

# An Evaluation of Grouping Techniques for State Dissemination in Networked Multi-User Games \*

Li Zou , Mostafa H. Ammar

College of Computing, Georgia Institute of Technology  
Atlanta, GA 30332  
{zou,ammar}@cc.gatech.edu

Christophe Diot

Sprint ATL  
1 Adrian Court, Burlingame, CA 94010  
cdiot@sprintlabs.com

## Abstract

*In a distributed multi-user game, entities need to communicate their state information to other entities. Usually only a subset of the game's entities are interested in information being disseminated by any particular entity. In a large scale distributed game, broadcasting messages containing each information to all participants and applying a relevance filter at the end host is wasteful in both network and processing resources. We consider techniques that address this problem by dividing the entities into groups and using multicast communication to disseminate information to the groups which would be interested in such information. We investigate two grouping strategies: cell-based grouping and entity-based grouping. Our goal is to understand the tradeoffs between grouping overhead and communication overhead and compare the cost of both strategies under various conditions.*

## 1. Introduction

In a distributed multi-user game [1, 2], several players move and interact in a virtual space, exchanging information amongst themselves over a network. In these systems, entities (e.g., avatars, tanks, monsters) need to communicate information about their states (e.g., change of color, position or shape) to other entities. One approach is to have each entity broadcast this information to all other entities. Over the network this would require the broadcast of a message from the machine representing the sending entity to the machines representing all the other entities. In most situations, however, only a subset of a game's entities need to receive such messages. When information is broadcast, entities usually will apply a filter on all received messages and

process only the ones that they need to be receiving. For example, a message indicating that a tank has moved its gun turret need only be sent to entities that can actually see the tank in the game. If such a message is broadcast, receiving entities will need to apply a position filter and discard the message if they cannot actually see the tank.

In a large-scale game with a large number of participating machines interconnected over a wide-area network (the Internet or an Intranet), this broadcast and filter approach can be quite wasteful. Network resources would be wasted broadcasting messages and processing resources would be wasted filtering out irrelevant data. In this paper we consider this problem and investigate the approach of dividing the entities into groups and targeting entity's information dissemination (using multicast communication) to specific groups that would be interested in the information. An important question is how one may determine such groups and what effect does the grouping strategy have on the overhead incurred in communication among entities in distributed gaming systems. Of particular interest to us is the trade-off between the overhead of group formation (especially since entities in such games can be dynamic thus requiring a dynamic grouping strategy) and the overhead of filtering unnecessary data by recipients. Clearly the broadcast approach described above is one extreme of this tradeoff: there is no group formation overhead since all entities always belong to the same group, while the filtering overhead is the highest.

We assume Internet-style multicast communication [3]. Groups are formed by receivers joining and leaving multicast groups asynchronously. Multicast packets are addressed to a group address. Receivers wishing to join the multicast group simply "listen" to the group address and inform a local designated router of this via the IGMP protocol [4]. A multicast routing protocol (such as DVMRP [5] or PIM [6]) operating within the network ensures the delivery of multicast packets to receivers. This typically requires the exchange of mes-

---

\*This work is supported by the AFOSR MURI Grant F49620-00-1-0327 and NSF Grant ANI-9973115

sages among routers within the network in order to establish (graft) or teardown (prune) multicast data delivery paths as needed. The group formation overhead we consider in our work is designed to account for the messaging required for multicast routing.

In this paper, we first describe two basic grouping strategies: *cell-based* and *entity-based* grouping. We then develop a model for a system which forms the basis of our evaluation of these grouping strategies. Our goal is to understand the various tradeoffs in the design of a grouping strategy and investigate the relative merits of the cell-based and entity-based schemes and how they compare to the simple broadcast approach.

We are not aware of a study that considers comparing the two strategies as ours. An approach similar to our cell-based grouping strategy has been studied by Van Hook et al. [7, 8]. They evaluated the approach using log files of DIS-based simulations [9, 10]. Lety et al [11] analytically studied the property of cell-based grouping using planar point process. They also proposed a dynamic cell-based protocol. Wong et al [12] presented a grouping algorithm based on the preference clustering. The entity-based protocol is an extreme case of the preference clustering approach where the number of clusters equals to the number of entities. Other efforts that try to avoid broadcasting and filtering all the data at end-host include interest-based filtering [13] and *Locales and Beacons* [14]. The former approach uses a network-layer protocol [15] to address data to the interested hosts. The latter divides the virtual world into locales which can be communicated independently and use beacons to find the locale in which an object resides. Our work is complementary to the work in [13] as we focus on the control cost of group formation, not just the data messages sent.

