Transaction and Networking Support for Group-oriented Mobile Commerce Services

Upkar Varshney, *Member*, *IEEE*, and Krish Ravikumar

***Abstract*—Broadband wireless services, including group-oriented mobile commerce services, are receiving a significant interest among researchers, developers, wireless service providers, and users. Some of these diverse services, including mobile auctions, mobile financial services, and multi-party interactive games, are transaction-oriented and will require the network and protocol support for transactions. In this paper, we focus on supporting transactions for group-oriented m-commerce services by analyzing and characterizing the transaction requirements, designing network protocols for supporting diverse transactions, and using simulation modeling techniques to evaluate the suitability of protocols. Our results show that protocols can support a range of performance requirements, including transaction completion probability and transaction delay, for group-oriented m-commerce services. The transaction delays are governed by user connectivity/dis-connectivity ratio and average disconnection duration, while the transaction blocking is governed primarily by the average duration of disconnection. It is also shown that for multi-parametric cases, increased application flexibility, in terms of a reduced number of entities for transaction completion, can compensate for the negative impact of increased or more frequent user disconnections by improving both transaction delay and blocking performance. Currently, we are evaluating the protocols under co-related dis-connectivity and failures of networking components. These results will be presented at the workshop.**

***Index Terms*—group transactions, mobile commerce, network protocols, simulation modeling.**

I. Introduction

The emerging broadband wireless services, including mobile commerce services, are receiving significant attention in terms of identifying new applications, designing frameworks, and engineering networking solutions [1]. Many m-commerce services including mobile auctions, multiparty games, mobile financial applications, mobile advertising, and mobile entertainment services (Figure 1) will require group communications, where several users have to simultaneously interact for an outcome or the same information is sent to multiple users [2-10]. The group-oriented m-commerce services are likely to be transaction-oriented, where the monetary value of many transactions could result in a financial impact of billions of dollars a year [11]. Before m-commerce services can be implemented and widely deployed, many challenges, including transaction support for diverse m-commerce services, must be addressed. The support for transactions is inherently complex due to the need to co-ordinate group membership, manage user inputs, and implement application synchronization. The support for transaction becomes even more difficult challenge due to brief dis-connectivity or intermittent connectivity in the underlying infrastructure and ad-hoc wireless networks, user mobility, and the varying performance of wireless links. These limitations will lead to poor transaction performance in terms of increased delays and lower transaction completion probability. In this paper, we address the problem of supporting transactions for group-oriented m-commerce services by analyzing and characterizing transactions requirements, designing transaction-management protocols, and evaluating the effectiveness of protocols for diverse m-commerce services using simulation modeling.

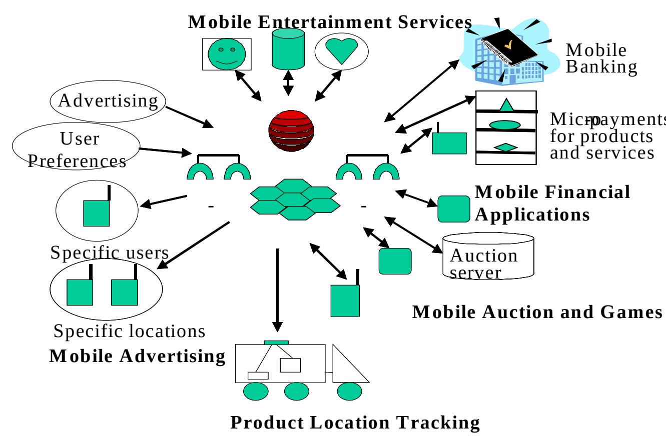

Fig. 1. Several group-oriented mobile commerce services

Although routing and related issues have received considerable attention [2-10], little work exists for transactions in mobile services such as basic requirements [7], mobile teamwork functions [12], and virtual guide function for a tourist group [13]. So far, the transaction support primarily deals with database access under connectivity problems such as pre-fetching data [14], detecting data conflicts between transactions [15], deploying transaction recovery [16], using

Manuscript received June 2004. This work was supported in part by the RCB Summer Research Grant, 2003.

U. Varshney is with the Department of Computer Information Systems, Georgia State University, Atlanta, GA 30302, USA (phone: 404-463-9139; fax: 404-651-3842; e-mail: uvarshney@gsu.edu).

K. Ravikumar was with Georgia State University during 2002-2003.

synchronization [17], reducing interaction under intermitted connectivity [18], supporting disconnected mode of operation [19], using middleware [20], using pre-write for data availability during frequent disconnection [21], performing recovery for transactions [22], measuring effect of disconnection on the properties for transactions [23], using recovery protocol [24], deploying transaction manager to retrieve state information [25] and access to commerce activities using state-machines and event management to handle disconnection [26]. A preliminary performance evaluation of group-oriented mobile services using basic time-out mechanisms has been presented in [30]. It can be observed that most of the work done so far primarily deals with database access and recovery for mobile transactions. The networking and transaction support for group-oriented m-commerce services have not been addressed. In this paper, we provide detailed treatment of networking protocols and transaction support for group-oriented mobile services. The major contributions of this paper are (a) comprehensive requirements of group-oriented m-commerce services (section II), (b) transaction-management protocols for group-oriented m-commerce services (section III), and, (c) detailed performance evaluation of protocols using simulation modeling (section IV). A discussion of current research and conclusion is presented in section V.

II. Requirements and Challenges of Group-Oriented M-Commerce Services

Examples of group-oriented m-commerce services are mobile auctions, interactive games, mobile entertainment services, mobile financial services, mobile advertising, proactive service management, product recommendation systems, and mobile inventory management [1].

A. Overview of Group-oriented M-commerce Services

Mobile auctions will involve users with hand-held devices joining one or more groups. A demanding service due to a number of users in several locations, auctions will require co-ordination of user inputs, continued connectivity, and symmetric real-time multicast. The transactions are likely to be few times in a week lasting from seconds to minutes. Using mobile agents, a user could become part of several auctions and mobile agents can also reduce the increased bidding traffic as the auction progresses [26].

