



ネットワークを使ったコミュニケーション

竹村治雄

奈良先端科学技術大学院大学



1. はじめに

本稿では VR システムをネットワークに接続してコミュニケーションを実施する場合のシステムの構成方法について解説する。本稿表題の「ネットワークを使ったコミュニケーション」には、それ以外の実画像を用いたコミュニケーション、電子メールや HTTP を用いたもの、2次元のデスクトップ会議システム等も考えられるがスペースの都合上、本稿では割愛させていただく。

VR システムを複数の利用者が同時に利用してコミュニケーションを図るアプリケーション（共有仮想環境 Collaborative Virtual Environment - CVE、または Networked Virtual Environment - NVE）は最近ではいろいろ構築されている[1-5]。VR システムを用いた協調作業の利点は、お互いの距離の物理的制約を乗り越えて空間を共有する形のコミュニケーションが可能に要約できる。もちろん、従来のデスクトップ会議システムも、共有エディタや共有ホワイトボードなどアプリケーションを共有して作業が可能であるが、CVE では、それぞれのユーザの存在する空間も共有可能であり、より密な協調作業が可能である。

以下、2 節では CVE における通信について、3 節では CVE のデータ共有方法について、4 節では実際の CVE システムの例を紹介する。

2. 共有仮想環境における通信

本節では、共有仮想環境に参加する端末とサーバ、あるいはサーバ同士の通信路について考察する。以下に、通信路ごとの特徴と共有環境実現のための通信路とした場合の問題点を考える。

2.1 物理ポートを用いた通信

LAN 以前ではシリアルポートやパラレルポートなどの計算機の物理ポートを介しての通信が試みられている。2

台のシステム間での通信には簡便な方法であるが、拡張性に問題がある。また、遠隔地間ではモデム同士により結合する必要があり、伝送速度が限られる。

2.2 LAN を利用した通信

ネットワークによる通信は、通信相手をネットワーク上の名前やアドレスで指定できる利点があり、物理的な接続と論理的な接続を分けて利用できる。また、プログラミングもネットワーク通信のライブラリを用いることで比較的容易に行える。通信プロトコルとしては LAN 上では TCP/IP および UDP/IP を用いるのが一般的である。ネットワークを利用することでネットワーク上の計算機を対象として仮想環境のプログラムを開発できる。プログラムは、共有データの管理手法など通信の高次の処理に専念できる。

2.3 WAN・インターネットによる通信

ここでの WAN・インターネットによる通信とは、複数の LAN がルータを介して接続されている状態での LAN 間通信のことである。最近では LAN 間接続の通信路の速度が向上し、基本的には LAN の場合と同じ感覚で利用が可能となりつつあるが、次のような留意点がある。

- ① 途中の経路に帯域の狭い場所があると全体のスループットが低下する。
- ② ルータを通過する毎にパケットの伝送遅延が起こる。
- ③ ブロードキャストパケットや特殊なパケットはルータを超えられない場合がある。
- ④ 信頼性が LAN よりは低くなる。

比較的空いているネットワークではパケットのロスはなく、UDP を用いてもデータのロスはほとんど無いが WAN ではそのような状況は期待できない。従って UDP を用いる場合はデータのロスに対する何らかの対策が必要で

ある。

2.4 データ配布の方法

共有仮想環境のデータ（共有環境に存在する仮想物体の形状など）を共有仮想環境に参加しているシステム（以下、ノード）に効率よく配布する方法について検討するには、通信のトポロジーや利用するプロトコルについてよく検討する必要がある。ここでは、簡単のために同一のデータを共有仮想環境に参加するすべてのノードに伝送する方法を例に考える。

ユニキャストによる方法

ユニキャストとは1対1の通信であり、通信路を明示的に確保してノード間のデータ交換を行う。この方法は簡便であり、ノード数がそれほど多くない場合にはよく用いられる。しかしながら、すべての参加ノードでデータを交換するためには、 N 個のノードについて $N * (N - 1)$ 本の通信路を確保し、各ノードは同一の packets を $(N - 1)$ 回再送する必要がある。パケット数の爆発が起こらないような参加者の比較的少ない場合は、ユニキャストはWANでも有効な方法である。

