

http://dx.doi.org/10.7236/IIBC.2016.16.1.253

IIBC 2016-1-35

## 밸브지점 균형과 교환 최적화 방법을 적용한 동적경제급전문제

### Dynamic Economic Load Dispatch Problem Applying Valve-Point Balance and Swap Optimization Method

이상운\*

Sang-Un, Lee \*

**요약** 본 논문은 경제급전 최적화 문제에 균형-교환 방법을 제안하였다. 제안된 알고리즘은 모든 발전기를 가능한 한 밸브지점으로 운영한다고 가정한다. 초기치로 최대 발전량  $P_i \leftarrow P_i^{\max}$ 로 설정하고, 각 발전기의 밸브지점  $v_k$ 까지 발전량을 감소시켰을 때의 평균 발전단가  $c_i = \frac{F(P_i) - F(P_{iv_k})}{(P_i - P_{iv_k})}$ 가 최대가 되는  $\max c_i$  발전기  $i$ 의 발전량을 밸브지점 발전단가  $P_{iv_k}$ 로 감소시켰으며,  $\Sigma P_i - P_d > 0$ 이면  $c_i = F(P_i) - F(P_i - 1)$ 의  $\max c_i$  발전기 발전량을  $P_i \leftarrow P_i - 1$ 로 감소시켜  $\Sigma P_i = P_d$ 의 균형을 맞추었다. 다음으로,  $\min \{ \max (P_i - P_i^{\min}), \max (P_i^{\max} - P_i) \} > \alpha \geq 10$ 의 범위에 대해 "10" 간격으로 감소시키는 성인걸음법으로,  $10 > \alpha \geq 1$  범위에 대해서는 "1"의 아기걸음법으로,  $P_i = P_i \pm \alpha$ 에 대한  $\max [F(P_i) - F(P_i - \alpha)] > \min [F(P_j + \alpha) - F(P_j)], i \neq j$ 이면  $P_i = P_i - \alpha, P_j = P_j + \alpha$ 로 발전량을 교환하는 방법으로 최적화를 수행하였다. 다음으로  $\alpha = 0.1, 0.01, 0.001, 0.0001$ 에 대해 미세한 교환을 수행하였다. 동적 경제급전 문제의 시험 사례에 제안된 알고리즘을 적용한 결과 기존의 휴리스틱 알고리즘 최적화 발전비용을 크게 감소시켜 경제적인 이익을 극대화 시켰다.

**Abstract** This paper proposes a balance-swap method for the dynamic economic load dispatch problem. Based on the premise that all generators shall be operated at valve-points, the proposed algorithm initially sets the maximum generation power at  $P_i \leftarrow P_i^{\max}$ . As for generator  $i$  with  $\max c_i$ , which is the maximum operating cost  $c_i = \frac{F(P_i) - F(P_{iv_k})}{(P_i - P_{iv_k})}$  produced when the generation power of each generator is reduced to the valve-point  $v_k$ , the algorithm reduces  $i$ 's generation power down to  $P_{iv_k}$ , the valve-point operating cost. When  $\Sigma P_i - P_d > 0$ , it reduces the generation power of a generator with  $\max c_i$  of  $c_i = F(P_i) - F(P_i - 1)$  to  $P_i \leftarrow P_i - 1$  so as to restore the equilibrium  $\Sigma P_i = P_d$ . The algorithm subsequently optimizes by employing an adult-step method in which power in the range of  $\min \{ \max (P_i - P_i^{\min}), \max (P_i^{\max} - P_i) \} > \alpha \geq 10$  is reduced by 10; a baby step method in which power in the range of  $10 > \alpha \geq 1$  is reduced by 1; and a swap method for  $\max [F(P_i) - F(P_i - \alpha)] > \min [F(P_j + \alpha) - F(P_j)], i \neq j$  of  $P_i = P_i \pm \alpha$ , in which power is swapped to  $P_i = P_i - \alpha, P_j = P_j + \alpha$ . It finally executes minute swap process for  $\alpha = 0.1, 0.01, 0.001, 0.0001$ . When applied to various experimental cases of the dynamic economic load dispatch problems, the proposed algorithm has proved to maximize economic benefits by significantly reducing the optimal operating cost of the extant Heuristic algorithm.

**Key Words** : Dynamic economic dispatch, Optimization, Valve-point, Adult-step, Baby-step

\*정회원, 강릉원주대학교 과학기술대학 멀티미디어공학과  
접수일자 : 2015년 8월 28일, 수정완료 : 2016년 1월 9일  
게재확정일자 : 2016년 2월 5일

Received: 28 August, 2015 / Revised: 9 January, 2016 /  
Accepted: 5 February, 2016

\*Corresponding Author: sulee@gwnu.ac.kr  
Dept. of Multimedia Eng., Gangneung-Wonju National University,  
Korea

## I. 서 론

전기에너지는 기본적으로 저장이 불가능하므로 모든 시간대별로 총 발전량은 손실을 포함한 총 부하량과 동일 또는 여유분을 갖도록 준비하여야 한다. 만약, 발전량이 부하량보다 적을 경우 주파수 난조 등 전체 계통에 큰 문제가 발생한다.

주어진 시간대의 부하를 공급할 수 있도록 발전비용이 각각 다른  $n$ 대의 발전기 출력을 각각 어떻게 배분해야 전체 발전비용(생산비용)이 가장 경제적인가를 결정하는 과정을 경제적 부하배분(economic load dispatch, ELD) 또는 간단히 경제급전(economic dispatch, ED)이라 한다. 전통적인 경제급전 문제는 단위시간에 대한 경제급전에 한정된 연구이며, 이를 정적경제급전(static ED, SED)이라 한다.<sup>[1]</sup> 반면에, 동적 경제급전(dynamic ED, DED)은 24시간 범위에 대해 시간대별로 정적경제급전문제를 확대한 경우로, 실제 운영시 보다 현실적인 발전계획에 적용된다.<sup>[2]</sup>

$n$ 개의 발전기가 전력을 공급하기 위해서는 각기 다른 발전량  $P_i^{\min} \leq P_i \leq P_i^{\max}$ 로 가동되어야 하며, 발전비용 함수는 실제로는 밸브 효과 (valve-effect)로 인해  $a_i P_i^2 + b_i P_i + c_i + |e_i \times \sin(f_i \times (P_i^{\min} - P_i))|$  형태를 나타내나 현실적으로는 밸브효과가 없는  $F(P_i) = a_i P_i^2 + b_i P_i + c_i$  함수를 적용한다.<sup>[2]</sup> 이는 함수의 기울기를 적용한 라그랑주 완화법(Lagrangian relaxation)으로 해를 구할 수 있기 때문이다.<sup>[3]</sup> 즉, 각기 다른 발전비용을 가진  $n$ 개의 발전기를 가동하여 요구량  $P_d$ 를 만족시키기 위해 각 발전기의 발전량  $P_i$ 를 얼마로 조절하였을 때 총 발전비용  $\Sigma F(P_i)$ 를 최소화시켜 가장 경제적으로 발전할 수 있는가의 최적화 문제가 경제급전이다.

