

1614 領域分割による繁忙期における駅ホーム上のマルチエージェントシミュレーションの高速化

学 [電] ○上水流 友望 (千葉工大院)

正 [電] 富井 規雄 (千葉工大)

Acceleration of a multi-agent passenger flow simulation on a station platform by regional decomposition

Tomomi KAMIZURU, Chiba Institute of Technology
Norio TOMII, Chiba Institute of Technology

During peak seasons, platforms of Shinkansen stations are very congested because a lot of passengers without a seat reservation make queues to wait for a train. Increase of passengers who get on and who get off makes dwell times longer than planned and makes departure of trains delayed. Thus, it is strongly required to arrange the passenger queues and to give a proper guidance to them so that passengers can move smoothly and prevent trains from being delayed. We are developing a multi-agent simulator in order to estimate effective measures to make passengers move smoothly and prevent delays of trains. We succeeded to reduce the simulation time by introducing two acceleration methods, namely division of simulation areas and reduction of unnecessary calculation.

Key Words : railway, dwell time, simulation, multi agent simulation, acceleration

1. はじめに

お盆や年末年始などの繁忙期には多くの旅客が新幹線を利用する。駅ホーム上では、整列しようとしている旅客や乗降している旅客、降車し階段へ向かう旅客、売店を利用しようとしている旅客など多くの旅客がそれぞれの目的地に向かって移動している。それらの旅客が大きな荷物を持って駅ホーム上を行き来するために、旅客や荷物が他の旅客の移動の妨げとなっている。

特に、整列している旅客は、これから整列する旅客や売店へ向かう旅客の移動の妨げとなることがある。また、降車する旅客にとっても、ドア付近に整列している旅客は、整列位置によってはホームに出る際の妨げとなることがあり、そのために、降車や乗車に時間がかかり、列車の発車に遅延が生じることもある。従って、ホーム上の旅客の整列位置や整列方法は、乗降客の快適な移動を確保することや、列車を遅延なく運行させる上で非常に重要であると言える。

鉄道会社では、ホーム上に旅客の整列位置をマークしたり、駅員による案内を行ったりする他、繁忙期で自由席車両に乗り切れない場合には、自由席の利用者を指定席車両のデッキに誘導（指定席誘導）することによって、旅客の円滑な移動や乗降時間の短縮に努めている。また、現在行なっていない案内方法や施策を施すことも考えており、新たに行う施策を事前に評価する方法が求められている。しかし、さまざまな施策を事前に評価することは容易ではない。そこで、筆者らは繁忙期における新幹線駅ホーム上の旅客や荷物、駅員の動作をシミュレートし、整列位置・整列方法や誘導方法が旅客の移動や乗降に与える影響を評価するためのシミュレータを開発している⁽¹⁾。同様のシミュレータにマルチエージェントを用いた乗降時間の推定シミュレータ⁽²⁾や駅構内での群衆歩行シミュレータ⁽³⁾、遅延連鎖シミュレータ⁽⁴⁾などがあるが、いずれも在来線を対象としており、繁忙期における多数の旅客と旅客の持つ荷物、駅員の動作などは考慮されていない。

シミュレータ上で旅客や荷物、駅員の動作を再現するためには、駅ホーム上にいる他の旅客や荷物の影響を計算す

る必要がある。本研究で用いるマルチエージェントシミュレーションにおいて、旅客の動作を再現するためには、すべてのエージェントが旅客の視野に入るか否かを計算する必要があるため、すべてのエージェント間の距離や角度を計算する必要がある。再現する旅客の人数に応じて計算量が大きく増加してしまうことが問題となる。そこで本論文では、エージェントを予め分割することで不要な計算を省略することによってシミュレーションの高速化を図る。

2. シミュレーションモデル

2.1 シミュレーションモデル

本研究では、シミュレーションモデルとしてポテンシャルモデルを用いる。ポテンシャルモデルは、エージェント同士が持つ電荷から引力や斥力を求めることによって移動位置を決定する。引力によって、整列位置や売店の利用を、斥力によって荷物や柱の回避といった行動を再現することができる。また、ポテンシャルモデルには、歩行速度や単位時間を自由に設定できるという利点がある。

2.2 旅客エージェント

旅客エージェントは次のような情報を持つ。

- 現在地: $P = (x, y)$
- 目的地: $G = (x, y)$
- 歩行速度: $v[m/s]$
- 最大歩行速度: $v_{max}[m/s]$
- 半径: $r[cm]$
- 向き: $d[度]$
- 視野角: $\theta[度]$
- 対人視野: $s_p[cm]$
- 対物視野: $s_o[cm]$

