An Efficient Implementation of Decentralized Optimal Power Flow

Balho H. Kim*

Abstract – In this study, we present an approach to parallelizing OPF that is suitable for distributed implementation and is applicable to very large inter-connected power systems. The approach could be used by utilities for optimal economy interchange without disclosing details of their operating costs to competitors. It could also be used to solve several other computational tasks, such as state estimation and power flow, in a distributed manner. The proposed algorithm was demonstrated with several case study systems.

Keywords: Parallel optimal power flow, Predictor-corrector proximal multiplier method, Regional decomposition

1. Introduction

Since Dantzig and Wolfe [1] proposed their decomposition principle for linear programming in 1960, extensive work on large-scale mathematical programming has followed including that performed by Lasdon and co-workers [2, 3], and Takahara and Mesarovic et al. [4]. Recently, motivated by this influential work, various approaches have been taken to parallelize power system problems including reactive power optimization problem and constrained economic dispatch problem [5, 6].

Dating back to the late 90's, Kim et al. [7, 8] modeled a multi-area Optimal Power Flow in a distributed manner, where they introduced the concept of artificial generators and loads on the border buses connecting the adjacent regions. Beginning with this noble approach, abundant numerical techniques have been developed for solving power system problems in decentralized methodologies [9, 10, 11, 12, 13, 14, 15].

In this study, we propose that the OPF be solved in a decentralized framework, consisting of regions, using a price-based mechanism that models each region as an economic unit. In each region, a local processor would perform its own OPF for the region and its border as in [7]. Regions interact by adjusting flows between themselves depending on the prices quoted for inter-regional interchanges.

2. Predictor-Corrector Proximal Multiplier Method

In this section, we present the general concept of the proposed decomposition coordination method for solving large-scale problems with separable structure: the Predictor-Corrector Proximal Multiplier Method (PCPMM) [16]. The PCPMM is basically based on the augmented Lagrangian method and a variant of the Proximal Point Algorithm (PPA), which has long been recognized as one of the attractive methods for convex programming and min-max-convex-concave programming.

Consider a typical convex program with separable structure of the form:

$$(P) \qquad \min_{x,z}\{f_a(x)+f_b(z): Ax=z\} \tag{1}$$

Then the augmented Lagrangian for problem (P) is defined as

$$L(x,z,\lambda)=f_a(x)+f_b(z)+\lambda^{+}(Ax-z)+\frac{\gamma}{2}\|Ax-z\|^2, \tag{2}$$

where λ denotes a Lagrange multiplier and γ is a constant. Augmented Lagrangians have several advantages compared to standard Lagrangians. However, the principal disadvantage for decomposition methods is the presence of the term $\frac{\gamma}{2}\|Ax-z\|^2$ in the L, which destroys the separability between x and z, since they are linked by the cross product term. This has long been recognized as one of the major drawbacks of the augmented Lagrangian approach, and a number of strategies have been proposed to remove this difficulty [17, 18, 19, 20, 21].

In 1958, Uzawa suggested to simply minimize the Lagrangian function L with respect to x and z (with λ fixed), then update the multiplier λ [22]. In the method, both f_a and f_b are assumed to be *strongly convex*, and this

* Dept. of Electrical Engineering, Hongik University, Seoul, Korea (bhkim@wow.hongik.ac.kr)
Received 17 February, 2006 ; Accepted 12 October, 2006

restricts its potential applications in many interesting problems.

In [20], Chen proposes another PPA-based splitting method. It bears similarity with algorithms produced from the alternating direction method in that it preserves separability, but is completely different both in the computational steps and in the assumptions involved in the problem's data. In addition, PCPMM can be used for parallel decomposition wherein a problem is decomposed into many subproblems and solved independently at each coordination step.

The key features of PCPMM in our study will be presented first, and its applications for the distributed OPF will be given in Section 3.

The predictor-corrector proximal multiplier method is based on the properties of Rockafellar's proximal method of multipliers, and its primal-dual application [23], which involves an augmented Lagrangian with an additional quadratic proximal term.

Consider the following convex problem:

$$\min\{f(x) : x \in R^n\}, \tag{3}$$

where f is a proper, lower-semi-continuous convex function. To solve the problem is to find x such that $0 \in \partial f(x)$.