マルチキャストによる方法

マルチキャストとは1対多、あるいは多対多の通信を行うための手法であり、ネットワーク上の特定のグループにのみ、一斉にデータを配布する。あるノードが他の参加ノードにデータを配布するためには、1度だけマルチキャストグループに対してデータを送信すればよい。大規模な共有環境に適した方法である。また、マルチキャストを利用すれば、参加ノード全体を複数のグループに分け、現在関心のあるグループ内でのみデータをやりとりすることで大規模なデータの効率的な共有が図れる。

マルチキャストではユニキャストと異なり、通常は配布したデータの信頼性は保証できない点が問題である。一方、信頼性を保証するためにデータを受け取ったことを配布元に通知する機構やエラー時の再配布の機構を伴った信頼性のあるマルチキャスト手法も存在するが、この場合はそのような機構が低速化を招き拡張性に欠けるという問題が生じる。

ブロードキャストによる方法

ブロードキャストとは1対多、多対多の通信を行うための手法であり、マルチキャストと異なり、ネットワーク上のすべてのノードに一斉にデータを配布する。そのた

め、多数のノードで共有環境を構築するには簡便な手法であるが、データを望まないノードに対してもパケットの配信を行うために、通信トラフィックが無駄に大きくなるという問題がある。また、この時すべてのノードが自分にとって必要なデータかどうかを調べるためにOSレベルで処理が低速化するという問題もある。

3. CVE システム

ここでは、前節の通信路を用いてCVEを構築する際の実装上の問題点を中心に解説する。具体的には複数の利用者が仮想環境を共有する際のデータ管理と通信を中心に述べる。

3.1 典型的なVRシステムのループ

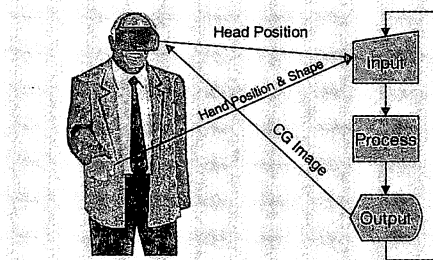


図1 典型的なVRシステムのフロー

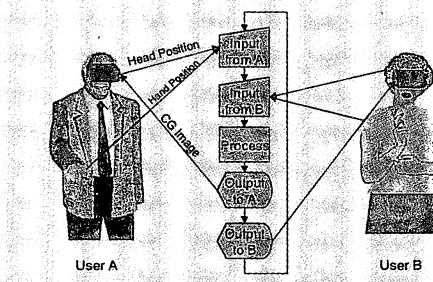


図2 一台の計算機による実装例

図1は典型的なVRシステムソフトウェアのフローを示したものである。図に示されるように、基本的にシステムはデバイスからの入力（Input）（HMDの3次元空間中の位置、データグローブなどの入力デバイス状態）を元に仮想環境の状態を更新し、出力用のデータ（仮想世界のCG画像）を生成し出力するというループを構成する。（実際のプログラミングではイベント駆動ベースでプログラミングされることが最近は多く、明示的なループは存在しないかもしれない。）通常、このループが1秒間に30～60回実行されることになる。

3.2 共有対象

図1のような仮想環境を複数構築し、これをネットワ

ークで接続してコミュニケーションを実現するにはどのような工夫が必要であろうか？ 音声だけを相互に伝送しても相手の姿は見えコミュニケーションが難しいのは容易に想像される。当たり前であるが、両者が存在する仮想環境が共有される必要がある。つぎに通信相手の仮想空間中での位置や姿勢が伝わる必要がある。さらに、利用者が環境に対して働きかける場合（たとえば仮想物体を掴むなどの操作を行う場合）、それに応じて仮想環境の状態が変化するので、これを他の利用者にも伝えなければならない。次節では、これらの点を二者間のCVEを例にあげて概観する。

