Unit Commitment by Lagrangian Relaxation and Genetic Algorithms

Chuan-Ping Cheng, Chih-Wen Liu, and Chun-Chang Liu

***Abstract*—This paper presents an application of a combined the Genetic Algorithms (GA's) and Lagrangian Relaxation (LR) method for the unit commitment problem. Genetic Algorithms (GA's) are a general purpose optimization technique based on principle of natural selection and natural genetics. The Lagrangian Relaxation (LR) method provides a fast solution but it may suffer from numerical convergence and solution quality problems. The proposed Lagrangian Relaxation and Genetic Algorithms (LRGA) incorporates Genetic Algorithms into Lagrangian Relaxation method to update the Lagrangian multipliers and improve the performance of Lagrangian Relaxation method in solving combinatorial optimization problems such as unit commitment problem. Numerical results on two cases including a system of 100 units and comparisons with results obtained using Lagrangian Relaxation (LR) and Genetic Algorithms (GA's), show that the feature of easy implementation, better convergence, and highly near-optimal solution to the UC problem can be achieved by the LRGA.**

I. Introduction

THE task of Unit Commitment (UC) involves scheduling the on/off status, as well as the real power outputs, of thermal units for use in meeting forecasted demand over a future short-term (24–168 hour) horizon. The resultant schedule should minimize the system production cost during the period while simultaneously satisfying the load demand, spinning reserve, physical and operational constraints of the individual unit. Since improved UC schedule may save the electric utilities millions of dollars per year in production costs, UC is an important optimization task in the daily operation planning of modern power systems.

Due to its goal, the UC problem has commonly been formulated as a nonlinear, large scale, mixed-integer combinatorial optimization problem with constraints. The exact solution to the problem can be obtained only by complete enumeration, often at the cost of a prohibitively computation time requirement for realistic power systems[1]. Research endeavors, therefore, have been focused on, efficient, near-optimal UC algorithms which can be applied to large-scale power systems and have reasonable storage and computation time requirements. A survey of literature on the UC methods reveals that various numerical optimization techniques have been employed to approach the UC problem. Specifically, there are priority list methods [2][3], integer programming [4][5], dynamic programming [6]–[11], branch-and-bound methods [15], mixed-integer programming [14], and Lagrangian relaxation methods [12][13]. Among these methods, the priority list method is simple and fast but the quality of final solution is rough. Dynamic programming methods which are based on priority lists are flexible but the computation time suffers from "curse of dimensionality." Branch-and-bound adopts a linear function to represent the fuel consumption and time-dependent start cost and obtains the required lower and upper bounds. The shortcoming of branch-and-bound is the exponential growth in the execution time with the size of the UC problem. The integer and mixed-integer methods adopt linear programming technique to solve and check for an integer solution. These methods have only been applied to small UC problem and have required major assumptions which limit the solution space. The Lagrangian relaxation method provides a fast solution but it may suffer from numerical convergence and solution quality problems.

Aside from the above methods, there is another class of numerical techniques applied to UC problem. Specifically, there are artificial Neural Network[16][17], Simulated Annealing (SA)[18], and Genetic Algorithms (GA's) [19][20][23][28]. These methods can accommodate more complicated constraints and are claimed to have better solution quality. SA is a powerful, general-purpose stochastic optimization technique, which can theoretically converge asymptotically to a global optimum solution with probability 1. One main drawback, however, of SA is that it takes much CPU time to find the near-global minimum. GA's are a general-purpose stochastic and parallel search method based on the mechanics of natural selection and natural genetics. GA's are a search method to have potential of obtaining near-global minimum.

In this paper, we apply an Lagrangian Relaxation and Genetic Algorithms (LRGA) method in solving the UC problem. The basic idea of LRGA is that Genetic Algorithms (GA's are incorporated into Lagrangian Relaxation (LR) method to update the Lagrangian multipliers and improve the performance of Lagrangian Relaxation method. A description of LRGA method is presented in Section II. Then a detailed application of LRGA method to UC is given in Section III. The analysis of the LRGA method is given in Section IV. Numerical tests on two cases using LRGA, LR and GA's are compared in Section V. Finally, a conclusion is given in Section VI.

II. Lagrangian Relaxation and Genetic Algorithms

The Lagrangian Relaxation (LR) method solves the unit commitment problem by "relaxing" or temporarily ignoring the coupling constraints and solving the problem as if they did not exist. The LR decomposition procedure, based on the dual optimization theory, generates a separable problem by integrating

Manuscript received August 10, 1998; revised April 26, 1999.

The authors are with the Department of Electrical Engineering National Taiwan University Taipei, Taiwan.

Publisher Item Identifier S 0885-8950(00)03809-8.

some coupling constraints into the objective function, through "penalty factors," which are functions of the constraint violation. The "penalty factors," referred to as Lagrangian multipliers, are determined iteratively. Instead of solving the primal problem, one can solve the dual by maximizing the Lagrangian function with respect to the Lagrangian multipliers, while minimizing with respect to the unit commitment control variable.