Let us consider the problem of finding, for an arbitrary maximal monotone operator T, an x satisfying $0 \in T(x)$.

To solve this problem is to find a fixed point of $P_\beta(x)$, the resolvent of T, which can be solved by the following iterative method:

$$x^{k+1} = P_\beta(x^k) = (I + \frac{1}{\beta}T)^{-1}(x^k) \cdot \tag{4}$$

Method (4) is called the *proximal point algorithm* (PPA). That is, starting from an initial point $u^0 \in R^n$, the iterative scheme of PPA is given as

$$u^{k+1} = (I + \frac{1}{\beta_k}\partial f)^{-1}(u^k) \Leftrightarrow u^{k+1} = \arg\min\{f(u) + \frac{\beta_k}{2}\|u - u^k\|^2\},$$
$$\Leftrightarrow \beta_k(u^k - u^{k+1}) \in \partial f(u^k) \cdot \tag{5}$$

Let us recall problem (1). In the proximal method of multipliers, one has to apply the proximal point algorithm to the penalized Lagrangian

$$L(x, z, \lambda^k) + \frac{1}{2\beta_k}\|Ax - z\|^2 \tag{6}$$

in each minimization step. Then, using (5), one can obtain

$$\beta_k(x^k - x^{k+1}) \in \partial f_a(x^{k+1}) + A^T(\lambda^k + \frac{1}{\beta_k}(Ax^{k+1} - z^{k+1})), \tag{7}$$

$$\beta_k(z^k - z^{k+1}) \in \partial f_b(z^{k+1}) + (\lambda^k + \frac{1}{\beta_k}(Ax^{k+1} - z^{k+1})) \cdot \tag{8}$$

This iterative scheme, however, cannot compute x^{k+1} and z^{k+1} separately because of the coupling term $(Ax^{k+1} - z^{k+1})$. Chen [16] suggests the following iterative scheme to remove this difficulty,

$$\beta_k(x^k - x^{k+1}) \in \partial f_a(x^{k+1}) + A^T(\lambda^k + \frac{1}{\beta_k}(Ax^k - z^k)), \tag{9}$$

$$\beta_k(z^k - z^{k+1}) \in \partial f_b(z^{k+1}) + (\lambda^k + \frac{1}{\beta_k}(Ax^k - z^k)) \cdot \tag{10}$$

where the term $(Ax^{k+1} - z^{k+1})$ has been replaced with $(Ax^k - z^k)$.

Similarly, from the conventional Lagrangian multiplier update rule

$$\lambda^{k+1} = \lambda^k + \frac{1}{\beta_k}(Ax^{k+1} - z^{k+1}) \tag{11}$$

one can produce a prediction $\tilde{\lambda}^{k+1}$ of λ^{k+1} by replacing the term $(Ax^{k+1} - z^{k+1})$ with $(Ax^k - z^k)$,

$$\tilde{\lambda}^{k+1} = \lambda^k + \frac{1}{\beta_k}(Ax^k - z^k) \cdot \tag{12}$$

Then (9) and (10) can be rewritten as

$$x^{k+1} = \arg\min\{f_a(x) + (\tilde{\lambda}^{k+1}) + Ax + \frac{\beta_k}{2}\|x - x^k\|^2\} \tag{13}$$

$$z^{k+1} = \arg\min\{f_b(z) + (\tilde{\lambda}^{k+1}) + z + \frac{\beta_k}{2}\|z - z^k\|^2\} \cdot \tag{14}$$

Rearranging (12)-(14), one finally obtains the following PCPMM algorithm for solving problem (1):

Algorithm - PCPMM

Step 1: Initialization

Step 2: Compute $\tilde{\lambda}^{k+1} = \lambda^k + \frac{1}{\beta_k}(Ax^k - z^k)$.

Step 3: Solve

$$x^{k+1} = \arg\min\{f_a(x) + (\tilde{\lambda}^{k+1}) + Ax + \frac{\beta_k}{2}\|x - x^k\|^2\}$$

$$z^{k+1} = \arg\min\{f_b(z) + (\widetilde{\lambda}^{k+1}) + z + \frac{\beta_k}{2}\left\|z - z^k\right\|^2\}$$

Step 4: Compute $\lambda^{k+1} = \widetilde{\lambda}^k + \gamma_k(Ax^{k+1} - z^{k+1})$.

