

# An Adaptive Interest Management Scheme for Limited Available Multicast Groups in Virtual Environments

Tsung-Yi Lee and Jyh-Ming Huang

Department of Information Engineering and Computer Science  
Feng Chia University  
Taichung, Taiwan

*Abstract - Massively Multiplayer Games (MMGs) is an attractive example of virtual environment applications on the Internet. In such application, a great deal of update messages will be introduced, and thus consume a large amount of network bandwidth and multicast addresses. Interest Management (IM) is a mechanism provided for diminishing the redundant communication traffics. In this paper, based on the well-known cell-based grouping technique, we propose a flexible scheme, termed as consolidate-based grouping, to effectively decrease the superfluous message rate, and adaptively match the limitation on the number of available multicast addresses. In our approach, two phases are included. Initially, the virtual environment is first partitioned into much more smaller cells, than those were obtained from the traditional cell-based grouping strategy, to largely reduce the superfluous messages. In the second phase, we apply a merging technique to inversely combine some similar neighbouring cells to match the constraint on the number of available multicast addresses. Simulation results show that, in most cases, our proposed grouping technique can conduct about 10% reductions on the superfluous message rate, compared to the conventional cell-based grouping strategy, at the same condition.*

**Keywords:** cell-based grouping, entity-based grouping, interest management, consolidate-based grouping, massively multiplayer games

## 1 Introduction

Massively Multiplayer Games (MMGs) is an attractive example of virtual environment applications on the Internet [3][4][8][10][11]. In MMGs, even hundreds or thousands of participants play and communicate with each other in the virtual world over a network. Three architectures, such as client/server based, peer-to-peer based, and hybrid client/server based, have been deployed for MMGs. In client/server based MMGs, participants (players) always connect to a central (gaming) server and exchange their update information via this server. In other words, specifically, every participant sends his update message to a gaming server for each game round, and the server tackles all effects and then immediately forwards these messages to relevant clients. As a large number of participants are usually invited into a MMG, and each participant is not interested in all update information, it seems inefficient if the server broadcasts all update messages to irrelevant participants and thus incurs a great

deal of superfluous messages and network bandwidth consumption [6]. Interest management (IM) is a mechanism provided for ensuring players to receive the messages they want, and thus can significantly reduce the system communication overheads [13][14][15].

Two well-known grouping strategies, entity-based grouping and cell-based grouping, have been proposed for interest management mechanism [9][15]. In entity-based grouping strategy, each entity (e.g. player) declares its location and area of interest (AOI), and maintains a multicast group. The group contains all entities those are currently locating within its AOI. In message exchange phase, every entity will receive the update messages from group entities that have changed their status. The merit of entity-based grouping strategy mainly comes from the fact that no superfluous messages will be induced. However, due to its computation complexity ( $O(n^2)$ ) [9][13], the entity-based grouping strategy is not very suitable for large-scale virtual environments (VE) [13][15]. In contrast, the cell-based grouping strategy usually divides the whole VE into many cells. Cells can be any sizes and shapes. Each cell is also associated with a multicast group. While an entity joins into the VE, it publishes its location at a cell, and subscribes all cells that are currently covered or intersected by its AOI. As an entity delivers its update message, all entities that subscribed its publishing cell will receive this message. However, in certain cases, some entities may just locate outside its current AOI, and are not interested in such information. The superfluous message transmissions will cause heavy communication traffics and thus degrade network performance. In order to reduce the superfluous message rate, a smaller cell size must be considered [1][9][13]. Unfortunately, the reduction on the cell size will result in more multicast group addresses to be needed [1][13]. Since the current multicast (group) addresses are not inexhaustible resources, many researchers have continuously investigated on the trade-off issues between the cell size and the number of available multicast addresses [2][5][6][7].