The LR decomposition procedure is dependent on the initial estimates of the Lagrangian multipliers and on the method used to update the multipliers. Another difficulty with LR based methods is that computational performance is very dependent on the method by which the Lagrangian multipliers are updated. Currently most techniques used for estimating the Lagrangian multipliers rely on a sub-gradient algorithm or heuristics.

The Lagrangian Relaxation and Genetic Algorithms (LRGA) method incorporates Genetic Algorithms (GA's) into Lagrangian Relaxation (LR) to update the Lagrangian multipliers and improve the performance of LR method. The Genetic Algorithms (GA's) combine the adaptive nature of the natural genetics or the evolution procedures of organs with functional optimizations. By simulating "the survival of the fittest" of Darwiniam evolution among chromosome structures, the optimal chromosome (solution) is searched by randomized information exchange. The three prime operators associated with the GA's are reproduction, crossover and mutation. An explanation of these genetic operators applied to UC problem is given in Explanation 4 of Section III. The LRGA method consists of a two-stage cycle. The first stage is to search for the constrained minimum of Lagrangian function under constant Lagrangian multipliers by two-state dynamic programming. The second stage is to maximize the Lagrangian function with respect to the Lagrangian multipliers adjusted by Genetic Algorithms. Fig. 1 shows the basic configuration of the LRGA algorithm.

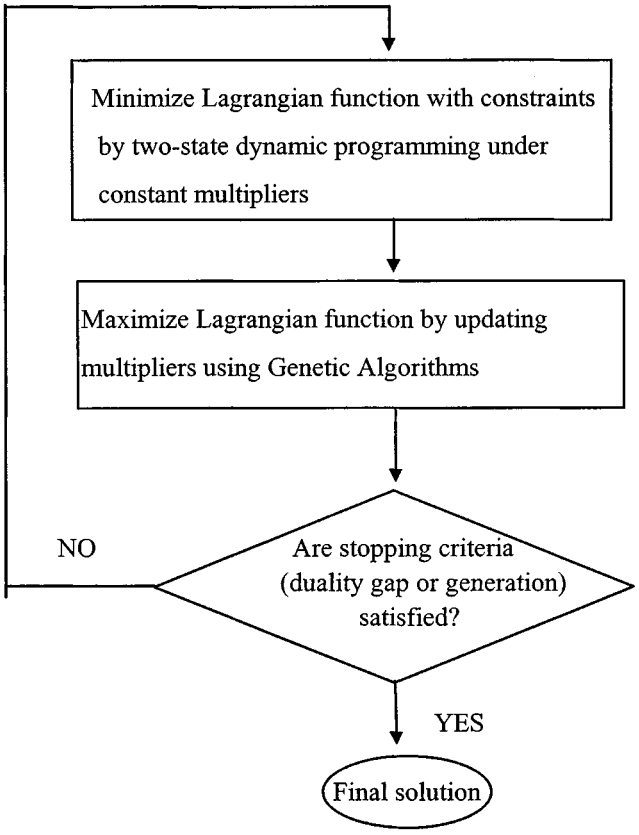


Fig. 1. The basic configuration of the LRGA algorithm.

III. Application to Unit Commitment Problem

In this section, we first formulate the UC problem, and then present a detailed LRGA algorithm for solving the UC problem.

The objective of the UC problem is the minimization of the total production costs over the scheduling horizon. Therefore, the objective function is expressed as the sum of fuel and start-up costs of the generating units. Mathematically, the function is as follows:

$$J(X, P) = \sum_{i=1}^{N} \sum_{h=1}^{H} \left[F_i(P_{ih}) + ST_i \left(1 - X_{i(h-1)}\right)\right] X_{ih} \tag{1}$$

Due to the operational requirements, the minimization of the objective function is subjected to the following constraints:

(a) power balance constraints

$$\sum_{i=1}^{N} P_{ih} X_{ih} \geq D_h \tag{2}$$

(b) spinning reserve constraints

$$\sum_{i=1}^{N} X_{ih} P_{i(\max)} \geq D_h + R_h \tag{3}$$

(c) generation limit constraints

$$P_{i(\min)} \leq P_{ih} \leq P_{i(\max)} \tag{4}$$

(d) minimum up-time constraints

$$X_{ih} = 1 \qquad \text{for} \sum_{t=h-up_i}^{h-1} X_{it} < up_i \tag{5}$$

(e) minimum down-time constraints

$$X_{ih} = 0 \qquad \text{for} \sum_{t=h-down_i}^{h-1} (1 - X_{it}) < down_i \tag{6}$$

where the notations used are

$F_i(P_{ih})$:fuel cost function of the i-th unit with generation output, P_{ih}, at the h-th hour. Usually, it is a quadratic polynomial with coefficients a_i, b_i and c_i as follows:

$$F_i(P_{ih}) = a_i P_{ih}^2 + b_i P_{ih} + c_i$$

N : the number of units,
H : the number of hours.
P_{ih} : the generation output of the i-th unit at the h-th hour,
P : the generation matrix with P_{ih}, $i = 1, \cdots, N$; $h = 1, \cdots, H$, as elements,
ST_i : start-up cost of the i-th unit,
X_{ih} : the on/off status of the i-th unit at the h-th hour, and $X_{ih} = 0$ when off, $X_{ih} = 1$ when on,

X : the schedule matrix with X_{ih}, $i = 1, \cdots, N$; $h = 1, \cdots H$, as elements,
D_h : load demand at the h-th hour,
R_h : spinning reserve at the h-th hour,
$P_{i(\min)}$: minimum generation limit of i-th unit,
$P_{i(\max)}$ maximum generation limit of i-th unit,
up_i : minimum up-time of i-th unit,
$down_i$: minimumdown-time of i-th unit.

