Secondary Control Strategies for DC Islanded Microgrids Operation

Andrea Martinelli[1], Alessio La Bella[2] and Riccardo Scattolini[2].

Abstract— **Secondary control architectures for islanded direct-current microgrids are getting interest since they are necessary to manage the voltage references in order to properly distribute the time-varying load demand. To this aim, we propose three different optimization-based secondary control approaches considering the internal units constraints, losses minimization and the continuous satisfaction of the load demand. The described approaches are based on a centralized, distributed and cluster-based optimization strategy, respectively.**

I. Introduction

A microgrid (mG) can be identified as a small-scale network cluster of electrical apparatus, e.g. loads, accumulators and generation units, which can work both islanded and connected to the main grid [1]-[2]. Recent advances in power electronics, batteries and direct-current (DC) energy sources have been fuelling a growing interest towards islanded DC mGs. The DC configuration, besides naturally requiring more straightforward modelling and control approaches, necessitate less power conversion stages with respect to the alternating-current (AC) configuration. One key challenge in DC mGs is the design of primary control strategies that ensure voltage reference tracking of Distributed Generation Units (DGUs). Recent solutions to this problem have been addressed, among others, by exploiting droop control [2], passivity-based control [3] and sliding mode control [4]. However, in the above-mentioned works, the mG closed-loop stability is ensured as far as the control inputs are not saturated, thus assuming that the DGUs are always able to track the voltage references despite the load variations. Higher-layer architectures are thus necessary to properly coordinate the voltage references of the primary controllers, guaranteeing the mG stability despite the presence of control input constraints and time-varying load disturbances. Secondary control approaches are proposed in [4] and [5], where the so-called *current sharing* property is achieved, meaning that the load demand is proportionally distributed among the DGUs independently on the network topology.

Contributions: In this paper, we propose three different secondary control strategies for low-voltage (LV) DC mGs. In all cases, the management of the voltage reference values is obtained by solving an optimization problem, whose solution is exploited to adjust the voltage set points of the primary controllers. Differently from previously mentioned approaches, we embed multiple goals in the management of the mG, namely the control effort balance, the line losses and the generation sources constraints. The first of the three methods is centralized and time-based, the second is distributed and time-based, and finally the third is solved by local DGUs' clusters and it is event-based.

Organization: In the next Section, we introduce the electrical model of an LVDC mG. In Section III, which is the core of this paper, the three secondary control strategies are introduced. Numerical simulations and comparisons among the architectures are performed in Section IV. In Section V, final conclusions and future developments are presented.

II. Microgrid model and primary control

As in [6], the LVDC mG is composed by a set of dynamical agents called *distributed generation units (DGUs)*. Agents' electrical interactions are captured by a weighted digraph $\mathcal{G}_{el} = (\mathcal{V}, \mathcal{E}, W)$. Each node in the set $\mathcal{V} = \{1, \dots, n\}$ hosts a DGU, and a weight $R_{ij}^{-1} \in \mathbb{R}^+$ is associated to each edge e_k belonging to $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. The numerical value assigned to R_{ij} corresponds to the electrical resistance of the power line that connects DGUs *i* and *j*. We collect those parameters in the *weight matrix* $W = \mathrm{diag}(R_{ij}^{-1}) \in \mathbb{R}^{m \times m}$, where $m = |\mathcal{E}|$ is the number of edges. For node i, $\mathcal{N}_i^+ = \{j \in \mathcal{V} \mid (i,j) \in \mathcal{E}\}$ denotes the set of out-neighbours, $\mathcal{N}_i^- = \{j \in \mathcal{V} \mid (j,i) \in \mathcal{E}\}$ the set of in-neighbours, and $\mathcal{N}_i = \mathcal{N}_i^+ \cup \mathcal{N}_i^-$ the set of neighbours. Graph topology can be expressed by means of the so called *incidence matrix* $B_{el} \in \mathbb{R}^{n \times m}$ associated to the electrical graph

$$B_{el} = [b_{ij}], \quad b_{ij} = \begin{cases} 1 & e_j \text{ enters node } i \\ -1 & e_j \text{ leaves node } i \\ 0 & \text{otherwise} \end{cases} . \tag{1}$$

An alternative description of $\mathcal{G}_{el}$ is provided by the *weighted Laplacian matrix*, defined as $\mathcal{L}_w(\mathcal{G}_{el}) = B_{el} W B_{el}'$.

Each DGU is composed, as in [7], by a static generator, e.g. a battery, that feeds a local load through a DC/DC *Buck converter*. Such converter permits to manipulate, by means of the switch, an output voltage which is strictly lower than the input voltage. Furthermore, we assume the switching frequency of the converter to be sufficiently greater than the cut-off frequency of the LC filter. This allows us, as reported in [8], to approximate the switching behaviour of the converter with a continuous average model, with the so called *state-space averaging method*. To this aim, we

[1] A. Martinelli, formerly with the Department of Electronics, Information and Bioengineering, Politecnico di Milano, Italy, is now with the Automatic Control Laboratory, ETH Zürich, Switzerland. Email to: andremar@control.ee.ethz.ch

[2] A. La Bella and R. Scattolini are with the Department of Electronics, Information and Bioengineering, Politecnico di Milano, Italy. Email to: alessio.labella@polimi.it (A. La Bella) and riccardo.scattolini@polimi.it (R. Scattolini).

