

Increasing the Resilience of Cyber Physical Systems in Smart Grid Environments using Dynamic Cells

Andrea Tundis, Carlos García Cordero, Rolf Egert, Alfredo Garro, Max Mühlhäuser

Abstract—Resilience is an important system property that relies on the ability of a system to automatically recover from a degraded state so as to continue providing its services. Resilient systems have the means of detecting faults and failures with the added capability of automatically restoring their normal operations. Mastering resilience in the domain of Cyber-Physical Systems is challenging due to the interdependence of hybrid hardware and software components, along with physical limitations, laws, regulations and standards, among others. In order to overcome these challenges, this paper presents a modeling approach, based on the concept of Dynamic Cells, tailored to the management of Smart Grids. Additionally, a heuristic algorithm that works on top of the proposed modeling approach, to find resilient configurations, has been defined and implemented. More specifically, the model supports a flexible representation of Smart Grids and the algorithm is able to manage, at different abstraction levels, the resource consumption of individual grid elements on the presence of failures and faults. Finally, the proposal is evaluated in a test scenario where the effectiveness of such approach, when dealing with complex scenarios where adequate solutions are difficult to find, is shown.

Keywords—Cyber-physical systems, energy management, optimization, smart grids, self-healing, resilience, security.

I. INTRODUCTION

MANY systems are designed and developed by integrating and structuring existing components. These components, in turn, are systems on their own with their own inherent complexity and characteristics. It is not only through the behavior of each individual component, but also through the interaction of all systems that the behavior of the whole is defined [1]. If these complex heterogeneous systems aim at providing more functionality and better performance than what the individual components can provide, individual resources and capabilities need to be managed together. When mechanical and logical systems are joined together, a particular class of systems called Cyber-Physical System (CPS) emerge [2] [3]. In particular, in a CPS, sensors and embedded systems are networked together to monitor and manage a range of physical processes through a continuous feedback system [4]. Indeed, cyber components manage data and generate commands for controlling physical components. Ubiquitous examples of these systems are Smart Grids, Autonomous Automotive Systems, Medical Monitoring Equipment, and Automatic Pilot Avionics among others [5], [6], [7], [8], [9].

As discussed in [10], particularly interesting CPSs in everyday life of people are Smart Grids (SGs). An important

aspect of SGs is the management of their cyber-physical resources when the system is operating, both in ideal and problematic conditions [11]. Especially, the management of resources in critical infrastructures that are often tied to the well-being of humans. For example, in many places and situations such as in hospitals, if electricity is not available for a few hours, human lives are at risk. So, it is important to leverage the resources left available after faults or failures in any component in order to ensure the proper and continuous operations by preparing and adapting to changing conditions as well as to disruptions [12].

Such ability to withstand faults and failures is known as *resilience* [13]. More specifically, when resilience is contextualized in the field of SGs, a more resilient Smart Grid (SG) is a more reliable one where energy production and consumption is optimized to reduce costs and maximize supply [14], [15]. Resilience is traditionally achieved by duplicating components and connections within a system [16]. In this approach, the degree of resilience is measured by the number of redundant components. Nonetheless, the more redundant components, the more resources are wasted during normal operations. SGs are systems expected to become ubiquitous in the near future [17], as a consequence it is important to prepare these critical systems by giving a new significance to the term resilience. A resilient SG is one that may continue supplying its goods, complying as much as possible with given constraints, despite faults, failures or normal operation where resources are conserved as much as possible [18]. In order to achieve this new form of resilience, models describing the level of resilience of a SG need to be developed, along tools, simulation techniques and algorithms.

In this paper, a representation model that takes into account supply, demand and resources utilization in SGs to manage resilience, is presented. In particular, such model is centered on a optimization-based approach. It enables to manage the smart grid as a collection of micro grids where each one collaborates with each other to find an optimal arrangement of components within the system. The goal of the system is to generate energy and satisfy demand, while minimizing resource utilization, no matter the operational circumstances. In particular, an heuristic algorithm, capable of identifying efficiently feasible and flexible component arrangements within the SG which takes into account faults and failures, is proposed and exploited.

The rest of the paper is organized as follows. Section II discusses the objectives and research challenges tackled in this paper. Specifically, how to model SGs flexibly, using Dynamic Cells, is discussed in Section III. After proposing a model, in Section IV a heuristic for identifying system configurations

Andrea Tundis, Carlos García Cordero, Rolf Egert and Max Mühlhäuser are with the Telecooperation Lab, Technische Universität Darmstadt (E-mail: {tundis, garcia, egert, max}@tk.tu-darmstadt.de).

Andrea Tundis and Alfredo Garro are with DIMES, University of Calabria (e-mail: {andrea.tundis, alfredo.garro}@dimes.unical.it).

that are resilient is presented. Whereas, in Section V, a case study illustrates how both the proposed model and the heuristics work together. In Section VI the main related works are reported and discussed. Finally, conclusions and future perspectives are outlined.

II. OBJECTIVES AND RESEARCH CHALLENGES

As a CPS, a Smart Grid has a cyber part and a physical part [8]. The cyber part encompasses computing and networking resources, whereas the physical part is made up of physical processes such as mechanical and electrical. SGs aim at routing energy as efficient as possible meeting diverse cyber and physical requirements. SGs must (i) find an optimal balance of energy production for a dynamic demand, (ii) collect data from devices within the grid to manage and discover information, (iii) organize either small micro grids or continental-scale grids, and (iv) integrate heterogeneous devices ranging from big transformers and power plants to smart household appliances [19].

Traditional electricity grids rely on only a few sources that generate high voltage electricity. These sources distribute the energy along a hierarchical structure of transformers where the voltage of electricity is reduced up to the point where end-users can utilize it [5]. The failure of a single power source has the potential of causing partial or total blackouts. Blackouts cause problems to both suppliers and consumers. From the supplier point of view, monetary loss is incurred from not provisioning consumers. From the consumer side, the lack of electricity might have large negative consequences on human wellness and health. SGs try to solve the issue of having single points of failure by distributing the production of energy between many small suppliers and large suppliers while at the same time flattening the hierarchical structure of the energy network.

SGs have the goal of managing power distribution in a smarter manner. This goal is driven by a set of three objectives:

- *Objective 1:* Whenever possible, all consumers must be fully supplied with energy while at the same time limiting their electricity consumption.
- *Objective 2:* If, for whichever reason, consumers cannot be fully supplied, the energy grid must reorganize itself so as to provide a minimum provision of energy to as many consumers as possible.
- *Objective 3:* In the event of faults or failures, their impact on the energy supply should be minimized. Furthermore, measures to repair the system must be taken.

Designing resilient SGs is not a trivial task as also stated in [20]. Beyond the three objectives just mentioned, SGs need to satisfy other requirements related to multi-national laws, physical constraints or business decisions [21]. In [22], many more objectives to achieve resilience are stated in the form of what the authors call “*self-properties*”. In the context of this self-properties, the above mentioned objectives can be mapped to self-management, stabilization, healing, organization, optimization and configuration.

Electricity is difficult to manage. In contrast with network communication flows in computing devices, electricity cannot

be easily routed. If electricity is fed into a channel, it will try to go wherever lower resistance is found within the channel. A SG needs to control the direction and current (amount) of electricity flows. Because electricity grids typically operate with AC current, synchronizing the electrical output of producers becomes a necessity. In the context under consideration, it is assumed that the SG has *splitting points*, specific gates/switches, deployed throughout the energy distribution channels. The manipulation of these splitting points enables the SG to control the aforementioned properties of electricity flow. The indicated objectives are meant to be achieved by controlling not only how much electricity is consumed and produced, but also how electricity is routed when splitting points are opened or closed to enable or disable the flow of electricity. It is also assumed that groups of producers that have been isolated due to closing splitting points are capable of synchronizing their output of electricity.

Given the objectives and limitations previously mentioned and well discussed in [23], the following three main research challenges, which are dealt with in this paper, have been identified.

