Optimal Wind Farm Collector System Topology Design Considering Total Trenching Length

Sudipta Dutta, *Student Member, IEEE*, and Thomas J. Overbye, *Fellow, IEEE*

***Abstract*—This paper addresses the optimal cable layout design of a collector system in a large-scale wind farm. The objective is the minimization of total trenching length which is the sum of lengths of all branches of the collector system tree. A graph-theoretic minimum spanning tree algorithm has been used as a starting algorithm, and improvements and modifications have been proposed to cater to the constraints and characteristics of a wind farm collector system. The contribution of this paper is three-fold. First, an algorithm has been proposed to further improve the results of the minimum spanning tree algorithm by creating external splice locations separate from the wind turbine locations in computing the cable layout configuration. Second, an algorithm has been proposed to compute the minimum trenching length layout configuration under the constraint of a prespecified maximum number of turbines connected to a feeder cable. Third, an algorithm has been developed to automatically compute the direction and magnitude of power flow on the different cables and to assign cable sizes accordingly.**

***Index Terms*—Clustering, collector system, economic analysis of wind farm operation, graph theoretic algorithms, minimum total trenching length, wind farm cable layout design.**

I. Introduction

THE goal of achieving 20% wind power penetration by 2030 in the U.S. has stimulated the installation of large-scale wind farms in recent years. This work addresses a topic which is extremely important in the planning, development, and installation stage of these wind farms, namely collector system design. Collector systems consolidate the power generated by turbine units distributed over the geographical area of the wind farm to a substation from where the generated power is transmitted to the electric grid. The difficulty of a collector system design project is that given the wind turbine locations and the substation location where the aggregated power is transmitted to the grid, depending on the dimension of the wind farm, there may be thousands of feasible layout configurations to choose from. Selecting an optimal design from these choices can be a challenging task. This work proposes three algorithms which generate a layout design for a collector system optimizing the total trenching length.

Collector system layout for a wind farm depends on several factors including turbine placement, terrain, reliability, landowner requirements, economics, and expected climatic conditions for the location [1]. Placement of turbines is based on a study of the wind direction and pattern, soil conditions, aviation restrictions, land agreements, and constructability considerations. The collector system substation is installed ideally within the central area of the wind farm to minimize collector system conductor lengths and optimize loss efficiencies within the collector system. However, the distance from the point of interconnection (POI) to the electric grid is also an important consideration in determining the location for the substation. The turbine locations and the substation will be the primary factors in the design and layout of the collector system considered here.

Collector systems can be structured in different configurations such as loop system and radial system depending on the desired level of collector system reliability. Since the majority of large wind farms built in the U.S. have a radial collector system, the design target of this work is to develop an optimal radial configuration with a collection system voltage of 34.5 kV. In this configuration, turbines are connected together in a "daisy chain" style, moving outward from the substation to the furthest located turbine.

A medium or low voltage power distribution system design has similarities with the wind farm collector system optimization problem. However, there are distinct differences between them. Collector systems operate at medium voltages and the substation steps the voltage up to a transmission level voltage. On the contrary, the input to a distribution substation is from a high voltage transmission line and the output is a number of medium voltage feeders. One more difference is that collector system cables are mostly underground whereas distribution system lines are mostly overhead, along streets. Overhead lines are more exposed to environment and inclement weather conditions and have a greater failure rate compared to underground cables. To increase reliability of distribution systems, looped or meshed configurations are common. However, wind farm collector system cables being mostly underground have a lower failure rate; so radial configurations are common. The area covered by a distribution system is also typically much larger compared to a collector system. The differences between distribution systems and wind farm collector systems indicate that an algorithm that generates the best layout for a distribution system might not generate the best one for a collector system. Hence, we argue that further research is required to develop dedicated algorithms for wind farm collector system layout design.

Studies have shown that an electric utility's power distribution system can account for up to 60% of capital budget and 20% of operating costs making it a significant expense [2]. Minimizing the cost of the distribution system can be a considerable challenge, as there are thousands of feasible design options to choose from. For these reasons, much research [2]–[5] has been

Manuscript received May 18, 2011; revised September 14, 2011; accepted January 12, 2012. Date of publication April 06, 2012; date of current version June 15, 2012. This work was supported by the Stanford University Global Climate and Energy Project (GCEP) and by the Power Systems Engineering Research Center (PSERC).

The authors are with the Electrical and Computer Engineering Department, University of Illinois at Urbana-Champaign, IL 61801 USA (e-mail: dutta4@illinois.edu; overbye@illinois.edu).

Digital Object Identifier 10.1109/TSTE.2012.2185817

focused on development of optimization algorithms to identify the lowest cost distribution configuration and hence the best design. Even with approximations, such programs can help reduce distribution costs by 5% to 10%. The similarities in the problems of distribution system optimization and wind farm collector system optimization indicates that there is scope of improvements on a similar scale by developing better designs for wind farm collector systems as well, thereby motivating the current work.

The IEEE PES Wind Plant Collector System Design Working Group has addressed issues related to the design of collector systems for wind plants. In [1], important design considerations such as feeder topology, collector design, interconnect, and NESC/NEC requirements for wind farms have been summarized. Protection issues of wind farms have been considered in [6], and in [7], design guidelines based on redundancy, reliability, and economics have been presented. A summary of design considerations including conductor selection, soil thermal properties, installation methods, and splicing, was presented in [8].

In a previous work [9], we addressed the cable layout design problem and proposed a clustering-based layout design that resulted in lower power losses and higher reliability compared to conventional radial cable layout configuration. The contribution of the current paper and also what differentiates this work from the previously mentioned references is that, here, an automatic collector system layout design for a wind farm is generated, that meets given design constraints such as a limit on number of turbines on a feeder, and the flexibility of introducing splices thereby further lowering the total length and hence cable costs and trenching costs.

This work treats the collector system design problem with a graph-theoretic approach building upon classical graph-theoretic algorithms, modifying them to cater to different constraints, and utilizing special features of the collector system design problem. Three novel algorithms have been proposed in this paper, the first further improving the results of a minimum spanning tree algorithm by creating external splice locations in computing the cable layout configuration, and the second computing the minimum length layout configuration under the constraint of a prespecified maximum number of turbines connected to a feeder cable. This second algorithm addresses the condition that the number of wind turbines that can be placed on a feeder string is limited by the conductor ampacity. The third algorithm automatically computes the direction and magnitude of active power flow on the different cables and assigns cable sizes accordingly.

