Generic Model of a Mobile Agent Network Suitable for Performance Evaluation

Vjekoslav Sinkovic and Ignac Lovrek

University of Zagreb

Faculty of Electrical Engineering and Computing, Department of Telecommunications

HR-1000 Zagreb, Unska 3, Croatia

Tel. + 385 1 612 98 02, E-mail: ignac.lovrek@fer.hr

The paper deals with a generic model of a mobile agent network that includes multi-agent system, and a set of processing nodes connected by a communication network. A mobile agent network is described as a queuing system where mobile agents represent information units that must be served. Network flows related to agent hosting and migration and serving capabilities of processing nodes are proposed. The response time defining an interval between acquiring the request and producing the response is defined as a performance measure for a mobile agent network. A case study discussing response time evaluation is included.

1. Introduction

The increased use of the network-centric applications creates challenges in developing and managing such applications. An agent paradigm is the promising choice because it is intrinsically communication and co-operation oriented [1, 2]. Agent concepts and a mobile software agent have become a part of the system and service architecture of next-generation networks.

A mobile agent represents a user in the network and can migrate autonomously from node to node to perform some processing on behalf of a user. Current research of the mobile agents includes the concepts, models, frameworks and programming systems, information, co-operation and transaction agents, and standardisation issues. Majority of the activities deal primarily with functionality issues. The same attention has not been paid yet to the mobile agent performance.

This paper introduces a model of mobile agents suitable for performance analysis. A generic model defines a set of co-operating and communicating agents as a mobile agent network and evaluates its response time as a basic performance parameter.

The paper is organised as follows: Mobile software agents are described in Section 2. Generic model of mobile agent network is defined in Section 3. Section 4 gives a case study related to processing performance and discusses response time optimisation.

2. Mobile Software Agents

An agent is an entity dealing on behalf of its user. Basic properties defining an agent are: intelligence, autonomy and mobility supporting its goal-orientation. A mobile software agent can migrate autonomously from node to node to perform some processing on behalf of a user [3-5].

This paper pays the attention to application areas related to distributed systems and networks, i.e. the agents for distributed problem solving and the use of an agent in communication system operation and management. In such situations an agent's mobility offers important advantages because of the reduced network load, increased asynchrony between communicating entities and parallel activities. Global end-to-end interactions, typical for client – server paradigm, are replaced by local interactions in a server visited by a mobile agent. Consequently, the need for long reliable connections is reduced, bandwidth requirements become decreased and repeated interactions less frequent.

A multi-agent system, A , consists of n co-operating and communicating mobile agents as follows:

$$A = \{agent_1, \dots, agent_k, \dots, agent_n\}.$$

The multi-agent system resides in a network in which the agents perform the services requested by their users. It can be considered as a distributed system whose processing nodes communicate with each other over a communication network, N . A set of processing nodes is denoted as follows:

$$S = \{S_1, \dots, S_i, S_j, \dots, S_{nc}\}.$$

Each node, S_i , is characterised by a set of services, s_i , it provides. A mobile software agent, $agent_k$, is defined by:

$$agent_k = \{name_k, address_k, service_k\},$$

where $name_k$ represents a unique agent identification, $address_k$ its location and $service_k$ functionality it provides. For $agent_k$ hosted by the node S_i or directed toward it $address_k = S_i$ and $service_k \in s_i$. In further explanations a triple $\{A, S, N\}$ will be referred to as a mobile agent network.

Node architecture includes a mobile agent platform [6, 7] that provides the following functionality:

- Agent Functions, responsible for execution and monitoring of the agents, their identification, communication and mobility,
- Security Functions, providing secure communication and protection against unauthorised use, and
- Communication Functions, that assure the protocols and services for agent transfer.

Agent migration is the operation crucial for performance issues. For simplicity reasons the migration model shown in Figure 1 is reduced to Agent Functions. The following steps are performed:

1. Agent $agent_k$ identifies itself on the originating node S_i .
2. Identification checking at S_i .
3. Transfer of $agent_k$ from the S_i to the destination node S_j .
4. Agent restoration at S_j .
5. Agent $agent_k$ identifies itself on S_j .
6. Identification checking at S_j .

After completing migration, $agent_k$ can be served by S_j .

