

# MMO 仮想環境における ユーザ密度の変化に対応した負荷分散手法

遠藤 伶<sup>†</sup> 高木 健士<sup>†</sup> 北望<sup>†</sup> 重野 寛<sup>†</sup>  
慶應義塾大学理工学部<sup>†</sup> 慶應義塾大学大学院理工学研究科<sup>†</sup>

大規模で参加ユーザ数の変動が激しい MMO 仮想環境のために、P2P を利用した高い拡張性を持つシステムが研究されている。そして MMO には、ユーザのグループ単位の行動による頻りに密集状態が生じるという特徴がある。そこで本稿では、そのような局所的密集状態によるメッセージ数の増加に対応できる負荷分散手法 GBLT を提案する。提案手法ではグループ内のユーザが互いに近接している事に着目し、グループに対する管理者を新たに追加する。それにより従来の管理者の負荷を低減し、密集状態に対応する。さらに、シミュレーション評価を行い密集状態に対する有用性を示す。

## Load distribution technique for high density of user in MMO Virtual Environment

Rei ENDO<sup>†</sup> Kenji TAKAGI<sup>†</sup> Nozomu KITA<sup>†</sup> Hiroshi SHIGENO<sup>†</sup>  
Faculty of Science and Technology, Keio University<sup>†</sup>  
Graduate School of Science and Technology, Keio University<sup>†</sup>

A high scalability systems using P2P are researched for MMO Virtual Environment that has large scale and change of number of user widely. A characteristic of MMO is that it occurs too high density frequently caused by the grouping work of users. Then, in this paper, we propose GBLT, the load distribution technique, that address increasing messages for high density. In the proposal, we focus on user in the same group which moves mutually close, and add new manager for the group. As a result, load of existing manager is decreased, and it addresses high density. In addition, we evaluate the proposal by the simulation and show effectivity for high density.

### 1 はじめに

仮想環境の中でも、参加ユーザ数の規模、そして商業的スケールが極めて大きなものとして Massively Multiplayer Online (MMO) という分野がある。いくつもの MMO は、常時数万人、最大では数十万人が同時に参加し [1]、その中でユーザはそれぞれが役割を持ち様々なアクティビティを行う。そして MMO では、仮想環境上で位置が近接している多数のユーザとお互いにコミュニケーションを取る事ができる。各ユーザが相互にコミュニケーションを取れる条件として、ユーザは仮想環境上で近接しているユーザからのメッセージを受け取れなければならない。その要件を満たすために、実用化されている MMO 仮想環境では、サーバ/クライアント・アーキテクチャのシステムを使用している。しかし、サーバは高コ

ストでありながら、計算能力や帯域などに限界がある。そのため、最大同時参加ユーザ数の増加に比例してサーバ台数を増やす必要があり、運営者の負担するコストが問題となる。また、MMO には時間帯によって参加するユーザ数が大きく変動するという特徴がある。よって、人口増加に対応できる高い拡張性を持った仮想環境システムが必要である。

これらの問題を解決する手段として P2P ネットワークを使用した分散型仮想環境 (Distributed Virtual Environment) が研究されている [2, 3]。P2P ネットワークでは、接続された対等な存在のコンピュータをピアと呼び、サービスをお互いに直接やりとりする事ができる。このピアとはサービスに参加する一般ユーザの PC によってもたらされる資源である。サーバが行っていた仮想環境の管理をピア間で分散する事で、サーバの必要台数を減らせる。

実際のMMOには複数のユーザがグループを組んで協力しあって行動するという特徴がある。そして、同じグループのユーザは同じ目的を持って行動するため、仮想環境上で位置が近接した状態となりやすい。その結果、局所的な密集状態が頻繁に発生する。しかし、分散仮想環境の既存研究では領域の一部にユーザが密集する状態が考慮されていない。そこで、ユーザが密集する状況下でも高い拡張性を維持できるような仮想環境が必要である。