Section II describes the mathematical considerations of the problem. Section III describes Dijkstra's algorithm and Section IV describes the process of K-means clustering. Both Dijkstra's algorithm and K-means clustering have been used in the proposed algorithms. Section V presents the proposed methodology with the subsections describing the algorithms. Section VI describes the available cable sizes and resistances. The results obtained by applying the proposed algorithms on wind turbine locations of a section of a real life large-scale wind farm are presented in Section VII. The cost of energy with the layouts generated by the proposed algorithms is also computed. Conclusions have been made in Section VIII. Finally, the method used for cost of energy computation has been presented in the Appendix.

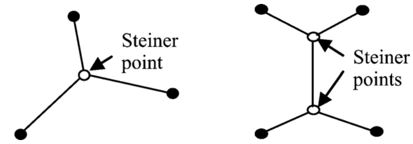


Fig. 1. Steiner points for groups of three and four points.

II. Mathematical Considerations

The problem of cable layout design for a wind farm collector system can be considered as finding a tree to meet required design characteristics in a graph $G=(V,B)$, where V represents the set of vertices or nodes which in this work are the locations of wind turbines and the substation, and B represents the set of branches or edges connecting the vertices which in this work are the connecting cables. Given the set of vertices, there can be numerous possible trees satisfying specific design constraints. More specifically, by Cayley's Tree formula, the number of nonidentical trees of order V is V^{V-2}. Therefore, in a wind farm with hundreds of turbines, performing an exhaustive search for the optimal tree to attain a desired objective becomes computationally expensive.

When the objective is to minimize the total length and hence to find a tree in the graph with minimum total length of edges, the problem is relatively easy and requires a minimum spanning tree algorithm to generate the required configuration. However, when an additional degree of freedom is introduced in the design space by allowing the addition of extra intermediate vertices to the graph in order to reduce the length of the spanning tree with the increased number of vertices, then the problem of optimal cable layout design to minimize the total length becomes a Steiner tree problem [10].

The minimum spanning tree problem which interconnects a given set of points or vertices by a network or graph of shortest total length of edges is superficially similar to the Steiner tree problem. In fact, a minimum spanning tree is a feasible but not usually optimal solution to the Steiner tree problem. However, the difference between the Steiner tree problem and the minimum spanning tree problem is that, in the Steiner tree problem, extra intermediate vertices and edges may be added to the graph in order to reduce the length of the spanning tree. These new vertices introduced to decrease the total length of connection are known as Steiner points or Steiner vertices.

For the Euclidean Steiner problem, Steiner points must have a degree of three, and the three edges incident to such a point must form three 120° angles. It follows that the maximum number of Steiner points that a Steiner tree can have is $N-2$, where N is the initial number of given points. For $N=3$, the solution is given by a Steiner point located at the Fermat point of the triangle formed by the given points. Fig. 1 shows the Steiner points for groups of three and four points, respectively.

For general N, and hence in the present case of large-scale wind farm collector system optimization, the Euclidean Steiner tree problem is NP-hard, and hence it is not known whether an optimal solution can be found by using a polynomial-time algorithm. The goal is then to connect the vertices (turbine locations) by edges (cables) of minimum total length in such a way that any two points may be interconnected by line segments either directly or via intermediate points similar to but not the same as Steiner points (intermediate splices) and line segments. The algorithm presented in Section V-A in this work achieves this goal in polynomial time.

For the Euclidean Steiner tree problem, the Steiner ratio [11], which is the largest possible ratio between the total length of a minimum spanning tree and the total length of a minimum Steiner tree, is $2/\sqrt{3} \approx 1.15$. An improvement of this order is difficult to achieve given that the problem is NP-hard. However, results show that the algorithm presented in Section V-A improved the solution obtained by the minimum spanning tree algorithm by a factor of approximately 1.01 in polynomial time.

III. Dijkstra's Minimum Spanning Tree Algorithm

A minimum spanning tree algorithm constructs a tree of minimum total length between specified nodes, where a tree is defined as a graph with one and only one path between every two nodes. This algorithm can be used to address the problem of finding the minimum total length layout design for a given wind farm, where the minimum total length gives the minimum total trenching length and the wind turbine locations are the nodes. The Dijkstra's minimum spanning tree algorithm [12] defines three sets of branches:

Set I) The branches definitely assigned to the tree under construction (they will form a subtree).

Set II) The branches from which the next branch to be added to set I will be selected.

Set III) The remaining branches (rejected or not yet considered).

The nodes are subdivided into two sets:

Set A) The nodes connected by the branches of Set I.

Set B) The remaining nodes (one and only one branch of Set II will lead to each of these nodes).

The algorithm starts by choosing an arbitrary node as the only member of Set A, and by placing all branches that end in this node in Set II. To start with, Set I is empty. From then onwards the following two steps are performed repeatedly.

Step 1) The shortest branch of Set II is removed from this set and added to Set I. As a result one node is transferred from Set B to Set A.

Step 2) The branches leading from the node which has just been transferred to Set A to the nodes that are still in Set B are considered. If the branch under consideration is longer than the corresponding branch in Set II, it is rejected; if it is shorter, it replaces the corresponding branch in Set II, and the latter is rejected. This is followed by a return to Step I and a repetition of the process until Sets II and III are empty. The branches in Set I form the tree required.

IV. K-Means Clustering

K-means clustering is one of the simplest unsupervised learning algorithms for classifying a given data set into a certain K number of clusters fixed *a priori* [13]. The main idea is to define K centroids, one for each cluster. The starting centroids must be chosen carefully since different starting locations of centroids may cause the algorithm to converge to different clusters. An idea is to select centroids from the given data set in a way that they are as much as possible far away from each other. The next step is to take each point belonging to the given data set and associate it to the nearest centroid forming clusters. When all the points have been grouped, the first step is completed. At this point K, new centroids are recalculated as centers of the clusters resulting from the previous step. After this, once again the same data set points are grouped so that each one is now associated with its nearest new centroid. This process continues until there is no change in the location of the centroids. This algorithm is as follows [14]:

Step 1) K points are placed into the space represented by the objects that are being clustered. These points represent initial group centroids.

Step 2) Each object is assigned to the group with the closest centroid.

Step 3) When all objects have been assigned, the positions of the K centroids are recalculated.