3.3 2ユーザによるCVEの実現

ここでは、共有仮想環境の具体的な構成方法を2ユーザ間の通信を例にして簡単に解説する。

1台の計算機による実装

一台の計算機でCVEを実現する場合は図2に示されるような構成を用いればデータ共有に関する問題は生じない。実装も比較的簡単に行える。この例ではそれぞれのデータを入力し、仮想世界を更新し表示するループを繰り返す。同じ室内など至近距離での協調作業環境を構築するには簡便な方法であるが、拡張性に欠ける。

2台の計算機による実装

2台の計算機を用いて2者間での共有仮想環境を行う場合の一例を図3に示す。この場合、仮想環境の状態はあらかじめVWDB (Virtual World Data Base) として各ユーザに配布されている。この例ではループは独立しており、それぞれが独立して共有空間をウォークスルーすることになり、通信相手の位置や状態が表示できない。VRMLファイルをWWWからダウンロードして、それぞれのネットワーク利用者がウォークスルーしている状態と同じと考えてよい。

互いの状況を相手に表示するためには、これらのデータを双方が送り合う図4の構成を必要とする。この構成はお互いが仮想空間の状態を変更しないという仮定の成立する状態では有効に機能する。データベースの状態を変更する場合は、この方法では変更を相手に伝えていないため、何の工夫もしなければ時間とともに両者のデータベースの内容が一致しなくなる。それぞれでの更新処理をしてもデータベースの内容が一致するためには、更新処理に対する要求（Aが仮想物体をつかもうとしたなどの操作）をそれぞれのユーザの計算機が同じ順序で行うことを保証

する必要がある。この処理は直列化とよばれる。具体的には、各ユーザからの入力装置の状態をすべてそれぞれのユーザサイトですべて集め、それをあらかじめ決められた順序でデータベースの更新に反映し、画像を生成し次のループ処理を実施する。従って、両方のサイトで直列化を行うことは、両方のサイトが同期してループを回すことになる。また、パケットの喪失は許されないため、TCPプロトコルなど信頼性が確保されたプロトコルを用いる必要がある。図4のような2者間の共有仮想環境でかつ通信遅延が少ない場合にはこのような方法でも良いが、後述するような多数の参加者がある環境への適用は困難である。

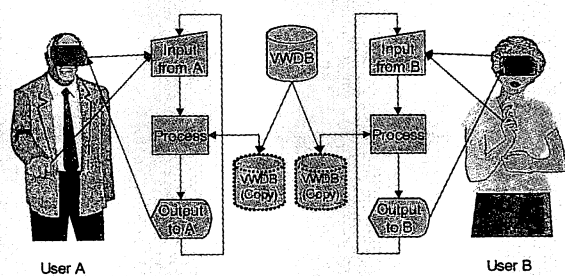


図3 複数ループによる共有仮想環境

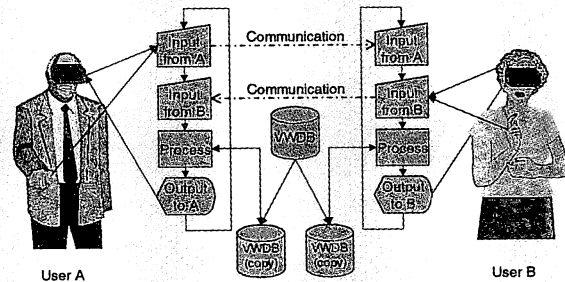


図4 複数ループによる共有仮想環境 (2)

また、この方法でサイト数が増えるとそれぞれのサイトでの処理が増大する。ひとつでも通信に時間がかかるサイトが存在すると全体のパフォーマンスが低下するなどの問題を生じる。このため、次に述べるようなサーバーを用いる方法が一般的によく用いられる。

