

並列災害救助シミュレーションの精度評価手法

菅 真樹[†] 合 田 憲 人[†]

地震等災害発生時の人命救助方法の検討を目的とした災害救助シミュレーションの必要性が高まっている。本稿では並列災害救助シミュレーションの結果の精度と計算時間に関するトレードオフを評価するとともに、並列シミュレーションの誤差を推定するための誤差モデルを提案する。評価の結果、シミュレーション結果の精度に関するトレードオフを確認するとともに誤差モデルの有用性を確認した。

Evaluation methodology of parallel simulation accuracy for the disaster relief act

MASAKI KAN[†] and KENTO AIDA[†]

Rescue simulation, which aims to discuss the disaster relief act, is desired in the real world. This paper presents experimental results for the tradeoff between accuracy and computation time for parallel rescue simulation, and proposes a model to estimate an error of the parallel simulation. The evaluation results showed the tradeoff and usefulness of the proposed model.

1. はじめに

地震等災害発生時の人命救助方法の検討を目的とした災害救助シミュレーションの必要性が高まっている。災害救助シミュレーションでは、火災や交通流等の個別の現象に関するシミュレーション、市民や消防などのエージェント活動のシミュレーションを統合する形でシミュレーションが行われる。本シミュレーションでは、個々のシミュレーションに多量の計算が必要とされ、またシミュレーションの対象領域の拡大に従って計算量が増大するため、広範囲の地域を対象としたシミュレーションを行う際に実用的な時間内で実行することが困難であるという問題がある。

この問題を解決するため、シミュレーション対象領域を複数の部分的な地域に分割し、各地域に対して異なる計算機を割り当てて計算をさせることにより、シミュレーションの並列化を実現し、シミュレーション時間の短縮、スケーラビリティの向上を図る方法が提案されている。このような並列シミュレーションではシミュレーションを行う際に、割り当てられた地域のデータだけでなく、他の計算機でシミュレーションされている隣接地域のデータが要求されるため、計算機

間でデータの交換を行う必要がある。

計算機間で大量の通信を発生させることはオーバーヘッドの増大につながり、計算時間を大きく増大させる原因となる。したがって並列シミュレーションでは、隣接する地域の境界付近では近似的なシミュレーションを行って計算機間の通信量を軽減することにより、シミュレーション結果の精度と性能の両立を図る必要がある。

本稿では、災害救助シミュレーションの並列化において、シミュレーション結果の精度と計算時間に関するトレードオフについて評価を行うとともに、事前に結果の精度を推定するための誤差モデルを提案する。

精度に着目して述べる。2 節で並列災害救助シミュレーションの結果を評価する尺度について説明し、3 節でその結果の誤差を推定するためのモデル式を提案する。4 節で実際にシミュレーションを行った結果からトレードオフ、誤差のモデルについて評価、検討を行う。これらにより、並列災害シミュレーションの結果の精度と計算時間両面から評価することの有用性を示す。

2. 災害救助シミュレーションの並列化

2.1 RoboCupRescue シミュレータ

本稿では、災害救助シミュレーションのモデルとして、RoboCupRescue プロジェクト¹⁾で開発されてい

[†] 東京工業大学
Tokyo institute of Technology

る RoboCupRescue シミュレータ Ver.1²⁾ において採用されている方式を対象とする。

本シミュレータは Space time³⁾ の概念を用いた分散シミュレーションカーネルで、以下のような方法でシミュレーションの並列化を行っている。本シミュレーションではまず初めにシミュレーション領域を複数に分割し、各領域をそれぞれ異なる計算機上で実行されるシミュレーションカーネルに割り当てる。カーネルとは、図 1 で示すように火災、交通流、市民等のシミュレータやエージェントを統括して担当領域のシミュレーションを進めるソフトウェアシステムであり、同一の領域を担当するカーネル、シミュレータ、エージェントをまとめてサブシステムと呼ぶ。分散シミュレーションは、図 2 のように、複数のサブシステムを結合することで構築される。

