A Semantic Middleware Supported Receding Horizon Optimal Power Flow in Energy Grids

***Abstract*—Energy management in electric grids with multiple energy sources, generators, storage devices, and interacting loads along with their complex behaviours requires grid wide control. Communication infrastructure that aggregates information from heterogeneous devices in the electric grid making the applications completely independent of physical connectivity are essential for building in the context of control applications. This investigation presents a semantic middleware that is used to implement a receding horizon based optimal power flow in smart grids. The presence of renewable energy sources, storage systems and loads dispersed all along the grid necessitates the use of grid wide control and a communication infrastructure to support it. To this extent the proposed middleware will serve as the basis for representing various components of the power grid. It is enriched with intelligence by semantic annotation and ontologies that provide situation awareness and context discovery. The middleware deployment is demonstrated by implementing the receding horizon OPF in a network in Steinkjer, Norway. Our results demonstrate the advantages of both the middleware and the algorithm. Furthermore, the results prove the added flexibility obtained in the grid due to the addition of renewable energy and storage systems. The significant advantage of the proposed approach is that the real-time monitoring infrastructure is used for improving the flexibility, reliability and efficiency of the grid.**

***Index Terms*—Middleware, Optimal Power Flow (OPF), Middleware Architecture, Renewable Energy, Energy Storage Systems, Multiple Energy Systems (MES), Receding Horizon Approach.**

NOMENCLATURE

Variables:

P_i^g, Q_i^g	Active and reactive power generated at bus i [p.u., p.u.];
P_{ij}	Active power injected by node i on (i,j) [p.u.];
$P_{(i,j)}^{\text{loss}}$, $Q_{(i,j)}^{\text{loss}}$	Active and reactive power loss on (i,j) [p.u., p.u.];
V_i, θ_i	Voltage magnitude and angle at bus i [p.u., rad];
b_i	State of Charge (SOC) of the storage unit at bus i [p.u.h];
r_i	Power exchanged with the storage unit at bus i [p.u.] (positive when discharging);

Parameters:

$y_s = g_s - \mathrm{j}b_s$	Complex series admittance for a transmission line [p.u.];
b_s, g_s	Series susceptance and conductance for a transmission line [p.u.];
b_c	Shunt susceptance [p.u.];
τ, θ^{sh}	Transformer tap ratio and phase shift angle [., rad];
P_i^d, Q_i^d	Active and reactive power load at bus i [p.u., p.u.];
P_i^r, Q_i^r	Renewable active and reactive power at bus i [p.u., p.u.];
$\underline{P}_i^g, \overline{P}_i^g$	Generator active power bounds at bus i [p.u.];
$\underline{Q}_i^g, \overline{Q}_i^g$	Generator reactive power bounds at bus i [p.u.];
R_i	Ramp rate limit of the generator at bus i [%].
$\overline{S}_{ij}$	Rating of line (i,j) [p.u.];
$\underline{V}_i, \overline{V}_i$	Minimum and maximum voltage magnitudes at bus i [p.u.];
$\underline{\theta}_i, \overline{\theta}_i$	Minimum and maximum phases, bus i [rad];
$\underline{B}_i, \overline{B}_i$	Minimum and maximum storage unit i capacity [p.u.h];
r_i^{loss}	Storage energy loss at bus i [p.u.];
r_i^{rated}	Maximum power supplied by the storage i [p.u.];
C_i	Linear cost term of generator i [€h/p.u.];
C_i^g	Generation cost at bus i [€];
C_i^b	Storage cost at bus i [€h^{-1}/p.u.];
H, T	Control horizon and simulation time [h];
ΔT	Time step [h].

I. INTRODUCTION

OPTIMAL Power Flow (OPF) is tool for both energy management and planning in energy grids. It dispatches available generation for optimizing an objective (e.g. reduce line losses) while respecting operating and physical constraints. Traditional OPF solves a static optimization problem to find the generator settings that will optimize grid operating cost or line-losses (see, for example [1]). The addition of newer components, namely the renewable generation from multiple energy sources (MES) and energy storage systems (ESS), necessitates dynamic optimization due to the intermittent generation and complex behaviours arising from their interconnection [2], [3]. To this extent, dynamic OPF was proposed that solves an optimization problem for a specified period considering the generation from MES and ESS [4]), and then repeats the computation during next periods. As there are frequent fluctuations in generation and demand the grid becomes vulnerable to the estimates of the generation and demand. To overcome this shortcoming, predictive OPF methods using receding horizon (RH) approach have been proposed recently [5]. The receding horizon approach is more robust to disturbances as it uses the forecasts and current measurements for computing the optimal network operating states.

Performing predictive OPF requires automation support to perform grid wide monitoring and control. The current levels of automation in power grids are more restricted to substation level. Other technical challenges in this direction include

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2017.2655047, IEEE Transactions on Industrial Informatics

interoperability among devices, presence of various vendor specific standards, protocols and data formats. Furthermore, control applications such as receding horizon OPF require support of third party applications such as forecasting or state estimation. In many instances, the OPF might well be developed by an independent vendor and used by the grid operator. Consequently, predictive OPF implementation in energy grids requires communication infrastructure support that makes the application totally oblivious to physical connectivity.

The problem of designing communication infrastructure for energy grids has been the focus of research for some time now [6], [7]. Widely studied middleware architecture is the GridStat [8] that aims to provide flexible communication to smart energy grids. It provides network resources to provide low-latency, reliable delivery of information and provides multi-cast feature, and uses a publish-subscribe model suitable for smart grid applications. In spite of being a mature technology, it lacks semantic capabilities. The service oriented architecture based middleware proposed by Zhou and Rodrigues [9] considers different end-user services (smart meters, power station outage management etc.) and therefore is more oriented towards the user, rather than device. Garcia et al. [10] proposed an intelligent agent based distributed architecture that uses a publish-subscribe mechanism too. The authors suggest the use of enterprise service bus to solve interoperability issues. The semantic features mentioned in the work do not really mean semantic annotated services. Sucic et al. [11] suggest an integration of IEC 61850 and device profile for web services. The authors highlight the semantic capabilities of IEC 61850, which makes use of the mechanism called Abstract Communication Services Interface or ASCI to establish a link between the abstract services of IEC 61850 and application layer-related implementations as Device Profiles for Web Services (DPWS) and Multimedia Message Service (MMS). Despite the addition of some "semantic" component, there are several weaknesses in the proposal such as the use of request-response mechanism of DPWS due to the dependence on web services and this is unsuitable for event driven system such as the smart grid. Moreover, the investigation does not discuss the semantic mechanisms, such as ontologies for device and service descriptions and annotations, or a language to create ontologies as OWL. More recently SmarC [12], a middleware proposal for the smart grid with the ability to process the data gathered from different elements of the grid, and use it insulate the applications from the complexity of the metering facilities, was proposed. The solution guarantees that any changes that happen in the lower levels will be updated for the future actions of the system in a seamless way for applications.

