

A two-layer stochastic Model Predictive Control scheme for microgrids

S. Raimondi Cominesi, M. Farina, L. Giulioni, B. Picasso, R. Scattolini

Abstract—A two-layer control scheme based on Model Predictive Control (MPC) operating at two different timescales is proposed for the energy management of a grid-connected microgrid (MG), including a battery, a microturbine, a photovoltaic system, a partially non predictable load, and the input from the electrical network. The high-level optimizer runs at a slow timescale, relies on a simplified model of the system, and is in charge of computing the nominal operating conditions for each MG component over a long time horizon, typically one day, with sampling period of 15 minutes, so as to optimize an economic performance index on the basis of available predictions for the PV generation and load request. The low-level controller runs at higher frequency, typically 1 minute, relies on a stochastic MPC (sMPC) algorithm, and adjusts the MG operation to minimize the difference, over each interval of 15 minutes, between the planned energy exchange and the real one, so avoiding penalties. The sMPC method is implemented according to a shrinking horizon strategy and ensures probabilistic constraints satisfaction. Detailed models and simulations of the overall control system are presented.

Index Terms—Distributed generation, hierarchical control, stochastic model predictive control, uncertainty.

I. INTRODUCTION

THE optimal management of energy in grid-connected micro-grids (MG) is of paramount importance due to the widespread and ever increasing diffusion of MG in energy production systems, see [1], [2], [3], [4] and references therein. However, the efficient use of MG is difficult due to the presence of distributed energy resources, which can be controllable (or dispatchable), such as gas turbines or Diesel engines, but also non dispatchable, such as wind turbines or photovoltaic (PV) generators, whose production heavily depends on the weather conditions, which in turn can be forecasted with some uncertainty over long time periods, typically 24 hours. In addition, also the loads suffer from the same problem: while some loads can be easily predicted, other critical loads can have a significant stochastic component to be accounted for. To mitigate these uncertainties, one can import/export power from/to the utility grid or include in the MG storage devices, like batteries, which have to be properly managed as well.

The planned use of the MG elements over the prediction horizon of 24 hours, usually with sampling period of 15 minutes, can be computed as the solution of an optimization problem, based on the Model Predictive Control (MPC) approach, where an economic performance index is minimized

and the available deterministic forecasts of the renewables power production and of the loads are considered. By resorting to Mixed Logical Dynamical (MLD) models [5], also logical constraints can be included in the optimization problem to take into account the operating conditions of the elements of the grid, such as latency times or charge/discharge constraints. Based on this initial plan, computed at the beginning of each day, a reference profile, sampled every 15 minutes, of the power and energy exchanges with the national power grid for the whole day ahead is agreed with the network manager and should be strictly followed to avoid penalties and additional costs. The effectiveness of this approach has already been tested both in simulation and in real benchmarks [3], [4], [6], [7], [8].

Although this deterministic solution is a significant aid in the planning of the MG resources over the day, the inherent uncertainty of the forecast of the renewables production and critical loads can lead to a large discrepancy between the energy really exchanged with the utility grid and the planned one. For this reason, several stochastic MPC (sMPC) algorithms have been presented, see e.g. [9], [10], [11], [12], [13], [14], to account for the uncertainty of non-dispatchable generation and loads. The proposed solutions usually rely on a two stage stochastic programming approach: a deterministic MPC problem is first solved, typically at a rate of 15 min, to plan the future use of the microgrid energy resources based on nominal profiles of loads and distributed generation. Then, stochastic MPC is applied at the same frequency (15 min) and with the same prediction horizon to compensate for unpredictable fluctuations of loads and renewable generators. In this paper, we propose a different approach based on the two-layer structure shown in Figure 1. At the higher layer, and similarly to the solutions described in [3], [4], [6], deterministic MPC is used to plan the use of the MG elements over the prediction horizon of 24 hours and with a sampling rate of 15 minutes. At the lower layer, a sMPC regulator runs at higher frequency, typically 1 minute, with the scope of compensating for the uncertainties and maintaining, over each interval of 15 minutes, the total energy exchange with the network as close as possible to the planned value. In this framework, according to the sMPC algorithm described in [15], [16], probabilistic constraints on the manipulated and/or controlled variables are considered. A shrinking-horizon implementation is used to set the terminal conditions of the MG elements, at the end of each period of 15 minutes, to the values assigned by the high-level regulator. The proposed approach is also capable of replanning at the higher layer the use of the MG elements when the constraints imposed at the lower

The authors are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milan 20133, Italy (e-mail: stefano.raimondi@polimi.it; marcello.farina@polimi.it; luca.giulioni@polimi.it; bruno.picasso@polimi.it; riccardo.scattolini@polimi.it).

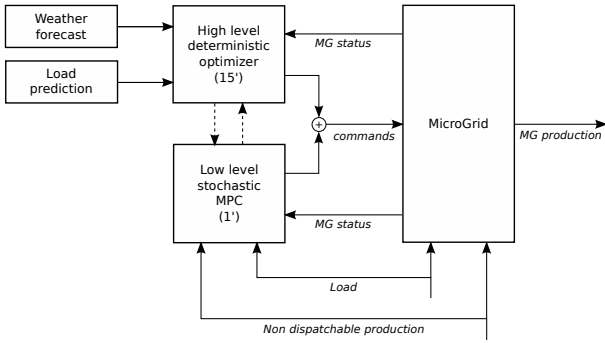


Fig. 1. Control scheme architecture.

layer cannot be fulfilled due to excessive deviations of the uncertain variables with respect to their initial nominal values.

A preliminary version of the the two-layer control scheme developed in the paper has been proposed in [17], where however, the models of the MG and their components were greatly simplified as well as the fundamental constraint on the energy exchange over the 15 minutes sampling time. The method we propose is of general validity for MG connected to the national grid and can be easily adapted to cope with different MG configurations. However, to present in detail the main steps of the design procedure, reference is made to a specific MG structure, described in Section II. The design of the high level layer is reported in Section III-B, while the core of the proposed approach is described in Section IV, where the adopted sMPC algorithm is presented. The results of simulation experiments are summarized in Section V, where different control strategies for the update of the high-level planning are discussed and commented. Finally, some conclusions are drawn in Section VI.

II. THE MICRO-GRID

The basic elements of a MG are dispatchable generators, such as microturbines or Diesel engines, non dispatchable generators, such as photovoltaic panels or wind turbines, energy storage systems, such as batteries or flywheels, controllable and noncontrollable (critical) loads, and thermal systems. The control design method proposed in this paper is of general validity and can be used for any MG configuration, however, in order to more clearly describe its main features, in the following we will consider a MG, consistent with the test facility (TF) available at R.S.E. S.p.A. and equipped with all the elements typical of microgrids. Specifically, the MG is equipped with a gas microturbine (T), a battery (B), a photovoltaic panel (P), a critical load (L), and the connection to the national power grid (N).

A. General description of the MG

The dispatchable units T and B are endowed with embedded controllers. Thus their setpoints u_T and u_B are control variables for the two-layer control structure here considered. The output variables are the actual generated power signals

y_T, y_B .

The photovoltaic panels are supposed not to be adjustable so that the totality of its produced power y_P flows through the MG. The loads are not controllable as well: let y_L be the electrical power to be supplied by the MG system. Daily predictions of both PV power production and load consumption are assumed to be available. Both variables are, however, affected by uncertainties.

The connection to N is bidirectional and allows for both purchase and sale of electric power. The net power exchanged between the MG and N (positive if power is absorbed) is denoted by y_N . The purchase cost and the sale price are known but not constant, being subject to market fluctuations.

B. Model of the MG

In this Section the dynamic models of the devices and the stochastic description of the uncertainties are illustrated. The microturbine and the battery are continuous-time systems. However, since the two-level sMPC algorithm described in the following sections runs at different sampling times for the low and the high layers, their models must be discretized with different sampling periods, namely $\tau_{\ell\ell} = 1$ min for the lower layer and $\tau_{H\ell} = 15$ min for the higher one. Correspondingly, two different discrete-time indices must be used: the one related to the fast time scale will be denoted by h , while the one related to the slow time scale will be denoted by k . The microturbine behavior, together with the one of its embedded controller, is based on a finite-state automata composed of four states, namely: *starting*, *running*, *stopping* and *stopped*. When the system is in the *running* mode, experimental tests carried out on T allowed us to identify the relationship between machine set point u_T and actual output y_T in terms of the following continuous-time transfer function:

$$G_T(s) = \frac{(1 + 11.25s)}{(1 + 6.41s)(1 + 1.36s)} e^{-26.4s}. \quad (1)$$

According to a standard discretization procedure with sampling period $\tau_{\ell\ell}$, see [18], the third order discrete-time realization of the continuous-time model (1) takes the form

$$\Sigma_T : \begin{cases} x_T(h+1) = A_T x_T(h) + B_T u_T(h) \\ y_T(h) = C_T x_T(h), \end{cases}$$

where, consistently with (1)

$$C_T(I - A_T)^{-1} B_T = 1. \quad (2)$$

The battery can be essentially modeled as a continuous-time integrator with internal state x_B representing its normalized state of charge (SOC) and with different charge η_{CH} and discharge η_{DCH} efficiencies. In order to model it in the short time scale, it is useful to define the discrete-time inputs

$$u_B(h) = \begin{bmatrix} u_B^{CH}(h) \\ u_B^{DCH}(h) \end{bmatrix}, \quad (3)$$

where $u_B^{CH}(h) \geq 0$ and $u_B^{DCH}(h) \geq 0$ represent the absolute value of the power absorbed and released, respectively, from the battery in a time period of one minute. The asymmetry of η_{CH}, η_{DCH} leads to a number of problems and subtleties in

their management which will be deeply analyzed in the sequel. In the fast discrete-time scale, the SOC dynamics is then described by the following system.