Multi-party interactive games include variety of games to mobile users in a user- or application-defined group. The games will involve millions of users worldwide and utilize location-based information in forming user groups. These will require continued connectivity, very low delays for producing real-time or near real-time performance for mobile participants, and symmetric real-time multicast with active participation from multiple users. The transactions are likely to be few times in a week lasting from seconds to minutes.

Mobile Financial Services will allow mobile users to conduct multi-party financial transactions including mobile payments. Financial services are likely to have the most economic impact, as mobile transactions worldwide will involve several hundred billion dollars [11]. These will require low delays, continued connectivity, and symmetric multicast & active participation from users. The transactions are likely to occur few times a week from seconds to minutes.

Mobile and Locational Advertising will turn wireless devices into a powerful marketing medium. User and location-specific messages will be sent for highly effective and targeted advertising [1]. These applications will involve millions of mobile users. The advertising is likely to be one-way, time and location-dependent [27, 28]. It requires asymmetric non-real-time multicast involving hundreds or more devices, and could occur few times a day from less than a second to few seconds. Higher delays and intermittent connectivity or brief dis-connectivity may be tolerated.

Mobile Entertainment Services/Mobile Distance Education will offer sophisticated entertainment and educational contents to millions of mobile users on a per event or subscription basis. The resource requirements of applications will be significant, and will also require asymmetric real-time multicast. The transactions can occur few times a day from few seconds to minutes, and major requirements include large bandwidth and low delays. The intermittent connectivity or brief dis-connectivity will have significantly effect on the overall user experience.

Proactive Service Management will provide users advance information related to activities such as wear and tear in different car parts (e.g., brakes) and the predicted date of replacement for individual components [1]. Since it will generate periodic information for users, it will require asymmetric non-real-time multicast. Transactions can occur few times in a week lasting few seconds to minutes. Higher delays and intermittent connectivity or brief dis-connectivity can be tolerated.

Product Recommendation Systems will provide recommendations on products & services from a 3rd party or customers. These will require asymmetric non-real-time multicast involving a large number of entities. Transactions can occur few times in a week from seconds to minutes, and higher delays and intermittent connectivity or brief dis-connectivity can be tolerated.

Mobile Inventory Management/ Product Location Services will allow mobile decision makers to reduce inventory by managing in-house and inventory-on-move and location tracking of product and services. Some delays can be tolerated and since fewer entities are involved, unicast can be used if multicast support does not exist. The transactions can occur several times in a day lasting few seconds to minutes. The response time of few seconds is tolerable, however intermittent connectivity or brief dis-connectivity may significantly increase the delay.

Patient Monitoring Services will allow doctors, nurses and healthcare providers to monitor the status of patients in homes, hospitals, and nursing homes [29]. Periodic transactions involving transmission of vital signs should be reliably supported, and non-periodic transactions created by major changes in vital signs must receive higher priority and initiate a set of actions from healthcare providers. These services require asymmetric multicast & active participation from users (devices). The transactions can occur several times in an hour and last from seconds to minutes. Both low delays and continued connectivity required, especially when vital signs are changing.

B. Requirements of Group-oriented M-commerce Services

Several group-oriented m-commerce services and their general requirements are shown in Figure 2. These can be divided into 4 classes:

Symmetric multicast and real-time: mobile auctions, interactive games, and possibly financial services

Asymmetric multicast and real-time: mobile distance education, entertainment services, and patient monitoring

Symmetric multicast and non-real-time: mobile inventory management

Asymmetric multicast and non-real-time: mobile advertising, proactive service management and product recommendation systems

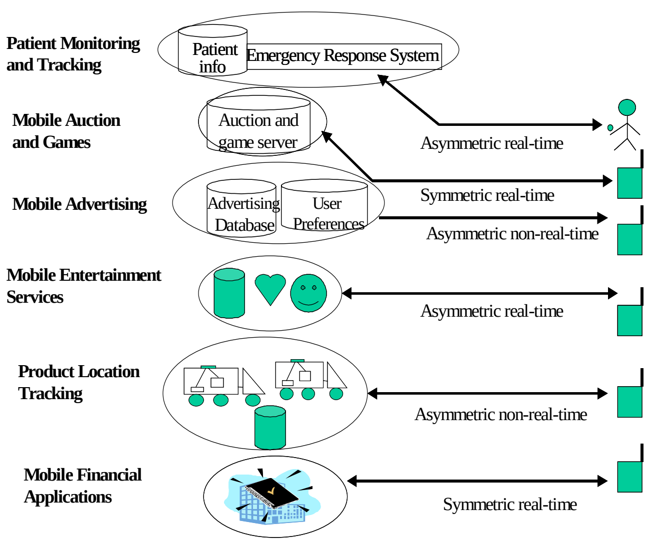

Fig. 2. General requirements of group-oriented m-commerce services

A transaction can be defined as a multi-step task where all steps must be executed in an ordered sequence. The outcome of transaction could be quite different if one or more steps are not properly executed or a certain sequence of execution is not followed. To avoid this, all or some steps of a transaction can be undone. Also a transaction should be isolated from other transactions that may affect the common values/parameters. M-commerce services present complex transaction requirements due to the diversity of applications, users, and the usage model. The diversity can be shown in terms of group communications (symmetric vs asymmetric), number of stages (single vs multiple), performance (real-time vs non real-time), connectivity and reliability requirements (Figure 3). One extreme is mobile advertising requiring asymmetric, single stage, non-real time support where reliability and connectivity are not critical. On the other side, mobile auctions and multi-party interactive games require symmetric, multi-stage, real-time support where reliability and connectivity are critical.

In this, paper we focus on supporting several different group-oriented mobile services by utilizing a more general approach. The group-oriented m-commerce services can be modeled using one or more stages and participants. The number and the inter-dependency of stages will vary significantly among applications (one simple stage with a large number of participants for mobile advertising to multiple stages with fewer users for mobile auctions). Each step requires a subset of mobile users to provide inputs/actions in reaching a consensus or outcome. The duration of transactions vary significantly, such as longer transactions for mobile multi-party interactive games and fairly short transactions for mobile advertising. For a general case, we assume a three-stage sequence: pre-processing, inputs from users, and post processing. The pre-processing stage could involve service advertisement and discovery, joining of members and setting up of rules and regulations, while post-processing will involve wrapping up the transaction including any verification. In some cases, it is possible to identify both the number and actual “critical” users whose response will be sufficient for the correctness of application. The number of stages could be chosen by the total number of users, minimum and maximum number of users per stage, delay tolerance per stage, processing delay per stage, and the maximum tolerable transaction delay. The current connectivity of mobile users could also be used in determining the number of stages.