そこで本稿では、ユーザのグループ行動が原因で発生する密集状態に対応した負荷分散手法GBLTを提案する。提案手法ではMMOに一般的にあるパーティと呼ばれる機能を利用する事でグループの成立、解散、構成メンバとその変化などの状態を容易に把握できる。パーティの成立時に、グループを管理するピアを新たに追加する事で上記の問題点を解決する。さらにシミュレーション評価を行って、提案手法の密集状態に対する有用性を示す。

## 2 関連研究

MMO仮想環境は、複数のユーザが参加可能な共有仮想環境である。共有仮想環境では仮想環境上で位置が近接しているユーザがコミュニケーションを取れる必要があり、そのためには位置が近接している隣人を発見できなくてはならない。そのような問題を隣人発見問題という。

隣人発見問題に用意に対応できるため、現在実際に運営されているMMOの大半が、サーバを使用した集中型管理手法を用いている。集中型管理手法[4]では、サーバが各ユーザの位置情報を一括して管理し、ユーザから送られてきた更新情報をそのユーザの隣人へと転送する。MMOの中には一日周期で数万から数十万の範囲で参加ユーザ数が大きく変化するようなサービスがある。そして、このようなシステムでは最大参加ユーザ数を収容できるだけのサーバ数を用意する必要がある。

しかし、予測した最大参加ユーザ数を超えた場合は仮想環境を維持できなくなり、逆に大きく下回った場合は用意したサーバが無駄なコストとなる。そのため、人口の急激な変化に対応できる拡張性を持った仮想環境管理手法として、P2Pを利用した分散型管理手法が多く研究されている。

分散型管理手法に関連する研究としてZFM[2]やPeer Clustering[3]などがあげられる。ZFMでは仮想環境を細かい領域に分割し、分割された領域をピ

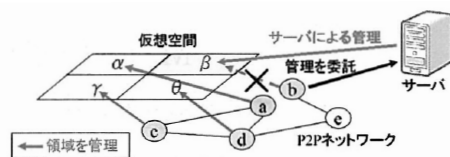


図 1: Peer Clustering

アが管理するという手法で仮想環境の分散処理を行っている。しかし、ピアはユーザが所持している一般的なPCであり、性能がサーバに比べて低い。そのため、ユーザの密集に敏感で、小規模の密集状態でもピアは領域を管理しきれなくなる。Peer Clustering[3]では、そのような密集状況下ではピアがサーバに領域の管理を委託する事でシステムを維持する。

しかし、MMOに参加するユーザの多くがグループ単位で行動を行うため、小規模の密集状態が頻繁に発生する。そのような場合では多くの領域がピアでは管理しきれなくなるので、安定したサービスを提供するには多くのサーバ台数を用意しなくてはならない。そこで、そのような密集状況下でも高い拡張性を維持できる仮想環境が必要である。

## 3 提案

本稿では、ユーザのグループ単位の行動によって発生する密集状態に着目し、MMO仮想環境におけるグループベース負荷分散手法 Group Based Load-distribution Technique (GBLT) を提案する。提案手法ではユーザグループ成立時に、そのグループを代表するピアを1つ決める。この際のグループの発生検知には、MMOのパーティを利用する。そして、グループ参加者に関してはその代表ピアが管理する事で、領域管理ピアの管理負荷を低減させる。

### 3.1 GBLT

図2にGBLTにおける管理の概要を示す。GBLTでは、仮想環境を地理的に分割し、ランダムに選ばれたピアが領域管理ピアとなって分割された領域を管理する。しかしピアの能力はサーバに比べて非常に低いため、管理しきれない場合はサーバに領域の管理を委託する。そして、グループが成立した場合は、そのグループを代表して管理するピアをLeaderピアとして決める。グループ参加者をLeaderピアが一括管理をする事で、密集状態での領域管理ピアの管理負荷を低減させる。