- *Research Challenge 1:* A novel modeling strategy to represent SGs with their inherent properties and limitations needs to be developed. The model must accommodate energy producers and consumers along with the restrictions that limit the flow of electricity using splitting points.
- *Research Challenge 2:* On top of a model, metrics for evaluating the fitness or goodness of a SG configuration need to be proposed. It is not sufficient to find adequate configurations where the supply of electricity is assured, it is also critical to identify configurations where resources are not wasted.
- *Research Challenge 3:* With a model and a set of metrics, rules and algorithms need to be developed in order to find optimal configurations of the main smart grid elements (producers, consumers and splitting points) given different environmental conditions.

In the following Sections the adopted model, approach and algorithm for facing with such research challenges are presented.

III. MODELING SMART GRIDS AS DYNAMIC CELLS

In this section, a novel model to represent SGs based on *Dynamic Cells* (DCs) is presented. A SG is a Dynamic Cell (DC) conceived as a stand-alone and recursive entity which is simultaneously a part of a bigger one [24]. The arrangement of entities creates hierarchies that have the ability of merging and splitting into other DCs. Each DC is responsible for its own stability and all DCs are responsible for finding an equilibrium in the system. Regardless of the depth of the hierarchies, each entity holds the same characteristics [25] that are useful in the context of SGs. In particular, (i) DCs achieve stability through the process of splitting and merging according to external conditions; (ii) although it is easy to identify independent DCs, there is no such thing as partial or incomplete DCs; furthermore (iii) DCs exist simultaneously as a part of a larger

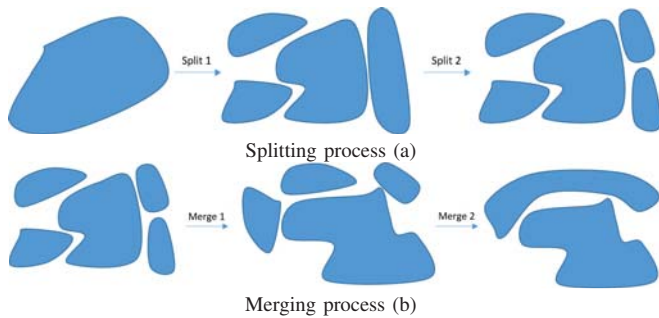


Fig. 1 Illustration of how Dynamic Cells merge and split

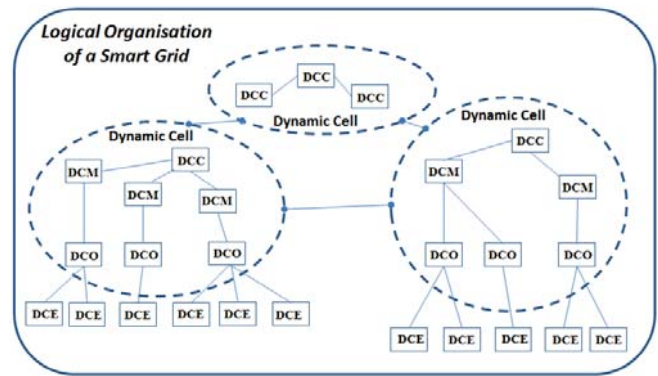


Fig. 2 Logical Organization of Smart Grids as Dynamic Cells.

International Science Index, Computer and Information Engineering Vol:11, No:1, 2017 waset.org/Publication/10007151

system and as individual systems that can sustain themselves. As a consequence, in optimal conditions, DCs tend to merge together to form large DCs (see Fig. 1 (b)). It is only through perturbation that DCs split in order to become stable (see Fig. 1 (a)). In this way, each DC becomes more resilient to perturbations as smaller DCs are easier to manage.

DCs can be utilized to organize and make SGs more resilient. Under normal conditions, the SG is treated as a single DC (optionally also as multiple DCs). When disruptions in the grid occur, the DC splitting process is used to isolate the affected parts and to reorganize its resources for the distribution of electricity. This approach enables the mitigation of negative effects in the whole system. Once the affected parts are restored to normal conditions, the system reorganizes itself by merging all affected parts into one.

In the next subsections the main concepts for enabling a Dynamic Cell-based representation of SGs are proposed. Different DC concepts to the energy sector are first mapped. Then the mathematical constraints that govern the interaction of the DC concepts are introduced. Finally, a strategy for realistically achieving this Dynamic Cell-based representation through the usage of splitting points are presented.

A. Conceptual Model and Design Constraints

To make the DC approach usable in the context of SGs, it is necessary to map different DC concepts to physical and logical components. In the proposed model, a *Dynamic Cell* is characterized by the following concepts:

- *DC-Elements* (DCE): basic physical components in the system able to generate (a battery, a solar panel, etc.) or consume (refrigerator, a television, hospital equipment, etc.) energy.
- *DC-Objects* (DCO): a set of DCEs managed by a single entity called *DC-Manager* (DCM). Examples of DCOs are houses, hospitals, or apartments. Every DCM can make decisions regarding the operation of its DCEs and can communicate with other DCM through higher-level entities called *DC-Coordinator* (DCC). A DCM itself has no information about the system outside the objects it manages. However, it reports the state of all its DCOs to its respective DCC.
- *DC-Coordinators* (DCC): a set of coordinators responsible for managing DCMs and communication

with other coordinators. They are responsible for the system-wide organization and coordination of DCMs.

Beside the key concepts introduced, specific evaluation indicators to assess the fitness of arbitrary configurations of DCOs, DCMs and DCOs have been defined. These indicators are useful to find system configurations that can satisfy the objectives presented in Section II. In particular, the main ones, named *Solution Quality Indicators* (*SQIs*), used in this research project/work are:

SQI₁: *Number of Existing DCs*: the goodness of the solution is characterized by the number of active DCs. In particular, since each DC can not provide surplus energy to other DCs, it is wasted. As a consequence the lower the number of DCs, the lower the total energy wasted/not used (due to the energy surplus of individual DCs). So less DCs indicate a better solution.

SQI₂: *Average DC Size*: the goodness of the solution is characterized by the average number of DC-Objects (or DC-Elements) associated to each DC. In this case the building of a possible solution tends to identify a system configuration with a balanced number of resources for each active DC. If this is the case, the difference of the size of each DC h and the average number of DCOs should not exceed a certain threshold Δ . In particular, $varh \leq \Delta$ where:

- Δ is a threshold determined by the number of DC-Objects,
- $varh = |\#DCObjects - \#AvgDCObjects|$,
- $\#AvgDCObjects = \frac{|DCO|}{|DCC_h|}$, and
- $|DCO_h|$ is the cardinality of DC-Objects that are logically managed by the DC h .

SQI₃: *DC Load Balancing*: in this case the quality of the solution is measured in terms of average energy wasted by each DC. In a good configuration, the difference of wasted energy E_h of each DC h should not exceed a certain threshold EG . In particular, $E_h \leq EG$, where:

- EG is a threshold in terms of wasted energy.
- E_h describes the energy wasted in a DC h which is defined as $EnergyProduced_h - EnergyConsumed_h$.

Unfortunately, the identification of suitable alternatives in case of dysfunctional behavior of the system is not trivial. Indeed, the identification of a solution in terms of system reconfiguration and therefore of energy redistribution among

resources in presence of faults and failures, is influenced by several factors. For example, priorities of single DC-Elements, or even of entire DC-Objects, according to specific types of energy supply contracts. For example, health care facilities, such as hospitals, have probably higher priority to be provided by energy than recreational facilities. Furthermore, providing supply for a cold-start of a power plant, which in turn provides additional supply, can be more useful for the overall system than supplying another consumer. From the other hand, elements/objects that are not supplied induce fines/penalty. To represent all these considerations in a more formal way, the following objective function and constraints are derived:

$$\max \sum_{j=1}^J \sum_{y=1}^Y \sum_{k=1}^K (g_{j,y} \cdot s_{j,y}) + (b_{j,y} \cdot p_{j,y}) + (e_{j,k} \cdot ep_k) \quad (1)$$

$$\sum_{j=1}^J \sum_{y=1}^Y (x_{i,j} \cdot p_{j,y} \cdot v_y) - (x_{i,j} \cdot s_{j,y} \cdot w_y) \leq EG \quad \forall i = 1, \dots, I \quad (2)$$

$$\sum_{i=1}^I x_{i,j} = 1, \quad \forall j = 1, \dots, J \quad (3)$$

$$s_{j,y} \in \{0, 1\}, \quad \forall j = 1, \dots, J, \quad \forall y = 1, \dots, Y \quad (4)$$

$$p_{j,y} \in \{0, 1\}, \quad \forall j = 1, \dots, J, \quad \forall y = 1, \dots, Y \quad (5)$$

$$x_{i,j} \in \{0, 1\}, \quad \forall i = 1, \dots, I, \quad \forall j = 1, \dots, J \quad (6)$$

$$e_{j,k} \in \{0, 1\}, \quad \forall j = 1, \dots, J, \quad \forall k = 1, \dots, K \quad (7)$$

