A Pheromone Based Mobile Agent Migration Strategy for Servicing Networked Robots

W. Wilfred Godfrey and Shivashankar B. Nair

Indian Institute of Technology Guwahati, Assam, India
{w.godfrey,sbnair}@iitg.ernet.in

Abstract. In this paper we describe how mobile agents carrying a resource as a payload can prove to be useful in searching networked robots that require their services. While the agents migrate within the network in a conscientious manner, robots requiring their services diffuse pheromones to attract and guide them through the shortest path. The bidirectional and parallel search on part of the robot and the agent culminates in a faster convergence. The paper also compares the results derived by using this method with those obtained using two other algorithms. The results and discussions clearly indicate that this pheromone based algorithm is better suited for both static and dynamic networked robotics scenarios.

1 Introduction

Of late there has been considerable research in finding applications for mobile agents [1, 2]. Even though such agents have useful properties and look intuitively advantageous, researchers have not been able to tap their potential to the fullest. While they have been used in scenarios for searching and routing [3], their more specific features like payload carrying ability as also cloning have hardly been exploited. Mobile agents have been used in applications with robots and robotic networks. Researchers have already used robots in conjunction with mobile agents for a variety of functions [4, 5]. Mobile agents, in each of these cases, have been used either as an alternative to the traditional communication paradigms or for collecting data and forwarding to a centralized server. Many other features of these agents such as inter-agent communication, goal oriented, adaptive and proactive behaviors still remain to be effectively utilized.

A model for a mobile agent based multi-robot network that uses immune system metaphors has been proposed by Godfrey and Nair [6]. The mobile agents assist in gathering and providing information to various robots forming a network. The agents used in their architecture migrate from one robot to another and provide a means for information retrieval and exchange, thereby providing for a mechanism for inter-robot information sharing. Information in this context could mean the knowledge of how a task is to be performed. Though the concept of information sharing is achieved in this architecture, the round robin strategy for mobile agent migration to discover a robot requiring a service seems grossly inefficient. Several network related strategies have

J. Suzuki and T. Nakano (Eds.): BIONETICS 2010, LNICST 87, pp. 533–541, 2012.

reportedly been implemented for agent migration [7, 8, 9]. Minar et al [10] describe two algorithms for mapping a network of hosts using mobile agents – viz. the conscientious and super conscientious approaches. These agents gather topological information of an underlying static network in a distributed fashion.

The work reported herein portrays a strategy to attract a specific mobile agent, in the network, carrying a resource as its payload, to a robot requesting for the same. Robotic nodes diffuse pheromones [11, 12, 13] into the network as and when they need a service while the mobile agents moving conscientiously within the network track these pheromones to eventually reach and service them. We have compared our method with those proposed by Gaber et al [14, 15, 16] and found that using pheromones seem to be a far better alternative in both static and dynamic multi- robot networks.

2 Resource Discovery Methods

A survey of resource discovery methods can be found in [17]. In this section we describe two existing methods which make use of the mobility of agents to discover resources and later compare them with the Pheromone-Conscientious proposed by us. Our test-bed comprises a network of robots, each capable of hosting mobile agents. The agents carry, as payload, code for various tasks to be executed by the robot. A robot is provided with a set of tasks for which it does not have the associated code, initially. It thus sends a request for the same into the network. The mobile agents, on receiving the request, migrate to this robotic node and provide the code to complete the service.

2.1 Conscientious Approach

A mobile agent using the conscientious approach [10] aims at uniformly visiting all the nodes comprising the network. The agent maintains a queue of "visited nodes". As it migrates, it tries to avoid nodes within this queue and in the process migrates to other nodes in its sojourn within the network. In doing so it maintains uniformity in its visits to the nodes and services those that have requested for a service which it carries as payload.

2.2 Gaber-Bakhouya's Random Walk and Cloning Based Approach (G-B Algorithm)

Gaber-Bakhouya's approach [14, 15] uses a mobile agent for locating resources available at distant nodes in a peer-peer network. Their algorithm works as follows-
The node which requires to locate a service creates a mobile agent termed as the request agent. A service could constitute a set of resources {R1, R2, R3, R4} each of which is available at separate nodes in the network. The request agent is capable of cloning at each hop. Based on the number of neighbors of the current node, this agent chooses one path to a node and sends a clone each along the others. Every agent and

clone has a Time to Live (TTL) parameter, which is decremented at each hop. If the TTL becomes zero before the service discovery is completed, the agent/clone contacts the requestor node to find whether any other agent has discovered the path to all these resources. If the requestor node responds negatively, the agent extends its TTL and proceeds in its search; else it kills itself. Once an agent discovers the path to all these resources, it sends back the path information to the requestor thereby completing its task. An adaptive immune approach [16] has reportedly been used to contain the population size of the clones. A reinforcement learning algorithm facilitates the strengthening of paths towards this set of resources.

