01 Oct 2010

Distributed Power Balancing for the FREEDM System

Rav Akella

Fanjun Meng

Derek Ditch

Bruce M. McMillin
Missouri University of Science and Technology, ff@mst.edu

et. al. For a complete list of authors, see https://scholarsmine.mst.edu/comsci_facwork/186

Recommended Citation

R. Akella et al., "Distributed Power Balancing for the FREEDM System," *Proceedings of the 2010 First IEEE International Conference on Smart Grid Communications (SmartGridComm) (2010, Gaithersburg, MD)*, pp. 7-12, Institute of Electrical and Electronics Engineers (IEEE), Oct 2010.
The definitive version is available at https://doi.org/10.1109/SMARTGRID.2010.5622003

Distributed Power Balancing for the FREEDM System

Ravi Akella[†], Fanjun Meng[*], Derek Ditch [†], Bruce McMillin[†], and Mariesa Crow[*]

[†]*Department of Computer Science*
[*]*Department of Electrical Engineering*
Missouri University of Science and Technology, Rolla, MO, USA 65409-0350
{ rcaq5c, fmtx9, dpdm85, ff, crow}@mst.edu

***Abstract*—The FREEDM microgrid is a test bed for a smart grid integrated with Distributed Grid Intelligence (DGI) to efficiently manage the distribution and storage of renewable energy. Within the FREEDM system, DGI applies distributed algorithms in a unique way to achieve economically feasible utilization and storage of alternative energy sources in a distributed fashion. The FREEDM microgrid consists of residential or industrial nodes with each node running a portion of the DGI process called Intelligent Energy Management (IEM). Such IEM nodes within FREEDM coordinate among themselves to efficiently and economically manage their power generation, utility and storage. Among a variety of services offered by the DGI, the *Power Balancing* scheme optimizes the distribution of power generation and storage among the IEMs. This paper presents the key aspects in implementing such a scheme and outlines the preliminary results obtained by integrating the proposed methodology with a functional SST model of FREEDM. The results demonstrate the potential benefits of adopting advanced 'smart' technology on a local grid.**

I. INTRODUCTION

The nation's electric power infrastructure is under challenges from grid growth in size, scale and complexity every day. There is growing belief among policymakers, business leaders, and other key stakeholders, around the idea that a smart grid is not only needed but well within reach. FREEDM microgrid is a smart grid with these goals of power and energy management and reliability enhancement achieved with advanced technologies of Solid State Transformer (SST), Distributed Renewable Energy Resource (DRER), Distributed Energy Storage Device (DESD) powered with Distributed Grid Intelligence (DGI).

Each node in the FREEDM system includes an SST, load, photovoltaic (PV) generation and a stationary battery. The DGI is a major cyber aspect in the FREEDM system with each node running a portion of DGI within the Intelligent Energy Management (IEM). The IEM nodes integrated with the SST at each household coordinate to manage the utilization, storage and distribution over the distributed micro grid. Non-uniformity of power utilization due to differences in household and peak hours along with uncertainty in renewable energy generation are some of the major challenges to be addressed in such a smart distribution grid. DGI renders a variety of services to each residential node through smart power management to balance the energy associated with DRER, load (utilization) and storage associated with the high-capacity battery. The IEM controls and reacts to the SST by computing a strategy which also involves migration of power through a gateway that connects an SST to the distribution bus. Among various algorithms adopted by the DGI is the proposed *Power Balancing* scheme to balance power flow through efficient distribution of energy and optimization of economics with in the system. This paper presents the key aspects in implementing such a scheme and outlines the preliminary results obtained by integrating the proposed methodology with a functional SST model of FREEDM.

II. Implementation of the Power balancing scheme on the FREEDM test bed

A. FREEDM microgrid test bed

The FREEDM microgrid is a test bed simulation for the FREEDM Intelligent Energy Management (IEM). The microgrid is a single phase 7.2kV distributed system. At each IEM node, the 7.2/0.12kV SST has its primary side connected to the microgrid feeder, and its secondary side to residential loads, DRER, and DESD. Currently, three IEM nodes are installed in a simulated microgrid. This simulation is developed under SimPowerSystem®.