The work of Alessio La Bella has been financed by the Research Fund for the Italian Electrical System in compliance with the Decree of Minister of Economic Development April 16, 2018.

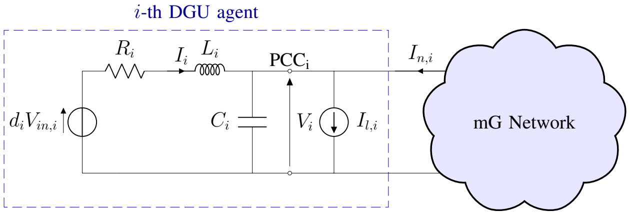


Fig. 1: Average DGU model connected to the mG

introduce a continuous variable $d_i \in (0,1)$, which constitutes the converter's duty cycle, and its complement $d_i^* = 1 - d_i$. The resulting average dynamical system is represented by the linear circuit in Figure 1, with equations

$$\Sigma_i : \begin{cases} C_i\dot{V}_i = I_i - I_{l,i} + I_{n,i} \\ L_i\dot{I}_i = -V_i - R_iI_i + d_iV_{in,i}. \end{cases} \tag{2}$$

The state $x_i = \begin{bmatrix} V_i & I_i \end{bmatrix}'$ is composed by the output voltage V_i and the inductance current I_i. The electrical parameters are the capacitance C_i, the inductance L_i and the inductor resistance R_i, while the exogenous terms are the load current $I_{l,i}$ and input voltage $V_{in,i}$. The manipulable input is the duty cycle d_i and, finally, the term $I_{n,i}$ is the coupling current with the rest of the mG. Several decentralized primary control strategies have been developed, as we reviewed in the Introduction, that guarantee each DGU to be able to sustain a desired voltage reference value V_i^r in spite of the couplings, see [6], [9]. From now on, we assume systems (2) to be controlled with such primary control laws, such that the relation $V_i = V_i^r$ is always satisfied, in nominal conditions, for each DGU.

III. Secondary Control

The aim of the secondary control layer is to manage the vector of reference values $\boldsymbol{V^r} = \begin{bmatrix} V_1^r & \dots & V_n^r \end{bmatrix}'$, in such a way to obtain a desired global performance and/or behaviour and without invalidating the stabilizing property of the primary controllers. The main idea is to implement local outer control loops that update each reference value V_i^r by solving an optimization problem. The adjustments of reference values must take place with time period which is sufficiently longer than the settling time of the inner loops. In this way, we are always guaranteed that systems (2) are in steady-state condition and that $\boldsymbol{V} = \boldsymbol{V^r}$, where $\boldsymbol{V} = \begin{bmatrix} V_1 & \dots & V_n \end{bmatrix}'$. Another consideration concerns the definition of desired global performances. Besides current sharing, which is regarded as the unique goal in the works on secondary control reviewed in the Introduction, one may like to achieve other objectives, such as

- distribute the load demand minimizing the line losses;
- consider the limitations on the duty cycles and on the generator dispensable current;
- penalize voltage deviations from nominal values.

A. Centralized architecture

Firstly, we want to address the secondary control problem as a fully centralized one. Aside from the simplicity of construction, we aim to find a centralized solution in order to provide a benchmark for the next distributed implementations. As anticipated, we consider a situation where systems (2) are in steady-state condition, and a perfect track of V_i^r is achieved. Steady-state equations are

$$\Sigma_i^{ss} : \begin{cases} \bar{I}_i = I_{l,i} - I_{n,i} \\ \bar{d}_i = \frac{\bar{V}_i + R_i(I_{l,i} - \bar{I}_{n,i})}{V_{in,i}} \end{cases}, \tag{3}$$

where $\bar{V}_i$, $\bar{I}_i$, $\bar{d}_i$ and $\bar{I}_{n,i}$ are the state variables, duty cycle and coupling current at the equilibrium. From now on, since we operate in a static environment, we will refer to equilibrium variables without using the upper bar notation. The objective is now to develop a global optimization problem, to determine the optimal vector $\boldsymbol{V}$ $(= \boldsymbol{V^r})$ with respect to a suitable cost function and constraints. For the sake of clarity, before stating the optimization problem some functions need to be defined.

1) Joule line losses: The current $I_{line,k}$ circulating in transmission line (i,j) can be expressed as $I_{line,k} = (V_j - V_i)/R_{ij}$. If we define $I_{line} = \begin{bmatrix} I_{line,1} & \dots & I_{line,m} \end{bmatrix}'$, we may exploit the incidence matrix B_{el} and weight matrix W to express the global dependence between the circulating currents and the voltages as $I_{line} = -WB_{el}'\boldsymbol{V}$.
Finally, the total power dissipated in the transmission lines is: $\mathcal{J}(\boldsymbol{V}) = I_{line}'W^{-1}I_{line} = \boldsymbol{V}'\mathcal{L}_w\boldsymbol{V}$.

