

## 放送型サイバースペースにおけるアバタの移動特性を考慮した 空間モデルデータのプリフェッチ方式

永石 博憲<sup>†</sup>      小川 剛史<sup>‡</sup>      原 隆浩<sup>†</sup>      西尾 章治郎<sup>†</sup>

<sup>†</sup> 大阪大学大学院情報科学研究科マルチメディア工学専攻  
〒 565-0871 大阪府吹田市山田丘 1-5

E-mail: {nagaishi.hironori, hara, nishio}@ist.osaka-u.ac.jp,

<sup>‡</sup> 大阪大学サイバーメディアセンター情報メディア教育研究部門

〒 560-0043 大阪府豊中市待兼山町 1-32

E-mail: ogawa@ime.cmc.osaka-u.ac.jp

大規模な仮想空間システムでは、クライアント数の増加に伴うサーバへの負荷の増大が問題となる。そこで筆者らの研究グループでは、サーバにおける負荷を削減するために放送型サイバースペースを提案し、これまでに空間モデルデータのスケジューリング方式を提案してきた。本稿では、クライアントにおいて空間モデルデータを効率よく受信するために、アバタの移動特性を考慮した空間モデルデータのプリフェッチ方式を提案する。また、本稿では提案したプリフェッチ方式を評価し、結果について議論する。

### A Prefetching Method Considering Mobility of Avatars in a Broadcast-based Cyberspace

Hironori NAGAISHI<sup>†</sup>      Takefumi OGAWA<sup>‡</sup>      Takahiro HARA<sup>†</sup>      Shojiro NISHIO<sup>†</sup>

<sup>†</sup>Dept. of Multimedia Eng., Graduate School of Information Science and Technology, Osaka University  
1-5 Yamadaoka, Suita-shi, Osaka, 565-0871 Japan

E-mail: {nagaishi.hironori, hara, nishio}@ist.osaka-u.ac.jp,

<sup>‡</sup>Infomedia Education Division, Cybermedia Center, Osaka University

1-32 Machikaneyama, Toyonaka, Osaka, 560-0043, Japan

E-mail: ogawa@ime.cmc.osaka-u.ac.jp

In large-scale networked virtual environments, the excessive load to the server occurs due to an increase of clients. Then, to reduce the server load, we have proposed the broadcast-based cyberspace and a scheduling method of the spacial data. In this paper, we propose a prefetching method of the spacial data at clients that considers the mobility of avatars for improving the waiting time of the data. Moreover, we evaluate the performance of our prefetching method by simulation experiments.

## 1 はじめに

近年のネットワーク技術の発展と高性能な計算機の普及に伴い、計算機上に大規模な仮想空間を構築し、地理的に離れている多数のユーザがインターネットを介してその仮想空間を共有する仮想空間システムが登場している。仮想空間システムの例として、仮想社会の参加者同士のコミュニケーション支援や行動解析に用いられる InterSpace[3] や PAW[4], FreeWalk[5] などが挙げられる。これまでに開発された仮想空間システムの多くはクライアント・サーバ型のアーキテクチャを用いて構成されるが、サーバ側に処理が集中することから、大規模なシステムではサーバにかかる負荷が増大する。

そのためユーザ数が増加するたびにサーバを多重化して対応しなければならない。

筆者らの研究グループでは、この問題を解決するため、放送型サイバースペースを提案している [7]。放送型サイバースペースでは、サーバ側で仮想空間を分割して部分空間を作成し、部分空間のモデルデータをプッシュ型の放送型配信を用いてクライアントへ提供する。また、必要となる部分空間のモデルデータをクライアントが受信するまでの待ち時間を削減するため、アバタの移動速度を考慮した放送スケジューリング手法 [6] を提案している。