Motivated by solving the aforementioned shortcomings characterized by both grouping scheme, in this paper, based on the traditional cell-based grouping strategy, we propose a flexible and compromising technique, termed as consolidate-based grouping, to effectively reduce the superfluous message rate, and adaptively match the desired limitation on the number of available multicast group addresses. The basic idea

behind our proposed scheme is first dividing the VE into a number of smaller cells (than normal situation) to largely reduce superfluous messages, and then applying a merging technique to inversely integrate some similar neighbouring cells to dramatically diminish the number of multicast addresses. Although such partition in the first phase will temporarily increase the number of multicast addresses needed, as will be seen in simulation results later, with an appropriate merging process, our grouping technique can not only adaptively fit the limitation on the number of multicast group addresses, but also significantly reduce the superfluous message rate about of 10%.

The rest of this paper is organized as follows: in the next section, we first briefly review two previous grouping strategies, entity-based and cell-based grouping strategies. And then, we describe the detail implementations for our proposed consolidate-based grouping technique in Section 3. In Section 4, we illustrate our simulation results. Conclusion remarks are finally drawn in Section 5.

## 2 Previous Grouping Strategies

Before describing our proposed grouping technique, it should be helpful to reviewing some relevant backgrounds first. In a VE, we generally use an entity (or avatar) to represent a participating client (player). So a player can communicate with others via his entity. As entities can dynamically move around everywhere in a VE, their update information should be appropriately and timely delivered to all other entities for keeping the VE consistent. As a consequence, this scenario will incur a large amount of communication overheads. In order to avoid the redundant communication traffics, and, an entity is often only interested in the update messages induced from its vicinity, the entity always declares its area of interest (AOI). While an entity changed its location, it publishes its new position to a gaming server, and then the server manager uses a specific strategy to decide which entities should be notified with this update information. Ideally, the players are only expected to receive their interesting and useful messages. As a consequence, an interest management (IM) mechanism based on a certain grouping strategy is always provided to filter the unwanted messages. Two commonly used grouping strategies, entity-based and cell-based, were proposed. Fig. 1 illustrates a typical virtual environment. Even an AOI can be modelled as any shapes (e.g. circle or rectangle), in this paper, we assume it is a circle.

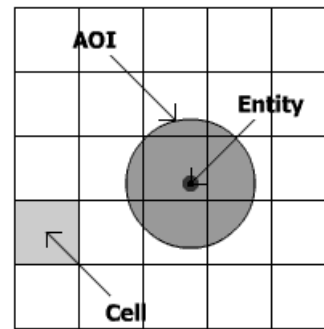


Fig. 1 A typical virtual environment

### 2.1 Entity-Based Grouping

The basic idea of entity-based grouping strategy is based on distance calculations and comparisons. In entity-based grouping strategy, an entity always maintains a group. For each game turn, every entity calculates the individual distances from all other entities, and checks those values with its AOI to determine whether the opposite entities should be included into its group or not. While an entity in the group changes its status, it will receive the update message from that entity member.

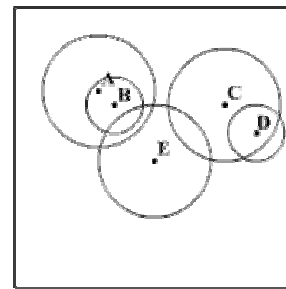


Fig. 2 Entity-based grouping strategy

Fig. 2 shows a simple example. Since the group of entity A contains the member of entity B (because entity B is locating within entity A's AOI), and vice versa, entities A and B therefore send messages to each other, if their statuses are altered. However, for entity C and entity D pair, as entity D is locating within entity C's AOI, only entity C can receive the update message from entity D. As for the entity E, since no other entities are included in its AOI, it does not receive any update message.

The main contribution of entity-based grouping strategy results from the absence of superfluous messages. So thus it can eliminate the redundant network traffics. However, due to its high computation complexity ( $O(n^2)$ ), the increment in participants will introduce heavy computation overheads. Consequently, various cell-based grouping techniques were proposed for solving this shortcoming [1][2][5][6][9][12]. In the following context, we use the terms cell and group interchangeably.

