

# A New Bio-inspired Approach to the Traveling Salesman Problem

Xiang Feng<sup>1</sup>, Francis C.M. Lau<sup>2</sup>, and Daqi Gao<sup>1</sup>

<sup>1</sup> East China University of Science and Technology, Shanghai 200237, China

<sup>2</sup> The University of Hong Kong, Hong Kong

xfeng@ecust.edu.cn, fcmlau@cs.hku.hk, gaodaqi@ecust.edu.cn

**Abstract.** The host-seeking behavior of mosquitoes is very interesting. In this paper, we propose a novel mosquito host-seeking algorithm (MHSA) as a new branch of biology-inspired algorithms for solving TSP problems. The MHSA is inspired by the host-seeking behavior of mosquitoes. We present the mathematical model, the algorithm, the motivation, and the biological model. The MHSA can work out the theoretical optimum solution, which is important and exciting, and we give the theoretical foundation and present experiment results that verify this fact.

**Keywords:** Bio-inspired algorithm, traveling salesman problem (TSP), mosquito host-seeking algorithm (MHSA), distributed and parallel algorithm.

## 1 Introduction

Some combinatorial optimization problems (COP) such as the traveling salesman problem (TSP) belong to a family of NP-complete problems [1] whose computational complexity increases exponentially with the number of parameters. The TSP is probably the best known COP and is NP-complete. The TSP is the problem to find the shortest path or circuit through  $n$  cities. The TSP and its variants have many practical applications—e.g., robot control, mobile computing, vehicle routing, automated guided vehicles scheduling, integrated circuits design, X-ray crystallography. Therefore, the TSP has attracted many researchers from different fields such as artificial intelligence, operations research, mathematics, physics, biology, etc. Although the TSP is easily formulated, it exhibits all aspects of COP as well as has served and continues to serve as the benchmark problem for new algorithmic ideas.

During the last decades, many algorithmic strategies have been proposed for finding near-optimum solutions to the TSP, including genetic algorithm (GA) [2], evolution strategies [3], evolutionary programming [4], ant colony optimization (ACO) [5], particle swarm optimization (PSO) [6], artificial immune system (AIS) [7], artificial neural network (ANN) [8], cellular automaton, generalized cellular automaton, cellular neural network, reaction and growth model, simulated annealing algorithm (SAA) [9] and elastic net (EN) [10]. Nature-inspired

algorithms (NAs) are among the best algorithms to solve dynamic TSPs nowadays. The algorithms mentioned above are all NAs, of which SAA and EN are physics-inspired algorithms, and the others are bio-inspired algorithms. There are some common features shared by all NAs:

- They draw from observations of physical processes that occur in nature.
- They mimic different natural systems and processes using mathematical models and algorithms.
- They have inherent parallelism.
- They are mainly used to solve COPs, such as the TSP.
- COPs are optimized using the NAs step by step (iteration).

In this paper, by means of the TSP, we will propose a new algorithm, called Mosquito Host-Seeking Algorithm (MHSA), which is a bio-inspired computing technique inspired by host-seeking behavior of mosquito swarms. We want to develop the MHSA as a new, biology-based, nature-inspired approach that has a substantive theoretical foundation, as well as the common features of NAs as mentioned above. We hope the MHSA can describe complex, high-dimensional, highly nonlinear, micro-evolutionary and random behaviors and dynamics arising from interactions of entities. It turns out that the MHSA is easy to use, and its simplicity implies greater flexibility, making the approach very easy to adapt to a wide range of optimization problems besides the TSP.

**Table 1.** The main notations in TSP

Notations	Meanings	Related to
$n$	total number of TSP cities	problem
$C_i$	the $j$ -th city ( $i = 1, n$ )	problem
$(x_i, y_i)$	coordinates of city $C_i$	problem
$d_{ij}$	distance between city pair $(C_i, C_j)$ ( $i, j = \overline{1, n}$ ) $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$	problem
$p_{ij}$	path between $C_i$ and $C_j$	
$x_{ij}$	if $x_{ij} = 1$ , then $p_{ij}$ can be passed; or else $x_{ij} = 0$ , then $d_{ij} = \infty$ , $p_{ij}$ cannot be passed	problem
$Z$	the shortest path through $n$ cities	solution
$r_{ij}$	(1)if $r_{ij} = 1$ , then $Z$ pass $p_{ij}$ ; or else $r_{ij} = 0$ , then $Z$ not pass $p_{ij}$ . (2)when $x_{ij} = 0$ , then $r_{ij} = 0$ .	solution

## 2 Problem Description for TSP

The TSP is to find the shortest path through  $n$  cities. The main notations in any TSP instance and the TSP matrix  $\mathbb{M}(t)$  are shown in Table 1 and Table 2, respectively.

