

文部科学省科学研究費補助金基盤研究(A)

研究成果報告書（平成 18～20 年度）

(課題番号 : 18200009)

大規模マルチエージェントシステムを用いた
参加型デザインの研究

2009 年 3 月

研究代表者 石田 亨（京都大学情報学研究科社会情報学専攻）

まえがき

文部科学省科学研究費補助金基盤研究(A)「大規模マルチエージェントシステムを用いた参加型デザインの研究」は、2006年4月から2009年3月の3年間行われた。その概要は以下のとおりである。

大型計算機の時代は言わば、コンピュータ室で稼動する情報システムの時代だった。そこではウォーターフォール型のシステムデザインが一般的であった。デスクトップコンピューティングの時代は、家やオフィスで稼動する情報システムの時代で、オブジェクト指向デザイン、人間中心設計(human-centered design)が力を持った。これに対し、ユビキタスコンピューティングの時代は、パブリックスペースで稼動する情報システムが求められる。そこで本研究では、ユビキタスピューティング時代の情報システムの設計法を提案した。即ち、ユビキタスコンピューティングを用いたサービスシナリオを、参加型デザインや社会中心設計(society-centered design)と呼ぶ新しい設計法を用いて構築しようとした。

具体的には、パブリックスペースにおける新サービスを、研究開発から実社会へ展開するための、参加型デザインを研究した。新しいサービスのアイデアは、大学や企業の研究所で生まれるが、ユビキタス技術により進化するパブリックスペースで、多くの利用者が使えるものへと完成させるには、従来になじみのない設計方法論が必要となる。本研究で提案した方法論は、大規模マルチエージェントシステムに基づくシミュレーション技術を背景に、利用者が参加するシミュレーションや実験(ユーザテスト)を繰り返し、徐々に実世界での運用に移行させるというものである。

こうした参加型デザインの方法論は、新サービスのアイデアから開発運用に至るまでを、シームレスに繋ぐものである。本研究では、以下の3段階からなる参加型デザインのプロセスを提案し、その実現方法と効果を、実問題を用いた実証を通じて明らかにした。

1) 計算型シミュレーション (Multiagent-Based Computational Simulation)

設計の初期段階では、多くのアイデアを効率よく評価することが求められる。それには、人間を含まない、モデルに基づくテストが適している。計算型シミュレーションは、利用者をエージェントとしてモデル化し、エージェント群に課せられた社会的制約をプロトコルとして記述し、大規模なシミュレーションを行うものである。そこで、社会的制約下での自律的エージェントの研究や、プロトコルを記述するためのシナリオの制作プロセスや、シナリオ実行制御などの研究に取り組んだ。

2) 参加型シミュレーション (Multiagent-Based Participatory Simulation)

設計が進んだ段階では、ユーザテストが求められるが、その際できるだけ少ない人数で確からしいテストを実施することが大切である。そのためには、漸増的に被験者の数を増やしつつ評価を繰り返すことのできる環境が求められる。参加型シミュレーションは、シナリオ制御のエージェントの一部を、被

験者が制御するアバター（化身）に置き換え、被験者とエージェントが混在した状態でシミュレーションを行うものである。その実現のために、エージェントと被験者が全く同等の知覚や運動能力を持つ仮想的な環境を構築した。

3) 拡張実験 (Multiagent-Based Augmented Experiment)

仮想空間のシミュレーションで十分なリアリティが確保できない場合には、仮想空間から実空間へシームレスに移行する手段が求められる。拡張実験は、本研究で初めて提案したもので、実空間での被験者実験を、仮想空間でのマルチエージェントシミュレーションで拡張するという、全く新しい実験方法である。拡張実験では、実空間での被験者の振る舞いを仮想空間に投影し、エージェントと共にシミュレーションを行なった。その状況は、モバイル端末を介して実空間の被験者に刻々伝えられた。

ところで、社会シミュレーションには、大きく分けて 2 つのアプローチがある。第一は複雑な社会現象の解明を目的としたもの（分析型）で、エージェントは可能な限り単純化してモデル化される。複雑な現象は、エージェント相互のインタラクションによって生まれると考える。第二は新しい社会システムの創造を目的とするもの（創造型）で、人間や組織の挙動を現実に近い形で再現する。エージェントのモデルは複雑になることを厭わない。新しいシステムを実装する前段の実験や、利用者に疑似体験を与える訓練などに用いられる。本研究では、主として第二のアプローチに取り組んだ。創造型の社会シミュレーションの応用は、証券取引、交通制御、航空演習、避難誘導、環境保全など多岐に渡る。こうした応用で、社会シミュレーションは参加型デザインと融合し、新しい社会システムの設計方法論へと発展することを期待する。

2009 年 3 月

石田 亨（研究代表者）
京都大学 情報学研究科 社会情報学専攻
606-8501 京都市左京区吉田本町
TEL 075 753 4821 FAX 075 753 4820
E-mail ishida@i.kyoto-u.ac.jp

研究組織

研究代表者 :	石田 亨	(京都大学情報学研究科社会情報学専攻)
研究分担者 :	服部 宏充	(京都大学情報学研究科社会情報学専攻)

研究経費

平成 18 年度	13,000 千円
平成 19 年度	12,000 千円
平成 20 年度	12,400 千円
合計	37,400 千円

研究発表

(1) 受賞

1. 2008, 電子情報通信学会フェロー.
石田 亨. デジタルシティの実証的研究への貢献.
2. 2008, 電気通信普及財団賞 テレコムシステム技術学生賞
Yuu Nakajima, Hironori Shiina, Shohei Yamane, Hirofumi Yamaki and Toru Ishida. Disaster Evacuation Guide: Using a Massively Multiagent Server and GPS Mobile Phones. *IEEE/IPSJ Symposium on Applications and the Internet (SAINT-07)*, 2, 2007.
3. 2006, JAWS 最優秀論文賞.
宮田直輝, 石田 亨. コミュニティに基づく大規模エージェント配置手法の評価. 合同エージェントワークショップ＆シンポジウム (JAWS-06), 2006.
4. 2006, 文部科学大臣表彰科学技術賞研究部門. 石田 亨, 中西英之. デジタルシティにおける仮想都市空間構成法の研究.
5. 2006, Best Paper Award.
Yichuan Jiang, Toru Ishida. Concurrent Agent Social Strategy Diffusion with the Unification Trend. *Pacific Rim International Workshop on Multi-Agents (PRIMA-06)*, Lecture Notes in Artificial Intelligence, 4088, Springer-Verlag, pp. 256 - 268, 2006.

(2) 著書・編書

1. Mika Yasuoka, Toru Ishida and Alessandro Aurigi. The Advancement of World Digital Cities. *Handbook on Ambient Intelligence and Smart Environments*, Springer-Verlag, 2009.
2. Hideyuki Nakanishi, Toru Ishida and Satoshi Koizumi. Virtual Cities for Simulating Smart Urban Public Spaces. *Urban Informatics: Community Integration and Implementation*. F. Marcus Ed. *Handbook of Research on Urban Informatics: The Practice and Promise of the Real-Time City*. Information Science

Reference, IGI Global, pp. 256-268, 2008.

(3) ジャーナル

1. Toru Ishida and Hiromitsu Hattori. Participatory Technologies for Designing Ambient Intelligence Systems. *Journal of Ambient Intelligence and Smart Environments*, IOS Press, Vol. 1, pp. 39-45, 2009
2. Yi-Chuan Jiang and Toru Ishida. Local Interaction and Non-Local Coordination in Agent Social Law Diffusion. *Expert Systems with Applications*, Vol. 34, No. 1, pp. 87-95, 2008.
3. 中島 悠, 椎名宏徳, 服部宏充, 八槇博史, 石田亨. マルチエージェントシステムを用いた避難誘導実験の拡張. 情報処理学会論文誌, Vol.49, No.6, pp.1954-1961, 2008.
4. 中西英之, 石田 亨, 小泉智史. 大規模実環境実験のためのマルチエージェントシミュレーション. 日本バーチャルリアリティ学会論文誌, Vol. 12, No. 4, pp. 509-517, 2007.
5. 宮田直輝, 石田 亨. 大規模マルチエージェントシステムにおけるエージェント配置. 電子情報通信学会論文誌 Vol. J90-D, No. 4, pp. 1023-1030, 2007.
6. 伊藤英明, 中西英之, 石田亨. 超越型コミュニケーションを用いた位置に基づく遠隔指示の分析. 情報処理学会論文誌, Vol. 48, No. 3, pp. 1372-1380, 2007.
7. 中島 悠 椎名宏徳, 山根昇平, 八槇博史, 石田 亨. 大規模マルチエージェントシミュレーションにおけるプロトコル記述と実行基盤. 電子情報通信学会論文誌, Vol. J89-D, No. 10, pp. 2229-2236, 2006.
8. 伊藤英明, 中西英之, 小泉智史, 石田 亨. 超越型コミュニケーション: 大規模公共空間のための位置依存型誘導法. 情報処理学会論文誌, Vol. 47, No. 2, pp. 547-554 2006.
9. 山根昇平, 石田 亨. 大規模マルチエージェントシステムのためのメタレベル制御機構. 情報処理学会論文誌, Vol. 47, No. 5, pp. 1363-1370, 2006.
10. 村上陽平, 杉本悠樹, 石田 亨. 仮想訓練システムのためのエージェントのモデル構築. 人工知能学会論文誌, Vol. 21, No. 3, pp. 243-250, 2006.
11. 鳥居大祐, 石田 亨, ブスケ フランソワ. 参加型アプローチによる交渉モデルの獲得. 人工知能学会論文誌, Vol. 21, No. 3, pp. 287-294, 2006.
12. 鳥居大祐, 石田 亨. 社会・環境シミュレーションにおけるインタラクション層の構築. 人工知能学会論文誌, Vol. 21, No. 1, pp. 28-35, 2006.

(4) 国際会議

1. Shoichi Sawada, Hiromitsu Hattori, Marika Odagaki, Kengo Nakajima and Toru Ishida. Participatory Simulation Platform Using Network Games. *Eleventh Pacific Rim International Conference on Multi-Agents (PRIMA-08)*, Lecture Notes in Artificial Intelligence, 5357, Springer-Verlag, pp. 370-380, 2008.
2. Shohei Yamane and Toru Ishida. An Agent Modeling Method Based on Scenario Rehearsal for Multiagent Simulation. *Pacific Rim International Conference on Multi-Agents (PRIMA-07)*, Lecture Notes in Artificial Intelligence, 5044, Springer-Verlag, 2009.
3. Yuu Nakajima, Takatoshi Oishi and Toru Ishida. Analysis of Pedestrian Navigation using Cellular Phones.

Pacific Rim International Conference on Multi-Agents (PRIMA-07), Lecture Notes in Artificial Intelligence, 5044, Springer-Verlag, 2009.

4. Yusuke Tanaka, Yuu Nakajima and Toru Ishida. Learning Driver's Model Using Hypothetical Reasoning. *Pacific Rim International Conference on Multi-Agents (PRIMA-07)*, Lecture Notes in Artificial Intelligence, 5044, Springer-Verlag, 2009.
5. Yuu Nakajima, Hironori Shiina, Shohei Yamane, Hirofumi Yamaki and Toru Ishida. Disaster Evacuation Guide: Using a Massively Multiagent Server and GPS Mobile Phones. *IEEE/IPSJ Symposium on Applications and the Internet (SAINT-07)*, pp. 2-8, 2007.
6. Toru Ishida and Shohei Yamane. Introduction to Scenario Description language Q. *International Conference on Informatics Research for Development of Knowledge Society Infrastructure (ICKS-07)*, pp. 137-144, 2007.
7. Yichuan Jiang and Toru Ishida. A Model for Collective Strategy Diffusion in Agent Social Law Evolution. *International Joint Conference on Artificial Intelligence (IJCAI-07)*, pp. 1353-1358, 2007.
8. Toru Ishida, Yuu Nakajima, Yohei Murakami and Hideyuki Nakanishi. Augmented Experiment: Participatory Design with Multiagent Simulation. *International Joint Conference on Artificial Intelligence (IJCAI-07)*, pp. 1341-1346, 2007.
9. Shohei Yamane and Toru Ishida. Meta-level Control Architecture for Massively Multiagent Simulations. *Winter Simulation Conference (WSC-06)*, pp. 889-896, 2006.
10. Yuu Nakajima, Hironori Shiina, Shohei Yamane, Hirofumi Yamaki and Toru Ishida. Caribbean/Q: A Massively Multi-Agent Platform with Scenario Description. *International Conference on Semantics, Knowledge and Grid (SKG-06)*, 2006.
11. Yichuan Jiang and Toru Ishida. A Hybrid Negotiation Mechanism among Agent Strategies. *International Conference on Semantics, Knowledge and Grid (SKG-06)*, 2006.
12. Daisuke Torii, Toru Ishida and Francois Bousquet. Modeling Agents and Interactions in Agricultural Economics. *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-06)*, pp. 81-88, 2006.

(5) シンポジウム・ワークショップ

1. Yuu Nakajima, Shohei Yamane, Hiromitsu Hattori, Toru Ishida. Evacuation Guide System based on Massively Multiagent System. *International Conference on Autonomous Agents and Multi-agent Systems(AAMAS-08)*, demo session, 2008.
2. Yichuan Jiang, Jiuchuan Jiang and Toru Ishida. Agent Coordination by Trade-off between Locally Diffusion Effects and Socially Structural Influences. *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-07)*, pp. 520-522, 2007.
3. Yichuan Jiang, Toru Ishida. Concurrent Agent Social Strategy Diffusion with the Unification Trend. *Pacific Rim International Workshop on Multi-Agents (PRIMA-06)*, Lecture Notes in Artificial Intelligence, 4088, Springer-Verlag, pp. 256 - 268, 2006. (best paper award)

4. Toru Ishida. Agent and Grid Technologies for Intercultural Collaboration. *Pacific Rim International Workshop on Multi-Agents (PRIMA-06)*, Lecture Notes in Artificial Intelligence, 4088, Springer-Verlag, pp. 1-4, 2006.
5. Yuu Nakajima, Hironori Shiina, Shohei Yamane, Hirofumi Yamaki and Toru Ishida. Protocol Description and Platform in Massively Multiagent Simulation. *AAMAS Workshop on Agent Technology for Disaster Management (ATDM-06)*, pp. 135-142, 2006.
6. Yohei Murakami and Toru Ishida. Participatory Simulation for Designing Evacuation Protocols. *AAMAS Workshop on: Agent Technology for Disaster Management (ATDM-06)*, 2006.
7. Naoki Miyata and Toru Ishida. Placing Agents in Massively Multi-Agent Systems. *AAMAS Workshop on Massively Multiagent Systems*, 2006.
8. Yichuan. C. Jiang and Toru Ishida Evolve Individual Social Laws to Global Social Convention by Diffusion from Superior Agents to Junior Ones. *AAMAS Workshop on: Massively Multiagent Systems*, 2006.
9. Shohei Yamane and Toru Ishida. Meta-level Control Architecture for Massively Multiagent Systems. *AAMAS Workshop on: Massively Multiagent Systems*, 2006.
10. Toru Ishida. Multiagent Simulation Meets the Real World. *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-06)*, pp. 123-125, 2006.

(6) シンポジウム・ワークショップ 国内

1. 澤田祥一, 服部宏充, 石田 亨, 小田垣まりか, 中島謙互. ネットワークゲームを用いた参加型シミュレーションプラットフォームの実現. 合同エージェントワークショップ&シンポジウム 2008 (JAWS-2008), 2008
2. 田中祐介, 中島 悠, 服部宏充, 石田 亨. 大規模マルチエージェント交通シミュレーションのための運転者のモデル化手法の提案. 第 22 回人工知能学会全国大会, オーガナイズドセッション「次世代知的交通システムにおける AI」, 3A3-09, 2008.
3. 神部洋介, 中島 悠, 石田 亨. GPS 携帯電話を用いた誘導支援システムの実装とその評価. 電子情報通信学会 知識処理と人工知能研究会・知能ソフトウェア工学合同研究会, AI2006-74 KBSE2006-82, pp. 57-60, 2007.
4. 中島悠, 大石隆俊, 石田 亨. 携帯電話を用いた歩行者の遠隔誘導の分析. 合同エージェントワークショップ&シンポジウム (JAWS-07), 2007.
5. 田中祐介, 中島 悠, 石田 亨. 仮説推論を用いた運転者モデルの学習. 第 21 回人工知能学会全国大会, 2E5-8, 2007.
6. 山根昇平, 石田 亨. マルチエージェントシステムのためのシナリオリハーサル. 第 21 回人工知能学会全国大会, 1C3-1, 2007.
7. 宮田直輝, 石田 亨. コミュニティに基づく大規模エージェント配置手法の評価. 合同エージェントワークショップ&シンポジウム (JAWS-06), 2006. (最優秀論文賞)
8. 中島 悠, 椎名宏徳, 山根昇平, 八槇博史, 石田 亨. マルチエージェントシステムを用いた避難誘導の拡張実験. 合同エージェントワークショップ&シンポジウム (JAWS-06), 2006.

9. 伊藤英明, 中西英之, 石田 亨. 超越型コミュニケーションの分析：位置に基づいた共同作業支援プラットフォーム. インタラクション, pp. 125-132, 2006.

(7) 解説

1. 中西英之, 石田 亨. インタラクションを投影するデジタルシティ. 日本建築学会建築雑誌, Vol. , No. 8, pp. 10-13, 2007.
2. 石田 亨, 鳥居大祐, 村上陽平, 寺野隆雄. 社会シミュレーションと参加型デザイン. 情報処理, Vol. 48, No. 3, pp. 271-277, 2007.

(8) 新聞

1. 「交通シミュレート技術. 日本 IBM と京大, 共同開発」, フジサンケイビジネスアイ 2008 年 6 月 11 日.
2. 「交通状況ミクロに予測. 日本 IBM, 京大と開発」, 日刊工業新聞 2008 年 6 月 11 日, 10 面.
3. 「車数百万台個別に再現. 交通量・CO₂ を予測」, 日経産業新聞 2008 年 6 月 11 日, 11 面.
4. 「デジタル都市で未来占う」, 日本経済新聞, 2007 年 6 月 17 日（日）朝刊 31 面, かがく Cafe.

目次

1 社会シミュレーションの現状	1
1.1 エージェントによる社会シミュレーション.....	1
1.2 社会システムの分析.....	1
1.3 社会システムの合成.....	2
2 参加型シミュレーション	2
2.1 シミュレーションへの人々の参加.....	2
3 参加型モデリング	3
3.1 モデリングへの人々の参加.....	3
3.2 農業経済の分析への応用.....	4
4 拡張実験	5
4.1 シミュレーションによる実験の拡張.....	5
4.2 避難誘導への応用.....	7
5 課題と将来	8

1 社会シミュレーションの現状

1.1 エージェントによる社会シミュレーション

コンピュータシミュレーションは、現象の理解、モデリング、予測、実験などを目的に、物理、化学をはじめ幅広い分野で利用されてきた。モデリングの主流は、システムの挙動を支配方程式で表現するトップダウン的手法である。それに対し、マルチエージェントシミュレーションは、要素と要素間の相互作用をモデリングするボトムアップ的手法である。従来、実験が困難であった、社会、経済、文化など、人間の意思決定が中心となる問題への接近法として研究されている。このようなマルチエージェントシミュレーションは、社会シミュレーション[Axtell 00, Gilbert 99, Moss 98, 鳥居 06b]と総称され、社会現象の理解と社会システムの創造の手段として注目され始めている。例えば、以下のような問題領域では、マルチエージェントシミュレーションによるアプローチが有効と考えられており、研究が活発化している。

1) 経済分析シミュレーション

シミュレーションによる市場経済分析の歴史は古く、また現在でも人工市場の構築と経済現象の再現は活発に研究されている。市場は、参加しているプレイヤーの選好や戦略が多様であり、非常に複雑な社会システムと言える。伝統的な経済理論では、市場に参加するプレイヤーの合理性が仮定されるが、実際には非合理的な振る舞いも少なくない[Poundstone 92]。そのため、合理的な個人を仮定した理論的分析ではなく、行動経済学で見られるように現実的な人間の行動分析が必要となる。そのような経済シミュレーションでは、任意の市場メカニズムの下で、プレイヤーごとの経済行動をモデル化し、分析できることが要求される。

2) 災害時避難シミュレーション

災害時避難シミュレーションは、家屋倒壊や火災などの災害状況下で、異なる属性を持つ避難者、救助者が、それぞれに避難行動、救助活動を行う様子を模擬する。避難者、救助者の行動は一様ではないため、個々人の行動にフォーカスしたミクロレベルの分析が求められる。さらには、局所的行動が全体に波及し、時にパニックを引き起こすなど、ミクロ、マクロの関係の分析も必要とされる。

社会シミュレーションは、その目的により大きく2種に分類できる。

1.2 社会システムの分析

社会シミュレーションは、複雑な社会現象の理解や社会システムの分析に利用される。マルチエージェントシステムが人間社会を素直に表現できるため、社会科学者による利用が活発化している。社会科学における実験や証明、発見の方法として、シミュレーションを用いる「人工社会」[Epstein 96]もその一部である。ここでの特徴的な概念は「創発」である。この概念は非線形システムを扱う複雑系理論に由来しており、要素どうしの相互作用から全く異なるレベルの現象が生まれることを表している。マルチ

エージェントシミュレーションは、創発における個体の属性や行動（ミクロレベル）と社会集団（マクロレベル）の関係を理解するために利用されている。ところで、エージェントのモデルが複雑になると、モデルの属性値とシミュレーション結果との因果関係の分析が難しくなる。そこで、各エージェントのモデルは簡潔なものとし、複雑さをエージェントの相互作用により生じさせる KISS (Keep it Simple, Stupid) 原理[Axelrod 97]に従うのが一般的である。

1.3 社会システムの合成

社会シミュレーションは、また、新しい社会システムの創造や、複雑なシステム開発の前段階の実験や訓練に用いられる。こうした合成型のシミュレーションは、システムの開発、検証に力点が置かれるため、エージェントのモデルが複雑になることは厭わない。例えば、ロボカップレスキュープロジェクトでは、都市における地震や火災の発生を想定した、災害救助シミュレータが開発されている[Kitano 01]。このシミュレータは、実空間で救命・救助活動を行っている人間やロボットを統合的に模擬し、救助支援システムを創造するために利用されている。また、Trading Agent Competition(TAC)[Lee 07, Niu 08]などの競技もシミュレーションの一環と言える。フライトイシミュレータのように単に操作手順を習得するのではなく、災害時や戦場での状況判断と現場での適切な社会的インタラクションを学ぶためのシステムも開発されている。利用者に疑似体験を与えることで、空間探索や意志決定を訓練できるからである。合成型のシミュレーションは、ユビキタスコンピューティングなど、社会に埋め込まれる情報システムを、人々がどのように利用するかを知るためにも用いることができる。現場での実証実験が不可能な場合には、こうしたシミュレーションの有用性は高い。なお、合成型のシミュレーションにステークホルダー（利害関係者）の様々な視点を反映するには、デザインプロセスに人間を参加させること（参加型デザイン）が大切となる[Ishida 07a, Torii 06, 鳥居 06a]。創造型のシミュレーションは、新しい情報システムの開発・検証に力点が置かれるため、モデルが複雑になることは厭わない。

2 参加型シミュレーション

2.1 シミュレーションへの人々の参加

エージェントと人間が協力して行うマルチエージェントシミュレーションは、参加型シミュレーション[Guyot 05, Guyot 06, Ishida 09, Murakami 05, Sawada 08, Yamane 06, 村上 06]と呼ばれている。例えば参加型シミュレーションを用いた訓練システムでは、指導員などの役割を演じるエージェントと訓練を受ける参加者によってシミュレーションが進行する。フライトイシミュレータなどの従来の訓練システムが、主に機械の操縦法を学ぶことを目的にしてきたのに対し、参加型シミュレーションは社会的インタラクションが大切になる。企業のマネジメントや集団の訓練などに利用されている。

図1(a)にマルチエージェントシミュレーションを示す。シミュレーションシナリオの記述に従って各々のエージェントの行動が制御されている。次に、図1(b)に示す参加型シミュレーションへと拡張する。

これを実現するには、シナリオで制御されていたエージェントの幾つかを、参加者が制御するアバターに置き換えるべきだ。エージェントとアバターは制御の方法が異なるだけで、シミュレーションでは同等に扱われる。本稿では、ソフトウェアで制御される行動主体をエージェント、参加者が制御する行動主体をアバターと呼び区別する。このようにすると、仮想空間でのシミュレーション過程を、参加者の振る舞いを含めてモニタを通じて観測することができる。

