Modeling Electrical Networks with Object Oriented Data Structures for Smart Control

Taha Selim Ustun,
Carnegie Mellon University, Pittsburgh, PA, USA

ABSTRACT

The electrical networks are becoming more complex with the introduction of distributed generators; power electronics based control devices and various types of loads. The electrical networks are no longer passive meshed systems, as they were a couple decades ago. Rather the generation, transmission and consumption co-exist almost at all levels of the electrical networks. With the increasing number of control parameters the management, protection and control of these electrical networks become a real challenge for power engineers. Furthermore, several loads and/or distributed generators may connect/disconnect from/to the network at any time. Therefore, the shape of the network is not fixed but variable. Power flow, power quality calculations as well as protection parameters such as relay settings, fault current settings etc. require that the exact structure of an electrical network should be monitored. This can be achieved by making use of communication lines in a central management system. In this paper, the electrical nodes such as relays, generators and loads are represented with Object Oriented (OO) data structures to model microgrids with graph theory. Dijkstra's algorithm is implemented to monitor the changes in the network and recognize new deployments. In this manner, the exact structure of the network can be monitored without central management and necessary adjustments can be made on the fly. The simulation of the proposed model shows the viability of such model and its effectiveness.

1. INTRODUCTION

The large-scale deployment of Distributed Generators (DGs) introduced unprecedented problems to power networks [1]. In an effort to tackle these problems, the microgrid concept has been introduced. A microgrid is a collection of loads and microgenerators along with some local storage and behaves just like a model-citizen from grid side thanks to intelligent control [2] .

It is a challenging task to manage microgrids as they have dynamic structures which change very often. The following may be counted among the reasons for the changes in the microgrid structure [3]:

- New DG or load deployments
- Islanding of the system
- Fault conditions
- Reconfiguration of the structure for maintenance

This dynamic behavior of microgrids is a major protection challenge since the conventional selectivity methods assume a fixed network structure and a predetermined relay hierarchy [4]. Whenever restructuring occurs, the selective levels assigned prior to that become erroneous. For a proper operation, the selective levels of relays should follow the changing conditions of the network.

New relay hierarchy should be extracted and corresponding time delays should be assigned before updating them with the help of communication lines [5]. This requires an algorithm which will determine the current structure of the system and yield the relay hierarchy at all branches of the network. There are some studies presented in the literature which emphasize the importance of such an adaptive selective operation such as in [6]. However, the prior discusses the issue qualitatively without any technical details whereas the latter implements an algorithm which includes a look-up table. This is a large set-back because it requires the knowledge of all possible microgrid configurations beforehand, plus human input for the preparation of this table and finally it requires that the microgrid should always match one of the predetermined structures. Moreover, any kind of a new deployment, which is very common to microgrids, requires that the whole selectivity table should be re-written.

The modeling of electrical networks with the OO models proposed in this paper and the implementation of Dijkstra's algorithm on it will make microgrid management easier from power flow, generation, load sharing and/or protection aspects.

The organization of this paper is as follows: Section II summarizes the challenges due to dynamic structure of the microgrid, Section III outlines the proposed modeling for management, Section IV sheds light on the implementation of graph theory and Dijkstra's Algorithm. It also, studies several case studies for microgrid reconfiguration and new deployments. Finally, Section V draws the conclusions.

2. DYNAMIC STRUCTURE OF MICROGRIDS

One of the key features of microgrids is their dynamic behavior. The connection/disconnection of a relay, load or generator at any given instance impacts the operation [7]. Connection of a load or a generator changes the load flow and generation settings. Therefore, the generation settings of the generators shall be updated, accordingly. Connection or disconnection of a relay changes the structure of a network and it requires adjustments. To further elaborate the challenges, as an example, we shall focus on the protection challenges due to dynamic behavior of microgrids . The challenges from other aspects can be detailed in similar fashion.

