Evaluation of Adaptive Nature Inspired Task Allocation Against Alternate Decentralised Multiagent Strategies

Richard Price and Peter Tiňo

School Of Computer Science
University Of Birmingham
Birmingham B15 2TT, UK
{ug73rxp, pxt}@cs.bham.ac.uk

Abstract. Adaptive multiagent algorithms based upon the behaviour of social insects are powerful *decentralised* systems capable of solving complex problems. The intelligence of such a system lies not within a single agent but is a product of a network of simple interactions. Under the context of a mail collection environment different techniques are implemented and evaluated. The paper creates a number of strategies to tackle task allocation problems of this type based upon the principles of self-organisation and greedy search. The paper also investigates factors that may affect their performance.

1 Introduction

Social insects have been colonising the planet for millions of years, part of their ability to survive in numerous climates is due to their ability to react to changing demands with little or no centralised control. Social insects are therefore powerful decentralised problem solving systems. Theories of self-organisation, originally developed in the context of physics and chemistry, can be extended to social insects. Algorithms based upon these principles have been shown to be effective on many complex problems including the travelling salesman problem, combinatorial optimisation and graph partitioning. The success of these algorithms and indeed social insects is (at least partly) due to their ability to adapt to dynamic environments.

The aim of this paper is to investigate how adaptive nature inspired techniques can be applied to a multiagent task allocation problem. The issue with all adaptive algorithms that are designed to deal with dynamically varying problems is that there is no standard way of evaluating their performance [1].

This paper explores nature inspired task allocation in a multiagent environment and compares its performance against other more established strategies.

The problem considered here is a variation of the mail retrieval proposal by Bonabeau et al. in [1]. Batches of mail produced by cities need to be assigned to processing centres all of which are spread across an area. Once produced, a batch of mail waits for collection at its origin and that city cannot produce another batch until it has been collected. Cities can produce different types of batches, each of which need to be processed differently by centres. Each centre has a collecting agent solely responsible for the selection and collection of batches of mail. Centres will eventually develop a queue of batches each of which must be processed in turn. It takes a set amount of time to process a batch of mail but it takes significantly longer to reconfigure the centre if the type of mail needing to be processed is different from the batch previously processed. Therefore the number of these changeovers should be minimised as best as possible.

However, a centre cannot exclusively deal with a single type of mail as a deadlock situation may occur with all the cities having a single type of mail waiting for collection. In addition each centre has a limited size queue of batches that have been collected and are awaiting processing. Therefore each centre needs to specialise as best as possible whilst being able to react to fluctuating demands. It is clear that the performance of each centre is directly dependent upon the decisions of the collecting agent.

The approach taken in this paper investigates a number of new areas. Firstly the project investigates decentralised task allocation rather than the centrally controlled approaches taken in [2, 3]. Secondly the paper compares a number of decentralised algorithms upon the mail retrieval problem to analyse how their performances compare and contrast. Finally the environment is more generalised than previous work in [1, 3, 4] allowing further investigation into the behaviours of the different algorithms.

2 Decentralised Approaches to Task Allocation

Developed by Bonabeau et al. in [5] the fixed response threshold algorithm can explain several aspects of the behaviour of social insects. In this model, individuals are more likely to engage in a task when the level of stimulus associated with that task exceeds their threshold. As a task's stimulus can only increase if the task is not performed at all or with not enough efficiency, removing individuals that normally conduct a specific task will result in the associated stimulus increasing. This will cause other individuals not normally associated with this task being stimulated to conduct it. This behaviour directly relates to the observations of the notable biologist Wilson in [6].

Every individual a is assumed to posses a set of response thresholds $\Theta_a = \{\theta_{a,0}, ..., \theta_{a,N}\}$. Each threshold $\theta_{a,t}$ correspond to a *type of task* $t = 0, 1, 2, ..., N$, that individual is able to perform. The initial values of the thresholds are randomly initialised to ensure that their roles are not predetermined.