Step 5: Repeat Steps 2-4.

At each iteration, the algorithm computes two proximal steps in the dual variables, the predictor step $\widetilde{\lambda}^{k+1}$ and the corrector step λ^{k+1}, and one proximal step in the primal variables. The algorithm preserves the good features of the proximal method of multipliers, with the additional advantage that it leads to a decoupling of the constraints, and is thus suitable for parallel implementation. It has been proved that under very mild assumptions on the problem's data, the method is globally convergent at a linear rate [16, 24].

3. Implementation of PCPMM to Parallel Optimal Power Flow

We propose a scenario where each individual utility solves a modified OPF that includes its own service area and the borders it shares with other utilities. The modified OPF is similar to a standard OPF except that artificial generators (dummy generators) are modeled at the border buses. Naturally, the OPFs solved in each region can be implemented with the fastest available algorithms. However, it is also possible for each utility (equivalently, region, or control area) to have a different OPF implementation for its own area.

The overall algorithm involves alternating solutions of individual OPFs and updates of prices. It converges, in principle, to a solution of the overall multi-utility OPF, yielding appropriate generation levels in each utility to minimize overall production costs. The multipliers on the constraints could be used to set prices for the exchange of real and reactive power. However, alternative ways to distribute savings, such as the split savings rule, can also be used.

To minimize the coupling between the regions and therefore maximize the solution speed, it is best to divide the overall system in a way that minimizes the number of transmission lines in the cutsets of lines defining the regions. Fortunately, this will usually be consistent with dividing a multi-utility system into individual utilities. This is because typical utilities tend to have a relatively complex mesh transmission system internally, but relatively few, often radial, connections externally. In summary, the most effective implementation of our parallel scheme corresponds well with the most likely institutional implementation: division into regions along utility boundaries. In our scheme, there is neither need for a uniform implementation of OPF across all utilities, nor need for all the utilities in the system to run full OPFs, so long as each region can represent dummy generators in its OPF or economic dispatch (ED). This means that the parallel OPF can be implemented across a multi-utility system without major disruption to existing OPF or ED investments by individual utilities.

In this section, we will first present how an OPF problem can be decomposed regionally using the mathematical decomposition techniques described in the previous section. Then details on the problem formulation and practical implementation of the Distributed OPF algorithm will be given, followed by a short description on the inclusion of contingency constraints into our distributed OPF scheme.

3.1 Regional Decomposition

In our distributed scheme, the regions buy and sell electricity from adjacent regions at prices that are coordinated by negotiations between adjacent regions. The price-setting itself can be performed without a centralized processor. The advantage of such decentralization is that only synchronization information needs to be exchanged globally, improving reliability in the event of communication failure.

To illustrate the main issues in regional decomposition, we will consider dividing a power system into two overlapping regions. In the following section, we define problem variables and constraints and then discuss the OPF formulation, decomposition, and implementation.

3.1.1 Variables

Because of our emphasis on the decomposition rather than on the OPF itself, we will follow [25] in not distinguishing the controls from the dependent variables in our formulation. Instead, we will distinguish the variables by their geographical relationship to the regional decomposition.

Consider Fig. 1, which shows the case of a single tie-line joining regions a and b. Notice that the border variables in the overlap region are denoted y, while the core variables in regions a and b are denoted x and z, respectively. Between and common to the two regions there is an overlap region, with a vector of variables denoted by y. The entries in y are defined as follows. For each tie-line we must include a bus in the border region. If there is no bus already there, we create a "dummy bus." Associated with each dummy bus are the real and reactive power flows through the bus and the voltage and angle at the bus. That is, the vector y has four entries for each tie-line.

In addition, in Fig. 1 we display vectors of variables x and z. The vector x consists of all the OPF variables that are relevant to region a but not already included in y. Similarly, z includes the region b variables not included in y. In summary, region a has state vector (x, y), while region b has state vector (y, z). The y variables are the overlap or border variables, while χ and z can be thought of as core variables for regions a and b, respectively. In typical systems, the vector y will be much smaller than vectors x and z and we will make use of this observation in our analysis of the decomposition.