Mathematical description for TSP is as follows:

$$\begin{aligned}
 \text{Minimize: } Z &= \sum_{i,j} d_{ij} \cdot r_{ij} \\
 \text{s.t. } & 1. r_{ij} \in \{0, 1\}; \\
 & 2. r_{ij} = r_{ji};
 \end{aligned}$$

$$\begin{aligned}
 3. \sum_i r_{ij} &= 2; \\
 4. \sum_j r_{ij} &= 2.
 \end{aligned}$$

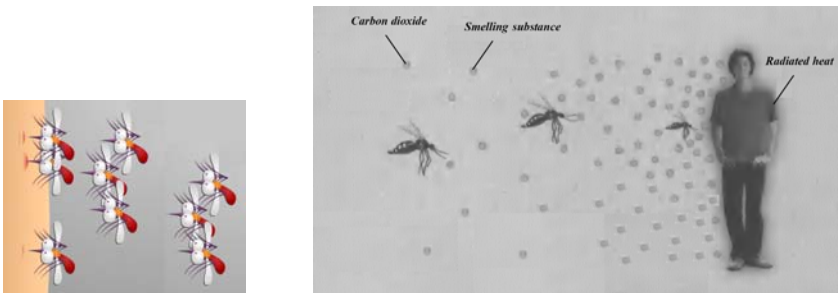
**Table 2.** TSP Matrix  $\mathbb{M}(t)$

Cities	$C_1$	...	$C_j$	...	$C_n$
$C_1$	$r_{11}, x_{11}, d_{11}$	...	$r_{1j}, x_{1j}, d_{1j}$	...	$r_{1n}, x_{1n}, d_{1n}$
$\vdots$	$\vdots$		$\vdots$		$\vdots$
$C_i$	$r_{i1}, x_{i1}, d_{i1}$	...	$r_{ij}, x_{ij}, d_{ij}$	...	$r_{in}, x_{in}, d_{in}$
$\vdots$	$\vdots$		$\vdots$		$\vdots$
$C_n$	$r_{n1}, x_{n1}, d_{n1}$	...	$r_{nj}, x_{nj}, d_{nj}$	...	$r_{nn}, x_{nn}, d_{nn}$

### 3 Mosquito Host-Seeking Model

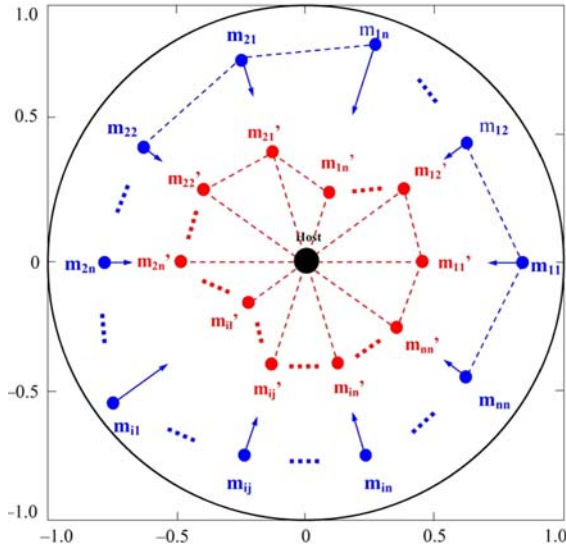
In the real world, both male and female mosquitoes are nectar feeders, but the female of many species is also capable of drinking blood. That is, only female mosquitoes search the host (object) to attack. Female mosquitoes hunt their blood host by detecting carbon dioxide ( $CO_2$ ) and 1-octen-3-ol ( $C_8H_{16}O$ ) from a distance. Female mosquitoes have host-seeking behavior, but male ones do not. As shown in Fig. 1, a swarm of mosquitoes are randomly searching the host (object) to attack. We can summarize three steps of mosquitoes host-seeking behavior as follows (see Fig. 2):

1. The mosquito looks for carbon dioxide or smelling substance;
2. It distinguishes the smell it loves, and then seeks towards a high-concentration location;
3. It makes a descent when it feels the radiated heat of the host.



**Fig. 1.** A swarm of mosquitoes **Fig. 2.** Steps of mosquitoes host-seeking behavior

The MHSA treats every entry of the TSP matrix  $\mathbb{M}(t)$  as an artificial mosquito,  $m_{ij}$ . The  $n$ -city TSP solving process is hence transformed into host-seeking behavior of a swarm of artificial mosquitoes ( $n \times n$  mosquitoes). The  $m_{ij}$ 's form the matrix  $\mathbb{M}(t)$ . For convenience, we let  $m_{ij}$  represent both an entry of the matrix as well as its corresponding artificial mosquito.