2.3 視野内判定

旅客エージェントは、旅客エージェントと他のエージェントとの距離ベクトル f_{ij} と旅客の向きベクトル d_i がなす角 θ_{f_{ij}, d_i} が式 1, 2 を満たすエージェントを図 1 のように視野内にいるエージェントと判断する。ただし、 $\|f_{ij}\|$ が

いずれかの壁と交差する場合は視野外とする。

$$\|\vec{f}_{ij}\| < s_p \quad (1)$$

$$\theta_{f_{ij}, d_i} < \theta/2 \quad (2)$$

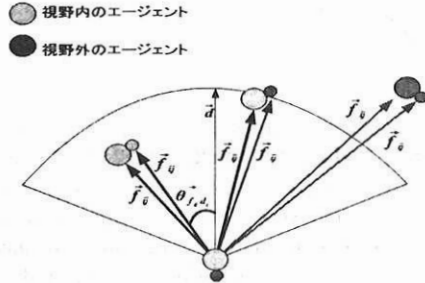


図1 エージェントの可視判定

2.4 速度ベクトル

旅客エージェントは次のような速度ベクトルを用いて移動位置の決定を行う。

● 目的地への引力

旅客エージェントの目的地への引力ベクトル $\vec{E}_{ig}$ は、旅客の現在地から目的地への距離ベクトルを $\vec{e}_{ig}$ 、目的地の電荷の大きさを Q_g とすると式3となる。

$$\vec{E}_{ig} = Q_g \cdot \frac{\vec{e}_{ig}}{\|\vec{e}_{ig}\|} \quad (3)$$

● 同方向へ進む旅客への引力

視野内にいる旅客エージェントの中で、同方向へ進む旅客に対して引力を持たせる。旅客エージェントが追従する旅客は、旅客エージェントの向きベクトル d_i と追従する旅客の向きベクトル d_k のなす角 θ_{d_i, d_k} が 15° 以内である旅客である。

同方向へ進む旅客への引力 $\vec{E}_{ik}$ は、視野内の面積を S_i 、視野内にいる対向している旅客の数を m_i 、旅客の現在地と追従する旅客との距離ベクトルを $\vec{e}_{ik}$ 、追従する旅客の電荷を Q_k とすると、式4となる。

$$\vec{E}_{ik} = \frac{m}{s} \cdot Q_k \cdot \frac{\vec{e}_{ik}}{\|\vec{e}_{ik}\|} \quad (4)$$

● 対向している旅客への斥力

視野内にいる旅客エージェントの中で、対向している旅客に対して斥力を持たせる。旅客エージェントに対向する旅客は、旅客エージェントの向きベクトル d_i と対向する旅客の向きベクトル d_j のなす角 θ_{d_i, d_j} が $120^\circ \sim 240^\circ$ である旅客である。

対向している旅客に対する斥力 $\vec{F}_{ij}$ は、旅客エージェントと対向している旅客との距離ベクトルを $\vec{f}_{ij}$ 、対向している旅客の数を n 、電荷を Q_j とすると、式5となる。余弦を乗じることで、並行移動や遠ざかる旅客への斥力を小さくする。

$$\vec{F}_{ij} = \sum_j^n \left(\frac{Q_j}{\|\vec{f}_{ij}\|^2} \cdot \frac{\vec{f}_{ij}}{\|\vec{f}_{ij}\|} \cdot \cos \theta_{d_i, d_j} \right) \quad (5)$$

● 壁への斥力

視界に入っている壁に対して斥力を持たせる。壁に

対する斥力 F_{iw} は、旅客エージェントと壁の最も近い位置への距離ベクトルを $\vec{f}_{iw}$ 、旅客エージェントの進行方向と $\vec{f}_{iw}$ とのなす角を $\theta_{E_i, f_{iw}}$ 、壁の電荷を Q_w 、壁の数を n とすると、式6となる。

$$\vec{F}_{iw} = \sum_w^n \left(\frac{Q_w}{\|\vec{f}_{iw}\|^2} \cdot \frac{\vec{f}_{iw}}{\|\vec{f}_{iw}\|} \cdot \left(\frac{1 + \cos \theta_{E_i, f_{iw}}}{4} \right) \right) \quad (6)$$