- SST is a power electronic based transformer designed A) to regulate active power flow tracking the control signal from DGI processes; B) to regulate unit power factor; C) to ride through temporary load imbalance and voltage sag; D) to regulate voltage level at both sides, etc. The functional SST model adopted for current research is limited to functions A and B.
- Typical DRER includes photovoltaic and wind energy. The PV installation is scalable at each IEM node to meet system design and customer requirements. One unit set of solar panels is rated 400VDC, 3kW at maximum insolation.
- A stationary battery is the most common DESD in current industry applications. A lead-acid stationary battery is selected out from a number of candidates as FREEDM DESD because of relative low cost and high reliability. The battery is rated at 120VDC, 35Ah, and can be scaled by changing the size of battery rack.

978-1-4244-6511-8/10/$26.00 ©2010 IEEE

- Load, DRER and DESD at IEM nodes are chosen in different scales in order to study different scenarios which are discussed in Section IV.

IEM	Total Load P(kW)	Total Load Q(kVA)
Node 1	10.020	7.010
Node 2	8.010	4.390
Node 3	8.010	4.620

TABLE I
FREEDM TEST BED: SPECIFICATIONS FOR EACH IEM

B. Distributed power balancing scheme

Distributed load balancing algorithms in computer science are designed to normalize the load of process execution among the peers of a distributed system. Intuitively, the nodes participating in the load balancing algorithm communicate their load changes with each other in an attempt to migrate the process execution task from a node with $High$ load to a node with Low load. The result of such a migration is that the nodes normalize their loads, thereby making the system stable. In this work, one such dynamic process migration scheme [1] is extended beyond its design to efficiently manage resources in a power distribution grid. The power balancing algorithm in the context of FREEDM micro grid power balancing scheme is explained below.

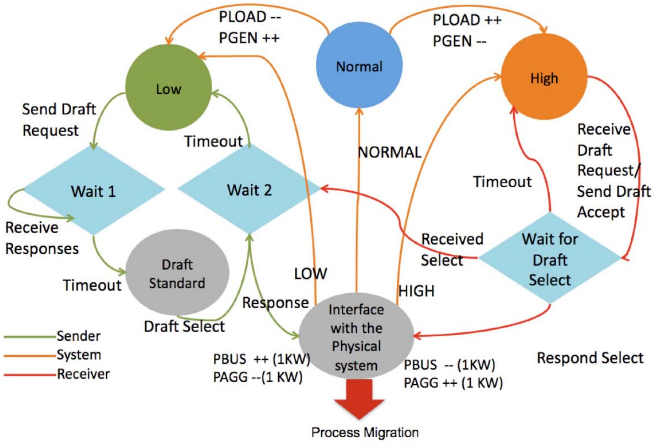

Fig. 1. State diagram of a distributed power balancing scheme

1) Simple version: Every IEM computes the SST's actual load on the distribution grid which can be defined as in Equation (1).

$$X_{Actual} = X_{Load} - X_{DRER} \tag{1}$$

where, X_{Actual} is the effective load which determines whether the node can $Supply$ or is in $Demand$, X_{Load} is the house load at the SST and X_{DRER} is the power generated by the distributed renewable energy source.

The IEM node is in a Low load state if $X_{Actual} < 0$, meaning that it has excess generation to $Supply$. It is in a $High$ load state if $X_{Actual} > Threshold$, where $Threshold$ can be decided based on an optimization heuristic. Otherwise, the IEM node is in a $Normal$ state. Here, the power that could be supplied by DESD (when it is discharging) is not considered since we do not consider the case in which the local battery discharges, to put power on to the shared bus. Also, it is assumed that the DESD is charging when the node is in a $Supply$ state and no other node is in a $Demand$ state. This DGI scheme consists of concurrent sub-processes with message passing communication among the IEMs on critical load changes. Each IEM node maintains a $Load\ table$ to store information it receives about other nodes in the system. Load table updating strategies are adopted to minimize cyber message traffic during frequent load changes. The power balancing process is triggered if at least one IEM node advertises a change of state from $Normal$ load level. An IEM node, on entering in to a Low load state advertises a $Draft\ Request$ message to the nodes in its load table that are in $High$ state and waits for response. A $High$ node, on receiving a $Draft\ Request$ message, responds to the sender by sending its demand with a special message called $Draft\ Age$. The $Draft\ Age$ which includes the demand to be met by the $High$ node in order to reach a $Normal$ load level, is evaluated as in Equation (2).