The rest of the paper is organized as follows: In section 2 we describe the cell-based grouping and entity-based grouping concepts and some analytical results. Then in section 3, we examine static models for the both strategies. This is followed in section 4 with discussions of simulation results of the static models. In section 5 we present a dynamic model designed to capture the effect of entity speed. Simulation results of the dynamic model are reported in section 6. The paper is concluded in section 7.

## 2. Grouping Strategies

We assume a two-dimensional “playing area”. Entities move within this area with the position of each entity defined by two coordinates that are known to that entity. Each entity has a *vision domain* which describes the set of locations within the playing area that this entity can see. An entity is only interested in receiving data from

other entities within its vision domain.

The playing area is divided into cells. Cells can be of any shape, however, in this paper, we only consider square cells. We assume that an entity knows the boundaries of its own vision domain and that it can determine whether a particular position within the playing area is within this domain. We also assume that an entity has enough information about the cell structure to be able to determine the set of cells that intersect with its own vision domain. Typically, the total area of these cells will be larger than the area represented by the vision domain. Figure 1 illustrates these basic concepts.

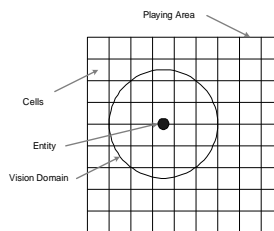


Figure 1. Entities, Cells and Vision Domains

Within this basic framework, there are two possible strategies for grouping entities for the purpose of communication among entities: *cell-based grouping* and *entity-based grouping*.

**Cell-Based Grouping** In the cell-based grouping approach, each cell is assigned a multicast group address. There are two sets of cells associated with each entity, a “sending” set and a “receiving” set. The definition of the sets are global and known to all entities. An entity wishing to communicate information to other entities first needs to compute the two sets locally according to the definition of the sets and the cell structure of the playing area. The entity then joins all the multicast groups associated with the cells in the “receiving” set and multicasts its information to the group addresses corresponding to the cells in the “sending” set. We assume that all information sent by an entity includes the entity’s own coordinates. When an entity receives a message on one of the multicast groups it has joined, it needs to perform a filtering function to determine whether the sender is actually in its vision domain. This is because some of the messages might come from the entities outside of its vision domain. However, the amount of the superfluous messages is less than that in the broadcast approach. As the entity moves and/or its vision domain changes shape it recomputes the two sets and may leave some multicast groups and join others.

The sending and receiving set construction algorithm must guarantee that each entity receive all the interesting information from others, provided the network is lossless. That is, if entity *A* can “see” entity *B*, then *A*’s

receiving set must intersect with  $B$ 's sending set. There are many possible definitions of the sets. We have analytically studied the properties of several type of sending and receiving sets in our other work. In this paper, we only consider the "traditional" definition of the sets where the "sending" set contains only the "home" cell in which the entity currently resides and the "receiving" set contains all the cells intersecting with the entity's vision domain. In this paper, for the simplicity, we only study the case with square cells and circle vision domains. We also assume the playing area is a square, which is divided into  $N \times N$  cells. We denote the multicast address assigned to cell  $(i, j)$  by  $G_{i,j}$ .  $i, j = 1, \dots, N$ . Figure 2 and Table 1 illustrate this cell-based grouping idea.

Entity	Join Group	Send Group
$X$	$G_{2,2}, G_{3,2}, G_{4,2}, G_{5,2}, G_{1,3}, G_{2,3}, G_{3,3}, G_{4,3}, G_{5,3}, G_{1,4}, G_{2,4}, G_{3,4}, G_{4,4}, G_{5,4}, G_{1,5}, G_{2,5}, G_{3,5}, G_{4,5}, G_{5,5}, G_{2,6}, G_{3,6}, G_{4,6}$	$G_{3,4}$
$Y$	$G_{4,4}, G_{5,4}, G_{6,4}, G_{4,5}, G_{5,5}, G_{6,5}, G_{4,6}, G_{5,6}, G_{6,6}, G_{4,7}, G_{5,7}, G_{6,7}$	$G_{5,6}$
$Z$	$G_{3,1}, G_{4,1}, G_{5,1}, G_{3,2}, G_{4,2}, G_{5,2}, G_{3,3}, G_{4,3}, G_{5,3}, G_{3,4}, G_{4,4}, G_{5,4}, G_{4,3}$	$G_{4,3}$

Table 1. Cell-Based Grouping Example

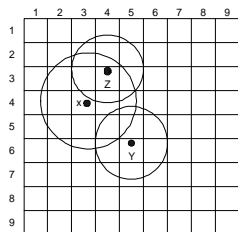


Figure 2. Cell-Based Grouping Example

Note that this cell-based approach degenerates to the broadcast approach when the playing area is only one cell. Note also that as the number of cells increases (everything else being equal), the amount of messages filtered out by recipients will decrease at the cost of an increase in the number of multicast groups that an entity needs to join.