Other smart grid proposals include CoSGrid (Controlling the Smart Grid) [13]: Self-Organizing smart grid services [14], a middleware architecture based on data collection using radio frequency identification [15]; Unified solution for advanced metering infrastructure [16] which contains an object oriented communication middleware and uses CORBA for the middleware architecture. However, these proposals lack information on semantic features and are therefore not suitable for the smart grid applications. Thus a review of literature reveals that the current middleware proposals lack semantic capabilities. Instead, significant intelligence and situational awareness can be obtained by providing semantic capabilities. Moreover, such a middleware should be more oriented towards the user requirement. Furthermore, a communication infrastructure and a receding horizon OPF implementation in an energy grid that considers a holistic approach in treating both information and energy flows has not been reported in literature. Moreover, a demonstration of the semantic middleware and receding horizon based OPF on a real network has not been reported to our best knowledge.

This investigation presents a middleware with semantic capabilities that enhances the intelligence of the existing communication infrastructure and a new way of implementing OPF that is best suited for energy grids having fluctuating generation from multiple sources and demand. To this aim the *I3RES middleware* developed within the European FP7 research project "ICT Intelligent Management of the RES optimal Operations" that organizes the energy management into a layered architecture. The top level of the architecture is the applications layer hosting the third party control applications such as the OPF. The lower layer is the physical layer having the heterogeneous devices. The middleware layer provides the interface between lower level devices to the application in real-time and thus serves as a communication infrastructure that enables grid wide control. To build intelligence into the middleware, ontology, semantic annotation and inference engine are added. This gives the middleware features such as service discovery and context awareness. At the control level, the benefits of adding forecasts on renewable generation and demand, the flexibility and the cost efficiency with the use of multiple energy sources within the grid are observed from the results. Further, the implementation of the receding horizon based OPF supported by a semantic middleware in real energy network is reported and the results are provided.

The paper is organized as follows. Section II presents the multi-objective receding horizon optimal power flow formulation and the DC approximation of the problem. The I3RES middleware architecture and definition is discussed in Section III. While Section IV discusses the semantic aspects of the middleware. Section V presents the implementation, deployment results of the middleware and ACOPF algorithm in a Norwegian microgrid. The results obtained illustrate the need to solve the ACOPF approach. Conclusions and future directions of research are discussed in Section VI.

II. Multi-Objective Receding Horizon Based OPF

A. Receding Horizon Based ACOPF

The receding horizon based ACOPF problem is the optimal operation schedule for the network that takes decision on the quantity of power to be produced from various generators connected to the grid and exchanged with storage devices each hour to minimize the grid operating cost and line-losses, while respecting the load balance, physical and operating constraints.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2017.2655047, IEEE Transactions on Industrial Informatics

The objectives can be modelled as

$$f^E(t) = C_0(t) + \sum_{i=1}^{N} \left(C_i^g(t) + C_i^b(t) \right), \tag{1a}$$

$$f^L(t) = \sum_{(i,e)\in\mathcal{E}} P_{(i,j)}^{\text{loss}}(t), \tag{1b}$$

where $f^E(t)$ and $f^L(t)$ are the grid operating cost and line losses, respectively. Additionally, the cost C_0 models the cost due to the utility and is given by

$$C_0 = c_0(t) P_0^g \tag{2}$$

where c_0 is actually time varying and depends on the day-ahead energy prices provided by the grid based on market clearing prices. To account for the bi-directional power flows, a negative lower bound $\underline{P}_0^g \leq 0$ on the injected active power is employed. A negative value of P_0^g indicates selling to utility leading to negative contribution to the OPF objective function, and vice-versa.

At each time step $\hat{t}$ we are given the initial SoC $b_i(\hat{t})\ \forall i$ and forecast of load $P_i^d(t),\ Q_i^d(t)\ \forall i,\ t \in [\hat{t}, \hat{t}+H]$ and generation $P_i^r(t),\ Q_i^r(t)\ \forall i,\ t \in [\hat{t}, \hat{t}+H]$. The RH based controller computes the optimal input sequence $\mathcal{U}^{\hat{t}} = \left(U_0^{\hat{t}}, U_1^{\hat{t}}, U_2^{\hat{t}}, \ldots, U_H^{\hat{t}} \right)$, where

$$U_t^{\hat{t}} = \left(P_i^g(\hat{t}+t), Q_i^g(\hat{t}+t), V_i(\hat{t}+t), \theta_i(\hat{t}+t), r_i(\hat{t}+t) \right), \tag{3}$$

by solving the finite-horizon optimal control problem:

$$\begin{aligned}
&\min_{\mathcal{U}^{\hat{t}}} \ J^{\hat{t}} \\
&\text{s.t.} \\
&P_{ij}(t) = \mathrm{Re}(V_i^2(t) y_{ii}^* + V_i(t) V_j(t) e^{\mathrm{j}\theta_{ij}(t)} y_{ij}^*), && \forall (i,j) \in \mathcal{E}, \forall t, \\
&Q_{ij}(t) = \mathrm{Im}(V_i^2(t) y_{ii}^* + V_i(t) V_j(t) e^{\mathrm{j}\theta_{ij}(t)} y_{ij}^*), && \forall (i,j) \in \mathcal{E}, \forall t, \\
&P_i^g(t) + P_i^r(t) - \sum_{j\in\mathcal{N}_i} P_{ij}(t) + r_i(t) = P_i^d(t), && \forall i, t, \\
&Q_i^g(t) + Q_i^r(t) - \sum_{j\in\mathcal{N}_i} Q_{ij}(t) = Q_i^d(t), && \forall i, t, \\
&|P_i^g(t+1) - P_i^g(t)| \leq R_i \Delta T && \forall i,\ \forall t \\
&S_{ij}^2(t) = P_{ij}^2(t) + Q_{ij}^2(t) \leq \overline{S}_{ij}^2, && \forall (i,j) \in \mathcal{E}, \forall t, \\
&b_i(t+1) = b_i(t) - \Delta T \left(r_i(t) + r_i^{\text{loss}} \right), && \forall i, t, \\
&-r_i^{\text{rated}} \leq r_i(t) \leq r_i^{\text{rated}}, && \forall i, t, \\
&\underline{B}_i \leq b_i(t) \leq \overline{B}_i, && \forall i, t, \\
&\underline{P}_i^g \leq P_i^g(t) \leq \overline{P}_i^g, \ \ \underline{Q}_i^g \leq Q_i^g(t) \leq \overline{Q}_i^g, && \forall i, t, \\
&\underline{V}_i \leq V_i(t) \leq \overline{V}_i, \quad \underline{\theta}_i \leq \theta_i(t) \leq \overline{\theta}_i, && \forall i, t, \\
&P_i^g(0) = P_{i,0}^g && \forall i, \\
&b_i(0) = b_{i,0} && \forall i,
\end{aligned} \tag{4}$$

where the objective function $J^{\hat{t}}$ is a multi-objective function consisting of the weighted sum over $\phi \in [0,1]$,

$$J^{\hat{t}} := \sum_{t=\hat{t}}^{\hat{t}+T} \phi f^E(t) + (1-\phi) f^L(t). \tag{5}$$

According to the receding horizon approach, once the optimization problem (4) has been solved at time $\hat{t}$ for a control horizon of H time slots, only the inputs at time $\hat{t}$ are applied to the grid, whereas the other computed control inputs are discarded. When the time horizon shifts (i.e. $\hat{t} \leftarrow \hat{t}+1$), (4) is repeated with new measured and/or estimated state and new forecasted generation and load. This optimal feedback policy allows to potentially compensate for any disturbance that has acted on the grid between two consecutive time steps.