3 Pheromone-Conscientious (P-C) Mobile Agent Migration Strategy for Multi-robot Systems

The work reported in this paper augments the multi-robot mobile agent architecture proposed in [6] with a combined pheromone cum conscientious strategy. The mobile agents opt for a conscientious approach while the robots requiring a service use pheromone diffusion to attract the concerned agents to satiate a service request. The Multi-Mobile Agent based Multi-Robot networked system has the following features:

i) Absence of a centralized server for hosting services.
ii) Mobile agents carrying payload (resources) keep migrating across different robotic nodes conscientiously.
iii) Robots need not have the resources (programs required for task execution) when they are deployed initially, thus facilitating a novice user to add and use robots on the fly.
iv) The robotic network could be static or dynamic.

A robot is issued a set of tasks by a human operator. A set of tasks for which the robot does not have relevant code to effect the execution could be looked upon as a service S = $\{C_{T1}\ C_{T2}, C_{T3}\ldots C_{Tn}\}$ where C_{Ti} is the code for the task T_i. A robot that needs a service is termed as a Robot Requesting Service (RRS). The mobile agents carry one or more of the C_{Ti}s as their payload. The work described in this paper intends to reduce the time required by the relevant mobile agents to reach and satiate the RRS with the requested C_{Ti}s, thereby providing for a quicker and on-site service. Once the programs for all the requested tasks are received at the RRS, they are ordered in the desired sequence, compiled and executed by the robot. This facilitates a novice user to hook on a robot to such a network and allow it to discover the essential programs. The main thrust of this paper is to describe a proactive RRS discovery process, on part of the mobile agents, using pheromones. The mobile agents normally use the conscientious method for migration within the network. They consciously avoid those robotic nodes recently visited and migrate towards those not yet visited by maintaining a “recently visited node list”. An RRS attracts the relevant agents by diffusing pheromones onto its immediate neighbors which in turn diffuse them further. Relevant agents that populate the network eventually hit the pheromones trail and on doing so, migrate thereon, along the shortest path to the RRS. The mobile agent and the RRS thus perform a proactive and concurrent bidirectional search facilitating a faster service within the network. While the mobile agent uses the conscientious approach till it senses a pheromone trail, the RRS spreads out a pheromone network around it to attract and guide the agent towards it.

3.1 Contents of the Pheromone

The virtual pheromone used herein has embedded within it, the following:

1) Task Identifier: Each task has an identifier and a pheromone is diffused specifically for that task so as to attract only those mobile agents that carry the relevant programs as their payload.

2) Concentration: Concentration of a pheromone trail has direct relevance to the proximity of the RRS. A higher concentration indicates that the RRS is close by. A mobile agent always chooses pheromone trails of higher concentration and hence reaches the RRS through the shortest available path.

3) Lifetime: Pheromones need to die out after the concerned mobile agent reaches and services an RRS, lest the agent be trapped at the RRS again for a redundant service. A life time for pheromones ensures that they die out after the concerned mobile agent reaches the RRS. However it may happen that they die out before the agent reaches the RRS. Therefore the RRS is programmed to periodically re-diffuse pheromones till the service request is completed.

4) Next Robot Identifier: Once a mobile agent selects a pheromone having the highest concentration at a node, it checks the Next Robot identifier of this pheromone to find the next robot node that will lead it to the RRS and then migrates towards it.

