Proceedings of IMECE2007
2007 ASME International Mechanical Engineering Congress and Exposition
November 11-15, 2007, Seattle, Washington, USA

IMECE2007-41224

A NEW EXTENSION OF PARTICLE SWARM OPTIMIZATION AND ITS APPLICATION IN ELECTRONIC HEAT SINK DESIGN

M.R. Alrasheed
Mechanical Engineering Department
The University of British Columbia
Vancouver, Canada
mrasheed@mech.ubc.ca

C.W. de Silva
Mechanical Engineering Department
The University of British Columbia
Vancouver, Canada
desilva@mech.ubc.ca

M.S.Gadala
Mechanical Engineering Department
The University of British Columbia
Vancouver, Canada
gadala@mech.ubc.ca

ABSTRACT

Particle Swarm optimization (PSO) is a robust stochastic evolutionary computation technique which is based on the movement and intelligence of swarms. In this paper the PSO algorithm is modified to improve its performance in a class of design applications in heat transfer. The developed approach includes a new term called a chaotic acceleration factor (Ca) into the algorithm, which enhances its convergence rate and its accuracy. The modified PSO is empirically tested with well- known benchmark functions. Next it is applied in plate-fin design with the objective of dissipating the maximum heat generation from an electronic component by minimizing the entropy generation rate to obtain the highest heat transfer efficiency.

INTRODUCTION

The recent trend in the electronic devices industry is moving toward denser and larger heat flux densities so that more powerful products require higher thermal performance from a cooling technique. For example, some 900 million computers are in use in the world today, with personal computers comprising approximately half of the total. This growth is reasonable in view of the past projections indicating that some 400 million computers were in use by the end of 2001[1]. These rapid advances in computer systems and other digital hardware have led to the associated increases in the thermal dissipation from microelectronic devices. This need fuels the interest of engineers and researchers in controlling the maximum operating temperature, and achieving long term reliability and efficient performance in electronic components. In electronic equipment, the temperature of each component must be maintained within an allowable upper limit, specified for each component from the viewpoint of operating performance and reliability. The power density in electronic systems is growing due to the high speeds of operation that are attained and the miniaturization of the associated components and devices. Generally, heat sinks are used to maintain the operating temperature for reliable operation of the electronic device. Choosing a suitable heat sink has become crucial to overall performance of electronic packages. The forced-air cooling technique, which is one of the effective methods for thermal management of electronic equipment cooling, is commonly used in electronic cooling area [2]. The development of a systematic and rather optimal design methodology for air-cooling heat sinks is undoubtedly very important in satisfying the current thermal necessities and for successful heat removal in the future generations of critical electronic components [3].

The performance of forced air convection heat sinks in electronic devices depend on a number of parameters including the thermal resistance, dimensions of the cooling channels, location and concentration heat sources as well as the airflow bypass due to flow resistance through the channel. In general, one of the most important goals of the heat sink design is to reduce the overall thermal resistance [4]. An alternative and related criterion for designing a heat sink is to maximize the thermal efficiency. Both criteria would affect the maximum heat dissipation. In a practical industrial design, different criteria are chosen depending on whether the primary objective is to maximize the transmitted heat, minimize the pumping power, or obtain the minimum volume or weight under the prescribed constraints such as component size and heat transfer time [5].

Recently, an evolutionary computation technique called particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Kennedy and Eberhart [6,7] and has been inspired by the behavior of schools of fish and flocks of birds. Unlike other heuristic techniques of optimization, PSO has a flexible and well-balanced mechanism to enhance and adapt to the global and local exploration abilities. PSO has its roots primarily in two methodologies. Perhaps more obvious are its ties to artificial life (A-life), and the behavior of flocks of birds, schools of fish, and swarms in particular. It is also related to evolutionary computation, and has ties to genetic algorithms and evolutionary strategies [8]. Moreover, particle swarm optimization has also been proved to perform well in evolutionary algorithms test functions and be used to solve many of the same kinds of problems as evolutionary algorithms, and it appears to be a promising approach and early testing has found the implementation to be effective with complex practical problems. PSO has been applied successfully to wide vary of areas such as electrical power systems [8], engineering structures [9], and electromagnetic area [10], etc. Up to authors' knowledge, PSO has not been yet applied in electronic cooling area so that one of the aims of current research is studying of applicability of PSO in such area.