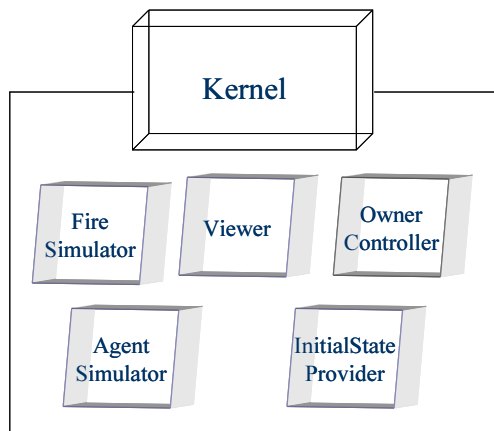


図 1 サブシステムの構成

ただし、本シミュレーションでは、ある地点のシミュレーションを行うためにはその周辺の状態をもとにして対象地点の計算を行う必要があるため、各サブシステムに割り当てられた地域の境界部分に関するシミュレーションでは、隣接するサブシステムの境界部分の状態を計算に用いる必要がある。そのため、本シミュレータでは隣接する地域の境界部分付近のシミュレーションをそれぞれの地域を担当するサブシステムが重複して行う。例えばサブシステム A、B が隣接した領域を担当する場合は A における重複部分（以下のりしる領域と呼ぶ）は本来隣接したサブシステム B の担当領域であるが、A のりしる領域のシミュレーション結果はシミュレーションが進むにつれ、B の結果と誤差が生じるため、この情報を定期的に更新（以下データ更新と呼ぶ）して補正することにより近似的なシミュレーション結果を得ている。

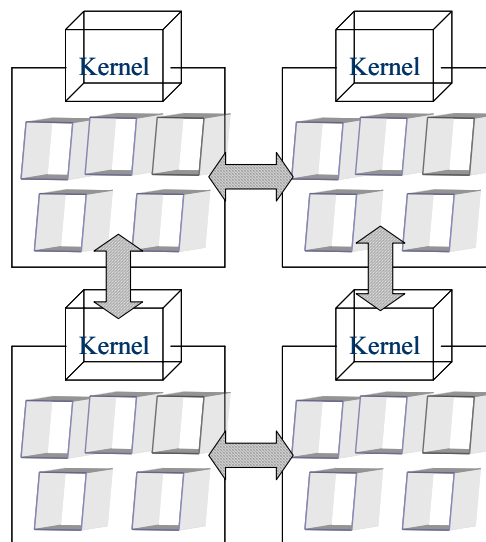


図 2 分散システムの構成

2.2 並列シミュレーションの結果精度尺度

前節で説明した並列シミュレーションでは、のりしる領域の近似シミュレーションや各領域のシミュレーションに用いられる乱数系列の違いから、並列シミュレーション結果が逐次シミュレーションと異なる可能性がある。

本稿では、並列シミュレーション結果が逐次シミュレーション結果とどの程度一致するかということと並列シミュレーションの精度と呼ぶ。このシミュレーション結果の精度と計算時間の間にはトレードオフの関係が存在すると考えられる。例えば、のりしる領域を小さくした場合、そのさらに外側の領域の状態が担当領域に与える影響を無視することになり、逐次シミュレーションの結果と誤差が生じる。だが、のりしる領域を小さくすることによって、サブシステムが計算するのりしる領域およびのりしる領域の更新のための通信時間が減少し、シミュレーション全体の計算時間は減少する。

災害救助シミュレーションにおいて、シミュレーションの結果はより早く正確に得ることが求められるが、たとえ厳密に精度が保証されなくても、シミュレーション結果にある程度の精度を保ちつつ、より短い時間でシミュレーション結果が得られる場合には、そちらが有用になりえる場合がある。例えば、現実よりも数倍短い時間である程度の精度を保ったシミュレーション結果を得ることは、災害時に災害対策を練る場合などにおいて、非常に有用であると考えられる。

このような並列シミュレーション結果の精度と計算時間のトレードオフを議論するために、並列シミュレ

シミュレーション結果の精度を評価するための評価尺度が提案されている⁴⁾。これは、対象地域全体に関する結果を評価する平均尺度と対象領域の各地点の結果を評価する座標尺度の2つから成る。本稿では、次節で述べる火災シミュレータを用いて、平均尺度と座標尺度双方から評価尺度の例を定義して並列シミュレーション結果を検討する。

2.3 火災シミュレータ

本節ではサブシステム中のシミュレータの例として、火災シミュレータについて説明する。

本火災シミュレータは、神戸市消防局・応用技術(株)高井(近畿大学)等によって開発された火災燃焼モデル⁵⁾を原型にされて実装されているRoboCupRescueシミュレータ Ver.0 向けのシミュレータを Ver.1 に移植したものである。このモデルは建物1棟毎に延焼を拡大させるモデルとなっており、具体的には、出火、燃焼、伝播および着火を判定・計算し、延焼の広がりを見表する。建物1棟毎に燃焼比率という値(以下 *fieryness* 値と呼ぶ)を持っており、基本的にはその値の変化を計算することによりシミュレーションが進行する。

