing causes the DC/DC converters with higher state-of-charge values to initially inject current into the DC-link in order to reduce their own state-of-charge values, see Figure 66a. As a result, the DC voltage rises, and the DC/DC converter with the lowest state-of-charge reactively draws current from the DC-link to compensate for the voltage increase. This creates a state-of-charge balancing functionality that results in a higher DC-link voltage during balancing compared to normal operation.

The same current sign convention as before applies: currents flowing toward the DC-link are positive, and those flowing away from the DC-link are negative. This means the state-of-charge decreases when the DC/DC converter current is positive and increases when the current is negative.

Another strategy is to do the opposite and assign $\mathrm{SoC}_{\mathrm{T}}$ to follow the highest state-of-charge value in the system, see Figure 66b. At the moment of balancing activation, this causes the DC/DC converters with lower-than-target state-of-charge to actively charge their storage systems, thereby lowering the DC-link voltage. The resulting voltage drop then forces the DC/DC converter

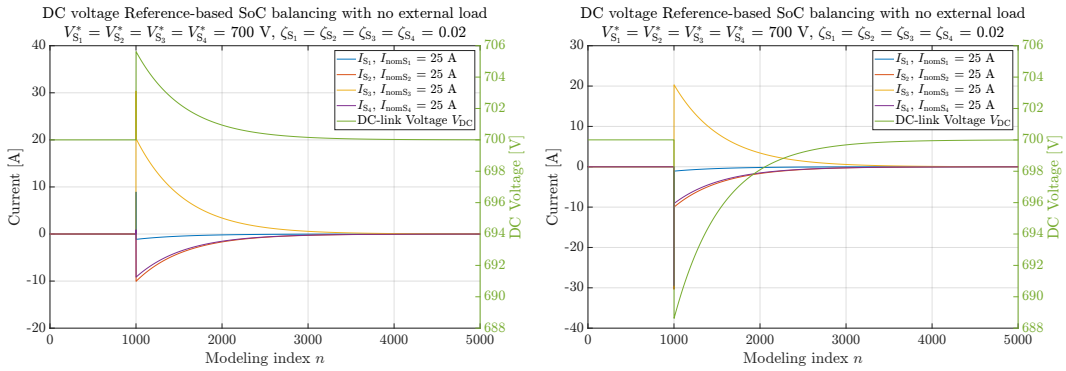


Figure 66. Loads and DC-link voltage during energy storage state-of-charge balancing, starting at modeling index $n = 1000$ utilizing the system's (a) lowest and (b) highest state-of-charge as the $\mathrm{SoC_T}$.

with the highest state-of-charge to inject current into the DC-link. This creates a state-of-charge balancing functionality that pulls the DC-link voltage lower than during normal operation.

Table 12 presents the currents and DC voltages from modeling indexes 999 to 1001 in both cases shown in Figure 66. When $\mathrm{SoC_T}$ is selected to follow the system's lowest state-of-charge, which in this case is Source 2, the source never actually actively participates in the balancing. At modeling index $n = 1000$, its current remains zero, as it was in the previous index, and it reacts only to the DC voltage rise at modeling index $n = 1001$. The same holds true for Source 3 when $\mathrm{SoC_T}$ follows the highest state-of-charge—Source 3 never actively contributes to the balancing and merely reacts to the DC voltage drop.

Table 12. Modeled currents and voltages during unoptimized energy storage state-of-charge balancing at modeling indexes $n = 999$, $n = 1000$ and $n = 1001$.

	Lowest SoC as $\mathrm{SoC_T}$			**Highest SoC as $\mathrm{SoC_T}$**		
	$n = 999$	$n = 1000$	$n = 1001$	$n = 999$	$n = 1000$	$n = 1001$
I_{S_1}	0.00 A	8.93 A	−1.13 A	0.00 A	−21.43 A	−1.08 A
I_{S_2}	0.00 A	0.00 A	−10.04 A	0.00 A	−30.36 A	−9.99 A
I_{S_3}	0.00 A	30.36 A	20.26 A	0.00 A	0.00 A	20.31 A
I_{S_4}	0.00 A	0.89 A	−9.15 A	0.00 A	−29.46 A	−9.10 A
$\sum \mathbf{I_S}$	0.00 A	40.18 A	−0.07 A	0.00 A	−81.25 A	0.15 A
V_{DC}	700 V	700 V	706 V	700 V	700 V	689 V

The selected balancing strategy—and the resulting SoC_T—does not affect the end result; the balancing proceeds at the same rate, and all energy storage systems will reach the same state-of-charge in the end, see Figure 67. However, the presented non-optimal balancing strategies cause disturbances to the DC-link voltage, as shown in Figure 66. In extreme cases, these disturbances can trigger unwanted undervoltage or overvoltage regulator activations. This effect was demonstrated earlier in this work using non-droop-compensated regulators in Section 3.1.7, and it is also evident later in Section 5.2.1. Therefore, the optimal approach would be to determine a value of SoC_T that produces V_B^* offsets in such a way that the balancing process has no impact on the DC-link voltage.

To ensure that the balancing has no effect on the DC-link voltage, the system's center of energy, denoted as SoC_C, must be calculated. This represents the state-of-charge around which the energies of the storage systems are balanced. The concept is analogous to the center of mass of a rod being balanced horizontally on a finger. When this point is used as the target SoC_T the actions of the energy-storage-controlling DC/DC converters become balanced and simultaneous. As a result, the DC-link voltage remains undisturbed, as shown in Figure 68.

Table 13 presents the currents and DC voltage during the optimized balancing, specifically their values at modeling indexes 999 to 1001—i.e., before, during, and after the balancing is activated. Unlike in the unoptimized case shown in Table 12, all sources begin supplying balancing current already at

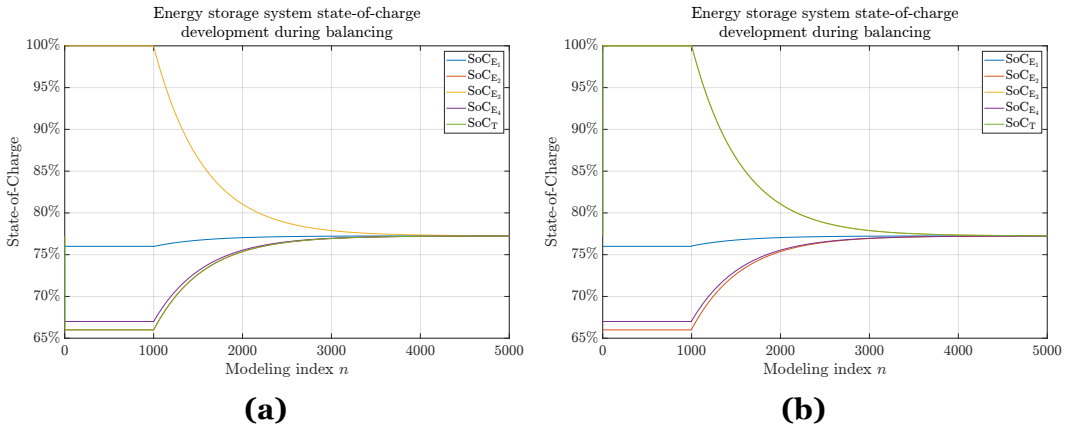
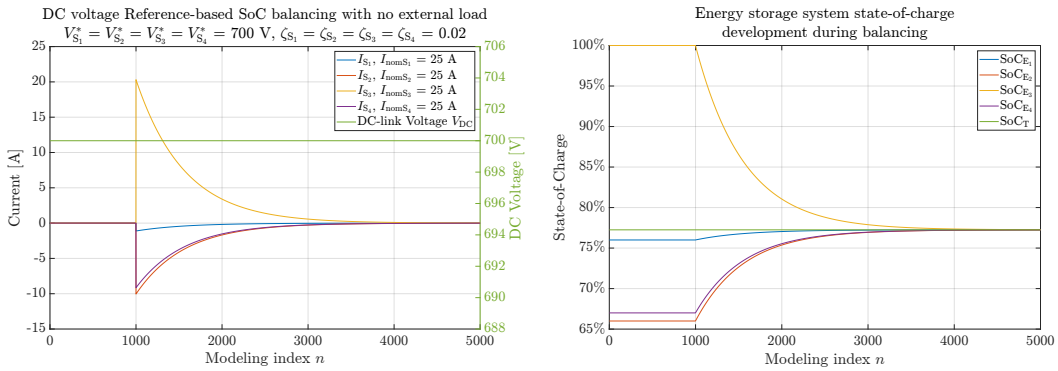


Figure 67. Energy storage systems' state-of-charge during balancing by DC/DC converters with (a) the lowest and (b) the highest states-of-charge used as SoC_T.



(a) **(b)**

Figure 68. Energy storage systems' (a) loading and (b) states-of-charge during balancing by DC/DC converters utilizing $\mathrm{SoC_C}$ as $\mathrm{SoC_T}$.

Table 13. Modeled currents and voltages during optimized energy storage state-of-charge balancing at modeling indexes $n = 999$, $n = 1000$ and $n = 1001$.

	$\mathrm{SoC_C}$ used as $\mathrm{SoC_T}$		
	$n = 999$	$n = 1000$	$n = 1001$
I_{S_1}	0.00 A	−1.12 A	−1.11 A
I_{S_2}	0.00 A	−10.04 A	−10.03 A
I_{S_3}	0.00 A	20.31 A	20.28 A
I_{S_4}	0.00 A	−9.15 A	−9.14 A
$\sum \mathbf{I_S}$	0.00 A	0.00 A	0.00 A
V_{DC}	700 V	700 V	700 V

index 1000, the moment the balancing is activated.

From Figure 68b, it can be seen that only Source 3 has a state-of-charge higher than the target $\mathrm{SoC_T}$. Therefore, it alone begins discharging its energy storage to the DC-link, while the others start charging their own systems. The overall behavior is nearly identical to the other cases presented earlier—the key difference being that in the optimized case, all sources actively participate in the balancing process instead of passively reacting to DC voltage changes.

If we make the incorrect assumption that the energy storage systems always have the same and constant capacities, it would be sufficient to calculate the average of their states-of-charge as the center of energy $\mathrm{SoC_C}$. However, it is prudent to assume that larger power systems may include energy storage systems of varying sizes. Additionally, in practice, the larger energy storage systems typically consist of smaller separate units operating in parallel. The

naming of these smaller units varies by supplier—in battery systems, for example, they have been referred to as strings or packs. These smaller units can be individually disconnected, for instance due to fault trips, and later reconnected on-the-fly, resulting in energy storage systems with effectively changing maximum energy capacities; see Figure 69.

Due to the different and on-the-fly capable changing maximum energy storage capacities, simply calculating an average is not sufficient to determine the center of energy. For this reason, we need to take a weighted average of the system as,

$$\mathrm{SoC_C} = \frac{\sum_{i=1}^{I} \mathrm{SoC_{E_i}} Q_{\mathrm{maxE}_i}}{\sum_{i=1}^{I} Q_{\mathrm{maxE}_i}}, \tag{4.5}$$

where $\mathrm{SoC_{E_i}}$ and Q_{maxE_i} are the state-of-charge and the maximum capacity in ampere-hours, respectively, of the energy storage E_i. An added benefit of this approach is that, if needed, the state-of-health can be taken into account in the Q_{maxE_i} calculation and thus all relevant variables would be included:

$$Q_{\mathrm{maxE}_i} = \mathrm{SoH_{E_i}} Q_{\mathrm{ratedE}_i}, \tag{4.6}$$

where $\mathrm{SoH_{E_i}}$ and Q_{ratedE_i} are the state-of-health and rated energy of the energy storage system in question, respectively.

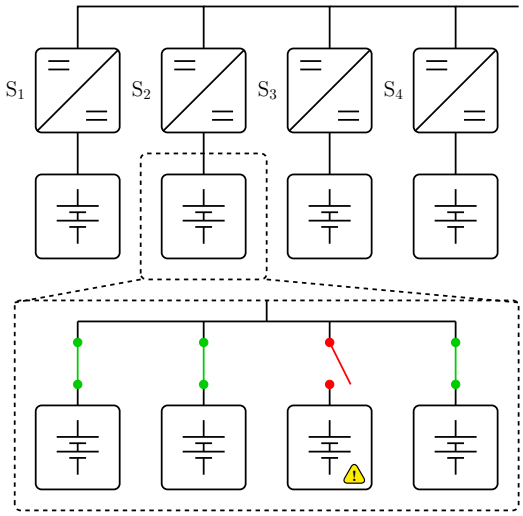


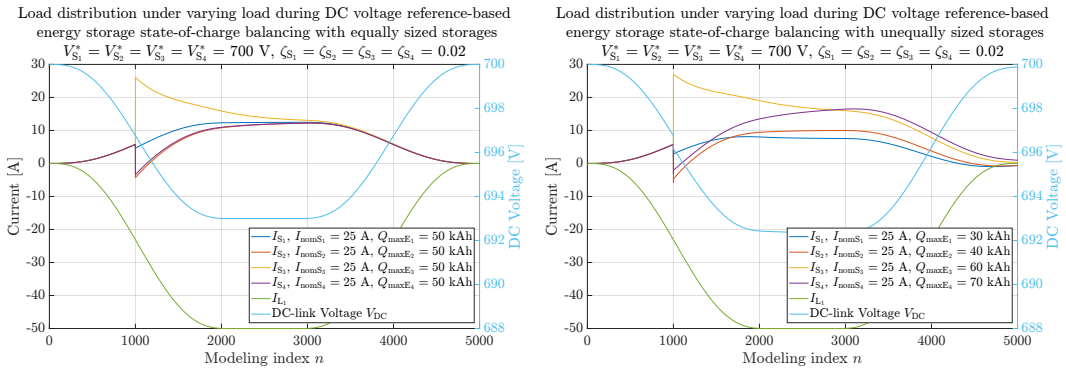
Figure 69. An example simplified single-line diagram of a large energy storage system comprised of multiple smaller units.

We now have the mathematical foundation to perform state-of-charge balancing of energy storage systems without disturbing the DC microgrid voltage. This applies even when operating concurrently with systems of different capacities and states of health. In this work, $\mathrm{SoC_C}$ is consistently used as the target $\mathrm{SoC_T}$. Appendix 10 presents the MATLAB code used to run the state-of-charge balancing algorithm under zero external load, implementing the strategies described earlier.

4.3.3 State-of-charge Balancing Under Varying Load With Equal and Unequal Storage Sizes

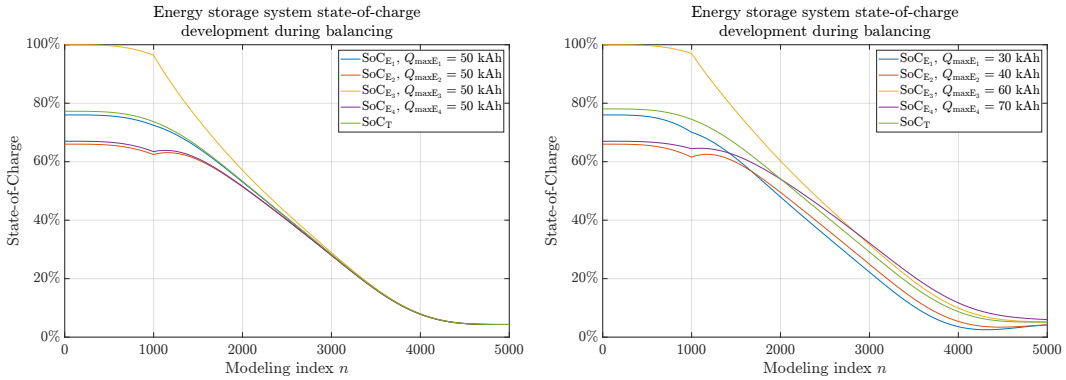
Introducing a varying load to systems with both equal and unequal energy storage sizes highlights the limitations of the current modeling approach. In order to keep the scope of this work focused, the balancing algorithm does not account for voltage sources transitioning into current-controlled operation upon reaching their current limits, nor for modeling how an energy storage system's discharging current limit varies with its state-of-charge. These aspects are deferred to future research. In this section, the applied load profile follows the same form as in the earlier modeling cases of Section 3.1, but is parametrized to range from −50 to 0 A in order to keep the states-of-charge within realistic bounds. Figure 70 presents the modeled results of state-of-charge balancing for both equal and unequal energy storage sizes under varying load.

In the system with equally sized energy storage systems, the converter currents converge during the modeling period, just as they do in the no-load case shown in Figure 68a. However, in the system with unequal storage sizes, the currents do not converge—even by modeling step $n = 5000$, when the external load has dropped to zero. The lack of convergence during loading is expected, as the different storage sizes influence the voltage references V_{B}^{*} via the center-of-energy calculation $\mathrm{SoC_C}$ in Equation (4.5). However, once the load is zero, current convergence would be expected if the state-of-charge balancing mechanism were effective. Figure 71 confirms this imbalance in the states-of-charge, indicating that balancing is not achieved. Therefore, additional correction is required to handle systems with unequally sized energy storage systems.



(a) **(b)**

Figure 70. Energy storage systems' loading while performing voltage reference-based balancing with (a) equal and (b) unequal energy storage sizes.



(a) **(b)**

Figure 71. Energy storage systems' states-of-charge while performing voltage reference-based balancing with (a) equal and (b) unequal energy storage sizes.

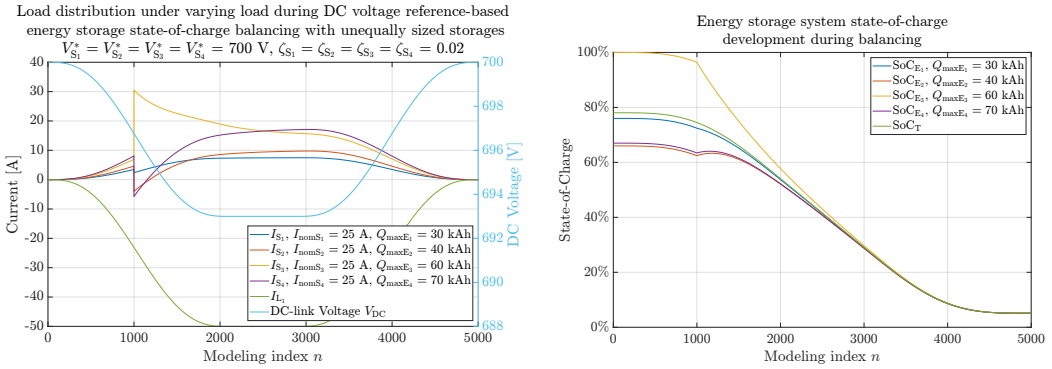
As shown in the modelings, the balancing mechanism functions correctly with equal-sized energy storage systems in both no-load and varying load scenarios, but fails when the storage sizes are unequal. From this, we can deduce that the issue is not offset-related and affecting all scenarios, but rather one of scaling. Moreover, since the balancing works with and without load in the case of equal storage sizes, the required scaling must reduce to unity when all storages are of equal size. This implies that the scaling factor should not involve a quantity representing a collective property, such as the sum of capacities, but rather something singular for each storage.

Additionally, since larger energy storage systems store more energy per percentage point of state-of-charge, achieving convergence toward the center-

of-energy $\mathrm{SoC_C}$ requires that the DC/DC converters connected to storages with higher capacities are loaded more aggressively than those with lower capacities. Scaling the current contribution of each DC/DC converter according to the ratio of its maximum capacity to the average capacity satisfies this condition. The resulting adjustment to the droop-based current reference is given by:

$$(\Gamma_{\mathrm{S}_i} + \Gamma_{\mathrm{B}_i}) \frac{Q_{\mathrm{maxE}_i}}{\overline{Q}_{\mathrm{maxE}}} = \left(\frac{V^*_{\mathrm{S}_i} - V_{\mathrm{DC}}}{\zeta_{\mathrm{S}_i} V^*_{\mathrm{S}_i}} + \frac{V^*_{\mathrm{B}_i}}{\zeta_{\mathrm{S}_i} V^*_{\mathrm{S}_i}} \right) \frac{Q_{\mathrm{maxE}_i}}{\overline{Q}_{\mathrm{maxE}}}, \tag{4.7}$$

where $\overline{Q}_{\mathrm{maxE}}$ is the average of the energy storage systems' maximum energy capacities. Figure 72 illustrates the improved current and state-of-charge behavior resulting from this scaling.



(a) **(b)**

Figure 72. Energy storage systems' (a) loads and (b) states-of-charge while performing voltage reference-based balancing with compensation for unequal energy storage sizes.

4.3.4 State-of-charge Balancing With Differently Sized DC/DC Converters

The final variable to explore in this context is the nominal current ratings of the DC/DC converters. A previous section covered the case where the converters were of equal size, while the connected energy storage systems varied in capacity. However, the converters themselves can also be intentionally dimensioned differently, or operate under temporary derated conditions due to exceptions. Figure 73 presents an example where some energy storage systems operate with partial capacity, potentially affecting the converter's

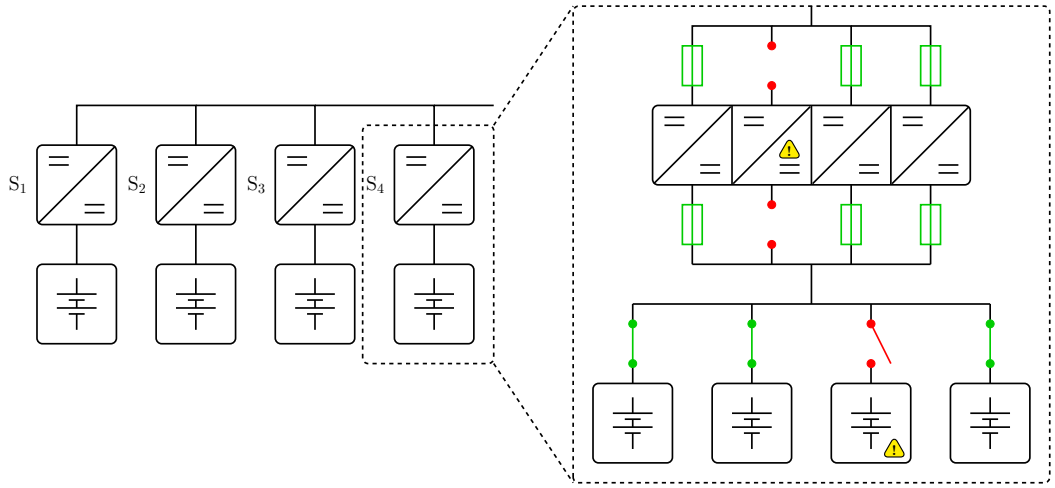


Figure 73. Simplified single-line diagram of a partial microgrid with DC/DC converters and energy storage systems capable of operating with only partial capacities..

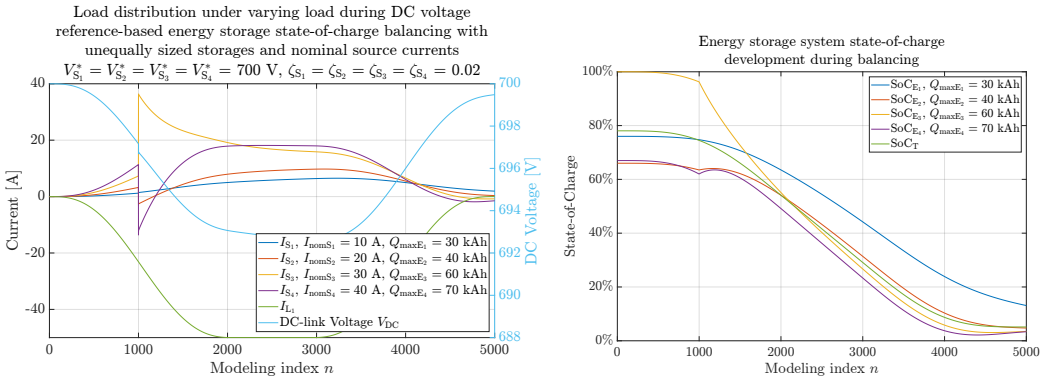
nominal current rating.

Depending on the converter manufacturer, such partial-capacity situations may lead either to a change in the converter's nominal current rating or merely to a modification of the enforced current limits. For modeling purposes, we assume here that the nominal rating I_{nomE_i} is adjusted to reflect the reduced capability.

To study this effect, we assign unequal nominal current ratings to the DC/DC converters and rerun the balancing calculations using the previously defined Γ-function from Equation (4.7). This function was originally introduced to compensate for unequal storage capacities, and here it is extended to account for differences in nominal converter current ratings.

Figure 74a shows that the resulting source currents behave asymmetrically, much like in the case of unequal storage capacities (Figure 70b). This imbalance is further reflected in the state-of-charge behavior shown in Figure 74b: the state-of-charge values do not converge.

When compared to the center-of-energy $\mathrm{SoC_C}$ used as the target value, it becomes evident that the converter with the lowest nominal current rating remains underloaded—its state-of-charge stays above the target—while the most powerful converter is overloaded, indicated by its state-of-charge consistently falling below the target. This behavior confirms that nominal cur-



(a) **(b)**

Figure 74. Energy storage systems' (a) loads and (b) states-of-charge while performing voltage reference-based balancing for a system with different energy storage sizes and DC/DC converter nominal current ratings.

rent imbalances must be accounted for in the balancing algorithm to achieve equal participation from all units and maintain long-term fairness in energy usage.

From the analysis of the modeling results in Figure 74, it can be deduced that the compensation for different nominal current ratings can be implemented with a similar, but inversed, algorithm as for the different storage sizes in Equation (4.7). Thus, when we implement scaling where the average of the nominal currents is divided by the inspected voltage source's nominal current, we get,

$$(\Gamma_{S_i} + \Gamma_{B_i}) \frac{Q_{maxE_i}}{\overline{Q}_{maxE}} \frac{\overline{I}_{nomS}}{I_{nomS_i}} = \left(\frac{V^*_{S_i} - V_{DC}}{\zeta_{S_i} V^*_{S_i}} + \frac{V^*_{B_i}}{\zeta_{S_i} V^*_{S_i}} \right) \frac{Q_{maxE_i}}{\overline{Q}_{maxE}} \frac{\overline{I}_{nomS}}{I_{nomS_i}}, \tag{4.8}$$

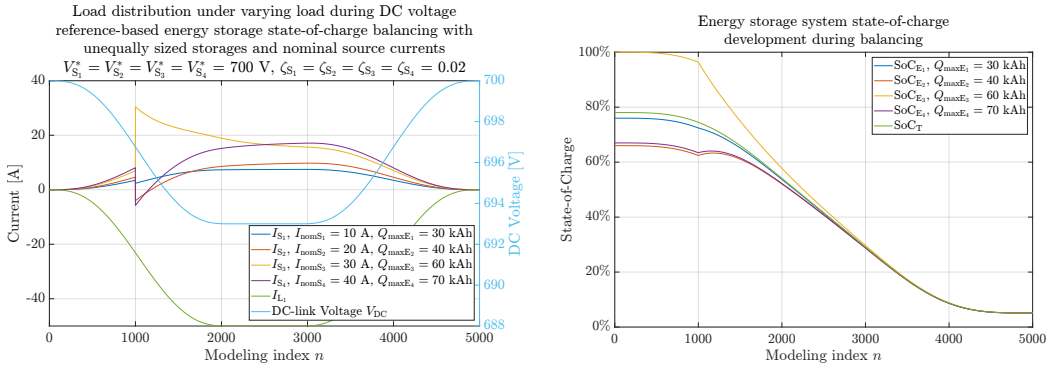
where $\overline{I}_{nomS_n}$ is the average of the voltage sources' nominal currents. It is, however, possible to simplify the equation by expanding the averaging portions, see Appendix 11. This gives a simpler presentation of the algorithm with just a division of the inspected energy storage's normalized energy capacity $Q^{p.u}_{maxE}$ by the inspected DC/DC converter's normalized nominal current $I^{p.u}_{nomS}$,

$$\Gamma_E = (\Gamma_S + \Gamma_B) \frac{Q^{p.u}_{maxE}}{I^{p.u}_{nomS}} = \left(\frac{V^*_S - V_{DC}}{\zeta_S V^*_S} + \frac{V^*_B}{\zeta_S V^*_S} \right) \frac{Q^{p.u}_{maxE}}{I^{p.u}_{nomS}}. \tag{4.9}$$

Modeling the system with the equation above, we have load distribution as shown in Figure 75a and energy storage states-of-charge as shown in Figure 75b. All currents now behave symmetrically, and the states-of-charge of the energy storages are converging to their center of energy used as the balancing target.

Now, with the Γ_E formulation in Equation (4.9), utilizing V_B^* from Equation (4.4) calculated based on the center of energy from Equation (4.5), and incorporating state-of-health into the maximum energy capacity in Equation (4.6), a comprehensive DC voltage reference-based state-of-charge balancing algorithm is achieved. This formulation adjusts the loading of the DC/DC converters in response to variations in all relevant parameters: energy storage capacity, state-of-health, and the converters' nominal current ratings.

This balancing implementation does not address several aspects related to the inverter and energy storage operation, and are left for future research: (i) the handling of situations where DC/DC converters reach their current limits or are stopped, and (ii) the adjustment of discharging and charging current limits based on the energy storage state-of-charge. Furthermore, configurations in which the nominal DC voltage references or droop coefficients differ intentionally between converters are excluded, as such variations are incompatible with the principles of the variable voltage-based DC microgrid control strategy.



(a) **(b)**

Figure 75. Energy storage systems' (a) loads and (b) states-of-charge while performing voltage reference-based state-of-charge balancing algorithm, compensated for different energy storage capacities and DC/DC converter nominal current ratings.