**Fig. 3.** Mosquito host-seeking model

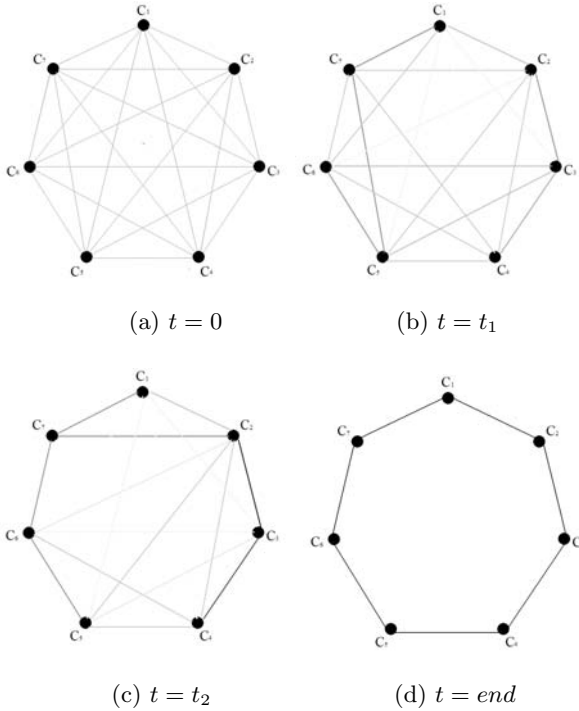
As shown in Fig. 3, all artificial mosquitoes (a swarm) are evenly distributed at an even radian surrounding a host. Each artificial mosquito is being attracted to seek towards the host by carbon dioxide, smelling substance, and radiated heat. The radial distance between an artificial mosquito and the host represents the corresponding artificial mosquito’s personal utility—that is, an artificial mosquito’s success value of host-seeking. The higher the consistency of carbon dioxide and smelling substance, the faster the artificial mosquitoes try to move towards the host. When all artificial mosquitoes stop moving, being in an equilibrium state, we arrive at an optimum solution to the TSP.

In more details, each artificial mosquito  $m_{ij}$  has a grayscale value  $r_{ij}$ , which will constantly change between 0 and 1 as the artificial mosquito moves. When all mosquitoes are in an equilibrium state, all  $r_{ij}$  will be 1 or 0.  $r_{ij} = 1$  represents the artificial mosquito  $m_{ij}$  attacking the host, as well as the path  $p_{ij}$  being black (the shortest path  $Z$  passes through this path). On the contrary,  $r_{ij} = 0$  represents that  $m_{ij}$  did not attack the host, as well as  $p_{ij}$  being white ( $Z$  does not pass through the path).

In addition, each artificial mosquito  $m_{ij}$  has a sex value  $x_{ij}$ .  $x_{ij} = 1$  represents  $m_{ij}$  is female, and  $x_{ij} = 0$  represents  $m_{ij}$  is male. Only grayscale values  $r_{ij}$  of female artificial mosquitoes will change with the motion of  $m_{ij}$ .  $r_{ij}$  of male artificial mosquitoes will always be 0—that is, the corresponding path  $p_{ij}$  will always be white.

$$\text{Motion of } m_{ij} \stackrel{(1)}{\Leftrightarrow} u_{ij} \stackrel{(2)}{\Leftrightarrow} r_{ij} \stackrel{(3)}{\Leftrightarrow} Z.$$

(1)(2)(3): MHS mathematical model and MHS algorithm. They will be proposed in the next section.



**Fig. 4.** An example of TSP (7 cities)( $t_1 < t_2$ )

The solving process of an example TSP using MHSA is shown in Fig. 4. The grayscales of all the paths between two cities change in parallel during the computing process.

For the example TSP in Fig. 4, the solution is  $Z: C_1 \rightarrow C_2 \rightarrow C_3 \rightarrow C_4 \rightarrow C_5 \rightarrow C_6 \rightarrow C_7 \rightarrow C_1$ .

$r_{12} = r_{21} = 1, r_{23} = r_{32} = 1, r_{34} = r_{43} = 1, r_{45} = r_{54} = 1, r_{56} = r_{65} = 1, r_{67} = r_{76} = 1, r_{71} = r_{17} = 1$ , the other  $r_{ij} = 0$ .

### 4 Mathematical Model of MHS

We define in this section the mathematical model of MHS for the TSP that involves  $n$  cities.

Firstly, we introduce a variable  $c_{ij}$ , which is weight of city pair  $(C_i, C_j)$ .

- When  $t = 0, c_{ij}(t = 0) = \max_{i,j} d_{ij} - d_{ij}$ ;
- When  $t > 0, c_{ij}(t > 0) \in [0, 1]$ .

In the mosquito host-seeking model,  $c_{ij}$  represents the relative strength of artificial mosquito  $m_{ij}$ . The larger  $c_{ij}$ , the stronger the artificial mosquito  $m_{ij}$ , and the easier it seeks and attacks the host.

$c_{ij}$  and  $r_{ij}$  will evolve with the motion of artificial mosquitoes.