$$Draft\ Age = X_{Actual} - Threshold \tag{2}$$

The Low node, on receiving Draft Ages from different $High$ nodes, will compute a $Draft\ Standard$ which is a selection of the node it is going to supply power to by evaluation of factors like its own predicted need, economics and other optimization metrics. The $Draft\ Age$ and $Draft\ Standard$ provide a means to incorporate multi-objective metrics for optimal and economic models of power distribution and management. For simplicity, each Low node responds to the request in a First-in First-out (FIFO) order. The Low node, on computation of draft standard, sends a unique $Draft\ Select$ message and initiates power migration by setting $P*$, the set point of the local SST's individual contribution to the shared power bus (2). On receiving the $Draft\ Select$ message from the Low node, the IEM node which was in demand obtains this power from the shared bus. The algorithm sets the migratable power in quanta of 1 KW; this means that each time a successful drafting takes place, 1 KW of power is migrated from the Low node to selected $High$ node. The migration takes place in unit step size until the time the Low node can supply to the demanding $High$ node or the $High$ node meets its sufficient demand or there is a change of load state in either of the nodes. The algorithm continues until all the nodes are in $Normal$ state. Figure 1 shows the process state diagram of a node participating in the power balancing algorithm. The results with implementation of such a simplistic power balancing scheme are presented in Section IV-A.

2) Power balancing with cost bidding: A natural extension of the scheme outlined in Section II-B1 is cost bidding. Cost is needed to incorporate economic metrics, fair distribution practices and to minimize the overall utility cost. Also, the DESD cannot be utilized efficiently with the simple power balancing scheme. The $High$ node advertises its demand cost along with the $Draft\ Age$ as in equation (3).

$$Cost_{High} = Draft\ Age * 100 \tag{3}$$

With every unit of power a $High$ node receives from the shared power bus (in response to a migration from a Low node), the associated cost is decreased by a factor of 100* 1 KW and the updated cost is advertised with subsequent migration requests. This multiplication factor of 100 is randomly chosen to prioritize the distribution from DRER generation against the neighboring battery. Also, it is assumed that the DESD at $High$ node would provide it with the sufficient energy until the time a Low node is ready to supply from its presumably cheaper DRER. For a Low node, the cost is evaluated as in Equation (4).

$$Cost_{Low} = 100 * X_{DRER} + X_{DESD} \tag{4}$$

With every migration step involving 1 KW of power, $Cost_{Low}$ is decremented by a factor of 100 representing the migration from DRER. Power migration does not take place once this cost is less than the DRER multiplication factor of 100, indicating that the cost remaining is associated with the battery which could be used now on a conditional basis by querying its $State\ of\ Charge$. The SST will automatically consume power from the utility to meet $inadequate\ aggregate\ demand$ as long as the utility electric is available with cheaper cost than DESD. The results with implementation of this power balancing scheme with cost bidding scheme are presented in Sections IV-B and IV-C.

C. Integration of power balancing scheme with Simulink model of SST

The experimental setup shown in Figure 2 constitutes of a virtual machine environment simulating the DGI process at each node, a Simulink model of the power system running on another machine and an S-Function that interfaces between the Simulink model and DGI process. The virtual nodes communicate with each other over virtual ethernet on one machine. Each virtual node makes calls to the simulation via different instances of the S-Function on unique ports. At each node, the $Power\ Balancing$ process obtains key parameters like X_{Load}, X_{DRER}, X_{DESD} and $X_{Gateway}$ from the local SST from time to time. In the experimental setup, the $SFUNC$ acts as the micro controller which interfaces the DGI process with the FREEDM micro grid model. The $Power\ Balancing$ process evaluates these parameters to obtain a state decision by which it classifies the node as being in $Normal$ state or $High$ state or Low state as mentioned in Section II-B1.