Appendix 12 presents the MATLAB code for performing the modelings from this section. The different energy storage capacities and DC/DC converter current ratings are available to be modified and the used settings are included as comments. The code also includes the different Γ_{E} versions as well for duplicating all modeling results.

4.4 Utilizing Drooping Coefficient

Another approach to performing state-of-charge balancing between energy storage systems is to assign each DC/DC converter an offset ζ_{B} to its base droop coefficient ζ_{S}. This results in differential current sharing between converters under load, as their drooped DC voltage references begin to differ only when the system is externally loaded.

In this droop coefficient offset-based method, the resulting Γ_{E} expression—analogous to the one presented earlier for DC voltage reference offset-based balancing in Equation (4.9)—takes the following form:

$$\Gamma_{\mathrm{E}} = \Gamma_{\mathrm{B}} \frac{Q_{\mathrm{maxE}}^{\mathrm{p.u}}}{I_{\mathrm{nomS}}^{\mathrm{p.u}}} = \frac{V_{\mathrm{S}}^{*} - V_{\mathrm{DC}}}{\left(\zeta_{\mathrm{S}} + \zeta_{\mathrm{B}}\right) V_{\mathrm{S}}^{*}} \frac{Q_{\mathrm{maxE}}^{\mathrm{p.u}}}{I_{\mathrm{nomS}}^{\mathrm{p.u}}}, \tag{4.10}$$

where ζ_{B} is the additional droop coefficient applied to enable balancing. Unlike the earlier method, where Γ_{S} and Γ_{B} were treated separately, here the balancing effect is embedded directly into the normal droop operation. This is due to the fact that adjusting ζ is mathematically equivalent to modifying the source's droop resistance R_{D_n}; see Appendix 13. As a result, no current is inherently injected unless an external load shifts the DC voltage.

This means the balancing effect is entirely dependent on the system having a non-zero external load. If no load is present to disturb the DC voltage away from the references, no balancing occurs, regardless of the droop offset values. Such a no-load condition may occur, for example, when a ship is connected to shore power and the onboard energy storages are idling in blackout prevention mode.

State-of-charge balancing using droop coefficient manipulation has been explored in prior research. For example, Xu, Wei, Yu, Guerrero, and Vasquez

(2021), Lu et al. (2012), and Ghanbari and Bhattacharya (2018) used similar approaches, while Saim et al. (2022) introduced power participation factors for more dynamic balancing, thereby adding non-linearities into the control.

The primary goal of state-of-charge balancing is to introduce controlled asymmetry in the loading of DC/DC converters to gradually drive all storage states-of-charges toward a common target $\mathrm{SoC_T}$. In this work, that target is defined as the center of energy $\mathrm{SoC_C}$ (Equation 4.5). In droop coefficient-based balancing, the offsets ζ_B should be assigned around the base droop value ζ_S, similarly to how voltage reference offsets V_B^* were applied in the earlier method.

There are two challenges and one inconvenience associated with this balancing method. The first challenge is that assigning an offset to the DC droop coefficient changes the voltage source's output voltage linearly, but not the output current. The linear relationship in voltage becomes evident when the equation for the drooped voltage reference is expressed with the offset coefficient,

$$V_\mathrm{DS}^* = V_\mathrm{S}^* \left(1 - \frac{I_\mathrm{S}}{I_\mathrm{nomS}}(\zeta_\mathrm{S} + \zeta_\mathrm{B})\right), \tag{4.11}$$

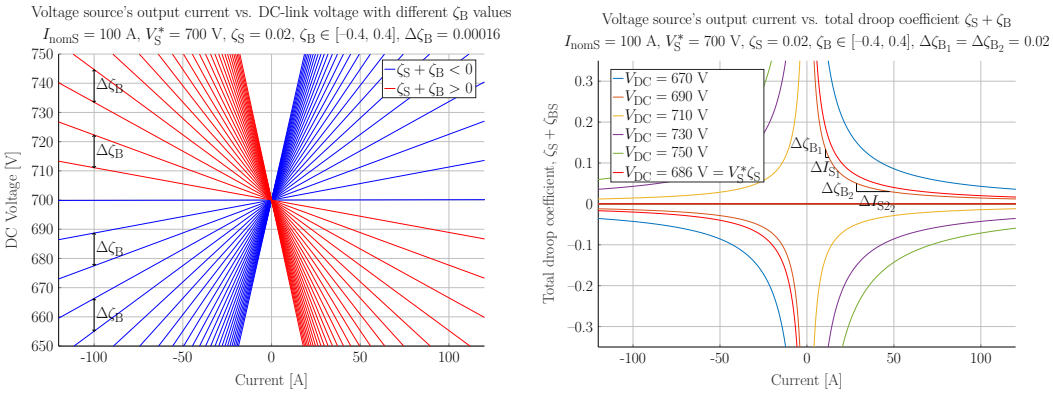
where the droop coefficients form a multiplier for the fraction. The derivation of the above from the equation with droop resistance (1.1) is presented in Appendix 13.

When we derive from the above equation the output current using the same principles as in Appendix 1, we get:

$$I_\mathrm{S} = \frac{V_\mathrm{S}^* - V_\mathrm{DC}}{(\zeta_\mathrm{S} + \zeta_\mathrm{B})V_\mathrm{S}^*} I_\mathrm{nomS}, \tag{4.12}$$

where the droop coefficients now introduce non-linearity through the denominator.

Equations (4.11) and (4.12) show that equal offsets ζ_{B_x} and ζ_{B_y} to already offset ζ_{S_x} and ζ_{S_y} values induce equal voltage reference differences, $\Delta\zeta_{\mathrm{B}_x} = \Delta\zeta_{\mathrm{B}_y} \Rightarrow \Delta V_{\mathrm{DS}_x}^* = \Delta V_{\mathrm{DS}_y}^*$, see Figure 76a, but this does not imply equal output current differences, $\Delta\zeta_{\mathrm{B}_x} = \Delta\zeta_{\mathrm{B}_y} \Rightarrow \Delta I_{\mathrm{S}_x} \neq \Delta I_{\mathrm{S}_y}$, see Figure 76b. Since DC-link voltage stability depends on balancing both the voltage reference and the output current, the DC-link voltage will deviate from the desired value. However, the effect on DC voltage diminishes as balancing progresses and the current differences converge, and it only poses a more significant



(a) **(b)**

Figure 76. Demonstration of droop coefficient induced (a) voltage linearity and (b) current non-linearity.

problem when compensating for large differences in states-of-charge.

The non-linearity also presents the second challenge which is visible in Figure 76b. Assigning an offset ζ_B such that it moves $\zeta_S + \zeta_B$ closer to zero increases the instability of the voltage source. In other words, the ζ_B values need to be defined so that the sum $\zeta_S + \zeta_B$ can never go too close to zero. Calculating ζ_B such that the minimum $|\zeta_B|$ is, for example, limited to half or one third of ζ_S, however, creates a system where individual DC/DC converters will not reverse their current to counteract the system load. This only affects the robustness of the balancing—that is, it slows it down—but does not prevent convergence.

Figures 77 and 78 present the modeling results for droop coefficient-based energy storage state-of-charge balancing. In order to keep the currents and states-of-charge within physically meaningful values, and at the same time give the modeling enough steps to converge the states-of-charges, the simulation parameters were adjusted. The load profile still utilizes the smoother-step function from Section 3.1, but now varies between −50 and 50 A, and the modeling was extended to 15,000 steps instead of the previous 5,000. The state-of-charge development during the balancing is presented in Figure 78a, and the droop coefficient variations in Figure 78b.

An operational inconvenience is that the system must continuously monitor the DC voltage level V_{DC}, and if it crosses the DC voltage reference value V_S^*,

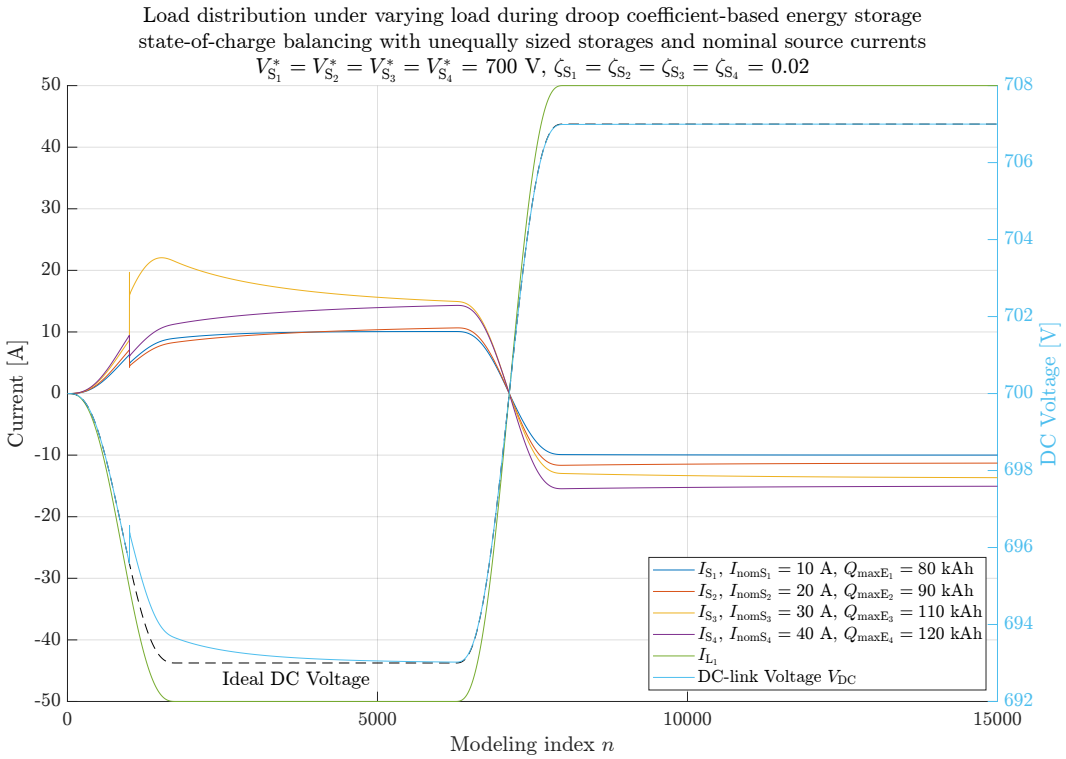
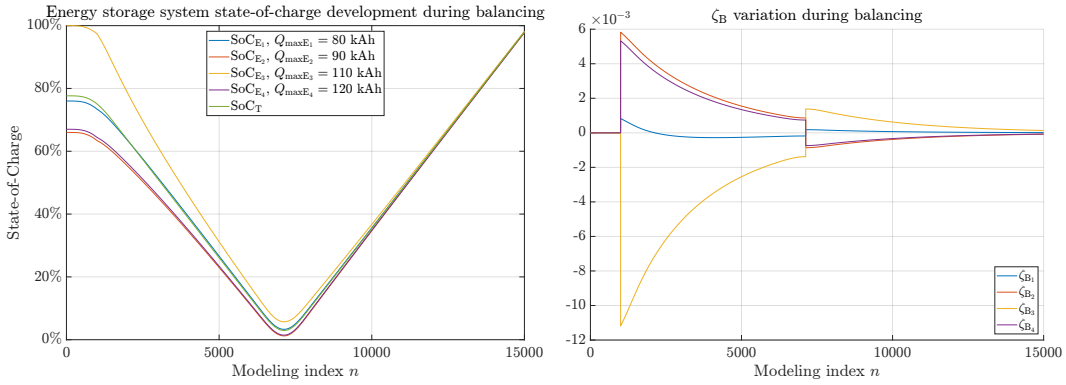


Figure 77. Load distribution during DC droop coefficient-based energy storage state-of-charge balancing.



(a) **(b)**

Figure 78. Energy storage systems' (a) states-of-charge and (b) balancing offsets ζ_{B_n} while performing DC droop coefficient-based state-of-charge balancing algorithm, compensated for different energy storage capacities and DC/DC converter nominal current ratings, with $\zeta_{B_{\text{scale}}} = 0.05$.

the ζ_{B_n} signs must be reversed, see also Figure 78b,

$$\zeta_{B_i} = \begin{cases} (\text{SoC}_{E_i} - \text{SoC}_T)\,\zeta_{B_{\text{scale}}}, & \text{when } V_{DC} > V^*_S, \\ -(\text{SoC}_{E_i} - \text{SoC}_T)\,\zeta_{B_{\text{scale}}}, & \text{otherwise,} \end{cases} \tag{4.13}$$

where $\zeta_{B_{scale}}$ is the scaling factor for the offset to control its magnitude in relation to the state-of-charge deviation. The sign reversal is required to invert the slope of the offset droop—switching between increasing and decreasing behavior—depending on whether the energy storage is charging or discharging.

Equations (4.10) and (4.13) now form the basis of a droop coefficient-based state-of-charge balancing algorithm applicable to arbitrary storage capacities and converter sizes. The method is not fully optimal due to the non-linear relationship between output current and voltage when droop coefficients are adjusted. In cases of extreme state-of-charge differences, the droop offset can become large, resulting in high output currents. Therefore, practical limitations must be considered, such as anticipating system-level voltage deviations and enforcing upper bounds on the droop offset. The algorithm could be further optimized by replacing the linear relationship in Equation (4.13) with a non-linear mapping of the offset as a function of state-of-charge deviation.

4.5 Conclusions on the Energy Storage Balancing Algorithms

The two presented state-of-charge balancing algorithms can be roughly categorized into active and passive balancing methods. The DC voltage reference offset V_B^*-based method actively induces current flow between the energy storages in order to balance the charges regardless of the external loading situation. Meanwhile, the DC voltage droop offset ζ_B-based method does not induce current flow between the storages, but rather passively aims to distribute the external loading unevenly between the voltage sources to, over time, balance the energy storages' states-of-charge. See Table 14 for an AI generated comparison of the two balancing methods.

Compared to the V_B^*-based algorithm, the ζ_B-based balancing algorithm is more complicated, slower, and presents additional challenges in maintaining DC-link voltage stability. Moreover, the coefficient-based algorithm depends on external load to induce current flow in the system, which leads to a voltage difference between the voltage sources' drooped DC voltage refer-

Table 14. Comparison of the developed state-of-charge balancing algorithms (OpenAI (2025b)).

Feature	**Voltage-Reference Offset (V_B^*)**	**Droop-Coefficient Offset (ζ_B)**
Mechanism	Add a balancing offset V_B^* to each source's droop reference; V_B^* is computed from the SoC deviation.	Modify the effective droop coefficient by adding ζ_B derived from SoC deviation; sign depends on whether V_{DC} is above or below V_S^*.
Dependency on External Load	Low; operation independent of load-induced deviations.	Moderate; redistribution follows droop law, effect depends on operating point and load.
Control Simplicity	High; one scalar offset per unit.	Medium; requires computing ζ_B with correct sign logic and mapping.
Impact on DC-Link Voltage	None by design; no impact on V_{DC}.	Non-zero; droop modification shifts DC voltage slightly, effect diminishes as balance is reached.
Response Speed	Fast; immediate effect at next control step.	Fast; immediate change in sharing, but current response is non-linear.
Scalability	High; p.u. scaling supports unequal sizes and many units.	High with tuning; requires bounded ζ_B for uniform behavior across many units.
Stability	High if offsets are bounded and clamped.	Medium – high; requires mapping of ζ_B for large imbalances to avoid excessive current steps.
Practical Feasibility	Good; requires write-access to V^* of each unit.	Good; requires run-time adjustability of droop coefficients, common in modern converters.
Advantages	Independent of load; no DC-link impact; simple and robust.	Works when V^* is fixed; leverages existing droop; predictable voltage response.
Disadvantages	Needs accurate SoC and bounded offsets; poor bounds can push unrealistic references.	Non-linear current response; can perturb V_{DC}; risks inter-converter currents if slope changes are excessive.
Best Use Case	Shipboard microgrids with variable voltage-based control where DC-link voltage must remain unaffected.	Systems with fixed V^* but adjustable droop, and moderate SoC deviations.

ences and the actual DC-link voltage. The algorithm then manipulates the slope of each voltage source's DC droop curve—either steepening or flattening it compared to the default—to create unequal voltage references among the DC/DC converters, thereby inducing current flow between them.

This elaborate symbiosis of external load, droop coefficient offsets, and their induced current flow between voltage sources is like a parasitic relationship, where the balancing functionality needs external load to exist. This, and the non-linear relationships presented earlier, are blatant disadvantages when compared with the V_B^*-based balancing algorithm, which is not dependent on any outside load, but inherently creates its own separately controlled current flow without any external intervention.

While it is possible to create a stable ζ_B-based balancing algorithm for multi-

ple differently sized energy storage systems run by differently sized DC/DC converters, it is, however, not likely worth the trouble in practice. If the system is capable of assigning the ζ_{B}, it is capable of assigning the V_{B}^{*} as well, and implementing a simpler and more capable algorithm. Thus, barring some special application area, ζ_{B}-based, and by mathematical equivalence R_{DS}-based, balancing development seems to be more of an academic exercise.

5 VARIABLE VOLTAGE-BASED POWER MANAGEMENT OF DC MICROGRIDS

The core principle of variable voltage-based power management is that load inverters within the microgrid use the DC-link voltage to independently determine their operating mode at any given time. Droop control forms the basis of the voltage's variable behavior, though it is not the only contributing factor. Droop control distributes the load proportionally among the voltage-supporting equipment, while variations in load demand or available generation capacity—which may change rapidly in exceptional situations, such as the sudden disconnection of a large source or load—cause corresponding changes in equipment loading. The DC voltage is directly influenced both by the sources, through their voltage references and droop coefficients, and by the loads, through their under- and overvoltage regulators. Additional influences include current limits on power flow to and from the DC-link, as well as the loads' reference values and any external constraints.

Traditional power management systems operate by continuously supervising the number and loading of active voltage sources. If available power becomes insufficient, new sources (e.g., generator sets) are started, or loads are limited or disconnected. This approach requires continuous low-level equipment control and depends entirely on the reliable operation of the central control system. The same applies during exceptional events: equipment relies on explicit control commands to limit power, shut down, or charge/discharge batteries. This dependence introduces operational delays and represents a clear single point of failure. To mitigate this, developers of power management and ship automation systems implement redundancies such as ring networks and duplicated controllers for port and starboard sections, in order to localize the consequences of failures.

In shipboard power systems, the variable voltage-based power management strategy provides significant advantages in robustness and fault resilience. Decision-making is distributed to the inverter level, eliminating the need for the central automation system to handle low-level control tasks such as starting additional sources or limiting loads when power becomes scarce. Instead, inverters perform these actions autonomously. The central control system remains responsible only for higher-level tasks, such as defin-

ing generator start-up priorities or disabling specific units, which lie outside the scope of the microgrid control strategy. These functions form the "sandbox" described earlier in Section 1.1.3, where the central system parameterizes rather than directly controls inverter operation. As a result, system robustness and fault resilience are greatly improved: even if the central control system fails completely, the inverters continue to operate correctly, and blackout is avoided.

5.1 System Control Hierarchy

While the variable voltage-based power management strategy is not hierarchical in nature, nor does it closely resemble the commonly used hierarchical control strategies, a control hierarchy nevertheless exists in all microgrid systems. This hierarchy is required to fulfill system-level operational requirements specific to the use case. In shipboard power systems, these include tasks such as starting and stopping equipment, assigning references, and setting other control parameters needed to operate the ship in different modes, for example, emission-free or shore-power modes. The implemented microgrid control strategy then defines the operational principles that ensure stable system behavior within the selected operating mode.

To clarify the division of responsibilities under the variable voltage-based power management strategy, a simplified control hierarchy is illustrated in Figure 79. The figure shows the two lowest levels of control required to manage a DC microgrid. At the lowest level are the inverters, represented with the widest combined base to indicate their primary role and overall responsibility. Above them is the microgrid controller, which parameterizes and supervises the inverter operation. In this context, parameterization refers to non-time-critical actions affecting the inverters as a whole, as opposed to continuous, time-critical low-level control of references and limits required by some other control strategies.

Inverters can operate either as current-controlled devices ▲, governed by their under- and overvoltage regulators, or as voltage-controlled devices ▲. As their operating principles differ fundamentally, it is useful to distinguish them explicitly, allowing clear identification of which device is primarily re-

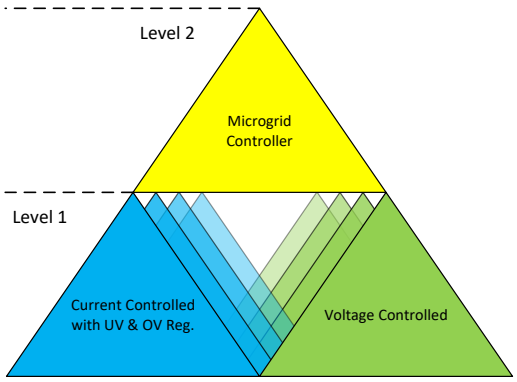


Figure 79. Lowest two levels of the control hierarchy in the variable voltage-based power management strategy.

sponsible for DC voltage control and which operates in a supporting role. All systems must also include a supervisory controller △. In small systems, this role can technically be performed by one of the inverters, or it can be handled by an external system, such as a ship power management controller. In all cases, a supervisory layer is required to coordinate inverter operation and provide a user interface.

As described in Section 1.1.3, under the variable voltage-based power management strategy the fast control required for continuous microgrid operation is located at the inverter level. The supervisory control system defines only the operating parameters and boundaries—the "sandbox"—within which the inverters function. The operating principles of inverters and DC microgrids are presented mathematically in Section 3.1. In practice, parameterization may include changing inverter modes between their internal operation states (e.g., speed, torque, or power control for motor drives), or assigning a new DC voltage reference due to a change in the system-level operating mode. These aspects are further demonstrated in later sections through real-life use cases.

Figure 80 shows an example DC microgrid in which the △, △, and △ correspond to the roles defined in Figure 79. In this example, the Grid Inverter at the top left is assumed to be connected to the national grid and is therefore capable of operating as a DC voltage source if required. From the DC microgrid's perspective, the DC/DC converter can also serve in either role: as a DC voltage supply or as a current source/sink.

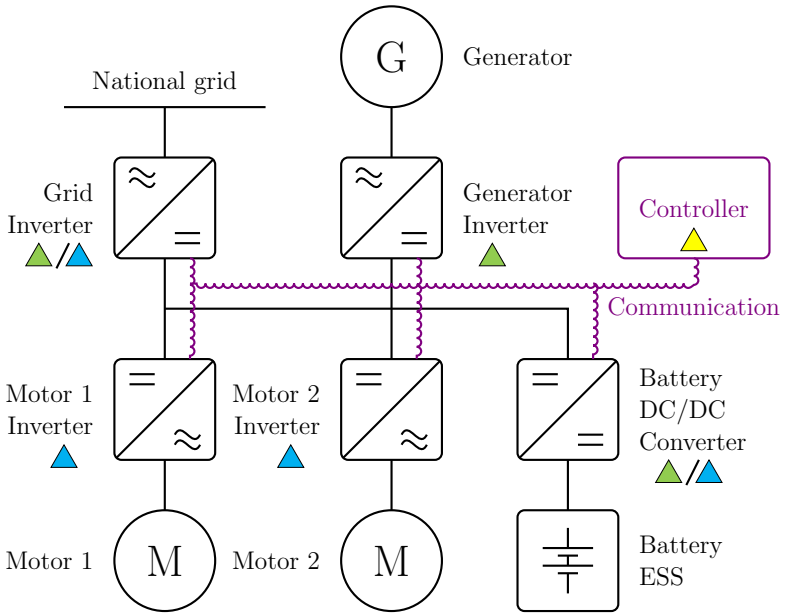


Figure 80. Example DC microgrid with inverters marked according to their hierarchical roles.

It should be noted that the system in Figure 80 is highly simplified in terms of both electrical components and control systems, and is presented solely as a conceptual reference. For example, it omits filters as well as a battery management system, which supervises the battery states, provides monitoring values, fault information, and charging and discharging current limits for the microgrid controller. A real system would also include an energy management system to estimate battery endurance, determine when charging is required and possible, and issue start commands for additional generators. Battery management is typically vendor-specific and integrated into the battery system itself, whereas energy management is system-level functionality. Its natural place is within the microgrid controller or, in some cases, a higher-level supervisory controller.

5.2 Ensuring Power Availability

Ensuring that power producers are not overloaded is one of the most crucial tasks in shipboard power management. Traditionally, this is achieved by a power management system that continuously calculates the producers' available power. The result is then translated into current or power limits

for the main consumers, or used as a trigger to disconnect non-critical loads or start additional generator sets. If this available power is miscalculated or temporarily exceeded—for example, due to the sudden disconnection of a power producer following a fault—the system may remain overloaded for too long because of communication delays. In such cases, the remaining power producers may also trip, leading to a blackout.

Figure 81 presents a system consisting of two loads and a single voltage source, whose output is limited to 100 A, similar to the configuration in Section 3.1.7.1. In this case, however, the loads are not equipped with undervoltage regulators. Once the source reaches its current limit, it switches to current-control mode, R_{MG} becomes infinite, and the DC voltage is no longer regulated. The loads continue to follow their current references, causing the DC voltage to collapse to zero and resulting in a blackout.

Figure 81 demonstrates the system behavior when a voltage source cannot supply sufficient current to maintain the DC-link voltage. As loads L_1 and L_2

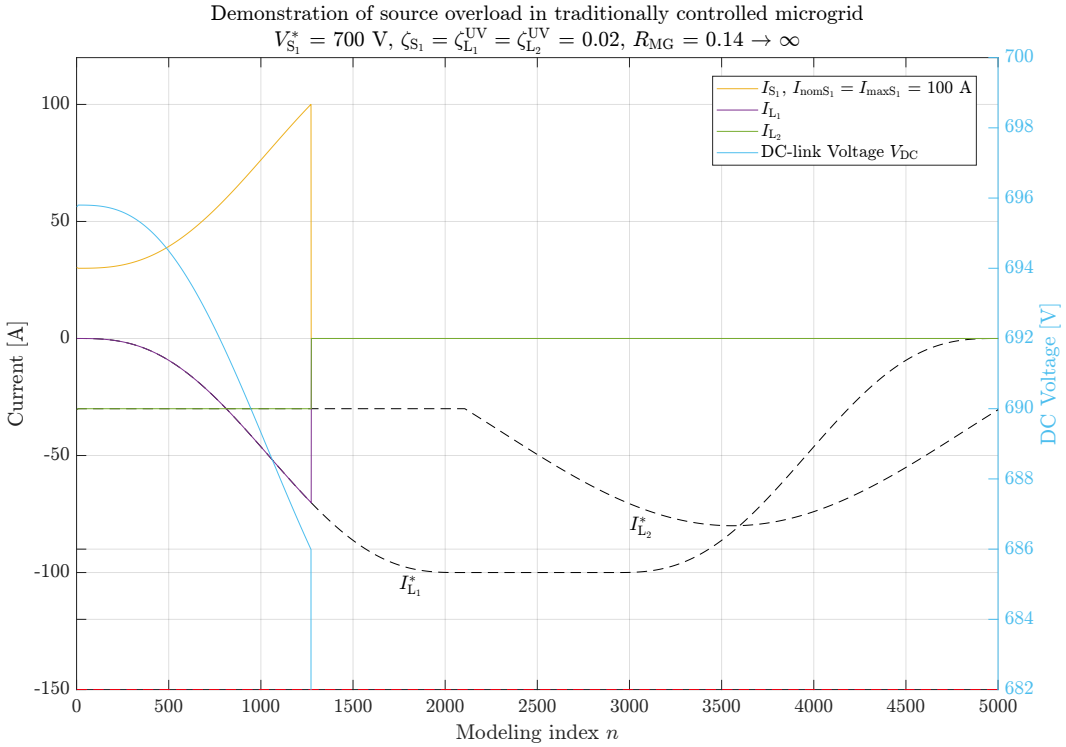


Figure 81. Demonstration of source overload and blackout in a traditionally controlled microgrid, where the control system failed to limit the loads in time.

together draw more current than the source S_1's maximum specified output of 100 A, the voltage collapses and the model stops the inverters. It should be noted that the quasi-static model behavior considers only the current imbalance between the loads and the source when calculating the DC-link voltage. It does not capture the detailed dynamic behavior of the capacitances in the inverters.

To place this in a broader context, it is useful to note that even in the far more common AC shipboard power systems, similar power-limiting functionalities are needed to prevent generator overloads. Modern inverter- or frequency-converter-based full-electric propulsion systems, for example, can independently monitor the ship's AC grid frequency and limit their output accordingly. This capability, however, requires additional measurement devices and is not inherently supported by all inverters. Availability depends on the specific supplier and product, and it is therefore uncertain whether equivalent functionality can be assumed across all devices.

In contrast, with DC microgrids employing the variable voltage-based power management strategy, neither continuous power-availability calculations nor additional measurement hardware are required for overload protection of voltage sources. All inverters are able to measure the DC voltage and thus deduce directly from it when and how much they must reduce their demand or support the voltage. With correct parameterization, the system becomes highly reliable and robust against power shortages.

