Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 18 (2013) 1899 – 1908

Procedia Computer Science

2013 International Conference on Computational Science

DDDAMS-based Dispatch Control in Power Networks

Nurcin Celik*, Aristotelis E. Thanos, Juan P. Saenz

University of Miami, 1251 Memorial Drive, McArthur Engineering Building Room 280, Coral Gables and 33146, USA

Abstract

Electricity networks need robust decision making mechanisms that enable the system to respond swiftly and effectively to any type of disruption or anomaly in order to ensure reliable electricity flow. Electricity load dispatch is concerned with the production of reliable electricity at the lowest costs, both monetary and environmental, within the limitations of the considered network. In this study, we propose a novel DDDAMS-based economic load dispatching framework for the efficient and reliable real-time dispatching of electricity under uncertainty. The proposed framework includes 1) a database fed from electrical and environmental sensors of a power grid, 2) an algorithm for online state estimation of the considered electrical network using particle filtering, 3) an algorithm for effective culling and fidelity selection in simulation considering the trade-off between computational requirements, and the environmental and economic costs attained by the dispatch, and 4) data driven simulation for mimicking the system response and generating a dispatch configuration which minimizes the total operational and environmental costs of the system, without posing security risks to the energy network. Components of the proposed framework are first validated separately through synthetic experimentation, and then the entirety of the proposed approach is successfully demonstrated for different scenarios in a modified version of the IEEE-30 bus test system where sources of distributed generation have been added. The experiments reveal that the proposed work premises significant improvement in the functional performance of the electricity networks while reducing the cost of dynamic computations.

Selection and peer review under responsibility of the organizers of the 2013 International Conference on Computational Science

Keywords: Dynamic data driven application systems (DDDAS); power grids; economic load dispatch; real-time decision making

1. Introduction

Reliance on foreign energy sources weakened by a fragile electrical grid poses major threats to the United States' security [1] [2]. Quantifying this statement, the Air Force (AF) alone spent $1.06 billion during 2007 on its worldwide energy needs as the largest consumer of petroleum within the Department of Defense.

* Corresponding author. Tel.: +1-305-284-2391; fax: +1-305-284-4040.
E-mail address: celik@miami.edu.

1877-0509 © 2013 The Authors. Published by Elsevier B.V.
Selection and peer review under responsibility of the organizers of the 2013 International Conference on Computational Science
doi:10.1016/j.procs.2013.05.359

Consequently, deployment of smart grid and energy efficient technologies has become an essential focus area for science and technology investment along with increased use of multi-scale simulations and autonomous systems. To this end, expanding renewable energy for facility use in conjunction with the implementation of a national clean energy smart grid, would reduce the risks and burdens on budgets and bolster national security.

Regarding smart grids, technological advances in microturbines, solar panels, reciprocating engines, digital controls and remote monitoring devices (among various others) have increased the opportunities and applications for "next generation" power grids, and have given customers great flexibility to tailor energy systems to their specific needs. At the same time, electric utility companies are exploring the possibilities that distributed generation may help address some of the requirements of the electric system, promoting greater energy security, economic competitiveness and environmental protection. Distributed generation has the following impacts on power networks: in terms of voltage violations and power qualities, the presence of distributed generation may help to reduce variations and impact voltage flicker and harmonics, in terms of power losses, the deployment of distributed generation will generally decrease the amounts of power lost in the system, and last but not least, the presence of distributed generation may enhance reliability if used to provide backup power. However, increasing penetration level of the distributed generation may increase security risks and cause crashes in the energy system, if it is not properly interfaced with the network.

Considering the issues highlighted above, effective monitoring and management of these critical infrastructures is vital to realize economic security and reliability in operations including cost savings in energy consumption, mitigating the debilitating impacts of possible central grid disturbances such as faults, power quality events, and most critically black-outs, improvements in reactive support and voltage profiles, removal of distribution and transmission bottlenecks, reduction of losses, and development of new transmission and generation systems. While initial demonstrations of important concepts that are integral to the power network configurations have been achieved, a fully functional master controller does not yet exist [3]. In this study, we investigate a novel dynamic data driven adaptive simulation (DDDAMS) framework that is designed for the efficient and reliable real-time dispatching of electricity under uncertainty.

The proposed framework includes 1) a database receiving data from electrical and environmental sensors of a power grid, 2) an algorithm for online state estimation of the demand nodes in the considered electrical grid using particle filtering, 3) an algorithm for effective culling and fidelity selection in simulation considering the trade-off between the computational requirements of simulations, and accuracy of anticipated dispatch results in terms of environmental and economic costs, and 4) data driven simulation for mimicking the system response behavior and generate a dispatch configuration which minimizes the total operational cost and power loss of the system, without posing security risks to the energy network.