2) Duty cycle: Let's define $d = \begin{bmatrix} d_1 & \dots & d_n \end{bmatrix}'$, and $R = \mathrm{diag}(R_1, \dots, R_n)$. The steady-state expression for duty cycle in (3) can be written in compact form for the whole mG as

$$d = V_{in}^{-1}(\boldsymbol{V} + RI_l - RI_n) \tag{4}$$

The vector $I_n = \begin{bmatrix} I_{n,1} & \dots & I_{n,n} \end{bmatrix}'$ of the coupling currents entering the DGUs is

$$I_n = B_{el}I_{line} = -\mathcal{L}_w\boldsymbol{V}. \tag{5}$$

Substituting (5) in (4) one obtains the expression of the duty cycles as a function of the voltages

$$d(\boldsymbol{V}) = V_{in}^{-1}(\mathbb{I}_{n\times n} + R\mathcal{L}_w)\boldsymbol{V} + V_{in}^{-1}RI_l, \tag{6}$$

where $\mathbb{I}_{n\times n}$ is the $n \times n$ identity matrix.
At this stage, the following function is introduced

$$\mathcal{D}(\boldsymbol{V}) = (d(\boldsymbol{V}) - \frac{1}{2}\mathbf{1}_n)'(d(\boldsymbol{V}) - \frac{1}{2}\mathbf{1}_n), \tag{7}$$

where $\mathbf{1}_n$ is a $n \times 1$ vector with 1 in all entries. Equation (7) measures the distance from the 50% of duty cycle operation, in order to avoid generator overloading while distributing the load demand.

3) Internal current: The internal currents $I = \begin{bmatrix} I_1 & \dots & I_n \end{bmatrix}'$ can be written, from the steady-state equation (3) and the coupling term (5), as

$$I(\boldsymbol{V}) = \mathcal{L}_w\boldsymbol{V} + I_l. \tag{8}$$

At this point, we also introduce the following function

$$\mathcal{N}(\boldsymbol{V}) = (\boldsymbol{V} - V^n)'(\boldsymbol{V} - V^n), \tag{9}$$

which expresses the cost for the voltage deviation from the vector of nominal voltages $V^n = \begin{bmatrix} V_1^n & \dots & V_n^n \end{bmatrix}'$.

The centralized global optimization problem follows.

$$\min_{\boldsymbol{V}} \quad \alpha\mathcal{J}(\boldsymbol{V}) + \beta\mathcal{N}(\boldsymbol{V}) + \gamma\mathcal{D}(\boldsymbol{V}) \tag{10a}$$

$$\text{s. t.} \quad \underline{V} \leq \boldsymbol{V} \leq \overline{V} \tag{10b}$$

$$\underline{I} \leq I(\boldsymbol{V}) \leq \overline{I} \tag{10c}$$

$$0 \leq d(\boldsymbol{V}) \leq 1 \tag{10d}$$

$$\alpha, \beta, \gamma \geq 0, \tag{10e}$$

where α, β and γ are positive scalar weights. Constraint (10b) confine voltages inside a suitable band, constraint (10c) restricts internal currents between $\underline{I}$ and $\overline{I}$, while (10d) limits each duty cycle between 0 and 1. Cost function (10a) is quadratic in $\boldsymbol{V}$, and it is composed by a combination of Joule losses, nominal values offsets and control efforts terms. To adopt this strategy a centralized controller must be designed, which operates at a fixed time sample solving problem (10), giving the proper voltage references to the DGUs.

B. Distributed architecture

In this section, we address the problem of reaching the same solution of (10) without the presence of a central entity, but just relying on local communications. To this aim, we proceed by formulating (10) as a *Distributed Constraint Optimization Problem (DCOP)*. In [10], a DCOP problem is described by the tuple

$$P = \langle X, D, F \rangle, \tag{11}$$

where $X = \{x_1, \ldots, x_n\}$ is a finite set of optimization variables and $D = \{D_1, \ldots, D_n\}$ is a set of finite domains of values associated to the variables, in such a way that variable x_i has domain D_i. Finally, $F = \{f_1, \ldots, f_k\}$ is a finite set of cost functions (or constraints), with

$$f_i \; : \; D_h \times \cdots \times D_j \to \mathbb{R}^+ \cup \{\perp\}, \tag{12}$$

where the symbol $\perp$ stays for *infeasible* value. Function f_i maps from the Cartesian product of a subset of domains associated to the subset of variables $x_{f_i} = \{x_h, \ldots, x_j\} \subseteq X$, to a positive scalar. Codomain of f_i is usually called *utility*. Furthermore, according to the terminology adopted in [10], the *utility* functions can be divided into *soft constraints*, when their codomain is $\mathbb{R}^+$, and *hard constraints*, when the codomain is $\{0\} \cup \{\perp\}$. Variables in the set x_{f_i} are referred to as the *scope* of f_i, while the *arity* of the cost function is the number $|x_{f_i}|$ of variables in its scope. A *complete assignment* σ is a value assignment to all variables in X, consistent with their domains, and such that all cost functions are satisfied, i.e. $f_i(\sigma) \neq \{\perp\}$, $\forall f_i \in F$. The objective of a DCOP is to find the complete assignment σ^* that minimizes the sum of cost functions