경제급전 최적화와 관련된 기존 알고리즘들은 대부분 수학적 최적화 기법인 라그랑주 완화법, 경사법, 감소된 경사법이나 인공지능 기법인 유전자 알고리즘, 신경망, 퍼지, 군집 알고리즘 등 비결정적이며, 휴리스틱 알고리즘으로 초기치를 랜덤하게 설정하고 수십 회 이상 수행한 결과 최소치를 가진 결과로 결정하여 동일한 해를 검증하기가 사실상 불가능하다. 또한, 다양한 휴리스틱 알고리즘을 적용함에도 불구하고, 대부분의 결과들은 거의 비슷한 결과를 얻고 있으며, 획기적으로 해를 개선한 알고리즘을 찾기가 현실적으로 불가능하다.

본 논문에서는 알고리즘이 간단하면서도 기존 알고리

즘들에 비해 해를 획기적으로 개선한 밸브지점 균형-교환 최적화(valve-point balance and swap optimization, VPBSO) 알고리즘을 제안한다. 2장에서는 동적 경제급전과 관련된 연구와 문제점을 고찰해 본다. 3장에서는 간단하면서도 획기적으로 해를 개선할 수 있는 알고리즘을 제안한다. 4장에서는 제안된 알고리즘을 적용하여 성능을 검증한다.

## II. 관련연구와 문제점

주어진 시간대에서 수급 균형을 맞추기 위해 총 발전량  $\Sigma P_i$ 는 총 부하량  $P_d$ 과 동일해야 한다. 이는 식 (1)과 같다.

$$\Sigma P_i = P_d \quad (1)$$

여기서 총 부하량  $P_d$ 로 총 요구량이 되며, 엄밀히 말해 수요 예측량과 공급 예비율의 합이다. 총 부하량을 수요량이라 하자.

경제급전 대상 발전소의 발전기  $n$ 개가 존재하며, 각각의 발전비용이 차이가 발생한다고 가정한다. 이 경우 수요량이  $\Sigma P_i^{\min} \leq P_d \leq \Sigma P_i^{\max}$ 를 충족시키기 위한 경제급전 문제의 목적함수는 경제급전 총 발전비용을 최소화시키는 것으로 식 (2)와 같다.<sup>[1,2]</sup>

$$\begin{aligned} \text{minimize } \Sigma F(P_i) &= \sum_{i=1}^n F(P_i) \\ \text{subject to } \Sigma P_i &= P_d \\ P_i^{\min} &\leq P_i \leq P_i^{\max} \end{aligned} \quad (2)$$

여기서  $F(P_i)$ 는 발전기  $i$ 의 발전비용(\$/MWh)으로 식 (3)을 적용한다. 왜냐하면 발전기를 가동하여 일정 출력에 도달하여 출력이 안정화될 때까지는 전력을 공급하지 않는 무부하 운전을 수행한다. 이 경우의 발전비용 함수는 식 (4)와 같다. 일단 발전기 출력이 안정화 된 이후에 전력을 공급하는 최초 시점이 최소 발전량  $P_i^{\min}$ 이며, 이후 발전 용량을 증가시키기 위해 밸브를 열면 진동 발생으로 인한 밸브효과로 발전비용  $F(P_i)$ 는 식 (3)과 같이 복잡한 비평활함수(non-smooth function) 형태를 나타낸다.<sup>[1,2]</sup>

$$F(P_i) = a_i P_i^2 + b_i P_i + c_i + |e_i \times \sin(f_i \times (P_i^{\min} - P_i))| \quad (3)$$

$$F(P_i) = a_i P_i^2 + b_i P_i + c_i \quad (4)$$

여기서  $P_i$ 는 발전기  $i$ 의 발전량 (KW)이며, 각 발전기의 출력 범위 제약조건은  $P_i^{\min} \leq P_i \leq P_i^{\max}$  이다.

표 1은 Attaviryanupap et al.<sup>[3]</sup>가 제시한 10-발전기의 동적 경제급전 예제 데이터이다. 표 1의 부하량 (수요량)을 각 시간대별로 표현하면 그림 1과 같다. 12시와 20시에 가장 많은 발전을 해야 하며, 1시에 가장 적은 발전량을 보이고 있다. 이는 전형적인 1일 발전량의 변동 곡선이다.

표 1. 실험 데이터  
 Table 1. Case Study Data

Data for the 10-Unit System							
$G_i$	$P_i^{\min}$ (MW/h)	$P_i^{\max}$ (MW/h)	$a$	$b$	$c$	$e$	$f$
1	150	470	0.00043	21.60	958.20	450	0.041
2	135	460	0.00063	21.05	1,313.60	600	0.036
3	73	340	0.00039	20.81	604.97	320	0.028
4	60	300	0.00070	23.90	471.60	260	0.052
5	73	243	0.00079	21.62	480.29	280	0.063
6	57	160	0.00056	17.87	601.75	310	0.048
7	20	130	0.00211	16.51	502.70	300	0.086
8	47	120	0.00480	23.23	639.40	340	0.082
9	20	80	0.10908	19.58	455.60	270	0.098
10	55	55	0.00951	22.54	692.40	380	0.094
Sum:	600	2,338					

Load Demand for 24 Hours					
Hour	Load (MW)	Hour	Load (MW)	Hour	Load (MW)
1	1,036	9	1,924	17	1,480
2	1,110	10	2,072	18	1,628
3	1,258	11	2,146	19	1,776
4	1,406	12	2,220	20	2,072
5	1,480	13	2,072	21	1,924
6	1,628	14	1,924	22	1,628
7	1,702	15	1,776	23	1,332
8	1,776	16	1,554	24	1,184

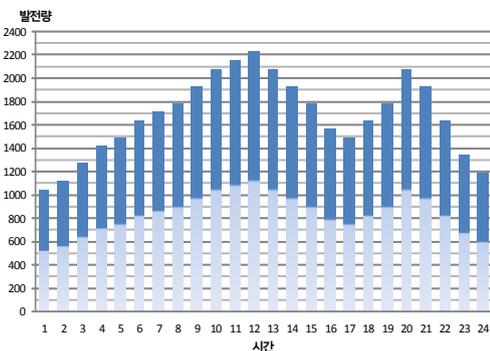


그림 1. 전형적인 시간대별 발전기 가동 현황  
 Fig. 1. Conventional Generator Operation with 24-Hour

표 1의 데이터들에 대해 최적화를 시킨 기존의 연구 결과는 표 2에 제시하였다.<sup>[2-20]</sup> 다양한 휴리스틱 방법으로 최적화를 시키는 연구가 진행되고 있으며, 아직까지는 10대의 발전기로 \$1,018,467.494/1일의 비용이 소요되는 경우가 최적의 경제급전으로 알려져 있다.