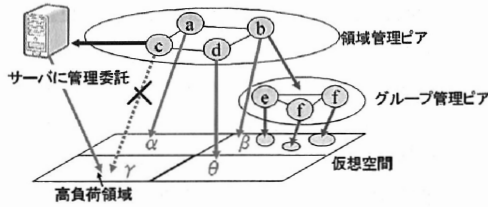


図 2: GBLT

### 3.2 パーティによるグループ成立検知

本提案ではユーザグループの成立を検知するために、MMOにおけるパーティという機能を利用する。多くのMMOではユーザのグループ行動を誘発するようなプレイシナリオが存在する。そして、そのようなMMOにはパーティと呼ばれる機能があり、ユーザはグループ行動を行う際にパーティを使用する。また、複数人でパーティ機能を使用する事を一般に「パーティを組む」と表現する。そしてユーザは、パーティを組む事によってパーティ内チャットなどの様々な追加機能を使用できる。この機能を利用して、グループの成立、解散を容易に検知できる。

### 3.3 領域管理ピアとLeaderピア

領域管理ピアは、仮想環境を構築する際に地理ベースで配置される静的な管理ピアである。領域管理ピア数はシステムによって固定であり、仮想環境上の地理的な領域は必ず1つの領域管理ピアによって管理されている。つまり、仮想環境全体を薄く広く管理するもので、既存研究では領域管理ピアのみでユーザ全体を管理している。

一方、Leaderピアは、ユーザグループベースで配置される動的な管理ピアである。MMOではグループ行動によって局所的に密集状態が発生し、一部の領域の管理負荷が増大する。そこで、グループが成立した時に、グループを初期化したピアが、Leaderピアと呼ばれる管理ピアになる。それによって、システム全体で管理に使用できるリソースを増やす事で、ユーザの密集状態に対応する。

提案手法では、領域管理ピアとLeaderピアを併用することで、管理負荷を管理ピア間で分散している。

### 3.4 領域管理ピアの負荷低減

図3に更新情報の共有手法を示す。(a)既存手法の図では、領域内で $N_1 \sim N_3$ の3ピアが仮想環境上で互い近接している。つまり図における3ピアは互いの更新情報を共有する必要がある。ピアは位置情報などを更新する際、領域管理ピアに自身の更新情

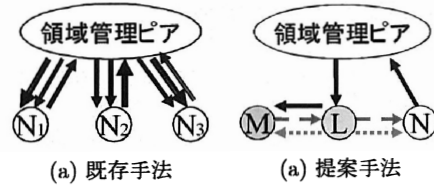


図 3: 更新情報の共有方法

報を送信する。更新情報を受信した領域管理ピアは、各ピアの位置から送信者と近接しているピアを検索し、そのピアに対して更新情報を転送している。

(b)提案手法では各ピアは、グループの代表者ならLeader、代表者以外のグループ参加者ならMember、そしてグループを組んでいないならNormalの状態を取る(それぞれ図中ではL, M, Nと表す)。Normalピアは既存手法と同じく領域管理ピアに更新情報を送信する。領域管理ピアは受信したNormalの更新情報をLeaderピアに転送し、Memberピアへは転送しない。Memberピアへは、Leaderピアが転送する事になる。Memberピアの更新情報はLeaderピアへ送信され、LeaderピアがNormalピアへと転送する事で共有される。そして、Leaderピアの更新情報はLeaderピアが直接Memberピア、Normalピアへと送信する事で共有される。提案手法では、グループを組んでいるLeaderピアとMemberピアの更新情報が領域管理ピアを経由せずに共有しているため、領域管理ピアの管理負荷を低減できる。

### 3.5 動作アルゴリズム詳細

#### a システム参加時の動作

仮想環境にピアが参加する際、ログインサーバへアクセスを行う。ログインサーバがP2Pネットワークへのブートストラッパーとして動作する。各ピアはログインサーバより、前回ログオフ時の自分の位置情報などの状態情報を取得する。自身の位置座標を取得したピアは、自身が存在する領域の識別子を取得し、P2Pネットワーク上で、その識別子に対応する領域管理ピアを探索する。そして、領域管理ピアに領域への参加を通知する。領域への参加を通知された領域管理ピアは、そのピアを自分の管理リストに追加する。管理リストは自分の管理領域にいるピアの状態が記載されたリストで、各領域管理ピアが個別に持っている。