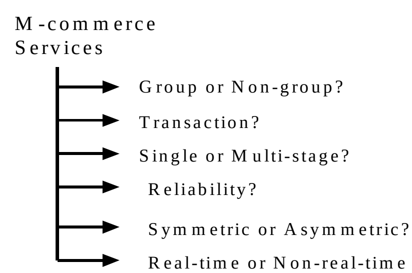

Fig. 3. Characterizing m-commerce services

Since there has been little work in group-oriented m-commerce services, it is difficult to characterize the requirements of group-oriented mobile commerce services. A preliminary set of preliminary metrics exists in [30] and we will use ETR (Entity to Transaction Ratio) and AFI (Application Flexibility Index). ETR describes number of users in a group transaction while AFI signifies the ratio of minimum number of users required for a transaction to the total number of users. Additionally, we include Transaction Frequency and Transaction Duration as additional metrics in characterizing the requirements of group-oriented m-commerce services (Table I).

TABLE I
DESCRIBING GROUP-ORIENTED M-COMMERCE SERVICES USING MULTIPLE METRICS

Services / Metrics	Mobile Auction/ Interactive Games	Mobile Financial Services	Patient Monitoring Services	Mobile and Locational Advertising	Mobile Entertainment Services/Mobile Distance Education	Proactive Service /Product Recommendation Systems	Mobile Inventory Management/Product Location
ETR	Thousands	Tens	Hundreds	Thousands	Hundreds	Hundreds	Tens
AFI	Low	Low	Low	High	Low	High	Low
Trans-Freq	1/day	1/week	1/Minute	1/Hour	1/day	1/week	1/hour
Trans-Dur	Minutes	Seconds	Seconds	Seconds	Minutes	Seconds	Seconds

C. Dis-connectivity and Intermittent Connectivity of Users

A transaction will involve several mobile users, some of which may experience brief dis-connectivity and/or intermittent connectivity [7, 27]. Although, instances of such connectivity problems are likely to be independent, however, co-related dis-connectivity or intermittent connectivity for users may exist due to the sharing of common environments (Figure 4). User 1 is unable to access application/wireless resources due to channel (radio frequency) failure, while Users 2 and 3 are unable due to partial/complete failure of components (base station, switch, or databases). User 4 has independent dis-connectivity due to out of range problem in ad hoc networks, while users 5 and 6 experience co-related dis-connectivity as the common channel is temporarily unusable. It is known that failures of base stations, switches, and database in infrastructure-oriented wireless networks are not very common [31], but the channel or RF links between base stations and mobile users can experience a range of problems affecting user's connectivity. There must be some differentiation between the transaction failures/dis-connectivity due to RF problems and the ones due to core network problems involving base station, switches and databases.

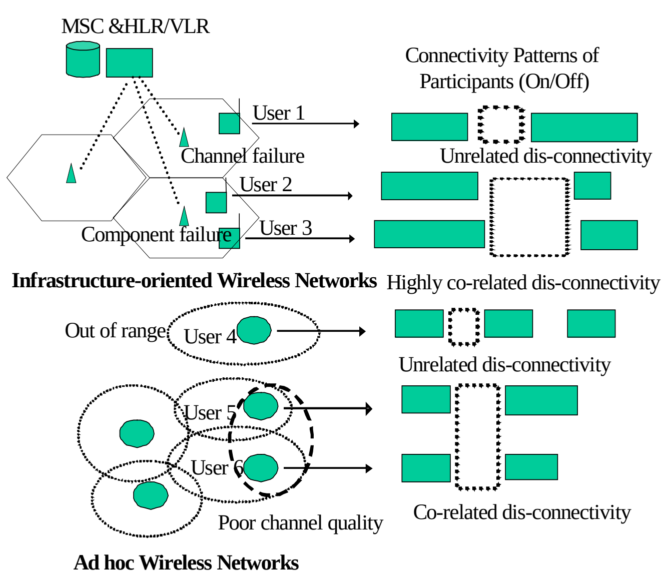

Fig. 4. Independent and co-related dis-connectivity of group users

Based on the duration and frequency of on-off periods, we propose to divide these into three classes: short-term off periods lasting from few to few hundred milliseconds and resulting due to channel behavior, handoff-problems; medium-term off periods lasting from few hundred milliseconds to few seconds due to lack of resources, obstacles, and user mobility; long-term off-periods lasting from few seconds to several minutes due to failure of devices, base stations, or other components. In this paper, we address both independent and co-related brief dis-connectivity and/or intermittent connectivity by designing protocols for transaction completion under short- and medium-term connectivity problems.

D. Example of an Application and Detailed Requirements

Mobile auctions can be modeled using multiple stages and participants with significant interdependency between stages. Pre-processing involves service advertisement and discovery, joining and verification of members, and setting up of auction rules and regulations, while post-processing will involve wrapping up the transaction including any verification. In between pre- and post processing, a typical stage involves single request and/or multiple responses (one from each participant) and requires a subset of mobile users to provide inputs/actions in reaching a consensus or outcome.

The duration of mobile auction transactions could vary significantly depending on number of users involved, type of auction, and number of bids per user. Since the lack of input from a certain user in a stage could result in a different outcome of the auction, safeguards must be put in place to avoid such problems. There are multiple different types of auctions with different set of rules and it may be possible to include additional rules to avoid unintended outcomes, we instead rely on network and protocols to provide excellent support for group-oriented applications including mobile auctions. In some cases, it is possible to identify both the number and actual "critical" users whose response will be sufficient for the correctness of application. Our future work will address how to guarantee correctness of group-oriented transactions using modifications in auction rules and protocols.