B. Receding Horizon Based Direct Current OPF

The classical AC Optimal Power Flow (ACOPF) is a static nonlinear and nonconvex problem and it leads to computation issues. The computation can be simplified using a DC model of the network [17] as:

$$\begin{aligned}
&\min_{\mathcal{U}^{\hat{t}}} \ J^{\hat{t}} \\
&\text{s.t.} \\
&P_i(t) = -\sum_{j=1}^{N} \frac{1}{x_{ij}} \theta_{ij}(t), && \forall i, t, \\
&\theta_{ij}(t) \leq \overline{\theta}_{ij}, && \forall (i,j) \in \mathcal{E}, \forall t, \\
&P_i^g(t) + P_i^r(t) - \sum_{j\in\mathcal{N}_i} P_{ij}(t) + r_i(t) = P_i^d(t), && \forall i, t, \\
&|P_i^g(t+1) - P_i^g(t)| \leq \Delta T R_i, && \forall i,\ \forall t \\
&b_i(t+1) = b_i(t) - \Delta T \left(r_i(t) + r_i^{\text{loss}} \right), && \forall i, t, \\
&-r_i^{\text{rated}} \leq r_i(t) \leq r_i^{\text{rated}}, && \forall i, t, \\
&\underline{B}_i \leq b_i(t) \leq \overline{B}_i, && \forall i, t, \\
&\underline{P}_i^g \leq P_i^g(t) \leq \overline{P}_i^g, && \forall i, t, \\
&P_i^g(0) = P_{i,0}^g, && \forall i, \\
&b_i(0) = b_{i,0}. && \forall i.
\end{aligned} \tag{6}$$

The bounds on buses phase $\underline{\theta}$, $\overline{\theta}$ are related to model the thermal limits of the transmission lines (i.e., $\underline{P}_{ij} \leq P_{ij}(t) \leq \overline{P}_{ij}$). Clearly, the DC model can only be used for optimizing the real power dispatches, and the line-losses cannot be modelled as the analysis assumes ideal transmission lines (i.e., $R = 0$).

1) Relaxation Approach for ACOPF: Convex relaxation of the ACOPF problem is another method to overcome the difficulty with solving the non-convex optimization model and is widely studied in the literature [18]. The relaxation techniques for ACOPF are valid under certain restrictive assumptions on the network parameters such as low $\frac{r}{x}$ ratio [19]. In contrary, distribution grids and microgrids have nonnegligible line resistances. Consequently, to apply RH based ACOPF to such networks, a large scale nonlinear and nonconvex optimization problem needs to be solved during each time-epoch. This investigation uses GAMS with IPOPT solver (to be discussed later) to solve the RH ACOPF and by setting the primal-dual gap to a reasonable accuracy, global optimum was achieved for the nonconvex optimization problem in our implementation.

III. I3RES Semantic Middleware Architecture and Definition

This investigation proposes a layered middleware architecture called the *I3RES Middleware Architecture* shown in Fig. 1. It is a software that alleviates the complexity associated with

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2017.2655047, IEEE Transactions on Industrial Informatics

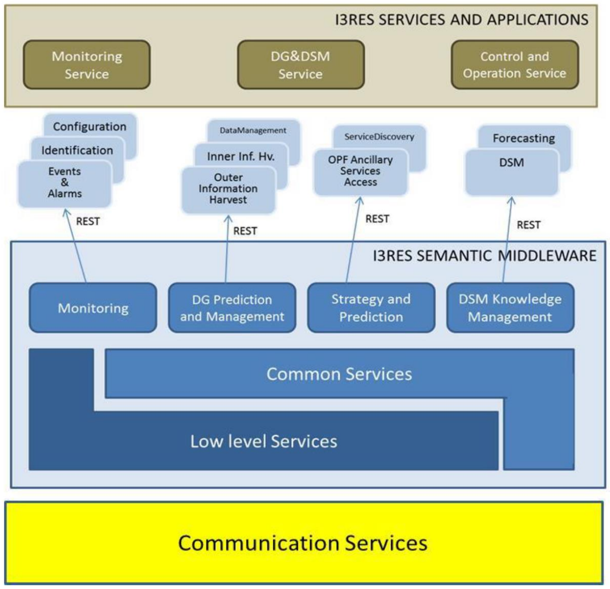

Fig. 1. High Level, Low Level and Common Services of the Middleware

the heterogeneity and interoperability of lower level devices. Thus, it transforms the data collected from various devices into homogeneous information for the application layer. The middleware provides the applications with the services and functionalities they need for accomplishing their tasks. A good approach in the backdrop is to define a set of "high-level services" with a common API that is used by different applications. This provides significant simplicity as the changes and updates in the lower layer become transparent to the application.

Thus, the proposed middleware has the features of the hardware abstraction solution and solves the heterogeneity and interoperability issues. It is organized into three layers: high-level, low-level and communication services (see, Fig. 1). The "high-level services layer" provides the necessary services and applications required for implementing the functional and non-functional requirements of the smart grid control algorithms (see, section IV for discussion on services). The low level services are connected to the hardware and communications layer. These services are used by the high-level services that provide information needed by the application layer. Middleware communication services include ancillary services that are used by both high level and low level services to provide control functionalities required by the applications layer. Finally, the lower layer consists of the physical components such as devices from various vendors and proprietary data formats.

The proposed middleware architecture provides smart grids desirable features such as flexibility, interoperability and scalability. In addition, by providing semantic features it provides the capability to apprehend the meaning of the information being transmitted and it increases the intelligence as well as the performance of the entire system. This is an addition to the current middleware architectures studied within the smart grid community that will optimize the time and resources to

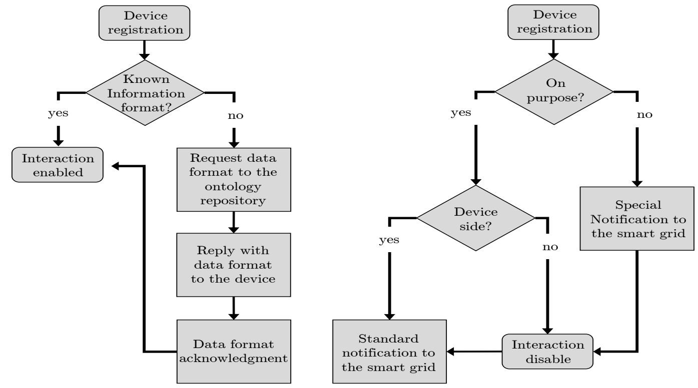

Fig. 2. Flow Chart for Device Interaction

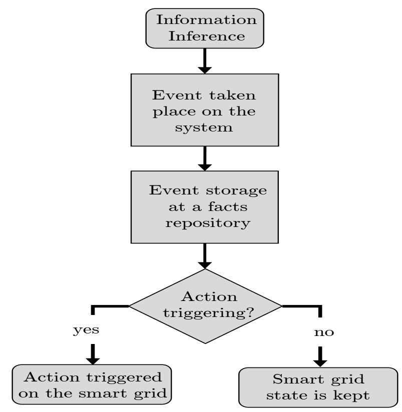

Fig. 3. Flow Chart for Information Inference

make decisions on "how" and "when" the services will be provided. The semantic middleware, as a basic part of the architecture, will embed the CIM information model [20], [21], as well as provide the mechanisms (set of common services) for monitoring the context and integrating and supporting the services. Currently many researchers are working on the CIM model for smart grids, however the discussion on the CIM profiles used in implementing the OPF is beyond the scope of this paper. Details of the CIM profiles for the microgrid model used in RH ACOPF is reported in [22].