**Entity-Based Grouping** The basic motivation behind entity-based grouping is the desire to eliminate the overhead of messages that are transmitted/received but are then filtered out by recipients. To this end we associate a single multicast group with each entity. We use the group identification  $G_X$  for entity  $X$ . An entity then multicasts all its information on this one group. In order to receive relevant information an entity needs to join

all groups corresponding to entities within its vision domain. For example, in Figure 3, Entities  $X, Y, Z$  and  $A$  will each multicast their information on addresses  $G_X, G_Y, G_Z$  and  $G_A$ , respectively. Entity  $X$  needs to join multicast groups  $G_A$  and  $G_Z$  only.

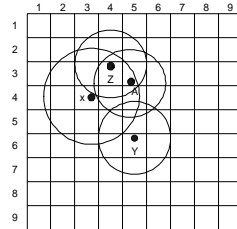


Figure 3. Entity-Based Grouping Example

The main question now is how does an entity know which groups it needs to join. In order to do this we continue to maintain the cell-based structure and corresponding multicast groups. As in the cell-based grouping, each entity joins the multicast groups corresponding to the cells with which its vision domain intersects. An entity positioned in a particular cell multicasts periodic *control messages* indicating its identity, its coordinates and its own (entity-based) multicast group address.

For example in Figure 3, Entities  $X, Y, Z$  and  $A$  will first join their corresponding cell-based groups and exchange control messages. After receiving these control messages, every entity will find out the entity-based groups it should join. As a result, entity  $X$  will join groups  $G_A$  and  $G_Z$ , entity  $Z$  will join group  $G_A$  and entity  $A$  will join group  $G_Z$ .

Note that, although the entity-based approach uses the cell-based grouping, it is used only to transmit control messages. These may be shorter and (depending on the dynamics of the game) may need to be transmitted less frequently than information messages. The advantage of the entity-based approach is that the amount of information messages that need to be filtered out by the recipients will be reduced. Our evaluation of the relative merits of the entity-based approach will consider these aspects.

### 3. A Static Model and Evaluation

Our evaluation efforts are based on a simple simulation model that tries to capture the details of the system operation that are necessary to provide an understanding of the relative performance of the grouping strategies. We begin in this section by examining a relatively static model of the system.

First we assume that the playing area is square and of unit side length. The playing area is divided into  $N \times N$  grids, therefore there are  $N^2$  square cells in area. We

also assume that all vision domains are circles centered at the location of the entity. The radius of a vision domain may vary.

In the static case a set of  $K$  entities is placed in the playing area (according to some random placement distribution). Distributions we have considered are uniform random placement over the playing area and normal distributions with origin at the center of the playing area with a given standard deviation. For vision domain radii, we use a constant distribution, a uniform distribution within given range and a normal distribution with given mean value and standard deviation.

All entities go through the following steps:

Step 1: Join appropriate cell-based groups.

Step 2: (For entity-based grouping) Send all necessary control messages; one message from each entity to the appropriate cell-based group.

Step 3: (For entity-based grouping) Join all appropriate entity multicast groups for entities within vision domain

Step 4: Send  $n$  information messages to the appropriate cell or entity multicast groups.

Although this is a static model, the parameter  $n$  is designed to represent a motion parameter. A small value of  $n$  represents a highly dynamic scenario where control messages (in the entity-based scheme) need to be sent frequently.

In this static simulation we define the following values:

Total number of cell-based group joins,  $J_{cell}$ : In step 1 above if entity  $i$  joins  $a_i$  cell-based multicast groups then  $J_{cell} = \sum_{i=1}^K a_i$ . The  $a_i$ 's will depend on the distribution of the entities, the radii of their vision domains and the division of the playing area.

Total number of entity-based group joins,  $J_{ent}$ : If in step 3 above, entity  $i$ , joins  $b_i$  entity-based groups, then  $J_{ent} = \sum_{i=1}^K b_i$ .

Total number of cell-based (information or control) message recipients,  $R_{cell}$ : If  $x_{ij}$  represents the number of entities in cell  $(i, j)$  and  $c_{ij}$  represent the number of entities that have joined cell-based group  $G_{i,j}$  (in step 1 above), then  $R_{cell} = \sum_{i=1}^N \sum_{j=1}^N x_{ij} \times c_{ij}$ .

Total number of entity-based information message recipients,  $R_{ent}$ : If  $d_i$  is the number of entities that have joined entity-based group for entity  $i$ , then  $R_{ent} = \sum_{i=1}^K d_i$ . Note that  $R_{ent} = J_{ent}$ .

