A Hybrid Prediction Schemes for Networked RPG

C. Yong Jung
Korea Advanced Institute Science & Technology (KAIST)
tyonarts@kaist.ac.kr

Bryan Jung
Pylon Media
chaniyl@empal.com

ABSTRACT

A major problem of network-based multiplayer games is caused by the network transmission delay. This delay leads to inconsistency and other problems. Dead reckoning is often used to reduce the effects of network induced delays and losses by applying prediction means. The quality of the prediction and, hence, the consistency of the distributed game, depends on the difference between the real and the predicted position of some objects. In this paper we discuss prediction methods for games and study their usefulness in regard to different game types. Using implementations, the methods are evaluated experimentally. In this paper, we have described our hybrid method for predicting movement using a mouse-based input device. We have introduced the method and described how to determine all the relevant parameters. We have also discussed how we may integrate the prediction method into the sample game that we have developed. Finally, we demonstrate the effectiveness of the proposed method as compared with some of the well-known motion prediction methods.

1. INTRODUCTION

In network games, developers must synchronize the interaction caused by multiple players on-line, and the subsequent game flow. There are times, however, when network delay causes objected movements to jitter, and in case of extreme network delay, synchronization itself breaks down. This means that it takes a while until the information, e.g., about the new position of the opponents' objects, reaches the receivers. While some technical means such as network level quality of service mechanisms can reduce or at least bound this delay, some delay will always exist. Caused by the transmission delay, it takes a while until the position of the counter player reached the local player. Thus, the clients will provide their local player with incorrect information. It is important to supply for a consistent state as far as possible. Different approached have been developed such as local presentation delay and dead reckoning.

In the first approach, the processing and presentation of events from the remote and the local system are synchronized. This requires that also local events be delayed. With dead reckoning, the next event such as a position update is predicted. This can reduce the impact of delay and of loss of events also. However, the accuracy of the prediction is critical. In this paper we study hybrid prediction schemes and evaluate their suitability for games by experiments.

2. RELATED WORK

Previous work in the area of networked multiplayer games often has focused on the issue of network delay [2]. Network delays impact the players' gaming experience and make it hard to achieve a consistent system state. Different approaches have been developed to cope with the latter problem, for example, dead reckoning schemes and local presentation delay. An example of local presentation delay called bucket synchronization has been implemented in MiMaze[1]. Pantel and Wolf [5][6][7] have studied the impact of delay on the user experience and found that delays up to 100ms are acceptable for car racing. Mauve et al.

[3][4] have proposed proxy architectures for networked games, which are expected to help with congestion control, robustness, provisioning of fairness and reduction of the impact of delay. Fitzek et al. have also designed an architecture featuring two wireless technologies. Kansal and Desai discuss the problem of handover times in larger Bluetooth networks. The work of Baatz also deals with the problem of long handover times and analyses interactions with higher-layer protocols.

3. A HYBRID PREDICTION SCHEMES USING RELOCATABLE LAYOUT

Different schemes are possible for the prediction of two categories: position and future event. The methods within the first category predict the future position of a game object based on the current and the past positions. Not only straight lines but also curvatures can be modeled. In the second case, not the future position of the game object but the future position and setting of the controlling input device is predicted. Based on the used model for the behavior of the game objects, the expected user input is then used to calculate the new object positions.

In principle, a further distinction would be possible, i.e., whether the prediction is done on the sender or the receiver side. Performing the prediction at the sender can have the advantages that more information is available and that the computation effort is needed only once.

Work on dead reckoning schemes has been performed by several other groups already before. This method is effective when the trace of object movements resembles a continuous curve, as in racing games, but it is not effective when objects change the direction of their movements often, as in strategy or RPG genre. In RPG, cursor is used for the movement of objects and pointing of items. Thus, if we can relocate the layout of the game screen according to specific functions, we can predict the movements of players from the directions of the cursor.

3.1. Input Prediction Schemes

Prediction Scheme 1: Prediction of the input by using the Lagrange polynom and assuming constant control device acceleration

3.2. A Hybrid Prediction Schemes: Relocatable Layout

Prediction Scheme 2: A hybrid prediction, based on Prediction Scheme 1, using Relocatable Layout

Figure 1 is the RPG screen used in the experiment, and table 1 is the description of each number. The main character is about to be attacked by opponents. This example shows how one can predict the movements of players from the cursor movements. The central cursor represents the current position and the white cursors represent the expected position.

Figure 1: Layout of sample RPG

Number	Description
1	Main character
2	Opponents
3	Gauge
4	Inventory window
5	Command button

Table 1: Description of numbers on Figure 1

Cursor movement to the area where there is no opponent can be interpreted as unwillingness to participate in the battle. Likewise, movement to the enemy-infested area can be predicted as willingness to participate in the battle.

Figure 2: Inventory window is located on left

If the cursor is positioning rapidly to the inventory menu, we can predict that the player is about to use one of the items in the inventory. The client does not need to transmit the result to the server, leading to the reduction of overall delay. If the player selects a certain item in the inventory, it does affect the entire game, so at that moment, the information is transmitted to the server. If the inventory is positioned where there are a lot of opponents, it can cause confusion in prediction.