Let  $u_{ij}(t)$  be the radial distance between an artificial mosquito  $m_{ij}$  and the host at time  $t$ , and let  $J(t)$  be the utility sum of all artificial mosquitoes, which we define as follows.

$$u_{ij}(t) = \exp ( -c_{ij}(t)r_{ij}(t)x_{ij}(t) ) ;$$

$$J(t) = \sum_{i=1}^n \sum_{j=1}^n u_{ij}(t) \tag{1}$$

At time  $t$ , the attraction function,  $P(t)$ , which is caused by the host is defined by

$$P(t) = \epsilon^2 \ln \sum_{i=1}^n \sum_{j=1}^n \exp[-u_{ij}^2(t)/2\epsilon^2] - \epsilon^2 \ln nn \tag{2}$$

where  $0 < \epsilon < 1$ . The larger  $P(t)$  is, the better.

The attraction of host causes the artificial mosquitoes to move to increase the corresponding artificial mosquitoes' minimal personal utility.

At time  $t$ , artificial mosquitoes' interaction behavior function,  $Q(t)$ , is defined by

$$Q(t) = \sum_{i=1}^n | \sum_{j=1}^n r_{ij}(t)x_{ij}(t) - 2 |^2 - \sum_{i,j} \int_0^{u_{ij}} \{ [1 + \exp(-10x)]^{-1} - 0.5 \} dx \tag{3}$$

Artificial mosquitoes can move towards the host along their own radial orbit under the the influence of these factors:

- the personal host-seeking behavior,
- the aggregate host-seeking behavior,
- the attraction of the host,
- the motion that is related to social coordinations among the swarm of artificial mosquitoes.

The four kinds of attraction factors can all contribute to the artificial mosquitos' movements towards the host. What is more, these factors produce hybrid attraction forces of the host. The general hybrid attraction function for artificial mosquito  $m_{ij}$ ,  $E_{ij}(t)$ , can be defined by

$$E_{ij}(t) = -\lambda_1 u_{ij}(t) - \lambda_2 J(t) - \lambda_3 P(t) - \lambda_4 Q(t) \tag{4}$$

where  $0 < \lambda_1, \lambda_2, \lambda_3, \lambda_4 < 1$ .

In order to dynamically solve the TSP, the artificial mosquito  $m_{ij}$  may alternately modify  $r_{ij}$  and  $c_{ij}$ , respectively, as follows:

$$dc_{ij}(t)/dt = -\lambda_1 \frac{\partial u_{ij}(t)}{\partial c_{ij}(t)} - \lambda_2 \frac{\partial J(t)}{\partial c_{ij}(t)} - \lambda_3 \frac{\partial P(t)}{\partial c_{ij}(t)} - \lambda_4 \frac{\partial Q(t)}{\partial c_{ij}(t)} \tag{5}$$

$$dr_{ij}(t)/dt = -\lambda_1 \frac{\partial u_{ij}(t)}{\partial r_{ij}(t)} - \lambda_2 \frac{\partial J(t)}{\partial r_{ij}(t)} - \lambda_3 \frac{\partial P(t)}{\partial r_{ij}(t)} - \lambda_4 \frac{\partial Q(t)}{\partial r_{ij}(t)} \tag{6}$$

where  $\frac{\partial Q(t)}{\partial u_{ij}(t)} = -\{ [1 + \exp(-10(t)u_{ij}(t))]^{-1} - 0.5 \}$ .

The motion equations for artificial mosquito  $m_{ij}$  are defined by

$$\begin{cases} du_{ij}(t)/dt = \Psi_1(t) + \Psi_2(t) \\ \Psi_1(t) = -u_{ij}(t) + \gamma v_{ij}(t) \\ \Psi_2(t) = [-\lambda_1 - \lambda_2 \frac{\partial J(t)}{\partial u_{ij}(t)} - \lambda_3 \frac{\partial P(t)}{\partial u_{ij}(t)} - \lambda_4 \frac{\partial Q(t)}{\partial u_{ij}(t)}] \{ [\frac{\partial u_{ij}(t)}{\partial r_{ij}(t)}]^2 + [\frac{\partial u_{ij}(t)}{\partial c_{ij}(t)}]^2 \} \end{cases} \tag{7}$$

where  $\gamma > 1$ . And  $v_{ij}(t)$  is a piecewise linear function of  $u_{ij}(t)$  defined by

$$v_{ij}(t) = \begin{cases} 0 & \text{if } u_{ij}(t) < 0 \\ u_{ij}(t) & \text{if } 0 \leq u_{ij}(t) \leq 1 \\ 1 & \text{if } u_{ij}(t) > 1, \end{cases} \tag{8}$$

### 5 The Parallel MHS Algorithm for TSP

Given the coordinates of  $n$  cities and  $x_{ij}$ , our mosquito host-seeking algorithm (MHSA) is as follows.

