Periodica Polytechnica
Mechanical Engineering

59(3), pp. 102-108, 2015
DOI: 10.3311/PPme.7850

RESEARCH ARTICLE

Analysis of Particle Swarm-Aided Power Plant Optimization

Axel Groniewsky[1*]

Received 02 December 2014; accepted after revision 23 February 2015

Abstract

Stochastic optimization algorithms are usually evaluated based on performance on high dimensional benchmark functions and results of these tests determine the direction of development. Benchmark functions however, do not emulate complex engineering problems. In this paper a power plant optimization problem is presented and solved under different constraints with multiple elite dependent and single elite dependent swarm intelligence. Although on benchmark problems multiple elite dependent algorithms usually outperform single elite dependent ones, if search space is represented by simulation software, diversity not just increases iterations but computation time as well and because of that conventional PSO (particle swarm optimization) exceeds modified ones.

Keywords

particle swarm optimization, power plant optimization, plant performance monitoring software, thermodynamic simulation

[1] Department of Energy Engineering, Faculty of Mechanical Engineering, Budapest University of Technology and Economics
H-1521 Budapest, P.O.B. 91, Hungary

[*] Corresponding author, e-mail: groniewsky@energia.bme.hu

1 Introduction

The increasing complexity of search space and growing number of variables, typical for engineering problems of our time have a great impact on optimization methods. Instead of traditional optimization techniques which have limited scope in practical applications heuristic search methods become more and more frequently used tools [1]. PSO can be considered one of the most important nature-inspired computing methods in optimization research [2]. It has several properties in common with other types of evolution-based collective intelligence, such as genetic algorithms including: random search in choice sets (search space) and population of individuals, but also differs from them since each individual can learn from itself and others to optimize its performance. When a particle locates a momentary best position (highest fitness value so fare), it shares the information with the other swarm members. As a result, all other swarm members change their positions to the direction of the target. The track and the velocity of a particle are defined individually and depend on its own experience and the experience of the most effective member of the swarm [3].

The application of PSO in the field of energy engineering is widespread. Al-Saedi et al. [4] elaborated an optimal power control strategy, for an inverter based Distributed Generation unit, in an autonomous microgrid operation based on real-time self-tuning method using PSO. Clarke et al. [5] used PSO to find the trade-off between specific work output and specific heat exchanger area of a binary geothermal power plant. To increase variety during optimum search Zafar et al. [6] applied fully informed PSO to reduce loss in power transmission. In order to compensate the instability of differential evolution and early convergence of PSO, Gnanambal et al. [7] uses hybridized DE-PSO algorithm to determine the maximum loadability limit of a power system. Ji et al. [8] combined PSO with gravitational search algorithm to solve economic emission load dispatch problems considering various practical constraints. Also for economic load dispatch Hosseinnezhad et al. [9] proposed a Species-based Quantum PSO where the number of groups in any iteration is determined considering the Hamming distance from the seed species to its border. Eslami et al. [10] proposed passive congregation PSO with chaotic sequence inertial weight to find optimal tuning and placement of power system stabilizer. Mariani et al. [11] presented a cost optimal shell and tube heat exchanger design using chaotic quantum-behaved PSO.

A great number of issues are solved with PSO in which the algorithm has been found to be robust, flexible, and stable. It is insensitive to local optimum or saddle and suitable to solve complex optimization problems with many parameters. PSO is fast in solving non-linear, non-differentiable multi-modal problems [12] and it does not require gradient computation. As literature shows conventional PSO is often modified to maintain diversity and avoid premature convergence even if the properties of search space do not justify an altered balance between variety and convergence speed. Complex engineering problems, where the evaluation of a single particle requires significant computational effort, diversity could cause major running time. The aim of this article is to demonstrate how conventional PSO outperforms a modified PSO when energy conversion system is optimized from thermodynamic viewpoint involving powerful plant performance monitoring software.