$$\Sigma_B : \begin{cases} x_B(h+1) = x_B(h) + B_B^{\text{CH}} u_B^{\text{CH}}(h) - B_B^{\text{DCH}} u_B^{\text{DCH}}(h) \\ y_B(h) = u_B^{\text{DCH}}(h) - u_B^{\text{CH}}(h). \end{cases}$$

In matrix form, recalling (3), $\Sigma_B = (A_B, B_B, C_B, D_B) = (1, B_B, 0, D_B)$ where

$$B_B = [B_B^{\text{CH}} \quad -B_B^{\text{DCH}}] = \frac{\tau_{\ell\ell}}{C_B^{\text{max}}} [\eta_{\text{CH}} \quad -\eta_{\text{DCH}}] \quad \text{and} \\ D_B = [-1 \quad 1].$$

where C_B^{max} is the nominal capacity of the battery. For consistency it must be verified that, for all $h \geq 0$

$$u_B^{\text{CH}}(h) \cdot u_B^{\text{DCH}}(h) = 0. \quad (4)$$

Photovoltaic production forecast is a problem that has been amply discussed in the literature. The most prominent approaches consist in the combination of the data available from any suitable technique of solar forecasting (*e.g.* mathematical models simulation, total sky imagery, satellite cloud motion vector, etc.) together with a characterization of the photovoltaic panels, in order to obtain an estimate of its contribution to the power balance. The reader might refer to [19] for a more detailed description of the problem. In this way it is possible, at the beginning of each day, to derive the predicted PV nominal daily production profile $y_P^*(h)$, $h = 0, \dots, 1439$ ¹. The real production $y_P(h)$ is given by

$$y_P(h) = y_P^*(h) + \hat{y}_P(h),$$

where $\hat{y}_P(h)$ is a stochastic process described by

$$\Sigma_P : \begin{cases} \hat{x}_P(h+1) = A_P \hat{x}_P(h) + B_P v_P(h) \\ \hat{y}_P(h) = C_P \hat{x}_P(h) + D_P v_P(h) \end{cases}$$

and v_P is a white noise whose characteristics have been estimated from the available historical time series.

The model of the real load profiles is obtained similarly. Letting $y_L^*(h)$, $h = 0, \dots, 1439$, be the nominal load profile, we have that

$$y_L(h) = y_L^*(h) + \hat{y}_L(h),$$

where

$$\Sigma_L : \begin{cases} \hat{x}_L(h+1) = A_L \hat{x}_L(h) + B_L v_L(h) \\ \hat{y}_L(h) = C_L \hat{x}_L(h) + D_L v_L(h) \end{cases}$$

and v_L is a white noise whose characteristics have been estimated from the available historical time series.

Finally note that, in this work, we rely on the following standing assumption.

Assumption 1: The load is constantly balanced by the network, namely, for all $h \geq 0$

$$y_L(h) = y_T(h) + y_B(h) + y_P(h) + y_N(h). \quad (5)$$

This is achieved either by drawing power from, or by selling the overproduced power to, the national grid N.

¹For simplicity, $h = 0$ denotes the time 00:00, while $h = 1439$ corresponds to time 23:59.

III. HIGH-LEVEL MODEL AND REFERENCE PLANNING

Based on the nominal predictions of both PV power production \bar{y}_P and load consumption \bar{y}_L , at the beginning of each day, the high-level control unit solves an optimization problem with the aim to set, for the future 24 hours and with sampling period τ_{He} , the nominal operation of T (power request, start and stop commands) and of B so as to guarantee the load balance while minimizing the operating costs of the MG.

A. Models for high-level optimization

Simplified and static models (where possible), including logic variables, of the available devices are used by the high-level optimizer. The modeling approach is quite standard and is summarized in this section. For consistency of notation, the nominal value of a generic variable y is denoted by \bar{y} . The discrete-time index for the high-level models is k .

1) *High-level model of the turbine:* The settling time of the model (1) of T is approximately 1 minute and motivates the adoption of a zero order model for the high-level problem. Therefore, in the *running* mode, consistently with (2)

$$\bar{y}_T(k) = \bar{u}_T(k). \quad (6)$$

The setpoint and, consequently, the steady-state power output of T, are bounded between the values $u_T^{\text{min}} = 50$ kW and $u_T^{\text{max}} = 100$ kW.

Taking the consumption of the auxiliary actuators into account (*i.e.*, the fuel pump and the cooling system), the net contribution of the microturbine to the microgrid power balance is negligible during the transient phases constituted either by the *starting* or by the *stopping* state. The *stopping* operation lasts an amount of time comparable with τ_{He} , while the duration of the *starting* phase changes according to the current machine's status. In particular, depending on whether it is cold or hot, the time required for the start-up (*latency time*) takes value t_{cold} or t_{hot} , respectively. We define by \bar{t} the time taken by T to become cold when stopped. Furthermore, to avoid damages and improper use of the device, the turbine must be in operation, once started, at least for a given *minimum operation time* t_{delay} .

The input of the model are the boolean on/off signal $\delta_T(k)$ and the power setpoint $\bar{u}_T(k)$. In order to model the logical states of T, two integer variables, namely $t_{\text{on}}(k)$, and $t_{\text{off}}(k)$, are introduced. In particular, variable t_{on} denotes the number of further time steps required to complete the latency time from the start signal, while t_{off} denotes the time steps spent since the last switch off. Consistently with (6), when T is on (*i.e.*, if $\delta_T(k) = 1$ and $t_{\text{on}}(k) \leq 0$), then $\bar{y}_T(k) = \bar{u}_T(k)$; on the contrary, if $\delta_T(k) = 0$ (*i.e.*, no switch-on signal has been delivered) or $t_{\text{on}}(k) > 0$ (*i.e.*, T is in the latency state), then $\bar{y}_T(k) = 0$.

At each time step, the auxiliary variables are updated as follows:

$$t_{\text{on}}(k+1) = \begin{cases} t_{\text{on}}(k) - 1 & \text{if } \delta_T(k) = 1 \\ t_{\text{hot}} & \text{if } (\delta_T(k) = 0) \wedge (t_{\text{off}}(k) < \bar{t}) \\ t_{\text{cold}} & \text{if } (\delta_T(k) = 0) \wedge (t_{\text{off}}(k) \geq \bar{t}), \end{cases}$$

$$t_{\text{off}}(k+1) = \begin{cases} t_{\text{off}}(k) + 1 & \text{if } \delta_T(k) = 0 \\ 0 & \text{if } \delta_T(k) = 1. \end{cases}$$

Finally, if $-t_{\text{delay}} \leq t_{\text{on}}(k) \leq 0$, then $\delta_T(k) = 1$.

2) *High-level model of the battery*: Denote with \bar{x}_B the normalized nominal SOC of B and

$$\bar{u}_B(k) = \begin{bmatrix} \bar{u}_B^{\text{CH}}(k) \\ \bar{u}_B^{\text{DCH}}(k) \end{bmatrix},$$

where $\bar{u}_B^{\text{CH}}(k) \geq 0$ and $\bar{u}_B^{\text{DCH}}(k) \geq 0$ represent the absolute value of the nominal power absorbed and released, respectively, from the battery. The nominal model of the battery is

$$\begin{cases} \bar{x}_B(k+1) = \bar{x}_B(k) + \frac{\tau_{\text{He}}}{C_B^{\text{max}}} (\eta_{\text{CH}} \bar{u}_B^{\text{CH}}(k) - \eta_{\text{DCH}} \bar{u}_B^{\text{DCH}}(k)) \\ \bar{y}_B(k) = \bar{u}_B^{\text{DCH}}(k) - \bar{u}_B^{\text{CH}}(k), \end{cases} \quad (7)$$

where, in the available TF, $C_B^{\text{max}} = 70$ kWh. The output $\bar{y}_B(k)$ is the power exchange between the battery and the micro-grid (if $\bar{y}_B < 0$, then the battery is absorbing power - hence, it is in the charging mode - whereas $\bar{y}_B > 0$ in case of power release - thus, the battery is in the discharging mode).

The following constraint, imposing that the battery is either absorbing or releasing power, has to be considered.

$$\bar{u}_B^{\text{CH}}(k) \cdot \bar{u}_B^{\text{DCH}}(k) = 0. \quad (8)$$

Notice that, under condition (8), model (7) is equivalent to a piecewise linear system. The charge and discharge power setpoints are upper bounded as well, so that, for all $k \in \mathbb{N}$, the constraints

$$\begin{cases} 0 \leq \bar{u}_B^{\text{CH}}(k) \leq u_B^{\text{max}} \\ 0 \leq \bar{u}_B^{\text{DCH}}(k) \leq u_B^{\text{max}} \end{cases}$$

are enforced. In the considered test facility, $u_B^{\text{max}} = 70$ kW. Finally, in order to reduce the risk of damaging the storage unit, the state of charge is bounded between $\bar{x}_B^{\text{min}}(k) = 0.15$ and $\bar{x}_B^{\text{max}}(k) = 0.9$.

3) *High-level model of the photovoltaic panels*: As it has been mentioned in the Introduction, the high-level optimizer is fed with the nominal PV production signal $\bar{y}_P(k)$, $k = 0, \dots, 95$, for the day ahead (with high-level sampling time), defined as

$$\bar{y}_P(k) = \frac{\sum_{t=0}^{n-1} y_P^*(nk+t)}{n}. \quad (9)$$

where $n = 15$. We also define $d_P(h) = y_P^*(h) - \bar{y}_P(\lfloor h/n \rfloor)$, so that, in view of (9), $\sum_{t=0}^{n-1} d_P(\lfloor h/n \rfloor \cdot n + t) = 0$.