1. Initialize the number of artificial mosquitoes to be  $n \times n$ . Initialize the sex of all artificial mosquitoes to be  $x_{ij}$ . Initialize all grayscale values  $r_{ij}$  of artificial mosquitoes  $m_{ij}$  to be the average values. Initialize all weight  $c_{ij}$  to be  $d_{ij}$ . Select the related coefficients  $\epsilon, \lambda_1, \lambda_2, \lambda_3, \lambda_4$  to be 0.8, 0.05, 0.05, 0.9, 0.9, respectively.
2. Calculate the  $u_{ij}(t)$  of each  $m_{ij}$  by Equation (1) in parallel, and  $du_{ij}(t)/dt$ .
3. For each artificial mosquito  $m_{ij}$ , calculate  $dr_{ij}(t)$  by Equation (6) and  $dc_{ij}(t)$  by Equation (5) in parallel.
4. For each artificial mosquito  $m_{ij}$ , update in parallel the grayscale value  $r_{ij}(t)$  by  $r_{ij}(t + 1) = r_{ij}(t) + dr_{ij}(t)/dt$ ; update in parallel the weight  $c_{ij}(t)$  by  $c_{ij}(t + 1) = c_{ij}(t) + dc_{ij}(t)/dt$ .
5. If all  $du_{ij}(t)/dt = 0$ , then finish successfully. Otherwise, go to step 2.

### 6 Proving the MHS Model

In the following, we derive some formal properties of the mathematical model presented above.

**Proposition 1.** *Updating the weights  $c_{ij}$  and grayscale values  $r_{ij}$  by Eq. (5) and Eq. (6) respectively amounts to changing the speed of artificial mosquito  $m_{ij}$  by  $\Psi_2(t)$  of Eq. (7).*

Denote the  $j$ -th terms of Eq. (5) and Eq. (6) by  $\langle \frac{dc_{ij}(t)}{dt} \rangle_j$  and  $\langle \frac{dr_{ij}(t)}{dt} \rangle_j$ , respectively. When  $r_{ij}$  is updated according to (6), the first and second terms of (6) will cause the following speed increments of the artificial mosquito  $m_{ij}$ , respectively:

$$\langle du_{ij}(t)/dt \rangle_1^r = \frac{\partial u_{ij}(t)}{\partial r_{ij}(t)} \langle \frac{dr_{ij}(t)}{dt} \rangle_1 = \lambda_1 [\frac{\partial u_{ij}(t)}{\partial r_{ij}(t)}]^2; \tag{9}$$

$$\begin{aligned} \langle du_{ij}(t)/dt \rangle_2^r &= \frac{\partial u_{ij}(t)}{\partial r_{ij}(t)} \langle \frac{dr_{ij}(t)}{dt} \rangle_2 = \lambda_2 \frac{\partial u_{ij}(t)}{\partial r_{ij}(t)} \frac{\partial J(t)}{\partial r_{ij}(t)} \\ &= \lambda_2 \frac{\partial u_{ij}(t)}{\partial r_{ij}(t)} \frac{\partial J(t)}{\partial u_{ij}(t)} \frac{\partial u_{ij}(t)}{\partial r_{ij}(t)} = \lambda_2 \frac{\partial J(t)}{\partial u_{ij}(t)} [\frac{\partial u_{ij}(t)}{\partial r_{ij}(t)}]^2 \end{aligned} \tag{10}$$

Similarly, the third and the fourth term of Eq. (6) will cause the following speed increments of the artificial mosquito  $m_{ij}$ :

$$\begin{aligned} \langle du_{ij}(t)/dt \rangle_3^r &= -\lambda_3 \frac{\partial P(t)}{\partial u_{ij}(t)} \left[ \frac{\partial u_{ij}(t)}{\partial r_{ij}(t)} \right]^2; \\ \langle du_{ij}(t)/dt \rangle_4^r &= -\lambda_4 \frac{\partial Q(t)}{\partial u_{ij}(t)} \left[ \frac{\partial u_{ij}(t)}{\partial r_{ij}(t)} \right]^2. \end{aligned}$$

Similarly, for Eq. (5), we have  $\langle du_{ij}(t)/dt \rangle_j^c, j = 1, 2, 3, 4$ . We thus obtain

$$\begin{aligned} &\sum_{j=1}^4 [\langle du_{ij}(t)/dt \rangle_j^c + \langle du_{ij}(t)/dt \rangle_j^r] \\ &= [-\lambda_1 - \lambda_2 \frac{\partial J(t)}{\partial u_{ij}(t)} - \lambda_3 \frac{\partial P(t)}{\partial u_{ij}(t)} - \lambda_4 \frac{\partial Q(t)}{\partial u_{ij}(t)}] \{ [\frac{\partial u_{ij}(t)}{\partial r_{ij}(t)}]^2 + [\frac{\partial u_{ij}(t)}{\partial c_{ij}(t)}]^2 \} = \Psi_2(t). \end{aligned}$$

Therefore, updating  $c_{ij}^{(j)}$  and  $r_{ij}^{(j)}$  by (5) and (6), respectively, gives rise to the speed increment of artificial mosquito  $m_{ij}$  that is exactly equal to  $\Psi_2(t)$  of Eq. (7).