표 2. 시험사례의 최적화 결과  
 Table 2. Optimization Results for the Test Cases

Method	Cost(\$)	Iteration
Sequential Quadratic Programming (SQP) <sup>[3]</sup>	1,051,163.000	50,000
Evolutionary Programming (EP) <sup>[3]</sup>	1,048,638.000	50,000
Hybrid EP and SQP (EP-SQP) <sup>[3]</sup>	1,031,746.000	30,000
MDE <sup>[4]</sup>	1,031,612.000	-
HDE <sup>[5]</sup>	1,031,077.000	-
PSO-SQP <sup>[6]</sup>	1,030,773.000	30,000
AIS-SQP <sup>[7]</sup>	1,029,900.000	-
MHEP-SQP <sup>[8]</sup>	1,028,924.000	30,000
DGPSO <sup>[9]</sup>	1,028,835.000	30,000
PSO-SQP(C) <sup>[6]</sup>	1,027,334.000	30,000
IDE <sup>[10]</sup>	1,026,269.000	-
IPSO <sup>[11]</sup>	1,023,807.000	-
EPSO <sup>[12]</sup>	1,023,772.460	20,000
CMAES <sup>[13]</sup>	1,023,740.000	-
EPSO-GM <sup>[2]</sup>	1,023,691.106	20,000
CS-DE <sup>[14]</sup>	1,023,432.000	-
AIS <sup>[15]</sup>	1,021,980.000	-
ABC <sup>[16]</sup>	1,021,576.000	-
DE <sup>[17]</sup>	1,019,786.000	-
CDE <sup>[18]</sup>	1,019,123.000	-
HHS <sup>[19]</sup>	1,019,091.000	-
ICA <sup>[20]</sup>	<b>1,018,467.494</b>	<b>600</b>

비평활함수에 대한 단위시간 기준 경제급전 최적화에 대해서는 Lee<sup>[21]</sup>가, 비평활함수 동적경제급전 문제에 대해서는 Lee<sup>[22]</sup>가 연구하였다. 그러나 대량의 발전기들을 대상으로 한 비평활함수의 24시간 동적경제급전에 대해 밸브지점을 기준으로 최적화시키는 방법에 대해서는 연구가 진행되지 않고 있다.

### III. 밸브지점 균형-교환 최적화 알고리즘

본 장에서는 제안하는 경제급전 최적화 알고리즘은 균형 단계(balance phase)와 교환 최적화 단계(swap optimization phase)를 수행한다. 첫 번째 단계는  $P_i \leftarrow P_i^{\max}$ 에 대해  $\Sigma P_i = P_d$ 로 균형을 맞추는 과정을 수행한다. 여기서 적용된 기법은 다음과 같다.

- (1) 각 발전기  $G_i$ 의  $P_i^{\min} \leq P_i \leq P_i^{\max}$ 에 대해 "1" 간격으로  $[a_i P_i^2 + b_i P_i + c_i + |e_i \times \sin(f_i \times (P_i^{\min} - P_i))| - (a_i P_i^2 + b_i P_i + c_i)]$ 를 계산한 값이 최소가 되는 발전량들을 밸브 지점으로 결정한다.

- (2)  $P_{i1} \leftarrow P_i^{\max}, P_{i2} \leftarrow P_{i v_k}$  로 설정한다. 여기서  $v_k$  는  $k$  번째 밸브 지점이다. 평균 발전단가  $c_i = \frac{F(P_{i1}) - F(P_{i2})}{P_{i1} - P_{i2}}$  를 계산한다.  $\max c_i$  인 발전기  $i$  에 대해  $P_{i1} \leftarrow P_{i2}, P_{i2} \leftarrow P_{i v_{k-1}}$  로 설정한다. 이 과정을  $\Sigma P_i \geq P_d$  일 때 까지 반복 수행한다.
- (3) 위 과정을 수행한 후 만약,  $\Sigma P_i - P_d > 0$  이면  $P_{i2} \leftarrow P_{i1} - 1$  로 설정하여  $c_i = F(P_i) - F(P_i - 1)$  을 계산하고,  $\max c_i$  인 발전기  $i$  에 대해  $P_{i1} \leftarrow P_{i1} - 1, P_{i2} \leftarrow P_{i1} - 1$  로 설정한다. 이 과정을  $\Sigma P_i - P_d = 0$  일 때까지 수행한다.
- (4) 위의 (2)를 수행한 횟수를  $l$  이라 하자. 기본적으로  $l$  을 수행하고 (3)을 수행한 경우,  $l-1$  까지 수행하고, (3)을 수행한 경우,  $l$  을 수행하고  $l+1$  번째의  $\max c_i$  에 대해  $P_{i1} \leftarrow P_{i v_{l+1}}, P_{i2} \leftarrow P_{i1}$  로 다음 상위 밸브 지점으로 발전량을 증가시키고 (3)을 수행한 경우 3가지 경우를 수행하여  $\min \Sigma F(P_i)$  인 경우를 선택한다.

두 번째 단계는  $P_i \pm \alpha$  의 발전비용에 대해  $\max [F(P_i) - F(P_i - \alpha)] > \min [F(P_j + \alpha) - F(P_j)], i \neq j$  이면  $P_i \leftarrow P_i - \alpha, P_j \leftarrow P_j + \alpha$  로 발전량을 교환하는 방법으로 해를 개선한다.

만약,  $\max_1 [F(P_i) - F(P_i - \alpha)] > \min_1 [F(P_j + \alpha) - F(P_j)], i = j$  이면 2번째의  $\max_2 [F(P_i) - F(P_i - \alpha)]$  와 2번째의  $\min_2 [F(P_j + \alpha) - F(P_j)]$  를 선택하여  $\max \{ \max_1 [F(P_i) - F(P_i - \alpha)] > \min_2 [F(P_j + \alpha) - F(P_j)], \max_2 [F(P_i) - F(P_i - \alpha)] > \min_1 [F(P_j + \alpha) - F(P_j)] \}$  로 교환을 수행한다.

이 방법은 다음과 같이 2가지 방법으로 수행한다.

- (1)  $H \geq \alpha \geq 10$  의 교환 범위에 대해 -10 간격의 내림 차순으로  $\alpha$  를 적용한다. 여기서  $H = \min \{ \max (P_i - P_i^{\min}), \max (P_i^{\max} - P_i) \}$  값에서 "1" 자리수를 절사한 값이다. 이를 성인걸음 (adult-step) 이라 하자.  $9 \geq \alpha \geq 1$  범위에서는 "-1" 간격으로  $\alpha$  를 적용한다. 이를 아기걸음 (baby-step) 이라 하자. 즉, 이 방법은 성인걸음-아기걸음 교환법이다.
- (2)  $H \geq \alpha \geq 1$  에 대해 바로  $\alpha = 1$  을 적용한다. 이를 거인걸음 (giant-step) 이라 하자. 즉, 이 방법은 거인걸음 교환법이다.

성인걸음-아기걸음 교환법과 거인걸음 교환법 중에서  $\min \Sigma F(P_i)$  을 선택한다. 마지막으로  $\alpha = 0.1 \rightarrow 0.01 \rightarrow 0.001 \rightarrow 0.0001$  에 대해 교환을 수행하여 최적화 값을 미세하게 조정한다.

### IV. 적용 결과 및 분석

$P_i^{\min} \leq P_i \leq P_i^{\max}$  에 대해 발전량을 1씩 증가시키면  $[a_i P_i^2 + b_i P_i + c_i + |e_i \times \sin(f_i \times (P_i^{\min} - P_i))]| - [a_i P_i^2 + b_i P_i + c_i]$  가 최소가 되는 발전량을 구하면 밸브 지점을 구한 결과는 표 3과 같다.