4) *High-level model of the load*: The nominal load profile needed for the high-level optimization problem is denoted by $\bar{y}_L(k)$ and, similarly to Section III-A3, it is defined as

$$\bar{y}_L(k) = \frac{\sum_{t=0}^{n-1} y_L^*(nk+t)}{n}. \quad (10)$$

Also, we define, for all $h \geq 0$, $d_L(h) = y_L^*(h) - \bar{y}_L(\lfloor h/n \rfloor)$.

5) *High-level model of the distribution network*: We define the variable $\bar{y}_N(k)$ as the nominal power exchange with the network over the k -th sampling interval. Consistently with Assumption 1, the following load balance equation holds for the variables of the high-level models:

$$\bar{y}_L(k) = \bar{y}_T(k) + \bar{y}_B(k) + \bar{y}_F(k) + \bar{y}_N(k). \quad (11)$$

6) *High-level model of the MG*: The previously described models and the logic/algebraic relations between the relevant variables, can be cast as a mixed logical dynamical (MLD) system [5]. Such reformulation is based on the definition of a set of auxiliary variables (either continuous or binary) included in a vector w_{MLD} , which allow for converting all logic relations to a set of linear inequalities with binary variables. This is done by resorting to the freely available software tool HYSDEL [20]. Thus, letting

$$x_{\text{MLD}}(k) = \begin{bmatrix} t_{\text{on}}(k) \\ t_{\text{off}}(k) \\ \bar{x}_B(k) \end{bmatrix} \quad \text{and} \quad u_{\text{MLD}}(k) = \begin{bmatrix} \delta_T(k) \\ \bar{u}_T(k) \\ \bar{u}_B^{\text{CH}}(k) \\ \bar{u}_B^{\text{DCH}}(k) \end{bmatrix}$$

be the state and input vectors, respectively, for the overall system and

$$y_{\text{MLD}}(k) = \bar{y}_N(k),$$

the MLD model used for the high-level optimization takes the form

$$\begin{cases} x_{\text{MLD}}(k+1) = \bar{A}x_{\text{MLD}}(k) + \bar{B}_u u_{\text{MLD}}(k) + \bar{B}_{\text{aux}} w_{\text{MLD}}(k) \\ y_{\text{MLD}}(k) = \bar{C}x_{\text{MLD}}(k) + \bar{D}_u u_{\text{MLD}}(k) + \bar{D}_{\text{aux}} w_{\text{MLD}}(k) \\ \bar{E}_{\text{aff}} \geq \bar{E}_x x_{\text{MLD}}(k) + \bar{E}_u u_{\text{MLD}}(k) + \bar{E}_{\text{aux}} w_{\text{MLD}}(k) \end{cases} \quad (12)$$

Notice that the output variable \bar{y}_N , which is defined by equation (11), is the only interaction element between the MG components that are otherwise not dynamically coupled.

B. High-level optimization

The aim of the upper layer is to optimize the cost of the electric production during the day, while satisfying the nominal load request profile, and subject to the operational constraints on the devices.

At the beginning of the day, the high-level optimizer returns the *initial* operation plan of each MG element for the whole day ahead. The initial planned energy exchange with the utility grid, which is shared with the network manager, is called *reference* plan and denoted by

$$\bar{y}_N^{\text{ref}}(k), \quad k = 0, 1, \dots, 95.$$

The corresponding high-level setpoints, though, could not be always tracked by the low-level controllers due to the uncertainties affecting the critical load and the PV production. If this occurs, the high-level controller should be able to update its plan accordingly in an event triggered fashion. More specifically, the high-level optimizer has the possibility to *update* its overall plan during the day but, in order to avoid penalties from the network manager, the difference between the component \bar{y}_N of its new issue and the reference plan \bar{y}_N^{ref} should be limited. As we shall see in Section III-B2, to account for this further constraint, the event-triggered updating high-level optimization takes a different form from the initial one.

1) *Initial high-level optimization*: The cost function to be minimized accounts for the turbine operational cost (denoted by c_T), the start-up cost (c_T^{st}), the cost of the energy absorbed by the network (c_N^{a}), and the price of the sold electric power (c_N^{s}). As for the battery, a fictitious cost (c_B^{v}) has been introduced on the absolute value of the setpoint variation in

order to preserve the life expectancy of the device. The cost function is hence defined by

$$J_{H\ell}^{\text{ref}} = \sum_{k=0}^{95} c_T(k) \bar{y}_T(k) + c_T^{\text{st}}(k) \max \{ \delta_T(k) - \delta_T(k-1), 0 \} + c_N^{\text{a}}(k) \max \{ \bar{y}_N(k), 0 \} + c_N^{\text{s}}(k) \min \{ \bar{y}_N(k), 0 \} + c_B^{\text{v}}(k) | \bar{y}_B(k) - \bar{y}_B(k-1) |, \quad (13)$$

and the optimization problem to be solved is

$$\min_{u_{\text{MLD}}(0:95)} J_{H\ell}^{\text{ref}}, \quad (14)$$

where $u_{\text{MLD}}(0 : 95)$ stands for the sequence of inputs $u_{\text{MLD}}(0), \dots, u_{\text{MLD}}(95)$, subject to: the dynamics (12), the load balance (11) and the operational constraints of the devices. In detail, for all $k = 0, \dots, 95$, $\bar{u}_T(k) \in [\bar{u}_T^{\text{min}}, \bar{u}_T^{\text{max}}]$, $\bar{u}_B^{\text{CH}}(k) \in [\bar{u}_{\text{CH}}^{\text{min}}, \bar{u}_{\text{CH}}^{\text{max}}]$, $\bar{u}_B^{\text{DCH}}(k) \in [\bar{u}_{\text{DCH}}^{\text{min}}, \bar{u}_{\text{DCH}}^{\text{max}}]$, $\bar{x}_B(k) \in [0.15, 0.9]$ and $\bar{u}_B^{\text{CH}}(k) = 0 \vee \bar{u}_B^{\text{DCH}}(k) = 0$.

Problem (14) is a mixed-integer linear program (MILP), which can be readily solved by available software tools.

2) *Updated high-level optimization*: During the day, it may be helpful to review a preceding high-level plan so as to better compensate for the unpredicted disturbances and/or possibly incorporate up-to-date nominal profiles of the PV production and load request. To this end, a boolean signal

$$\mathcal{F} : \{k \in \mathbb{N} | 1 \leq k \leq 95\} \rightarrow \{0, 1\}$$

is introduced and the high-level optimization is updated at each time k in which the event-trigger is on, i.e. such that $\mathcal{F}(k) = 1$. For instance, if the updating occurs at all time instants (i.e., $\mathcal{F}(k) = 1$ for all k), then one has a closed-loop high-level controller. Any choice for \mathcal{F} is possible and, in particular, it is interesting the case of signals \mathcal{F} that are specified by some algorithm taking the effective behavior of the lower-layer system into account: e.g., if the lower-layer algorithm encounters systematic difficulties in guaranteeing feasibility, it may send an alert to the top layer optimizer by setting \mathcal{F} equal to 1 and thus asking for a replanning of the MG operation. Different performance resulting from various updating policies \mathcal{F} shall be compared in the simulations reported in Section V.

Let \bar{k} be such that $\mathcal{F}(\bar{k}) = 1$. The cost function associated with the updating optimization problem is defined by

$$J_{H\ell}^{\text{upd}}(\bar{k}) = \sum_{k=\bar{k}}^{95} c_N(k) | y_N^{\text{ref}}(k) - y_N(k) | + c_T(k) \bar{y}_T(k) + c_T^{\text{st}}(k) \max \{ \delta_T(k) - \delta_T(k-1), 0 \} + c_B^{\text{v}}(k) | \bar{y}_B(k) - \bar{y}_B(k-1) |, \quad (15)$$

where the terms appearing in (13) and quantifying the cost/price for the energy exchanged with the network are replaced by a term penalizing the discrepancy from the reference plan. The updated plan to be passed to the low-level controller is then the issue of the following optimization problem:

$$\min_{u_{\text{MLD}}(\bar{k}:95)} J_{H\ell}^{\text{upd}}, \quad (16)$$

subject to: the dynamics (12), the load balance (11) and the same operational constraints of the devices as in problem (14).

IV. LOW-LEVEL MODEL AND CONTROL

The aim of the low-level controller is to compensate for the uncertainties on the load and PV predictions by refining the control actions, i.e., inputs $u_T(h)$ and $u_B(h)$, if both available. The scope is to minimize the deviations of the real outputs with respect to the nominal ones computed by the high-level optimizer and, at the same time, to reduce, under a certain threshold, the probability that the difference between the real energy exchanged with the network and the value computed by the high-level optimizer overcomes a limit value. On the other hand, the low-level model is not allowed to modify the state of the turbine (running/non running) computed at the high level because the prediction horizon considered at the lower layer is shorter than the start-up/shut-down time of the device. Therefore, the contribution of the turbine is available at the low level only if the turbine itself is in running mode. At the lower layer of the proposed architecture, the detailed dynamic models of the devices and the stochastic description of the uncertainties presented in Section II are used.

A. Low-level modelling

The sampling time for the low-level controller is $\tau_{\ell\ell}$. For better clarity, note that we have $n = \tau_{H\ell}/\tau_{\ell\ell} = 15$ samples of each high-level step $[k, k+1)$ and, at time $k\tau_{H\ell} + t\tau_{\ell\ell}$, one has $h = nk + t$.

Indeed, the result of the low-level controller complements the control values \bar{u}_B and \bar{u}_T obtained by the high-level optimizer in such a way that, at time $h = nk + t$, the corresponding inputs to the real system are $u_B(nk + t) = \bar{u}_B(k) + \hat{u}_B(nk + t)$ and $u_T(nk + t) = \bar{u}_T(k) + \hat{u}_T(nk + t)$.