A. *High Level Description*

There are several actions that can be performed by the middleware in connecting the services of the application layer, and the hardware devices. The actions are:

1) *Device Registration:* This step includes the device registration or unregistration depending on whether a new device is added or an existing device is removed from the smart grid as shown in Fig. 2. During registration of the device, it must be ensured that the services provided by the new element can be used. This requires a common format to homogenize communications between the system and the new devices. In the event that the information format within the system is unknown, then it can be obtained using queries in standardized languages

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2017.2655047, IEEE Transactions on Industrial Informatics

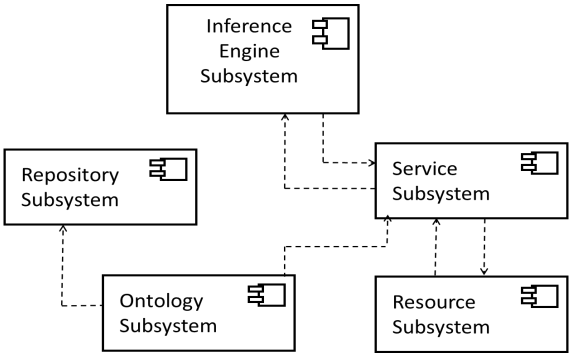

Fig. 4. Subsystems of the Middleware Computation Architecture

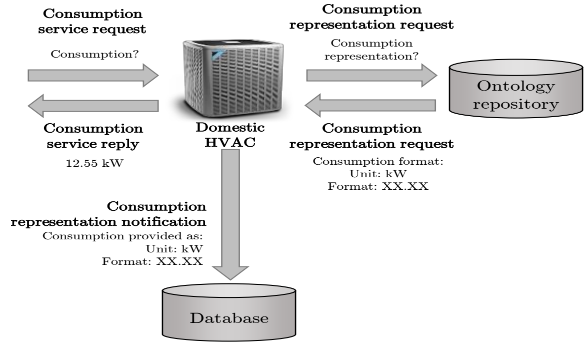

Fig. 5. Ontology Sequence

such as XML or JSON. This is a good option in system using ontologies to represent inner information, as the ontology may have been designed in an *ad-hoc* manner for the system. The removal of the device may be due to two reasons: deliberate action or an accident. In practice, the system is usually aware of the intentional removal of the device. In the event of an accidental removal the system will be notified about the change.

2) *Service provisioning:* The services provided to the smart grid are classified as simple and composed services. A simple service interacts with a single hardware device (e.g. temperature sensor), whereas the composed service interacts with several hardware devices and data (e.g. temperature and humidity sensors). These services are used for monitoring and interaction with end-user's information requests.
3) *Information inference:* A desirable feature of the smart grid automation is to have an entity that can infer information from the context and perform consequent services. Notice two important elements: i) a collection of events that have taken place for a certain amount of time, and ii) a collection of rules that are previously predetermined for the smart grid events. As a result, whenever there is a collection of events that matches certain rules, then a response action is triggered as shown in Fig. 3. Non-semantic system cannot perform consequent services based on the context inference, however semantic capabilities can be obtained by connecting them to semantic gateway that adds required semantic annotations to the data.

B. Computation Architecture

The computation architecture consists of the subsystems shown in Fig. 4. The *Inference engine* is the subsystem tackling all the information related to information inference. It consists of an action collector for all the events that takes place in the system, an inference manager that will compare the actions collected in a repository with that of several prefixed rules and the facts related to the system, and an action triggering component to activate the actions that must be taken whenever there is a match between an event in the system (facts) and how it is governed (rules).

Services manages the capabilities provided to any actor outside the smart grid. Three different components are designed for this *Service Subsystem*: a Factory for Service design, a Request Manager that will handle queries and answer those queries regarding services, and a Container for all the services that are available.

The *Resource subsystem* describes the physical equipment related to the smart grid and does not deal with any service of the equipment. It consists of three components: a) prominent features of the hardware that is currently used, b) sensor and actuator used in the system, and c) input/output operations that require interaction with a piece of hardware.

The *Ontology Subsystem* handles the formatting of the data transfer to the representation style using a component called *Format*, while another component will keep the ontology updated whenever a new device or service becomes available. The *Repository Subsystem* is mainly used for ontology storage.

IV. Semantic Middleware Approach

One can visualize that with the deployment of the middleware, the complexity and heterogeneity of the hardware is hidden to the upper layers as information required is provided in a seamless manner and completely independent of the type of device (smart meters, controllers etc.). The performance of the middleware can be enhanced by adding semantics which denotes the capacity of the system to "understand" information from communication messages. In addition, semantics can be used to take actions based on knowledge extracted from the information and harvesting device information.

Semantics provides flexibility to the smart grid as there is a more dynamic way to read the connected devices. Due to the distributed nature of smart grids, data collection from components (such as wireless smart meters, and home load controllers) may be unreliable, or have issues related with hardware constraints (incapability to send responses to requests or even to receive requests, etc.). Consequently, a more dynamic way to acknowledge the available devices and services in real-time is required. Semantics mitigate this challenge by employing uniform format among inner communication devices, and provide understanding of the consequences of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2017.2655047, IEEE Transactions on Industrial Informatics

modifying a group of hardware devices (either by addition or by removal).

Semantics can be enabled by using ontologies i.e., a collection of entities belonging to a set of common characteristics (e.g. electric power, software engineering) along with their relations under a more or less complicated hierarchy. An ontology can be regarded as a smart dictionary where not only terms are present, but also how they relate to each other. An ontology that is part of a system should be kept updated so as to mirror the different changes that may intervene (e.g. new devices and/or services, outdated ones, new concepts that are being established, etc.).

Figure 5 shows the ontology sequence for providing semantics to the middleware. The new device to be added sends a request asking for the data models to be used, and then it follows the hierarchy and representation of the information currently used in the system. The device then provides a data description with different features (units, thresholds, etc.). The data representation is completed and stored in the ontology repository. Whenever, a service is invoked, an answer with the data represented in the given format is retrieved.

