Distributed Load Balancing for FREEDM system

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

[*]*Department of Electrical Engineering*
[†]*Department of Computer Science*
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 provides a unique way in applying distributed algorithms to achieve economically feasible and optimal utilization and storage of alternative energy sources in a distributed fashion. The FREEDM microgrid consists of residential nodes with each node running a portion of the DGI process called Intelligent Energy Management (IEM). Such IEM nodes within the FREEDM neighborhood coordinate among themselves to efficiently and economically manage their power generation, utility and storage. Among a variety of services offered by the DGI , the $Load\ Balancing$ 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 the functional SST model of FREEDM. The results demonstrate the potential benefits of adopting advanced 'smart' technology on a regional grid.**

***Index Terms*—Smart grid, FREEDM, Solid-State Transformer, Load Balancing, Distributed system**

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 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 residential node in the FREEDM system includes an SST, house load, photovoltaic (PV) generation and a stationary battery. The DGI is a major cyber aspect in the FREEDM system with each residential node running a portion of DGI, called the Intelligent Energy Management (IEM). The IEM nodes integrated with the SST at each household coordinate to manage the utilization, storage and distribution over 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, house load (utilization) and storage associated with the high-capacity battery. The IEM controls and reacts to the SST by computing an optimal strategy which also involves migration of power through the gateway that connects an SST to the distribution bus. Among various algorithms adopted by the DGI is the proposed $Load\ Balancing$ scheme to efficiently balance power flow through optimal 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 the functional SST model of FREEDM.

II. IMPLEMENTATION OF THE LOAD BALANCING SCHEME ON THE FREEDM TEST BED

A. FREEDM microgrid test bed

The FREEDM microgrid is a test bed simulation for 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 primary side connected to the microgrid feeder, and secondary side to residential loads, DRER and DESD. Currently, three IEM nodes are installed in microgrid. This simulation is developed under SimPowerSystem®.

- SST is a power electronic based transformer 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. A functional SST model is adopted for current research, 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 requirement. One unit set of solar panels is rated 400VDC, 3kW at maximum insolation.
- Stationary battery is the most common DESD in current industry applications. 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.

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

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 load 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, there by 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. Load balancing algorithm in the context of FREEDM micro grid is explained below.

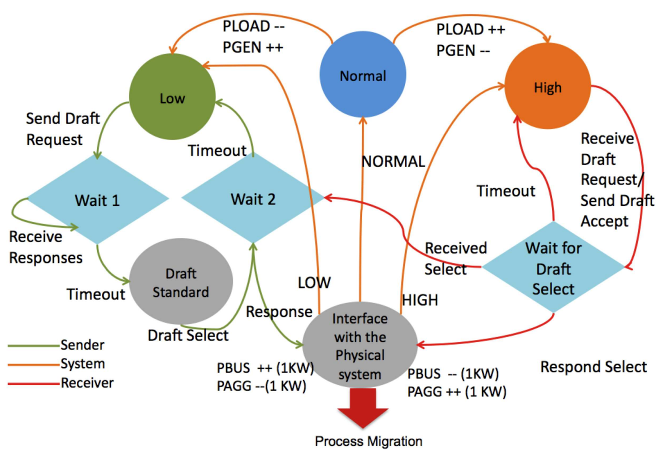

Fig. 1. State diagram of a distributed load balancing scheme

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

$$X_{Actual} = X_{Load} - X_{DRER} \quad (1)$$

where, X_{Actual} is the effective load which determines whether the node can $Supply$ or it 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 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 DESD would be charging when the node is in a Low load state and no other node is in a demanding. However, the DESD element is incorporated in the Load balancing scheme with cost bidding, outlined in Section II-B2. 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 load balancing process is triggered if at least one 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 $Draft\ Request$ message, responds to the sender by sending its demand with a special message called $Draft\ Age$. The $Draft\ Age$ which currently includes the demand to be met by the $High$ node in order to reach a $Normal$ load level, is evaluated as in Equation (2).