$$\sigma^* = \arg\min_{\sigma} \sum_{f_i \in F} f_i(\sigma). \tag{13}$$

To fit with our environment, we define the set of variables $X = \{V_1, \ldots, V_n\}$ as the set of voltages, and domains D are defined by constraint (10b). Notice that, within DCOP framework, voltages domains are discrete. This means that the quality of the solution depends on domains' granularity. Then, we can define the duty cycle and the internal current:

$$d(x_i, x_{j\in\mathcal{N}_i}) = \frac{x_i + R_i\left(I_{l,i} - \frac{\sum_{j\in\mathcal{N}_i}(x_j - x_i)}{R_{ij}}\right)}{V_{in,i}},$$

$$I(x_i, x_{j\in\mathcal{N}_i}) = \frac{\sum_{j\in\mathcal{N}_i}(x_j - x_i)}{R_{ij}} - I_{l,i},.$$

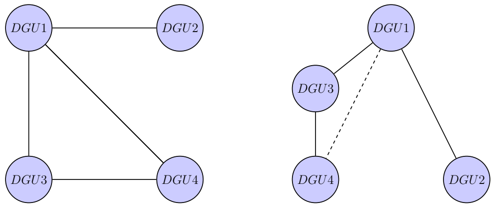
As concerns local cost functions $f_k \in F$, we define:

- *Unary* cost functions to represent term $\mathcal{N}(\boldsymbol{V})$:
$$f_k^{\mathcal{N}}(x_i) = \beta(x_i - x_i^n)^2. \tag{14a}$$
- *Binary* cost functions to represent term $\mathcal{J}(\boldsymbol{V})$:
$$f_k^{\mathcal{J}}(x_i, x_j) = \alpha\frac{(x_j - x_i)^2}{R_{ij}}. \tag{14b}$$
- *k-ary* cost functions to represent $\mathcal{D}(\boldsymbol{V})$, $d(\boldsymbol{V})$ and $I(\boldsymbol{V})$:
$$f_k^{\mathcal{D}}(x_i, x_{j\in\mathcal{N}_i}) = \gamma(d - 1/2)^2 \tag{14c}$$
$$f_k^{d}(x_i, x_{j\in\mathcal{N}_i}) = \begin{cases} 0 & \text{for } 0 \leq d \leq 1 \\ \perp & \text{otherwise} \end{cases} \tag{14d}$$
$$f_k^{I}(x_i, x_{j\in\mathcal{N}_i}) = \begin{cases} 0 & \text{for } I_{min} \leq I \leq I_{max} \\ \perp & \text{otherwise} \end{cases} \tag{14e}$$

Notice that functions $f_k^{\mathcal{N}}$, $f_k^{\mathcal{J}}$ and $f_k^{\mathcal{D}}$ are implemented as soft constraints, while f_k^d and f_k^I as hard constraints. In this way, the centralized problem (10) and the distributed one (13) are equivalent, aside from the domains' discretization introduced by DCOP. In fact, constraint (10b) is represented by the set of domains D, while constraints (10c) and (10d) are represented by soft constraints (14d) and (14e), respectively. Finally, the cost functions in (10) and (13) are identical. The following approximation

$$\min \sum f_k(\cdot) \approx \min \alpha\mathcal{J}(\cdot) + \beta\mathcal{N}(\cdot) + \gamma\mathcal{D}(\cdot) \tag{15}$$

depends on density of domains in D. In general, (15) is the more accurate the more the average cardinality of sets D_i is high. Such a characteristic will be observed in simulation in Section IV. The main difference between problems (10) and (13) is the solution search protocol. In the centralized problem, as already discussed, each node refers to a central entity which is in charge of the optimization procedure. The advantage of DCOP modelling is the existence of efficient *distributed algorithms*, which operate with message exchange protocols among agents. A comprehensive description of DCOP algorithms, including their computational characteristics, comparisons and typical applications is found in [10]. We selected for the following simulations the so called *Distributed Pseudotree Optimization Procedure (DPOP)*, proposed by [11]. This specific algorithm relies on local communications only and guarantees to achieve the optimal solution in a finite time. The pseudocode of *DPOP* is described in detail in Algorithm 1 of [11]. For the sake of completeness, we give here an high-level description of its modus operandi. Firstly, the electrical graph is arranged as



(a) (b)

Fig. 2: Pseudo-tree arrangement (b) of graph (a).

a *pseudo-tree* (it is known that this arrangement is possible for any graph).

Definition 3.1: A pseudo-tree arrangement of a graph $\mathcal{G}$ is a rooted tree with the same vertices as $\mathcal{G}$ and the property that adjacent vertices of the original graph fall in the same branch of the tree.