Listing 1. Example of Ontology and Semantically Annotated Data

```
<owl:DatatypeProperty rdf:ID="SmartMeter">
            <rdfs:label xml:lang="en">SmartMeter
            </rdfs:label>
      <rdfs:subClassOf rdf:resource="\#Device"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="SimpleService">
            <rdfs:label xml:lang="en">SimpleService
            </rdfs:label>
      <rdfs:subClassOf rdf:resource="\#Service"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="ComposedService">
            <rdfs:label xml:lang="en">ComposedService
            </rdfs:label>
      <rdfs:subClassOf rdf:resource="\#Service"/>
</owl:DatatypeProperty>

<Device>
      <SmartMeter>
            <name>REX</name>
            <capability>12.8kW</capability>
            <capability>24.6 euro</capability>
            <capability>2 euro/kW</capability>
      </SmartMeter>
</Device>

<Service>
      <SimpleService>
            <measurement>17.8</measurement>
            <unit>C</unit>
      </SimpleService>
</Service>

<Service>
      <ComposedService>
            <measurement>acceptable</measurement>
            <unit>comfortLevel</unit>
      </ComposedService>
</Service>
```

When ontologies are used, they are bound to produce semantically annotated data, i.e. data that have been refined in order to match a representation format. Rather than having the data processed as stream of characters, a data type, or a string, they will be structured and hierarchized by the ontology, which can also be considered as a pattern for building semantically annotated data. A depiction of how both a simple ontology and semantically annotated data look like can be observed in Listing 1. This illustration shows the annotated data employ labels that are defined by the ontology in the first place.

A. Context Awareness

Context awareness denotes the ability of a system to understand the deployed conditions, react to changes in external factors, and provide a more accurate response to any request. The context determines not only the effect of the change on the entity but also the change in its behaviour with respect to time; therefore, the applications will have to adapt themselves to changing context conditions. Context awareness will involve several aspects from the middleware, like an ontology for information inference and resources for data collection.

B. Semantic Middleware Services

1) High Level Services: The upper layer of the middleware is called "high level services layer" and provides the appropriate services directly to the applications. The services required from this layer are obtained from the functional and non-functional requirements of the middleware. Important services offered by this layers include:

1) Inner Information Harvest, which provides methods for obtaining the data needed to implement the applications, such as power consumption, customer data, generation and consumption forecast, etc.;
2) Outer Information Harvest, which provides access to the data available from external sources, such as weather forecast, energy pricing services, etc.;
3) Data Management, which adapts and translates data according to the data model;
4) Events and Alarms, which manages the events and alarms generated within the control platform (I3RES architecture);
5) OPF Ancillary Services, which provides the ancillary services needed by the OPF applications.

2) Low Level Services: The middleware low-level services are connected to the hardware and communications layer. These services are used by high-level services layer to retrieve the information needed by the applications layer. Eight low level services were defined (see Fig. 1): Service Discovery and Registry, Orchestration, Events and Alarms, Ontology, Identification, Context Discovery, Virtual Device, and Data Management.

3) Common Services: The middleware common services layer includes the ancillary services that are used by both high-level and low-level services to provide the functionality required by the applications layer. The common services of the I3RES middleware are (see Fig. 1):

1) Reasoning service providing all mechanisms for forecast algorithms such as weather, pricing, consumption and generation.
2) Security service, used to provide security for the data.
3) Configuration, providing the mechanisms needed to configure and set-up all the components connected to the I3RES middleware.
4) Information Harvest, providing mechanisms to obtain a data that is not accessible to a particular service directly.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2017.2655047, IEEE Transactions on Industrial Informatics

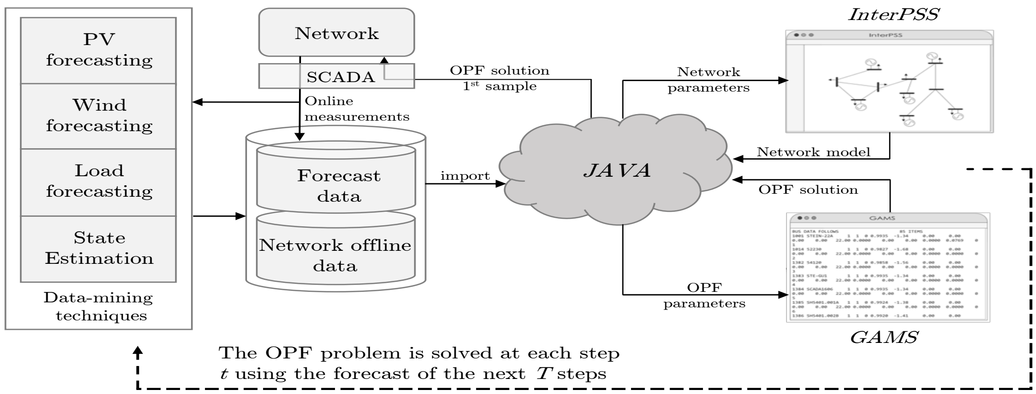

Fig. 6. The Java framework implemented to solve the optimization problem, combining InterPSS and GAMS software features.

Parameter	Value
$\underline{V}$	0.95 p.u.
$\overline{V}$	1.05 p.u.
$\underline{B}$	0.025 MWh a.v.
$\overline{B}$	0.5 MWh a.v.
r^{rated}	1 MW a.v
b^{loss}	0.01 MWh a.v.

TABLE I. Network Parameters

V. Results

A. Case Study: Demo Network in Steinkjer

This section presents the deployment results of the middleware and the multi-objective RH OPF algorithm in a microgrid located at Steinkjer, Norway. It is a radial network that consists of 3 hydro-generators, 32 aggregating load stations, 50 links, and 84 nodes. The RES generation is largely due to solar panels and has grid level storage units. Table I shows the network parameters in per unit (p.u) with a base of 100 MVA and absolute values (a.v.) of storage parameters. The hydro-generation supplies around 10% of the total energy depending on the season and weather condition. To achieve power balance in the microgrid, the network operator buys the energy from an utility at the day-ahead market. To illustrate the performance of RH ACOPF, the obtained results are compared with the RH DCOPF. Such comparison emphasizes that, while the latter schedules real power sources considering the variations in energy prices, it does not bring into account line-losses and reactive power. In turn, this means that power quality and voltage magnitudes cannot be controlled using DCOPF.

B. Solution of the Optimization Problem

The power system modelling and optimization tools used for solving the finite-horizon ACOPF problem are:

- InterPSS (Internet technology based Power System Simulator)[1], a free and open source software for design, analysis, and simulation of power systems. It allows to add and/or change the network topology as well as parameters by using a graphical user interface or the JAVA APIs.
- GAMS (General Algebraic Modeling System)[2] is a high-level modelling system for mathematical programming and optimization. It consists of a language compiler and a stable integrated high-performance solver. GAMS allows to solve complex, large-scale applications as well as build large maintainable models capable of adapting to different problems. Furthermore, it has the APIs needed to integrate its functionalities in a JAVA framework. The InterPSS and GAMS can be combined to solve large-scale optimization problem.