Figure 1 – Agent migration: functional description

3. Generic Model of Mobile Agent Network

Generic model suitable for performance analysis of a mobile agent network includes structural and behavioural part. Regarding the structure, the model consists of the processing part represented by a set of nodes, S , capable to host the agents and communication part represented by a network, N , that connects the nodes and provides agent migration (Figure 2). Users interact with the mobile agent network by requesting services at the originating nodes. User requests are random events, as well as an activation of the agents capable to

handle the services. An agent execution preceded by its migration from an originating to a destination node, and agent response to a user, are also stochastic processes. This is why a mobile agent network is described as a queuing system where an agent represents an information unit that must be served.

Node S_i collects the requests from its users and activates the agents. After completing the service, the agent returns the response to the user. Requests and responses form an input flow, G_i , and output flow, R_i , respectively. Agent migration is defined by network flows between two nodes, X_{ij} and X_{ji} . The agents accepted for execution at node S_i represent its internal flow, L_i . With respect to agent hosting and execution capability, the nodes are classified as follows:

- S_i is a sink node if it accepts the agents sent from other nodes, but does not initiate an agent migration towards other nodes ($X_{ij} = 0$, $X_{ji} \geq 0$, for all j),
- S_i is a source node if it transfers the agents to other nodes, but it can not accept the agents from other nodes ($X_{ji} = 0$, $X_{ij} \geq 0$ for all j),
- S_i is an active source node if it serves the agents ($L_i > 0$),
- S_i is a passive source node if it does not serve the agents ($L_i = 0$),
- S_i is a neutral node if it serves local agents only, without accepting/transferring the agents from/to other nodes ($X_{ij} = X_{ji} = 0$, for all j).

Processing intensity of the node S_i is denoted by B_i .

Temporal and spatial presentation of an agent migration and execution is given in Figure 3. $agent_k$ enters the network at node S_i , at t_0 . Supposing that $agent_k$ can be served on several nodes, including the local one (S_i), performance based decision can be stated as follows:

- a) Agent handling at S_i is described by holding time, $t_{qi} = t_{wi} + t_{si}$, where t_{wi} defines waiting time and t_{si} serving time.
- b) Agent transfer from S_i to S_j , handling at S_j and return to S_i is described by holding time extended by communication delay $t_{qj}' = T_{ij} + t_{qj} + T_{ji}$, where T_{ij} and T_{ji} represent agent transfer time from S_i to S_j and back.
- c) Minimum time for serving an agent is defined by $\min \{t_{qi}, \min (t_{qj}')\}$, for all j fulfilling $service_k \in s_j$. The corresponding node is the optimum solution for a serving agent.

Response time defining an interval between acquiring the request and producing the response is proposed as a performance measure for a mobile agent network. The same measure is used for evaluating parallel and distributed processing in telecommunications [8, 9].

Figure 2 – Structure of generic model and agent flows

Response time calculation is based on the following assumptions:

- Input flows are stationary Poisson processes with arrival intensity G_i [erl/s],
- Total input flow is represented by $G = \sum_{i=1}^{nc} G_i$.
- Processing nodes are single servers with exponential distribution of service time. Mean service time equals $t_{si} = 1/B_i$ and mean holding time equals $t_{qi}(L_i) = 1/(B_i - L_i)$.
- An agent must be served at the selected destination. Its forwarding is not allowed.

$$T(L, X) = \sum_{i=1}^{nc} \frac{L_i}{G} t_{qi}(L_i) + \sum_{i=1}^{nc} \sum_{j=1}^{nc} \frac{X_{ij}}{G} T_{ij}(X) \quad (1).$$

4. Case Study: Response Time Analysis

A mobile agent network in which communication delay from S_i to S_j does not depend on the node pair has been considered. Term (1) can be simplified and optimisation problem giving the minimum mean response time can be formulated as follows:

$$T(L) = \sum_{i=1}^{nc} \frac{L_i}{G} t_{qi}(L_i) + \frac{X}{G} T(X) \quad (2)$$

$$\sum_{i=1}^{nc} L_i = G \quad (3)$$

Term (2) defines a function to be minimised and term (3) optimisation constraints. Two additional functions are used in calculation:

- $t_i(L_i)$ defining growth of holding time for a node S_i for infinitesimal growth of L_i , and
- $t(X)$ defining growth of communication delay.