#### b グループが存在しない場合の動作

初期状態のように領域内にグループが存在しない場合、全てのピアはNormal状態である。このよう

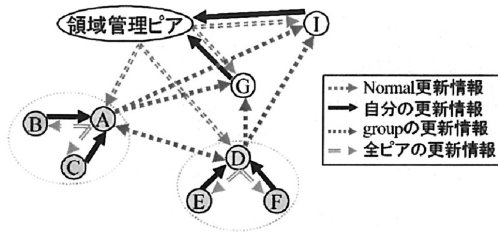


図 4: 領域内にグループがある場合の更新情報共有な場合、領域管理ピアは管理リスト内の全てのピアから更新情報を受け取り、全てのピアに仮想環境上の隣人の更新情報を転送する。

ピアが領域を移動する場合は、領域管理ピアに移動先の領域を通知し、移動先の領域管理ピアの IP アドレスを受け取る。したがって、P2P 上でのピア探索を行わないため、その遅延は発生せずに次の領域管理ピアの管理下に入る事ができる。移動先の領域管理ピアのアドレスを通知した後、元の領域管理ピアは領域を移動したピアを管理リストから削除する。

#### c グループが存在する場合の動作

まず、グループが成立する時の動作について説明する。グループはパーティが発生した際に成立する。この際、パーティに勧誘した側のピアが Leader、勧誘された側が Member の状態に遷移する。Leader ピアは領域管理ピアにグループ成立の通知をする。通知を受け取った領域管理ピアは Leader ピアに領域内のピアの IP アドレスを渡す。以後、領域管理ピアは Member ピアが自分の管理領域に存在する事と、Member ピアのアドレスは引き続き管理リストに保持するが、Member ピアと直接の情報交換は行わなくなる。状態情報の一部を保持するのは、Leader ピアにトラブルが発生した場合、Member ピアと再び通信するためである。

図 4 に、グループが存在する場合の領域内での更新情報の共有の様子を示す。領域管理ピアと Leader ピアが以下のように動作する。

- 領域管理ピア
 

領域管理ピアは管理リスト内の全 Normal ピアから更新情報を受け取り、全 Normal ピアに近接している Normal ピアの更新情報を転送する。そして、Leader ピアには全 Normal ピアの更新情報を転送する。
- Leader ピア
 

Leader ピアは以下に挙げる更新情報を受け取る。

- 領域管理ピアからの全 Normal ピアの更新情報
- 他 Leader ピアからの各グループ参加者 (Leader 自身とその Member ピア) の更新情報
- 管理 Member ピアからの更新情報

この時点で Leader ピアは領域内の全ピアの更新情報を把握する事になる。

そして、以下に挙げる更新情報を転送する。

- 他 Leader ピアへの、自身のグループ参加者の更新情報
- 管理 Member ピアへの、近接しているピアの更新情報
- グループに近接している Normal ピアへの、自分のグループ参加者の更新情報

以上で、全ピアが近接ピアの更新情報を共有できる。

#### d システム離脱時の動作

MMO では、ピアはシステムからの離脱時に明示的な処理を行う。その際、Leader ピアの場合はグループ参加者の一人を次の Leader ピアとして選択して、Leader を交代する。次の Leader ピアに対して、管理に必要な情報を受け渡し、自分の状態を Normal に変更する。新しい Leader ピアは自身の状態を Leader に変更し、Member ピアへ Leader の交代を宣言してからグループの管理を開始する。

離脱ピアが領域管理ピアであった場合は、自分に最も近い識別子を持つピアに管理情報を受け渡す。

最後に、自身の領域管理ピアに離脱を通知し、ログインサーバに現在の状態を記録してから離脱する。

#### e 領域飽和時の動作

領域に参加するピアが増加しすぎて、領域管理ピアに能力以上の負荷がかかる場合について述べる。領域内のピア数が増えると、領域の管理負荷が領域管理ピアの能力を超える可能性がある。その場合には、サーバに管理を委託する事で領域を維持する。管理負荷は、ネットワーク、CPU、メモリなどの使用率から測定する事ができる。ピアの処理能力に応じてあらかじめ決めてある閾値を超えた時点で、サーバに領域の情報を送信し、領域管理を委託する。