A response threshold algorithm combines the associated threshold with the corresponding stimulus intensity of a task to calculate the probability that an individual will engage in that task. A threshold response function ensures that when the stimulus exceeds an individual's corresponding threshold that individual is likely to engage in that task. Correspondingly if the stimulus is less than an individual's threshold then it should be unlikely that the individual engages in that task. Finally if the stimulus is equal to the individual's threshold then there should be a 50% chance of that individual engaging in that task. Formally [1],

$$T_{\theta_{a,t}}(S_j) = \frac{(S_j)^n}{(S_j)^n + (\theta_{a,t})^n}, \tag{1}$$

where $T_{\theta_{a,t}}(S_j)$ is the probability that the individual a will perform task j of type t. The probability is directly related to the magnitude of the stimulus S_j and the individual's response threshold $\theta_{a,t}$ to that type of task. In this paper, the stimulus S_j is the time waited of batch of mail j whilst $\theta_{a,t}$ is the threshold of agent a corresponding to type of mail t. In addition, the steepness of the threshold response function can be altered through the parameter $n > 2$.

However, fixing an agent's threshold limits the agent's ability to adapt to its environment and this model cannot account for several aspects of social insect behaviour. A fixed threshold model assumes that an individual's role is predetermined, in addition to excluding ageing and learning from the task allocation process. Therefore a fixed threshold model is only a valid model of social insect behaviour over a sufficiently short period of time where thresholds are considered relatively stable.

Theraulaz et al. [7] extended the fixed threshold model by allowing variable thresholds. This model allows thresholds to vary through time in a self-reinforcing way according to what action an agent takes. In our paper, each time an agent a collects a batch of mail of type t, its threshold for collecting that type of batch again is lowered by a small amount $\epsilon > 0$

$$\theta_{a,t}^{new} = \theta_{a,t}^{old} - \epsilon. \tag{2}$$

In addition, that agent's thresholds for all other types of batches q are increased by a small amount $\phi > 0$,

$$\theta_{a,q}^{new} = \theta_{a,q}^{old} + \phi, \quad q \neq t. \tag{3}$$

In [1], Bonabeau et al. refers to ϵ and ϕ as learning and forgetting coefficients, respectively. A response threshold function such as (1) is still used to select tasks. In addition each threshold $\theta_{a,t}$ is restricted to a positive interval $[\theta_{min}, \theta_{max}]$[1].

The variable response threshold algorithm does not assume that roles are predetermined and allows the age of an individual to affect response thresholds. In addition a number of experiments and observations imply the existence of a reinforcement process within social insects [8].

The variable response threshold algorithm has been used in several nature inspired systems. Bonabeau et al. in [1] showed that the use of variable thresholds caused individuals to become highly responsive to stimulus associated with specific tasks whilst others only became weakly responsive. By removing these responsive individuals from the experiment individuals with previously high thresholds become more responsive to the associated available tasks. This behaviour is analogous to the observations by Wilson in [6] in contrast to the

[1] if $\theta_{a,t}^{new} < \theta_{min}$, then $\theta_{a,t}^{new} = \theta_{min}$, and if $\theta_{a,t}^{new} > \theta_{max}$, then $\theta_{a,t}^{new} = \theta_{max}$

fixed threshold model in which thresholds cannot respond to perturbations in the environment. This paper implements both the fixed and variable response threshold algorithms to further analyse their behaviour.

Other work also investigates the behaviour of the response threshold algorithm. In [4], Cicirello et al. showed that by using a variable response threshold algorithm and a dynamic queueing policy based upon wasp behaviour is capable of producing an efficient and robust system that can adapt to dynamically changing factory environments. Campos et al. [3] explain the similarities between the variable response threshold algorithm and a more established market-based approach. Overall related work shows that the variable response threshold algorithm can be used to create a self-organised system that is flexible, efficient and robust.