### 2.2 Cell-Based Grouping

The cell-based grouping strategies always perform a cell partition task. The gaming server first divides the VE

into several cells (as shown in Fig. 1) with the same or different sizes. For simplicity, the cell size is always fixed. We denote a cell as  $C_{ij}$  in this paper. Many factors, such as the number of available multicast groups, the receiver capabilities, and the distribution of entities, will significantly affect the decision of cell size [6]. Generally, the number of partitioned cells is confined by the number of available multicast group addresses. In cell-based grouping scheme, each cell is also associated with a multicast group. While an entity enters into the VE, it publishes itself at a cell and subscribes all cells (i.e. groups) that are partially intersected with or fully covered by its AOI. When an entity changes its status, only the entities subscribed its publishing cell will receive the update message. For example, as shown in Fig. 3, entity A publishes at cell  $C_{22}$  and subscribes cells  $C_{11}$ ,  $C_{12}$ ,  $C_{13}$ ,  $C_{21}$ ,  $C_{22}$ ,  $C_{23}$ ,  $C_{31}$ ,  $C_{32}$  and  $C_{33}$ . Entity C publishes at cell  $C_{24}$  and subscribes cells  $C_{14}$ ,  $C_{15}$ ,  $C_{23}$ ,  $C_{24}$ ,  $C_{25}$ ,  $C_{33}$ ,  $C_{34}$ , and  $C_{35}$ . If entity A sends its update message out, both entities B and E will receive that message. Similarly, entity E will receive the update messages from entities A, B and C. Entity D can also receive message from entity C. However, unfortunately, because the entities A, B and C are not within E's AOI, the update messages issued from them to entity E will be redundant, and thus result in network bandwidth consumption. So it is an important issue to eliminating the superfluous messages. The main disadvantage of cell-based grouping strategy comes from the fact that a huge amount of superfluous messages may be introduced. In contrast, its merits are characterized by the low computation overhead and potential scalability.

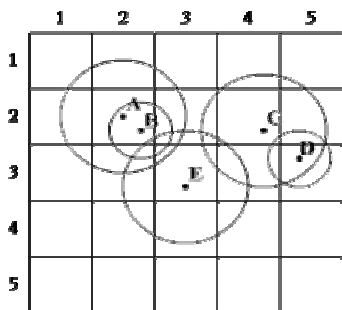


Fig. 3 Cell-based grouping strategy

Generally speaking, in cell-based grouping strategy, the larger the cell size becomes, the more superfluous messages will be induced [1][6][9][13]. In [6], the authors have shown that when the cell size is greater than the diameter of entity's AOI, the superfluous message rate would be over 70%. On the contrary, the smaller the cell size is, the more multicast group addresses are necessary. As a consequence, how to decide an appropriate cell size to minimizing the superfluous message rate becomes a challenge. In the following section, we propose a compromising approach, named as consolidate-based grouping technique, to effectively and adaptively reduce the superfluous messages while operates on the limited number of multicast group addresses.

### 3 Consolidate-Based Grouping Technique

As the mentioned before, for a MMG implemented with a cell-based grouping strategy, the system communication overhead is closely dependent on the cell size, and the multicast addresses are not inexhaustible resources. Therefore, the number of available multicast groups must be taken into account during design procedure. Motivated by these reasons, in this section we propose a flexible scheme, termed as consolidate-based grouping technique, to adaptively solve these problems. In our approach, two processing phases, partition phase and merging phase, are necessary. In the first step, we divide the VE into much more cells, with a cell size smaller than that was determined by a conventional cell-based grouping strategy, to greatly reduce the superfluous messages. Even this step will temporarily result in a large amount of multicast groups being needed, the subsequent merging phase, as will be stated in the following paragraphs, can dramatically diminish the needed number of multicast group addresses so as to meet the limitation.

#### 3.1 Merging Phase

In cell-based grouping strategy, since a cell can be simultaneously subscribed by several entities, so different groups might have the similar or the same entity sets. If we can merge those similar groups into a larger one, we are expected that the number of available multicast group addresses can be dramatically reduced. As a result, in the second phase, we apply this concept to iteratively merge similar neighbouring cells, between which their diversity metric is less than or equal to a predetermined merging threshold, to decrease the needed number of multicast addresses. Neighbouring cell means a cell that is closely located on the up, left, right, or down side of a specific cell. For instance, the neighbouring cells of cell  $C_{22}$  in Fig. 4 are cells of  $C_{12}$ ,  $C_{21}$ ,  $C_{23}$ , and  $C_{32}$ . The diversity metric will be defined and calculated as below.