표 3. 10 발전기의 밸브지점  
Table 3. Valve-points for 10 Generators

발전기	$P_i^{\max}$	$v_5$	$v_4$	$v_3$	$v_2$	$v_1 = P_i^{\min}$
$G_1$	470	456	380	303	227	150
$G_2$	460	-	397	310	222	135
$G_3$	340	-	-	297	185	73
$G_4$	300	-	241	181	120	60
$G_5$	243	-	223	173	123	73
$G_6$	160	-	-	-	122	57
$G_7$	130	-	130	93	57	20
$G_8$	120	-	-	-	85	47
$G_9$	80	-	-	-	52	20
$G_{10}$	55	-	-	-	-	55

각 발전기의 밸브 지점간 평균 발전단가  $c_i$  는 다음과 같다.

$$\begin{aligned}
 G_1 : & 470 \xrightarrow{38.2435} \xrightarrow{456} \xrightarrow{22.0492} \xrightarrow{380} \xrightarrow{21.8647} \xrightarrow{303} \xrightarrow{21.7970} \xrightarrow{227} \xrightarrow{21.8521} \xrightarrow{150} \\
 G_2 : & 460 \xrightarrow{28.7781} \xrightarrow{397} \xrightarrow{21.4293} \xrightarrow{310} \xrightarrow{21.4344} \xrightarrow{222} \xrightarrow{21.3411} \xrightarrow{135} \\
 G_3 : & 340 \xrightarrow{27.8917} \xrightarrow{297} \xrightarrow{21.0140} \xrightarrow{185} \xrightarrow{20.9266} \xrightarrow{73} \\
 G_4 : & 300 \xrightarrow{24.6025} \xrightarrow{241} \xrightarrow{24.2126} \xrightarrow{181} \xrightarrow{24.0562} \xrightarrow{120} \xrightarrow{24.1196} \xrightarrow{60} \\
 G_5 : & 243 \xrightarrow{35.0681} \xrightarrow{223} \xrightarrow{21.9799} \xrightarrow{173} \xrightarrow{21.9009} \xrightarrow{123} \xrightarrow{21.8219} \xrightarrow{73} \\
 G_6 : & 160 \xrightarrow{25.7918} \xrightarrow{122} \xrightarrow{18.0732} \xrightarrow{57} \\
 G_7 : & 130 \xrightarrow{17.2240} \xrightarrow{93} \xrightarrow{16.5331} \xrightarrow{57} \xrightarrow{17.0000} \xrightarrow{20} \\
 G_8 : & 120 \xrightarrow{26.8100} \xrightarrow{85} \xrightarrow{24.0926} \xrightarrow{47} \\
 G_9 : & 80 \xrightarrow{37.7080} \xrightarrow{52} \xrightarrow{27.4809} \xrightarrow{20}
 \end{aligned}$$

$\Sigma P_i^{\min} = 690 < P_d < \Sigma P_i^{\max} = 2,358$  에 대해 출력량을 감소시키는 순서는 다음과 같다.

표 4. VPBSO 알고리즘 수행 결과  
 Table 4. The Results of VPBSO Algorithm

	$P_i^{\min}$	$P_i^{\max}$	Hours											
			01H	02H	03H	04H	05H	06H	07H	08H	09H	10H	11H	12H
$G_1$	150	470	150.000	150.000	150.000	303.248	379.872	379.872	456.497	456.497	456.497	456.497	456.497	456.497
$G_2$	135	460	309.532	222.266	309.533	309.533	309.533	396.799	396.799	396.800	396.799	396.799	396.800	396.800
$G_3$	73	340	73.000	180.828	291.427	286.179	283.555	294.423	291.798	315.929	293.235	283.753	323.062	327.455
$G_4$	60	300	60.000	60.000	60.000	60.000	60.000	60.000	60.000	60.000	180.830	300.000	300.000	300.000
$G_5$	73	243	73.000	122.866	73.000	73.000	73.000	122.866	122.866	172.733	222.599	222.599	222.600	222.600
$G_6$	57	160	118.878	122.450	122.450	122.450	122.450	122.450	122.450	122.450	122.450	122.450	122.450	160.000
$G_7$	20	130	129.590	129.590	129.590	129.590	129.590	129.590	129.590	129.590	129.591	129.590	129.591	129.591
$G_8$	47	120	47.000	47.000	47.000	47.000	47.000	47.000	47.000	47.000	47.000	85.312	120.000	120.000
$G_9$	20	80	20.000	20.000	20.000	20.000	20.000	20.000	20.000	20.000	20.000	20.000	20.000	52.057
$G_{10}$	55	55	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000
$\Sigma P_i = P_d$			1,036	1,110	1,258	1,406	1,480	1,628	1,702	1,776	1,924	2,072	2,146	2,220
$\Sigma F(P_i)$			28,252.655	29,777.362	32,893.860	36,169.124	37,813.599	40,909.504	42,560.272	44,266.657	47,669.410	51,377.064	53,788.277	55,214.151
균형과정			$l-3$	$l$	$l$	$l$	$l+1$	$l$	$l-1$	$l$	$l$	$l$	$l$	$l$
교환과정			성인결음 -아기결음											

	Hours											
	13H	14H	15H	16H	17H	18H	19H	20H	21H	22H	23H	24H
$G_1$	456.497	456.497	456.497	303.248	379.872	379.872	456.497	456.497	456.497	379.872	150.000	150.000
$G_2$	396.799	396.799	396.800	396.799	309.533	396.799	396.800	396.799	396.799	396.799	396.799	222.267
$G_3$	283.753	283.235	315.929	297.047	283.555	294.423	315.929	283.753	293.235	294.423	178.428	304.692
$G_4$	300.000	180.830	60.000	60.000	60.000	60.000	60.000	300.000	180.830	60.000	60.000	60.000
$G_5$	222.599	222.599	172.733	122.866	73.000	122.866	172.733	222.599	222.599	122.866	172.733	73.000
$G_6$	122.450	122.450	122.450	122.450	122.450	122.450	122.450	122.450	122.450	122.450	122.450	122.450
$G_7$	129.590	129.590	129.591	129.590	129.590	129.590	129.591	129.590	129.590	129.590	129.590	129.591
$G_8$	85.312	47.000	47.000	47.000	47.000	47.000	47.000	85.312	47.000	47.000	47.000	47.000
$G_9$	20.000	20.000	20.000	20.000	20.000	20.000	20.000	20.000	20.000	20.000	20.000	20.000
$G_{10}$	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000	55.000
$\Sigma P_i = P_d$	2,072	1,924	1,776	1,554	1,480	1,628	1,776	2,072	1,924	1,628	1,332	1,184
$\Sigma F(P_i)$	51,377.064	47,669.410	44,266.657	39,263.645	37,813.599	40,909.504	44,266.657	51,377.064	47,669.410	40,909.504	34,580.071	31,318.456
균형과정	$l$	$l$	$l$	$l$	$l+1$	$l$						
교환과정	성인결음 -아기결음											

$\Sigma P_i^{\max} = 2,358$ 에 대해  $\max C_i$ 는  $G_1 : 470 \xrightarrow{2,344}{38.2435} 456$ 로  $G_1$  발전기의 출력량이 470에서 다음 밸브 지점 456으로 감소되면  $\Sigma P_i = 2,344$ 로 감소된다. 이후  $\Sigma P_i^{\min} = 690$ 까지 다음 순서로 발전량이 감소된다. 여기서  $G_i(x, y)$ 에서  $x$ 는 현 시점의 출력량이며,  $y$ 는 다음 밸브 지점의 발전량이다.