By assigning nonnegative Lagrangian multipliers λ^h, $h = 1, \cdots, H$, and μ^h, $h = 1, \cdots, H$, to the constraints (2) and (3), respectively, we can then form the corresponding Lagrangian function

$$\begin{aligned} &L(X, P, \lambda, \mu) \\ &\quad = J(X, P) + \sum_{h=1}^{H} \lambda^h \left(D_h - \sum_{i=h}^{N} P_{ih} X_{ih} \right) \\ &\qquad + \sum_{h=1}^{H} \mu^h \left(D_h + R_h - \sum_{i=1}^{N} P_{i(\max)} X_{ih} \right) \end{aligned} \tag{7}$$

The Lagrangian relaxation procedure solves the UC problem by "relaxing" or temporarily ignoring the coupling constraints and solving the problem as if they did not exist. This is done through the dual optimization procedure which attempts to reach the constrained optimum by maximizing the Lagrangian function L with respect to the Lagrangian multipliers λ^h and μ^h, while minimizing with respect to the control variable P_{ih} and X_{ih} in UC problem, that is:

$$q^* = \max_{\lambda, \mu} q(\lambda, \mu)$$

where

$$q(\lambda, \mu) = \min_{X, P} L(X, P, \lambda, \mu)$$

subject to constraints (4),(5) and (6) Assume that λ and μ are fixed, we minimize the Lagrangian function L as follows. First from Eq.(7) we have

$$\begin{aligned} &L(X, P, \lambda, \mu) \\ &\quad = \sum_{h=1}^{H} \sum_{i=1}^{N} \left[F_i(P_{ih}) + ST_i \left(1 - X_{i(h-1)}\right) \right] \\ &\qquad \cdot X_{ih} + \sum_{h=1}^{H} \lambda^h \left(D_h - \sum_{i=1}^{N} P_{ih} X_{ih} \right) \\ &\qquad + \sum_{h=1}^{H} \mu^h \left(D_h + R_h - \sum_{i=1}^{N} P_{i(\max)} X_{ih} \right) \\ &\quad = \sum_{i=1}^{N} \left\{ \sum_{h=1}^{H} \left[F_i(P_{ih}) + ST_i \left(1 - X_{i(h-1)}\right) \right. \right. \\ &\qquad \left. \left. - \lambda^h P_{ih} - \mu^h P_{i(\max)} \right] \right. \\ &\qquad \left. \cdot X_{ih} + \sum_{h=1}^{H} (\lambda^h D_h + \mu^h D_h + \mu^h R_h) \right. \end{aligned}$$

The term

$$\sum_{h=1}^{H} \left[F_i(P_{ih}) + ST_i \left(1 - x_{i(h-1)}\right) \lambda^h P_{ih} - \mu^h P_{i(\max)} \right] X_{ih}$$

can be solved separately for each generating unit, without regard for what is happening on the other generating units. Then the minimum of the Lagrangian function L is found by solving the minimum for each generating unit over all time periods; that is

$$\begin{aligned} \min L(X, P, \lambda, \mu) = \sum_{i=1}^{N} \min \sum_{h=1}^{H} \\ \cdot \left[F_i(P_{ih}) + ST_i \left(1 - X_{i(h-1)}\right) \right. \\ \left. - \lambda^h P_{ih} - \mu^h P_{i(\max)} \right] X \\ + \sum_{h=1}^{H} (\lambda^h D_h + \mu^h D_h + \mu^h R_h) \end{aligned}$$

subject to constraints (4),(5) and (6). The minimum of Lagrangian function $L(X, P, \lambda, \mu)$ is easily found by solving a two-state dynamic programming problem in two variables for each unit. In order to maximize the Lagrangian function with respect to the Lagrangian multipliers, the adjustment of Lagrangian multipliers must be done carefully. Most references to adjust Lagrangian multipliers use a combination of subgradient search and various heuristics to achieve a rapid solution. In this paper, we use the Genetic Algorithms to adjust the Lagrangian multipliers and improve the performance of Lagrangian Relaxation method. A step-by-step LRGA method for the UC problem is outlined as follows.