● 歩行速度

旅客エージェントの歩行速度 v_i は、引力と斥力の合力から求める。旅客 i に対する引力 $\vec{E}_i$ は式7、斥力 $\vec{F}_i$ は、式8となる。

$$\vec{E}_i = \vec{E}_{ig} + \vec{E}_{ik} \quad (7)$$

$$\vec{F}_i = \vec{F}_{ij} + \vec{F}_{il} + \vec{F}_{iw} + \vec{F}_{ib} \quad (8)$$

旅客の現在の速度を $\vec{v}_i$ とすると、単位時間 dt 後の速度 V_{new} は式9、10となる。

$$\vec{v}_{imp} = \vec{v}_i + \frac{(\vec{E}_i - \vec{F}_i) \cdot dt}{\|\vec{v}_i + \vec{E}_i \cdot dt\|} \quad (9)$$

$$\vec{V}_{new} = \begin{cases} \vec{V}_{imp} & (\|\vec{V}_{imp}\| \leq v_{max}) \\ v_{max} \cdot \frac{\vec{V}_{imp}}{\|\vec{V}_{imp}\|} & (\|\vec{V}_{imp}\| > v_{max}) \end{cases} \quad (10)$$

単位時間後の旅客の位置 p_{new} は、現在の位置 p_i より式11によって求められる。

$$p_{new} = p_i + \vec{V}_{new} \cdot dt \quad (11)$$

2.5 対象ホーム

シミュレーションの対象とする範囲は、島式ホーム上の自由席車両を中心にした範囲（長さ155m・約6両分幅12m）とした。本研究で用いるホームを図2に示す。

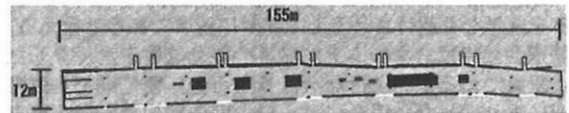


図2 シミュレーションに用いるホーム

3. ホームの領域分割による計算回数削減

3.1 従来研究の計算回数

従来研究⁽¹⁾では、エージェントの現在位置に関係なくすべてのエージェントとの視野内判定(2.3)を行っていた。すべてのエージェントが他のすべてのエージェントとの視野内判定を行うための計算回数 N_1 は、シミュレーション中の旅客エージェントが n 人のとき、式12となる。

$$N_1 = n * n \quad (12)$$

視野内判定を行うためには、エージェント間の距離や角度を求める必要がある。視野内判定を行う手順を次に示す。

1. エージェント間の距離ベクトルを計算
2. 距離ベクトルの長さを計算
3. 求めた距離ベクトルの長さが旅客の対人視野より短い判定
4. 短い場合5へ、長い場合視野外と判定する

5. 距離ベクトルとエージェントの向きベクトルがなす角を計算
6. 求めたなす角が視野角の半分よりも小さいか判定
7. 小さい場合 8へ、大きい場合視野外と判定される
8. 距離ベクトルと交差する壁があるか判定
9. 交差する壁がある場合視野外と判定され、ない場合視野内と判定される

3.2 旅客の視野

対人視野が s_p である旅客エージェントの視野内に入る可能性のあるエージェントは、エージェント間の距離が s_p より短いエージェントである。従って、エージェント間の距離が s_p より長いエージェントは、視野内に入る可能性がないため距離や角度を計算する必要がない。そこで、エージェント間の距離を計算せずに、エージェント間の距離が対人視野 s_p より長いエージェントを計算から除外するためにホームの分割を行う。本研究では、旅客エージェントの対人視野 s_p を 750cm とした。

3.3 ホームの領域分割と領域番号

エージェント間の距離が s_p より長いエージェントを除外するために、図 3 に示すようにホームを s_p 毎に分割する。分割した領域には領域番号を振り、左端を領域 0 として左から順に右端まで振る。本研究では、ホームの長さを 155m としたため、領域 0 から領域 21 までの領域に分割した。すべてのエージェントは、分割した領域のいずれかに分類される。分類される領域番号 $spaceNumber$ は、旅客エージェントの現在位置の x 座標 (ホーム長手方向) を s_p で除算し、端数を切り捨てた値とした。

領域番号 $spaceNumber$ を求める式を、式 13 に示す。本研究では、ホームの幅が 12m であり、y 座標 (ホーム短手方向) での領域分割は処理が削減されるエージェントが存在しないため、式 13 では、x 座標のみを用いている。

$$SpaceNumber = \lfloor x/s_p \rfloor \quad (13)$$

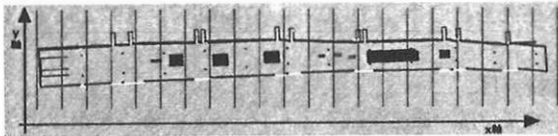


図3 旅客視野で分割した領域

3.4 視野内に入る可能性のあるエージェント

ある旅客エージェントの視野内に入る可能性のある他のエージェントは、その旅客エージェントの分類された領域 s とその両隣の領域 $s+1$, $s-1$ に分類されたエージェントだけである。2つ以上離れた領域に分類されたエージェントは、エージェント間の距離が長さ s_p の領域が1つ以上離れているため視野内に入る可能性はない。