$$\begin{aligned} & \frac{\Sigma P_i^{\max} \cdot 2,344}{38.2435} G_1(456, 380) \xrightarrow{2,316}{37.7080} G_6(52, 20) \xrightarrow{2,296}{35.0681} G_5(223, 173) \\ & \xrightarrow{2,233}{28.7781} G_2(397, 310) \xrightarrow{2,190}{27.8917} G_3(297, 185) \xrightarrow{2,158}{27.4809} G_9(20, -) \xrightarrow{2,123}{26.8100} \\ & G_8(85, 47) \xrightarrow{2,085}{25.7918} G_6(122, 57) \xrightarrow{2,026}{24.6025} G_1(241, 181) \xrightarrow{1,966}{24.2126} G_1(181, \\ & 120) \xrightarrow{1,928}{24.0926} G_8(47, -) \xrightarrow{1,867}{24.0562} G_1(120, 60) \xrightarrow{1,807}{24.1196} G_1(60, 0) \xrightarrow{1,731}{22.0492} \\ & G_1(380, 303) \xrightarrow{1,681}{21.9799} G_5(173, 123) \xrightarrow{1,631}{21.9009} G_5(123, 73) \xrightarrow{1,554}{21.8647} G_1(303, \\ & 227) \xrightarrow{1,504}{21.8219} G_2(73, -) \xrightarrow{1,428}{21.7970} G_1(227, 150) \xrightarrow{1,351}{21.8521} G_1(150, -) \xrightarrow{1,264}{21.4293} \end{aligned}$$

$$\begin{aligned} & G_2(310, 222) \xrightarrow{1,176}{21.4344} G_2(222, 135) \xrightarrow{1,089}{21.3411} G_2(135, -) \xrightarrow{977}{21.0140} G_3(185, \\ & 73) \xrightarrow{865}{20.9266} G_3(73, -) \xrightarrow{800}{18.0732} G_6(57, -) \xrightarrow{763}{17.2240} G_7(93, 57) \xrightarrow{727}{16.5331} G_7 \\ & (57, 20) \xrightarrow{690}{17.0000} G_7(20, -) \end{aligned}$$

01H의 경우  $P_d = 1,036$ 으로  $\frac{1,089}{21.3411} G_2(135, -)$ 까지 발전량을 감소시킨 횟수  $l = 23$ 이다. 이 경우  $\Sigma P_i = 1,089$ ,  $P_d = 1,036$ 으로  $\Sigma P_i - P_d = 53$ 의 여유량이 발생한다. 이 여유량에 대해 각 발전기 별로  $F(P_i) - F(P_i - 1)$ 을 계산하여  $\max [F(P_i) - F(P_i - 1)]$ 일 발전기의 발전량을  $P_i \leftarrow P_i - 1$ 로 설정하였다.  $l - 1$ 번째에 대해서와  $l + 1$ 번째의 발전기 발전량을 증가시키고  $\Sigma P_i = P_d$ 를 맞추는 과정을 수행하여 최소 발전비용 값을 선택하였다. 교환 방법은 3장에서 제안된 방법과 동일하게 수행되었다.

제안된 알고리즘을 표 1의 데이터에 대해 수행한 결과는 표 4에 제시되어 있다.

제안된 VPBSO 알고리즘을 기존의 최적 연구 결과인 EPSO-GM<sup>[2]</sup>, ICA<sup>[20]</sup>와 발전비용을 비교한 결과는 표 5와 같다. EPSO-GM은 H5에서, ICA는 H1과 H13에서 최적의 결과를 얻었는데 반해 제안된 VPBSO 알고리즘은 나머지 21개의 시간대에서 최적치를 얻었다. 표 5로부터 제안된 VPBSO 알고리즘은 기존의 최적의 성능을 가진 ICA에 비해 1일 동안 동일한 전력량을 생산하는데 \$ 6,907.463의 비용을 절감하였다.

표 5. 10 발전기의 알고리즘 비교  
Table 5. Comparison of Algorithms for 10 Generators

Hour	Load	EPSO-GM <sup>[2]</sup>	ICA <sup>[20]</sup>	VPBSO	EPSO-GM 대비 비용절감	ICA 대비 비용절감
H1	1,036	28,511.541	28,238.754	28,252.655	-238.886	13.901
H2	1,110	30,373.729	29,828.077	29,777.362	-596.367	-50.715
H3	1,258	33,282.826	33,347.045	32,893.860	-389.966	-453.185
H4	1,406	36,429.445	36,296.715	36,169.124	-260.321	-127.591
H5	1,480	37,672.973	37,991.334	37,813.599	140.626	-177.735
H17		38,157.079	37,983.869	37,813.599	-343.480	-170.070
H6		41,415.959	41,387.159	40,909.504	-506.455	-477.655
H18	1,628	41,496.734	41,294.355	40,909.504	-587.230	-384.851
H22		41,555.343	41,280.418	40,909.504	-645.839	-370.914
H7	1,702	43,115.004	42,844.529	42,560.272	-554.732	-284.257
H8		44,375.058	44,600.484	44,266.657	-108.401	-333.827
H15	1,776	45,505.049	44,649.659	44,266.657	-1,238.392	-383.002
H19		44,635.630	44,374.060	44,266.657	-368.973	-107.403
H9		48,576.927	47,885.318	47,669.410	-907.517	-215.908
H14	1,924	48,215.013	47,818.061	47,669.410	-545.603	-148.651
H21		47,954.137	47,915.540	47,669.410	-284.727	-246.130
H10		52,009.385	51,887.342	51,377.064	-662.321	-510.278
H13	2,072	51,894.407	51,357.359	51,377.064	-457.343	19.705
H20		51,905.558	51,862.515	51,377.064	-528.494	-485.451
H11	2,146	54,036.432	53,788.277	53,235.332	-801.100	-552.945
H12	2,220	55,656.686	55,605.118	55,214.151	-442.535	-390.967
H16	1,554	40,209.207	39,816.706	39,263.645	-945.562	-553.061
H23	1,332	34,863.138	34,952.455	34,580.071	-283.067	-372.384
H24	1,184	31,893.846	31,462.345	31,318.456	-575.390	-143.889
Sum		1,023,711.106	1,018,467.494	1,011,560.031	-12,151.075	-6,907.463

또한, EPSO-GM<sup>[2]</sup>과 ICA<sup>[20]</sup>는 휴리스틱 방법으로 초기치를 랜덤하게 할당함으로써 인해 알고리즘을 수행할 때 마다 다른 결과를 나타내는 단점을 갖고 있다. 예로, H5와 H17은 동일한 전력을 생산함에도 불구하고 알고리즘 수행 결과는 다른 값을 나타낸다. 이러한 결과는 H5와 H17을 포함해 5개의 부류에서 나타나고 있다. 반면에, 제안된 알고리즘은 결정론적 알고리즘으로 항상 동일한 결과를 나타내는 장점이 있다.

제안된 알고리즘의 적합성을 검증하기 위해 추가적으로 표 6의 40 발전기에 대해 실험을 수행하였다. 표 6의

40 발전기에 대한 최적화 연구 결과는 표 7과 같다. 표에서 실제 발전량 데이터를 제시한 경우에 한해 MS-Office EXCEL 2007로 검증한 결과는 ( )에 제시하였으며, 발전량 데이터를 미 제시한 경우 (?)로 표기하였다.<sup>[1,23-32]</sup>

제안된 VPBSO의 교환 방법은 거인 걸음법으로  $\alpha = 1$ 에 대해  $P_i \pm \alpha$ 로 수행되었다. 표 7에서 최적 결과를 얻은 IPSO와 제안된 VPBSO의 결과를 비교하여 표 8에 제시하였다. VPBSO는 IPSO<sup>[30]</sup>에 비해 \$39,7494/Hour를 절감시키는 효과를 얻었다.