Step 4) Steps 2 and 3 are repeated until the centroids no longer move. This produces a separation of the objects into clusters.

V. Methodology

In this paper, Dijkstra's minimum spanning tree algorithm has been used to generate the basic least total length design of the wind farm. In addition, an algorithm has been developed to introduce intermediate splice points similar to Steiner's vertices to reduce the total length obtained by the minimum spanning tree algorithm. In an electrical collector system, there is a limit on the number of turbines on a feeder cable depending on the maximum size and ampacity of available cables. To address this problem in the design, the wind turbine locations have been clustered using K-means clustering and the number of turbines in each cluster is restricted to the maximum limit by an algorithm. All the above-mentioned algorithms result in undirected graphs. However, for the wind farm collector system design, an important concern is determining the cable sizes which depend on the ampacity of required cables which in turn depend on the actual power carried by the cable. This needs a direction of power flow on the cables and hence the undirected graphs have to be converted to directed graphs. Another algorithm has been proposed for the same.

For the purpose of this work, it has been assumed that turbine locations in the wind farm are already available and that the wind farm has a single collector substation at the POI to the grid. Therefore, the input to the algorithms includes the locations of the wind turbines and the substation. Also, in this work issues of laying cables such as land unavailability within the wind farm area or trenching restrictions have not been considered and will be addressed in a future work.

A. Introduction of Intermediate Splices

The algorithm for introduction of intermediate splices improves the results obtained by Dijkstra's minimum spanning tree algorithm. The algorithm has the following steps:

Step A.1) The algorithm is initialized by the connected graph obtained by applying Dijkstra's minimum spanning tree algorithm on all nodes including the substation. Four sets are defined as follows: Set 1= {the substation node}, Set 2 = {all other nodes}, and Set 3, Set 4 null sets.

Step A.2) A node is selected from Set 2 such that it is connected to a node in Set 1 according to the minimum spanning tree algorithm. The former is named Node A and the latter Node B. The distances between Node A and the nodes in Set 1 are computed. If the least distant node is Node B, the existing branch in between Nodes A and B is retained and moved to Set 3. Also, the five nodes created in Step A.3 of the previous iteration (provided this is not the first iteration) are deleted from Set 1. However, if the least distant node in Set 1 is not Node B, it (the least distant node) is defined as a "splice node." The splice node location is copied to Set 4. A branch is created between Node A and the splice node which replaces the existing branch between Node A and Node B in Set 1 and is moved to Set 3. The four remaining nodes created in the previous iteration are deleted from Set 1.

Step A.3) Multiple equidistant points are created on the just moved branch to introduce new nodes. In this work, five equidistant points are chosen. Node A and the five newly created nodes are moved to Set 1.

Steps A.2 and A.3 are repeated until Set 2 is empty. This indicates that all the nodes have been considered. Finally, Set 4 contains the set of newly created intermediate splice nodes, and Set 3 contains the branches that form a tree with the original set of nodes and introduced splice nodes.

The process in Steps A.1 to A.3 has been illustrated in Fig. 2. Fig. 2(a) shows the initialization of a five-node graph (four turbine nodes indicated by dots and one substation node indicated by a square) by Dijkstra's minimum spanning tree algorithm. At this stage, Set 1 consists of the substation node, i.e., Node 1, and Set 2 consists of the other four nodes, i.e., Nodes 2, 3, 4, and 5. At the first stage, Node B = Node 1, and Node A = Node 2, since it is the only node connected to a node in Set 1. The least distance between Node A and a node in Set 1 corresponds to Node B, since Set 1 contains only Node 1 at this stage. Hence the branch between these is retained as shown in Fig. 2(b). This branch is replaced by five equidistant nodes as shown in Fig. 2(c) and these are moved to Set 1. Set 1 now contains the five newly created nodes, Node 1, and Node 2. Node 2 is removed from Set 2. Set 3 contains the branch between Node 1 and Node 2 and Set 4 is empty. At the next stage, Node 3 is selected and distances between this node and the seven nodes in Set 1 are computed. It is found that Node 2 is the least distant from Node 3, so the branch between these is retained and no splice nodes introduced as shown in Fig. 2(d). This process is followed until the first splice node (indicated by a diamond) is found in Fig. 2(h), and Set 2 becomes empty.

An important point to note in this algorithm is that Set 1 is initialized with the substation node. The characteristics of spanning trees having only one unique path between any pair of nodes ensures that if Set 2 is nonempty, a node can always be found in Set 2 that is connected to a node in Set 1. The modified tree with increased number of nodes and branches grows outward from the substation node with every iteration, adding intermediate splice nodes as necessary to reduce the total spanning length.

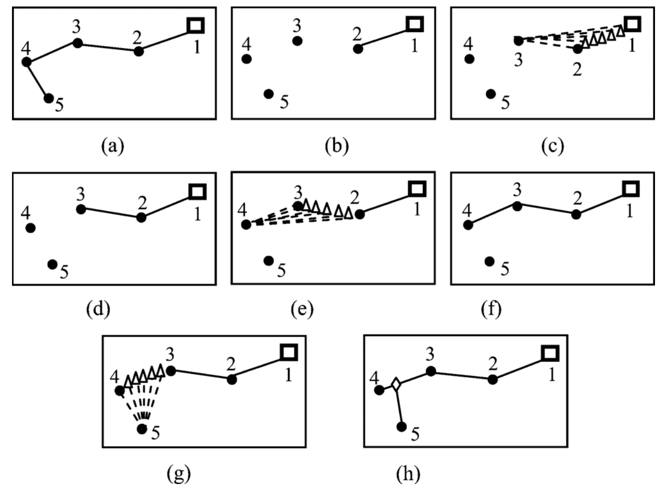


Fig. 2. Addition of intermediate splice nodes.

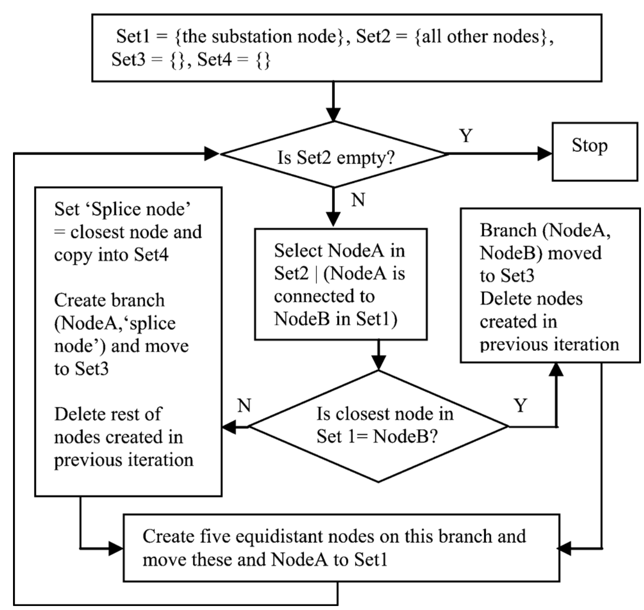


Fig. 3. Algorithm for computing intermediate splice locations.