**Proposition 2.** *The first and second terms of Eqs. (5) and (6) will enable the artificial mosquito  $m_{ij}$  to move towards the host, that is, the personal utility of the artificial mosquito  $m_{ij}$  increases, in direct proportion to the value of  $(\lambda_1 + \lambda_2)$ .*

According to Eqs. (9) and (10), the sum of the first and second terms of Eq. (5) and (6) will be

$$\begin{aligned} &\langle du_{ij}(t)/dt \rangle_1^r + \langle du_{ij}(t)/dt \rangle_2^r + \langle du_{ij}(t)/dt \rangle_1^c + \langle du_{ij}(t)/dt \rangle_2^c \\ &= [\lambda_1 + \lambda_2 \frac{\partial J(t)}{\partial u_{ij}(t)}] \{ [\frac{\partial u_{ij}(t)}{\partial r_{ij}(t)}]^2 + [\frac{\partial u_{ij}(t)}{\partial c_{ij}(t)}]^2 \} \\ &= (\lambda_1 + \lambda_2) x_{ij}^2(t) [r_{ij}^2(t) + c_{ij}^2(t)] [-u_{ij}(t)]^2 \\ &\geq 0. \end{aligned}$$

Therefore, the first and second terms of (5) and (6) will cause  $u_{ij}(t)$  to monotonically increase.

**Proposition 3.** *For MHS, if  $\epsilon$  is very small, then decreasing the attraction forces  $P(t)$  of the host (Eq. (2)) amounts to increasing the minimal utility of artificial mosquitoes.*

Supposing that  $H(t) = \max_{i,j} \{-u_{ij}^2(t)\}$ , we have

$$[\exp(H(t)/2\epsilon^2)]^{2\epsilon^2} \leq [\sum_{i=1}^n \sum_{j=1}^n \exp(-u_{ij}^2(t)/2\epsilon^2)]^{2\epsilon^2} \leq [nn \exp(H(t)/2\epsilon^2)]^{2\epsilon^2}.$$

Taking the logarithm of both sides of the above inequalities gives

$$H(t) \leq 2\epsilon^2 \ln \sum_{i=1}^n \sum_{j=1}^n \exp(-u_{ij}^2(t)/2\epsilon^2) \leq H(t) + 2\epsilon^2 \ln nn.$$

Since  $nn$  is constant and  $\epsilon$  is very small, we have

$$H(t) \approx 2\epsilon^2 \ln \sum_{i=1}^n \sum_{j=1}^n \exp(-u_{ij}^2(t)/2\epsilon^2) - 2\epsilon^2 \ln nn = 2P(t).$$

It turns out that the attraction force  $P(t)$  at time  $t$  represents the maximum of  $-u_{ij}^2(t)$  among all the artificial mosquitos  $m_{ij}$ , which is the minimal personal utility of the artificial mosquitoes at time  $t$ . Hence the increase of the attraction function  $P(t)$  will result in the increase of the minimum of  $u_{ij}(t)$ .



**Proposition 4.** *Updating  $c_{ij}$  and  $r_{ij}$  according to Eqs. (5) and (6) amounts to increasing the minimal utility of the artificial mosquitoes in direct proportion to the value of  $\lambda_3$ .*

The speed increment of artificial mosquito  $m_{ij}$ , which is related to the attraction function  $P(t)$ , is given by

$$\begin{aligned} \left\langle \frac{du_{ij}(t)}{dt} \right\rangle_3 &= \left\langle du_{ij}(t)/dt \right\rangle_3^r + \left\langle du_{ij}(t)/dt \right\rangle_3^c \\ &= \lambda_3 \frac{\partial P(t)}{\partial u_{ij}(t)} \{ [\frac{\partial u_{ij}(t)}{\partial r_{ij}(t)}]^2 + [\frac{\partial u_{ij}(t)}{\partial c_{ij}(t)}]^2 \}. \end{aligned}$$

Denote by  $\langle \frac{dP(t)}{dt} \rangle$  the differentiation of the attraction function  $P(t)$  with respect to time  $t$  arising from using Eqs.(5), (6). We have

$$\begin{aligned} \left\langle \frac{dP(t)}{dt} \right\rangle &= \frac{\partial P(t)}{\partial u_{ij}(t)} \left\langle \frac{du_{ij}(t)}{dt} \right\rangle_3 \\ &= \lambda_3 [\frac{\partial P(t)}{\partial u_{ij}(t)}]^2 \{ [\frac{\partial u_{ij}(t)}{\partial r_{ij}(t)}]^2 + [\frac{\partial u_{ij}(t)}{\partial c_{ij}(t)}]^2 \}. \\ &= \lambda_3 \omega_{ij}^2(t) u_{ij}^2(t) x_{ij}^2(t) [r_{ij}^2(t) + c_{ij}^2(t)] [u_{ij}(t)]^2 \\ &\geq 0. \end{aligned}$$

where,

$$\omega_{ij}(t) = \exp[-u_{ij}^2(t)/2\epsilon^2] / \sum_{i=1}^n \sum_{j=1}^n \exp[-u_{ij}^2(t)/2\epsilon^2].$$

It can be seen that using Eqs. (5) and (6) give rise to monotonic increase of  $P(t)$ . Then by Proposition 3, the increase of  $P(t)$  will result in the increase of the minimal utility, in direct proportion to the value of  $\lambda_3$ .