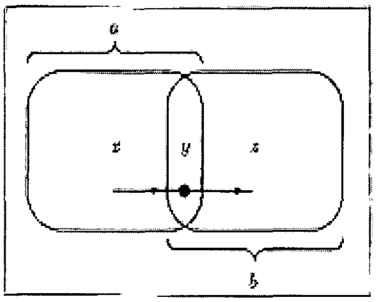


Fig. 1. Decomposition of a power system into two overlapping regions, a and b.

3.1.2 Constraints

We assume that the constraints on the system involve x and y or y and z, but not x and z nor x, y, and z. That is, we assume that the constraints in each region involve only the core variables and the border variables for that region. This assumption is reasonable for the power flow equations, since the bus admittance matrix couples only those variables pertaining to buses that are connected directly by a line. For example, a tie-line limit would be represented as a constraint on the flows to and from the border buses. If some of the other constraints are functions of both x and z elements, then this can be handled by moving more of the state vector into the border vector y. That is, our assumption on the dependence of constraints on core and border variables can always be satisfied, but it may require us to increase the dimension of y by enlarging the border region.

With this assumption, we can write the power flow constraints for region a in the form $g_a(x, y) = 0$ and for region b in the form $g_b(y, z) = 0$. Similarly, we can write the inequality constraints for region a in the form $h_a(x, y) \le 0$ and for region b in the form $h_b(y, z) \le 0$. The functions h_a and h_b represent the line flow, voltage, and contingency constraints in the individual regions.

Define the two sets: $A = \{(x, y) : g_a(x, y) = 0, h_a(x, y) \le 0\}$ and $B = \{(y, z) : g_b(y, z) = 0, h_b(y, z) \le 0\}$. Then a feasible power flow solution is a point (x, y, z) that satisfies $(x, y) \in A$ and $(y, z) \in B$.

3.2 OPF Formulation

With the above definition on variables and constraints, the OPF problem can be written as

$$\min_{\substack{(x,y)\in A\\(y,z)\in B}} \{f_a(x) + f_b(z)\}, \tag{15}$$

where we assume that the cost functions f_a and f_b are convex approximations to the actual cost functions in each region and that there is a unique solution to (15). We decompose problem (15) into regions by duplicating the border variables and imposing coupling constraints between the two variables.

First, define the copies of y to be y_a and y_b, assigned to the regions a and b, respectively. Then problem (15) is equivalent to:

$$\min_{\substack{(x,y)\in A\\(y,z)\in B}} \{f_a(x) + f_b(z) + \frac{\gamma}{2}\|y_a - y_b\|^2 : y_a - y_b = 0\}. \tag{16}$$

The quadratic term added to the objective does not affect the solution since the constraint $y_a - y_b = 0$ will make the quadratic term equal to zero at any solution; however, when we decompose the problem, this term will significantly aid in convergence [26].

3.3 Decomposition

Next we apply the three decomposition algorithms described in the previous chapter to obtain sub-problems for a distributed implementation.

Algorithm PCPMM

Similarly, using Algorithm-PCPMM (11)-(14), we obtain the following regional OPF problems:

$$(x^{k+1}, y_a^{k+1}) = \arg\min_{(x,y_a)\in A}\{f_a(x) + \frac{\beta_k}{2}\|y_a - y_a^k\|^2 + (\tilde{\lambda}^{k+1})^T y_a\}, \tag{17}$$

$$(z^{k+1}, y_a^{k+1}) = \arg\min_{(y_b,z)\in B}\{f_b(z) + \frac{\beta_k}{2}\|y_b - y_b^k\|^2 - (\tilde{\lambda}^{k+1})^T y_b\}, \tag{18}$$

where the Lagrange multiplier $\tilde{\lambda}$ is updated by (11) and (12).

A natural implementation of the proposed Algorithm-PCPMM is given in Fig. 2.

