OPNET-Based Performance Analysis of a Multi-Agent Architecture for Managing the Mobile Content Delivery Process

Adel Aneiba[1], Claude C. Chibelushi[2]

[1] Faculty of Information Technology Benghazi- University, Libya,
a.aneiba@uob.edu.ly
[2] Faculty of Computing, Engineering and Sciences, Staffordshire University, UK
c.c.chibelushi@staffs.ac.uk

Abstract. This paper addresses the problem of mobile content delivery failure in wireless data networks, and the resulting wastage of communication resources. In these networks, many content delivery transactions fail due to inadequate device or channel capability, possibly after a partial delivery of the content requested by the user. The paper evaluates the performance of a solution devised to enhance and optimise the delivery of mobile content, as a new approach for reducing the probability of wasting valuable communication resources. The proposed solution is a layered multi-agent architecture which is offered in two alternative configurations: a centralised-decision configuration, and a distributed-decision configuration. Furthermore, a baseline configuration (with no agents for managing the content delivery process) is used in the paper, for the purpose of comparative performance evaluation. The simulation results have shown that on average, under heavy traffic conditions and for two levels of device capability (low or high performance device), the distributed-decision configuration outperforms the other two configurations, in terms of lower agent communication overhead, admitting more transactions and reducing bandwidth utilisation. Overall, compared to the baseline system, the layered multi-agent system performs more efficiently in heavy traffic networks and for poor device capability. However, as would be expected, the multi-agent system performs worse than the baseline system under conditions of high device capability, due to the overhead introduced by the communication between agents. The results support the intuitive expectations of agent behaviour in telecommunication systems.

1 Introduction

As an abstract view, the term “mobile content” means any type of media or resource which is viewed or played on mobile devices. Mobile content is accessed via a mobile device which is connected through a wireless channel, often with low available bandwidth. The mobile device may have limited processing, storage, input and output resources. This limitation introduces several data and information management challenges related to device capability, content presentation, and bandwidth utilisation. Not all content is appropriate for all subscriber devices or channel conditions. Therefore, mobile operators need to deliver to their subscribers only the content that their devices can access, store, support, or display correctly, and which can be delivered reliably by the wireless channel [1-2]. Due to the diversity of wireless terminals, content formats supported by different terminals may not be compatible with each other. It is possible that pictures or videos sent by the transmitter cannot be displayed on the receiver. There are four main factors that are critical to the successful delivery of mobile content. They are: the mobile content, the mobile device, the wireless channel, and the end-user experience. These factors were identified based on an interview with a mobile service provider, and from secondary research sources [3-4]. These factors should be taken into consideration in any mobile content delivery management process.

A major contribution of this paper is the set of findings of a quantitative performance assessment, which show the impact of the proposed architecture. The experimental study was conducted through software simulation of a UMTS network, using the OPNET tool. The study found that the proposed multi-agent architecture can be beneficial in terms of serving more client transactions, and improving bandwidth utilisation, especially under conditions of heavy traffic and poor device capability. To the best knowledge of the authors, such a quantitative analysis, from the standpoint of mobile content delivery management, is not available in the literature.

2 Mobile Content Delivery Agent Architecture

The goal of the work presented in this paper is evaluate the performance of a multi-agent architecture and its configurations which were presented in [5], the architecture is called Mobile Content Delivery Agent Architecture (MCDAA), which adopts a layered multi-agent approach. The key issues, addressed by MCDAA, include the gathering and integration of diverse information sources by collaborating agents, and the provision of automated decision-making for more efficient use of the resources provided by mobile service providers. This is done such that prospective delivery of mobile content is possible within the current conditions of the device, channel, content and the available resources.

MCDAA has four layers: the Client Layer, the Radio Resource Layer, the Management Layer, and the Content Layer. It comprises four agents, which are the Device Profile Agent (DPA), the Wireless Channel Profile Agent (WCPA), the Decision Making Agent (DMA), and the Content Profile Agent (CPA). The architecture is designed as two configurations: a centralised-decision configuration and a distributed-decision configuration. The configurations differ in the distribution of the decision regarding the capability of the various components (device, channel and content source). In addition to the two MCDAA configurations, a baseline configuration is used in this paper, for the purpose of comparative performance evaluation. The baseline configuration does not use agents for managing the content delivery process [5]. MCDAA is designed to operate in wireless data networks, such as 3G, LTE where mobile clients access information sources via wireless channels.