本稿では、クライアントにおける待ち時間をさらに削減するため、クライアントがバッファをもつ場合のモデルデータのプリフェッチ方式を提案する。

クライアント側では、アバタの移動先となる部分空間を構成するモデルデータが必要となるため、プリフェッチを行う場合にはアバタの移動先となる部分空間を予測する必要がある。移動先となる部分空間は、仮想空間におけるアバタの移動特性に大きく依存すると考えられるため、提案方式では、アバタの移動特性を考慮する。アバタの移動特性として、アバタの進行方向があまり変化しない場合と、頻繁に変化する場合を想定し、それぞれの場合に有効となる予測方式を適応的に切り替えることで、待ち時間を削減する。

以降、2章では、放送型サイバースペースの概要と想定環境について説明する。3章で提案方式の詳細を述べ、4章でシミュレーションによる提案方式の評価結果を示す。最後に6章で本稿のまとめを述べる。

## 2 放送型サイバースペース

放送型サイバースペースの概略を図1に示す。放送型サイバースペースは、クライアント・サーバ型のアーキテクチャを用いて構成される。クライアント・サーバ間のネットワーク帯域を分割し、一方ではオンデマンド型、他方では放送型のデータ配信方式を用いてデータを交換する。ユーザが操作するアバタの仮想空間内における位置情報や仮想空間の更新データなど、更新に要求されるリアルタイム性が高いデータは、前者の帯域を用いて交換され、仮想空間のモデルデータなど、更新に要求されるリアルタイム性が比較的低いデータは、後者の帯域を用いて送信される。

サーバは、仮想空間をいくつかの部分空間に分割し、部分空間を構成するモデルデータを放送の基本単位となるデータアイテムとして、クライアントに対して一定の周期で繰り返し放送する。放送型のデータ配信方式を用いることにより、各クライアントはサーバ側で更新された最新のモデルデータを放送から取得できるため、モデルデータの更新コストを削減できる。また、常に最新のモデルデータが放送されるため、各クライアントが保持しているモデルデータ間の整合性が保たれる。

本稿における放送型サイバースペースの想定環境を以下に示す。

- 仮想空間内を移動するアバタの総数を  $N$ 、アバタの平均速度を  $v$  とする。
- 円環状に接続された仮想空間を同じ広さの  $M$  個の部分空間に分割し、各部分空間の幅を  $W$  とする。

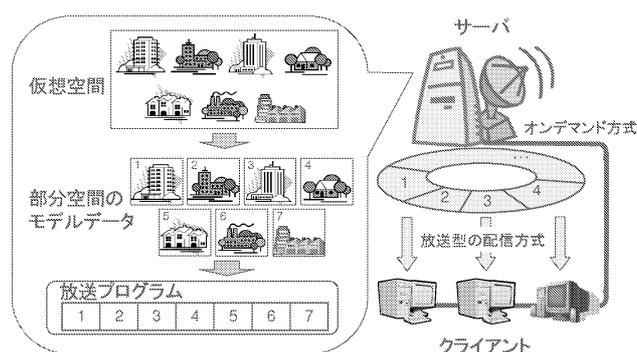


図 1: 放送型サイバースペース

- サーバは、各データアイテムを1周期の放送プログラムにおいて1度ずつ放送する。
- 各データアイテムのサイズはすべて同じとし、1つのデータアイテムの放送に必要な時間を  $L$  とする。1周期の放送に必要な時間は  $ML$  となる。
- クライアントは、データアイテムの受信が完了するまでは端末上にその部分空間を表示できない。

放送型のデータ配信を用いる場合、クライアントは必要なデータを即座に受信することはできず、必要なデータを取得する際の待ち時間を削減することが重要な課題となる。

## 3 空間モデルデータのプリフェッチ方式

本章では、提案する空間モデルデータのプリフェッチ方式について説明する。

クライアントにおいて、アバタが部分空間を移動している間に、次の移動先となる部分空間のモデルデータをバッファに受信しておくことで、待ち時間を削減することができる。

一方、クライアントのバッファ容量には限りがあるため、バッファの空き領域がなくなった場合は、バッファにあるデータアイテムのうち、アバタの次の移動先となる可能性の低い部分空間のデータアイテムをバッファから削除する必要がある。そこで、提案方式では、アバタの移動先となる可能性の高いものから順に、部分空間に順位をつける。部分空間の順位を決定する手法については、次節で詳しく説明する。以下では、部分空間の順位に基づくデータアイテムのプリフェッチ方式について説明する。