2 The PSO concept

Consider an unconstrained *D*-dimensional minimization problem as follow:

$$Min\ f(X),\ X=\left[x^1,\ldots x^j,\ldots x^D\right] \tag{1}$$

where *X*, as a member of the swarm is a solution to be optimized in a form of a *D*-dimensional vector. Assumed that x_i^j is the position and v_i^j is the velocity of the *i*th particle on the *j*th dimension their values can be updated by iteration as follows [13]:

$$\begin{aligned} v_i^j &= v_i^j + c_1 \cdot r1_i^j \cdot \left(pbest_i^j - x_i^j\right) + \\ &+ c_2 \cdot r2_i^j \cdot \left(gbest_i^j - x_i^j\right), \end{aligned} \tag{2}$$

$$x_i^j = x_i^j + v_i^j \cdot \Delta t, \tag{3}$$

where $X_i=\left(x_i^1 \ldots x_i^j \ldots x_i^D\right)$ and $V_i=\left(v_i^1 \ldots v_i^j \ldots v_i^D\right)$ represents the position and velocity, respectively of the *i*th particle in the *D*-dimensional search space while $pbest_i=\left(pbest_i^1 \ldots pbest_i^j \ldots pbest_i^D\right)$ and $gbest_i=\left(gbest_i^1 \ldots gbest_i^j \ldots gbest_i^D\right)$ represents the best position of the *i*th particle and the overall best position of the swarm discovered so fare. Δt refers to the time steps between two iterations and can be considered as 1. The acceleration constants *c1* and *c2* are the cognitive and social learning rates, respectively, denoting the relative importance of *pbest* and *gbest* positions. $r1_i^j$ and $r2_i^j$ are randomly generated numbers in the range [0,1].

Since its introduction many researchers have worked on improving the performance of PSO by modifying the velocity updating strategy of the original algorithm. The ratio of global and local exploration of new areas depends on the quality and quantity of the elite examples who share their information with neighbors. In this paper single elite and multiple elite algorithms are tested.

In Canonical PSO (CPSO) [12] only the best particle shares information with neighbors. Its velocity updating differs from the original algorithm in the use of inertia weight *w* alone, which keep balance between global and local search abilities:

$$\begin{aligned} v_i^j &= w_i \cdot v_i^j + c_1 \cdot r1_i^j \cdot \left(pbest_i^j - x_i^j\right) \\ &+ c_2 \cdot r2_i^j \cdot \left(gbest_i^j - x_i^j\right). \end{aligned} \tag{4}$$

Although several variants of inertia weight are proposed the one applied in canonical form is linearly descending $w_i = w_{max} - [(w_{max} - w_{min}) / i_{max}] \times i$).

Comprehensive Learning Strategy PSO (CLPSO) [14] applies multiple elite examples to prevent premature convergence. The velocity update algorithm of CLPSO is presented in Eq. (5):

$$v_i^j = w_i \cdot v_i^j + c_1 \cdot r1_i^j \cdot \left(pbest_{fi(j)}^j - x_i^j\right), \tag{5}$$

where $f_i = (f_i(1) \ldots f_i(j) \ldots f_i(D))$ defines which neighbors' *pbest* the particle *i* should follow. $pbest_{fi(j)}^j$ can be the corresponding dimension of any particle's *pbest* including its own *pbest*. It always depends on a probability factor called *Pc* learning probability:

$$Pc_i = 0{,}05 + 0{,}45 \cdot \frac{\left(\exp\left(\frac{10(i-1)}{ps-1}\right)-1\right)}{\exp(10)-1}. \tag{6}$$

Besides $v_{\max}^j$ maximum velocity has to be given for both algorithms to determine constraints:

$$v_i^j = \min\left(v_{\max}^j, \max\left(v_{\min}^j, v_i^j\right)\right). \tag{7}$$

3 Case study

Previously introduced algorithms are tested on the thermodynamic model of a LANG-BBC 215 MW steam turbine (Fig. 1) to determine highest system efficiency (η_0) under different constraints. These units were operating in Dunamenti Power Plant and Tisza II Thermal Power Plant between the 70's and 90's. The superheated steam is generated by superheater SH of boiler, it next expands in high pressure turbine HPST, it then reheated in reheater RH, and expands first in intermediate pressure turbine IPST than in low pressure turbine LPST to condenser pressure. In the main condenser MC steam condensates at constant pressure and saturation temperature. Feedwater than delivered to the regenerative system by the pump EP. The regenerative system composed of 3 low pressure feedwater heaters E1..E3 and 3 high pressure ones E5..E7 separated by DEA deaerator and a main feedwater pump. MFP is driven by an auxiliary turbine PT where extraction steam expands to condenser pressure. After condensation in auxiliary condenser

