Problem-Size Scalability of Amoeba-based Neurocomputer for Traveling Salesman Problem

Liping Zhu†, Masashi Aono‡, Song-Ju Kim‡, Masahiko Hara†‡

†Tokyo Tech-RIKEN International School, Tokyo Institute of Technology, Japan
‡Flucto-Order Functions Research Team, RIKEN-HYU Collaboration Research Center,
RIKEN Advanced Science Institute, Japan
Email: lpzhu@riken.jp

Abstract—A single-celled amoeboid organism, the true slime mold *Physarum polycephalum*, exhibits rich spatiotemporal oscillatory dynamics and sophisticated parallel problem solving capabilities. To investigate impacts on the accuracy and speed of the parallel problem solving as a result of the increase of the number of processing units, we use our previously demonstrated experimental system that leads the organism to search for a solution to the Traveling Salesman Problem (TSP). In this study, increasing the problem size N of TSP from 4 to 8, we show the following results; 1) the accuracy to reach a good solution was robustly maintained independently of N; and 2) the time required to reach a solution and the amount of information that the organism acquired from the optical feedback did not grow drastically. These results hint that the search ability of our system is enhanced by "economical" dynamics of the organism to find a high quality solution at lower exploration cost.

1. Introduction

In the amoeba-based computer [1, 2, 3], with the assistance of optical feedback to implement a recurrent neural network model, the amoeboid organism changes its shape by expanding and shrinking its photosensitive branches so that its body area can be maximized and the risk of being illuminated can be minimized. Our system finds a good solution of the 8-city TSP with a high probability.

Our latest results suggested that the organism might have performed "economical" search, i.e., quickly gathering useful information from the illuminations to reach a good solution without wasting exploration cost [4]. When the extent of search space diverges, can the system save the exploration resources, particularly the time required to reach a good solution? For the N-city TSP, our system uses N^2 branches of the organism, and the number of all solutions grows rapidly as a factorial function $(N-1)!/2$. In this study, increasing N from 4 to 8, we try to draw the growth curve of the exploration time as a function of N. As shown in Fig. 1, we fix the physical space size of the experimental condition for all N, so that we can compare the exploration time under the condition that information propagation velocities among the branches of the organism are equalized

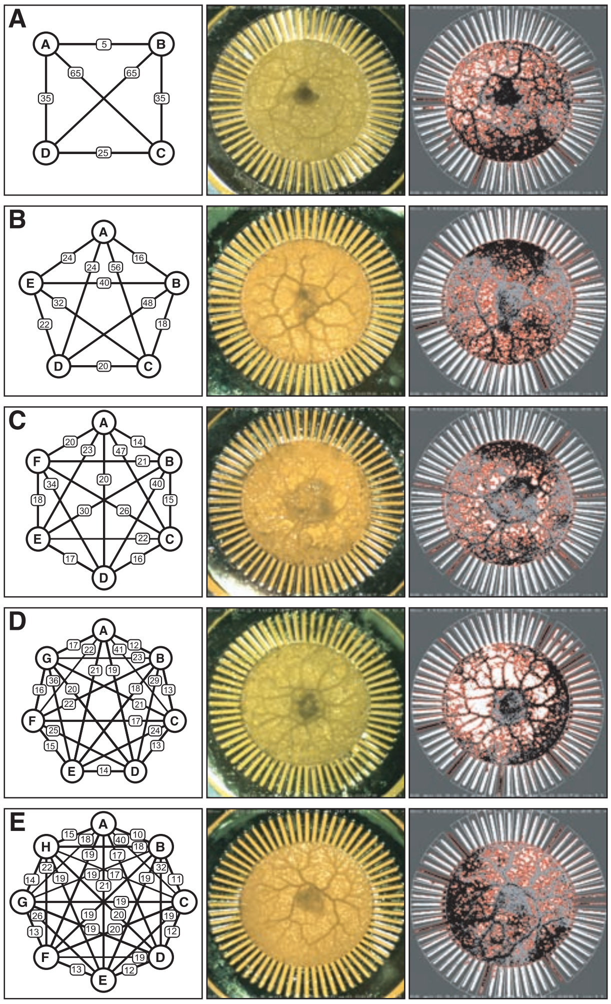

Figure 1: Topological maps (left), initial configurations (center), and valid solutions reached (right), where black and gray pixels represents that the thickness of the corresponding region of the organism was increased and decreased, respectively. Red pixels show the border between the body of the organism and unoccupied agar region (white pixels). White trapezoids indicate the illuminated regions. A, B, C, D, and E are examples of 4-, 5-, 6-, 7-, and 8-city TSP, respectively. N^2 branches are needed for solving N-city TSP. Thus, $64 - N^2$ lanes were disabled with constant illuminations for the cases where $N < 8$.