**Proposition 5.** *Updating  $c_{ij}$  and  $r_{ij}$  by Eqs. (5) and (6) gives rise to monotonic increase of the whole utility of all artificial mosquitoes, in direct proportion to the value of  $\lambda_2$ .*

Similar to Proposition 2, it follows that when an artificial mosquito  $m_{ij}$  modifies its  $c_{ij}$  and  $r_{ij}$  by Eqs. (5) and (6), differentiation of  $J(t)$  with respect to time  $t$  will not be negative—i.e.,  $\langle \frac{dJ_R(t)}{dt} \rangle \geq 0$ , and it is directly proportional to the value of  $\lambda_2$ .

**Proposition 6.** *Updating  $c_{ij}$  and  $r_{ij}$  by Eqs. (5) and (6) gives rise to monotonic decrease of the artificial mosquitoes' behavior interaction function  $Q(t)$ , in direct proportion to the value of  $\lambda_4$ .*

As in the above, we have

$$\begin{aligned} \left\langle \frac{du_{ij}(t)}{dt} \right\rangle_4 &= -\lambda_4 \frac{\partial Q(t)}{\partial u_{ij}(t)} \{ [\frac{\partial u_{ij}(t)}{\partial r_{ij}(t)}]^2 + [\frac{\partial u_{ij}(t)}{\partial c_{ij}(t)}]^2 \}; \text{ and} \\ \left\langle \frac{dQ(t)}{dt} \right\rangle &= \frac{\partial Q(t)}{\partial u_{ij}(t)} \left\langle \frac{du_{ij}(t)}{dt} \right\rangle_4 \\ &= -\lambda_4 [\frac{\partial Q(t)}{\partial u_{ij}(t)}]^2 \{ [\frac{\partial u_{ij}(t)}{\partial r_{ij}(t)}]^2 + [\frac{\partial u_{ij}(t)}{\partial c_{ij}(t)}]^2 \} \\ &\leq 0. \end{aligned}$$

## 7 Simulations

We give the experimental results here which demonstrate the effectiveness of the MHSA in handling large-scale TSP problems. We show the actual times

and iterations used to solve TSP problems on a cluster, which can verify the efficiency and parallelism of our MHS algorithm. All the experiments presented in this section were carried out on a cluster where each machine has a Pentium 4 2.0 GHz CPU with 512 Kbytes of L2 cache and 512 Mbytes of DDR SDRAM, and the machines are interconnected via Fast Ethernet.

### 7.1 Effectiveness of MHS Algorithm

We pick a number of TSP problems of different scales to test our MHS algorithm. No matter how large the scale is, the MHS algorithm can always converge and obtain the solution to the TSP.

In the real world, there is little collaboration among mosquitoes. Each mosquito seeks the host all by itself. In our mosquito host-seeking model, each artificial mosquito evolves and moves in parallel almost without information exchange. Therefore, the MHS algorithm has inherent parallelism, but only approximate solutions can be obtained. As shown in Fig. 5 and Fig. 6, even if the MHS algorithm can converge and obtain better solutions to the TSP, the MHS algorithm is easy to jump into local optima.

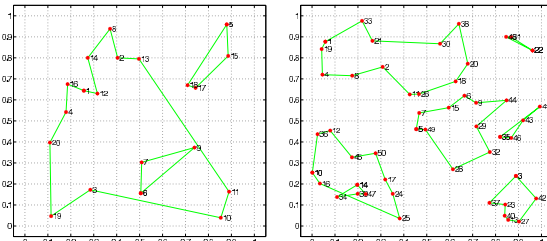


Fig. 5. The results of 20-city and 50-city TSPs using MHS

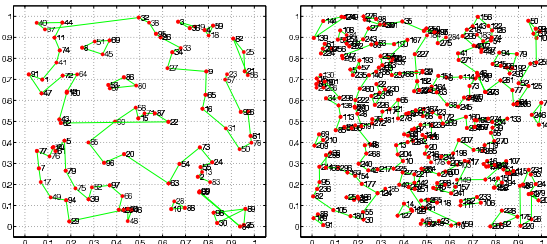


Fig. 6. The results of 100-city and 300-city TSPs using MHS

### 7.2 Efficiency and Parallelism of MHS Algorithm

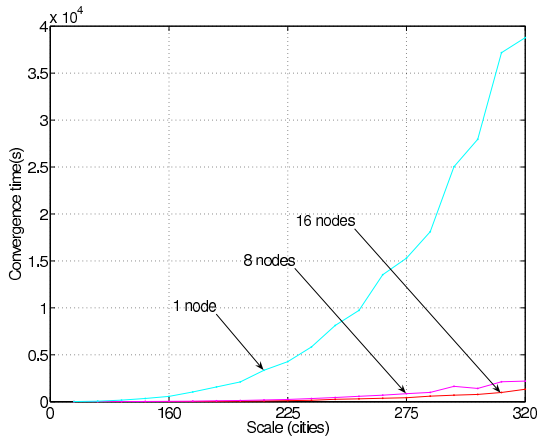
The MHS algorithm provides a valuable alternative to traditional methods because of its inherent parallelism. The grayscale values and weights,  $r_{ij}$ ,  $c_{ij}$ , can be computed and updated in parallel without any information exchange, which