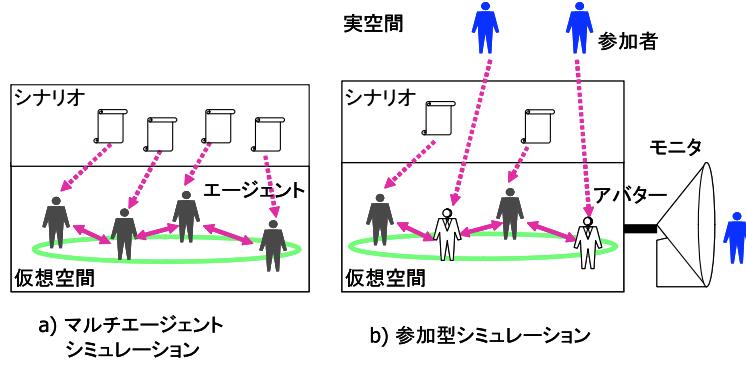


図1 参加型シミュレーション

図1(b)に示すように、参加型シミュレーションは、1) 人やグループをモデル化するエージェント(agent), 2) 参加者やそのグループを表すアバター/avatar, 3) インタラクションを規定するシナリオ(scenario), 4) アバターを制御する参加者(human subject), 5) 実空間を表現する仮想空間(virtual space), 6) 仮想空間で行われるシミュレーションを可視化するモニタ(monitor)から構成されている。¹

3 参加型モデリング

3.1 モデリングへの人々の参加

問題解決の過程に、それに関わるステイクホルダーが参加すると、政府などがトップダウンに解決するのに比べ、現実的な解を得られることが多い。市民参加によるまちづくりやごみ処理場の立地問題などがその例である。マルチエージェントシミュレーションを制度設計に利用する場合には、こうしたステイクホルダーの視点を反映する必要がある。専門家が収集したデータや資料だけに基づいてエージェントをモデル化するだけでは、ステイクホルダーの視点が十分に反映されるとは言えない。

¹ これは概念的な説明であって実装は様々である。例えば仮想市場ではアバターは陽に存在せず、参加者が仮想空間を直接操作する。その場合にも、エージェントと参加者はシミュレーションにおいて同等に扱われる。

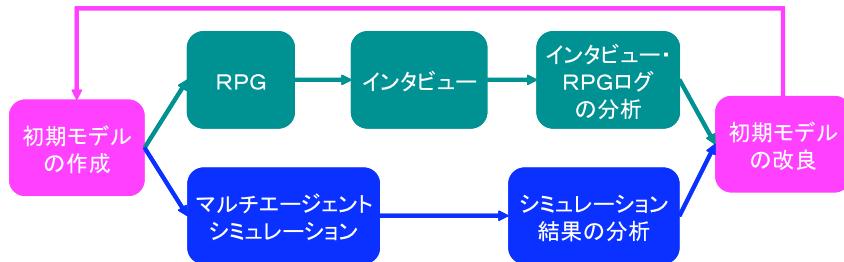


図 2 参加型モデリングのプロセス

図 2 に参加型モデリングのプロセスを示す。まず、文献や調査を基にエージェントの初期モデルが構築される。次に、ステイクホルダーを取り巻く環境を再現した RPG (Role Playing Game)が実施される。その後、RPG におけるステイクホルダーの意志決定の過程を理解するためのインタビューが行われる。RPG ログデータとインタビュー結果の解析によりエージェントモデルの改良が行われる。

複数回の RPG によってエージェントモデルが改良されると、そのモデルを用いてマルチエージェントシミュレーションが実施される。シミュレーションのインターフェースは RPG を想起させるものがよい。ステイクホルダーは RPG を体験しているのでシミュレーション結果を容易に理解でき、モデルの改善を提案することができる。このように、RPG とマルチエージェントシミュレーションを用いることにより、現実に近いエージェントモデルを獲得できる。

3.2 農業経済の分析への応用

フランス国際農業研究所(CIRAD: Centre de coopération internationale en recherche agronomique pour le développement)は国際稲研究所 (IRRI: International Rice research Institute) と共同プロジェクトを行い、ベトナムやタイにおける農作物選択や土地利用の分析に参加型モデリングを適用している [Bousquet 02]。例えばタイ東北部 (Khon Kaen 北部の Nam Phong 地方) では、ここ 30 年間、高台における換金作物の広がりが観察されている。これは、低地に向いた米の値段が下落する一方で、高台に向いたサトウキビの値段が安定しているためと考えられる。このプロジェクトはこの仮設を検証し、農家の意思決定プロセスの現状を理解し、将来のあるべき土地利用を描き出すことを目的としている。



図 3 RPG の様子

図3は、ステイクホルダーを取り巻く環境をボード上に再現した RPG を表している。RPG の結果はエージェントモデルに反映され、その後、土地利用に関するシミュレーションが行われる。図4は、高台から低地に至るまで作付けシミュレーションの結果を、RPG に参加した農家が評価している様子を表している。このシミュレーションにより、1) 換金作物を高台で栽培するインセンティブが農家にあること、2) 栽培作物の決定には大規模農家の影響が大きく、小規模な農家は栽培戦略をほとんど持っていないことが明らかになった。また、各農家は RPG とシミュレーション結果の評価に参加することを通じて、サトウキビ価格の決定プロセスと、他の農家と協調して作物を栽培するメリットを理解している。

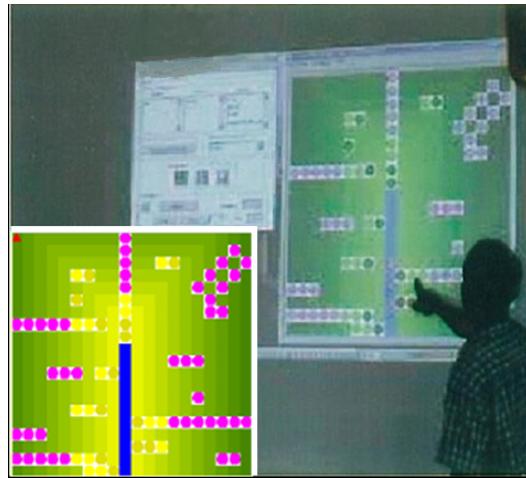


図4 シミュレーションによるエージェントモデルの評価
(左下はスクリーンショット)

このように参加型モデリングでは、意思決定過程のモデルが得られるだけでなく、RPG やシミュレーションによって参加者である農家が問題を理解することを助けている。RPG により参加者同士の議論が活発になり、他の農家の作物選択に対する考え方や計画を相互に学んでいる。また、この事例では、ほとんどの農家はシミュレーション結果を妥当なものとして受け入れた。以上のように、参加型モデリングは、ステイクホルダーが納得する形で解決法を提供できる。また、問題解決プロセスに慣れていないコミュニティにも適用できる。しかし、ボードを用いた RPG で複雑な交渉を表現することは難しい。また、ログ解析は人手に負うところが大きい。ログ解析への機械学習の適用などを検討する必要がある。

4 拡張実験

4.1 シミュレーションによる実験の拡張

これまで、人々が新しい技術をどう用いるかを予測するための実証実験が行われてきた。しかし、今後普及が予想されるユビキタスコンピューティングの場では、大量の電子デバイスが駅などの公共空間に

埋め込まれる。実験には機器の準備に加え公共空間の占有が必要となり、実現は容易ではない。また、公共空間での実証実験には多数の被験者が必要となるが、現実には十分な被験者を集めることは難しい。一方、シミュレーションは、様々な実験を行うのに適している。実験環境を整えるのが簡単であるのに加え、統制も容易である。また、実験の経緯を容易に記録することができる。

そこで、実空間での実験と仮想空間でのシミュレーションを統合することを考える。拡張実験（augmented experiment）は、少数の被験者による実証実験をマルチエージェントシミュレーションによって拡張する手法である[Ishida 07a]。図5は拡張実験がどのように実現されるかを示している。図5(a)は実空間での実験を、図5(b)は、実験に仮想空間をいかに導入するかを示している。実空間に配置されたセンサーが参加者の行動を仮想空間に投影している。センサーは、カメラでもRFIDでもGPSでもよい。この投影によって、実験全体の様子を仮想空間内で様々な視点からモニタすることができる。

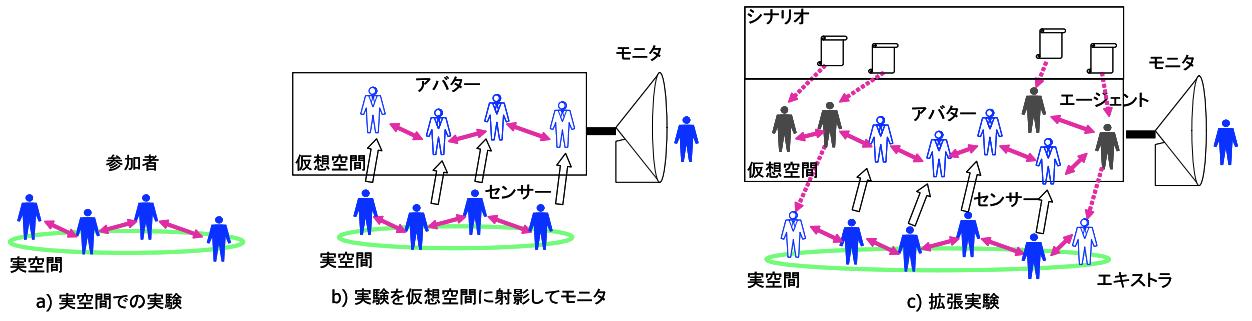


図5 拡張実験

拡張実験では、実空間での実験と並行して仮想空間でマルチエージェントシミュレーションが実行される。参加者に現実感を与えるために、エキストラを参加者の周辺に置くこともある。仮想空間での参加型シミュレーションとは異なり、参加者は実空間で自らの五感を用いて環境を理解し行動する。実空間での実験とシミュレーションの状況はモニタを介して知ることができる。モニタ上に現れる仮想空間内のアバターを通じて、実空間の参加者とコミュニケーションを行うこともできる。

図5(c)は拡張実験の構成を示している。拡張実験は、1) 人やグループをモデル化するエージェント、2) 参加者やそのグループを表すアバター、3) インタラクションを規定するシナリオ、4) アバターを制御する参加者、5) 実空間を表現する仮想空間、6) 実空間で行われている実験と仮想空間で行われるシミュレーションを統合して可視化するモニタ、7) 参加者やそのグループを仮想空間に投影するためのセンサー(sensor)、8) 仮想空間と実空間をつなぐコミュニケーションチャネル(communication channel)、そして 9) 仮想空間のエージェントの指示を受けて、実空間で参加者に対応するエキストラ(human extra)から構成される。

4.2 避難誘導への応用

拡張実験の例として、屋内（京都駅）と屋外（京都大学周辺）で行われた実験を紹介する。共に、実空間での実験をシミュレーションで拡張しているのだが、センサや仮想空間を実現するテクノロジーが異なっている。

京都駅構内の実験を図6に示す。この実験ではセンサーとして、天井に取り付けられた28台の自由曲面カメラが用いられている。センサーからの入力映像は連結され1枚の映像となる。それを解析することにより、乗客の位置と動きが捕捉される。仮想空間としては、京都大学で開発された3次元仮想都市システムFreeWalkを用いている[Nakanishi 04, 伊藤 06]。モニタは仮想空間を映し、アバターの動きがホームでの乗客の動きを刻々描き出している。モニタを観察する管制官と乗客とのコミュニケーションチャネルには携帯電話が用いられている。管制官はモニタ上のアバターを指定しコミュニケーションを始めることができる。下り回線を管制官が制御することによって、災害時に輻輳を避けながら現場とコミュニケーションをとるためのアイデアである。マルチエージェントシミュレーションが、実験と同時に実行され、その様子がモニタに映し出される。シミュレーションの状況は、携帯電話を通じて駅構内の参加者にも送られる。参加者は、シミュレーションの様子を頭に描きながら行動を選択していくのだが、実際の実空間では仮想空間の様子は見えない。さらに、実験とは関係のない乗降客が目の前を通り、災害の臨場感は十分とは言えない。



図6 屋内での拡張実験

図7に京都大学の周辺（屋外）で行われた避難誘導実験の様子を示す。この実験では、センサーにGPSを用い、参加者の位置と動きを捕捉している。仮想空間としては、2次元の地図を用いている。モニタで観察されるアバターの動きは、大学周辺の参加者の動きを刻々表したものである。モニタを観察する管制官と参加者とのコミュニケーションチャネルには、やはり携帯電話が用いられている。3000体のエージェントを用いたシミュレーションが、実験と同時に実行されモニタに映し出される[Nakajima 07, 中島08]。シミュレーションには、大規模なエージェント群を制御できるIBM東京基礎研究所のCaribbean[Yamamoto 99]と、京都大学のシナリオ記述言語Q[Ishida 02, Ishida 07b, Nakajima 06]が用いられた。シミュレーションの状況は地図上に表示され、携帯電話を通じて刻々大学周辺の参加者に送られる。参加者はシミュレーションの様子を頭に描きながら行動を選択していく。これは屋内実験と同じ枠組みなのだが、屋内実験で覚えた違和感はない。避難と関係のない市民が実験参加者の注意を惹くこともない。拡張実験の臨場感は、実験の場に大きく依存することが分かる。



図7 屋外での実験

5 課題と将来

マルチエージェントシミュレーションには多くの課題がある。工学的課題はいかにシミュレーションを実現するかに関する課題であり、科学的課題はいかにシミュレーション結果を解釈するかに関する課題である。工学的課題の筆頭は大規模なシミュレーションを実現することである。100万体のエージェント群を実現するには、膨大な並行処理を実装しなければならないが、シナリオライターに並行プログラミングの経験を期待することはできない。参加型シミュレーションの場合には、さらに、人とエージェント群のインタラクション設計が重要となる。こうした工学的課題が解決され、大規模シミュレーションが実現可能となったとしよう。交通シミュレーションが行われ、仮想空間で多数の事故が発生したとしよう。この結果は何を意味するのだろうか。人は仮想空間で実空間と同じように振舞うのだろうか。身体性の欠如は意思決定に根本的な相違をもたらすのではないか。こうした問い合わせに答えるには、参加型シミュレーションに関わる科学的研究が必要となる。シミュレーションで得られた結果の、何が重要で何が重要でないのだろう。エージェントの作りこみによって、どのような結果でも得られるのではないかという疑問が生じる。シミュレーションから確かな知見を得るために、解釈の検証方法が確立される必要がある。

参考文献

- [Axelrod 97] R. Axelrod. *The Complexity of Cooperation: Agent-based Models of Competition and Collaboration.* Princeton University Press, 1997.
- [Axtell 00] R. Axtell. Why Agents? On the Varied Motivation for Agent Computing in the Social Sciences. Brookings Institution CSED Technical Report No. 17, 2000.
- [Bousquet 02] F. Bousquet, O. Barreteau, P. Aquino, M. Etienne, S. Boissau, S. Aubert, C. Le Page, D. Babin, and J. C. Castella. Multi-Agent Systems and Role Games: Collective Learning Processes for Ecosystem Management. M. Janssen Ed. *Complexity and Ecosystem Management*, Edward Elgar Publishers, pp. 248-285, 2002.
- [Epstein 96] J. M. Epstein, and R. Axtell. *Growing Artificial Societies - Social science from the bottom up -*. MIT Press, 1996.
- [Gilbert 99] N. Gilbert and K. G. Troitzsch. *Simulation for the Social Scientist*. Open University Press, 1999.
- [Guyot 05] P. Guyot, and A. Drogoul. Using Emergence in Participatory Simulations to Design Multi-Agent Systems. *Fourth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2005)*, pp. 199—203, 2005.
- [Guyot 06] P. Guyot, A. Drogoul, and S. Honiden. Power and Negotiation: Lessons from Agent-Based Participatory Simulations. *Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2006)*, pp. 27-33, 2006.
- [Ishida 02] T. Ishida. Q: A Scenario Description Language for Interactive Agents. *IEEE Computer*, Vol. 35, No. 11, pp. 42-47, 2002.
- [Ishida 07a] T. Ishida, Y. Nakajima, Y. Murakami and H. Nakanishi. Augmented Experiment: Real-World Experiment Empowered by Multiagent Simulation. *International Joint Conference on Artificial Intelligence (IJCAI-2007)*, pp. 1341-1346, 2007.
- [Ishida 07b] T. Ishida, and S. Yamane. Introduction to Scenario Description language *Q*. *International Conference on Informatics Research for Development of Knowledge Society Infrastructure (ICKS-07)*, pp. 137-144, 2007.
- [Ishida 09] T. Ishida and H. Hattori. Participatory Technologies for Designing Ambient Intelligence Systems. *Journal of Ambient Intelligence and Smart Environments*, IOS Press, Vol. 1, pp. 39-45, 2009.
- [Kitano 01] H. Kitano, and S. Tadokoro. RoboCup Rescue: A Grand Challenge for Multiagent and Intelligent Systems. *AI Magazine*, Vol. 22, No. 1, pp. 39-52, 2001.
- [Lee 07] S. J. Lee, A. Greenwald, and V. Naroditskiy. RoxyBot-06: An (SAA)2 TAC Travel Agent. *Twentieth International Joint Conference on Artificial Intelligence (IJCAI-2007)*, pp. 1378-1383, 2007.
- [Moss 98] S. Moss. Social Simulation Models and Reality: Three Approaches. *Multi-agent Systems and Agent-Based Simulation (MABS-98)*, Lecture Notes In Computer Science, 1534, pp. 45-60, 1998.
- [Murakami 05] Y. Murakami, Y. Sugimoto and T. Ishida. Modeling Human Behavior for Virtual Training Systems. *Twentieth National Conference on Artificial Intelligence (AAAI-2005)*, pp. 127-132, 2005.

- [Nakajima 06] Y. Nakajima, H. Shiina, S. Yamane, H. Yamaki and T. Ishida. Caribbean/Q: A Massively Multi-Agent Platform with Scenario Description. *International Conference on Semantics, Knowledge and Grid (SKG-2006)*, 2006.
- [Nakajima 07] Y. Nakajima, H. Shiina, S. Yamane, H. Yamaki and T. Ishida. Disaster Evacuation Guide: Using a Massively Multiagent Server and GPS Mobile Phones. *IEEE/IPSJ Symposium on Applications and the Internet (SAINT-07)*, pp. 2-8, 2007.
- [Nakanishi 04] H. Nakanishi, S. Koizumi, T. Ishida, and H. Ito. Transcendent Communication: Location-based Guidance for Large-Scale Public Spaces. *International Conference on Human Factors in Computing Systems (CHI-2004)*, pp. 655-662, 2004.
- [Niu 08] J. Niu, K. Cai, S. Parsons, E. Gerding, and P. McBurney. Characterizing Effective Auction Mechanisms: Insights from the 2007 TAC market design competition. *Seventh International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2008)*, pp. 1079-1086, 2008.
- [Poundstone 92] W. Poundstone. *Prisoner's Dilemma*. Doubleday, 1992.
- [Sawada 08] S. Sawada, H. Hattori, M. Odagaki, K. Nakajima and T. Ishida. Participatory Simulation Platform using Network Games. *Eleventh Pacific Rim International Conference on Multi-Agents (PRIMA-2008)*, Lecture Notes in Artificial Intelligence, 5357, pp. 370-380, 2008.
- [Torii 06] D. Torii, T. Ishida and Francois Bousquet. Modeling Agents and Interactions in Agricultural Economics. *Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2006)*, pp. 81-88, 2006.
- [Yamamoto 99] G. Yamamoto, and Y. Nakamura. Architecture and Performance Evaluation of a Massive Multi-Agent System. *Third International Conference on AUTONOMOUS AGENTS (AGENTS-99)*, pp. 319-325, 1999.
- [Yamane 06] S. Yamane, and T. Ishida. Meta-level Control Architecture for Massively Multiagent Simulations. *Winter Simulation Conference (WSC-06)*, pp. 889-896, 2006.
- [伊藤 06] 伊藤英明, 中西英之, 小泉智史, 石田 亨. 超越型コミュニケーション: 大規模公共空間のための位置依存型誘導法. 情報処理学会論文誌, Vol. 47, No. 2, pp. 547-554, 2006.
- [村上 06] 村上陽平, 杉本悠樹, 石田 亨. 仮想訓練システムのためのエージェントのモデル構築. 人工知能学会論文誌, Vol. 21, No. 3, pp. 243-250, 2006.
- [中島 08] 中島 悠, 椎名宏徳, 服部宏充, 八槇博史, 石田亨. マルチエージェントシステムを用いた避難誘導実験の拡張. 情報処理学会論文誌, Vol. 49-6, 2008.
- [鳥居 06a] 鳥居大祐, 石田 亨, ブスケ フランソワ. 参加型アプローチによる交渉モデルの獲得. 人工知能学会論文誌, Vol. 21, No. 3, pp. 287-294, 2006.
- [鳥居 06b] 鳥居大祐, 石田 亨. 社会・環境シミュレーションにおけるインタラクション層の構築. 人工知能学会論文誌, Vol. 21, No. 1, pp. 28-35, 2006.

主要論文

(1) 著書・編書

1. Hideyuki Nakanishi, Toru Ishida and Satoshi Koizumi. Virtual Cities for Simulating Smart Urban Public Spaces. *Urban Informatics: Community Integration and Implementation.* F. Marcus Ed. *Handbook of Research on Urban Informatics: The Practice and Promise of the Real-Time City.* Information Science Reference, IGI Global, pp. 256-268, 2008.

(2) ジャーナル

1. 中島 悠, 椎名宏徳, 服部宏充, 八槻博史, 石田亨. マルチエージェントシステムを用いた避難誘導実験の拡張. 情報処理学会論文誌, Vol.49, No.6, pp.1954-1961, 2008.
2. 宮田直輝, 石田 亨. 大規模マルチエージェントシステムにおけるエージェント配置. 電子情報通信学会論文誌 Vol. J90-D, No. 4, pp. 1023-1030, 2007.
3. 伊藤英明, 中西英之, 小泉智史, 石田 亨. 超越型コミュニケーション: 大規模公共空間のための位置依存型誘導法. 情報処理学会論文誌, Vol. 47, No. 2, pp. 547-554 2006.
4. 村上陽平, 杉本悠樹, 石田 亨. 仮想訓練システムのためのエージェントのモデル構築. 人工知能学会論文誌, Vol. 21, No. 3, pp. 243-250, 2006.
5. 鳥居大祐, 石田 亨. 社会・環境シミュレーションにおけるインタラクション層の構築. 人工知能学会論文誌, Vol. 21, No. 1, pp. 28-35, 2006.

(3) 国際会議

1. Yuu Nakajima, Hironori Shiina, Shohei Yamane, Hirofumi Yamaki and Toru Ishida. Disaster Evacuation Guide: Using a Massively Multiagent Server and GPS Mobile Phones. *IEEE/IPSJ Symposium on Applications and the Internet (SAINT-07)*, pp. 2-8, 2007.
2. Toru Ishida, Yuu Nakajima, Yohei Murakami and Hideyuki Nakanishi. Augmented Experiment: Participatory Design with Multiagent Simulation. *International Joint Conference on Artificial Intelligence (IJCAI-07)*, pp. 1341-1346, 2007.
3. Shohei Yamane and Toru Ishida. Meta-level Control Architecture for Massively Multiagent Simulations. *Winter Simulation Conference (WSC-06)*, pp. 889-896, 2006.
4. Daisuke Torii, Toru Ishida and Francois Bousquet. Modeling Agents and Interactions in Agricultural Economics. *International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-06)*, pp. 81-88, 2006.

(4) 解説

1. 中西英之, 石田 亨. インタラクションを投影するデジタルシティ. 日本建築学会建築雑誌, Vol. , No. 8, pp. 10-13, 2007.

Chapter XVIII

Virtual Cities for Simulating Smart Urban Public Spaces

Hideyuki Nakanishi
Osaka University, Japan

Toru Ishida
Kyoto University, Japan

Satoshi Koizumi
Osaka University, Japan

ABSTRACT

Many research projects have studied various aspects of smart environments including smart rooms, home, and offices. Few projects, however, have studied smart urban public spaces such as smart railway stations and airports due to the lack of an experimental environment. We propose virtual cities as a testbed for examining the design of smart urban public spaces. We developed an intelligent emergency guidance system for subway stations and used the virtual subway station platform to analyze the effects of the system. This experience allows us to argue that simulations in virtual cities are useful to pre-test the design of smart urban public spaces and estimate the possible outcome of real-life scenarios.