This algorithm has been further illustrated with a flowchart in Fig. 3.

B. Limit on Maximum Number of Turbines on A Feeder

Feeder cables have limited current carrying capacity. This limits the number of turbines that can be connected to a feeder. Assuming a limit of Nmax turbines on a feeder, the algorithm for incorporating this constraint in the automatic cable layout design is as follows:

Step B.1) The algorithm is initialized with a value of number of clusters required. This value is calculated by finding the ratio of total number of turbines and Nmax.

Step B.2) The turbines are clustered with K-means clustering algorithm and the size of the largest cluster is found.

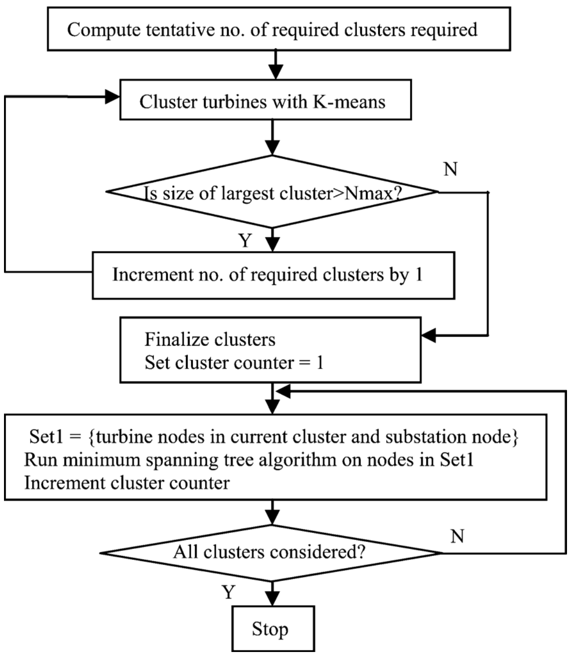


Fig. 4. Algorithm for limiting number of turbines on a feeder.

Step B.3) If the size of the largest cluster exceeds the prespecified Nmax, the value of number of required clusters is increased by unity and Steps B.2 and B.3 are redone. If the size of the largest cluster is within limit, the value for required cluster number is finalized and the turbine nodes are grouped according to K-means clustering algorithm.

Step B.4) For every cluster, the nodes corresponding to the turbines and the substation location are grouped in a set called Set A. Next the minimum spanning tree algorithm is applied on elements of Set A.

This algorithm has been further illustrated with a flowchart in Fig. 4.

Depending on a visual inspection of the geographical distribution of wind turbines, Step B.2 can be varied to cluster the actual wind turbine locations or the radial angles of turbine locations at substation location. The differences between these two clustering methods will be explained in more detail in the section describing results.

To ensure the convergence of the K-means clustering algorithm to the same clusters in each run, the starting centroids are selected by the algorithm in a way that the K centroids are as far away from each other as possible. For example, when the turbines are clustered by radial angles, the turbines are first sorted by increasing or decreasing angle values and then K equispaced locations are selected from this list to give the K starting centroids.

C. *Conversion of Undirected to Directed Graph*

In this work, assignment of cable sizes is based on maximum real power loading of cables and maximum real power generated by the wind turbines.

The algorithms described in the previous sections result in undirected trees. The algorithm described in this subsection converts the undirected trees to directed ones. The input to the algorithm is the connected undirected graph obtained by applying one of the previously described algorithms on substation and wind turbine locations. The output is the direction and magnitude of active power flow on the branches or cables, an assignment of cable sizes for different cables from a list of available cable sizes as shown in Table I, and power losses on all connecting cables. The design is for the situation when all the turbines generate the rated power. Thereby the power injected by each turbine at the node corresponding to its location equals the rated power. The algorithm is as follows:

Step C.1) Two sets are initialized. Set 1 is the set of all turbine and splice nodes. Set 2 is the set of all branches.

Step C.2) A node is selected from Set 1 such that it corresponds to only one branch in Set 2. This node is named Node A. The purpose of this step is to select one of the terminal nodes in the graph. Node A is denoted a "from node." The node to which it (Node A) is connected is denoted a "to node." The power flow direction is from the "from node" to the "to node." The flow on the cable is the power injected at the "from node" and the power injected at the "to node" location is incremented by the power flow on the cable. The connecting cable is assigned a cable size from continuous ampacity rating of available conductors in Table I and assuming a 34.5-kV medium voltage system. The resistance of the cable is computed from resistance data in Table I. The resistance is used to find the I^2R power losses on the cable.

Step C.3) Node A ("from node") is deleted from Set 1 and the branch connecting the "to node" and "from node" is deleted from Set 2 thus resulting in a smaller dimension graph.

Step C.4) Steps C .2 and C .3 are repeated until Set 1 is empty.

This algorithm has been further illustrated with a flowchart in Fig. 5.

TABLE I
ACSR CABLE SIZES, PROPERTIES, AND COSTS

Al Strand Conductor Size	Continuous Ampacity (Amps) [15]	DC Resistance at 25 deg C (mΩ/m) [16]	AC Resistance at 25 deg C (mΩ/m)	Cost ($/m)
1/0	150	0.5482	0.5482	28
4/0	211	0.2741	0.2741	35
500 kcmil	332	0.1161	0.1184	42
750 kcmil	405	0.0774	0.0813	85
1000 kcmil	462	0.0577	0.0633	125

VI. AVAILABLE CONDUCTOR SIZES AND PROPERTIES

Some of the common cable sizes used in large-scale wind farms can be found in [8]. In this work, it is assumed that aluminum conductor steel reinforced (ACSR) cables of only the sizes mentioned in Table I are available for the collector system cable layout. Multiple (double and triple) circuits of cables were used to provide sufficient ampacity levels to the collector system as generated power from turbine units gets consolidated gradually closer to the substation.