あるエージェントの視野に入る可能性のある領域を図4に示す。図4は、領域Cにいるエージェントの視野に入る可能性のある領域を示している。このエージェントの視野内に入るエージェントは、領域BCDのいずれかに分類されているエージェントだけである。領域Aにいるエージェントは、領域Cから見ると領域Bの長さ750cmよりも離れているため、視野内に入る可能性はない。

4. 情報共有による計算回数削減

4.1 従来研究の手法

旅客エージェントの速度ベクトル計算は、2.4にあるようにエージェント同士の距離や距離ベクトルを用いてい

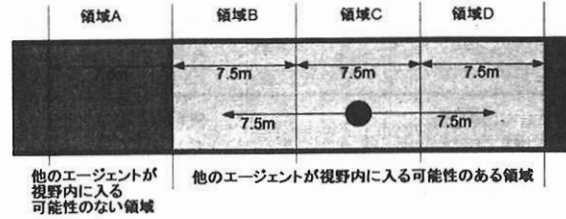


図4 エージェントの視野に入る可能性のある領域

る。従来研究では、双方のエージェントがそのエージェントの現在位置を基準に距離や距離ベクトルを求めていたため同じ計算を繰り返していた。本論文では、これらを片方のエージェントのみが計算を行い、計算結果を用いて双方の斥力ベクトルを求めることで計算回数の削減を行う。また、情報共有に加えて領域分割を用いることでエージェント間の組み合わせを減らし計算回数を削減する。

4.2 共有する情報

本論文では、2つのエージェント間で次の情報を共有する。

- 距離: $\|f_{ij}\|$
- 距離ベクトル: $\vec{f}_{ij}$
- エージェントの向きがなす角: θ_{d_i, d_j}
- エージェント間の壁の有無

これらのうち距離、エージェントの向きがなす角、エージェント間の壁の有無はどちらのエージェントを基準にして求めても同じ値となる。距離ベクトルは、基準となるエージェントが求めた距離ベクトルの向きを反転させることでもう一方のエージェントを基準にした場合の距離ベクトルが求められる。これらを共有することで旅客エージェント間の追従ベクトルと斥力ベクトルを同時に計算することが出来る。

4.3 同じ領域内における計算回数削減

同じ領域内にいるエージェント同士の計算は、情報共有を行わない場合、各エージェントが他のエージェントとの計算を行うため、 $n(n-1)$ 回となる。情報共有を行う場合、その領域内のエージェントから2つのエージェントでの組み合わせで計算を行うことでエージェント間の計算が終了するため nC_2 回となる。3エージェントでの計算回数を図5に示す。

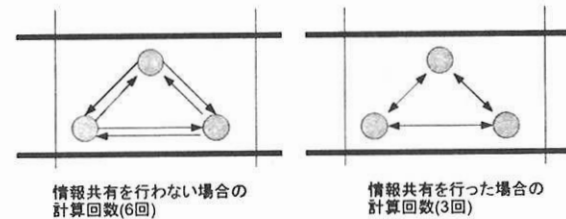


図5 3エージェントでの同じ領域内における計算回数

4.4 隣接する領域における計算回数削減

領域 n での隣接する領域との斥力計算を行うと、領域 n と領域 $n+1$ にいるエージェント間の斥力計算が完了する。次に領域 $n+2$ で隣接する領域との斥力計算を行うと、領域 $n+2$ と領域 $n+1$ にいるエージェント間の斥力計算が完了する。これにより、図6のように領域 $n+1$ は、隣接する領域との斥力計算を行う必要がなくなる。従って、本論文では、奇数番目の領域でのみ隣接する領域との斥力計算を行うことで偶数番目の領域の斥力計算を削減する。