The rest of the paper is organized as follows. In Section 2, we provide the background and literature review on dynamic data driven application systems (DDDAS) paradigm. In Section 3, we describe our proposed DDDAMS framework for economic load dispatching in distributed power grids. In Section 4, we describe the characteristics, topology, and sensory data of the considered IEEE-30 bus system; and discuss the performance of the proposed framework via results obtained using this system. Finally in Section 5, we provide conclusions and discussions on the planned future venues for this work.

2. Previous Works on DDDAS Paradigm

Dynamic data driven application systems have been conceived as a powerful tool that allows more effective measurement processes in a variety of application areas, bringing along challenges in the aspects of applications, mathematical algorithms, systems software, and in the way data is collected [4]. In a study of contaminant tracking, numerical procedures for multi-scale interpolation are introduced in order to map sensor data and allow continuous update of the simulation [5]. Another study [6] introduces the use of online data acquisition and a filtering control to recognize out-of-order data. [7] draws attention to the challenges of automatically adapting simulations when experimental data indicates that a simulation must change. To this end, a simulation is first run to gain insight about a phenomenon. Then, this insight is used to determine what new observations should be collected, and the simulation is adapted to reflect these observations. Generalizing software to anticipate all possible ways it could change is difficult, and attempting to do so usually comes at the expense of performance, as well as makes the code unmanageably complex [8]. However, the problem of software adaptation can be simplified by taking advantage of the flexibilities and constraints of a simulation at the same time. Without flexibility, automatic adaptation is impossible because there is no way to know which alternatives should be considered. Without constraints, automatic adaptation is infeasible because there are too many alternatives to consider in a timely fashion. [7], therefore, propose a semi-automated adaptation approach that exploits the flexibility and constraints of model abstraction opportunities to automate simulation adaptation. While their study does not involve manual or automatic modification of the code or application of optimization methods which can make the software extremely complex to control, it is still in need of human intervention to determine the most likely places of the code in need to be changed. In our proposed research, changes in level of detail of data acquisition and the choice of certain parameters over others allow the automatic multi-fidelity adaptation in the simulation model.

In [9], researchers from the University of Kentucky and University of Miami's Rosenstiel School of Marine and Atmospheric Science report their efforts to build data driven application systems for short-range forecasting of weather and wildfire behavior from real time weather data, images, and sensor streams. [11] and [12] propose a simulation-based shop floor planning and control system, where the same simulation model (executing in the fast mode and after going through some modification) is employed at the planning stage after it is used as a real-time task generator at the control stage. In their approach, the real-time simulation drives the manufacturing system by sending and receiving messages using socket-based communication links. While the use of real-time simulation as a task generator is common ground for these works and our current study, the adaptive simulation scheme steering the measurement process for selective data update and incorporating the real-time dynamic data into the executing simulation model is novel in our research. Such adaptivity allows us to save computational power usage while keeping the model accurate enough by wisely drawing conclusions via the embedded algorithms which are developed as part of this research.

