LF-017

グラフィック・アクセラレータによる マルチエージェント・シミュレータの実装

The implementation of a multi agent simulator using graphic accelerator

赤嶺　有平†　　遠藤　聡志†　　山田　孝治†
Yuhei Akamine　　Satoshi Endo　　Koji Yamada

１．はじめに

本論文では，グラフィック・アクセラレータ（ビデオ・グラフィック・アダプタとも呼ばれる．以下 VGA と表記する）のピクセルシェーダ（プログラマブルな画像生成ハードウエア）においてマルチエージェント・シミュレータを実装する試みについて述べる．マルチエージェントシステム（以下 MAS と略す）は，従来の還元主義的な手法では解析が難しいとされている複雑系現象をモデル化し表現可能であることから近年注目を集めている手法である．

MAS を利用するためのツールとして，SWARM[1]や KK-MAS[2]などがある．これらのツールは，簡単な記述で MAS シミュレータを生成することが可能であるが，シミュレーション実行速度に関しては考慮されておらず，大規模なシミュレーションを効率的に行うことは難しい．

MAS は，解析対象とする系を自律的に行動し主体性を持つエージェントの集合で表現する．各エージェントは，環境を知覚し状況を判断することで自身の状態を変化させる．また，各エージェントの行動決定処理は，エージェントの種類によって決まっており，種類別に条件分岐することで処理ルーチンを一元化できる．したがって，知覚可能な環境の範囲を限定し移動可能な空間を格子で表現することで，セルオートマトン（CA）モデルとして記述することが可能である．CA モデルに変換することで，CA モデルの並列性を生かした各種の高速化手法が MAS シミュレーションに適用できる．

一方，エンターテインメント産業の発展にともない，リアルタイム３次元画像生成処理の高速化要求が高まり，画像生成処理の大部分を VGA が行えるようになっている．さらに，これまで固定的であった VGA の処理がプログラマブルに変更可能となり，VGA を用いて画像生成以外の処理を実装することも可能となっている．

CA は，VGA のプログラマブルなユニットのうち，ピクセルシェーダと呼ばれるユニットで実装可能なことが知られている．以下，２節でピクセルシェーダの詳細と，ピクセルシェーダを用いた CA シミュレータの実装方法について述べ，３節で MAS モデルを CA モデルに変換する手法について説明する．４節は，本研究において VGA 上に実装した道路交通モデルを紹介する．

２．ピクセルシェーダと CA

ピクセルシェーダは，VGA のレンダリングパイプラインにおいて最終段の処理，すなわち実際に描画されるピクセルの色を決定するユニットである．VGA のレンダリングパイプラインを図１に示す[3]．頂点データとは，３次元オブジェクトの形状を表すデータである．頂点シェーダは，頂点データに対して回転や透視変換などの処理を行う．ピクセルシェーダは，変換された頂点座標のピクセルカラーを出力する．ピクセルシェーダでは，ビデオ RAM に配置されているテクスチャ画像の任意のピクセルカラーにアクセスし，最終的な出力ピクセルカラーを決定する．これらの処理は，全てのピクセルに対して同様に行われ，通常 VGA はこれらの処理を並列に行う．

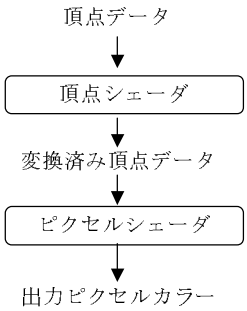

図１　レンダリングパイプラインの構造

```
float4 test(float2 uv : TEXCOORD0) : COLOR
{
    UV = uv;
    float s = st(1,-1)+st(0,-1)+st(-1,-1)+st(1,0)+st(-1,0)
              +st(1,1)+st(0,1)+st(-1,1);
    if(s == 3.0f) {
        s = 1.0f;
    }else if(s == 2.0f) {
        s = st(0,0);
    }else {
        s = 0.0f;
    }
    return float4(s,s,s,1.0f);
}
```

図２　HLSL によるピクセルシェーダ記述の例

CA では，空間を一様な格子状に分割する．分割された各部分空間をセルと呼ぶ．各セルは有限の状態を持ち，定められた規則（近傍則と呼ばれる）により同期的に状態遷移する．各セルの時刻 t+1 における状態は，時刻 t の近傍のセルの状態に応じて決定する．したがって，テクスチャ画像のピクセルのカラー値を CA における時刻 t の状態とみなし，最終出力画像のピクセルのカラー値を時刻 t+1 の状態とすることでピクセルシェーダのみで CA シミュレーションを行うことが可能である．

ピクセルシェーダをプログラミングするには，ピクセルシェーダ・アセンブラと呼ばれるアセンブリ言語を用いるか，HLSL（上位レベルシェーダ言語）とよばれる C 言語に似た構文を持つ記述言語を利用する[3]．図２は，

†琉球大学工学部情報工学科

ピクセルシェーダでライフゲームの CA 近傍則を記述した例である．一般には，HLSL を用いる方法が容易である．HLSL では，変換済み頂点データを入力として出力ピクセルカラーを出力する関数としてピクセルシェーダを記述する．CA シミュレーションを行う場合は，頂点シェーダからテクスチャ座標を取得することで近傍セルの状態へのアクセスを実装する．頂点シェーダから受け取るテクスチャ座標は，出力ピクセルの座標と等しくなるように調整する必要がある．