Assuming that receiving an information message represents 1 unit of cost, we denote the cost of receiving a control message by  $\alpha$  and the message cost of joining and leaving a multicast group by  $\beta$ . Note that we will typically have  $\alpha \leq 1$  while  $\beta$ 's value can be larger than one.

We now use the following cost functions:

- Cost for the cell-based grouping approach is  $C_{cell} = nR_{cell} + \beta J_{cell}$
- Cost for the entity-based grouping approach is  $C_{ent} = nR_{ent} + \alpha R_{cell} + \beta(J_{cell} + J_{ent})$

Comparing the two costs (and recalling that  $R_{ent} = J_{ent}$ ) yields

$$\begin{aligned} C_{cell} > C_{ent} & \text{ if } \frac{R_{ent}}{R_{cell}} < \frac{n-\alpha}{n+\beta} \\ C_{cell} < C_{ent} & \text{ if } \frac{R_{ent}}{R_{cell}} > \frac{n-\alpha}{n+\beta} \end{aligned} \quad (1)$$

That is, given all the cost coefficients, which strategy is better is solely decided by the ratio  $\frac{R_{ent}}{R_{cell}}$  and by the values of  $n$ ,  $\alpha$  and  $\beta$ .

#### 4. Numerical Evaluation of Static Model

We performed a set of extensive simulations for the static case to study the effect of various parameters on the cost. In these simulations the costs shown in section 3 are evaluated and averaged for a large number of random placements of entities in the playing area. We report on these results in this section. Unless specified, the parameters we used are: number of entities is 500, uniformly distributed within the playing area,  $n = 10$ ,  $\alpha = \beta = 1$ , the playing area is divided into  $30 \times 30$  grids. All costs we report are per entity costs (i.e., costs derived in section 3 divided by  $K$ , the number of entities).

Figure 4 shows  $J_{cell}$ ,  $R_{cell}$  and  $J_{ent}$  for different cell sizes as a function of the vision domain radius of entities. We consider constant vision domain radii for all entities.  $R_{ent}$  is independent of the number of cells on the playing area. In broadcast (i.e.,  $1 \times 1$  grid) case, each entity only needs to join 1 cell-based group but every message sent by an entity will reach all the entities in the playing area. As the number of grids increases,  $R_{cell}$  decreases and  $J_{cell}$  increases; but when the number of grids is large,  $R_{cell}$  slowly decreases to a limiting value whereas  $J_{cell}$  increases as a linear function of the number of cells in the playing area. This suggests that to reduce overall cost in both cell-based grouping and entity-based grouping we should divide the playing area into fewer cells, especially when the vision domain radius of entity is large. This conclusion is independent of the number of entities because on average  $J_{cell}$  is independent of the number of entities.

Figure 5 shows the static cost as a function of the vision domain radius under different cell sizes. Again we use constant vision domain radii for all entities. We can see that having large number of cells reduces cost only when the vision domain radius is small but increases cost rapidly when the vision domain radius increases. When the vision domain radius is not very small (e.g., greater

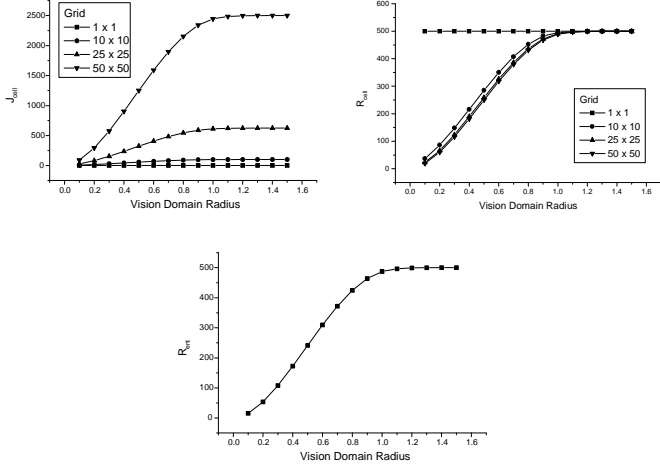


Figure 4.  $J_{cell}$ ,  $R_{cell}$  and  $R_{ent}$  in the static case

than 0.1),  $10 \times 10$  grid gives very good performance for a large value range of  $\alpha$ ,  $\beta$  and  $n$ .