1) SFUNC: The S-Function interface is the means by which developers may extend Simulink models using programming languages such as C/C++, Fortran, or Matlab. The DGI S-Function is implemented using C and C++. It operates as a separate processing thread that provides a communication layer between the simulation and the local DGI process. It defines both the client-side and server-side marshaling protocol, while running an instance of the latter. This marshaling mechanism allows the DGI process to connect and request the system state at its local SST. Each SST has a unique copy of the $SFUNC$ interface, as it would in a real system. Periodically, the DGI process will request the current system state from the simulated SST. The period length on which this occurs is adjustable in our simulation system, but in a full implementation, this would happen as frequently as possible, in order to minimize state inconsistency. After the DGI process requests the system state and performs the power balancing procedures, the DGI process then makes another call to the S-Function interface to change the $P*$ of the simulated SST. The $SFUNC$ thread then writes this value to a local variable that Simulink will check during the next simulated time step.

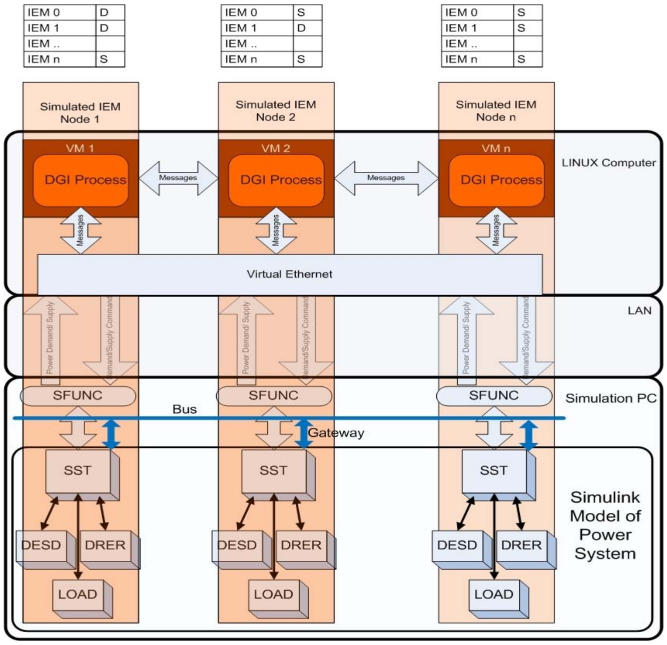

Fig. 2. Simulation Architecture of Power Balancing scheme

Having achieved the appropriate model that integrates well with the power balancing scheme, the following tests were performed and results are presented in Section IV.

III. Algorithm analysis

The algorithms presented in Section II-B not only provide a straight-forward implementation of the classical load balancing problem, but are also provably optimal within a given margin of error. Optimal, in this case, would be the case where all of the distributed renewable energy sources are applied to load demands within the microgrid, minimizing the power consumption from the utility. In this case, there may exist IEM nodes with an excess generation (in Low state), a generation deficiency (in $High$ state), or neither of these (in $Normal$ state). Stated more formally as a decision problem:

Problem 3.1 (Power Migration Decision Problem): Does a power migration strategy exist that migrates power from the m IEM nodes with excess power generation to the n IEM nodes with a generation deficiency, resulting in a draw from the utility grid of at most g units?

$$G = \sum_{i \in I} X_{Actual} = \underbrace{\sum_{i=1}^{n} X_{Load,i}}_{\text{Local Load}} - \underbrace{\sum_{j=1}^{m} X_{DRER,j}}_{\text{Local Capacity}} \quad (5)$$

As detailed in Section II-B2, the cost bidding method provides a way to choose the 'best' power source for a given definition of 'best' by use of a cost metric. This extension to the load balancing algorithm can be modelled by the knapsack problem in computer science. The knapsack problem is placed into a class of algorithms known as NP-Complete [2], which means that there is no known algorithm that can find an answer that is both correct and fast. Specifically, it is not known whether an algorithm exists that can solve a given instance in a time that can be bounded polynomially by the size of the input. However, since the implemented approach migrates quanta of power instead of whole quantities, an approximation algorithm can be used to solve the problem quickly within a given bounds of the optimal answer. This is the approach that the DGI implementation uses.

Theorem 3.1 (Power Migration from Knapsack): There exists a migration strategy with a draw from the utility grid of at most g units if and only if m objects of value $v_1, v_2, \ldots, v_m$ and weights $w_1, w_2, \ldots, w_m$ can be placed into a knapsack such that $\sum_{i=1}^{m} w_i + g = L$, where L is the capacity of the knapsack and equal to the sum of the loads: $L = \sum_{j=1}^{n} X_{Load,j}$ and g is the remaining draw from the utility grid. The goal is to then minimize the value of the knapsack.