Selectivity is a well known protection concept which means isolating the fault with the nearest relay in an effort to minimize its effect on the rest of the system. This requires that in case of a fault, the relays should react according to a hierarchy. In conventional protection systems designed for passive networks, the relays which are downstream and closer to the fault point are required to operate first. However, if the fault current is very large and downstream relays are not capable of interrupting it, then other relays with larger capacities are expected to operate and isolate the fault. Implementation of selectivity is not that straightforward with the introduction of DGs. The very concepts of downstream and upstream relays are prone to change according to the status of the microgrid. The operating mode, i.e. grid-connected or islanded-mode, changing network structure with alternative paths and new deployments are some of the factors that would alter the selectivity parameters.

Consider the system shown in Figure 1. In this network, all branches have generation and load, and various alternative network structures can be formed through the combination of relays.

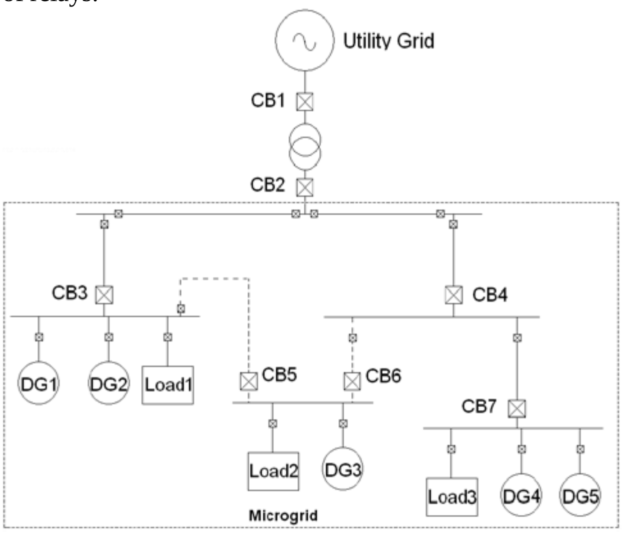

Figure 1: A sample microgrid

As first case, assume that the Circuit Breakers (CBs) CB1, CB2, CB3, CB4, CB6 and CB7 are closed whereas CB5 remains open. When a fault occurs at the terminals of Load 2, then the most downstream relay will be Load 2's own relay (represented by the little box) and selectivity implies that it should interrupt the connection. If Load 2's relay fails to achieve that in a predetermined time (delay), then the proper sequence for the selective operation should be CB6, CB4 and finally CB2. In similar fashion, should a fault occur at the terminals of Load 3, the proper selective operation requires the sequence: Load 3's relay, CB7, CB4 and CB2.

If CB4 is disconnected for any reason, for example maintenance or breakdown, in order to keep the integrity of the network CB5 closes. The line between Load 1 and Load 2 (protected by CB5) has therefore been added to form a loop structure when necessary and protect the microgrid against contingencies and failures.
Now, there is only one branch for the power flow instead of two. For this new microgrid structure all selective levels, time steps and time delay calculations shall be repeated. Following the same examples should a fault occur at Load 2 or Load 3, the proper relay hierarchies are; Load2's relay, CB5, CB3, CB2 and Load 3's relay, CB6, CB5, CB3, CB2, respectively.

The above mentioned factors require that the selectivity hierarchy of the relays should be dynamic and updated frequently. An algorithm should be employed which determines the network structure whenever the status of a critical relay is changed. A critical relay refers to a relay the status of which changes the structure of the network. Following this definition relays of CB2, CB3, CB4, CB5 and CB6 are all critical relays whereas Load 2's relay, DG1's relay are non-critical relays.

3. OBJECT ORIENTED MODELING OF ELECTRICAL NETWORKS

As mentioned in the previous section the varying structure of the microgrid requires a system which can represent the network in computer environment and monitor the changes occurring therein. In this manner, the operation settings of protective devices, generators, loads and other auxiliaries can be calculated by a central microgrid controller and updated into relative devices [8] [9]. When all the connected devices are recognized as nodes and their connection/disconnection is followed in the modeling system, then the microgrid can be defined with different methods such as the graph theory.

