in IEEE/ACM Transactions on Networking Journal, Vol. 12 , No. 2, pp. 247-260, April 2004

SCORE : a Scalable Communication Protocol for Large-Scale Virtual Environments

Emmanuel Léty[1], Thierry Turletti[1], François Baccelli[2]
Emmanuel.Lety@UDcast.com, {Thierry.Turletti,Francois.Baccelli}@inria.fr
[1]INRIA, 2004 route des Lucioles, BP 93, 06902 Sophia Antipolis, FRANCE.
[2]INRIA-ENS, 45 rue d'Ulm, 75005 Paris, FRANCE.

***Abstract*— This paper describes and analyses SCORE, a scalable multicast-based communication protocol for Large-Scale Virtual Environments (LSVE) on the Internet. Today, many of these applications have to handle an increasing number of participants and deal with the difficult problem of scalability. We propose an approach at the transport-layer, using multiple multicast groups and multiple agents. This approach involves the dynamic partitioning of the virtual environment into spatial areas and the association of these areas with multicast groups. It uses a method based on the theory of planar point processes to determine an appropriate cell-size, so that the incoming traffic at the receiver side remains with a given probability below a sufficiently low threshold. We evaluate the performance of our scheme and show that it allows to significantly improve the participants' satisfaction while adding very low overhead.**

***Index Terms*— Area Of Interest Manager (AOIM), Cell-based grouping, communication protocol, Large-Scale Virtual Environments (LSVE), multiple multicast groups, scalability.**

I. INTRODUCTION

This paper describes and analyses SCORE, a scalable multicast-based communication protocol for Large-Scale Virtual Environments (LSVE) on the Internet. Such Virtual Environments (VE) include massively multi-player games, Distributed Interactive Simulations (DIS) [1], and shared virtual worlds. Today, many of these applications have to handle an increasing number of participants and deal with the difficult problem of scalability. Moreover, the real-time requirements of these applications make the scalability problem more difficult to solve. In this paper, we consider only many-to-many applications, where each participant is both source and receiver. We also make the assumption that a single data flow is generated per participant. However, we believe that most of the results and mechanisms presented in this paper can be easily adapted to more complex applications that use several media types or layered encodings [2].

The use of IP multicast solves part of the scalability problem by allowing each source to send data only once to all the participants without having to deal with as many sequential or concurrent unicast sessions as the number of participants. However, with a large number of heterogeneous users, transmitting all the data to all the participants dramatically increases the probability of congestion within the network and particularly at the receiver side. Indeed, processing and filtering all the packets received at the application level could overload local resources, especially if the rendering module is already processor intensive [3]. [4] shows that in a group communication setting, the percentage of useless (or *superfluous*) information received by each participant increases with the number of data flows and the number of users. This is not surprising since within a VE, each participant simultaneously interacts with only a limited set of other participants. The superfluous information represents a cost in terms of network bandwidth, routers buffer occupation and end-host resources, and is mainly responsible for the degradation of performance in LSVE.

We argue that the superfluous received traffic has to be filtered out before it reaches the end-host. The main difficulty in this filtering mechanism comes from the heterogeneity and the dynamicity of the receivers, not only in terms of bandwidth and processing power but also in terms of data of interest, virtual and physical locations. In [5] and [6], network-layer approaches are proposed to introduce ”filters” in the router forwarding process, customizing the data delivered to multicast receivers. However, these propositions require modifications in the routers and are unfortunately not yet deployed in the Internet.

The aim of this paper is neither to propose a new IP multicast model nor to come up with a network-layer approach, adding new mechanisms in the routers. Instead, we present a transport-layer filtering mechanism with multiple agents, assuming that all the users are capable of receiving multicast transmissions. Our approach involves the dynamic partitioning of the VE into spatial areas called *cells* and the association of these cells with multicast groups. We describe a method, based on the theory of planar point processes, to determine an appropriate *cell-size* so that the incoming traffic at the receiver side remains with a given probability below a sufficiently low threshold. We then propose mechanisms to dynamically partition the VE into cells of different sizes, depending on the density of participants per cell, the number of available multicast groups, and the link bandwidth and processing resources available per participant.

The rest of the paper is organized as follows. Section II reviews the limitations of the current IP multicast model, presents the cell-based grouping strategy, and examines the tradeoff in selecting the cell-size parameter. Section III describes a model allowing one to evaluate various mean values of interest. The section then analyses the impact of the cell-size on the traffic received at the receivers and several quantities such as the participant's mean residence time within a multicast group. Section IV describes SCORE, a scalable communication protocol that implements a dynamic cell-based grouping strategy using a limited number of multicast groups. Section V evaluates the performance and the overhead of SCORE using a set of intensive experimentations. Finally, Section VI discusses related works, and Section VII concludes the paper and presents directions for future work.

II. Motivation

In this section, we examine the different limitations in using multiple multicast groups and the issues involved in selecting the best size of cell.

A. Multiple multicast groups limitations

There are several limitations on the use of multiple multicast groups. First, we have to consider that today, multicast groups are not inexhaustible resources: the number of available multicast groups in IPv4 is limited to 268 million Class D addresses[1]. Moreover, there is an increasing number of applications that require several multicast addresses, such as layered coding based video-conferencing, or DIS applications. Therefore, the widespread use of multicast increases the probability of address collisions. A few solutions have already been proposed in the literature to solve the multicast address allocation problem. For example, a scalable multicast address assignment based on DNS has been proposed in [7]. Another alternative could be the use of the Multicast Address Set Claim (MASC) protocol which describes a scheme for the hierarchical allocation of Internet Class D addresses [8]. Some alternatives to the current IP multicast model have also been proposed: [9] describes a multicast address space partitioning scheme based on the port number and the unicast host address. In *Simple Multicast*, a multicast group is identified by the pair (address of the group, address of the core of the multicast tree), which gives to each core the full set of Class D addresses space [10]. In *EXPRESS*, a multicast *channel* is identified by both the sender's source address and the multicast group [11]. Finally, with IPv6, the multicast address space will be as large as the unicast address space, so this will solve the multicast address assignment problem. However, all these proposals are still active research areas and are not currently available on the Internet.

Secondly, multicast addresses are expensive resources. The routing and forwarding tables within the network are limited resources with limited size. For each multicast group, all the routers of the associated multicast tree have to keep information on which ports are in the group. Hosts and routers also need to report periodically their IP multicast group memberships to their neighboring multicast routers using IGMP[12]. Moreover, some routing protocols such as DVMRP[13] rely on the periodic flooding of messages throughout the network. All this traffic has a cost, not only in terms of bandwidth but also in terms of join and leave latency, which should be taken into consideration for interactive applications [14]. Indeed, when a participant sends a join request, it can take several hundreds of milliseconds before the first multicast packet arrives. Such costs should be obviously considered in Large-Scale Multicast Applications (LSMA) and argue in favor of a bigger cell-size, and therefore, of a limited number of multicast groups.

[1]IPv4 Class D addresses use 28-bits address space.

B. The cell-size tradeoff

In this paper, we focus on the *cell-based grouping strategy* which basically consists in partitioning the VE into cells and assigning to each cell a multicast group. During the session, each participant identifies the cell he is currently "virtually" located in, and sends his data to the associated multicast group. To receive the data from the other participants included in the area in which he is interested in (i.e., his *area of interest*), each participant has to join the multicast groups associated with the cells that intersect his area of interest. Similarly, when a participant moves, he needs to leave the multicast groups associated with the cells which do not intersect his area of interest anymore.

The cell-based grouping strategy is particularly suitable on VEs that can easily be partitioned into virtual areas (e.g., virtual Euclidean spaces). However, the main difficulty in this partitioning is to find the appropriated cell-size. Indeed, decreasing the cell-size increases the overhead associated with dynamic group membership whereas increasing the cell-size increases the unwanted information received per participant [15].

Two approaches are possible to estimate the best cell-size in a LSVE: the first approach requires the pre-calculation of a static cell-size parameter, which remains the same during the whole session. The second approach consists in dynamically re-estimating the cell-size during the session, taking into account various parameters. To motivate the choice of one of these two approaches, let us first identify the parameters involved in the cell-size calculation and then, examine the impact of the variability of these parameters on the appropriate cell-size.

- **The number of available multicast groups** is an important parameter to take into account for the cell-size calculation because it gives a lower bound on the cell-size. As the number of multicast groups used is inversely proportional to the size of the cell, a small set of available multicast groups will lead to a bigger cell-size.
- **The receivers capacities** are determined by the link capacities and the processing power available per receiver. Typically, this parameter limits the amount of traffic that the receivers can handle. Assuming each user roughly generates the same amount of traffic, the incoming traffic per receiver grows linearly with the total number of sources contained in the multicast groups to which he has subscribed. In other words, the incoming traffic per receiver is a function of the number of entities located in the cells included or intersected by his area of interest. Nevertheless, some of these participants may be located outside the area of interest but inside a cell that includes

this area of interest. The ratio between the corresponding number of unwanted participants and the total number of sources received represents the percentage of *superfluous* traffic received. So, the cell-size and more particularly the ratio between the cell-size and the size of the area of interest, have a direct impact on the amount of unwanted traffic.