To evaluate and validate the feasibility of the proposed solution, simulation experiments were carried out. For these experiments, each MCDAA configuration and the baseline configuration were implemented, as a wireless data network, using the OPNET simulation tool.

3. The Simulation Setup

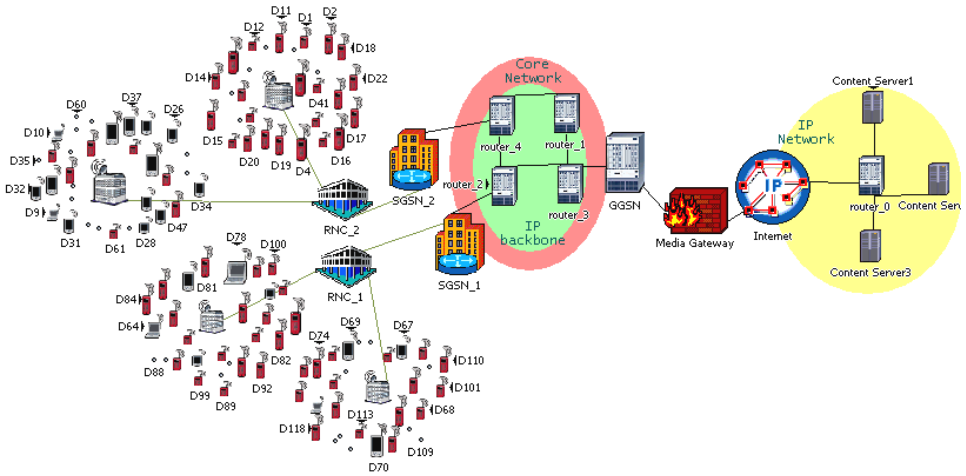

Fig. 1. Network topology used in the experiments

One important goal of the simulation study is to evaluate and validate the proposed architecture. It examines and evaluates the impact of the use of software agents on the mobile content delivery process, and how the agents perform within different situations and scenarios. The first objective of these experiments is to test if the performance of the mobile content delivery process can be improved by using software agents. The second objective is to optimise the configurations, with respect to the allocation of decision-making responsibilities, the software agents within the environment, so as to minimise wastage of resources. The experiments focus on the type of scenario which is of most interest in this study for assessing the performance gain, where the end user requests content, which is then delivered to the device.

To evaluate MCDAA in a systematic way, [6] has suggested an approach which can be described in terms of a three-dimensional space which includes: the set of possible applications; the set of possible multi-agent system (MAS) architectures or configurations; and the set of attributes used to evaluate the architectures or the configurations. The suggested approach is to investigate substantial parts of this space rather than just single points. This approach, besides enabling a more systematic investigation of the space, will lead to a deeper understanding of MASs and their applications, which, in turn, will contribute to the long-term goal of obtaining general design principles of MASs [6-7]. In this paper, this claim is supported since MAS can work in different ways, as seen in [1]. The experimentation with several configurations under several performance metrics will give a solid background for development strategies for MAS in the future, in the area of telecommunications. Therefore, this approach has been adapted to the problem of mobile content delivery process by presenting two configurations of MCDAA alongside the baseline configuration. These will then be compared based on criteria, as mentioned in the previous paragraph which will involve analysing data on all of following networks attributes: TCP performance, bandwidth utilisation, agent communication overhead, number of successful and failed transactions, admitted transactions, and scalability.

The study uses the UMTS network model provided by OPNET and rewrites the application level to implement the MCDAA, which is similar in form to a client-server application system. The basic network configuration, which is used in the simulations, consists of 120 mobile stations accessing three remote content servers gateway (proxy) through a radio access network, as shown in Fig. 1. The network architecture has been adopted from the OPNET network standard models library and has been modified based on is a well-known topology for mobile content delivery according to [8-9].