Furthermore:

$$g_{j,y} \geq 0, \quad \forall j = 1, \dots, J, \quad \forall y = 1, \dots, Y \quad (8)$$

$$b_{j,y} \leq 0, \quad \forall j = 1, \dots, J, \quad \forall y = 1, \dots, Y \quad (9)$$

$$w_y \geq 0, \quad \forall y = 1, \dots, Y \quad (10)$$

$$v_y \leq 0, \quad \forall y = 1, \dots, Y \quad (11)$$

$$ep_k \leq 0, \quad \forall k = 1, \dots, K \quad (12)$$

$$EG \geq 0 \quad (13)$$

The parameters used in the previous constraint definitions are defined as follows:

- $i = 1, \dots, I$ are the possible number of activable *Dynamic Cells (DCs)* in the network.
- $j = 1, \dots, J$ are the possible number of *DC-Objects* within each *DC*.
- $y = 1, \dots, Y$ are the number of available *DC-Elements* that are electricity consumers.
- $k = 1, \dots, K$ are the number of available *DC-Elements* that are electricity producers.
- $g_{j,y}$ is the gain value when *DC-Element y* belonging to the *DC-Object j* is supplied.
- $s_{j,y}$ indicates if the *DC-Element y* belonging to the *DC-Object j* is actually supplied.
- $b_{j,y}$ is the penalty value when *DC-Element y* belonging to the *DC-Object j* is not supplied.
- $p_{j,y}$ indicates if *DC-Element y* belonging to the *DC-Object j* is actually not supplied.

- $x_{i,j}$ is the DC-DC-Object assignment variable.
- w_y is the energy consumed by the y -th *DC-Element*.
- v_y is the energy produced by the y -th *DC-Element*.
- $e_{j,k}$ provides information on which k -th *DC-Element* producer is associated to which j -th *DC-Object*.
- ep_k indicates the costs of using the k -th producing *DC-Element*.

Moreover, the objective function (1) aims at cumulating the gain of supplying *DC-Elements* as well as the penalty of not supplying other *DC-Elements*. Constraint (2) aims to guarantee that, the sum of the electricity consumed by the *DC-Elements* (that are actually supplied) and belonging to a *DC-Object j*, which in turn is part of the *DC i*, has to be no more than the sum of the energy produced inside the *DC i*; in particular, according to the *SQI₃*, the smaller the value of *EG*, the better the quality of the solution identified, as there is less waste of energy. Constraint (3) states that each *DC-Object j*, must be assigned to a single *DC i*. Whereas, (4), (5), (6) and (7) define the four sets of binary decision variables. Finally, the kind of parameters expected in input are defined in (8), (9), (10), (11), (12), (13).

B. Enabling Dynamic Cells through the Splitting Point Model

In the previous Section is mentioned that the flow of electricity seeks to cover each reachable part of the network. As a consequence, the concept of *Splitting Points*, which can be used to represent parts of the network to allow or deny supply for the area, has been introduced in Section II. Indeed, this physical split is also necessary to maintain network stability. As explained before *DC-Objects* contain *DC-Elements*, which in turn can be consumers as well as electricity producers (solar panels, wind turbines, etc.). In case the network is split into multiple *DCs*, the electricity producing *DC-Elements* in its *DC* starts supplying their part of the network. However, it is necessary for all producers in a network to be synchronized. To enable such behavior and to allow the *DCs* to stabilize themselves, if possible, it is necessary to have a complementary physical model (based on the Splitting Point concept) which allows to physically implement the reorganization of the system by providing management capabilities for the electrical flow.

In this regard, Fig. 3 (a) shows the logical model of a SG based on Dynamic Cells, whereas Fig. 3 (b) shows the same grid layout, where *Splitting Points (SPs)* indicate points where it is possible to separate the network. These *SPs* enable the SG to be divided as well as to be organized differently. An example of the extended SG is shown in Fig. 3 (c), with four *SPs* sp_1, sp_2, sp_3, sp_4 and six *DC-Objects* $dco_1, dco_2, dco_3, dco_4, dco_5, dco_6$.

Under normal operation conditions the SG is considered as a single DC. In case of failures of the power plant or when some attacks occur, the SG exploits the introduced *SPs* to split the network. In this way, it is able to better manage the resources in the network, isolate potential faulty parts that can be a threat to the system and ultimately stabilize the components. Potential example configurations (see Fig. 4) for the introduced grid can be the follows:

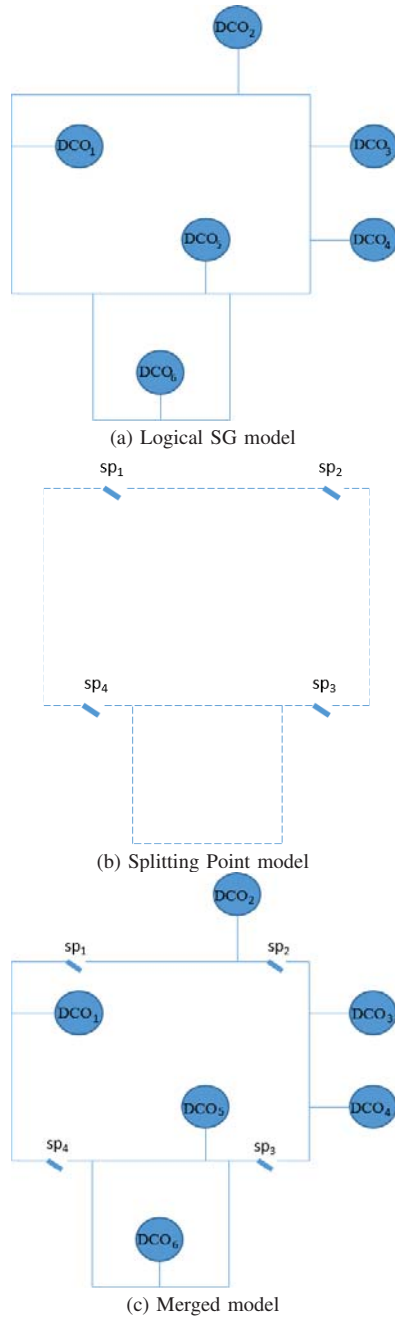


Fig. 3 Layers of the Dynamic Cells based model

Configuration 1: By using splitting point sp_2 and sp_4 the network is partitioned into two DCs. DC #1 contains the DC-Objects dco_1 and dco_2 and DC #2 consists of dco_3, dco_4, dco_5 and dco_6 .

Configuration N: By using the splitting points sp_1, sp_2 and sp_4 the network is partitioned into three DCs. DC #1 consists solely of dco_1 , DC #2 contains dco_2, dco_3 and dco_4 . DC #3 consists of dco_5 and dco_6 .

To benefit from the splitting ability it is necessary to define strategies and general rules of when and how DCs must split or merge. This is not the purpose of this work, but nevertheless some important guidelines are specifically identified for SG

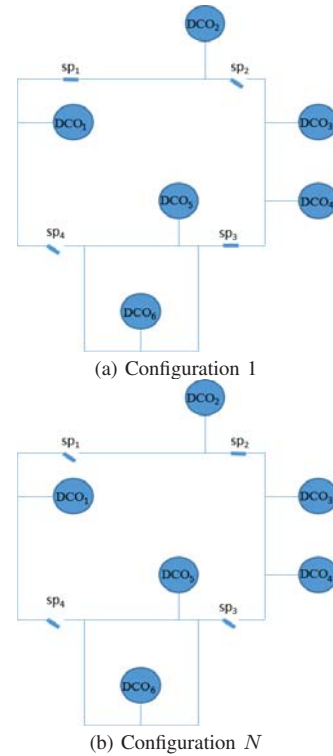


Fig. 4 Alternative Dynamic Cells configuration

environments.