A core aim of this paper is to compare the performance of the variable response threshold algorithm against viable alternatives, such the variable response probability algorithm introduced below. Each agent a has an internal value $P_{a,t}$ for each type of mail t. $P_{a,t}$ represents the probability of collecting mail of type t by agent a. Each time an agent a collects a batch of mail of type t, its probability for collecting that type of mail again is increased, while probabilities for collecting all other types of mail are decreased:

$$Q_{a,t} = P_{a,t}^{old} \cdot (1+\alpha), \tag{4}$$

$$Q_{a,q} = P_{a,q}^{old} \cdot (1-\alpha), \quad q \neq t, \tag{5}$$

$$P_{a,j}^{new} = \frac{Q_{a,j}}{\sum_r Q_{a,r}}, \tag{6}$$

where $0 < \alpha < 1$.

Note that updates in the variable response probability algorithm are multiplicative in nature, whereas the threshold updates in the variable response threshold algorithm by Theraulaz et al. [7] are additive.

As mentioned earlier the environment created for this paper is different in many respects than alternate approaches. Each agent, regardless of strategy, is supplied with the same information about mail awaiting collection (such as type and time waited). No agent, in any strategy, has access to information about other agent's actions or states. Therefore each strategy is completely decentralised. Many task allocation techniques implement centralised control and/or communication between agents to optimise the overall performance [2, 3]. This paper offers a fresh outlook at decentralised task allocation algorithms within both stationary and dynamically changing environments.

3 Experiments

The experiments in this section analyse the performance of the all the strategies upon increasingly complex environments. However, all the experiments have a few common features. Each simulation is run for a period of 10,000 ticks and each experiment uses 100 simulations for each strategy to ensure comprehensive testing.

Both fixed and variable response threshold algorithms used the following parameter settings: $\theta_{min} = 0$, $\theta_{max} = 100$, $n = 2$. In addition, the variable response threshold algorithm used $\epsilon = 5$ and $\phi = 5$. Parameter α in the variable response probability algorithm was set to 0.2. These parameter settings where chosen as they tended to lead to good performance levels when tested upon the multiple versions of the environment (with varying degrees of complexity) used in this paper.

The work presented here also implemented two base case strategies designed to be the minimum level of acceptable performance for the other strategies. By far the simplest of the algorithms, *'first in - first out'* (FIFO), collects mail in the same order it is produced. Slightly more sophisticated than FIFO is the *greedy algorithm* that always attempts to collect the same type of mail it is currently processing. The type of mail currently being processed has the highest priority, otherwise mail is collected according to the time it has waited. One would expect that the greedy algorithm performs to a higher standard than FIFO by processing more mail and incurring fewer changeovers.

Performance can be evaluated by how much mail each strategy is able to process whilst minimising the number of changeovers. The tables in this section show for each experiment the average amount of mail processed and the average number of changeovers of each strategy over 100 runs. In addition the standard deviations of these figures are shown in brackets below the averages.

We tested for significance in differences between alternate strategies across multiple runs of the same experiment using t-test. Throughout this section the symbol $*$ signifies that a strategy is statistically significantly worse (with probability 0.95) in comparison to the variable response threshold algorithm. Analogously, symbol $+$ signifies that a strategy is statistically significantly worse in comparison to the variable response probability algorithm (with probability 0.95).

3.1 Experiment 1

The initial comparison used an environment of six cities (producing mail), two centres (collecting/processing the mail) and two types of mail. The results of the experiment are shown in Table 1.

As expected, the FIFO strategy was outperformed by every other strategy. In addition, the greedy algorithm was only able to slightly increase throughput and decrease changeovers. The fixed response threshold algorithm outperformed both of these algorithms but was unable to compete with the variable response threshold (VRT) and variable response probability (VRP) algorithms. The VRT algorithm was able to significantly increase throughput and decrease changeovers by almost 50% in comparison to the fixed response threshold algorithm. The VRP algorithm decreased changeovers dramatically, well below all other algorithms, while additionally closely matching the throughput of the VRT algorithm.