$C_{11}$	$C_{12}$	$C_{13}$
$C_{21}$	$C_{22}$	$C_{23}$
$C_{31}$	$C_{32}$	$C_{33}$

Fig. 4 Neighbouring cells of cell  $C_{22}$  (in shaded)

Assume that  $C_1$  and  $C_2$  are two neighbouring cells and associated with entity set  $S_1$  and  $S_2$ , respectively. We define the diversity metric (DM) between cells  $C_1$  and  $C_2$  as follows:

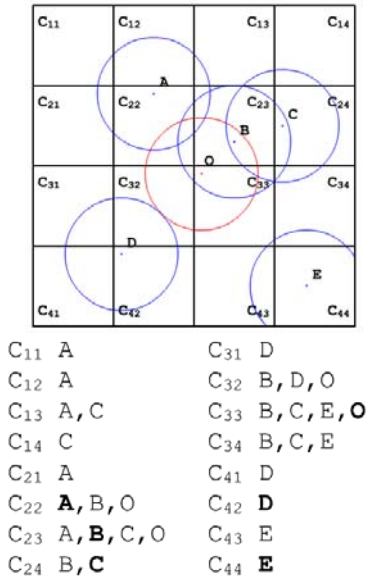
$$DM = \begin{cases} 1 - \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}, & \text{if } |S_1 \cap S_2| \neq \phi \\ 0 & \text{if } |S_1 \cap S_2| = \phi \end{cases}$$

Where  $|S_i|$  represents the cardinality of entity set  $S_i$ .

For example, given two cells  $C_1$  and  $C_2$  with their respective entity set  $S_1 = \{A, B, C, D, E\}$  and  $S_2 = \{A, B, C, D, F, G\}$ , the diversity metric between them can be computed as  $DM = 1 - (4/7) = 0.43$ .

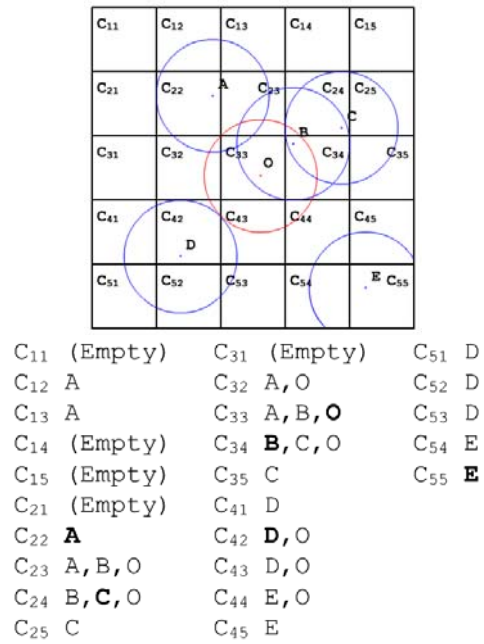
In merging phase, we iteratively integrate those neighbouring cells with a lower diversity metric, compared to a predetermined merging threshold, into a larger group. This process should be continuously performed until all cells are checked and no cells can be further merged with each other. Finally, the desired multicast group limitation can be met. As will be seen in the simulation results, our proposed grouping technique can adaptively decrease the number of needed multicast groups, if an appropriate merging threshold is selected in advance.

For realizing the detail consolidation process, we take the following examples to explain and suppose the number of available multicast addresses (groups) is 16. Fig. 5 shows an original VE that was divided by the conventional cell-based grouping strategy. The names of entities that subscribe a specific cell are also listed on the right-hand side. The letter with bolder type indicates the name of entity that is publishing at a cell. For instance, entity A is publishing at cell  $C_{22}$ , and entities B and O subscribe the same cell. In this scenario, if each entity is assumed to send one update message, the superfluous message rate can be computed as  $0.5556 (=5/9)$ .