INTRODUCTION

Virtual cities are three-dimensional graphical representations of digital cities (Ishida, 2002a). This chapter describes virtual cities for testing smart environments installed in large-scale

crowded urban public spaces such as airports and railway stations. Smart environments are living spaces with embedded abilities to perceive what their inhabitants are doing and support their lives. Since living spaces vary with respect to scale, smart environments of various sizes have been

developed: smart rooms (Bobick et al., 1999), smart classrooms (Brotherton & Abowd, 2004), smart homes (Kidd et al., 1999), smart offices (Addlesee et al., 2001), and smart conference sites (Sumi & Mase, 2001). There have been few attempts, however, to develop smart environments in urban public spaces such as airports and railway stations, even though they are an essential part of our everyday life. A manifest reason for this is the sheer vastness of such spaces. Many researchers have built their own room (Bobick et al., 1999) and home (Kidd et al., 1999) in which to conduct their projects, but it is almost impossible to build urban public spaces in research laboratories. Conference sites are too large to be purpose-built as laboratories, too. However, there is no need to do so, since researchers prefer

to take advantage of actual conference events as testbeds for evaluating their systems (Sumi & Mase, 2001). Such deployment is preferred also for testing smart classrooms (Brotherton & Abowd, 2004) and smart offices (Addlesee et al., 2001). Deployment of smart urban public spaces is challenging, because it is difficult not only to attach sensors to the spaces but also to involve the visitors in situ; it can be awkward or prohibited to ask them to participate in experiments. Furthermore, it is usually impossible to shut visitors out of the space and occupy it in order to conduct experiments with study participants. We propose virtual cities as a solution. We present a user testing method that utilizes virtual cities populated with scenario-controlled software agents developed by us (Ishida, 2002b).

Figure 1. Evacuation simulation in the virtual Kyoto station



More than 300,000 passengers pass through Kyoto station, the main railway station in Kyoto City, every day. In this station we installed a guidance system that tracks passengers to help their navigation based on their current positions (Nakanishi et al., 2004). Beyond conventional navigation systems, which passively present route information, our system proactively sends instructions to the individuals' mobile phones to control their routes and avoid congestion. The system's primary application is crowd control in emergency situations. Fortunately, we were permitted to attach positioning sensors to the station's subway platform and install the system, though we were not allowed to conduct experiments that would employ many subjects playing the role of an escaping crowd. To conduct experiments without occupying urban public spaces, we developed a virtual city simulator integrating a large number of software agents and humans into the same crowd. This simulator can produce complex group behaviors such as escaping crowds. This simulator enables the agent-based user testing described in this chapter.

The next section explains how agents, humans, and avatars are integrated in the agent-based user testing. The third section presents an experiment, which was conducted to see how the user testing method can work on our guidance system. In the fourth section we discuss implications obtained from the experiment. The fifth section summarizes related work. The sixth section concludes this chapter.

AGENT-BASED USER TESTING

Augmented Experiments by Agents and Humans

We developed an AR (Augmented Reality) based user interface for the virtual city simulator. Even in a crowded urban public space, it is not difficult to test smart environments that support individuals (e.g. normal pedestrian navigation systems (Abowd et al, 1997), since an experiment in which just one person or a group is taking part does not disturb the space. In contrast, it is extremely

Figure 2. Subjects in the augmented experiment



intrusive to test smart environments that support crowds (e.g. crowd navigation systems such as our emergency guidance system). To solve this problem, we contrived multi-agent crowd simulations that can be overlaid onto physical spaces. A large number of agents in the simulation augment an on-site small-scale experiment. We call this kind of experiment an “augmented experiment” (Ishida et al., 2007). Augmented experiments enable us to conduct large-scale experiments in an urban public space with minimum interference with its daily operation. In the experiment performed on the subway platform at Kyoto Station we overlaid an evacuation simulation in which a hundred agents escaped from the platform to the upstairs concourse through the central staircase. Figure 1 presents a couple of screenshots of this simulation visualized in the virtual Kyoto Station. To avoid disrupting the station’s operation, only three subjects escaped in each evacuation trial. The AR based user interface was necessary for the subjects to experience such a simulation on the physical platform.

The AR based user interface was a mobile phone which displays four symbols: a cross, a triangle, a circle, and a double circle. See-through head-mounted displays are not suitable for presenting the simulation of augmented experiments, since it is unsafe to mask the field of view of a walking person with a wide-field image such as a virtual crowd. Semitransparent images are safer but it is harder for users to recognize features. As Figure 2 shows, we used mobile phones, because there are always people looking at their phone’s screen, reading and writing text messages while walking around in crowded places. Mobile phones are a simple means of enabling AR. Since small images of a 3D virtual space are difficult to understand, instead of displaying visual simulations, the mobile phones displayed symbols that directly represented what the subjects of the experiment needed to recognize. To produce a situation in which the subjects were often blocked by a surrounding virtual crowd and could not advance

freely, we used four symbols representing the following four different degrees of density and slowness of the crowd: When the subjects were facing a crowd too dense to advance through and they had to stop walking immediately, a cross was presented on the screen. If they found a triangle or a circle on the screen, they could move ahead slowly for one or two meters, as they were approaching the crowd. A double circle was presented whenever it was possible for them to walk freely. Before starting each evacuation trial we asked the subjects to keep to these rules. Note that their mobile phones displayed a symbol and also a guidance message (described later). Which symbol and which message a subject’s phone displayed was determined by the simulation and his or her position, which was tracked by the guidance system. The direction of subjects’ move was estimated by the predefined evacuation route in the experiment.

Participatory Simulations by Agents and Avatars

Since it is conceivable that augmented experiments are not the best solution, we formed another group of subjects, who sat in a laboratory room and controlled their avatars in Figure 1’s virtual Kyoto Station. This kind of human-in-the-loop multi-agent simulation is called a “participatory simulation” (Guyot et al., 2005). As Figure 3 shows, the subjects held a gamepad in one hand to control their avatars and a mobile phone in the other hand to receive the guidance messages (they did not receive the symbols described in the previous section). We designed our participatory simulations (VR mode) to contrast with the augmented experiments (AR mode): AR subjects walked around a physical space, whereas VR subjects controlled their avatars just like they would play a videogame. AR subjects received the symbols representing the current situation, while VR subjects could see the visual simulation from their avatars’ viewpoint.

Figure 3. Subjects in the participatory simulation

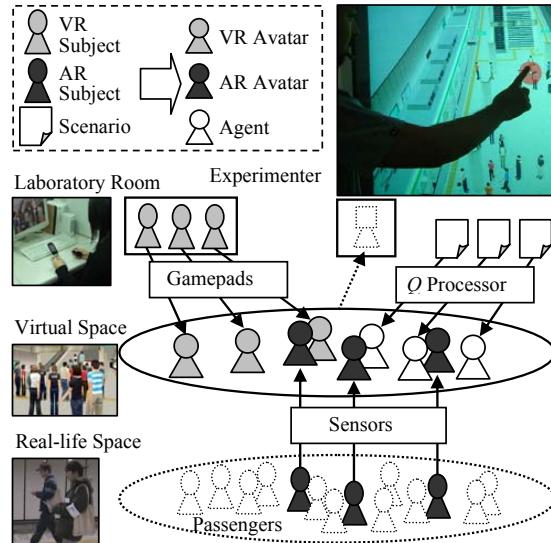


User Testing by Agents, Humans, and Avatars

In general, multi-agent simulations of group behaviors (Noda & Stone, 2003) show a different movement in each trial. If the AR and VR subjects could share the same simulations, we could exclude unnecessary variance in the analysis results of questionnaires, recorded data, and so on. Thus, we designed our simulator to be able to create both symbols and 3D animations from the same simulation, and to use both the sensors and the gamepads to control avatars. Figure 4 depicts agent-based user testing, which is a combination of augmented experiments and participatory simulations. The virtual space contains the VR and AR subjects' avatars, and the agents. The AR subjects and also passengers inhabit the real-life space. The laboratory room includes the VR subjects and an experimenter who monitors the simulation displayed on a large screen and administrates the experiment.

VR and AR subjects, and agents, have different input and output means in order to take part in the simulations. To integrate these heterogeneous participants, we attached a wrapper to the simulator's walking and seeing modules. VR

Figure 4. Agent-based user testing



subjects manipulate their gamepads to input the direction which they wish to advance. AR subjects' movements are tracked by sensors that inform the system of their current positions. Agents are controlled by their assigned simulation scenarios, in which their next destinations are specified. The wrapper converts these heterogeneous input data into changes in velocity and orientation. According to these changes, the walking module determines the following positions based on the pedestrian model (Okazaki & Matsushita, 1993) and the gait model (Tsutsuguchi et al, 2000). The wrapper also presents the same data managed by the seeing module in different forms. The data are positions and other parameters of the agents and the avatars. Agents can directly access the data to perceive the situation within the limitation of their visual power and visual field. VR subjects can see the 3D animations drawn according to the data, while AR subjects use their mobile phones to receive notifications of situational changes.

USER TESTING IN KYOTO STATION

The goal of the experiment was a trial use of the agent-based user testing. We tried to confirm that

our method was useful to pre-test the design of smart urban public spaces and estimate the possible outcome of real-life scenarios.

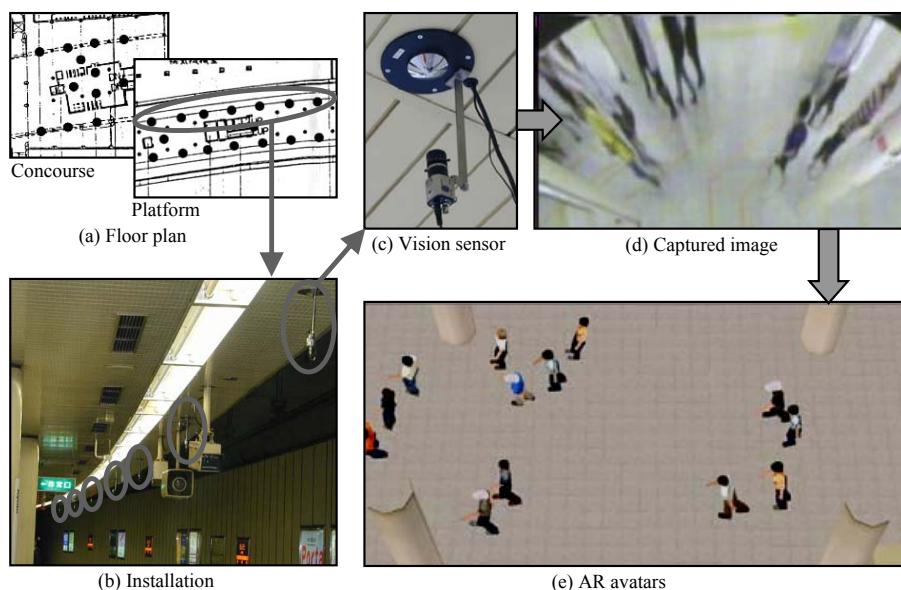
Smart Environment Tested in the Experiment

General emergency guidance is usually offered through public address systems which announce general information that is meaningful to the whole crowd. An example of this general guidance is: "There is a fire. Please use the nearest staircase to exit." Our emergency guidance delivered through mobile phones disseminates site-specific information that is suitable for each person. An example of this location-based guidance would be: "Please do not use the nearest staircase because it is too crowded." Location-based guidance systems need sensors to know the location of the addressed person and also the movement of the surrounding crowd.

We attached a vision sensor network to the station. We attached twelve sensors to the concourse and sixteen sensors to the platform. Figure 5(a) is the floor plan, on which the black dots show the sensors' positions, and Figure 5(b) shows how

they have been installed. The vision sensor network can track passengers between the platform and the ticket gate. In Figure 5(c), you can see a CCD camera and a reflector with a special shape (Nakamura & Ishiguro, 2002). If we could expand the field of view (FOV) of each camera, we could reduce the number of required cameras. However, a widened FOV causes minus (barrel) distortion in the images taken by conventional cameras. The reflector of our vision sensor can eliminate such distortion. The shape of the reflector can tailor a plane that perpendicularly intersects the optical axis of the camera to be projected perspectively to the camera plane. As shown in Figure 5(d), this optical contrivance makes it possible to have a large FOV without distortion. From the images taken by the cameras, the regions of moving objects are extracted using the background subtraction technique. The position of each moving object is determined based on geographical knowledge, including the position of the cameras, the occlusion edges in the views of the cameras, and the boundaries of walkable areas. Figure 5(e) shows AR avatars synchronized with the retrieved positions of AR subjects.

Figure 5. Positioning sensors installed in Kyoto Station



In Figure 2, the AR subjects wore a cap with a halogen lamp attached. The lamp and infrared filters covering the sensors were necessary to avoid errors in tracking the movements of the subjects on the platform, which was crowded with passengers. Passengers did not keep their distance from the subjects because they apparently ignored our experiment. Our location-based guidance system can work without this trick when the platform is sparsely populated.

The system needs the subjects' email addresses to send them guidance messages. In the experiment we registered the addresses before the evacuation began. We suppose that a real-life system would automatically register the addresses via the smart cards that people use to pass through the ticket gates. The system cannot work unless delay in email delivery is short. In the experiment the delay was about several seconds, which was short enough. We kept the email messages short to avoid distracting the subjects. Our previous prototype (Nakanishi et al., 2004) supported vocal instructions instead of guiding email messages, and we did not evaluate ways of conveying the guidance but did evaluate the psychological responses the guidance induces. Thus, the results of this experiment can be applied to both previous and current prototypes.

Hypotheses

Among other things, emergency guidance must be trustworthy in order to safely lead escaping people, who tend to lose their composure and recklessly follow others around them (Helbing et al., 2000; Sugiman & Misumi, 1988). A study on interpersonal trust in remote communication suggests that there are two independent factors in trust (Greenspan et al., 2000). One is the emotion-oriented attribute, which builds supportive impressions. In the case of emergency guidance, this is the degree to which the guide seems to be willing to help people to escape safely. The other factor is the cognitive-oriented attribute,

which increases confidence that a task will be successfully completed. In other words, the guide is believed to be able to grasp the situation perfectly and manage the evacuation efficiently. We supposed that location-based guidance is better than general guidance in terms of cognitive-oriented trust. Since location-based guidance takes into account the guided person's position and his/her surrounding situation, we expected that the guidance could give a feeling that the system is monitoring the scene of evacuation and issuing optimal instructions, with the guided person finally feeling at ease. Our hypotheses are that location-based guidance: 1) would be perceived as trustworthy; 2) would be useful as a navigation aid; and 3) might induce more calmness, which is important to prevent panic. We did not pressure the subjects into a panic, since even just a few people could cause serious trouble if they were to panic in a crowded station. Instead, we merely asked the subjects to arrive at the destination as soon as they could. The previous research showed that this sort of moderate pressure does work to a certain extent for investigating evacuations (Sugiman & Misumi, 1988).

Procedure

We recruited nine VR and eight AR subjects for a total of seventeen subjects. All of them were undergraduate students. We paid them for their participation. In each evacuation trial, three VR and three AR subjects escaped together with a hundred agents. To make the evacuation as simple as possible, the system guided everyone to the central staircase that soon became crowded as shown in Figure 1. In an actual evacuation on the subway platform more than a thousand of people would escape. A hundred agents, however, were enough to produce a crowded situation around a single staircase.

The subjects took part in an evacuation where they were guided with either the location-based method or the general way, and answered a ques-

tionnaire about the trustworthiness and usefulness of experienced guidance and calmness during the evacuation. Each subject repeated this twice in random order to experience both guidance methods. We analyzed the data of eight VR and five AR subjects, because one VR and three AR subjects were unable to experience one of the two kinds of guidance due to system problems.

In both location-based and general guidance evacuation, the system sent the subjects five guidance messages based on each subject's current location as follows. The first message was sent when the subject started escaping. The second, third, and fourth messages were sent when the subject passed through the spots that are fifteen, ten, and five meters away from the staircase respectively. Delay in email delivery did not become a problem since the subjects walked according to the messages that told them to move ahead slowly or stop walking. The final message was sent when the subject began climbing the staircase.

The messages used in the location-based guidance included information on the direction in which to proceed and the crowdedness around the staircase. At the moment the evacuation started, the subjects received the message "Please escape through the front staircase." Then, after they began walking and had advanced a certain distance, they were told to "Please keep going toward the staircase." When they arrived at the tail of the crowd jostling at the bottom of the staircase, they were guided with "Please use this staircase even though it is crowded." When they were about to finish passing through the crowd, they were told "Please do not hurry because you will pass the crowd soon." Finally, they were advised to "Please go up the staircase calmly," when halfway up the staircase to the concourse. The general guidance messages, on the other hand, were the plain emergency announcements: "Please escape through a staircase close to you," "Please do not hurry when fleeing from here," "Please choose the staircase nearest to you," "Please keep calm during the evacuation," and, "Please use a staircase nearby for evacuation."

Results

After each evacuation, all the VR and AR subjects answered the same questionnaire, which contained three items about trustworthiness, three items about usefulness, six items related to the calmness, and twelve items prepared for hindering the subjects' interpretation of the experimenter's intention. All the items were measured using a nine-point Likert scale ranging from 1 to 9. We analyzed the VR subjects' data and the AR subjects' data separately. We used a two-sided paired t-test to analyze differences in the impressions of the two guiding methods.

From the VR subjects' data, we found that location-based guidance was more trustworthy ($t(7)=3.1, p<.05$). This is the result of comparing the means of the "TRUST" index (general: 13.1, location-based: 19.6, 3 was the lowest, 27 was the highest). We obtained this index by summing the scores of these three items: "How much were you willing to follow the guidance?"; "How trustworthy was the guidance?"; and "How persuasive was the guidance?" (Cronbach's α : .84). We also found that location-based guidance was more useful ($t(7)=2.8, p<.05$). This was obtained from the "USEFUL" index (general: 15.3, location-based: 20.2, 3 was the lowest, 27 was the highest), which was made from these three items: "How useful were the guidance messages?"; "How easily could you understand the messages?"; and "How kind was the guidance service?" (Cronbach's α : .70). Figure 6 summarizes these results.

In the AR subjects' data, the TRUST and USEFUL indexes did not show significant differences, but we instead observed that the subjects were calmer under location-based guidance ($t(4)=3.1, p<.05$). This comes from the "CALM" index (general: 9.8, location-based: 14.6, 2 was the lowest, 18 was the highest), which was the sum of two items: "How unhurriedly did you escape?" and "How calm did you feel during the evacuation?" (Cronbach's α : .91). Figure 7 shows this finding. In the VR subjects' data there was no significant

difference in the CALM index.

Note that all the items that had significant difference are presented in the above analysis. We could not find any result that indicated superiority of the general guidance.

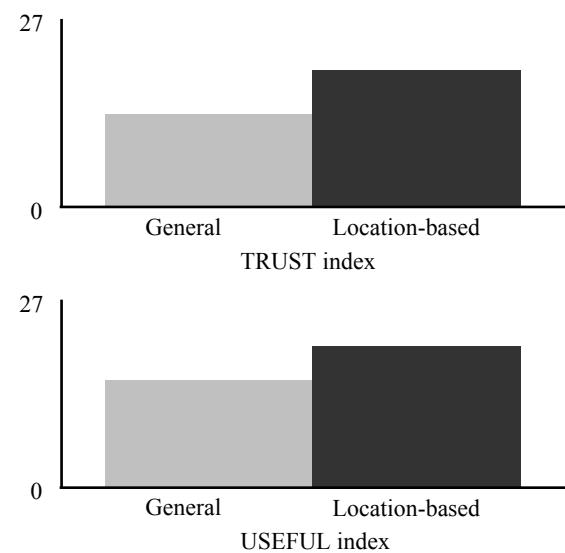
IMPLICATIONS

Analyzing the Findings of the Experiment

Interestingly, the location-based guidance elicited different responses from the VR and the AR subjects. The VR subjects appraised the method—the location-based guidance was trustworthy and useful, and the AR subjects became aware of their feelings—the location-based guidance kept them calm. Through analysis of the recorded videos, we explored what caused these different reactions.

First, we found that the VR subjects felt that the location-based guidance system was context-aware and tracking the current situation to make decisions. The VR subjects were watching the PC's screen drawing a visual simulation from their avatars' viewpoint. When the guidance's

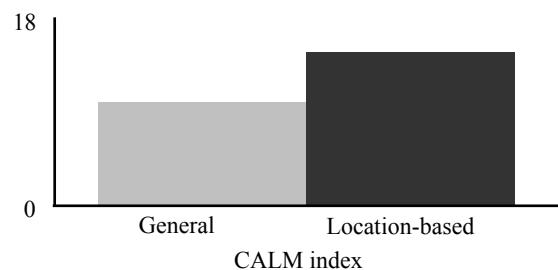
Figure 6. VR subjects' response



message informed them of the crowded staircase, they could actually see it. On the other hand, the AR subjects were just signaled to slow down or stop when they received the same message. It was not easy for them to know whether the system was context-aware or not.

Next, we observed that the AR subjects physically experienced the evacuation. The physical experience means that the subjects actually moved their bodies on the platform. When the virtual crowd around the staircase forced the AR subjects to stop walking and wait for the cross symbol—the “pause” instruction—to disappear, they received a different message for each guidance method. The location-based message, “Please do not hurry because you will pass the crowd soon,” was predictive, which might have helped to keep the subjects calm. In contrast, the general message, “Please keep calm during the evacuation,” did not predict anything, thus the subjects could not estimate how long they had to keep waiting, something that might have made them uneasy. If the AR subjects could see the virtual crowd as the VR subjects could do, they would have felt more uneasy and consequently the ‘calm’ effect of the location-based message could be observed more clearly. On the other hand, the VR subjects held down the “up” key to keep going forward before reaching the crowd and also after colliding with it. They did not need to release the key, since their avatars would automatically stop walking once the avatars hit other agents or avatars. This kind of operation is also observed

Figure 7. AR subjects' response



in videogame playing. The difference in guidance messages barely affected the VR subjects because whether they were walking or pausing did not matter to them.

Understanding Agent-Based User Testing

A lack of visual representations prevented the AR subjects from recognizing the crowd of agents and avatars around the staircase, so they were unable to notice the context-awareness of the guidance system. However, physical reality like what they felt when they were blocked by a surrounding crowd led them to indirectly evaluate the context-aware guidance system highly. Since the VR subjects' experience was deliberately differed from that of the AR subjects, it may be possible to generalize this as follows: In the physical space, software agents have difficulty in appearing visually, but they can interact with humans via mobile notification devices in terms of physical movement, even though they do not have a physical body. In a virtual space the agents can easily display their appearance and simulate social interaction with humans by means of 3D computer animations but can barely make them feel physical interaction. The kind of results that can be obtained from agent-based user testing depends significantly on the user interface for interacting with the agents. The VR and the AR interfaces, then, could provide results based on social and physical interactions. The results were different but both of them showed the advantages of the location-based guidance as we hypothesized in the previous section. If we had used a user interface that combined the two kinds of interactions, we might have different results. The development of such a user interface is a future work.

RELATED WORKS

Toward the goal of substituting agents for humans, researchers have been trying to develop various social agents (Nagao & Takeuchi, 1994), such as sales agents (Cassell et al., 1999) and trainer agents (Rickel et al., 2002). These agents are, however, still far from being used in practice. Meanwhile, agents have been used as movie extras (Macavinta, 2002). To realize the dream, it seems to be a much shorter way to develop agents with the ability to follow a scenario describing what to do than agents with complete autonomy. As described in this paper, the user testing of smart environments exemplified how such scenario-controlled agents could substitute for humans.

Our virtual city simulations appear to be similar to SpaceTag (Tarumi et al., 2000), which is location-based informational objects overlaid on the real world. Our simulator is, however, endowed with the gait (Tsutsuguchi et al., 2000), pedestrian (Okazaki & Matsushita, 1993), and multi-agent interaction models (Murakami et al., 2003) so that it can overlay complex group behaviors. Moreover, subjects can take part in our simulations from both mobile and desktop environments, whereas the SpaceTag system is an information-accessing mechanism for mobile users and has little meaning for desktop users. Social mixed reality systems (Crabtree et al., 2004; Okada et al., 2001) can enable mobile and desktop users share the same physical space, though they are not sufficient for the agent-based user testing. Our simulator, on the other hand, is equipped with a wrapper for integrating mobile and desktop users, and software agents. Furthermore, they can become the subjects of experiments, since the simulator is connected to the implemented smart environment, which is our guidance system.