The telemeter and Dispatch steps require intra-regional communication of data and control signals. The loop termination criterion requires global communication, while the Exchange step only requires communication between adjacent regions. In the case of multiple regions, each region will solve an OPF for its core and border variables. In Fig. 1, we illustrated a single tie-line between regions a and b. At the borders bus, there is a real and reactive

Initialize $x^0, y_a^0, y_b^0, z^0, \lambda^0$;
$k = -1$;
Telemeter load and topology data from each region to its processor;
Repeat{
Increment k;
In parallel, solve the regional OPF for region a and region b;
Exchange y_a^k and y_b^k between regional processors;
Update λ^{k+1};
} Until y_a^k and y_b^k converge to within tolerance;
Dispatch generators according to OPF solution.

Fig. 2. Distributed implementation of parallel OPF.

power flow, measured for example in the direction from a to b, and a voltage magnitude and phase. The dummy generators are created by duplicating this bus. The border variables are then $y_a = (p_a q_a v_a \theta_a)^T$ and $y_b = (p_b q_b v_b \theta_b)^T$, respectively, the real and reactive power flow and the voltage magnitude and phase at the copies of the dummy bus in regions a and b. The iterative process drives the values of y_a and y_b together.

4. Case Studies

Several case studies were performed to demonstrate the proposed distributed OPF algorithm. The objectives of the case studies are, first, to verify the viability of the algorithms in practical implementation and, second, to test and compare the overall performance of the algorithms. Performance comparisons are based on the cpu times and number of iterations required for desired accuracy.

For the case studies, a state-of-the art Interior-Point OPF code (INTOPF) [7, 25] was employed. Non-contingency constrained AC OPFs were performed for all cases with real and reactive generator limits and line and voltage constraints imposed. All computations were performed on the Pentium-IV processor.

4.1 Case Study Systems

Data from the IEEE 118-bus Reliability Test System and eight Texas utilities in the ERCOT (Electric Reliability Council of Texas) power pool were used to demonstrate the performance of the algorithm. Table 1 summarizes the test systems. The first column denotes the system identification number, which will be used throughout the paper instead of real names, the second column shows the total number of buses in each system, while the third and fourth columns reveal the number of regions and the number of core buses in each region. The fifth column displays the number of tie-lines that interconnect the regions, while the sixth column indicates the total number of lines in each complete system. The last column shows the total per unit loads in the systems. The five smaller systems consist of two, three, or four copies of two IEEE Test Systems, while the four Texas systems use data from two to eight Texas utilities.

Table 1. Case study systems.

No	Buses	Regions	Core Buses	Ties	Lines	Load
1	360	3	118,118,118	6	570	126
2	753	4	271,105,128,237	12	1100	209
3	1459	6	271,105,128,237,365,325	28	2145	395
4	1777	8	271,105,128,237,365,325,74,213	59	2587	462

The objective to be minimized is the production cost for active and reactive power. The cost of reactive power is assumed to be 10^{-3} of the active power cost for each generator, while real power costs were adapted from [27] and [28].

In order to see how the algorithm responds to small changes in system status, we solved a base-case and several change-cases for each system. Each base-case was solved from a flat start with initially no interchange on any tie-line, while the change-cases were solved using the solution of the base-case as a starting point. The change-cases were as follows:

- increase in demand of 5% at all demand buses;
- increase in demand of 10% at all demand buses;
- an outage of a single generator with capacity equal to approximately 2-3% of the total system demand.

The change-cases demonstrate the tracking behavior of the algorithm for an on-line application.

4.2 Stopping Criterion

We chose the maximum mismatch between the border variables as the stopping criterion. To select the tolerance on the maximum mismatch, we experimented with the performance of the algorithm. We found that the choice 0.03 per unit maximum mismatch yielded a solution with total costs that were within 0.1% of the optimal production costs from the serial algorithm. Typically, the mismatches on most buses were much smaller than 0.03 per unit.

4.3 Test Results

Selected case study results are presented in this section. To compare the overall performance of the algorithm, the total cpu times and iteration counts are tabulated. Then the speed-ups and efficiency of the algorithms are discussed.

The cpu time results from the undecomposed and the parallel implementation of INTOPF code, which are summarized in Tables 3 and 4, respectively, where all the cpu times include the overheads necessary for reading data and communicating among processors. As seen in Table 3, the cpu times and the number of buses have almost a linear relationship. Table 4 shows that the first iteration of the INTOPF algorithm takes much more cpu time than each subsequent iteration. Table 5 indicates the measured cpu time for the base-case for the serial and parallel implementations of the proposed algorithm. The estimated efficiencies for the larger systems are between about 55 and 60%, based on the 0.03 per unit tie-line mismatch criterion.