The choice of InterPSS and GAMS in a JAVA platform is motivated by the ability of GAMS API to conveniently exchange inputs and outputs with InterPSS using the *GAMSDatabase* class. In addition, the GAMS API allows the seamless integration with other applications such as the InterPSS— by providing appropriate classes. Moreover, the *GAMSModelInstance* class can solve closely related model instances and is suitable for flexibility, i.e. dynamical addition or removal of components to/from models before the optimization run.

To integrate the different software tools, a JAVA framework is used as shown in Fig. 6. The APIs of InterPSS and GAMS help the seamless integration of these tools within the JAVA framework. As the RH ACOPF is a non-convex optimization problem with discontinuous first order derivatives, the Interior Point OPtimizer solver (IPOPT) is chosen to solve the optimization model in the GAMS environment. It consists of a library designed for large-scale non-linear optimization of continuous systems and implements an interior point method[3] that tries to provide global minimum. The RH ACOPF algorithm was implemented in i7 quad-core processor with 8 GB random access memory where each execution of the RH ACOPF took 10s using the IPOPT solver. In our analysis, we found that the solver converged to the global minimum most of the time (verified using primal-dual gap).

C. Forecast Models

This section presents a succinct description of the models used for forecasting renewable generation and demand. As

[1]https://www.interpss.org

[2]https://www.gams.com

[3]https://projects.coin-or.org/Ipopt

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2017.2655047, IEEE Transactions on Industrial Informatics

many factors, both intrinsic and external to the grid, influence the renewable generation and demand accurate forecasts remain a challenge. Consequently, good forecasting models, providing a reasonable trade-off between accuracy and speed are important for building the RH ACOPF. The photovoltaic (PV) panels are the renewable generators in the microgrid and their generation varies with meteorological factors such as global solar radiation (GSR), temperature, humidity, wind velocity, wind direction, and air temperature. The forecasting models in this investigation estimate the GSR from meteorological data and then use PV models to compute the energy generated. Though there are several models for forecasting GSR, the artificial neural networks (ANN) have outperformed other models due to their ability to learn from data patterns, and provide accurate forecasts [23]. Taking a clue from existing results, different ANN models such as the generalized regression neural networks, radial basis function neural network, multi-layer perceptron, minimum resource allocation network (MRAN), and other approaches were studied for forecasting GSR from meteorological data. The results obtained with forecasting models are reported in Tab. II. Based on existing results, MRAN provided good accuracy. Furthermore, its ability to add/prune neurons improved the computational efficiency of the model. Therefore, MRAN was selected as the PV forecasting tool in our analysis.

Similarly, electricity demand depends on social, economic, and geographical factors [24]. Demand forecast models in this investigation use hourly demand and weather data collected from the distribution network for two years as inputs, and predicted demand as output. Two regression based data-mining models: support vector machines (SVM) and random forests (RF) were selected over other methods based on the error analysis. Table III shows the performance of the RF models for the data obtained. Here, error $= \frac{\hat{P}^d - P^{d,a}}{P^{d,a}}$ with $\hat{P}^d$ predicted demand and $P^{d,a}$ actual demand. RF is selected as the forecasting method over SVM due to its simplicity. A detailed review of the forecasting methods used in this investigation is reported in [24], [25].

[4]RH DCOPF results are shown in dotted and thin lines

Model	MAPE [%]	MSE [%]
MRAN	2-5	2-8
RBFN	12-16	18-22
RBFN (time-series)	3-8	4-12
MLP	7-8	7-12

TABLE II. Performance of Solar Forecasting Models. MLP- Multi Layer Perceptron, RBFN- Radial Basis Function Network

Data	Cumulative min/mean/max/std	Differential min/mean/max/std
Consumption	0.04/0.14/0.49/0.08	2.84/6.28/16.25/2.25
Consumption + Weather	0.04/0.16/1.56/0.15	2.39/6.89/16.28/2.40
Working days (Consumption + Weather)	0.05/0.49/5.14/0.57	2.11/7.12/26.46/3.16

TABLE III. Performance of Demand Forecasting Models

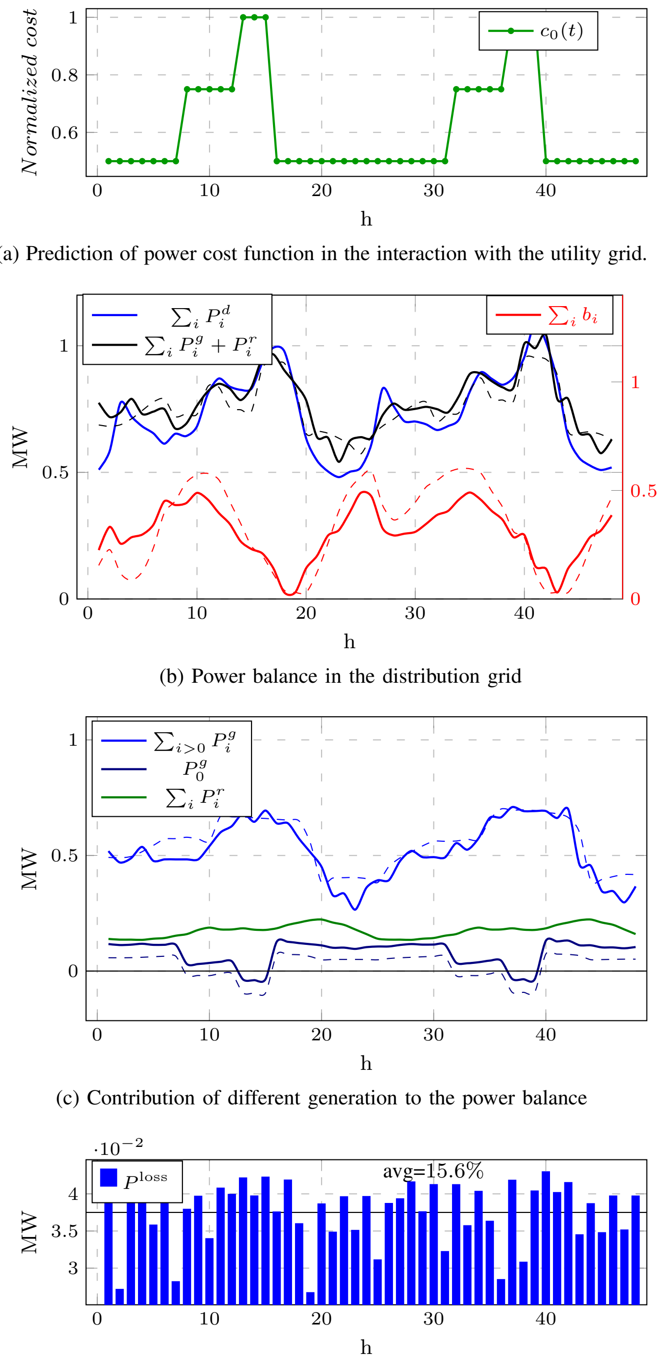

Fig. 7. Simulation with time varying generation cost using MPC based ACOPF controller.[4]