The LRGA Algorithm for UC
Step 1: Initialize the parameters such as the size of population, the mutation rate, the crossover rate, the max generation, the duality gap, etc.
Step 2: Initialize a population of chromosomes λ and μ, (Lagrangian multipliers)
Step 3: While (generation_number < max_generation or the duality gap is greater than a predetermined threshold) do {decode every chromosome to obtain normalized $\lambda^h_{\text{decode}}$ and μ^h_{decode};
translate normalized $\lambda^h_{\text{decode}}$ and μ^h_{decode} to actual λ^h and μ^h;
Step 4: Solve the constrained minimum of Lagrangian function L for each unit to obtain P_{ih} and X_{ih} for $h = 1 \cdots T$; $i = 1 \cdots N$; using two-state dynamic programming;
Step 5: Calculate the dual value using obtained P_{ih} and X_{ih}

$$q(\lambda, \mu) = \sum_{h=1}^{H} \sum_{i=1}^{N} \left[F_i(P_{ih}) + ST_i \left(1 - X_{i(h-1)}\right) \right] X_{ih}$$

$$+\sum_{h=1}^{H}\lambda^h\left(D_h-\sum_{i=1}^{n}P_{ih}\right)$$
$$+\sum_{h=1}^{H}\mu^h\left(D_h+R_h-\sum_{i=1}^{N}P_{i(\max)}X_{ih}\right)$$

Step 6: Using X_{ih} to solve economic dispatch to obtain P^*_{ih}; Calculate the primal value

$$J^*=\sum_{h=1}^{H}\sum_{i=1}^{N}\left[F_i(P^*_{ih})+ST_i\left(1-X_{i(h-1)}\right)\right]X_{ih});$$

Step 7: calculate the relative duality gap

$$\varepsilon=\frac{J^*-q}{q};$$

Step 8: calculate Fitness function (FIT)

$$\text{FIT}=\frac{1}{1+K\bullet\left(\dfrac{F_{\max}}{F_r}-1\right)}$$

Step 9: Rank chromosomes according to their FIT;
Step 10: Select fittest parents for reproduction;
Step 11: Apply crossover & mutation to obtain new chromosomes in order to maximize Lagrangian function;
Step 12: generation_number:= generation_number +1;}
end while
Step 13: print out the final solution;

Some explanations regarding the LRGA method are given in the following.

Explanation 1: A chromosome in the LRGA method corresponds to an encoded normalized Lagrangian multipliers matrix, shown in Fig. 2

The advantage of using Lagrangian multipliers matrix instead of unit on/off state as the encoded parameter is that the number of bits of chromosome will be entirely independent of the number of units. The more encoding bits there are, the higher the resolution and the slower the convergence. In this paper, we use 12 bits to represent Lagrangian multipliers. The decoding of the encoded normalized Lagrangian multipliers can be expressed as:

$$\lambda^h_{\text{decode}}=\sum_{j=1}^{12}(\lambda^h_j\times 2^{-j}),\qquad \lambda^h_j\in\{0,1\}$$
$$\mu^h_{\text{decode}}=\sum_{j=1}^{12}(\mu^h_j\times 2^{-j}),\qquad \mu^h_j\in\{0,1\}$$

Normalized Lagrangian multipliers	λ^1	λ_1^1	λ_2^1	λ_3^1	..	..	λ_{10}^1	λ_{11}^1	λ_{12}^1
	:	:	:	:	..	..	:	:	:
	λ^H	λ_1^H	λ_2^H	λ_3^H	..	..	λ_{10}^H	λ_{11}^H	λ_{12}^H
	μ^1	μ_1^1	μ_2^1	μ_3^1	..	..	μ_{10}^1	μ_{11}^1	μ_{12}^1
	:	:	:	:	..	..	:	:	:
	μ^H	μ_1^H	μ_2^H	μ_3^H	..	..	μ_{10}^H	μ_{11}^H	μ_{12}^H

Fig. 2. Matrix form of a chromosome.

Translating normalized $\lambda^h_{\text{decode}}$ and μ^h_{decode} to actual Lagrangian multipliers λ^h and μ^h using

$$\lambda^h=\lambda_{\min}+\lambda^h_{\text{decode}}(\lambda_{\max}-\lambda_{\min});$$
$$\mu^h=\mu_{\min}+\mu^h_{\text{decode}}(\mu_{\max}-\mu_{\min});$$

where $\lambda_{\min}$, $\lambda_{\max}$, $\mu_{\min}$ and $\mu_{\max}$ are the maximum and minimum values of Lagrangian multipliers.

For example, assume that an encoded normalized Lagrangian multiplier is

$$[\lambda^h_1,\lambda^h_2,\lambda^h_3,\lambda^h_4,\lambda^h_5,\lambda^h_6,\lambda^h_7,\lambda^h_8,\lambda^h_9,\lambda^h_{10},\lambda^h_{11},\lambda^h_{12}]$$
$$=[1,0,0,1,1,0,0,1,0,0,1,1],$$
$$\lambda_{\max}=60\quad\text{and}\quad\lambda_{\min}=20.$$