- **The density of participants** represents the ratio between the number of participants and the size of the VE. In the cell-based grouping strategy, the area of interest is approximated by the smallest set of cells covering the area of interest. In the rest of the paper, we refer to the difference between these two areas as the *superfluous area*, see Figure6. So, the density of participants in a VE not only has an impact on the average number of participants located in the area of interest, but also on the superfluous area. Consequently, the participant density has an impact on the average superfluous traffic. A smaller cell-size could allow a better approximation of the area of interest and a significant reduction of superfluous area and its corresponding traffic. Thus, depending on the participant density, the superfluous traffic and its negative impact on the application performance could also be significantly reduced.
- **The participant velocity** can be used in a cell-based grouping VE to estimate the bandwidth overhead generated when participants cross cells, and the ratio between the join and leave latency and the mean time that the participant stays in each cell. In cell-based grouping, each cell is assigned to a multicast group. Therefore, joining and leaving a cell in a VE corresponds to joining or leaving an IP multicast group in reality. Even though there are enough multicast addresses available to assign each cell, there are several concerns while using multiple multicast groups. First, join and leave control messages use some additional bandwidth between the end-users and their nearest multicast routers. Second, when the participants join or leave multicast groups, they create a significant processing overhead among the routers of the associated multicast trees. Finally, there is a huge concern with the join and leave latency, especially for interactive VE in which the real-time requirement of the application is essential to preserve.

III. Models and simulations

This section introduces models of area of interest and of participants based on random point processes which are inspired by the stochastic geometry approach proposed in [16]. This model allows us to evaluate various mean values of interest and later on to address issues pertaining to mobility.

A. Static participants

First, we restrict the problem to static participants using the following assumptions :

- The participants are static and located on the plane according to a random homogeneous Poisson point process of intensity λ [17];
- The cells form an infinite regular square grid on the plane;
- The area of interest $IArea$ is a square of area r^2 centered on a typical participant (referred to as the observer in what follows).

We denote by s the cell-size (i.e., the distance between two adjacent horizontal or vertical cell boundaries), and $CellArea$ the cell area s^2. We focus here on the distribution of the number M of cells intersecting the area of interest and on that of N, the number of participants located in these cells (excluding the observer).

Let $\lfloor x \rfloor$ denote the integer part of the real number x, namely the largest integer smaller than or equal to x. Let

$$k = k(r,s) = \lfloor \frac{r}{s} \rfloor \quad (1)$$

$$p = p(r,s) = \frac{r}{s} - \lfloor \frac{r}{s} \rfloor . \quad (2)$$

Note that $0 \le p < 1$. We prove below that:

1) The law of M is a point mass distribution on the three integers $(k+1)^2$, $(k+1)(k+2)$ and $(k+2)^2$, with parameters

$$P[M=(k+2)^2] = p^2, \quad (3)$$

$$P[M=(k+1)(k+2)] = 2p(1-p), \quad (4)$$

$$P[M=(k+1)^2] = (1-p)^2. \quad (5)$$

2) The generating function of N, is given by the following formula:

$$E[z^N] = p^2 e^{-\lambda s^2 (k+2)^2 (1-z)} + 2p(1-p) e^{-\lambda s^2 (k+1)(k+2)(1-z)} + (1-p)^2 e^{-\lambda s^2 (k+1)^2 (1-z)}. \quad (6)$$

Consider the configuration *seen* by the observer. Assume the observer to be located at point $(r/2, r/2)$, so that the area of interest I is the square $[0,r] \times [0,r]$. Seen from this participant, the grid is as randomly shifted. More precisely, from well known properties of renewal processes [18], this typical configuration is that where the grid has one of its intersection points at (X,Y), where X and Y are independent random variables, each with a uniform distribution on the interval $[0,s]$. Under such a configuration, if $X \le x_0$, where x_0 is defined by the relation

$$x_0 = r - s \lfloor \frac{r}{s} \rfloor ,$$

then the number of cells which intersect the horizontal sides of I is exactly $k+2$, with k defined as above. If $X > x_0$, this number is $k+1$. The same argument gives the number of cells intersecting the vertical sides of I. Using the independence and the uniformity, we obtain that with probability $\left(\frac{x_0}{s}\right)^2 = p^2$, the number of cells intersecting I is $(k+2)^2$. We obtain the other point masses of the law of M via similar arguments.

We now give the proof of the second formula. We have

$$N = \sum_{i=1}^{M} N_i ,$$

where the random variables N_i give the numbers of participants in the cells which intersect I. Each of these variables is Poisson

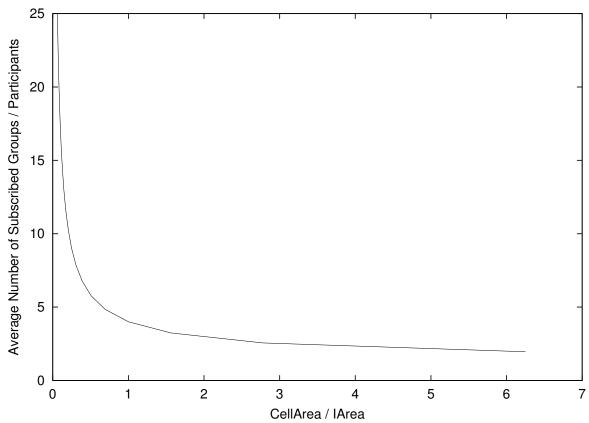

Fig. 1. Average number of subscribed groups / participant

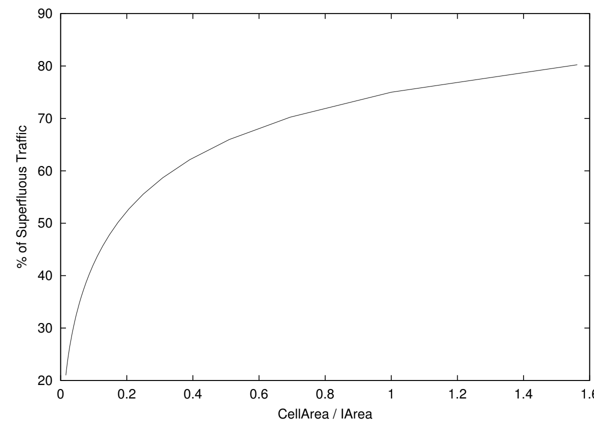

Fig. 2. Percentage of superfluous traffic

with parameter λs^2. In addition, the random variables N_i and M are independent. Therefore, we can apply the rule giving the generating function of a random sum of random variables, which states that the generating function of N is $\psi(\phi(z))$ where ϕ is the generating function of N_1 and ψ that of M [19]. Here, we have

$$\phi(z) = e^{-\lambda s^2(1-z)}$$

and

$$\psi(z) = p^2 z^{(k+2)^2} \; + \; 2p(1-p)z^{(k+1)(k+2)} + \; (1-p)^2 z^{(k+1)^2},$$

and the second formula follows immediately from this. As direct consequences of these formulas, we obtain the following expressions for:

1) The mean value of M and its variance:

$$\begin{aligned} E[M] &= p^2(k+2)^2 + 2p(1-p)(k+1)(k+2) \\ &\quad +(1-p)^2(k+1)^2 \\ \text{var }(M) &= p^2(k+2)^4 + 2p(1-p)(k+1)^2(k+2)^2 \\ &\quad +(1-p)^2(k+1)^4 - (E[M])^2. \end{aligned}$$

2) The mean value of N : $E[N] = \lambda s^2 E[M]$.
3) The variance of N (with the above notation) :

$$\begin{aligned} \text{var }(N) &= E[M]\text{var } N_1 + E[N_1]^2 \text{var } M \\ &= E[M]\lambda s^2 + \text{var } M \lambda^2 s^4. \end{aligned} \quad (7)$$

4) The probability that N is less than a threshold n, where n is a non–negative integer :

$$\begin{aligned} P[N \leq n] &= p^2 g_n((k+2)^2) \\ &\quad +2p(1-p)g_n((k+1)(k+2)) \\ &\quad +(1-p)^2 g_n((k+2)^2), \end{aligned} \quad (8)$$

where

$$g_n(m) = e^{-\lambda s^2 m} \sum_{i=0}^{n} \frac{(\lambda s^2 m)^i}{i!}. \quad (9)$$

Now, we analyze the impact of $CellArea$ and the participant intensity on the traffic N received by participant, the average number of subscribed multicast groups per participant, and the percentage of superfluous traffic received. In order to be as generic as possible, we focus more particularly on the impact of the ratio between $CellArea$ and $IArea$ (i.e., $\frac{s^2}{r^2}$). Note that we assume here that all the participants generate the same amount of traffic.