5.2.1 Loads' Automatic Support of the Voltage

With the variable voltage-based power management strategy, every inverter in the DC microgrid can deduce the loading situation directly from the DC-link voltage. Source overload situations are handled autonomously by the loads via their undervoltage regulators, without intervention from the central control system (see Figure 82). This functionality is inherent to the inverters themselves: as long as an inverter is operating, it can support the grid voltage. The supervisory control system is therefore not required to perform continuous low-level control of the inverters at fast time scales. Instead, it only defines the operational boundaries in the form of DC voltage references,

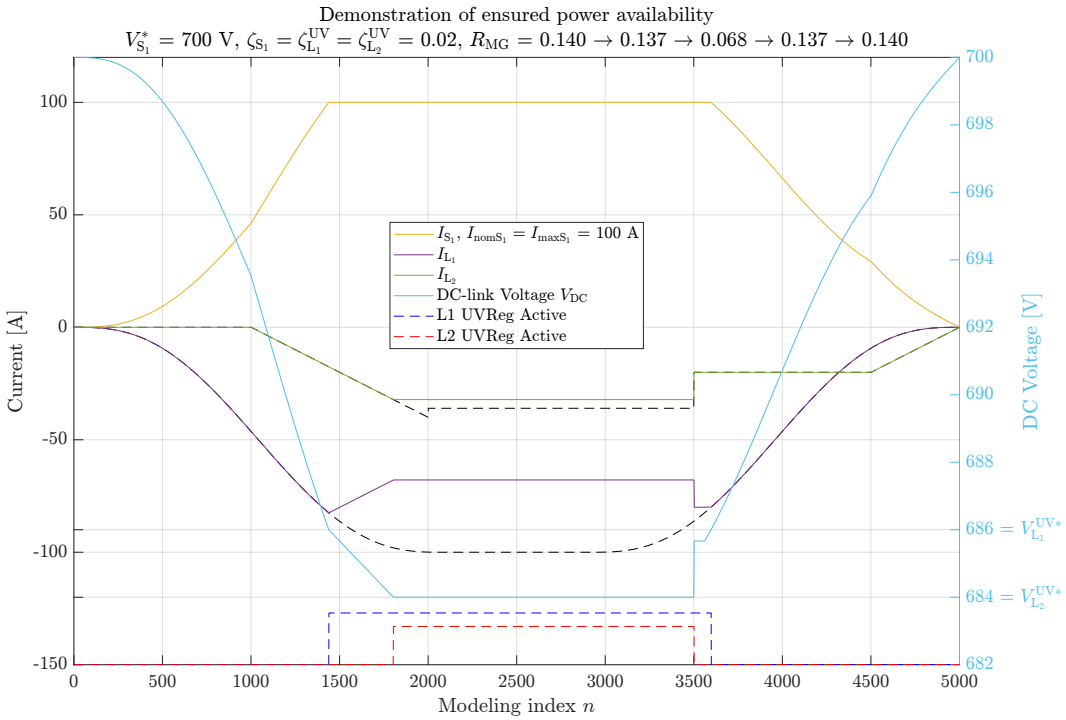


Figure 82. Demonstration of inverters' independent load adjustment by voltage regulators, to ensure the voltage source is not overloaded.

undervoltage and overvoltage regulator limits, and droop coefficients within which the inverters operate.

In the example of Figure 82, the system's only voltage source S_1 is limited to its nominal current of 100 A, as in Figure 81. The modeled loads follow predetermined current references: Load L_1 varies between 0 A and 100 A, while Load L_2 varies between 0 and 40 A. L_1's undervoltage regulator activation threshold is set to 686 V with 2% droop, corresponding to the DC-link voltage when S_1 is at full load with its 700 V reference and 2% droop. L_2's undervoltage regulator is set slightly lower, at 684 V with the same droop.

As the combined load increases to 100 A, S_1 reaches its nominal current and the DC-link voltage falls to 686 V, activating L_1's undervoltage regulator. As L_2's demand continues to rise, L_1 reduces its own load, lowering the DC voltage further. At 684 V, L_2's regulator also activates, stabilizing the system. When L_2's reference decreases, its regulator contribution becomes unnecessary and it disengages. As L_1's reference decreases further, the DC-link voltage rises above its activation threshold and L_1 also disengages, returning both loads to normal current-controlled operation.

Because the source current was fixed at 100 A, S_1 effectively operated as a current source. The loads then adjusted their operation automatically to maintain DC-link stability and prevent blackout. The DC voltage remained within safe limits, and the loads were utilized up to their maximum feasible contribution. Once the load references decreased sufficiently, the regulators disengaged simultaneously with the source leaving current-limited operation, and the system returned to normal operation.

It should be emphasized that the system behavior in this scenario is critically dependent on the undervoltage and overvoltage regulators. These regulators are integral to the variable voltage-based power management strategy, as they enable the loads to support the grid voltage when required. Their implementation must therefore be predictable and consistent for correct load sharing and voltage support. The regulator models used in this and the following examples (unless otherwise specified) are those described in Section 3.1.7.1, which incorporate conditional droop compensation.

For comparison, Figure 83 shows the same scenario with undervoltage regulators implemented without conditional droop compensation. In this case,

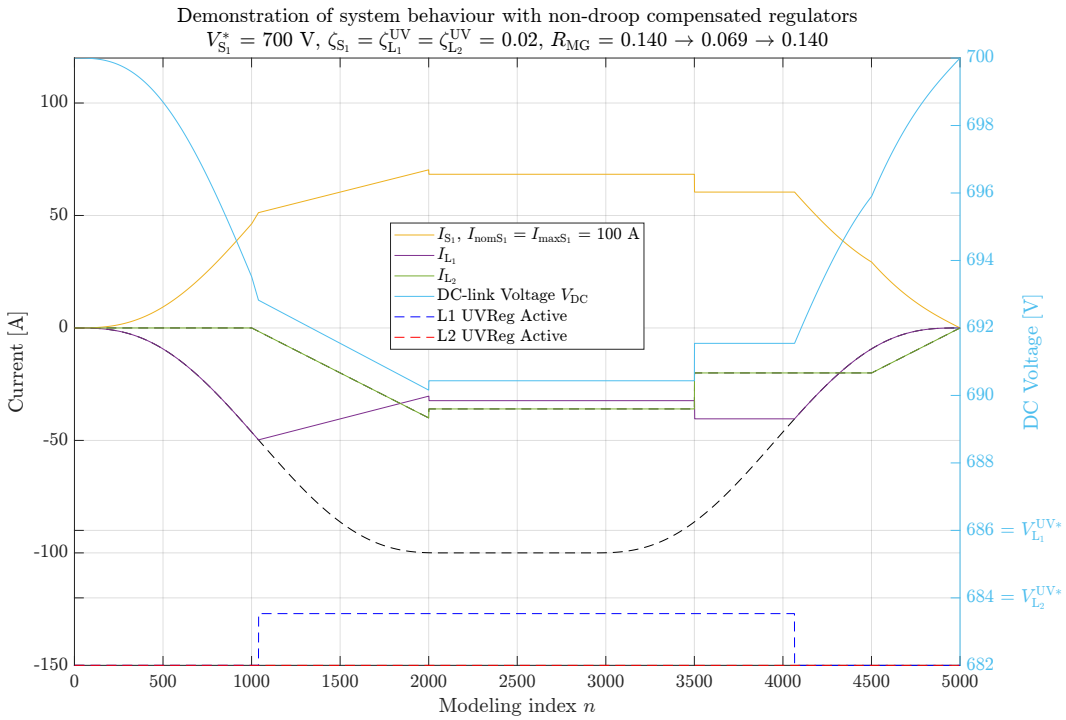


Figure 83. Demonstration of undervoltage regulator operation without conditional droop compensation.

L_1's regulator shifts its activation threshold upward to such an extent that it activates already when S_1 is loaded to only about 50% of nominal capacity. L_2's regulator never activates, and S_1 is never utilized near its full potential.

5.2.2 Automatic Stand-by Voltage Source Startup

Startup of a new power source, such as a generator set, can be directly linked to the DC-link voltage. Because droop control distributes load proportionally, a given DC-link voltage corresponds to a specific loading percentage of the active voltage sources. This enables straightforward trigger conditions for automatic startup of additional sources, provided that all voltage sources share the same DC voltage reference and droop coefficient.

Figure 84 presents a modeled scenario of generator startup based on DC-link voltage. Initially, up to $n = 1000$, Load L_1 consumes 50 A, supplied by a 75 A voltage source. As the load begins to increase, the primary source

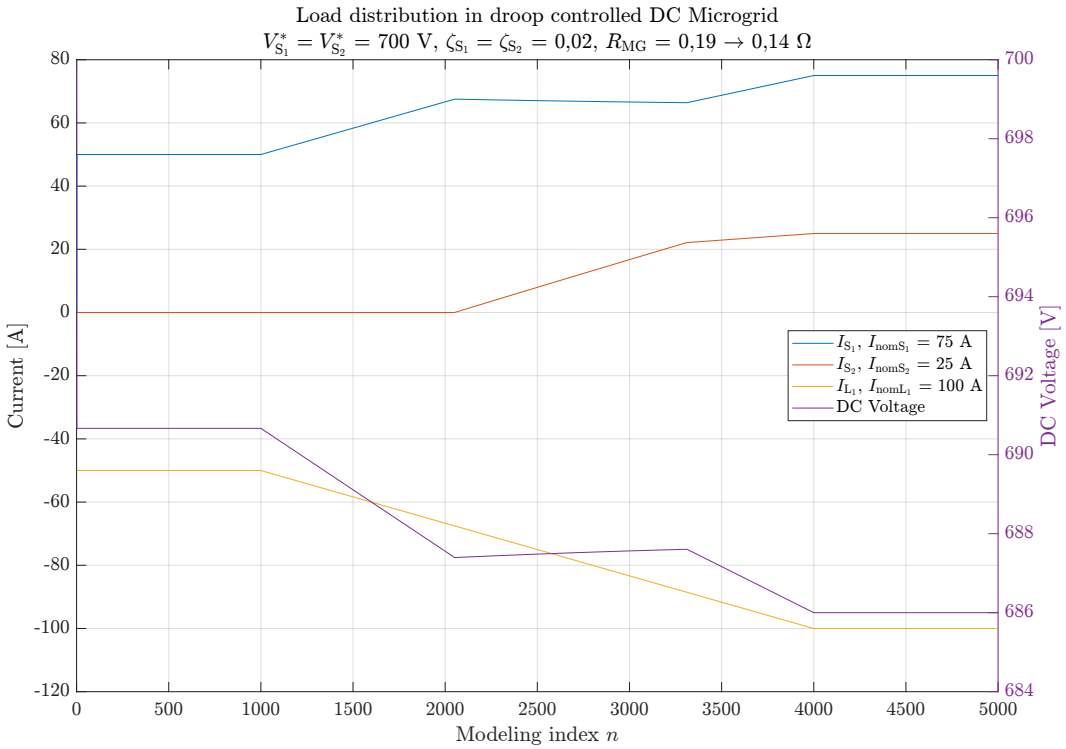


Figure 84. Modeled situation where load increases steadily from 50 to 100 A and a secondary voltage source is automatically started when the primary source reaches 90% of capacity.

reaches 90% of capacity at $n \approx 2050$.

The secondary 25 A voltage source, representing the standby generator set, is programmed to start when the DC-link voltage falls to 90% of the drooped value:

$$V^*_{\mathrm{S}_1} \cdot \left(1 - 0.9 \cdot \zeta_{\mathrm{S}_1}\right) = 700\,\mathrm{V} \cdot \left(1 - 0.9 \cdot 0.02\right) = 687.4\,\mathrm{V}. \tag{5.1}$$

This voltage level corresponds to the situation where the operating source is at 90% of nominal load, which occurs at $n \approx 2050$. Since all voltage sources share load proportionally when they have identical voltage references V^* and droop coefficients ζ, the DC-link voltage provides a direct and unambiguous indication of system loading, regardless of the number of active sources.

Between $n \approx 2050$ and $n \approx 3300$, the 25 A source is ramped up smoothly by increasing its DC voltage reference from the actual DC-link level to the common reference of 700 V used by the 75 A source. Between $n \approx 3300$ and $n = 4000$, the load continues to rise, and the droop-controlled sources allow the DC-link voltage to decrease until the system reaches 100 A at $n = 4000$. Because the standby source entered operation in a controlled and autonomous manner, system operation remained uninterrupted and no external control intervention was required.

5.3 Automatic Limitation of Excess Power Generation

Traditionally, backpower scenarios have most often resulted from exceptional situations, such as the sudden loss of a large consumer due to a fault trip or emergency stop. Combustion engine – based generator sets are relatively slow to respond in such cases because of their inertia, and may themselves enter backpower and trip. This makes management of backpower events more challenging for the control system than overload situations, where a coarse but effective solution is simply to shed loads. While this section concentrates on excessive power generation from renewable sources, the same principles can also be applied to conventional backpower events through the use of DC/DC converters and energy storage systems.

Until now, only DC voltage drops caused by drooping have been considered.

It is important to note that, depending on the relative mix of sources and loads, drooping can equally result in a voltage rise. This becomes increasingly relevant as renewable sources such as solar power are integrated into shipboard systems. Renewables are typically operated to extract the maximum available power at all times, since it is effectively cost-free, by running their inverters in current-controlled mode with maximum power point tracking. This may lead to situations where production exceeds consumption, and without a corrective mechanism, the voltage sources onboard may experience backpower and trip.

Within the principles of the variable voltage-based power management strategy, there are two approaches to preventing backpower. One approach is to assign the current source an overvoltage regulator threshold $V^{\mathrm{OV}*}$ slightly below the voltage source reference V^*, ensuring that the voltage source output remains marginally positive. Alternatively, $V^{\mathrm{OV}*}$ can be set significantly above V^* while simultaneously imposing a minimum current limit on the voltage source, forcing it into current-controlled operation rather than backpower. This latter method deliberately allows a temporary condition of excess power generation, causing the DC-link voltage to rise and thereby activating load-side overvoltage regulators. In this controlled manner, additional voltage margin is created and unstable interactions between sources are avoided.

As with undervoltage regulators in Section 5.2.1, the correct operation of this functionality depends critically on the implementation of the overvoltage regulator. The modeled scenarios presented in the following subsections therefore employ the same regulator implementation described earlier in this work.

5.3.1 Limiting by Overvoltage Regulator Only

The same principle used throughout the variable voltage-based power management strategy also applies here: inverters can deduce their required action directly from the DC-link voltage. Current sources, such as renewable power units, can be modeled as positive loads with their own under- and overvoltage regulators. As noted earlier, motor loads are not directly con-

trollable by the system, and the same applies to renewable sources: they are designed to operate autonomously, for instance by following a maximum power point tracker in the case of a solar inverter. Just as normal loads trigger their undervoltage regulators when the DC-link voltage falls too low, positive loads can trigger their overvoltage regulators when the DC voltage rises too high, thereby limiting their output independently of the central controller (see Figure 85).

In the example of Figure 85, the system consists of a single 100 A voltage source, a 50 A constant base load, and a current source, modeled as a negative load, whose output varies between 20 and 60 A. The voltage source reference is 700 V with 2% droop, while the current source overvoltage regulator is set to 699 V. For this approach to function, the regulator must be set extremely close to the voltage source reference; otherwise, the source either cannot reduce its output sufficiently, or it reduces its output excessively, leading to backpower.

This requirement is problematic. In the example, the regulator must be set just 1 V (0.14%) below the source reference to achieve a practically reason-

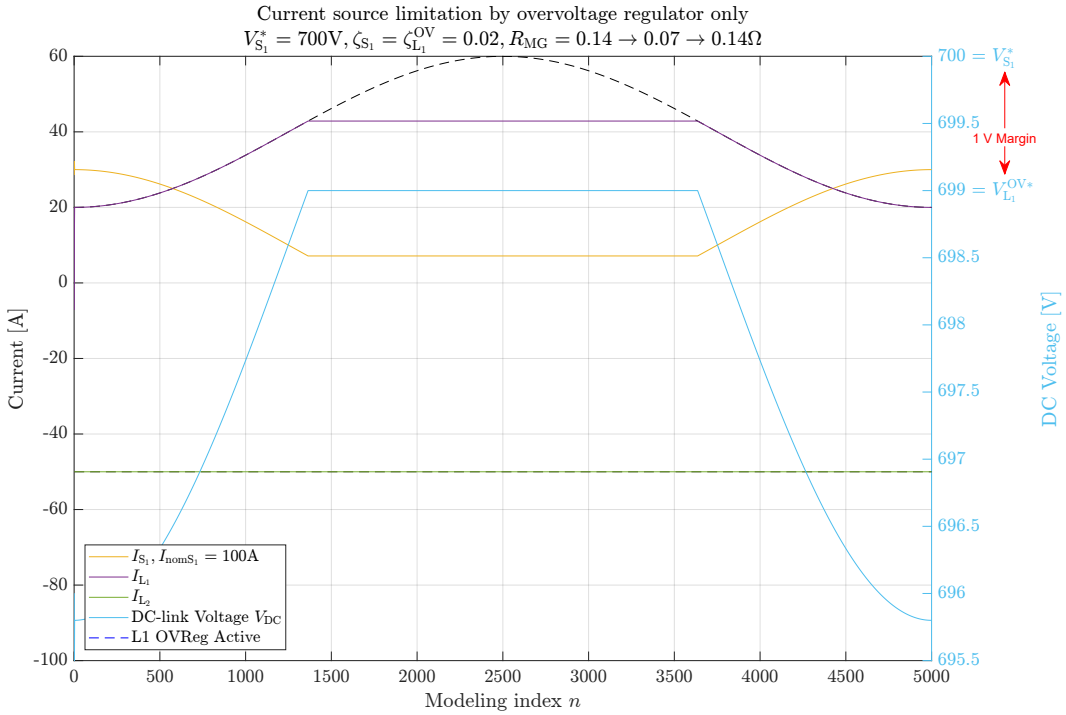


Figure 85. Limiting voltage source unloading by utilizing only a current source's overvoltage regulator.

able 7% minimum voltage source output. In reality, no voltage source can operate reliably within such a narrow margin. The result would be strong fluctuations of the source output around zero, as it attempts unsuccessfully to share load with the current source, which is effectively operating as a voltage source. In practice, this would likely cause the load to switch repeatedly between current-controlled and voltage-controlled operation, introducing further disturbances into the system.

The underlying requirement is not about fine-tuning load sharing but about preventing the voltage source from being driven into backpower. Using regulators alone to achieve this would necessitate unrealistically large droop coefficients in order to create sufficient voltage margins. While voltage regulators are essential for enabling inverters to act on DC-link voltage deviations, additional mechanisms are required to provide the necessary operating margin for stable backpower prevention.

5.3.2 Assigning Minimum Current for Voltage Source

Another option to control current source output and prevent the voltage source from entering backpower is to impose a minimum output on the voltage source. For example, limiting the voltage source minimum output to 7% of nominal, as in Figure 86, forces the voltage source to change to current-controlled operation when output reaches that level, thereby preventing its output from falling too low.

Since the voltage source cannot reduce its output below the minimum, while the current source continues injecting current into the DC-link, the DC voltage rises steeply once generation exceeds consumption. At the point where the voltage source reaches its current limit and transitions to current control, there is momentarily no device actively regulating the DC-link voltage. This deliberate rise in voltage allows the current source to be assigned a significantly higher overvoltage regulator threshold than in the case of Figure 85, thereby creating a larger operating margin and improving system stability. Consequently, the voltage source does not oscillate around zero output in an attempt to share load with the current source.

In the example of Figure 86, the current source's overvoltage regulator is set

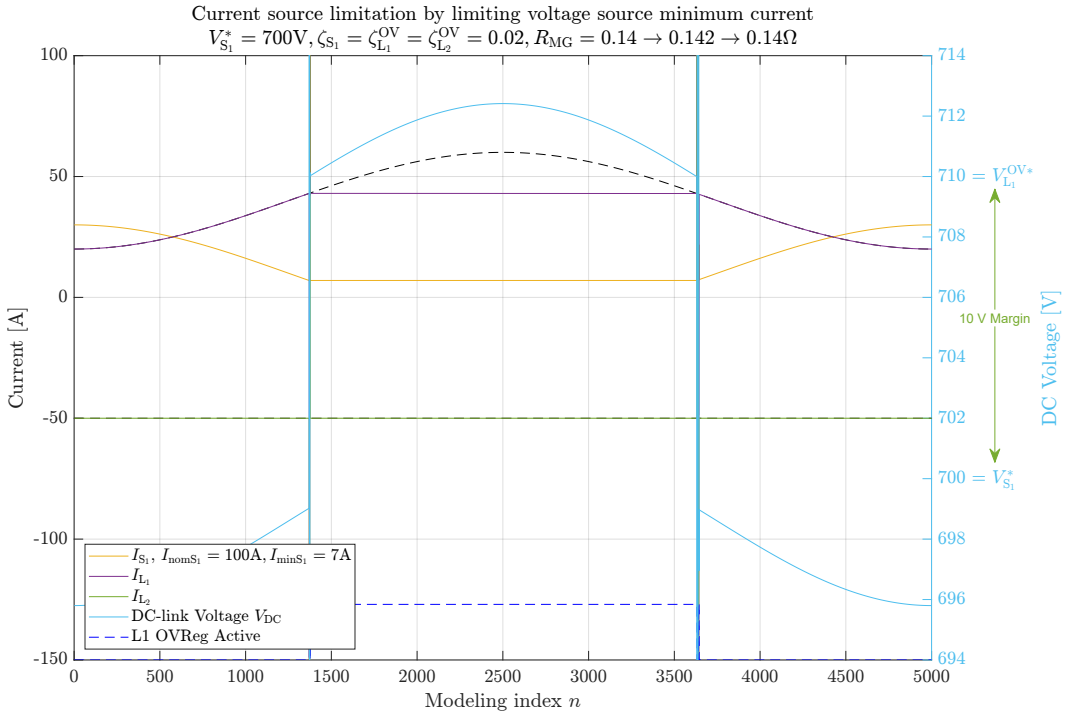


Figure 86. Limiting voltage source unloading by imposing a minimum current output and allowing the DC-link voltage to rise.

to 710 V. This makes the margin between the voltage reference and the regulator limit ten times larger than in Figure 85, giving the system substantially more flexibility. Furthermore, because the voltage source effectively operates as a current source, its output remains much more stable, as it is no longer competing with the current source for load sharing.

Figure 86 also shows unstable regions around the moments when the voltage source and the load change their operation modes. These artifacts result from the quasi-static modeling, since during the transitions R_{MG} is momentarily infinite and the DC voltage mathematically tends toward $\pm\infty$, depending on the loading situation. In the model, Code 3 limits the maximum DC voltage to 1,000 V to prevent numerical divergence in such situations, with the overvoltage regulator subsequently constraining the otherwise uncontrolled voltage. In real systems, inverter capacitors introduce dynamics that allow smoother transitions between operation modes.

5.4 Power Supply Prioritization

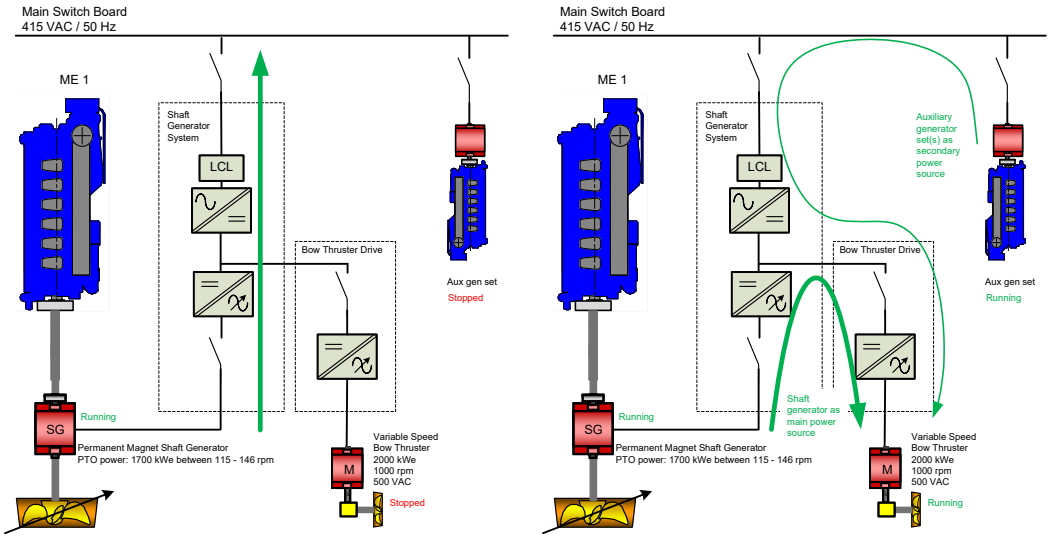
Power supply prioritization is a fundamental requirement of any practical microgrid control strategy. From both ecological and economic perspectives, it would be unacceptable to adopt a strategy that cannot prioritize the use of more favorable power sources over less favorable ones. It has been noted that droop control alone does not provide supply prioritization (Ko, Lee, Choi, and Kim (2019)), as its purpose is to distribute load rather than to rank sources. While this statement is correct, it is not the full picture. Supply prioritization can in fact be achieved by exploiting the variable voltage behavior of the DC microgrid induced by droop control.

Two distinct scenarios may be described as supply prioritization. The first is a planned condition, where one source is fully loaded before another begins contributing. The second is an exceptional condition, where a backup source must react immediately to a disturbance; this latter case is addressed separately in Section 5.6.

This section focuses on the first case: intentional sequencing where the secondary source remains unloaded until the primary is fully utilized. This differs from the situation in Section 5.3, where a renewable source operates as a current source while a generator set continuously maintains the DC-link voltage. In that configuration, increased renewable output reduces the generator contribution but does not replace it entirely, and vice versa. Here, by contrast, an increase in load raises the output of the primary source until its limit is reached, at which point the secondary source begins supplying. As load decreases, the secondary source reduces its output first, followed only later by the primary.

5.4.1 Victorian Reliance II: Real-life Use Case on Controlled Prioritization of Primary and Secondary Power Sources

A practical example of controlled source prioritization was presented by Alho and Laaksonen (2020), implemented on the ships *MV Tasmanian Achiever II* and *MV Victorian Reliance II*, as illustrated in Figure 87. The system consisted of three inverters: one connected to a 1.7 MW shaft generator, one to



(a) Power flow during PTO operation while sailing.

(b) Power flow when thruster is running.

Figure 87. Principle-drawing of presented inverter system utilizing voltage-based power supply prioritization.

a 2.0 MW bow thruster motor, and one to the AC grid. During normal sailing operation, the shaft generator operates in Power-Take-Out (PTO) mode, meant for supplying electrical power to the ship grid (Figure 87a). When the bow thruster is started, however, the export of power to the AC grid is ramped down, and the shaft generator output is redirected entirely to the motor. Only when the generator output is insufficient to cover the motor demand does the AC grid side inverter contribute (Figure 87b).

The single-line diagram in Figure 87 originates from the shaft generator system supplier (WE Tech Solutions Oy). As the supplier was not involved in the selection or design of the auxiliary generators, their power ratings were not known at the time of diagram preparation. The purpose of the diagram in this work is solely to illustrate the basic principles of the two operational scenarios; therefore, exact details—for example, breaker closing statuses—are omitted, as such information can be deduced from the depicted power flows.

In this configuration, the generator-side inverter must actively maintain the DC-link voltage, i.e., it serves as the system's voltage source. The grid-side inverter is allowed to contribute to the DC-link only if the motor demand exceeds the generator's nominal capacity; otherwise, it remains idle, neither

supplying to nor absorbing from the DC-link. The generator inverter must also be prevented from exceeding its nominal rating.

The generator inverter is the sole voltage source and operates with droop control, meaning that the DC-link voltage directly reflects its loading. The grid-side inverter is operated in current-control mode with zero reference, effectively acting as a load, but equipped with an undervoltage regulator set to activate at the DC-link voltage corresponding to the generator at nominal load. No droop characteristic is applied to this regulator, since the inverter is not intended to share load proportionally but only to supplement the generator once its limit is reached. If droop were applied, the generator load would continue to increase with motor demand, which would contradict the prioritization requirement.

5.4.2 Modeling the System From Motor Load

The mathematical model of the system is constructed from the motor loading data recorded onboard and reported in Alho and Laaksonen (2020) (Figure 88), the known powers in Figure 87, and the modeling principles introduced in Section 3.1. The thruster motor has a nominal power of 2,000 kW and the generator inverter 1,700 kW, leaving the grid-side inverter to supply the remaining 300 kW. Its nominal power must therefore be at least this value.

As demonstrated in Figure 55, the use of power references does not, with the applied droop coefficients, significantly affect the inverter currents. The modeling framework is, however, expressed in terms of nominal DC current ratings. In practice, such ratings are not directly available, since inverter parameterization is typically specified for the AC side. To resolve this, a nominal DC voltage is assigned for the sole purpose of converting the known power ratings into nominal DC currents required by the model. The absolute values of the resulting currents are not physically meaningful, as they depend on the arbitrarily chosen nominal voltage. Nevertheless, by consistently using the same nominal voltage when converting these currents back into powers, the modeled powers correctly represent the behavior of the system.

