Dynamic Clustering in Delaunay-Based P2P Networked Virtual Environments

Matteo Varvello
Eurecom-Thomson
France
matteo.varvello@eurecom.fr

Ernst Biersack
Eurecom
Sophia-Antipolis, France
ernst.biersack@eurecom.fr

Christophe Diot
Thomson
Paris, France
christophe.diot@thomson.net

1. INTRODUCTION

A Networked Virtual Environment (NVE) is a distributed virtual environment built with Internetworked Virtual Reality technology. NVEs were introduced in the 80s for military simulators; they were rapidly applied to games, i.e. World of Warcraft (WoW) [8]. Nowadays the leading application for NVEs is Second Life. SL is a complex on-line society within a 3D virtual world [7].

The state of the art for NVE based system is a client-server architecture. Recently, p2p architectures have been proposed to solve the scalability issue of NVEs [3, 4, 1].

The feasibility of p2p NVEs resides in the locality of interest and content [6]. For a given user an Area-Of-Interest (AOI) is defined; state updates are filtered according to the AOI radius.

The hard networking problem that arises is how to construct a flexible "while stable" overlay that maps the concept of neighborhood in the NVE. The Delaunay/Voronoi network seems to be an appealing solution [3].

A Delaunay network in 2-dimensions (2D) is a network of computers constructing a Delaunay triangulation on a generic plane. Avatars coordinates in the NVE are used to construct the triangulation.

We define the maintenance cost C_m as the number of messages each peer exchanges in the Delaunay network to maintain a valid Delaunay triangulation. In a dynamic environment C_m increases with player density and velocity. Since in NVEs the popularity of different areas follows a power-law distribution [1], the peers of players concentrated around some attractive points may see a considerable volume of maintenance traffic.

To address this issue, we propose a dynamic clustering algorithm: each peer in the network monitors his cost of maintenance and triggers the creation of a cluster as soon as the volume of traffic generated exceeds a given threshold. Members of a cluster then *expand* their coordinates in order to reduce the effect of the concentration of peers. Simulation results show that our solution is very effective to limit the amount of maintenance traffic below a chosen threshold.

2. A SCALABLE CLUSTERING FOR DELAUNAY P2P NVES

In this Section we present a scalable clustering for Delaunay p2p NVEs. Users are organized via a p2p network based on the Delaunay triangulation. We give here the definition of a Delaunay triangulation:

DEFINITION 1. *The Delaunay Triangulation $T(n)$ of n points in two dimensions is a triangulation of points where no point d lies inside the circumference C_{abc} circumscribed about any triangle T_{abc}.*

We call a "peer" the end-user participating in the NVE. The virtual representation of a peer is called an "avatar". Avatars coordinates are used to define a "node" in the Delaunay triangulation.

We consider a finite population of peers equal to N that do not leave or join the network, i.e. there is no churn. We define B_t the threshold to trigger the creation of a cluster as soon as the volume of traffic for the maintenance of the triangulation grows too much.

When avatars move, angular relationships among their nodes vary; as a consequence links can be activate or deactivate to maintain a valid triangulation. We call this event a "flip operation" [2]. Intuitively the rate of flip operations increases with player density and velocity and affects the maintenance cost of a Delaunay network.

In order to detect high density zones in the NVE each peer in the network monitors his cost of maintenance C_m. If $peer_a$ notices $C_m \geq B_t$, implicitly his rate of flip operations is too high. $Peer_a$ uses this information to detect a possible aggregation point in the NVE and reacts by proposing his neighbors the creation of a cluster.

The neighbors of $peer_a$ check their value of C_m and decide to agree or not in the clustering process. Peers who agree in the cluster creation start an initialization phase. During this phase the area occupied by the cluster (S_c) and its center (X_c, Y_c) is computed . A peer is elected to be responsible of the cluster, we refer to him as the cluster-head.

External nodes refer to the cluster considering it as a virtual node at coordinates X_c, Y_c. The cluster-head is in charge to monitor this virtual node as a normal node in the triangulation. Figure 1 shows how the Delaunay network looks after a cluster creation.

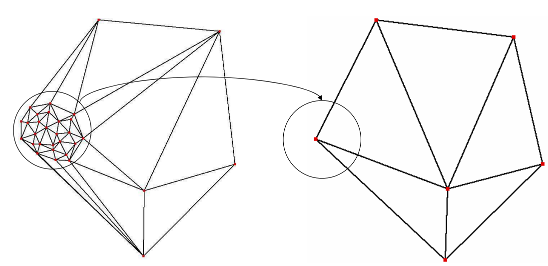

Figure 1: Example of clustering in a Delaunay network

A cluster is formed when the density in a part of the NVE is too high. To reduce this density members of a cluster expand their coordinates. The expansion is unique for all cluster members and the angular relationship among avatars stays the same, i.e. the intra-cluster Delaunay links are not modified. Indeed the density within a cluster is reduced. Reducing the local density allows to obtain a reduction on C_m for the involved peers.

Despite a cluster is represented by a point, cluster members occupy the entire area S_c of a cluster. Join and leave events are so managed considering the intersection between S_c and avatars AOI.

When the last avatar has left the cluster, it gets automatically dissolved. It results as a disconnection event of a node in the triangulation and it is handled by the cluster-head.