Proof: Theorem 3.1 follows from a simple renaming of variables. The m IEM nodes have an excess power generation represented by the weights $w_1, \ldots, w_m$ and the global load demand is L. Then, g is the remaining draw on the utility grid. ■

The core approximation algorithm can be expressed in pseudocode as shown in Algorithm 1. This algorithm uses a greedy approach to solving the problem. This means that it makes the optimal decision for a smaller problem as an approximation to the globally optimal solution. For the knapsack, the greedy approach uses the value of price per unit of power as the minimization decision term. In terms of the knapsack problem, choose the heaviest object with the least value to insert into the knapsack. Do this repeatedly until the knapsack cannot fit anymore whole items. Next, break the remaining items into smaller pieces. Naturally, as the pieces become infinitesimally small, the resulting solution approaches the optimal solution. Since this adds infintely more computations, this reduces the problem to the NP-Complete version of the problem previously discussed. If the size of the pieces is limited, however, the solution is guaranteed to be within a bounds of the optimal solution.

Let C be the cost of the least expensive power source (e.g. $C = \min_{x \in X_{DRER}}(cost(x))$). Then mC is a lower bound on the cost that can be achieved from any solution. Let ϵ be a parameter of the amount of error in the solution. A polynomial-time approximation algorithm can then be devised in terms of m and $1/\epsilon$.

Algorithm 1 Least Cost Fractional Knapsack

```
1: Given $\epsilon > 0$, let $K = \frac{\epsilon C}{m}$.
2: For each source $s_i$, define $cost'(s_i) = \left\lfloor \frac{cost(s_i)}{K} \right\rfloor$
3: Add up to $K$ entries of each source in increasing order of
   $cost'(s_i)$ into the set $S'$, such that $\sum_{s \in S'} cost'(s) \leq L$.
4: Output $S'$, the least cost set.
```

Theorem 3.2: The least cost set, S' output by the algorithm satisfies: $cost(S') \leq (1 + \epsilon) \cdot OPT$

Proof: Let O denote the optimal set. For any source s, because of rounding down, $K \cdot cost'(s)$ can be larger than $cost(s)$, but not by more than K. Therefore,

$$K \cdot cost'(O) - cost(O) \leq mK$$

Line 3 of the algorithm must return a set at least as good as O under the new costs. Therefore,

$$\begin{aligned} cost(S') \leq K \cdot cost'(O) &\leq cost(O) + mK = OPT + \epsilon C \\ &\leq (1 + \epsilon) \cdot OPT, \end{aligned}$$

where the last line follows the observation that $OPT \geq C$[3]. ■

For the FREEDM DGI, Theorem 3.2 shows that given a quanta of 1 kW, if the optimal draw from the utility was 15 kW, the algorithm would guarantee a solution of no more than $\frac{1kW}{15kW} = 0.0\overline{6} \cdot OPT$ above the optimal solution.

IV. Results

Simulations were executed on a Dell M4400 2.53GHz multicore computer with three additional Virtual Machines implementing the DGI processes in each of the three IEM nodes. Three simulation tests are reported to depict basic operation of the system, Test101, Test103 and Test203. In each test, various power and energy changes were introduced into the Simulink simulation. These processes detected and broadcasted change of their local state (X_{Load}, X_{DRER}, X_{DESD}, and $X_{Gateway}$ with an applied cost function), negotiated, and executed power transfers.

A. Test 101: Power migration from single IEM node without cost factor

IEM 01 without DRER and DESD: Major load increase to $High$ load status (t = 0.525s). The $High$ load broadcasts the demand of extra power in order to reduce the area (FREEDM microgrid) electric consumption. IEM 02 with DESD and large scale DRER: PV generation is used to charge local battery (at low SOC) since its local load is at $Normal$ state. Then the extra power migrates to IEM 01 to relieve the $High$ load demand (t = 0.5852s). The battery is discharged to meet the inadequate demand at IEM 01 (t = 1.061s) since energy cost is not introduced in this scenario. IEM 03 with DRER and DESD: load increase is within $Normal$ level; therefore PV generation is used to reduce IEM node total consumption. The battery idles to avoid unnecessary discharge/charge.