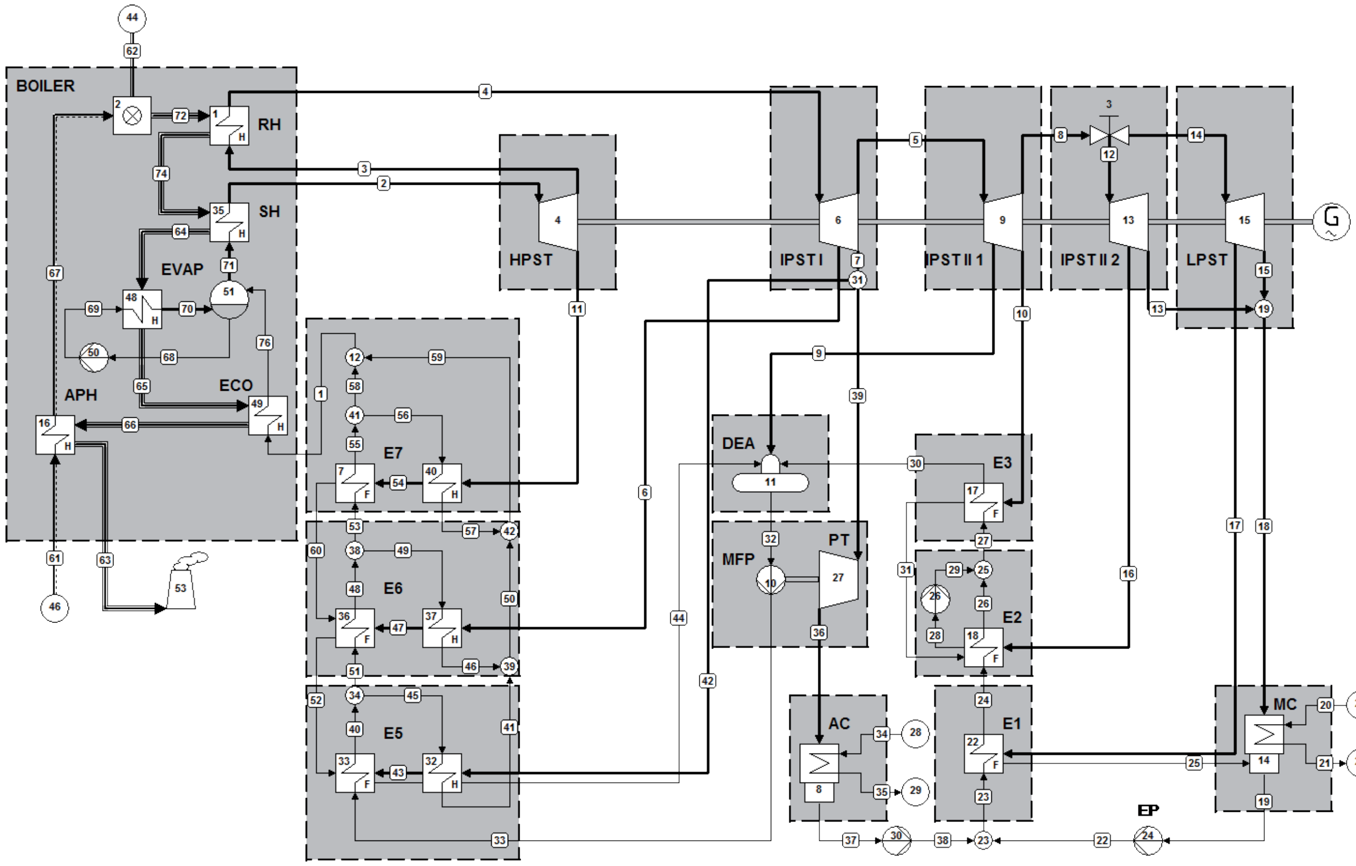

Fig. 1 Simplified scheme of the LANG-BBC 215 MW steam turbine cycle

AC feedwater delivered to low pressure heater E1 by pump. Drains from heater E3 are cascaded to heater E2 and delivered to cold side of E3 by drain pump. Drains from E1 are directed to the condenser. The high pressure feedwater heaters with external desuperheaters and drain coolers have Nekolny-Ricard arrangement. State properties corresponding to Fig. 1 are given in Table 4.