Regarding T in *running mode*, the following variables are considered by the low-level controller.

$$\hat{u}_T(h) = u_T(h) - \bar{u}_T(\lfloor h/n \rfloor) \quad (17a)$$

$$\hat{x}_T(h) = x_T(h) - x_T^{\text{eq}}(\lfloor h/n \rfloor) \quad (17b)$$

$$\hat{y}_T(h) = y_T(h) - \bar{y}_T(\lfloor h/n \rfloor) \quad (17c)$$

where $x_T^{\text{eq}}(k) = (I - A_T)^{-1} B_T \bar{u}_T(k)$ and $\bar{y}_T(k) = C_T x_T^{\text{eq}}(k) = \bar{u}_T(k)$.

Similarly, recalling (3), the following input variable is considered by the low-level controller for B:

$$\hat{u}_B(h) = u_B(h) - \bar{u}_B(\lfloor h/n \rfloor) \quad (18a)$$

Note that the state response of the low-level (fast) model Σ_B to the nominal input $\bar{u}_B(k)$ with initial condition $x_B(kn) = \bar{x}_B(k)$ is not constant in the interval $[kn, (k+1)n)$. This signal, denoted $\bar{\bar{x}}_B(h)$, takes initial value $\bar{\bar{x}}_B(h) = \bar{x}_B(\lfloor h/n \rfloor)$ and evolves according to

$$\bar{\bar{x}}_B(h+1) = \bar{\bar{x}}_B(h) + B_B^{\text{CH}} \bar{u}_B^{\text{CH}}(\lfloor h/n \rfloor) - B_B^{\text{DCH}} \bar{u}_B^{\text{DCH}}(\lfloor h/n \rfloor)$$

We now define the state $\hat{x}_B(h)$ as the variable evolving according to Σ_B with input $\hat{u}_B(h)$ and with initialization $\hat{x}_B(nk) = x_B(nk) - \bar{x}_B(k)$. In view of the linearity of the model one has that

$$\hat{x}_B(h) = x_B(h) - \bar{\bar{x}}_B(h) \quad (18b)$$

The output variable (which depends solely on the inputs) is

$$\hat{y}_B(h) = y_B(h) - \bar{y}_B(\lfloor h/n \rfloor) = \hat{u}_B^{\text{DCH}}(h) - \hat{u}_B^{\text{CH}}(h) \quad (18c)$$

Considering P and L, the variables \hat{x}_P , \hat{y}_P , \hat{x}_L , and \hat{y}_L are used to represent the deviations of the behavior of such systems with respect to the nominal predictions, as discussed in Section II-B.

We introduce a new variable, named $\hat{\epsilon}(h)$, which allows to quantify the integral of the deviation between the real electrical input from the network and the one resulting from the high-level optimizer between two subsequent high-level sampling times. Assuming that $\hat{\epsilon}(h) = 0$ for any $h = kn$, i.e., that such discrepancy is reset to zero at the beginning of any high-level sampling time, the dynamics of $\hat{\epsilon}(h)$, $h = kn, \dots, (k+1)n$ is then given by the equation

$$\hat{\epsilon}(h+1) = \hat{\epsilon}(h) + \hat{y}_N(h)$$

Regarding the energy exchanged with the network recall that, in view of the definition of the nominal profiles $y_P^*(h)$ and $y_L^*(h)$ and (9)-(10), the signal $\bar{y}_N(k)$, generated by the high-level optimization, represents the average value of the network electrical input over the k -th interval. Consistently with this and (5), the nominal electrical input $y_N^*(h)$ is then given by $y_N^*(h) = \bar{y}_N(k) + d_N(h)$ with $d_N(h) = d_L(h) - d_P(h)$. From this it follows that $\hat{y}_N(h) = y_N(h) - \bar{y}_N(\lfloor h/n \rfloor) - d_N(h) = \hat{y}_L(h) - \hat{y}_B(h) - \hat{y}_T(h) - \hat{y}_P(h)$ for all $h \geq 0$.

Letting $\hat{x}(h) = [\hat{x}_B(h) \quad \hat{x}_T(h) \quad \hat{\epsilon}(h) \quad \hat{x}_P(h) \quad \hat{x}_L(h)]'$, $v(h) = [v_P(h) \quad v_L(h)]'$, and $\hat{u}(h) = [\hat{u}_B(h) \quad \hat{u}_T(h)]'$, the evolution of $\hat{x}(h)$, $h = kn, \dots, (k+1)n$, is given by

$$\Sigma_{\ell\ell} : \hat{x}(h+1) = A\hat{x}(h) + B_u\hat{u}(h) + B_vv(h) \quad (19)$$

where

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & A_T & 0 & 0 & 0 \\ 0 & -C_T & 1 & -C_P & C_L \\ 0 & 0 & 0 & A_P & 0 \\ 0 & 0 & 0 & 0 & A_L \end{bmatrix} \quad (20)$$

$$B_u = \begin{bmatrix} B_B & 0 \\ 0 & B_T \\ -D_B & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad B_v = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -D_P & D_L \\ B_P & 0 \\ 0 & B_L \end{bmatrix}.$$

B. The stochastic MPC algorithm for the low-level system

The main goal of the low-level controller is to compensate for the disturbances on the load profile and on the power delivered by the PV in a reactive fashion and to guarantee the fulfillment of the operational constraints. Thus, a stabilization problem of the origin of system $\Sigma_{\ell\ell}$ is considered. Since the nominal condition of the system (and also its structure, as the microturbine can be switched off) can significantly vary at each high-level sampling time, a ‘‘shrinking-horizon’’ MPC algorithm is adopted. Moreover, in view of the probabilistic nature of the model, the control algorithm is cast as a stochastic MPC.

1) The control law and the shrinking-horizon approach:

Consistently with the fact that the model $\Sigma_{\ell\ell}$ is in charge of the dynamics over the time interval $[nk, nk+n)$ only and according to the discussion in Section IV-A, at the beginning of each high-level sampling time (i.e., when $h = nk$, for all $k \geq 0$) the following initialization of the low-level model is in order

$$\hat{x}(nk) = \begin{bmatrix} x_B(nk) - \bar{x}_B(k) \\ x_T(nk) - \bar{x}_T(k) \\ 0 \\ \hat{x}_P(nk) \\ \hat{x}_L(nk) \end{bmatrix} \quad (21)$$

For simplicity of presentation, but with a slight abuse of notation, the time index of the input variables \hat{x} , \hat{u} , and v is reset so that, in equation (19), $\hat{x}(nk+t)$, $\hat{u}(nk+t)$, and $v(nk+t)$ are replaced by $\hat{x}(t)$, $\hat{u}(t)$, and $v(t)$, respectively and our analysis focuses in the interval $[0, n)$ solely.

At any time $t < n$ a measure of the state $\hat{x}(t)$ is assumed to be available and, based on this, an input sequence $\{\hat{u}(t|t) \hat{u}(t+1|t) \dots \hat{u}(n-1|t)\}$ of length $n-t$ minimizing a certain performance index $J_{\ell\ell}$, possibly under some state and input constraints, is looked for². Hence, letting $\hat{u}_{\text{opt}}(t|t)$ be the first element of the optimal sequence, according to the ‘‘shrinking-horizon’’ principle, the command $\hat{u}(t) = \hat{u}_{\text{opt}}(t|t)$ is applied and, at time $t+1$, the optimization procedure is repeated over shorter input sequences $\{\hat{u}(t+1|t+1) \hat{u}(t+2|t+1) \dots \hat{u}(n-1|t+1)\}$ of length $n-t-1$. Because of the stochastic disturbances acting on the system, similarly to [15] the adopted control law takes the form, for $t, \dots, n-1$

$$\hat{u}(j|t) = \tilde{u}(j|t) + \hat{K}^M(\hat{x}(j|t) - \tilde{x}(j|t)) \quad (22)$$

where $\tilde{x}(j|t)$ is the mean value of the random variable $\hat{x}(j|t)$ and $\tilde{u}(j|t)$ is the mean value of the random variable $\hat{u}(j|t)$. Moreover, as it will be clarified in Section IV-B3 (see, e.g., constraints (31) on the structure of \hat{K}^M), the value taken by the gain \hat{K}^M depends on the adopted battery mode and is such that $F^M = A + B_u\hat{K}^M$ is Schur stable.

Under the assumption that v is a zero-mean white noise with variance V , the variable \tilde{x} evolves as follows.

$$\begin{cases} \tilde{x}(j+1|t) = A\tilde{x}(j|t) + B_u\tilde{u}(j|t) \\ \tilde{x}(t|t) = \hat{x}(t). \end{cases} \quad (23)$$

As for the variance X of \hat{x} , its evolution (i.e., for $j \geq t$) is given by

$$\begin{cases} X(j+1) = \hat{F}^M X(j) \hat{F}^{M'} + B_v V B_v' \\ X(t) = 0. \end{cases} \quad (24)$$

The MPC algorithm at the low level is aimed at minimizing the performance index

$$J_{\ell\ell}^M(\hat{x}(t)) = \mathbb{E} \left[\sum_{j=t}^{n-1} (\|\hat{x}(j|t)\|_Q^2 + \|\Delta\hat{u}(j|t)\|_R^2) + \|\hat{x}(n|t)\|_S^2 \right],$$

with respect to the sequence of control values $\{\tilde{u}(t|t) \dots \tilde{u}(n-1|t)\}$, subject to the dynamics (19)

²Where, as usual, the notation $z(j|t)$ stands for the prediction made at time t of the variable $z(j)$ that, depending on the particular circumstances, can be either a numerical value or a random variable.