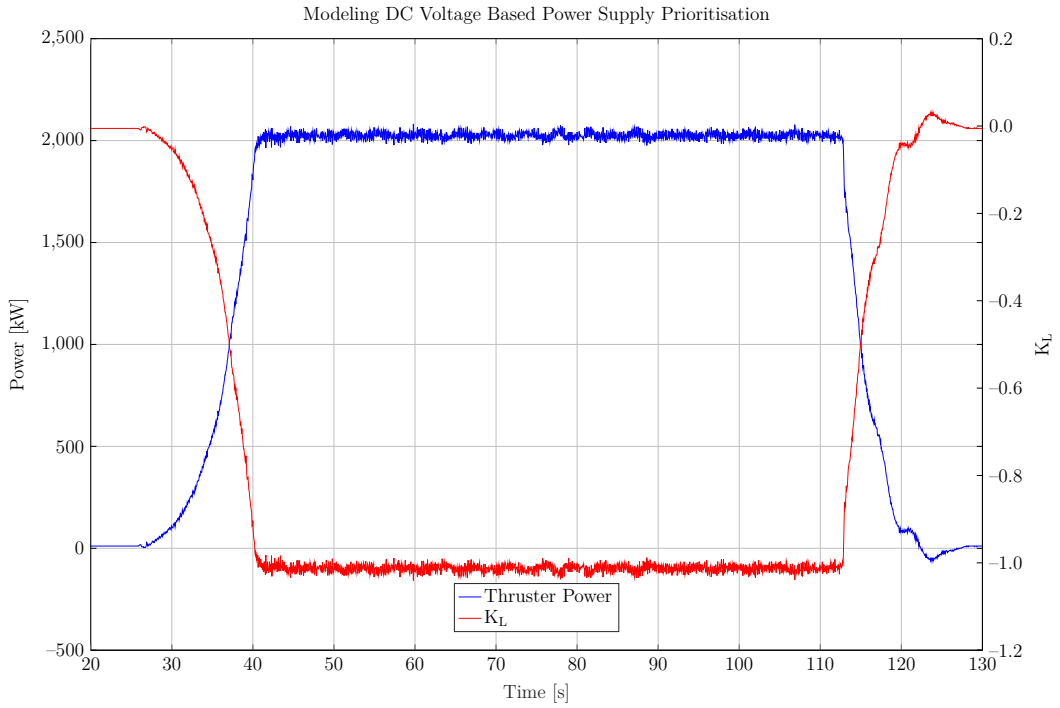


Figure 88. Motor load data used as the basis for mathematical modeling of the prioritization system.

The motor load data is provided in kilowatts and is transformed into the normalized load factor $\mathrm{K_L}$ required for the calculations by dividing the power by the negated nominal motor power of 2,000 kW, consistent with the adopted load sign convention:

$$\mathrm{K_L}(t) = \frac{P_\mathrm{L}(t)}{-2{,}000\,\mathrm{kW}}. \tag{5.2}$$

The grid-side inverter, acting as the auxiliary source when the thruster is running, is modeled as a current-controlled device whose participation is enabled only when the DC-link voltage falls below the activation level of its undervoltage regulator. The behavior is represented according to the principles in Section 3.1.9:

$$\mathrm{K_{Grid}} = \begin{cases} 0, & \text{if } V_\mathrm{DC} > V_\mathrm{Grid}^{\mathrm{UV}*}, \\ -\dfrac{I_\mathrm{L} + I_\mathrm{Gen}}{I_\mathrm{nomGrid}} \in [0,\infty), & \text{otherwise}, \end{cases} \tag{5.3}$$

where $V_\mathrm{Grid}^{\mathrm{UV}*}$ is the grid-side inverter's designated undervoltage regulator activation limit, I_L is the modeled bow thruster inverter current, I_Gen is the modeled generator inverter current and I_nomGrid the modeled grid-side in-

verter nominal current.

This representation makes the grid-side inverter a conditionally activated current-controlled device that effectively behaves as a negative load, without a detailed physical model. It simply compensates the current difference between the bow-thruster inverter and the generator inverter. As shown later, this produces an unnaturally stable DC voltage because the inverter's output is prescribed rather than derived from a control model. Since the inverter output is scaled algebraically, the exact nominal current rating is not critical, provided the scaling is applied consistently. In this modeling, the grid-side inverter nominal power is set to 1,000 kW.

The parameters used in the modeling are summarized in Table 15. Circulating currents are neglected in order to isolate the prioritization behavior. The generator inverter voltage reference V^*_{Gen} of 787 V was determined from the original zero-load data, where droop is absent. From the same data, the droop coefficient was calculated as 2%, based on the ratio of DC voltages at zero load and at 1,700 kW load. Although Alho and Laaksonen (2020) reports a droop of 3% according to the inverter parameterization, the measured voltage characteristics are consistent with a value of 2%. The reason for this discrepancy has not been established and lies outside the scope of this work. Since the objective here is to reproduce the physical behavior of the system rather than the exact inverter parameter settings, a droop coefficient of 2% is adopted.

As seen in Figure 88, the real-life data naturally contains a time element, since the data points were recorded at specific intervals and can be mapped

Table 15. Parameters used in the power supply prioritization model.

Parameter	Value
Voltage Rating V_{rated}	$400\,\text{V}$
P_{nomGen}	$1{,}700\,\text{kW}$
P_{nomGrid}	$1{,}000\,\text{kW}$
P_{nomL}	$2{,}000\,\text{kW}$
I_{nomGen}	$P_{\text{nomGen}}/V_{\text{rated}} = 4{,}250\,\text{A}$
I_{nomGrid}	$P_{\text{nomGrid}}/V_{\text{rated}} = 2{,}500\,\text{A}$
I_{nomL}	$P_{\text{nomL}}/V_{\text{rated}} = 5{,}000\,\text{A}$
V^*_{Gen}	$787\,\text{V}$
ζ_{Gen}	0.02
$V^{\text{UV}*}_{\text{Grid}}$	$V^*_{\text{Gen}}(1.0 - \zeta_{\text{Gen}}) \approx 771.3\,\text{V}$
R_{MG}	$V^*_{\text{Gen}}\zeta_{\text{Gen}}/I_{\text{nomGen}} \approx 0.004\,\Omega$

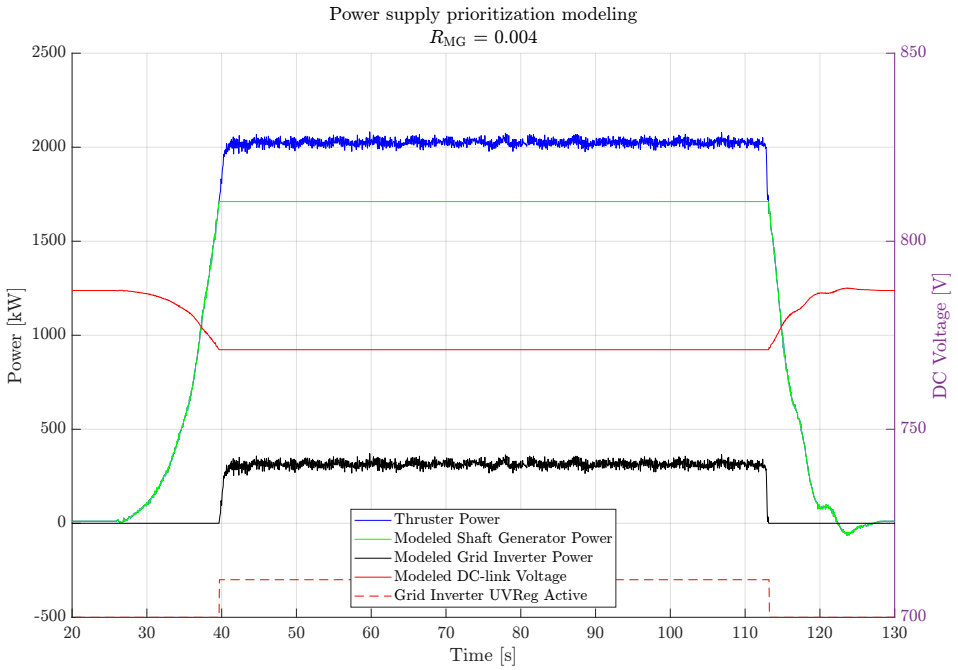
across a time window. However, as mentioned in Section 3.1.3.1, the time aspect is irrelevant for quasi-static modeling. In the figures presenting the modeling results in later sections, time is shown on the x-axis only because the corresponding real-life data is included as well. In the modeling itself, each data point simply represents a step in the calculation, without any implication of the elapsed time between steps.

The real-life inverters naturally include DC-link capacitors, which introduce short-term dynamics into the system that quasi-static modeling cannot replicate. As discussed in Section 1.1.1 and shown in Sections 3.1.4.1 and 5.2, the absence of such dynamics in the modeling paradigm leads to instantaneous steps in voltage and current without transients. The capacitors in the inverters serve to smooth rapid variations in the DC voltage, ensuring stable operation of the power electronics, and to provide a limited energy buffer against sudden large load steps. Apart from the special case where an inverter is disconnected while its capacitors are fully charged, in which case they may discharge over several seconds, the influence of the capacitors during normal operation is practically irrelevant at the time scales of seconds. For example, with a bow thruster acceleration ramp lasting 15 s, the capacitors have a negligible effect on the DC voltage behavior.

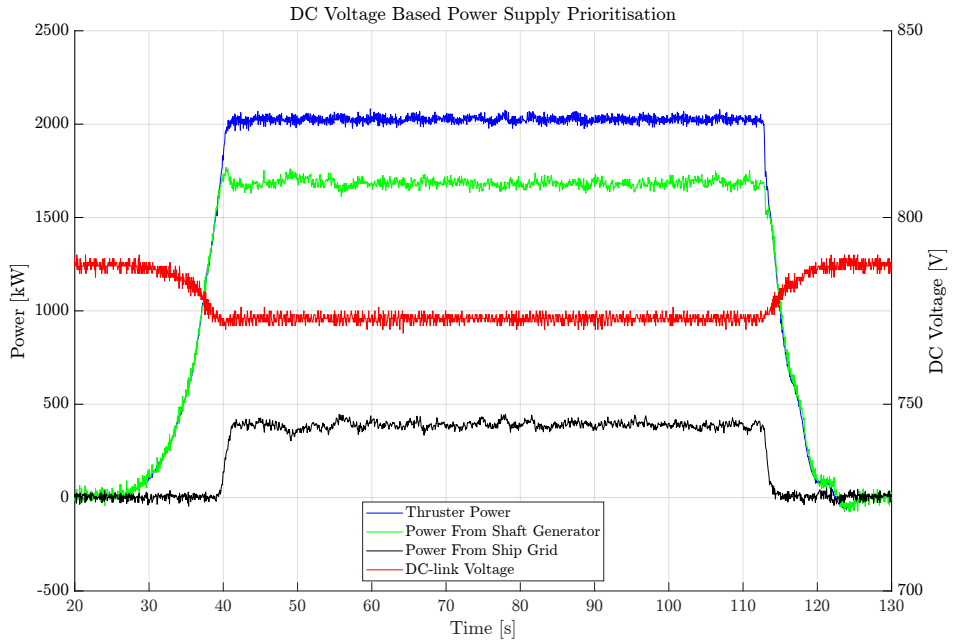
It should be also emphasized that the inclusion of DC-link capacitors would not affect the steady-state results. For example, if the quasi-static modeling yields a DC-link voltage of x V in a condition where the system has remained unchanged for several seconds, a time-domain model including capacitances will converge to exactly the same value. The capacitors influence only the transient trajectories, not the equilibrium points.

5.4.3 Model Analysis and Comparison With Real-life

Figure 89a presents the modeling results, where the blue line is the measured bow thruster power and the rest of the signals are modeled based on it and the defined modeling parameters. The load sign here is inverted compared to the Section 3.1 as to match with the data from Alho and Laaksonen (2020). Appendix 14 presents the MATLAB code for performing the modeling.



(a) Modeled DC voltage-based power supply prioritization using load data.



(b) Real-life DC voltage-based power supply prioritization.

Figure 89. Comparison of (a) modeled DC voltage-based power supply prioritization with (b) real-life data.

As the undervoltage regulator of the grid-side inverter is modeled as rigid with zero drooping, as described in Section 3.1.9, it immediately supplies all excess power not provided by the generator inverter once activated, see

Figure 89a. This produces artificial zero watt difference in power consumption and production and a somewhat unnatural generator inverter output and DC voltage, which appear artificially flat when the grid-side inverter begins contributing. On a fundamental level, however, this is consistent with real-life operation, as seen in Figure 89b.

The measured traces are noisier, but the overlaid comparison in Figure 90 shows close agreement between modeled and measured behavior. During undervoltage-regulator operation, the modeled and measured DC voltages differ by approximately 3 V. This offset appears only when the grid-side inverter's voltage regulator is active, suggesting that it originates within the inverters (e.g., measurement calibration or regulator bias). The 3 V deviation is too small to account for the discrepancy between the modeled and measured grid-side inverter powers. In the model, the grid-side inverter exactly compensates the power difference between the generator- and motor-side inverters so that the net power imbalance is zero. Because the modeled generator-side inverter power aligns reasonably well with the measurements, the remaining difference in grid-side inverter power is attributable to unmodeled losses (converter and cabling), waveform/filter effects, and measurement errors.

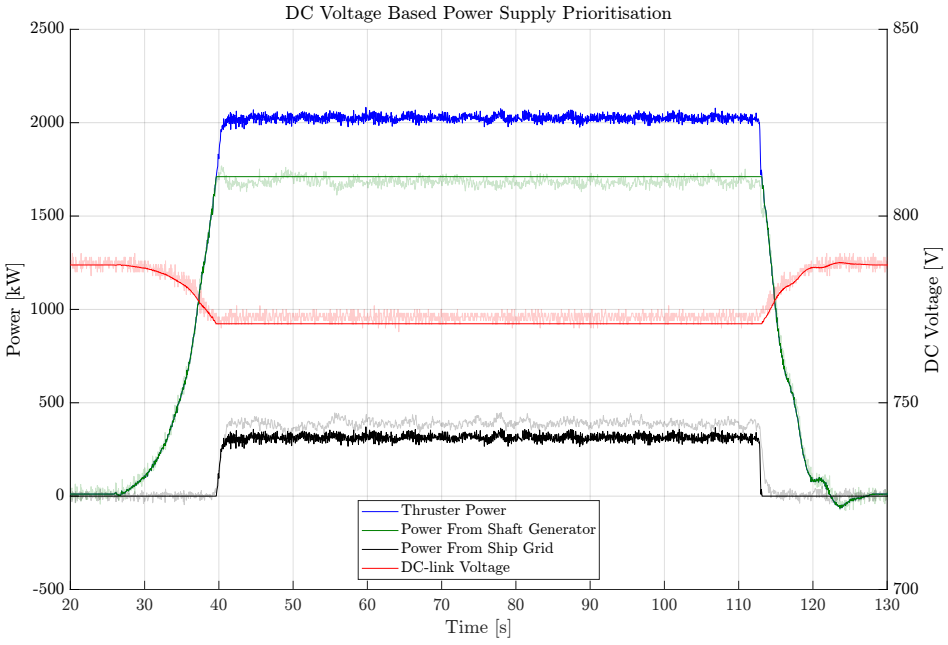
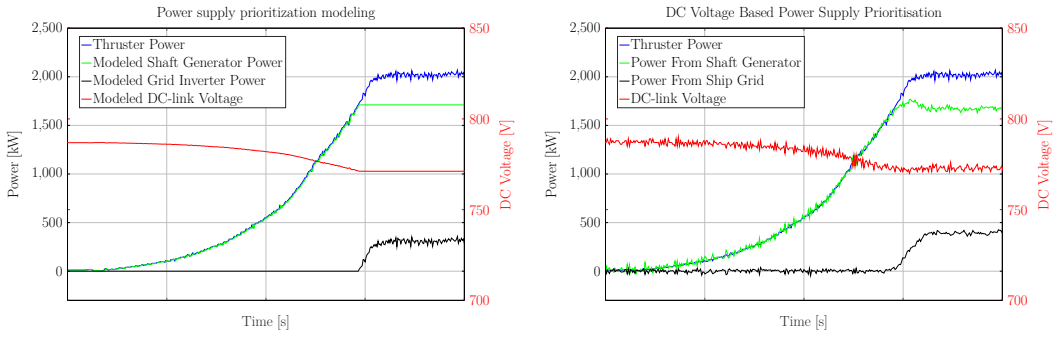


Figure 90. Overlaid modeled and real data from power supply prioritization.

In the model, when the motor load increases, the generator inverter maintains the DC-link voltage. Due to droop control, the voltage decreases as generator loading increases. Only once the voltage reaches the activation limit of the grid-side inverter's undervoltage regulator does it begin contributing to the DC-link. This process is shown in more detail in Figure 91a. When the motor load decreases, the grid-side inverter is unloaded first, prior to the generator inverter. Thus, the modeled functionality reproduces exactly the intended prioritization behavior and closely matches the real-life measurements.

In the real-life data, practically identical behavior is observed, with only the earlier mentioned slight difference in power distribution (see Figure 91b). Although some noise is present in the DC-link voltage, the undervoltage regulator activates cleanly and begins supporting the DC-link voltage. After a brief overshoot, the generator inverter stabilizes at approximately 1,700 kW.

When analyzing the modeled behavior in the voltage domain, the linear effect of droop control is clearly visible as the DC-link voltage decreases with increasing generator loading. Once the grid-side inverter's undervoltage regulator activates, its output increases vertically, since the DC-link voltage is no longer permitted to fall further. This ensures that the generator inverter load remains stable. If the DC voltage were allowed to decrease further, the generator inverter would continue to share more load, contrary to the prioritization objective.



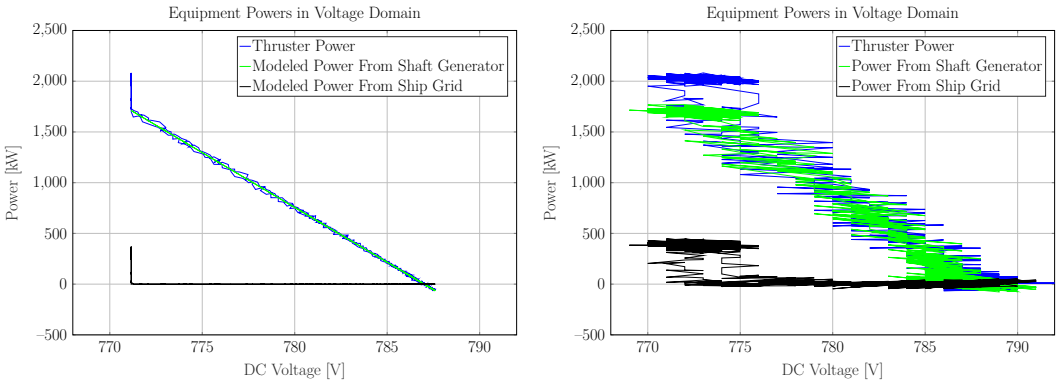
(a) DC voltage-based power supply prioritization modeled using load data only.

(b) Real-life DC voltage-based power supply prioritization operation.

Figure 91. More detailed comparison of (a) modeled DC voltage-based power supply prioritization with (b) real-life data.

The limitation of generator output is therefore achieved through the non-drooped undervoltage regulator, rather than by imposing a strict maximum power limit on the generator inverter. It is generally unwise to enforce a hard output ceiling on power-generating units, since all such units can tolerate moderate overloads for limited durations before for example the generator windings start heating up. This operational flexibility is critical to avoid situations where even a minor, momentary overload could otherwise cause an unnecessary blackout just like in Figure 81.

Due to the measurement noise in the powers and the DC-link voltage, the analysis of the real-life data is less distinct than in the modeled case, see Figures 92b and 92a, respectively. Nevertheless, a clear correspondence is observed between the two data sets: the DC-link voltage decreases linearly as the generator inverter power increases, and once the grid-side inverter's undervoltage regulator activates, the voltage drop halts. From the figures it can be estimated that the total voltage drop from 0 to 100% load with 2% droop is about three times larger than the amplitude of the noise in the DC-link voltage measurement. This suggests that a 2% droop represents approximately the minimum practical setting for the system while still providing a sufficient control margin.



(a) DC voltage-based power supply prioritization modeled using load data only.

(b) Real-life data on DC voltage-based power supply prioritization.

Figure 92. Voltage domain comparison of (a) modeled DC voltage-based power supply prioritization with (b) real-life data.

5.4.4 Conclusions on the Modeling Paradigm Usability

The comparison between simulation results and real-life data demonstrates that the proposed modeling paradigm provides an accurate representation of system behavior. It can therefore be used by system integrators and inverter suppliers to predict performance under various scenarios prior to laboratory-based control system development. In this way, the modeling can verify that planned control solutions fulfill customer requirements before committing to the potentially time-consuming development and testing of control systems.

As mentioned in Section 3.1.3, the simplified approach of modeling the DC-link voltage without accounting for voltage drops in cables or busbars introduces only negligible error in a single-cabinet system such as the one presented above. As discussed in Section 3.1.10, these voltage drops can nevertheless be incorporated through nodal analysis, further extending the paradigm's applicability and accuracy for larger and more distributed systems.

5.5 Stopping Non-critical Loads

Stopping non-critical loads is a standard action in ship power management to reduce the demand on power-generating equipment. Non-critical loads are defined as those not essential for ship functionality or survivability, and their temporary disconnection in exceptional situations is therefore considered acceptable. The practical distinction between loads that can be limited and those that must be stopped outright depends on the nature of the equipment. For example, non-critical loads may include for example cabin heating and cooling, which are not practical to control individually through limits or references. In such cases, it is more effective to disconnect the entire section of the ship's grid supplying this equipment.

This action may be carried out manually by the crew, in which case it is entirely separate from the ship's automation system. However, the automation system can also be given the capability to disconnect non-critical loads autonomously when certain conditions are met, provided that the crew is

confident it will not do so inappropriately. In this case, automatic load disconnection becomes an integral part of the overall power management or microgrid control strategy.

Within the variable voltage-based power management strategy, the disconnection of non-critical loads operates in a manner analogous to the automatic startup of standby generators described in Section 5.2.2. Inverters supplying non-critical loads are programmed to shut down when the DC-link voltage falls below a defined threshold, as illustrated in Figure 93. This reduces the demand on the voltage sources and causes the DC-link voltage to rise proportionally. For this reason, non-critical loads must not be restarted automatically based solely on the DC-link voltage, unless a sufficient hysteresis margin between the stopping and starting thresholds is implemented. Otherwise, the system risks entering an oscillatory cycle of repeated load connection and disconnection.

Figure 93 presents the modeled operation of a system with one voltage source and one large load of 100 A, as well as two smaller non-critical loads of 5 A and 7 A. Initially, the voltage source operates below its nominal rating while

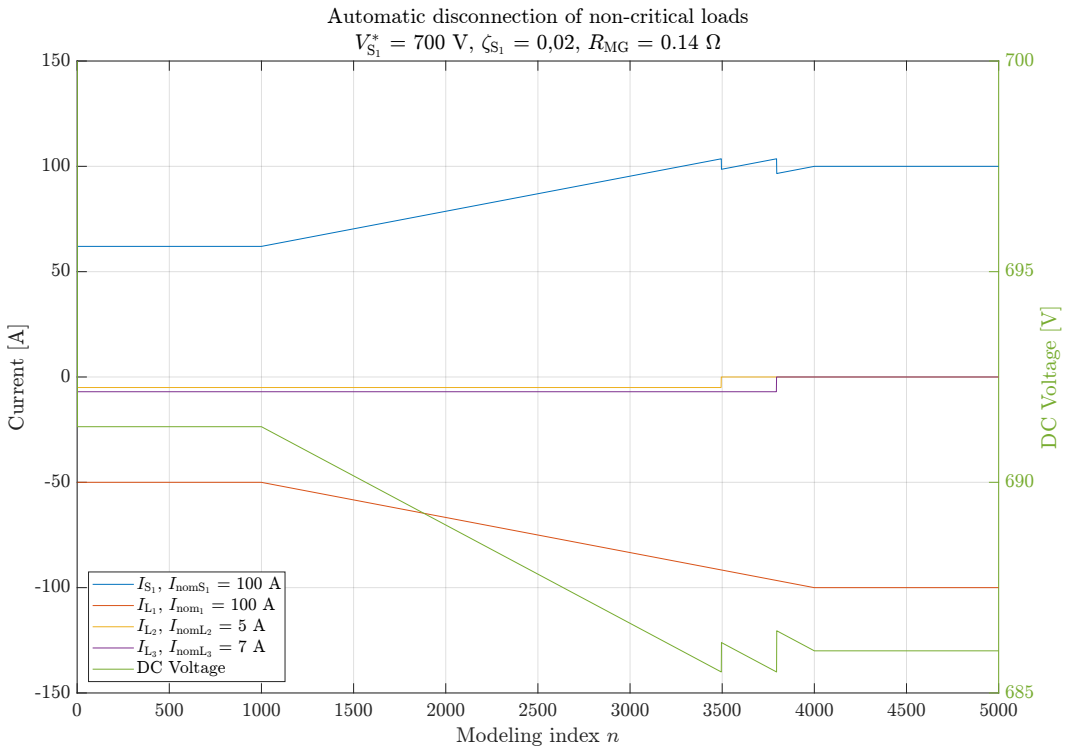


Figure 93. Stopping non-critical loads based on DC voltage level.

the large load is gradually increased. At modeling index $n = 3500$ the voltage source becomes overloaded and the first non-critical load disconnects; at $n \approx 3800$ the second non-critical load disconnects as well. With the two auxiliary loads shed, the voltage source is relieved from overload as the large load reaches 100 A. Disconnection is based on the DC-link voltage, with a predefined priority order ensuring that the second load is disconnected only if the first has already been shed, even though both have the same threshold of 685.5 V.

A key design consideration is that the non-critical loads selected for disconnection should not represent excessively large power steps. In systems relying solely on combustion engine-based generator sets without battery buffering, the sudden disconnection of a large load relative to the engine capacity may cause engine rushing and even engine trip. Careful selection and sizing of non-critical loads are therefore essential for safe and effective implementation of this strategy.

5.6 Blackout Prevention

Blackout prevention refers to managing rapidly occurring situations that create an immediate risk of a total power loss, the clearest example being the sudden disconnection of the ship's last active power source due to a fault. In such cases, immediate actions are required to prevent a complete blackout. Blackout prevention can be supported by equipment with sufficiently fast response and adequate power production capacity, such as a shaft generator operating in boosting mode (providing additional torque to the main engine), or an energy storage system. In most cases, however, load limitations are also required to ensure that the backup source is not overloaded during the transition.

Designing and testing blackout prevention involves addressing a range of challenges. The time window between, for example, a generator disconnection and a total ship blackout is extremely short. Effective prevention therefore requires decision-making at the lowest level of system architecture. Implementing such functionality with centralized microgrid control strategies is difficult: by the time a central controller detects the condition, processes

the measurements, and issues commands to adjust generation and loads, the blackout may already have occurred.

The key challenges include, but are not limited to:

- **Reaction time:** the available window may be as little as a few tens of milliseconds.
- **Power sources:**
  - Is sufficient backup power available?
  - Can the equipment tolerate the sudden increase in power output, or even a reversal of power flow, required by blackout prevention?
  - Can the source respond quickly enough?
- **Power consumers:**
  - How can non-critical loads be disconnected rapidly?
  - How should critical loads that cannot be disconnected respond?
  - Is there a priority order among the critical loads?
  - Is it acceptable that some propulsors or thrusters are completely unloaded in this event?

5.6.1 Variable Voltage-based Blackout Prevention Principle

For the variable voltage-based control strategy, a blackout-inducing event and the associated corrective actions loosely resemble the principle of limiting excess power generation discussed in Section 5.3.2. In the latter case, the DC voltage is allowed to rise uncontrollably until it is stabilized by the overvoltage regulator of a current source. In blackout prevention, by contrast, the DC voltage decreases uncontrollably and is stabilized by one or more loads or other current sources, such as a battery-connected DC/DC converter, through their undervoltage regulators.

As mentioned earlier, one of the key design challenges is the definition of power consumer types and their "priority order," i.e., the sequence in which

consumers begin to unload and by how much. In a purely centralized control strategy, this would require the central controller to calculate in real time how much each consumer must be unloaded and then transmit the new limits, a process for which there is no time. In a distributed control strategy, such as the variable voltage-based approach, this priority order is defined through parameterization of the equipment. Parameterization can be carried out in advance, for example when the ship's operating mode is changed, or even during commissioning for simple systems. Each piece of equipment then reacts independently based on the DC-link voltage.

In practice, the implementation of a priority order is straightforward. One illustrative example is shown below, where voltage thresholds are assigned such that $V_{\mathrm{A}} > V_{\mathrm{B}} > V_{\mathrm{C}} > V_{\mathrm{D}}$:

1. The standby generator receives a start command when the DC voltage falls to V_{A}, as in Section 5.2.2.
2. Non-critical loads are disconnected at voltage V_{B}, as described in Section 5.5.
3. A battery-connected DC/DC converter activates its undervoltage regulator at V_{C}.
4. Finally, if still required, propulsion and thruster inverters activate their undervoltage regulators at V_{D}.

Figure 94 presents an example system depicting the priority order assignments in the above listing. In Figure 94, the nominal operating voltage is denoted by V_{nomOp}, and the energy storage voltage, serving as the practical lower bound, is denoted by V_{ESS}. With the DC/DC converter hardware topology described in Alho et al. (2024), V_{ESS} represents a critical threshold to which the voltage must never fall. If the DC-link voltage were to drop to this level, the converter's IGBTs would effectively operate as a diode bridge, causing the loss of control over the energy storage and leading to uncontrolled discharging.

In this example, the standby generator receives the start command when the DC voltage first falls below V_{A}. Although the generator may not connect quickly enough to support the system, the command is issued immediately.