Experience shows that system efficiencies are unimodal functions of the variables [15] which means that a local optimum is also a global one. In most cases, the efficiency function is a very flat function of the variables in the neighborhood of the optimum. Although the search space representing all theoretically possible parameter set is greater than the set of physically possible solutions and in complicated systems the variables are often badly scaled, with properly chosen constraints discontinuities of optimization landscape can be avoided. If however - independently from parameter set – simulation does not converge it causes discontinuities in search space and verify the use of heuristic search algorithms. Fig. 2 shows system efficiency of the steam turbine as the functions of p_3 and p_6 extraction pressures.

This optimization landscape has more discontinuities which are not indicated with iteration number (Fig. 3).

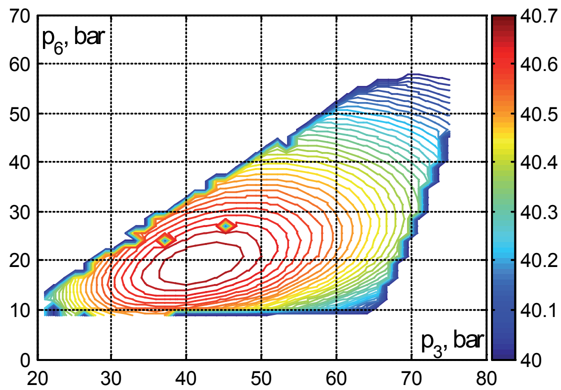

Fig. 2 2-dim search space of LANG-BBC 215 MW steam turbine

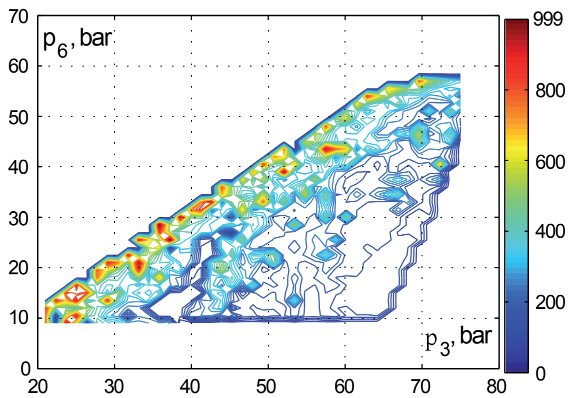

Fig. 3 Iteration map of LANG-BBC 215 MW steam turbine

Design parameters representing dimensions of search space are as follows: live steam parameters, isentropic efficiency and extraction pressures of steam turbine, terminal temperature differences (TTD) and drain cooler approaches (DCA) of feedwater heaters. This way parameter sets contains variables with significant and insignificant effects on system efficiency as well. Also, some of the variables have no optima and have to fall on either the lower or upper limit of search space.

CPSO and CLPSO were tested on a small and a large search space (Table 1) to see how ratio of calculable and incalculable solutions affects the algorithms.

Table 1 Decision variables with the upper and lower bounds

Range	wide (R1)		narrow (R2)	
Bound	lower	upper	lower	upper
T_2, °C	440	580	520	580
η, 1	0,78	0,9	0,84	0,9
p_2, bar	130	170	150	170
p_{11}, bar	25	80	35	45
p_6, bar	10	40	15	25
p_7, bar	5	20	6	10
p_9, bar	2	10	3	5
p_{10}, bar	1	5	1	2
p_{16}, bar	0,4	2	0,5	0,8
p_{17}, bar	0,1	0,8	0,1	0,3
TTD_{E1}, °C	2	15	2	8
TTD_{E2}, °C	2	15	2	8
TTD_{E3}, °C	2	15	2	8
DCA_{E3}, °C	2	15	2	8
TTD_{E5}, °C	2	15	2	8
DCA_{E5}, °C	2	15	2	8
TTD_{E6}, °C	2	15	2	8
DCA_{E6}, °C	2	15	2	8
TTD_{E7}, °C	2	15	2	8
DCA_{E7}, °C	2	15	2	8

Parameters of CPSO and CLPSO are set according to [12] and [14] respectively. Number of particles, number of iterations, initial (w_{max}) and final (w_{min}) inertia weights for both algorithms are 25, 120, 0,9 and 0,4 respectively. Cognitive ($c1$) and social ($c2$) learning rates of CPSO are 1, learning probability factor (c) of CLPSO is 1,49445, refreshing gap (m) is 7.