- バッファに空き領域がある場合は、バッファ内のデータアイテムに対応する部分空間の順位

に関わらず、放送されるデータアイテムを逐次受信する。

- バッファに空き領域がなく、バッファ内で最も順位の低い部分空間より順位の高い部分空間のデータアイテムが放送される場合は、最も順位の低い部分空間のデータアイテムをバッファから削除し、放送されるデータアイテムを受信する。

クライアントは、各データアイテムの放送が開始される度に、以上の方式に従ってデータアイテムのプリフェッチを行う。また、アバタが移動先の部分空間から移動元の部分空間へ引き返した場合を考慮して、移動元の部分空間のデータアイテムをバッファにキャッシュする。

### 3.1 部分空間の順位付け

それぞれの部分空間を  $S_i$  ( $i = 1, 2, \dots, M$ ) とし、以降、 $i$  の増加する方向を順方向、減少する方向を逆方向と呼ぶ。現在アバタが存在する部分空間を  $S_h$  ( $h = 1, 2, \dots, M$ ) とするとき、 $S_h$  に対する  $S_t$  ( $t = 1, 2, \dots, M$ ) の順位  $rank(h, t)$  を、以下の式によって定義する。ただし、 $t \neq h$  とする。

$$rank(h, t) = \begin{cases} pR_{h,t} & (|R_{h,t}| \leq n, pR_{h,t} > 0) \\ 2pR_{h,t} - n & (|R_{h,t}| > n, pR_{h,t} > 0) \\ -2pR_{h,t} + n - 1 & (pR_{h,t} < 0) \end{cases}$$

$R_{h,t}$  は、 $S_h$  と  $S_t$  の位置関係を表す整数である。 $|R_{h,t}|$  は  $S_h$  と  $S_t$  との距離を表し、 $S_h$  と  $S_t$  が隣接する場合は 1 となる。 $R_{h,t}$  は、 $S_h$  に対して  $S_t$  が順方向に位置する場合は正の値、逆方向に位置する場合は負の値をとる。 $p$  は、アバタが次に移動すると予測される部分空間の  $S_h$  に対する位置関係を表し、順方向の場合は 1、逆方向の場合は  $-1$  となる。以降、 $p$  を予測方向と呼ぶ。 $p$  を決定する方式については次節で述べる。 $n$  は、予測方向にあるいくつ先までの部分空間のデータアイテムを優先してバッファに取得するかを表す自然数である。以降、 $n$  を取得距離と呼ぶ。

分割数  $M = 7$ 、 $S_h = S_3$  である場合の、 $S_3$  に対する各部分空間の位置関係  $R_{3,t}$  を図 2 に示す。配列中の数字は位置関係  $R_{3,t}$  を表している。また、 $p$  と  $n$  の値に対する各部分空間  $S_t$  の順位の例を図 3 に示す。配列中の数字は、与えられた  $p$  および  $n$  の値によって決定される順位  $rank(3, t)$  を表している。 $p = 1$  の場合、 $S_h$  に対して順方向側にある隣接部分空間の順位が最も高く、 $|R_{h,t}| \leq n$  を満たす部分空間は、 $S_h$  に近いものから順に  $n$  個先までの部分空間の順位が  $1, 2, \dots, n$  となる。 $p = -1$  の

	S1	S2	S3	S4	S5	S6	S7
$R_{3,t}$	-2	-1	0	1	2	3	-3

図 2: 部分空間の位置関係

	S1	S2	S3	S4	S5	S6	S7
$p = 1, n = 1$	4	2	0	1	3	5	6
$p = 1, n = 2$	5	3	0	1	2	4	6
$p = -1, n = 1$	3	1	0	2	4	6	5

図 3: 部分空間の順位

場合は、 $S_h$  に対して逆方向側にある部分空間の順位が同様に定義される。これ以外の部分空間の順位は、 $|R_{h,t}|$  の値が大きくなるにしたがって下がる。