When occupation and replication of the environment are impossible, there is an alternative to our method: exploitation of a similar environment. For example, one study uses a university building as a structural copy of a shopping mall

to evaluate a shopping guide system (Bohnenberger et al., 2002). To use this method, one has to be satisfied with a rough mock-up of the actual space unless an existing physical space is found whose structure imitates the space precisely. In our method, it is possible to use the actual space if the effort is made to construct a 3D model of the space and describe the scenario for group behavior. Our simulator FreeWalk/*Q* (Nakanishi and Ishida, 2004) is a combination of a virtual city “FreeWalk” and a scenario description language “*Q*” (Ishida, 2002b). *Q* is based on Scheme, which is a dialect of the Lisp programming language invented by Guy Lewis Steele Jr. and Gerald Jay Sussman. *Q*’s model is an extended finite state machine. According to the scenario written in *Q*, agents can walk, speak, and gesture in FreeWalk’s virtual space. Since FreeWalk/*Q* can simulate many kinds of group behaviors including evacuation, our method can be applied to various smart environments installed in crowded places.

CONCLUSION

We proposed agent-based user testing, that is a method for conducting experiments to test smart environments installed in large-scale crowded urban public spaces such as airports and central railway stations, since it is difficult to replicate such environments in a laboratory. Such places do not allow us to ask visitors to participate in our experiments or prevent visitors from entering so that we can conduct our experiments with selected subjects. For this method, we developed a virtual city simulator integrating agents, humans, and avatars. In testing the emergency guidance system installed in Kyoto’s central railway station, the simulator overlaid a virtual crowd consisting of a hundred agents, three humans, and three avatars on the station platform so that the experiment could be carried out with little interference to the station’s daily operation. The result of the

experiment supported our hypotheses that our location-based guidance system installed in Kyoto station was superior in trustworthiness, usefulness, and inducing calmness. This motivated us to conduct a totally physical experiment to confirm these effects.

ACKNOWLEDGMENTS

We express our thanks to the Municipal Transportation Bureau and General Planning Bureau of Kyoto city for their cooperation. This work would have been impossible without the invaluable participation of Toyokazu Itakura, Ryo Watanabe, Shinji Konishi, Shunsuke Tanizuka, Takatoshi Oishi, and Armando Rubio Torroella. We thank Shigeyuki Okazaki, Toshio Sugiman, Ken Tsutsuguchi, CRC Solutions, Mathematical Systems, and CAD Center for their support in the development of the guidance system and the social interaction platform. The platform is available at <http://www.ai.soc.i.kyoto-u.ac.jp/freewalk/> and <http://www.ai.soc.i.kyoto-u.ac.jp/Q/>.

REFERENCES

- Abowd, G. D., Atkeson, C. G., Hong, J. I., Long, S., Kooper, R., & Pinkerton, M. (1997). Cyberguide: A Mobile Context-aware Tour Guide. *Wireless Networks*, 3(5), 421-433.
- Addlesee, M., Curwen, R., Hodges, S., Newman, J., Steggles, P., Ward, A., & Hopper, A. (2001). Implementing a Sentient Computing System. *IEEE Computer*, 34(8), 50-56.
- Bobick, A. F., Intille, S. S., Davis, J. W., Baird, F., Pinhanez, C. S., Campbell, L. W., Ivanov, Y. A., Schutte A., & Wilson, A. (1999). The KidsRoom: A Perceptually-Based Interactive and Immersive Story Environment. *Presence: Teleoperators and Virtual Environments*, 8(4), 369-393.

- Bohnenberger, T., Jameson, A., Kruger, A., & Butz, A. (2002). Location-Aware Shopping Assistance: Evaluation of a Decision-Theoretic Approach. *Proceedings of International Conference on Human-Computer Interaction with Mobile Devices and Services*, LNCS2411, 155-169.
- Brotherton, J. A., & Abowd, G. D. (2004). Lessons Learned from eClass: Assessing Automated Capture and Access in the Classroom. *ACM Transactions on Computer-Human Interaction*, 11(2), 121-155.
- Cassell, J., Bickmore, T., Billinghurst, M., Campbell, L., Chang, K., Vilhjalmsson, H., & Yan, H. (1999). Embodiment in Conversational Interfaces: Rea, *Proceedings of International Conference on Human Factors in Computing Systems*, 520-527.
- Crabtree, A., Benford, S., Rodden, T., Greenhalgh, C., Flintham, M., Anastasi, R., Drozd, A., Adams, M., Row-Farr, J., Tandavanitj, N., & Steed, A. (2004). Orchestrating a Mixed Reality Game 'On the Ground'. *Proceedings of International Conference on Human Factors in Computing Systems*, 391-398.
- Greenspan, S., Goldberg, D., Weimer, D., & Basso, A. (2000). Interpersonal Trust and Common Ground in Electronically Mediated Communication. *Proceedings of Computer Supported Cooperative Work*, 251-260.
- Guyot, P., Drogoul, A., & Lemaitre, C. (2005). Using Emergence in Participatory Simulations to Design Multiagent Systems, *Proceedings of International Conference on Autonomous Agents and Multiagent Systems*, 199-203.
- Helbing, D., Farkas, I. J., & Vicsek, T. (2000). Simulating Dynamical Features of Escape Panic. *Nature*, 407(6803), 487-490.
- Ishida, T., Nakajima, Y., Murakami, Y., & Nakanishi, H. (2007). Augmented Experiment: Participatory Design with Multiagent Simulation. *Proceedings of International Joint Conference on Artificial Intelligence*, 1341-1346.
- Ishida, T. (2002a). Digital City Kyoto: Social Information Infrastructure for Everyday Life. *Communications of the ACM*, 45(7), 76-81.
- Ishida, T. (2002b). Q: A Scenario Description Language for Interactive Agents. *IEEE Computer*, 35(11), 54-59.
- Kidd, C. D., Orr, R., Abowd, G. D., Atkeson, C. G., Essa, I. A., MacIntyre, B., Mynatt, E. D., Starner, T., & Newstetter, W. (1999). The Aware Home: A Living Laboratory for Ubiquitous Computing Research. *Proceedings of International Workshop on Cooperative Buildings*, 191-198.
- Macavinta, C. Digital Actors in Rings Can Think. <http://www.wired.com/entertainment/music/news/2002/12/56778>, 2002.
- Murakami, Y., Ishida, T., Kawasoe, T., & Hishiyama, R. (2003). Scenario Description for Multiagent Simulation. *Proceedings of International Conference on Autonomous Agents and Multiagent Systems*, 369-376.
- Nagao, K., & Takeuchi, A. (1994). Social Interaction: Multimodal Conversation with Social Agents. *Proceedings of AAAI Conference on Artificial Intelligence*, 22-28.
- Nakamura, T., & Ishiguro, H. (2002). Automatic 2D Map Construction using a Special Catadioptric Sensor. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 196-201.
- Nakanishi, H., Koizumi, S., Ishida, T., & Ito, H. (2004). Transcendent Communication: Location-Based Guidance for Large-Scale Public Spaces. *Proceedings of International Conference on Human Factors in Computing Systems*, 655-662.
- Nakanishi, H., & Ishida, T. (2004). FreeWalk/Q: Social Interaction Platform in Virtual Space. *Proceedings of ACM Symposium on Virtual Reality*

Software and Technology, 97-104.

Noda, I., & Stone, P. (2003). The RoboCup Soccer Server and CMUnited Clients: Implemented Infrastructure for MAS Research, *Autonomous Agents and Multi-Agent Systems*, 7(1-2), 101-120.

Okada, M., Tarumi, H., Yoshimura, T., & Moriya, K. (2001). Collaborative Environmental Education Using Distributed Virtual Environment Accessible from Real and Virtual Worlds. *ACM SIGAPP Applied Computing Review*, 9(1), 15-21.

Okazaki, S., & Matsushita, S. (1993). A Study of Simulation Model for Pedestrian Movement with Evacuation and Queuing. *Proceedings of International Conference on Engineering for Crowd Safety*, 271-280.

Rickel, J., Marsella, S., Gratch, J., Hill, R., Traum, D. R., & Swartout, W. R. (2002). Toward a New Generation of Virtual Humans for Interactive Experiences, *IEEE Intelligent Systems*, 17(4), 32-38.

Sugiman T., & Misumi J. (1988). Development of a New Evacuation Method for Emergencies: Control of Collective Behavior by Emergent Small Groups. *Journal of Applied Psychology*, 73(1), 3-10.

Sumi, Y., & Mase, K. (2001). Digital Assistant for Supporting Conference Participants: An Attempt to Combine Mobile, Ubiquitous and Web Computing. *Proceedings of International Conference on Ubiquitous Computing*, LNCS2201, 156-175.

Tarumi, H., Morishita, K., Ito Y., & Kambayashi, Y. (2000). Communication through Virtual Active

Objects Overlaid onto the Real World. *Proceedings of International Conference on Collaborative Virtual Environments*, 155-164.

Tsutsuguchi, K., Shimada, S., Suenaga, Y., Sonehara, N., & Ohtsuka, S. (2000). Human Walking Animation based on Foot Reaction Force in the Three-dimensional Virtual World. *Journal of Visualization and Computer Animation*, 11(1), 3-16.

KEY TERMS

Augmented reality: overlapping virtual objects with a physical environment in order to provide additional information to people in the environment

Avatar: a graphically or physically embodied representation of a human user for virtual or augmented environments

Augmented experiment: an experiment in which virtual subjects participate

Participatory simulation: a simulation which interacts with human participants

Pedestrian mode: a model for determining the direction and velocity of human walking

Gait mode: a model for moving human legs

Multi-agent interaction model: a model for deciding next behavior based on perceptual information of other agents

augmented experiment successfully affects route selection of the participants.

マルチエージェントシミュレーションを用いた 避難誘導実験の拡張

中島 悠^{†1} 植名 宏徳^{†1,*1} 服部 宏充^{†1}

八榎 博史^{†2} 石田 亨^{†1}

避難誘導システムやカーナビゲーションシステムに代表される大規模な社会情報システムを開発する場合、利用者全体の移動効率のような群集視点の分析と群衆中の大量の被験者の振舞いのような個人視点の分析が求められる。しかし、これらの調査のために、大量的被験者を用いた実験をすることは困難である。本研究では、大規模な社会情報システムの開発支援を目標として、大規模社会情報システムを分析する環境の設計と実験問題への適用を行った。大規模マルチエージェントシミュレーション技術とGPS携帯電話を利用することで、小規模な被験者実験を拡張する環境を設計した。さらに、避難誘導システムを題材に選び、分析環境の実装と被験者実験を行った。実験後のインタビューにより、避難中の経路選択という問題において、仮想的な群衆を被験者に見せることで被験者の意志決定に影響を与えることを確認した。

Augmentation of Experiment in Evacuation Navigation with Multiagent Simulation

YUU NAKAJIMA,^{†1} HIRONORI SHINA,^{†1,*1}
HIROMITSU HATTORI,^{†1} HIROFUMI YAMAKI^{†2}
and TORU ISHIDA^{†1}

社会情報システムの双方が求められる。どちらの分析においても、大人数がシステムを利用している状況が必要となる。

社会情報システムは人を含むシステムであるため、そのテストでは大規模な被験者実験を行うことが望ましい。しかし、都市という広大な空間で、大量の被験者を用いた実験を行うことはコストが大きい。また一方で、大量の被験者を用意する代わりに、利用者をモデル化したエージェントを用いたシミュレーションにより分析する方法もある。しかし、利用者の完全なモデルを実装することは非常に困難である。本研究において、我々は、大規模な社会情報システムの開発支援を目標として、以下の2つの課題に取り組んだ。

- 大規模社会情報システムを分析する環境の設計

The ubiquitous environment enables us to build socially embedded systems such as evacuation guidance system and car navigation system. In developing such a system, it is necessary to estimate the user behavior in a crowd and the crowd behavior. However, it is difficult to perform tests on such a system given the large number of human subjects. In this research, we built a framework of augmented experiment in evacuation navigation system and we execute a practical experiment. We built the environment that show a virtual crowd to human subjects based on GPS-capable mobile phones and agent technology. We carry out an augmented experiment using the system and let participants of a small experiment feel in a crowd. The interview of experiment shows that the

1.はじめに

現在の日本では、GPSやRFIDのような測位デバイス、WLANやBluetoothといったネットワーク機器が普及し、ユビキタスな情報環境が進展してきている。この環境を利用することで、現在位置や周辺環境などの個人がおかれた状況、年齢や職業などの個人が有する属性に合わせたサービスを提供する社会情報システムの実現が可能になる。たとえば、経路情報共有に基づくナビゲーションシステム¹⁾やオンラインマンドバスシステム²⁾などの研究がされている。その流れの中で、新しい社会情報システムを分析、評価するための環境に対するニーズは高まっている。

社会情報システムの特徴は、広範囲に存在する利用者が、情報や資源を共有することで互いに影響を及ぼしながらシステムを利用する。大規模な社会情報システムを評価する場合、利用者全体の移動効率のような群集視点の分析と群衆中の利用者の振舞いのような個人視点の分析が求められる。どちらの分析においても、大人数がシステムを利用している状況が必要となる。

社会情報システムは人を含むシステムであるため、そのテストでは大規模な被験者実験を行うことが望ましい。しかし、都市という広大な空間で、大量の被験者を用いた実験を行うことはコストが大きい。また一方で、大量の被験者を用意する代わりに、利用者をモデル化したエージェントを用いたシミュレーションにより分析する方法もある。しかし、利用者の完全なモデルを実装することは非常に困難である。本研究において、我々は、大規模な社会情報システムの開発支援を目標として、以下の2つの課題に取り組んだ。

- 大規模社会情報システムを分析する環境の設計

大規模社会情報システムの分析には、少數の被験者に対して、群衆中におけるシステム利用の体験を与える環境が求められる。我々はその環境を拡張実験のアプローチに基づ

^{†1} 京都大学大学院情報学研究科社会情報学専攻
Department of Social Informatics, Kyoto University

^{†2} 名古屋大学情報連携基盤センター
Information Technology Center, Nagoya University

*1 現在、富士通株式会社
Presently with Fujitsu Limited

- ・ 実問題への適用
いて設計する。

大規模社会情報システムを分析するための環境を実問題に適用する・避難誘導システムを分析する環境を実装し、その環境を用いて少人数の被験者による実験を拡張する。以下、2章、3章では、大規模社会情報システムを分析する手法とその環境の設計について説明する。4章では、分析する対象として取り上げた大規模避難誘導システムについて述べる。5章では、GPS 携帯電話とマルチエージェントシミュレーションを組み合わせた分析環境の実装について説明する。6章では、大規模誘導システムの実験に適用した例を示す。

2. 大規模社会情報システムの分析手法

大規模社会情報システムの分析において、大量の被験者を用いた実験の実施は非常に困難である。我々はこの問題に対して、拡張実験と名付けた新しいシステムの分析手法を提案している³⁾。

拡張実験とは、利用者をエージェントによって代替することで、実験に参加する人間の被験者の数を抑えうえで、多数の利用者が当該システムを使用する状況を擬似的に再現するアプローチである。

拡張実験では、現実空間の被験者実験と並行して仮想空間でマルチエージェントシミュレーションが実施される。並行してシミュレーションを行う目的は、現実空間の実験だけでは不足する被験者数を補うことである。被験者は、所有する端末に仮想空間の状況を受け取ることで、あたかも多数の人間と同時にシステムを利用しているかのように実験に参加する。一方で、実験実施者は、現実空間と仮想空間が統合された状況を観測することで、群衆全体の動きを分析する。そうすることで、大量の被験者による実証実験を行うことなく、システムの問題点を洗い出す。

シミュレーションを利用してある仮想的な状況における人間の振舞いを調べる他の手法として、参加型シミュレーションがあげられる^{4),5)}。参加型シミュレーションとは、マルチエージェントシミュレーション上の一端のエージェントを人間の操作するアバタと置きかえ、シミュレーション上に実際の人間の行動を取り込むものである。人間はテレビゲームをするように、ジョイスティックやキーボードなどの入力機器でエージェントを操作する。参加型シミュレーションでは、被験者は人工的に作られた仮想空間で観測および行動する。それに対し、拡張実験では、被験者は現実空間で観測および行動する。この対象空間の違いが2つの手法の大きな相違点である。拡張実験の利点は、実際のシステムの利用者

が、人工的な空間ではなく実世界において、分析対象となるシステムを実際に使用する実験を行えることである。

3. 大規模社会情報システムの分析環境の設計

我々の考える大規模社会情報システムとは、屋外の広範囲に存在する利用者が互いに影響を及ぼしながらサービスを利用するシステムである。ここでは大規模社会情報システムの代表例として、都市における歩行者誘導システムを考える。

大規模な社会情報システムに拡張実験のアプローチを適用する際の課題は、現実空間の情報を仮想空間へ取り込む方法と、仮想空間の情報を現実空間へ取り込む方法である。拡張実験のコンセプトに基づいて設計された、大規模社会情報システムを分析する環境の構成を図1に示す。

3.1 仮想空間への情報の取り込み

仮想空間では、大規模エージェントプラットフォーム上で、システムの仮想的なユーザを再現するシミュレーションが実施される。このシミュレーション上のエージェントには、分析対象のシステムのユーザの振舞いをモデル化したものが実装される。

仮想空間で行われるシミュレーションに被験者の振舞いを取り込むため、センサにより取り込んだ位置情報に基づきアバタを生成する。屋内の限られた範囲における実験⁶⁾では、センサとしてカメラを使用できる。しかし、屋外の広域な範囲における実験では、実験領域すべてをカバーするだけのカメラを設置することはできない。そこで、測位センサとしてGPS を用いることとした。

実験実施者は、群衆視点からの分析をするために、利用者全体の行動を観測したい。そのためには、実世界の少数の被験者の行動と仮想空間のエージェント群の行動を同時に扱える必要がある。そこで、現実空間の情報と仮想空間の情報を重畠し、超越的な視点から観測を行いうインターフェースを用いた⁷⁾。図2に示すような超越型のインターフェースを用いることで、実験実施者は現実空間の情報を取り込んだ仮想空間を俯瞰しながら、被験者とユーザエージェントを区別することなく扱うことができる。

3.2 現実空間への情報の取り込み

拡張実験を実施するには、仮想の公共空間の中で活動するエージェントの振舞いを現実空間にいる被験者に伝えなければならない。そこで問題となるのは、仮想空間の情報を適切なリアリティと速度をもって現実空間の被験者に伝える方法である。

たとえば、透過型ヘッドマウントディスプレイに仮想空間の状況を表示する方法が考えら

3 マルチエージェントシミュレーションを用いた避難誘導実験の拡張

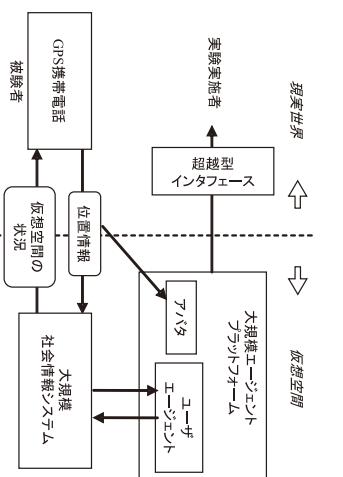


図 1 大規模社会情報システムの分析環境の構成
Environment for analyzing large-scale
socially embedded system.

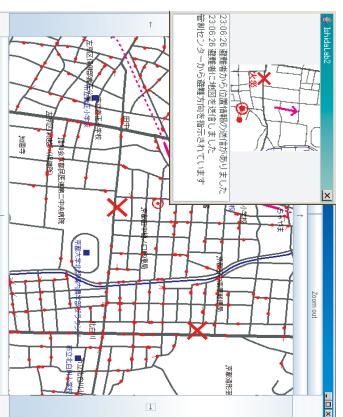


図 2 現実空間の情報と仮想空間の情報を重畠する
インターフェース
Fig. 2 Interface layering information of
real world and virtual space.

れる。しかし、街中を移動する被験者にそれを装着させるのは危険であるし、費用の面でも手間の面でも高いコストがかかる。これは、開発段階において小規模な実験でシステムの問題点を探すという目的にはそぐわない。そこで、我々は、仮想空間情報の伝達手段として携帯電話を選択した。

4. 大規模避難誘導システム

本研究の分析対象として、都市災害時に用いるGPS携帯電話を利用した大規模避難誘導システムを選んだ。都市規模の大規模な誘導となると同じ情報をブロードキャストするだけでは、すべての利用者にとって有用な情報を配信することはできない。そこで、個々の利用者に対して誘導エージェントを割り当てることで、システムに集約されている情報を個別化して利用者に配信するシステムを想定した。実装した大規模避難誘導システムの構成を図3に示す。

環境データベースに蓄積される情報は、指定避難場所を含む地図情報、都市内に設置されたセンサなどで収集される被害情報、避難者の位置情報である。地図情報は、国土地理院発行の空間データ基盤数値地図(1/25000)を用いた。

誘導エージェントは、利用者ごとに1つずつ作成され、対応する利用者に対して誘導情報の配信を行う。はじめに誘導エージェントは、GPS経由で利用者の位置情報を取得する。続いて、誘導エージェントは、指定避難所の情報を環境データベースから取り出し、最寄り

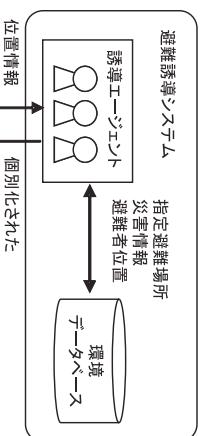


図 3 大規模避難誘導システムの概観
Overview of large-scale evacuation
guidance system.

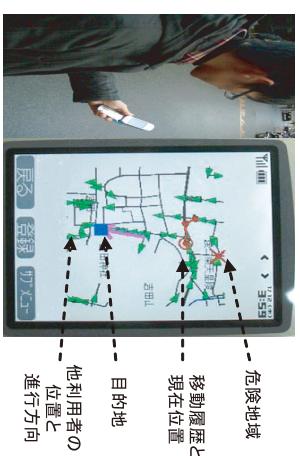


図 4 避難誘導システムで配信される誘導情報
Navigation information provided
by evacuation guidance system.

の避難所を検索する。誘導エージェントは、利用者の現在地付近の誘導地図を作成し、それを利用者に配信する。誘導地図には、目的とする避難場所の位置、避難場所への進行方向、火災などの危険地域、自分の現在位置、移動履歴、他者の位置と進行方向が表示される(図4)。

5. 避難誘導システムの分析環境

大規模避難誘導システムを対象として、少人数の被験者実験をマルチエージェントシミュレーションにより拡張するためのシステムを構築した。その構成を図5に示す。

現実空間の被験者を仮想空間にアバタとして再現するために、GPSにより位置情報を取得した。仮想空間の状況を伝達するために、それぞれの被験者が持つ携帯電話のブラウザ機能を用いた。誘導エージェントが、周囲の避難者エージェントの状況を表示することで、避難者に対して仮想的な群衆の状況を伝える。これにより、避難者は大群衆と同時にシステムを利用するような環境で実験に参加する。また、実験実施者は、仮想空間と現実空間を重ね合わせた情報を提供する観測モニタを通じて、拡張された実験環境を観測する。以下では、図5の内容を詳細に説明する。

5.1 避難者エージェントのプラットフォーム

誘導システムの仮想的なユーザを生成するために、大規模マルチエージェントプラットフォーム上に、避難者をモデル化したエージェントが動作するシミュレータを実装する。このプラットフォームとしてCaribbean/Qを用いた⁸⁾。このプラットフォームは、シナリオ

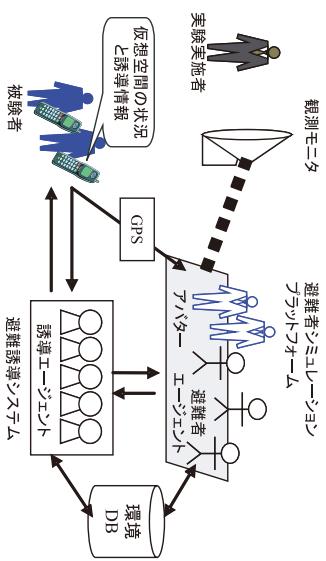


Fig. 5 Environment for analyzing large-scale evacuation guidance system.