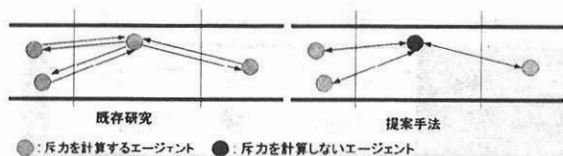


図6 隣接する領域にいるエージェントとの計算回数

5. フローチャート

領域分割、情報共有を用いた場合における斥力計算処理のフローチャートを図7に示す。

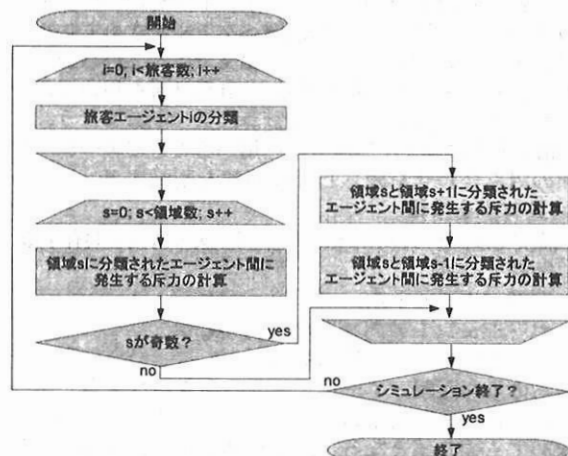


図7 高速化を行う斥力計算処理のフローチャート

6. 実験

6.1 実験1 - 高速化の検証

6.1.1 実験方法

本研究では、高速化を検証するための実験として次のような条件で処理時間の測定を行った。実験1では、階段に出現し整列する旅客と降車位置に出現し階段まで移動する旅客、各250人が行動を完了するまでの処理を行った。旅客は一定の間隔で出現し、整列位置、降車位置はランダムに決定した。

実験1-1 従来研究での処理時間

実験1-2 領域分割のみを行った場合の処理時間

実験1-3 領域分割及び情報共有を行った場合の処理時間

6.1.2 実験環境

- OS : Windows 7 Home Professional
- プロセッサ : Intel(R)Core(TM)i7-2600
CPU@3.4GHz3.4GHz
- 実装メモリ : 16.0GB

6.1.3 実験結果

各実験の処理速度を表1に示す。領域分割と情報共有を行った結果、処理時間を約7分の1に短縮できたことがわかる。

表1 処理時間(ミリ秒)

実験1-1	4,465,991
実験1-2	734,714
実験1-3	667,167

6.2 実験2 - 旅客流動の再現

6.2.1 実験方法

本シミュレータを用いた実験として旅客流動の再現があり、旅客の歩行が快適になると考えられる次のような整列方法に対して実験を行なった。

整列方法1 整列する旅客の先頭がドアの前であり、L字型に整列する(図8)。

整列方法2 整列する旅客の先頭がドアの前からずれており、L字型に整列する(図8)。

整列方法3 整列する旅客の先頭がドアの前であり、横一列に整列する(図8)。



図8 3種類の整列方法

整列客が旅客に与える影響を評価するために次の3つの時間の測定を行なった。

1. 旅客が80人整列するまでの時間(整列時間)
2. すべての整列客が乗車するまでの時間(乗降時間)
3. 降車した旅客がホームから退場するまでの時間(退場時間)

6.2.2 実験結果

各整列方法で10回測定した時間の平均を示す(表2)。整列時間と退場時間の点では、整列方法3が、乗降時間の点では、整列方法2が有利であることがわかる。

表2 旅客流動に要した時間(秒)

	整列時間	乗降時間	退場時間
整列方法1	388.50	78.24	223.05
整列方法2	369.18	73.71	204.84
整列方法3	351.12	78.00	197.25

7. 終わりに

本論文では、旅客流動シミュレータの旅客の視野に注目し、ホームを分割することで視野に入る可能性のある旅客を限定し、シミュレーションの高速化を行った。また、2エージェント間の斥力を同時に計算することで領域内、領域間の計算回数を減らし、シミュレーションの高速化を行った。

参考文献

- 1) 上水友望, 富井規雄:「マルチエージェントモデルによる繁忙期における駅ホーム上の旅客流動シミュレーション」, 電気学会 産業応用部門大会 2012 年 8 月
- 2) 稲木達哉, 富井規雄:「駅ホーム上のマルチエージェントによる乗降時間の推定」, 電気学会 交通・電気鉄道研究会, TER-11-062, 2011 年 11 月.
- 3) 阿久澤 あずみ:「駅構内における群集歩行シミュレーションモデルの研究」, 中央大学大学院理工学研究科情報工学専攻 修士論文, 2006 年 3 月
- 4) 高橋 郁人, 大木 徹憲:「高頻度運下での都市鉄道の遅延連鎖シミュレーションシステムの開発と遅延対策の分析」, 第18回鉄道技術・政策連合シンポジウム, 2011 年 12 月