The 120 nodes (mobile devices) are distributed over four cells. The number of nodes was chosen because it is the maximum that a SGSN component can attach according to the network configuration used. Table1 summarises the important simulation parameters used in the experiments. These parameters are those required by the OPNET package to determine an application-level configuration.

Table 1. Main simulation parameters.

Parameters	Value
Inter-Request Time (Seconds)	Exponential (120)
File Size (Bytes)	Uniform (6000-10000)
Type of Service	Best Effort (0)
Mobile Device Numbers	120 Users
Simulation Time	3 Hours

In order to study the effect of adopting multi-agent systems in the mobile content delivery management process, by assessing the overall network performance, several performance metrics have been chosen.

4. Simulation Scenarios

For assessing the efficiency of MCDAA, two scenarios were found to cover different circumstances for the performance study. The scenarios use different mobile device capabilities and traffic conditions. Scenario A is for investigating how MCDAA behaves under a heavy network traffic and high device capabilities settings. Scenario B is for investigating how MCDAA performs under a heavy network traffic and low device capabilities settings. These scenarios will be used to assess how MCDAA configurations can perform in a worst-case scenario where the network is operating under heavy traffic condition (i.e. more transactions generated by clients). The scenarios are summarized in Table 2. Each of the scenarios is simulated under each MCDAA configuration (centralised-decision or distributed-decision).

Table 2. MCDAA Evaluation Scenarios.

Scenarios	Network Traffic	Device Capability
Scenario A	Heavy Traffic	High
Scenario B	Heavy Traffic	Low

Each scenario will be tested under worst network conditions (heavy traffic) and under two device profiles (low and high capabilities). The "low" indicates that the device can only be able to process 60% of its transactions and "high" indicates that the device is capable of handling 95% of its transactions. The primary investigation was conducted on a range of device capabilities (for low capability: 30%, 40%, 50% and 60%; and for high capability: 70% 80%, 90% and 95%), and it was found that the best representations are 60% for the low and 95% for the high capability. The selection of two device profiles was chosen to show how the system performs for various client device capabilities at different ends of the spectrum.

Heavy network traffic indicates that mobile devices issue 3600 transactions. The number of transactions was generated automatically by the system based on the simulation settings. The traffic model is based on the HTTP application provided by the OPNET trace library. The measured performance of the two architectural configurations will be compared against the findings from [10].

5. Performance Metrics and Measurements

According to [6,7,11], it is possible to evaluate MAS architectures with respect to several quality attributes, both performance-related attributes and more general quality attributes, such as scalability. The following performance-related metrics have been identified as important for MCDAA configurations. These performance metrics will show the efficiency of the proposed approach. However, some of these metrics seem to be correlated, especially 2 and 3. The reason that both have been included is that, agent communication will be measured in the uplink direction as this link has a minimum amount of bandwidth dedicated to the client request, while the bandwidth utilisation can be significant in the downlink direction as the content download process operates in this path.

1. **Agent Communication Overhead:** This measures the additional load which the multi-agent system places on the underlying communication infrastructure. This will be measured by the uplink bandwidth required for the MCDAA agents to collaborate. The overhead is measured in terms of bits sent by mobile stations and received by the Node B.
2. **Bandwidth Utilisation:** Shows the links average and peak bandwidth usage in bits per second. This will be measured by displaying, for each configuration, the average bandwidth usage in bits per second.

3. **Admitted Transactions:** This statistic shows the ability of MCDAA to allow more client requests to be admitted onto the network, when eliminating failed transactions. This will count how many client requests have been admitted or granted by the system under each configuration.

6. Verification and Validation of MCDAA Simulation Model

One of the most important aspects in developing a simulation model is its credibility; hence, the validation and verification of any simulation model is necessary. Multi-agent models are difficult to validate because these models represent a new approach to simulation for which traditional validation methods are not always applicable [12]. Given these challenges, it is essential to know what an appropriate validation process is for such models. According to [13] there are various validation techniques used in model verification and validation.

6.1 Verification Strategy

MCDAA model code has been critically tested and verified by employing several mobile content delivery scenarios. In the reviewed literature, many common-sense suggestions have been gathered which can be given for use in the verification process:

1. A wide variety of output statistics under a variety of settings of the input parameters have been generated.
2. The MCDAA simulation model has been designed and developed based on an industrial standard (OPNET).