### 3.2 アバタの移動先部分空間の予測

想定環境では、アバタが次に移動する可能性のある部分空間は現在アバタが存在する部分空間の隣接空間に限られる。以下では、アバタの移動特性を考慮して、予測方向  $p$  を決定するための方式について説明する。

#### DR(Direction) 方式

アバタが現在の進行方向を維持して今後も移動を継続するものと予測し、アバタの現在の進行方向を  $p$  とする。したがって、アバタの現在の進行方向が順方向の場合は  $p = 1$ 、逆方向の場合は  $p = -1$  となる。DR 方式は、アバタが特定の目的地に向かって移動しているなど、進行方向の変化がほとんど起こらない場合に有効であると考えられる。

#### PO(Position) 方式

アバタが現在の位置から最も近い部分空間に移動すると予測する。例えば、現在のアバタの位置が順方向側にある隣接部分空間に近い場合は  $p = 1$ 、逆方向側にある隣接部分空間に近い場合は  $p = -1$  となる。PO 方式は、アバタが付近を歩き回っているなど、アバタの進行方向が頻繁に変化する場合に有効であると考えられる。

プリフェッチ方式では、アバタの移動特性に応じて DR 方式と PO 方式を適応的に切り替える。

## 4 性能評価

提案方式の性能を評価するためにシミュレーションを行った。以下ではその結果を示し、結果に対する考察を加える。

## 4.1 アバタの移動モデル

シミュレーションでは、アバタの移動モデルとして、ランダムウェイポイントとランダムウォークを組み合わせたものを用いる。

- ランダムウェイポイント  
アバタは、ある範囲内でランダムに選択された移動速度で、ランダムに決定した目的地に向かって最短経路で直進する。目的地に到達したアバタは、一定時間その場所に待機したのち、新たに移動速度と目的地を決定して移動を再開する。
- ランダムウォーク  
アバタは、一定の距離を進むか、一定時間が経過するたびに、移動速度と進行方向をランダムに選択して移動を継続する。どちらの条件で進行方向を変更するかを最初に指定する。

本シミュレーションで用いる移動モデルでは、アバタはランダムウェイポイントに従って目的地に移動したのち、一定時間ランダムウォークに従って移動するパターンを繰り返す。こうした移動パターンは、例えば仮想空間内のショッピングモールで行動するアバタの動きなどに見られる。アバタは、目的の店に向かって最短距離で移動したのちに店内を歩き回って買い物をし、再び次の店に向かう、といった行動を繰り返す。

アバタが次に移動する部分空間の予測方式として、アバタがランダムウェイポイントに従って行動している場合は DR 方式を、ランダムウォークに従って行動している場合は PO 方式を適用する。

## 4.2 シミュレーション環境

シミュレーションで用いたパラメータを表 1 に示す。シミュレーションでは、上下と左右が接続された、一辺の長さが 2,520m の正方形の領域を仮想空間とし、仮想空間を一次元方向に 32 分割した。サーバは、分割された個々の部分空間のモデルデータを一つのデータアイテムとし、32Mbps の帯域を用いて周期的に放送する。放送プログラムは、隣接する部分空間のデータアイテム同士がほぼ半周期離れて放送されるように作成した。また、仮想空間のデータサイズを 1 GB とする。このとき、各データアイテムの放送時間  $L$  は 8 秒となる。

シミュレーションでは、 $1/3$  秒を単位時間とし、仮想空間上に 1,000 人のアバタをランダムに配置した。アバタは、ランダムウェイポイントに従って行動を開始し、目的地に到達したのち、ランダムウォークに従ってその周辺を 300 秒間歩き回るも

表 1: パラメータ

パラメータ	値
仮想空間のデータサイズ	1[GB]
仮想空間の大きさ	$2520 \times 2520[m^2]$
仮想空間の分割数 $M$	32
放送帯域	32[Mbps]
アバタの総数 $N$	1000[人]
アバタの平均移動速度 $v$	1.25[m/s]
アバタの移動速度の分散	0.3
隣接部分空間への移動回数	300[回]