A Smart Grid can be defined as $SG = \langle DC, DCO, DCE, SP, E \rangle$ where:

- DC a set of I Dynamic Cells $\{dc_1, \dots, dc_i, dc_d, \dots, dc_I\}$
- DCO a set of J DC-Objects $\{dco_1, \dots, dco_j, dco_h, \dots, dco_J\}$
- DCE a set of K DC-Elements $\{dce_1, \dots, dce_K\}$
- SP a set of F Splitting Points $\{sp_1, \dots, sp_F\}$ and,
- E is a set of C Edges $\{e_1, \dots, e_C\}$ such that $\{e_c \in E \mid e_c \subseteq (X, Y), \text{ where } X, Y \in \{DC, DCO, SP\} \wedge X, Y \notin (SP, SP)\}$

In order to define merging and splitting rules for a Smart Grid, the following predicates, along with their definition, have been introduced:

NEIGHBORHOOD. Neighborhood describes the physical closeness of two Dynamic Cells. For a more formal description we define a predicate *neighbours* that takes two Dynamic Cells $dc_j, dc_d \in DC$ as an input and outputs *true* only if there exists at least one splitting point $sp_f \in SP$ such that $\exists e_a, e_b \in E$ with (dc_j, sp_f) and (dc_d, sp_f) .

Definition 1 (Neighbours): Two Dynamic Cells dc_i, dc_j are neighbours if $neighbours(dc_i, dc_j) = true$.

DCO-MEMBERSHIP. Let *dco-membership* be a predicate that takes a DC-Object $dco_f \in DCO$ and a DC dc_j as an input and outputs *true* only if $dco_f \in dc_j \wedge \nexists dc_d \in DC$ with $dco_f \in dc_d \wedge d \neq j$.

Definition 2 (DCO-Membership): A DC-Object dco_f is member of a Dynamic Cell dc_j if $dco-membership(dco_f, dc_j) = true$.

DC-EXISTENCE. Describes the general existence of a Dynamic Cell dc_j . Let dc -exists be a predicate which takes a Dynamic Cell as an input and outputs *true* if there exists at least one DC-Object $dco_f \in h_j$.

Definition 3 (DC-Existence): A Dynamic Cell dc_j exists if dc -exists(dc_j) = *true*.

DC-MERGEABLE. Describes the capability of two Dynamic Cells being merged together. Let dc -mergeable be a predicate that takes two Dynamic Cells $dc_j, dc_d \in DC$ as an input and outputs *true* only if the following conditions hold:

- 1) $neighbours(dc_j, dc_d) = true$;
- 2) dc -exists(dc_j) = *true* \wedge dc -exists(dc_d) = *true*;
- 3) \exists at least one $dce_K \in DCE$ in either dc_j or dc_d that is not supplied, but the sum of the excess electricity provided by dc_j, dc_d is bigger or equal to the supply necessary for dce_K .

Definition 4 (DC-Mergeable): Two Dynamic Cells $dc_j, dc_d \in DC$ are mergeable if dc -mergeable(dc_j, dc_d) = *true*.

DCO-MERGEABLE. Describes the capability of placing two DC-Objects in one Dynamic Cell. Let dco -mergeable be a predicate that takes two DC-Objects dco_j, dco_h as an input and outputs *true* if

- 1) $\exists dc_j, dc_d$ with dc -mergeable(dc_j, dc_d) = *true*;
- 2) dco -membership(dco_j, dc_j) = *true* and dco -membership(dco_h, dc_d) = *true*.

Definition 5 (DCO-Mergeable): Two DC-Objects dco_j, dco_h are dco-mergeable if dco -mergeable(dco_j, dco_h) = *true*.

DC-UNDERSUPPLIED. A Dynamic Cell is undersupplied if it contains a DC-Element that is not supplied. Let dc -undersupplied be a predicate that takes a Dynamic Cell dc_j as an input and outputs *true* if $\exists dce_k, \exists dco_j$ such that $dce_k \in dco_j$ is not supplied and dco -membership(dco_j, dc_j) = *true*.

Definition 6 (DC-Undersupplied): A Dynamic Cell dc_j is dc-undersupplied if dc -undersupplied(dc_j) = *true*.

DCE-MEMBERSHIP. Let dce -membership be a predicate that takes a DC-Element dce_k and a Dynamic Cell dc_j as an input and outputs *true* if $\exists dco_j$ DC-Object such that $dce_k \in dco_j$ and dco -membership(dco_j, dc_j) = *true*.

Definition 7 (DCE-Membership): A DC-Element dce_k is dce-membership with a Dynamic Cell dc_j if dce -membership(dce_k, dc_j) = *true*.

SP-NEIGHBORHOOD. SP-Neighborhood describes the neighborhood of Dynamic Cells by having a direct connection to the same Splitting Point. Let sp -neighbours be a predicate that takes two Dynamic Cells dc_i, dc_j and a Splitting Point sp_k as an input and outputs *true* if $\exists SB_{i,j}$ Splitting Boundary, such as $SB_{k,i} = SB_{k,j} = 1$ where $i \neq j$.

Definition 8 (SP-Neighbours): Two Dynamic Cells $dc_i, dc_j \in DC$ are sp-neighbours via Splitting Point $sp_k \in SP$ if sp -neighbours(dc_i, dc_j, sp_k) = *true*.

IV. A HEURISTIC TO SOLVE THE DYNAMIC CELLS GROUPING PROBLEM

Integer programming is a NP-hard problem, as a consequence the identification of an optimal solution in the overall space of solutions could be very time consuming. To

achieve the objectives described in the previous section and cope with the problems of electricity resources and energy management, it is necessary to provide a way to manage the evolution of the network smartly. In this case typically, heuristic methods represent a good trade-off between the quality of the result and the time necessary to compute a solution. In the following, the pseudo-code of a defined heuristic algorithm, based on the concepts defined above, is presented.

In particular, it exploits a combination of different strategies to find feasible solutions for the merging and splitting of cells as previously discussed. This solution aims to manage SGs in a more intelligent and efficient way by taking demand and supply into account and mitigate potential risks.

Algorithm 1: Pseudocode for the general heuristic

Input: Smart Grid Description
Result: Returns a locally best solution for merging cells

```

1 boundaryTable  $\leftarrow$  buildInitialBoundaryTable();
2 strategy  $\leftarrow$  defaultResearchStrategy(boundaryTable);
3 currentSolution  $\leftarrow$  generateInitialSolution(strategy);
4 bestSolution  $\leftarrow$  currentSolution;
5 while stopCriteriaNotReached do
6     strategy  $\leftarrow$  chooseResearchStrategy(strategy);
7     candidateSolution  $\leftarrow$ 
8         generateCandidateSolution(strategy);
9     checkConstraints(candidateSolution);
10    if constraintsNotViolated() then
11        evaluateCurrentSolution();
12        if isSolutionImproved then
13            currentSolution  $\leftarrow$  candidateSolution;
14            bestSolution  $\leftarrow$  candidateSolution;
14 return bestSolution;
```

The reason for choosing heuristics is that, given a fixed time budget, it represents a different approach in the implementation of a computationally expensive activity such as simulation optimization. Heuristic methods represent the latest developments in the field of direct search methods that are frequently used for simulation optimization. Indeed, many of these techniques offer a good trade-off between global search for promising solutions within the entire feasible region (exploration) and the local search of promising sub-regions (exploitation), thereby resulting in efficient global search strategies as demonstrated in other different application domains [26].

A. Initialization Step

In case of an event that disturbs the overall equilibrium of the systems, like the failure of a power plant, the initialization phase is started. This phase consists of the first four steps of the code shown as algorithm 1 with the overall goal to rapidly generate a save and valid solution. To do this the system does not check the requirements but the solution is built in a way such that all the requirements are fulfilled. In particular, step 0 allows to identify:

- i) the maximum number of *cells* according to the splitting points present in the network;
- ii) the *neighbours* of each cell;
- iii) redundancies of splitting points in the network.

This is done by dividing the network into the maximum number of Dynamic Cells by opening all existing Splitting Points. To represent the correlation between Dynamic Cells and the Splitting Points in the network, the following algorithm is used to set the entries in the Splitting Boundary (SB) table.