N	4	5	6	7	8
#(Solutions)	3	12	60	360	2520
Shortest	100	100	100	100	100
Longest	200	200	200	200	200
Av. Dist.	153.3	150.0	145.2	146.0	149.1
Std. Dev.	50.3	30.0	22.6	18.5	16.9

Table 1: Statics of TSP maps used in the experiments. The number of all valid solutions, shortest and longest route distances, and averages and standard deviations of route distances of all solutions are shown.

independently of N.

This study is an extension of our research of which we presented the latest results in our paper [4] that appears prior to this article. Therefore, to avoid overlapping of the detailed explanations of common experimental setups, we majorly describe only the new contents of this study.

2. Methods

2.1. TSP Maps

Figure 1 shows the maps used in the experiments. We designed these maps so that they give unimodal-like distributions of route distances of valid solutions. As shown in Table 1, for all maps, the shortest and longest routes take distances 100 and 200, respectively. For each map, the shortest and longest routes exist uniquely, and the average route distance of all solutions (i.e., the peak of the unimodal distribution) is located at about 150.

2.2. Experimental Conditions

When the organism is placed in a multi-lane stellate chip resting on an agar plate, the branches of the organism inherently grow to occupy the entire region of the lanes but withdraw when illuminated by visible light. According to the recurrent neural network model proposed by Hopfield and Tank [5], the N-city TSP can be tackled using N^2 lanes. In this study, for all N-city cases, we use 64-lane chip prepared for solving the 8-city TSP. As shown in the center column of Fig. 1, for the cases where $N < 8$, $64 - N^2$ lanes are disabled by constant illuminations so that the organism can elongate up to N^2 branches in the enabled (not disabled) lanes. Each enabled lane is labeled with Pn indicating the city name P and its visiting order n. When the organism sufficiently elongates its branch in lane Pn, we consider that city P was visited nth.

To prepare for the initial condition for the computing, we illuminate all the lanes and wait until the center disk of the chip is fully covered with the body of the organism. Table 2 shows the averaged volumes of the organism used for the experiments.

N	4	5	6	7	8
Amoeba Vol.	12.02	11.67	12.15	11.91	14.15
γ	.00495	.0058	.0067	.0076	.0081

Table 2: Experimental Parameters. Average values of the volume of the organism (mg) and values of parameter γ used in the experiments are shown.

After the computing starts, the organism changes its shape in the enabled lanes by expanding or shrinking its branches at each period (1 to 2 min) of autonomous oscillation. This spatiotemporal oscillatory movements are exploited as search dynamics in our computing scheme. Monitoring the changes in the shape of the organism at every 6 sec, the optical feedback system automatically updates the illumination pattern, which is a grayscale image[1] projected using a PC projector. The optical feedback system determines whether each lane is illuminated or not in accordance with a modified Hopfield-Tank neural network algorithm [2]. We set the values of a parameter γ of the algorithm as shown in Table 2, following the instruction given in [3, 4].

The complex oscillatory dynamics of the organism produce fluctuated growth movements of the branches that evoke a variety of illumination patterns. There are 2^{N^2} different illumination patterns that give the entire extent of search space. Through trial-and-error process to explore the search space, less frequently illuminated N branches elongate exclusively, and the system finally reaches a valid solution as shown in the right column in Fig. 1. The system is judged as reaching the solution when the illumination pattern was kept unchanged for more than 10 min.