D. ACOPF Versus DCOPF

1) Demo Steinkjer: To illustrate the advantages of the ACOPF its performance is compared with that of DCOPF. The ACOPF has multiple objectives of reducing the line-losses and operating cost. The DCOPF aims to dispatch the generation to reduce the operating cost. In our analysis, we set $\phi = 0.5$ i.e. equal weights are assigned to both economic and line-loss objectives. Figure 7 compares the RHC based ACOPF (in thick lines) and DCOPF (thin and dashed lines).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2017.2655047, IEEE Transactions on Industrial Informatics

The variations in normalized energy prices provided by the utility in the day-ahead electricity market during the test period is shown in Fig. 7a. The energy consumption from the utility, RES generation during the test period, and the total demand on the grid with the ACOPF is shown in Fig. 7b. It is interesting to note that there is a difference between the total power generated and demand due to the power supplied by the ESS. Furthermore, Fig.7c shows that the power is bought/sold from/to the utility grid during the periods of decreasing/increasing energy prices. The behaviour is observed both in ACOPF and DCOPF due to the receding horizon approach. Results revealed that the ACOPF reduces the cost and line-losses by 6.54% and 15.6%, respectively. We remind that DCOPF cannot include the line-losses. Furthermore, the voltage profiles and power quality are maintained using the ACOPF, whereas the DCOPF model does not provide any guarantee. The results also proved that judicious integration of RES and ESS provided significant cost and operational benefits.

E. Middleware Deployment

The middleware presented in this paper is based on the SMArc proposal [12]. The following key indicators were selected to ascertain the middleware performance:

- The solution must be designed considering the requirements of the low-capability devices used in smart grids.
- Security mechanisms must be provided in order to avoid privacy threats.
- The middleware solution should provide components distribution mechanisms.
- Semantic mechanisms to provide annotation of data gathered, as well as ontologies and inference engines support.

The middleware solution provided the services and mechanisms that were required from it for implementing the ACOPF. The solution was tested by performing client application data queries to the middleware. In the test, a large number of clients sent data queries to the middleware service that provides the measurement of a device. The results of this test present a total throughput of 2.47 requests/s and an average response time of 1.6s. The average service registration time of the middleware was around 453.4ms, while the average time for a composed service registration was around 661ms.

F. Flexibility

Flexibility is provided by both the middleware and the RH ACOPF. Using the semantic capability the middleware provides the smart grid a dynamic way to read and connected devices. Semantics mitigate the problem of heterogeneity by providing uniform format for inner communication among devices and provide understanding about the consequences of modifying a group of hardware devices.

The RH ACOPF provides flexibility by integrating RES and ESS. The storage units and renewable energy are scheduled in a way to reduce the operating cost and line-losses in the grid. This provides additional flexibility to optimize the operation, guarantee voltage profiles and power quality, and operate the grid reliably. In addition, the RH ACOPF approach is flexible as the computation burden is not increased significantly due to the introduction/removal of energy components. To study this feature, we added and removed RES and ESS and recorded the corresponding computation time t_c. We took the variable amount of RES and ESS to be equal to 10% of the grid connected load; such value was selected based on the current availability and location of renewable generators in the demo Steinkjer, Norway. Then we experimented at 0%, 25%, 50% and 100% of such amount. The computation time was computed by averaging over 50 runs for each of the cases and is reported in Table IV. One can observe that the computation burden does not change significantly either by adding/removing components.

% RES+ESS	t_c min [s]	t_c max [s]	t_c Average [s]
0	1.098	1.628	1.243
25	1.262	2.132	1.618
50	1.292	2.243	1.423
100	1.682	4.45	2.013

TABLE IV. Execution Time with RES Components added/removed

G. Case Study: IEEE 14, 30 and 57 Bus Systems

A comparison of multi-objective RH ACOPF and DCOPF approaches was performed by optimizing the IEEE 14, 30 and 57 bus systems. As said before, the RH DCOPF cannot model the line-losses while the RH ACOPF optimizes both economic objectives and line-losses. For example, the results in Table V show that while the economic savings with RH ACOPF with a weighting factor of 0.5 is lower than RH DCOPF, there is a significant improvement on line-losses.

Test Case	RH DCOPF	RH ACOPF	
	% Savings	% Savings	% Savings
		line-losses	Operating Cost
IEEE 14 bus	9.19	29.7672	7.4
IEEE 30 bus	9.16	30.7414	6.76
IEEE 57 bus	13.14	46.7855	5.7

TABLE V. Comparison of RH DCOPF and RH ACOPF
*losses are not considered in the DCOPF

The flexibility of the RH ACOPF approach at various scales was studied by optimizing the IEEE 14, 30, and 57 buses. The average computation time t_c for the various test cases and different % of components added or removed from the grid for 50 trials is shown in Tab. VI. As earlier, the RES plus ESS contribution is considered to be 0-10% of total demand. The results show the reasonableness of the computational times for all cases.

H. Comparison of RH ACOPF and Multi-period ACOPF using GA

The performance of the RH ACOPF was compared with that one of the multi-period ACOPF solved using genetic algorithm (GA) with a population size of 50 over 100 generations. Some results for the IEEE 30 bus test case are reported in Table VII

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2017.2655047, IEEE Transactions on Industrial Informatics

%RES+ESS	IEEE 14 t_c [s]	IEEE 30 t_c [s]	IEEE 57 t_c [s]
0	0.568	1.226	3.136
25	0.480	1.230	2.663
50	0.444	1.245	3.542
100	0.682	1.257	3.648

TABLE VI. Average Computation Time with Components Added or Removed

Time	$\sum P_d$ [p.u.]	$\sum P_{res}$ [p.u.]	Cost with MOPF GA [€]	Cost with RH ACOPF [€]
1	1.3181	0.1202	401.627	391.714
2	1.5082	0.1375	390.181	398.757
3	2.0027	0.1826	585.0142	398.416
4	1.8587	0.1695	523.832	389.417
5	1.7586	0.1603	596.2363	388.617
6	1.6897	0.1541	498.1267	385.927
7	1.5833	0.1414	434.037	391.696
8	1.6842	0.1536	479.6833	426.141
9	1.6609	0.1514	411.0325	464.879
10	1.7603	0.1605	503.3151	487.234
11	2.0931	0.1909	535.8765	502.776
12	2.2480	0.2050	615.6887	515.668
13	2.1659	0.1975	700.0301	549.623
14	2.1324	0.1944	605.9772	589.352
15	2.1445	0.1955	646.7081	616.051
16	2.4097	0.2917	636.2925	599.112
17	2.5616	0.2336	738.1125	540.228
18	2.5248	0.2303	752.6921	462.806
19	2.1249	0.1938	548.6096	376.548
20	1.6384	0.1494	502.1489	303.085
21	1.4112	0.1287	425.5487	275.419
22	1.3010	0.1186	386.0702	267.925
23	1.2425	0.1133	361.289	284.556
24	1.2960	0.1182	375.6284	337.979
Total			12653.7579	10343.926

TABLE VII. Comparison of Multi-period GA based OPF and RH ACOPF

and show that the RH ACOPF saves 22.33% more than the GA tuned multi-period ACOPF algorithm for a time horizon of 24 hours.