Thermodynamic analysis of the optimization process is carried out in GateCycle (GC) plant performance monitoring software using JANAF data for the properties of ideal gases and IAPWS-IF97 for the properties of water and steam. PSO algorithms are developed and all optimization runs are controlled in MATLAB however dynamic data exchange is performed via Microsoft EXCEL. Following steps are performed at each iteration:

Step 1. PSO provides new design variables for GC;

Step 2. after simulation with new variables, GC provides thermodynamic properties for PSO search algorithms;

Step 3. based on new thermodynamic data, PSO evaluates the objective function and based on results creates new design variables.

4 Results

Performance of swarm intelligences were compared based on different constraints shortlisted in Table 2.

Table 2 Constraints of runs

Type	P1	P2	P3	P4	P5	P6
Algorithm		CPSO			CLPSO	
Range	R1	R1	R2	R1	R1	R2
$v_i^j / v_{\lim}^j$	0,33	0,166	0,33	0,33	0,166	0,33

Live steam parameters (p_2, T_2) and isentropic efficiency (η) have no optima therefore in optimal conditions they reach the upper limit of search space and for the same reason TTDs and DCAs reach the lower one. Since optimal extraction conditions – if stage efficiency is constant - can be estimated either by keeping the temperature change of feedwater heaters constant ($\Delta T_{FWH,i} = \Delta T_{FWH}^{\Sigma} / n$) Tor by keeping the rate of temperature change constant ($q_i = \sqrt[n]{T_{FWH\,n,out} / T_{FWH\,1,out}}$) both methods were calculated and compared with results of PSO based optimum search. All simulations were repeated at least 5 times. Table 3 contains the best results of simulations under different conditions.

Results show that regardless of the size of search space or velocity constraints, CPSO outperforms CLPSO. It provides better results than $\Delta T_{FWH,\,i}$ and exceeds q_i method under P3 condition. CLPSO only surpass $\Delta T_{FWH,\,i}$ under P6 condition and does not exceed q_i. Because of high diversity, multiple elite dependent CLPSO requires significantly more iterations for a successful run than single elite dependent CPSO and have more non-convergent solutions per iteration as well. Since total computation time exceeded 2200 hours (Intel Core 2 Duo E8500, 4GB RAM) iteration threshold was not increased. Both PSOs had their best results in small search spaces (P3, P6).

Figure 4 and Figure 5 shows minimum, maximum, and mean values for system efficiencies (η_0) and for average number of non-convergent solutions per iteration (ANCS) under different conditions.

Table 3 Results of simulations under different constraints

	$\Delta T_{FWH,i}$	q_i	P1	P2	P3	P4	P5	P6
η_0, 1	42,886	42,934	42,934	42,930	42,958	42,859	42,442	42,903
T_2, °C	580	580	580	580	580	580	580	580
η , 1	0,9	0,9	0,9	0,9	0,9	0,9	0,9	0,9
p_2, bar	170	170	170	170	170	170	163,013	170
p_{11}, bar	45,798	45,798	43,448	44,039	45,798	51,565	45,780	41,164
p_6, bar	29,313	24,923	20,938	22,021	22,222	24,975	21,358	20,792
p_7, bar	15,219	11,156	9,999	10,285	11,046	14,697	13,163	10
p_9, bar	6,731	4,365	3	4,218	4,289	7,459	6,737	3,600
p_{10}, bar	2,851	1,725	1,492	1,857	1,901	2,792	3,778	1,568
p_{16}, bar	0,942	0,573	0,5	0,716	0,644	0,896	1,495	0,527
p_{17}, bar	0,246	0,172	0,129	0,159	0,149	0,208	0,418	0,185
TTD_{E1}, °C	2	2	2	2	2	2	7,442	2,429
TTD_{E2}, °C	2	2	2	2	2	2	8,395	2,125
TTD_{E3}, °C	2	2	2	2	2	2,792	9,530	2,472
DCA_{E3}, °C	2	2	2	11,147	2,002	2	8,805	2
TTD_{E5}, °C	2	2	2	2	2	2,999	4,781	2,257
DCA_{E5}, °C	2	2	2	15	2	2,071	5,801	3,007
TTD_{E6}, °C	2	2	2	2	2	4,258	9,540	3,274
DCA_{E6}, °C	2	2	8	2	2	9,121	5,967	2
TTD_{E7}, °C	2	2	2	2	2	2	11,236	5,232
DCA_{E7}, °C	2	2	6,737	15	2	11,256	14,603	5,805