Figure 3: Relocated layout (Inventory window is allocated on right side, map window is allocated on the other side)

Thus, the screen layout needs to be changed. In other words, the inventory and status windows need to exchange positions.

3.3. A Hybrid Prediction Schemes: No prediction

A difficulty in raising the accuracy of prediction is the behavior of beginners. Beginners tend to show certain kinds of behavior that are hard to expect. On the other hand, skilled players do a set pattern of preparation movements, leading to easy prediction. Also, beginners usually take a while to get used to window position-changing situations. Thus, assuming that the player is a beginner, if the hit ratio within a certain period is low, no prediction scheme is effective.

4. EVALUATION SCENARIOS

The prediction schemes have been evaluated using four different situation scenarios. For this purpose, according our RPG and the prediction schemes have been implemented. The situation scenarios are:

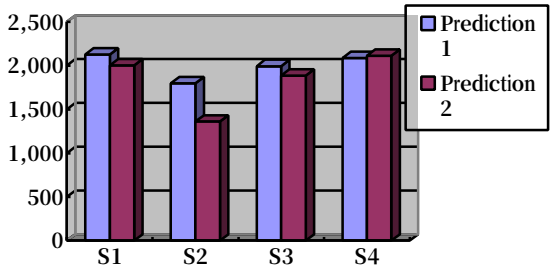
S1: situation of the beginning; the player has few items and seldom attack the opponents.
S2: second situation; the player attacks the opponents mainly.
S3: third situation; the player chats with the opponents or exchanges items mainly.
S4: fourth situation; the player sneak away mainly.

For the evaluation of the described presentation delay scheme, we implemented it in a sample RPG. This game is based on commercial networked RPG, which has been developed by authors.

4.1. Implementation

A replay function has been implemented for the sample RPG. This allows to record and replay user input, control device operation and game object positions. The quality of a prediction scheme is defined as the difference between the positions calculated by the main module and the prediction within 10 minutes. To get a general overview about the quality of a prediction scheme we determine the average deviation between the prediction and the real position of the object. Additionally, in order to get a better situation-dependent and more concrete idea of the deviations, we also implemented a viewer program. This program is used to visualize the trajectory a certain game object has driven by reading the recorded trace data and presenting it.

As will be discussed in more detail, our investigations show that there are measurable differences between situations in RPG. The results are shown in Figure 4.



	Situation 1	Situation 2	Situation 3	Situation 4
Prediction 1	2,126	1,798	1,993	2,085
Prediction 2	2,003	1,361	1,884	2,110

Figure 4: Results: Sum of Deviations

In general, our hybrid prediction seems to be suited well for RPG genre. Their results are typically better than those of the input event prediction schemes (Prediction Scheme 1). In situation 4, however, prediction 1 is better than prediction 2. The reason is that situation 4 and action game are much the same at the level of the trace of object movements.

5. CONCLUSIONS

The experiments show that prediction schemes can be useful for networked RPG. They cannot reduce the latency directly, but they can be used to reduce its impact. While presentations delay schemes can provide for an improved consistency of a networked game, they introduce additional delay. In this paper, we have described our hybrid method for predicting movement using a mouse-based input device. We have introduced the method and described how to determine all the relevant parameters. We have also discussed how we may integrate the prediction method into the sample game that we have developed. Finally, we demonstrate the effectiveness of the proposed method as compared with some of the well-known motion prediction methods. Our method does well under single and multiple prediction steps.

References

[1] L. Gautier and C. Diot: "Design and Evaluation of MiMaze, a Multiplayer Game on the Internet", Proc. IEEE Multimedia (ICMCS'98), Austin, TX, USA, 1998, pp. 233-236.

[2] Tristan Henderson: "Latency and User Behaviour on a Multiplayer Game Server", Proc.Third International Workshop on Networked Group Communication (NGC2001), November 7-9, 2001, UCL, London, UK.

[3] Martin Mauve: "How to Keep a Dead Man from Shooting", Proc. of the 7th International Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services (IDMS) 2000, Enschede, The Netherlands, 2000, pp. 199-204.

[4] Martin Mauve: "Consistency in Replicated Continuous Interactive Media", Proc. of the ACM Conference on Computer Supported Cooperative Work (CSCW) 2000, Philadelphia, PA, USA, 2000, pp. 181-190.

[5] Lothar Pantel: "Approaches for the Treatment of End-to-End Delay within Multiplayer Games" (in German), Studienarbeit, TU Darmstadt, 2000.

[6] Lothar Pantel, Lars Wolf: "On the Suitability of Dead Reckoning Schemes for Games", First Workshop on Network and System Support for Games (NetGames2002), April 16-17, 2002, Braunschweig, Germany.

[7] Lothar Pantel, Lars Wolf: "On the Impact of Delay on Real-Time Multiplayer Games", NOSSDAV '02, May 12-14, 2002, Miami, FL, USA.

[8] R. Steinmetz und C. Engler: "Human Perception of Media Synchronization", Technical Report 43.9310, IBM European Networking Center Heidelberg, Heidelberg, Germany, 1993.

[9] Jose Pablo Zagal, Miguel Nussbaum, Ricardo Rosas: "A Model to Support the Design of Multiplayer Games", Presence, Vol. 9, No. 5, October 2000, pp. 448-462.