with (22) and to the state and input (probabilistic) constraints specified in the following. Matrices Q , R and S are positive definite and symmetric, while $\Delta\hat{u}(j|t)$ is given by

$$\Delta\hat{u} = \begin{bmatrix} \hat{y}_B \\ \hat{u}_T \end{bmatrix} = D_\Delta\hat{u} \quad (25)$$

where

$$D_\Delta = \begin{bmatrix} -1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$\Delta\hat{u}$ is therefore a vector containing: (i) \hat{y}_B , i.e., the difference between the real net power exchanged by B with the network and its nominal value given by the high-layer optimizer; (ii) \hat{u}_T , i.e., the difference between the real input to T and the nominal one. As shown in [15], $J_{\ell\ell}^M(\hat{x}(t)) = J_m(\hat{x}(t)) + J_v(M)$, where:

$$J_m(\hat{x}(t)) = \sum_{j=t}^{n-1} \left(\|\tilde{x}(j|t)\|_Q^2 + \|\Delta\tilde{u}(j|t)\|_R^2 \right) + \|\tilde{x}(n|t)\|_S^2, \quad (26)$$

where, from (25), $\Delta\tilde{u}(j|t) = \mathbb{E}[\Delta\hat{u}(j|t)] = D_\Delta\tilde{u}(j|t)$. Moreover

$$J_v(M) = \sum_{j=t}^{n-1} \left(\text{tr}[(Q + \hat{K}^{M'} D'_\Delta R D_\Delta \hat{K}^M) X(j)] \right) + \text{tr}[S X(n)]. \quad (27)$$

It is worth noting that, since the evolution (including the initial value) of $X(j)$, for $j = t, \dots, n$ is fully determined as in (24), the only degree of freedom affecting the value of J_v is the battery mode $M \in \{\text{CH}, \text{DCH}\}$ (in view of the fact that in general $\hat{K}^{\text{CH}} \neq \hat{K}^{\text{DCH}}$). Therefore, this term does not play a significant role in the minimization of $J_{\ell\ell}^M$ and will be discarded in the cost function considered in the low-level optimization problem.

2) *Handling probabilistic state and input constraints:* Due to the stochastic nature of the model, input and state constraints must be imposed in a probabilistic sense. According to [21], a probabilistic constraint of the type

$$\mathbb{P}[b'z \geq \alpha^{\max}] \leq p, \quad (28)$$

where $b \in \mathbb{R}^n$, $\alpha^{\max} \in \mathbb{R}$ and $0 \leq p \leq 1$ are given and z is a random vector taking values in \mathbb{R}^n , is implied by the condition

$$b'\mathbb{E}[z] \leq \alpha^{\max} - \sqrt{b'\text{Var}[z]b} \cdot f(p) \quad (29)$$

where the function $f(p)$ depends on the a-priori assumptions on the distribution of z . For example, in v_P and v_L are characterized by a Gaussian probability distribution, $f(1-p) = \mathcal{N}^{-1}(1-p)$ where \mathcal{N} is the cumulative probability function of a Gaussian variable with zero mean and unitary variance; on the other hand, according to the Cantelli-Chebyshev inequality, the more stringent choice $f(p) = \sqrt{\frac{1-p}{p}}$ must be used for unknown distributions. Therefore, probabilistic constraints of the type (28) are turned into conditions on the mean value and the variance in the form (29) that, in view of (23) and (24), can be easily handled. However, it is worth remarking that, in the current framework (and differently from [15]), we set $\tilde{x}(t|t) = \hat{x}(t|t)$, and therefore $\hat{u}(t|t) = \tilde{u}(t|t)$ is a deterministic variable, while

$\hat{x}(j|t)$ and $\hat{u}(j|t)$ are stochastic ones for $j = t+1, \dots, n$. In view of this, condition (29) reduces to a deterministic constraint on $\hat{x}(t|t)$ and $\hat{u}(t|t)$. Moreover, since $x_B(h+1)$ does not depend on $v(h)$ (see equation (19)), deterministic constraints on the battery state of charge at time $t+1$ are enforced as well.

Remark 1: In the literature, many sMPC algorithms have been proposed, see the recent review [21]. Among them, the scenario, or randomized, approach (see e.g. [22], [23]) is one of the most promising, since it does not require specific assumptions on the model structure and noise distribution. As a partial drawback, randomized methods can require to generate a quite large number of samples of the disturbance realizations, so that the computational effort required can be huge. For this reason, in this work the deterministic reformulation of the probabilistic constraints according to (28), (29) has been mainly considered. However, a comparison among the performance of this method and the one of a scenario approach derived from [22] is briefly discussed in Section V

3) *Input constraints for the battery and mode operation management:* In this section we first present how the nonlinear constraint (4) is enforced in our setup. Two important facts are in order.

- 1) In each optimization step, the prediction is computed under the assumption that the battery operates in a fixed mode over the prediction horizon.

This implies that either $\bar{u}_B^{\text{CH}}(k) + \hat{u}_B^{\text{CH}}(j|t) = 0$, for all $j = t, \dots, n-1$ (battery in the discharge mode, i.e. $M = \text{DCH}$), or $\bar{u}_B^{\text{DCH}}(k) + \hat{u}_B^{\text{DCH}}(j|t) = 0$, for all $j = t, \dots, n-1$ (battery in the charge mode, i.e., $M = \text{CH}$). In view of this, the low-level optimization problem is decomposed into two problems that are solved in parallel: one corresponding to $M = \text{CH}$ and one corresponding to $M = \text{DCH}$, and the selected battery operation mode is the one corresponding to the minimum cost function.

It is worth remarking that, however, in view of the closed loop nature of the proposed low-level control scheme and the adopted shrinking-horizon implementation, at each time step only the first element of the optimal sequence is applied and afterwards the low-level optimization is repeated. Therefore the resulting control law can alternate between the charge and the discharge mode. This possibility is important especially when the high-level optimizer selects a small value for the power exchange between the battery and the micro-grid.

- 2) The battery operation mode adopted by the real system (denoted with symbol \bar{M}) may be not consistent with the one selected by the high layer: in a few words, for example $\bar{u}_B^{\text{CH}}(k) = 0$ does not imply that $u_B^{\text{CH}}(h) = 0$ for all $h = 0, \dots, n-1$. This first implies that, for each mode, a different set of inequalities is induced. More specifically

$$\begin{cases} \hat{u}_B^{\text{CH}}(j|t) & \geq -\bar{u}_B^{\text{CH}}(k) \\ \hat{u}_B^{\text{DCH}}(j|t) & = -\bar{u}_B^{\text{DCH}}(k) \end{cases} \text{ if } \bar{M} = \text{CH} \quad (30a)$$

$$\begin{cases} \hat{u}_B^{\text{CH}}(j|t) &= -\bar{u}_B^{\text{CH}}(k) \\ \hat{u}_B^{\text{DCH}}(j|t) &\geq -\bar{u}_B^{\text{DCH}}(k) \end{cases} \text{ if } M = \text{DCH} \quad (30b)$$

The second main implication of this fact is that, as a matter of fact, one of the two battery input variables of the low-level model (19) (i.e., either \hat{u}_B^{CH} or \hat{u}_B^{DCH}) must verify an equality constraint.

For instance, in charge mode $M = \text{CH}$, this means that we must set $\hat{u}_B^{\text{DCH}} = \bar{u}_B^{\text{DCH}} = -\bar{u}_B^{\text{DCH}}$. Since the only degree of freedom is \bar{u}_B^{DCH} , while the relationship between \bar{u}_B^{DCH} and \hat{u}_B^{DCH} is given by (22), the former equality can be enforced by letting the second column of the gain matrix \hat{K}^{CH} (i.e., the column corresponding with input u_B^{DCH}) be zero. This, indeed, is not critical as far as the controllability properties of the system are concerned, since the two battery inputs are, in a way, redundant.

Summing up, the gains \hat{K}^M must be designed in such a way that, for all $M \in \{\text{CH}, \text{DCH}\}$, $A + B\hat{K}^M$ is Schur stable and, at the same time

$$\begin{cases} k_{i2}^{\text{CH}} = 0 & \text{for all } i = 1, \dots, 5 \\ k_{i1}^{\text{DCH}} = 0 & \text{for all } i = 1, \dots, 5 \end{cases} \quad (31)$$

where k_{ij}^M is the (i, j) -th entry of matrix \hat{K}^M .

Finally recall that, from the definition of matrices \hat{K}^M and D_Δ , and from the control law (22), in operation mode CH, $\hat{u}_B^{\text{CH}}(j|t) - \bar{u}_B^{\text{CH}}(j|t) = -e_1' D_\Delta \hat{K}^{\text{CH}} (\hat{x}(j|t) - \tilde{x}(j|t))$ while, in operation mode DCH, $\hat{u}_B^{\text{DCH}}(j|t) - \bar{u}_B^{\text{DCH}}(j|t) = e_1' D_\Delta \hat{K}^{\text{DCH}} (\hat{x}(j|t) - \tilde{x}(j|t))$, where e_i denotes the i -th orthonormal basis vector. From this we derive that, in general, in operation mode M, $U_B^M(j) = \text{Var}[\hat{u}_B^M(j|t)] = e_1' D_\Delta \hat{K}^M X(j) \hat{K}^{M'} D_\Delta' e_1$.

In addition to constraint (4), a further input constraint for the battery is that both the charge $\bar{u}_B^{\text{CH}}(k) + \hat{u}_B^{\text{CH}}(j|t)$ and the discharge $\bar{u}_B^{\text{DCH}}(k) + \hat{u}_B^{\text{DCH}}(j|t)$ power values are bounded by a maximal value u_B^{max} .

Summing up, the overall constraints to be imposed on the battery inputs are therefore the following.