記述言語 Q と大規模エージェントサーバ Caribbean を組み合わせたものである。

Q は、エージェントの内部メカニズムには言及せず、エージェントと外界とのインタラクションプロトコルを、シナリオとして記述するための言語である^⑨。現実空間での実験と並行して行われるシミュレーションでは、エージェントに適切な人間行動のモデルを与えることが重要となる。そのモデル化において、人間行動の中身を記述する旧来のエージェント記述方式ではなく、 Q のように人間と外部世界とのインタラクションプロトコルの様態を、そのままシナリオとして記述する方式が効果的であることが示されている^⑩。

Caribbean は、イベント駆動オブジェクトとしてエージェントを管理し、数十万のエージェントを持つ大規模エージェントサーバである^⑪。都市上の市民のシミュレーションを行うためには、その規模に十分なスケーラビリティの確保が必要である。

Caribbean/ Q の特長は、シナリオ記述によりエージェントを制御できること、数十万のエージェントを管理できるスケーラビリティを備えていることである。この特長のため、仮想群衆を再現するマルチエージェントシミュレーションのプラットフォームに Caribbean/ Q を選択した。

5.2 避難者エージェントの生成

大規模エージェントプラットフォーム上に実装される避難者エージェントは、 Q により記述されたシナリオに基づいて、避難誘導システムの仮想的なユーザとして振る舞う。避難者エージェントに与えたシナリオを図 6 に示す。

避難者エージェントは、定期的に位置情報を誘導エージェントに送信する。誘導エージェントは、受け取った位置情報に応じて誘導地図を作成し、それを避難者エージェントに送信

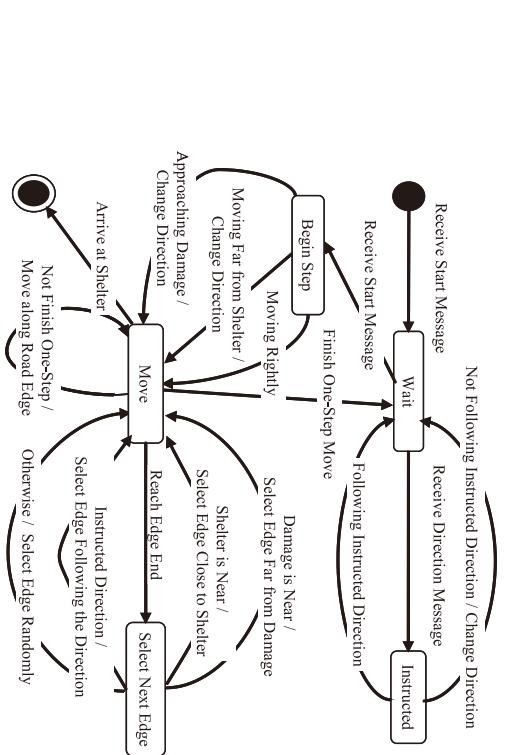


Fig. 6 避難者エージェントのシナリオ

する。避難者エージェントは、その誘導地図を見ながら仮想空間を移動する。

避難者エージェントは、環境データベースから道路データを読み出し、その上を移動する。道路のデータは道路節点と節点どうしがつなぐ道路区間から構成される。避難者エージェントは、誘導地図に記された目的地、危険地域を考慮して、道路上沿って避難を行う。避難者エージェントは、1回の移動ステップで以下のように行動する。

- (1) 避難者エージェントは目的地を決定する。前回の行動が終了してから追加された被害情報によって、現在の目的地が不適切なものになっていないかを確認する。「危険地域を通過せずに到達できる避難場所の中で、現在地から一番近いものを選ぶ」というルールに基づいて目的地の選択を行う。
- (2) 避難者エージェントは次の道路節点を決定する。「危険地域に近づかず」、目的地に近づける道路節点を選ぶ」というルールに基づいて道路節点の選択を行う。
- (3) 避難者エージェントは移動を行う。1回の移動ステップにおける避難者エージェントの移動距離は前回移動してから経過した時間によって決定する。

今回実装したシミュレーションでは、避難者エージェント群に単純な単一のシナリオを与えた。消防士や警察などの個人的社会的な役割を反映したシナリオや、「怪我の有無など個

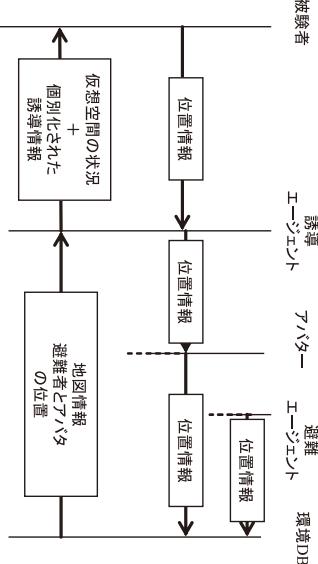


図 7 現実空間情報の仮想空間への取り込みと仮想空間情報の現実空間への伝達の流れ

Fig. 7 Sequence diagram: connecting real world and virtual space.

人のおかれている状況を反映したシナリオを適宜用意することで、より多様な人間が存在する状況のシミュレーションが可能となる。

5.3 実験環境の観測モニタ

3.1節で述べたように、実験実施者に現実空間の情報と仮想空間の情報を提供するために、図2に示される超越型インターフェースを用いた。その観測モニタ上では、実際の被験者と避難者エージェントを区別することなく扱うことができる。つまり、少数の被験者実験であると類似した状況を提供できる。この環境を用いることで、大人数が誘導システムを使用するときに都市で起こることを分析できる。

5.4 現実空間と仮想空間の接続

現実空間の被験者の振る舞いを仮想空間に取り込むために、GPSを使用した。GPSを経由して獲得された現実空間の被験者の位置は、仮想空間上にアバタとして再現される。本システムでは、KDDIが提供するGPSサービスを使用した。

我々は、仮想空間情報の伝達手段として携帯電話を用いた(図4)。携帯電話は仮想空間の情報を伝達する手段としては非常に簡素なものである。携帯電話の小さな画面に一人称視点のリアルな3次元映像を表示したところで没入感は得られないうえに、画像作成や通信のコストが大きくなり仮想空間の情報の伝達速度が遅くなる。そこで、実験課題の遂行に必要な仮想空間の状況のみを被験者に配信することにした。

本実験環境では、誘導情報とともに仮想空間上の避難者エージェントの位置が記された地

図が被験者の携帯電話に伝達される。現実空間の情報と仮想空間の情報の伝達経路を図7に示す。

はじめに、被験者の位置情報が誘導エージェントに送信される。つづいて、誘導エージェントはアバタの位置情報を更新する。仮想空間にはアバタの位置情報と避難者エージェントの位置情報が集約される。誘導エージェントは、環境データベースから地図情報と群衆の情報を取得し、その地点に応じた誘導情報と仮想空間の環境情報を被験者に伝える。被験者はその情報を考慮して、意志決定を行う。このようにして、避難者は群衆と同時にシステムを利用するような環境で実験に参加することができる。

6. 実験

5章の分析環境を用いて、大規模避難誘導システムの被験者実験を拡張実験のコンセプトに基づいて実施した。この実験の目的は、仮想群衆の提示が被験者に与える影響を調べることである。

6.1 実験設定

本実験で想定したシナリオは、都市に地震が発生し断続的に火災などの二次災害が発生する状況で、市民が誘導システムを使いながら指定の避難所まで移動するというものである。これは、刻々と変わる環境下で、避難者が素早く安全な場所への移動を目指す状況を想定したものである。

実験では、京都大学周辺の1辺が4 kmの矩形領域を対象領域とした。対象領域中に、5力所の避難所を設定した。現実空間の実験には10人程度(1回目13人、2回目10人)の人間の被験者が参加した。現実空間の被験者実験と並行して実行されるシミュレーションでは、3,000体の避難者エージェントからなる仮想群衆を生成した。

被験者に配信する地図には、誘導情報とともに仮想空間上の避難者エージェントの位置を描画した。二次災害は、あらかじめ定められた位置に定められたタイミングで発生し、環境データベースに随時書き込まれた。

実験開始から終了までの流れは以下のとおりであった。実験開始とともに初期災害の発生が被験者に通知される。被験者は各々の初期位置から避難を開始する。誘導エージェントは、避難者の位置情報を受け取ると、環境データベースより情報を取得し、避難者に与える誘導地図を作成して送信する。被験者はその情報を参照しながら、避難場所へ向かう。すべての被験者が広域指定避難所に到達したときを実験の終了とした。実験後には、被験者の主観的な評価を得るためにアンケートとインタビューを行った。

6 マルチエージェントシミュレーションを用いた避難誘導実験の拡張

6.2 実験参加者のタスク

被験者に与えたタスクは、各自が持つGPS携帯電話により避難誘導システムを利用して避難所に向かうことである。避難者は、それぞれの避難開始位置からシステムに指示される避難所まで移動した。この実験において、被験者は、1分間隔の自動更新と手動による地図情報の更新ができた。

事前説明において、被験者に実験シナリオを伝えた。また、被害の発生地点を避けた移动するよう教示した。避難者エージェントは「仮想の避難者である」とだけ説明し、避難者エージェントについていたり、避けたりするような特別な説明は与えなかった。実験の前には、システムの使い方の説明をし、数分の練習をさせた。

実験実施者は、超越型のモニタを通じて、被験者と避難者エージェントが混在した仮想空間を観察した。被験者が実験区域から外れるなどのトラブルがあったとき、実験実施者は、そのモニタを通じてメールにより被験者の支援をした。また、実験実施者は、モニタ上で被験者や避難者エージェントを選択することで、それが参照している誘導地図を確認することができた。

6.3 アンケート結果

アンケートは、避難誘導システムに関する6問（信頼性に関する2問、有用性に関する2問、簡便性に関する2問）と拡張実験に関する6問（仮想群衆の影響に関する3問、実験全体に関する3問）、実験意図の察知を防ぐための5問の計17問で構成した。実験トラブルによりアンケートに答えられなかつた1人を除く、22人の被験者から回答を集めることができた。以下では、拡張実験に関する6問について説明する。

仮想群衆が被験者に与えた影響を調べるために、避難行動の「巧拙」、避難行動時の「冷静さ」、実験状況の「臨場感」について、仮想群衆が表示されていた場合と非表示であった場合とで比較する質問をした。集計結果を表1に示す。仮想群衆の有無により評価が違った人の割合をみると、仮想群衆の提示は、「巧拙」「冷静さ」に関しては6割以下、「臨場感」に関しては4割以下の人にしか影響を与えることができなかつたことが分かる。

また、実験全体における行動と状況を調べるために、避難行動の「巧拙」と「冷静さ」、実験状況の「臨場感」に関する質問をした。各質問は、1点から9点までのリカート尺度である。「巧拙」を尋ねた「うまく避難できましたか?」という質問の得点は平均6.0、標準偏差2.3であった。「冷静さ」を尋ねた「落ち着いて行動できましたか?」という質問の得点は平均7.0、標準偏差1.8であった。「臨場感」を尋ねた「避難状況に臨場感が出ていましたか?」という質問の得点は平均3.3、標準偏差1.7であった。

Table 1 Questionnaire about effect of evacuee agent.

	仮想群衆 表示	仮想群衆 非表示	特に 違いなし	影響を 受けた割合
どちらがうまく 避難できたか	4人	9人	9人	13/22=59%
どちらの避難に 臨場感があつたか	7人	1人	14人	8/22=36%

表2 避難時の経路選択の基準に関するインタビュー（A,Bが1組目のインタビュー、C,D,Eが2組目のインタビュー）
Table 2 Interview about factors of route selection.

インタビュー：目的的に向かう際に、何を基準に通る道を選びましたか？
被験者A：わかりやすい道と、なるべく最短経路を選ぶようにしました。あと、1回目は、仮想群衆ですか？それがあ歩いてないところを選ぶようにしました。（インタビュー：わかりやすい道とは具体的にどのような道ですか？）大きい道ですね、基本的にには、
被験者B：僕は、最短経路と、さっきの方とは逆なんですが、群衆がいる方がいい道なのかなと思ってそっちを選ぶようにしました。
被験者C：えっと、一番は最短経路らしい道ということと、あとは通れない道、赤いバッテンですね。それと1回目は人の多さです。ごちゃごちゃしている方は面倒くさそうだから少ない方を通ってみようという感じで道を選びました。
被験者D：私も前の方とほとんど同じなんですけど、あと、道の中で自分が通ったことのある道があればそこを通るようにしました。
被験者E：私も最短経路に近いようにということと、あと曲がる角が少ないように、なるべく直線が多くて迷わないようにということと、あとその人の多さというのが気になつたんで、本当に、実際に人がいたらただその後を付いていったと思うんですけど、そういうわけじゃなくて単にああ、人がいるという感じだったんで、なるべくそういう人の行きそうな太い道を避けて、さっさと逃げれそうな細い道で行きました。

6.4 インタビュー結果

実験後のインタビューでは、「システムを用いてスマーズに避難できたか」、「配信した情報は見やすかったか」、「経路選択の際にどのような情報を参考にしていたか」について尋ねる。「巧拙」を尋ねた「うまく避難できましたか?」について説明する（表2）。

インタビューを受けたすべての被験者は、経路選択に際して、他の避難者の動きを意識したと回答した。被験者A,C,Dは、避難者が少なく表示された経路を選び、被験者Bは

まったく逆に、避難者が多く表示された経路を選択したと回答した。被験者Eは、実際に人が見えている場合と仮想群衆が見えている場合で違う行動をとったと発言している。

被験者Eの発言は、本実験における経路選択は、実際の大規模な被験者実験における経路選択とは異なる方針であったことを意味している。しかし、このインタビュー結果は、提案する実験環境を用いることで、少人数の被験者らに大群衆の中でのシステム利用を体験させ、その状況下における意志決定を分析できることを示している。

6.5 考 察

拡張実験の1つの目的は、被験者に群衆中のシステムの体験を想起させることである。

6.3節のアンケートは、仮想群衆の提示は、避難行動の「巧拙」や「冷静さ」、実験状況の「臨場感」に関しては、3分の1から3分の2程度の人にしか影響を与えるなかったという結果だった。それに対して、6.4節のインタビュー結果は、仮想群衆の提示が被験者の経路選択に影響を与えることを示すものだった。つまり、避難中の経路選択という問題において、仮想群衆を被験者に見せることにより、実験の臨場感を増加させることはできなかつたが、被験者の意志決定には影響を与えたといえる。

拡張実験のもう1つの目的は、超越型インタフェースを用いて、群衆全体を分析する視点から実験を観測することである。このとき、仮想空間上を行動する1体1体のエージェントが被験者の代理としての性質を十分に備えていなければ、仮想群衆の全体の行動結果を解析する意味は乏しい。

インタビュー結果で得られたように避難者は周辺の人間の動きを意識して行動するはずであるが、今回実装したエージェントは周辺の人間の振舞いを考慮して経路を選択する能力を備えていなかった。そのため、群衆の全体の振舞いは、実際の大規模実験とはかけ離れたものとなっていると予想される。しかし、群衆中でどう振る舞うか分からぬユーチャのモデルを事前に準備することは困難である。そこで、拡張実験により得られた行動ログやインタビュー結果を用いることで、順次、エージェントモデルを精緻していくプロセスが考えられる³⁾。

通常の狭い範囲の被験者実験ではカメラにより実験の一部始終を記録可能である。また、仮想空間で行われるシミュレーションではアバタの操作系列を詳細に記録可能である。それに対して、屋外の社会情報システムの実験では、その規模の大きさのため被験者の行動を記録するコストが高くなるという問題が起こる。今後は、限られたログを用いて、エージェントモデルを精緻する方法を検討する必要がある。

7. おわりに

人を含んだ大規模な社会情報システムを開発する際には、利用者全体の移動効率のような群集視点の分析と群衆中の利用者の振舞いのような個人視点の分析が必要である。しかし、大人数による被験者実験のコストは非常に大きい。我々は、少人数の被験者により大規模な社会情報システムの分析をすること目標として、以下の2つの課題に取り組んだ。

• 大規模社会情報システムを分析する環境の設計

大規模社会情報システムを分析する環境として、現在広く普及しているGPS携帯端末とエージェント技術を用いることで、仮想空間のマルチエージェントシミュレーションと現実空間の実験を結びつける環境を設計した。

• 実問題への適用

大規模避難誘導システムの実験を拡張できる環境を実装した。その環境を使って実験を行い、仮想群衆の提示により被験者の経路選択に影響を与えることを確認した。

大規模な社会実験を、多数の人を動員することなく実現するためには、群衆全体の行動分析ができること、実験参加者に群衆中の行動を想起させることの2つが必要である。

群衆全体の行動の分析には、より再現性の高いシミュレーションが必要である。そのためには、群衆中で行動するエージェントのモデルをより正確にする必要がある。拡張実験で得られる結果を利用してエージェントモデルを精緻することで、より効果的な拡張実験の環境を作り出すことができる。

避難誘導の実験において、少数の被験者実験にもかかわらず、被験者に群衆が存在する状況下における意志決定を想起させることができた。今後の課題は、実験に参加する被験者とエージェントの割合を変えた実験を行うことで、人間の群衆とエージェントからなる仮想群衆の差を明確にすることである。人間の被験者だけが参加する統制実験の結果とエージェントを加えた拡張実験の結果を比較することで、拡張実験の効果を定量的に評価したい。

謝辞 KDDI研究所森川大輔氏の実験協力に感謝する。本研究は、日本学術振興会科学研究費基盤研究(A)(18200009, 2006-2008), 総務省戦略的情報通信研究開発推進制度(2005-2007)の補助を受けた。

参 考 文 献

- Yamashita, T., Izumi, K., Kurumatani, K. and Nakashima, H.: Smooth traffic flow with a cooperative car navigation system, *AAMAS-05*, pp.478-485 (2005).

8 マルチエージェントシミュレーションを用いた避難誘導実験の拡張

- 2) Noda, I., Ohta, M., Shinoda, K., Kumada, Y. and Nakashima, H.: Evaluation of Usability of Dial-a-Ride Systems by Social Simulation, Vol.2927, pp.167-181 (2004).
- 3) Ishida, T., Nakajima, Y., Murakami, Y. and Nakanishi, H.: Augmented Experiment: Participatory Design with Multiagent Simulation, *IJCAI-07*, pp.1341-1346 (2007).
- 4) Guyot, P., Drogoul, A. and Lemaitre, C.: Using emergence in participatory simulations to design multi-agent systems, *AAMAS-05*, pp.199-203 (2005).
- 5) Drogoul, A., Vanberge, D. and Meurisse, T.: Multi-agent Based Simulation: Where Are the Agents?, *MABS-02*, pp.1-15 (2002).
- 6) Ito, H., Nakanishi, H. and Ishida, T.: An Analysis of Location-based Remote Instructions by Transcendent Communication, *IPSJ Journal*, Vol.48, No.3, pp.125-132 (2007).
- 7) Nakanishi, H., Koizumi, S., Ishida, T. and Ito, H.: Transcendent communication: Location-based guidance for large-scale public spaces, *CHI-04*, pp.655-662 (2004).
- 8) Nakajima, Y., Shima, H., Yamane, S., Yamaki, H. and Ishida, T.: Caribbean/Q: A Massively Multi-Agent Platform with Scenario Description Language, *SKG-06*, p.26 (2006).
- 9) Ishida, T.: Q: A Scenario Description Language for Interactive Agents, *IEEE Computer*, Vol.35, No.11, pp.42-47 (2002).
- 10) Murakami, Y., Ishida, T., Kawasee, T., et al.: Scenario Description for Multi-Agent Simulation, *AAMAS-03*, pp.369-376 (2003).
- 11) Yamamoto, G. and Tai, H.: Performance evaluation of an agent server capable of hosting large numbers of agents, *AGENTS-01*, pp.363-369 (2001).

中島 悠



平成 18 年京都大学大学院情報学研究科修士課程修了。現在、同博士課程在学中。平成 18 年日本学術振興会特別研究員 (DC1)。大規模マルチエージェントシステム、マルチエージェントシミュレーションに关心を持つ。

椎名 宏徳
平成 18 年京都大学大学院情報学研究科修士課程修了。同年富士通株式会社入社。大規模ナビゲーションシステムに興味を持つ。



服部 宏充 (正会員)

平成 16 年名古屋工業大学大学院博士課程修了。博士 (工学)。平成 16 年日本学術振興会特別研究員 (PD)。リバーブル大学、マサチューセッツ工科大学客員研究員を経て、平成 19 年京都大学大学院情報学研究科助教。マルチエージェントシステム、意思決定支援に興味を持つ。



八榎 博史 (正会員)

平成 11 年京都大学大学院情報学研究科博士課程修了。博士 (情報学)。京都大学大学院情報学研究科助手、同講師を経て、現在、名古屋大学情報連携基盤センター准教授。マルチエージェントシステムおよびネットワーク応用に興味を持つ。



石田 亨 (フェロー)

昭和 53 年京都大学大学院情報工学専攻修士課程修了、日本電信電話公社電気通信研究所入所。工学博士。IEEE フェロー。情報処理学会フェロー。現在、京都大学大学院情報学研究科教授。マルチエージェントシステム、Semantic Web 技術に興心。デジタルシティ、異文化コラボレーションプロジェクトを推進。



論 文

大規模マルチエージェントシステムにおけるエージェント配置

宮田 直輝[†] 石田 亨[†]

Placing Agents in Massively Multi-Agent Systems

Naoki MIYATA[†] and Toru ISHIDA[†]

あらまし 近年、多数のエージェントを扱う大規模なマルチエージェントシステムが開発されている。大規模マルチエージェントシステムは、分散化による利点を得るために、複数のエージェントサーバを接続するアプローチが取られる。エージェントが複数のサーバに分散する場合、システムの性能を効率的に改善するためには、エージェントの計算負荷とエージェント間のインタラクションに注目してエージェントを配置する必要がある。本稿では、互いにインタラクションを頻繁に行うエージェント集合であるコミュニティに対する評価を基に、その配置を決定する Community-based Load Balancing (CLB) アルゴリズムを提案する。CLB は解の精度を保証することができないため、本稿ではシミュレーションを行った。そして CLB は先行研究と比べて、計算コストが大きくなるが、エージェントのインタラクションが構成するネットワークが正則、または小世界ネットワークの場合にインタラクションコストが小さな配置を行うことを示した。

キーワード 大規模マルチエージェントシステム、エージェント配置

1. まえがき

近年、多数のエージェントを扱う大規模マルチエージェントシステムが開発されている[1]。大規模マルチエージェントシステムを構築することで、より多くの利用者にサービスを提供するシステムを構築することや、より複雑な系を対象とするシミュレーションを行うことができる。例えば、数百万規模の交通をマルチエージェントシステムにより再現するシミュレーション[2] や、多数の利用者を想定したサービスをマルチエージェントシステムによって提供するシステム[3] の開発が行われている。こうした大規模なマルチエージェントシステムを構築する際に生じる問題を解決するための研究が、現在進められている[4]。

大規模マルチエージェントシステムにおける問題点の 1 つは、エージェント数の増加に従ってシステムの性能が低下することである。システムが扱うエージェントの数が増加すると、エージェントの計算負荷や利用者からのアクセス負荷が増加し、リアルタイムなサービスを提供することや、効率的にシミュレーション

を行うことが困難になる。

大規模マルチエージェントシステムの性能が低下する問題に対して、エージェントシステムを分散化するアプローチをとることができる。複数のサーバからなる分散マルチエージェントシステムを構築することで、システムのリソースを増加させることができる。その結果、システムのリソース不足による性能の低下を解決することができる。こうしたシステムの分散配置は、分散 OS やワークステーションのクラスタ化に適用されており、その有効性が示されている[5]。

しかし、サーバを分散配置する場合、分散 OS などに適用してきた計算負荷に基づく手法では不十分である。エージェントが分散配置されることで、インタラクションがサーバ間で発生し、システムのインタラクションコストは増加する。インタラクションが頻繁に発生する場合、このようなインタラクションコストの増加を無視することはできない。従って、エージェントの計算負荷とインタラクションコストに基づいて、その配置を決定しなければならない。

本稿では、都市規模の交通シミュレーションや誘導システムといった、数千体、数万体ものエージェントを扱う大規模な分散マルチエージェントシステムにおいて、計算負荷を分散し、通信コストを小さくするよ

[†] 京都大学社会情報学専攻 〒 606-8501 京都市左京区吉田本町
Department of Social Informatics, Kyoto University
Yoshida-Honmachi, Kyoto 606-8501, Japan

うなエージェント配置を決定する手法を提案する。本研究では上記の問題を、各サーバの計算負荷が閾値よりも小さく、サーバ間で発生するインタラクションコストを最小にするようなエージェントの配置を求める、エージェント配置問題として定式化する。そして、定式化したエージェント配置問題の最適解を求めるることは困難であるため、エージェント配置問題に対する近似解を求める手法を提案する。本研究では、提案手法によるシステムのインタラクションコストの変化を検証するために、シミュレーションを行った。

2. 関連研究

分散環境におけるプロセスまたはエージェントの配置手法は、High Performance Computing(HPC) やモバイルエージェント、マルチエージェントの分野で提案されている。これらの手法は、プロセスやエージェントの計算負荷と通信コストに基づいてその配置を決定することで、システムの性能を改善する手法を提案している。