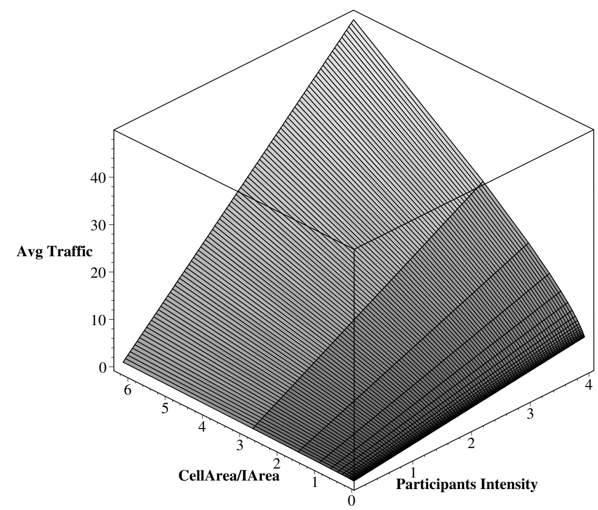

Fig. 3. Average traffic (number of sources) / participant

Figure 1 shows that the average number of subscribed multicast groups per participants $E[M]$ decreases sharply as $CellArea$ approaches $IArea$. However, as $CellArea$ increases further, the average number of subscribed groups decreases slowly to 1.

Figure 2 plots the average percentage of superfluous traffic out of the total received traffic by a participant. Since the participants are located on the plane according to a random homogeneous Poisson point process, this percentage is equal to: $1 - \frac{r^2}{E[M]s^2}$. We observe that when $CellArea$ is larger than $IArea$, more than 70% of the traffic is superfluous. This figure also suggests that when $CellArea$ is smaller than $IArea$, a slight diminution of $CellArea$ decreases significantly the superfluous traffic received. However, it is important to note that 70% of superfluous traffic is acceptable compared to the situation where all the users communicate on a single

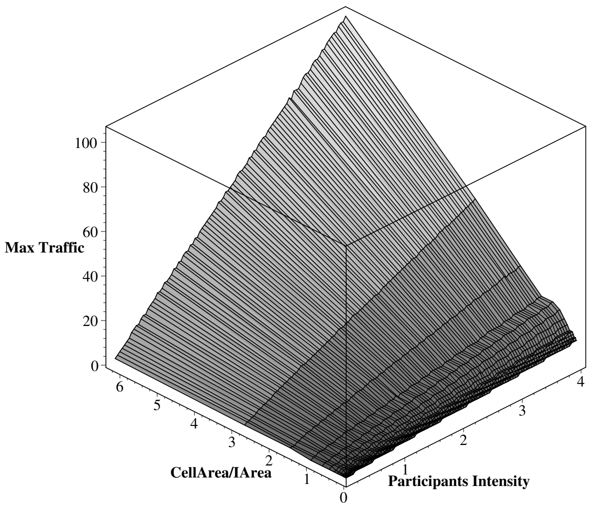

Fig. 4. Max traffic threshold (number of sources) / participant with p = 0.95

multicast group [3]. Indeed, with a single multicast group and a large number of participants, almost all the traffic received would be superfluous.

Figure 3 shows the average traffic received by a participant, depending on the intensity of participants in the VE, and the ratio between $CellArea$ and $IArea$. The participant intensity represents here the average number of participants per $IArea$: λr^2. Such a way to express the density of participant in a VE is very useful, as it allows us to modify $CellArea$ without having an impact on the value of the density. The results show that for a given value of participant intensity, it is possible to find the largest ratio between $CellArea$ and $IArea$, so that the average traffic remains under a sufficiently low threshold. The average traffic is given by : $\lambda s^2 E[M]$.
Finally, Figure 4 probably shows the most interesting results. In order to satisfy participants in a VE, it is better to determine an appropriate $CellArea$ so that the incoming traffic remains with a high probability below the maximum traffic that they can handle. This probability reflects the tradeoff between the satisfaction of the users and the required number of multicast groups. Figure 4 shows that for a given intensity of participants, it is possible to find the largest $CellArea$ (i.e., the smallest number of multicast groups), so that the incoming traffic remains below a sufficiently low threshold with a probability of 0.95. Moreover, for a given $CellArea$, we observe that this traffic increases linearly with the intensity of participants.

B. Dynamic participants

This section introduces a model of mobility which is compatible with the assumption that the point process of participants is Poisson at any time, and which allows us to derive various mean values of interest in relation with mobility. This includes quantities such as:

- The handover in and out of a multicast group, defined as the time point process intensity of the boundary crossings of the corresponding cell by moving participants;
- The mean residence time of a typical participant within a multicast group, namely within the corresponding cell.

The assumption is still that participants are initially located according to a Poisson point process of intensity λ. No participant enters or leaves the game. Nevertheless, each participant moves on the plane according to an independent random motion described as follows: a pair of random variables (V_i, θ_i), is associated with participant i, where $V_i \in \mathbb{R}_+$ is the random velocity of the participant and $\theta_i \in [0, 2\pi)$ his random direction. It is assumed that all pairs (V_i, θ_i) are independent and identically distributed and that the random variables (V_1, θ_1) are independent, with V_1 of density f on $\mathbb{R}_+$ and with θ_i uniform on $[0, 2\pi)$. Thanks to the so called displacement theorem (see [17], p. 61), the point process giving the location of all participants is still a Poisson point process of intensity λ at any time t, so that the results of the previous section are still valid at any such time.

Let σ be a fixed segment of length u, which we can assume to be located on the horizontal axis without loss of generality. The set of participants with a motion pair equal to (v, θ) and which cross σ between time 0 and t is that initially located in a parallelogram of area $uvt|\sin\theta|$. The set of participants with a motion pair in the set $[v, v + dv] \times [\theta, \theta + d\theta]$ is Poisson with intensity $\lambda f(v) dv \frac{d\theta}{2\pi}$. Therefore, the mean number of participants crossing σ between time 0 and t is

$$\int_0^\infty \int_0^{2\pi} uvt|\sin\theta|\lambda f(v) dv \frac{d\theta}{2\pi} = \frac{2\lambda u E[V] t}{\pi}.$$

Consider a typical cell, namely a square with perimeter $4s$. From what precedes, we get the following expression for the mean value of the handover in and out this cell per unit of time:

$$H = \frac{8\lambda s E[V]}{\pi}. \tag{10}$$

Due to the displacement theorem, we can still use the Poisson law for the number of participants in this cell at any time. Its mean value is λs^2. Since the intensity of the entrances into the cell is $\frac{H}{2}$, Little's law gives the following expression for the mean residence time of a participant in a typical multicast group:

$$E[W] \quad = \quad \frac{2\lambda s^2}{H} = \frac{\pi s}{4E[V]}. \tag{11}$$

Figure 5 shows the mean residence time per cell $E[W]$ as a function of the participant mean velocity $E[V]$. We express the velocity in cell-size per second. We observe that the mean residence time decreases exponentially as the mean velocity approaches 1 cell-size per second. This result argues in favor of a limited velocity in LSVE, so that the residence time per cell remains higher by orders of magnitude than the join and leave latency. Indeed, a participant needs to anticipate his join request by subscribing to the multicast groups which map the cells where he can go during the time corresponding to a join latency. Hence, his velocity and the cell-size impact on the number of multicast groups he needs to join by anticipation, and therefore on the IGMP traffic generated.

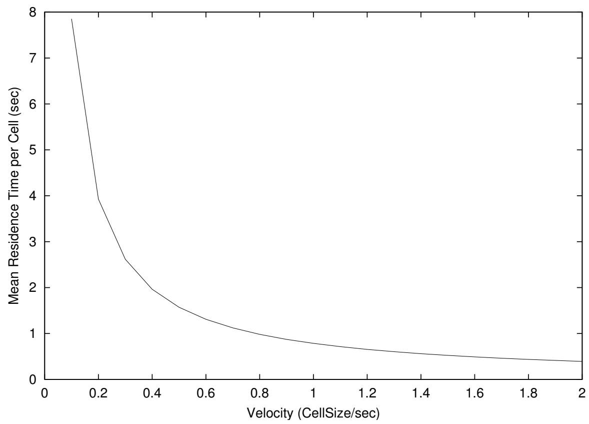

Fig. 5. Mean residence time

C. Discussion

These models are all based on the assumption that participants are distributed according to planar Poisson point processes. This assumption is primarily made for mathematical convenience. In further studies, models with more clustering such as compound Poisson point processes could also be considered. This was not done here as important properties such as the displacement theorem do not hold for such models.

IV. Description of SCORE

In Section III, we showed that the cell-size should take different values, depending on the density of participants and on the maximum traffic that participants are able to handle. This result argues in favor of a dynamic partitioning of VEs into cells of different sizes. However, our model can only be applied if we suppose that the distribution of participants in the VE follows a random homogeneous Poisson point process. In a real VE, such a global distribution is not realistic, however if we split the VE into *zones* or parts, we approximate a Poisson distribution of participants inside each zone, with different intensities within each zone. In our scheme, we take into account this model and the corresponding results to split the VE into different zones and to compute an appropriate cell-size in each zone. We implemented this scheme to show its feasibility, then performed several experimentations on our testbed in order to prove its advantages (i.e., the improvements in performance), and to evaluate its cost (i.e., control messages overhead and cost of dynamic join and leave). The goal of this scheme is to make VEs scalable with thousands of heterogeneous users on the Internet. We claim that this solution works with a limited number of available multicast groups. We believe that today, such many-to-many applications, with potentially thousands of users, require minimal management and administration support.