3. 誤差モデル

本節では並列シミュレーションと逐次シミュレーション結果の一貫性を保持するための条件について述べるとともに、並列シミュレーション結果の誤差を示すモデルを提案する。

3.1 一貫性保持のための厳密モデル

並列シミュレーション結果と逐次シミュレーション結果の一貫性を厳密に保持するための、のりしる領域の決定方法が提案されている⁶⁾。本システムでは、ある地点の状態はその周囲の局所的な状態から決定されることを前提としている。遠い距離にある物体間の相互作用は近い距離にある物体間の相互作用の伝播によって実現される。サブシミュレータの1計算stepあたりの伝播速度が大きくと v 、通信間隔を u stepとした場合には以下の式(1)により求められる値より大ききのりしる領域を設定すれば並列シミュレーション結果と逐次シミュレーション結果の一貫性を保持できる。

$$v(1+u) \quad (1)$$

2.3節で述べた火災シミュレータのモデルでは、1計算stepにおいてある建物はその周囲20m以内に存在する建物にのみ影響を与えるため、式(1)における v は20mである。例えば、ここで u を4stepに指定した場合、式(1)よりのりしる領域が100m以上あれば

シミュレーション結果に誤差が生じることはないことがわかる。つまり、式(1)はあるサブシミュレータのオブジェクトが次のデータ更新までに影響を与える最大の範囲を示している。しかし、実際には20m/stepでの影響伝播は極めて稀であり、多くの場合より少ないのりしる領域でシミュレーション結果の一貫性を保つことができると考えられる。

3.2 一貫性保持のための期待モデル

本節では、実際のシミュレーションにおいて多くの場合期待される最大の伝播速度を期待伝播速度と定義し、本火災シミュレータの場合の値を求める。

本火災シミュレータのモデルはある建物のシミュレーションを行う際、燃焼過程では周囲の建物の影響を受けず、延焼、着火過程でのみ周囲の建物の影響を受ける。細かい計算式、定義などは文献1)12章を参照していただきたい。着火過程の計算は式(2)により行われ、 FF は火災影響量、 FH は延焼力、 $TRAN$ は伝播係数を表す¹⁾。 FF がある条件を満たした場合に着火イベントが発生する。

$$FF = FH \times TRAN \quad (2)$$

ここで、RoboCupRescueプロジェクトで使用されている神戸市長田区の約1800m×1500mの建物データから、平均的な建物パラメータ(床面積など)の算出し、以下の仮定

- (1) 火災シミュレーションの計算stepが30秒
- (2) すべての建物が平均的なパラメータをもつ同じ建物
- (3) 建物間の見通し角¹⁾は定数の2.94ラジアンとする。
- (4) 燃え移る方向と風向きが同じ
- (5) 消防活動¹⁾は存在しない

したがって、 FH の計算を行うと、式(3)のようになる。

$$FH = 15.99 \times \Delta P(t) \quad (3)$$

$\Delta P(t)$ は着火後の時間経過が t 秒の時のその建物の燃焼比率の変化量であり、同様の仮定のもとで計算すると式(4)のようになる。

$$\Delta P(t) = \frac{1}{1 + \exp(-0.02t + 3.12)} - \frac{1}{1 + \exp(-0.02t + 3.75)} \quad (4)$$

また、 $TRAN$ についても同様の仮定の元で計算を行うと式(5)のようになる。

$$TRAN = \frac{3.546}{D^2} \quad (5)$$

D は、燃焼している建物と燃え移る先の建物の直線

距離である。

式(2),(3),(5)より、FFは式(6)のようになる。

$$FF \doteq 56.7 \times \frac{\Delta P(t)}{D^2} \quad (6)$$

建物の着火条件は単位時間最大影響量(単位時間に各建物から受けた最大の延焼影響量 FF)が規定値(ここでは0.01)を超えた場合とする。 FF が0.01を超すには、式(4,6)より $t = 30$ の時には $D \leq 13.76$ 、 $t = 60$ の時には $13.76 < D$ 、 $t = 90$ の時には $17.73 < D$ の条件が必要であることがわかる。つまり着火直後の建物が30秒後にさらに隣接する建物を着火させるには建物間距離が13.76m以内でなければならないことがわかる。