Algorithm 2: GenerateInitialBoundaryTable: Pseudocode for generation the initial table *SB* with all switches open

```

1 foreach  $dc_i \in DC, i = 1, \dots, I$  do
2   foreach  $dco_j \in dc_i$  do
3     foreach  $sp_f \in SP, f = 1, \dots, F$  do
4       if  $\exists e_c \in E | e = (dco_j, sp_f)$  then
5          $SB_{f,i} = 1;$ 
    
```

The SB table is a way to represent the connection between Dynamic Cells via the existing splitting points in the network. In addition to the existence of a connection to the splitting points, it also contains information about their current status which can be either open (represented as 1) or closed (represented as 0). Note that open Splitting Points means that the Dynamic Cells normally connected by this Splitting Point are currently disconnected. If a table entry is empty this means that there is no edge between the Dynamic Cell and a potential Splitting Point. An example for an initial splitting boundary table is given in Table I.

TABLE I
 EXAMPLE OF THE INITIAL SPLITTING BOUNDARY TABLE *SB*

	dc_1	dc_2	...	dc_I
sp_1	1	1	...	1
sp_2	1	1	...	-
...
sp_F	-	-	...	-

By exploiting definition 8, (*SP-Neighbours*), it is also possible to identify redundant splitting points in the network. Redundancy is present if there exist two Dynamic Cells $dc_i, dc_j \in DC$ and a Splitting Point sp_k such that $sp_neighbours(dc_i, dc_j, sp_k) = true$ and $\exists sp_f$ with $f \neq k$ such that $sp_neighbours(dc_i, dc_j, sp_f) = true$.

After the splitting boundary table is built, the next three steps (from 2 to 4) of the algorithm 1, identify a valid initial solution which satisfies the system constraints. In particular this phase uses a bottom up approach to generate a solution based on two strategies:

- *Splitting Strategy*: Divide the overall system into the maximum amount of possible Dynamic Cells. This is done by using the information provided by the Splitting Boundary table.
- *Self-Containment Strategy*: For each Dynamic Cell all of its electricity producing DC-Elements are activated.

B. Stopping Criteria Step

This Step is represented as line 5 in the algorithm 1. It determines when the algorithm terminates and returns the result. In particular two specific stopping criteria have been considered:

- Countable units, such as iteration cycles of the algorithm or a specific amount of time (seconds, milliseconds, etc.).
- Achieving a specific threshold value for the solution (e.g a specific score which rates the quality of a solution).

C. Recovery Step

This is the most important phase of the heuristic and includes the logic represented in line six and seven of the algorithm 1. This steps aims to generate system configurations using different strategies. These configurations can ultimately be used to restore the grid up to a satisfactory state of operation. In the following, the proposed strategies are described.

Greedy Strategy: This strategy is centered around the concept of simply maximizing the fitness of the objective function. It is applied on each single Dynamic Cell and tries to switch on the elements that provide the maximum amount of fitness. This strategy employs a solving technique that is typically applied to the knapsack problem. Where given a set of items (in this case DC-Element), each with a weight (energy consumed when it is on) and a value (gain if it is working), determine which (DC-Element) to include in a collection (to keep working) so that the total weight (energy consumed) is less than or equal to a given limit (energy produced in the DC). This strategy is adopted as long as is possible to switch on DC-Elements in each single DC.

Merging Strategy: this strategy aims to identify pairs of DCs, from whose merging, there is an electricity overproduction, which allows to supply at least one more DC-Element. To enable this strategy, the DCs must be *dc-mergeble* (See section III-B).

Algorithm 3: Pseudo-code for the Merging strategy

```

1 foreach  $dc_i \in DC, with dc\_undersupplied(dc_i) = true$  do
2    $candidateNeighbours \leftarrow$  get all  $dc_d \in DC$  with
    $dc\_mergeble(dc_i, dc_d) = true;$ 
3   foreach  $h_k \in candidateNeighbours$  do
4     calculate energy excess  $e_k = e(dc_k)$  of Dynamic
     Cell  $dc_k;$ 
5   select  $dc_k \in candidateNeighbours$  with  $e_k =$ 
      $\max(e(dc_k) \in candidateNeighbours);$ 
6    $merge(dc_i, dc_k);$ 
    
```

Replacement Strategy: This strategy aims to find the optimal supply of elements in a DC to maximize the fitness of the objective function. This includes the decision to sacrifice supplied elements to use this electricity to switch on elements with a higher fitness value. Algorithm 4 shows the pseudo-code for this strategy.

Energy Saving Strategy: This strategy aims to maximize the use of energy producers in the network. It tries to switch off

Algorithm 4: Pseudo-code for the Replacement strategy

```

1 foreach  $dc_i \in DC$ , with  $dc\text{-undersupplied}(h_i) = true$  do
2   if  $\exists dce\text{-membership } dce_j, dce_k$  with  $dce_j$  is supplied,
    $dce_k$  is not supplied and  $j \neq k$  then
3     If the overall fitness would increase by supplying
      $dc_k$  and switching off  $dc_j$  then
4       switchOff( $dc_j$ );
5       switchOn( $dc_k$ );

```

energy producers for which their own supply costs exceed the sum of the gains from the objects that are supplied by this producer.

Algorithm 5: Pseudo-code for the Energy Saving strategy

```

1 foreach  $dc_i \in DC$  do
2   foreach  $dce_j \in DCE$  which produces electricity and
    $dce\text{-membership}(dce_j, dc_i) = true$  do
3      $sum \leftarrow$  sum of fitness of  $dce_d \in DCE$  with
      $dce\text{-membership}(dce_d, dc_i) = true$  and
     supplied by  $dce_j$ 
4     if costs of  $dce_j > sum$  then
5       switchOff( $dce_j$ );

```

D. Constraints Verification Step

This step refers to line 8 and 9 of the algorithm 1. This phase aims to check whether the solution generated in the previous steps is valid and respects the necessary constraints. More precisely there are two major constraints that need to be verified:

DC Load Balancing Constraint: The amount of energy a DC consumes must be equal or less than the sum of the energy that it produces. Algorithm 6 shows the pseudo-code for checking the load balancing constraints.

DC Object Memberships Constraint: Each DC-Object must be only assigned to a single DC. For checking this constraint it is enough to check if there exists a DC such that for each DC-Object the *DCO-Membership* is true.

E. Evaluation and Updating Step

This phase includes steps from 10 to 14. In particular, since the candidate solution, identified in the recovery step does not violate the constraints according to Constraints Verification phase it is checked if the value of the candidate solution is better than the value of the current solution. In case the solution improves the fitness value of the old one the candidate solution becomes the current solution and as a consequence the "current" best solution. Otherwise, if the value of the candidate solution is not better than the value of the current solution, then the current solution is kept and the research continues in order to find another candidate solution.

Algorithm 6: Pseudo-code for checking the Load Balancing constraints

```

input: CandidateSolution  $CS$  with a specific
configuration of Dynamic Cell  $DC$  and a
corresponding set of constraints  $C$ 
1 foreach  $dc_i \in DC$  do
2    $constraint - dc_i \in C \leftarrow false$ ;
3    $sum\_prod = 0$ ;
4    $sum\_cons = 0$ ;
5   foreach  $dco_j$  with  $dco\text{-membership}(dco_j, dc_i) =$ 
   true do
6     foreach  $dce_k \in dco_j$  do
7        $sum\_prod = sum\_prod + prod(dce_k)$ ;
8        $sum\_cons = sum\_cons + cons(dce_k)$ ;
9   if  $sum\_prod < sum\_cons$  then
10     $constraint - dc_i \leftarrow true$ ;
11 foreach  $constraint - dc_j \in C$  do
12   if  $constraint - dc_j = true$  then
13     $CS$  is an invalid solution;

```

TABLE II
DC-OBJECT – DC-ELEMENTS CONSUMERS ASSIGNMENT

	dce_1	dce_2	dce_3	dce_4	dce_5
dco_1	1	1	0	0	1
dco_2	0	0	1	0	1
dco_3	0	1	0	1	0
dco_4	1	0	1	1	1
...
dco_{10}	1	1	0	0	1

V. A CASE STUDY

In this section a small example of a potential SG is used to exemplify the feasibility of both: (i) the proposed Dynamic Cells-based model for its representation along with, (ii) the proposed heuristic algorithm for increasing the system resilience by managing its resources in case of faults or failures due to external attacks or natural events.