$$Draft\ Age = X_{Actual} - Threshold \quad (2)$$

The Low node, on receiving Draft Ages from different $High$ nodes will compute a $Draft\ Standard$ which is an optimized 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. It can be observed that the $Draft\ Age$ and $Draft\ Standard$ provide a means to incorporate multi objective function for optimal and economic models of power distribution and management. For simplicity, currently 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 the power migration by making a setting called $P*$, which is the set point of the local SST's individual contribution on to the shared power bus. On receiving the $Draft\ Select$ message from the Low node, the IEM which was in demand obtains this power from the shared bus. Right now, 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 between the Low node and the selected $High$ node. The migration takes place in unit step size till 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 till all the nodes are in $Normal$ state. SST will automatically consume power from utility to meet $inadequate\ aggregate\ demand$ as long as the utility electric is available with cheaper cost than DESD. Figure 1 shows the process state diagram of a node participating in the Load balancing algorithm. The results with implementation of such a simplistic load balancing scheme are presented in Section III-A.

2) Load balancing with cost bidding: A natural extension of the scheme outlined in Section II-B1 is cost bidding. The potential need for such a leap is to incorporate economic metrics, fair distribution practices and to minimize the overall utility cost. Also, the DESD cannot be utilized efficiently with the simple load balancing scheme. In Section III-A, it can be observed that the battery discharges itself in an attempt to meet the demand requested by the $High$ node. This could however, be avoided by including the cost metric as outlined in this section. The $High$ node would advertise its demand cost along with the $Draft\ Age$ as in equation (3).

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

With every unit of power it receives from the shared power bus (in response to a migration from a Low node), it would decrement its cost by a factor of 100* 1 KW and advertises the updated cost 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 till 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, $Cost_{Low}$ would be 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 constraints associated with such a mechanism like fractional output from DRER (to make $X_{DRER} < 1$), variation of cost due to dynamic changes of power output from DRER and DESD were dealt within the implementation. The results with implementation of this load balancing scheme with cost bidding scheme are presented in Sections III-B and III-C.

C. Integration Load 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 $Load\ Balancing$ 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 $Load\ Balancing$ 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.

The DGI process uses the same communication protocol implementation, in order to maintain consistency. 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 communication delay. After the DGI process requests the system state and performs the load balancing procedures, the DGI process then makes another call to the S-Function interface to change the set-point 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.

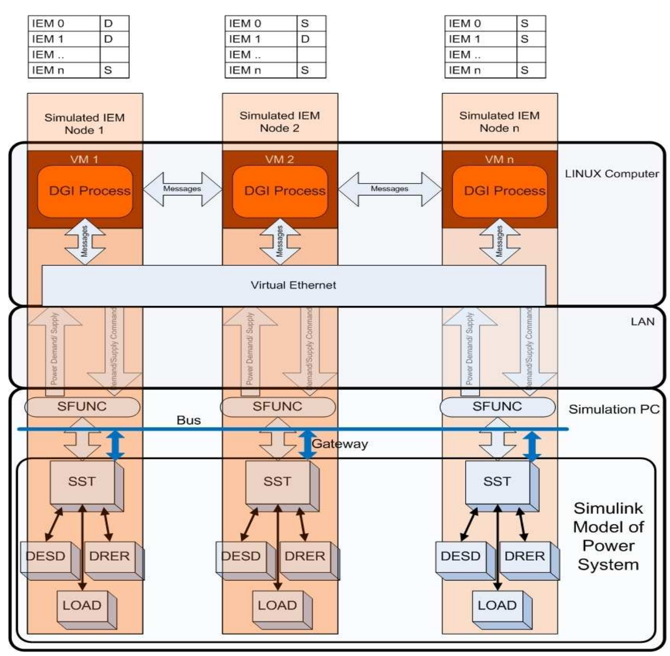

Fig. 2. Simulation Architecture of Load Balancing scheme

Having such an experimental setup, several models of $Functional\ SST$ were developed and tested to match up with the expectations. Having achieved the appropriate model that integrates well with the load balancing scheme, the following tests were performed and results are presented in Section III.