If $M = \text{CH}$

$$\begin{cases} \bar{u}_B^{\text{CH}}(j|t) &\geq -\bar{u}_B^{\text{CH}}(k) + \sqrt{U_B^{\text{CH}}(j)} f(p_{u,B}) \\ \hat{u}_B^{\text{CH}}(j|t) &\leq -\bar{u}_B^{\text{CH}}(k) + u_B^{\text{max}} - \sqrt{U_B^{\text{CH}}(j)} \cdot f(p_{u,B}) \\ \hat{u}_B^{\text{DCH}}(j|t) &= -\bar{u}_B^{\text{DCH}}(k) \end{cases} \quad (32a)$$

If $M = \text{DCH}$

$$\begin{cases} \bar{u}_B^{\text{CH}}(j|t) &= -\bar{u}_B^{\text{CH}}(k) \\ \hat{u}_B^{\text{DCH}}(j|t) &\geq -\bar{u}_B^{\text{DCH}}(k) + \sqrt{U_B^{\text{DCH}}(j)} f(p_{u,B}) \\ \hat{u}_B^{\text{CH}}(j|t) &\leq -\bar{u}_B^{\text{DCH}}(k) + u_B^{\text{max}} - \sqrt{U_B^{\text{DCH}}(j)} f(p_{u,B}) \end{cases} \quad (32b)$$

where $p_{u,B}$ is the tolerated probability that the thresholds are infringed. Finally, recall that, at step $j = t$, $X(t) = 0$ and therefore $U_B^M(t) = 0$, meaning that the constraints are indeed deterministic for $j = t$.

4) *State constraints on the battery:* Recall that variable $\bar{x}_B(j)$ describes the evolution of the battery SOC in the fast timescale, under the nominal input $\bar{u}_B(j)$. To impose that the

battery state of charge $x_B(j) = \bar{x}_B(j) + \hat{x}_B(j)$ is confined within $[x_B^{\text{min}}, x_B^{\text{max}}]$, the following constraints are enforced

$$\begin{cases} \bar{x}_B(j) + \tilde{x}_B(j|t) &\leq x_B^{\text{max}} - \sqrt{e_1' X(j) e_1} \cdot f(p_{x,B}) \\ \bar{x}_B(j) + \tilde{x}_B(j|t) &\geq x_B^{\text{min}} + \sqrt{e_1' X(j) e_1} \cdot f(p_{x,B}) \end{cases} \quad (33)$$

where $p_{x,B}$ is the tolerated probability that the thresholds x_B^{min} or x_B^{max} are infringed.

5) *Input constraints on the turbine:* If $\bar{u}_T(k) > 0$ (i.e., the high-level optimizer has planned to use the turbine during the k -th sampling interval), then the preview power request to the turbine $\bar{u}_T(k) + \hat{u}_T(j|t)$ should be confined within a prescribed interval $[u_T^{\text{min}}, u_T^{\text{max}}]$.

In details, the random variable $\hat{u}_T(j|t) = e_3' \hat{u}(j|t)$ is the component relative to the turbine input of the total input vector $\hat{u}(j|t) = \tilde{u}(j|t) + \hat{K}^M (\hat{x}(j|t) - \tilde{x}(j|t))$. The mean value of $\hat{u}_T(j|t)$ is $\bar{u}_T(j|t)$, while its variance is $U_T(j) = e_3' \text{Var}[\hat{u}(j|t)] e_3 = e_3' \hat{K}^M X(j) \hat{K}^{M'} e_3$. Thus, according to (29), the input constraints on the turbine are

$$\begin{cases} \bar{u}_T(k) + \bar{u}_T(j|t) &\leq u_T^{\text{max}} - \sqrt{U_T(j)} \cdot f(p_{u,T}) \\ \bar{u}_T(k) + \bar{u}_T(j|t) &\geq u_T^{\text{min}} + \sqrt{U_T(j)} \cdot f(p_{u,T}) \end{cases} \quad (34)$$

where $p_{u,T}$ is the tolerated probability that the thresholds u_T^{min} or u_T^{max} are infringed.

6) *Final state constraint on \hat{e} :* In order to avoid penalties due to discrepancies between the real electric input and what agreed on by the high-level optimizer, it is imposed that, at the end of the prediction horizon (i.e., at the time corresponding to the $(k+1)$ -th high-level sampling time), the absolute value of \hat{e} is below a tolerance threshold ϵ^{max} . That is,

$$\begin{cases} \tilde{\epsilon}(n|t) &\leq \epsilon^{\text{max}} - \sqrt{e_3' X(n) e_3} \cdot f(p_\epsilon) \\ -\tilde{\epsilon}(n|t) &\leq \epsilon^{\text{max}} - \sqrt{e_3' X(n) e_3} \cdot f(p_\epsilon), \end{cases} \quad (35)$$

where p_ϵ is the tolerated probability that the thresholds $-\epsilon^{\text{max}}$ or ϵ^{max} are infringed.

7) *The low-level MPC controller: résumé and feasibility issue:* In this section we give an overview of the algorithm for the selection of the low-level control action $\hat{u}(t)$. Let $\tilde{u}_{\text{opt}}(t|t)$ be the first element of the control sequence that solves the following optimal control problem

$$\min_{M \in \{\text{CH}, \text{DCH}\}} \min_{[\tilde{u}(t|t) \dots \tilde{u}(n-1|t)]} J_m(\hat{x}(t)) \quad (36)$$

subject to: the dynamics (23)

the constraints (32), (33), (34), (35),

where the expression of $J_m(\hat{x}(t))$ is given in equation (26), then let

$$\hat{u}(t) = \tilde{u}_{\text{opt}}(t|t). \quad (37)$$

It is worth noting that, considering the operational constraints (32), (33), (34), since we reset $X(t) = 0$ for all $t = 0, \dots, n-1$, at each time t the correct operativity regions are indeed satisfied with probability 1. This holds provided that the optimization problem (36) results feasible. However, recursive feasibility of problem (36) is not guaranteed as it may be disrupted by the presence of unusual disturbance realizations. In particular, constraint (35) may be conflicting with the operational constraints (32)-(34). On the other hand,

since it is always possible to obtain the load balance through the power exchange y_N with the electrical network, it is possible to attain the main operational constraints and to offload the effects of the disturbance on the variable $\hat{\epsilon}$: in other words, feasibility can be recovered if constraint (35) is removed. In details, the following procedure is adopted.

Algorithm 1 (Complete low-level controller):

- **Step 0.** Consider the optimization problem (36): if it is feasible, compute $\hat{u}(t)$ as in equation (37), let $t = t + 1$ and go to step 0; if it is infeasible, go to step 1.
- **Step 1.** Set to 1 the probabilities $p_{u,T}$, $p_{u,B}$, and $p_{x,B}$ in constraints (32)-(34) and consider problem (36): if it is feasible, compute $\hat{u}(t)$ as in equation (37), let $t = t + 1$ and go to step 0; if it is infeasible, go to step 2.
- **Step 2.** Set to 1 the probability p_ϵ in constraint (35) and consider problem (36): if it is feasible, compute $\hat{u}(t)$ as in equation (37), let $t = t + 1$ and go to step 0; if it is infeasible, go to step 3.
- **Step 3.** Remove constraint (35), solve problem (36) and get $\hat{u}(t)$ as in equation (37), let $t = t + 1$ and go to step 0.

Recursive feasibility of the complete low-level control algorithm is established by the following result.

Proposition 1: If $\hat{x}_B(t)$ is such that $x_B^{\min} \leq \bar{x}_B(t) + \hat{x}_B(t) \leq x_B^{\max}$, then the optimization problem considered in step 3 is feasible and the state at time $t + 1$ satisfies $x_B^{\min} \leq \bar{x}_B(t + 1) + \hat{x}_B(t + 1) \leq x_B^{\max}$.

Proof. It suffices to show that the optimization problem in step 3 is feasible, since the subsequent condition on $\hat{x}_B(t + 1)$ follows from the main operational constraint (33). As a consequence of step 1, $f(p_{u,T}) = 0$ and the intervals defining constraint (34) on $\bar{u}_T(j|t)$ are nonempty. The result is then achieved by showing that the sequence of battery inputs $\bar{u}_B^{CH}(j|t) = -\bar{u}_B^{CH}(k)$, for all $j = t, \dots, n - 1$, is admissible: indeed, in view of step 1, $f(p_{u,B}) = 0$ and constraint (32) is satisfied; as for constraint (33), there also holds that $f(p_{x,B}) = 0$, then notice that the considered input sequence ensures that $\bar{x}_B(j) + \hat{x}_B(j|t)$ is constant and hence equal to $\bar{x}_B(t) + \hat{x}_B(t)$. ■

Remark 2 (Turbine switched off): If $\bar{u}_T(k) = 0$, i.e., the high-level optimizer has planned not to use the turbine during the k -th sampling interval, then this decision cannot be contradicted by the low-level optimizer. In this case, the control algorithm described in the previous sections should be simply modified by removing, both in model (19) and in the subsequent presentation, all the equations and terms involving the state and the control of the turbine. Notice, however, that it is no more possible to select the control gain \hat{K}^M so that $\hat{F}^M = A + B_u \hat{K}^M$ is a Schur matrix because the system is single-input and has two integrators (i.e., the dynamics of both \hat{x}_B and $\hat{\epsilon}$). Nonetheless, the main role of \hat{K}^M is to maintain the variance X of \hat{x} as contained as possible and there are not specific properties required to \hat{K}^M (in principle, any matrix is a feasible choice). Thus, when the turbine is switched-off, a good choice for \hat{K}^M is one ensuring the best possible variance reduction of the $\hat{\epsilon}$ component of the state. For this reason, in the proposed implementation of the algorithm described in

Section V, \hat{K}^M has been chosen so as to act as a deadbeat controller on $\hat{\epsilon}$. □

8) *Properties guaranteed by the algorithm:* As it has been already discussed in Section IV-B7, the main operational constraints ensure that the system is correctly and safely operated.