NOMENCLATURE

a	Fin height, m.
a / s	Aspect ratio of fin height to fin spacing
A	Total surface area of heat sink including the fins and exposed other surface, m^2.
A_c	Cross –sectional area of the fin, m^2.
A_{ch}	Cross –sectional area of the channel, m^2.
b	Base thickness of the heat sink, m.
d	Fin thickness, m.
D_h	Hydraulic diameter of the channel.
f_{app}	Apparent friction factor for hydrodynamically developing flow.
F_d	Drag Force, N.
$f \,.\, Re_{D_h}$	The fully developed flow factor Reynolds number group.
h	Specific Enthalpy, kJ/kg. Equations (2 &4).
h	Convective heat transfer coefficient, W/ m^2K
k	Thermal conductivity of the fin, W/mK
K_c	Coefficients of a sudden contraction.
K_e	Coefficients of a sudden expansion.
k_f	Fluid thermal conductivity, W/mK
L	Heat sink length, m.
m	Fin heat transfer parameter, Equation(16)
$\dot{m}$	Mass flow rate, kg/s.
N_{u_b}	Nusselt number.
P	Perimeter of the fin, m. Equation (16).
P_r	Prandtl number.
P	Pressure, kPa.
Q	Heat dissipation rate, W.
q''	Heat transfer rate between each fin and stream, W/ m^2
R	Overall heat sink thermal resistance, K/W
Re_{D_h}	Channel Reynolds number.
Re_b	Reynolds number.
Re_b*	Modified channel Reynolds number.
R_{fin}	Thermal resistance of a single fin, K/W.
s_{in}	Specific Entropy at the inlet, kJ/kg.K.
s_{out}	Specific Entropy at the outlet, kJ/kg.K.
s	Spacing between two fins, m.
$\dot{S}_{gen}$	Entropy generation rate, W/K.
T_b	Temperature of the fin base, ^oC.
T_e	Ambient Temperature, ^oC.
T_w	Temperature of the surface of the fin, ^oC.
V_{ch}	Channel velocity, m/s.
V_f	Stream velocity, m/s.
W	Heat sink width, m.
ΔT	Temperature excess, T_b-T_e, ^oK.
ν	Kinematics viscosity of fluid, m^2/s.
ρ	Air density, kg/ m^3

ENTROPY GENERATION MINIMIZATION (EGM) OF A HEAT SINK

The idea of using an entropy generation rate to estimate the heat transfer enhancement was first proposed by Bejan [11] as a performance assessment criterion for thermal systems. A fin can generate the entropy associated with the external flow and because the fin is non-isothermal it can also generate entropy internally. As all the thermodynamic systems, the entropy in heat sink is generated from the irreversibility due to the heat transfer with finite temperature differences and the friction of fluid flow. The basic thermodynamic equations for the stream channel as an open system in steady flow are

$$\dot{m}_{in} = \dot{m}_{out} = \dot{m} \qquad (1)$$

$$\dot{m}\ h_{in} + \iint q'' dA - \dot{m}\ h_{out} = 0 \qquad (2)$$

$$\dot{S}_{gen} = \dot{m}\ s_{out} - \dot{m}\ s_{in} - \iint \frac{q'' dA}{T_w} \qquad (3)$$

The canonical $dh = T\,ds + \left(1/\rho\right) dP$ form may be written as

$$h_{out} - h_{in} = T_e\,(s_{out} - s_{in}) + \frac{1}{\rho}\,(P_{out} - P_{in}) \qquad (4)$$

It was assumed that the temperature and density do not change significantly between inlet and outlet. Combining equations (2) through (4), the entropy generation rate can be written in this form:

$$\dot{S}_{gen} = \iint_A q'' \left(\frac{1}{T_e} - \frac{1}{T_w}\right) dA - \frac{\dot{m}}{\rho T_e}\left(P_{out} - P_{in}\right) \qquad (5)$$

Knowing that:

$$\dot{m} = A_{ch}\ \rho V_f \qquad (6)$$

$$F_d = A\ \left(P_{in} - P_{out}\right) \qquad (7)$$

Obtaining

$$\dot{S}_{gen} = \frac{1}{T_e^{\,2}} \iint_A q'' \left(T_w - T_e\right) dA - \frac{1}{T_e} F_d\, V_f \qquad (8)$$

where F_d is the drag force.

Equation (8) represents the entropy generation rate caused by fin heat transfer in external flow. The fin generates entropy internally due to the fin is nonisothermal

$$\left(\dot{S}_{gen}\right)_{internal} = \iint_A \frac{q''}{T_w} dA - \frac{Q}{T_b} \qquad (9)$$

Where T_w and T_b represent the local and the base temperature respectively and Q is the heat dissipation rate of the heat sink. The heat transfer rate between the fin and the stream is q'' and theoretically a heat sink is required to satisfy an equation of the form

$$\iint q''\, dA = Q \qquad (10)$$

Adding equations (8) and (9) the entropy generation rate for a single fine can be written in this expression:

$$\dot{S}_{gen} = \frac{Q \cdot \Delta T}{T_e^{\,2}} + \frac{F_d\, V_f}{T_e} \qquad (11)$$

A uniform stream with velocity V_f and the absolute temperature T_e passes through the fin as shown in Figure 1. Fluid friction appears in the form of the drag force F_d along the direction of V_f. Equation (11) shows that fluid friction and inadequate thermal conductivity jointly contribute to degrading of the thermodynamic performance of the fin. Thus, the optimal thermodynamic size of the fin can be computed by minimizing the entropy generation rate given by equation (11) subjected to necessary design constrains.

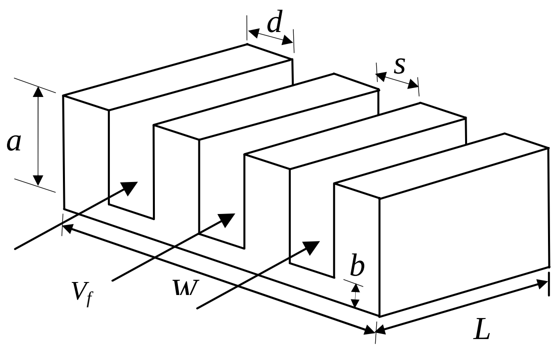

Fig.1: Schematic diagram of a plate fin.