As an example, the reader may observe the Figure 2b, which is a pseudo-tree arrangement of the graph represented in Figure 2a. Then, each node of the pseudo-tree computes the utility of its neighbourhood by means of local functions (14), and builds a message containing utility data. Messages are thus propagated from the leaves to the root of the tree. The root node is in charge to select the value of its domain that maximises the aggregated utility of all underlying nodes. Finally, down-streams decisions are made by all the other nodes, relying on optimal decision made by their parents.

C. Cluster-based architecture

The working paradigm of the previous two architectures is to force inner loops, through the variation of all the voltage references $\boldsymbol{V^r}$, with a sampling time long enough not to spoil their stability. A critical condition may arise when a sudden load variation occurs in a node, between two consecutive samplings. In such a case, some node may face duty cycle (or current) saturation before the next computation of vector $\boldsymbol{V^r}$. Therefore, the stability property of the inner loops is not guaranteed any more. To counteract this trouble, we introduce a cluster-based architecture, which can be regarded as middle-way between the centralized and the distributed one. The idea behind this method is that the near-saturation node must identify a surrounding cluster which is, as a group, able to adsorb the load variation without affecting the other nodes in the network. This solution, even if clearly sub-optimal with respect to the previous two, has the double advantage to be timely and to only employ resources in a neighbourhood of the load variation. One could conceive to implement a fixed-time sampling method for the optimal operation of the mG, as the ones described in Section III-A and III-B, together with a faster event-triggered cluster-based procedure for the sudden disturbances rejection. In detail, let's start with considering equations (2) in steady-state form. By expliciting, in both equations, the internal current, one obtains

$$I_{l,i} - I_{n,i} = (d_i V_{in,i} - V_i)/R_i. \tag{16}$$

Static equations (16) can be written, with the help of (5), in compact form

$$Kd - \mathcal{Y}\boldsymbol{V} - I_l = \boldsymbol{0}. \tag{17}$$

Matrix $\mathcal{Y} = \mathcal{L}_w + R^{-1}$ is usually addressed to as *nodal admittance matrix*, and $K = \operatorname{diag}(V_{in,1}/R_1, \ldots, V_{in,n}/R_n)$. Since $\mathcal{Y}$ is the sum of a Laplacian and a diagonal matrix with positive entries, it is positive definite.

Assumption 1: Consider a sudden load variation $\Delta I_{l,i}$ in the i^{th} node, such that it does not exists a corresponding Δd_i which satisfies either the duty cycle or the internal current constraint.

We want then to modify the voltage reference values of a certain cluster $C \subseteq \mathcal{V}$ surrounding node i. We can arbitrarily order equations (17) in such a way that the $c = |C|$ nodes of the selected cluster are represented by the first c equations, while the remaining $\bar{c} = n - c$ equations describe the rest of the network. So, equations (17) are partitioned as

$$\left[\begin{array}{c|c} K_c & \boldsymbol{0} \\ \hline \boldsymbol{0} & K_{\bar{c}} \end{array}\right] \left[\begin{array}{c} \Delta d_c \\ \hline \Delta d_{\bar{c}} \end{array}\right] - \left[\begin{array}{c|c} Y_c & Y_b \\ \hline Y_b' & Y_{\bar{c}} \end{array}\right] \left[\begin{array}{c} \Delta V_c \\ \hline \Delta V_{\bar{c}} \end{array}\right] + \\ - \left[\begin{array}{c} \Delta I_{l,c} \\ \hline \Delta I_{l,\bar{c}} \end{array}\right] = \boldsymbol{0} \tag{18}$$

Since we do not want to affect reference values and duty cycles of the rest of the mG, we set $\Delta V_{\bar{c}} = \Delta d_{\bar{c}} = \boldsymbol{0}$. Moreover, since we assume the load variation to takes place in a node in C, also $\Delta I_{l,\bar{c}} = \boldsymbol{0}$. Finally, expression (18) becomes

$$\underbrace{\left[\begin{array}{c|c} -Y_c & K_c \\ \hline -Y_b' & \boldsymbol{0} \end{array}\right]}_{A} \underbrace{\left[\begin{array}{c} \Delta V_c \\ \hline \Delta d_c \end{array}\right]}_{x} = \underbrace{\left[\begin{array}{c} \Delta I_{l,c} \\ \hline \boldsymbol{0} \end{array}\right]}_{b}. \tag{19}$$

Constraint (19) guarantees both steady-state load compensation and outer nodes independence. Additionally, we must guarantee cluster feasibility by enforcing local constraints on voltages, duty cycles and internal currents. Let's start with defining the duty cycle and the internal current, stressing out the separation between C and $\bar{C}$

$$\begin{bmatrix} d_c \\ d_{\bar{c}} \end{bmatrix} = K^{-1}(\mathcal{Y}\boldsymbol{V} + I_l) = \begin{bmatrix} K_c^{-1}(Y_c V_c + Y_b V_{\bar{c}} + I_{l,c}) \\ K_{\bar{c}}^{-1}(Y_b' V_c + Y_{\bar{c}} V_{\bar{c}} + I_{l,\bar{c}}) \end{bmatrix},$$

$$\begin{bmatrix} I_c \\ I_{\bar{c}} \end{bmatrix} = \mathcal{L}_w \boldsymbol{V} + I_l = \begin{bmatrix} \mathcal{L}_c V_c + \mathcal{L}_b V_{\bar{c}} + I_{l,c} \\ \mathcal{L}_b' V_c + \mathcal{L}_{\bar{c}} V_{\bar{c}} + I_{l,\bar{c}} \end{bmatrix}.$$