This section is organized as follows. First, we introduce a user satisfaction metric and present the role of the *agents* in our scheme. Then, we describe the information exchange process between participants and agents and finally we present the mapping algorithm and the handover management mechanism.

A. User satisfaction metric

An ideal situation from the end-user viewpoint can be defined as a situation where the received traffic contains no superfluous data. However, this situation is far from being realistic, considering the cost of multicasting, and therefore, the limitation in the number of available multicast groups (see Section II-A). Moreover, participants have limited network and CPU processing cycles resources. If the participant's area of interest is so large that the traffic he receives cannot be processed in real time, no mechanism could enable him to receive all the data he is interested in. Indeed, in this case, even if the subscribed cells exactly match his area of interest, the received traffic exceeds his capacity. For this purpose, we define the user satisfaction metric S as:

$$S = \frac{U_r}{\min(U_t, C)} \tag{12}$$

where U_r stands for the interesting data rate received and processed; U_t represents the data rate (received or not received), in which the user is interested (in the case of a homogeneous Poisson point process of intensity λ, this would be proportional to λr^2); and C stands for the receiver's capacity, which is the maximum data rate that the receiver can handle (limited by his network connectivity and/or processing power). When a participant receives and processes all the data he is interested in, this satisfaction metric is maximal whatever the superfluous traffic rate. Notice that for a particular user, S is also maximal when U_r is equal to C. This is true even though only a part of the data, in which the participant is interested, is received by the application. We justify the choice of this metric by the necessary tradeoff between the superfluous data rate received, the network state, and the overhead associated with dynamic group membership. Note that with this satisfaction metric, the goal of our scheme is not to adapt to the worst receiver in terms of network connectivity and processing power, but to maximize the satisfaction of the receiver with the lowest S value. This approach often referred as *max-min fairness* is described in [20].

B. Agents responsibility

Let us define *agents* as servers or processes running at different parts of the network (e.g., on a campus LAN, hosted by an ISP or by LSVE developers). Administrators of LSVE are responsible for deploying such agents on the Internet and for positioning them as close as possible to their potential users. Agents are not servers, i.e., they do not aim to process any global state for the VE, so they do not receive data traffic sent between participants. Actually, agents dynamically determine zones with the VE by considering the distribution of participants and they calculate appropriate cell-sizes according to the density of participants in each zone. Agents also have to periodically process the satisfaction of each participant according to his capacity, the size of his area of interest and the density of participants within his current zone. The computation of the participant satisfaction is done in a very simple way, using our Poisson model in the plane within each zone. Once this computation is done, agents can determine new zones (or inversely they can aggregate existing zones), and

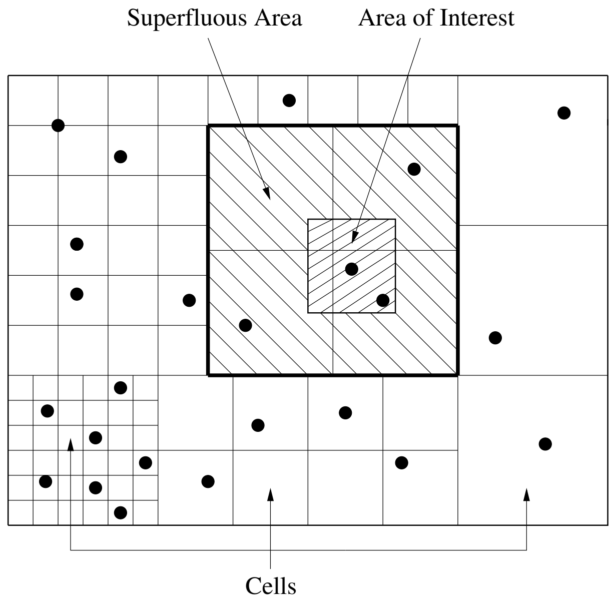

Fig. 6. Partitioning with different cell-sizes

modify the cell-sizes within the zones where the participants with the lowest satisfaction are located. Therefore our approach requires the dynamic partitioning of the VE into cells of different sizes, and the association of these cells with multicast groups. Agents have to dynamically determine appropriate cell-size values in order to maximize users' satisfaction. During the session, four successive operations are required:

- Partition the VE into several *zones*, according to the distribution of users, the users satisfactions, and the VE structure (e.g., rooms, walls, etc.).
- Compute the appropriate cell-size for each zone, according to the parameters listed in Section II-B (see Figure 6).
- Divide each zone into cells, according to his computed cell-size, and assign a multicast group address to each cell of each zone.
- Inform the participants of which multicast groups they need to join in order to interact with participants located around them.

In the rest of the paper, we refer to the first three operations as the *mapping algorithm*. We also designate the results of these operations as the *mapping information*.

C. Mapping information

In order to communicate mapping information to users, i.e., the association between cells and multicast group addresses, it is necessary to find a way to identify and name these cells within the VE. Moreover, the VE could be a structured environment with walls and rooms of different sizes. Two participants can be very close to each other but as a wall is separating them, there is no possible interaction. This specific information should be taken into account before partitioning a VE into different zones.

First, the VE is statically partitioned into several large parts that we called *start-zones* in the rest of the paper. These start-zones are actually defined according to the intrinsic structure of the VE (e.g., rooms, floor, walls, etc.) and can't ever be combined. Each start-zone is statically partitioned into indivisible *zone-units* which are the smallest unitary zones that compose the start-zone. During the session, start-zones are dynamically divided into zones which all have the same cell size. So, cells are mapping of multicast groups to a number of zone-units. As agents decide to define new zones in order to take into account changes in the distribution of participants, they identify these zones as sets of one or more contiguous zone-units belonging to the same start-zone.
To summarize, a zone is a subset of a start-zone and is composed by n contiguous zone-units $(n \geq 1)$. Within a given zone, all cells have the same size but two distinct zones could have different cell-sizes.

D. Participants-to-Agent communication

Figure 7 shows the different levels of communication in our scheme :

- Each participant subscribes to one or more multicast groups but sends data packets on a single group.
- Each participant is connected to a single agent, using a UDP unicast connection.
- Agents communicate with each other on a single multicast group: the *Agent Multicast Group* (AMG).

A participant has to subscribe to two different kinds of multicast groups:

- *data groups* associated to the cells that intersect his area of interest. Note that a participant only sends data to the multicast group associated to his current cell.
- *control groups* associated to the *start-zones* that intersect his area of interest. For these groups, a participant is only a receiver. Agents use control groups to send mapping information relative to the start zones. These informations are periodically sent for each start-zone (period = $P_{mapping}$), and contain the mapping information for all the zones belonging to the start-zone (i.e., the cell-size for each zone and the associated multicast groups addresses).

For each of these groups, the participant has to make early *join*s taking into account his speed, and the join-latency value[2]. For control groups, the $P_{mapping}$ period is also taken into account in order for the participant to receive the mapping information before his area of interest intersects new cells belonging to new start-zones.

Each participant is connected to his nearest agent using a UDP connection. We do not use a TCP connection for scalability reasons. Each time a participant enters a new zone-unit, he sends a short message to his agent. This message (20 bytes) contains his identity, his position in the zone-unit, his current size of area of interest and his capacity [21]. Therefore each agent is able to track the location of its connected users in the VE. In order to evaluate the density of participants within each zone, agents exchange information on the AMG

[2]The join-latency value can be dynamically updated during the session.

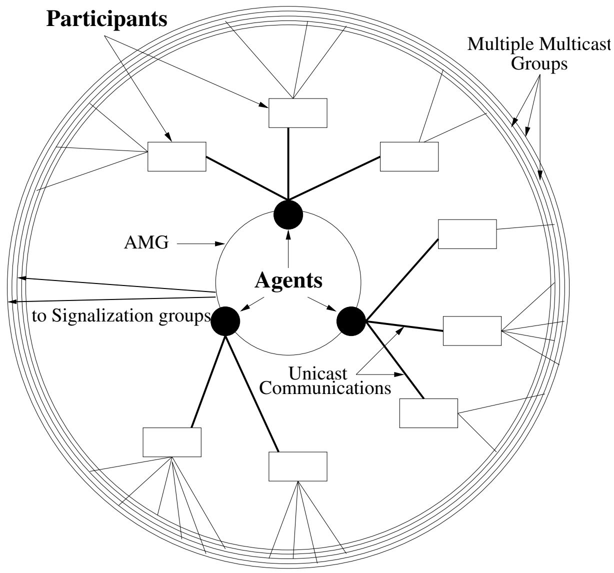

Fig. 7. Different levels of communication

multicast group. However, agents do not need to send the exact virtual position of their associated users. Only the number of users per zone-unit is necessary to allow agents to compute periodically the density of participants per zone-unit.