제안된 VPBSO 알고리즘은 균형을 빠르게 맞추고 교환도 간단히 수행하며, 항상 동일한 결과를 얻는 장점이 있다. 또한, 이러한 장점과 더불어 실험에 적용된 10 발전기의 24시간용 동적 경제급전 문제와 40 발전기의 1시간용 정적 경제급전 문제에 대해 최적의 결과도 얻을 수 있었다.

표 6. 40 발전기 데이터  
Table 6. Case Study of 40 Generators Data

Fuel Cost Function of 40 Generators							
$G_i$	$P_i^{\min}$	$P_i^{\max}$	$a$	$b$	$c$	$e$	$f$
1	36	114	0.00690	6.73	94.71	100	0.084
2	36	114	0.00690	6.73	94.71	100	0.084
3	60	120	0.02028	7.07	309.54	100	0.084
4	80	190	0.00942	8.18	369.03	150	0.063
5	47	97	0.01140	5.35	148.89	120	0.077
6	68	140	0.01142	8.05	222.33	100	0.084
7	110	300	0.00357	8.03	287.71	200	0.042
8	135	300	0.00492	6.99	391.98	200	0.042
9	135	300	0.00573	6.60	455.76	200	0.042
10	130	300	0.00605	12.90	722.82	200	0.042
11	94	375	0.00515	12.90	635.20	200	0.042
12	94	375	0.00569	12.80	654.69	200	0.042
13	125	500	0.00421	12.50	913.40	300	0.035
14	125	500	0.00752	8.84	1760.40	300	0.035
15	125	500	0.00708	9.15	1728.30	300	0.035
16	125	500	0.00708	9.15	1728.30	300	0.035
17	220	500	0.00313	7.97	647.85	300	0.035
18	220	500	0.00313	7.95	649.69	300	0.035
19	242	550	0.00313	7.97	647.83	300	0.035
20	242	550	0.00313	7.97	647.81	300	0.035
21	254	550	0.00298	6.63	785.96	300	0.035
22	254	550	0.00298	6.63	785.96	300	0.035
23	254	550	0.00284	6.66	794.53	300	0.035
24	254	550	0.00284	6.66	794.53	300	0.035
25	254	550	0.00277	7.10	801.32	300	0.035
26	254	550	0.00277	7.10	801.32	300	0.035
27	10	150	0.52124	3.33	1055.10	120	0.077
28	10	150	0.52124	3.33	1055.10	120	0.077
29	10	150	0.52124	3.33	1055.10	120	0.077
30	47	97	0.01140	5.35	148.89	120	0.077
31	60	190	0.00160	6.43	222.92	150	0.063
32	60	190	0.00160	6.43	222.92	150	0.063
33	60	190	0.00160	6.43	222.92	150	0.063
34	90	200	0.00010	8.95	107.87	200	0.042
35	90	200	0.00010	8.62	116.58	200	0.042
36	90	200	0.00010	8.62	116.58	200	0.042
37	25	110	0.01610	5.88	307.45	80	0.038
38	25	110	0.01610	5.88	307.45	80	0.038
39	25	110	0.01610	5.88	307.45	80	0.038
40	242	550	0.00313	7.97	647.83	300	0.035
Sum:	4,817	12,722			$P_d = 10,500 MW$		

표 7. 40 발전기 최적화 결과  
 Table 7. Optimization Results for the 40 Generators

Method	Min Cost	Mean Cost	Max Cost	St Dev
Genetic Algorithm (GA) <sup>[1]</sup>	119,732.25 (123,966.653)	-	-	-
Embedded Evolutionary Programming-PSO (CEP-PSO) <sup>[20]</sup>	123,670.000 (?)	124,145.60	124,900.00	-
Particle Swarm Optimization (PSO) <sup>[21]</sup>	122,980.450 (?)	-	-	-
Evolutionary Programming (EP) <sup>[22]</sup>	122,624.350 (?)	-	-	-
Hybrid Evolutionary Programming with SQP (HEP-SQP) <sup>[23]</sup>	122,379.630 (?)	-	-	-
Modified Particle Swarm Optimization (MPSO) <sup>[24]</sup>	122,232.265 (?)	-	-	-
Hybrid Particle Swarm with SQP (HPS-SQP) <sup>[24]</sup>	122,094.670 (?)	-	-	-
Chaotic Differential Evolution and Quadratic Programming (DEC-SQP) <sup>[27]</sup>	121,741.9793 (121,532.104)	122,295.12	122,839.29	386.18
New Particle Swarm Optimization Solution to Nonconvex (NPSO-LRS) <sup>[28]</sup>	121,664.430 (?)	122,209.31	122,981.59	-
Improved PSO (IPSO) <sup>[29]</sup>	121,485.7041 ( <b>121,504.702</b> )	121,699.301	122,168.116	155.619
Artificial Immune System (IS) <sup>[30]</sup>	121,489.110 (?)	121,737.73	122,163.56	106.88
SA-CLONAL <sup>[31]</sup>	121,486.12 (?)	121,507.03	121,591.70	-
Fuzzy CIS (FCIS) <sup>[30]</sup>	121,446.710 (?)	121,662.67	121,863.39	138.04
Genetic Algorithm (GA) <sup>[32]</sup>	121,432.177 (121,532.104)	-	-	-
Cultural Immune System (CIS) <sup>[30]</sup>	121,423.680 (?)	121,717.97	122,147.93	144.81