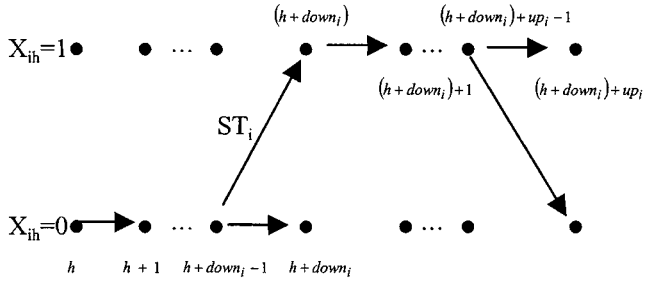
Then the decoding of the encoded normalized Lagrangian multiplier is calculated as

$$\lambda^h_{\text{decode}}=\sum_{j=1}^{12}(\lambda^h_j\times 2^{-j})=0.598$$

The actual Lagrangian multiplier is calculated as

$$\lambda^h=\lambda_{\min}+\lambda^h_{\text{decode}}(\lambda_{\max}-\lambda_{\min})$$
$$=20+0.598(60-20)$$
$$=43.92$$

Explanation 2: The minimum of Lagrangian function $L(X, P, \lambda, \mu)$ is easily found by solving as a two-state dynamic programming problem in two variables for each unit as was done for the forward dynamic-programming solution of the unit commitment problem itself. The scheme of two-state dynamic programming can be visualized in the figure below, which shows the only two possible states (i.e. $X_{ih}=0$ or 1) and subjects to the minimum up/down time for unit i.



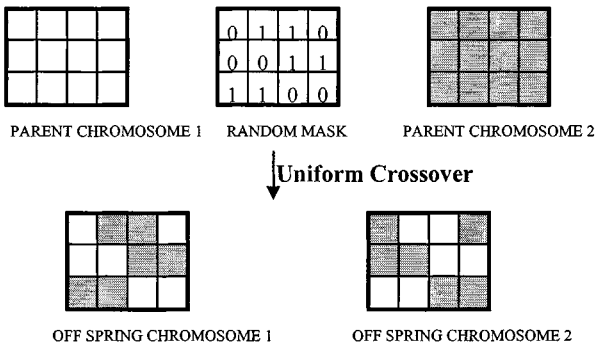


Fig. 3. Uniform crossover operator.

where ST_i is the start-up cost for unit i. The solution to

$$\min \sum_{h=1}^{H} \left[F_i(P_{ih}) + ST_i\left(1 - X_{i(h-1)}\right) - \lambda^h P_{ih} - \mu^h P_{i(\max)}\right] X_{ih}$$

for unit i

means finding the path with the least cost from the initial time $(h = 1)$ to the last time $(h = T)$ and subjecting to 1 constraints (4),(5) and (6).

Explanation 3: Since LRGA method uses the relative duality gap as its basis of the converging rule, the fitness function of the chromosome is expressed as follows:

$$\text{FIT} = \frac{1}{1 + K \bullet \left(\dfrac{F_{\max}}{F_r} - 1\right)}$$

where K: a scaling coefficient,

$$F_r\colon = \frac{1}{\text{the relative duality gap}} = \frac{1}{\varepsilon}$$

$F_{\max}$: the maximum of F_r within the population.

Explanation 4: There are three genetic operators in the LRGA, namely, reproduction, crossover, and mutation. The reproduction operator is a prime selection operator that an old chromosome is copies into a "mating pool" according to the "roulette wheel parent selection" technique[21],[27]. In this technique, the chromosome with larger fitness value have a higher probability to be selected.

The crossover operator recombines the extremely important features of two chromosome to make the offspring chromosome not only inherit some important characteristics from their parent chromosomes but also have the chance to get closer to the optimal solution. In the LRGA, we adopt a new crossover technique known as "uniform crossover" which exchanges bits between the parent chromosome to create two new offspring chromosomes by a randomly generated mask. The scheme of "uniform crossover" is shown in Fig. 3. In the random mask, the "1" represents bit swapping and "0" denotes bit unchanged.

The mutation operator allows us to create new chromosome in the population and provides background variation depending on a mutation probability. The scheme of mutation operator is shown in Fig. 4.

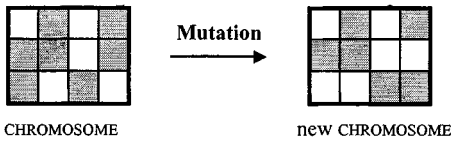


Fig. 4. Mutation operator.

Explanation 5: The "roulette wheel parent selection" technique is used to select the "best" parent chromosomes according to their fitness. It consists of the following steps:

Step 1: Sum the fitness of all chromosomes in the population; call it the FITSUM.

Step 2: Generate a random number, R, between 0 and FITSUM.

Step 3: Return the first chromosome whose fitness, added to the fitness of preceding chromosomes, is greater than or equal to R.