Then, the following cluster-level optimization problem is introduced.

$$\min_{\Delta V_c, \Delta d_c, \epsilon} \quad \alpha x' x + \beta \Delta V_c' \mathcal{L}_c \Delta V_c + \gamma \epsilon \tag{20a}$$

$$\text{s. t.} \quad Ax = b - \tilde{\epsilon} \tag{20b}$$

$$\underline{\Delta V}_c \leq \Delta V_c \leq \overline{\Delta V}_c \tag{20c}$$

$$\underline{\Delta I}_c \leq \mathcal{L}_c \Delta V_c + b \leq \overline{\Delta I}_c \tag{20d}$$

$$\underline{\Delta d}_c \leq \Delta d_c \leq \overline{\Delta d}_c \tag{20e}$$

$$\alpha, \beta, \gamma, \epsilon > 0, \tag{20f}$$

where α, β and γ are positive weights and $\tilde{\epsilon} = \begin{bmatrix} \epsilon & \boldsymbol{0} \end{bmatrix}'$. Cost function (20a) is made up of three terms: the first one represents the variation of voltages and duty cycles, the second the variation of Joule losses and the third is a *slack* variable. Constraint (20b) is the structural relation (19) with the addition of the slack. In this way, we know that a feasible solution is found when the slack variable assumes value $\epsilon = 0$. Constraints (20c), (20d) and (20e) enforce the usual limits on voltages, internal currents and duty cycles, respectively.

Proposition 3.1: A necessary condition for the solution of problem (20) is that $\text{rank}(Y_b) < c$.

Proof: The system of linear equations (19) always has a solution. In fact, by invoking the Rouché-Capelli theorem, we can state that the system (19) has a solution if and only if $\text{rank}(A) = \text{rank}(A|b)$. This condition is always verified, because vector b is linearly dependent from one of the columns of $\left[\begin{array}{cc} K_c & \mathbf{0} \end{array}\right]'$. Then, notice that $\text{rank}(-Y_c|K_c) = c$. Moreover, Y_b' has c columns, and so $\left[\begin{array}{cc} -Y_b' & \mathbf{0} \end{array}\right]$ can present, at most, c independent rows. It follows that $\text{rank}(Y_b) \leq c$. In particular, when $\text{rank}(Y_b) = c$, we can find an explicit solution by eliminating zero or dependent rows from $\left[\begin{array}{cc} -Y_b' & \mathbf{0} \end{array}\right]$, obtaining the equivalent reduced system $\hat{A}x = \hat{b}$. Since in this case $\hat{A}$ is square and full rank, we have that

$$x = \hat{A}^{-1}\hat{b} = \left[\begin{array}{c|c} Q & U \\ \hline W & Z \end{array}\right] \left[\begin{array}{c} \Delta I_{l,c} \\ \hline \mathbf{0} \end{array}\right] = \left[\begin{array}{c} Q \\ \hline W \end{array}\right] \Delta I_{l,c} \quad (21)$$

Thanks to the properties of block matrix inversion, we have that

$$\begin{aligned} Q &= Y_c^{-1} - Y_c^{-1}K_c(Y_bY_c^{-1}K_c)^{-1}Y_bY_c^{-1} = \mathbf{0}, \\ W &= (Y_bY_c^{-1}K_c)^{-1}Y_bY_c^{-1} = K_c^{-1}. \end{aligned}$$

Then,

$$\left[\begin{array}{c} \Delta V_c \\ \hline \Delta d_c \end{array}\right] = \left[\begin{array}{c} \mathbf{0} \\ \hline K_c^{-1}\Delta I_{l,c} \end{array}\right]. \quad (22)$$

So the load variation in the i^{th} node is just compensated with the duty cycle of the node i itself

$$\Delta d_{c,i} = \frac{R_i}{V_{in,i}}\Delta I_{l,i}, \quad (23)$$

while all other voltages and duty cycles variations are maintained at zero. This solution is in contrast with Assumption 1. Therefore, we have to guarantee that $\text{rank}(Y_b) < c$. ■

Remark 1: Matrix Y_b describes how cluster C is connected with the rest of the network. In particular, $\text{rank}(Y_b)$ expresses the number of "independent neighbours" of C, that is the number of nodes of $\bar{C}$ that are independently connected to C. In this light, Proposition 3.1 provides a structural condition for the solution of problem (20). In fact, when $\text{rank}(Y_b) < c$, there exists at least one node in C that is able to modify its state without affecting external nodes.