Conditions for grouping the nodes are the following:

$$t_i(L_i) \geq \alpha + t(X), L_i = 0, S_i \in V_{sp}$$

$$t_i(L_i) = \alpha + t(X), 0 < L_i < G_i, S_i \in V_{sa}$$

$$\alpha \leq t_i(L_i) \leq \alpha + t(X), L_i = G_i, S_i \in V_n$$

$$\alpha = t_i(L_i), L_i > G_i, S_i \in V_o,$$

where α denotes an optimisation threshold, V_o a set of sink nodes, V_n , a set of neutral nodes, V_{sa} , a set of active sources and V_{sp} , a set of passive sources. Optimisation algorithms from [10] are used in examples that follow.

Figure 3 – Agent migration in space and time

For the generic model defined as a product form Markovian queuing network, total mean response time, $T(L, X)$, equals to the sum of total mean holding time and mean communication delay, as follows:

A mobile agent network taken as an example consists of four processing nodes with processing intensity $B = \{100, 40, 60, 20\}$ [erl/s]. Input flow is given by a vector $G = \{24.37, 25.14, 27.54, 10.15\}$ [erl/s]. Graphical interpretation of the optimisation problem is given in Figure 5. The curves show the growth in the holding time for all nodes. Points G1, G2, G3 and G4 determine working conditions of a stationary multi-agent system, where all requests are handled locally, i.e. $L_i = G_i$ for all i . In such a system the response time equals to holding time.

Figure 5 – Response time optimisation problem

By introducing thresholds $\alpha = 0.03$ and $\alpha + t(X) = 0.15$ three regions describing underload (sink region), normal load (neutral region) and overload (source region) are defined. Nodes S_4 and S_2 are overloaded, S_3 is neutral and S_1 is underloaded. A feasible solution for reducing mean response time in a multi-agent system is to transfer the agents from S_4 and S_2 to S_1 .

Figure 6 – Delay influence on agent migration

Figure 6 shows how communication delay effects agent migration. Agent migration is analysed for four different communication delays:

$$T(X) = \{0., 7.905, 16.54, 147.9, 189.0\} \text{ [ms]}.$$

Mean response time obtained is the following:

$$T(L) = \{33.157, 41.3844, 50.7419, 197.846\} \text{ [ms]}$$

For $T(X) = 0$ majority of the agents are transferred to the node S_1 the processing intensity of which is the highest. Local flows are the following:

$$L = \{53.14, 10.36, 23.7, 0\} \text{ [erl/s]}$$

For the biggest communication delay, $T(X) = 189$ ms, all agents are served locally, i.e. without migration. Corresponding local flows follow:

$$L = \{24.37, 25.14, 27.54, 10.15\} \text{ [erl/s]}$$

5. Conclusion

Generic model of a mobile agent network suitable for performance evaluation has been proposed. It includes a multi-agent system and a set of processing nodes connected by a communication network providing agent migration. A mobile agent network is described as a queuing system where mobile agents represent information units that must be served. Response time analysis given as a case study presents how communication delay effects agent migration performance. In the future work agent service capability and complexity, as well as processing node server classes, will be introduced.

References

1. D. N. Chorafas, "Agent Technology Handbook, McGraw-Hill, New York, 1998.
2. W. Cockarine, M. Zyda, "Mobile Agents", Manning, Greenwich, Connecticut, 1998.
3. J.M. Bradshaw, ed., "Software Agents", AAAI Press/The MIT Press, Menlo Park, 1997.
4. W. Brenner, R. Zarnekow, H. Wittig, "Intelligent Software Agents - Foundations and Applications", Springer, Berlin, 1998.
5. R. Murch, T. Johnson, "Intelligent Software Agents", Prentice Hall PTR, Upper Saddle River, New Jersey, 1999.
6. V. A. Pham and A. Karmouch, "Mobile Software Agents: An Overview", IEEE Communications Magazine, July 1998, pp. 26-37.
7. N.M. Karnik and A.R. Tripathi, "Design Issues in Mobile-Agent Programming Systems", IEEE Concurrency, July-September 1998, pp. 52-61.
8. G. Nemeth, I. Lovrek, V. Sinkovic, "Scheduling Problems in Parallel Systems for Telecommunications", Computing, Vol. 58, No. 3, pp. 199-223, 1997.
9. V. Sinkovic, V., I. Lovrek, G. Nemeth, "Load Balancing in Distributed Parallel Systems for Telecommunications", Computing, Vol. 63, No. 3, pp. 201-218, 1999.
10. A.N. Tantawi, D. Towsley, Optimal Static Load Balancing in Distributed Computer Systems. Journal of the ACM, Vol. 32, No.2, April 1985. pp. 445-465.