IV. Analysis of the LRGA Method

The Lagrangian Relaxation (LR) method for solving UC problems involves two optimization processes; one for solving the individual subproblems and the other for adjusting the values of the Lagrangian multipliers. Even though the optimum solution to the individual unit subproblems can easily be found, global optimum of overall original problem is not guaranteed as a consuquence of the non convexity of the primal function. The dual problem always has a lower dimension than the primal problem. The difference in value between the primal and dual function yields the duality gap which provides a measure of the suboptimality of the solution. Most of the LR research has therefore concentrated on finding an appropriate technique for updating the Lagrangian multipliers, while minimizing the duality gap. Most of the studies update the Lagrangian multipliers using sub-gradient algorithm. In [29] the Lagrangian multipliers are updated using a sub-gradient algorithm and the Dynamic Programming method is used for solving the individual unit subproblems. The use of sub-gradient algorithm to update these multipliers can result in infeasible spinning reserve solution [13]. Another difficulty with LR based methods is that computational performance is very dependent on the method by updating the Lagrangian multipliers.

The Lagrangian Relaxation and Genetic Algorithms (LRGA) method incorporates Genetic Algorithms (GA's) into Lagrangian Relaxation (LR) method to update the Lagrangian multipliers and improve the performance of LR method. The GA's combine an artificial survival of the fittest with genetic operators abstracted from nature to form a surprisingly robust mechanism that is suitable for a variety of optimization problems. One of the advantages of GA's is using stochastic operators instead of deterministic rules to search a solution. By simulating "the survival of the fittest" of Darwiniam evolution among chromosome structures, the optimal chromosome (solution) is searched by randomized information exchange, thus allowing it to escape from local optimum in which other algorithm might land.

TABLE I
COMPARISON OF SEARCH PERFORMANCE BETWEEN LRGA AND LR

iteration	method	Duality gap	Dual cost($)	Primal cost($)
1	LRGA	0.373	15404	21152
	LR	∞	0	40000
2	LRGA	0.084	18973	20567
	LR	1.67	14982	40000
3	LRGA	0.054	19521	20567
	LR	0.965	18344	36024
4	LRGA	0.026	19654	20162
	LR	0.502	19214	28906
5	LRGA	0.020	19759	20162
	LR	0.844	19532	36024
6	LRGA	0.012	19919	20162
	LR	0.037	19442	20170

The three prime operators associated with the GA's are reproduction, crossover and mutation. Another attractive property of GA's is that it searches for many optimum chromosomes in parallel. The advantage of using normalized Lagrangian multipliers instead of units on/off state as the encoded parameter is that the number of bits of chromosome will be entirely independent of number of units and only dependent of number of hours.

V. NUMERICAL SIMULATIONS

In this section, two cases are studied to illustrate the effectiveness of the proposed LRGA method in terms of its solution quality. The first case compares LRGA with Lagrangian Relaxation (LR) in terms of duality gap. The second case compares LRGA with LR and Genetic Algorithms (GA's) in terms of production cost. The LRGA program is coded in Turbo C and implemented on a compatible personal computer (PC 486DX2-66). In order to avoid misleading results due to the stochastic nature of the LRGA, 20 runs were averaged for each case, with each run starting with random initial populations.

Case 1: 3-unit system by LRGA and LR: In this case, a simple three-generator, four-hour unit commitment schedule determined by the proposed LRGA method is compared to-that determined by the Lagrangian Relaxation(LR)[1]. The system unit data and the load demands are shown in Appendix I. The control parameter settings of LRGA are population size 60, probability of crossover 0.8, probability of mutation 0.1 and duality gap 0.02 . The performance of the LRGA compared with that of LR is given in Table I. The CPU times of LRGA and LR are about l-second for the case. From this result, it is shown that the performance by LRGA is better than that of LR in terms of duality gap and primal cost.

Case 2: Up to 100-Unit Systems by LRGA, LR and GA's: In this case, the up to 100-generator, 24-hour unit commitment schedule determined by the proposed LRGA method is compared to that determined by the Lagrangian Relaxation (LR) and Genetic Algorithms (GA's)[28]. The system unit data and the load demands are shown in Appendix II. The control parameter settings of LRGA are probability of crossover 0.8, probability of mutation 0.0333 and duality gap 0.02. The performance of the LRGA compared with that of LR and GA's is given in Table II. From this result, it is shown that the performance by LRGA is better than that of LR and GA's in terms of production cost. Since the CPU times are not reported in reference [28] and no knowledge of what kind of computer hardware is used in [28], CPU times are not comparable. We show CPU times of LRGA only. It is observed that CPU times of LRGA increase with the number of units. However with the progress in the hardware of parallel computing, the speed of LRGA can be greatly improved by means of parallel processing of GA's.

TABLE II
COMPARISON OF FINAL PRODUCTION COST FOR UP TO 100-UNIT SYSTEMS AMONG LRGA, LR AND GA 'S

unit	LR	GAs	LRGA				
	Production cost($)			Population size	Generation	Duality gap	CPU time(sec)
10	565825	565825	564800	60	500	0.0392	518
20	1130660	1126243	1122622	80	500	0.0386	1147
40	2258503	2251911	2242178	80	500	0.0349	2165
60	3394066	3376625	3371079	60	500	0.0371	2414
80	4526022	4504933	4501844	60	500	0.0483	3383
100	5657277	5627437	5613127	60	500	0.0418	4045

VI. CONCLUSION

The paper solves the unit commitment (UC) problem by the Lagrangian Relaxation and Genetic Algorithms (LRGA) which incorporated the Genetic Algorithms (GA's) into the Lagrangian Relaxation (LR) method to update the Lagrangian multipliers and improve the performance of LR method. The advantage of using normalized Lagrangian multipliers instead of units' on/off state as the encoded parameters is that the number of bits of chromosome will be entirely independent of number of units and only dependent of number of hours. This is particularly attractive in large-scale systems. The numerical tests and results show that better solution of the unit commitment (UC) problem can be obtained by the LRGA method than LR and GA's.