のとする。アバタは、隣接する部分空間へ 300 回移動したのち、行動を終了する。このとき、単位時間ごとに、同じ正規分布に従う乱数を生成し、それをアバタの移動速度とする。本シミュレーションでは、乱数の平均を 1.25 m/s、分散を 0.3 としており、アバタが仮想空間の分割方向に対して垂直に直進する場合、一つの部分空間の通過時間は約 63 秒となる。

クライアントがバッファに保持できる部分空間のデータアイテムの最大個数を  $c$  とし、以降  $c$  をバッファサイズと呼ぶ。バッファは初期状態において空であるとする。アバタの移動先となる部分空間のデータアイテムがバッファにない場合に、アバタが現在存在する部分空間の端に到達してからそのデータアイテムの受信を完了するまでにかかる時間が待ち時間となる。

以上のような環境設定の下で、 $p$  方向にある隣接部分空間にアバタが移動した場合をヒット、それ以外をミスとし、アバタごとに予測方向のヒット回数およびミス回数を測定した。また、アバタが次の部分空間に移動する際、移動先となる部分空間のデータアイテムがバッファにあった場合をヒットとし、アバタごとにデータアイテムのヒット回数を測定した。

## 4.3 予測方向のヒット率

アバタの次の移動先となる部分空間の予測に DR 方式のみを適用した場合、PO 方式のみを適用した場合、および両方式を組み合わせて適用した場合における予測方向のヒット率を図 4 に示す。縦軸は予測方向のヒット率を表している。これは、予測方向のヒット回数を隣接部分空間への移動回数で割った値である。

本シミュレーションでは、DR 方式を適用した場合のヒット率が PO 方式に比べて高く、DR 方式と

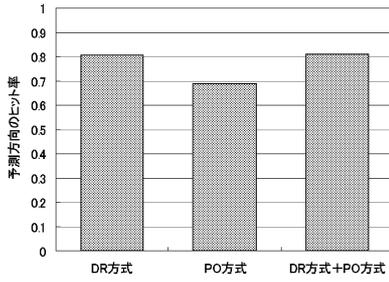


図 4: 予測方向のヒット率

PO方式を組み合わせた場合のヒット率とはほぼ等しくなっているが、各方式における予測方向のヒット率は、アバタの移動特性によって変化すると考えられる。例えば、ランダムウォークに従って周辺を歩き回る時間が長くなると、PO方式のみを適用した場合のヒット率が高くなることが予想される。

#### 4.4 平均待ち時間

$c$ の値を変化させたときの平均待ち時間を図5に示す。横軸はバッファサイズ  $c$ 、縦軸は平均待ち時間を表す。比較のために、プリフェッチを行わない場合、およびプリフェッチ方式としてFIFO(First In First Out)方式を適用した場合の平均待ち時間を示している。プリフェッチを行う場合、平均待ち時間は  $c$ の値の増加に伴って減少しているが、提案方式を適用した場合はFIFO方式に比べて平均待ち時間が小さいことが分かる。提案方式において、 $n=2$ の場合の平均待ち時間は  $n=1$ の場合よりも小さく抑えられている。これは、アバタがランダムウェイポイントに従って  $p$ 方向に移動を継続した場合に、 $p$ 方向にある2個先の部分空間のデータアイテムをプリフェッチしておくことで、アバタの全体的な待ち時間を削減できたためであると考えられる。また、 $c$ の値の増加に伴って平均待ち時間が減少しており、減少の割合は  $c \leq 3$ において最大となっている。したがって、本シミュレーション環境では、クライアントがデータアイテム3個分程度のサイズのバッファを確保できれば、提案方式によって待ち時間を大幅に削減することができる。

#### 4.5 プリフェッチ成功率

$c=10$ として取得距離  $n$ を変化させた場合の、DR方式およびPO方式におけるプリフェッチ成功率を図6に示す。横軸は取得距離  $n$ 、縦軸はプリフェッチ成功率を表す。プリフェッチ成功率は、データアイテムのヒット回数を隣接部分空間への移動回数で割り、それをすべてのアバタについて平均した値である。PO方式では、 $n=1$ の場合にプリフェッチ