Table 2. Number Iterations for parallel OPF: Algorithm-PCPMM.

System Number	No.1	No.2	No.3	No.4
Based case	6	7	8	10
5% Demand Incr	5	6	7	9
10% Demand Incr	4	5	5	9
outage case	4	5	5	9

Table 3. CPU time for undecomposed OPF (sec).

System Number	No.1	No.2	No.3	No.4
Base case	10.6	32.7	54.6	77.4

Table 4. Cumulative CPU time for parallel (decomposed) OPF (sec).

System Number	No.1	No.2	No.3	No.4
Iteration=1	1.3	5.8	6.8	6.9
Iteration=5	4.1	10.1	11.8	11.3
Iteration=10	6.4	13.4	16.5	17.1

Table 5. Speed-Up and Efficiency.

System Number	No.1	No.2	No.3	No.4
Cputime (Undecomposed)	10.6	32.7	54.6	77.4
Cputime (Decomposed)	4.6	10.8	14.6	17.1
Speed-Up	2.3	3.0	3.7	4.5
Efficiency (%)	76.7	75.0	61.7	56.3

The case study results show that almost all of the potential production cost savings are achieved within 4 or 5 iterations. If we terminate after 4 or 5 iterations, then the efficiency improves to 80%, with production costs still within 0.1% of optimal amount. The size of our test systems is modest and the ratio of the number of borders to core variables is large. We expect better performance for larger systems with lower ratios of border to core variables.

5. Conclusion

We have demonstrated a parallel algorithm for the OPF problem that is cable of distributed implementation. Based on the case study results, the proposed algorithm has a great advantage to the conventional undecomposed algorithm.

Our future study is first to explore ways to improve convergence of the algorithm. An important challenge is to theoretically analyze the improvement in convergence speed due to the quadratic term. Finally, incorporation of contingency constraints will also be studied. We will investigate ways to represent security constraints and to solve the Security Constrained Optimal Power Flow (SCOPF) efficiently and reliably in a distributed manner.

Acknowledgements

This work was funded by the Ministry of Education and Human Resources Development for BK21 (2nd Phase) Project 'Research on New Energy Resource & Power System Interface' and done during the sabbatical year (09/01/2005 – 08/31/2006).

References

[1] G. B. Dantzig and P. Wolfe, "Decomposition principle for linear programs", *Operations Research*, 8, January 1960.

[2] L. S. Lasdon. *Optimization Theory for Large Systems*, The Macmillan Company, New York, 1970.

[3] L. S. Lasdon and S. D. Schoeffer, "A multilevel technique for optimization," *Proc, JACC, Troy*, NY, 1965.

[4] M. D. Mesarovic, D. Macko, and Y Takahara, *Theory of Hierarchical, Multilevel, Systems*, Mathematics in Science and Engineering. Academic Press, New York, 1970.

[5] N. I. Deeb and S. M. Shahidehpour, "Linear reactive power optimization in a large power network using the decomposition approach," *IEEE Transactions on Power Systems*, vol. 5, no. 2, pp. 428-438, May 1990.

[6] J. Wernerus and L. Soder, "Area price based multi-area economic dispatch with transmission losses and constraints," *Paper SPT PS 75-05-0337 presented at*

IEEE/KTH Stockholm Power Tech Conference, Stockholm, Sweden, June 18-77, 1995.

[7] B. H. Kim and R. Baldick, "Coarse-grained distributed optimal power flow," *IEEE Transactions on Power Systems*, vol. 12, no. 2, pp. 932-939, May 1997.

[8] R. Baldick, B. H. Kim, C. Chase, and Y. Luo, "A first distributed implementation of optimal power flow," *IEEE Transactions on Power Systems*, vol. 14, pp. 853-863, August 1999.

[9] A. J. Conejo, J. A. Aguado, "Multi-area coordinated decentralized DC optimal power flow," *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1272-1278, November 1998.

[10] A. G. Bakirtizis and P. N. Biskas, "A decentralized solution to the DC-OPF of interconnected power system," *IEEE Transactions on Power Systems*, vol. 18, pp. 1007-1013, August 2003.