Execution trace at IEM 01	Execution trace at IEM 02
Time: 6.01035, on this HIGH node, Set P*: 8.82826 Time: 6.01035, cost (Draft age) of: 482 sent to: IEM 02 ⋮ Time: 6.04756, received 1KW supply from: IEM 02 ....Cost after migration: 382 Time: 6.8112, cost (Draft age) of: 383 sent to: IEM 02 ⋮ Time: 6.9986, received 1KW supply from: IEM 02 ....Cost after migration: 283 Time: 7.7798, cost (Draft age) of: 281 sent to: IEM 02 ⋮ Time: 7.9798, received 1KW supply from: IEM 02 ....Cost after migration: 181 Time: 8.722, on this HIGH node Set P*: 8.83601 Time: 8.722, cost (Draft age) of: 483 sent to: IEM 02 ⋮ Time: 8.923, received 1KW supply from: IEM 02 ....Cost after migration: 383 Time: 9.70495, on this HIGH node Set P*: 8.83879 Time: 9.70495, cost (Draft age) of: 383 sent to: IEM 02 ⋮ Time: 9.90634, received 1KW supply from: IEM 02 ....Cost after migration: 283 Time: 10.7354, cost (Draft age) of: 284 sent to: IEM 02 ⋮ Time: 10.9354, received 1KW supply from: IEM 02 ....Cost after migration: 184 Time: 11.7641, on this HIGH node Set P*: 8.8435 Time: 11.7641, cost (Draft age) of: 484 sent to: IEM 02	⋮ Time: 6.0224, Draft age of: 482 received from: IEM 01 Time: 6.0224, send Draft Select to: IEM 01 Time: 6.0224, supplied 1 KW load to: IEM 01 ....Set P* = 1.2145, Cost after migration: -379 ⋮ Time: 6.9934, Draft age of: 383 received from: IEM 01 Time: 6.9934, send Draft Select to: IEM 01 Time: 6.9934, supplied 1 KW load to: IEM 01 ....Set P* = 0.2145, Cost after migration: -278 ⋮ Time: 7.9646, Draft age of: 281 received from: IEM 01 Time: 7.9646, send Draft Select to: IEM 01 Time: 7.9646, supplied 1 KW load to: IEM 01 ....Set P* = -0.7855, Cost after migration: -177 ⋮ Time: 8.90571, send Draft Request to: IEM 01 Time: 8.91115, new cost at this LOW node: -476 Time: 8.91115, Draft age of: 483 received from: IEM 01 Time: 8.91115, send Draft Select to: IEM 01 Time: 8.91115, supplied 1 KW load to: IEM 01 ....Set P* = -1.7855, Cost after migration: -376 ⋮ Time: 9.8998, Draft age of: 383 received from: IEM 01 Time: 9.8998, send Draft Select to: IEM 01 Time: 9.8998, supplied 1 KW load to: IEM 01 ....Set P* = -2.7855, Cost after migration: -275 ⋮ Time: 10.9288, Draft age of: 284 received from: IEM 01 Time: 10.9288, send Draft Select to: IEM 01 Time: 10.9288, supplied 1 KW load to: IEM 01 ....Set P* = -3.7855, Cost after migration: -174 ⋮