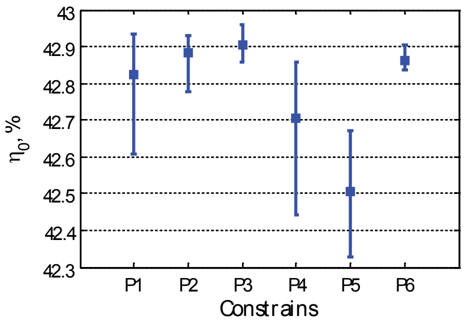

Fig. 4 System efficiencies under different constraints

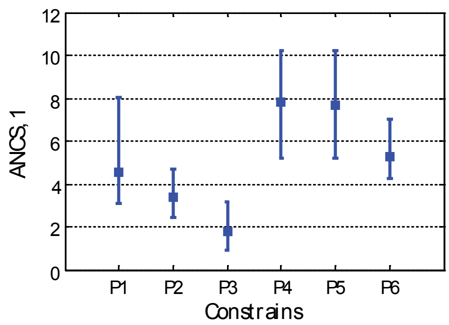

Fig. 5 Average number of non-convergent solutions under different constraints

In a fixed search space, reduced velocity increases the search efficiency of conventional PSO, decreases standard deviation and the number of non-convergent solutions per iteration. Experience shows however that too small velocity maximum decreases global search ability and increases iteration. Well-chosen parameter sets and constraints could increase the effectiveness of the algorithm and reduce the average number of non-convergent solutions per iteration.

Since under given iteration threshold CLPSO did not converged, results of algorithm cannot be assessed. Also, high diversity in velocity updating increases the number of non-convergent solutions (Fig. 6, Fig. 7). As evaluation time of a particle depends on the number of internal iterations of simulation software which is always higher when particles do not converge, average computational time for CLPSO is higher than for CPSO.

Table 4 Flow properties of LANG-BBC 215 MW steam turbine for base case

	m, kg/s	p, kPa	T, K	h, kJ/kg	s, kJ/kg-K		m, kg/s	p, kPa	T, K	h, kJ/kg	s, kJ/kg-K
1	178,6	16188,0	524,6	1093,0	2,778	33	178,6	16188,0	425,0	649,8	1,843
2	178,6	16188,0	813,2	3410,0	6,442	34	638,9	200,0	285,1	50,6	0,181
3	163,6	3844,2	606,5	3055,1	6,539	35	638,9	200,0	291,1	75,3	0,266
4	163,6	3844,2	813,2	3538,9	7,227	36	7,2	4,0	302,1	2317,1	7,690
5	139,6	895,3	618,6	3150,6	7,341	37	7,2	4,0	302,1	121,4	0,422
6	11,1	1844,6	711,8	3335,2	7,291	38	7,2	1373,0	302,2	123,0	0,423
7	13,0	895,3	618,6	3150,6	7,341	39	7,2	895,3	618,6	3150,6	7,341
8	125,0	153,0	438,5	2803,6	7,483	40	178,6	16188,0	444,0	731,5	2,031
9	8,6	430,5	539,6	2997,4	7,409	41	4,6	16188,0	543,9	1186,3	2,953
10	6,0	153,0	438,5	2803,6	7,483	42	5,8	895,3	618,6	3150,6	7,341
11	15,0	3844,2	606,5	3055,1	6,539	43	5,8	895,3	453,8	2787,3	6,655
12	45,2	153,0	438,5	2803,6	7,483	44	31,9	895,3	431,7	669,3	1,927
13	38,4	3,9	301,7	2354,9	7,825	45	4,6	16188,0	444,0	731,5	2,031
14	79,7	153,0	438,5	2803,6	7,483	46	10,8	16188,0	568,0	1309,2	3,174
15	73,8	3,9	301,7	2352,8	7,819	47	11,1	1844,6	526,7	2918,7	6,612
16	6,9	63,2	368,4	2671,2	7,558	48	174,0	16188,0	478,1	880,5	2,354
17	5,9	16,7	329,3	2496,5	7,650	49	10,8	16188,0	478,1	880,5	2,354
18	112,2	3,9	301,7	2353,5	7,821	50	15,4	16188,0	560,9	1272,3	3,109
19	118,1	3,9	301,7	119,7	0,417	51	174,0	16188,0	444,0	731,5	2,031
20	10007,7	200,0	288,1	63,2	0,224	52	26,1	1844,6	452,5	760,8	2,132
21	10007,7	200,0	294,2	88,3	0,311	53	163,3	16188,0	478,1	880,5	2,354
22	118,1	1373,0	301,8	121,3	0,418	54	15,0	3844,2	540,9	2871,3	6,218
23	125,3	1373,0	301,8	121,4	0,418	55	163,3	16188,0	517,5	1059,4	2,714
24	125,3	1373,0	325,9	222,0	0,739	56	10,8	16188,0	517,5	1059,4	2,714
25	5,9	16,7	329,3	235,2	0,783	57	10,8	16188,0	568,2	1310,7	3,177
26	125,3	1373,0	356,2	348,8	1,110	58	152,5	16188,0	517,5	1059,4	2,714
27	138,2	1373,0	356,6	350,5	1,115	59	26,2	16188,0	564,0	1288,2	3,137
28	12,9	63,2	360,4	365,4	1,161	60	15,0	3844,2	483,0	897,7	2,420
29	12,9	1373,0	360,5	367,0	1,161	61	200,0	101,3	288,1	-0,6	6,869
30	138,2	1373,0	381,5	455,4	1,400	62	11,1	300,0	288,1	-1,1	10,172
31	6,0	153,0	362,9	376,0	1,190	63	211,1	101,2	429,0	155,4	7,586
32	178,6	430,5	419,4	616,2	1,804						