III. TRANSACTION SUPPORT FOR GROUP-ORIENTED M-COMMERCE SERVICES

In this section, we address how to support group-oriented mobile commerce services by designing protocols for transaction management. We have used the following factors in the design of protocols:

- Support for both real-time as well as non real-time transactions
- Support for single stage as well as multi-stage transactions
- Support for reliability and having the ability to deal with failures and connectivity problems
- Support for managing group membership
- Support for co-ordination of inputs from multiple users

A transaction stage can involve reliable transmission of information on application/transaction to member users, reliable reception of user inputs (response) using one or more time-outs and retransmissions. All inputs must be processed and critical users for next stage are also determined based on current state of transaction, user polling, and current list of group-members. In a stage multiple non-overlapping requests and responses could exist. The number of stages and time-out for different users in a stage are important and challenging issues. To ensure transaction completion, applications could allow certain range of time. Also, if failure of a critical user occurs, then transaction must be aborted. Also, if non-critical stages exist, then transaction should not be aborted after a time-out during those stages.

The protocols are shown in Table II and some of the operation is illustrated in Figure 5. Protocol 1 is simple and does not introduce transaction blocking on its own. It offers use of other entities, but employs no time-out, group management, or reliability. It can support asymmetry of group communications and is primarily designed for single-stage group transactions such as mobile and user advertising, mobile inventory and proactive service management. Protocol 2 involves deriving dynamic time-outs for different stages of a transaction, and, can be used for symmetric or asymmetric group communications and reliable transactions. It manages group memberships, co-ordinates user inputs, and supports distributed processing of group application by treating all users equal. Protocol 3 can offer all of the functionality of Protocol 2, but works in a more centralized applications environment as users are treated with different status (priority) in its operation. To overcome failure of single source, it has the additional ability to assign protocol functions to other user(s) if the source crashes, which may overcome the effects of single-point failure and result in a more dependable support for transactions. Both Protocols 2 and 3 derive dynamic time-outs for individual stages of a transaction using transaction response time, number of stages, time left, and predicted time interval for a user input. The protocols can co-ordinate multi-stage inputs from users, and communicate with the applications on certain users whose participation is necessary for a transaction. The protocols can operate deadlock free even if users crash or do not provide inputs and the intelligence to support diverse set of transactions. Protocol 2 and Protocol 3 can be used for mobile auctions, interactive games, mobile financial services, patient monitoring services, mobile entertainment services, and more sophisticated m-commerce services.

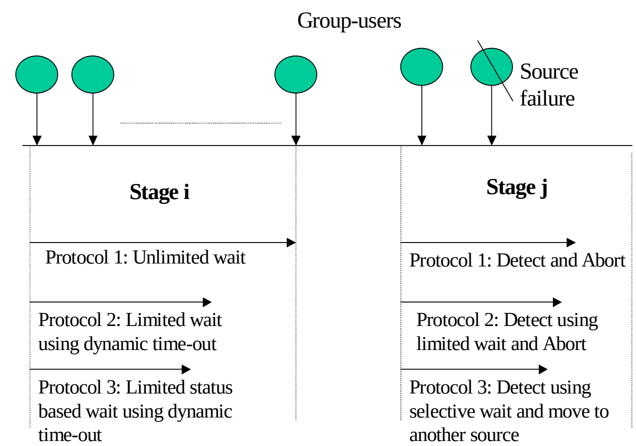

Fig. 5. Multi-stage transactions and operation of protocols

IV. MODELING AND PERFORMANCE EVALUATION

The performance of protocols can be computed by deriving transaction completion probability and transaction delay under different values of ETR, AFI, SMR, TSR, DMR, CDR, CDI and TDP. These metrics describe group-size, application flexibility, number of sources, transaction frequency, level of dis-connectivity, ratio of connectivity and dis-connectivity, co-related intermittent connectivity, and a product of transaction durations and dis-connectivity.

Simulation Modeling: For detailed performance evaluation, we identified several features and characteristics of applications and protocols including the number of stages for applications, the number of users required for every stage, and, transactions frequency, duration, delay, completion probability. A basic structure of the simulation model to support transactions, connectivity and dis-connectivity patterns, multi-stage management of user inputs is shown in Figure 6. The model can generate transactions and on-off connectivity periods with exponentially distributed inter-arrival intervals and durations, and supports time-out based mechanisms for multi-stage multi-input process. It can also capture the behavior of multiple heterogeneous wireless networks and involves both multicast and unicast transactions with diverse requirements.

TABLE II
MAJOR STEPS OF THE PROTOCOLS

Protocol	Set of Actions	Comments

Protocol 1
(Basic
Protocol)
	Receive TRNS_RQRMNTS (Users, Delay)
	Derive LIST_DESTN
	Send REQ_INFO to Server1, 2, …S
	Receive RES_INFO and Derive USER_INFO
	While (USER_FAILURE = NO)
	If PUSH=YES
	Send USER_INFO to LIST_DESTN
	Else While (REQUEST = YES)
	Process TRANS_REQ & Derive USERAPP_REQ
	Send USERAPP_REQ to LIST_DESTN
	For Every User in LIST_DESTN
	While (USER_INPUT=NO) and (USER_LV=NO)
	Wait
	Process multiple USERAPP_RES
	Output TRANS_RES
		Get the transaction requirement of application
		
		Contact and receive information from servers
		
		If permanent disconnection or failure, abort the transaction
		
		If one-way, push information to users
		Otherwise send requests to multiple users and manage multiple responses
		
		Output transaction results

Protocol 2
(Dynamic
Adaptation for
Transaction
Support)
	Receive TRNS_RQRMNTS (Users, Delay, Critical users)
	Derive LIST_DESTN
	Send REQ_INFO to Server1, 2, …S
	Receive RES_INFO and Derive USER_INFO
	Derive NUM_STAGES, DELAY_STAGES, CRITICAL_USERS_STAGES, and NOM_TIMEOUT
	
	While (USER_FAILURE = NO)
	While CUR_STAGE <=NUM_STAGES
	Derive TIMEOUT[1-ReqUser] (NOM_TIMEOUT, NUM_STAGES, CUR_STAGE, COR_MOBILITY)
	For J= 1 to NUM_CRITICAL_USERS
	While (USER_INPUT=NO)
	If WAIT-TIME>=TIMEOUT[J]
	STATUS_TRANS_STAGE=TERMINATED
	Else Wait
	Process TRANS_STAGE
	Increment CUR_STAGE
	