For a heat sink set, the temperature excess of ΔT is related to the overall heat sink thermal resistance.

$$\Delta T = Q \cdot R \qquad (12)$$

where R is the overall heat sink thermal resistance. So, the entropy change rate of a heat sink set can be written as

$$\dot{S}_{gen} = \frac{Q^2\ R}{T_e^{\,2}} + \frac{F_d V_f}{T_e} \qquad (13)$$

The entropy generation rate in equation (13) is a function of both heat sink resistance and viscous dissipation. In the general thermal design of a heat sink, the goal can be either to minimize the total thermal resistance or to maximize the total efficiency. The minimization of the entropy generation rate is equivalent to the minimization of the total thermal resistance and pressure drop. Therefore, the design strategy of minimizing entropy generation rate has the same effect as maximizing the thermal efficiency, surface area, and convective coefficients. Additionally, the optimal flow velocity and viscous dissipation can be found through minimization of the entropy generation rate. In the heat sink optimization, one important implication is that since the size parameter is naturally linked directly to the volume and the weight, it should be considered as one of the design constraints in the minimization of the entropy generation rate. The overall heat sink resistance is given by

$$R = \frac{1}{\left(\frac{N}{R_{fin}}\right) + h\left(N - 1\right) s\ L} + \frac{b}{k\ LW} \qquad (14)$$

where N is the number of fins and R_{fin} is the thermal resistance of a single fin:

$$R_{fin} = \frac{1}{\sqrt{h\ P\ k\ A_c}\ \ \tanh(m\ a)} \quad (15)$$

with

$$m = \sqrt{\frac{h\ P}{k\ A_c}} \quad (16)$$

and P is the perimeter of the fin and A_c is the cross-sectional area of the fin. The total drag force on the heat sink may be obtained by considering a force balance on the heat sink. Specifically,

$$\frac{F_d}{\left(1/2\right)\rho V_{ch}^{\ 2}} = f_{app}\ N\ (2\ a\ L + s\ L) + ... + K_c\ (a\ W) + K_e\ (a\ W) \quad (17)$$

where f_{app} is the apparent friction factor for hydrodynamically developing flow, and the channel velocity V_{ch} is related to the free stream velocity by

$$V_{ch} = V_f\ \left(1 + \frac{d}{s}\right) \quad (18)$$

The apparent friction factor f_{app} for a rectangular channel may be computed using a form of the model developed by Muzychka and Yovanovich [12] for developing laminar flow:

$$f_{app}\ Re_{D_h} = \left[\left(\frac{3.44}{\sqrt{L'}}\right)^2 + \left(f\ \ Re_{D_h}\right)^2\right]^{1/2} \quad (19)$$

where

$$L' = \frac{L}{D_h\ Re_{D_h}} \quad (20)$$

and D_h is the hydraulic diameter of the channel and $f\,.Re_{Dh}$ is the fully developed flow factor Reynolds number group given by

$$f\,.Re_{D_h} = 24 - 32.527\left(\frac{s}{a}\right) .... + 46.721\left(\frac{s}{a}\right)^2 - 40.829\left(\frac{s}{a}\right)^3 ... + 22.954\left(\frac{s}{a}\right)^4 - 6.089\left(\frac{s}{a}\right)^5 \quad (21)$$

The expansion and contraction loss coefficients may be computed using the simple expressions for a sudden contraction and a sudden expansion:

$$K_c = 0.42\left(1 - \left(1 - \frac{N\ d}{W}\right)^2\right) \quad (22)$$

$$K_e = \left(1 - \left(1 - \frac{N\ d}{W}\right)^2\right)^2 \quad (23)$$

This paper adopts fully-developed and developing flow presented by Teertstra et al. [13]. So, the heat transfer coefficient h can be computed using the model developed by Teertstra et al. [13].

$$N_{u_b} = \left\{ \left(\frac{Re_b^{\ *}\ Pr}{2}\right)^{-3} + \ldots + \left(0.664\sqrt{Re_b^{\ *}}\ Pr^{1/3}\ \sqrt{1 + \frac{3.65}{\sqrt{Re_b^{\ *}}}}\right)^{-3} \right\}^{-1/3} \quad (24)$$

where

$$Re_b^{\ *} = Re_b\ \left(\frac{s}{L}\right) \quad (25)$$

$$Re_b = \frac{s\,.V_{ch}}{\upsilon} \quad (26)$$

PARTICLE SWARM OPTIMIZATION (PSO)

Particle swarm optimization is based on a relatively simple concept, and can be implemented in a few lines of computer code. It requires only simple mathematical operators, and is computationally inexpensive in terms of both memory requirement and speed. It exhibits some evolutionary computation attributes such as it is initialized with a population of random solutions, it searches for optima by updating generations, and updating is based on the previous generations. PSO does not suffer, however, from some of the difficulties of evolutionary algorithms. For example, a particle swarm system has memory, which the genetic algorithm (GA) does not have. In particle swarm optimization, individuals who fly past optima are pulled to return toward them, and knowledge of good solutions is retained by all particles [7]. While a GA can handle combinatorial optimization problems, PSO was initially used to handle continuous optimization problems. Subsequently, PSO has been expanded to handle combinatorial optimization problems and both discrete and continuous variables as well. Efficient treatment of mixed-integer nonlinear optimization problems (MINLP) is a rather difficult problem in the optimization field. Unlike other evolutionary computing (EC) techniques, PSO can be realized using only a small program, in MINLP. This feature of PSO is one of its main advantages when compared with other optimization techniques [14].