3. Results

3.1. Accuracy

For each N, the number of experimental trials carried out and performance evaluations in finding a solution are shown in Table 3. Although the number of all valid solutions grows rapidly, the averaged route distance of the solutions reached remained relatively small and did not grow. In the 8-city case, the system exhibited relatively the best performance in finding a good solution. Indeed, the averaged route distance was 133.0 which is evaluated as "top 18.7%," i.e., the number of solutions shorter than 133 is 470 and is smaller than 18.7% of the number of all 2520 solutions. In Fig. 2B, we compared the original distribution of valid solutions obtained from the given maps (listed in Table 1) and the experimentally obtained distribution of solutions found by the organism (listed in Table 3). It is clear that the system found a good solution more accurately than

[1] Each image pattern determines illuminated and non-illuminated lanes that are colored with white and black, respectively

N	4	5	6	7	8
#(Trials)	7	6	7	9	16
Best	100	100	100	128	117
Worst	200	174	174	151	165
Av. Dist. (Top %)	137.1 (33.3)	133.0 (33.3)	135.6 (40.0)	139.4 (39.7)	133.0 (18.7)

Table 3: Statics of Experimental Results. For each N, the number of trials performed, best and worst solutions found, and averaged route distance of the solutions reached together with its relative rank in percentage are shown.

random sampling from the original distribution, and the accuracy was maintained robustly independent of N.

3.2. Speed

Fig. 2C shows how the averaged time required to reach a valid solution grows as a function of N. The growth appears to be "linear" at least for these small Ns, although we cannot assert it because the number of sample data for each N is not sufficient.

3.3. Information Exploration

We counted the number of the patterns evoked in the course of solution search, excluding overlapped occurrences of identical patterns. This counts represents the extent of search space explored by the system and can be interpreted as the accumulated amount of information that the organism obtained from the optical feedback system. Fig. 2D shows the growth function of the acquired information. Our data suggested that the growth is "linear."

4. Discussion and Conclusion

In this study, we investigated how the expansion of search space increases the use of exploration resources, such as time and information required for accurate solution search. The increase in the problem size N broadens the search space "nonlinearly," as the number of enabled lanes N^2 and number of all valid solutions $(N-1)!/2$ grow rapidly. However, for a small problem size N ranging from 4 to 8, our system needed only "linearly" grown exploration resources to maintain the same quality of accurate solution search. It would be hard to cover the nonlinearly widened search space by a succession of "random" decision making using linearly grown exploration resources [6, 7, 8]. Thus, we consider that the search ability of our system was enhanced by some "economical" dynamics to attain accurate solution search at lower exploration cost.

This observation is consistent with what we reported in our paper posted prior to this article [4]. Changing

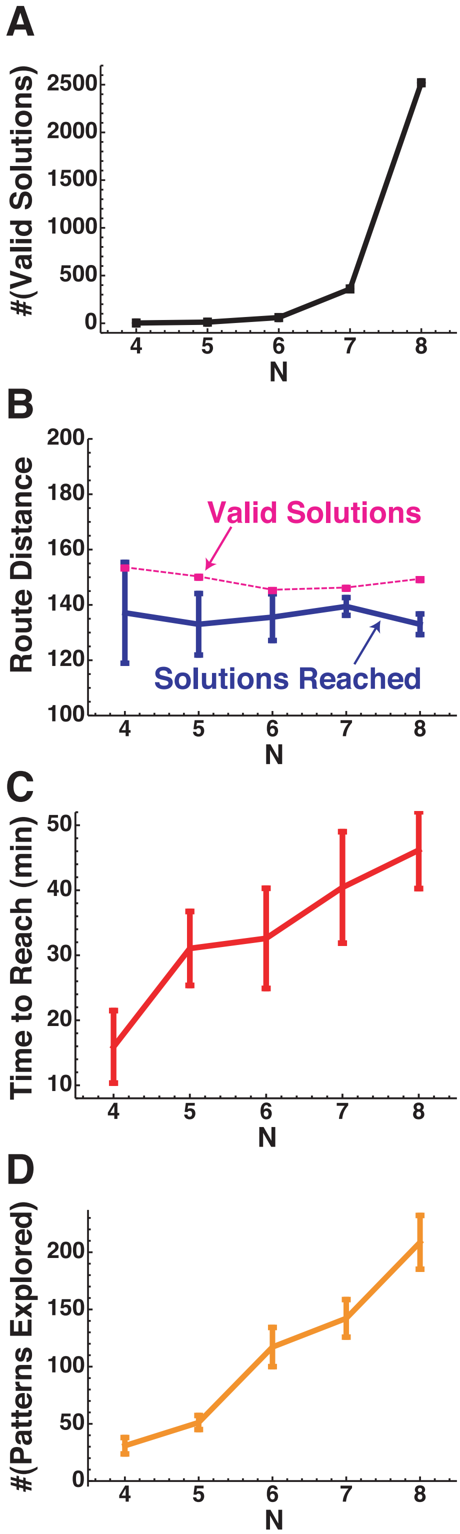

Figure 2: (A) Growth function $(N-1)!/2$ of the number of all valid solution. (B) Averaged route distances of given valid solutions (magenta) and solution reached in the experiments (blue). (C) Growth of the averaged time required to reach a valid solution. (D) Growth of the averaged number of illumination patterns explored. All error bars indicate *standard error* $(= standard\ deviation/\sqrt{\#(Trials)})$.