6.2 Validation Strategy

Comparison to another validated model was selected as the validation technique for the work reported herein, due to its applicability and appropriateness to the MCDAA model. A set of results produced by the simulation model has been validated by comparing the output of MCDAA model to known results of the simulation model which was validated by [10]. This was done with respect to load balancing under agent communication overheads, as both multi-agent systems fall under the umbrella of the general problem of multi-agent systems for "dynamic resource allocation".

The validation experiment was run as a series of simulations with similar traffic pattern and rate for all configurations corresponding to an aggregated offered load of 0.2, 0.4, 0.6, 0.8, 1, 1.2, 1.4, 1.6, 1.8 and 2 Erlang[2], respectively. These simulations give a measure of the ability to utilize resources (i.e. bandwidth) and balance the load, as seen in Fig.3.

---

[2] Erlang is a dimensionless unit which is used in cellular communication systems as a statistical measure of offered load

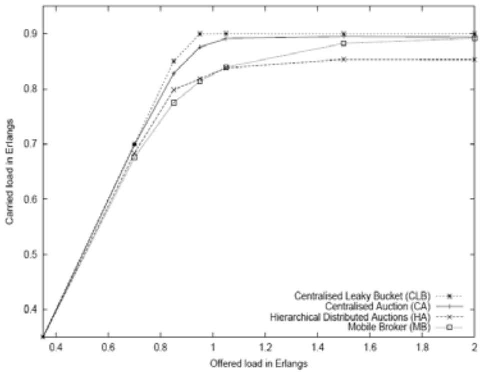

Fig. 2 Agent communication overhead, as measured in [10]

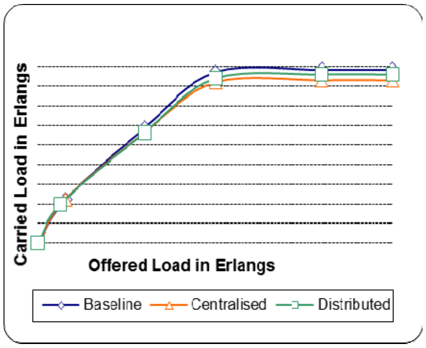

Fig. 3 MCDAA agent communication overhead

Despite the fact that both studies ([10] and this research) have different implementation approaches, it was found from two graphs (Fig. 2 and Fig. 3) that both studies behave in a similar manner and they match the theoretical expectation that the more agent communicate, the less offered load can be obtained. As can be seen from Fig. 3, the MCDAA system in both its configurations offers less load on the network compared to the baseline configuration. This is because the interaction between agents increases the volume of traffic, which in turn reduces the proportion of real traffic carried. In addition, the distributed configuration offers more load than the centralised one; this is due to the fact that the centralised configuration exchanges more messages for each transaction than the distributed one.

The two studies have found that scalability seems to be better supported by distributed architectures than centralised architectures. Firstly, the distribution of control at the distributed architecture is divided between a number of components (device, radio controller, and the proxy). Secondly, the risk for communication bottlenecks is smaller. The above analysis supports the research findings reported in this paper, but the best way to conclusively show the benefits of the MCDAA and determine the level of improvement is through a real world implementation. It is hoped that the results of this study may help influence a mobile service provider to experiment with MCDAA or a similar type of transaction management system in the future.

7. Simulation Results and Analysis

The findings from the two experiments indicated that when the device capability is poor, more savings of network resources can be made. Therefore, for the sake of simplicity and to avoid repetition, the results of the two experiments (i.e. for the two device capabilities) are presented as an average of the results obtained for the chosen traffic and device setting. In the graphs, the *x*-axis represents the temporal progression of the simulation and the *y*-axis represents the relevant performance metric.