In summary, compared with other methods, PSO has the following advantages [15]:

- Faster and more efficient: PSO may get results of same quality in significantly fewer fitness and constraint evaluations.

- Better and more accurate: In demonstrations and various application results, PSO is found to give better and more accurate results than other algorithms reported in the literature.

- Less expensive and easier to implement: The algorithm is intuitive and does not need specific domain knowledge to solve the problem. There is no need for transformations or any other manipulations. Implementation in difficult optimization areas requires relatively simple and short coding.

1. The Structure of PSO

The PSO algorithm consists of the following basic components [7,8,16]:

- *Particle position vector* $\vec{x}$: This vector is containing the current location of the solution for each particle in the search space.
- *Particle velocity vector* $\vec{v}$ *:* it represents the amount in which vector $\vec{x}$ (both vectors have consistent units) will change in magnitude and direction in the next iteration. The velocity is the step size, the amount by which the changing the $\vec{v}$ values changes the direction in which the particle will move through the search space, that is, it causes the particle to make a turn. The velocity vector is used to control the range and resolution of the search.
- *Inertia weight w (t):* This is a control parameter that is used to control the impact of the previous velocities on the current velocity. Hence, it influences the trade-off between the global and local exploration abilities of particles.
- *Best solution pbest:* This is the best solution of the objective function has been discovered by a particular particle.
- *Best Global Solution gbest*: This is the best global solution of objective function has been discovered by all particles of the population.

Basically, PSO can be expressed mathematically for a given problem of *D*-dimensions i.e. D- design variables, for each particle *i* and each channel $d = [1 \ldots D]$,

$$v_{id}^{t+1} = \omega . v_{id}^{t} + \rho_1 . rand_1 \cdot \left(p_{id} \cdot x_{id}^{t} \right) + \cdots + \rho_2 . rand_2 \cdot \left(p_{gd} \cdot x_{id}^{t} \right) \quad (27)$$

$$x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1} \quad (28)$$

$$w = w_{max} - \left(\frac{w_{max} - w_{min}}{t_{max}} \times t \right) \quad (29)$$

where

- w is the inertia factor.
- v_{id} is the value of channel d in the velocity vector $\vec{v}$ for particle *i*.
- ρ_1 is the cognitive learning rate.
- ρ_2 is the social learning rate.
- $rand_1$ and $rand_2$ are random values in the range of [0..1], (accelerating) is the current position of particle *i* along dimension *d*.
- t is current iteration number.
- t_{max} is the maximum number of iterations.

2. Parameter selection

In PSO calculation, the main parameters, which have great effect on convergence and correctness of the algorithm, include:

1. Population size

Population size is related to search space scale. If population size is too small, the algorithm is easy to converge to local optimum, if the size is too large, it will need large computer memory so that it requires a long calculation time. In fact, the selected population size is problem-dependent and population size of 20-50 is most common. It was found early on that smaller populations that were common for other evolutionary algorithms (such as genetic algorithms and evolutionary programming) were optimal for PSO in terms of minimizing the total number of evaluations (population size times the number of generations) needed to obtain a sufficient solution [17].

2. Learning factors

The two learning factors ρ_1 and ρ_2 in equation (27) represent the weighting of the stochastic acceleration terms that direct each particle toward *pbest* and *gbest* positions. Early experience with particle swarm optimization led to setting the acceleration constants ρ_1 and ρ_2 each equal to 2.0 for almost all applications [18].

3. Inertia Weight Factor

The inertia weight factor was introduced by Shi and Eberhart [19,20] to control the momentum of the particle by weighing the contribution of the pervious velocity i.e. controlling how much memory or knowledge of previous flight direction will influence the new velocity. In terms of value of the inertia factor, it starts with 0.9 and decrease gradually with time until it reach 0.4.

4. Constraint of velocity

In velocity updating, v_{max} and v_{min} are two important parameters, which determine in which regions between the present position and target position are searched [21]. Usually, we set $v_{min} = -v_{max}$. If v_{max} is too high, particles may fly beyond the feasible domain. If v_{max} is too small, on the other hand, particles may not explore sufficiently beyond locally good regions. In fact, they could become trapped in local optima, unable to move far enough to reach a better position in the searching space. In general, v_{max} is often set at about 10–30% of the dynamic range of the variable on each dimension [22].

PROPOSED NEW DEVELOPMENTS:

A premature convergence to a local minimum is one of PSO weakness. Because of premature convergence, particles will not be able to do more exploration within the feasible domain and eventually particles get trapped at local optima. A new parameter has been introduced to the PSO which is called chaotic acceleration factor (C_a) to improve the speed and efficiency of the search.