## 4 シミュレーション評価

本説では、提案手法 GBLT の密集状態における有用性を示すために、既存手法である Peer Clustering[3] との比較評価を行う。

### 4.1 シミュレーション条件

表 1 にシミュレーションで使用したパラメータを示す。仮想環境のサイズ、最大グループサイズなどは既存研究 [3] を参考に決定したものである。ピアの処理能力は、本稿では送信するメッセージとし、上り回線 512Kbps の ADSL 回線を使用した場合を想定して決定した。MMO によって更新情報の形態は異なるが、250 メッセージというのは、実際のピアの能力としては低めの値である。

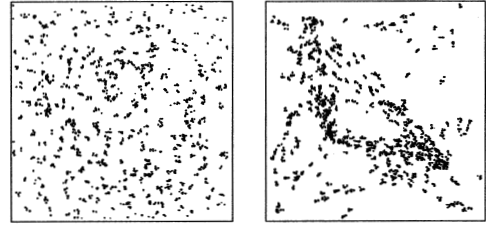
表 1: シミュレーション条件

分割した領域数	20 × 20 個
分割した領域の形状	六角格子
総ピア数	2000 人
仮想環境の横断にかかる時間	10 分
ピアの処理能力	250 メッセージ
最大グループサイズ	1~5 人
シミュレーション時間	7200 秒
シミュレーション回数	10 回
DHT	Pastry

シミュレーションでは、ユーザは仮想環境をランダムに移動しながらパーティを組む。ユーザの移動モデルとしては Random Way Point モデルと Hot Point Near モデル [5] を使用した。図 5 にそれぞれのモデルにおけるユーザ分布の一例を示す。

Random Way Point モデルでは、ユーザは一定の確率で移動角度を決定し移動する。そして一定の確率で最大グループサイズ以下のパーティを組み、パーティを組んだユーザはグループ単位で移動する。つまり、グループによる行動が原因で局所的な密集が発生する状況を再現したモデルである。

Hot Point Near モデルでは、ユーザは Hot Point への角度を中心とした正規分布に従う確率で移動角度を決定する。Hot Point とはユーザが密集しやすい仮想環境上の点であり、ランダムに複数配置される。そして Random Way Point モデルと同様に一定確率でパーティを組む。これは、グループによる局所的密集状態に加え、Hot Point という大規模な密集状態が発生する状況を再現したモデルである。



(a) Random Way Point (b) Hot Point Near

図 5: ユーザ行動モデル

Hot Point の位置の違いによる影響をなくす為、シミュレーションでは Hot Point の位置は固定した。

### 4.2 最大グループサイズの変化による影響

最大グループサイズの変化による平均サーバ管理人口の測定結果を示し、最大グループサイズがシステムの拡張性に与える影響について考察する。

#### a Random Way Point モデルの場合

図 6 に Random Way Point モデルを使用して、最大グループサイズを変化させた場合の平均サーバ管理人口について示す。図において横軸は最大グループサイズ、縦軸は平均サーバ管理人口を示している。

平均サーバ管理人口は、領域管理ピア、Leader ピアによる P2P 上の分散管理では管理しきれずに、サーバに委託されたユーザ数を示す。

既存手法では最大グループサイズが大きくなるに従って、平均サーバ管理人口も増加している。最大グループサイズの増加は集団で行動するユーザの増加を示し、従って局所的に人口密集状態ができやすくなる事を示している。それによって、密集状態となった領域は領域管理ピアでは維持できず、サーバに委託される。