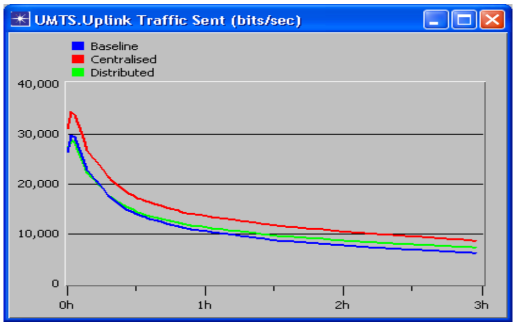

Fig. 4 Agent communication overhead under heavy traffic. The graph shows the averages of the measurements obtained for high and low device capability (Scenarios A and B).

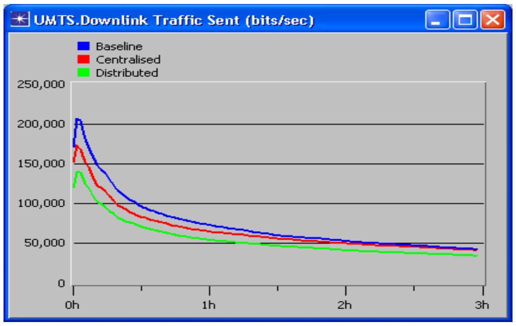

Fig. 5 Bandwidth utilisation under heavy traffic. The graph shows the averages of the measurements obtained for high device capability and low device capability (Scenarios A and B).

Figure 4 displays the impact of MCDAA agent communication overhead, in the uplink direction under the heavy traffic condition. The figure shows that the centralised-decision configuration has the highest communication overhead. This is due to the fact that each request is accompanied by an agent message inserted by the DPA, which explains why the centralised-decision configuration has the highest uplink rate.

The distributed-decision configuration has a slightly lower agent communications overhead than the centralised-decision configuration and a higher overhead than the baseline. This is due to the DPA stopping prospective transactions from taking place, due to the poor capability of the device, it can be concluded that when the device and the channel are capable of handling the content request, agent communication can be a serious overhead. In contrast, when the channel is really loaded and the device is fully capable, MCDAA agent communications can be beneficial in terms of serving more users and admitting more transactions, as seen in Scenario A.

Fig. 5 shows the impact of MCDAA on bandwidth utilisation for the download content process. As seen from the graph, both MCDAA configurations download fewer transactions than the baseline configuration. This is because the baseline network tries to respond to all the user requests regardless of whether these transactions are valid or not, therefore a portion of the traffic is wasted on these failed transactions. However, theoretically, a high throughput is a good thing but not always correct since the system is not aware of the device capability, channel condition and the content size, it just delivers what it can deliver. In this case, the generated throughput can be just a waste. This waste is managed much better by both MCDAA configurations as seen from the graph.

In addition, more transactions have been allowed by the centralised-decision configuration than the distributed-decision configuration. This is because the WCPA in the distributed configuration has decided that many transactions will fail due to the channel condition and have no chance to get through. Therefore, it has stopped them while the centralised-decision configuration has obtained the channel data rate and based on previous history has decided to respond to more transactions. The interesting finding in this situation is that 80% of the available bandwidth has been successfully utilised by the distributed-decision configuration and 74% by the centralised-decision configuration, while the baseline managed to utilise just 58% from the available bandwidth. This finding indicates how important it is to develop a system to predict the channel conditions, especially when huge data traffic is transmitted, as in content streaming and downloading.

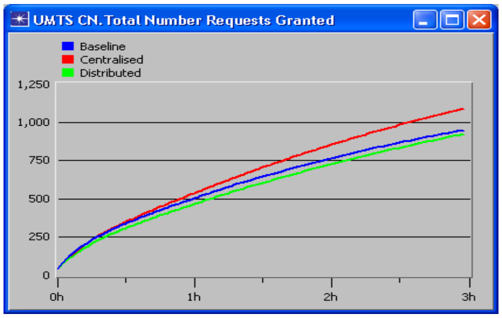

Fig. 6 Admitted transactions under heavy traffic. The graph shows the averages of the measurements obtained for high device and low device capability (Scenarios A and B)

Fig. 6 shows the impact of the MCDAA on the system admission control function in the core network for data traffic, Scenario A shows that when MCDAA is installed, more transactions are admitted than for the baseline network. Although due to the fact that the graph shows the average of the worst and best case scenarios, the centralised-decision configuration is seen to admit more transactions than the other two configurations. This is because the DPA cannot make any decision locally, therefore in many cases some of these transactions will eventually fail due to poor device capability.