Table 1. Average performances of strategies in Experiment 1. FRT, VRT and VRP stand for the fixed response threshold, variable response threshold and variable response probability algorithms, respectively.

Strategy	Mail Processed	Changeovers
FIFO	$318.73^{*,+}$ (10.676)	$162.21^{*,+}$ (3.817)
Greedy	$321.23^{*,+}$ (13.348)	$159.75^{*,+}$ (5.695)
FRT	$332.94^{*,+}$ (20.840)	$151.37^{*,+}$ (11.755)
VRT	401.40 (22.609)	88.27^{+} (10.419)
VRP	400.78 (33.108)	18.93 (7.137)

3.2 Experiment 2

Further experiments with increasingly complex environments showed similar results – variable response threshold algorithm processes the most mail whilst the VRP algorithm maintains significantly lower changeovers. A typical example (using 30 cities, 10 centres and 2 types of mail) is presented in Table 2.

Table 2. Average performances of strategies in Experiment 2 (30 cities, 10 centres, 2 types of mail).

Strategy	Mail Processed	Changeovers
FIFO	$1587.20^{*,+}$ (27.167)	$805.80^{*,+}$ (9.945)
Greedy	$1632.51^{*,+}$ (44.587)	$756.49^{*,+}$ (29.533)
FRT	$1485.05^{*,+}$ (82.864)	$508.32^{*,+}$ (58.587)
VRT	2318.61 (35.622)	222.67^{+} (29.330)
VRP	2207.87^{*} (143.795)	65.90 (14.721)

3.3 Experiment 3

The previous experiments only investigated stationary environments where the probabilities of different types of mail appearing remained constant. It was reasoned that because of the relatively high value of parameter α in the variable response probability model ($\alpha = 0.2$) and the multiplicative nature of updates in this model, the response probabilities of agents quickly specialise to a type of mail to process (the probability of picking an alternate type of mail rapidly decreases to negligible values). This makes the model rather inflexible in dynamically changing environments. The next experiment setup an environment where initially one type of mail was twice as likely to appear as the alternate, however after 5000 ticks these probabilities are reversed: There were 9 cities, 3 centres and 2 types of mail. The results are presented in Table 3.

Table 3. Average performances of strategies VRT and VRP in Experiment 3 (dynamically changing environment, 9 cities, 3 centres, 2 types of mail).

Strategy	Mail Processed	Changeovers
VRT	633.68	121.03^{+}
	(29.083)	(17.555)
VRP	507.80^{*}	32.30
	(51.358)	(7.243)

In this experiment, the variable response threshold algorithm consistently devoted two centres to the dominant mail type in the first half of the simulations. At the point where the mail type probabilities switched, the algorithm reliably caused the behaviour of one of the collecting agents to specialise to the alternate and now dominant type of mail. However the VRP algorithm was unable to adapt as suitably to the dynamic probabilities within the environment. Using this strategy, the collecting agents behaviour did not alter despite the new environmental conditions resulting in a significantly lower overall throughput, although the changeovers incurred remained minimal.

4 Discussion

The main findings of this paper are that the adaptive algorithms, namely the variable response threshold (VRT) and variable response probability (VRP) algorithm, where able to significantly outperform the simpler approaches. The performance of these algorithms remained stable over increasingly complex environments. The difference in performance between the fixed and variable response threshold algorithms highlight how a dual reinforcement process enables collecting agents to adapt well to most environments.

Particularly of interest was how the VRP algorithm was able to incur a very small amount of changeovers compared to every other strategy. The changeovers occurred very early in the simulation before the collecting agents could fully specialise to one type of mail. Once the collecting agents had adapted to the environment, changeovers occurred with little or no frequency. Further experiments (not included in this paper) also show that in complex *stationary* environments the VRP algorithm is able to process more mail than the VRT algorithm.