표 8. 40 발전기의 알고리즘 비교  
 Table 8. Comparison of Algorithms for 40 Generators

Gen <sup>t</sup>	$P_i^{\min}$	$P_i^{\max}$	IPSO <sup>[20]</sup>		VPBSO	
			$P_i$	$F(P_i)$	$P_i$	$F(P_i)$
1	36	114	111.7576	941.0496	110.7998	925.0964
2	36	114	111.5584	937.7332	110.7998	925.0964
3	60	120	97.3999	1,190.5485	97.3999	1,190.5485
4	80	190	179.7331	2,143.5503	179.7331	2,143.5503
5	47	97	91.6560	770.1281	87.7999	706.5001
6	68	140	140.0000	1,596.4643	140.0000	1,596.4643
7	110	300	300.0000	3,216.4240	259.5996	2,612.8845
8	135	300	300.0000	3,052.3095	284.5996	2,779.8365
9	135	300	284.5997	2,798.2312	284.5996	2,798.2302
10	130	300	130.0000	2,502.0650	130.0000	2,502.0650
11	94	375	168.7998	2,959.4585	158.7998	2,895.1323
12	94	375	94.0000	1,908.1668	94.0000	1,908.1668
13	125	500	214.7598	3,792.0703	214.7598	3,792.0703
14	125	500	394.2794	6,414.8612	394.2793	6,414.8601
15	125	500	304.5196	5,171.1984	394.2793	6,436.5860
16	125	500	304.5196	5,171.1984	394.2793	6,436.5860
17	220	500	489.2794	5,296.7114	489.2793	5,296.7107
18	220	500	489.2794	5,288.7658	489.2793	5,288.7651
19	242	550	511.2794	5,540.9299	511.2793	5,540.9292
20	242	550	511.2794	5,540.9099	511.2793	5,540.9092
21	254	550	523.2794	5,071.2903	523.2794	5,071.2903
22	254	550	523.2853	5,071.4098	523.2794	5,071.2903
23	254	550	523.2847	5,067.3304	523.2794	5,067.2237
24	254	550	523.2794	5,067.2237	523.2794	5,067.2237
25	254	550	523.2794	5,275.0891	523.2794	5,275.0891
26	254	550	523.2794	5,275.0891	523.2794	5,275.0891
27	10	150	10.0000	1,140.5240	10.0000	1,140.5240
28	10	150	10.0000	1,140.5240	10.0000	1,140.5240
29	10	150	10.0000	1,140.5240	10.0000	1,140.5240
30	47	97	90.3329	748.4522	87.7999	706.5001
31	60	190	190.0000	1,643.9913	190.0000	1,643.9913
32	60	190	190.0000	1,643.9913	190.0000	1,643.9913
33	60	190	190.0000	1,643.9913	190.0000	1,643.9913
34	90	200	200.0000	2,101.0170	164.7998	1,585.5443
35	90	200	200.0000	2,043.7270	164.7998	1,539.8704
36	90	200	200.0000	2,043.7270	164.7998	1,539.8704
37	25	110	110.0000	1,220.1661	110.0000	1,220.1661
38	25	110	110.0000	1,220.1661	110.0000	1,220.1661
39	25	110	110.0000	1,220.1661	110.0000	1,220.1661
40	242	550	511.2794	5,540.9299	511.2793	5,540.9292
Sum	4,817	12,722	10,500.0000	121,532.104	10,500.0000	121,464.9526
$P_d$	10,500					

## V. 결론

본 논문은 1일 24시간 동안 가장 경제적인 비용으로 전력을 생산할 수 있는 동적 경제급전의 최적화 문제의

해를 구하는 알고리즘을 제안하였다. 기존 알고리즘들은 휴리스틱 방법들을 적용하여 다수의 시행 결과 얻은 해들 중에서 최소치를 선택하는 방법을 적용하고 있음에도 불구하고 알고리즘을 수행할 때마다 다른 결과를 나타내는 단점이 있다. 반면에, 본 논문에서 제안한 VPBSO 알고리즘은 간단하면서도 항상 동일한 결과를 얻는 결정론적 알고리즘이며, 기존의 휴리스틱 방법들에 비해 보다 향상된 결과를 얻었다.

제안된 VPBSO 알고리즘은 가능한 모든 발전기를 벨브지점으로 작동시켜 벨브효과를 최소화하도록 하였다. 또한, 발전기별로 각기 다른 발전비용 효율성을 최적화시키기 위해 교환 방법을 적용하였다.

제안된 알고리즘을 10 발전기의 24시간 동적 경제급전 문제에 적용한 결과 기존의 최적 해를 얻은 ICA<sup>[20]</sup>에 비해 동일한 전력을 생산함에도 불구하고 \$ 6,907.463/1일의 비용을 절감하였다. 40 발전기의 경우 IPSO<sup>[28]</sup>에 비해 \$39.7494 /Hour를 절감시켜 가장 좋은 결과를 얻었다.

제안된 VPBSO 알고리즘은 간단하면서도 빠르게 항상 동일한 최적의 결과를 얻는 장점으로 인해 경제급전 문제의 실용화에 적용할 수 있을 것이다.

## References

- [1] A. A. El-Fergany, "Solution of Economic Load Dispatch Problem with Smooth and Non-Smooth Fuel Cost Functions Including Line Losses Using Genetic Algorithm," International Journal of

- Computer and Electrical Engineering, Vol. 3, No. 5, pp. 706-710, doi: 10.7763/IJCEE.2011.V3.407, Oct. 2011.
- [2] P. Sriyangong, "A Hybrid Particle Swarm Optimization Solution to Ramping Rate Constrained Dynamic Economic Dispatch," World Academy of Science, Engineering and Technology, Vol. 47, pp. 374-379, 2008.
- [3] P. Attaviriyapap, H. Kita, E. Tanaka, and J. Hasegawa, "A Hybrid EP and SQP for Dynamic Economic Dispatch with Non-smooth Fuel Cost Function," IEEE Trans. on Power System, Vol. 17, No. 2, pp. 411-416, doi: 10.1109/MPER.2002.4312139, Apr. 2002.
- [4] F. Li, R. Morgan, and D. Williams, "A Hybrid Genetic Approaches to Ramping Rate Constrained Dynamic Economic Dispatch," Electric Power System Research, Vol. 43, No. 2, pp. 97 - 103, doi: 10.1016/S0378-7796(97)01165-6, Nov. 1997.
- [5] X. Yuan, L. Wang, Y. Yuan, Y. Zhang, B. Cao, and B. Yang, "A Modified Differential Evolution Approach for Dynamic Economic Dispatch with Valve Point Effects," Energy Conversion and Management, Vol. 49, No. 12, pp. 3447 - 3453, doi: 10.1016/j.enconman.2008.08.016, Dec. 2008.
- [6] T. A. A. Victoire and A. E. Jeyakumar, "Reserve Constrained Dynamic Dispatch of Units With Valve-Point Effects," IEEE Trans. Power System, Vol. 20, No. 3, pp. 1273-1282, doi: 10.1109/TPWRS.2005.851958, Aug. 2005.
- [7] S. Hemamalini and S. Simon, "Maclaurin Series-based Lagrangian Method for Economic Dispatch with Valve-point Effect," IET General Transmission Distribution, Vol. 3, No. 9, pp. 859 - 871, doi:10.1049/iet-gtd.2008.0499, Sep. 2009.
- [8] T. A. A. Victoire and A. E. Jeyakumar, "A Modified Hybrid EP-SQP Approach for Dynamic Dispatch with Valve-point Effect," International Journal of Electrical Power & Energy Systems, Vol. 27, No. 8, pp. 594-601, doi:10.1016/j.ijepes.2005.06.006, Oct. 2005.
- [9] T. A. A. Victoire and A. E. Jeyakumar, "Deterministically guided PSO for Dynamic Dispatch Considering Valve-point Effect," Electric Power Systems Research, Vol. 73, No. 3, pp. 313-322, doi:10.1016/j.epr.2004.07.005, Mar. 2005.
- [10] Y. Lu, J. Zhou, H. Qin, Y. Wang, and Y. Zhang, "Chaotic Differential Evolution Methods for Dynamic Economic Dispatch with Valve-point Effects," Engineering Applications of Artificial Intelligence, Vol. 24, No. 2, pp. 378 - 387, doi:10.1016/j.engappai.2010.10.014, Mar. 2011.
- [11] H. Dakuo, G. Dong, F. Wang, and Z. Mao, "Optimization of Dynamic Economic Dispatch with Valve-point Effect using Chaotic Sequence based Differential Evolution Algorithms," Energy Conversion and Management, Vol. 52, No. 2, pp. 1026 - 1032, doi:10.1016/j.enconman.2010.08.031, Feb. 2011.
- [12] P. Sriyangong, "An Enhanced Particle Swarm Optimization for Dynamic Economic Dispatch Problem Considering Valve-Point Loading," In Proceeding of the Fourth IASTED International Conference on Power and Energy Systems (AsiaPES 2008), pp. 167-172, 2008.
- [13] N. Nomana and H. Iba, "Differential Evolution for Economic Load Dispatch Problems," Electric Power System Research, Vol. 78, No. 8, pp. 1322 - 1331, doi:10.1016/j.epr.2007.11.007, Aug. 2008.
- [14] R. Balamurugan and S. Subramanian, "Differential Evolution-based Dynamic Economic Dispatch of Generating Units with Valve-point Effects," Electric Power Components and Systems, Vol. 36, Nop. 8, pp. 828 - 843, doi:10.1080/15325000801911427, Jun. 2008.
- [15] S. Hemamalini and S. P. Simon, "Dynamic Economic Dispatch using Maclaurin Series based Lagrangian Method," Energy Conversion and Management, Vol. 51, No. 11, pp. 2212-2219, doi:10.1016/j.enconman.2010.03.015, Nov. 2010.
- [16] K. S. Hindi and M. R. Ab Ghani, "Dynamic Economic Dispatch for Large Scale Power