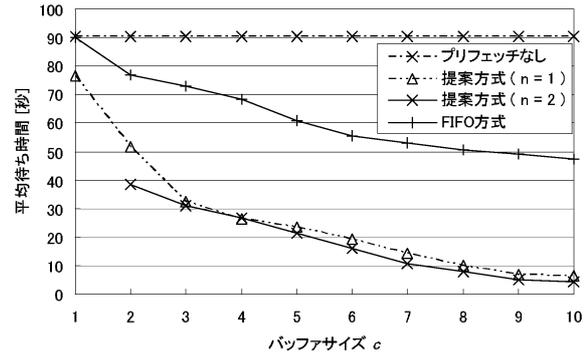


図 5:  $c$ を変化させた場合の平均待ち時間

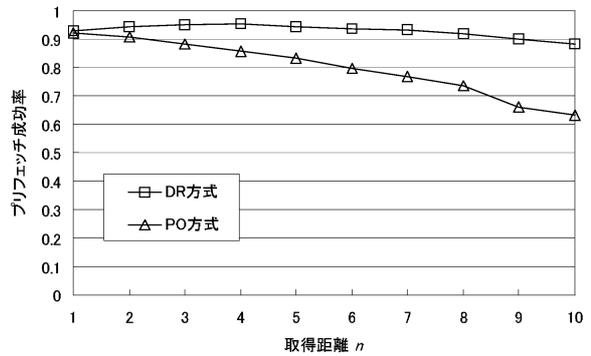


図 6:  $n$ を変化させた場合のプリフェッチ成功率 ( $c=10$ )

成功率が最も高くなっている。本シミュレーションでは、アバタがランダムウォークに従って移動する場合、連続した三つ以上の部分空間にまたがって歩き回る状況は起こらないため、 $n \geq 2$ のときはPO方式の有効性が低下するものと考えられる。DR方式では、 $n=4$ の場合にプリフェッチ成功率が最も高くなっているが、プリフェッチ成功率が最も高くなる際の  $n$ の値は、データアイテムの放送時間とアバタの移動速度に依存すると考えられる。

バッファサイズ  $c$ を変化させた場合のプリフェッチ成功率を図7に示す。横軸はバッファサイズ  $c$ 、縦軸はプリフェッチ成功率を表す。なお、それぞれの  $c$ の値におけるプリフェッチ成功率は、 $n$ の値を適切に定めることによって得られる最大値である。比較のために、FIFO方式を適用した場合を示している。グラフより、提案方式を適用した場合のプリフェッチ成功率は、 $c$ の値の増加に伴って1に漸近しており、FIFO方式に比べ、アバタが次の部分空間へ移動する際の待ち時間が0となる可能性が高いことが分かる。

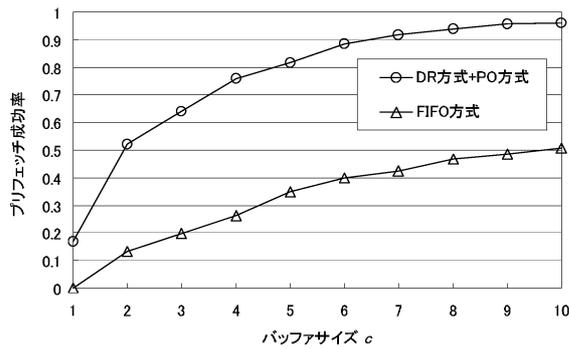


図 7:  $c$  を変化させた場合のプリフェッチ成功率

## 5 関連研究

提案方式では、アバタが次に移動する部分空間をアバタの移動特性に基づいて予測しているが、これまでに、アバタの過去の移動速度および進行方向の履歴に基づいてアバタの今後の移動特性を予測する方式が提案されている。