3.2 Diffusion of Pheromones

Diffusion of pheromones is carried out by a robotic node for every pheromone it receives from another by laying them across all its neighbors except the one from which it was received. Since the pheromone concentration gradient needs to be formed, the concentration of the pheromone as also its life time, are reduced proportionately. This maximum concentration (C_{max}) of a pheromone trail is laid between the RRS and its immediate one-hop neighbors. Successive diffusions have a concentration gradient given by equation 1.

$$\Delta C = 100/d \tag{1}$$

where d is the spanning length. The percolated pheromone concentration P_{pc} is given by equation 2.

$$\begin{aligned} P_{pc}(n) &= C_{max} && \text{for } n=1 \\ &= P_{pc}(n-1) - \Delta C && \text{for } n>1 \end{aligned} \tag{2}$$

where n is the hop count from the RRS. Similar to the concentration gradient, a gradient of the pheromone life time (ΔL) is also used to ensure that they disappear in accordance. Farther the pheromone from the RRS, the lesser is its life time. The lifetime gradient is proportional to the concentration gradient and is given by equation 3.

$$\Delta L = L_{max} (\Delta C/100) = L_{max}/d \quad (3)$$

where L_{max} is the maximum value of life time. While laying the pheromones, their lifetimes are reduced over the hops using equation (4).

$$P_{lt}(n) = L_{max} \quad \text{for } n=1$$
$$= P_{lt}(n-1) - \Delta L \quad \text{for } n>1 \quad (4)$$

where n is the hop count from the RRS. If the tasks issued need to be procured and executed quickly, the RRS will correspondingly change the values of P_{lt}, ΔC and d to allow a higher and longer diffusion into the network. This will increase the chances of the relevant agents hitting these pheromones trails. The maximum life time needs to be fixed based on the application scenario. For instance a larger network would, on an average, mean that the mobile agent takes more time to find the pheromone trail. Under such conditions we prescribe a larger life time.

4 Experimental Results and Analysis

We present a comparison of results obtained by using the three methods (viz. Conscientious, G-B and P-C) for scenarios with single and multiple RRSs populating a static robotic network. We also present a short discussion on how the three algorithms would perform when used in a dynamic network.

4.1 With One RRS and Three Agents

Simulations for several scenarios were carried out with 25 robotic nodes. Table 1 shows the results of four such scenarios depicted in Figure 1 (a)-(d) using a single RRS requesting a service S = {C_{T1}, C_{T2}, C_{T3}}. In case of the Conscientious and PC methods, the initial locations of three mobile agents, carrying three different resources, were changed for every run of the simulation as shown in the Figure 1. Likewise, the static resources were shifted accordingly for the G-B method. The number of hop counts taken by each of the agents to reach the RRS with their resources to service it is shown in Table 1. The hops taken, agents spawned/cloned and the messages/pheromones diffused together give us an idea of the amount of network resources expended in effecting a service. All these consume energy and/or time. The P-C method, carried out for 25 robotic nodes had a pheromone spanning depth of 2. As is evident, the P-C method outperforms its Conscientious counterpart due to the parallel and bidirectional search used. The use of pheromones greatly reduces the wayward movement of the mobile agents carrying resources. The P-C method takes in all 12 hops (for all three agents) and 12 pheromone diffusions in the first network scenario. In the G-B method, wherein the RRS sends agents and their clones to discover the static resources, the first agent discovers all the resources in 10 hops. The clones generated by this agent move in parallel and take an extra 10 hops. These agents may further clone and hop or pass messages all of which amount to a value greater than 15. Though these hops occur in parallel and can be attributed to a faster search, they consume both network bandwidth and energy of the nodes. The G-B algorithm also prescribes message passing between the agent/clone and the RRS every time the TTL becomes 0. The same is true for updating the path, and sending the path back to the RRS. These overheads have not been calculated here but if added makes the P-C algorithm perform far better as the mobile agents actually reach the services to the RRS. The path calculated by the learning mechanism used in the G-B algorithm is also restricted to a certain set of resources comprising a service. This path may prove to be suboptimal when proper subsets of the resources are taken into consideration.