よって期待伝播速度は最大13.76m/stepであるといえる。これは前述した仮定が含まれるため、実際には20m/stepで火災が伝播する可能性は依然として存在し、また周囲複数の建物から同時に影響を受ける場合についても考慮していない。だが、期待伝播速度を採用して式(1)によりのりしろ領域を決定した場合にも、多くの場合分散シミュレーションの同一性を十分保持できることが期待される。

また、上述の仮定にすべての建物が等間隔にマッピングされているという仮定を加えると最も近接の建物との距離は約17.15mとなる。その建物に延焼するのにかかる時間は前述の計算より60秒であることから、期待伝播速度は8.575m/stepと求められる。この値によるのりしろ領域の場合でも分散シミュレーションの同一性を保持できることが期待される。

3.3 誤差モデル

本節では、本稿が提案する並列シミュレーション結果の誤差を推定するためのモデルを定義する

ある離れたオブジェクトは、あるサブシステムの担当領域から離れているほど、サブシステムに与える影響は小さくなるがいえる。よってここで、担当領域の境界から x の距離に存在するオブジェクト全てが、自身の担当領域のオブジェクト全体に与える影響を αe^{-x} と仮定する。ここで α は定数項で正の値をとることとする。

そのとき、距離 V までの範囲に存在するオブジェクト全ての影響は以下のように示すことができる。

$$\int_0^V \alpha e^{-x} dx \quad (7)$$

一方、のりしろ領域の幅を a とすると、のりしろ領域が自身の担当領域のオブジェクト全体に与える影響は式(7)と同様に、

$$\int_0^a \alpha e^{-x} dx \quad (8)$$

とすることができる。のりしろ領域は実際に計算を行うので、式(8)で表される影響量は担当領域の誤差にならない。そこで、実際に担当領域のオブジェクトに対し外側のオブジェクトが与える影響量は以下の式(9)のように示すことができる。

$$Error = \int_a^V \alpha e^{-x} dx \quad (9)$$

ただし、 $Error > 0$ の場合、 $Error \leq 0$ の場合は $Error = 0$ とする。

これを図示したものが図3である。

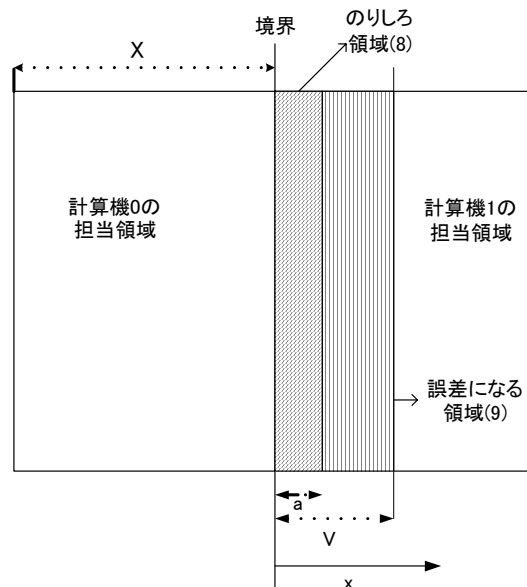


図3 誤差モデル図解

距離 a 、 V はともにシミュレーション領域の大きさによってその重みが変わると考えられるため、以降は担当領域の辺に対する割合で示すこととする。

V は一般には分散シミュレーションにおける、あるオブジェクトがデータ交換までに与える影響範囲である。これは分散シミュレーション担当領域の一辺の長さを X とすると、式(1)と同様に次のように求められる。

$$V = \frac{v(1+u)}{X} \quad (10)$$

ここで、 $a - V > 0$ 、すなわち $Error = 0$ であることは3.1節、3.2節のモデルにおいて、並列シミュレーションの結果の一貫性を保持できる条件であることを示している。3.1節の厳密モデルと3.2節の期待

モデルの違いは V の導出方法であり、前者は V が可能性がある最大の値をとり、後者は平均的なパラメータによって期待される伝播範囲の値をとる。

4. 評価実験

本節では、並列シミュレーション結果の精度と計算時間のトレードオフを示すための評価結果を示すとともに本稿が提案する誤差モデルの評価を行う。

4.1 実験方法

本稿では次のように評価尺度をより厳密に定義し、評価実験を行った。

燃焼建物数 領域全体における建物オブジェクトの $fieryness$ 値が 0 より大きい建物の数。平均尺度。
 $fieryness$ の総和 領域全体の建物オブジェクトの $fieryness$ 値の総和。平均尺度。