We have also designed a flow control mechanism between participants and agents along with a dynamic mechanism allowing each agent to know when a participant disconnects from the VE. This creates a soft state in the agent and adds reliability to the UDP transport. Participants have to send low rate *keep-alive* packets so that agents can detect a possible disconnection and have an accurate number of participants in the different zones. According to the number of participants connected, agents compute the minimal sending packet period[3] and send it back to participants. If the participant's timer expires before the participant crosses a new zone-unit, he will send a keep-alive packet including only his identity and his current position.

Connection to the Virtual Environment

We assume that before starting a session, participants have already downloaded the VE description and know the agent's multicast group address. When a new participant wants to enter the VE, he first needs to find the "closest" agent before registering and starting a login process. In our scheme, end-users discover agents by sending "Hello" packets on the agent multicast group address (they do not need to request membership to that group). This agent discovery could be done using either an incremental TTL-based mechanism or an RTT-based mechanism, depending on the distance metric we decide to choose. As soon as an agent receives a "Hello" packet from a new participant, it opens a UDP connection with it and starts the login process. Afterwards, an optional authentication process can start.

[3]This information is included within the remapping information sent by agents through the control groups.

Mapping algorithm

The same mapping algorithm is used by each agent. Agents use this algorithm to dynamically define zones in the VE, and to dynamically compute an appropriate cell-size within each zone, considering the distribution of participants and their satisfactions. However, the number of cells within a zone is inversely proportional to the cell-size for that zone. Thus, a limited number of multicast groups limits the minimal size of cells (remember that each cell is associated with a unique multicast group). Considering the entire VE, the number of cells remains always the same during the whole session. Nevertheless, the number of multicast groups assigned for each zone dynamically changes according to the evolution of the distribution of participants in the VE.

In order to allow the agents to easily compute the mapping information, we only consider square cells, with an integer number of cells per zone. Throughout the session, agents periodically compute the average density of participants per multicast group, by dividing the number of connected participants with the number of available multicast groups for the application. We refer to this density as the *remapping threshold* of the mapping algorithm. As participants arrive and move in the VE, agents keep track of the density of participants in each zone.

The mapping algorithm consists in three successive operations:

- At first, calculation is done in order to define a cell-size for each zone by only taking into account the distribution of participants in the VE. To perform this calculation, the density of participants per cell in each zone is compared to the remapping threshold.
- Then, the participants with the lowest satisfaction are identified as well as their distribution in the VE. If agents detect a concentration of unsatisfied participants within a part of a zone, this zone is divided into two new zones in order to isolate these participants. If not, the zone remains unchanged. A smallest cell-size is computed in the zones which contain the participants with the lowest satisfactions, so that they can better approximate their area of interest and therefore improve their goodput.
- The final operation is less frequently performed. During this operation, agents can decide to aggregate contiguous zones, if the cell-sizes are the same for these zones and if they belong to the same start-zone. Note that two start-zones can never be merged.

Two possible reasons can lead to the division of a zone into smaller cells:

- It is possible to find a smaller cell-size where the average density of participants per cell still exceeds the remapping threshold.
- The participants with the lowest satisfactions are located in this zone.

In the first case, agents can use the density of participants in the zone to compute a more appropriate cell-size. In the second case, agents first determine the distribution of unsatisfied participants within the zone. In order to detect a concentration of unsatisfied users in only a part of the zone, agents first compute the minimal satisfactions of participants for each zone-unit of that zone. Then they compare these satisfactions with the average satisfaction of all the zone-units of that zone. Afterwards, they can decide to split or not the zone into two new ones. Conversely, agents can decide to remap a zone using bigger cells. This remapping occurs when the density of participants per cell is smaller than the remapping threshold.

Handover management

To inform the participants on which multicast groups they need to join in order to interact with participants located around them, agents have to deal with two different situations. The first situation occurs when a participant is moving in the VE and is about to enter in an area where he does not know the mapping information. The second situation occurs when agents decide that the cell-size of a part of the VE is no longer appropriate; for example if the density of participants in this area suddenly increases. In this case, a new cell-size needs to be computed and the participants who are currently located in this area need to update their group memberships. Moreover, participants need to keep interacting between each other without suffering from this *remapping*. We call this critical operation the *handover management* [22], [23].

Here are the successive operations required to perform a handover:

- When a participant receives the new mapping information, he joins the new groups which map his area of interest. However, the participant keeps sending in the old multicast groups.
- As explained in Section IV-D, agents periodically send the mapping information in each control group. However, when agents decide to change the mapping information for a zone, they temporarily increase their sending rate for the corresponding control group.
- The participant waits for the reception of n mapping information packets before sending to the new multicast groups instead of the old ones. However, if he starts receiving data from the new groups, he immediately switches to the new groups.
- When a participant did not receive any data from the old multicast groups for a given period of time, he leaves these groups.

V. Evaluation of SCORE

In order to evaluate the performance and the overhead of SCORE, we have implemented the algorithms described in section IV and run a set of intensive experimentations [21] on PC stations: 7 PCs under Linux 2.2 connected on a 10Mb/s Ethernet. To allow experimentations with a very large number of participants, we added an option for the participants to disable the reception of data packets. When this option is used, the participant sends normally his data traffic to the multicast group associated to his current cell but only subscribes to the control groups (not to the data groups). This considerably reduces the CPU load used for the participant and enables us to run a large number of participants on a same machine.

In the following experimentations, we use 1000 participants: 996 of them with the data reception disabled are run on 3 PCs (332 participants per PC) and the 4 remaining participants that receive data traffic are run on 4 others PCs. One agent is enough to handle a set of 1000 participants: it takes only 10% of CPU of a PentiumPro 200MHz machine. To simplify participants' movements, we use a square VE without walls. The ($1200 \times 1200\ units^2$) VE is partitioned into 3×3 square start-zones with 144 available multicast groups. Each start-zone includes 2×2 square zone-units, so the size of a zone can be 1,2,3 or 4 times the size of a zone-unit. For example, in case zones are composed of 1 or 4 zone-units, the number of cells per zone takes its value in $\{1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121\}$. However, the total number of cells in the VE remains always less than or equal to the number of available multicast groups. In order to evaluate the performance of the mapping algorithm, we compare it with a static partitioning strategy dividing the VE into 12×12 squares cells of the same size. To simulate heterogeneous participants, each participant has a capacity C that is randomly selected at the beginning of the experimentation. For example, if a participant was able to handle a maximum of 20 sources, but 40 participants were located in the cells intersecting its area of interest, then only half of its incoming traffic was received and processed. The presence of variability is introduced in the VE using both the participants velocities and the notion of "hot" and "cold" start-zones: i.e., zones in which the probabilities to contain participants are respectively higher and lower than the average. At the beginning, participants are first randomly placed in the VE with a uniform distribution along x-axis and y-axis. Then the destination start-zone is randomly selected taking into account probabilities to contain participants of each start-zone [21].

Furthermore, to analyze the different experimentations, the following parameters are used:

- *Area of interest (IArea)* expressed according to the cell area in the static case (which is equal to the ratio between the VE area ($1200 \times 1200\ units^2$) and the number of available multicast groups (i.e., 144)),
- *Remapping period (RP)* standing for the period in seconds between two different remapping decided by agents,
- *Participants velocity (V)* in the VE in units per second (we have compared two cases: $V = 10$ units/s and $V = 100$ units/s given that with a static partitioning the cell area is equal to $100 \times 100\ units^2$),
- *Distribution of participants' capacity (C)*: capacities are randomly selected with a uniform distribution on either the interval $[20, 40]$ or the interval $[10, 50]$ sources/sec.

In all the experimentations, we use 1000 participants and a set of 144 available multicast groups, so the remapping threshold is equal to $\frac{1000}{144} = 6.94$.

A. Performance evaluation using the satisfaction metric

In the following set of experiments, we analyze the cumulative distribution of participants' satisfactions based on the model described in Section III. Then, we compare data traffic received per participant with and without SCORE.

1) Comparison of satisfactions static/dynamic: Figure 8 compares satisfactions obtained with a static partitioning and a dynamic partitioning. Two levels of heterogeneity are shown, with capacities uniformly distributed between either $[20, 40]$ sources/sec or $[10, 50]$ sources/sec. We have done experimentations [21] with ten different values of $IArea$ (between 1 $CellArea$ and 0.01 $CellArea$), but we only present in the paper two of them: $IArea = 0.25CellArea$ for the left figure and $0.04CellArea$ for the right figure. Whatever the level of heterogeneity between participants, the dynamic partitioning curve remains always below the static partitioning curve. For example, in the left figure, we observe that for the dynamic partitioning case, less than 5% of participants for $C \in [20, 40]$, (and less than 20% of participants for $C \in [10, 50]$) have a satisfaction value less than 0.8; whereas for the static case, between 40% and 50% of participants have a satisfaction value less than 0.8. In the right figure, minimal satisfactions are respectively 0.9 ($C \in [20, 40]$) and 0.6 ($C \in [10, 50]$) for the dynamic case and 0.55 and 0.3 for the static case. These results clearly demonstrate the scalability improvements of SCORE with respect to a static partitioning approach. However, when the area of interest is very large, performance decreases whatever the partitioning mode. On the opposite, when the area of interest is very small, participants' satisfactions tend towards 1 whatever the partitioning mode, and the SCORE mechanism becomes useless in this case.