3. Proposed DDDAMS Framework

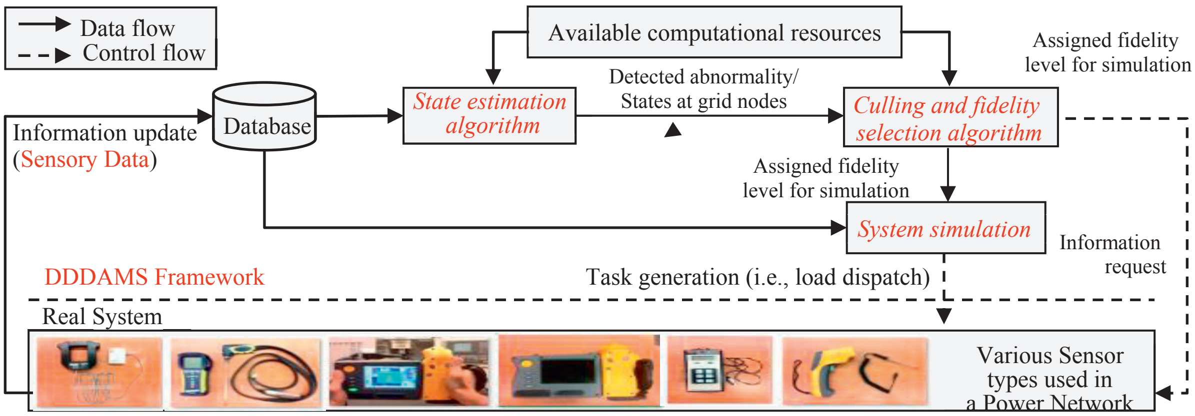

Fig. 1. Overview of proposed DDDAMS framework with embedded algorithms

In order to address the challengs mentioned in Section 1, a dynamic data driven adaptive multi-scale simulations (DDDAMS) framework is proposed in this study (see Figure 1 for overview). The overall scheme envisioned is a robust multi-scale federation of simulation models that support planning and control decisions in electrical networks and their connection to central power grids. The basic structure of the proposed DDDAMS system consists of the application (real system – electrical power network), the grid computing modules, a message-oriented web server, databases where the updated measurement values are stored, and the real-time (RT) DDDAM-simulation. Decision making capabilities of the proposed framework are enabled through embedded algorithms. Data sources (sensors) installed in each component (e.g., regions in electrical network) obtain data from the real system and transmit it to the RT DDDAM-simulation through the web server. Given the updated sensory data and available computational power, algorithms embedded in the RT DDDAM-simulation are invoked to determine the system state and level of detail (fidelity) that the simulation model should run at. Then, the DDDAM-simulation adjusts itself to continue running on this new fidelity to evaluate the future behavior of the system and generate control tasks (environmental economic load dispatch in the considered application). Once the best task alternative is chosen, it is executed to drive the actual system as planned at the control stage where switching between the planning and control stages may occur either periodically or as a result of a system change (event-based). This process continues while the DDDAM-simulation is running. The skeleton and working principles of the components of the proposed framework are detailed in the following sub-sections.

3.1. Online State Estimation of a Networked Electrical Grid using Particle Filtering

Efficient state estimation is crucial in power networks due to its major impact on the control of the power flow and security of the system. While the state estimation has been extensively studied for transmission systems, the techniques used at the transmission level are not suitable for distribution networks mainly due to the lack of reliable (diverse) measurement data and accurate measurement models. State estimation in medium voltage networks at the distribution level is conducted by evaluating the voltage and angles of each and every bus in the network, and is a non-linear problem. As such, a reliable state estimation necessitates the timely evaluation of massive datasets containing electrical information which usually are collected by the Supervisory Control and Data Acquisition (SCADA) system. However, this is not a trivial task due to two main reasons. First, handling all the state variables of the nodes in distributed power networks in real time is itself a significant challenge due to the computational power requirements associated with its processing. Second, the data collected via the SCADA system is often noisy and erroneous causing further inaccuracies in the estimation. Addressing these challenges, the proposed algorithm aims at producing accurate estimation of these states in real-time against imperfect and massive datasets using a smart sampling approach. By using the previously estimated states, and incoming dynamic measurements obtained from both environmental (i.e., temperature) and electrical sensors (i.e. real power injections, reactive power injections, real power flows, currents, etc.), the algorithm updates estimations for state of each node in the network at each decision cycle.

In this work, the proposed state estimation algorithm embeds two particle filtering (PF) sub-procedures that can either be used separately or combined (see Figure 2). The first sub-procedure yields to an aggregate state estimation (for *major* states) incorporating the measurements from environmental sensors such as temperature and seasonality as depicted partially in (1), while the second one refines this estimation by using the measurements from electrical sensors such as voltage magnitudes, power injections, power flow, and current (for *minor* states) as depicted in (2). In this proposed algorithm, the states of the buses are defined by the real and reactive power injections in those buses. Frequency of data collection is determined on the basis of load variation and response times of the different available energy generation sources. The minimum and maximum of these frequencies are governed by the fastest possible response time of energy generation sources and duration in which the load variation is kept within a threshold, respectively. Evidently, higher frequencies of data collection lead to higher accuracies in estimations. On the other hand, lower frequencies result in lighter computational weights. As a consequence, the optimal (or near optimal) frequency for data collection should be decided considering this trade-off between estimation accuracy and its associated computational burden. Once this frequency is set, the algorithm generates four state variables corresponding to real and reactive power injections at either week-day or weekend-day for each bus.

When a new environmental measurement is available, the first PF sub-procedure is initiated. A sample with size N_1 is drawn from the prior power injection probability density function and their weights are assigned. If the effective number of particles (the number of samples that have significant weights) is substantially small, the resampling step is invoked and new samples are drawn (still at the same stage). When the effective number of particles is sufficient, the major state variables are estimated and stored. Then, the second sub-procedure is commenced using new electrical measurement to increase the accuracy of these major states via newly introduced "*minor*" states. Here, a new sample with size N_2 is drawn and as in the first sub-procedure, weights are assigned to them. If necessary, the resampling step is used, and new estimations for the minor state variables are computed. Since the electrical measurements are collected more frequently than the environmental ones, the estimation obtained from the second sub-procedure is used to correct the estimation obtained from the first one.