**Table 3.** Convergence times and speeds of MHSA with scale

Scale cities	16 parallel nodes		8 parallel nodes		1 parallel node	
	time(s)	iterations	time(s)	iterations	time(s)	iterations
110	0.983	242	1.90	166	16.767	176
160	2.583	600	5.74	348	58.827	393
200	5.453	1046	12.62	602	169.793	816
225	10.8	1759	25.87	973	343.651	1312
250	15.868	2227	42.034	1293	569.017	1832
275	29.419	3327	63.617	1722	1044.953	2842
300	38.901	4159	99.039	2276	1571.968	3711
320	63.718	5711	130.258	2679	2113.836	4351
340	84.762	6806	183.987	3410	3358.083	5839
360	118.984	8404	247.028	4122	4276.371	6758
375	155.671	9816	336.842	4988	5839.789	8245
390	262.693	12544	446.898	5834	8130.714	9963
405	307.017	13909	578.865	6845	9729.199	11014
420	374.298	15629	699.978	7718	13528.261	13064
435	425.01	17180	861.57	8609	15294.553	14319
450	594.781	19854	997.091	9528	18108.043	15921
465	698.079	21922	1641.718	11881	25035.577	18459
480	774.948	23927	1418.417	11523	27951.384	20024
495	987.354	26780	2122.651	13738	37178.713	22961
510	1332.217	30418	2196.459	14275	38786.457	24314

is the foundation of the MHSA’s parallelism. The experimental results have verified the good parallelism of the MHSA (see Table 3). We use 1, 8, 16 computing nodes of the cluster, respectively.

As shown in Table 3, the convergence time and the number of iterations increase steadily when the TSP scales up. Table 3 includes the sequential version which comes from our experimental results using one computing node of the cluster. The other parts are for the parallel version, with using 4 and 16 computing nodes of the cluster. “Iterations” and “time” are the number of iterations and the time the MHSA takes to converge.



**Fig. 7.** Convergence times of MHSA with scale

As shown in Fig. 7, the convergence time of the sequential version increases exponentially with the scale, which is similar to all existing exact methods. When parallelized, the convergence time drops significantly, which speaks for the high parallelism of MHSA.

## 8 Conclusion

In this paper, we propose a novel mosquito host-seeking algorithm as a new branch of nature-inspired algorithms for solving TSP problems. The MHS algorithm is inspired by the host-seeking behavior of mosquitoes. We discuss the mathematical model, the algorithm, the motivation, and the biological model, and give experimental results of the MHSA being applied to the TSP. The MHSA can work out the theoretical optimum solution, which is important and exciting.

**Acknowledgements.** This project is supported by the Hong Kong General Research Fund under Grant No. HKU 713708E, the National Science Foundation of China (NSFC) under Grant No. 60575027, the High-Tech Development Program of China (863) under Grant No. 2006AA10Z315, and the Specialized Research Fund for the Doctoral Program of Higher Education under Grant No. 20060251013.

## References

1. Crescenzi, P., Kann, V.: A Compendium of NP Optimization Problems (1998), <ftp://ftp.nada.kth.se/Theory/Viggo-Kann/compendium.pdf>
2. Holland, J.: *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge (1976)
3. Schwefel, H.: *Evolution and Optimum Seeking*. John Wiley, New York (1995)
4. Porto, V.: Evolutionary programming. In: Bäck, T., Fogel, D., Michalewicz, Z. (eds.) *Handbook of Evolutionary Computation*, Institute of Physics, Bristol (1997)
5. Dorigo, M., Maniezzo, V., Colnari, A.: Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Trans. Systems, Man, Cybernet.*, Part B 26(1), 29–41 (1996)
6. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. In: *IEEE Conf. Neural Networks*, pp. 1942–1948. IEEE Press, Los Alamitos (1995)
7. Farmer, J., Packard, N., Perelson, A.: The Immune System, Adaptation, and Machine Learning. *Physica D* 2, 187–204 (1986)
8. Hertz, J., Krogh, A., Palmer, R.: *Introduction to the Theory of Neural Computation*. Addison-Wesley, Reading (1991)
9. Kirkpatrick, S., Gelatt, C., Vecchi, M.: Optimization by Simulated Annealing. *Science* 220(4598), 671–680 (1983)
10. Durbin, R., Willshaw, D.: An Analogue Approach to the Travelling Salesman Problem Using an Elastic Net Method. *Nature* 326(6114), 689–691 (1987)