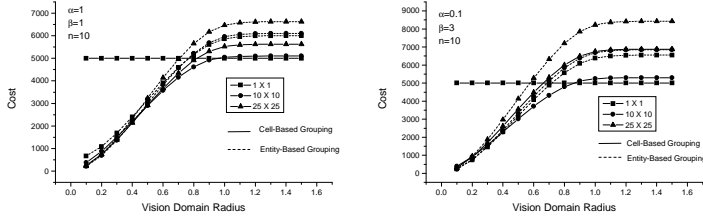


Figure 5. Static Cost as a Function of Vision Domain Radius

We then consider which strategy is better under a given configuration. Figure 6 shows the corresponding  $\frac{R_{ent}}{R_{cell}}$  to Figure 5. We can see from the figures that when  $\frac{R_{ent}}{R_{cell}} > \frac{n-\alpha}{n+\beta}$ , the cell-based approach is better.

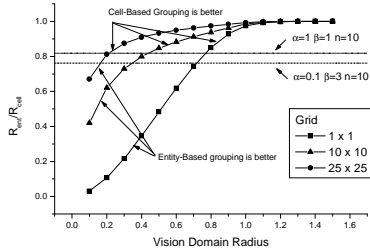


Figure 6. The effect of number of cells

Simulations show that in most cases which approach is better is mainly decided by the average vision domain radius, it has little relationship with the other properties of the entities and the playing area, such as number of entities, entity placement distributions and vision do-

main radius distributions. Due to space limitation, we omit these graphs here.

In the cost function we derived above, the parameter  $n$  is an important one. Different applications may have very different values of  $n$ . Figure 7 shows the relationship between cost and  $n$ . The cost is a linear function of  $n$ , its coefficients are  $R_{cell}$  and  $R_{ent}$  in the two approaches, respectively. Since  $R_{cell}$  is always greater than  $R_{ent}$ , the cost of cell-based grouping will increase faster than that of entity-based grouping. From Equation 1 we can derive at the crossing points (i.e., the points at which the cost of both strategies equal)  $n = \frac{\alpha+\beta}{1-\frac{R_{ent}}{R_{cell}}} - \beta$ .

Because  $\frac{R_{ent}}{R_{cell}}$  is an increasing function of vision domain radius, the value of  $n$  at crossing point is also an increasing function.

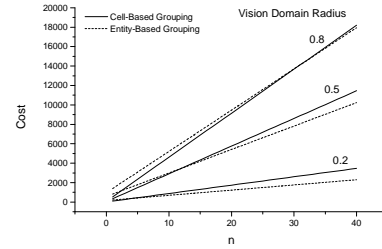


Figure 7. Relationship between parameter  $n$  and static costs

$\beta$  is another important parameter, depending on the property of the multi-user game and the infrastructure of the multicast network,  $\beta$  may be much larger than 1 (recall that the cost of receiving a data message is 1). Figure 8 shows the relationship between costs and  $\beta$ . From this figure we can see a large  $\beta$  will favor cell-based grouping. We can derive the value of  $\beta$  at crossing points:  $\beta = (n - \alpha) / \frac{R_{ent}}{R_{cell}} - n$ . Since  $\frac{R_{ent}}{R_{cell}}$  is an increasing function of vision domain radius, the value of  $\beta$  at crossing point is a decreasing function of vision domain radius.

## 5. A Dynamic Model and Its Evaluation

We model the dynamic system as a continuous state, discrete time Markov process. The initial state is that obtained from the static simulation described above. After the randomly placed entities join the appropriate groups, and then send  $n$  information messages, another placement of the entities is obtained by having entities move in a random direction that is independently chosen for each entity. Each entity moves a distance that is chosen from a distribution which may be different for different entities. Entities whose motion carries them to

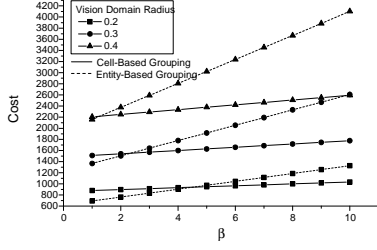


Figure 8. Relationship between parameter  $\beta$  and static costs

the boundary of the playing area are mirror-reflected at the boundary. After the motion is complete, each entity leaves all the groups (entity or cell) that are not needed any more and joins new ones as needed. Each entity then sends  $n$  messages before moving again.

For example in Figure 9, in simulation step  $i - 1$ , all entities send  $n$  data messages to their entity-based groups, i.e.,  $G_X$ ,  $G_Y$  and  $G_Z$ . Then all the entities choose a random direction and distance and make a movement accordingly. The simulation step  $i$  begins with all the entities leaving the cell-based groups that no longer required and joining new cell-based groups. Table 2 shows the cell-based and entity-based groups joined and left by the entities at the beginning of step  $i$ .