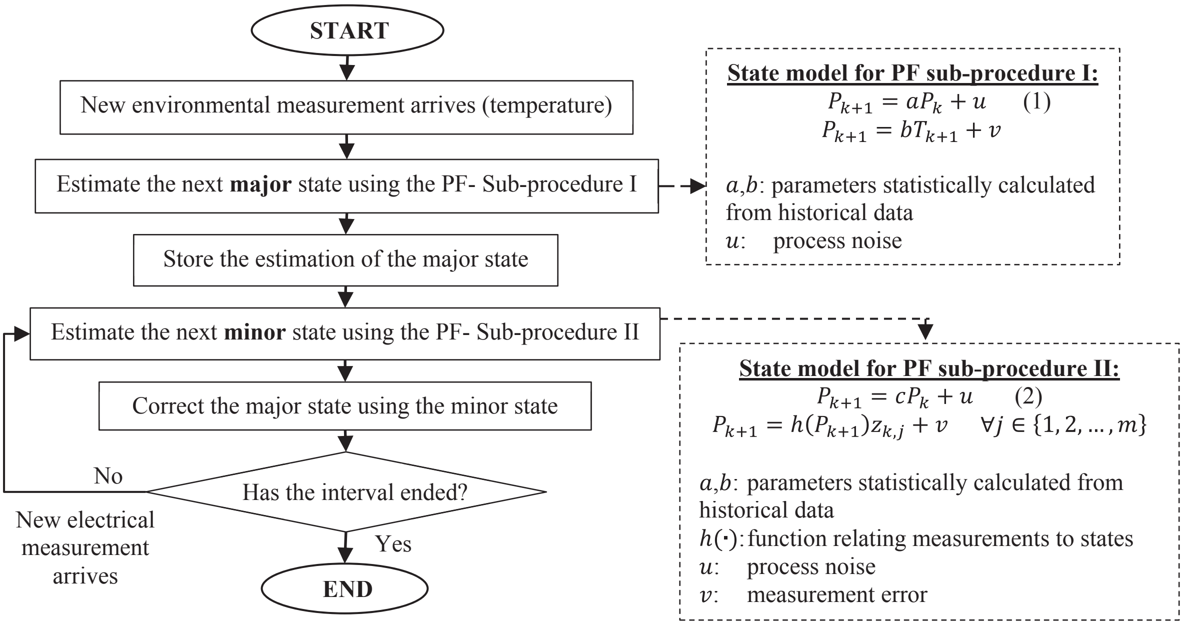

Fig. 2. Operations of the state estimation algorithm using particle filtering

It should be noted that electrical measurements, incorporated into the second sub-procedure can be handled flexibly considering the trade-off between algorithm's estimation accuracy and the computational burden it requires. Therefore, if a higher accuracy is desired, the algorithm will be obliged to run with larger datasets comprised of measurements obtained at more frequent time intervals. On the other hand, if a faster response time is preferred, only selective measurements gathered at specified times or places will be used. This capability is enabled in this second sub-procedure through the usage of several sets of measurements that are available within each predefined interval. This particular characteristic leads to a remarkable difference from the generic particle filtering technique. In the generic filter, a single measurement parameter is used to estimate the posterior probability distribution for each state. However, in the proposed sub-procedure, several measurements can be incorporated at the same stage and thus potentially provide better state estimations. On the other hand, certain minor states can be omitted, depending on the desired accuracy level and the usage of computational sources. The assessment of the minor states in the second sub-procedure may be used to improve the estimation of the major states. In this way, an enhanced efficiency of the algorithm can be achieved by combining the two sub-procedures and running them simultaneously.

3.2. Effective Culling and Fidelity Selection in Simulation

The goal of the proposed effective culling and fidelity selection algorithm is to determine the best feasible level of detail that the DDDAM-Simulation should run at in order to derive the dispatch while ensuring that minimal computational resources are used. In our proposed framework, culling and fidelity selection are performed based on the estimated states that are evaluated in the online state estimation algorithm and the expected computational burden needed for this simulation.

Here, if the algorithm concludes that the system is operating under normal conditions or the variation in the load is restricted to a some particular sub-network, the amount and type of data pulled from the real system become area-specific, and the level of detail of the simulation is set to the most detailed one in that sub-network whereas the rest of the network is simulated at the most aggregated level in order to save from computational resources (namely Fidelity 1). Therefore in Fidelity 1, the simulation only modifies the dispatch of the sources in this sub-network and the swing bus (used to ensure load balancing). On the other hand, if the state estimation algorithm diagnoses several abnormalities or load variations dispersed throughout the entire network at the expense of heavier computational burden, the DDDAM-Simulation is run for the entire network at the highest detail possible (namely Fidelity 2) in order to update its dispatch decisions.