CHAOTIC ACCELERATION FACTOR:

Chaos is a kind of characteristic of nonlinear systems that under specific conditions is sensitive to initial conditions. Although it appears to be stochastic, it occurs in a deterministic nonlinear system under deterministic conditions. Because chaos queues can experience all the states in a specific area without repeat, chaotic search becomes a novel tool used as an optimizer [23]. This paper is introducing a new factor is called a chaotic acceleration factor (C_a) to be added to a new position equation (28). Ca is extracted from the logistic map equation, which is one of chaotic sequences, is defined as follows:

$$C_a^{t+1} = \mu \cdot C_a^t \cdot \left(1 - C_a^t\right) \tag{30}$$

where μ is the control parameter and t is the iteration number. While the above equation is deterministic, it exhibits chaotic dynamics when $\mu = 4$ and $C_a^0 \notin \{0, 0.25, 0.5, 0.75, 1\}$ that is, it exhibits the sensitive dependence on initial conditions, which is the basic characteristic of chaos. By using C_a, the chaotic phenomena are incorporated into particle swarm optimization (PSO) to improve the quality of solutions. One of PSO weakness is that the particles have a tendency to be trapped in a local optimum. C_a will ensure that the particles be exploring the search space and not be trapped in one the local optimum. Moreover, C_a will enrich the searching behavior and improve the computational speed. C_a will be multiplying by the new velocity term in equation (28). C_a is trying to make a balance between global exploration and local exploitation.

$$x_{id}^{t+1} = x_{id}^{t} + C_a^{t+1} \cdot v_{id}^{t+1} \tag{31}$$

NUMERICAL SIMULATION

1. VALIDITY OF THE PROPOSED METHOD

To validate the proposed algorithm, it is firstly used to solve four non-linear functions. The first function is the Sphere function described by equation (32):

$$f(x) = \sum_{i=1}^{D} x_i^2 \qquad -5.12 \le x_i \le 5.12 \qquad D = 30 \tag{32}$$

The second function is Rosenbrock's function described by equation (33):

$$f(x) = 100\,(x_1^2 - x_2^2)^2 + (1 - x_1)^2 \qquad -2.048 \le x_i \le 2.048 \tag{33}$$

The third function is Brainin's function described by equation (34):

$$f(x_1, x_2) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x_1 + 10$$

$$x_1 \in [-5, 10] \quad , x_2 \in [0, 15] \tag{34}$$

The fourth function is Schaffer's function described by equation (35):

$$f(x) = 0.5 - \frac{\left(\sin\sqrt{x_1^2 + x_2^2}\right) - 0.5}{\left(1.0 + 0.001\left(x_1^2 + x_2^2\right)\right)^2} \qquad x_i \in \{-10, 10\} \tag{35}$$

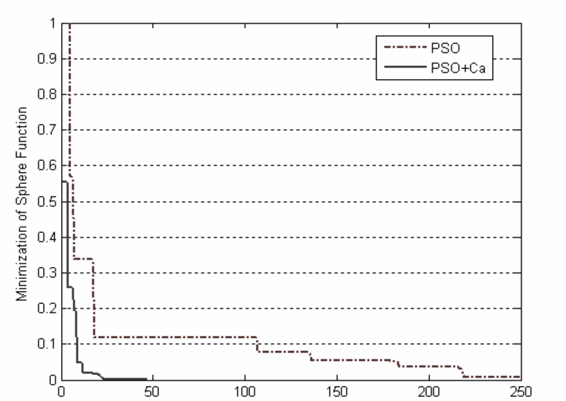

Fig.2: Minimization of Sphere Function Using PSO and PSO+C_a.

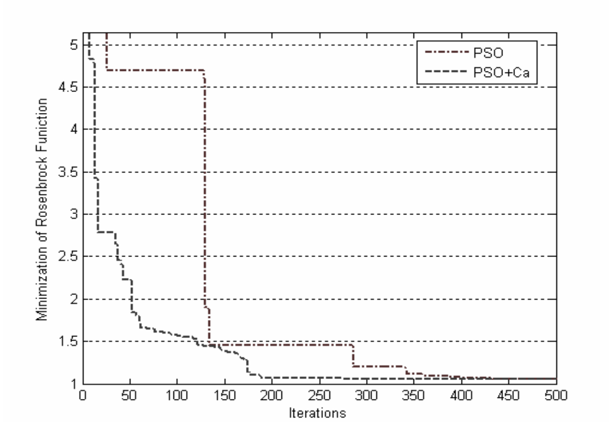

Fig.3: Minimization of Rosenbrock Function Using PSO and PSO+C_a.

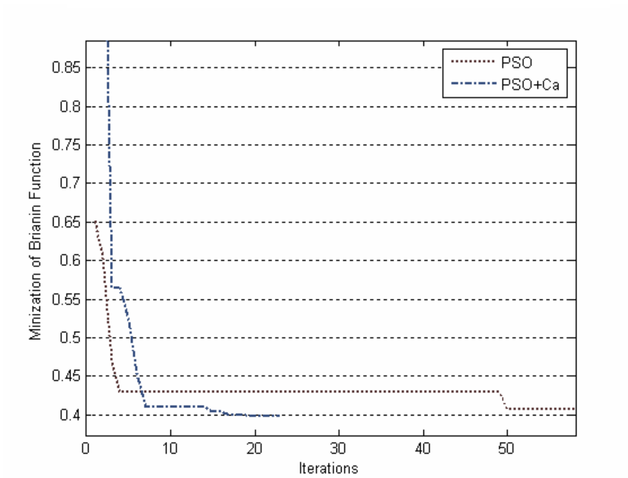

Fig.4: Minimization of Brainin's Function Using PSO and PSO + C_a.