	Output TRANS_RES
		Get the transaction requirement of M-commerce application
		
		Derive stages and nominal time-out information
		
		Derive/predict time-out values for all users the group
		
		Detect user response
		
		Wait until user responds or time-out expires
		
		Manage multiple inputs from “critical” users
		
		Output transaction results

Protocol 3
(Dynamic
Adaptation and
Status-based
Time-out for
Transaction
Support)
	Receive TRANS_REQRMNTS (Users, Delay, Critical users)
	Derive LIST_DESTN
	Send REQ_INFO to Server1, 2, …S
	Receive RES_INFO
	Derive USER_INFO
	Derive NUM_STAGES, DELAY_STAGES, CRITICAL_USERS_STAGES, NOM_TIMEOUT_STATUS
	
	If SOURCE_FAILURE = NO
	While (USER_FAILURE = NO)
	While CUR_STAGE <=NUM_STAGES
	Derive TIMEOUT[1-ReqUser] (NOM_TIMEOUT, NUM_STAGES, CUR_STAGE, COR_MOBILITY)
	For J= 1 to NUM_CRITICAL_USERS
	While (USER_INPUT=NO)
	If WAIT-TIME >=TIMEOUT_STATUS[J]
	STATUS_TRANS_STAGE=TERMINATED
	Else Wait
	Process TRANS_STAGE
	Increment CUR_STAGE
	Else
	If SOURCE_SEARCH (NEW_SOURCE) = TRUE
	Move TRANS_STATUS to NEW_SOURCE
	Else ABORT_TRANS
	
	Output TRANS_RES
		Get the transaction requirement of M-commerce application
		
		Derive/predict time-out values for source(s) and other users in the group
		
		Detect user response, permanent disconnection or failure
		
		If permanent disconnection or failure of critical user, start recovery process, Else wait until status-based time-out expires or user responds
		
		Manage multiple inputs from “critical” users
		
		Output transaction results

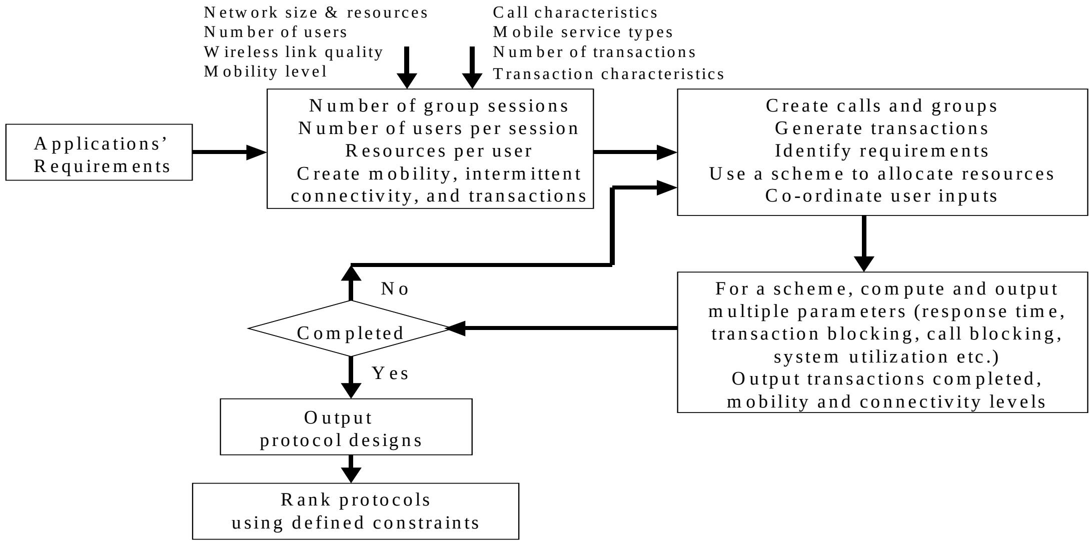

Fig. 6. Basic structure of simulation model

The simulation model is being enhanced to include support for both independent and co-related dis-connectivity periods and the use of information from other layers on coordinating user inputs for multiple stages. For performance evaluation purposes, we assumed that the transaction inter-arrival intervals and transaction durations are independently exponentially distributed. The minimum group size was three with single source and a user was involved in one transaction at a time (Table III).

TABLE III
SIMULATION PARAMETERS

Parameters	Range of values
Number of users	100
Transaction inter arrival interval	180 seconds
Transaction duration	30, 45, 60, 75, and 90 seconds
Number of users in a transaction	3, 4, 5, 6, 7, 8, or 9
Percentage of users required for a transaction	50, 60, 70, 80, 90, or 100
User connectivity/dis-connectivity ratio	2:1, 4:1, 6:1, 8:1, 9:1, 10:1, or 20:3

Effect of Transaction Duration*:* To measure the effect of transaction duration, the user connectivity/dis-connectivity ratio was 9:1, required user percentage at 90%, the average time-out values were 1 second for non-source users in Protocol 3 and 2 seconds for all in Protocol 2 and source for Protocol 3. The delay for all three protocols increases with an increased transaction duration as expected, with Protocol 1 having the highest transaction delay due to unlimited wait and Protocol 3 having the lowest delay due to limited and selective wait (Figure 7). We also found that the delay for aborted transactions is significantly lower than the delays for completed transactions for Protocols 2 and 3 and an increase in transaction duration does not seem to affect the difference between delays for aborted and completed transactions.