TABLE II
MIGRATION STEPS FROM EXECUTION TRACES OF TEST 103 GENERATED BY THE POWER BALANCING ALGORITHM

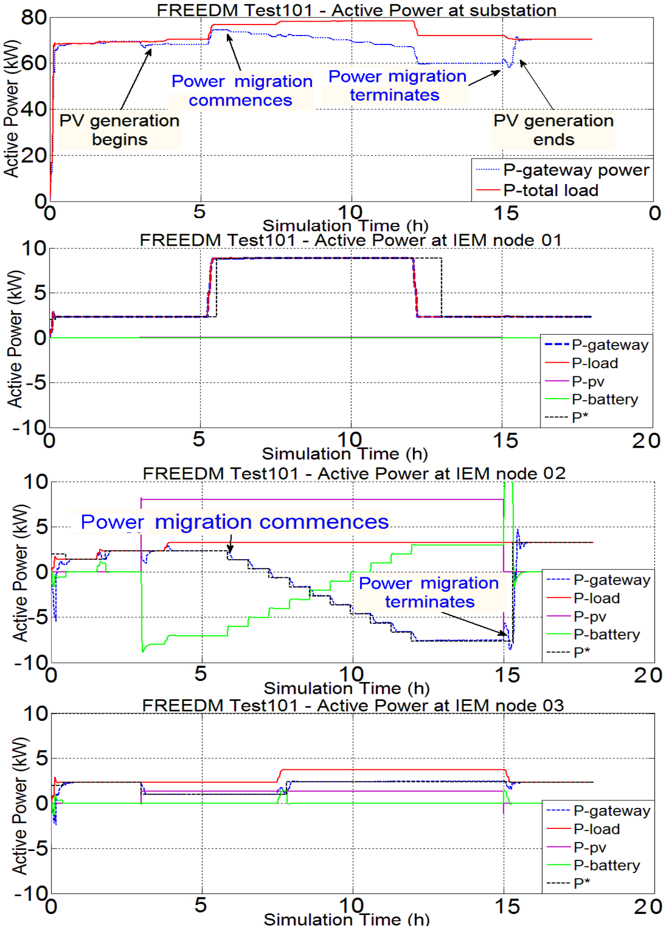

Fig. 3. Test 101: Basic 2-Node Migration

B. Test 103: Power migration with cost bidding, from PV of single IEM node

To achieve economic optimization and efficient migration of power across the microgrid, we incorporate a cost function in the power balancing scheme. In this test, we set the battery energy cost higher than the PV generation and utility grid energy cost. It can be observed from the result of this test in Figure 4 that IEM 01 only receives the excess generation from DRER of IEM 02, but not from the DESD. The inadequate demand is automatically compensated for by utility power. The corresponding traces generated by the power balancing scheme at IEM 01 and 02 are presented in Table II.

C. Test 203: Power migration from multiple IEM nodes with cost bidding

This test is a natural extension of Test103 with IEM 03 transitioning to *Low* state; bidding competition between IEM 02 and IEM 03 occurs in this test. Similar to Test101, IEM 02 begins migrating power to IEM 01 to relieve the *High* demand. IEM 03 also reaches *Low* load state at (t = 0.9761s) and becomes a supplier candidate. So IEM 02 and 03 both migrate power to IEM 01. The assumption of on-peak hour is adopted and therefore the battery power overbids the utility power.

V. CONCLUSION AND FUTURE WORK

The proposed power balancing scheme was successfully integrated with the functional model of the FREEDM system and it was shown to provide power distribution and management. Algorithmic properties of power balancing with respect