As far as the terminal value of $\hat{\epsilon}$ is concerned, the guaranteed properties depend on the implemented step of the algorithm at time $t = n - 1$: if the complete low-level controller selects the control action by resorting to step 0 or 1, then constraint satisfaction is guaranteed in probability (specifically, condition (35) implies that $\mathbb{P}[|\hat{\epsilon}(n)| > \epsilon^{\max}] \leq \min\{2p_\epsilon, 1\}$); if the control action is selected by resorting to step 2, then condition (35) reduces to $-\epsilon^{\max} \leq \tilde{\epsilon}(n|t) \leq \epsilon^{\max}$, which merely implies that $\mathbb{P}[|\hat{\epsilon}(n)| > \epsilon^{\max}] \leq 1$. As such, condition (35) does not guarantee anything, in general, unless a particular distribution is assumed on the random variable v (e.g., if v is a Gaussian white-noise, condition $\tilde{\epsilon}(n|t) \leq \epsilon^{\max}$ also guarantees that $\mathbb{P}[\hat{\epsilon}(n) > \epsilon^{\max}] \leq \frac{1}{2}$). Finally, nothing can be ensured on $\hat{\epsilon}(n)$ if the algorithm is forced to resort to step 3: in this case, one can only attempt to minimize the damage by modifying the weighting matrix Q in the performance index so as to considerably penalize the component related to the variable $\hat{\epsilon}$. On the other hand, it is apparent that in front of sudden and large variations of the load and of the PV power, the only way to instantaneously guarantee the power balance is to modify the power exchange with the grid.

V. SIMULATION STUDY

In this section, simulation tests are presented that witness the effectiveness of the proposed control approach. Specifically, the advantages deriving by the introduction of the low-level controller are first discussed. Then a comparison between the presented stochastic algorithm and a scenario-based approach is presented. Finally, the results achievable by means of different high level plan update policies are illustrated. Representative examples have been selected among an extensive collection of algorithm executions.

a) *Benefits of the two-level structure:* According to the procedure illustrated in Section III-B1, realistic forecast data of both the photovoltaic production and the load consumption, as well as the market prices, are needed to solve the high-level optimization problem. These profiles are represented by the dashed blue lines in Figure 2 (the actual data are also represented therein by a solid red line), while Figure 3 collects the unitary energy production and trade prices. The microturbine start-up cost c_T^{st} in equation (13) – is considered to be constant and equal to € 0.175. The resulting optimal setpoint of the MG devices and the profile of the power exchange with the network are jointly reported in Figure 4.

Figure 5 shows the difference between the nominal setpoint profiles of the MG devices and those obtained with the low-level control action compensating for the fluctuations of the real photovoltaic generation and load consumption. In order to emphasize the contribution of the lower layer

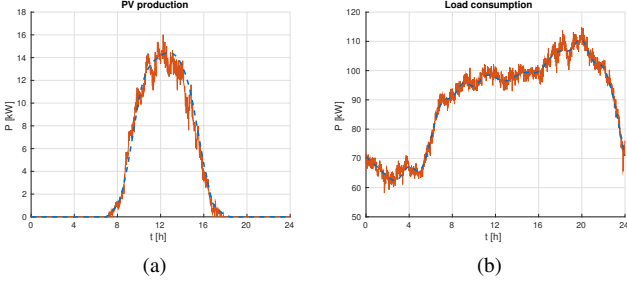


Fig. 2. (a) Photovoltaic forecast (dashed blue line) and real production (solid red line); (b) Load forecast (dashed blue line) and real consumption (solid red line).

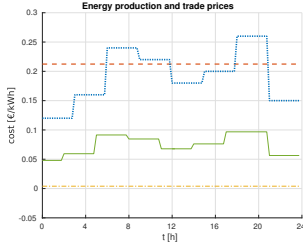


Fig. 3. Unitary energy price used in the cost function $J_{H\ell}^{\text{ref}}$ of equation (13): c_T (dashed red line), c_N^a (blue dotted line), c_N^s (green solid line), c_B^v (yellow dash-dot line).

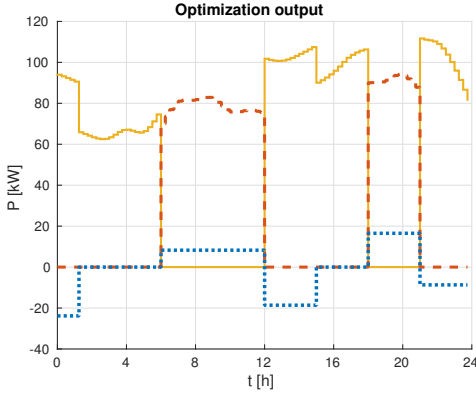


Fig. 4. Profiles generated from the high-level controller: $\bar{y}_B(k)$ (dotted blue line), $\bar{y}_T(k)$ (dashed red line), $\bar{y}_N(k)$ (solid yellow line); power is positive when flowing from the component to the MG.

controller, the deviation from the nominal profile of the energy exchange with the distribution network, *i.e.*, the values $\hat{\epsilon}(n)$ at the end of each k -th time instant, is shown in Figure 6 in two cases: (i) when the low-level control is active; (ii) when only the nominal profiles are applied to the MG devices. Note that the action of the low-level controller allows to reduce both the number of discrepancies with the original plan (*i.e.*, occurrences of $|\hat{\epsilon}(n)| > \epsilon^{\text{max}}$) and the total unpredicted energy exchange, while maintaining the power production profile of each MG component close to the nominal values, thus contributing to the economically optimal energy management. In particular, in this example, the number of discrepancies drops from 76 to 4 while the

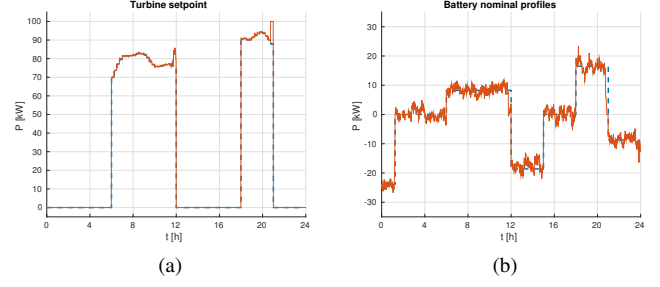


Fig. 5. (a) Turbine power production: $\bar{y}_T(k)$ (dashed blue line), $y_T(h)$ (solid red line); (b) Battery power production: $\bar{y}_B(k)$ (dashed blue line), $y_B(h)$ (solid red line).

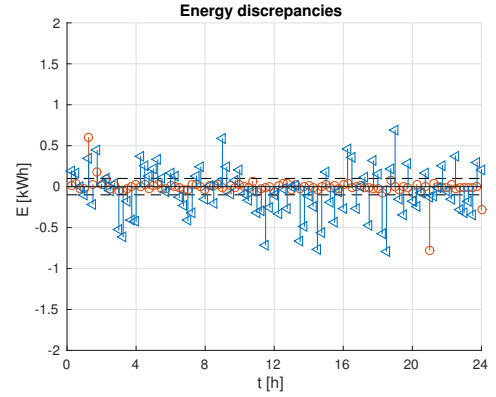


Fig. 6. Discrepancy between the exchanged energy with the network and the nominal one during the high-level control intervals (*i.e.*, the values $\hat{\epsilon}(n)$ at the end of each k -th time instant): controlled vs uncontrolled case (red circles and blue triangles, respectively).

total amount of unplanned energy exchanges with the network is reduced from 22.95 kWh to 1.83 kWh.

Notice that in this simulation, as much as in the following one, an additional constraint has been added to the optimization problems specifying the final value of the battery state of charge, so as to ensure that the shown results are exclusively ascribable to the effect of the control action.

b) Comparison with the scenario approach: To implement the low-level control problem according to a scenario-based optimization approach, both the constraint on the unpredicted energy exchange with the main grid, and the cost function have been modified. Specifically, the condition on $\hat{\epsilon}$ has been reformulated by means of the following soft constraints in order to guarantee feasibility.

$$\hat{\epsilon}(n) \leq \epsilon^{\text{max}} + \rho \quad \wedge \quad -\hat{\epsilon}(n) \leq \epsilon^{\text{max}} + \rho \quad \rho \geq 0,$$

where ρ represents a slack variable and $\hat{\epsilon}$ depends, through (19), on both the input sequence $\hat{u}(t : n - 1)$, and the K_n sample extractions of $v(t : n - 1)$.

Concerning the cost function, finally, we adopted the sampled average over K_n noise realizations.

The performance achievable with the two different approaches in the same framework of the previous example have been compared in Table I, where the number of discrepancies and

TABLE I
COMPARISON OF THE TWO LOW-LEVEL IMPLEMENTATION

	sMPC	scenario-based approach
N discrepancies	4	4
Energy [kWh]	1.83	1.89

the value of unexpected energy exchange are collected.

It appears that the results are definitely comparable. However, note that, with the scenario-based approach, the number of samples to be generated might become quite large in order to guarantee some properties of the method. For instance, in our case and according to [22], taking $p = 0.05$, and a confidence parameter $\theta = 10^{-6}$, 326 samples are needed, introducing a total of 652 constraints for each of the two MPC problems to be solved at any time instant.

c) Update of the high level plan: As described in Section (III-B2), the introduction of updates of the high-level optimization during the control phase allows to better counteract the effect of significant and unpredicted deviation of the real data from the forecasted values. Performance improvements obtainable via suitable updating policies \mathcal{F} are exemplified by the following simulation test, in which we assumed the case of a large variation of the photovoltaic power production from its forecasted values.