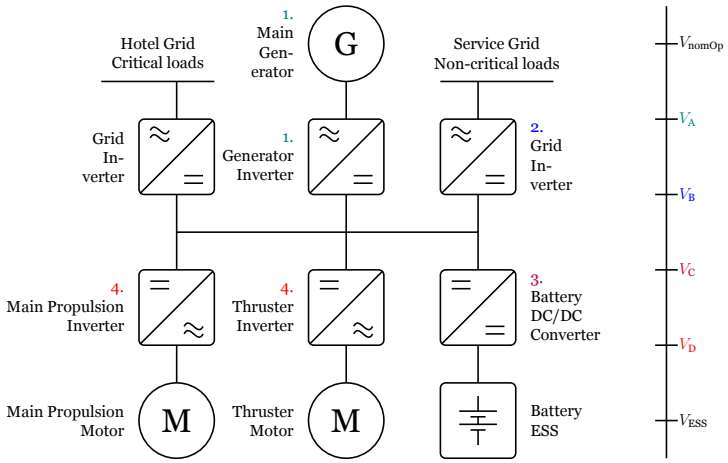


Figure 94. Single-line diagram with voltage scale used for blackout-prevention priority order example.

If the DC voltage continues to fall, the non-critical loads disconnect themselves according to their predefined priority logic. Should further support be required, batteries or other suitable sources provide current via undervoltage regulators. If the voltage still declines, for example because the battery DC/DC converter reaches its discharge current limit, load reduction is extended to the propulsion and thruster inverters. While the goal is to maintain propulsion, it is preferable to partially reduce propulsion power until the standby generator comes online than to risk a total blackout.

This example illustrates only one possible implementation of equipment priority ordering. In practice, numerous other ship-specific design factors and end-user requirements may apply, which fall outside the scope of this work. A real-life blackout prevention system is described in Section 5.6.2.

5.6.2 Aurora Botnia: Real-life Use Case on Variable Voltage-based Blackout Prevention

Aurora Botnia, see Figure 95, sailing between Vaasa, Finland, and Umeå, Sweden, features electric propulsion supported by battery energy storage systems within an AC/DC hybrid power distribution grid (Figure 96). The four main generators are located in the propulsion switchboards on the 690 V AC grid, while the batteries and the principal consumers—namely the main propulsion and thruster inverters—are located in separate DC microgrids.

Figure 95. *Aurora Botnia*, picture used with permission from Wasaline.

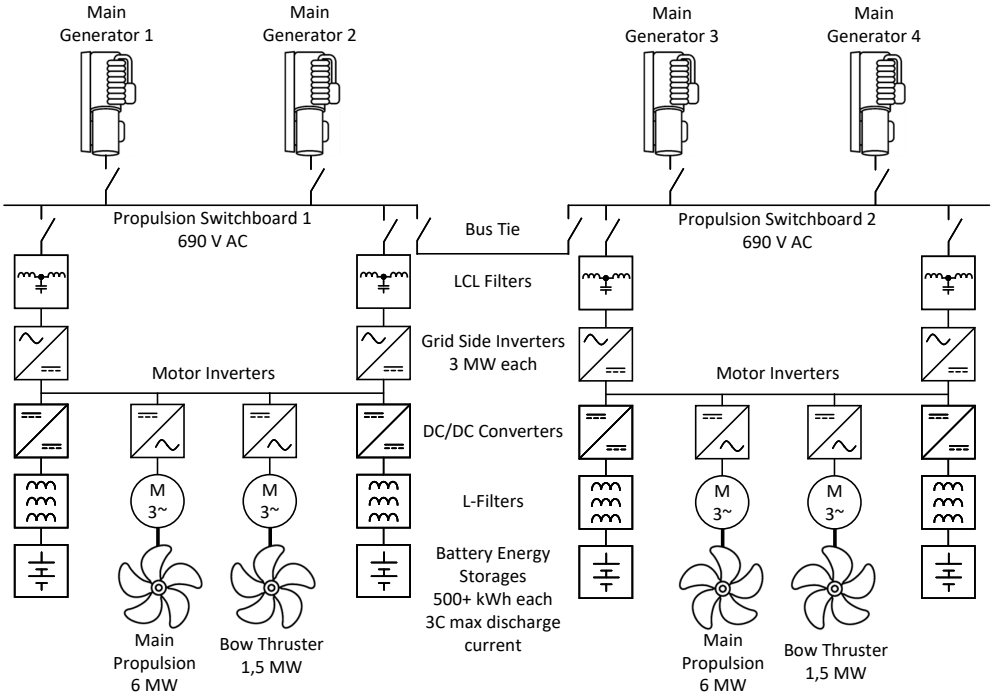


Figure 96. Simplified single-line diagram of *Aurora Botnia*'s AC/DC hybrid power distribution system, used with permission from Wasaline and WE Tech Solutions Oy.

The DC microgrids apply the variable voltage-based control strategy and are capable of performing ship-level blackout prevention using the battery systems.

The electric main propulsion motors are rated at 6.0 MW each, and the bow thrusters at 1.5 MW each. The four battery systems each have a capacity exceeding 500 kWh, with a 3C, or 300% of nominal current, discharge capability—equivalent to approximately 1.5 MW of power output per system, subject to state-of-charge.

In normal operation, the batteries are employed for emission-free maneuvering near shore and for hybrid operation with the main generators in peak-shaving and load-scheduling modes. They are also capable for blackout prevention during sea operation (e.g., in traditional maneuvering mode when power for the propulsors is provided only by the main generators), and in port in the event of a shore connection failure. The DC microgrid management principles follow those presented in this work, with differences in the practical implementation of the inverter under- and overvoltage regulators and the considerations specific to an AC/DC hybrid system.

Official blackout-prevention sea trials were conducted during the night of 20 – 21 January 2022. All tests were successful and no blackouts occurred. Figures 97 and 98 show the inverter system behavior (currents, limits, and regulator statuses) and the AC grid behavior during one such test.

The test was conducted with the bus tie closed (propulsion switchboards interconnected) and only a single main generator in operation. The vessel was operating in traditional maneuvering mode, where the sole generator served as the only active voltage source maintaining the 690 V AC grid. The grid-side inverters acted as voltage sources for the DC microgrids, while all four battery DC/DC converters were in standby mode—running in current control with zero reference, effectively monitoring the DC-link voltage and reacting only via their under- and overvoltage regulators.

Prior to the test, propulsion load was gradually increased, as shown in Figure 97 (port-side DC microgrid perspective). The ramp rate was deliberately slow, consistent with operation on a single generator; propulsion power ramps are adjusted automatically based on the number of active generators to avoid excessive transients (Alho et al. (2024)). At time point A, the only

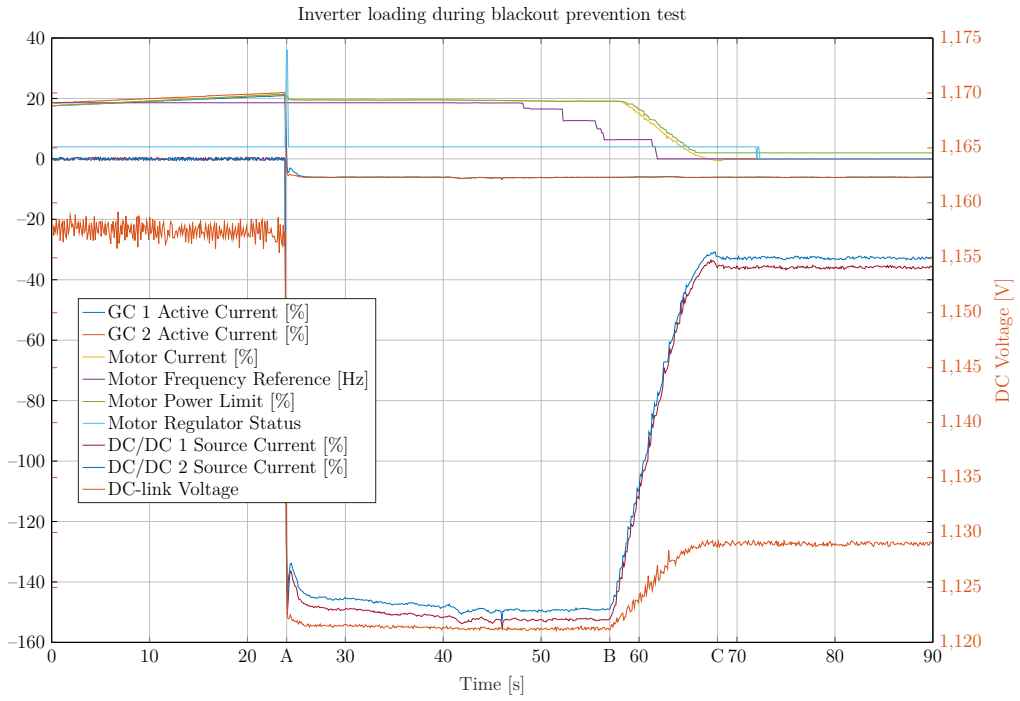


Figure 97. *Aurora Botnia*'s system behavior during a blackout-prevention test, where GC denotes Grid Converter, data used with permission from Wasaline and WE Tech Solutions Oy.

running main generator was emergency-stopped via the e-stop button in the control room, resulting in its instantaneous disconnection.

At the moment of disconnection, no power sources remained online and the DC-link voltage fell rapidly from approximately 1,158 V to 1,121 V. The DC/DC converters immediately responded: their undervoltage regulators activated and their output increased from zero to approximately 150% of rated discharge. Simultaneously, the grid-side inverters (labeled Grid Converters in this system) shifted from supplying power to the DC grid (about +20%) to absorbing power from it (about − 8%), reflecting reversal of current flow from DC to AC. Between time points B and C, propulsion load was manually reduced to zero by the ship's captain, which correspondingly lowered the DC/DC converter output and restored the DC-link voltage.

The ship's design principles also imposed an automatic propulsion power limitation at the moment of generator disconnection. Consequently, the propulsion inverters ran momentarily against their undervoltage regulators but quickly received new power limits from the control system. This prevented the batteries from being discharged at their maximum allowable cur-

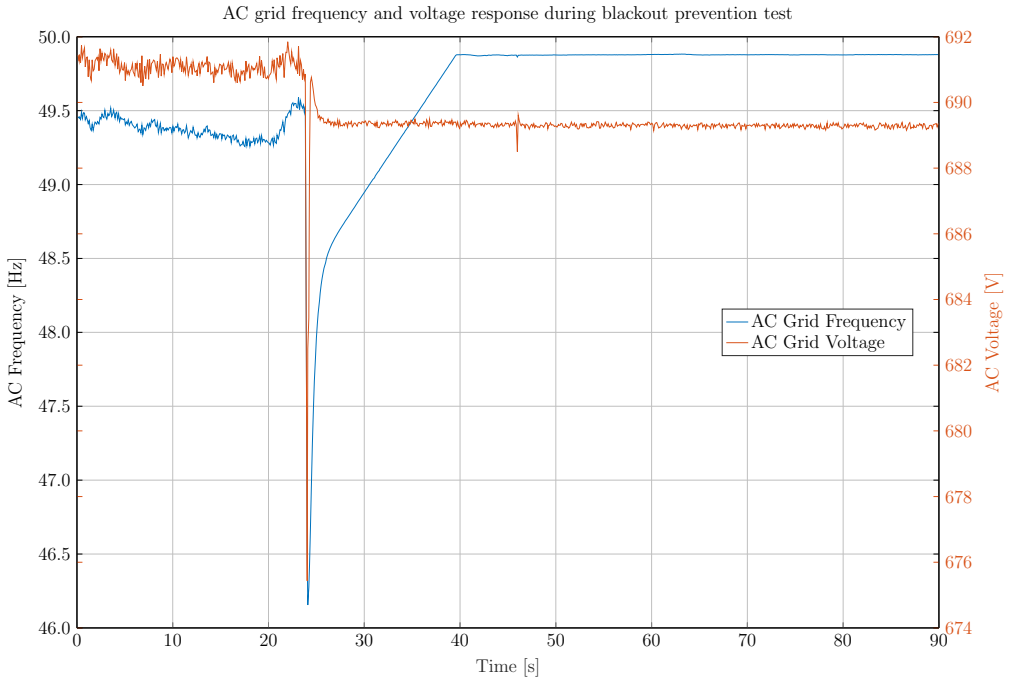


Figure 98. *Aurora Botnia*'s AC grid frequency and voltage during blackout-prevention test, data used with permission from Wasaline and WE Tech Solutions Oy.

rent, ensuring that they could continue supporting the vessel for longer if required.

Figure 98 shows the AC grid response. As the vessel operates with an AC/DC hybrid topology, where main generators are connected on the AC side, the test was performed from control's perspective in the worst case scenario where the grid-side inverters operated as DC voltage sources. In blackout prevention scenarios, they must be able to switch to AC voltage source operation in accordance with grid frequency. To avoid false triggering, the frequency threshold was set relatively low, allowing the frequency to dip significantly before operation mode switching, yet no blackout occurred. In a pure DC distribution system, with all sources located on the DC link, the grid-side inverters would consistently operate as AC voltage sources, and such frequency dips would not occur.

This real-life test demonstrates the effectiveness of the variable voltage-based DC microgrid control strategy in handling extremely fast, blackout-inducing exception situations. Without active communication or central commands, and relying solely on the DC-link voltage as the indicator of system state, the

DC/DC converters successfully prevented a blackout when the last generator tripped offline.

6 DISCUSSION

As noted at the outset of this dissertation, while DC microgrids are increasingly adopted in various fields, their control strategies remain largely tied to AC-oriented paradigms. This is evident from the literature review, which showed that DC voltage is often treated analogously to AC frequency: both are assumed to require strict, continuous stabilization. While stability is important, the DC voltage level in a DC microgrid has significantly more permissible variation compared to AC frequency. This flexibility enables the use of DC voltage as an information-carrying medium.

Although some researchers have implemented DC microgrids that allow the DC voltage to vary, their approaches have remained bound to traditional centralized control concepts. In the literature, so-called DC-bus signaling typically refers to a centralized controller adjusting system-level operation according to the DC-link voltage. In practice, this offers limited capability for handling exceptional situations or ensuring system flexibility. The DC voltage is merely used as a trigger to change operating modes via the central controller, similar to a manual system. Communication delays persist, all actions must be pre-programmed, and the controller itself constitutes a single point of failure.

This dissertation introduces a novel DC microgrid control strategy, referred to as the *variable voltage-based power management strategy*. The approach combines two key features: (i) the ability of inverters to deduce the loading state of the voltage-supporting equipment directly from the DC microgrid voltage level, and (ii) the capability of load inverters to support the DC-link voltage via under- and overvoltage regulators. This combination distributes fast decision-making, traditionally requiring system-level control, down to the inverters themselves—i.e., the lowest and most robust level of system architecture. The DC-link voltage thereby functions as a communication medium: droop-controlled voltage sources adjust their output proportionally to the deviation between the DC-link voltage and their reference, while loads use their under- and overvoltage regulators to provide support when thresholds are reached. Together, these mechanisms enable both planned and exceptional responses without external commands.

The variable voltage-based power management strategy is built upon this

information-carrying characteristic of the DC-link voltage. Because droop control makes the voltage vary linearly with the proportional output currents of voltage-supporting equipment, any given voltage corresponds directly to a proportional loading state—provided that all devices use identical voltage references and droop coefficients. Consequently, every device connected to the DC-link inherently knows the loading condition of the sources and can react independently, without central coordination, to support system stability.

This distributes decision-making to the lowest system architecture level, allowing system-level actions during exceptional events to be executed within milliseconds by the equipment itself. This approach avoids the need to preprogram all possible scenarios in advance: instead, equipment can deduce the appropriate response autonomously, based on its parameterization and the observed DC-link voltage.

The variable voltage-based power management strategy was validated by first establishing a mathematical foundation and a modeling paradigm. This platform enabled the development of the required under- and overvoltage regulators for load inverters, which are core elements of the strategy. Next, real-life data were compared with the results of the modeling paradigm to verify the operation of the strategy in representative shipboard scenarios. The comparison confirmed both the accuracy of the modeling approach and the operational capabilities of the strategy in practice.

6.1 Modeling Paradigm

The modeling paradigm developed in this work encompasses the three main components of a DC microgrid: voltage sources, loads, and the DC-link, see Figure 99. Beyond their base operation, the paradigm also incorporates circulating currents and advanced features such as energy storage balancing and under- and overvoltage regulators, whose behavior is directly tied to the proposed control strategy.

A central principle of the paradigm is the use of proportional values for all currents. This allows droop parameters to be expressed proportionally as

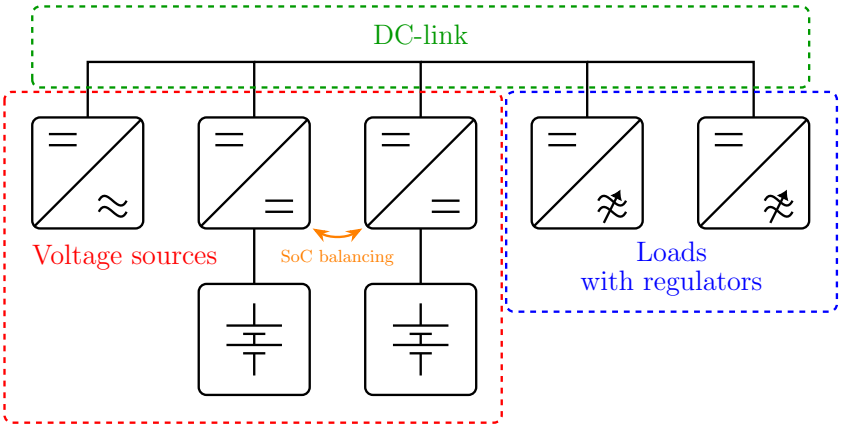


Figure 99. Principle single-line diagram of an inverter system, depicting the main modeled components.

well. Instead of calculating individual parameters for each device supporting the DC-link voltage, it is possible to apply a uniform droop coefficient, for example, of 2.0% to all voltage sources.

The paradigm also extends to include cable and busbar resistances, enabling nodal analysis of the system. In practical installations, such resistances are non-negligible and introduce additional effective droop, particularly in large systems with long distribution distances. These resistances influence inverter undervoltage and overvoltage activation thresholds and may introduce nonlinearities when effective droop coefficients are no longer equal across equipment.

6.2 Energy Storage State-of-Charge Balancing

As energy storage systems gain increasing prominence in shipboard power systems, compatibility of the microgrid control strategy with their operation becomes essential. For the variable voltage-based power management strategy, the main risk of incompatibility lies in state-of-charge management methods that introduce excessive DC voltage deviations during normal operation, which could cause unwanted activation of under- or overvoltage regulators.

One aspect that can cause such deviations is state-of-charge balancing between multiple energy storages of different sizes and nominal currents, con-

nected via DC/DC converters of different ratings. If the converters are not controlled in a coordinated manner, balancing can be achieved but at the expense of significant DC-link voltage fluctuations.

This dissertation develops state-of-charge balancing algorithms based on offsetting either the DC voltage reference or the droop coefficient. Both methods are integrated into the DC microgrid modeling paradigm and are designed to balance state-of-charge across any number of heterogeneous storage systems. Particular attention is given to ensuring compatibility between these balancing mechanisms and the variable voltage-based strategy.

6.3 Real-life and Modeled Scenarios Incorporating the Variable Voltage-based Strategy

To verify feasibility, a series of scenarios were modeled that reflect real-world shipboard power system challenges. These scenarios include overload prevention, automatic voltage source startup, and non-critical load disconnection. In overload prevention, loads autonomously support voltage sources to prevent cascading failure of the voltage sources. In renewable generation scenarios, current-source inverters (e.g., solar) are limited by overvoltage regulators to avoid backpowering voltage sources while still maximizing renewable output.

In addition, real-life data from ships already applying different principles of the variable voltage-based strategy are presented. Data from *MV Tasmanian Achiever II* and *MV Victorian Reliance II* Alho and Laaksonen (2020) illustrate controlled power source prioritization, while additional data from *Aurora Botnia* demonstrate how the strategy enables blackout prevention. In the latter case, when the last voltage source onboard was suddenly disconnected, the distributed inverter-based response prevented blackout without requiring central commands.

7 CONCLUSIONS

7.1 Research Outcomes

Section 1.2 listed the dissertation's research questions, which cover the core aspects required to verify the validity of the variable voltage-based power management strategy for DC microgrids in shipboard applications. While the questions were designed to be general enough to support the evaluation of any control strategy, their emphasis reflects the unique features of the proposed approach. These include, in particular, the ability of load inverters to operate temporarily as voltage sources via their under- and overvoltage regulators, thereby actively supporting the DC-link voltage.

7.1.1 RQ1: Modeling Paradigm for the Variable Voltage-based Strategy

The principal challenge in modeling the variable voltage-based strategy lies in representing both the loads and the DC-link voltage accurately. Although numerous modeling tools exist, ranging from Simulink to PSCAD, they do not natively incorporate the under- and overvoltage regulator functionalities of load inverters. These regulators are central to the proposed strategy, as they enable loads to participate in maintaining the DC-link voltage. Without the ability to model these behaviors explicitly, the operation of the control strategy cannot be evaluated.

In addition, accurate modeling of the DC-link voltage itself is critical, as it dictates the overall system behavior. A fully detailed representation would require the inclusion of inverter capacitances and other time-variant dynamics. In this work, however, the paradigm applies a quasi-static approach, representing the DC-link through virtual droop resistances of the inverters and, where relevant, cable and busbar resistances. While the omission of capacitances means that fast transients are not captured, this limitation does not undermine the objectives of this study, which are concerned with the operational principles of the control strategy rather than detailed transient

phenomena.

Nevertheless, incorporating cable and busbar resistances, together with inverter capacitances, is an important direction for further academic and industrial research. In practical systems with long distribution distances, voltage drops directly affect equipment parameterization and performance. Similarly, modeling transient dynamics is essential for analyzing system selectivity and the design of protective devices such as fuses.

The mathematical foundations of the developed modeling paradigm are presented in Section 3.1. The source codes of the utilized MATLAB scripts are presented throughout the work.

7.1.2 RQ2: Inverter Under- and Overvoltage Regulators

The variable voltage-based power management strategy relies on allowing the DC-link voltage to deviate according to the droop characteristics of the voltage sources. These deviations communicate the loading state of the sources, enabling the loads to support them when required. Such support is provided independently through the loads' under- and overvoltage regulators, whose function is to temporarily change a load into a voltage-supporting device at predefined DC-link voltage levels.

For this principle to function correctly, the regulator algorithms must be designed to activate and deactivate at the appropriate voltage thresholds and to behave consistently and predictably. Section 3.1.7 presented the development of regulators with two essential features: built-in drooping functionality, which enables stable parallel operation with other voltage sources, and conditional droop compensation, which ensures activation precisely at the desired voltage level. The regulators' performance was demonstrated through modeling in a system where two loads provided voltage support to a source with a limited maximum output.

7.1.3 RQ3: Energy Storage State-of-Charge Balancing Compatible with the Control Strategy

Energy storage systems are expected to be a central element of future shipboard power systems, and their effective utilization requires reliable state-of-charge balancing. The purpose of state-of-charge balancing is to control the DC/DC converters linking the storages to the DC microgrid in such a way that their states-of-charge converge over time. For converters operating in current-control mode this task is relatively straightforward, but the situation becomes considerably more complex when multiple converters operate in parallel as voltage sources with different rated currents and storage capacities.

As noted in RQ2, the variable voltage-based power management strategy relies on loads assisting the voltage sources via under- and overvoltage regulators when the DC-link voltage deviates from its reference. These voltage deviations are the natural result of droop-controlled voltage source operation and are both expected and desired. However, they should only reflect genuine load changes. Voltage fluctuations caused by poorly designed state-of-charge balancing can inadvertently trigger the regulators of other devices, leading to unintended interventions. Therefore, state-of-charge balancing must be implemented in a manner that avoids introducing spurious deviations in the DC-link voltage.

Chapter 4 presented two state-of-charge balancing algorithms: a DC voltage reference offset method (Section 4.3) and a droop coefficient offset method (Section 4.4). Modeling results showed that both approaches achieve balancing, but with differing characteristics. The reference-offset method is simpler to implement and can operate without inducing any deviation in the DC-link voltage. In contrast, the droop-offset method is more cumbersome: it depends on external load changes to induce current flow, which it then manipulates via droop adjustments, making it less effective in preventing voltage deviations.

7.1.4 RQ4: Modeling of Shipboard Power System Scenarios

To validate the usability and performance of the variable voltage-based power management strategy in shipboard applications, it was necessary to test the approach against a range of scenarios representative of real operating conditions. These included both everyday and exceptional situations, in order to demonstrate the breadth of the strategy's capabilities.

As discussed in RQ1 and RQ2, the proposed approach is a distributed control strategy: the loads and sources operate independently, while the microgrid controller provides only the operational boundaries, references, and limits. Fast decision-making is executed locally by the inverters themselves.

Chapter 5 addressed the following scenarios:

1. Section 5.2.1: Loads' automatic support of voltage sources against overload
2. Section 5.2.2: Automatic startup of a standby voltage source
3. Section 5.3: Two implementations for current sources (e.g., renewables) supporting voltage sources against backpower
4. Section 5.4: Voltage source prioritization, first introduced by Alho and Laaksonen (2020)
5. Section 5.5: Disconnection of non-critical loads
6. Section 5.6: Blackout prevention

The control strategy performed as intended in all of the above scenarios. Its strengths were most evident in the exceptional cases (1, 3, and 6), where failure to act would otherwise result in blackout. In scenario 1, the loads' undervoltage regulators protected the voltage source against overload; in scenario 3, the current source's overvoltage regulator prevented the voltage source from backpowering; and in scenario 6, the battery-connected DC/DC converters sustained system operation when the last running generator tripped, thereby avoiding blackout.

7.1.5 RQ5: Modeling Comparison Against Real-life Data

To verify that the mathematical model reflects real system behavior, a direct comparison was carried out between modeled results and real-life measurements. Scenario 4 in RQ4 provided such an opportunity: in a real ship, a shaft-generator system was operated in a way that enabled voltage source prioritization. Since measurement data from this case were available, the same situation was modeled using only the bow thruster load profile taken from the vessel, with all other system components represented through the mathematical model.

Section 5.4.3 presented the comparison between real-life data and the modeled results. By basing the model solely on the recorded load profile, the comparison ensured that similarities or deviations directly reflected the accuracy and limitations of the modeling paradigm. The analysis demonstrated that the model captured the system's operational principles with sufficient fidelity to support both validation of the proposed control strategy and its use in predicting system behavior under similar scenarios.

7.1.6 RQ6: Control Strategy Operation in Real-life Systems

This dissertation also examined two real-life vessels that illustrate partial application of the variable voltage-based power management strategy. Section 5.4.1 presented power supply prioritization, first introduced in Alho and Laaksonen (2020), while Section 5.6.2 detailed a blackout prevention case study onboard *Aurora Botnia*. Both cases were supported with measured operational data.

The power supply prioritization example builds on previously published research, whereas the blackout prevention case study provides new, previously unpublished evidence of the strategy's applicability. Together, these cases demonstrate that the principles validated through mathematical modeling are not only theoretically sound but also practically effective in demanding shipboard environments.

7.2 Scientific Contributions

This dissertation presented a novel DC microgrid control strategy, based on utilizing the DC-link voltage as an information-carrying medium and on the capability of inverters to act independently according to that voltage through their regulators. By combining the droop characteristics of voltage sources with the under- and overvoltage regulator functionalities of loads, it is possible to build a distributedly controlled DC microgrid that is both robust and fault-tolerant, with reaction times in the millisecond range.

The feasibility of the strategy was demonstrated through the development of a new quasi-static mathematical modeling paradigm for DC microgrids, capable of explicitly representing the operation of load inverters with under- and overvoltage regulators. This paradigm was numerically validated and then used to model a range of scenarios, from everyday operating conditions to blackout-inducing exceptions, thereby showing how the strategy handles both normal and extreme situations. Real-life data from two ships were also incorporated: one demonstrating power supply prioritization and another blackout prevention. In the prioritization case, the scenario was additionally modeled mathematically, allowing direct comparison of real-life and modeled behaviors, which confirmed that the paradigm accurately captures real-world operation.

A critical requirement of the strategy is that all DC-link voltage deviations should originate from actual changes in system loading, not from unrelated processes. One such process is energy storage state-of-charge balancing. Since state-of-charge balancing does not inherently reflect the ship's overall loading condition, it must not introduce spurious voltage deviations that could trigger incorrect responses from other devices. To address this, two balancing algorithms were developed that can handle multiple storage units and DC/DC converters of arbitrary size, while keeping the DC-link voltage—and thus the strategy's operation—undisturbed.

Thanks to its distributed nature, the strategy enables shipboard power systems to remain resilient even in severe conditions. Because loads can independently support the voltage sources via their undervoltage regulators, the system inherently prevents source overloads and avoids blackout. As demonstrated in the case of *Aurora Botnia*, properly designed systems ap-

plying the variable voltage-based strategy are capable of blackout prevention even when the last remaining generator trips.

With the variable voltage-based DC microgrid control strategy, shipboard power management reduces fundamentally to DC voltage management. Allowing droop-induced deviations in the DC voltage provides a direct measure of current imbalance between production and consumption. A stable DC-link voltage indicates balanced power and the absolute voltage level reflects the loading state of the voltage-maintaining equipment.

Traditional shipboard power management depends on continuously calculating the so-called "power available" to determine remaining generation capacity. In contrast, the proposed strategy reduces this dependency by enabling equipment to manage power distribution autonomously, since all necessary information is already embedded in the DC-link voltage. With properly designed and parameterized regulators, even loads—which conventionally remain indifferent to source loading—can contribute to voltage stability. This shifts the role of ship automation and power management systems away from low-level, instantaneous power control, and toward higher-level operational and supervisory tasks.