7. Conclusions

This paper reports a performance evaluation of the MCDAA architecture. It presents a simulation model for MCDAA and shows how it behaves under different conditions. In addition, the paper illustrates the simulation model and scenarios used for simulating UMTS wireless networks, with and without MCDAA. From the simulation experiments, it has been observed that using software agents in telecommunication networks can improve network performance, in terms of serving more client transactions and bandwidth utilisation.

The results show that a distributed-decision configuration utilizes the bandwidth better than the centralised-decision configuration when the network is operating under heavy-traffic and poor device capabilities. In addition, the centralised-decision configuration tends to have more messages (an agent message accompanies each client request), and that could require a large bandwidth, whereas the distributed-decision configuration tends to be better at utilizing a given bandwidth over the time.

Furthermore, communication in a distributed-decision configuration has a tendency to be more local than in a centralised-decision configuration, using smaller parts of the network (the DPA takes decision at the client device, or the WCPA at the RNC…etc). The evaluation methodology of this research can be a good foundation for future research in this area. Indeed, it can essentially serve as a guide for those interested in choosing a particular solution for the management of mobile content tasks in wireless data networks.

References

[1] Florence Agboma and Antonio Liotta. 2012. Quality of experience management in mobile content delivery systems. Telecommun. Syst. 49, 1 (January 2012), 85-98.

[2] Lengheimer, M.; Binder, G.; Rosler, T., "Content management systems for mobile, context-dependent augmented reality applications," Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on , vol., no., pp.1521,1526, 26-30 May 2014.

[3] Higashino, M.; Hayakawa, T.; Takahashi, K.; Kawamura, T.; Sugahara, K., "Management of Streaming Multimedia Content Using Mobile Agent Technology on Pure P2P-based Distributed E-learning System,"Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on , vol., no., pp.1041,1047, 25-28 March 2013.

[4] Bosunia, M.R.; Anbin Kim; Jeong, D.P.; Chanhong Park; Seong-Ho Jeong, "Efficient data delivery based on contentcentric networking," Big Data and Smart Computing (BIGCOMP), 2014 International Conference on , vol., no., pp.300,304, 15-17 Jan. 2014

[5] Aneiba, A; Chibelushi, C, "A Multi-Agent Architecture for a Reliable Mobile Content Delivery Management", NNGT Int. J. on Artificial Intelligence, Vol. 3, Jun 2015, Journal ISSN Online: 2356-5888

[6] Davidsson, P., Johansson, S., Svahnberg, M., Characterization and Evaluation of Multi-Agent System Architectural Styles, Software Engineering for Multi-Agent Systems IV, LNCS series, Vol. 3914, Springer, pages 179-188, 2006.

[7] Davidsson, P., and Wernstedt, F., A Framework for Evaluation of Multi-Agent System Approaches to Logistics Network Management Multi-Agent Systems: An Application Science, Kluwer, 2004.

[8] Ilyas, M., Mahgoub, I, 2005, Mobile computing handbook, Boca Raton: Auerbach Publications.

[9] Talukder, Asoke K., Yavagal, Roopa R., 2007, Mobile computing: technology, applications, and service creation, McGraw-Hill.

[10] Carbo, J., Sanchez-Pi, N., Griol, D. , Molina, J., Comparing Agent Interactions of Distributed and Centralized Multi-Agent Systems for Context-Aware Domains, Highlights on Practical Applications of Agents and Multi-Agent Systems Communications in Computer and Information Science Volume 365, 2013, pp 158-165

[11] Clements, P., Kazman, R., Klein, M., Evaluating Software Architectures. Addison Wesley, 2002.

[12] Li, Y., Brimicombe, A.J., Li, C., Agent-based services for the validation and calibration of multi-agent models, Computers, Environment and Urban Systems, Volume 32, Issue 6, November 2008, Pages 464-473.

[13] Sargent, Robert G., "Verification and validation of simulation models," Simulation Conference, WSC 2008, pp.157-169.