Entity	Join	Leave
X	$G_{6,3}, G_{7,3}, G_{6,4}, G_{7,4}, G_{6,5}, G_{7,5}, G_{6,6}, G_{7,6}, G_Y$	$G_{2,2}, G_{3,2}, G_{4,2}, G_{5,2}, G_{2,2}, G_{1,3}, G_{2,3}, G_{1,4}, G_{2,4}, G_{1,5}, G_{2,5}, G_{2,6}, G_Z$
Y	$G_{3,5}, G_{3,6}, G_{3,7}, G_{3,8}, G_{4,8}, G_{5,8}$	$G_{4,4}, G_{5,4}, G_{6,4}, G_{6,5}$
Z	$G_{6,1}, G_{6,2}, G_{6,3}$	$G_{3,4}, G_X$

Table 2. Dynamic Model Example

After joining and leaving necessary cell-based groups, every entity sends one control message to the multicast group corresponding to the cell in which it resides. For example, entities  $X$ ,  $Y$  and  $Z$  will send control messages which contain coordinates and entity-based group addresses to groups  $G_{5,4}$ ,  $G_{4,6}$  and  $G_{5,2}$  respectively. Using received control messages, every entity can find out the entity-based groups it should join or leave, in our example at step  $i$ , entity  $X$  will leave  $G_Z$  and join  $G_Y$ , entity  $Z$  will leave  $G_X$ . Then each entity will send  $n$  data messages again to their corresponding entity-based groups before moving into simulation step  $i + 1$ .

We define  $J_{cell}^l, J_{ent}^l, R_{cell}^l, R_{ent}^l$  as the number of joins and information message recipients (as defined

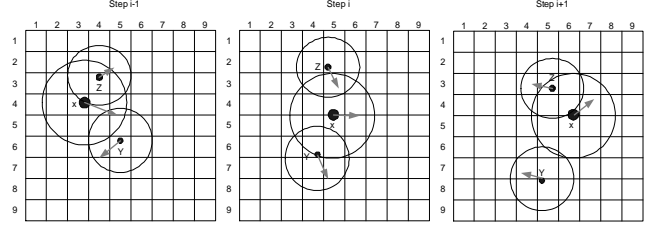


Figure 9. Dynamic Model Example

previously) in the  $l$ th step of the Markov process. We also further define  $L_{cell}^l$  and  $L_{ent}^l$  as the number of group leaves in the  $l$ th step. The total cost of each step is then given by

$$C_{cell}^l = nR_{cell}^l + \beta J_{cell}^l + \beta L_{cell}^l \quad (2)$$

$$C_{ent}^l = nR_{ent}^l + \alpha R_{cell}^l + \beta (J_{cell}^l + J_{ent}^l) + \beta (L_{cell}^l + L_{ent}^l) \quad (3)$$

We are interested in the time averages of the costs and these are given simply as  $C_{cell/ent} = (1/T) \sum_{l=1}^T C_{cell/ent}^l$ . Similarly  $J_{cell}$ ,  $J_{ent}$ ,  $R_{cell}$  and  $R_{ent}$  are also time average of the corresponding quantities. Note that we do not consider the cost of initializing the first state, simply the cost of transitions after the first state.

## 6. Numerical Evaluation of Dynamic Model

In this section we report the results of simulating of our dynamic model. Because the initial distribution of entity placement is not important when we take a long time average, we only simulate with an initial uniform distribution of entity placement. Unless specified, the parameters we used are: the number of entities are 500, uniformly distributed within the playing area. Entities have constant vision domain radius 0.5. The motion steps are also constant in magnitude and the motion direction are uniformly distributed. The playing area is divided into  $30 \times 30$  grids. We also have  $n = 10$ ,  $\alpha = \beta = 1$ .

Figure 10 shows  $J_{cell}$ ,  $R_{cell}$ ,  $J_{ent}$  and  $R_{ent}$  as a function of the motion step for different cell divisions.  $J_{ent}$  and  $R_{ent}$  are independent of the cell divisions. So for given parameters  $\alpha$ ,  $\beta$ ,  $n$ , the optimal number of cells is decided by  $J_{cell}$  and  $R_{cell}$ . These values, in turn, are functions of vision domain radius and motion step. The relationship between  $J_{cell}$ ,  $R_{cell}$  and number of cells in the dynamic case is similar to the relationship between  $J_{cell}$ ,  $R_{cell}$  and number of cells in the static model, so we can draw similar conclusions. From the figure, we can see small motion step favors the use of a large number of cells.