3. MAS から CA への変換

本論文では，エージェントの種類が少ない MAS モデルを想定する．ここで，同種のエージェントとは，環境から知覚した情報と自身の状態が同一ならば，まったく同じ行動決定を行うエージェントを指す．エージェントの種類が少ない場合，行動決定処理は，エージェントの種類によって条件分岐することで処理ルーチンを一元化することができる．さらに，エージェントの移動を格子上に限定し，各格子点に進入できるエージェントの数を制限する．これにより，エージェントが各格子点に存在するか否かを表現する状態とエージェントが持つ状態を CA の状態として表現することができる．エージェントの移動は，移動元のセルの状態を移動先のセルの状態にコピーすることで対応できる．移動元のセルは，エージェントが存在しない状態に遷移する．これを CA の近傍則で表現するには，以下の２段階に分けて記述する．

1. 着目セルのエージェントの行動決定
2. エージェントのコピーとクリア

これらを２種類の近傍則として記述し交互に状態遷移を行う．すなわち，最初の規則によっていったん全てのセルを更新した後，２番目の規則に切り替えて全セルの更新を行う．

最初の規則では，着目セルにエージェントが存在する場合は，環境を知覚し行動を決定する．移動する場合は，

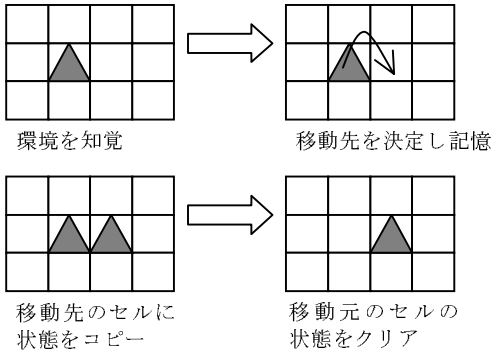

図3:マルチエージェント･モデルの CA 近傍則としての表現

移動先を状態として保存する．２番目の規則は，着目セルにエージェントが存在し，エージェントの移動先状態が「移動」状態になっていれば，着目セルのエージェントをクリアする．「停止」状態の場合は何もしない．エージェントが存在しない場合は，近傍セルの移動先状態を調べて着目セルに到達するエージェントの状態をコピーする．

4. 道路交通モデルの実装

本研究では，前述の手法を用いて道路交通モデルについて実装を行った．実装したモデルは，一車線の一方通行の道路を想定し，エージェントを速度 v を持つ車として走行させる．車は，一回の更新処理につき v セル分前方に移動する．車は，前方に十分な空きがあれば vmax に達するまで加速し，空きがない場合は減速する．これを CA モデルの近傍則として表すと，状態としてセルが車，空き，移動状態のいずれかを表す速度 v を持ち，近傍側は以下の２つの規則を交互に適用する．v が負の値のとき「空き」を表す．

1. 着目セルが「車」状態（v≧0）なら前方の空きセルを調べ，十分な空きがあり v＜vmax なら 1 加算する．
2. 着目セルが「移動」状態（v>0）なら「空き」状態（v=0）に遷移する．着目セルが「空き」状態なら着目セルに到達する「移動」状態の近傍セルを調べて速度 v をコピーする．

以上の規則を，ピクセルカラーの赤の輝度を表す値を速度 v に割り当てて HLSL で記述した．また，空間の格子サイズは，256x256，すなわち 256 セル分の車線のシミュレーションを 256 並列で行うこととした．

実装したシミュレータを，GeForceFX5900XT を搭載した VGA カードで実行した．同カードは，コアクロック 400MHz で動作し，1 クロックにつき 8 つのピクセルシェーダを並列実行する．実行結果から，各エージェントが衝突しないように加速していく様子が確認できた．また，実行速度は，毎秒 150 更新程度であった．これは，一秒に約 2 千万セル更新可能であり，1 セルの更新に 40 クロック必要であることを示している．前述の近傍則の HLSL による記述をコンパイルした結果 160 個のアセンブラ命令が生成されたため，単純に 4 以上の並列化がハードウエアにより実現されていることがわかる．

5. おわりに

本論文では，MAS シミュレータを VGA のピクセルシェーダを用いて実装する手法について述べた．まず，ピクセルシェーダにより CA シミュレータが実装可能であることを示し，特定の条件をみたす MAS モデルが CA の近傍側で表現可能であることを示した．また，簡単な道路交通モデルのシミュレータを VGA によって実装したところ，ハードウエアにより並列実行されていることが示された．

VGA のピクセルシェーダは，最近追加された機能であり，今後さらに性能向上が期待できる機能である．また，今回実験に用いた VGA カードは一世代前のハードウエアであり，動的フロー制御が行えないモデルである．したがって，動的フロー制御が可能な VGA カードによって実装した場合，さらに高速化されることが期待できる．

参考文献

[1] Nelson Minar,Roger Burkhart,Chris Langton,Manor Askenazi:THE SWARM SIMULATION SYSTEM,SFI,1996.
[2] 構造計画研究所　創造工学部：”MAS コミュニティ”, http://www2.kke.co.jp/mas/
[3] マイクロソフト株式会社：DirectX9.0 プログラマーズ・マニュアル,2002