At this point, we need to introduce a protocol that permits the faulty node i to identify a suitable surrounding cluster, in order to enforce the cluster-level optimization problem (20). We propose a procedure in which the faulty node ask to its neighbours for the necessary data to solve the optimization problem. As described with pseudo-code in Algorithm 1, the protocol provides for two kind of messages that nodes can exchange. The first one is the HELP message, i.e. a request of help to satisfy load demand. The second one is the DATA message, which is sent as a response to a HELP message, which actually contains necessary data to define the constraints (20b)-(20e), e.g. DGUs actual duty and current bounds, inductance and resistance values. Then, the faulty node solves problem (20). If the slack variable ϵ is greater then zero, that is the current cluster is not able to adsorb the disturbance, node i will send HELP messages to its longer distance neighbours, solving enlarged optimization problems until a feasible solution is found ($\epsilon = 0$).

Algorithm 1 Cluster control

```
 1: procedure ClusterControl
 2:     $h = 1$
 3:     node $i$ sends HELP messages to $h$-step neighbors $j$
 4:     nodes $j$ send DATA messages to node $i$
 5:     node $i$ solves optimization problem (20)
 6:     if $\epsilon = 0$ then return $\Delta V_c$
 7:     else
 8:         $h = h + 1$
 9:         goto 3
10:     end if
11: end procedure
```

IV. Simulations

The aim of this section is twofold. Firstly, we want to compare the centralized and distributed algorithms presented in Sections III-A and III-B, respectively. Secondly, we show a situation in which the cluster-based technique is triggered by a sudden load variation in a node.

1) Centralized VS distributed: In order to fairly compare the two fixed-time methods, we implement them to the same mG configuration, with topology as in Figure 3 and electrical parameters as in Table I. In Table II it is shown that the distributed cost function tends to the centralized one as the number of points in the variables domains increases. That is, approximation (15) is the more accurate the more domains are dense. On the other hand, Table II displays that the distributed algorithm requires a power-wise computation time to be solved, that increases with the domains points. Therefore, the choice of domain density is a trade-off between optimality of the cost function and computation time. We want to stress that, even with a domain of 17 points, the distributed algorithm cost function is very close to the centralized one, as well as the computation time.

2) Cluster-based regulation: In Figure 4a it is represented a step-wise disturbance current load for node 1. The loads of all other nodes are assumed to be constant for the whole duration of the experiment. The vertical dashed lines correspond to the time instants when the fixed-time algorithm, either centralized or distributed, recomputes the optimal voltage reference values. In Figures 4b, 4c and 4d, the electrical behaviour of node 1 is described by dotted lines when only the centralized algorithm is used. In continuous lines we depict its behaviour when the centralized is operated together with the cluster-based algorithm. A disturbance peak, which is not perceived by the fixed-time algorithm, occurs between 10 and 15 minutes. Node 1 cannot handle

Parameter	Symbol	Value
Input voltage	V_{in}	$23 \div 25\ V$
Load	I	$1.5 \div 2.5\ A$
Converter capacitance	C	$0.9 \div 1.1\ mF$
Converter inductance	L	$45 \div 55\ mH$
Converter resistance	R	$0.1 \div 0.3\ \Omega$
Switching frequency	f	$1\ kHz$
Line resistance	R_L	$0.5 \div 1.5\ \Omega$

TABLE I: MG electrical parameters, where $a \div b$ expresses randomly generated values between a and b.

	Distr.					Centr.
Domain points	5	9	17	33	65	∞
Cost function	0.801	0.715	0.576	0.551	0.548	0.547
Time (s)	0.08	0.41	2.42	27.65	399.98	1.58

TABLE II: Performance comparison between distributed and centralized algorithm.

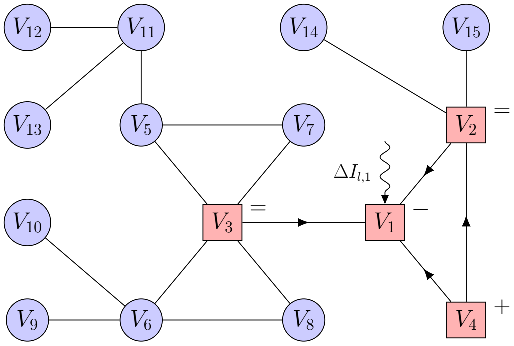


Fig. 3: mG topology and identified cluster (red-squares)