The cable ampacity is based on conditions [15] that cables are installed in sand with minimum cover of approximately 1 m, load factor is 100%, and maximum ambient earth temperature

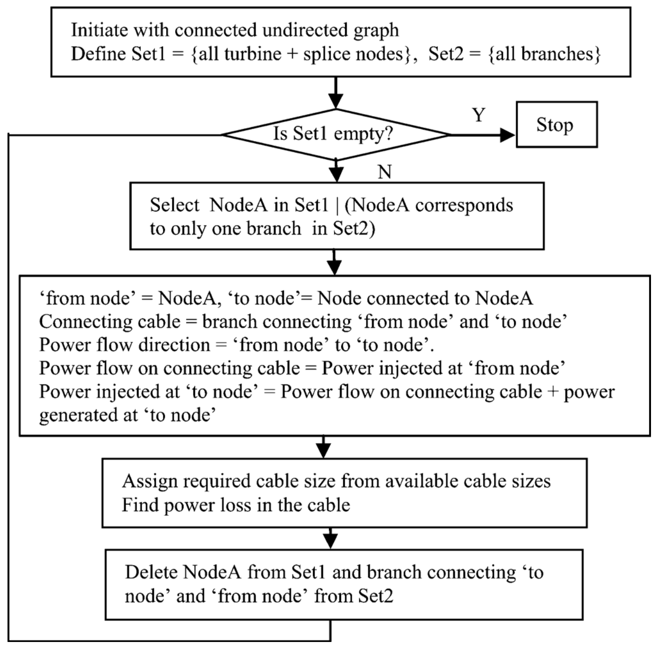


Fig. 5. Algorithm for conversion of undirected to directed graph.

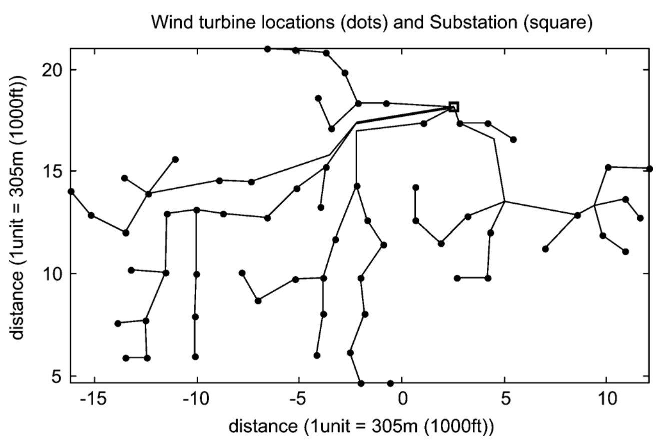


Fig. 6. Actual cable connection diagram. Total trenching length = 43 186 m. Total cable length = 43 186 m.

is 20 °C. The dc resistances of the different conductors were obtained from [15] and [16]. The ac resistances are computed from the dc resistances according to the method described in [17] and are tabulated in Table I. Approximate cable costs are also provided.

VII. Results

The proposed algorithms in this work have been tested on an example of a wind farm consisting of 66 turbines on a flat terrain such as is common in the U.S. Midwest. Fig. 6 shows the locations of the 66 wind turbines with dots, the substation with a square, and the actual cable layout configuration.

Case I—Minimum total length configuration without limits on maximum number of turbines on a feeder and without introducing intermediate splice nodes.

In this case, the Dijkstra's minimum spanning tree algorithm is applied on all the turbine locations and the substation location

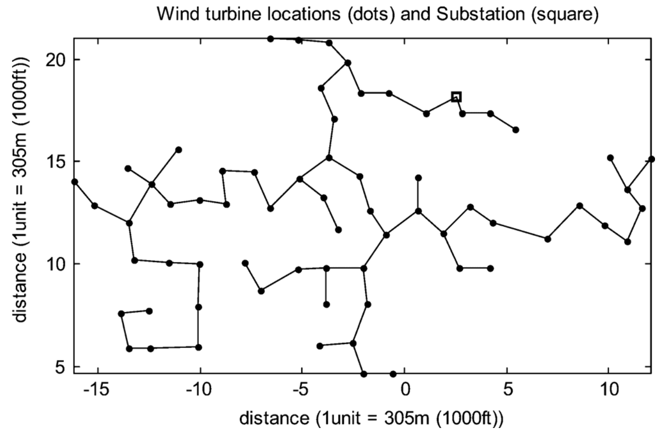


Fig. 7. Shortest total length connection (using Dijkstra's algorithm). Total trenching length = 33 896 m. Total cable length = 40 197 m.

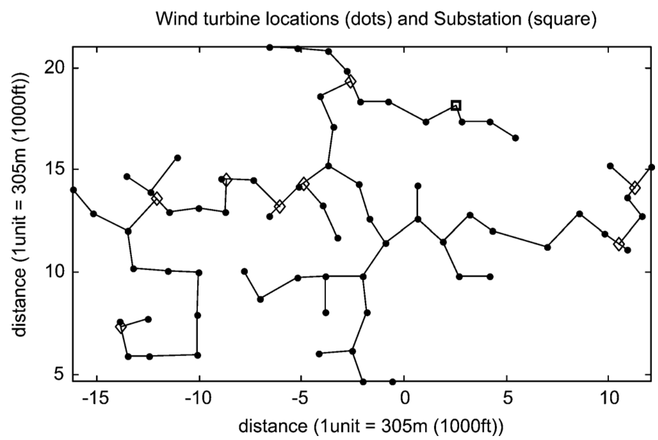


Fig. 8. Shortest total length connection with introduction of intermediate splices (diamonds). Total trenching length = 33 732 m. Total cable length = 39 672 m.

for laying out cables from wind turbines to substation so that total trenching length is minimized. The configuration is shown in Fig. 7. The algorithm described in Section V-C was also applied to convert the undirected graph (resulting from Dijkstra's algorithm) to a directed graph to determine direction of power flow on each cable, and assign cable sizes depending on available conductor sizes.

As can be seen there are two feeders coming into the substation, one carrying the power generated from 3 turbines and the other from 63 turbines. None of the cables listed in Table I can provide the required ampacity in a single circuit as loading increases closer to the substation. Thus multiple circuits of cables have to be used. This is reflected by a larger value for the total cable length compared to the total trenching length.