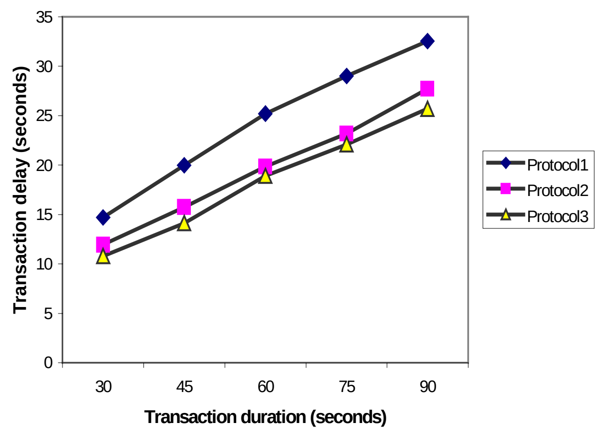

Fig. 7. Effect of transaction duration on average transaction delays

Effect of Varying Connectivity/Dis-connectivity Ratio: As the average user connectivity/dis-connectivity is changed to 9-1, the transaction delay reduces for all three protocols as expected (Figure 8). When the ratio becomes 18-2, the average delay reduces for Protocol 2 and Protocol 3 due to an increased transaction blocking. The lowest delays are for 19-1 due to the lowest average disconnection time for individual user. The delay increases when the ratio becomes 20-3. We also found that both delays and the difference among delays decrease with improvement in connectivity/dis-connectivity ratio, and, eventually the differences become insignificant. Therefore, for a high ratio, Protocol 1 is suitable due to low delay performance and inherent zero transaction blocking. As the average values of connectivity and dis-connectivity durations are varied with the same ratio, we found some very interesting results (Figure 9). The delay for Protocol 1 remains about the same, however, the delays for Protocol 2 and Protocol 3 decrease with increased transaction blocking. For example, the transaction blocking for Protocol 3 rises from about 20% to about 80%. Thus, the delays in Protocol 2 and Protocol 3 are governed by user connectivity/dis-connectivity ratio as well as the disconnection duration, while transaction blocking for both Protocol 2 and Protocol 3 is governed primarily by the average duration of disconnection (Table IV).

TABLE IV
EFFECT OF CONNECTIVITY/DIS-CONNECTIVITY ON PROTOCOLS

	Protocol 1	Protocol 2	Protocol 3
Transaction Delay	Connectivity/dis-connectivity (C/D) ratio	C/D Ratio Length of Disconnection	C/D Ratio Length of Disconnection
Transaction Blocking	N/A	Length of Disconnection	Length of Disconnection

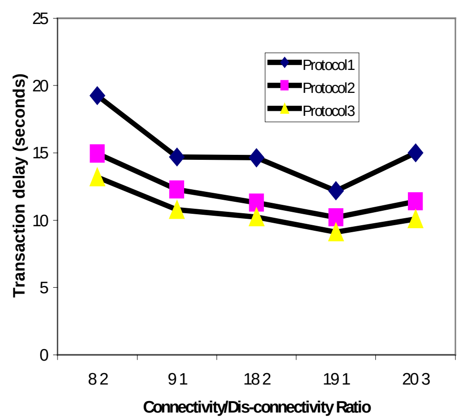

Fig. 8. Transaction delay under ratio of connectivity/dis-connectivity

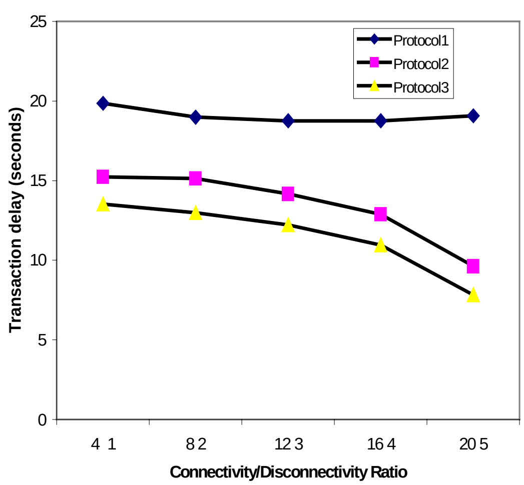

Fig. 9. Transaction delay under durations of connectivity and dis-connectivity

Effect of Required User Percentage: To observe the effect of required user percentage, the connectivity/dis-connectivity was 9-1. With the reduced number of users needed for transaction completion the average delay drops for all three protocols, as expected. Also, the difference among delays for protocols reduces and becomes insignificant (Figure 10). Thus, for a flexible application, Protocol 1 is ideal as it combines low delay performance with zero transaction blocking. The transaction blocking for Protocol 2 and Protocol 3 reaches to the same value with increased application flexibility (Fig. 11).

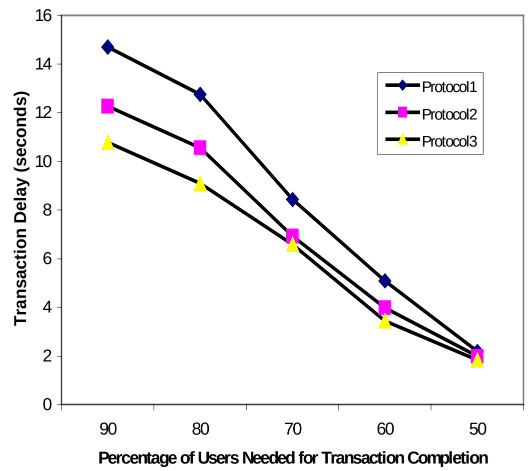

Fig. 10. Average delay for changed required user percentage.

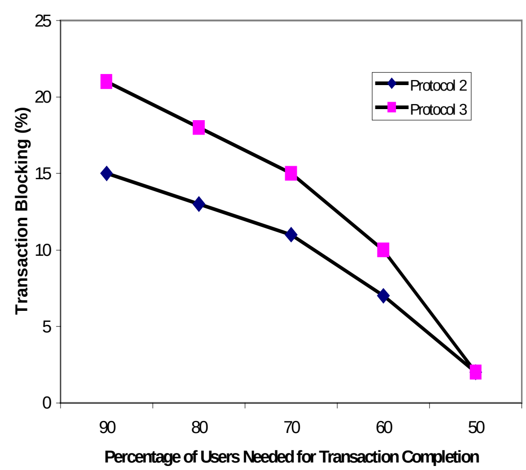

Fig. 11. Average transaction blocking for changed required user percentage.