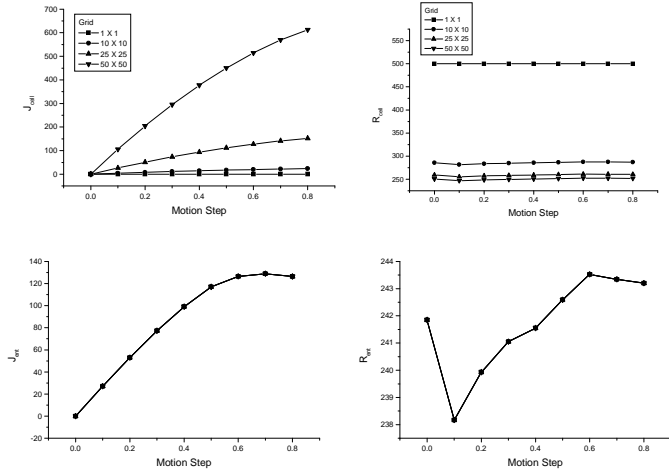


Figure 10.  $J_{cell}$ ,  $R_{cell}$ ,  $J_{ent}$  and  $R_{ent}$  in the dynamic case

Figure 11 shows the cost as a function of the motion step for different cell sizes. We can see that only when the motion step is very small, is dividing the playing area into a large number of cells advantageous.

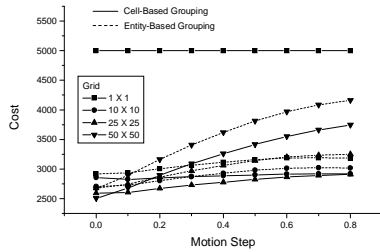


Figure 11. Dynamic cost as a function of motion step

Figure 12 shows the combined effect of vision domain radius and motion step on the dynamic costs. We can see from the figure that the vision domain radius is a dominant factor. Which approach is better is mainly decided by the vision domain radius with a small vision domain radius favoring the entity-based grouping approach.

Figure 13 shows the relationship between the average cost per entity and the total number of entities in the playing area as a function of motion step. The average cost should be a linear function of the number of entities in the playing area, because the number of peers an entity can see is proportional to the density of the entities which, in turn, is proportional to the total number of entities in the playing area given the playing area is fixed. Hence, the relative costs of the entity-based and cell-based grouping approaches are independent of the number of entities.

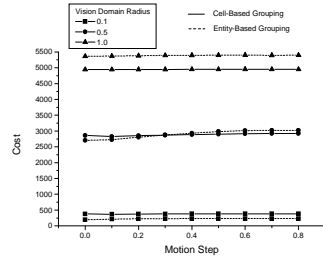


Figure 12. The combined effect of motion step and vision domain radius

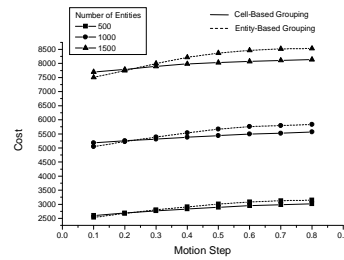


Figure 13. The effect of number of entities

Figure 14 shows the relationship between the cost and distributions of motion step and vision domain radius. We can see from the figure that generally the distribution of these two radii has little effect on the costs.

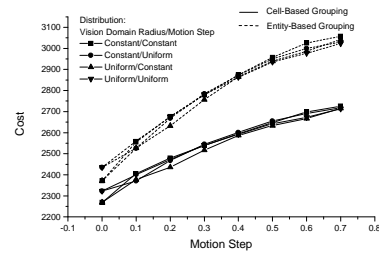
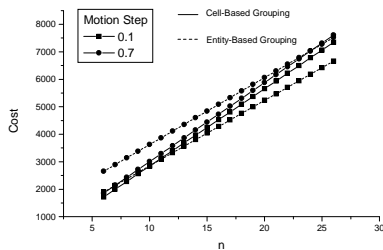


Figure 14. The effect of distribution of motion step and vision domain radius

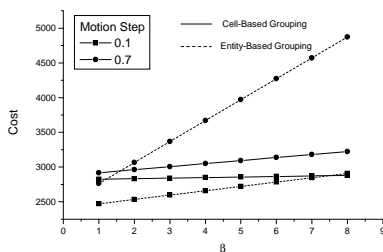
Figure 15 shows the effect of the number of data messages per step,  $n$ , on the cost. The costs of both strategies are linear functions of  $n$ . Because  $R_{cell}$  is always greater than  $R_{ent}$ , the cost of cell-based grouping always grows faster. Since on average,  $R_{cell}$  and  $R_{ent}$  are both independent of the motion step, increasing the motion step does not change the slope of the linear functions, it only slightly increases their offsets. As a result, the value of  $n$  at the crossing point of the cost functions of the cell-based and entity-based grouping approaches is

an increasing function of the motion step.