3. EVALUATION

We implemented a simulator in Matlab to evaluate the performance of our clustering scheme. A centralized unit maintains the Delaunay triangulation among nodes and manages the clustering process. We do not count heartbeat messages, since they are an additive cost for all peers. The packet size is set to $D = 100$ *byte*. We consider one level of clustering only. We consider a Restricted Random Waypoint [5] trip model to simulate avatars behavior.

There are N avatars in a virtual world of size X, Y. Within a world we define n_s domains of size X_s, Y_s. We use a Random Waypoint trip model [5] restricted to n_s domains for $(N - f)$ avatars and a simple Random Waypoint trip model in the entire world for the remaining f avatars. The motivation is to simulate peers interests in some specific zones of the NVE. The n_s domains act as area characterized by an high level of attraction.

We consider $N = 25$ avatars that move in a virtual world of size $X = 300$ *units*, $Y = 300$ *units*. Each avatar estimates his position 20 times per second. The simulation time is $T = 300$ *seconds* and the speed of avatars is $v_i = 20$ *units/s*. We are interested in observing interactions among avatars, this is the reason why we set high speed and sample rate and a short simulation time. All these parameters are application dependent and affect our solution only in the choose of B_t.

Figure 2 shows the per peer maintenance cost C_m of a Delaunay network for the depicted virtual world. From this analysis we get insight on how to choose B_t; in this example we set $B_t = 4$ *kbps*.

In Figure 2 it is straightforward the dependence of C_m from the local density of nodes in the triangulation. When $f = 0\%$, avatars are concentrated around the points of interest and C_m assumes large values. When $f = 100\%$ avatars are uniformly distributed in the NVE and the values of C_m result very small.

Applying our clustering scheme we see how C_m is always bounded by the target value $B_t = 4$ *kbps*. When $f = 0\%$ all peers agree in creating two clusters in correspondence of the two zones of attractions. When $f = 30\%$, the first 8 P-ids represent peers that move according to a Random Waypoint trip model in the entire world. Their distribution is uniform in the NVE and they are not interested in creating any cluster. The remaining 17 P-ids suffer for an high local density and find great benefits from the clustering process.

Finally for $f = 100\%$ all avatars are uniformly distributed in the entire NVE and C_m never exceeds the fixed threshold. In this case no peer is ever proposing to create a cluster and

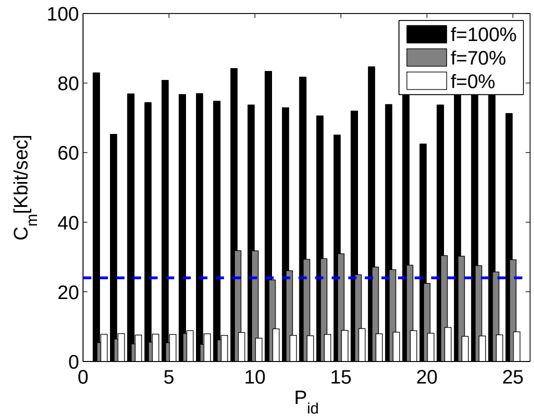

Figure 2: Per peer maintenance cost for a Delaunay network - Simulations - $N = 25$, $X = 300$ *units*, $Y = 300$ *units*, $T = 300$ *seconds*, $v_i = 20$ *units/s*

the algorithm is never executed.

4. CONCLUSION AND FUTURE WORK

This paper proposes a scalable clustering for p2p NVEs based on the Delaunay network. Our analysis shows the impact of the local density of avatars in a p2p Delaunay network for NVEs. The impossibility to limit the behavior of avatars is reflected in unpredictable maintenance cost. Using avatars in the NVE as detectors of density we propose a clustering algorithm that constructs a hierarchical Delaunay network.

The analysis proposed underlines the effect of player density and velocity in a Delaunay NVE. Simulation results show that our solution is very effective to limit the amount of maintenance traffic below a chosen threshold.

One avenue of future work is to implement a distributed version of the algorithm and to test it in a realistic environment. A complex protocol can be built on top of the proposed overlay exploiting the clustering strategy to construct an efficient data dissemination scheme.

5. REFERENCES

[1] A. Bharambe, J. Pang, and S. Seshan. Colyseus: A Distributed Architecture for Online Multiplayer Games. In *NSDI '06*, 2006.

[2] A. Bowyer. Computing Dirichlet Tessellations. *Computer journal*, pages 162–166, 1981.

[3] S.-Y. Hu, J.-F. Chen, and T.-H. Chen. VON: A Scalable Peer-to-Peer Network for Virtual Environments. *Network, IEEE*, 20(4):22–31, 2006.

[4] J. Keller and G. Simon. SOLIPSIS: A Massively Multi-Participant Virtual World. In *Int. Conf. on Parallel and Distributed Techniques and Applications*, 2003.

[5] Mobility Models. `http://ica1www.epfl.ch/RandomTrip/`.

[6] K. L. Morse. Interest Management in Large-Scale Distributed Simulations. Technical Report ICS-TR-96-27, 1996.

[7] Second Life. `http://www.secondlife.com/`.

[8] WoW. `http://www.worldofwarcraft.com/`.