7.3 Next Possible Steps and Future Research

Clear next steps and development opportunities have been identified during this work. These fall into two broad categories: the control strategy itself and the modeling paradigm. Both are relevant from academic and industrial perspectives: academically, as avenues for developing algorithms and methods for the wider research community; and industrially, as opportunities to create solutions that provide competitive advantages in the marine sector.

7.3.1 The Control Strategy

As some of the principles of the variable voltage-based power management strategy have already been demonstrated in real-life shipboard systems, the

next logical step is a full-scale implementation in the longer term. For shipboard power systems, this would require the main power distribution to be entirely DC, with all producers, major consumers, and energy storage units connected directly to the DC microgrid. The hotel and service-related AC grids would remain as separate AC microgrids, appearing as loads from the DC perspective. From a development point of view, this transition would require a virtual environment capable of accurately modeling the ship power system, together with model-based design tools that enable flexible development of multiple controllers operating in tandem. Furthermore, implementing multiple safeguards against blackouts introduces challenges related to fault tolerance in power flow and control, as well as efficient utilization of energy storage.

If the physical distances between equipment are short enough that cable and busbar voltage drops can be neglected, the simplified single-DC-voltage model of the paradigm would suffice for operational studies. Where distances are longer and resistances non-negligible, the nodal analysis introduced in this work becomes essential. For more realistic preparation of ship operation, additional load profiles should be incorporated into the modeling to study concurrent use cases—for instance, the simultaneous operation of bow thrusters and main propulsion during port maneuvers.

Expanding the set of real-life scenarios to be modeled will help verify the robustness of the strategy and support the development of inverter functionalities and microgrid controllers optimized for these use cases. The strategy is not only applicable to the marine sector but also to other domains, including space systems and isolated land-based microgrids. These applications similarly demand systems that are robust, reliable, and highly autonomous.

7.3.2 The Modeling Paradigm

Several avenues for further development of the modeling paradigm were identified. A major limitation of the present implementation is its quasi-static nature, which omits time-dependent dynamics. The inclusion of inverter capacitances enables time-variant modeling of transients, allowing the paradigm to address phenomena such as instabilities induced by con-

stant power loads.

The nodal analysis approach implemented here extends the model to include cable and busbar resistances. These resistances cause additional voltage drops that, from the DC-link perspective, appear as added droop and alter the effective sharing between sources and loads. The situation becomes more complex when inverter voltage regulators are considered, since the voltage drops observed by the loads are no longer proportional solely to the combined source loading, but also depend on which particular load is drawing current. In the presence of constant-power loads, such interactions can introduce nonlinear behavior into the system response.

A further extension of the paradigm—beyond the present quasi-static scope—would be to incorporate dynamic elements such as capacitances and inductances, enabling the modeling of transients. This capability is essential for analyzing protection behavior, particularly fuse coordination and selectivity, which remain unresolved challenges in DC microgrid design.

The following development directions have been identified for the modeling paradigm:

- **DC-link voltage modeling:** Extend the current quasi-static approach by including inverter capacitances to capture dynamic, load-induced phenomena.
  - The most obvious shortcoming of the current modeling paradigm is its quasi-static nature, which limits its usage to relatively long-term phenomena. While this is sufficient for power management – related studies, it is not sufficient for sub-second time-level modeling.
- **Fast transient and fuse modeling:** Integrate short-term transient dynamics and fuse behavior to enable selectivity and protection system modeling.
  - One of the largest challenges for DC distribution grids is selectivity, i.e., how to select and parameterize protection devices so that they open the circuit only when required to isolate faults downstream, but allow even high short-circuit currents to pass when a fault is being isolated elsewhere in the grid. Isolation refers to

short-circuit currents from multiple sources converging on the fault and tripping the protection device (e.g., fuse) in front of it.

- **Load and source diversity:** Incorporate more accurate models and operating profiles for a broader range of devices, including batteries, fuel cells, and motor drives.
  - For example, modeling how a combustion engine – based generator set could be overloaded by a large load step, creating more dynamics in the model, or how an energy storage system's current limits behave in terms of its internal variables and state of charge.
- **Energy storage charging and discharging:** The quasi-static modeling is incapable of accurately modeling energy storage charging over time. Adding time dynamics will help to address this aspect.
  - The quasi-static modeling is completely incapable of predicting the accumulation or depletion of energy in an energy storage system over specific time period. In order to model systems including energy storage, this basic capability is essential.
- **Directly connected sources:** Model the operation of directly DC-connected devices (e.g., batteries, fuel cells) in a DC microgrid.
  - Connecting batteries, fuel cells, or other energy storage directly to the DC-link is not uncommon, but changes the system control dynamics considerably. In order to model these systems, the internal electrical aspects of the energy storage must be represented, as they directly affect DC voltage behavior.
- **State-of-charge balancing optimization:** The current implementation cannot handle current-limited or stopped DC/DC converters during balancing and requires further development.
  - An obvious shortcoming of the current implementation of state-of-charge balancing is the missing handling of current-limited or stopped DC/DC converters. Hitting current limits is not uncommon, so this issue must be addressed.
- **Dynamic topologies:** Apply nodal analysis to large-scale dynamic distribution grids capable of changing their topology.

  - Especially in the marine sector, electrical grid topologies are often changed via breakers—for example, to electrically separate the port and starboard sides of the ship in preparation for faults. Ring topologies are also becoming more common, as they provide alternate current paths. Handling such topology changes is required for the modeling paradigm.

- **Optimization and AI:** Investigate algorithms, potentially AI-based, to optimize the operation of sources and energy storage controlling DC/DC converters in large-scale, non-ideal DC microgrids with dynamic topologies.

  - Even with resistance-based models only, changing grid topologies affect the current paths and thus the resistances between sources, energy storages, and loads. The operation of the equipment must be dynamically adjusted according to the grid topology in order to maintain desired capabilities, such as blackout prevention.

- **Iterative modeling steps:** To accurately model the phenomena happening within a single step involving different types of equipment, an iterative approach should be implemented to handle possible interdependent mode transitions.

  - During the modeling of load exceptional situations, it was identified that inverter current limits and voltage regulators may cause complex interdependent control dynamics within a single modeling step. As the calculations inside a step are currently done only once, such interdependencies are only addressed in subsequent steps, by which time the situation may already have developed away from the actual physical behavior. As this quasi-static paradigm is intended as a stepping stone toward a time-variant modeling paradigm, these control dynamics must be handled already at this stage, otherwise they will lead to inaccuracies and unwanted behaviors later.

REFERENCES

Abdolmaleki, Babak and Bergna-Diaz, Gilbert. (2022). Distributed Control and Optimization of DC Microgrids: A Port-Hamiltonian Approach. *IEEE Access*, *10*, 64222-64233.

Abhishek, A., Ranjan, A., Devassy, S., Kumar Verma, B., Ram, S. K., & Dhakar, A. K. (2020). Review of hierarchical control strategies for DC microgrid. *IET Renewable Power Generation*, *14*(10), 1631–1640.

Alho, T. M. R., & Laaksonen, H. (2020). DC Voltage Level – Based Power Supply Prioritization in Multi – Inverter System: Marine Use Case. In *2020 International Conference on Smart Energy Systems and Technologies (SEST)* (p. 1-6).

Alho, T. M. R., Shahparasti, M., & Laaksonen, H. (2024). Shipboard power systems. In M. G. Simões & T. D. C. Busarello (Eds.), *Power Electronic Converters and Systems, 2nd Edition, Volume 2: Applications* (chap. 21). IET, The Institution of Engineering and Technology.

Bidram, A., Reno, M. J., Abadi, S. A. G. K., Aparicio, M. J., & Bauer, D. (2024). Trends in dc Microgrids: From the control and protection perspective. *IEEE Electrification Magazine*, *12*(2), 33–47.

Bryan, J., Duke, R., & Round, S. (2004). Decentralized generator scheduling in a nanogrid using DC bus signaling. In *IEEE Power Engineering Society General Meeting, 2004.* (p. 977-982 Vol.1).

Bureau Veritas. (2025, 1). *Rules for the Classification of Steel Ships - NR467, Pt C, Ch 2, Sec 3* (Nos. NR467, Pt C, Ch 2, Sec 3).

Chen, D., Xu, L., & Yao, L. (2013). DC Voltage Variation Based Autonomous Control of DC Microgrids. *IEEE Transactions on Power Delivery*, *28*(2), 637-648.

Clerici, A., Paris, L., & Danfors, P. (1991). HVDC conversion of HVAC lines to provide substantial power upgrading. *IEEE Transactions on Power Delivery*, *6*(1), 324-333.

Dhir, R., Garg, R., & Mahajan, P. (2022). A Review of Microgrid Control Strategies. In *2022 IEEE Delhi Section Conference (DELCON)* (p. 1-5).

DNV AS. (2024, 7). *Rules for classification: Ships — Part 6 Chapter 2* (No. DNV-RU-SHIP Pt.6 Ch.2).

Edrington, C. S., Ozkan, G., Papari, B., & Perkins, D. (2022). Distributed adaptive power management for medium voltage ship power systems. *Journal of Marine Engineering & Technology*, *21*(5), 281–296. https://doi.org/10.1080/20464177.2021.1894783

Garg, A., Joshi, B. M., & Oruganti, R. (2016). Overview of real-time power management strategies in DC microgrids with emphasis on distributed control. In *2016 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)* (pp. 1–6).

Ghanbari, N., & Bhattacharya, S. (2018). SoC Balancing of Different Energy Storage Systems in DC Microgrids Using Modified Droop Control. In *IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society* (p. 6094-6099).

Ghanbari, N., Bhattacharya, S., & Mobarrez, M. (2018). Modeling and Stability Analysis of a DC Microgrid Employing Distributed Control Algorithm. In *2018 9th IEEE International Symposium on Power Electronics for Distributed Generation Systems (PEDG)* (p. 1-7).

Guerrero, J. M., Vasquez, J. C., Matas, J., de Vicuna, L. G., & Castilla, M. (2011). Hierarchical Control of Droop-Controlled AC and DC Microgrids—A General Approach Toward Standardization. *IEEE Transactions on Industrial Electronics*, *58*(1), 158-172.

Hu, R., & Weaver, W. W. (2016). Dc microgrid droop control based on battery state of charge balancing. In *2016 IEEE Power and Energy Conference at Illinois (PECI)* (p. 1-8).

Ito, Y., Zhongqing, Y., & Akagi, H. (2004). DC microgrid based distribution power generation system. In *The 4th International Power Electronics and Motion Control Conference, 2004. IPEMC 2004.* (Vol. 3, p. 1740-1745 Vol.3).

Jin, Z., Savaghebi, M., Vasquez, J. C., Meng, L., & Guerrero, J. M. (2016). Maritime DC microgrids - a combination of microgrid technologies and maritime onboard power system for future ships. In *2016 IEEE 8th International Power Electronics and Motion Control Conference (IPEMC-ECCE Asia)* (p. 179-184).

Justo, J. J., Mwasilu, F., Lee, J., & Jung, J.-W. (2013). AC-microgrids versus DC-microgrids with distributed energy resources: A review. *Renewable and Sustainable Energy Reviews*, *24*, 387-405. https://-doi.org/10.1016/j.rser.2013.03.067

Kant, P., Singhal, P., Mahto, M. K., & Jain, D. (2022). Control strategies for DC Microgrids: An overview. In *2022 2nd International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)* (p. 1-6).

Khushoo, M., Sharma, A., & Kaur, G. (2022). DC microgrid-A short review on control strategies. *Materials Today: Proceedings*, *71*, 362-369. (World Engineering Conference on Contemporary Technologies TM 2022) https://doi.org/10.1016/j.matpr.2022.09.409

Kim, K., Park, K., Roh, G., & Chun, K. (2018). DC-grid system for ships: a study of benefits and technical considerations. *Journal of International Maritime Safety, Environmental Affairs, and Shipping*, *2*(1), 1–12. https://doi.org/10.1080/25725084.2018.1490239

Ko, B.-S., Lee, G.-Y., Choi, K.-Y., & Kim, R.-Y. (2019). A Coordinated Droop Control Method Using a Virtual Voltage Axis for Power Management and Voltage Restoration of DC Microgrids. *IEEE Transactions on Industrial Electronics*, *66*(11), 9076-9085.

Lasseter, R., & Paigi, P. (2004). Microgrid: a conceptual solution. In *2004 IEEE 35th Annual Power Electronics Specialists Conference (IEEE Cat. No.04CH37551)* (Vol. 6, p. 4285-4290 Vol.6).

Li, C., Dragicevic, T., Diaz, N. L., Vasquez, J. C., & Guerrero, J. M. (2014). Voltage scheduling droop control for State-of-Charge balance of distributed energy storage in DC microgrids. In *2014 IEEE International Energy Conference (ENERGYCON)* (p. 1310-1314).

Lu, X., Sun, K., Guerrero, J. M., Vasquez, J. C., Huang, L., & Teodorescu, R. (2012). SoC-based droop method for distributed energy storage in DC microgrid applications. In *2012 IEEE International Symposium on Industrial Electronics* (p. 1640-1645).

Manandhar, U., Ukil, A., & Kiat Jonathan, T. K. (2015). Efficiency comparison of DC and AC microgrid. In *2015 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA)* (p. 1-6).

Ni, K., Hu, Y., & Li, X. (2017). An Overview of Design, Control, Power Management, System Stability and Reliability in Electric Ships. *Power Electronics and Drives*, *2*(2), 5–29. https://doi.org/10.5277/ped170211

OpenAI. (2025a). *ChatGPT GPT-5 (August 24 Version)*. https://chatgpt.com/share/68ab1414-09d8-8001-a76c-2f0cabd693dd. (DC microgrid control strategy comparison table)

OpenAI. (2025b). *ChatGPT GPT-5 (August 24 Version)*. https://chatgpt.com/share/68ab2470-78b8-8001-96cd-f99a3e502a76. (State-of-charge balancing algorithm comparison table)

P., Monica and M., Kowsalya and Guerrero, Josep M. (2021). Logarithmic droop-based decentralized control of parallel converters for accurate current sharing in islanded DC microgrid applications. *IET Renewable Power Generation*, *15*(6), 1240-1254. https://doi.org/10.1049/rpg2.12103

Papadimitriou, C. N., Kleftakis, V. A., Rigas, A., & Hatziargyriou, N. D. (2014). A DC-microgrid control strategy using DC- bus signaling. In *MedPower 2014* (p. 1-8).

Perlin, Ken. (2002). Improving noise. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques* (p. 681 – 682). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/566570.566636

Rashad, M., Ashraf, M., Bhatti, A. I., & Minhas, D. M. (2018). Mathematical modeling and stability analysis of DC microgrid using SM hysteresis controller. *International Journal of Electrical Power & Energy Systems*, *95*, 507-522. https://doi.org/10.1016/j.ijepes.2017.09.001

Saim, A., Houari, A., Barrios, M. A., Ait-Ahmed, M., Machmoum, M., & Guerrero, J. M. (2022). Power management strategy with SoCs balancing of a battery powered shipboard DC Microgrid. In *2022 IEEE International Conference on Electrical Sciences and Technologies in Maghreb (CISTEM)* (Vol. 4, p. 1-6).

Shekhar, A., Ramírez-Elizondo, L., & Bauer, P. (2017). DC microgrid islands on ships. In *2017 IEEE Second International Conference on DC Microgrids (ICDCM)* (p. 111-118).

Wang, F., Zhang, Z., Ericsen, T., Raju, R., Burgos, R., & Boroyevich, D. (2015). Advances in Power Conversion and Drives for Shipboard Systems.

Proceedings of the IEEE, *103*(12), 2285-2311.

Wang, Y., Wei, F., & Zuo, Z. (2021). Modeling and Stability Analysis of DC Microgrid with Constant Power Loads. In *2021 40th Chinese Control Conference (CCC)* (p. 6799-6805).

Xu, L., Wei, B., Yu, Y., Guerrero, J. M., & Vasquez, J. (2021). Coordinated Control of Diesel Generators and Batteries in DC Hybrid Electric Shipboard Power System. *Energies*, *14*(19). https://doi.org/10.3390/en14196246

Yu, X., Huang, A., Burgos, R., Li, J., & Du, Y. (2013). A fully autonomous power management strategy for DC microgrid bus voltages. In *2013 Twenty-Eighth Annual IEEE Applied Power Electronics Conference and Exposition (APEC)* (p. 2876-2881).

Zhang, L., Wu, T., Xing, Y., Sun, K., & Guerrero, J. M. (2011). Power control of DC microgrid using DC bus signaling. In *2011 Twenty-Sixth Annual IEEE Applied Power Electronics Conference and Exposition (APEC)* (p. 1926-1932).

Zhang, L., Zhang, W., Zeng, F., & Yang, X. (2018, sep). A Review of Control Strategies In DC Microgrid. *Journal of Physics: Conference Series*, *1087*(4), 042035. https://doi.org/10.1088/1742-6596/1087/4/042035

Zohrabi, N., Shi, J., & Abdelwahed, S. (2019). An overview of design specifications and requirements for the MVDC shipboard power system. *International Journal of Electrical Power & Energy Systems*, *104*, 680-693. https://doi.org/10.1016/j.ijepes.2018.07.050

APPENDICES

Appendix 1

Deriving inverter current with drooping coefficient

If we assume that a voltage source is perfectly capable of controlling the DC-link voltage, V_{DC}, to its internal voltage reference, V_{DS}^{*}, we can rewrite equation 1.2 as

$$\begin{aligned} V_{\mathrm{DC}} &= V_{\mathrm{DS}}^{*} \\ &= V_{\mathrm{S}}^{*}\left(1 - \frac{I_{\mathrm{S}}}{I_{\mathrm{nomS}}}\zeta_{\mathrm{S}}\right). \end{aligned}$$

And then solve for I_{S}:

$$\begin{aligned} V_{\mathrm{DC}} &= V_{\mathrm{S}}^{*}\left(1 - \frac{I_{\mathrm{S}}}{I_{\mathrm{nomS}}}\zeta_{\mathrm{S}}\right) \\ V_{\mathrm{DC}} &= V_{\mathrm{S}}^{*} - \frac{I_{\mathrm{S}}\zeta_{\mathrm{S}}V^{*}}{I_{\mathrm{nomS}}} \\ I_{\mathrm{nomS}}V_{\mathrm{DC}} &= I_{\mathrm{nomS}}V_{\mathrm{S}}^{*} - I\zeta_{\mathrm{S}}V_{\mathrm{S}}^{*} \\ I_{\mathrm{S}}\zeta_{\mathrm{S}}V_{\mathrm{S}}^{*} &= I_{\mathrm{nomS}}V_{\mathrm{S}}^{*} - I_{\mathrm{nomS}}V_{\mathrm{DC}} \\ I_{\mathrm{S}} &= \frac{I_{\mathrm{nomS}}V_{\mathrm{S}}^{*} - I_{\mathrm{nomS}}V_{\mathrm{DC}}}{\zeta_{\mathrm{S}}V_{\mathrm{S}}^{*}} \\ I_{\mathrm{S}} &= \frac{V_{\mathrm{S}}^{*} - V_{\mathrm{DC}}}{\zeta_{\mathrm{S}}V_{\mathrm{S}}^{*}}I_{\mathrm{nomS}} \end{aligned}$$

Appendix 2

MATLAB source code for modeling source circulating currents with output limitation and possibility to stop

```
%% Initialize variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [10; 20; 30; 40];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
SourceMeasErrors = [1; 0.5; -1; 0]; % Define equipment measurement errors
if ~isequal(length(InomS), length(DCRefs), length(DCDroops), length(SourceMeasErrors))
    error('C3_ModelingSourceCircCurrBoolArrays:SourceParameterMismatch', ...
        'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
GammaCMatrix = zeros(numberOfSources,numberOfSources);
SourceCurrentMatrices = zeros(numberOfSources,numberOfSources,length(steps));
SourceTotalCurrentArrays = zeros(numberOfSources,length(steps));
SourceLoadSupplyingCurrentArrays = zeros(numberOfSources,length(steps));
B_VS = ones(numberOfSources, 1);
B_runS = ones(numberOfSources, 1);
posLims = ones(numberOfSources, 1) * inf;
negLims = ones(numberOfSources, 1) * -inf;

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources, 1);
droopConduc = zeros(numberOfSources, 1);
R_MG = 0;

%% Create smootherstep-based load profile varying between -100 and 0 A
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
s = 6*x.^5 - 15*x.^4 + 10*x.^3;
load = -100 * s;

%% Run through the calculations
for n=steps
    %% DC voltage handling
    % Calculate R_MG
    for r=1:numberOfSources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = (B_VS(r) && B_runS(r))/droopResis(r);
    end
    R_MG = 1/sum(droopConduc);

    % Calculate the DC voltage
    if n==1 % If first step use default DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, load(n), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), sum(SourceLoadSupplyingCurrentArrays(:,n-1)), load(n), R_MG);
    end

    %% Source handling
    B_runS(2) = n < 2500; %<- Stop source 2 at modeling index 2500
    posLims(3) = 0.6; %<- Limit source 3 positive output to 60%

    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        GammaMatrix(j,j) = Gamma(DCVolt(n), DCRefs(j), DCDroops(j));
    end

    % Populate Gamma_C matrix
    for a = 1:numberOfSources
        for b = 1:numberOfSources
            if a==b
                % Diagonal
                GammaCMatrix(a,a) = Gamma_Ca(DCVolt(n), SourceMeasErrors(a), DCRefs(a), DCDroops(a));
            else
                % Off-diagonal (circulating current from source a to b)
                GammaCMatrix(a,b) = Gamma_Cab(DCVolt(n), SourceMeasErrors(a), droopResis(a), ...
                    InomS(b), SourceMeasErrors(b), droopResis(b));
            end
        end
    end

    % Calculate the source current matrix
```

```
    SourceCurrentMatrices(:,:,n) = (GammaMatrix + GammaCMatrix) * diag(InomS);

    % Check for limited and shutdown sources
    for k = 1:numberOfSources
        I_k = sum(SourceCurrentMatrices(k,:,n));
        B_VS(k) = I_k < posLims(k)*InomS(k) && I_k > negLims(k)*InomS(k);

        % If source is limited or stopped, zero its corresponding column
        % and line
        if ~(B_VS(k) && B_runS(k))
            SourceCurrentMatrices(k,:,n) = 0;
            SourceCurrentMatrices(:,k,n) = 0;
        end

        % If source is limited but running, set its corresponding diagonal
        % value to the limited one
        if ~B_VS(k) && B_runS(k)
            SourceCurrentMatrices(k,k,n) = min(max(I_k, negLims(k)*InomS(k)), posLims(k)*InomS(k));
        end
    end

    % Calculate source total currents
    SourceTotalCurrentArrays(:,n) = SourceCurrentMatrices(:,:,n)*ones(numberOfSources,1);

    % Calculate load supplying currents
    SourceLoadSupplyingCurrentArrays(:,n) = diag(SourceCurrentMatrices(:,:,n));
end
```

Appendix 3

MATLAB source code for inverter class

```
%% Author Info
% Made by Timo M. R. Alho
% email: timo.alho@uwasa.fi
%        timo.alho@alhoengineering.fi

classdef inverterAdvReg
    %% Private variables for object operation
    properties (Access = private)
        IL_nom = 0
        V_nom = 0
        VUVRef_L = 0
        ZetaUV_L = 0
        VOVRef_L = 0
        ZetaOV_L = 0
        Ref = 0
        IL_percent = 0
        IL_A = 0
        PL_percent = 0
        PL_W = 0
        V_DC = 0
        UVRegActive = 0
        OVRegActive = 0
        usedK = 0
        VUVRef_L_internal
        VOVRef_L_internal
        useAdvancedRegulator = 0
        usePowerReference = 0
        GammaUV_L
        GammaOV_L
        VUVRef_L_used
        VOVRef_L_used
        notLimitedK
        running = 1
        DCMeasError = 0;
        posLim = Inf
        negLim = -Inf
        posLimActive
        negLimActive
        circulatingCurrents = 0
        voltageRegulatorEnabled = 1
    end

    methods
        %% Constructor for creating the inverter object
        function obj = inverterAdvReg(nominalCurrent_A, ...
                                      uvRegulatorActivationLimit_V, ...
                                      uvRegulatorDroop_percent, ...
                                      ovRegulatorActivationLimit_V, ...
                                      ovRegulatorDroop_percent, ...
                                      useAdvancedRegulator, ...
                                      usePowerReference, ...
                                      V_nom)

            obj.IL_nom = nominalCurrent_A;

            if uvRegulatorActivationLimit_V == 0
                obj.VUVRef_L = 0;
                obj.ZetaUV_L = 2.0;
            else
                obj.VUVRef_L = uvRegulatorActivationLimit_V;
                obj.ZetaUV_L = uvRegulatorDroop_percent;
            end


            if ovRegulatorActivationLimit_V == 0
                obj.VOVRef_L = 0;
                obj.ZetaOV_L = 2.0;
            else
                obj.VOVRef_L = ovRegulatorActivationLimit_V;
                obj.ZetaOV_L = ovRegulatorDroop_percent;
            end

            obj.VUVRef_L_internal = obj.VUVRef_L;
            obj.VOVRef_L_internal = obj.VOVRef_L;
            obj.useAdvancedRegulator = useAdvancedRegulator;
            obj.usePowerReference = usePowerReference;

            if V_nom == 0
                obj.V_nom = 700;
            else
                obj.V_nom = V_nom;
            end
```

```
end

%% Setting Functions
function objWithNewRef = setRef(obj,Ref)
    obj.Ref = Ref;
    objWithNewRef = obj;
end

function objWithVoltage = setV_DC(obj, V_DC, DCMeasError)
    obj.DCMeasError = DCMeasError;
    obj.V_DC = V_DC + DCMeasError;
    objWithVoltage = obj;
end

function objWithLimits = setLimits(obj, posLim, negLim)
    obj.posLim = posLim;
    obj.negLim = negLim;
    objWithLimits = obj;
end

function objWithRunning = setRun(obj, runStatus)
    obj.running = runStatus;
    objWithRunning = obj;
end

function objWithCircCurr = setCirCurr(obj, circCurr)
    obj.circulatingCurrents = circCurr;
    objWithCircCurr = obj;
end

function objWithRegActive = setRegulatorActive(obj, voltageRegulatorEnabled)
    obj.voltageRegulatorEnabled = voltageRegulatorEnabled;
    objWithRegActive = obj;
end


%% Execute functions
function objWithCalculatedCurrents = calculateActualK(obj)
    if obj.usePowerReference
        KToUse = ((obj.Ref * obj.V_nom) / obj.V_DC);
    else
        KToUse = obj.Ref;
    end

    if obj.useAdvancedRegulator == 0
        VUVRef_L_internalFinal = obj.VUVRef_L;
    elseif obj.V_DC <= obj.VUVRef_L
        VUVRef_L_internalFinal = max(obj.VUVRef_L_internal, obj.VUVRef_L * (1 + (KToUse * obj.ZetaUV_L)));
    else
        obj.VUVRef_L_internal = ...
            obj.VUVRef_L * (1 + ((obj.IL_A/obj.IL_nom) * obj.ZetaUV_L));
        VUVRef_L_internalFinal = obj.VUVRef_L_internal;
    end


    if obj.useAdvancedRegulator == 0
        VOVRef_L_internalFinal = obj.VOVRef_L;
    elseif obj.V_DC >= obj.VOVRef_L
        VOVRef_L_internalFinal = min(obj.VOVRef_L_internal, obj.VOVRef_L * (1 + (KToUse * obj.ZetaOV_L)));
    else
        obj.VOVRef_L_internal = ...
            obj.VOVRef_L * (1 + ((obj.IL_A/obj.IL_nom) * obj.ZetaOV_L));
        VOVRef_L_internalFinal = obj.VOVRef_L_internal;
    end

    obj.GammaUV_L = ((VUVRef_L_internalFinal - obj.V_DC)/(obj.ZetaUV_L*obj.VUVRef_L));
    obj.GammaOV_L = ((VOVRef_L_internalFinal - obj.V_DC)/(obj.ZetaOV_L*obj.VOVRef_L));

    obj.VUVRef_L_used = VUVRef_L_internalFinal;
    obj.VOVRef_L_used = VOVRef_L_internalFinal;

    if obj.voltageRegulatorEnabled == 1
        finalKWithRegs = max(min(KToUse, obj.GammaOV_L), obj.GammaUV_L) * obj.running;

        if finalKWithRegs == obj.GammaUV_L && obj.running
            obj.UVRegActive = 1;
        else
            obj.UVRegActive = 0;
        end

        if finalKWithRegs == obj.GammaOV_L && obj.running
            obj.OVRegActive = 1;
        else
            obj.OVRegActive = 0;
        end
    else
        obj.UVRegActive = 0;
        obj.OVRegActive = 0;
        finalKWithRegs = KToUse * obj.running;
    end

    if finalKWithRegs + obj.circulatingCurrents/obj.IL_nom >= obj.posLim && obj.running
        obj.posLimActive = 1;
```

```
            obj.negLimActive = 0;
            obj.UVRegActive = 0;
            obj.OVRegActive = 0;

            finalK = obj.posLim;
        elseif finalKWithRegs + obj.circulatingCurrents/obj.IL_nom <= obj.negLim && obj.running
            obj.posLimActive = 0;
            obj.negLimActive = 1;
            obj.UVRegActive = 0;
            obj.OVRegActive = 0;

            finalK = obj.negLim;
        else
            obj.posLimActive = 0;
            obj.negLimActive = 0;
            finalK = finalKWithRegs;
        end

        obj.usedK = finalK;
        obj.notLimitedK = KToUse;

        obj.PL_percent = finalK * (obj.V_DC/obj.V_nom);
        obj.PL_W = obj.IL_nom * finalK * (obj.V_DC/obj.V_nom);
        obj.IL_percent = finalK;
        obj.IL_A = obj.IL_nom * finalK;

        objWithCalculatedCurrents = obj;
    end

    %% Getting functions
    function outputCurrentPercent = getIL_percent(obj)
        outputCurrentPercent = obj.IL_percent;
    end

    function outputCurrentA = getIL_A(obj)
        outputCurrentA = obj.IL_A;
    end

    function outputPowerPercent = getPL_percent(obj)
        outputPowerPercent = obj.PL_percent;
    end

    function outputPowerW = getPL_W(obj)
        outputPowerW = obj.PL_W;
    end

    function [uvRegulatoActive, ovRegulatoActive] = getRegulatorActivations(obj)
        uvRegulatoActive = obj.UVRegActive;
        ovRegulatoActive = obj.OVRegActive;
    end

    function usedK = getUsedK(obj)
        usedK = obj.usedK;
    end

    function notlimitedK = getnotLimitedK(obj)
        notlimitedK = obj.notLimitedK;
    end

    function [uvRegulatorGamma, ovRegulatorGamma] = getRegGammas(obj)
        % if obj.voltageRegulatorEnabled == 1
            uvRegulatorGamma = obj.GammaUV_L;
            ovRegulatorGamma = obj.GammaOV_L;
        % else
        %     uvRegulatorGamma = 0;
        %     ovRegulatorGamma = 0;
        % end
    end

    function [VUVRef, VOVRef] = getRegVRefs(obj)
        % if obj.voltageRegulatorEnabled == 1
            VUVRef = obj.VUVRef_L_used;
            VOVRef = obj.VOVRef_L_used;
        % else
        %     VUVRef = 0;
        %     VOVRef = 0;
        % end
    end

    function DCRef = getActiveDCRef(obj)
        if ~(obj.UVRegActive || obj.OVRegActive)
            DCRef = obj.VUVRef_L;
        else
            DCRef = obj.UVRegActive * obj.VUVRef_L_used + ...
                    obj.OVRegActive * obj.VOVRef_L_used;
        end
    end

    function DCDroop = getActiveDCDroop(obj)
        if ~(obj.UVRegActive || obj.OVRegActive)
            DCDroop = obj.ZetaUV_L;
        else
            DCDroop = obj.UVRegActive * obj.ZetaUV_L + ...
```

```
                        obj.OVRegActive * obj.ZetaOV_L;
            end
        end

        function DroopConduc = getDroopConduc(obj)
            DroopConduc = (obj.running && (obj.UVRegActive || obj.OVRegActive)) / ...
                        ((getActiveDCRef(obj) * getActiveDCDroop(obj)) / obj.IL_nom);
        end

        function [posLimActive, negLimActive] = getActiveLims(obj)
            posLimActive = obj.posLimActive;
            negLimActive = obj.negLimActive;
        end

        function running = getRun(obj)
            running = obj.running;
        end
    end
end
```