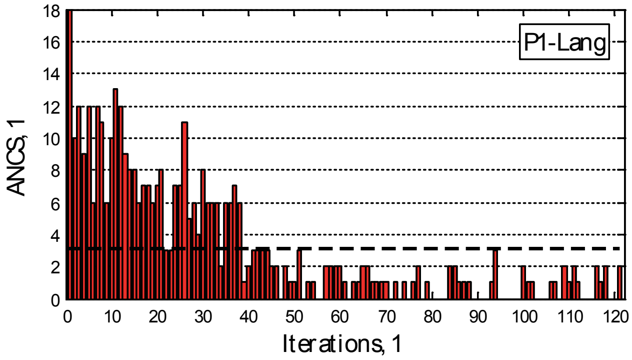

Fig. 6 Number of non-convergent solutions – CPSO

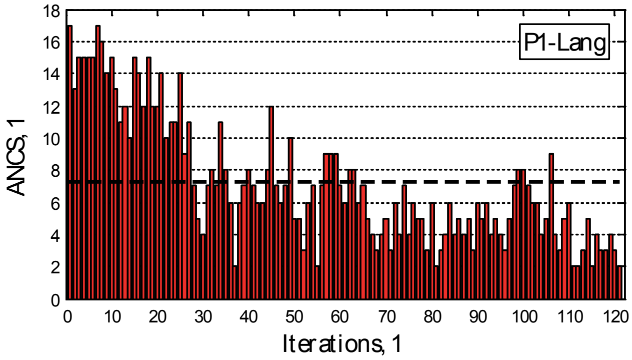

Fig. 7 Number of non-convergent solutions – CLPSO

5 Conclusion

PSO is suitable to optimize thermodynamic models of energy conversion systems. Sensitivity of the algorithm to discontinuities of optimization landscape depends on the velocity update algorithm, the size of the velocity relative to search space and the ratio of calculable and incalculable parameter sets. For a fixed search space, reduced velocity increases the search efficiency of conventional PSO, decreases standard deviation and the number of non-convergent solutions per iteration. Small velocity maximum however decreases global search ability and increases iteration. Well-chosen parameter sets and constraints could increase the efficiency of the algorithm and reduce the average number of non-convergent solutions per iteration. Because of high diversity, multiple elite dependent CLPSO requires significantly more iterations for a successful run than single elite dependent CPSO and have more non-convergent solutions per iteration as well. Due to high computational time CLPSO is less suitable for power plant optimization if thermodynamic model is developed in a plant performance monitoring software. Computation time is affected only slightly by iteration threshold and swarm size but is significantly affected by the quality of the variables. This is primarily due to the fact that evaluation time of a particle depends on the number of internal iterations of simulation software, which is always higher when particles do not converge.