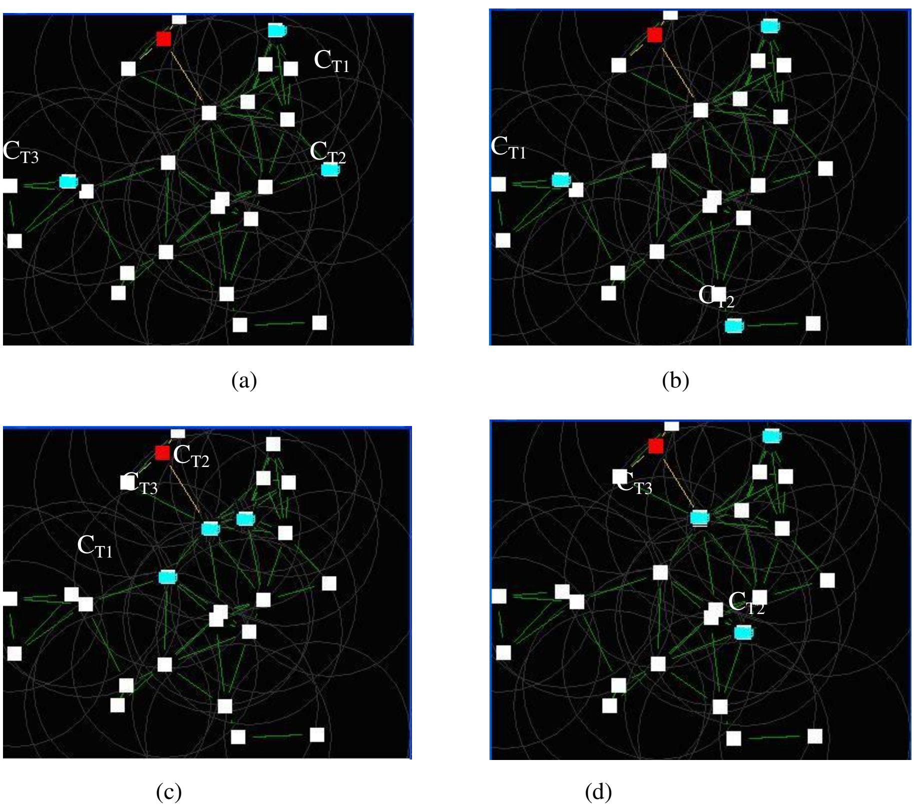

Fig. 1. Snapshots of simulations using 25 nodes. Scenarios (a)-(d) depict the placement of the agents/resources (in Blue) and that of the Requesting node (in Red).

Table 1. Table depicting the number of hops taken by three agents in the Conscientious and P-C methods to bring the resources to the RRS together with the pheromones diffused in the latter. The number of agents, number of hops consumed to locate the resource locations in the G-B method are also shown.

Scenarios	Conscientious Method			P-C Method				G-B Method		
	No. of hops			No. of hops			No. of pheromones diffused by the RRS	No. of hops taken by the Agent that first locates all the resources	No. of clones generated by the Agent that first located all the resources	No. of other Clones and Messages
	1st Agent	2nd Agent	3rd Agent	1st Agent	2nd Agent	3rd Agent				
(a)	26	10	45	5	3	4	12	10	10	Greater than 15
(b)	17	37	33	8	7	4	12	14	10	Greater than 15
(c)	29	6	15	5	5	6	12	4	2	Greater than 15
(d)	31	5	8	4	4	10	12	7	6	Greater than 15

4.2 Multiple RRS

If the above experiment were to be repeated with the multiple RRSs requesting for the same resources (viz. C_{T1}, C_{T2}, C_{T3}) at the same time, the G-B method would generate a large number of clones as also messages. It could result in an increase in energy consumed per node and also bandwidth. Though the G-B algorithm describes an immune approach to restrict clonal expansion, it has not taken into account the amount of processing involved per node and the subsequent delays caused in the search. As the number of RRSs increase clones too increase thereby increasing the effective time required for a migration. On the other hand, the P-C method is conservative in its use of both energy and network bandwidth and strives to bring the resources to the agent in times comparable to that of the G-B algorithm. A modified P-C algorithm for closely knit RRSs that request the same resource described in [18], allows for decreasing the service period.

4.3 Dynamic Multi-robot Network

Unlike the G-B algorithm that caters only to static networks, the P-C algorithm can be used in conjunction with dynamic networks with a moderate degradation in performance. In a dynamic network the links to neighboring robot nodes could make or break. Under these conditions, if the G-B algorithm were used, the agents would never be able to trace their path backwards. Even a single link failure can cause an agent to be lost. In the P-C algorithm, the agents could be made to come closer to the RRS by re-diffusing pheromones with low life times at a higher frequency and greater spanning depth. Since the pheromones have a short life time, those between separated nodes will die out faster. The higher re-diffusion frequency will allow fresh and new pheromones to be laid in the current state of the network.