[11] P. N. Biskas, A. G. Bakirtzis, "A decentralized implementation of DC optimal power flow on a network of computers," *IEEE Transactions on power Systems*, vol. 20, no. 1, pp. 25-33, February 2005.

[12] J. A. Aguado and V. H. Quintana, "Inter-utilities power exchange coordination: A market-oriented approach," *IEEE Transactions on Power Systems*, vol. 16, pp. 513-519, 2001.

[13] X. Wang, Y. H. Song, and Q. Lu, "Lagrangian decomposition approach to active power congestion management across interconnected regions," *IEE Proc. Gener. Transm. Distrib.*, vol. 148, pp. 497-503, 2001.

[14] K. Xie and Y. H. Song, "Power market oriented optimal power flow via an interior point method," *IEE Proc. Gener. Transm. Distrib.*, vol. 148, pp. 549-556, 2001.

[15] W. D. Rosehart, C. A. Canizares, and V. H. Quintana, "Multiobjective optimal power flows to evaluate voltage security costs in power networks," *IEEE Transactions on Power Systems*, vol. 18, pp. 578-587, 2003.

[16] G. Chen. *Proximal and Decomposition Method for Convex Programming*, PhD thesis, University of Maryland Baltimore, 1993.

[17] E. Gelman and J. Mandel, "On multilevel iterative methods for optimization problems," *Mathematical Programming*, 48, pp. 1-17, 1990.

[18] O. Guler, "On the convergency of the proximal point algorithm for convex minimization," *SIAM Journal on Control and Optimization*, 29(2), pp. 403-419, March 1991.

[19] P. Tseng. "Applications of a splitting algorithm to decomposition in convex programming and variational inequalities," *SIAM Journal on Control and Optimization*, 29(1), pp. 119-138, January 1991.

[20] N. Watanabe, Y. Nishimura, and M. Matsubara, "Decomposition in large system optimization using the method of multipliers," *SIAM Journal on Control and Optimization*, 25(2), pp. 181-193, June 1978.

[21] A. P. Wierzbiki and S. Kurcyusz, "Projection on a cone, penalty functionals and duality theory for problems with inequality constraints in Hilbert space," *SIAM Journal on Control and Optimization*, 15(1), pp. 25-56, January 1977.

[22] H. Uzawa. *Studies in Linear and Nonlinear Programming*, Springer-Verlag, 1958.

[23] R. T. Rockafellar, "Hamiltonian trajectories and duality in the optimal control of linear systems with convex costs," *SIAM Journal on Control and Optimization*, 27(5): 1007-1025, September 1987.

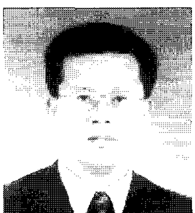
[24] D. G. Luenberger. *Linear and Nonlinear Programming*, Addison-Wesley Publishing Company, Reading, Massachusetts, second edition, 1984.

[25] Y. Wu, A. S. Debs, and R. I. Marsten, "A direct nonlinear predictor-corrector primal-dual interior point algorithm for optimal power flows," *IEEE Transactions on Power Systems*, vol. 9, no. 2, pp. *876-883*, May 1994.

[26] J. Batut and A. Renaud, "Daily generation scheduling optimization with transmission constraints: A new class of algorithms," *IEEE Transactions on Power Apparatus and Systems*, vol. 7, no. 3, pp. 982-989, August 1992.

[27] Reliability Test System Task Force of the application of probability methods subcommittee, "The IEEE Reliability Test System-1996," *Paper 96 WM 326-9 PWRS presented at IEEE/PES Winter Meeting, Baltimore, MD, January 21-25*, 1996.

[28] M. L. Baughman et al., *Electric Utility Resource Planning and Production Costing Projects: Final Report,* Center for Energy Studies, The University of Texas at Austin, 1993.



Balho H. Kim
He received his B.S. degree from Seoul National University, Korea, in 1984, and his M.S. and Ph.D. degrees from the University of Texas, Austin in 1992 and 1996, respectively. He was with KEPCO (the Korea Electric Power Corporation) from 1984 to 1990 and then joined Hongik University in 1997 where he is presently an Associate Professor of Electrical Engineering.