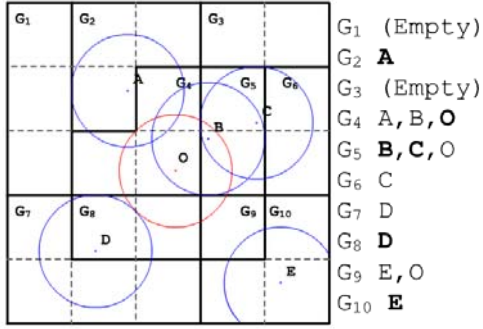
**Fig. 5 A VE partitioned with conventional cell-based grouping technique**

However, if we apply our proposed technique to the same situation as Fig. 5, we can further divide the original VE into much more smaller cells (say 25 cells, see Fig. 6). Even this partition process will temporarily introduce an increment on the number of multicast groups (i.e. 25 groups), the superfluous message rate has been greatly dropped to a value of  $0.2857 (= 2/7)$ .



**Fig. 6 A VE after performing the partition task with our proposed technique**

Subsequently, in the second phase, we begin a recursive merging process. We assume the merging threshold is  $1/3$ , and start this process from cell  $C_{11}$  in a cell-by-cell manner and an index-increasing (row-major) order. For each cell, we check its neighbouring cells, in an up, left, right, and down side sequence, to decide whether the two neighbouring cells should be merged together or not. If the computed diversity metric (DM) among them is less than or equal to the merging threshold, a consolidation task shall be performed. As the merging process cannot be continued, we return to the initial merging cell and repeat the same procedure to other neighbouring cells. For example, cell  $C_{11}$  cannot be merged with cell  $C_{12}$  due to their  $DM=1$ . However, it can be combined with cell  $C_{21}$  since their  $DM=0$ . After merging the cells, we take the newly merged cell (e.g.  $C_{21}$ ) as the initial merging cell for further consolidation processing. As a consequence, the cell  $C_{31}$  is eventually merged to form the group  $G_1$ , which is illustrated in Fig. 7. For another example and referring to Figs. 6 and 7, begin with the cell  $C_{12}$ , we can formulate the group  $G_2$  in conjunction with cells  $C_{13}$  and  $C_{22}$ , because of  $DM = 0$ . The merging process is terminated until all cells are carefully checked and no cells can be further integrated. Finally, the desired number of multicast groups could be derived. Fig. 7 shows out all final groups which are formulated from a VE presented in Fig. 6. The bolder lines indicate the group boundaries. It is worth noting that, with our proposed grouping technique and an appropriate threshold, the number of multicast groups is indeed reduced to 10, and the superfluous message rate is still with the value of  $0.2857 (2/7)$ .



**Fig. 7 A final VE after applying our proposed grouping technique**

### 3.2 Merging Threshold

In our proposed grouping scheme, the merging threshold plays a key point to the second phase. Generally, the larger threshold value is, the less multicast groups appear. A large merging threshold may cause the number of multicast groups exceeding the given limitation. Many factors, such as the density of entities, cell size, AOI, and the number of available multicast groups, will significantly affect the decision of merging threshold. Since a MMG is a real-time application in general, it seems inefficient if the merging threshold is calculated during each merging phase. As a consequence, in this paper, according to the past gaming experiences and the parameters just mentioned, we create a merging threshold table in advance for merging reference.

## 4 Simulations and Results

To evaluate the effectiveness of our proposed grouping technique, we conduct several simulations under two different scenarios: (1) a 100 \* 100 units VE with 100 entities, (2) a 200 \* 200 units VE with 400 entities. All entities are generated randomly. In addition, for simplicity, we assume all entities have the same AOI size and with a diameter of 20 units. For each scenario, we execute 10 runs and take the average superfluous message rate as our metric.