By the nature of the VRP model, parameter α determines the speed of specialisation. Higher values of α lead to faster specialisation, but also to greater inflexibility of the model to adapt to changes in the environment. In dynamic environments, the VRT algorithm consistently outperformed the VRP algorithm. Further observations showed that the more dynamic the environment, the larger the performance gap between the VRT and VRP algorithms becomes.

This paper offers a fresh outlook at decentralised task allocation algorithms within both static and dynamic environments. The transfer of social insect inspired algorithms from static to dynamic environments has rarely been tested [3].

The work presented here also analysed the VRT algorithm in more detail and was shown to have many diverse features. The results highlight that the algorithm is capable of creating a self-organised system through stigmergy alone. This self-organised system also adapts well to most environments. In addition, natural phenomena particularly in comparison to the work of Wilson in [6] are reproducible. The behaviour of the algorithm can be altered through the parameters. Particularly of interest was how the VRT algorithm may reinforce a hypothesis suggested by Anderson in [9]: *"There must be a critical window of correlation of activity among individuals in order for self-organisation to occur. That is above some upper threshold and below some lower threshold, self-organisation breaks down, and the emergent properties no longer exist."*

5 Conclusion

We compared the variable response threshold (VRT) algorithm of Theraulaz et al. [7] for decentralised task allocation with four alternate strategies, namely FIFO, greedy algorithm, fixed response threshold algorithm (FRT) and variable response probability (VRP) algorithm. Each of these strategies where analysed and compared upon increasingly complex environments. It appears that the VRP algorithm can be most suitable in stationary environments where the probabilities of the types of mail appearing remained constant. However if the probabilities of mail types appearing are dynamic within the environment the VRP algorithm is less flexible than the VRT algorithm.

Overall the area of self-organisation is intriguing and new developments are being discovered at a rapid pace. Perhaps in the future such systems will become more widely accepted, as their behaviour is better understood. Until such a time the area will be dominated by theoretical problems and relatively few real-world applications. Even though social insect colonies and other biological systems have utilised self-organisation in the real world with great success for millions of years.

References

1. Bonabeau E., Sobkowski A., Theraulaz G., Deneubourg J.L.: Adaptive Task Allocation Inspired by a Model of Division of Labour in Social Insects. In: Bio-Computation and Emergent Computing. World Scientific (1997) 36-45.
2. Waldspurger, C.A., Hog T., Huberman B.A., Kephart J.O.: Spawn: A Distributed Computational Economy. IEEE Trans. Softw. Engineer **18** (1992) 103-117.
3. Campos M., Bonabeau E., Theraulaz G., Deneubourg J.L.: Dynamic Scheduling and Division of Labour in Social Insects. Adaptive Behaviour **8(2)** (2001) 83-92.
4. Cicirello V.A., Smith S.F.: Wasp Nests for Self-Configurable Factories. In Agents 2001. Proceedings of the Fifth International Conference on Autonomous Agents. ACM Press (2001) 473-480.
5. Bonabeau E., Theraulaz G., Deneubourg J.L.: Quantitative Study of the Fixed Threshold Model for the Regulation of Division of Labour Insect Societies Proc. Roy. Soc. London **B 263** (1996) 1565-1569.
6. Wilson E.O.: The Relation Between Caste Ratios and Division of Labour in the Ant Genus Pheidole (Hymenoptera: Formicidae). Behav. Ecol. Sociobiol. **16** (1984) 89-98.
7. Theraulaz G., Goss S., Gervet J., Deneubourg J.L.: Task Differentiation in Polistes Wasp Colonies: A Model for Self-Organizing Groups of Robots. In Proceedings First International Conference on Simulation of Adaptive Behaviour: From Animals to Animats. (1991) 346-355.
8. Withers G.S., Fahrbach S.E., Robinson G.E.: Selective neuroanatomical plasticity and division of labour in the honeybee. Nature **364** (1993) 238-240.
9. Anderson C.: Self-Organization in Relation to Several Similar Concepts: Are the Boundaries to Self-Organization Indistinct? Biol. Bull, **202** (2002) 247-255.