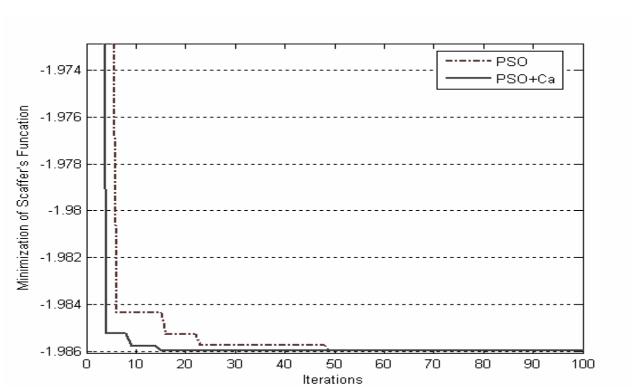

Fig.5: Minimization of Schaffer's Function Using PSO and PSO + C_a.

As shown in Fig 2-5, in general, both PSO's algorithms successfully had reached the global solution for each test function within the allowable generations. For all test functions, the modified PSO algorithm showed better convergence performance over the original PSO algorithm. However, the modified PSO algorithm converged to the minimum at earlier generations the original PSO algorithm did so that the modified PSO algorithm has faster convergence rate than the original algorithm when the both algorithms solved same number of evaluations of each fitness functions.

In the experiment, the modified and original PSO algorithms were implemented in MATLAB 7.1 and were run at a computer with processor Pentium 4 CPU 3.20GHz with 1GB of RAM. Computing CPU time was recorded for each running trial of PSO algorithms. The average computing time of modified PSO algorithm for all test functions is 10.5s, 15.8s, 60.1s, 18.2s, and 32s respectively. Whereas the average computing time of original PSO algorithm for all test functions is 79.6s, 67.6s, 110.1s, 64s, and 65.6s respectively.

Function	Optimum	PSO	PSO+C_a
Sphere	0	204	52
Rosenbrock	0	1203	374
Brainin	0.3977	704	101
Schaffer	0	696	169

Table1: the optimum solution for test function and number of iteration needed by both PSO's algorithms to reach the optimum solution.

The numerical simulation results suggest that the modification strategy proposed can well accelerate the convergence rate of the PSO algorithm as it is shown in Table1.

When a population based optimization method is applied to solve a real world problem. A trade-off has to be made between the algorithm convergence rate and the solution precision. The modified PSO algorithm proposed in this paper can successfully satisfy both convergence rate and solution precision. This feature is very attractive to real world problem.

2. HEAT SINK DESIGN OPTIMIZATION

The goal is to get the optimal number of fins N, optimum height of fins a, optimum thickness of each fin d, and the optimum flow velocity of cooling flow V_f. The objective function is

$$\dot{S}_{gen} = \frac{Q^2 R}{{T_e}^2} + \frac{F_d V_f}{T_e} \tag{36}$$

Fig.1 shows the geometrical configuration of a plate-fin sink with horizontal inlet cooling flow. Configuration data is as follows:

- Both the base length L and the width W are 50 mm.
- The total heat dissipation of 30 W is uniformly applied over the base plate of the heat sink with a base thickness *b* of 2 mm.

- The thermal conductivity of the heat sink k is 200 W/m.K.
- The ambient air temperature T_e is 25 C.
- The conductivity of the air k_f is 0.0267 W/m.K.
- The air density ρ is 1.177 kg/m.
- The kinematic viscosity coefficient ν is 1.6 (10^{-5}) m^2/s.
- The design variables $[x_1, x_2, x_3, x_4]^T = [N, a, d, V_f]$.
- The design boundaries corresponding to each design variable are

$$2 \le x_1 \le 40 \qquad (37)$$

$$25 \text{ mm} \le x_2 \le 140 \text{ mm} \qquad (38)$$

$$0.2 \text{ mm} \le x_3 \le 2.5 \text{ mm} \qquad (39)$$

$$0.5 \ {}^{m}\!/\!{}_{s} \le x_4 \le 2 \ {}^{m}\!/\!{}_{s} \qquad (40)$$

The number of fins must be an integer that can be restricted in the following domain:

$$2 \le N \le int\left[1 + \left(\frac{W - d}{d}\right)\right] \qquad (41)$$

The spacing s between two fins is equal to the following:

$$s = \left(\frac{W - d}{N - 1}\right) - d \qquad (42)$$

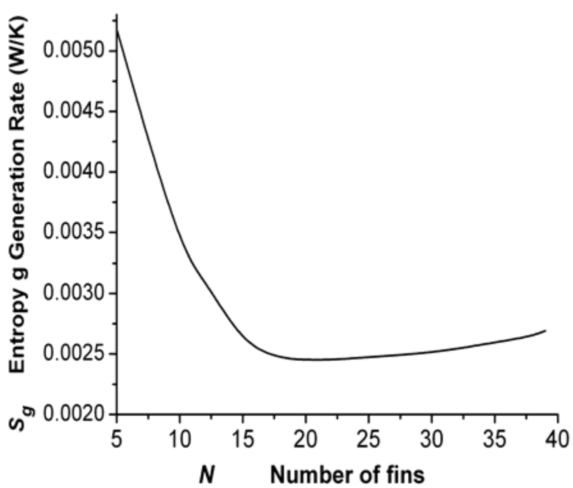

Fig.6: Optimum Entropy Generation Rate with vary of *N*

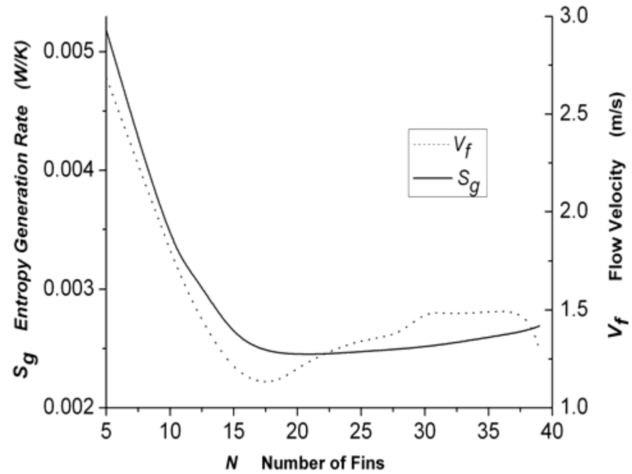

Fig.7: Optimum Entropy Generation Rate and Optimum flow Velocity with different values of *N*

Fig.6-8 show the parametric optimization of heat sink as result of applying the proposed PSO algorithm to minimize the entropy generation rate. As N the number of fins increases the entropy generation rate will decrease dramatically due to increasing the surface area of the heat sink. When N reaches 22 fins and up then the entropy generation rate will start increasing gradually and the search will start go away from the optimal solution location. As it is clear from the figure the optimal solution location is located N values vary between 19 and 22 fins. The optimal solution of the entropy generation rate is 0.002504 W/K. Fig. 7-8 show that behaviors of some of design variables (V_f = Flow Velocity, d = Thickness of Fin) along with optimization process of the entropy generation rate for different values of *N*.

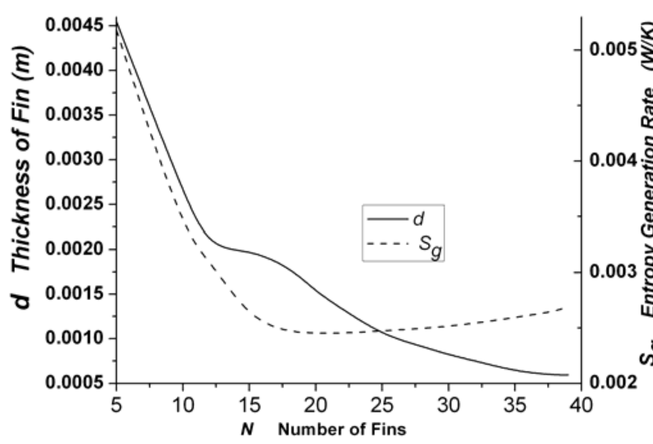

Fig.8: Optimum Entropy Generation Rate and Optimum Thickness of Fin with different values of *N*

For a comparative evaluation between the proposed PSO and a deterministic design optimization, Work of Shih and Liu [3] considered here,. They had applied augmented Lagrange multiplier (ALM) method to minimize the entropy generation rate of heat sink which had the same configurations that had been used above.

	N	a (mm)	d (mm)	V_f (m/s)	s (mm)	$\dot{S}_{gen}$ (W/K)	V_{oL} (mm^3)
Current Work	21	106	1.4	1.25	1.2	0.002504	155820
Shih&Liu	20	134	1.61	1.05	.9368	0.002967	220740

Table 2: Results obtained in this work and the paper by Shih and Liu.

Table.2 shows the results that were obtained by applying the proposed PSO method and ALM method. The last column shows the total structural volume of the heat sink indicated as V_{oL} (mm^3). The larger value of V_{oL} indicates the further structural mass required to manufacture the heat sink. The following conclusions can be made from Table 2:

- The height of fins *a* and the total structural volume of the heat sink V_{oL} are reduced by 30%. This implies that less material is needed to produce the heat sink, which may in turn reduce the cost of the cooling system.
- The entropy generation rate $\dot{S}_{gen}$ is slightly lower than the results obtained by Shih & Liu, which means that the overall thermal resistance is decreased.
- The thickness of each fin d and the spacing s between two fins are slightly lower than what has been reported by Shih and Liu.
- The free stream velocity V_f is 20% higher, which means more pumping power is required, and that will increase the running cost of the cooling system.

CONCLUSION

In general, the particle swarm optimization (PSO) algorithm has some superior features such as:

- PSO algorithms use objective function information to guide the search in the problem space. Therefore, PSO can easily deal with non-differentiable and non-convex objective functions. Additionally, this property relieves PSO algorithms of assumptions and approximations, which are often required by traditional optimization methods.
- PSO algorithms use probabilistic rules for particle movements, not deterministic rules. Hence, PSO algorithms are a type of stochastic optimization algorithm that can search a complicated and uncertain area. This makes PSO algorithms more flexible and robust than conventional methods.

However, improving convergence of the PSO algorithm is an important objective when solving complex real world problems is considered. In this paper, a novel chaotic acceleration term is added into original PSO algorithm to improve its convergence performance. The proposed PSO algorithm was studied and compared with the original PSO algorithm by using a suit of four well-known test functions. Next, the applicability of the proposed PSO algorithm to the optimal heat sink design has been investigated. The proposed PSO process was presented for the design of a plate-fin heat sink, with the objective of maximum dissipation of the heat generated from electronic components. The entropy generation rate was used in the fitness function, to realize the highest heat transfer efficiency. A practical application was presented as an illustrative example. A comparative evaluation was made with respect to results available in the literature.