Table 1 shows the average superfluous message rate for different cell sizes in scenario (1). All cases are conducted with conventional cell-based grouping technique. For comparisons, we also perform simulations on the same situations, to which our proposed grouping technique is applied. Table 2 illustrates the results for different cell sizes and merging thresholds. As we can see from Table 1 and Table 2, in scenario (1), if the number of available multicast addresses is limited to 49, we must partition the VE into 49 cells (cell size equals to 16 units) with conventional cell-based grouping strategy to meet the restriction, and thus derive the average superfluous message rate 0.74. While, if we apply our proposed grouping technique to, as Table 2 shown, we can further divide the VE into much more smaller cells first, with different cell sizes, such as 10, 12, and 14 units first. And then, we can use the distinct thresholds 0.4547, 0.45, and 0.4129, respectively, to merge similar cells into 49 cells.

Surprisingly, the average superfluous message rate can be achieved to the values of 0.68148, 0.71789, and 0.73558, respectively. The improved superfluous message rates in Table 2 are marked with a shaded background. Similarly, the same explanations can also be migrated to those corresponding cases in scenario (2) (the relevant results are demonstrated in Tables 3 and 4). For other scenarios, we have similar conclusions. Due to the space limitation, we omit here.

**Table 1. Average superfluous message rate for scenario (1) (with conventional cell-based grouping strategy)**

Cell Size	Number of Cells (groups)	Superfluous rate
10	100	0.59844
12	81	0.66537
14	64	0.71054
16	49	0.74000
18	36	0.76615
20	25	0.76899

**Table 2. Average superfluous message rate for scenario (1) with our grouping technique**

Cell Size	Number of Cells (after merged)	Threshold	Superfluous rate
10	81	0.2754	0.61488
	64	0.3779	0.64471
	49	0.4547	0.68148
	36	0.5364	0.72432
	25	0.6424	0.78513
12	64	0.3444	0.68493
	49	0.4500	0.71789
	36	0.5454	0.75785
	25	0.6495	0.81007
14	49	0.4129	0.73558
	36	0.5097	0.77195
	25	0.6310	0.82116
16	36	0.4604	0.77120
	25	0.5857	0.81389
18	25	0.5075	0.80111

**Table 3. Average superfluous message rate for scenario (2) (with conventional cell-based grouping strategy)**

Cell Size	Number of Cells (groups)	Superfluous rate
10	400	0.60324
12	289	0.65893
14	225	0.70498
16	169	0.73847
18	144	0.76750
20	100	0.77988

**Table 4. Average superfluous message rate for scenario (2) with our grouping technique**

Cell Size	Number of Cells (after merged)	Threshold	Superfluous rate
10	289	0.3592	0.63855
	225	0.4417	0.67523
	169	0.5172	0.71688
	144	0.5643	0.74322
	100	0.6537	0.79787
12	225	0.3820	0.68526
	169	0.4751	0.72409
	144	0.5211	0.74553
	100	0.6255	0.79681
14	169	0.4361	0.73595
	144	0.4895	0.75575
	100	0.5975	0.80162
16	144	0.4261	0.75526
	100	0.5610	0.79966
18	100	0.5514	0.80849

As the simulation results illustrated, we believe that our proposed consolidate-based grouping technique is valuable for improving superfluous message rate in virtual environment applications. In addition, it should be noted that even though we have many ways to achieve the desired number of cells (groups), it seems a good choice if we directly divide the VE into cells with AOI-radius-match size, and then apply our grouping technique to merge them.

In a short summary, theoretically, the smaller the cell size is, the less superfluous messages introduced, and the

more multicast groups needed. Under the constraint on the available multicast groups, instead of using traditional cell-based grouping strategy, our proposed consolidate-based grouping technique can not only match the grouping requirements, but also achieve the potential improvement on the superfluous message rate.

## 5 Conclusion

Massively Multiplayer Game (MMG) is an interesting application scenario in virtual environments. In a client/server based MMG, the effect of network communication traffics on system performance has to be carefully studied. Cell-based grouping strategy is a commonly used approach in interest management mechanism to decrease the redundant communication overheads. In this paper, we propose a compromising cell-based merging technique, termed as consolidate-based grouping, to effectively and adaptively diminish the superfluous messages while matching the limitation of available multicast group addresses. In our technique, we first divide the virtual environment into much more smaller cells, than those were obtained from the traditional cell-based grouping strategy, to largely reduce the superfluous messages, and then apply an appropriate predetermined threshold to recursively merge similar neighbouring cells together so as to coincide with the limited number of available multicast groups. Simulation results show that, compared to the conventional cell-based grouping strategy, in most cases, our proposed grouping technique can not only reduce about 10% superfluous message rate, but also fit the multicast group restriction.