Over the years, the international standard IEC 61850 has defined many OO data and communication models for power system networks especially for substations. However, IEC 61850 and its various parts are continuously evolving with new additions and amendments. In this paper, the authors wish to propose one such future amendment. The authors believe that there is a significant need for a data model to represent the information with regards to the various node points such as bus bars along the distribution network. Such a data model would allow valuable information such as load profile or generation capacity connected to a particular point within the network to be communicated across to control equipment.

Thus, the authors are proposing the Electrical Network Node (ENN) model shown in Figure 2. This node is defined by following OO Modeling rules and Unified Model Language (UML) representation [10]. The node includes some public data to represent its properties such as node ID, operating settings `*node settings*` which vary for different node classes and connection data such as, '*IDs of the connected nodes*'.

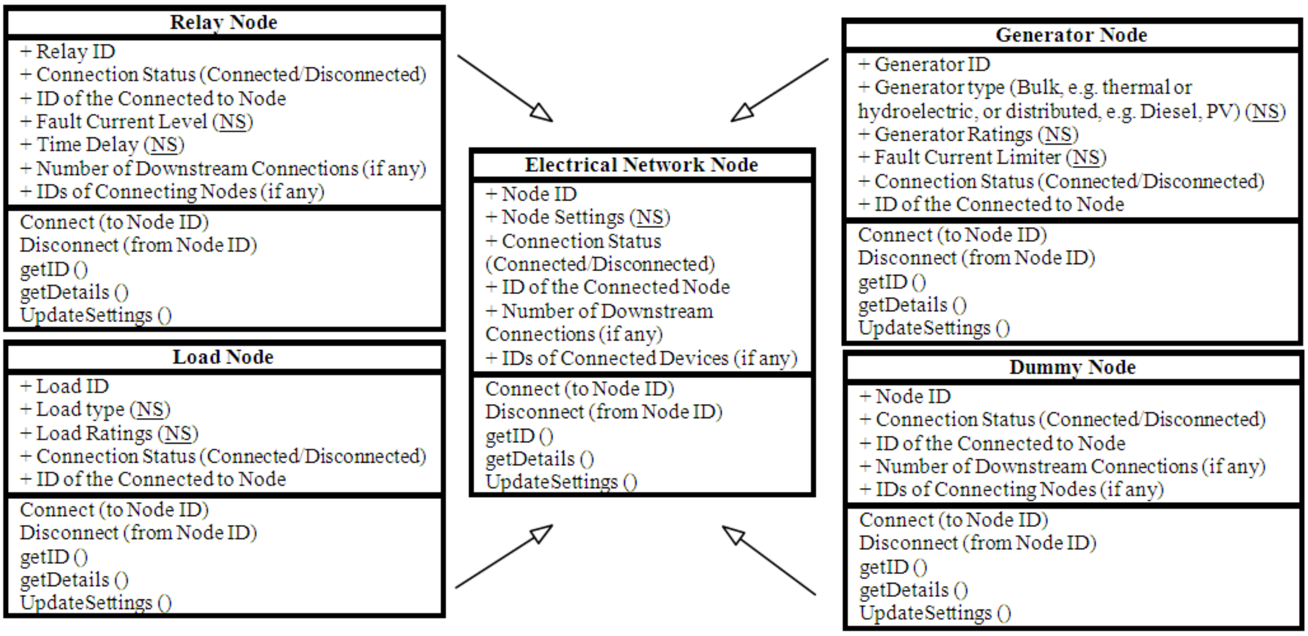

Figure. 3. Electrical Network Node and 4 specific instances of the model

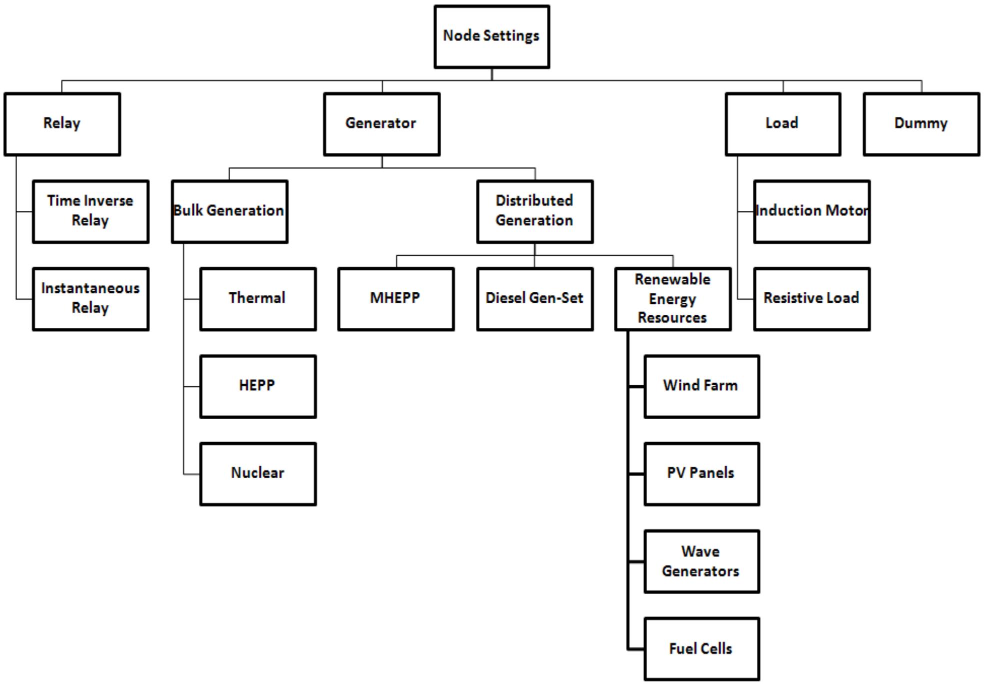

Figure 4. Abstraction of the Node Settings for different Node types handled by Update Settings Function

By following the IEC 61850 syntax, it would be possible to further develop the object class model as a Logical Device (LD). This paper is to focus on the discussion of the need for such a model and will not detail IEC 61850 fitting.

The common data sets for different instances of the EEN are node IDs, the connection status of that particular node, ID of the upstream node to which the node is connected to as well as the number of downstream nodes which are connected the node under consideration and their IDs.

The different specific instances of the ENN will have different node settings (NS) depending on the type of the node and the relevant characteristics. As shown in Figure 2, the general object class ENN has four different sub-classes which are:

i) Relay Node
ii) Load Node
iii) Generator Node
iv) Dummy Node

These sub-classes could be modeled as Logical Nodes (LN) as per the IEC 61850 standard and various data models already exist in the standard to allow for this.

These sub-classes are proposed in the most comprehensive manner so that the modeling shall be versatile and it shall be possible to model different network systems. Despite the fact that different node sub-classes have same data entry 'node settings', depending on the node type this abstraction has different sub-groups for detailed modeling. The different sub-groups of this abstraction are shown in Figure 3.

The relay element can be modeled by using the LN RDIR from the standard set of documents, but however further advances are surely necessary. For instance, relay node should have at least two attributes which represent the operation settings of the relay. The first sub-group of attributes represents the details of a time-inverse relay while the second sub-group of attributes is used to model instantaneous relays In similar fashion the generators are categorized under two main headings such as bulk generation and distributed generation. The former is required if the microgrid is connected to a larger generation system while the latter is a vital element for distributed generators such as diesel gen-sets, micro hydroelectric power plants (MHEPP) and other renewable energy resources.