Appendix 4

MATLAB source code for modeling a system of two loads with their undervoltage regulators

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [100];            % Source nominal currents
DCRefs = [700];       % Source DC voltage references
DCDroops = [0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('E2_ModelingLoadVoltRegsOptReg:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
sourceCurrentArrays = zeros(numberOfSources,length(steps));
B_VS = ones(numberOfSources, 1);
B_runS = ones(numberOfSources, 1);
posLims = ones(numberOfSources, 1) * inf;
negLims = ones(numberOfSources, 1) * -inf;

%% Initialize load modeling variables
% Order of the loads is as follows: [L1, L2...]
InomL = [100; 100]; % Load nominal currents
% UVRefs = [DCRefs(1) * (1 - DCDroops(1)*0.8)];
UVRefs = [700*0.98; 684];
UVDroop = [0.02; 0.02];
% OVRefs = [710];
OVRefs = [710; 710];
OVDroop = [0.02; 0.02];
if ~isequal(length(InomL), length(UVRefs), length(UVDroop), length(OVRefs), length(OVDroop))
    error('E3_ModelingLoadVoltRegsOptRegBoolArrays_twoloads:SourceParameterMismatch', ...
          'The number of load parameters do not match each other.');
end
numberOfLoads = length(InomL);
KappaMatrix = zeros(numberOfLoads,numberOfLoads);
loadCurrentArrays = zeros(numberOfLoads,length(steps));
loads(1,1) = inverterAdvReg(InomL(1), ...
                            UVRefs(1), ...
                            UVDroop(1), ...
                            OVRefs(1), ...
                            OVDroop(1), ...
                            1, ... % 0 = normal drooping regulator
                            ...    % 1 = drooping regulator with conditional droop compensation from the dissertation
                            0, ... % 0 = reference is considered current reference
                            ...    % 1 = reference is considered power reference
                            700); % Nominal voltage for power calculations

loads(1,2) = inverterAdvReg(InomL(2), ...
                            UVRefs(2), ...
                            UVDroop(2), ...
                            OVRefs(2), ...
                            OVDroop(2), ...
                            1, ... % 0 = normal drooping regulator
                            ...    % 1 = drooping regulator with conditional droop compensation from the dissertation
                            0, ... % 0 = reference is considered current reference
                            ...    % 1 = reference is considered power reference
                            700); % Nominal voltage for power calculations

%% Create smootherstep-based load current reference profile for L1 varying between -1.0 and 0.0
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
KL1 = -(6*x.^5 - 15*x.^4 + 10*x.^3);
loadProfiles(1,:) = KL1;

%% Create reference profile for L2
load2CurrentRef = min((-40+40 * sin(steps/5000 * 2*pi+pi/13)) / 100, -0.3);
loadProfiles(2,:) = load2CurrentRef;

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources+numberOfLoads, 1);
droopConduc = zeros(numberOfSources+numberOfLoads, 1);
R_MG = 0;

%% Run through the calculations
for n=steps
    %% Load handling
    for l=1:numberOfLoads
```

```
        if n==1
            loads(l) = setV_DC(loads(l), DCRefs(1), 0);
        else
            loads(l) = setV_DC(loads(l), DCVolt(n-1), 0);
        end
        loads(l) = setRef(loads(l), loadProfiles(l,n));
        loads(l) = calculateActualK(loads(l));
        KappaMatrix(l,l) = getUsedK(loads(l));
    end

    % Calculate the load currents
    loadCurrentArrays(:,n) = KappaMatrix * InomL;

    %% DC voltage handling
    % Calculate R_MG
    droopConduc = droopConduc * 0;
    for r=1:numberOfSources % Go though the sources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = (B_VS(r) & B_runS(r))/droopResis(r);
    end

    for l=1:numberOfLoads % Go though the loads
        [UVRegHit, OVRegHit] = getRegulatorActivations(loads(l));
        if UVRegHit == 1
            droopConduc(numberOfSources+l) = 1/((UVRefs(l) * UVDroop(l)) / InomL(l));
        end

        if OVRegHit == 1
            droopConduc(numberOfSources+l) = 1/((OVRefs(l) * OVDroop(l)) / InomL(l));
        end
    end
    R_MG = 1/sum(droopConduc);

    % Calculate the DC voltage
    if n==1 % If first step use default DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, sum(loadCurrentArrays(:,n)), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), sum(sourceCurrentArrays(:,n-1)), sum(loadCurrentArrays(:,n)), R_MG);
    end

    %% Source handling
    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        GammaMatrix(j,j) = min(max(Gamma(DCVolt(n), DCRefs(j), DCDroops(j)), negLims(j)), posLims(j));
        B_VS(j) = GammaMatrix(j,j) > negLims(j) & GammaMatrix(j,j) < posLims(j);
    end

    % Calculate the source currents
    sourceCurrentArrays(:,n) = GammaMatrix * InomS;
end
```

Appendix 5

MATLAB source code for modeling a system with load inverter class incorporating load circulating currents and capability for current limitation and stopping

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [10; 20; 30; 40];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('F1_ModelingWholeSystemWithCircCurr:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);

%% Initialize load modeling variables
% Order of the loads is as follows: [L1, L2...]
InomL = [100]; % Load nominal currents
UVRefs = [700*(1-0.02*0.8)];
UVDroop = [0.02];
OVRefs = [710];
OVDroop = [0.02];
if ~isequal(length(InomL), length(UVRefs), length(UVDroop), length(OVRefs), length(OVDroop))
    error('E3_ModelingLoadVoltRegsOptRegBoolArrays_twoloads:SourceParameterMismatch', ...
          'The number of load parameters do not match each other.');
end
numberOfLoads = length(InomL);
loads(1,1) = inverterAdvReg(InomL(1), ...
                            UVRefs(1), ...
                            UVDroop(1), ...
                            OVRefs(1), ...
                            OVDroop(1), ...
                            1, ...    % 0 = normal drooping regulator
                            ...       % 1 = drooping regulator with conditional droop compensation from the dissertation
                            0, ...    % 0 = reference is considered current reference
                            ...       % 1 = reference is considered power reference
                            700);     % Nominal voltage for power calculations

%% Create smootherstep-based load current reference profile for L1 varying between -1.0 and 0.0
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
KL1 = -(6*x.^5 - 15*x.^4 + 10*x.^3);
loadProfiles(1,:) = KL1;

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources+numberOfLoads, 1);
droopConduc = zeros(numberOfSources+numberOfLoads, 1);
R_MG = 0;

I_C = zeros(numberOfSources+numberOfLoads);
I_nomAll = [InomS; InomL];
X = zeros(numberOfSources+numberOfLoads, 1);
B_V = ones(numberOfSources+numberOfLoads, 1);
B_run = ones(numberOfSources+numberOfLoads, 1);
measErrors = [1; 0.5; -1; 0; -1.5];
activeDCRefs = [DCRefs; UVRefs];
activeDroops = [DCDroops; UVDroop];
GammaCMatrix = zeros(numberOfSources+numberOfLoads);

ICSumArrays = zeros(numberOfSources+numberOfLoads,length(steps));
IArrays = zeros(numberOfSources+numberOfLoads,length(steps));

posLims = ones(numberOfSources+numberOfLoads, 1) * inf;
negLims = ones(numberOfSources+numberOfLoads, 1) * -inf;

%% Run through the calculations
for n=steps
    % Use this to set load limit to 50% at index 2250
    % if n==2250
    %     negLims(5) = -0.5;
    % end

    % Use this to stop the load at index 2750
    % if n==2750
    %     B_run(5) = 0;
```

```
% end

%% Load handling
for l=1:numberOfLoads
    % Set load's DC voltage measurement
    if n==1
        loads(l) = setV_DC(loads(l), DCRefs(1), measErrors(numberOfSources+l));
    else
        loads(l) = setV_DC(loads(l), DCVolt(n-1), measErrors(numberOfSources+l));
    end
    % Assign load its limits...
    loads(l) = setLimits(loads(l), posLims(numberOfSources+l), negLims(numberOfSources+l));
    % its run status...
    loads(l) = setRun(loads(l), B_run(numberOfSources+l));
    % and load
    loads(l) = setRef(loads(l), loadProfiles(l,n));
    % Tell load to calculate its K value
    loads(l) = calculateActualK(loads(l));
    % Assign the K value to the Xi array
    X(numberOfSources+l) = getUsedK(loads(l));
    % Check the voltage regulator statuses
    [UVRegHit, OVRegHit] = getRegulatorActivations(loads(l));
    % If voltage regulator is active, set the load as voltage regulating device
    B_V(numberOfSources+l) = UVRegHit || OVRegHit;

    % Update arrays for system's active DC references and droops
    activeDCRefs(numberOfSources+l) = getActiveDCRef(loads(l));
    activeDroops(numberOfSources+l) = getActiveDCDroop(loads(l));
end

% Calculate the new load currents
IArrays((numberOfSources+1:numberOfSources+numberOfLoads),n) = ...
    diag(X(numberOfSources+1:numberOfSources+numberOfLoads)) * ...
    I_nomAll(numberOfSources+1:numberOfSources+numberOfLoads);

%% DC voltage handling
% Calculate R_MG
droopConduc = droopConduc * 0;
for r=1:numberOfSources+numberOfLoads % Go though the sources
    if r <= numberOfSources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = (B_V(r) & B_run(r))/droopResis(r);
    else
        [UVRegHit, OVRegHit] = getRegulatorActivations(loads(r-numberOfSources));
        if UVRegHit == 1
            L1UVRegActive(n) = 1;
        end

        droopResis(r) = 1/getDroopConduc(loads(r-numberOfSources));
        droopConduc(r) = getDroopConduc(loads(r-numberOfSources));
    end
end

R_MG = 1/sum(droopConduc);
R_MGArray(n) = R_MG;

% Calculate the DC voltage
if n==1 % If first step use default DC voltage and source currents
    DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, ...
        sum(IArrays((numberOfSources+1:numberOfSources+numberOfLoads),n)), R_MG);
else % Else use previous step DC voltage and source currents
    DCVolt(n) = calculateDCVoltage( ...
        DCVolt(n-1), ...
        sum(IArrays(1:numberOfSources,n-1)), ...
        sum(IArrays((numberOfSources+1:numberOfSources+numberOfLoads),n)), ...
        R_MG);
end

%% Source handling
% Populate the Gamma-matrix with new values according to
% the DC-link voltage and inverter parameters
for j=1:numberOfSources
    X(j) = min(max(Gamma(DCVolt(n), activeDCRefs(j), activeDroops(j)), negLims(j)), posLims(j));
    B_V(j) = X(j) > negLims(j) & X(j) < posLims(j);
end

% Populate Gamma_C matrix
for a = 1:numberOfSources+numberOfLoads
    for b = 1:numberOfSources+numberOfLoads
        if B_V(a) && B_V(b)
            if a==b
                % Diagonal
                if a <= numberOfSources % Only sources, loads are handled in the class
                    GammaCMatrix(a,a) = [Gamma_Ca(DCVolt(n), measErrors(a), activeDCRefs(a), activeDroops(a))];
                end
            else
                % Off-diagonal (circulating current from source a to b)
                GammaCMatrix(a,b) = Gamma_Cab(DCVolt(n), measErrors(a), droopResis(a), ...
                    I_nomAll(b), measErrors(b), droopResis(b));
            end
        else
            GammaCMatrix(a,b) = 0;
        end
```

```
        end
    end

    % Calculate the currents
    I_C = (diag(X) + GammaCMatrix) * diag(I_nomAll);

    % Check for limited and shutdown devices
    for k = 1:numberOfSources+numberOfLoads
        I_k = sum(I_C(k,:));

        if k <= numberOfSources % Check sources
            B_V(k) = I_k < posLims(k)*I_nomAll(k) && I_k > negLims(k)*I_nomAll(k);
        else
            % For loads assign them the circulating current value
            % B_V for loads is assigned when K is calculated
            loads(k-numberOfSources) = setCirCurr(loads(k-numberOfSources), (I_k - I_C(k,k)));
        end

        % If device is limited or stopped, zero its corresponding column and line
        if ~(B_V(k) && B_run(k))
            I_C(k,:) = 0;
            I_C(:,k) = 0;
        end

        % If device is limited but running, set its corresponding diagonal
        % value to the limited one
        if ~B_V(k) && B_run(k)
            I_C(k,k) = min(max(I_k, negLims(k)*I_nomAll(k)), posLims(k)*I_nomAll(k));
        end
    end

    % Calculate source total currents
    ICSumArrays(:,n) = I_C*ones(numberOfSources+numberOfLoads,1);

    % Calculate load supplying currents
    IArrays(:,n) = diag(I_C);
end
```

Appendix 6

Conductance matrix of a system with four voltage sources and a single load

$$
\mathbf{G}_{\text{nodes}} = \begin{bmatrix}
G_{\text{DS}_1} + G_{\text{C}_{1+}} & -G_{\text{C}_{1+}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -G_{\text{DS}_1} \\
-G_{\text{C}_{1+}} & G_{\text{DS}_2} + G_{\text{C}_{1+}} + G_{\text{C}_{2+}} & -G_{\text{C}_{2+}} & 0 & 0 & 0 & 0 & 0 & -G_{\text{DS}_2} & 0 \\
0 & -G_{\text{C}_{2+}} & G_{\text{DS}_3} + G_{\text{C}_{2+}} + G_{\text{C}_{3+}} & -G_{\text{C}_{3+}} & 0 & 0 & 0 & -G_{\text{DS}_3} & 0 & 0 \\
0 & 0 & -G_{\text{C}_{3+}} & G_{\text{DS}_4} + G_{\text{C}_{3+}} + G_{\text{C}_{4+}} & -G_{\text{C}_{4+}} & 0 & -G_{\text{DS}_4} & 0 & 0 & 0 \\
0 & 0 & 0 & -G_{\text{C}_{4+}} & G_{\text{DL}_1} + G_{\text{C}_{4+}} & -G_{\text{DL}_1} & 0 & -G_{\text{C}_{4-}} & 0 & 0 \\
0 & 0 & 0 & 0 & -G_{\text{DL}_1} & G_{\text{DL}_1} + G_{\text{C}_{4-}} & -G_{\text{C}_{4-}} & G_{\text{DS}_4} + G_{\text{C}_{3-}} + G_{\text{C}_{4-}} & -G_{\text{C}_{3-}} & 0 \\
0 & 0 & 0 & -G_{\text{DS}_4} & 0 & -G_{\text{C}_{4-}} & 0 & -G_{\text{C}_{3-}} & G_{\text{DS}_3} + G_{\text{C}_{2-}} + G_{\text{C}_{3-}} & -G_{\text{C}_{2-}} \\
0 & 0 & -G_{\text{DS}_3} & 0 & 0 & 0 & 0 & -G_{\text{C}_{2-}} & G_{\text{DS}_2} + G_{\text{C}_{1-}} + G_{\text{C}_{2-}} & -G_{\text{C}_{1-}} \\
0 & -G_{\text{DS}_2} & 0 & 0 & 0 & 0 & 0 & 0 & -G_{\text{C}_{1-}} & G_{\text{DS}_1} + G_{\text{C}_{1-}} \\
-G_{\text{DS}_1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

Appendix 7

MATLAB source code for performing nodal analysis on a system of four voltage sources and one load

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [10; 20; 30; 40];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('G1_NodalAnalysis:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);

%% Initialize load modeling variables
% Order of the loads is as follows: [L1, L2...]
InomL = [100]; % Load nominal currents
% UVRefs = [700*(1-0.02*0.8)];
UVRefs = [100];
UVDroop = [0.02];
% OVRefs = [710];
OVRefs = [710];
OVDroop = [0.02];
if ~isequal(length(InomL), length(UVRefs), length(UVDroop), length(OVRefs), length(OVDroop))
    error('G1_NodalAnalysis:SourceParameterMismatch', ...
          'The number of load parameters do not match each other.');
end
numberOfLoads = length(InomL);
loads(1,1) = inverterAdvReg(InomL(1), ...
                            UVRefs(1), ...
                            UVDroop(1), ...
                            OVRefs(1), ...
                            OVDroop(1), ...
                            1, ...   % 0 = normal drooping regulator
                            ...      % 1 = drooping regulator with conditional droop compensation from the dissertation
                            0, ...   % 0 = reference is considered current reference
                            ...      % 1 = reference is considered power reference
                            700);    % Nominal voltage for power calculations

%% Create smootherstep-based load reference profile for L1 varying between -1.0 and 0.0
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
KL1 = -(6*x.^5 - 15*x.^4 + 10*x.^3);
loadProfiles(1,:) = KL1;

%% Initialize DC voltage modeling variables
DCVoltages = zeros(numberOfSources+numberOfLoads, length(steps));
droopResis = zeros(numberOfSources+numberOfLoads, 1);
droopConduc = zeros(numberOfSources+numberOfLoads, 1);
conductanceMatrix = zeros(10);
UpsilonArray = [DCRefs(1);
                DCRefs(2);
                DCRefs(3);
                DCRefs(4);
                sum(DCRefs.*InomS)/sum(InomS);
                0;
                0;
                0;
                0;
                0];
UpsilonArray_pre = UpsilonArray;
dcNodeCurrentMatrix = [0; 0; 0; 0; 0; 0; 0; 0; 0; 0];

%% Initialize cable resistances
RC1p = 0.001;
RC2p = 0.001;
RC3p = 0.001;
RC4p = 0.01;
RC1n = 0.001;
RC2n = 0.001;
RC3n = 0.001;
RC4n = 0.01;

I = zeros(numberOfSources+numberOfLoads, 1);
I_nomAll = [InomS; InomL];
X = zeros(numberOfSources+numberOfLoads, 1);
B_V = ones(numberOfSources+numberOfLoads, 1);
B_run = ones(numberOfSources+numberOfLoads, 1);
```

```
% measErrors = [1; 0.5; -1; 0; -1.5];
measErrors = [0; 0; 0; 0; 0];
activeDCRefs = [DCRefs; UVRefs];
activeDroops = [DCDroops; UVDroop];

ICSumArrays = zeros(numberOfSources+numberOfLoads,length(steps));
IArrays = zeros(numberOfSources+numberOfLoads,length(steps));

posLims = ones(numberOfSources+numberOfLoads, 1) * inf;
negLims = ones(numberOfSources+numberOfLoads, 1) * -inf;

%% Run through the calculations
for n=steps
    %% Load handling
    for l=1:numberOfLoads
        % Set load's DC voltage measurement
        if n==1
            loads(l) = setV_DC(loads(l), DCRefs(1), measErrors(numberOfSources+l));
        else
            loads(l) = setV_DC(loads(l), DCVoltages(5,n-1), measErrors(numberOfSources+l));
        end
        % Assign load its limits...
        loads(l) = setLimits(loads(l), posLims(numberOfSources+l), negLims(numberOfSources+l));
        % its run status...
        loads(l) = setRun(loads(l), B_run(numberOfSources+l));
        % and load
        loads(l) = setRef(loads(l), loadProfiles(l,n));
        % Tell load to calculate its K value
        loads(l) = calculateActualK(loads(l));
        % Assign the K value to the Xi array
        X(numberOfSources+l) = getUsedK(loads(l));
        % Check the voltage regulator statuses
        [UVRegHit, OVRegHit] = getRegulatorActivations(loads(l));
        % If voltage regulator is active, set the load as voltage regulating device
        B_V(numberOfSources+l) = UVRegHit || OVRegHit;

        % Update arrays for system's active DC references and droops
        activeDCRefs(numberOfSources+l) = getActiveDCRef(loads(l));
        activeDroops(numberOfSources+l) = getActiveDCDroop(loads(l));

        I(numberOfSources+l) = X(numberOfSources+l) * I_nomAll(numberOfSources+l);
    end

    %% DC voltage handling
    % Calculate droop resistances and conductances
    for r=1:numberOfSources+numberOfLoads % Go though the sources
        if r <= numberOfSources
            droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
            droopConduc(r) = (B_V(r) & B_run(r))/droopResis(r);
        else
            droopResis(r) = 1/getDroopConduc(loads(r-numberOfSources));
            droopConduc(r) = getDroopConduc(loads(r-numberOfSources));
        end
    end

    conductanceMatrix = [
        droopConduc(1)+1/RC1p, -1/RC1p, 0, 0, 0, 0, 0, 0, -droopConduc(1);
        -1/RC1p, droopConduc(2)+1/RC1p+1/RC2p, -1/RC2p, 0, 0, 0, 0, -droopConduc(2), 0;
        0, -1/RC2p, droopConduc(3)+1/RC2p+1/RC3p, -1/RC3p, 0, 0, 0, -droopConduc(3), 0, 0;
        0, 0, -1/RC3p, droopConduc(4)+1/RC3p+1/RC4p, -1/RC4p, 0, -droopConduc(4), 0, 0, 0;
        0, 0, 0, -1/RC4p, droopConduc(5)+1/RC4p, -droopConduc(5), 0, 0, 0, 0;
        0, 0, 0, 0, -droopConduc(5), droopConduc(5)+1/RC4n, -1/RC4n, 0, 0, 0;
        0, 0, 0, -droopConduc(4), 0, -1/RC4n, droopConduc(4)+1/RC3n+1/RC4n, -1/RC3n, 0, 0;
        0, 0, -droopConduc(3), 0, 0, 0, -1/RC3n, droopConduc(3)+1/RC2n+1/RC3n, -1/RC2n, 0;
        0, -droopConduc(2), 0, 0, 0, 0, 0, -1/RC2n, droopConduc(2)+1/RC1n+1/RC2n, -1/RC1n;
        -droopConduc(1), 0, 0, 0, 0, 0, 0, 0, -1/RC1n, droopConduc(1)+1/RC1n;
        ];

    dcNodeCurrentMatrix = [
        I(1) + (1/RC1p) * (UpsilonArray(2,1) - UpsilonArray(1,1));
        I(2) + (1/RC1p) * (UpsilonArray(1,1) - UpsilonArray(2,1)) + (1/RC2p) * (UpsilonArray(3,1) - UpsilonArray(2,1));
        I(3) + (1/RC2p) * (UpsilonArray(2,1) - UpsilonArray(3,1)) + (1/RC3p) * (UpsilonArray(4,1) - UpsilonArray(3,1));
        I(4) + (1/RC3p) * (UpsilonArray(3,1) - UpsilonArray(4,1)) + (1/RC4p) * (UpsilonArray(5,1) - UpsilonArray(4,1));
        I(5) + (1/RC4p) * (UpsilonArray(4,1) - UpsilonArray(5,1));
        -I(5) + (1/RC4n) * (UpsilonArray(7,1) - UpsilonArray(6,1));
        -I(4) + (1/RC3n) * (UpsilonArray(8,1) - UpsilonArray(7,1)) + (1/RC4n) * (UpsilonArray(6,1) - UpsilonArray(7,1));
        -I(3) + (1/RC2n) * (UpsilonArray(9,1) - UpsilonArray(8,1)) + (1/RC3n) * (UpsilonArray(7,1) - UpsilonArray(8,1));
        -I(2) + (1/RC1n) * (UpsilonArray(10,1) - UpsilonArray(9,1)) + (1/RC2n) * (UpsilonArray(8,1) - UpsilonArray(9,1));
        -I(1) + (1/RC1n) * (UpsilonArray(9,1) - UpsilonArray(10,1));
    ];

    UpsilonArray = UpsilonArray_pre + pinv(conductanceMatrix) * dcNodeCurrentMatrix;

    DCVoltages(1,n) = UpsilonArray(1,1) - UpsilonArray(10,1);
    DCVoltages(2,n) = UpsilonArray(2,1) - UpsilonArray(9,1);
    DCVoltages(3,n) = UpsilonArray(3,1) - UpsilonArray(8,1);
    DCVoltages(4,n) = UpsilonArray(4,1) - UpsilonArray(7,1);
    DCVoltages(5,n) = UpsilonArray(5,1) - UpsilonArray(6,1);

    UpsilonArray_pre = UpsilonArray;

    %% Source handling
    % Populate the Gamma-matrix with new values according to
```

```
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        X(j) = min(max(Gamma(DCVoltages(j,n)+measErrors(j), activeDCRefs(j),...
            activeDroops(j)), negLims(j)), posLims(j)) * B_run(j);
        B_V(j) = X(j) > negLims(j) & X(j) < posLims(j);
    end