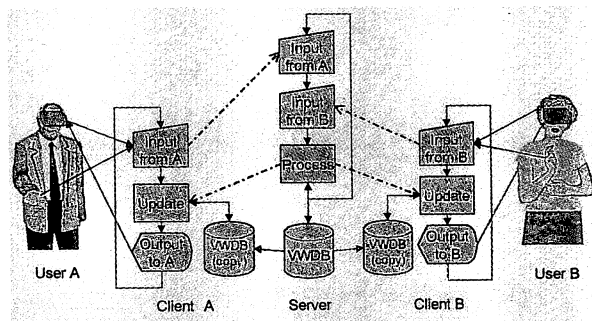


図5 サーバ・クライアント方式

サーバを用いた共有仮想空間

ローカルにデータベースを直列化した更新要求に基づき更新するのではなく、1箇所のサーバ計算機が一括してデータの更新を行う方法である。更新されたデータベースの差分情報がサーバから各計算機に送付される。従って、ネットワークの接続形態はサーバにクライアントが直接接続するスター型になる。この構成は、今日一般的に用いられる方法である。この方法は2台から十数台比較的小規模なCVEを実装するのに適している。サーバは各計算機からの入力を受け付け、それによってVWDBを更新し、変更情報と他のサイトのユーザの状態をすべての計算機に送り返す。ただし、共有環境の規模が大きくなるとサーバで処理すべき情報量が増えるため、遅くなるという欠点がある。このためVWDBをいくつかの部分に分割し、利用者の仮想環境中での位置情報をもとに接続するサーバを動的に切り替える方法が大規模なCVEでは用いられる。また、それぞれのユーザに伝達すべき情報をユーザ間の距離などをもとに選択することで通信データ量の削減したり、データの重要度に応じてTCP、UDP、マルチキャストなどパケットの種類は変える方法など様々な応用形態がある[5]。

3.4 デッド・レコニング

WANのようにサイト間の距離が長くなると、通信路の帯域幅に関わらず通信に時間遅延が発生する。遠隔のユーザが互いに時刻の等しい情報をもとに作業を行うためには、ローカルサイトの情報についても過去の情報を使う必要があるが、操作性の観点から非現実的である。デッド・レコニングとは、そのような問題を緩和するために、他のユーザの位置情報などを、過去のデータをもとにローカルサイトで各自予測して処理をしておき、予測した時刻のデータが実際に到着したときに誤差に応じて修正を行う手法である。デッド・レコニングは広域ネットワークを使用する共有仮想環境において一般的な手法である。また、パケットの送出頻度を減らすことができるのでトラフィックの軽減にも有効な手法である。もちろん、仮想物体が予測可能な比較的単純な経路を通る場合に応用は限られる。

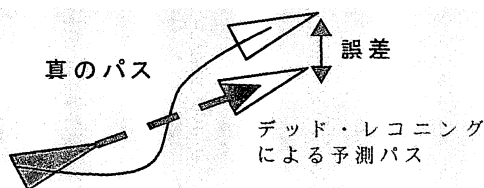


図6 デッド・レコニング (推測航法)