The environmental economic load dispatch problem considered in this work is the determination of the output of electric resources to reliably meet the system demand minimizing economic and environmental costs, while ensuring that constraints of power balance, and capacity limits are met. The total economic cost of the generated electricity is provided by $\sum a_i + b_i P_i + c_i P_i^2$, where a_i, b_i, and c_i are cost coefficients of the i^{th} generator, and P_i is the amount of the real power obtained from the i^{th} generator. The total environmental cost of the generated electricity is provided by $\sum\left[10^{-2}(\alpha_i + \beta_i P_i + \gamma_i P_i^2) + \zeta_i e^{\lambda_i P_i}\right]$ where α_i, β_i, γ_i, ζ_i, and λ_i are the coefficients of the i^{th} generator's emissions. The total power loss is defined as $P_{loss} = \sum_{i=1}^{N}(P_i - L_i)$, where P_{loss} is the total power loss, N is the total number of buses, and L_i is the real load at bus i. The capacity and real and reactive power balance constraints that must be satisfied in the economic dispatch problem are given in (3)-(5) where V_i is the voltage magnitude at bus i, δ_i is the voltage angle at bus i, C_{ij} is the transfer conductance between buses i and j, T_{ij} is the transfer susceptance between buses i and j, Q_i is the reactive power generated at the i^{th} bus, and R_i is the reactive load at bus i.

$$P_i^{min} \leq P_i \leq P_i^{max} \qquad \forall i \tag{3}$$

$$P_i - L_i - V_i \sum_{j=1}^{N} V_j[C_{ij}cos(\delta_i - \delta_j) + T_{ij}sin(\delta_i - \delta_j)] = 0 \qquad \forall i \tag{4}$$

$$Q_i - R_i - V_i \sum_{j=1}^{N} V_j[C_{ij}sin(\delta_i - \delta_j) + T_{ij}cos(\delta_i - \delta_j)] = 0 \qquad \forall i \tag{5}$$

3.3. Realization of Architecture in Virtual Setting

In this work, data driven simulations of the considered real system (a test power network) are built in a multi-scale manner where models are federated in a distributed computing environment. Each process in our considered power network has different inputs, outputs, and controls which can be modeled via different modeling techniques, such as different statistical distributions, differential equations, or process simulators for different levels of fidelities considering the modeling accuracy and computing power. For instance, if the culling and fidelity selection algorithm concludes that the system is operating under normal conditions, the amount and type of data pulled from the real system become minimal and the level of detail of the simulation is set to the most aggregated level in order to save from computational resources. If the culling and fidelity selection algorithm detects an abnormality, the amount and type of data pulled from the real system as well as the level of detail of the simulation increases gradually. Once the model fidelity is determined, the data driven simulation adjusts itself to evaluate the future behavior of the system. Changes in computational resource availabilities due to collecting, processing, and analyzing data from the sensors (based on the selected fidelity) are monitored via an automated .NET based Grid Computing framework.

3.4. Communication using Web Services and Time Synchronization

For the execution of data driven simulations, there is a need of communication between heterogeneous and distributed system simulations. In this work, we facilitate this feature by the communication server that has been developed in earlier works [13][14] using Web Services technology (state-of-the-art distributed computing technology) that overcomes barriers of standard communication via the usage of W3C standard protocols including XML, WSDL, and SOAP. The communication server provides a backbone which complements the computation structure provided by grid-based computing. Web Services provide an elegant mechanism for communicating among distributed components without keeping much state information, as required in conventional means such as socket based interaction.

4. Experiments and Preliminary Results

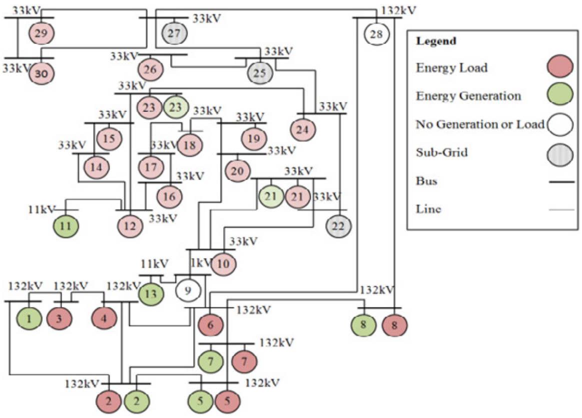

Fig. 3. IEEE-30 Bus Test System

Table 1: IEEE-30 Load Data

Bus	Original Load		Increased Load		Bus	Original Load		Increased Load	
	MW	MVAR	MW	MVAR		MW	MVAR	MW	MVAR
1	0	0	0	0	16	3.5	1.8	3.709	1.906
2	21.7	12.7	22.293	13.048	17	9	5.8	9.401	0.83
3	2.4	1.2	2.423	1.268	18	3.2	0.9	3.379	0.917
4	7.6	1.6	7.694	1.637	19	9.5	3.4	9.613	3.6
5	94.2	19	94.347	19.788	20	2.2	0.7	2.273	0.703
6	0	0	0	0	21	17.5	11.2	18.459	11.247
7	22.8	10.9	24.158	11.549	22	0	0	0	0
8	30	30	30.298	30.072	23	3.2	1.6	3.339	1.69
9	0	0	0	0	24	8.7	6.7	8.852	6.874
10	5.8	2	5.804	2.041	25	0	0	0	0
11	0	0	0	0	26	3.5	2.3	3.559	2.364
12	11.2	7.5	11.485	7.598	27	0	0	0	0
13	0	0	0	0	28	0	0	0	0
14	6.2	1.6	6.42	1.646	29	2.4	0.9	2.524	0.945
15	8.2	2.5	8.349	2.613	30	10.6	1.9	11.081	1.913