    % Calculate the currents
    I = diag(X) * I_nomAll;
    IArrays(:,n) = I;
end
```

Appendix 8

MATLAB source code for performing nodal analysis on a system with two loads

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [100];            % Source nominal currents
DCRefs = [700];       % Source DC voltage references
DCDroops = [0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('G3_NodalAnalysisTwoLoads:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);

%% Initialize load modeling variables
% Order of the loads is as follows: [L1, L2...]
InomL = [100; 100]; % Load nominal currents
UVRefs = [700*0.98; 684];
UVDroop = [0.02; 0.02];
OVRefs = [710; 710];
OVDroop = [0.02; 0.02];
if ~isequal(length(InomL), length(UVRefs), length(UVDroop), length(OVRefs), length(OVDroop))
    error('G3_NodalAnalysisTwoLoads:SourceParameterMismatch', ...
          'The number of load parameters do not match each other.');
end
numberOfLoads = length(InomL);
loads(1,1) = inverterAdvReg(InomL(1), ...
                            UVRefs(1), ...
                            UVDroop(1), ...
                            OVRefs(1), ...
                            OVDroop(1), ...
                            1, ...   % 0 = normal drooping regulator
                            ...      % 1 = drooping regulator with conditional droop compensation from the dissertation
                            1, ...   % 0 = reference is considered current reference
                            ...      % 1 = reference is considered power reference
                            700);    % Nominal voltage for power calculations

loads(1,2) = inverterAdvReg(InomL(2), ...
                           UVRefs(2), ...
                           UVDroop(2), ...
                           OVRefs(2), ...
                           OVDroop(2), ...
                           1, ... % 0 = normal drooping regulator
                           ...    % 1 = drooping regulator with conditional droop compensation from the dissertation
                           1, ... % 0 = reference is considered current reference
                           ...    % 1 = reference is considered power reference
                           700); % Nominal voltage for power calculations

%% Create smootherstep-based load reference profile for L1 varying between -1.0 and 0.0
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
KL1 = -(6*x.^5 - 15*x.^4 + 10*x.^3);
loadProfiles(1,:) = KL1;

%% Create static reference profile for L2
load2CurrentRef = min((-40+40 * sin(steps/5000 * 2*pi+pi/13)) / 100, -0.3);
loadProfiles(2,:) = load2CurrentRef;

%% Initialize DC voltage modeling variables
DCVoltages = zeros(numberOfSources+numberOfLoads, length(steps));
droopResis = zeros(numberOfSources+numberOfLoads, 1);
droopConduc = zeros(numberOfSources+numberOfLoads, 1);
conductanceMatrix = zeros(10);
UpsilonArray = [DCRefs(1);
                sum(DCRefs.*InomS)/sum(InomS);
                sum(DCRefs.*InomS)/sum(InomS);
                0;
                0;
                0];
UpsilonArray_pre = UpsilonArray;
dcNodeCurrentMatrix = [0; 0; 0; 0; 0; 0];

%% Initialize cable resistances
% Case 1
RC1p = 0.010;
RC2p = 0.001;
RC1n = 0.010;
RC2n = 0.001;
```

```
% % Case 2
% RC1p = 0.001;
% RC2p = 0.010;
% RC1n = 0.001;
% RC2n = 0.010;
%
% % Case 3
% RC1p = 0.010;
% RC2p = 0.010;
% RC1n = 0.010;
% RC2n = 0.010;

I = zeros(numberOfSources+numberOfLoads, 1);
I_nomAll = [InomS; InomL];
X = zeros(numberOfSources+numberOfLoads, 1);
B_V = ones(numberOfSources+numberOfLoads, 1);
B_run = ones(numberOfSources+numberOfLoads, 1);
% measErrors = [1; 0.5; -1; 0; -1.5];
measErrors = [0; 0; 0; 0; 0];
activeDCRefs = [DCRefs; UVRefs];
activeDroops = [DCDroops; UVDroop];

ICSumArrays = zeros(numberOfSources+numberOfLoads,length(steps));
IArrays = zeros(numberOfSources+numberOfLoads,length(steps));

posLims = ones(numberOfSources+numberOfLoads, 1) * inf;
negLims = ones(numberOfSources+numberOfLoads, 1) * -inf;

posLims(1) = 1.0;

%% Run through the calculations
for n=steps
    %% Load handling
    for l=1:numberOfLoads
        % Set load's DC voltage measurement
        if n==1
            loads(l) = setV_DC(loads(l), DCRefs(1), measErrors(numberOfSources+l));
        else
            loads(l) = setV_DC(loads(l), DCVoltages(numberOfSources+l,n-1), measErrors(numberOfSources+l));
        end
        % Assign load its limits...
        loads(l) = setLimits(loads(l), posLims(numberOfSources+l), negLims(numberOfSources+l));
        % its run status...
        loads(l) = setRun(loads(l), B_run(numberOfSources+l));
        % and load
        loads(l) = setRef(loads(l), loadProfiles(l,n));
        % Tell load to calculate its K value
        loads(l) = calculateActualK(loads(l));
        % Assign the K value to the Xi array
        X(numberOfSources+l) = getUsedK(loads(l));
        % Check the voltage regulator statuses
        [UVRegHit, OVRegHit] = getRegulatorActivations(loads(l));
        % If voltage regulator is active, set the load as voltage regulating device
        B_V(numberOfSources+l) = UVRegHit || OVRegHit;

        % Update arrays for system's active DC references and droops
        activeDCRefs(numberOfSources+l) = getActiveDCRef(loads(l));
        activeDroops(numberOfSources+l) = getActiveDCDroop(loads(l));

        I(numberOfSources+l) = X(numberOfSources+l) * I_nomAll(numberOfSources+l);
    end

    %% DC voltage handling
    % Calculate droop resistances and conductances
    for r=1:numberOfSources+numberOfLoads % Go though the sources
        if r <= numberOfSources
            droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
            droopConduc(r) = (B_V(r) & B_run(r))/droopResis(r);
        else
            droopResis(r) = 1/getDroopConduc(loads(r-numberOfSources));
            droopConduc(r) = getDroopConduc(loads(r-numberOfSources));
        end
    end

    conductanceMatrix = [
        droopConduc(1)+1/RC1p, -1/RC1p, 0, 0, 0, -droopConduc(1);
        -1/RC1p, droopConduc(2)+1/RC1p+1/RC2p, -1/RC2p, 0, -droopConduc(2), 0;
        0, -1/RC2p, droopConduc(3)+1/RC2p, -droopConduc(3), 0, 0;
        0, 0, -droopConduc(3), droopConduc(3)+1/RC2n, -1/RC2n, 0;
        0, -droopConduc(2), 0, -1/RC2n, droopConduc(2)+1/RC1n, -1/RC1n;
        -droopConduc(1), 0, 0, 0, -1/RC1n, droopConduc(1)+1/RC1n
        ];

    dcNodeCurrentMatrix = [
        I(1) + (1/RC1p) * (UpsilonArray(2,1) - UpsilonArray(1,1));
        I(2) + (1/RC1p) * (UpsilonArray(1,1) - UpsilonArray(2,1)) + (1/RC2p) * (UpsilonArray(3,1) - UpsilonArray(2,1));
        I(3) + (1/RC2p) * (UpsilonArray(2,1) - UpsilonArray(3,1));
        -I(3) + (1/RC2n) * (UpsilonArray(5,1) - UpsilonArray(4,1));
        -I(2) + (1/RC1n) * (UpsilonArray(6,1) - UpsilonArray(5,1)) + (1/RC2n) * (UpsilonArray(4,1) - UpsilonArray(5,1));
        -I(1) + (1/RC1n) * (UpsilonArray(5,1) - UpsilonArray(6,1));
        ];
```

```
    UpsilonArray = UpsilonArray_pre + pinv(conductanceMatrix) * dcNodeCurrentMatrix;

    DCVoltages(1,n) = UpsilonArray(1,1) - UpsilonArray(6,1);
    DCVoltages(2,n) = UpsilonArray(2,1) - UpsilonArray(5,1);
    DCVoltages(3,n) = UpsilonArray(3,1) - UpsilonArray(4,1);

    UpsilonArray_pre = UpsilonArray;

    %% Source handling
    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        X(j) = min(max(Gamma(DCVoltages(j,n)+measErrors(j), activeDCRefs(j),...
            activeDroops(j)), negLims(j)), posLims(j)) * B_run(j);
        B_V(j) = X(j) > negLims(j) & X(j) < posLims(j);
    end

    % Calculate the currents
    I = diag(X) * I_nomAll;
    IArrays(:,n) = I;
end
```

Appendix 9

Deriving DC voltage source current with DC voltage reference-based state-of-charge balancing functionality

Following the same principles as in Appendix 1, let's assume the V_{DC} follows exactly $V_{\mathrm{DS}}^{*} + V_{\mathrm{B}}^{*}$. Thus the DC voltage equation is as follows:

$$V_{\mathrm{DC}} = V_{\mathrm{DS}}^{*} + V_{\mathrm{B}}^{*}$$
$$V_{\mathrm{DC}} = V_{\mathrm{S}}^{*}\left(1 - \frac{I_{\mathrm{S}}}{I_{\mathrm{nomS}}}\zeta_{\mathrm{S}}\right) + V_{\mathrm{B}}^{*}$$

To distinguish between normal voltage source and energy storage related currents, let's rename here the currents I_{S} and I_{nomS} as I_{E} and I_{nomE} respectively. Let's keep the V_{S}^{*} and ζ_{S} names as they are, as they point to the operation as a voltage source, not to is the device connected to energy storage or not. Now, let's solve for I_{E}:

$$V_{\mathrm{DC}} = V_{\mathrm{S}}^{*}\left(1 - \frac{I_{\mathrm{E}}}{I_{\mathrm{nomE}}}\zeta_{\mathrm{S}}\right) + V_{\mathrm{B}}^{*}$$
$$V_{\mathrm{DC}} = V_{\mathrm{S}}^{*} - \frac{V_{\mathrm{S}}^{*} I_{\mathrm{E}} \zeta_{\mathrm{S}}}{I_{\mathrm{nomE}}} + V_{\mathrm{B}}^{*}$$
$$V_{\mathrm{DC}} - V_{\mathrm{S}}^{*} - V_{\mathrm{B}}^{*} = -\frac{V_{\mathrm{S}}^{*} I_{\mathrm{E}} \zeta_{\mathrm{S}}}{I_{\mathrm{nomE}}}$$
$$\frac{V_{\mathrm{S}}^{*} I_{\mathrm{E}} \zeta_{\mathrm{S}}}{I_{\mathrm{nomE}}} = V_{\mathrm{S}}^{*} - V_{\mathrm{DC}} + V_{\mathrm{B}}^{*}$$
$$V_{\mathrm{S}}^{*} I_{\mathrm{E}} \zeta_{\mathrm{S}} = \left(V_{\mathrm{S}}^{*} - V_{\mathrm{DC}} + V_{\mathrm{B}}^{*}\right) I_{\mathrm{nomE}}$$
$$I_{\mathrm{E}} = \frac{V_{\mathrm{S}}^{*} - V_{\mathrm{DC}} + V_{\mathrm{B}}^{*}}{V_{\mathrm{S}}^{*} \zeta_{\mathrm{S}}} I_{\mathrm{nomE}}$$
$$I_{\mathrm{E}} = \left(\frac{V_{\mathrm{S}}^{*} - V_{\mathrm{DC}}}{V_{\mathrm{S}}^{*} \zeta_{\mathrm{S}}} + \frac{V_{\mathrm{B}}^{*}}{V_{\mathrm{S}}^{*} \zeta_{\mathrm{S}}}\right) I_{\mathrm{nomE}}$$
$$= \left(\Gamma_{\mathrm{S}} + \Gamma_{\mathrm{B}}\right) I_{\mathrm{nomE}}$$

Appendix 10

MATLAB code for modeling energy storage state-of-charge balancing with identically dimensioned DC/DC converters and storages at zero load

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [25; 25; 25; 25];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('I1_ModelingESS:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
sourceCurrentArrays = zeros(numberOfSources,length(steps));
B_VS = ones(numberOfSources, 1);
B_runS = ones(numberOfSources, 1);
posLims = ones(numberOfSources, 1) * inf;
negLims = ones(numberOfSources, 1) * -inf;

%% Initialize energy storage variables
initSoCs = [ 0.76; 0.66; 1.00; 0.67];
SoCArray = ones(numberOfSources,length(steps)) .* initSoCs;
ESSMaxCaps = [50000; 50000; 50000; 50000];

%% Initialize ESS Balancing Variables
V_BSArray = zeros(numberOfSources,length(steps));
GammaEMatrix = zeros(numberOfSources,numberOfSources);
droop_scale = 0.02;
V_scale = 1;
SoC_TArray =  ones(1,length(steps)) * sum(SoCArray(:,1) .* ESSMaxCaps(:)) / sum(ESSMaxCaps);

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources, 1);
droopConduc = zeros(numberOfSources, 1);
R_MG = 0;

%% Force load to zero
load = zeros(1,length(steps));

%% Run through the calculations
for n=steps
    %% DC voltage handling
    % Calculate R_MG
    for r=1:numberOfSources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = (B_VS(r) & B_runS(r))/droopResis(r);
    end
    R_MG = 1/sum(droopConduc);

    % Calculate the DC voltage
    if n==1 % If first step use default values
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, load(n), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), ...
            sum(sourceCurrentArrays(:,n-1)), load(n), R_MG);
    end

    %% Source handling
    if n>1
        SoC_TArray(n) = min(SoCArray(:,n-1));
        % SoC_TArray(n) = max(SoCArray(:,n-1));
        % SoC_TArray(n) = sum(SoCArray(:,n-1) .* ESSMaxCaps(:)) / sum(ESSMaxCaps);
    end

    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        if n>=1000
            V_BSArray(j,n) = ((SoCArray(j,n-1) - SoC_TArray(n)) / (droop_scale * V_scale));
        end

        GammaMatrix(j,j) = min(max(Gamma(DCVolt(n), DCRefs(j), DCDroops(j)) ...
                           + (V_BSArray(j,n)/(DCDroops(1) * DCRefs(j))), negLims(j)), posLims(j));
        B_VS(j) = GammaMatrix(j,j) > negLims(j) & GammaMatrix(j,j) < posLims(j);
```

```
    end

    % Calculate the source currents
    sourceCurrentArrays(:,n) = (GammaMatrix * InomS);

    % Update the soc values
    for j=1:numberOfSources
        if n>1
            SoCArray(j,n) = SoCArray(j,n-1) + -sourceCurrentArrays(j,n) / ESSMaxCaps(j);
        end
    end
end
```

Appendix 11

Simplifying the DC voltage reference-based state-of-charge balancing Γ_{E}-equation

$$
\begin{aligned}
\Gamma_{\mathrm{E}} &= (\Gamma_{\mathrm{S}} + \Gamma_{\mathrm{B}}) \frac{Q_{\mathrm{maxE}}}{\overline{Q}_{\mathrm{maxE}_n}} \frac{\overline{I}_{\mathrm{nomS}_n}}{I_{\mathrm{nomS}}} \\
&= (\Gamma_{\mathrm{S}} + \Gamma_{\mathrm{B}}) \frac{Q_{\mathrm{maxE}}}{\frac{1}{N} \sum_{n=1}^{N} Q_{\mathrm{maxE}_n}} \frac{\frac{1}{N} \sum_{n=1}^{N} I_{\mathrm{nomS}_n}}{I_{\mathrm{nomS}}} \\
&= (\Gamma_{\mathrm{S}} + \Gamma_{\mathrm{B}}) \frac{Q_{\mathrm{maxE}}}{\sum_{n=1}^{N} Q_{\mathrm{maxE}_n}} \frac{\sum_{n=1}^{N} I_{\mathrm{nomS}_n}}{I_{\mathrm{nomS}}}
\end{aligned}
$$

An array $\mathbf{X}$'s element X_n divided by the sum of the array's elements $\Sigma\mathbf{X}$ results by definition in normalized value of said element $X_n^{\mathrm{p.u}}$, such that sum of the normalized elements equals one, $\Sigma\mathbf{X}^{\mathrm{p.u}} = 1$. Thus we can replace the fractions $\frac{Q_{\mathrm{maxE}}}{\sum_{n=1}^{N} Q_{\mathrm{maxE}_n}}$ and $\frac{\sum_{n=1}^{N} I_{\mathrm{nomS}_n}}{I_{\mathrm{nomS}}}$ with $Q_{\mathrm{maxE}}^{\mathrm{p.u}}$ and $\frac{1}{I_{\mathrm{nomS}}^{\mathrm{p.u}}}$ respectively, and we will have more simplified form of the equation:

$$
\Gamma_{\mathrm{E}} = (\Gamma_{\mathrm{S}} + \Gamma_{\mathrm{B}}) \frac{Q_{\mathrm{maxE}}^{\mathrm{p.u}}}{I_{\mathrm{nomS}}^{\mathrm{p.u}}}
$$

Appendix 12

MATLAB code for modeling energy storage state-of-charge balancing with possibility for different capacities and nominal currents

```
%% Initialize general modeling variables
steps = 1:5000;

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
% InomS = [25; 25; 25; 25];            % Source nominal currents
InomS = [10; 20; 30; 40];            % Source nominal currents
DCRefs = [700; 700; 700; 700];       % Source DC voltage references
DCDroops = [0.02; 0.02; 0.02; 0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('I2_ModelingESSVarLoad:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
sourceCurrentArrays = zeros(numberOfSources,length(steps));
B_VS = ones(numberOfSources, 1);
B_runS = ones(numberOfSources, 1);
posLims = ones(numberOfSources, 1) * inf;
negLims = ones(numberOfSources, 1) * -inf;
InomSAve = sum(InomS)/length(InomS);

% posLims(3) = 0.5;

%% Initialize energy storage variables
initSoCs = [ 0.76; 0.66; 1.00; 0.67];
SoCArray = ones(numberOfSources,length(steps)) .* initSoCs;
% ESSMaxCaps = [50000; 50000; 50000; 50000];
ESSMaxCaps = [30000; 40000; 60000; 70000];
ESSMaxCapsAve = sum(ESSMaxCaps)/length(InomS);

%% Initialize ESS Balancing Variables
V_BSArray = zeros(numberOfSources,length(steps));
GammaEMatrix = zeros(numberOfSources,numberOfSources);
droop_scale = 0.02;
V_scale = 1;
SoC_TArray =  ones(1,length(steps)) * sum(SoCArray(:,1) .* ESSMaxCaps(:)) / sum(ESSMaxCaps);

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources, 1);
droopConduc = zeros(numberOfSources, 1);
R_MG = 0;

%% Create smootherstep-based load profile varying between -50 and 0 A
x = (2*(steps)./max(steps))*1.2;
x(x > 1.2) = 2.4 - x(x > 1.2);
x(x > 1) = 1;
s = 6*x.^5 - 15*x.^4 + 10*x.^3;
load = -50 * s;

%% Run through the calculations
for n=steps
    %% DC voltage handling
    % Calculate R_MG
    for r=1:numberOfSources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = (B_VS(r) & B_runS(r))/droopResis(r);
    end
    R_MG = 1/sum(droopConduc);

    % Calculate the DC voltage
    if n==1 % If first step use default values
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, load(n), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), ...
            sum(sourceCurrentArrays(:,n-1)), load(n), R_MG);
    end

    %% Source handling
    if n>1
        SoC_TArray(n) = sum(SoCArray(:,n-1) .* ESSMaxCaps(:)) / sum(ESSMaxCaps);
    end

    % Calculate the normalized nominal currents and maximum energy capacities
    InomSNorm = InomS ./ sum(InomS);
    ESSMaxCapsNorm = ESSMaxCaps./ sum(ESSMaxCaps);

    % Populate the Gamma-matrix with new values according to
```

```
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        if n>=1000
            V_BSArray(j,n) = ((SoCArray(j,n-1) - SoC_TArray(n)) / (droop_scale * V_scale));
        end

        % No scaling
        % GammaMatrix(j,j) = min(max(Gamma(DCVolt(n), DCRefs(j), DCDroops(j)) ...
        %                       + (V_BSArray(j,n)/(DCDroops(1) * DCRefs(j))), negLims(j)), posLims(j));

        % Scaling for different energy storage sizes
        % GammaMatrix(j,j) = min(max((Gamma(DCVolt(n), DCRefs(j), DCDroops(j)) ...
        %                       + (V_BSArray(j,n)/(DCDroops(1) * DCRefs(j)))) ...
        %                       * (ESSMaxCaps(j)/ESSMaxCapsAve), negLims(j)), posLims(j));

        % Scaling for different energy storage sizes and nominal currents
        % GammaMatrix(j,j) = min(max((Gamma(DCVolt(n), DCRefs(j), DCDroops(j)) ...
        %                       + (V_BSArray(j,n)/(DCDroops(1) * DCRefs(j)))) * (ESSMaxCaps(j)/ESSMaxCapsAve) ...
        %                       * (InomSAve/InomS(j)), negLims(j)), posLims(j));
        GammaMatrix(j,j) = min(max((Gamma(DCVolt(n), DCRefs(j), DCDroops(j)) ...
                              + (V_BSArray(j,n)/(DCDroops(1) * DCRefs(j)))) ...
                              * (ESSMaxCapsNorm(j)/InomSNorm(j)), negLims(j)), posLims(j));

        B_VS(j) = GammaMatrix(j,j) > negLims(j) & GammaMatrix(j,j) < posLims(j);
    end

    % Calculate the source currents
    sourceCurrentArrays(:,n) = (GammaMatrix * InomS);

    % Update the soc values
    for j=1:numberOfSources
        if n>1
            SoCArray(j,n) = SoCArray(j,n-1) + -sourceCurrentArrays(j,n) / ESSMaxCaps(j);
        end
    end
end
```

Appendix 13

Proof that change in droop resistance R_{DS} is change in droop coefficient ζ_{S}

Let's take the drooped voltage's equation 1.1, add R_{B} as the balancing offset resistance, and rewrite the resistance values as specified in equation 3.7:

$$\begin{aligned} V^*_{\mathrm{DS}} &= V^*_{\mathrm{S}} - (R_{\mathrm{DS}} + R_{\mathrm{B}})I_{\mathrm{S}} \\ &= V^*_{\mathrm{S}} - \left(\frac{-\Delta V^*_{\mathrm{S}}}{\Delta I_{\mathrm{S}}} + \frac{-\Delta V^*_{\mathrm{B}}}{\Delta I_{\mathrm{B}}}\right) I_{\mathrm{S}} \end{aligned}$$

As resistance value is mathematically the slope of a curve in current/voltage-plane, we can calculate the slope by selecting two points on the curve and using them as the Δs. The easiest points are the zero and nominal load points, remembering that at nominal load the voltage has dropped according to $V^*\zeta$.

$$V^*_{\mathrm{DS}} = V^*_{\mathrm{S}} - \left(\frac{V^*_{\mathrm{S}}\zeta_{\mathrm{S}}}{I_{\mathrm{nomS}}} + \frac{V^*_{\mathrm{B}}\zeta_{\mathrm{B}}}{I_{\mathrm{B}}}\right) I_{\mathrm{S}}$$

In the balancing offset resistance calculation, as the base droop resistance R_{DS} and its offset R_{B} are being modeled for the same voltage source, from this follows $I_{\mathrm{B}} = I_{\mathrm{nomS}}$. Also $V^*_{\mathrm{B}} = V^*_{\mathrm{S}}$, as an offset to a resistance does not change voltage reference, which would generate current.

$$\begin{aligned} V^*_{\mathrm{DS}} &= V^*_{\mathrm{S}} - \left(\frac{V^*_{\mathrm{S}}\zeta_{\mathrm{S}}}{I_{\mathrm{nomS}}} + \frac{V^*_{\mathrm{S}}\zeta_{\mathrm{B}}}{I_{\mathrm{nomS}}}\right) I_{\mathrm{S}} \\ &= V^*_{\mathrm{S}} - \left(\frac{(\zeta_{\mathrm{S}} + \zeta_{\mathrm{B}})V^*_{\mathrm{S}}}{I_{\mathrm{nomS}}}\right) I_{\mathrm{S}} \\ &= V^*_{\mathrm{S}} - \frac{I_{\mathrm{S}}V^*_{\mathrm{S}}}{I_{\mathrm{nomS}}}(\zeta_{\mathrm{S}} + \zeta_{\mathrm{B}}) \\ &= V^*_{\mathrm{S}}\left(1 - \frac{I_{\mathrm{S}}}{I_{\mathrm{nomS}}}(\zeta_{\mathrm{S}} + \zeta_{\mathrm{B}})\right) \end{aligned}$$

This is completely analogous with the V^*_{DS} equation 1.2, only with ζ_{B} portion added. Thus showing that an offset to the droop resistance R_{DS} only creates an offset to the droop coefficient ζ_{S}.

Appendix 14

MATLAB code for modeling power supply prioritization based on real-life data

```
%% Initialize general modeling variables
steps = 1:3000;
voltageRating = 400;
equipmentPowerArray = zeros(3,length(steps));

%% Initialize source modeling variables
% Order of the sources is as follows: [S1, S2, S3, S4]
InomS = [1700000/voltageRating];            % Source nominal currents
DCRefs = [787];       % Source DC voltage references
DCDroops = [0.02]; % Source DC droop coefficients
if ~isequal(length(InomS), length(DCRefs), length(DCDroops))
    error('M1_powerSupplyPrioritization:SourceParameterMismatch', ...
          'The number of source parameters do not match each other.');
end
numberOfSources = length(InomS);
GammaMatrix = zeros(numberOfSources,numberOfSources);
sourceCurrentArrays = zeros(numberOfSources,length(steps));
B_VS = ones(numberOfSources, 1);
B_runS = ones(numberOfSources, 1);
posLims = ones(numberOfSources, 1) * inf;
negLims = ones(numberOfSources, 1) * -inf;

%% Initialize load modeling variables
% Order of the loads is as follows: [L1, L2...]
InomL = [2000000/voltageRating; 1000000/voltageRating]; % Load nominal currents
% UVRefs = [DCRefs(1) * (1 - DCDroops(1)*0.8)];
UVRefs = [300; DCRefs(1)*(1-DCDroops(1))];
UVDroop = [0.02; 0.02];
% OVRefs = [710];
OVRefs = [900; 900];
OVDroop = [0.02; 0.02];
if ~isequal(length(InomL), length(UVRefs), length(UVDroop), length(OVRefs), length(OVDroop))
    error('M1_powerSupplyPrioritization:SourceParameterMismatch', ...
          'The number of load parameters do not match each other.');
end
numberOfLoads = length(InomL);
KappaMatrix = zeros(numberOfLoads,numberOfLoads);
loadCurrentArrays = zeros(numberOfLoads,length(steps));
loads(1,1) = inverterAdvReg(InomL(1), ...
                            UVRefs(1), ...
                            UVDroop(1), ...
                            OVRefs(1), ...
                            OVDroop(1), ...
                            1, ... % 0 = normal drooping regulator
                            ...    % 1 = drooping regulator with conditional droop compensation from the dissertation
                            0, ... % 0 = reference is considered current reference
                            ...    % 1 = reference is considered power reference
                            700); % Nominal voltage for power calculations

loads(1,2) = inverterAdvReg(InomL(2), ...
                            UVRefs(2), ...
                            UVDroop(2), ...
                            OVRefs(2), ...
                            OVDroop(2), ...
                            1, ... % 0 = normal drooping regulator
                            ...    % 1 = drooping regulator with conditional droop compensation from the dissertation
                            0, ... % 0 = reference is considered current reference
                            ...    % 1 = reference is considered power reference
                            700); % Nominal voltage for power calculations

%% Use the prioritization data as the load reference for L1
prioritizationData = readmatrix("prioritizationData.csv");
load_kW = -prioritizationData(:,3);
loadProfiles(1,:) = load_kW / 2000;

%Synchronize timestamps with the original data
csvTimeThruster = (steps * 0.05) + 2.65;

%% Use L2 as the grid side inverter
loadProfiles(2,:) = zeros(length(steps), 1);

%% Initialize DC voltage modeling variables
DCVolt = zeros(1,length(steps));
droopResis = zeros(numberOfSources+numberOfLoads, 1);
droopConduc = zeros(numberOfSources+numberOfLoads, 1);
R_MG = 0;

R_MGArray = zeros(1,length(steps));
```

```
L1UVRegActive = zeros(1,length(steps));
L2UVRegActive = zeros(1,length(steps));

%% Run through the calculations
for n=steps
    %% Load handling
    for l=1:numberOfLoads
        if n==1
            loads(l) = setV_DC(loads(l), DCRefs(1), 0);
        else
            loads(l) = setV_DC(loads(l), DCVolt(n-1), 0);
        end

        %% L1 is handled normally but L2 is rigid voltage source
        if l==1
            loads(l) = setRef(loads(l), loadProfiles(l,n));
            loads(l) = calculateActualK(loads(l));
            KappaMatrix(l,l) = getUsedK(loads(l));
        else
            loads(l) = setRef(loads(l), loadProfiles(l,n));
            loads(l) = calculateActualK(loads(l));

            [UVRegHit, OVRegHit] = getRegulatorActivations(loads(l));
            % If L2 hits its UV regulator limit, for its output to the
            % difference between the generator and load, i.e. rigid voltage source
            if UVRegHit
                KappaMatrix(l,l) = max(-((KappaMatrix(1,1) * (InomL(1)/InomL(2))) ...
                    + (GammaMatrix(1,1) * (InomS(1)/InomL(2)))),0);
            else
                KappaMatrix(l,l) = getUsedK(loads(l));
            end
        end
    end

    % Calculate the load currents
    loadCurrentArrays(:,n) = KappaMatrix * InomL;

    %% DC voltage handling
    % Calculate R_MG
    droopConduc = droopConduc * 0;
    for r=1:numberOfSources % Go though the sources
        droopResis(r) = (DCRefs(r) * DCDroops(r) / InomS(r));
        droopConduc(r) = (B_VS(r) & B_runS(r))/droopResis(r);
    end

    for l=1:numberOfLoads % Go though the loads
        [UVRegHit, OVRegHit] = getRegulatorActivations(loads(l));
        if UVRegHit == 1
            if l==1
                L1UVRegActive(n) = 1;
                droopConduc(numberOfSources+l) = 1/((UVRefs(l) * UVDroop(l)) / InomL(l));
            elseif l==2
                % Load L2's UV regulator is not droop-based, so it must not influence the R_MG
                L2UVRegActive(n) = 1;
            end
        end

        if OVRegHit == 1
            droopConduc(numberOfSources+l) = 1/((OVRefs(l) * OVDroop(l)) / InomL(l));
        end
    end
    R_MG = 1/sum(droopConduc);
    R_MGArray(n) = R_MG;

    % Calculate the DC voltage
    if n==1 % If first step use default DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCRefs(1), 0, sum(loadCurrentArrays(:,n)), R_MG);
    else % Else use previous step DC voltage and source currents
        DCVolt(n) = calculateDCVoltage(DCVolt(n-1), sum(sourceCurrentArrays(:,n-1)), sum(loadCurrentArrays(:,n)), ...
            R_MG);
    end

    %% Source handling
    % Populate the Gamma-matrix with new values according to
    % the DC-link voltage and inverter parameters
    for j=1:numberOfSources
        GammaMatrix(j,j) = min(max(Gamma(DCVolt(n), DCRefs(j), DCDroops(j)), negLims(j)), posLims(j));
        B_VS(j) = GammaMatrix(j,j) > negLims(j) & GammaMatrix(j,j) < posLims(j);
    end

    % Calculate the source currents
    sourceCurrentArrays(:,n) = GammaMatrix * InomS;

    % Calculate the equipment powers
    equipmentPowerArray(1,n) = sourceCurrentArrays(1,n) * voltageRating; % Generator inverter
    equipmentPowerArray(2,n) = loadCurrentArrays(1,n) * voltageRating; % Thruster inverter
    equipmentPowerArray(3,n) = loadCurrentArrays(2,n) * voltageRating; % Grid Inverter
end
```