4. 共有仮想環境の例

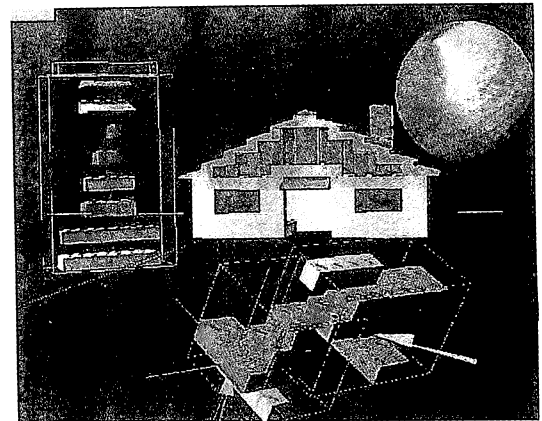


図7 VLEGO (II)の実行画面

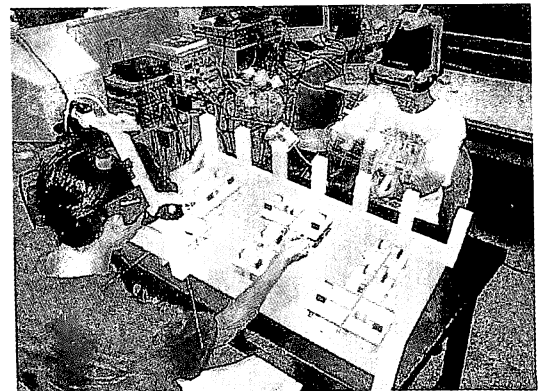


図8 VLEGO (II)での協調作業実施例

実際の共有仮想環境の構築事例として筆者等が開発したVLEGO (II)を紹介する[6]。VLEGO (II)は、複数の利用者が複数ボタンの3次元マウスを両手に把持して、仮想物体を操作する。通常の仮想環境に見られる基本的操作に加えて、VLEGO (II)では、種々の操作支援手法を取り入れ操作性の改善を図っている。図7はVLEGO (II)の実行画面の様子である。操作者のそれぞれの手に対応する3次元カーソル、操作対象仮想物体、3次元プリミティブメニュー、色変更3次元メニューなどが表示されている。さらに協調作業の実施は、二人の操作者が光学的シースルーHMDを装着して、お互いの動作を確認しながら形状組み立てや、仮想物体配置作業を実施できる。光学的シースルーHMDを用いるのは、仮想の環境に加えて実環境を共有して協調作業を実施することを可能にするためである。図8に実際にVLEGO (II)を用いて、協調作業を行っている様子の合成画像を示す。

VLEGO (II)では、協調作業に参加する利用者の数が比較的少ないことを考えて、サイト間の通信はTCPプロトコルによるユニキャスト通信を用い、通信の信頼性を確保している。また、操作の直列化は全サイトを同期させた上

でそれぞれのサイトで独立して行っている。

5. まとめ

ネットワークを利用したVR コミュニケーションとして CVE の構成方法について概説した。紙面の都合で詳細は紹介できなかったが、ネットワークで仮想環境を快適に共有するにはさまざまな創意工夫が必要である。特に多人数参加型のリアルタイムシミュレーション環境は構築が難しく解決すべき課題も多い。特に通信路遅延が存在する限りは、現実とまったく同じ共有環境の構築は不可能である。人間の知覚特性をうまく利用することで、実使用上不都合の無い環境を構築することが必要である。

参考文献

- [1] S. Benford, J. Bowers, L E. Fahlen, C. Greenhalgh, J. Mariani, and T. Rodden, "Networked Virtual Reality and Cooperative Work," Presence, Vol.4, No.4, pp. 364-386, 1995.
- [2] M. R. Macedonia, M. J. Zyda,, D. R. Pratt, P. T. Barham, and S. Zeswitz, "NPSNET: A Network Software Architecture for Large Virtual Environment," Presence,

Vol.3, No.4, 1994.

- [3] A. Pope : "The SIMNET Network and Protocols," BBN Report No.7102, BBN Systems and Technologies, Cambridge, Massachusetts, 1989.
- [4] <http://www.sics.se/dive/>
- [5] <http://www-npsnet.cs.nps.navy.mil>
- [6] K. Kiyokawa, H. Takemura, H. Iwasa, and N. Yokoya: "Collaborative immersive workspace through a shared augmented environment", Proc. SPIE, Vol.3517, Intelligent Systems in Design and Manufacturing, pp.2-13, Boston, MA Nov. 1998.

【著者略歴】

竹村治雄

1987年大阪大学大学院基礎工学研究科単位取得退学、同年国際電気通信基礎技術研究所入社、3次元ユーザインタフェース、臨場感通信会議システムに関する研究に従事。1992年同社主任研究員、1994年奈良先端科学技術大学院大学情報科学研究科助教授。現在に至る。1998年～1999年 トロント大学客員助教授。