The proposed DDDAMS framework is validated via a modified version of the IEEE-30 bus test system from the Department of Electrical Engineering at the University of Washington. The original version of the IEEE-30 bus test system includes 31 buses and 41 lines, with energy generation at 6 buses, energy loads at 22 buses and 5 buses with no generation or loads. In our modified version of the IEEE-30 bus system we have chosen to add 3 sources of distributed generation to the system, located at buses 7, 21 and 23, arbitrarily and divide the network so that 19 of the buses form a sub-network, as shown in Figure 3.

In order to evaluate the performance of the DDDAMS framework, two different scenarios are considered. In the first scenario, load variation occurs only in the buses within a sub-network; while in the second scenario, load variation may occur throughout the entire network. To this end, the best feasible fidelity is determined for each scenario when the variation in the loads are less than 3%, 6%, and 9%, respectively. Table 1 shows the original loads versus loads obtained from the state estimation algorithm when there is a change of up to 3%. In our experiments, the simulation is run for 30 independent iterations.

4.1. Scenario 1:Load Variation Occurs only within the Sub-network

This first scenario represents the situations when there are load changes only in some specific buses whose impact may potentially be enclosed within a sub-network, and may not be felt in the entire network. Here, a non-dominated solution set including a best compromise dispatch obtained using the fuzzy logic method presented in [15] (highlighted in red in Figure 4) is shown for benchmarking purposes. Their best compromise solution has a cost and emissions of $633.27 per hour, with emissions that total 0.2673 tons per hour. The non-dominated solution set has energy dispatch options that range from $560.07 to $722.81 per hour, in terms of cost, and from 0.2671 to 0.2673 tons per hour in terms of emissions.

When loads within a sub-network vary up to 3%, and the fidelity selection algorithm decides to run the simulation at Fidelity 1 (capturing the considered sub-network and swing bus only), a best compromise solution that has a cost of $634.67 per hour with emissions of 0.26725 tons per hour is achieved (shown in light green in Figure 5). The total computational time used for this solution is 181.4516 seconds. Here, the non-dominated solution set (shown in green in Figure 5) ranges from $600.59 to $658.75 per hour in terms of cost and from 0.2672 to 0.2673 tons per hour in terms of emissions. If the fidelity selection algorithm determines that a full update of the energy dispatch is necessary and decides to run the simulation at Fidelity 2, the best compromise solution (shown in light blue in Figure 5) has a cost of $665.10 per hour with emissions of 0.26721 tons per hour. The total computational time used for this solution is 483.5126 seconds. In this case, the non-dominated solution set (shown in blue in Figure 5) ranges from $524.28 to $697.35 per hour, and from 0.2672 to 0.2674 tons per hour. It is important to highlight that if the information from both of fidelities is combined and a collective non-dominated solution set is constructed, the best compromise solution of this set becomes the solution with the least emissions, from the non-dominated solution set obtained using Fidelity 1 (highlighted in red in Figure 5). Therefore, in this scenario the use of Fidelity 1 for simulation produces satisfactory results with savings in computational time of 302.061 seconds.

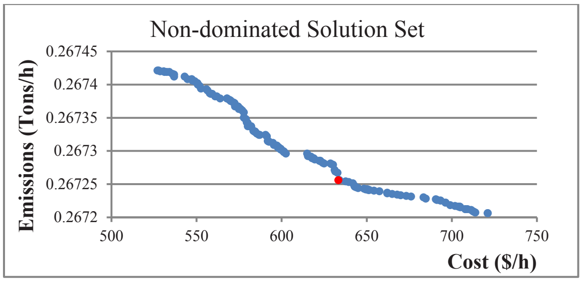

Fig. 4. Base loads non-dominated solution set

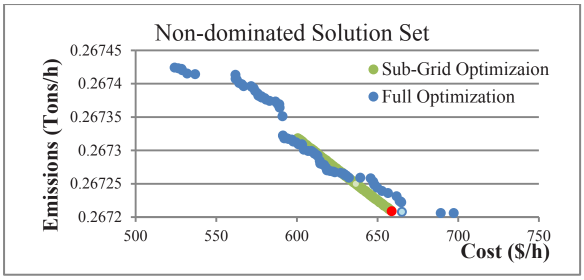

Fig. 5. 3% load changes within the sub-network

When loads within a sub-network vary up to 6% and the simulation is run at Fidelity 1, the best compromise solution has a cost of $635.73 per hour with emissions of 0.2673 tons per hour, while the total computational time used for solution is 208.01 seconds. In this case, the non-dominated solution set ranges from $608.39 to $667.05 per hour in terms of cost and from 0.2672 to 0.2673 tons per hour in terms of emissions. If the simulation is run at Fidelity 2, the best compromise solution has a cost of $657.38 per hour with emissions of 0.2672 tons per hour, while the total computational time used for the solution is 471.40 seconds. Here, the non-dominated solution set ranges from $535.22 to $693.58 per hour, and from 0.2672 to 0.2674 tons per hour. In this case, when a collective non-dominated solution set is constructed, the best compromise solution of this set is from the non-dominated solution set of the Fidelity 1 with a cost of $538.38 per hour and emissions of 0.26739 tons per hour. The use of Fidelity 1 also produces satisfactory results with savings in computational time of 263.39 seconds.