Figure 3 and 4/

The modeling of loads is kept very simple and only two different sub-groups have been proposed which differentiate between the rotating machine loads and resistive loads which are hard-to-control and lightweight loads, respectively.
The detailed characteristics listed in node settings shall be acquired from the international standard IEC 61850. IEC 61850 is bound to have a significant impact on how electric power systems are to be designed and built for many years to come [11].

The model-driven approach of the IEC 61850 standard describes the communication between devices in a substation and the related system requirements. Throughout the preparation of IEC 61850 standard International Electro-technical Commission identified the several aspects of devices which are crucial for proper operation. [12]
The ENN data model shown in Figure 2 has five different services which are needed to:

a) Get connected to another node,
b) Get disconnected from an already-connected node,
c) Receive the ID of a particular node for identification purposes,
d) Acquire the settings of a particular node for management purposes,
e) Update the current settings of the node with the new operation points stipulated by the central management unit.

Among these nodes, the dummy node might be of particular interest. It, in fact, does not represent a specific device but a common coupling point where different connections meet. For example, the network shown in Figure 1 required a dummy node to connect Circuit Breaker 2 (CB2) to CB3 and CB4. Even if microgrid gets islanded, i.e. CB2 opens, CB3 and CB4 will remain connected over the dummy node. At any given instance, the new connection or disconnection of a device shall be represented by these OO models with abstracted node setting groups.

Consider the case shown in Figure 4 where a relay has a relay, a generator and a load located downstream. When each one of these downstream devices requires connecting to *Relay X* they will send a connection signal with *Connect (Relay X)* service. The variable holding the number of connections in *Relay X* and the array which holds the IDs of connected nodes will be updated. If the details of *Relay X* are retrieved with ***RelayX.getDetails()*** command, in addition to relay characteristics the returned data will include,:

Data Attribute	***Value***
Number of connections	3
IDs of connected Devices	{DG, Relay Y, Load}

When the same service is called for the downstream nodes, for instance *DG* as in ***DG.getDetails(),*** the retrieved data shall include two variables in addition to DG characteristic data. One of them is a Boolean operator, 'Connection Status', which is set to TRUE in this instance signifying that the DB is currently connected. The other attribute '*ID of the Connected to Node'* is a pointer pointing towards the upstream node to which DG is connected.

When a connected node requires to disconnecting, for instance Load node, it shall use the service ***Load.Disconnect (Relay X)***. The connection variables in Load will be changed as:

Data Attribute	***Value***
Connected	*False*
ID of the Connected to Node	*N/A*

While the related variables in *Relay X* will be updated as follows:

Data Attribute	***Value***
number of connections	*2*
IDs of connected Devices	*{DG, Relay Y}*

Following this modeling procedure the changes occurring in the microgrid can be monitored instantaneously and the relevant power management, protection or other adjustments can be performed immediately.

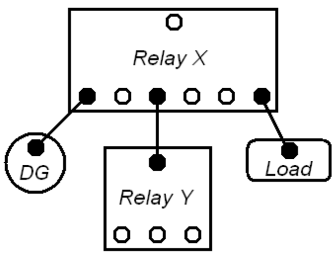

Figure 4. Netwok after Connect (Relay X) Service

4. IMPLEMENTATING DIJKSTRA'S ALGORITHM FOR MICROGRID HIERARCHY DETERMINATION

It is proposed in [13] to model the microgrid system according to graph theory and implement Dijkstra's algorithm in order to extract the relay hierarchy. Since this method does not require the knowledge of the network structure beforehand, it is very robust; it easily accepts new deployments and serves well for plug-and-play purposes.