文献 [6] では、分散配置されたノードに対して、あるノードが保持する多数のタスクを、システムの性能が最大となるように割り当てる手法が提案されている。分散されたタスクの負荷は、そのデータ量と分散方法によって定まるため、静的に解析することで適当な配置を行うことができる。一方で、マルチエージェントシステムでは、エージェントが自律的に動作するため、その振る舞いは動的に変化し、あらかじめ予測することは困難である。従って、システムの負荷が動的に変化する中で、システムの性能を改善するように配置を決定しなければならない。

文献 [7], [8] では、エージェントがタスクを実行する際の通信コストを最小にするように、移動を行う手法が提案されている。これらの手法は主に 1 体のエージェントに注目し、その通信コストが小さくなるように移動を決定する。しかし、エージェント間でインタラクションが発生するマルチエージェントシステムにおいては、あるエージェントの移動は、他のエージェントの通信コストに影響を与える。従って、複数のエージェントの通信コストを考慮してその配置を決定しなければならない。

文献 [9], [10], [11] では、エージェントの計算負荷と通信コストに基づいて、エージェントの配置を決定する手法が提案されている。これらの手法に共通する特徴として、エージェント単体の評価に基づいて、その

移動と移動先を決定していることが挙げられる。例えば、[10] が提案する Comet アルゴリズムは、エージェント単体に対する評価を基に、移動するエージェントとその移動先を決定し、過負荷のサーバの計算負荷が閾値よりも小さくなるまで移動を反復する。しかし、4.2 節で述べるように、エージェントの集合に対する評価を基に、その配置を決定することで、よりインタラクションコストを小さくすることができる。

3. エージェント配置問題

本章では、エージェントが複数のサーバに分散する際に考慮すべき点を述べ、エージェント配置問題として定式化を行う。

3.1 分散環境におけるエージェント配置

本研究で考えるマルチエージェントシステムは、ネットワーク上で接続された m 台のサーバに n 体のエージェントが分散配置されているものとする。このマルチエージェントシステム上では、エージェントは必要に応じて相互にインタラクションを行う。互いに異なるサーバに存在するエージェント間のインタラクションは、互いのエージェントが存在するサーバ間でメッセージを交換することで実現可能であるとする。そして、各サーバの性能は同じであり、サーバ間の接続は全て等価であると仮定する。このようなシステムにおいて達成すべき事柄は、以下の 2 点である。

- 各サーバにエージェントの計算負荷を分散する
- サーバ間で発生する通信コストを最小にする

まず、エージェントの計算負荷を各サーバに分散する必要がある。あるサーバに対して負荷が集中することによって性能が低下することを防ぐため、各サーバの性能が大きく低下しないよう、エージェントを分散配置する必要がある。

次に、サーバ間で発生するインタラクションコストを抑える必要がある。あるエージェントが異なるサーバに存在するエージェントに対してインタラクションを行う場合、インタラクションのためのメッセージをサーバ間で通信する必要がある。従って、サーバ間のインタラクションが発生することで、システムにおけるインタラクションコストが増加する。こうしたインタラクションコストの増加を抑えるために、頻繁にインタラクションを行うエージェント同士を同じサーバに配置し、サーバ間で発生するインタラクションの頻度を少なくする必要がある。

エージェント間のインタラクションには、頻繁にそ

の相手と頻度が変化する場合と、変動はあるが、統計的にはその変動が小さい場合が想定される。本研究におけるインタラクションは、統計的には変動が小さい場合を想定する。変動が小さい場合、それまでのインタラクションの傾向から、インタラクションコストが小さくなるような配置を行うことができる。一方で、インタラクションの変動が大きい場合、その変動を予測して配置を決定することが必要である。

3.2 エージェント配置問題の定式化

表1の表記に基づいて、エージェント配置問題を以下のように定式化する。

エージェント集合 A の計算負荷 $W(A)$ は、(1) 式に示すように、エージェント a_j の計算負荷 w_j の総和で与えられるとする。

$$W(A) = \sum_{a_j \in A} w_j \quad (1)$$

また、表記を簡単にするために、エージェント集合間のインタラクションコストとして、(2) 式で定まる $P(A_i, A_j)$ を定義する。 $P(A_i, A_j)$ は 2 つのエージェント集合 A_i, A_j 間のインタラクションコストの総和である。

$$P(A_i, A_j) = \sum_{a_k \in A_i} \sum_{a_l \in A_j} p(a_k, p_l) \quad (2)$$

ここで、サーバ s_1 が過負荷であり、他のサーバ $s_i (i = 2, 3, \dots, m)$ は過負荷でないとする。そして、過負荷のサーバから過負荷でないサーバにエージェントを移動することで、各サーバの計算負荷を閾値以下にすることができるものとする。つまり、初期条件として (3), (4), (5) 式が満たされているものとする。

$$W(A_1) > T_h \quad (3)$$

$$W(A_i) < T_h (i = 2, 3, \dots, m) \quad (4)$$

$$\sum_{1 \leq i \leq m} W(A_i) < mT_h \quad (5)$$

このとき、過負荷のサーバ s_1 から他のサーバ $s_i (i = 2, 3, \dots, m)$ にエージェントを移動して、負荷の分散を行うことを考える。 s_1 から s_i に移動するエージェントの集合を $M_{1,i}$ とおくと、エージェントを移動した後の各サーバが保持するエージェント集合 A'_i は、(6) 式で与えられる。

表1 表記
Table 1 Notations

a_i	エージェント $i (i = 1, 2, \dots, n)$
s_i	エージェントサーバ $i (i = 1, 2, \dots, m)$
w_i	エージェント a_i の計算負荷
$W(A_i)$	エージェント集合 A_i の計算負荷
$p(a_i, a_j)$	a_i と a_j の間のインタラクション頻度
$P(A_i, A_j)$	A_i と A_j の間のインタラクション頻度
A_i	移動前に s_i が保持するエージェント集合
A'_i	移動後に s_i が保持するエージェント集合
T_h	エージェントサーバの計算量閾値

$$A'_i = \begin{cases} A_1 \setminus \cup_{2 \leq j \leq m} M_{1,j} & (i = 1) \\ A_i \cup M_{1,i} & (i = 2, 3, \dots, m) \end{cases} \quad (6)$$

上記の式を用いて、サーバの負荷を均等にし、かつサーバ間で発生するインタラクション量を抑えるという目的は (7), (8) 式として定式化することができる。

$$\min \sum_{1 \leq k \leq m} \sum_{k < l \leq m} P(A_i, A_j) \quad (7)$$

$$s.t. \quad W(A'_i) < T_h \quad (i = 1, 2, \dots, m) \quad (8)$$

(7), (8) 式を満たす $M_{1,i} (i = 2, 3, \dots, m)$ が、エージェント配置問題における最適なエージェント移動集合である。

$|A_1|$ が大きい場合、全ての配置の中から (7) 式を満たすエージェントの配置を決定することは困難である。なぜならば、 $|A_1|$ 体のエージェントに対して、エージェントの移動集合の選び方は $m^{|A_1|}$ 通り存在するからである。そこで 4. 章で、エージェント配置問題に関する近似アルゴリズムを定める。

4. エージェント配置問題に対する近似解

本章では、エージェント配置問題に対する本稿が提案する近似解法と、その解法の特徴について述べる。

4.1 Community-based Load Balancing

過負荷になった s_1 は、以下の情報が取得できるとする。

- 他のサーバの計算負荷
- 内部エージェントの計算負荷
- 内部エージェント間のインタラクション頻度
- 各内部エージェントと外部サーバ間のインタラクション頻度

エージェント集合 A の評価値として、 $C(A)$ を定義する。 $C(A)$ は、 A を加えても計算負荷が閾値を超えない。

Algorithm 1 Community-based Load Balancing

```

Require:  $W(A_1) > T_h$ ,  $W(A_i) < T_h (i = 2, 3, \dots, m)$ ,
 $\sum_{1 \leq i \leq m} W(A_i) < mT_h$ 
loop
     $cost_{min} \leftarrow \infty$  : minimum  $C(A)$  in the following loop
     $M_{min} \leftarrow \emptyset$  : set of agents providing  $cost_{min}$ 
     $toservind \leftarrow null$  : index of the server which  $M_{min}$  moves to
    for  $a_i \in A_1$  do
         $M, s \leftarrow OptimalSet(a_i)$ 
        if  $cost_{min} > C(M)$  then
             $cost_{min} \leftarrow C(M)$ 
             $M_{min} \leftarrow M$ 
             $toservind \leftarrow$  index of  $s$ 
        end if
    end for
    if  $W(A_1) \leq T_h$  and  $cost_{min} \geq 0$  then
        end loop
    end if
    move  $M_{min}$  to  $toservind$  ( $A_1 \leftarrow A_1 \setminus M_{min}$ ,
     $A_{toservind} \leftarrow A_{toservind} \cup M_{min}$ )
end loop

```

ないサーバのうち， A が最もインタラクションを行うサーバに移動した場合の，インタラクションコストの変化量である。 A を移動可能なサーバが存在しない場合，その評価値は無限大であるとする。

$$C(A) = \begin{cases} P(A, A_1 \setminus A) \\ - \max_{\{m_k | W(A_k \cup A) < T_h\}} P(A_k, A) \\ (\exists m_i | W(A_i \cup A) < T_h) \\ \infty \quad (\forall m_i | W(A_i \cup A) > T_h) \end{cases} \quad (9)$$

我々が提案する Community-based Load Balancing (CLB) の擬似コードを Algorithm 1 に示す。CLB はエージェントの移動集合 $M_i (i = 2, 3, \dots, 4)$ の近似解を，山登り法で求める。まず，操作 $OptimalSet$ を定義する。 $OptimalSet$ は 1 体のエージェントを引数にとり，そのエージェントが共に移動する最適なエージェント集合と，その移動先を返す。CLB は，各エージェントに対して $OptimalSet$ を適用し， $C(A)$ が最も小さいエージェント集合を， $OptimalSet$ により得られる移動先サーバに移動する。この操作を移動元サーバの計算負荷が閾値を下回り，かつ評価値が負の(移動によりインタラクションコストが減少する)エージェント集合が見つからなくなるまで繰り返す。

次に，操作 $OptimalSet$ について述べる。 $OptimalSet$ は引数としてエージェントを受け取り，そのエージェントと共に移動する最適なエージェント集合と，移動先サーバを返す。最適なエージェント集合を求

Algorithm 2 ApproximateOptimalSet (a)

```

Move  $\leftarrow \emptyset$  : set of agents which move with  $a$ 
Candidate  $\leftarrow \{a\}$  : set of candidates for  $Move$ 
 $C_{min} \leftarrow \infty$  : minimum  $C(A)$  in the following loop
 $M_{min} \leftarrow Move$  : set of agents providing  $C_{min}$ 
repeat
    Select  $a_i \in Candidate$  which has the least  $c(a_i)$ 
     $Move \leftarrow Move \cup \{a_i\}$ 
     $Candidate \leftarrow Candidate \setminus \{a_i\}$ 
    if  $C_{min} > C(Move)$  then
         $M_{min} \leftarrow Move$ 
         $C_{min} \leftarrow C(Move)$ 
    end if
    Add  $\{a_j | p(a_i, a_j) > 0, a_j \in A_1 \setminus (Candidate \cup Move)\}$ 
until  $W(Move \cup A_i) > T_h (i = 2, 3, \dots, m)$ 
return  $M_{min}$  and  $s$  (server which  $M_{min}$  moves to in  $C_{min}$ )

```

めることは，一般的に困難であるため，本稿では $OptimalSet$ の近似解を求める $ApproximateOptimalSet$ を提案する。 $ApproximateOptimalSet$ の擬似コードを Algorithm 2 に示す。まず，移動エージェント集合 $Move$ と，移動候補エージェント集合 $Candidate$ を定める。初期状態で $Move$ は空集合であり， $Candidate$ は引数で与えられたエージェントのみを含む集合である。そして以下の操作を， $W(Move)$ が全てのサーバの許容量を上回るまで反復する。まず，候補エージェントの中から，(10) 式で定められる評価値 $c(a_i)$ が最も小さいエージェントを選択し，移動エージェント集合に加える。 $c(a_i)$ は，そのエージェントを $Move$ に加えることによる， $C(Move)$ の変化量である。

$$c(a_i) = C(Move + a_i) - C(Move) \quad (10)$$

次に，移動エージェント集合に加えたエージェント a と同じサーバに存在し，かつ a とインタラクションを行い， $Candidate$ や $Move$ に含まれないエージェントを候補エージェントに加える。以上の操作の反復の中で，最も評価値が小さいエージェント集合を選択し，評価値 $C(A)$ が得られる移動先のサーバと共に返す。

4.2 特 徵

CLB は，エージェントの配置を決定するために，移動するエージェント集合を考えている。通常，大規模なマルチエージェントシステムにおいては，過負荷なサーバから他のサーバに負荷を分散するためには，複数のエージェントを移動することが必要である。複数のエージェントを移動する場合，移動するエージェントの集合を考えることで，単体ごとにエージェントを移動した場合よりも，インタラクションコストを小さくすることができる。

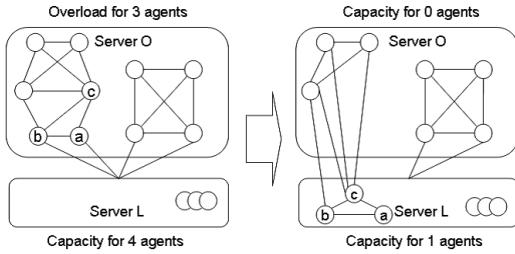


図 1 エージェント単体に対する評価に基づいて配置を決定する場合
Fig. 1 Agent movement based on evaluation of each agent

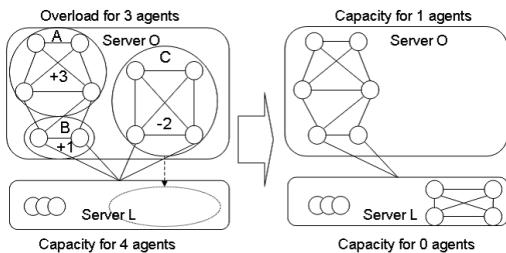


図 2 コミュニティに対する評価に基づいて配置を決定する場合
Fig. 2 Agent movement based on evaluation of each community

くすることができる。

エージェント単体を評価する場合と、コミュニティを評価する場合で、移動後のインタラクションコストが異なる例を図 1 と図 2 に示す。これらの図には、エージェントと、その間のインタラクション、サーバが表されている。ここでサーバ A はエージェント 3 体分過負荷であるとし、サーバ B は 4 体分の許容量があるとする。図 1 が、エージェント単体の評価値に基づいて、エージェントの移動を決定する場合である。エージェントの評価値を、移動によるインタラクションコストの変化量とすると、図 1 の場合はエージェント a, b, c の順に移動が行われる。エージェントを移動した後の外部インタラクションコストは、開始前の 4 から 6 に増加する。エージェント a が移動したこと、b, c が続いて移動したが、a が属するコミュニティが、移動先のサーバの許容量よりも大きいため、コミュニティが分割され、インタラクションコストが増加した。

図 2 が、コミュニティに対して評価を行った場合である。A, B, C のコミュニティの評価値は、移動することによるインタラクションコストの変化量であると

すると、それぞれ +3, +1, -2 である。このとき、評価値が最も小さいコミュニティ C を移動することで、インタラクションコストは 4 から 2 に減少する。従って、エージェント単体の評価に基づく場合と比較すると、インタラクションコストが 4 小さい。コミュニティの評価に基づいて移動を決定することで、コミュニティの分割によるインタラクションコストの増加を防ぐことができる。

5. エージェント配置シミュレーション

本章では、CLB の有効性を検証するために行ったシミュレーションについて述べる。

5.1 シミュレーション内容

このシミュレーションは、複数のサーバを接続した分散マルチエージェントシステムをシミュレートしたものである。このシミュレーションでは、エージェント数 $n = 1,000$ 、サーバ数 $m = 10$ としている。各エージェント間のインタラクション頻度 $p(a_i, a_j)$ は、小世界ネットワーク [12] の概念を用いて設定した。小世界ネットワークとは、高度に構造化され、かつ任意のノード間の経路長が短いネットワークである。実世界では、人間のコミュニティやインターネットなどに見られる。マルチエージェントシステムは対象とする系を反映するため、エージェント間のインタラクションは小世界ネットワークの性質を持っているといえる。本シミュレーションでは、エージェントの隣接接点を 6 に設定し、隣接接点とのインタラクションコストを 1 とした正則なネットワークを構成し、確率 p で枝を張り替えることで小世界ネットワークを構成した。

まず初期配置として、エージェントをクラスタごとに分割し、各サーバに配置する。具体的には、 $s_i (i = 1, 2, \dots, m)$ に存在するエージェント集合を $\{a_j | (i-1)n/m + 1 \leq j \leq in/m\}$ とした。この初期配置は、エージェント移動のために適用する各アルゴリズムに対して共通である。次に、各サーバに対して負荷の平均値をランダム値で定める。そして、各エージェントには、サーバに設定した負荷の平均値にランダム値をかけたものを設定する。エージェントの計算負荷とインタラクションコストには相関がないものとした。本シミュレーションでは、 T_h をエージェントの負荷の総和に 1.1 を掛けた値を設定した。エージェントの計算負荷を定めた後、計算負荷が閾値 T_h を超えたサーバに対して各手法を適用し、負荷の分散を行う。そして、(11) 式によって与えられる移動後のシステム

Algorithm 3 Sequentially Load Balancing

Require: $W(A_1) > T_h$, $W(A_i) < T_h$ ($i = 2, 3, \dots, m$),
 $\sum W(A_i) < mT_h$

Ensure: $W(A_i) < T_h$ ($i = 1, 2, \dots, m$)

```

repeat
    gainmin ← ∞ : minimum gain in the following loop
    amin ← null : agent providing gainmin
    for ai ∈ A1 do
        if gainmin > gain(ai) then
            amin ← ai
            gainmin ← gain(ai)
        end if
    end for
    move amin to s (server which amin moves to in
    gainmin) (A1 ← A1 \ {amin}, Aj ← Aj ∪ {amin})
until W(A1) < Th and gain(ai) ≥ 0 (∀ai ∈ A1)

```

全体のインタラクションコストの変化を記録する。

$$\sum_{1 \leq i \leq m} \sum_{i < j \leq m} P(A_i, A_j) \quad (11)$$

以上の操作を 1 回の反復として、負荷分散によるエージェントの移動数と、システム全体のインタラクションコストの比較を行う。

エージェント移動の際に適用したアルゴリズムは、以下の 2 手法である。

- Sequentially Load Balancing
- Community-based Load Balancing

本シミュレーションでは、エージェントをコミュニティの評価に基づいて移動する CLB と比較するために、エージェント単体を評価し、移動を決定する Sequentially Load Balancing(SLB) との比較を行った。SLB の擬似コードを Algorithm 3 に示す。SLB は、(12) 式に基づいてエージェントを評価する。エージェントの評価値は、エージェントを移動することによるインタラクションコストの変化量である。

$$gain(a_i) = \sum_{a_j \in A_1} p(a_i, a_j) - \max_{w_i + W(A_k) < T_h} \sum_{a_l \in A_k} p(a_i, a_l) \quad (12)$$

SLB は、評価値が最も小さいエージェントを選択し、そのエージェントが最もインタラクションを行うサーバに移動する操作を、移動元のサーバの計算負荷が閾値を下回り、かつインタラクションコストを改善するエージェントが見つからなくなるまで繰り返す。

5.2 シミュレーション結果

図 3 は適用手法ごとにシミュレーションを行い、そ

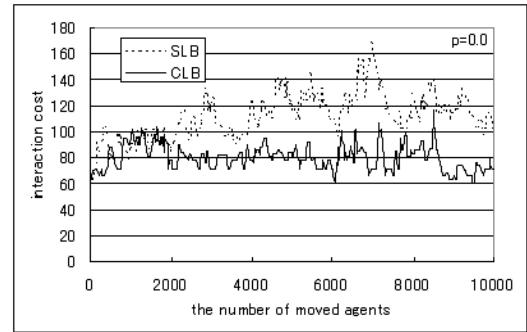


図 3 インタラクションコストの変化
Fig. 3 The change of the interaction cost

の結果を記録したグラフである。このグラフの縦軸はサーバ間で発生するインタラクションコストの総和であり、(11) 式で与えられる。横軸は、過負荷のサーバから移動を行ったエージェントの数を表す。このグラフ中には、SLB と CLB を適用した場合のインタラクションのコストを適用回数ごとに記録している。

CLB を適用することで、SLB を適用する場合よりも、インタラクションコストが小さな配置を行うことができる。具体的には、SLB を適用した場合のインタラクションコストの最大値は 168、平均値が 110.45 であるのに対して、CLB を適用した場合は最大値が 117、平均値が 79.77 であった。

次に、小世界ネットワークを構成する際の確率 p を変動させ、手法ごとのインタラクションコストの比較を行った。本稿では、各設定に対して、6 回のシミュレーションを行った。そのシミュレーション結果を表 2 に示す。 $I(p)$ は (13) 式で定義され、確率 p で小世界を構成する際の、SLB の CLB に対する改善度を表す。(13) 式における Average interaction cost とは、エージェントの移動数が 10,000 から 20,000 までのインタラクションコストの平均値である。

$$I(p) = 1 - \frac{\text{Average interaction cost in CLB}}{\text{Average interaction cost in SLB}} \quad (13)$$

10,000 から 20,000 までの平均値を取った理由は、ある程度エージェントの移動を繰り返した後の、定常状態の平均値を測定するためである。

表 2 より、エージェントのインタラクションが構成するネットワークが正則、または小世界ネットワークの場合に、インタラクションコストはより改善する。改善度はネットワークが正則な場合に最も大きく、SLB のインタラクションコストを約 27% 改善している。こ

表 2 平均インタラクションコストと改善度の比較
Table 2 the average interaction cost and improvement

probability p	SLB	CLB	$I(p)$
0	110.946	80.880	0.27099
0.0001	111.296	84.373	0.24190
0.001	111.680	81.719	0.26827
0.01	129.515	105.013	0.18918
0.1	327.989	288.883	0.11923
1.0	1418.101	1422.055	-0.0027

れは、ネットワークのクラスタリング係数が高いことで、移動元のエージェントにコミュニティが存在する確率が高くなり、CLB が SLB よりも効率的な配置を行ふことができることを示している。

しかし、インタラクションコストが減少する一方で、エージェントの配置を決定するために必要となる計算量は大きくなる。ここでは、エージェントがインタラクションを行うエージェントの平均数を c とし、過負荷のサーバが計算負荷を閾値以下にするために移動しなければならないエージェントの数を N とする。SLB では、各エージェントの評価値を求める計算量は c に比例し、 s_1 が保持する全てのエージェントの評価値を求めるための計算量は $O(c|A_i|)$ である。そして、最も評価値が小さいエージェントを選択して移動する操作を、移動元のサーバの負荷が閾値を下回るまで繰り返すため、エージェントの移動に必要な計算量は $O(Nc|A_i|)$ である。

次に CLB の計算時間を考える。まず、OptimalSet の操作に必要な計算量を S とする。CLB は各エージェントに対して OptimalSet を適用し、最も評価値が小さいコミュニティを選択して移動を行うため、エージェントの移動に必要な計算量は $O(NS|A_1|)$ である。本稿で提案した OptimalSet の近似解法である ApproximateOptimalSet に必要な計算時間を探る。Candidate に含まれる 1 体のエージェントの評価値を求める計算時間は $O(c)$ である。その操作を Candidate に含まれるエージェントに対して反復する。クラスタリング係数が高く、Candidate に含まれるエージェント数を $O(c)$ とすると、1 体のエージェントを移動エージェント集合に追加するために必要な時間は $O(c^2)$ である。その反復をコミュニティの移動先サーバがなくなるまで反復するため、その反復回数を $O(N)$ とすると、総計算時間は $O(Nc^2)$ となる。よって、CLB の総計算時間は $O(N^2c^2|A_i|)$ である。

従って、1 台のサーバが保持するエージェント数が

表 3 サーバ数を増加させた場合の CLB の平均計算時間
Table 3 the average computation time in changing the number of servers

サーバ数 (台)	10	20	50	100
CLB の平均計算時間 (分)	0.03	0.05	0.09	0.16

表 4 サーバが保持するエージェント数を増加させた場合の CLB の平均計算時間
Table 4 the average computation time in changing the number of agents in a server