EWMA 方式 [2] では、アバタの過去の移動ベクトルに対して新しいものほど高い重みを付け、それらのベクトルの和をアバタの次の移動ベクトルとする。アバタの最近の移動特性が予測結果に大きく影響するため、アバタが円形状の軌跡を描いて移動する場合など、アバタの進行方向の変化が一定である場合に有効である。したがって、DR 方式への EWMA 方式の応用が可能であると考えられる。EWMA 方式は、計算にかかるオーバーヘッドを抑えられる反面、アバタの移動特性の変更が予測結果に反映されるまでに時間がかかるため、アバタの移動特性が頻繁に変化する場合には予測精度が低くなる。

アバタの操作には一般的にマウスが用いられることに注目し、ユーザのマウス操作の予測結果を仮想空間上のアバタの動きにマッピングすることでアバタの動きを予測する手法 [1] が提案されている。操作するマウスの速度と加速度との相関性によってユーザの今後のマウス操作を予測することで、アバタの移動特性が頻繁に変化する場合であっても、変化の開始時点でこれを検知できる。放送型サイバースペースでは、アバタが次の部分空間に移動する時点での移動特性を予測すればよいため、この手法の適用によって、アバタの移動先となる部分空間の予測精度が向上する可能性は低いと考えられる。

## 6 まとめ

本稿では、放送型サイバースペースにおいて、クライアントが部分空間を構成するモデルデータを

保持するためのバッファをもつ場合に、適切なモデルデータを放送からプリフェッチする方式を提案した。提案方式では、ユーザが操作するアバタの移動パターンに応じてアバタの移動先となる部分空間を予測し、その部分空間のモデルデータをバッファにプリフェッチすることで、アバタが仮想空間を移動する際の待ち時間を削減する。提案方式を適用することにより、クライアントのバッファサイズが小さい場合であっても、待ち時間を大幅に削減することができる。

今後は、アバタが仮想空間を移動する際の速度変化や、部分空間の接続構造がより複雑な場合を考慮したプリフェッチ方式について検討する必要がある。

**謝辞** 本研究の一部は、科学研究費補助金基盤研究 (B)(2)(15300033), (A)(2)(17200006), 特定領域研究 (2)(16016260), および、文部科学省 21 世紀 COE プログラム (研究拠点形成費補助金)「ネットワーク共生環境を築く情報技術の創出」の研究助成によるものである。ここに記して謝意を表す。

## 参考文献

- [1] A. Chan, R. Lau and B. Ng, "A Hybrid Motion Prediction Method for Caching and Prefetching in Distributed Virtual Environments," Proc. ACM VRST, pp. 135–142, 2001.
- [2] J. Chim, R. Lau, A. Si, H. Leong, D. To, M. Green and M. Lam, "Multi-Resolution Model Transmission in Distributed Virtual Environment," Proc. ACM VRST, pp. 25–34, 1998.
- [3] 井上 雅之, 宇佐美 潔忠, 清末 悌之, 石橋 聡, 長谷 雅彦, "3 次元仮想社会 InterSpace におけるコミュニティ形成過程とコミュニケーションメディア利用推移に関する考察," 情報処理学会論文誌, Vol. 41, No. 10, pp. 2670–2678, 2000.
- [4] 松田 晃一, 三宅 貴浩, "パーソナルエージェント指向仮想社会 PAW(第 2 版) の構築と評価," 情報処理学会論文誌, Vol. 41, No. 10, pp. 2698–2707 (2000).
- [5] Nakanishi, H., "FreeWalk: A Social Interaction Platform for Group Behavior in a Virtual Space," International Journal of Human Computer Studies, Vol. 60, No. 4, pp. 421–454, 2004.
- [6] 小川 剛史, 塚本 昌彦, 西尾 章治郎, "放送型サイバースペースにおけるアバタの移動速度を考慮した放送スケジューリング手法," 情報処理学会マルチメディア, 分散, 協調とモバイルシンポジウム (DICOMO2003), pp. 325–328, 2003.
- [7] 小川 剛史, 塚本 昌彦, 西尾 章治郎, "放送帯域幅と仮想空間内でのユーザの移動速度を考慮した放送型サイバースペースの提案," 情報処理学会研究報告 (データベースシステム研究会 2003-DBS-129, 放送コンピューティング研究グループ, 2003-BCCgr-4 合同研究発表会), pp. 49–56, 2003.