In order to be able to implement Dijkstra's algorithm, the microgrid should be represented as a graph similar to the one shown in Figure 5. The components should be represented as nodes, or vertices, while the connections should be represented as edges. This requires storage of network data in an array or a linked list. Also the connections between the DGs, CBs and Loads should also be stored in a matrix or linked list structure. For real time response of the proposed technique, the real time data should be updated when a node disconnects from the system or an edge disappears and an alternative edge is connected. All these necessitate continuous monitoring of the microgrid and utilization of communication lines between the nodes. This should not be considered as a drawback, since such a system is already needed for smartgrids . Furthermore, most of new generation microgrid protection systems incorporate a central protection unit and communication lines as in [6, 7] .

In this paper, selectivity application shall be studied as a test case. It must be noted that the proposed method can also be used for power flow, load sharing and/or generation planning purposes. For the proper application of selectivity, the main goal is to determine the relay hierarchy. It is evident that, there is only one path between the point of origin, CB2, and the destinations, all leaf nodes such as DG1, DG2, Load1, and Load2. This eliminates the effect of distance and simplifies the existing problem to a path finding problem. In other words, Dijkstra's algorithm will be used to find the paths between CB2 and leaf nodes and identify the relay hierarchy.

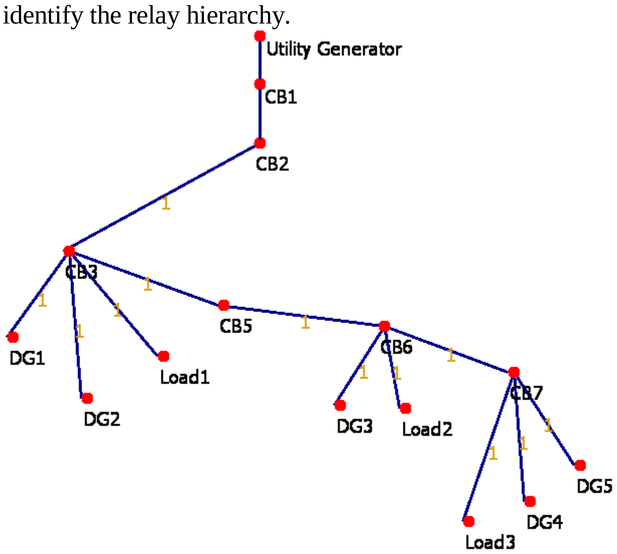

Figure 5: Modeling Case 2 with graph theory

In these graphs, all the distances are marked as one, since the shortness of the path is of no concern. Nevertheless, there is another reason for using unit distances as it has a usage for selectivity purposes. The distance yields number of the selective levels, n, between the origin and the destination. If the leaf nodes are not assigned any time delay (meaning that they will react immediately in case of a fault) the base time delay is calculated based on the parameter n. Alternatively, if it is desirable to assign time delays for the relays of the leaf nodes then the calculations will be based on value "n+1".

For the implementation of Dijkstra's algorithm on these graph representations, a C# implementation provided in [14] is used. Firstly, the algorithm is run to find the shortest path (i.e. the only path in our case) between CB2 and DG4 for Case 1. Figure 6 shows that the path is successfully highlighted on the graph and the proper hierarchy is shown in `Report` area. In order to change from Case 1 to Case 2 following services are executed to perform required connections/disconnections:

Relay4.Disconnect(Relay2)
Relay6.Disconnect(Relay4)
Relay7.Disconnect(Relay4)
Relay5.Connect(Relay3)
Relay6.Connect(Relay5)
Relay7.Connect(Relay6)

The algorithm is executed again to find the path between CB2 and DG4. The path is successfully found without a centralized monitoring for grid structure. The shortest paths and the distances obtained for both of the cases are given in Table I.