- Systems: A Lagrangian Relaxation Approach,” *Electric Power and Energy Systems*, Vol. 13, No. 1, pp. 51–56, doi:10.1016/0142-0615(91) 90018-Q, Feb. 1991.
- [17] R. Balamurugan and S. Subramanian, “An Improved Differential Evolution based Dynamic Economic Dispatch with Non-smooth Fuel Cost Function,” *Journal of Electrical Systems*, Vol. 3, No. 3, pp. 151 - 161, Sep. 2007.
- [18] J. C. Lee, W. M. Lin, G. C. Liao, and T. P. Tsao, “Quantum Genetic Algorithm for Dynamic Economic Dispatch with Valve-point Effects and Including Wind Power System,” *Electrical Power and Energy Systems*, Vol. 33, No. 2, pp. 189 - 197, doi:10.1016/j.ijepes.2010.08.014, Feb. 2011.
- [19] X. S. Han, H. B. Gooi, and D. S. Kirschen, “Dynamic Economic Dispatch: Feasible and Optimal Solutions,” *IEEE Transactions on Power System*, Vol. 16, No. 1, pp. 22 - 28, doi:10.1109/PSS.2001.970332, Feb. 2001.
- [20] A. Rabiei, A. Soroudi, and B. Mohammadi, “Imperialist Competition Algorithm for Solving Non-convex Dynamic Economic Power Dispatch,” *Energy*, Vol. 44, No. 1, pp. 228–240, doi:10.1016/j.energy.2012.06.034, Aug. 2012.
- [21] S. U. Lee, “Improved Valve-Point Optimization Algorithm for Economic Load Dispatch Problem with Non-convex Fuel Cost Function,” *Journal of IIBC*, Vol. 15, No. 6, pp. 257–266, doi:10.7236/JIIBC.2015.15.6.257, Dec. 2015.
- [22] S. U. Lee, “A Swap Optimization for Dynamic Economic Dispatch Problem with Non-smooth Function,” *Journal of KSCI*, Vol. 17, No. 11, pp. 189–196, doi:10.9708/jksci/2012.17.11.189, Nov. 2012.
- [23] N. Sinha and B. Purkayastha, “PSO Embedded Evolutionary Programming Technique for Nonconvex Economic Load Dispatch”, *IEEE PES-Power Systems Conference and Exposition*, Vol. 1, pp. 66–71, doi:10.1109/PSCE.2004.1397447, Oct. 2004.
- [24] T. A. A. Victoire and A. E. Jeyakumar, “Hybrid PSO-SQP for Economic Dispatch with Valve-point Effect,” *Electric. Power Systems Research*, Vol. 71, No. 1, pp. 51–59, doi:10.1016/j.epsr.2003.12.017, Sep. 2004.
- [25] N. Sinha, R. Chakrabarti, and P. K. Chattopadhyay, “Evolutionary Programming Techniques for Economic Load Dispatch,” *IEEE Trans. on Evolutionary Computing*, Vol. 7, No. 1, pp. 83–94, doi: 10.1109/TEVC.2002.806788, Feb. 2003.
- [26] J. B. Park, K. S. Lee, J. R. Shin, and K. Y. Lee, “A Particle Swarm Optimization for Economic Dispatch with Non-smooth Cost Function,” *IEEE Trans. Power System*, Vol. 20, No. 1, pp. 34–42, doi:10.1109/TPWRS.2004.831275, Feb. 2005.
- [27] L. S. Coelho and V. C. Mariani, “Combining of Chaotic Differential Evolution and Quadratic Programming for Economic Dispatch Optimization with Valve-Point Effect,” *IEEE Trans. on Power Systems*, Vol. 21, No. 2, doi:10.1109/TPWRS.2006.873410, May 2006.
- [28] A. I. Selvakumar, and K. Thanushkodi, “A New Particle Swarm Optimization Solution to Non-convex Economic Dispatch Problems,” *IEEE Trans. on Power Systems*, Vol. 22, No. 1, pp. 42–51, doi:10.1109/TPWRS.2006.889132, Feb. 2007.
- [29] J. B. Park, W. N. Lee, and J. R. Sin, “An Improved Particle Swarm Optimization for Economic Dispatch Problems with Non-Smooth Cost Functions,” *International Journal of Innovations in Energy Systems and Power*, Vol. 1, No. 1, pp. 1–7, doi:10.1109/PES.2006.1709300, 2006.
- [30] R. Goncalves, C. Almeida, J. Kuk, and M. Delgado, “Solving Economic Load Dispatch Problem by Natural Computing Intelligent Systems,” *15th International Conference on Intelligent System Applications to Power Systems (ISAP)*, pp. 1–6, doi:10.1109/ISAP.2009.5352843, Nov. 2009.
- [31] A. Khamseh and YY. Alinejad-Beromi, “Hybrid

CLONAL Selection Algorithm with SA for Solving Economic Load Dispatch with Valve-Point Effect," Canadian Journal on Electrical and Electronics Engineering, Vol. 2, No. 10, pp. 463-467, 2011.

- [32] P. H. Chen and H. C. Chang, "Large-Scale Economic Dispatch by Genetic Algorithm," IEEE Trans. on Power Systems, Vol. 10, No. 4, pp. 1919-1926, doi:10.1109/59.476058, Nov. 1995.

## 저자 소개

### 이 상 운(정회원)



- 1987년 : 한국항공대학교 항공전자공학과 (학사)
  - 1997년 : 경상대학교 컴퓨터과학과 (석사)
  - 2001년 : 경상대학교 컴퓨터과학과 (박사)
  - 2003년 : 강원도립대학 컴퓨터응용과 전임강사
  - 2004년 ~ 2007년 2월 : 국립 원주대학 여성교양과 조교수
  - 2007년 3월 ~ 2015년 3월 : 강릉원주대학교 멀티미디어공학과 부교수
  - 2015년 4월 ~ 현재 : 강릉원주대학교 멀티미디어공학과 정교수
- <주관심분야 : 소프트웨어 프로젝트 관리, 개발 방법론, 분석과 설계 방법론, 시험 및 품질보증, 소프트웨어 신뢰성, 그래프 알고리즘>
- E-Mail : sulee@gwnu.ac.kr