Execution trace at IEM 01	Execution trace at IEM 02
: : : Time: 0.49016, on this NORMAL node, Set P*: 2.32931 Time: 0.601035, on this HIGH node, Set P*: 8.82826 Time: 0.601035, cost (Draft age) of: 482 sent to: IEM 02 Time: 0.604756, received 1KW supply from: IEM 02 ....Cost after migration: 382 Time: 0.68112, cost (Draft age) of: 383 sent to: IEM 02 Time: 0.69986, received 1KW supply from: IEM 02 ....Cost after migration: 283 Time: 0.77798, cost (Draft age) of: 281 sent to: IEM 02 Time: 0.79798, received 1KW supply from: IEM 02 ....Cost after migration: 181 Time: 0.8722, on this HIGH node Set P*: 8.83601 Time: 0.8722, cost (Draft age) of: 483 sent to: IEM 02 Time: 0.8923, received 1KW supply from: IEM 02 ....Cost after migration: 383 Time: 0.970495, on this HIGH node Set P*: 8.83879 Time: 0.970495, cost (Draft age) of: 383 sent to: IEM 02 Time: 0.990634, received 1KW supply from: IEM 02 ....Cost after migration: 283 Time: 1.07354, cost (Draft age) of: 284 sent to: IEM 02 Time: 1.09354, received 1KW supply from: IEM 02 ....Cost after migration: 184 Time: 1.17641, on this HIGH node Set P*: 8.8435 Time: 1.17641, cost (Draft age) of: 484 sent to: IEM 02 Time: 1.19641, received 1KW supply from: IEM 02 ....Cost after migration: 384 Time: 1.38194, on this NORMAL node, Set P*: 2.33382 : :	Time: 0.60224, Draft age of: 482 received from: IEM 01 Time: 0.60224, send Draft Select to: IEM 01 Time: 0.60224, supplied 1 KW load to: IEM 01 ....Set P* = 1.2145, Cost after migration: -379 Time: 0.69934, Draft age of: 383 received from: IEM 01 Time: 0.69934, send Draft Select to: IEM 01 Time: 0.69934, supplied 1 KW load to: IEM 01 ....Set P* = 0.2145, Cost after migration: -278 Time: 0.79646, Draft age of: 281 received from: IEM 01 Time: 0.79646, send Draft Select to: IEM 01 Time: 0.79646, supplied 1 KW load to: IEM 01 ....Set P* = -0.7855, Cost after migration: -177 Time: 0.890571, send Draft Request to: IEM 01 Time: 0.891115, new cost at this LOW node: -476 Time: 0.891115, Draft age of: 483 received from: IEM 01 Time: 0.891115, send Draft Select to: IEM 01 Time: 0.891115, supplied 1 KW load to: IEM 01 ....Set P* = -1.7855, Cost after migration: -376 Time: 0.98998, Draft age of: 383 received from: IEM 01 Time: 0.98998, send Draft Select to: IEM 01 Time: 0.98998, supplied 1 KW load to: IEM 01 ....Set P* = -2.7855, Cost after migration: -275 Time: 1.09288, Draft age of: 284 received from: IEM 01 Time: 1.09288, send Draft Select to: IEM 01 Time: 1.09288, supplied 1 KW load to: IEM 01 ....Set P* = -3.7855, Cost after migration: -174 Time: 1.19573, new cost at this LOW node: -472 Time: 1.19573, Draft age of: 484 received from: IEM 01 Time: 1.19573, send Draft Select to: IEM 01 Time: 1.19573, supplied 1 KW load to: IEM 01 ....Set P* = -4.7855, Cost after migration: -372

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

III. RESULTS

Simulations were executed on a Dell M4400 2.53GHz multicore computer with three additional Virtual Machines implementing the DGI process in each of the three IEM nodes. Three simulation tests are reported to depict basic operation of the system, which are named as Test 101, Test 103 and Test 203 for notational convinience. 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: Energy 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 would broadcast 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 local load is at $Normal$ state. Then the extra power migrates to IEM 01 to relieve $High$ load demand (t = 0.5852s). The battery is discharged to meet the inadequate demand at IEM 01 (t = 1.061s), since energy cost factor is not introduced in this scenario. IEM 03 with DRER and DESD: load increase is within $Normal$ level; therefore