誤差の総和 建物それぞれの $fieryness$ の値を逐次シミュレーションの結果と比較し、その誤差の総和をとったもの。座標尺度。

一致しない建物の割合 建物それぞれの $fieryness$ の値を逐次シミュレーションの結果と比較し、逐次的場合と同一の値を得られなかった建物オブジェクトの全体に対する割合。座標尺度。

実験環境は 100Base-T ネットワークで接続された 4 ノード (dual PentiumIII 1GHz, 1GB Memory) 構成の PC クラスタを使用した。この PC クラスタ上で RoboCupRescue シミュレータ Ver.1 を実行し、神戸市長田区の一部の地図 (約 1800m × 約 1500m) のシミュレーションを行った。シミュレーションの時間粒度 (1Step あたりにシミュレーションされる時間) は 30 秒で 15 分間のシミュレーションを行った。領域の分割方法は、対象領域を短冊状に縦に 4 分割し、分割後の各部分地域を各ノードに割り当てている。

逐次実行の場合は、上記の 1 ノードのみを用いて、並列シミュレーションプログラムを実行した。データ交換の間隔は 2step に指定されており、実際のデータ交換は行われぬが、そのための計算処理は行われる。

4.2 精度と計算時間のトレードオフ

4.2.1 のりしろ領域を固定

のりしろ領域を 10% に固定して、データ交換の間隔を変化させて並列シミュレーションを行った場合の結果を表 1 に示す。表 1 より、まず並列シミュレーション結果と計算時間の間にはトレードオフがあることが確認できる。交換間隔を大きくすると、通信回数が減少するため計算時間の短縮が見られる。シミュレーション結果に関しては交換間隔が 3step の場合までは逐次実行の場合と完全に一致することがわかる。交換

間隔を大きくすると、のりしろ領域のさらに外側が与える影響が大きくなるため、誤差は増大するはずであるが、交換間隔が 5step と 6step の場合に着目すると、平均尺度のアプローチである燃焼建物数と $fieryness$ total のデータに関しては交換間隔が 5step である場合の方が 6 の場合よりも逐次実行の結果と比較して誤差が大きい結果となった。

1 つのサブシステムの担当領域の幅 X は 450m であり、のりしろ領域の幅は 45m である。期待伝播速度が 13.76m/step の場合、シミュレーション結果の同一性が保たれる $a - V > 0$ の条件を満たすのは、式 (10) より交換間隔 u が 2step 以下の場合である。これは実験結果を見た場合にも同一性が保たれている。また、期待伝播速度を 8.575m/t とした場合のシミュレーション結果の同一性が保たれる交換間隔は 4step 以下の場合であり、この場合もほぼ同一性が保たれることが期待される結果であることがわかる。

また、全体の 1.37% 程度の建物が逐次シミュレーションの場合と異なることを許容すると、計算アルゴリズム等を工夫することなく、約 6 割程度の計算時間でシミュレーションを終了することが可能になることがわかる。

4.2.2 交換間隔を固定

交換間隔を 3step (シミュレーション時間 90 秒) に固定して、のりしろ領域を変化させて並列シミュレーションを行った結果を表 2 に示す。

表 2 によりのりしろ領域を大きくすることによって、サブシステムが計算を行う領域が増大するため、それだけ多くの計算時間を必要とし、4.2.1 節と同様にトレードオフが確認できる。

また、期待伝播速度が 13.76m/step で、交換間隔が 3step ならば、式 (10) より並列シミュレーション結果の同一性を保つために必要なのりしろ領域の割合は 12.2% 以上である。表 2 の結果でも同一性を保持できていることが確認できる。

さらに、期待伝播速度が 8.57m/step である場合には、同一性を保持するために必要なのりしろ領域の割合は 7.6% 以上である。この場合にも表 2 の結果において同一性は保持できている。

この結果とのりしろ領域を固定した場合の結果から考えて、平均的なパラメータによる伝播速度である期待伝播速度によって、シミュレーション結果の同一性を保つのに必要な条件を算出することが可能であるといえる。