Case II—Reduced total length configuration compared to Case I by the introduction of intermediate splice nodes without limits on maximum number of turbines on a feeder.

In this case, the algorithm described in Section V-A is applied followed by the algorithm in Section V-C. The resulting configuration is shown in Fig. 8.

As in Case I, direction of power flow on each cable is determined, and cable sizes assigned depending on available conductor sizes. Comparing Figs. 7 and 8, it can be seen that the

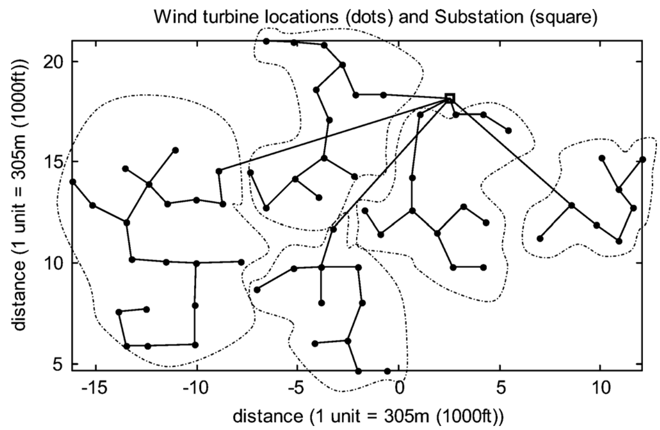


Fig. 9. Five feeders each with less than or equal to 20 turbines. Total trenching length = 41 844 m. Total cable length = 41 844 m.

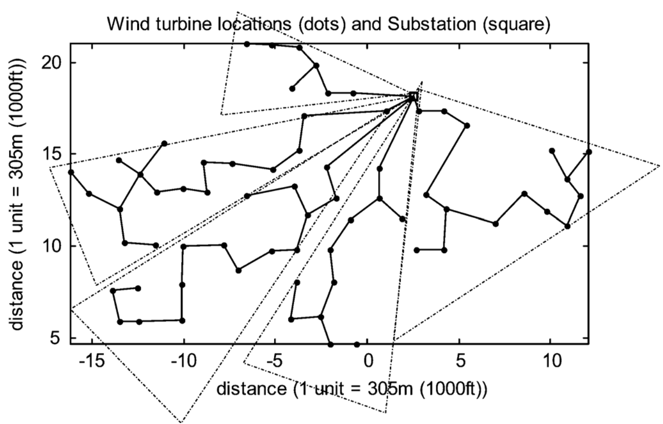


Fig. 10. Five feeders each with less than or equal to 20 turbines. Total trenching length = 39 037 m. Total cable length = 39 037 m.

total trenching length decreases by a factor of about 1.01 when the intermediate splice nodes are introduced.

However, the disadvantage of the configuration in Fig. 8 is that there are two feeders coming into the substation each connecting to 3 and 63 turbines, respectively. Similar to the previous case, the total trenching length is lower than the total cable length because multiple cable circuits have to be used to provide required ampacity closer to the substation. In a practical collector system, there may be limits on the maximum number of turbines on a feeder. This condition is addressed in Case III.

Case III—Minimum total length configuration with limits on maximum number of turbines on a feeder and without introducing intermediate splice nodes.

The algorithms described in Sections V-B and V-C are used in this case. Figs. 9 and 10 show the resulting configurations. Fig. 9 corresponds to clustering based on wind turbine locations, and Fig. 10 corresponds to clustering based on radial angles of wind turbine locations at the substation location. In each figure, the dashed–dotted lines show the clusters. The number of clusters is determined so that the maximum number of turbines in each cluster is less than or equals 20 turbines in this work. However, the maximum limit can be varied according to the design requirement. As can be seen, in Fig. 9, five clusters are formed

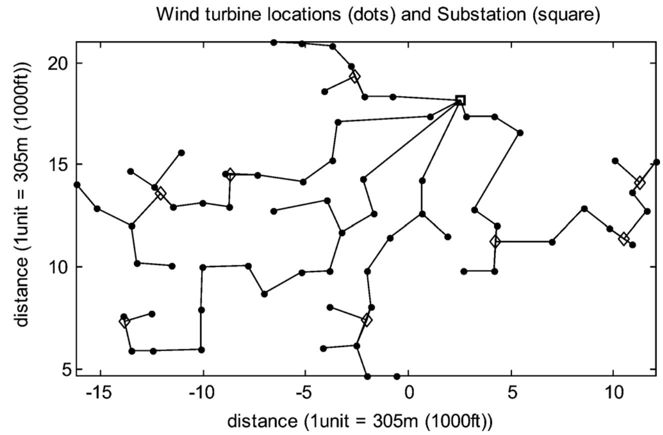


Fig. 11. Five feeders each with less than or equal to 20 turbines; intermediate splices (diamonds) introduced. Total trenching length = 38 870 m. Total cable length = 38 870 m.

with 20 turbines in the largest cluster, and in Fig. 10, five clusters are formed with 17 turbines in the largest cluster.

In this particular wind farm, the geographical distribution of wind turbine locations are such that Fig. 10 (and hence a radial clustering) depicts a better choice for clustering compared to Fig. 9. The disadvantage of Fig. 9 for this wind farm is that some cables connecting turbines of a cluster to the substation may pass through or very close to turbines in another cluster thus requiring application of heuristics to improve the design generated by the algorithm. However, clustering based on turbine locations can be useful in designing collector system configurations in a wind farm where the terrain and other factors lead to placement of groups of wind turbines at large distances from each other. We argue that the choice whether turbine locations or radial angles are to be clustered depends on the specific wind farm under study and can be done by a visual inspection of relative locations. Once the clusters are defined, Dijkstra's algorithm is run on individual clusters to get the least total cable length layout under this constraint.

In Figs. 9 and 10, the total cable length equals the total trenching length. This is because the maximum number of turbines on a feeder is limited. So one of the cable sizes from Table I could be assigned to each of the cables, even those close to the substation, without violating the ampacity limits.

Case IV—Reduced total length configuration by introduction of intermediate splice nodes and considering limits on maximum number of turbines on a feeder.

In this case, the turbine locations are first clustered based on radial angles followed by introduction of intermediate splice nodes thus interlacing the algorithms described in Sections V-B and V-A. This is followed by applying the algorithm described in Section V-C. The resulting configuration is shown in Fig. 11. Due to the introduction of intermediate splice nodes, the total length is less than in Fig. 10.