A. System Description

The SG under consideration is defined by ten DC-Objects ($J=10$) each of which includes some of the five DC-Elements ($Y=5$) described in table II. As an example, the DC-Object dco_1 is composed by the DC-Elements dce_1 , dce_2 and dce_5 ; whereas the DC-Object dco_2 is composed by the DC-Elements dce_3 , and dce_5 , etc. Furthermore, initially the net is supplied by a Main Source of Energy that can be for example a Power Plant. Table III provides information about each DC-Element consumer. In particular, it reports the energy consumed and the related gain when a DC-Element is supplied, as well as the penalty value when it is not supplied. Likewise, other two tables are used to describe the DC-Object – DC-Elements Producers assignment and the DC-Elements Producers description where there is Energy Produced (kWh) instead of Energy Consumed (kWh). Then, according to the approach proposed in Section 3 six Splitting

TABLE III
 DC-ELEMENTS CONSUMER DESCRIPTION

	Energy Consumed (kWh)	Gain	Penalty
dce_1	60	20	5
dce_2	100	50	20
dce_3	90	45	15
dce_4	120	48	18
dce_5	75	30	12

TABLE IV
 DC – DC-OBJECT - INITIAL CONFIGURATION

	dco_1	dco_2	dco_3	dco_4	dco_5	dco_6	dco_7	dco_8	dco_9	dco_{10}
dc_1	1	1	1	1	1	1	1	1	1	1
dc_2	0	0	0	0	0	0	0	0	0	0
dc_3	0	0	0	0	0	0	0	0	0	0
dc_4	0	0	0	0	0	0	0	0	0	0
dc_5	0	0	0	0	0	0	0	0	0	0

Points ($F=6$) are introduced so as to enable a dynamic cell based modeling (representation). By exploiting this set of physical devices (such as *switches*, a possible organization of the Smart Grid is shown in Figure 5. Furthermore the QI_3 metric is adopted according to the constrain (2), so as to identify the desired configuration. In this case, in the final configuration each Dynamic Cell dc_i cannot waste more than $EG=35$ kWh of energy. In the next subsection some iterations of the algorithm are shown.

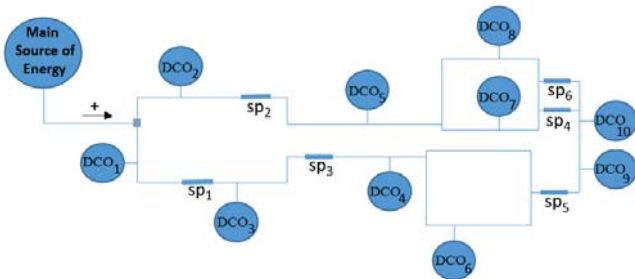


Fig. 5 Reference Scenario

B. Algorithm Execution

Table IV and V show the initial configuration of the network. Initially, all the Splitting Points are closed/connected (see Table V) and the Smart Grid works as a unique Cell (see Table IV), in which all the DC-Elements belonging to the DC-Objects of the net are supplied from the Main Source of Energy (e.g. Power Plant). In this case the following two tables are initialized as followed. In particular, Table IV shows that every DC-Objects belong to one DC and exactly to the same DC (dc_1), whereas Table V shows that all the Splitting Points are closed (value=1), so as to enabling the configuration reported in Table IV. Starting from this configuration when the Main Source of Energy fails, and as a consequence does not provide energy to the network, the algorithm combine the *Splitting Strategy* to identify a fast, valid and secure configuration and then the *Greedy Strategy*, by activating all the DC-Element producers, in order to supply as much as possible DC-Element consumers inside each DC. The resulting

TABLE V
 SPLITTING BOUNDARY TABLE – INITIAL CONFIGURATION

	dc_1	dc_2	dc_3	dc_4	dc_5
sp_1	1	1	-	-	-
sp_2	1	-	-	-	1
sp_3	-	-	1	1	-
sp_4	-	-	1	-	1
sp_5	-	-	1	1	-
sp_6	-	-	1	-	1

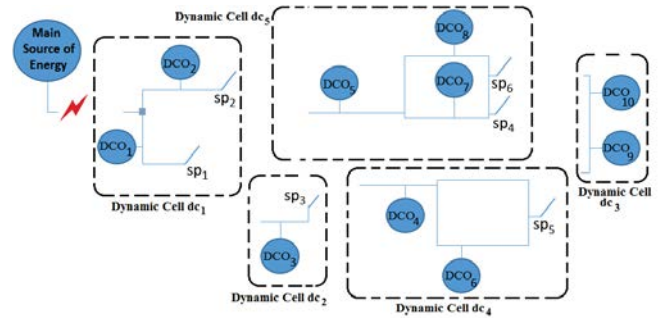


Fig. 6 Smart Grid split in the maximum DC size

configuration is shown in Figure 6, whereas Table VI and VII show respectively how DC-Objects (and as a consequence they DC-Elements) are distributed in the network and how such configuration is enabled by opening all the Splitting Points (value=0). Furthermore all the DC-Elements producers are turned on. By iterating on the Splitting Boundary table, it is guaranteed that many invalid configurations are avoided a priori such as the following represented in Tables VIII, since any direct connection (and as a consequence any Splitting Point) exists between the Dynamic Cells dc_1 and dc_3 . Whereas Figure 7 shows the final configuration of the system after some iterations, in which the initial DC dc_1 , dc_2 , dc_4 and dc_5 (along with their DC-Objects and DC-Elements) are merged in one Cell dc_{1245} ; whereas Dynamic the Cell dc_3 is still managed apart. This configuration is also represented in Table IX and that show the DC-Objects distribution and the management of the Splitting Points.

TABLE VI
 DC – DC-OBJECT – FIRST CONFIGURATION

	dco_1	dco_2	dco_3	dco_4	dco_5	dco_6	dco_7	dco_8	dco_9	dco_{10}
dc_1	1	1	0	0	0	0	0	0	0	0
dc_2	0	0	1	0	0	0	0	0	0	0
dc_3	0	0	0	0	0	0	0	0	1	1
dc_4	0	0	0	1	0	1	0	0	0	0
dc_5	0	0	0	0	1	0	1	1	0	0

TABLE VII
 SPLITTING BOUNDARY TABLE – FIRST CONFIGURATION

	dc_1	dc_2	dc_3	dc_4	dc_5
sp_1	0	0	-	-	-
sp_2	0	-	-	-	0
sp_3	-	-	0	0	-
sp_4	-	-	0	-	0
sp_5	-	-	0	0	-
sp_6	-	-	0	-	0

TABLE VIII
 INVALID CONFIGURATION

	dc_1	dc_2	dc_3	dc_4	dc_5	dc_6	dc_7	dc_8	dc_9	dc_{10}
dc_1	1	1	0	0	0	0	0	0	1	1
dc_2	0	0	1	0	0	0	0	0	0	0
dc_3	0	0	0	0	0	0	0	0	0	0
dc_4	0	0	0	1	0	1	0	0	0	0
dc_5	0	0	0	0	1	0	1	1	0	0

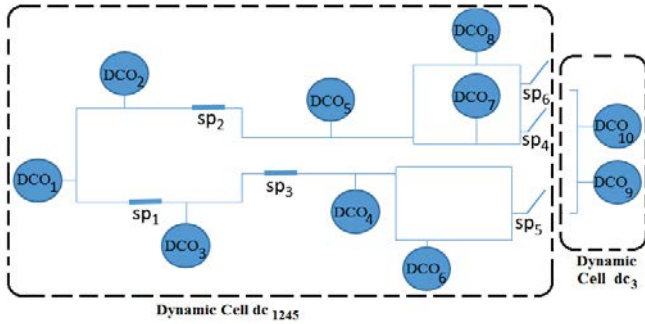


Fig. 7 A valid reconfiguration

C. Results Assessment

By considering the above mentioned configuration, it worth noting how the system evolves by try to find a possible solution that fulfills the objective functions avoiding the violation of the constrains. In particular, Figure 8 reports the evolution of the Objective Function evaluated by comparing a Random Search with the execution of the proposed Heuristic Algorithm. In particular, by using the proposed Heuristic it is interesting to notice that the value of the Objective function increases faster because the search space is limited and only a subset of configuration are evaluated, thanks to the combination of the different strategies adopted (as described in Section IV).