APPENDIX I
UNIT DATA AND LOAD DEMAND FOR CASE 1

UNIT DATA FOR 3-UNIT SYSTEM

Unit	Pmax	Pmin	a	b	c
1	600	100	500	10	0.0020
2	400	100	300	8	0.0025
3	200	50	100	6	0.0050

LOAD DEMAND FOR 4-HOUR

Time (hour)	P_{load} (MW)
1	170
2	520
3	1100
4	330

APPENDIX II
UNIT DATA AND LOAD DEMANDS FOR CASE 2

LOAD DEMAND FOR 24-HOUR

Hour	P_{load}	Hour	P_{load}
1	700	13	1400
2	750	14	1300
3	850	15	1200
4	950	16	1050
5	1000	17	1000
6	1100	18	1100
7	1150	19	1200
8	1200	20	1400
9	1300	21	1300
10	1400	22	1100
11	1450	23	900
12	1500	24	800

The problem data were scaled appropriately for the problems with more units. The reserve was assumed to be 10% of the demand.

DATA FOR THE 10-UNIT SYSTEM

	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
Pmax (MW)	455	455	130	130	162
Pmin (MW)	150	150	20	20	25
a ($/h)	1000	970	700	680	450
b ($/MWh)	16.19	17.26	16.60	16.50	19.70
c ($/MW2-h)	0.00048	0.00031	0.002	0.00211	0.00398
min up (h)	8	8	5	5	6
min dn (h)	8	8	5	5	6
hot start cost($)	4500	5000	550	560	900
cold start cost($)	9000	10000	1100	1120	1800
cold start hrs (h)	5	5	4	4	4
initial status (h)	8	8	-5	-5	-6

	Unit 6	Unit 7	Unit 8	Unit 9	Unit 10
Pmax (MW)	80	85	55	55	55
Pmin (MW)	20	25	10	10	10
a ($/h)	370	480	660	665	670
b ($/MWh)	22.26	27.74	25.92	27.27	27.79
c ($/MW2-h)	0.00712	0.0079	0.00413	0.00222	0.00173
min up (h)	3	3	1	1	1
min dn (h)	3	3	1	1	1
hot start cost($)	170	260	30	30	30
cold start cost($)	340	520	60	60	60
cold start hrs (h)	2	2	0	0	0
initial status (h)	-3	-3	-1	-1	-1

REFERENCES

[1] A. J. Wood and B. F. Wollenberg, *Power Generation, Operation & Control*: John Wiley & Sons, Inc., 1996.

[2] R. M. Burns and C. A. Gibson, "Optimization of Priority Lists for A Unit commitment Program," in *IEEE/PES 1975 Summer Meeting*, Paper A 75 453-1.

[3] G. B. Sheble, "Solution of the Unit Commitment Problem by the Method of Unit Periods," *IEEE Trans. on Power Systems*, vol. 5, no. 1, pp. 257–260, 1990.

[4] T. S. Dillon and K. W. Edwin, "Integer Programming Approach to The Problem of Optimal Unit Commitment with Probabilistic Reserve Determination," *IEEE Trans. on Power Systems*, vol. PAS-97, no. 6, pp. 2154–2166, 1978.

[5] L. L. Garver, "Power Generation Scheduling by Integer Programming - Development of Theory," *IEEE Trans. on Power Systems*, vol. 102, pp. 730–735, 1963.

[6] W. L. Snyder Jr., H. D. Powell Jr., and J. C. Rayburn, "Dynamic Programming Approach to Unit Commitment," *IEEE Trans. on Power Systems*, vol. 2, pp. 339–350, 1987.

[7] P. G. Lowery, "Generation Unit Commitment by Dynamic Programming," *IEEE Trans. on Power Systems*, vol. 102, pp. 1218–1225, 1983.

[8] C. K. Pang and H. C. Chen, "Optimal Short-Temm thermal Unit Commitment," *IEEE Trans. on Power Systems*, vol. 95, no. 4, pp. 1336–1346, 1976.

[9] C. K. Pang, G. B. Sheble, and F. Albuyeh, "Evalution of Dynamic Programming Based Methods and Multiple Area Representation for Thermal Unit Commitments," *IEEE Trans. on Power Systems*, vol. PAS-100, no. 3, pp. 1212–1218, 1981.

[10] C. C. Su and Y. Y. Hsu, "Fuzzy Dynamic Programming: An Application to Unit Commitment," *IEEE Trans. on Power Systems*, vol. 6, no. 3, pp. 1231–1237, 1991.