5 Conclusions

In this paper, we have discussed a pheromone based migration strategy for servicing networked robots. A comparison of this method with the Conscientious and the G-B algorithms is also presented. Observations indicate the viability of using the P-C algorithm which is conservative in its consumption of both network bandwidth and energy. Further unlike the G-B algorithm, it can perform even when the network is dynamic. The G-B algorithm, on the contrary may prove to be beneficial in terms of time in static networks provided we neglect bandwidth problems and energy consumed per node. We thus conclude that the P-C algorithm described herein seems to perform better than the G-B algorithm when used in conjunction with multi-robot networks.

References

1. Outtagarts, A.: Mobile Agent Based Applications: A Survey. International Journal of Computer Science and Network Security 9(11), 331–339 (2009)
2. Giovanni, V.: Mobile Agents: Ten Reasons for Failure. In: 5th International Conference on Mobile Data Management, Berkeley, California, pp. 298–299 (2004)
3. Chen, W., Zhang, Y.: A Multi-constrained Routing Algorithm based on Mobile Agent for MANET Networks, pp. 16–19. IEEE Computer Society, IEEE Press, New York (2009)
4. Kambayashi, Y., Takimoto, M., Kodama, Y.: Controlling biped walking robots using Genetic algorithms in mobile agent environment. In: 3rd IEEE International Conference on Computational Cybernetics, pp. 29–34 (2005)
5. Cragg, L., Hu, H.: A Multi-agent system for distributed control of networked mobile robots. Measurement and Control 38(10), 314–319 (2005)
6. Godfrey, W.W., Nair, S.B.: An Immune System Based Multi-robot Mobile Agent Network. In: Bentley, P.J., Lee, D., Jung, S. (eds.) ICARIS 2008. LNCS, vol. 5132, pp. 424–433. Springer, Heidelberg (2008)
7. Zhang, J., Ma, J., Yang, J., Cheng, L.: An Improved Migration Strategy of Mobile Agent. In: 4th International Conference on Fuzzy Systems and Knowledge Discovery, vol. 1, pp. 577–581 (2007)
8. Hayfa, Z., Hammadi, S.: A Migration strategy of mobile agents for the transport network applications. Journal of Mathematical Computation and Simulation 76(5-6), 345–362 (2008)
9. Kwon, H.C., Kim, H.H., Lee, J.T., Yoo, K.J.: A Migration strategy of mobile agent. In: 8th International Conference on Parallel and Distributed Systems, pp. 706–712 (2001)
10. Minar, N., Kramer, K.H., Maes, P.: Cooperating Mobile Agents for Mapping Networks. In: 1st Hungarian National Conference on Agent Based Computing, pp. 34–41 (1999)
11. Kitamura, N., Yoshiyuki, N., Koji, F.: Development of a desktop swarm robot system based on pheromone communication. Artificial Life Robotics 14(3), 329–331 (2009)
12. Parunak, H., Dyke, V., Brueckner, S., Sauter, J.A.: Digital Pheromones for Coordination of Unmanned Vehicles. In: 1st International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 246–263 (2002)

13. David, P., Daily, M., Estowski, R., Howard, M., Lee, C.: Pheromone Robotics. In: Autonomous Robots, vol. 11(3), pp. 319–324. Kluwer Academic Publishers (2001)
14. Gaber, J., Bakhouya, M.: Mobile Agent-Based Approach for Resource Discovery in Peer-to-Peer Networks. In: Joseph, S., Despotovic, Z., Moro, G., Bergamaschi, S. (eds.) AP2PC 2006. LNCS (LNAI), vol. 4461, pp. 63–73. Springer, Heidelberg (2008)
15. Bakhouya, M., Gaber, J.: A Query Routing Approach Based on Users' Satisfaction for Resource Discovery in Service-Oriented Networks. World Wide Web 13(1-2), 61–73 (2010)
16. Bakhouya, M., Gaber, J.: Adaptive Approach for the Regulation of a Mobile Agent Population in a Distributed Network. In: The Fifth International Symposium on Parallel and Distributed Computing, pp. 360–366 (2006)
17. Meshkova, E., Riihijärvi, J., Petrova, M., Mähönen, P.: A Survey on Resource Discovery mechanisms, peer-to-peer and service discovery frameworks. Computer Networks 52(11), 2097–2128 (2008)
18. Godfrey, W.W., Nair, S.B.: Mobile Agent Cloning for Servicing Networked Robots. In: Desai, N., Liu, A., Winikoff, M. (eds.) PRIMA 2010. LNCS, vol. 7057, pp. 336–339. Springer, Heidelberg (2012)