As in the previous case, the total cable length equals the total trenching length because the maximum number of turbines on a feeder is limited.

Case V—Same turbine clusters as actual layout.

It should be noted that the maximum number of turbines on a feeder for the actual layout (Fig. 6) is 24 whereas in the layout in

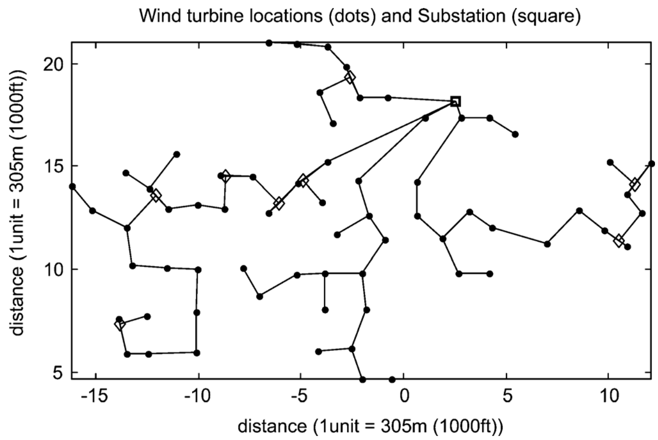


Fig. 12. Layout with turbine clusters same as actual layout; intermediate splices (diamonds) introduced. Total trenching length = 37 131 m. Total cable length = 37 131 m.

TABLE II
RESULTS SUMMARY

Layouts	Max. turbines on feeder	Total trenching length (m)	Total cable length (m)	Trenching costs (× 10^6 $)	Cable costs (× 10^6 $)
Case I (min. spanning tree)	63	33896	40197	1.69	2.13
Case II (splices)	63	33732	39672	1.68	2.06
Case III (normal clustering)	20	41844	41844	2.09	1.36
Case III (radial clustering)	17	39037	39037	1.95	1.29
Case IV (splices and radial clustering)	17	38870	38870	1.94	1.28
Case V (Same clusters as Actual layout)	24	37131	37131	1.86	1.37
Actual layout	24	43186	43186	2.16	1.44

Fig. 11, the maximum number is 17. When the developed algorithms are applied on the same turbine clusters as the actual one, and hence with the same maximum limit, the layout obtained is shown in Fig. 12.

As is Case IV, the total cable length equals the total trenching length because the maximum number of turbines on a feeder is limited.

The results for all the cases are summarized in Table II.

It should be noted that adding a constraint, namely limiting number of turbines on a feeder as in Cases III and IV results in an increase of the total trenching length compared to Cases I and II. For the same reason, the total trenching length in Case V is lower than the previous layout (Case IV). Also, it is noticed that the higher the number of turbines allowed to be connected to a feeder, greater are the cable costs since a greater number of higher sized and hence higher cost cables are required as branch currents add up close to the substation. Furthermore, radial clustering generates better results compared to clustering by turbine locations.

TABLE III
COSTS OF ELECTRIC ENERGY

Layouts	Cost of energy ($/MWh)	Layouts	Cost of energy ($/MWh)
Case I (min. spanning tree)	46.800	Case IV (splices and radial clustering)	46.123
Case II (splices)	46.733	Case V (Same clusters as Actual layout)	46.160
Case III (normal clustering)	46.195	Actual layout	46.163
Case III (radial clustering)	46.124		

The costs of energy for the different layouts are also calculated and presented in Table III. Since the costs of energy are levelized costs, there is not a significant change between different layouts. However, with the layouts in Cases IV and V, the cost of energy values are slightly smaller compared to the actual layout. The procedure for estimating the cost of energy is described in the Appendix.

It should be noted that the objective of the current paper is to minimize the trenching lengths which is achieved. Also, with the layouts in Cases IV and V, significant savings of, respectively, $380 000 and $370 000 are made compared to the actual layout in cabling and trenching. These savings are important considerations during project planning. It should also be noted that an important contribution of this work is that these layouts are generated automatically, thus saving manual labor.

VIII. CONCLUSION

This paper proposes three novel algorithms for generating a basic design for a wind farm collector system cable layout configuration. The first algorithm improves on a minimum spanning tree design by creating external splice locations separate from the wind turbine locations. The second algorithm addresses the constraint of a prespecified maximum number of turbines connected to a feeder cable. The third proposed algorithm computes direction and magnitude of power flow on each cable, assigns cable sizes from a table of available cable sizes.

Results show that the algorithms proposed in this paper can be used to generate a design that has minimum total trenching length, also taking into account constraints on maximum number of turbines on a feeder. The total length of the minimum spanning tree is lowered by a factor of 1.01 by the introduction of intermediate splice nodes, but this total length increases when the constraint of 20 maximum turbines on a feeder are applied. The designs generated in Cases IV (Fig. 11) and V (Fig. 12) achieve, respectively, 10% and 14% reduction in total trenching length compared to the actual cable layout configuration shown in Fig. 6. Future work will involve taking into account constraints such as landowner requirements and trenching restrictions. However, the major contribution of this work is in the automatic generation of a starting layout design which is optimal with respect to total length and can be modified with heuristics to incorporate additional constraints.

TABLE IV
GLOSSARY OF PARAMETERS

Symbol	Quantity	Value
P_R	Power rating/turbine	1 MW
N	No. of turbines	66
V_{avg}	Average wind speed	8.5 m/s
CF	Capacity factor	Computed by eq (1)
D	Rotor diameter	52 m
P_{loss}	Power loss in collector system	Computed as a part of Algorithm *V-C* (MW)
AE	Annual energy	Computed by eq (2) (MWh/yr)
CC	Capital cost	Computed by eq (3) ($)
C_T	Cost of turbine	$\$1\times 10^6$ /turbine
C_t	Trenching cost	$50/m
L	Total trenching length	Computed as a part of Algorithm *V-C* (m)
C_c	Total cable cost	Computed as a part of Algorithm *V-C* ($)
AC	Annual cost	Computed by eq (4) ($/yr)
A	Annual loan payments	Computed by eq (5) ($/yr)
E	Annual return on equity	15% ($/yr)
M	Operation and maintenance costs	3% of CC ($/yr)
CRF	Capital recovery factor	Computed as a part of eq (5) (/yr)
P	Principle borrowed	75% of CC ($)
i	Rate of interest	7%
n	Loan term	20 yrs
CEE	Cost of electric energy	Computed by eq (6) ($/MWh)

APPENDIX

The parameters used in the economic analysis are presented in Table IV.