In the case where loads vary up to 9%, the results produced by both fidelities have some similarities to those where loads change up to 3%. The best compromise solution set obtained from the simulation running at Fidelity 2 dominates the best compromise solution obtained from that of running at Fidelity 1. However, the best compromise solution from the collective non-dominated solution set is the solution with least emissions from the Fidelity 1. This indicates that even when load changes are as significant as 9%, the simulation running at Fidelity 1 has reasonable results, while enabling savings of 468.78 seconds in computation.

Under the three cases evaluated in this scenario, the solution with the least emissions from the non-dominated solution set of the sub-grid optimization procedure is part of the collective non-dominated solution set. Therefore, it can be concluded that when the changes in the loads are within the considered sub-network, the least emissions solution from the non-dominated solution set obtained from Fidelity 1 is a viable operational alternative that runs 354.5% faster that of Fidelity 2.

4.2. Scenario 2: Load Variation Occurs throughout the Entire Network

In this scenario, we consider the cases where load changes may occur throughout the entire network up to 3%, 6% and 9%, respectively. When the loads vary up to 3% and the fidelity selection algorithm runs the simulation at Fidelity 1, a best compromise solution (shown in light green in Figure 6) that has a cost of \$631.46 per hour with emissions of 0.2673 tons per hour is achieved. The total computational time used for this simulation is 178.48 seconds. Here, the non-dominated solution set (green in Figure 6) ranges from \$601.39 to \$667.96 per hour in terms of cost and from 0.2672 to 0.2673 tons per hour in terms of emissions. If the fidelity selection algorithm runs the simulation at Fidelity 2, the best compromise solution (light blue in Figure 6) has a cost of \$636.34 per hour with emissions of 0.2672 tons per hour, while the total computational time used for the simulation is 536.20 seconds. When the collective non-dominated solution set is constructed, the best compromise solution is also the least emissions solution from the non-dominated solution set of the one obtained from Fidelity 1 (red in Figure 6). For this scenario, it can be concluded that running the simulation at Fidelity 1 produces satisfactory results with savings of 357.72 seconds in computational time.

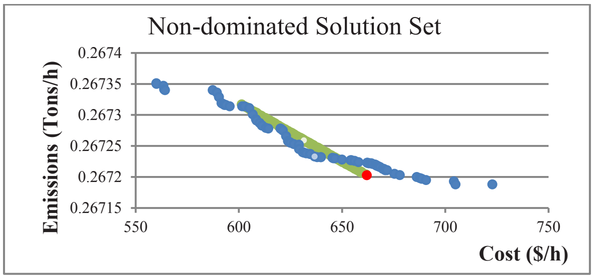

Fig. 6. 3% load changes throughout the entire grid

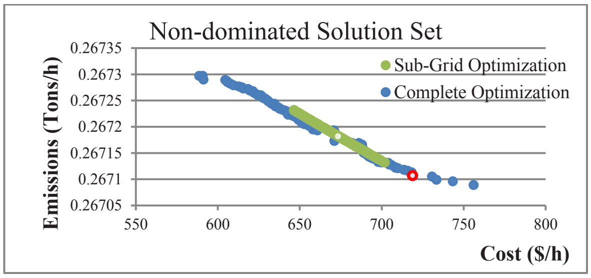

Fig. 7. 9% load changes throughout the entire grid

When the loads vary up to 6% and the fidelity selection algorithm runs the simulation with Fidelity 1, the best compromise solution has a cost and emissions of \$668.33 and 0.2672 tons per hour, respectively. The total computational time used for the simulation is 183.61 seconds. If the fidelity selection algorithm runs the simulation at Fidelity 2, the best compromise solution has a cost of \$685.87 per hour with emissions of 0.2672 tons per hour, with total computational time of 663.13 seconds. In this case, when the collective non-dominated solution set is constructed, the best compromise solution is also the least emissions solution from the non-dominated solution set obtained from the simulations running at Fidelity 1 with savings of 479.52 seconds in computational time. For the case when loads vary up to 9%, the best compromise solution obtained from the simulation running at Fidelity 2 is also the best compromise solution of the collective non-dominated solution set (shown in light blue and red in Figure 7). On the other hand, while the simulation running at Fidelity 1 produces 419.70 seconds of savings in computational time, very few of the solutions appear in the collective non dominated solution set.