2) Comparison of satisfactions for different distributions of capacities: Figure 9 compares mean satisfactions of 10 participants (i.e., 1% of the overall LSVE population), for two different distributions of receiver capacities. In the first distribution (called *non-uniform distribution*), their capacities are uniformly distributed in $[10, 20]$, whereas the 990 remaining participants have higher capacities uniformly distributed in $[30, 50]$. In the second distribution called *uniform distribution*, capacities of the 1000 participants are uniformly distributed in $[10, 20]$. The left figure shows the case where the area of interest is large (equal to $CellArea$). We observe that the two curves are similar and that very few participants obtain a maximal satisfaction. Indeed, when the area of interest is large, the superfluous incoming traffic could become very important. So, whatever their capacities, the participants with the lowest satisfactions are almost all located within the "hot" start-zones. Thus, for both distributions of capacities, the mapping algorithm only allocates more multicast groups within those start-zones.

When the area of interest decreases, (e.g., in the left figure with $IArea = 0.49CellArea$), more and more participants with capacities uniformly distributed in $[30, 50]$ obtain a maximal satisfaction. As soon as the cell sizes have been computed on the different zones according to the density, these participants obtain a maximal satisfaction. So, all the remaining multicast groups can be allocated to zones in which the 10 low-capacity participants are located. Note that less than 40% of satisfactions are less than 0.5 for the non-uniform case, whereas this percentage reaches 80% for a uniform distribution of receivers' capacities. This result shows the aptitude of SCORE to handle heterogeneous participants.

3) Received data traffic per participant: Figure 10 compares the mean participant's incoming data rate (in sources/sec) for a static partitioning scheme and a dynamic partitioning scheme using $RP = 1s$. Remember that this traffic is used to compute satisfactions of participants. We can observe that the gap between the 2 curves is almost constant independent of the size of the area of interest. However, relative gaps between curves differs: for $IArea = 0.16CellArea$, the incoming data traffic is 50% less in the dynamic case (20 sources/sec) than in the static case (30 sources/sec); whereas for $IArea = CellArea$, it is only 30% less (50 sources/sec vs. 65 sources/sec). This result shows that mechanisms implemented in SCORE enable participants to better approximate their areas of interest using smaller cell sizes, especially in places where the density of participants is important. Indeed, in such high density places, a small reduction of the superfluous area strongly decreases the superfluous incoming data traffic.

B. Overhead of SCORE

1) Impact of SCORE on Multicast Routing Protocols: It is realistic to assume that multicast-enabled routers can support the needs of multiple multicast groups as required by SCORE. Firstly, even if each participant subscribes to multiple multicast groups, each participant will only send data traffic to a single multicast group. Therefore, with respect to a given participant, a single (S, G) entry will be active in each multicast router. Regarding the other multicast groups where the participant behaves as a passive receiver only, the only impact might be the addition of an outgoing interface in pre-existing entries of each multicast router present in the corresponding multicast tree. Secondly, it is important to realize that, in SCORE, the fact that each participant could be a member of several multicast groups, is limited by the assumption that SCORE deals with a limited number of multicast groups. This implies that routing/forwarding tables could contain several (S_i, G) for a given group G. So depending on the underlying multicast routing protocol, these entries could also be aggregated into a single $(*, G)$ entry [24].

In the following experimentations, we evaluate the overhead of SCORE focusing on the signaling and control traffic. Then

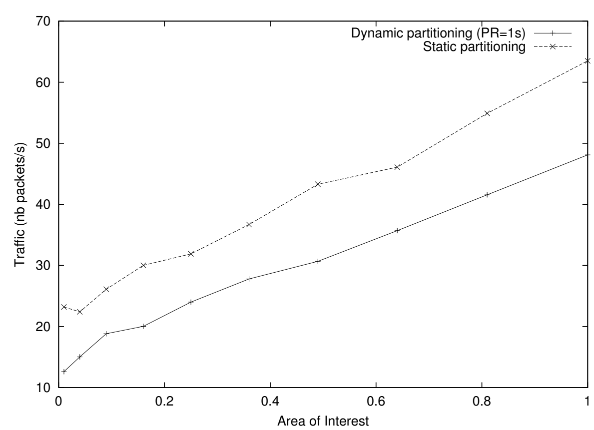

Fig. 10. Received data traffic per participant ($V = 10$ units/s)

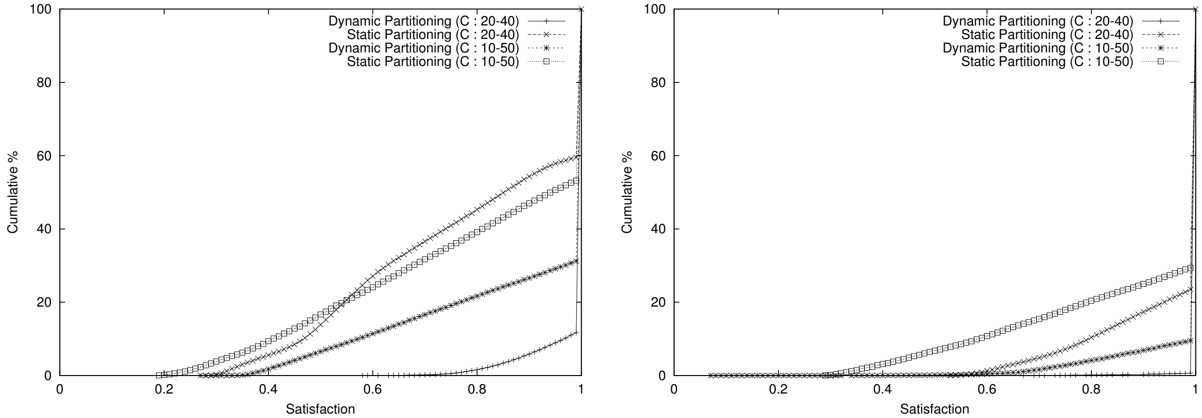

Fig. 8. Comparison of satisfactions with $V = 100$ units/s, $RP = 1s$, $IArea = 0.25$ (left) and $IArea = 0.04$ (right)

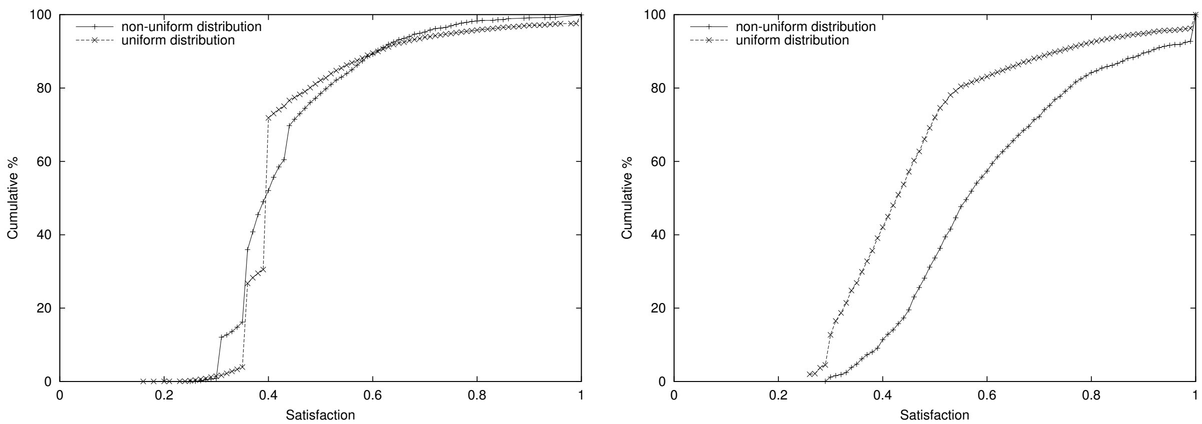

Fig. 9. Comparison of satisfactions with 2 capacity distributions with $IArea = 1$ (left) and $IArea = 0.49$ (right)

we compare the IGMP traffic generated by participant with and without SCORE.

2) Received signaling traffic per participant: Figure 11 shows the signaling traffic multicast by agents to participants and the control traffic sent by each participant to his agent according to the size of the area of interest. Note that sizes of signaling and control packets are respectively 8 and 16 bytes (plus 24 bytes of UDP/IP headers). The maximal signaling traffic is obtained for $RP = 1s$ and $IArea = CellArea$ and remains less than 1.5 packet/s. In this worst case, the right figure means that a participant subscribes in average to 1.5 multicast groups and receives a mean traffic rate of 48 bytes/s (i.e., 0.38kb/s). The left figure shows the control traffic and the “keep-alive” traffic sent by two participants to their agents with $V = 10$ units/s. The overhead is very low, less than 0.1 packet/s for the “keep-alive” traffic and about 0.05 packet/s for the control traffic. We used two different participants in order to show that the “keep-alive” traffic decreases when the control traffic increases, and conversely.