サーバ 1 台のエージェント数 (体)	100	200	500	1000
CLB の平均計算時間 (分)	0.03	0.3	5.8	101

増加するにつれて、 $|A_i|$ と N が増加し、CLB の計算量は大きく増加する。1 台のエージェントサーバに存在するエージェント数を 100 とし、サーバ数を増加させた場合の、CLB の平均計算時間を表 3 に示す。表 4 には、サーバ数を 10 台とし、サーバ 1 台あたりに存在するエージェント数を増加させた場合の CLB の平均計算時間を示す。本計測は CPU1.7GHz、メモリ 512MB の計算機で、Java を用いて実装されたプログラム上で行い、エージェントのインタラクションを正則ネットワークとして、エージェントの隣接接点数を 6 とした。これらの結果より、エージェント数に比例してサーバが存在する場合、CLB の計算量の増加は小さい。しかし、サーバ 1 台あたりのエージェント数が増加するに従って、CLB の計算量は大きく増加する。

6. む す び

本研究では、分散マルチエージェントシステムにおいて、システムの性能を改善するためのエージェント配置に関する議論を行った。分散マルチエージェントシステムを構築する場合、エージェントの計算負荷とインタラクションコストを考慮して、その配置を決定する必要がある。

本研究の主な貢献は、インタラクションコストを小さくするエージェント配置手法である、Community-based Load Balancing(CLB) を提案したことである。CLB の特徴は、互いに頻繁にインタラクションを行うエージェント集合であるコミュニティに注目し、コミュニティに対する評価を用いてエージェントの配置を決定することである。本稿では、エージェント配置に関するシミュレーションを用いることで、エージェントのインタラクションが構成するネットワークが正則、または小世界ネットワークである場合に、エージェント単体に対する評価に基づいて配置を決定する SLB

アルゴリズムよりも、インタラクションコストが小さくなることを示した。

一方で、この手法ではエージェントの移動による安定性や収束性を保証することはできない。安定性や収束性は、分散アルゴリズムにおける重要な性質である。こうした性質を保証するためにエージェントの移動時にサーバ間で協調を行う必要があるが、本研究ではその方法に言及していない。サーバ間でエージェントの配置を決定する際に、こうした性質を保証するための協調方法を検討する必要がある。

CLB の計算量も解決すべき問題の一つである。本稿では、エージェント集合に対する評価に基づいて配置を決定する基本的なアルゴリズムとして CLB を提案した。しかし、CLB はサーバ 1 台あたりのエージェント数が増加するにつれて計算量が大きく増加するという問題がある。従って、CLB をより効率的に実行するための手法を提案する必要がある。

謝 辞

本研究は、日本学術振興会科学研究費基盤研究(A)(18200009, 2006-2008) 戦略的情報通信研究開発推進制度地域情報通信技術振興型研究開発(2005-2007)の助成を受けて行われた。

文 献

- [1] L. Gasser and K. Kakugawa: "Mace3j: fast flexible distributed simulation of large, large-grain multiagent systems.", AAMAS, pp. 745–752 (2002).
- [2] M. Balmer, N. Cetin, K. Nagel and B. Raney: "Towards truly agent-based traffic and mobility simulations", AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, Washington, DC, USA, IEEE Computer Society, pp. 60–67 (2004).
- [3] H. Tai and G. Yamamoto: "An agent server for the next generation of web applications", DEXA '00: Proceedings of the 11th International Workshop on Database and Expert Systems Applications, Washington, DC, USA, IEEE Computer Society, p. 717 (2000).
- [4] T. Ishida, L. Gasser and H. Nakashima Eds.: "Massively Multi-Agent Systems I, First International Workshop, MMAS 2004, Kyoto, Japan, December 10-11, 2004, Revised Selected and Invited Papers", Vol. 3446 of Lecture Notes in Computer Science Springer (2005).
- [5] I. Ahmad and A. Ghafoor: "Semi-distributed load balancing for massively parallel multicenter systems", IEEE Transactions on Software Engineering, **17**, 10, pp. 987–1004 (1991).
- [6] B. Hong and V. K. Prasanna: "Distributed adaptive task allocation in heterogeneous computing environments to maximize throughput", ipdps, **01**, p. 52b (2004).
- [7] T.-H. Chia and S. Kannapan: "Strategically mobile agents", MA '97: Proceedings of the First International Workshop on Mobile Agents, London, UK, Springer-Verlag, pp. 149–161 (1997).
- [8] T. Kawamura, S. Joseph, A. Ohsuya and S. Honiden: "Quantitative evaluation of pairwise interactions between agents", ASA/MA 2000: Proceedings of the Second International Symposium on Agent Systems and Applications and Fourth International Symposium on Mobile Agents, London, UK, Springer-Verlag, pp. 192–205 (2000).
- [9] 能登正人, 沼澤政信, 粟原正仁: "エージェントの移動性を考慮したエージェント間通信のトラフィック量に関する実験と評価", 電気学会論文誌 C, **124-C**, 3, pp. 904–911 (2004).
- [10] K.-P. Chow and Y.-K. Kwok: "On load balancing for distributed multiagent computing.", IEEE Transactions on Parallel and Distributed Systems, **13**, 8, pp. 787–801 (2002).
- [11] M.-W. Jang and G. Agha: "Adaptive agent allocation for massively multi-agent applications.", in T. Ishida, et al. [4], pp. 25–39.
- [12] D. J. Watts and S. H. Strogatz: "Collective dynamics of 'small-world' networks.", Nature, **393**, 6684, pp. 440–442 (1998).

(平成 18 年 5 月 19 日受付, 8 月 9 日再受付)

宮田 直輝



2006 年京都大学工学部情報学科卒業, 現在, 同大学院社会情報学専攻修士課程に在学中。大規模マルチエージェントシステムに関心を持つ。

石田 亨 (正員)



1976 年京都大学工学部情報工学科卒業, 1978 年同大学院修士課程修了。同年日本電信電話公社電気通信研究所入所。ミュンヘン工科大学, パリ第六大学, メリーランド大学客員教授など経験。工学博士。IEEE フェロー。情報処理学会フェロー。現在, 京都大学大学院情報学研究科社会情報学専攻教授, 上海交通大学客員教授。自律エージェントとマルチエージェントシステム, セマンティック Web 技術に取り組む。デジタルシティ, 異文化コラボレーションプロジェクトを推進。

超越型コミュニケーション： 大規模公共空間のための位置依存型誘導法

伊藤英明[†], 中西英之[†]
小泉智史^{††}, 石田亨[†]

居住環境や労働環境におけるコミュニケーション支援の研究が盛んに行われてきている一方で、我々の日常生活にとって重要であるはずの大規模公共空間を対象としたコミュニケーション支援の研究は不十分である。そこで我々は、ユビキタス環境の普及によって可能になりつつある「超越型」のコミュニケーション支援システムを提案し、それによって大規模公共空間の管理スタッフが訪問者に対して誘導等の手助けを効果的に行えることを示す。このシステムでは、訪問者のコンテキスト情報が仮想公共空間にリアルタイムで反映される。管理スタッフは画面に鳥瞰視点で表示される仮想公共空間を見渡して現地の状況を把握し、画面上の特定位置をポイントングすることでコミュニケーション対象となる訪問者を選択する。我々が行った実験では、大規模空間における人々の動きを把握する際に、鳥瞰視点と現地における一人称視点は互いに補完的に機能することが分かり、システムの有効性が示された。我々はこのシステムを、大規模屋内空間である京都駅に設置した。

Transcendent Communication: Location-Based Navigation for a Large Scale Public Space

HIDEAKI ITO,[†], HIDEYUKI NAKANISHI,[†], SATOSHI KOIZUMI^{††},
and TORU ISHIDA[†]

Many studies have been conducted on supporting communication in home and office spaces, but relatively few studies have explored supporting communication in large-scale public spaces, despite the importance of such environments in our daily lives. We propose a transcendent means of communication as an emerging style in this pervasive computing era: a system that allows administrative staff to effectively help visitors in large-scale public spaces. The visitors' context is used to provide a bird's-eye view of a simulated public space for the staff to grasp the situation and point at a particular location within the view to indicate the visitors they intend to address. The results of an experiment showed synergic effects between the bird's-eye view and the first-person one in determining the spatial movements of people. In an indoor large-scale public space, Kyoto station, we installed our prototype and learned the implications of its use.

1. はじめに

我々が生活する空間は家庭、職場、公共空間に分けられる。視覚的コミュニケーションの研究の多くは家

庭と職場を対象としている。そのため、物理的に隔てられた空間を結び²⁾⁸⁾¹¹⁾、結ばれた空間の利便性を向上させる⁵⁾ために計算機ネットワーク技術をいかに活用するかが研究の主眼であった。

一方、公共空間の重要性は増すばかりであり、我々は大規模な公共空間の安全性確保と利用価値向上のためにネットワーク技術を役立てるべく研究を進めている。大規模公共空間には管理スタッフと出入りする訪問者の2つの立場が存在する。管理スタッフは要望に応じて案内や誘導などの支援を訪問者に対して行う。管理スタッフが訪問者と効率よくコミュニケーションを行えるか否かが大規模公共空間の快適さを左右するといえる。本研究の目的は、ユビキタス環境の普及に

† 京都大学社会情報学専攻

Dept. of Social Informatics, Kyoto University
現在、オムロン(株)センシング&コントロール研究所

Presently with Sensing and Control Technology Laboratory, OMRON Corporation

†† JST CREST デジタルシティプロジェクト

JST CREST Digital City Project

現在、(株)国際電気通信基礎技術研究所 知能ロボティクス研究所

Presently with Intelligent Robotics and Communication Laboratories, ATR

より可能になった、こうした状況で行うにふさわしい管理スタッフと訪問者の新たなコミュニケーション形態を生み出すことがある。

我々は公共空間の監視カメラと放送設備を融合させるアプローチをとっている。通常、監視カメラは異常事態の発見のためだけに用いられるが、本研究では公共空間の監視カメラ映像を利用した訪問者の誘導手法を提案する。監視カメラから得られた画像は状況把握のほか、誘導先地点や誘導内容の決定にも用いられる。

放送設備に関しては、公共空間に設置されたスピーカーの音量を独立に制御できる機能や局所的な音声通信が実現できる特別な装置があれば、対象訪問者を限定し、その位置情報を活用した誘導が可能である。

位置情報に基づく情報支援を大規模公共空間に適用する場合、各監視カメラから得られる断片的な画像を参照するよりも大局的に統合された視点で空間全体を把握することが好ましい。そこで、現実の大規模公共空間の状況をその全体が見渡せるように3次元仮想空間上に視覚化できる鳥瞰視点を監視インターフェースに採用した。管理スタッフは仮想空間上の位置を指示することで訪問者とのコミュニケーションを行う場所を指定できる。このような形態は従来の視覚的コミュニケーションには見られず、超越型コミュニケーション²⁰⁾²¹⁾と呼んでいる。

この仮想空間は実環境から取得されるコンテキスト情報に基づいて生成される。上記の仮想鳥瞰視点を生成するには少なくとも訪問者の位置情報が必要である。この位置情報によって訪問者の移動状況の把握と誘導対象とする訪問者の指定が可能になる。開発した超越型コミュニケーション環境のプロトタイプでは、現実空間上の訪問者の位置に対応した仮想空間上の位置に人体モデルが表示される。

また、仮想空間上でこの人体モデルを指定することで、対応する訪問者との音声通話を開始することもできる。デジタルシティプロジェクト⁶⁾のもとでこうしたプロトタイプを開発し、すでに開発していた避難シミュレータ⁴⁾も活用して超越型コミュニケーションの評価を実施した。さらに、このプロトタイプを実世界の大規模公共空間である京都駅へ設置した。

京都駅のように1日の乗降客が30万人を超える屋内の公共空間では人命の安全確保のための危機管理が不可欠である。こうした非常時の避難誘導において超越型コミュニケーション環境が果たすべき役割は大きい。通常の監視カメラや放送設備では不可能な、複数の集団に対しての個別の避難誘導を、単一の仮想空間上で実施できるためである。

2. 超越型コミュニケーション環境の設計

超越型コミュニケーション環境では管理者の立場にあるユーザが鳥瞰視点から描かれた仮想空間を通して実世界の状況を把握し、仮想空間上の人物や領域を指定することでコミュニケーションを行う個人または集団を選定できる。図1に概念図を示す。実世界を眺望しながら選定した相手に語りかける管理スタッフが描かれている。

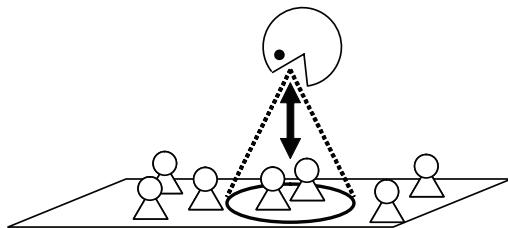


図1 超越型コミュニケーションの概念図

Fig. 1 Conceptual figure of transcendent communication

以下では超越型コミュニケーションと、対面コミュニケーションの再現と強化を目指す視覚的コミュニケーションとの差異について論じる。図2は両者の違いを明確にするために対面コミュニケーション、視覚的コミュニケーション、超越型コミュニケーションを2つの面から比較したものである。

1つめはコミュニケーション参加者の位置関係の違いである。視覚的コミュニケーションでは物理的に離れたユーザ間で対面コミュニケーションが可能である。超越型コミュニケーションではユーザが物理的に離れて存在するか否かよりも各ユーザが位置する階層の違いが重要である。鳥瞰視点にいるユーザは実世界全体を眺望できる立場にあり、実際に実空間に存在する、内在視点のユーザの立場とは明確に区別される。通常、両視点のユーザは物理的に離れて存在することが多いが、同一の空間にいても構わない。

2つめの違いはコミュニケーション空間の形にある。視覚的コミュニケーションでは対人距離やアイコンタクトのような社会的合図¹⁴⁾が仮想空間上に提供される。実世界ではこうした合図に基づいて対面コミュニケーションが開始されることが多い。よって対面指向のコミュニケーションでもきっかけとして同様の合図が必要になる。一方、超越型コミュニケーションではこうした社会的合図によってコミュニケーションが開始されることはない。仮想空間上に表現される現実空間の状況に応じて鳥瞰ユーザがコミュニケーションすべき内在ユーザを決定する。また、超越型コミュニケ

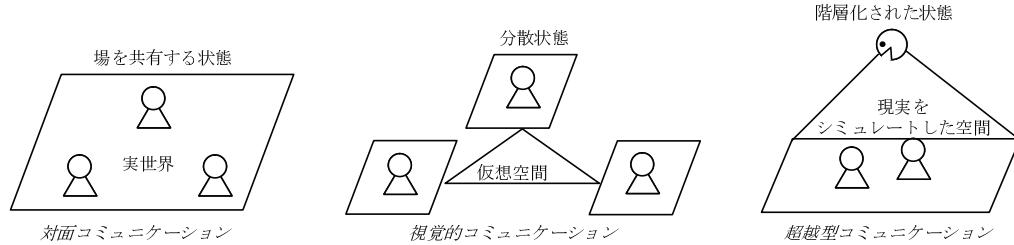


図 2 対面 / 視覚的 / 超越型コミュニケーションの比較
Fig. 2 Comparison of FTF, visual, and transcendent communication

ションの場合は個々の内在ユーザ自身よりもその周囲の状況に依存した対話が行われる点が特徴的である。

従来の視覚的コミュニケーションの参加者どうしは基本的に相互に対等な関係にあり、プライバシーの保護が重要な課題になる¹⁾¹⁸⁾。超越型コミュニケーションの参加者間には鳥瞰ユーザが内在ユーザに対する特権的能力を持つという非対称的な関係が成立している。この特徴は公共空間の管理者のように少数の担当者が実世界の訪問者など大勢の参加者を支援する場合に効果的だといえる。

3. シミュレーションによる評価

3.1 目的

プロトタイプを実装する前に、超越型コミュニケーションに用いる鳥瞰視点インターフェースと、従来の視覚的コミュニケーションに用いられる一人称視点インターフェースとの差異が、ユーザの認知行動に与える影響を確認する目的のシミュレーション実験を実施した。この実験では、鳥瞰視点からの観測と、一人称視点による主観的な経験の組み合わせによって、群集の行動に対する理解がどれだけ促進されるかを調査した。

3.2 実験装置

実験には、これまでに開発してきた3次元仮想空間シミュレータ FreeWalk¹³⁾とシナリオ記述言語 Q⁷⁾を利用して避難のシナリオを構築した。図3に示すように、鳥瞰視点の被験者は仮想空間上で群集の行動シミュレーションを観測し、内在視点の被験者はアバターとして仮想空間内で実際に行動する。

3.3 実験内容

管理スタッフがユーザに1対1で対応するナビゲーションの例では、鳥瞰視点が効果的であることが知られている³⁾が、避難のシナリオで管理スタッフが対応するユーザは複数である。従って、避難のような群集行動では、管理スタッフは鳥瞰視点によるマクロな情報に加え一人称視点によるミクロな情報を併用するこ

とで、群集の誘導に良い影響を与えることが考えられる。そこで両者の併用による群集行動に対する理解の相乗効果を確認するための比較実験を行った。

実験では大学生96名からなる被験者を4つのグループ、一人称視点のみ(FPグループ)、鳥瞰視点のみ(BEグループ)、一人称視点のあと鳥瞰視点(FP-BEグループ)、鳥瞰視点のあと一人称視点(BE-FPグループ)に分け、それぞれを試す順序も考慮した。1度の実験に1グループあたり6名が参加し、計4回の実験を行った。

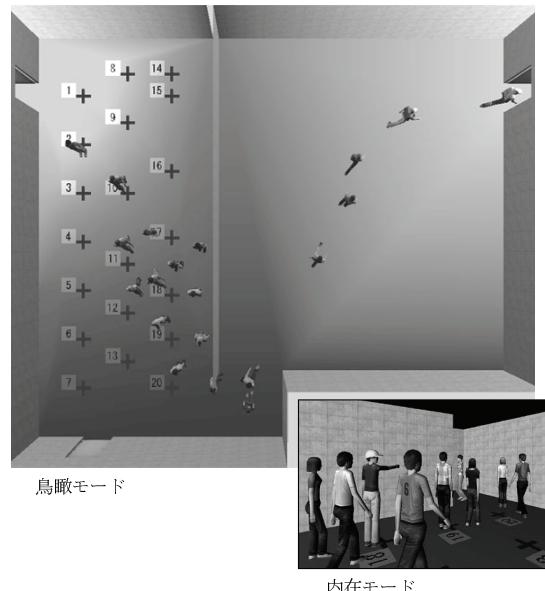


図3 避難シミュレーションシステム

Fig. 3 Evacuation simulator

シミュレーション開始時には参加者全員が図3に示すような一部を除き壁で左右に仕切られた部屋の左側にいる。4名の誘導者は16名の避難者が左側の間違った出口に向かわないように注意しつつ右側にある正し

い出口に誘導しなければならない。

6名の被験者の実験における役割は、FP条件とBE条件では異なる。FP条件では、6名全員が避難者役となり、ソフトウェアで実装された他の10名の避難者エージェントとともに避難する。この際、4名の誘導者は全てエージェントがその役を担う。一方、BE条件では、16名の避難者および4名の誘導者は全員エージェントで実装し、6名の被験者はその避難行動を鳥瞰視点から観察するのみである。

このようにFP条件とBE条件で被験者の担う役割が異なる理由は、一人称視点と鳥瞰視点という視点の違いが、避難における群集行動の学習効果にどのような影響を与えるかを調べるためにある。

被験者は誘導手法の違いに起因する次の2つの群集行動を観測、経験することになる。

指差誘導法：誘導者は腕を振り上げて非常口を指し、「出口は向こうだ！」と叫ぶ。誘導者自身は最後に避難する。

吸着誘導法：誘導者は周囲にいる避難者にだけ「自分について来い」と小声でささやき、出口へと向かう。この場合、出口に向かう人の流れが形成される。

被験者には実験の前後で2度、17項目の質問からなるテストを行う。テストでは、被験者は群集行動に関する説明文を読み、その行動を引き起こした誘導手法を二者択一で答える。このテストは群集行動への理解を問うものであり、杉万らによる前回の実験¹⁷⁾をもとに作成した。実験の前後でテストの得点に有意差があるか否かを調べるためにt検定を用いた。有意差の存在は被験者自身の観測または経験により群集行動の理解に対する学習効果があったことを示す。

3.4 結 果

表1に、9つの質問について実験に参加する前後ににおけるテスト結果の平均値の差を検定した結果を示す。表中の値はt値であり、No1～9の各質問において、実験参加前には誤って回答され、実験参加後には正しく回答された得点の上昇分が、統計的に有意であるか否かを評価したものである。なお、残り8つの質問はどのグループでも有意差が認められなかったので記載を省略した。

こうした評価結果は質問設定に依存するとはいえるが、鳥瞰視点からの観測が群集行動の理解に必要であったことは明確である。FPグループでは避難行動に関するNo.3からNo.9の質問において学習効果がみられないが、一人称視点に価値がないわけではない。BEやFP-BEの両グループが学習しなかったNo.6とNo.7

の質問をBE-FPグループが学習できている点は興味深い。これらの質問は群集行動における密集度に関連している。

以上の結果から、群集行動の全体像を把握する背景知識を得ることで、被験者は自身の主観的経験から獲得した密集度の感覚に基づいて群集行動の状況を推論できると考えられる。

結論として、群集の移動状況の理解には鳥瞰視点が効果的であること、こうした状況理解は一人称視点による主観的経験から得た感覚によりさらに深められることが導かれた。これは、鳥瞰視点によるマクロな理解の後、一人称視点によるミクロな理解が全体の学習効果を高めるという、人の理解のプロセスとしても直感的に納得のいく結果である。

4. プロトタイプ開発

シミュレーションにより、超越型コミュニケーションの有効性を示す結果が得られたため、大規模公共空間への適用に向けたプロトタイプを開発した。

4.1 京都駅避難誘導システム

4.1.1 システム概要



図4 避難誘導システム
Fig. 4 Evacuation guidance system

図4に、我々が開発した避難誘導システムの画面を示す。駅のホーム上を携帯電話を手に避難する人々が表示されている。また、大型タッチディスプレイの前に立ち指差しをしているのは遠隔誘導センターで避難

No.	質問 (全項目とも「吸着誘導法」が正解)	FP	BE	FP-BE	BE-FP
1	誘導者が最初に避難する	4.3***	2.2*	2.3*	4.0***
2	誘導者が避難状況を観測しない	2.8**	4.4***	4.0***	4.2***
3	誘導者が避難者と同じように避難する	1.6	2.2*	1.9*	2.9**
4	避難行動が他者の行動に起因する	1.2	2.1*	3.3**	2.9**
5	誤った非常口への避難を防ぐ役割の者がいない	1.6	4.9***	3.7***	4.5***
6	他の避難者に追従して避難する	1.3	1.0	0.7	2.1*
7	避難者が集団を形成する	1.6	0.5	1.2	1.9*
8	誘導者と避難者が一緒に避難する	0.7	2.0*	0.2	3.4**
9	他の避難者と同じ行動をとろうとする	0.9	2.5*	1.5	0.9

p<.05, **p<.01, *** p<.001 (df=23)

表 1 質問および回答集計結果 (一对標本による片側 t-検定による)

Table 1 Summary of the result of the quiz

誘導にあたる管理者を想定している。タッチディスプレイには仮想空間が鳥瞰視点で描かれている。駅は静的モデル、訪問者を示す人体モデルは CG アニメーションでそれぞれ表示され、人体モデルの位置は駅で避難する歩行者の位置と同期して更新される。

また、この写真では誘導者が訪問者を視覚化した人体モデルを指差している。こうした指差し操作がタッチディスプレイによって検知されると、誘導スタッフのヘッドセットと、人体モデルに対応する訪問者の携帯電話の間で通話が可能になり、超越型コミュニケーションが実現する。このように、我々が開発した誘導システムでは、近年ほぼ 1 人 1 台の割合で普及する携帯電話を訪問者が誘導スタッフとコミュニケーションを行うデバイスとして利用する。もちろん携帯電話の回線は非常事態では使用できなくなることが多い。しかし、我々のシステムにとっては現状で最も実用的なデバイスである。

なお、指定した訪問者の電話番号を誘導システムが取得するには、いくつかの方法が考えられる。現時点では実験者が各人体モデルに事前に電話番号の割り当てを行っておく方法をとっている。より良い方法として、利用者全員に指定した場所からシステムに電話をかけてもらい、指定場所の情報を使って人体モデルと発信者の電話番号を対応づけることができる。将来的には、各訪問者の携帯電話にタグやセンサを取り付けて自動登録を行う、あるいは携帯電話を用いて超指向性のスピーカおよびマイクロホンアレイを天井に設置することが考えられる。

4.1.2 システム構成

図 5 に、開発した京都駅避難誘導システムの構成を示す。開発システムは、音声通信ハンドラ、視覚センサネットワーク、3 次元シミュレータの 3 つから構成される。以下、各構成部の詳細を記述する。