Effect of Time Out: The transaction delays for Protocol 2 and Protocol 3 are significantly lower than protocol 1 (Figure 12). The transaction blocking and the delays for Protocol 2 and Protocol 3 are about the same due to similar time-out values. However, with an increased time-out, the transaction delay for Protocol 2 approaches to that of Protocol 1, rendering it less desirable than Protocol 1. The performance of Protocol 3 is better due to status-based time-out values. We found that transaction blocking for both the Protocol 2 and Protocol 3 reduces with Protocol 3 producing higher transaction blocking due to lower time-out for non-source users. Also, the delays for aborted transactions in Protocol 2 were actually higher than delays for completed transactions in Protocol 3 after time-out reached 2 seconds. This emphasizes that the optimal time out value is an important factor in the efficiency of the protocols.

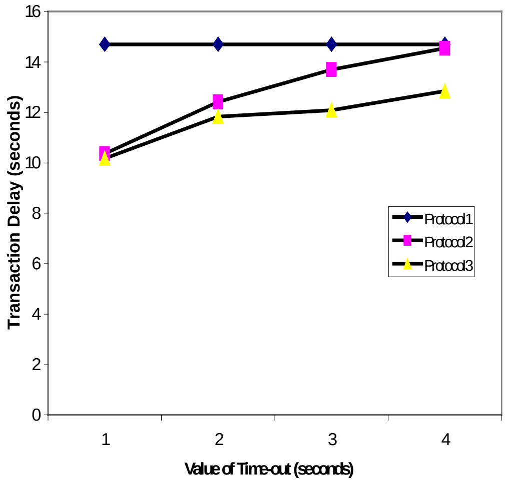

Fig. 12. Transaction delays for different time-out values

Combined Effect of Connectivity/Dis-connectivity Ratio and User Percentage: To achieve multi-parametric performance evaluation, we combined connectivity/dis-connectivity ratio with user percentage. In general, the transaction delay reduces with improved connectivity ratio at fixed required user percentage. However, as the required user percentage is increased, the delays do not fall that drastically (Figure 13) as the decrease in delay is partly offset by increased required user percentage. The transaction blocking increases at a slower pace as the negative influence due to higher required user percentage is somewhat offset by the positive influence due to better connectivity ratio (Figure 14).

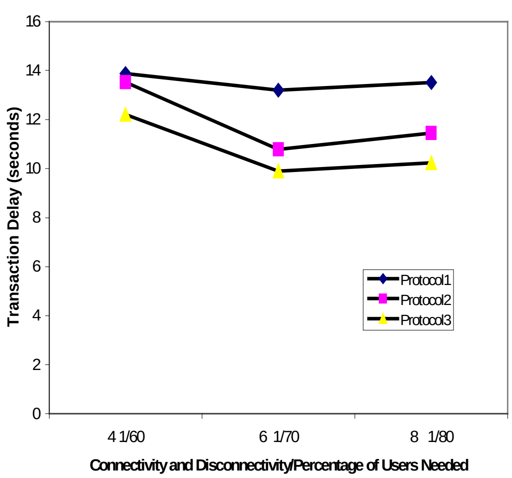

Fig. 13. Transaction delay for connectivity/dis-connectivity and user percentage

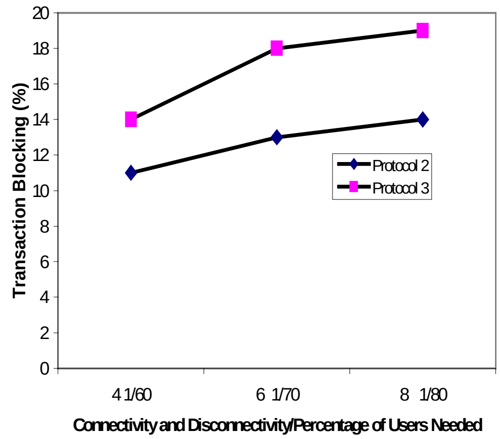

Fig. 14. Transaction blocking for connectivity/cis-connectivity and user percentage

From the results, the following important and interesting observations can be made. The delays for transactions and the difference among delays decrease with improvement in user connectivity/dis-connectivity ratio, and, eventually the differences in delays become insignificant. The transaction delays in Protocol 2 and Protocol 3 are governed by user connectivity/dis-connectivity ratio and average disconnection duration, while the transaction blocking for both Protocol 2 and Protocol 3 is governed primarily by the average duration of disconnection. With increased application flexibility, the average delay and the difference among delays for all three protocols reduce. Thus, for a highly flexible application, Protocol 1 is ideal as it combines low delay performance with zero transaction blocking. Also, the blocking performance of Protocol 2 and Protocol 3 is same for a highly flexible group-oriented application. With an increased time-out, the transaction delay for Protocol 2 approaches to that of Protocol 1, rendering it less desirable than Protocol 1. The delays for aborted transactions for Protocol 2 were actually higher than delays for completed transactions for Protocol 3 under some cases, emphasizing the need of carefully selecting time out value. For multi-parametric cases, increased connectivity problems can be offset by increased application flexibility in terms of both transaction delay and blocking.

V. CONCLUSIONS AND WORK IN PROGRESS

In this paper, we characterized transaction requirements of group-oriented m-commerce services, presented several protocols for transactions, and evaluated the effectiveness of protocols for diverse transactions using simulation modeling. The results show that high completion probability could be achieved by all three protocols under different conditions. It is also found that group applications with higher flexibility would be awarded with significantly higher transaction completion probability by all the three protocols. The flexibility of group-application can compensate for the impact of increased or more frequent disconnections. Protocol 1 can provide support for non delay-sensitive transactions, while Protocol 2 and Protocol 3 can support a range of delay sensitive transactions, and Protocol 2 is suitable for transactions requiring low delay and higher probability of transaction completion.

In addition to potential financial impact in billions of dollars, the design and implementation of protocols for group-oriented m-services services can form the basis for several innovative and multi-disciplinary research projects in m-commerce services. It can also lead to wide-scale deployment of group-oriented m-commerce services.

References

[1] U. Varshney, and R. Vetter, “Framework, Applications, and Networking Support for M-commerce”, ACM/Kluwer Journal on Mobile Networks and Applications (MONET), vol. 7, no. 3, June 2002, pp. 185-198.

[2] S. Deering, “Host Extension for IP Multicasting”, Internet RFC 1112, August 1989

[3] N. Nikaein, and C. Bonnet, “Wireless Multicasting in an IP Environment”, In Proc. 5th International Workshop on Mobile Multimedia Communication MoMuc ’98, Oct. 1998.