Three different update strategies have been considered: (i) periodic, with a one hour period (*i.e.*, $\mathcal{F}(k) = 1$ for all $k = 4i$, with $i = 1, 2, \dots, 23$); (ii) triggered by the low layer performance (this happens, in particular, if at the end of the control horizon the algorithm is forced to implement one of the backup solution 2–3 or if the difference between the actual and the estimated state of charge of the battery \hat{x}_B is greater or equal to the 10% of its total capacity); (iii) triggered by the availability of more reliable forecast data (in the following example we considered just one update between 9.15 and 9.30 A.M). Figure 7 collects the initial photovoltaic production forecasts (dashed blue line) and the real data (solid red line), together with the updated forecast that is made available to the control structure in case (iii) and identified by the dotted green line. We considered the same load profiles and the same unitary costs reported in Figures (2)(b)–(3). Figure (8)–(10) show the nominal profiles of the MG devices and the network power exchange obtained by the algorithm with the alternative high-level plan update strategies. The results of the corresponding control action is then collected in table (II) and compared, in terms of discrepancy in the energy exchange with the distribution network, to those achieved without any update of the initial setpoint. As reasonably expected, the algorithm performs better with realistic forecast data; it is quite interesting, although, in the authors' opinion, to observe how the introduction of a periodic update policy or, even better, the interaction between the two layers of the algorithm can improve the results of the controlled system, partially compensating for unreliable input data.

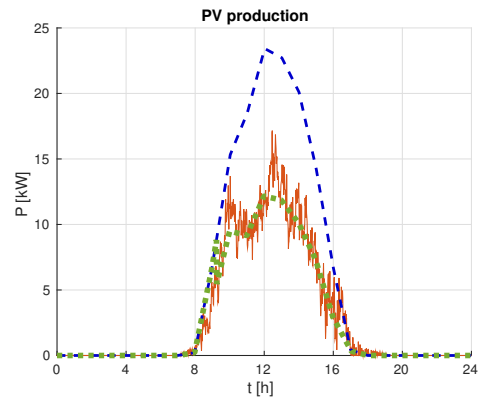


Fig. 7. Initial photovoltaic forecast (dashed blue line) and updated (dotted green line), together with the real production (solid red line).

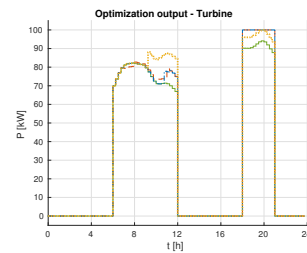


Fig. 8. Turbine high-level setpoint $\bar{y}_T(k)$: initial (solid green line) and updated profiles for strategy (i) (dashed red line), strategy (ii) (dot-dashed blue line) and strategy (iii) (dotted yellow line).

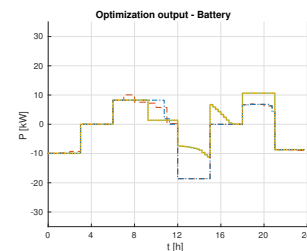


Fig. 9. Battery setpoint $\bar{y}_B(k)$: initial (solid green line) and updated profiles for strategy (i) (dashed red line), strategy (ii) (dot-dashed blue line) and strategy (iii) (dotted yellow line).

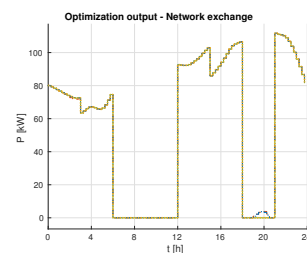


Fig. 10. Nominal power exchange with the network $\bar{y}_N(k)$: initial (solid green line) and updated profiles for strategy (i) (dashed red line), strategy (ii) (dot-dashed blue line) and strategy (iii) (dotted yellow line).

VI. CONCLUSIONS

This work presents a two-layer algorithm for the optimal energy management in micro-grids. The high-level *off-line*

TABLE II
COMPARISON OF THE DIFFERENT UPDATE STRATEGIES RESULTS

	No Update	strategy (i)	strategy (ii)	strategy (iii)
N discrepancies	11	11	13	1
Energy [kWh]	15.07	5.49	5.42	0.39

economic optimization is combined with a low-level *online* stochastic MPC. The optimizer produces a solution based on the nominal forecasts of the PV production and of the load consumption, while the MPC algorithm allows for the minimization of the discrepancies with the optimal plan in the presence of stochastic disturbances. The update of the high-level plan consent to re-define the nominal profiles according to the temporarily optimal solution, enhancing the performance of the whole control algorithm and contributes to compensate for any forecast error. The potential of the proposed approach is witnessed by the illustrated simulations and motivates further research on this topic. At the moment an extension to account for the thermal energy needs of the MG is under investigation. Preliminary results concerning the high-level implementation are available in [24].

ACKNOWLEDGMENTS

The authors are indebted to A. Guagliardi, C. Sandroni, and E. Corsetti (RSE S.p.A.) for fruitful discussions and suggestions. S. Raimondi Cominesi is financially supported by RSE S.p.A.

REFERENCES

- [1] US Department of Energy, "The smart grid: an introduction," in *Technical Report*, 2009.
- [2] R. Lasseter and P. Paigi, "Microgrid: a conceptual solution," in *IEEE 35th Annual Power Electronics Specialists Conf.*, vol. 6, 2004, pp. 4285–4290.
- [3] A. Parisio, E. Rikos, and L. Glielmo, "A model predictive control approach to microgrid operation optimization," *IEEE Trans. on Control Systems Technology*, vol. 22, no. 5, pp. 1813–1827, 2014.
- [4] S. Bracco, F. Delfino, F. Pampararo, M. Robba, and M. Rossi, "A mathematical model for the optimal operation of the University of Genoa Smart Polygeneration Microgrid. Evaluation of technical, economic and environmental performance indicators," *Energy*, vol. 64, pp. 912–922, 2014.
- [5] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35(3), pp. 407–427, 1999.
- [6] E. Minò, "Modellizzazione e controllo ottimo di una rete di generazione distribuita," *MS Thesis, R. Scattolini supervisor, Politecnico di Milano*, 2007.
- [7] W. Qi, J. Liu, and P. D. Christofides, "A distributed control framework for smart grid development: Energy/water system optimal operation and electric grid integration," *Journal of Process Control*, vol. 21, no. 10, pp. 1504–1516, 2011.
- [8] I. Prodan and E. Zio, "A model predictive control framework for reliable microgrid energy management," *International Journal of Electrical Power & Energy Systems*, vol. 61, pp. 399–409, 2014.
- [9] A. Parisio and L. Glielmo, "Stochastic model predictive control for economic/environmental operation management of microgrids," in *Control Conference (ECC), 2013 European*. IEEE, 2013, pp. 2014–2019.
- [10] L. Ji, D. Niu, and G. Huang, "An inexact two-stage stochastic robust programming for residential micro-grid management-based on random demand," *Energy*, vol. 67, pp. 186–199, 2014.
- [11] T. Niknam, R. Azizipanah-Abarghooee, and M. R. Narimani, "An efficient scenario-based stochastic programming framework for multi-objective optimal micro-grid operation," *Applied Energy*, vol. 99, pp. 455–470, 2012.
- [12] P. Meibom, R. Barth, B. Hasche, H. Brand, C. Weber, and M. O'Malley, "Stochastic optimization model to study the operational impacts of high wind penetrations in Ireland," *Power Systems, IEEE Transactions on*, vol. 26, no. 3, pp. 1367–1379, 2011.
- [13] W. Su, J. Wang, and J. Roh, "Stochastic energy scheduling in microgrids with intermittent renewable energy resources," *Smart Grid, IEEE Transactions on*, vol. 5, no. 4, pp. 1876–1883, 2014.
- [14] P. Patrinos, S. Trimboli, and A. Bemporad, "Stochastic MPC for real-time market-based optimal power dispatch," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*. IEEE, 2011, pp. 7111–7116.
- [15] M. Farina, L. Giulioni, L. Magni, and R. Scattolini, "A probabilistic approach to model predictive control," in *Proceedings of the IEEE Conference on Decision and Control*, pp. 7734–7739, 2013.
- [16] —, "An MPC approach to output-feedback control of stochastic linear discrete-time systems," available at arXiv:1408.6723, 2014.
- [17] S. Raimondi Cominesi, M. Farina, L. Giulioni, B. Picasso, and R. Scattolini, "Two-layer predictive control of a micro-grid including stochastic energy sources," in *2015 American Control Conference (ACC)*. IEEE, 2015, pp. 918–923.
- [18] K. Åström and B. Wittenmark, *Computer Controlled Systems: theory and design*. Prentice-Hall, 1984.
- [19] J. K. T. O. S. Pelland, J. Remund and K. D. Brabandere, "Photovoltaic and solar forecasting: state of the art," International Energy Agency, Tech. Rep., 2013.
- [20] F. Torrisi and A. Bemporad, "HYSDEL — A tool for generating computational hybrid models for analysis and synthesis problems," *IEEE Transactions on Control Systems Technology*, vol. 12, pp. 235–249, Mar. 2004. [Online]. Available: <http://control.ee.ethz.ch/index.cgi?page=publications;action=details;id=2223>
- [21] M. Farina, L. Giulioni, and R. Scattolini, "Stochastic linear model predictive control with chance constraints—a review," *Journal of Process Control*, vol. 44, pp. 53–67, 2016.
- [22] G. Schildbach, L. Fagiano, and M. Morari, "Randomized solutions to convex programs with multiple chance constraints," *SIAM Journal on Optimization*, vol. 23, no. 4, pp. 2479–2501, 2013.
- [23] X. Zhang, S. Grammatico, G. Schildbach, P. Goulart, and J. Lygeros, "On the sample size of random convex programs with structured dependence on the uncertainty," *Automatica*, vol. 60, pp. 182–188, 2015.
- [24] S. Raimondi Cominesi, A. Del Corno, E. Corsetti, G. A. Guagliardi, C. Sandroni, and R. Scattolini, "Thermal and energy production optimization in microgrids with model predictive control," in *International Conference on Renewable Energies and Power Quality (ICREPO)*, 2016.