In conclusion, the least emissions solution from the non-dominated solution set obtained from the simulation running at Fidelity 1 becomes a viable operational alternative in this scenario when the changes in the loads are up to 6% at each bus. Otherwise, the results obtained from the simulation running at Fidelity 2 are superior to that of the simulation running at Fidelity 1.

5. Conclusion and Future Work

In this work, a DDDAMS framework involving state-of-the-art information technologies, including data driven simulations, grid computing, web services, sensor network, and database has been presented for environmental and economic dispatch control in distributed power networks. To enable adaptive fidelity switching of the DDDAM-Simulation against available computational resources and sensory updates from the real system, two algorithms have also been developed, including a state estimation algorithm, and a culling and fidelity selection algorithm. The proposed framework has been demonstrated on a modified IEEE-30 bus system. Results revealed that when load variations within the considered sub-network is less than 9%, the simulation should be run at Fidelity 1 (only modifying the dispatch of the sources in this sub-network and the swing bus), in order to have accurate results while at the same time saving significantly from computational resources. Extensions to this work are possible in the methodological and technological aspects. Methodological extensions can be performed in organization of the simulation into numerous fidelities, and the role of information sharing. Technologically, the effect of integrating high-speed sensor networks into the DDDAMS system on the system performance can be studied.

Acknowledgements

This research was supported in part by the Air Force Office of Scientific Research.

References

[1] Kougentakis A, Kenworthy T, Weiss DJ. Clean energy for the wild blue yonder: Expanding renewable energy and efficiency in the Air Force, *Report submitted to Center for American Progress*. 2009.
[2] CNA Analysis and Solutions, Military Advisory Board. Powering America's defense: Energy and the risks to national security. 2009.
[3] Galvin Electricity Initiative. Master controller requirements specification for perfect power systems. 2007. Accessed June 2012 at http://www.galvinpower.org/sites/default/files /documents/MasterController_VCRevision.pdf.
[4] Darema F. Dynamic data driven applications system: A new paradigm for application simulations and measurements, In Proceedings of International Conference on Computational Science 2004: 662-669.
[5] Douglas CC, Shannon CE, Efendiev Y, Ewing R, Ginting V, Lazarov R, Cole MJ, Jones G, Johnson CR, Simpson J. A note on data driven contaminant simulation, Lecture Notes in Computer Science 2004: 701-708.
[6] Mandel J, Chen M, Franca LP, Johns C, Puhalskii A, Coen JL, Douglas CC, Kremens R, Vodacek A, Zhao W. A note on dynamic data driven wildfire modeling, Lecture Notes in Computer Science 2004: 725-731.
[7] Carnahan JC, Reynolds PF. Requirements for DDDAS flexible point support, In Proceedings of the 2006 Winter Simulation Conference. 2006: 2101-2108.
[8] Parnas DL. Designing software for ease of extension and contraction, *IEEE Transactions on Software Engineering* 1979; **5**: 128-138.
[9] Douglas CC, Lodder RA, Beezley JD., Mandel J, Ewing RE, Efendiev Y, Qin G, Iskandarani M, Coen J, Vodacek A, Kritz M, and Haase G. DDDAS approaches to wildland fire modeling and contaminant tracking, In Proceedings of the 2006 Winter Simulation Conference 2006: 2117-2124.
[10] Patrikalakis NM, McCarthy JJ, Robinson AR, Schmidt H, Evangelinos C, Haley PJ, Lalis S, Lermusiaux PFJ, Tian R, Leslie WG, and Cho W. Towards a dynamic data driven system for rapid adaptive interdisciplinary ocean forecasting. Invited Paper in Dynamic Data Driven Application Systems, Darema, F., editor. Kluwer Academic Publishers, Amsterdam 2004.
[11] Son YJ, Joshi SB, Wysk RA, Smith JS. Simulation-based shop floor control, Journal of Manufacturing Systems 2002; **21**(5): 380-394.
[12] Wysk R, Hall D, Pegden D, Son Y, McGinnis L, Zhou C. ITR: Collaborative Research: As times go on - adaptive and scalable time synchronization mechanism for federations of distributed simulations. Unpublished Report. 2004.
[13] Lee S, Zhao X., Shendarkar A, Vasudevan K, Son Y. Fully dynamic epoch (FDE) time synchronization method for distributed supply chain simulation, *International Journal of Computer Applications in Technology* 2008; **31**(3-4): 249-262.
[14] Lee D, Dongarra J, Ramakrishna RS. visPerf: monitoring tool for grid computing. Computational Science. In Proceedings of International Conference on Computational Science 2003; **2659** :233-243
[15] Abido MA. Multiobjective evolutionary algorithms for electric power dispatch problem. *IEEE Trans Evol Comput* 2006; **10**: 315-329.