4.3 誤差モデルの評価

本節では、 $Error > 0$ の場合の誤差のモデル式の

表 1 のりしろ領域を固定した場合の結果

交換間隔 (step)	のりしろ領域 (%)	計算時間 (s)	燃焼建物数	fieryness total	誤差の総和	一致しない建物の割合
2	10	379.283	2102	1310.08	0	0.00%
3	10	324.201	2102	1310.08	0	0.00%
5	10	274.585	2115	1314.27	3.262	0.42%
6	10	255.539	2106	1310.14	4.592	1.03%
10	10	233.475	2090	1303.07	14.995	1.37%
15	10	217.549	2040	1255.63	57.142	3.58%
30	10	143.36	1770	1148.15	161.419	4.12%
逐次実行	—	683.278	2102	1310.08	—	—

表 2 交換間隔を固定した場合の結果

のりしろ領域 (%)	交換間隔 (step)	計算時間 (s)	燃焼建物数	fieryness total	誤差の総和	一致しない建物の割合
20	3	395.287	2102	1310.08	0	0.00%
10	3	324.201	2102	1310.08	0	0.00%
7	3	290.385	2102	1310.08	0	0.00%
5	3	281.381	2102	1310.08	0	0.00%
3	3	261.934	2100	1310.99	5.7266	0.95%
1	3	247.571	2074	1286.71	26.325	3.09%
逐次実行	—	683.278	2102	1310.08	—	—

正当性について検討する。表 1 の誤差の総和のグラフと、 $v=8.57\text{m/step}$, $\alpha = 1$ とした時の誤差のモデル式 (9) のグラフを、交換ステップ間隔を x 軸にとって描いたものが図 4 である。

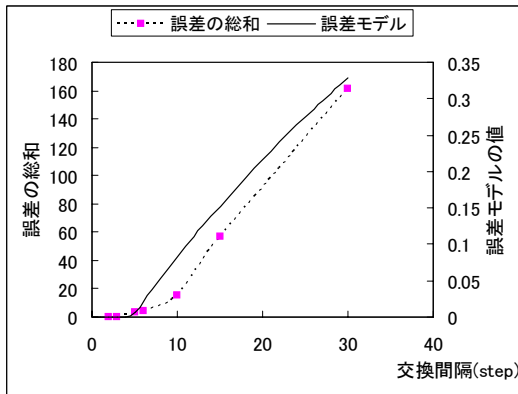


図 4 誤差の総和と、モデル式のグラフ

これを見ると、良く似た形状をしたグラフになっていることがわかる。よって、 α を適切に決定することにより、誤差モデルにより分散シミュレーションに現れる誤差を導出できると考えられる。

5. ま と め

本稿では、RoboCupRescue シミュレータを用いてシミュレーション結果の精度と計算時間の両面からシミュレーションの評価を行い、トレードオフの存在を確認するとともに、並列シミュレーション結果の誤差を推定するモデルを提案した。

評価の結果、シミュレーション結果と計算時間の間にトレードオフが存在することを確認するとともに、提案した誤差モデルの有用性を確認した。

今後はさらに評価尺度と結果精度のモデルについて検討を行い、要求されるシミュレーション結果の精度から、最適な並列計算方法を提案を目指す予定である。

謝辞 本研究を進めるにあたり、RoboCupRescue シミュレータ Ver.1 の使用について御協力いただいた電気通信大学竹内郁雄教授、小藤哲彦氏に感謝いたします。

参 考 文 献

- 1) 田所諭、北野宏明監修：ロボカップレスキュー緊急大規模災害救助への挑戦，共立出版，2000。
- 2) 小藤哲彦，竹内郁雄：複数のシミュレータを統合する大規模分散シミュレーションカーネル，情報処理学会研究報告 (DPS) No104-14,2001。
- 3) K.M.Chandy and R.Sherman. Space, time, and simulation. *Proceedings of the SCS Multi-conference on Distributed Simulation*, Vol.21, No.2, pp53-57, March 1989.
- 4) 合田憲人，菅真樹：ロボカップレスキューシミュレータの並列化，計測自動制御学会 SI2001, pp289-290, 2001。
- 5) 神戸市消防局編集，阪神・淡路大震災における火災状況 (神戸市域)，神戸市防災安全公社/東京法令出版株式会社
- 6) 小藤哲彦，竹内郁雄：RoboCupRescue Ver.1 におけるシミュレーションの分散化，計測自動制御学会 SI2001,pp287-288,2001。