3) Network load caused by the participants: In Figure 12, we plot the number of subscriptions per second, depending on the area of interest (relative to an average size of cell), the remapping period and the velocity. To obtain the number of IGMPv2 Reports and IGMPv2 Leaves, this number should be multiplied by a factor of two. However, if several participants are located on the same LAN, the number of IGMPv2 packets sent might be reduced as a result of the IGMP-v2 Max Response Time field present in each IGMP-v2 Query packet combined with the duplicate Report suppression mechanism of IGMP-v2.

In Figure 12 left, we observe that in the case of dynamic partitioning the number of reports doubles when the area of interest increases from $0.01 * CellArea$ to $0.49 * CellArea$, and seems to stabilize for larger area. This can be explained by the fact that the area of interest is multiplied by 50 in the left part of Figure 12 left and by 2 in the right part. Thus, the number of cells intersected by the Area of interest grows also much faster in the first part. As each cell is associated with a multicast group, the evolution of the number of IGMP Reports is directly correlated with this behaviour. In this Figure, we also notice that the subscription frequency is 2 times larger where $RP = 6s$ and 4 times larger where $RP = 1s$, compared with a static partitioning strategy.

In Figure 12 right, if we compare the dynamic partitioning strategy with the static strategy where $V = 100$ units/s, we observe that the frequency of IGMP reports in the former case is twice larger than in the latter. However, even if the velocity clearly has a direct impact on the subscription frequency,

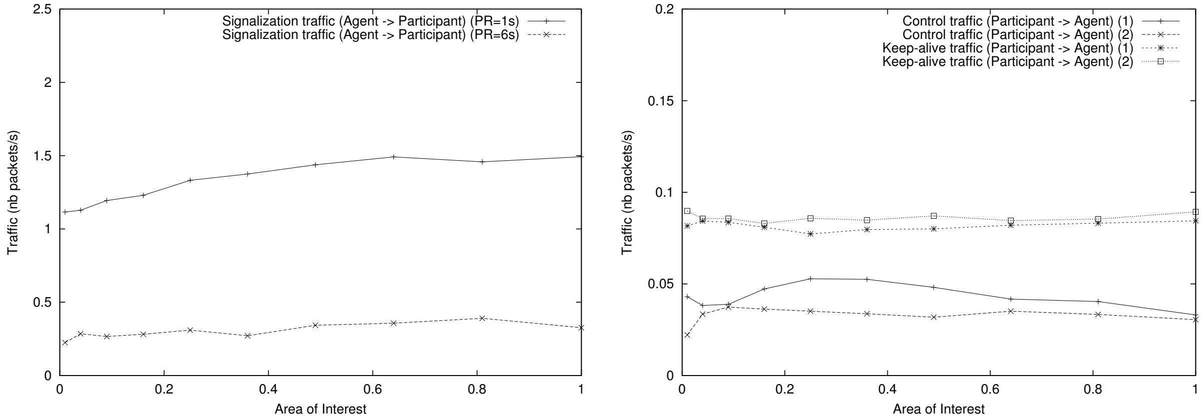

Fig. 11. Signaling traffic received and Control traffic sent (packets/s) with $V = 10$ units/s

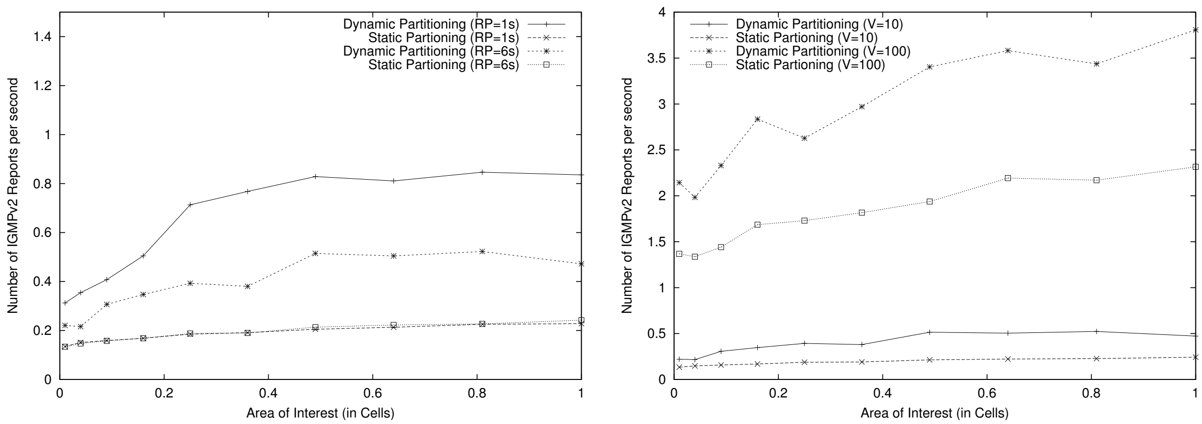

Fig. 12. Comparison of subscription frequency with $V = 10$ units/s (left) and $PR = 6s$ (right)

the same comparison for $V = 10$ units/s shows that the relative difference between the number of IGMP reports for the two partitioning strategies decreases. Indeed, SCORE allows the reduction of cell-size in the areas where the majority of participants are located (as the dynamic partitioning strategy takes into account the density of participants). With $V = 100$ units/s and $RP = 6s$, this statement is mainly true when a remapping of the virtual environment happens. Between two remappings, the distribution of participants changes more drastically compared with the case where $V = 10$ units/s. We have already seen in Figure 8, that this also has an impact on the participant satisfaction.

C. Multicast groups analysis

In the following experimentations, we analyze the use of multicast groups within the SCORE scheme.

1) Number of multicast groups subscribed per participant: Figure 13 shows the evolution of the number of multicast groups subscribed per participant during a session with $RP = 6s$ and $V = 10$ units/s. We have used 10 different values of area of interest in our experimentations [21] but have only plotted in the paper the curves corresponding to $IArea = 0.81CellArea$ and $IArea = 0.36CellArea$. We observe that when $IArea \leq 0.49CellArea$, the number of subscribed multicast groups evolves in the same interval $[1, 4]$ both for the static and dynamic cases (see the right figure). There are several reasons. First, when the area of interest is small, the number of intersected cells is small. Second, since the area of interest is small, agents do not need to compute smaller cell sizes because participants' satisfactions are already maximal. In this case, the number of groups per zone is not increased by a remapping phase and the number of subscribed groups per participant remains low. However, when the area of interest is larger (e.g., $IArea = 0.81CellArea$ in the left figure), more and more cells are intersected by the area of interest. So, the incoming data traffic increases and agents have to remap the "hottest" zones that include unsatisfied participants. This explains the higher number of subscribed groups per participant in the dynamic case.

2) Distribution of participants within multicast groups: Figure 14 shows the distribution of participants within multicast groups when $IArea = 0.04$. First, we can observe a peak around $N = 7$ participants. This peak corresponds to the remapping threshold value (i.e., 6.64, see Section V). This clearly demonstrates that SCORE can adapt to non-uniform and dynamic distributions of participants. On the contrary, the static case leads to a waste of filtering resources: 30% of multicast groups do not contain any participants, and almost

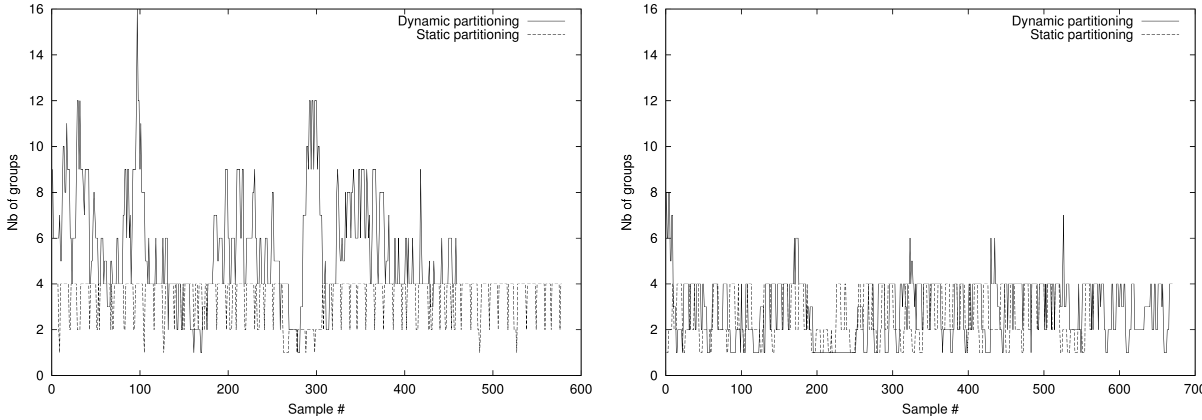

Fig. 13. Subscriptions per participant with $V = 10$ units/s, $RP = 6s$, $IArea = 0.81$ (left) and $IArea = 0.36$ (right)

half of multicast groups contain less than 3 participants. It is interesting to note that the percentage of multicast groups that contain a large number of participants is higher in the static case than in the dynamic case.