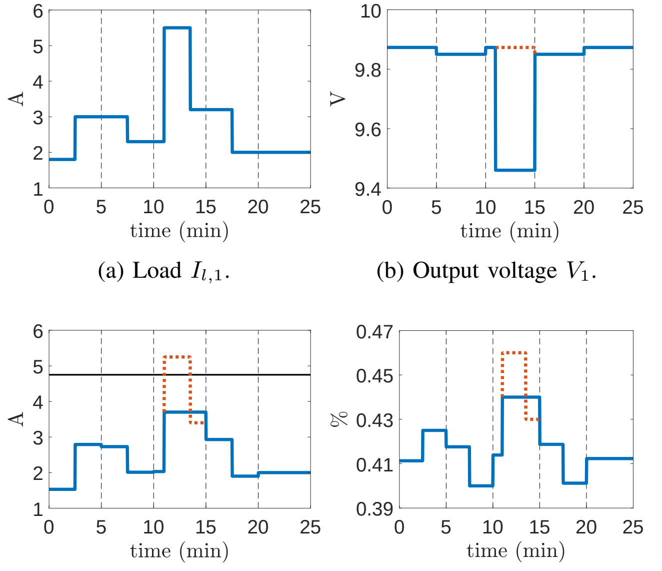
it internally due to the constraints on internal current, as it can be observed in Figure 4c. As a consequence, Algorithm 1 is triggered. By referring to Figure 3, node 1 sends HELP messages to its neighbours 2, 3 and 4, and they replies with DATA messages containing necessary data. At this point, node 1 executes problem (20) and finds the following solution that satisfies the constraints:

$$\Delta V_1 = -0.42\,V,\ \ \Delta V_2 = 0\,V,\ \ \Delta V_3 = 0\,V,\ \ \Delta V_4 = 0.12\,V.$$

Hence, after the cluster-based method, node 1 reduces its reference value, while its neighbours increase it. This results in a raise of flows towards node 1 (see Figure 3), that contributes counteracting the disturbance. Note also that, in order to enforce a flows redistribution inside the cluster which does not affect external nodes, the reference of nodes 2 and 3 is maintained constant by the algorithm.

V. Conclusion

In this paper, we focused on developing secondary control strategies for DC microgrids operation. A centralized and a distributed method were introduced, implemented in a fixed-time setting, as well as a cluster-based optimization strategy. The authors believe significant improvements still could be reached. Firstly, an analysis of other existing DCOP algorithms is advisable (see e.g. [12]), in order to compare their efficacy with the one we implemented in this work. Furthermore, finer cluster selection strategies may be explored, for instance exploiting the mG topology to suitably identify the best clusters to absorb the load demand.



(a) Load $I_{l,1}$.

(b) Output voltage V_1.

(c) Internal current I_1.

(d) Duty cycle d_1.

Fig. 4: Current, voltage and duty cycle trends in node 1. Dashed line: centralized algorithm only; continuous line: centralized + cluster-based algorithm.

References

[1] A. La Bella, S. R. Cominesi, C. Sandroni, and R. Scattolini, "Hierarchical predictive control of microgrids in islanded operation." *IEEE Trans. Automation Science and Engineering*, vol. 14, no. 2, pp. 536–546, 2017.

[2] J. M. Guerrero, J. C. Vasquez, J. Matas, D. Vicuna, L. Garcia, and M. Castilla, "Hierarchical control of droop-controlled AC and DC microgrids - a general approach toward standardization," *IEEE Transactions on Industrial Electronics*, vol. 58, pp. 158–172, 2011.

[3] A. Martinelli, P. Nahata, and G. Ferrari-Trecate, "Voltage stabilization in MVDC microgrids using passivity-based nonlinear control," in *Proceedings of the 57th IEEE Conference on Decision and Control*, 2018, pp. 7022–7027.

[4] M. Cucuzzella, S. Trip, C. D. Persis, X. Cheng, A. Ferrara, and A. V. der Schaft, "A robust consensus algorithm for current sharing and voltage regulation in DC microgrids," *IEEE Transactions on Control Systems Technology*, 2018, accepted for publication.

[5] M. Tucci, L. Meng, J. M. Guerrero, and G. Ferrari-Trecate, "Plug-and-play control and consensus algorithms for current sharing in DC microgrids," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12 440 – 12 445, 2017, 20th IFAC World Congress.

[6] P. Nahata, R. Soloperto, A. Martinelli, and G. Ferrari-Trecate, "A passivity-based approach to voltage stabilization in DC microgrids with ZIP loads," *Automatica*, 2018, submitted. Available online at https://infoscience.epfl.ch/record/253266.

[7] M. Tucci, S. Riverso, and G. Ferrari-Trecate, "Line-independent plug-and-play controllers for voltage stabilization in DC microgrids," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 3, pp. 1115–1123, 2018.

[8] R. D. Middlebrook and S. Cuk, "A general unified approach to modelling switching-converter power stages," in *1976 IEEE Power Electronics Specialists Conference*, 1976, pp. 18–34.

[9] M. Cucuzzella, S. Rosti, A. Cavallo, and A. Ferrara, "Decentralized sliding mode voltage control in DC microgrids," in *Proceedings of 2017 American Control Conference*, 2017, pp. 3445–3550.

[10] F. Fioretto, E. Pontelli, and W. Yeoh, "Distributed constraint optimization problems and applications: A survey," *Journal of Artificial Intelligence Research*, vol. 61, pp. 623 – 628, 2018.

[11] A. Petcu and B. Faltings, "A scalable method for multiagent constraint optimization," in *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, 2005, pp. 266–271.

[12] P. J. Modi, W. Shen, M. Tambe, and M. Yokoo, "ADOPT: asynchronous distributed constraint optimization with quality guarantees," *Artificial Intelligence*, vol. 161, no. 1, pp. 149–180, 2005.