[11] Z. Ouyang and S. M. Shahidehpour, "An Intelligent Dynamic Programming for Unit Commitment Application," *IEEE Trans. on Power Systems*, vol. 6, no. 3, pp. 1203–1209, 1991.

[12] A. Merlin and P. Sandrin, "A new Method for Unit Commitment at Electricite De France," *IEEE Trans. on Power Systems*, vol. 102, pp. 1218–1225, 1983.

[13] F. Zhuang and F. D. Galiana, "Toward a more Rigorous and Practical Unit Commitment by Lagrangian Relaxation," *IEEE Trans. on Power Systems*, vol. 3, no. 2, pp. 763–772, 1988.

[14] J. A. Muckstadt and R. C. Wilson, "An Application of Mixed-lnteger Programming Duality to Scheduling Thermal Generating Systems," *IEEE Trans. on Power Systems*, pp. 1968–1978, 1968.

[15] A. I. Cohen and M. Yoshimura, "A Branch-and-Bound Algorithm for Unit Commitment," *IEEE Trans. on Power Systems*, vol. PAS-102, no. 2, pp. 444–451, 1983.

[16] Z. Ouyang and S. M. Shahidehpour, "A Hybrid Artificial Neural Network/Dynamic Programming Approach to Unit Commitment," *IEEE Trans. on Power Systems*, vol. 7, no. 1, pp. 236–242, 1992.

[17] H. Sasaki, M. Watabable, J. Kubokawa, N. Yorino, and R. Yokoyama, "A Solution Method of Unit Commitment by Artificial Neural Networks," *IEEE Trans. on Power Systems*, vol. 7, no. 1, pp. 974–985, 1992.

[18] F. Zhuang and F. D. Galiana, "Unit Commitment by Simulated Annealing," *IEEE Trans. on Power Systems*, vol. 5, no. 1, pp. 311–317, 1990.

[19] D. Dasgupta and D. R. McGregor, "Themmal Unit Commitment using Genetic Algorithms," *IEE Proc. C, Gener. Transm. Distrib.*, vol. 141, no. 5, pp. 459–465, 1994.

[20] C. L. Huang, J. S. Tzeng, P. C. Yang, and H. T. Yang, "Implementation of Genetic Algorithm for Unit commitment," in *1993 Proceedings of The 14th Symposium on Electrical Power Engineering*, Taiwan, R.O.C., pp. 439–446.

[21] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*: Addison Wesley, 1989.

[22] F. T. Lin, C. Y. Kao, and C. C. Hsu, "Applying the Genetic approach to Simulated Annealing in Solving Some NP-Hard Problems," *IEEE Trans. on Systems, Man, And Chyernetics*, vol. 23, no. 6, pp. 1752–1767, 1993.

[23] G. B. Sheble and T. T. Maifeld, "Unit Commitment by Genetic Algorithm and Expert System," *Electric Power System Research 30*, pp. 115–121, 1994.

[24] D. C. Walters and G. B. Sheble, "Genetic Algorithm solution of Economic Dispatch with valve point loading," *IEEE Trans. on Power Systems*, vol. 8, no. 3, pp. 1325–1332, 1993.

[25] H. C. Weng and C. Y. Kao, "A Genetic Algorithm Approach for the Bounded Integer Programming Problem," Master Dissertation, National Taiwan Univ., 1993.

[26] C. L. Lin and C. Y. Kao, "A Genetic Algorithm Approach for the Sequencing Problems:the Traveling Salesman Problem and Scheduling Problem," Master Dissertation, National Taiwan Univ., 1993.

[27] P.-H. Chen and H.-C. Chang, "Large-scale economic dispatch by Genetic Algorithm," *IEEE Trans. on Power Systems*, vol. 10, no. 4, pp. 1919–1925, Nov. 1995.

[28] S. A. Kazarlis, A. G. Bakirtzis, and V. Petridis, "A Genetic Algorithm solution to the unit commitment problem," *IEEE Trans. on Power Systems*, vol. 11, no. 1, pp. 83–92, Feb. 1996.
[29] A.I. Cohen and S.H. Wan, "A method for solving fuel constrained unit commitment problem," *IEEE Trans. on Power System*, pp. 608–614, 1987.

Chuan-Ping Cheng was born in Taiwan in 1959. He graduated from National Taipei Institute of Technology in Electrical Engineering in 1980. He is currently working toward his Ph.D. degree at National Taiwan University. His research interests include power system analysis and optimal theory.

Chih-Wen Liu was born in Taiwan in 1964. He received the B.S. degree in electrical engineering from National Taiwan University in 1987, Ph.D. degree in electrical engineering from Cornell University in 1994. Since 1994, he has been with National Taiwan University, where he is associate professor of electrical engineering. His research interests include application of numerical methods to power system and motor control.

Chun-Chang Liu was born in Taiwan in 1925. He received his B.S. degree in electrical engineering from National Taiwan University in 1951. Since 1951, he has been with National Taiwan University, where he is professor of electrical engineering. From 1987 to 1988, he was a visiting professor at the Kyushu University in lapan. His current research interests include power system analysis and electrical machine analysis.