REFERENCES

[1] Bar-Cohen, A., "Least-Energy Optimization of Air-Cooled Heat Sinks for Sustainable Development," *IEEE Transactions on Components and Packaging Technologies*, Vol. 26, No. 1, pp.16-25, March 2003.

[2] Ogiso, K., "Assessment of Overall Cooling Performance in Thermal Design of Electronics Based on Thermodynamics," *Journal of Heat Transfer*, Vol. 123, pp.999-1005, October 2001.

[3] Shih, C.J., and Liu,G.C., "Optimal Design Methodology of Plate-Fine Heat Sinks for Electronic Cooling Using Entropy Generation Strategy," *IEEE Transactions on Components and Packaging Technologies*, Vol. 27, No. 3, pp.551-559, Sep. 2004.

[4] Lyergar, M.K., *Resource Constrained Heat Sink Optimization*, PhD Thesis, University of Minnesota, MN, 2002.

[5] Wei, X., and Joshi, Y., "Optimization Study of Stacked Micro-Channel Heat Sinks for Micro-Electronic Cooling," *IEEE Inter Society Conference on thermal Phenomena*, San Diego, CA, pp. 441-448, May 2002.

[6] Kennedy, J. and Eberhart, R. C., "Particle swarm optimization," *Proceedings of IEEE International Conference on Neural Networks*, Perth, Australia, pp. 1942-1948, 1995.

[7] Eberhart, R. C. and Kennedy, J., "A new optimizer using particle swam theory," *Proceedings of the Sixth International Symposium on Micromachine and Human Science*, Nagoya, Japan, pp. 39-43, 1995.

[8] Abido, M.A., "Optimal power flow using particle swarm optimization," *International Journal of Electric Power and Energy Systems*, Vol. 24, pp. 563-571, 2002.

[9] Ren, F.M., and Li, L.J.,"Application of particle swarm optimization method in engineering structures," *Journal of Harbin Institute of Technology (New Series)*, v 14, n SUPPL. 2, January, 2007, p 142-145

[10] Mikki, Said M., and Kishk, Ahmed A.,"Quantum particle swarm optimization for electromagnetic," *IEEE Transactions on Antennas and Propagation*, v 54, n 10, October, 2006, p 2764-2775

[11] Bejan, A., *Energy Generation Minimization*, Orlando, FL, CC Press, 1995.

[12] Muzychka, Y.S., and Yovanovich, M.M., "Modeling Friction Factors in Noncircular ducts for Developing Laminar Flow," *Proceedings 2nd AIAA Theoretical Fluid Mechanics Meetings*, Albuquerque, NM, June 15-18,1998.

[13] Teertstra, P., Yovanovich, M.M., Culham, J.R., and Lemczyk, "Analytical Forced Convection Modeling of Plate Fin Heat Sink," *Proceedings 15th IEEE SEMI-THERM Symposium*, San Diego, CA, 1999.

[14] Fukuyama, Y., "Fundamentals of Particle Swarm Techniques," In Lee K.Y. and El-Sharkawi M.A. (eds.), *Modern Heuristic Optimization techniques with Applications to Power Systems*, IEEE Power Engineering Society , pp.45-51, 2003.

[15] Xiaohui, H., Eberhart, R.C., and Yuhui, S., “Engineering optimization with particle swarm,” *Proceedings of the 2003 IEEE Swarm Intelligence Symposium* , Indianapolis, IN, pp. 53-57, April 2003.

[16] Carlisle, A.J., *Applying the Particle Swarm Optimizer to Non-Stationary Environments*, PhD Thesis, Auburn University, Auburn, AL, 2002.

[17] Karray, F.O. and de Silva, C.W., *Soft Computing and Intelligent Systems Design—Theory, Tools, and Applications*, Addison Wesley, Harlow, U.K., 2004.

[18] Alrasheed, M.R., de Silva, C.W. and Gadala, M.S., "Evolutionary optimization in the design of a microchannel heat sink," in the book *MECHATRONIC SYSTEMS Devices, Design, Control, Operation, and Monitoring*, CRC Press, Taylor & Francis, Boca Raton, FL (In Press 2007).

[19] Shi, Y., and Eberhart, R.C., “A modified particle swarm optimizer,” *Proceedings of the IEEE International Conference on Evolutionary Computation* Piscataway NJ: IEEE Press (1998), pp. 69–73.

[20] Shi Y, Eberhart RC. , “Parameter selection in particle swarm optimization”, In: Evolutionary Programming VII. Lecture Notes in Computer Science, vol. 1447. Berlin: Springer; 1998. p. 591–600.

[21] Lee, K. Y. and El-Sharkawi, M.A (Eds.), *Modern Heuristic Optimization Techniques with Applications to Power Systems*, IEEE Power Engineering Society, 2002.

[22] Eberhart, R.C., and Yuhui, Shi, “Particle swarm optimization: developments, applications and resources,” *Proceedings of the 2001 Congress on Evolutionary Computation*, Vol. 1, Seoul, South Korea, pp. 81-86, May 2001.

[23] Coelho, Ld.S., and Mariani, V.C., “ Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect,” *IEEE Transactions on Power Systems*, vol. 21, n 2, May 2006, 989-96