Whereas Fig. 9 shows how the wasted energy decreases by opportunely managing the resources (that produce and consume energy) in the Smart Grid, so as to increase its resilience in terms of efficiency. Whereas, as expected, Table XI provides information on the quality of the configuration identified in terms of produced and consumed energy and, as a consequence, wasted energy in each resulting Cell.

TABLE IX
 A VALID CONFIGURATION

	dc_1	dc_2	dc_3	dc_4	dc_5	dc_6	dc_7	dc_8	dc_9	dc_{10}
dc_1	1	1	1	1	1	1	1	1	0	0
dc_2	0	0	0	0	0	0	0	0	0	0
dc_3	0	0	0	0	0	0	0	0	1	1
dc_4	0	0	0	0	0	0	0	0	0	0
dc_5	0	0	0	0	0	0	0	0	0	0

TABLE X
 SPLITTING BOUNDARY TABLE – A VALID CONFIGURATION

	dc_1	dc_2	dc_3	dc_4	dc_5
sp_1	1	1	-	-	-
sp_2	1	-	-	-	1
sp_3	-	-	1	1	-
sp_4	-	-	0	-	0
sp_5	-	-	0	0	-
sp_6	-	-	0	-	0

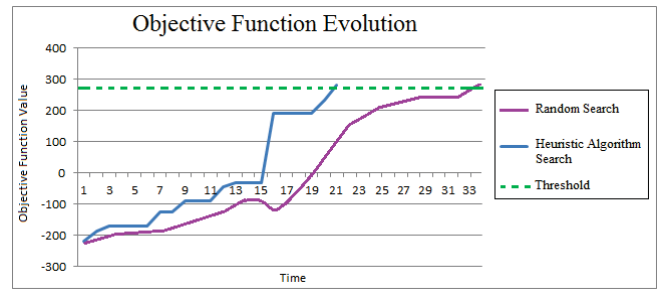


Fig. 8 Fashion of the Objective Function value

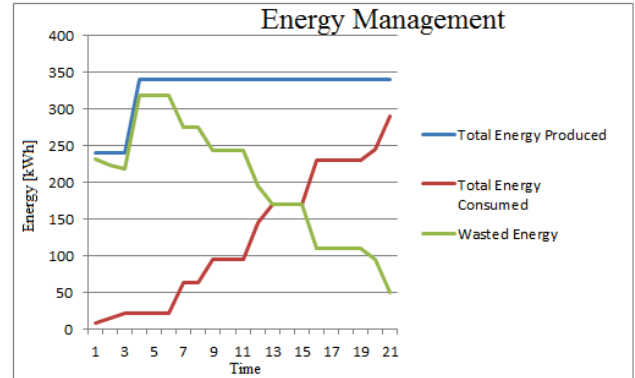


Fig. 9 Fashion of the energy value during the reconfiguration of the Smart Grid

VI. RELATED WORK

The management and control of a Smart Grid is not a trivial task. The introduction of a two-way communication infrastructure into the electrical grid yields many new control and management possibilities [27]. However, the combination of electrical grid and communication infrastructure also increases the number of possible failures and attacks on the network [28]. The Smart Grid aims to provide efficiency, reliability and safety, with smooth integration of renewable energy resources [29]. The modern Smart Grid is assumed to have a fully deployed monitoring and control infrastructure and a strong integration of distributed energy resources. The currently deployed architectures might not be usable when all these new elements are taken into account[30]. These problems mainly derive from the impact of distributed energy resources, the presence of controllable loads and quality constraints of the electrical grid.

Aside from the traditional hierarchical organized electrical grid, agent-based systems have gained a lot of attention. In general there are three different approaches to model the structure of Smart (Micro) Grids with the help of agents. A Smart Grid can be represented in a centralized manner where all the data gathered in the network is sent to a central server. This server has therefore a global view on the network and can make optimal decisions[30]. Alternatively it can be organized as a distributed system where local instances have to communicate and coordinate their actions. The third possibility is a hybrid approach, which aims at combining the aspects of distributed and centralized organization. The distributed and hybrid approaches have gotten a lot of attention since

TABLE XI
SMART GRID RECONFIGURATION ASSESSMENT BASED ON THE DC LOAD
BALANCING CONSTRAINT

Dynamic Cell	Produced Energy (kWh)	Consumed Energy (kWh)	Wasted Energy (kWh)
dc_{1245}	230	200	$30 \leq 35 = EG$
dc_3	110	90	$20 \leq 35 = EG$

their structure circumvents major flaws of the central approach and have additional benefits like increasing scalability and reducing energy consumption in the network [31]. A lot of work is based on these multi-agent systems (MAS) covering a variety of different tasks in Smart Grids. Control, management and monitoring mechanisms based on MAS are among others described in [21], [32], [33]. Ramchurn et al. used distributed self-organized agents to smooth demand peaks in the electrical grid by a demand side management approach which allows the consumers to adapt their consumption based on pricing information of the grid. Colson et al. and Vaccaro et al. made use of Smart Micro Grids to increase the resilience and self-healing capabilities of the overall network. In [34] and [35] hybrid strategies are described to increase resilience and to deploy self-healing mechanisms in case of errors in the Smart Grid. Distributed mechanisms for demand and response management are presented in [36], [37].

Unfortunately, despite numerous research efforts (mainly based on Multi Agent Systems), none of them can be fully exploited natively. Indeed, although Software Agents are natively distributed and are well suited to represent individual elements with capability of mobility and information exchanging, they cannot split or merge the network at different levels of granularity. Conversely, this work introduces a model based on the Dynamic Cells that aims at enabling such features in order to improve resilience mechanism in CPSs and, in particular, in Smart Grid environments.

VII. CONCLUSION AND FUTURE PERSPECTIVES

This paper has focused on the resilience of Smart Grids through the management of their resources in the event of malfunctions or disruptions. The main objective was to identify acceptable system configurations that would enable normal operations as much as possible.

A reference model based on the Dynamic Cells has been developed for representing the system organization as well as specific concepts for representing the system. The concept of *Splitting Points* have been used for extending the model and for enabling the dynamic evolution and the SG reconfiguration process. Moreover, centered on such model, a heuristic algorithm has been defined for automatically managing the resources of the Smart Grid in order to find resilient configurations.

Ongoing work concerns the development of a simulation environment and tools based on, both, the above presented Dynamic Cells-based model and the heuristic algorithm. The simulator and tools are aimed at enabling researchers to simulate and evaluate Smart Grid systems when faults or failures occur. The integration of intrusion detection

mechanisms, to cope with potential attacks to the model that might hamper resilience, is also under consideration. Finally, a wide experimentation and employment of these models not only to Smart Grids, but also to other CPSs application context is considered.

VIII. ACKNOWLEDGEMENTS

The work in this paper was partially performed in the context of the PolyEnergyNet (PEN) research project and partially funded by the Germany Federal Ministry for Economic Affairs and Energy (BMWi) under grant no. "0325737E".