TABLE I. THE PATH FROM "CIRCUIT BREAKER 2"

	Case 1		Case 2	
Node	Dist	Path	Dist	Path
CB3	1	CB2-CB3	1	CB2-CB3
CB4	1	CB2-CB4	-	-
***DG1**	2	CB2-CB3-DG1	2	CB2-CB3-DG1
***DG2**	2	CB2-CB3-DG2	2	CB2-CB3-DG2
***Load1**	2	CB2-CB3-Load1	2	CB2-CB3-Load1
CB5	-	-	2	CB2-CB3-CB5
CB6	2	CB2-CB4-CB6	3	CB2-CB3-CB5-CB6
CB7	2	CB2-CB4-CB7	4	CB2-CB3-CB5-CB6-CB7
***DG3**	3	CB2-CB4-CB6-DG3	4	CB2-CB3-CB5-CB6-DG3
***Load2**	3	CB2-CB4-CB6-Load2	4	CB2-CB3-CB5-CB6-Load2
***DG4**	3	CB2-CB4-CB7-DG4	5	CB2-CB3-CB5-CB6-CB7-DG4
***DG5**	3	CB2-CB4-CB7-DG5	5	CB2-CB3-CB5-CB6-CB7-DG5
***Load3**	3	CB2-CB4-CB7-Load3	5	CB2-CB3-CB5-CB6-CB7-Load3

* denotes the leaf nodes

The extracted data, the relay hierarchy and the distances, can be used to do necessary adjustments for management and protection purposes. Whenever the structure of the microgrid changes, due to disconnections or new deployments, knowledge of the point of origin and the destinations (which are CB2 and leaf nodes, respectively) is sufficient to extract the new relay hierarchy. Leaf nodes will be DGs, loads or storage devices. When connected to the network, they may have a special heading or a label which indicates that they are leaf nodes.

In Figure 7, three new deployments, i.e. CB8, DG6 and Load 4 are added to Figure 6. The following three commands are realized for this change:

Relay8.Connect(Relay6)
Load4.Connect(Relay8)
DG6.Connect(Relay8)

Dijkstra's algorithm is run on the graph and the new deployments are successfully identified in grid hierarchy. It is shown that with this simple arrangement, the path from the known origin to known destinations can be found for any possible network structure. Furthermore, if there is a new deployment of branches, relays or leaf nodes, they will be automatically considered in path calculation process provided that vertex and edge data are updated accordingly.

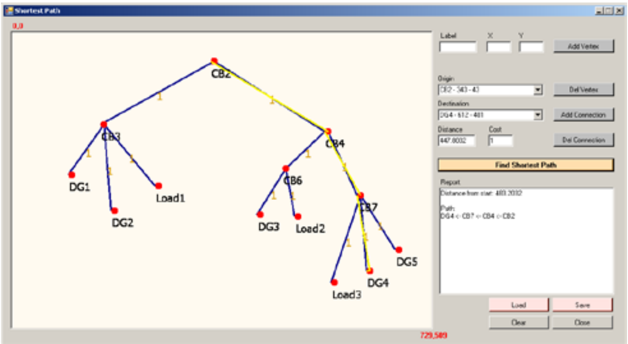

Figure 6. Dijkstra's Algorithm run for case 1, Path from CB2 to DG4

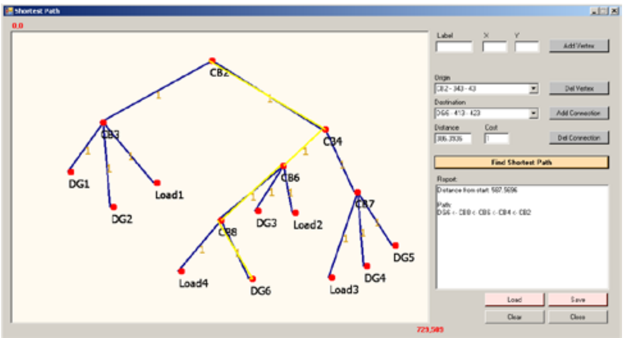

Figure 7. Dijkstra's Algorithm run after new deployments, Path from CB2 to DG6

5. CONCLUSIONS

OO based models are proposed for microgrid modeling. The proposed models make it possible to define information data specific to various electrical nodes within a network in terms of connections between the nodes and the devices connected to these nodes. In this manner the changing structure of a particular network can be followed and the new operating points can be calculated, then updated.