VI. Conclusions

This investigation presented a semantic middleware and multi-objective receding horizon (RH) based ACOPF for microgrids having distributed renewable generation and storage. The middleware was used to aggregate information from various devices across the grid and provide it to the ACOPF application. This made the ACOPF application totally oblivious to the heterogeneity and complexity of the physical device.

The ACOPF implementation required forecast on demand and renewable generation. The minimum resource allocation network and random forest were selected as the forecast models for RES and demand, respectively due to their accuracy. This investigation used software tools InterPSS and GAMS (with IPOPT solver) integrated using JAVA framework to solve the ACOPF problem. The solution method was based on IPOPT and provided global optimal solutions in most instants (verified using duality gap).

Our results showed that the RH ACOPF reduces the operating cost and line-losses by 6.54% and 15.6%, respectively compared to the DCOPF. Furthermore, a savings of 22.3% (for 24 hours time period) was observed with the RH ACOPF with respect to multi-period ACOPF solved using genetic algorithm. This results illustrates the ability to the RH ACOPF to deal with intermittent renewable generation. In addition, it provides good voltage stability, power quality, and reactive power flows. Similarly, the middeware provided the required services for implementing RH ACOPF. Furthermore, the middleware had a response and service registration time of 1.6s and 661ms, respectively. Studying the Quality of Service of the middleware and extending the ACOPF with a plug and play feature of the RES and ESS are the future course of this investigation.

References

[1] A. Maffei, S. Srinivasan, L. Iannelli, and L. Glielmo, "A receding horizon approach for the power flow management with renewable energy and energ storage systems," in *2015 AEIT International Annual Conference (AEIT)*, 2015, pp. 1–6.

[2] P. Siano, C. Cecati, H. Yu, and J. Kolbusz, "Real time operation of smart grids via fcn networks and optimal power flow," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 4, pp. 944–952, 2012.

[3] F. Yao, Z. Y. Dong, K. Meng, Z. Xu, H. H.-C. Iu, and K. P. Wong, "Quantum-inspired particle swarm optimization for power system operations considering wind power uncertainty and carbon tax in Australia," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 4, pp. 880–888, 2012.

[4] R. Jabr, S. Karaki, and J. Korbane, "Robust multi-period OPF with storage and renewables," *IEEE Transactions on Power Systems*, vol. PP, no. 99, pp. 1–10, 2014.

[5] M. Almassalkhi and I. Hiskens, "Model-predictive cascade mitigation in electric power systems with storage and renewables part ii: Case-study," *IEEE Transactions on Power Systems*, vol. 30, no. 1, pp. 78–87, Jan 2015.

[6] T. Strasser, P. Siano, and V. Vyatkin, "New trends in intelligent energy systems–an industrial electronics point of view," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 4, pp. 2420–2423, 2015.

[7] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "A survey on smart grid potential applications and communication requirements," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 28–42, 2013.

[8] H. Gjermundrod, D. E. Bakken, C. H. Hauser, and A. Bose, "Gridstat: A flexible qos-managed data dissemination framework for the power grid," *Power Delivery, IEEE Transactions on*, vol. 24, no. 1, pp. 136–143, 2009.

[9] L. Zhou and J. J. Rodrigues, "Service-oriented middleware for smart grid: Principle, infrastructure, and application," *Communications Magazine, IEEE*, vol. 51, no. 1, pp. 84–89, 2013.

[10] A. P. Garcia, J. Oliver, and D. Gosch, "An intelligent agent-based distributed architecture for smart-grid integrated network management," in *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*. IEEE, 2010, pp. 1013–1018.

[11] S. Sučić, B. Bony, and L. Guise, "Standards-compliant event-driven soa for semantic-enabled smart grid automation: Evaluating iec 61850 and dpws integration," in *Industrial Technology (ICIT), 2012 IEEE International Conference on*. IEEE, 2012, pp. 403–408.

[12] J. Rodíguez-Molina, J.-F. Martínez, P. Castillejo, and R. Diego, "Smarc: a proposal for a smart sematic middleware architecture focussed on smart city energy management," *International Journal of Distributed Sensor Networks*, vol. 2013, 2013.

[13] D. Villa, C. Martín, F. J. Villanueva, F. Moya, and J. C. López, "A dynamically reconfigurable architecture for smart grids," *Consumer Electronics, IEEE Transactions on*, vol. 57, no. 2, pp. 411–419, 2011.

[14] A. Awad and R. German, "Self-organizing smart grid services," in *Next Generation Mobile Applications, Services and Technologies (NGMAST), 2012 6th International Conference on*. IEEE, 2012, pp. 205–210.

[15] Z. Shaomin, Z. Jinchong, and W. Baoyi, "Study on middleware of data collection based on IEC 61970 and rfid technology in smart grid," in *2012 2nd International Conference on Applied Robotics for the Power Industry (CARPI)*.

[16] A. P. Garcia, J. Oliver, and D. Gosch, "A unified solution for advanced metering infrastructure integration with distribution management system," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*. IEEE, 2010, pp. 566–571.

[17] F. Li and R. Bo, "DCOPF-based LMP simulation: algorithm, comparison with ACOPF, and sensitivity," *IEEE Transactions on Power Systems*, vol. 22, no. 4, pp. 1475–1485, 2007.

[18] L. Gan, N. Li, U. Topcu, and S. Low, "On the exactness of convex relaxation for optimal power flow in tree networks," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 465–471.

[19] S. Bose, S. H. Low, T. Teeraratkul, and B. Hassibi, "Equivalent relaxations of optimal power flow," *IEEE Transactions on Automatic Control*, vol. 60, no. 3, pp. 729–742, 2015.

[20] 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, 2011.

[21] A. Naumann, I. Bielchev, N. Voropai, and Z. Styczynski, "Smart grid automation using IEC 61850 and CIM standards," *Control Engineering Practice*, vol. 25, pp. 102–111, 2014.

[22] i3RES consortium. (2015) Deliverable d6.1: Smart grid model. [Online]. Available: http://http://www.i3res.eu/v1/

[23] A. K. Yadav and S. Chandel, "Solar radiation prediction using artificial neural network techniques: A review," *Renewable and Sustainable Energy Reviews*, vol. 33, pp. 772–781, 2014.

[24] I. Landa-Torres, I. Unanue, I. Angulo, M. R. Russo, C. Campolongo, A. Maffei, S. Srinivasan, L. Glielmo, and L. Iannelli, "The application of the data mining in the integration of res in the smart grid: Consumption and generation forecast in the i3res project," in *2015 IEEE 5th International Conference on Power Engineering, Energy and Electrical Drives (POWERENG)*. IEEE, 2015, pp. 244–249.

[25] i3RES consortium. (2015) i3RES: ICT-based intelligent integration of Renewable Energy Sources. [Online]. Available: http://http://www.i3res.eu/v1/