REFERENCES

- [1] D. Wachholder and C. Stary, "Enabling emergent behavior in systems-of-systems through bigraph-based modeling," in *System of Systems Engineering Conference (SoSE), 2015 10th*, may 2015, pp. 334–339.
- [2] I. S. Danda B. Rawat Joel J.P.C. Rodrigues, *Cyber-Physical Systems: From Theory to Practice*. CRC Press, 2015.
- [3] J. Shi, J. Wan, H. Yan, and H. Suo, "A survey of Cyber-Physical Systems," *2011 International Conference on Wireless Communications and Signal Processing, WCSP 2011*, 2011.
- [4] K. G. S. L. M. P. Gaddadevara Matt Siddesh Ganesh Chandra Deka, *Cyber-Physical Systems: A Computational Perspective*. Chapman and Hall/CRC, 2015.
- [5] N. B. M. Isa, T. C. Wei, A.Yatim, and A. H. M. Yatim, "Smart grid technology: Communications, power electronics and control system," in *International Conference on Sustainable Energy Engineering and Application (ICSEEA)*. IEEE, oct 2015, pp. 10–14. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7380737>
- [6] K. Sampigethaya, R. Poovendran, and T. Br, "Cyber-physical integration in future aviation information systems," in *2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC)*. IEEE, 2012, pp. 7C2—1.
- [7] J. Plourde, D. Arney, and J. M. Goldman, "OpenICE: An open, interoperable platform for medical cyber-physical systems," *2014 ACM/IEEE International Conference on Cyber-Physical Systems, ICCPS 2014*, p. 221, 2014.
- [8] S. Karnouskos, "Cyber-physical systems in the SmartGrid," *IEEE International Conference on Industrial Informatics (INDIN)*, pp. 20–23, 2011.
- [9] L. Rogovchenko-Buffoni, A. Tundis, M. Z. Hossain, M. Nyberg, and P. Fritzson, "An Integrated Toolchain For Model Based Functional Safety Analysis," *Journal of Computational Science*, vol. 5, no. 3, pp. 408–414, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.jocs.2013.08.009>
- [10] Y. Strengers, "Smart Energy in Everyday Life: Are You Designing for Resource Man?" *interactions*, vol. 21, no. 4, pp. 24–31, jul 2014. [Online]. Available: <http://doi.acm.org/10.1145/2621931>
- [11] H. H. Yan, J. F. Wan, and H. Suo, "Adaptive Resource Management for Cyber-Physical Systems," in *Mechatronics and Applied Mechanics*, ser. Applied Mechanics and Materials, vol. 157. Trans Tech Publications, 2012, pp. 747–751.
- [12] P. Smith, D. Hutchison, J. P. G. Sterbenz, M. Schöller, A. Fessi, M. Karaliopoulos, C. Lac, and B. Plattner, "Network resilience: A systematic approach," *IEEE Communications Magazine*, vol. 49, no. 7, pp. 88–97, 2011.
- [13] G. Denker, N. Dutt, S. Mehrotra, M.-O. Stehr, C. Talcott, and N. Venkatasubramanian, "Resilient dependable cyber-physical systems: a middleware perspective," *Journal of Internet Services and Applications*, vol. 3, no. 1, pp. 41–49, 2012. [Online]. Available: <http://dx.doi.org/10.1007/s13174-011-0057-4>
- [14] M. Amin, "Challenges in reliability, security, efficiency, and resilience of energy infrastructure: Toward smart self-healing electric power grid," in *Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE*, jul 2008, pp. 1–5.
- [15] N. A. S. Abdullah, N. L. M. Noor, and E. N. M. Ibrahim, "Resilient organization: Modelling the capacity for resilience," *International Conference on Research and Innovation in Information Systems, ICRIS*, vol. 2013, pp. 319–324, 2013.

- [16] X. Zhao, Z. Zhou, Z. Li, and Z. Qin, "Redundancy deployment strategy based on energy balance for wireless sensor networks," in *Communications and Information Technologies (ISCIT), 2012 International Symposium on*, oct 2012, pp. 702–706.
- [17] E. Rodriguez-Diaz, J. C. Vasquez, and J. M. Guerrero, "Intelligent DC Homes in Future Sustainable Energy Systems: When efficiency and intelligence work together," *IEEE Consumer Electronics Magazine*, vol. 5, no. 1, pp. 74–80, jan 2016.
- [18] P. Smith and A. Schaeffer-Filho, "Management Patterns for Smart Grid Resilience," in *Service Oriented System Engineering (SOSE), 2014 IEEE 8th International Symposium on*, apr 2014, pp. 415–416.
- [19] D. Seo, H. Lee, and A. Perrig, "Secure and Efficient Capability-Based Power Management in the Smart Grid," in *Parallel and Distributed Processing with Applications Workshops (ISPAW), 2011 Ninth IEEE International Symposium on*, may 2011, pp. 119–126.
- [20] A. Pahwa, S. A. DeLoach, B. Natarajan, S. Das, A. R. Malekpour, S. M. Shaful Alam, and D. M. Case, "Goal-Based Holonic Multiagent System for Operation of Power Distribution Systems," *IEEE Transactions on Smart Grid*, vol. 6, no. 5, pp. 2510–2518, 2015.
- [21] C. M. Colson, M. H. Nehrir, and R. W. Gunderson, "Distributed multi-agent microgrids: A decentralized approach to resilient power system self-healing," *Proceedings - ISRCS 2011: 4th International Symposium on Resilient Control Systems*, pp. 83–88, 2011.
- [22] A. Berns and S. Ghosh, "Dissecting Self- * Properties," *2009 Third IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, pp. 10–19, 2009.
- [23] D. E. Olivares, A. Mehrizi-Sani, A. H. Etemadi, C. A. Cañizares, R. Iravani, M. Kazerani, A. H. Hajimiragha, O. Gomis-Bellmunt, M. Saeedifard, R. Palma-Behnke, G. A. Jiménez-Estévez, and N. D. Hatziargyriou, "Trends in Microgrid Control," *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1905–1919, 2014.
- [24] A. Koestler, *The Ghost in the Machine*. Macmillan, 1967.
- [25] M. Calabrese, A. Amato, V. di Lecce, and V. Piuri, "Hierarchical-granularity holonic modelling," *Journal of Ambient Intelligence and Humanized Computing*, vol. 1, no. 3, pp. 199–209, 2010.
- [26] P. Legato and R. Mazza, "A Simulation Optimization based Approach for Team Building in Cyber Security," *Journal of Simulation and Process Modelling*, 2015.
- [27] M. Erol-Kantarci and H. T. Mouftah, "Energy-Efficient Information and Communication Infrastructures in the Smart Grid: A Survey on Interactions and Open Issues," *IEEE Communications Surveys and Tutorials*, vol. 17, no. 1, pp. 179–197, 2015.
- [28] A. Hahn and M. Govindarasu, "Cyber attack exposure evaluation framework for the smart grid," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 835–843, 2011.
- [29] C. Cecati, C. Citro, A. Piccolo, and P. Siano, "Smart operation of wind turbines and diesel generators according to economic criteria," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4514–4525, 2011.
- [30] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "Smart Grid Technologies: Communication Technologies and Standards," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 4, pp. 529–539, nov 2011. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6011696>
- [31] R. M. Oviedo, F. Ramos, S. Gormus, P. Kulkarni, and M. Sooriyabandara, "A comparison of centralized and distributed monitoring architectures in the smart grid," *IEEE Systems Journal*, vol. 7, no. 4, pp. 832–844, 2013.
- [32] S. Ramchurn, P. Vytelingum, A. Rogers, and N. Jennings, "Agent-Based Control for Decentralised Demand Side Management in the Smart Grid," *AAMAS '11, Taipei*, pp. 5–12, 2011. [Online]. Available: <http://eprints.soton.ac.uk/271985/>
- [33] A. Vaccaro, V. Loia, G. Formato, P. Wall, and V. Terzija, "A Self-Organizing Architecture for Decentralized Smart Microgrids Synchronization, Control, and Monitoring," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 1, pp. 289–298, feb 2015. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6863653>
- [34] S. Chouhan, J. Ghorbani, H. Inan, A. Feliachi, and M. A. Choudhry, "Smart MAS restoration for distribution system with Microgrids," *IEEE Power and Energy Society General Meeting*, 2013.
- [35] A. Felix, H. S. V. S. K. Nunna, S. Doolla, and A. Shukla, "Multi agent based restoration for smart distribution system with microgrids," *2015 IEEE Energy Conversion Congress and Exposition, ECCE 2015*, pp. 2341–2347, 2015.
- [36] A. Barbato, A. Capone, L. Chen, F. Martignon, and S. Paris, "A distributed demand-side management framework for the smart grid," *Computer Communications*, vol. 57, pp. 13–24, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.comcom.2014.11.001>
- [37] R. Deng, Z. Yang, F. Hou, M.-Y. Chow, and J. Chen, "Distributed Real-Time Demand Response in Multiseller–Multibuyer Smart Distribution Grid," *IEEE Transactions on Power Systems*, vol. 30, no. 5, pp. 2364–2374, 2015.