In this work, underground cables interconnect wind turbines and the substation at 34.5 kV. Assuming the units generate power at unity power factor, the current is projected to be a maximum of 16.73A.

Assuming Rayleigh distribution, capacity factor of the turbines can be computed as [18]

$$\mathrm{CF} = 0.087 \times V_{\mathrm{avg}} - 1000 \times \frac{P_R}{D^2}. \tag{1}$$

It has been assumed that the wind turbines generate either the peak power or zero. Hence, the collector system loss factor equals the capacity factor. Hence, the annual energy generated is found by the product of the nameplate capacity of the wind farm reduced by the collector system losses, number of hours in a year, and the capacity factor of the wind farm

$$\mathrm{AE} = (\mathrm{PR} \times N - P_{\mathrm{loss}}) \times 8760 \times \mathrm{CF}. \tag{2}$$

The capital cost of the wind farm project is computed as

$$\mathrm{CC} = 1.05 \times (N \times C_T + L \times C_t + C_c). \tag{3}$$

A multiplication factor of 1.05 is used to take into account other components of the capital cost such as site preparation, grid connections, project development, and feasibility study. It is assumed that the wind project is financed by a loan which is 75% of the capital cost and the remaining by equity. The capital cost is spread out over the projected lifetime of the wind farm.

The annual cost incurred is given by [18]

$$\mathrm{AC} = A + E + M. \tag{4}$$

The annual payment on the loan is given by [18]

$$A = P\left[\left\{\frac{i(1+i)^n}{(1+i)^n - 1}\right\}\right] = P \times \mathrm{CRF}(i, n). \tag{5}$$

Finally, a levelized cost estimate for energy delivered by the wind farm has been found as follows [18]:

$$\mathrm{CEE} = \frac{\mathrm{AC}}{\mathrm{AE}}. \tag{6}$$

REFERENCES

[1] "IEEE PES wind plant collector system design WG, Wind power plant collector system design considerations," in *Proc. 2009 IEEE Power and Energy Society General Meeting*, Calgary, Canada, Jul. 2009.

[2] H. L. Willis, H. Tram, M. V. Enge, and L. Finley, "Optimization applications to power distribution," *IEEE Comput. Appl. Power*, vol. 8, no. 4, pp. 12–17, Oct. 1995.

[3] T. Short, *Electric Power Distribution Handbook*. Boca Raton, FL: CRC Press, 2003.

[4] E. Miguez, J. Cidras, E. Dorado, and J. Dornelas, "An improved branch-exchange for large-scale distribution network planning," *IEEE Trans. Power Syst.*, vol. 17, no. 4, pp. 931–936, Nov. 2002.

[5] S. K. Goswami and S. K. Basu, "A new algorithm for the reconfiguration of distribution feeders for loss minimization," *IEEE Trans. Power Del.*, vol. 7, no. 3, pp. 1484–1490, Jul. 1992.

[6] "IEEE PES wind plant collector system design WG, wind power plant grounding, overvoltage protection, and insulation coordination," in *Proc. 2009 IEEE Power and Energy Society General Meeting*, Calgary, Canada, Jul. 2009.

[7] "IEEE PES wind plant collector system design WG, wind power plant substation and collector system redundancy, reliability, and economics," in *Proc. 2009 IEEE Power and Energy Society General Meeting*, Calgary, Canada, Jul. 2009.

[8] "IEEE PES wind plant collector system design WG, design and application of cables and overhead lines in wind power plants," in *Proc. 2010 IEEE Transmission and Distribution Conf.*, New Orleans, LA, 2010.

[9] S. Dutta and T. Overbye, "A clustering based wind farm collector system cable layout design," in *Proc. 2011 IEEE Power and Energy Conf. at Illinois*, Urbana, IL, Feb. 2011.

[10] M. Hazewinkel, "Steiner Tree Problem," in *Encyclopaedia of Mathematics*. New York: Springer, 2001 [Online]. Available: http://eom.springer.de/s/s110270.htm, ISBN: 978-1556080104

[11] D. Z. Du and F. K. Hwang, "The Steiner ratio conjecture of Gilbert and Pollak is true," *Proc. National Academy of Sciences*, vol. 87, pp. 9464–9466, Dec. 1990 [Online]. Available: http://www.pnas.org/content/87/23/9464.full.pdf

[12] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[13] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Mathematical Statistics and Probability, University of California Press*, Berkeley, CA, 1967, pp. 281–297.

[14] A Tutorial on Clustering Algorithms [Online]. Available: http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html

[15] Maine Mountain Power, LLC, Preliminary Engineering for the Black Nubble Wind Farm 34.5 kV Collector System and 115 kV Interconnection Facility 54 MW Facility Jun. 18, 2007 [Online]. Available: http://www.maine.gov/doc/lurc/projects/redingtonrevised/Documents/Section01_Development_Description/Development_Electric/E_Pro_Reports/5.1_Electrical_Power_Line_Report.pdf

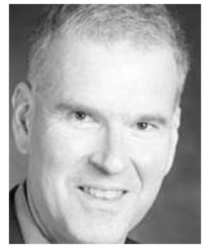
[16] Kerite Engineering Catalog [Online]. Available: http://www.kerite.com/pdf/Kerite%20Engineering%20Catalog.pdf.

[17] J. H. Neher and M. H. McGrath, "The calculation of the temperature rise and load capability of cable systems," *AIEE Trans.*, vol. 76, pp. 752–64, Oct. 1957.

[18] G. M. Masters, *Renewable and Efficient Electric Power Systems*. Hoboken, NJ: Wiley-Interscience, 2004.

Sudipta Dutta (S'08) received the B.E. degree from Jadavpur University, Kolkata, India, and the M.Tech. degree from the Institute of Technology, Banaras Hindu University, Varanasi, India, both in electrical engineering. She is currently working toward the Ph.D. degree at University of Illinois at Urbana-Champaign.

Her research interests are in the area of wind power forecasting, integration of storage with renewable energy, and wind farm collector system design.



Thomas J. Overbye (S'87–M'92–SM'96–F'05) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Wisconsin-Madison.

He is currently the Fox Family Professor of Electrical and Computer Engineering at the University of Illinois Urbana-Champaign. His research interests include power system visualization, power system analysis, and computer applications in power systems.