## 6 References

- [1] Rassul Ayani, Farshad Moradi and Gray Tan, "Optimizing Cell-size in Grid-Based DDM," *proceedings of the Fourteenth Workshop on Parallel and Distributed Simulation, 2000. PADS 2000*, pp. 93-100.
- [2] Tianying Chang, Popescu, G. and Codella. C, "Scalable and Efficient Update Dissemination for Distributed Interactive Applications," *proceedings of the 22nd International Conference on Distributed Computing Systems*, July 2002, pp.143-150.
- [3] Nathaniel E. Baughman and Brian Neil Levine, "Cheat-Proof Payout for Centralized and Distributed Online Games," *proceedings of INFOCOM 2001, the Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, vol.1, pp. 104-113.
- [4] L. Gautier and C. Diot, "Design and Evaluation of MiMaze, a Multi-Player Game on the Internet," *proceedings of International Conference on Multimedia Computing and Systems*, 1998, pp. 233-236.
- [5] E. Lety, T. Turletti and F. Baccelli, "Cell-based

Multicast Grouping in Large-Scale Virtual Environments,” *Tech. rep., INRIA, France*, July 1999.

- [6] E. Lety, T. Turetletti and F. Baccelli, “SCORE: A Scalable Communication Protocol for Large-Scale Virtual Environment,” *IEEE/ACM Transactions on Networking*, April 2004, pp. 247-260.
- [7] B. N. Levine, J. Crowcroft, C. Diot, J.J. Garcia-Luna-Aceves and James F. Kurose, “Consideration of Receiver Interest for IP Multicast Delivery,” *proceedings of INFOCOM 2000, the Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, vol.2, pp. 470-479.
- [8] Yow-Jian Lin, Katherine Guo and Sanjoy Paul, “Sync-MS: Synchronized Message Service for Real-Time Multi-Player Distributed Games,” *proceedings of 10th IEEE International Conference on Network Protocols, 2002*. pp. 155-164.
- [9] Chia-Hao Liu, Chen-Hsing Wen, Hsing-Lung Chen, “Tracking-needless grouping: An Efficient and Scalable Grouping Schema in Networked Virtual Environments,” *proceedings of the First IEEE Consumer Communications and Networking Conference, 2004. CCNC 2004*, pp. 477-482.
- [10] Lothar Pantel and Lars C. Wolf, “On the Suitability of Dead Reckoning Schemes for Games,” *First Workshop on Network and System Support for Games, April 2002*, pp.79-84.
- [11] V. Ramakrishna, Max Robinson, Kevin Eustice and Peter Reiher, “An Active Self-Optimizing Multiplayer Gaming Architecture,” *Autonomic Computing Workshop, 2003*, pp. 32-41.
- [12] El Saddik, A and Dufour, A, “Peer-to-Peer Suitability for Collaborative Multiplayer Games,” *the Seventh IEEE International Symposium on Distributed Simulation and Real-Time Applications, 2003*, pp.101-107.
- [13] Gray Tan, Rassul Ayani, YuSong Zhang and Farshad Moradi, “Grid-Based Data Management in Distributed Simulation,” *proceedings of the 33rd Annual Simulation Symposium, 2000. SS 2000*, pp. 7-13.
- [14] Lihua Wang, Stephen John Turner and Fang Wang, “Interest Management in Agent-based Distributed Simulations,” *proceedings of the Seventh IEEE International Symposium on Distributed Simulation and Real-Time Applications, 2003*, pp. 20-27.
- [15] Li Zou, Mostafa H. Ammar and C. Diot, “An Evaluation of Grouping Techniques for State Dissemination in Networked Multi-User Games,” *proceedings of the Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2001*, pp.33-40.