Acknowledgement

The project presented in this article is supported by GE Energy.

References

[1] Groniewsky, A. "Exergoeconomic optimization of a thermal power plant using particle swarm optimization." *Thermal Science.* 17 (2). pp. 509-524. 2013. DOI: 10.2298/tsci120625213g

[2] Huang, H., Qin, H., Hao, Z., Lim, A. "Example-based learning particle swarm optimization for continuous optimization sciences." *Information Sciences.* 182 (1). pp. 125-138. 2012. DOI: 10.1016/j.ins.2010.10.018

[3] Singiresu, S. R. "*Engineering Optimization: Theory and Practice.*" 4th ed. John Wiley & Sons Inc. 2009.

[4] Al-Saedi, W., Lachowicz, S. W., Habibi, D., Bass, O. "Voltage and frequency regulation based DG unit in an autonomous microgrid operation using particle swarm optimization." *International Journal of Electrical Power & Energy Systems.* 53 (1). pp. 742–751. 2013. DOI: 10.1016/j.ijepes.2013.06.002

[5] Clarke, J., McLeskey, J. T. "Multi-objective particle swarm optimization of binary geothermal power plants." *International Journal Applied Energy.* 138. pp. 302–314. 2015. DOI: 10.1016/j.apenergy.2014.10.072

[6] Ul-Hassan, H. T., Zafar, R., Mohsin, S. A., Lateef, O. "Reduction in power transmission loss using fully informed particle swarm optimization." *International Journal of Electrical Power & Energy Systems.* 43 (1). pp. 364–368. 2012. DOI: 10.1016/j.ijepes.2012.05.028

[7] Gnamabali, K., Babulal, C. K. "Maximum loadability limit of power system using hybrid differential evolution with particle swarm optimization." *International Journal of Electrical Power & Energy Systems.* 43 (1). pp. 150–155. 2012. DOI: 10.1016/j.ijepes.2012.04.033

[8] Jiang, S, Ji, Z, Shen, Y. "A novel hybrid particle swarm optimization and gravitational search algorithm for solving economic emission load dispatch problems with various practical constraints." *International Journal of Electrical Power & Energy Systems.* 55 (1). pp. 628–644. 2014. DOI: 10.1016/j.ijepes.2013.10.006

[9] Hosseinnezhad, V., Rafiee, M., Ahmadian, M., Ameli, M. T. "Species-based Quantum Particle Swarm Optimization for economic load dispatch." *International Journal of Electrical Power & Energy Systems.* 63 (1). pp. 311–322. 2014. DOI: 10.1016/j.ijepes.2014.05.066

[10] Eslami, M., Shareef, H., Mohamed, A., Khajehzadeh M. "An efficient particle swarm optimization technique with chaotic sequence for optimal tuning and placement of PSS in power systems." *International Journal of Electrical Power & Energy Systems.* 43 (1). pp. 467–478. 2012. DOI: 10.1016/j.ijepes.2012.07.028

[11] Mariani, V. C., Duck, A. R. K., Guerra, F. A., Coelho, L. S., Rao, R. V. "A chaotic quantum-behaved particle swarm approach applied to optimization of heat exchangers." *Applied Thermal Engineering.* 42. pp. 119–128. 2012. DOI: 10.1016/j.applthermaleng.2012.03.022

[12] Shi, Y. H., Eberhart, R. C. "A Modified Particle Swarm Optimizer." *IEEE International Conference on Evolutionary Computation, Anchorage, Alas.* USA. 1. pp. 318-321. 1998.

[13] Eberhart, R., Kennedy, J. "A New Optimizer Using Particle Swarm Theory." *Micro Machine and Human Science, Proceedings of the Sixth International Symposium*. 1995.

[14] Liang, J. J., Qin, A. K., Suganthan, P. N., Baskar, S. "Comprehensive Learning Particle Swarm Optimizer for Global Optimization of Multimodal Functions." *IEEE Transactions on Evolutionary computation.* 10 (3). pp. 281-295. 2006. DOI: 10.1109/tevc.2005.857610

[15] van der Lee, P. E. A., Terlaky, T., Woudstra, T. "A new approach to optimizing energy systems." *Computer Methods in Applied Mechanics and Engineering*. 190 (40). pp. 5297-5310. 2001. DOI: 10.1016/s0045-7825(01)00162-1