**Figure 15. Relationship between parameter  $n$  and dynamic cost**

Figure 16 shows how the costs change as a function of  $\beta$  for both strategies. The coefficient of  $\beta$  in the entity-based grouping cost is always greater than the corresponding coefficient in the cell-based grouping cost. Therefore, the cost of the entity-based grouping always grows faster. From the dynamic cost equations we get that at the crossing points:  $\beta = ((n - \alpha)R_{cell} - nR_{ent}) / (J_{cell} + L_{cell})$ . Because  $R_{cell}$  and  $J_{cell}$  are independent of the motion step whereas  $J_{cell}$  and  $L_{cell}$  are increasing functions of the motion step, the value of  $\beta$  at the crossing points is a decreasing function of the motion step.



**Figure 16. Relationship between parameter  $\beta$  and dynamic cost**

## 7. Concluding Remarks

In this paper we have considered schemes for subdividing entities in a networked multi-user game into multicast groups. The main objective of this subdivision is to reduce the overall cost of state dissemination for such gaming applications. The cost we considered had two main components: data-message reception cost (including messages received but filtered out) and group formation control message cost. Our work targeted an understanding of the tradeoff between these costs in the design

and configuration of grouping strategies. To this end we developed two simple models that captured many of the interesting features of the operation of a multi-user game. We then reported on the results of simulations using these models that allowed us to examine the effect of various parameters on the overall cost of the system.

Our work in this area needs to continue with consideration of other grouping strategies such as hybrids of the cell-based and entity-based approaches. Our simple models assume linear cost functions, synchronized data transfer and instantaneous group joining and leaving. One can use a more detailed cost modeling with the appropriate models of the network topology and the operation of network protocols. Ultimately, one should consider whether modifications to the existing Internet multicast support can reduce the overheads associated with multi-user games.

## References

- [1] L. Gautier and C. Diot, "Design and evaluation of mimaze, a multi-player game on the internet," in *Proceedings of IEEE Multimedia Systems Conference*, June 1998.
- [2] J. W. Barrus, R. C. Waters, and D. B. Anderson, "Locales and beacons: Efficient and precise support for large multi-user virtual environments," Tech. Rep. TR96-02a, Mitsubishi Electric Research Laboratory, Cambridge, MA, 1996. <http://www.merl.com/reports/TR96-02a/index.html>.
- [3] K. Almeroth, "The evolution of multicast: From the Mbone to inter-domain multicast to Internet2 deployment," *IEEE Network Special Issue on Multicasting*, January/February 2000.
- [4] S. Deering, "Host extensions for IP multicasting," <http://ds.internic.net/rfc/rfc1112.html>.
- [5] D. Waitzman, C. Partridge, and S. Deering, "Distance vector multicast routing protocol," <http://ds.internic.net/rfc/rfc1075.html>.
- [6] S. D. Deering, D. Farinacci, V. Jacobson, C. Liu, and L. Wei, "The PIM architecture for wide-area multicast routing," *ACM Transactions on Networks*, APRIL 1996.
- [7] D. J. V. Hook, S. J. Rak, and J. O. Calvin, "Approaches to relevance filtering," in *11th DIS Workshop on Standards for the Interoperability of Distributed Simulations*, September 1994.
- [8] S. J. Rak and D. J. V. Hook, "Evaluation of grid-based relevance filtering for multicast group assignment," in *14th DIS Workshop on Standards for the Interoperability of Distributed Simulations*, March 1996.
- [9] "IEEE standard for distributed interactive simulation – application protocols (IEEE std 1278.1-1995)," 1995. IEEE Computer Society.
- [10] "IEEE standard for distributed interactive simulation – communication services and profiles (IEEE std 1278.2-1995)," 1995. IEEE Computer Society.
- [11] E. Lety, T. Turelletti, and F. Baccelli, "Cell-based multicast grouping in large-scale virtual environments," Tech. Rep. 3729, INRIA, July 1999.
- [12] T. Wong, R. Katz, and S. McCanne, "An evaluation of preference clustering in large-scale multicast applications," in *Proceedings of IEEE INFOCOM*, (Tel-Aviv, Israel), March 2000.
- [13] B. N. Levine, J. Crowcroft, C. Diot, J. J. Garcia-Luna-Aceves, and J. F. Kurose, "Consideration of receiver interest in content for IP delivery." Submitted for Publication, 1999.
- [14] R. C. Waters, D. B. Anderson, J. W. Barrus, D. C. Brogan, M. A. Casey, S. G. McKeown, T. Nitta, I. B. Sterns, and W. S. Yerazunis, "Diamond park and spline: A social virtual reality system with 3D animation, spoken interaction, and runtime modifiability," Tech. Rep. TR95-16a, Mitsubishi Electric Research Laboratory, Cambridge, MA, 1995. <http://www.merl.com/reports/TR95-16a/index.html>.
- [15] B. N. Levine and J. J. Garcia-Luna-Aceves, "Improving internet multicast with routing labels," in *IEEE International Conference on Network Protocols*, pp. 241–50, October 1997.