After modeling the microgrid according to graph theory, Dijkstra's algorithm is implemented to find the path from the point of common coupling to different parts of the network. This algorithm extracts the hierarchy of different components in the network. This feature is very crucial for plug and play purposes in electrical networks.

6. REFERENCES

[1] T. S. Ustun, et al., "Recent developments in microgrids and example cases around the world--A review," Renewable and Sustainable Energy Reviews, vol. 15, pp. 4030-4041, 2011.
[2] B. Lasseter, "Microgrids," in IEEE 2001 WM Panel, Role of Distributed Generation in Reinforcing the Critical Electric Power Infrastructure, 2001.
[3] T. S. Ustun, et al., "Investigation of Microgrid Behavior While Operating Under Various Network Conditions," presented at the IEEE International Conference on Smart Grid Engineering (SGE'12), UOIT, Oshawa, Canada, 2012.
[4] H. H. El-Tamaly and A. H. M. El-sayed, "A new technique for setting calculation of digital distance relays," in Power Systems Conference, 2006. MEPCON 2006. Eleventh International Middle East, 2006, pp. 135-139.
[5] A. H. Osman, et al., "Transmission line distance relaying using on-line trained neural networks," Power Delivery, IEEE Transactions on, vol. 20, pp. 1257-1264, 2005.
[6] A. Oudalov and A. Fidigatti, "Adaptive Network Protection in Microgrids," More Microgrids Europe www.microgrids.eu/documents/519.pdf.
[7] T. S. Ustun, et al., "A microgrid protection system with central protection unit and extensive communication," in Environment and Electrical Engineering (EEEIC), 2011 10th International Conference on, 2011, pp. 1-4.
[8] T. S. Ustun, et al., "Fault Current Coefficient and Time Delay Assignment for Microgrid Protection System With Central Protection Unit," Power Systems, IEEE Transactions on, vol. PP, pp. 1-1, 2012.
[9] T. S. Ustun, et al., "A central microgrid protection system for networks with fault current limiters," in Environment and Electrical Engineering (EEEIC), 2011 10th International Conference on, 2011, pp. 1-4.
[10] C. R. Ozansoy, et al., "Object Modeling of Data and DataSets in the International Standard IEC 61850," Power Delivery, IEEE Transactions on, vol. 24, pp. 1140-1147, 2009.
[11] C. R. Ozansoy, et al., "The Application-View Model of the International Standard IEC 61850," Power Delivery, IEEE Transactions on, vol. 24, pp. 1132-1139, 2009.
[12] I. T. WG17, "Introduction to IEC 61850-7-420: Distributed Energy Resources (DER) Object Modeling," White Paper, vol. Ver 2., July 31, 2009.
[13] T. S. Ustun, et al., "Implementation of Dijkstra's Algorithm in a Dynamic Microgrid for Relay Hierarchy Detection," in Second IEEE International Conference on Smart Grid Communications (SmartGridComm), Belgium, 2011.
[14] P. Faria. (2009). Shortest path with Dijkstra and C#. Available: http://letmetutoryou.wordpress.com/2009/05/26/shortest-path-with-dijkstra-and-c/

7. Author(s)

Principal Author: **Taha Selim USTUN** received B.E. degree in Electrical and Electronics Engineering from Middle East Technical University, Turkey in 2007 and Master of Engineering Science degree from University of Malaya, Malaysia in 2009. He has received Ph.D in Electrical Engineering from Victoria University, Melbourne, Australia. Currently, he is an Assistant Professor in Electrical Engineering, School of Electrical and Comppputer Engineering, Carnagie-Mellon University, PA, USA. His research Interests are Power Systems Protection, Communication in Power Networks, Distributed Generation, Microgrids and Smartgrids.

Presenter: The paper is presented by Taha Selim Ustun.