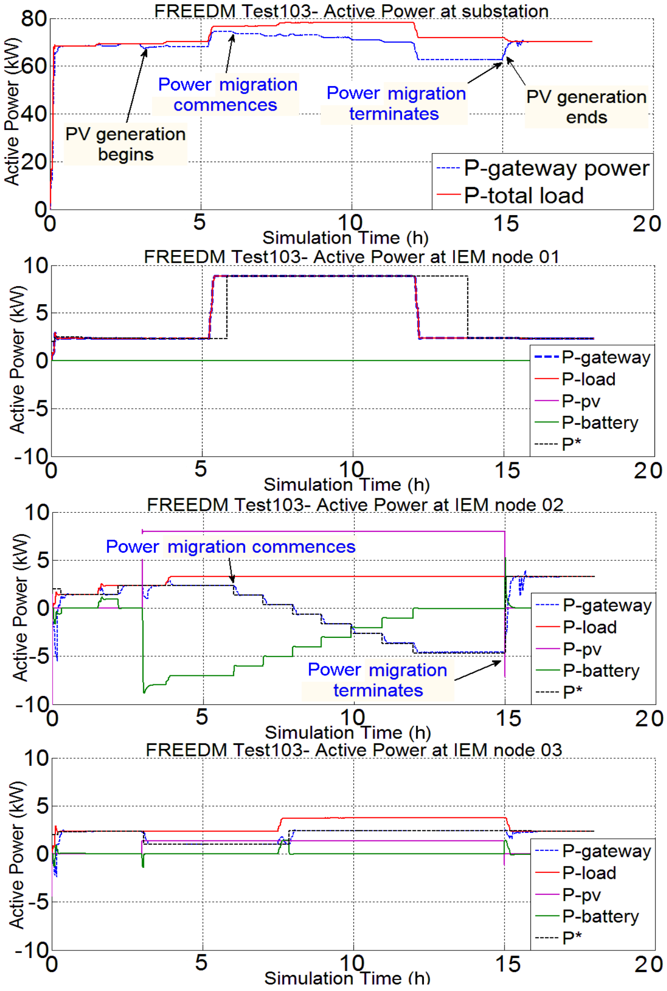

Fig. 4. Test 103: 2-Node Migration with cost function

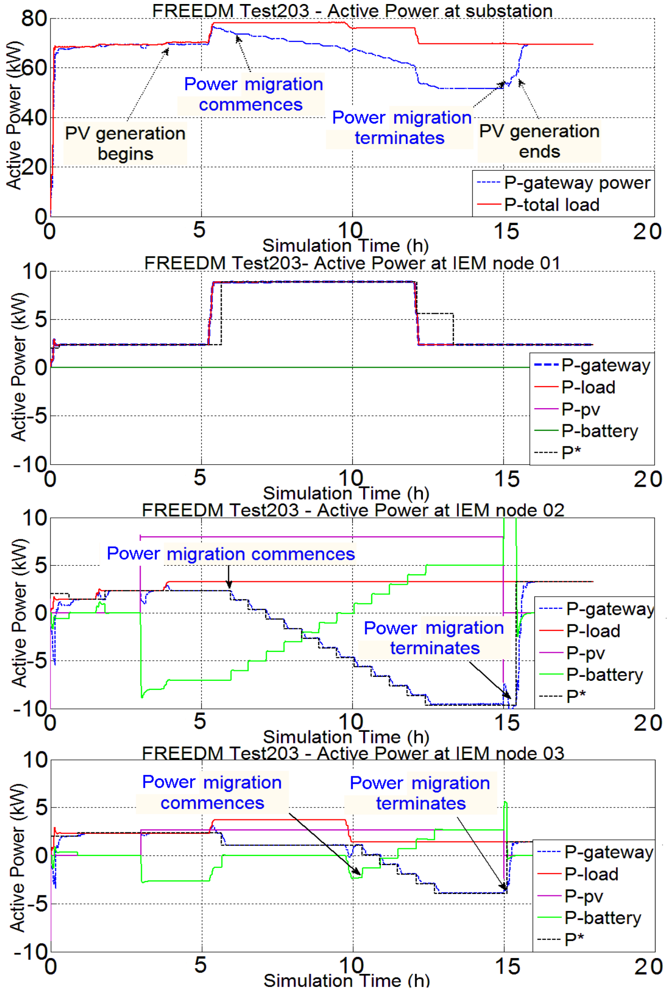

Fig. 5. Test 203: 3-Node Migration

to multi-objective optimization constraints were analyzed in [4]. Future work involves optimization of the bidding scheme to benchmark power management against multi-objective optimization constraints [5], analyzing the effects of DGI processes on the microgrid and utility grid power quality, and analyzing stability issues using an average model or switching model of the SST.

VI. Acknowledgement

The authors acknowledge the support of the Future Renewable Electric Energy Distribution Management Center; a National Science Foundation supported Engineering Research Center, under grant NSF EEC-0812121 and NSF CSR award CCF-0614633.

References

[1] L. M. Ni, C.-W. Xu, and T. B. Gendreau, "A distributed drafting algorithm for load balancing," *IEEE Trans. Softw. Eng.*, vol. 11, no. 10, pp. 1153–1161, 1985.

[2] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences)*. W. H. Freeman, 1979.

[3] V. V. Vazirani, *Approximation Algorithms*. Springer, March 2004.

[4] D. Ditch and B. McMillin, "Optimality of distributed grid intelligence for power distribution," in *Proceedings of the FREEDM System Center Annual Conference*, Tallahassee, FL, 2010.

[5] B. R. Sathyanarayana, G. T. Heydt, and B. M. McMillin, "Distribution energy management via multiobjective optimization," in *Proceedings of the FREEDM System Center Annual Conference*, Tallahassee, FL, 2010.