VI. Related Work

There has been a lot of published work on the issue of evaluating grouping strategies for LSVE, but few of them consider network aspects. [25] analyses the performance of a grid-based relevance filtering algorithm that estimates the cell-size value which minimises both the network traffic and the use of scarce multicast resources. However, the paper shows specific simulations done using different granularity of grids for several types of DIS entities, but the generic case is not studied. [26] compares the cost of cell-based and entity-based grouping strategies using both static and dynamic models but the paper does not propose any solution to calculate the cell-size value.

Several architectures such as NPSNET [27], DIVE [28], MASSIVE-2 [29] and SPLINE [30] have already been designed using multiple multicast groups. In NPSNET, the world is partitioned into hexagonal cells which are associated with multicast groups. In the DIVE architecture, the objects in the virtual world are hierarchically composed and associated with a set of hierarchical multicast groups. MASSIVE-2 is a collaborative virtual environment in which the spatial structure is mapped onto a hierarchy of multicast groups. SPLINE [30] is a multi-server architecture that splits the virtual world into several zones (or *locales*) in which multicast transmission is used. [31] also suggests an *octree*-based approach for interest management using multicast groups. The department of Defense has been pursuing its own architecture, called HLA [32], for virtual environment interoperability which has been recently adopted by the IEEE. HLA filtering mechanisms are based on DIS experience with multicast and use the concept of *routing spaces*. A routing space is made of *subscription* regions corresponding to member's expression of interest and *update* regions that express what a member is able to produce; regions are rectangle areas in the routing space. However, none of these different works have presented an architecture to dynamically partition the VE into multicast groups, taking into account the density of participants per cell and the participants' capacities.

VII. Conclusion and future work

We have described SCORE, a multicast-based communication protocol that enables LSVE applications to run on the Internet today. The intensive experimentations done using the SCORE implementation show that this protocol significantly improves scalability of such applications without adding critical overhead. Moreover, the scheme is flexible enough to benefit from new functionalities like the support for source filtering in IGMPv3[33]. However, we have shown that in some particular cases, a static partitioning scheme is sufficient. This situation occurs when the available number of multicast groups is large enough or when participants have high link bandwidth and processing resources available.

Directions for future work include the extension of the communication protocol to multi-flow sources, the detailed impacts of SCORE on multicast routing protocols, and the experimentation of this communication protocol with a real LSVE application on the Internet. We are currently integrating SCORE into the V-Eye application [34].

Acknowledgments

The authors would like to thank the anonymous reviewers for helpful suggestions.

References

[1] J. M. Pullen, M. Myjak, and C. Bouwens, "Limitations of internet protocol suite for distributed simulation in the large multicast environment," *RFC 2502*, February 1999.

[2] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," in *Proceedings ACM SIGCOMM*, Stanford, August 1996.

[3] E. Léty, L. Gautier, and C. Diot, "Mimaze, a 3d multi-player game on the internet," in *Proceedings 4th International Conference on VSMM (Virtual Systems and MultiMedia)*, Gifu, Japan, November 1998.

[4] 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," in *Proceedings IEEE INFOCOM*, 2000.

[5] B. N. Levine and J. J. Garcia-Luna-Aceves, "Improving internet multicast with routing labels," in *Proceedings ICNP*, Atlanta, GA, 1997.

[6] M. Oliveira, J. Crowcroft, and C. Diot, "Router level filtering for receiver interest delivery," in *Proceedings NGC*, November 2000.

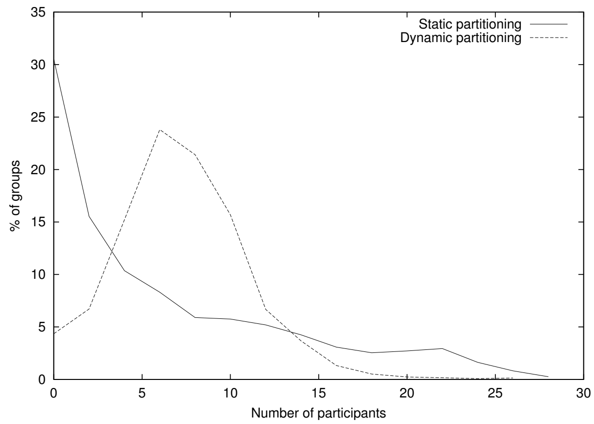

Fig. 14. Distribution of participants within multicast groups with $V = 10$ units/s, $RP = 1s$ and $IArea = 0.04$

[7] M. Sola, M. Ohta, and T. Maeno, "Scalability of internet multicast protocols," in *Proceedings of INET*, 1998.

[8] S. Kumary, P. Radoslavov, D. Thaler, C. Alaettinoglu, D. Estrinz, and M. Handley, "The masc/bgmp architecture for inter-domain multicast routing," in *Proceedings ACM SIGCOMM*, September 1998.

[9] S. Pejhan, A. Eleftheriadis, and D.Anastassiou, "Distributed multicast address management in the global internet," *IEEE Journal on Selected Areas in Communications*, pp. 1445–1456, October 1995.

[10] T. Ballardie, R. Perlman, C-Y Lee, and J. Crowcroft, "Simple scalable internet multicast," *UCL Research Note RN/99/21*, April 1999.

[11] H. W. Holbrook and D. R. Cheriton, "Express multicast making multicast economically viable," in *Proceedings ACM SIGCOMM*, September 1999.

[12] W. Fenner, "Internet group management protocol, version 2," *RFC-2236*, November 1997.

[13] D. Waitzman, C. Partridge, and S. Deering, "Distance vector multicast routing protocol," *RFC-1075*, November 1988.

[14] L. Rizzo, "Fast group management in igmp," in *Proceedings 4th Hipparch Workshop*, June 1998.

[15] Daniel J. Van Hook, Steven J. Rak, , and James O. Calvin, "Approaches to relevance filtering," in *Proceedings 11th DIS Workshop*, September 1994.

[16] F. Baccelli and S. Zuyev, "Poisson-voronoi spanning trees with applications to the optimization of communication networks," *in Proceedings of Operations Research*, 1999.

[17] J. Kingman, *Poisson processes*, Oxford Studies in Probability, No3. Clarendon Press, 1993.

[18] F. Baccelli and P. Brémaud, *Elements of Queueing Theory*, Springer Verlag, 1994.

[19] W. Feller, *An Introduction to Probability Theory and Its Applications*, vol. 2, Chapman and Hall, Limited, 1971.

[20] D. Bertsekas and R. Gallager, *Data Networks*, chapter 6, pp. 524–529, Prentice-Hall, 1987.

[21] E. Léty, *Une architecture de communication pour environnements virtuels distribués à grande échelle sur l'Internet*, Ph.D. thesis, Université de Nice Sophia Antipolis, décembre 2000.

[22] E. Léty and T. Turletti, "Issues in designing a communication architecture for large-scale virtual environments," in *Proceedings the 1st International Workshop on Networked Group Communication*, Pisa, Italy, November 1999.

[23] E. Léty, T. Turletti, and F. Baccelli, "Cell-based multicast grouping in large-scale virtual environments," Tech. Rep. RR-3729, INRIA, July 1999.

[24] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma, and L. WeiP. Tsuchiya, "Protocol independent multicast-sparse mode (pim-sm): Protocol specification," *RFC-2362*, June 1998.

[25] S. J. Rak and D. J. Van Hook, "Evaluation of grid-based relevance filtering for multicast group assignment," in *Proceedings 14th DIS workshop*, 1996.

[26] L. Zou, M. H. Ammar, and C. Diot, "An evaluation of grouping techniques for state dissemination in networked multi-user games," in *ICNP*, Toronto, Canada, 1999.

[27] M. R. Macedonia, M. J. Zyda, D. R. Pratt, P. T. Barham, and S. Zeswitz, "Npsnet : A network software architecture for large scale virtual environments," *MIT Presence 3(4)*, 1994.

[28] C. Carlsson and O. Hagsand, "Dive - a multi user virtual reality system," in *Proceedings IEEE VRAIS*, Seattle, Washington, September 1993.

[29] C. Greenhalgh, "Dynamic, embodied multicast groups in massive-2," Tech. Rep. NOTTCS-TR-96-8 1, University of Nottingham, 1996.

[30] J. W. Barrus, R. C. Waters, and D. B. Anderson, "Locales: Supporting large multiuser virtual environments," *IEEE Computer Graphics and Applications*, pp. 16(6):50–57, November 1996.

[31] H. Abrams, K. Watsen, and M. Zida, "Three-tiered interest management for large-scale virtual environments," in *Proceedings VRST*, Taipei, Taiwan, 1998.

[32] J. Dahman, J. R. Weatherly, and F. Kuhl, *Creating Computer Simulation Systems: An Introduction to The High Level Architecture*, Prenctice Hall, 1999.

[33] B. Cain, S. Deering, B. Fenner, I. Kouvelas, and A. Thyagarajan, "Internet group management protocol, version 3," *RFC-3376*, October 2002.

[34] A. Gourdon, "V-eye: A virtual eye lsve application," http://www-sop.inria.fr/planete/software/V-Eye/, November 2002.