一方、提案手法 GBLT では最大グループサイズが増加しても、平均サーバ管理人口は増加していな

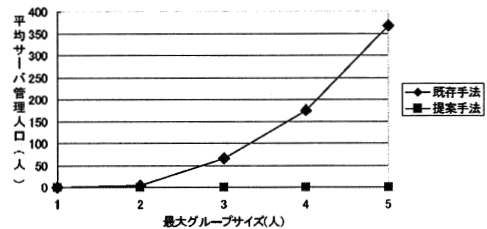


図 6: RWP モデルにおける最大グループサイズと平均サーバ管理人口の関係

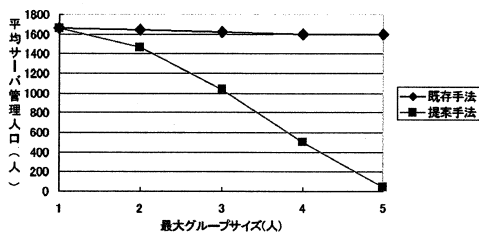


図 7: HP モデルにおける最大グループサイズと平均サーバ管理人口の関係

い、大きなグループができると密集状態ができやすくなるが、同時に Leader ピアによって管理される Member ピアの数も増える。また、グループの数が増えたとしても、それにつれて Leader ピアの数も増える。つまり、ユーザのグループ行動によって密集状態が発生しても、その分領域管理ピアの負荷は分散されるため、平均サーバ管理人口の増加を抑えることができる。

#### b Hot Point Near モデルの場合

図 7 に Hot Point Near モデルを使用して、最大グループサイズを変化させた場合の平均サーバ管理人口について示す。ここで Hot Point 数は、実際の MMO における一般的な値である 3 個とした。

既存手法は、どの場合も 1600 人以上がサーバによって管理されている。これは、Hot Point 周辺の領域は広い範囲に渡って人口が密集するため、大半の領域がピアでは管理しきれないためである。

提案手法 GBLT でも、Random Way Point モデルと比較すると平均サーバ管理人口が多い。これは、Hot Point による密集状態が、グループと直接関係が無いためである。しかし、最大グループサイズが増えるにつれ Leader ピアによる管理も増えるため、平均サーバ管理人口は減少する。最大グループサイズが 5 人の時は 50 人以下に抑えられている。

Random Way Point モデルのようにグループによって局所的な密集が発生しても、サーバ管理されるユーザ数が増加しない。また、Hot Point による大規模な密集に対しても、最大グループサイズが 5 人の時は、サーバ管理されるユーザを全体の 3% 程度に抑えられる。以上より、提案手法 GBLT はユーザが密集するような状況でも高い拡張性を維持できる。

## 5 結論

本稿では、MMO という大規模な仮想環境を想定した負荷分散管理手法 GBLT を提案した。MMO は、ユーザ数が大規模である事に加えて、ユーザのグループ行動により局所的な密集状態が頻繁に発生するという特徴がある。そのため、密集状況下でも高い拡張性を維持できる負荷分散手法が必要である。

GBLT では、MMO 独自のパーティシステムを利用してグループの成立を検知し、グループ内にグループを管理する管理ピアを新たに追加する事で密集状態に対応した。

GBLT をシミュレーションにより評価した結果、ユーザのグループ行動が原因で密集状態が発生しても、Leader ピアおよび領域管理ピア間で処理を分散する事ができ、サーバの負荷が増加しない事が示された。また、Hot Point による大規模な密集に対しても、最大グループサイズが 5 人の時は、サーバの負荷をユーザ全体の 3% 程度に抑える事が可能である。以上より GBLT の密集状態に対応した仮想環境の管理手法として有用である。

#### 謝辞

本研究の一部はグローバル COE プログラム「アクセス空間支援基盤技術の高度国際連携」により行われました。

#### 参考文献

- [1] FINAL FANTASY XI official web. <http://www.playonline.com/ff11/index.shtml>(2008/2 現在).
- [2] 斉藤賢爾, 飯村卓司. Peer-to-peer 技術. 情報処理, Vol. 46, pp. 906-909, 2005.
- [3] Alvin Chen and Richard R. Muntz. Peer clustering: A hybrid approach to distributed virtual environments. In *ACM Netgames'06*, pp. 30-31, 2006.
- [4] Bruno Van Den Bossche, Tom Verdickt, and Bart De Vleeschauwer. A platform for dynamic microcell redeployment in massively multiplayer online games. *ACM NOSSDAV '06 Newport*, pp. 14-20, 2006.
- [5] S. Rueda, P. Morillo, J.M. Orduna, and J. Duato. On the characterization of peer-to-peer distributed virtual environments. In *IEEE Virtual Reality Conference 2007*, pp. 10-14, 2007.