[4] G. Xylomenos, and G. Polyzos, “IP Multicast for Mobile Hosts”, IEEE Communications Magazine, January 1997.

[5] J. S. Baras, and I. Secka, “High Performance IP Multicasting over Wireless Satellite-Terrestrial Networks”, Research Report (CSHCN T.R. 97-32), Center for Satellite and Hybrid Communication Networks, University of Maryland, 1997.

[6] A. Acharya, and B. Badrinath, “A Framework for Delivering Multicast Messages in Networks with Mobile Hosts”, ACM/Baltzer Journal on Mobile Networks and Applications (MONET), October 1996.

[7] U. Varshney, “Multicast over Wireless Networks”, Communications of the ACM, Vol. 45, no. 12, Dec. 2002, pp. 31-37.

[8] E. Royer and C-K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks", IEEE Personal Communications, April 1999.

[9] J. Broch, D. Maltz, D. Johnson, Y-C. Hu, and J. Jetcheva, "A Performance Comparison of Multi-hop Wireless Ad Hoc Network Routing Protocols", In Proc. ACM Mobicom 1998.

[10] S-J. Lee, W. Su, J. Hsu, M. Gerla, and R. Bagrodia, “A Performance Comparison Study of Ad Hoc Wireless Multicast Protocols”, In Proc. IEEE INFOCOM, 2000.

[11] U. Varshney, “Mobile Payments”, IEEE Computer, vol. 35, no. 12, December 2002, pp. 120-121.

[12] E. Kirda, P. Fenkam, G. Reif, and H. Gall, “A Service Architecture for Mobile Teamwork” In Proc. 14th International Conference on Software Engineering and Knowledge Engineering, 2002, pp. 513-518.

[13] U. Walther, and S. Fischer, “Metropolitan Area Mobile Services to Support Virtual Groups”, IEEE Transactions on Mobile Computing, vol. 1, no. 2, Apr-Jun 2002, pp. 96-110.

[14] K.-Y. Lam, G. H. Li and T-W. Kuo, “A Multi-version Data Model for Executing Real-time Transactions in a Mobile Environment”, In Proc. 2nd ACM International Workshop on Data Engineering for Wireless and Mobile Access, pp. 90–97, 2001.

[15] V.C.S. Lee, L. Kwok-Wa, S. H. Son “Real-time Transaction Processing with Partial Validation at Mobile Clients”, In Proc. 7th Int. Conference on Real-Time Computing Systems and Applications, 2000, pp. 473-477.

[16] M. M. Gore and R. K. Ghosh, “Recovery of Mobile Transactions”, In Proc. 11th International Workshop on Database and Expert Systems Applications, London, Sept. 2000, pp. 23-27.

[17] A. M. Keller, O. Densmore, W. Huang, and B. Razavi, “Zippering: Managing Intermittent Connectivity in DIANA”, ACM/Kluwer Journal on Mobile Networks and Applications (MONET), vol. 2, no. 4 (January 1998), pp. 357-364.

[18] M. Samulowitz, F. Michahelles, and C. Linnhoff-Popien, “Adaptive Interaction for Enabling Pervasive Services”, In Proc. 2nd ACM International Workshop on Data Engineering for Wireless and Mobile Access, 2001, pp. 20-26.

[19] Y. Weinsberg and I. Ben-Shaul, “A Programming Model and System Support for Disconnected-Aware Applications on Resource-constrained Devices”, In Proc. 24th Int. Conference on Software engineering, 2002, pp. 374-384.

[20] L. Capra, G. S. Blair, C. Mascolo, W. Emmerich, and P. Grace, “Exploiting Reflection in Mobile Computing Middleware”, ACM Mobile Computing and Communications Review (MC2R), vol. 6, no. 4, Oct. 2002, pp. 34-44.

[21] S. K. Madria, and B. Bhargava, “A Transaction Model for Mobile Computing”, In Proc. International Database Engineering and Applications Symposium, July 1998 (IDEAS'98), pp. 92-102.

[22] C. Pedregal-Martin, and K. Ramamritham, “Support for Recovery in Mobile Systems”, IEEE Transactions on Computers, Vol. 51, no. 10, Oct 2002, pp. 1219-1224.

[23] P. Serrano-Alvarado, C. L. Roncancio, and M. Adiba, “Analyzing Mobile Transactions Support for DBMS”, In Proc. 12th International Workshop on Database and Expert Systems Applications, Sept. 2001, pp. 595-600.

[24] D. VanderMeer, A. Datta, K. Dutta, K. Ramamritham, and S. B. Navathe, “Mobile User Recovery in the Context of Internet Transactions”, IEEE Transactions on Mobile Computing, vol. 2, no. 2, April-June 2003, pp. 132-146.

[25] J. Veijalainen, V. Terziyan, H. Tirri, “Transaction Management for M-commerce at a Mobile Terminal”, In Proc. 36th Annual Hawaii International Conference on System Sciences (HICSS), Jan. 2003, pp. 89-98.

[26] J. Sairamesh, S. Goh, I. Stanoi, C. S. Li, and S. Padmanabhan, “Self-managing, Disconnected Processes and Mechanisms for Mobile E-business”, In Proc. 2nd Int. Workshop on Mobile Commerce, pp.82-89, 2002.

[27] U. Varshney, “Multicast Support for M-commerce Applications”, IEEE Computer, vol.35, no. 2, Feb. 2002, pp.115-117.

[28] U. Varshney "Location Management for Mobile Commerce Applications in Wireless Internet”, ACM Transactions on Internet Technologies, vol. 3, no. 3, August 2003, pp. 236-255.

[29] U. Varshney, “Patient Monitoring using Wireless Networks”, in Proc. of Americas Conference on Information Systems (AMCIS), August 2004.

[30] U. Varshney, “Performance Evaluation of Group-oriented Mobile Services”, to appear in ACM/Kluwer Journal on Mobile Networks and Applications (MONET).

[31] M. Shooman, Reliability of Computer Systems and Networks. New York, Wiley Interscience, 2002