the experimental condition, our system became capable of achieving the same quality of accurate solution search using shorter searching time and less information. In other words, the organism could be tuned to select "economical" path in solution search requiring fewer exploration resources, although we have not yet clarified why the organism could become "smarter."

We consider the economical search process to be a series of correlated movements produced by the organism's intrinsic dynamics of which we can tune their parameters somehow. Indeed, the authors have modeled the intrinsic dynamics of the organism [9, 10]. When applied to solving the "multi-armed bandit problem," a problem of managing the trade-off between the accuracy and speed of resource allocation, the model works more economical than other well-known algorithms.

The search dynamics of the amoeba-based computer are understood as a hybrid of the Hopfield-Tank neural network dynamics [5] of the optical feedback and complex spatiotemporal oscillatory behavior [11, 12] of the intrinsic dynamics of the organism . It is one of our future subject to evaluate the net effect of the organism's intrinsic dynamics to enhance the system's search ability by comparing our system with a simulation model, which is a hybrid of the Hopfield-Tank neural net and some random decision-making unit. We believe that our studies on the economical search dynamics will contribute to understanding and developing a wide variety of smart systems to achieve efficient uses of resources in uncertain environments.

Acknowledgments

This work was supported by KAKENHI (22700322). The authors thank Dr. Takashi Isoshima and Ms. Yuki Hasegawa (RIKEN) for their cooperation 's for experiments.

References

[1] M. Aono, M. Hara, K. Aihara, "Amoeba-based neurocomputing with chaotic dynamics," *Communications of the ACM*, vol.50 (9), pp.69–72, 2007.

[2] M. Aono, Y. Hirata, M. Hara, K. Aihara, "Amoeba-based chaotic neurocomputing: combinatorial optimization by coupled biological oscillators," *New Generation Computing*, vol.27, pp.129–157, 2009.

[3] M. Aono, L. Zhu, M. Hara, "Amoeba-based Neurocomputing for 8-City Traveling Salesman Problem," *International Journal of Unconventional Computing*, to appear.

[4] M. Aono, L. Zhu, S.-J. Kim, M. Hara, "Performance Enhancement of Amoeba-based Neurocomputer for 8-City Traveling Salesman Problem," *Proc. NOLTA 2011*, 2011.

[5] J. J. Hopfield, D. W. Tank, "Computing with Neural Circuits: A model," *Science*, vol.233, pp.625–633, 1986.

[6] M. R. Garey, D. S. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness," *W. H. Freeman and co.*, New York, 1979.

[7] J. H. Holland, "Adaptation in Natural and Artificial Systems (Second Edition)," *The MIT Press*, Cambridge, Massachusetts, 1992.

[8] E. Bonabeau, M. Dorigo, G. Theraulaz, "Swarm Intelligence: From Natural to Artificial Systems," *Oxford University Press*, New York, 1999.

[9] S.-J. Kim, M. Aono, M. Hara, " Tug-of-war model for multi-armed bandit problem," *In: Calude, C. (Ed.), Unconventional Computation. Lecture Notes in Computer Science*, Springer 6079, pp.69–80, 2010.

[10] S.-J. Kim, M. Aono, M. Hara, "Tug-of-war model for two-bandit problem: Nonlocally correlated parallel exploration via resource conservation," *BioSystems*, vol.101, pp.29–36, 2010.

[11] Y. Hirata, M. Aono, M. Hara, K. Aihara, "Spontaneous mode switching in coupled oscillators competing for constant amounts of resources," *Chaos*, vol.20, pp.013117, 2010.

[12] M. Aono, Y. Hirata, Y. Hara, K. Aihara, "Resource-competing oscillator network as a model of amoeba-based neurocomputer," *In: Calude, C. (Ed.), Unconventional Computation. Lecture Notes in Computer Science*, Springer 5715, pp.56–69, 2009.