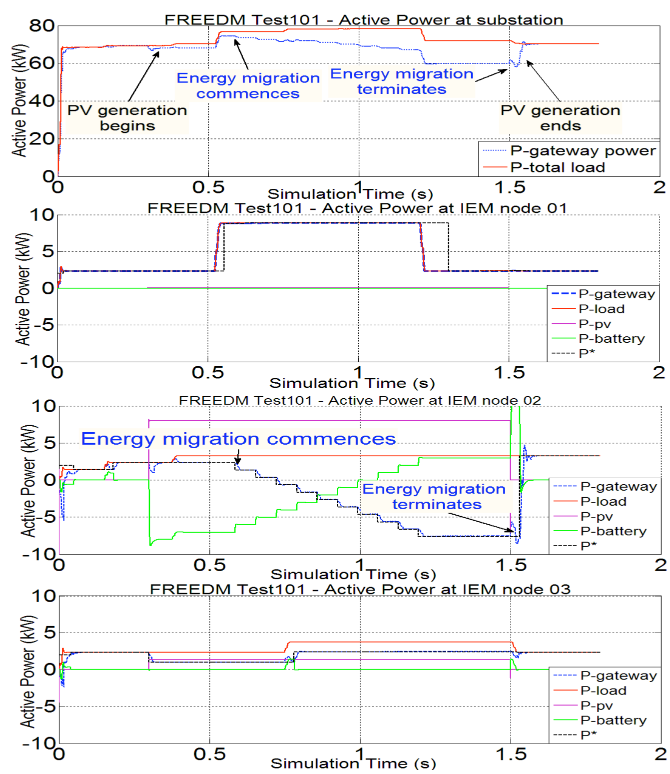

Fig. 3. Test 101: Basic 2-Node Migration

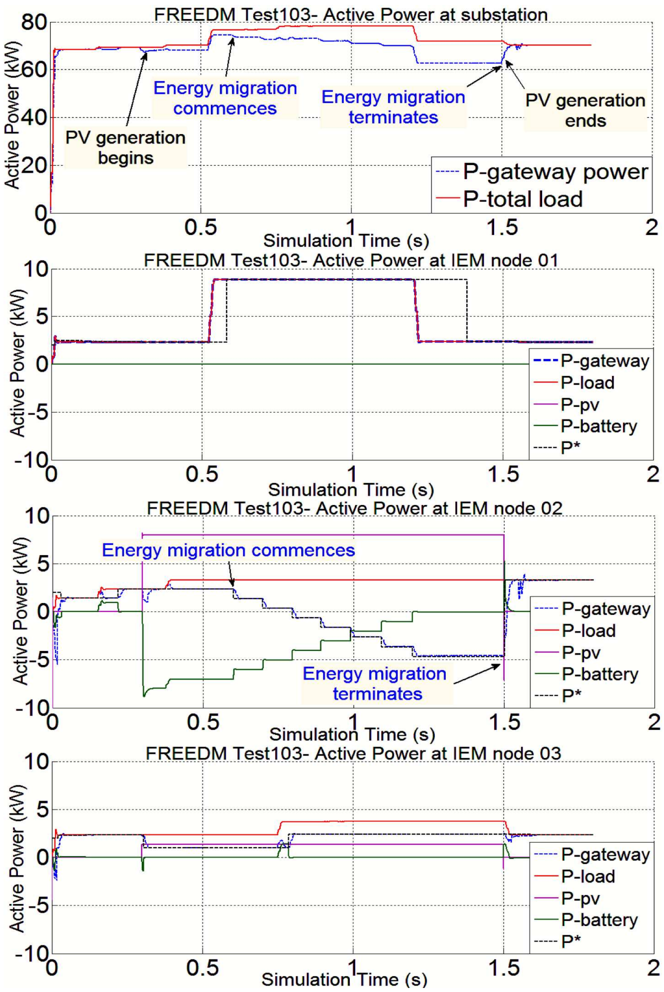

Fig. 4. Test 103: 2-Node Migration with cost function

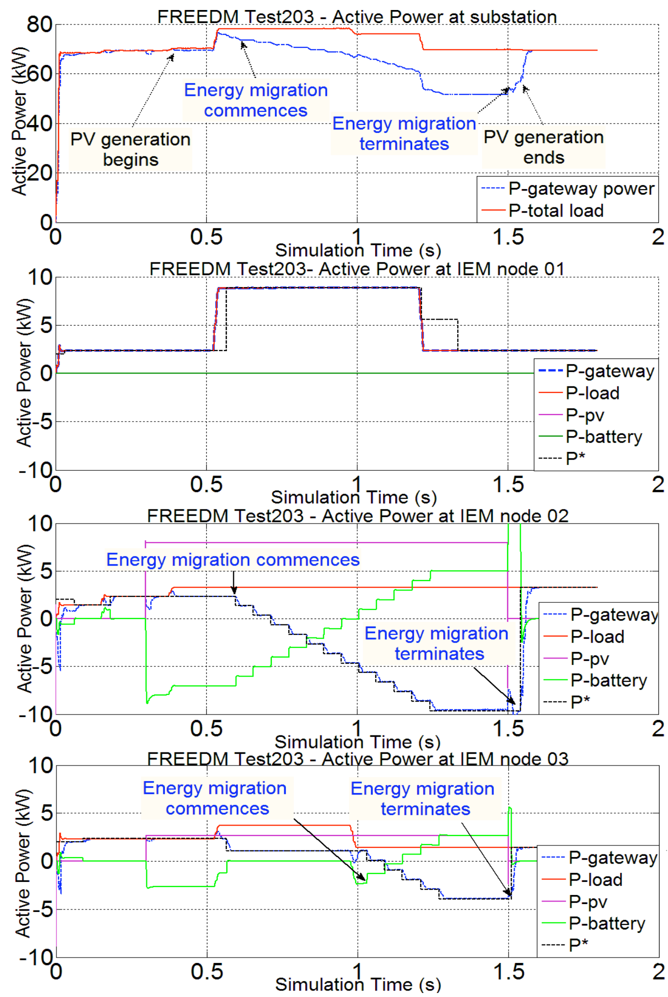

Fig. 5. Test 203: 3-Node Migration

PV generation is used to reduce IEM node total consumption. And battery is idling to avoid unnecessary discharge/charge.

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

To achieve economic optimization and efficient migration of power across the microgrid, we incorporate cost function in the load balancing scheme. In this test, we assuming the battery energy cost is higher than 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 with utility power. The corresponding traces generated by the load balancing scheme at IEM 01 and 02 are presented in Table II.

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

This test is a natural extension of Test 103 with IEM 03 transitioning in to *Low* state; there by, bidding competition between IEM 02 and IEM 03 occurs in this test. Similar to Test 101, IEM 02 begins migrating energy to IEM 01 to relieve the *High* demand. IEM 03 also reaches LOW status at (t = 0.9761s) and becomes a supplier candidate. So IEM 02 and 03 both migrate energy to IEM 01 after cost evaluation process. The assumption of on-peak hour is adopted and therefore the battery energy overbids the utility energy.

IV. Conclusion and Future Work

The proposed load balancing scheme was successfully integrated with the Functional model of the FREEDM system and it was proved to provide reliable and optimal power distribution and management. Algorithmic properties of load balancing with respect to multi-objective optimization constraints were analyzed in [2]. Future work involves optimization of the bidding scheme to benchmark load balancing power management against multi-objective optimization constraints [3], and analysis the effects of DGI process on microgrid and utility grid power quality and stability issues using an average model or switching model of SST.

V. 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] D. Ditch and B. McMillin, “An application of load balancing for optimal power distribution,” in *Proceedings of the FREEDM System Center Annual Conference*, Tallahassee, FL, 2010.

[3] 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.