4.1.3 音声通信ハンドラ

音声通信ハンドラは、誘導スタッフの音声と訪問者

の携帯電話の間の通話を実現する。これにはアナログ電話回線ポートを備えた特殊なインターフェースカードとヘッドセット、及び 1 台の計算機を用いる。このインターフェースカードとヘッドセットは同じ計算機に接続されており、ヘッドセットのマイクから入力された音声は、カードを介して訪問者と接続された電話回線ポートへと送られる。

マイクからカードへの音声通信は計算機によって制御できるため、前述のようにタッチディスプレイ上の指差し操作により選択された訪問者へのみ、誘導スタッフの音声を送信することが可能である。またこのカードは複数の電話回線ポートを備えており、一度に集団へ話しかけることも可能である。集団へ話しかける場合は、誘導スタッフはドラッグ操作による矩形範囲指定や、一定の大きさをもった円領域により人体モデルを一括指定する。計算機は、選択された全ての人体モデルに対応する訪問者と接続された電話回線ポートへ、誘導スタッフのマイク入力データをおく。

以上のように、指差し操作と音声通話起動処理を連動させておくだけで誘導者は誘導指示の開始と終了を極めて容易に行い、仮想空間上に視覚化された任意の個人や任意の集団に話しかけることができる。

さらに、FreeWalk は複数ユーザによる使用に対応しており、1 つの仮想空間を複数の誘導者がそれぞれ違った鳥瞰視点で観測することもできる。1 人の訪問者に複数の誘導者が同時に話しかけることも可能である。この場合、携帯電話では多人数での同時通話ができないため、音声通信ハンドラが各誘導者の音声を統合する役割を果たす。

なお、音声通信ハンドラは、訪問者同士の音声通信はサポートしない。その理由は、我々の研究目的は、鳥瞰視点をもつユーザと内在視点をもつユーザとの間に発生する階層化された超越型コミュニケーションの性質を解明することにあり、本システムはそのプロトタイプという位置づけであるためである。

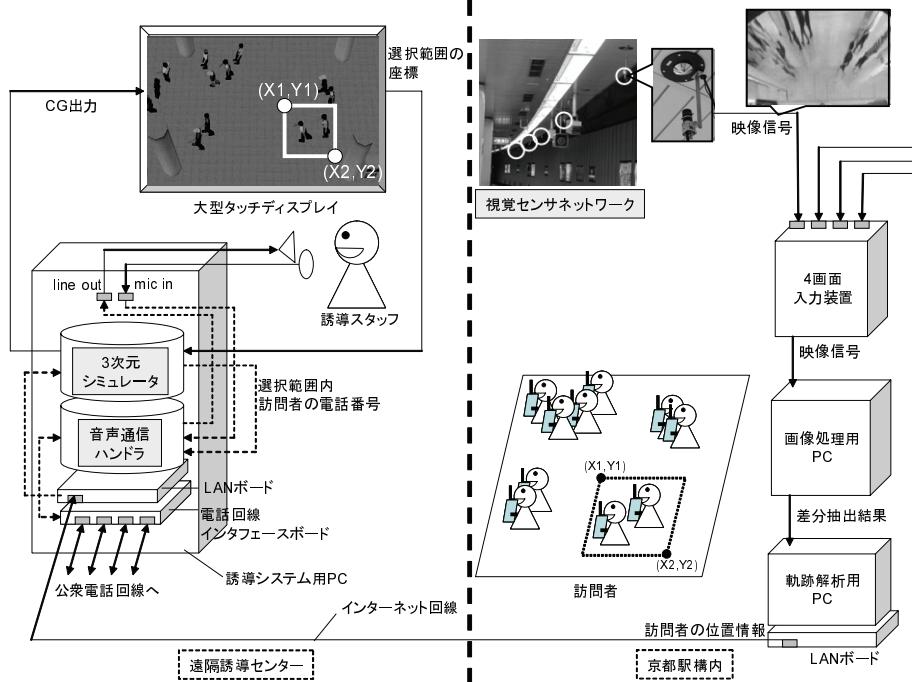


図 5 京都駅誘導システム構成図

Fig. 5 System architecture of Kyoto station guidance system

4.1.4 視覚センサネットワーク

京都駅構内に訪問者の位置情報を取得する視覚センサネットワークを設置した。視覚センサの台数はコンコースに 12 台、ホームに 16 台、計 28 台である。図 6 の見取図中の黒い丸がセンサの設置位置を示している。

図 5 中央上の写真が視覚センサであり、CCD カメラと特殊な形状の反射鏡で構成されている¹²⁾。各カメラの画角を広くすれば必要なカメラ数を減らせるが、通常、画角を広くすると画像にマイナスの歪みが生じる。反射鏡はこの歪みを抑える働きをする。反射鏡の形状は、カメラ光軸に直交する平面を等比で撮影できるように設計された自由曲面である。この光学的な仕掛けにより広い画角で歪みのない画像が得られる。

京都駅に設置した視覚センサネットワークでは視覚センサの他、カメラ 4 台の画像を統合する 4 画面入力装置と画像処理 PC を各 7 台、軌跡解析用 PC1 台を使用している。まず 4 画面入力装置は視覚センサ 4 台分の画像を 1 つに統合して画像処理 PC に送る。画像処理 PC は背景差分法によって移動物体の領域を抽出する。抽出結果は軌跡解析 PC に送られ、視覚センサの設置位置や移動可能な領域の境界といった物理的位置関係の知識に基づき移動物体の位置を求める。

このシステムでは、4 台のカメラを配置した 10m 四方程度の領域において、20 人程度の訪問者を同時に検

出することが可能である。一方、検出された訪問者が過去検出された訪問者と同一であるかを識別し、同一の場合にはその位置を追跡する識別性能に関しては、上記領域において数人を識別することが可能である。駅ホームでの人の行動ルールに関する知識を追加する等の手段により、識別性能をさらに向上させることができるもの。

このように、視覚センサネットワークは屋内環境下で効果を発揮し、GPS よりも高い精度で人物の動きを検知できる。また、デバイスフリーなアプローチでもありシステムに位置を知らせるための特別な機器をユーザは携行しなくてよい。よって、システムの利用を意識していない訪問者も含めてシミュレートされた駅の空間上に視覚化できる。

4.1.5 3 次元シミュレーター

視覚センサネットワークで検出した訪問者の位置情報は、およそ 10 ミリ秒間隔でインターネットを介し 3 次元仮想空間シミュレータ FreeWalk に送信される。この位置情報は、物理的および社会的な歩行者の行動に関するルールを用いて修正される。

ここで、物理的なルールとは他者、壁、支柱等との衝突回避のルールであり、磁気モデル¹⁵⁾により実装されている。一方、社会的なルールには、他者に追従する、他者とは一定の距離を置く¹⁶⁾といった群集行

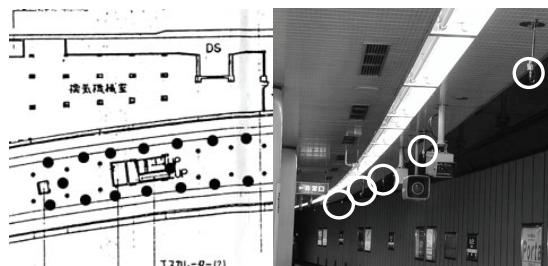


図 6 駅構内に設置した視覚センサネットワーク
Fig. 6 Vision sensor network in the station

動に関するものと、改札ゲート前では一列に並ぶ、話すときは円の形に集まる¹⁰⁾といった文化的な慣習にに関するものがあり、これらのルールは Q 言語により記述される。こうして修正された位置情報を基に、運動力学に基づく歩行アニメーション生成アルゴリズムにより¹⁹⁾訪問者の移動が 3 次元空間上の歩行アニメーションとして再現される。一方、駅自体の空間モデルとしては、デジタルカメラで撮影した静止画と構内の平面図に基づいた簡単な幾何モデルを組み合わせた 3 次元モデルを構築した。

4.1.6 評価

プロトタイプの評価は、一般利用者および被験者の安全を考慮し、平日早朝に、誘導スタッフ 1 名、訪問者 2 名の最も基本的な構成で実施した。その内容は、誘導スタッフが、ホーム上の同地点にいる 2 名の訪問者を、それぞれ南北別の階段を経由させて 1 階上の改札まで誘導するというものである。その際誘導スタッフがシステムを使用する様子をビデオカメラにより記録、分析を行なった。

その結果、誘導スタッフは訪問者を誘導する際に、一通り指示を出した後は、

- (1) 訪問者が途中で立ち止まる、急転回するなど、異常な歩行動作を行なった場合
- (2) 誘導スタッフ自身が予定していた道筋から、訪問者が大きく外れた場合
- (3) 事前に誘導スタッフ自身で設定したランドマークを訪問者が通過した場合

のいずれかのイベントが発生した際、該当する訪問者を選択し確認と指示を行うことが明らかになった。

このことから、誘導スタッフはシステムとのインタラクションを通じて訪問者の位置や動きの妥当性を評価し、その評価結果を対話開始の社会的合図の代用としていることが考えられる。

5. おわりに

大規模公共空間において鳥瞰視点と個別の音声コ

ミュニケーション機能を備えた、超越型コミュニケーション環境を提案した。超越型コミュニケーションにより、管理スタッフは訪問者の状況を把握しながら位置情報に基づいた誘導を行うことができる。また、群集行動のシミュレーションにより、鳥瞰視点と内在視点を併用した場合の学習効果を統計的に分析し、このコミュニケーション環境が従来の内在視点のものと補完的に働くことによる有効性を確認した。さらに超越型コミュニケーションの実例として、京都駅に設置したプロトタイプを紹介した。

6. 謝辞

本研究は JST CREST デジタルシティプロジェクトの支援により行われました。日頃から多大なご協力をいただいている京都市交通局、京都市総合企画局に感謝します。視覚センサネットワークを設置いただいた石黒浩教授、避難シミュレーションシステムの構築にご協力いただいた杉万俊夫教授、岡崎甚幸教授、筒口拳氏に感謝いたします。また、避難誘導シミュレータ開発における菱山玲子氏、河添智幸氏、板倉豊和氏、株式会社 CRC ソリューションズ、株式会社数理システム、株式会社 CAD センターのご協力に感謝します。

参考文献

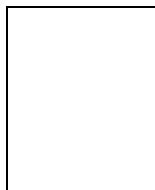
- 1) Borning, A. and Travers, M. Two Approaches to Casual Interaction over Computer and Video Networks. Proc. CHI91, (1991), 13-19.
- 2) Buxton, W. Telepresence: Integrating Shared Task and Person Spaces. Proc. Graphics Interface 92, (1992), 123-129.
- 3) Fukatsu, S., Kitamura, Y., Masaki, T. and Kishino, F. Intuitive Control of "Bird's Eye" Overview Images for Navigation in an Enormous Virtual Environment. Proc. VRST98, (1998), 67-76.
- 4) Hareesh, P.V., Shibano, N., Nakanishi, H., Kashiwagi, M. and Sawada, K. Evacuation Simulation: Visualization Using Virtual Humans in a Distributed Multi-User Immersive VR System. Proc. VSMM2000, (2000), 283-289.
- 5) Huang, E.M. and Mynatt, E.D. Semi-Public Displays for Small, Co-located Groups. Proc. CHI2003, (2003), 49-56.
- 6) Ishida, T. Digital City Kyoto: Social Information Infrastructure for Everyday Life. CACM, 45(7), (2002), 76-81.
- 7) Ishida, T. Q: A Scenario Description Language for Interactive Agents. IEEE Computer, 35(11), (2002), 54-59.

- 8) Ishii, H., Kobayashi, M. and Arita, K. Iterative Design of Seamless Collaboration Media, CACM, 37(8), (1994), 83-97.
- 9) Kelly, P.H., Katkere, A., Kuramura, D.Y., Moezzi, S. and Chatterjee, S. An Architecture for Multiple Perspective Interactive Video, Proc. Multimedia95, (1995), 201-212.
- 10) Kendon, A. Spatial Organization in Social Encounters: the F-formation System. A. Kendon, Ed., Conducting Interaction: Patterns of Behavior in Focused Encounters, Cambridge University Press, (1990), 209-237.
- 11) Kuzuoka H. Spatial Workspace Collaboration: a SharedView Video Support System for Remote Collaboration Capability. Proc. CHI92, (1992), 533-540.
- 12) Nakamura, T. and Ishiguro, H. Automatic 2D Map Construction using a Special Catadioptric Sensor. Proc. IEEE/RSJ IROS2002, (2002), 196-201.
- 13) Nakanishi, H. FreeWalk: A Social Interaction Platform for Group Behavior in a Virtual Space. IJHCS, (2004), Vol.60, No.4, 421-454.
- 14) Okada, K., Maeda, F., Ichikawa, Y. and Matsushita, Y. Multiparty Videoconferencing at Virtual Social Distance: MAJIC Design. Proc. CSCW94, (1994), 385-393.
- 15) Okazaki, S. and Matsushita, S. A Study of Simulation Model for Pedestrian Movement with Evacuation and Queuing. Proc. International Conference on Engineering for Crowd Safety, (1993), 271-280.
- 16) Reynolds, C.W. Flocks, Herds, and Schools: A Distributed Behavioral Model. Proc. SIGGRAPH87, (1987), 25-34.
- 17) Sugiman, T. and Misumi, J. Development of a New Evacuation Method for Emergencies: Control of Collective Behavior by Emergent Small Groups. Journal of Applied Psychology, 73(1), (1998), 3-10.
- 18) Tang, J.C. and Rua, M. Montage: Providing Teleproximity for Distributed Groups. Proc. CHI94, (1994), 37-43.
- 19) Tsutsuguchi, K., Shimada, S., Suenaga, Y., Sonehara, N. and Ohtsuka, S. Human Walking Animation based on Foot Reaction Force in the Three-dimensional Virtual World. Journal of Visualization and Computer Animation, 11(1), (2000), 3-16.
- 20) 中西 英之, 小泉 智史, 石田 亨, 伊藤 英明. 超越型誘導のための仮想都市シミュレータ. 第 18 回人工知能学会全国大会, (2004), 2G2-01.
- 21) 中西 英之, 小泉 智史, 石黒 浩, 石田 亨. 市民参加による避難シミュレーションに向けて. 人工知能学会誌, Vol. 18, No. 6, (2003), pp. 643-648.
 (平成 0 年 0 月 0 日受付)
 (平成 0 年 0 月 0 日採録)



伊藤 英明

1976 年生 . 1999 年京都大学工学部資源工学科卒業 . 2004 年京都大学情報学研究科社会情報学専攻博士課程修了 . 同年オムロン(株)入社 . 現在 , HCI の実用化研究に従事 . 人工知能 , 最適制御に興味を持つ .



中西 英之(正会員)

1996 年京都大学工学部情報工学科卒業 . 1998 年同大学院工学研究科情報工学専攻修士課程修了 . 同年日本学術振興会特別研究員 . 2001 年同大学院情報学研究科社会情報学専攻博士課程修了 . 博士(情報学) . 同年より同専攻助手 . 2000 年スタンフォード大学客員研究員 . 2005 年ジョージア工科大学客員研究員 . HCI, CSCW に興味を持つ . 2002 年度情報処理学会坂井記念特別賞 . 2004 年度人工知能学会全国大会優秀賞 . 2004 年度テレコムシステム技術賞 .



小泉 智史(正会員)

2000 年東京工業大学大学院総合理工学研究科知能システム科学専攻修了 . 工学博士 . 同年より科学技術振興事業団 CREST 「デジタルシティ のユニバーサルデザイン」プロジェクト研究員 . 現在(株)国際電気通信基礎技術研究所知能ロボティクス研究所研究員 . 人工知能 , 画像処理 , 全方位視覚に興味をもつ .



石田 亨(正会員)

1976 年京都大学工学部情報学工学科卒業 , 1978 年同大学院修士課程修了 . 京都大学大学院情報学研究科社会情報学専攻教授 . 工学博士 . IEEE Fellow . 情報処理学会フェロー . 人工知能 , コミュニケーション , 社会情報システムに興味を持つ .

仮想訓練システムのためのエージェントのモデル構築

Constructing Agent Model for Virtual Training Systems

村上 陽平
Yohei Murakami

京都大学大学院情報学研究科社会情報学専攻
Department of Social Informatics, Kyoto University
yohei@kuis.kyoto-u.ac.jp, http://www.lab7.kuis.kyoto-u.ac.jp/~yohei/

杉本 悠樹^{*1}
Yuki Sugimoto

(同 上)
sugimoto@kuis.kyoto-u.ac.jp

石田 亨
Toru Ishida

(同 上)
ishida@i.kyoto-u.ac.jp, http://www.lab7.kuis.kyoto-u.ac.jp/~ishida/

keywords: multiagent systems, software agent, virtual reality

Summary

Constructing highly realistic agents is essential if agents are to be employed in virtual training systems. In training for collaboration based on face-to-face interaction, the generation of emotional expressions is one key. In training for guidance based on one-to-many interaction such as direction giving for evacuations, emotional expressions must be supplemented by diverse agent behaviors to make the training realistic. To reproduce diverse behavior, we characterize agents by using a various combinations of operation rules instantiated by the user operating the agent. To accomplish this goal, we introduce a user modeling method based on participatory simulations. These simulations enable us to acquire information observed by each user in the simulation and the operating history. Using these data and the domain knowledge including known operation rules, we can generate an explanation for each behavior. Moreover, the application of hypothetical reasoning, which offers consistent selection of hypotheses, to the generation of explanations allows us to use otherwise incompatible operation rules as domain knowledge. In order to validate the proposed modeling method, we apply it to the acquisition of an evacuee's model in a fire-drill experiment. We successfully acquire a subject's model corresponding to the results of an interview with the subject.

1. はじめに

近年，協調作業やリーダシップの養成を目的とした集団作業の訓練に，仮想訓練システムが用いられ始めている [Rickel 99, Traum 02] . これは，実世界で必要となる複数の被験者をエージェントで代用することで，集団作業の訓練に伴う実施コストを削減できるためである。ただし，このような訓練では，仮想空間内においてエージェントと学習者との間にインタラクションが発生するため，エージェントによる，説得力のある社会的インタラクションの実現が必要とされる [Johnson 00, Cavazza 02] .

これまでエージェントの現実的なインタラクションは，非言語的な振舞いや，感情的な振舞い，発話といった表現の仕方をより実際の人間に似せることで実現されていた [Cassell 00, Marsella 02, Lester 00, Pelachaud 02] . これは，従来のアプローチが協調作業などで見られる face-to-face のインタラクションに焦点を当てていたためである。一方，避難時の指示のように，一対多のインタラク

ションに基づく誘導方法の訓練では，避難者エージェントは感情表現だけでなく，エージェントごとに多様に振舞うことも訓練を現実的なものにするために必要である [Wray 03, Sukthankar 04] . これに対し，声量や動きの速さといった身体的な違いを再現することで行動の多様性を実現することはできるものの，性格の違いによる多様性の実現は容易ではない。

しかし，実世界では人の指示に率先して従う者や，なかなか従わず周りに合わせる者がいるように，同じ状況であっても性格により人それぞれ振舞いが異なる場合が多い。訓練をより現実的にするには，このような性格の違いによる多様なエージェントの振舞いを実現することも重要である。そこで本研究では，訓練において多様なエージェントを構築するために，ユーザがエージェントを操作するユーザ参加型のシミュレーションを実施し，その操作ログから，ドメインに依存した多様なユーザの操作モデルを獲得することを目的とする。なお，ここでの操作モデルとは，エージェントの一人称視点でユーザが何を観測し，どのような操作を行ったかを表す操作ル

*1 現職：西日本電信電話株式会社

ルの集合と定義する。したがって、性格の違いは被験者の利用した操作ルールの違いとしてモデル化される。このような目的を達成するために、我々が取り組んだ課題は次の二点である。

ユーザ固有のモデルの獲得 訓練の設定に適合した多様なエージェントを構築するために、ユーザ参加型のシミュレーションからユーザごとの操作モデルを獲得する必要がある。そこで、全被験者のログデータから一般的な操作モデルを帰納的に獲得するのではなく、各被験者のログデータからそのログデータを説明できる被験者固有の操作モデルを獲得する。

モデルの一貫性の保証 性格の異なるユーザの操作モデルを獲得するために、多種多様な操作ルールが必要である。ただし、そのような操作ルールの中には、互いに両立することのない操作ルールも存在するため、モデル構築時には、モデルの一貫性を保ちつつ操作ルールを選定しなければならない。

本研究では上記の課題を踏まえて、操作モデルの獲得に仮説推論の枠組を適用する。仮説推論は、これまで故障や病気の診断に適用してきた実用的な枠組である [Poole 87]。故障の原因や病因が症状を生むという関係を用いて、可能な原因を仮説として立てて推論を進め、観測結果を論理的帰結として説明できた場合に、仮説が正しかったとする推論である。「人は各状況で実行可能な自分の行動ルールのうち最も優先度の高いものを実行する」といった合理的な意思決定モデルに基づけば、被験者の行動も自身の操作ルールとその優先度によって決定されると言える。したがって、診断問題と同様に被験者の操作モデルの獲得は、観測された一人の被験者の振舞い(観測結果)を矛盾なく導く、操作ルールとその優先度の組合せ(仮説の組合せ)を見出す探索問題として定式化できる。

なお、ここでの矛盾とは、同じ観測条件で動作の異なる操作ルールを一人の被験者が利用することを指す。これは、全く同じ状況に対して異なる行動をとることを意味し、矛盾した振舞いと言える。仮説推論では、このような矛盾を規制するために制約知識を記述することができる。操作ルールの選定時に、この知識により常に無矛盾性を確認することで、最終的に獲得される操作モデルの一貫性が保証される。

このように仮説推論を適用することで、被験者が利用する可能性のある操作ルールと被験者の振舞いさえ収集できれば操作モデルの獲得が可能になる。社会心理学に関する専門家の知識を用いて、観測された振舞いから利用された操作ルールを演繹的に推論することも可能であるが、これらの知識には経験的な知識も多く含まれており、知識獲得がボトルネックとなる。さらに、訓練という目的上、ドメインや環境に依存した被験者の操作モデルを獲得しなければならないことが、知識の獲得をより困難にしている。このような理由から、被験者の操作モデルの獲得には仮説推論の適用が有効であると考えられる。

以下、本論文では初めに参加型シミュレーションについて説明し、参加型シミュレーションを用いた被験者のモデリングのプロセスについて述べる。次に本研究で用いる技術的な語彙を定義し、それらの語彙を用いて、仮説推論を利用したモデリング手法について説明する。最後に、避難者役のユーザの操作モデルを獲得するために、避難ドメインに提案手法を適用し、その有用性を確認する。

2. 参加型シミュレーション

参加型シミュレーションとは、仮想空間上で行われるマルチエージェントシミュレーションにおいて、一部のエージェントをユーザが完全に操作するものである。このようにして仮想空間を人間とエージェントが共有することで、仮想空間内でエージェントとユーザのインタラクションを間接的に実現することができる。本研究では、仮想訓練システム上のエージェントを人間が操作する。

このような参加型シミュレーションを用いて被験者の操作モデルを獲得する利点として次の三点が挙げられる。

- 各被験者の操作履歴だけでなく、シミュレーション時に被験者が観測した環境の情報も、ログデータから算出し獲得することができる。
- シミュレーション時の被験者の画面を録画することで、被験者の状況を完全に記録することができる。録画した動画を被験者に見せながらインタビューすることで、被験者から効果的に操作ルールを収集する。
- 被験者の不足を補うために、プログラムの制御するエージェントで被験者を代用することができる。

本研究では、この参加型シミュレーションを用いてインタビューを行った結果得られた操作ルールを領域知識として用いて、参加型シミュレーションのログデータから導かれた各被験者の振舞い(観測事象)を説明することで、各被験者の実行した操作ルールを獲得する。ただし、モデリング対象の被験者が領域知識内のどの操作ルールを備えているかが不明であるため、その被験者がある操作ルールを実行したと一旦仮説を立て推論し、矛盾なく説明が達成されれば立てた仮説を真とする仮説推論を用いて操作モデルを求める。このような参加型シミュレーションと仮説推論を組み合わせたモデリングのプロセスを以下に示す(図1)。

ステップ1：マルチエージェントシミュレーション 文献や専門家へのインタビューなど事前知識により構築したモデルを評価するために、マルチエージェントシミュレーションを行う。

ステップ2：参加型シミュレーション 一部のエージェントを被験者が操作し、参加型シミュレーションを実施する。

ステップ3：観測事象の記述 参加型シミュレーションのログデータから、観測事象となる被験者の振舞いを抽出し、述語論理により記述する。

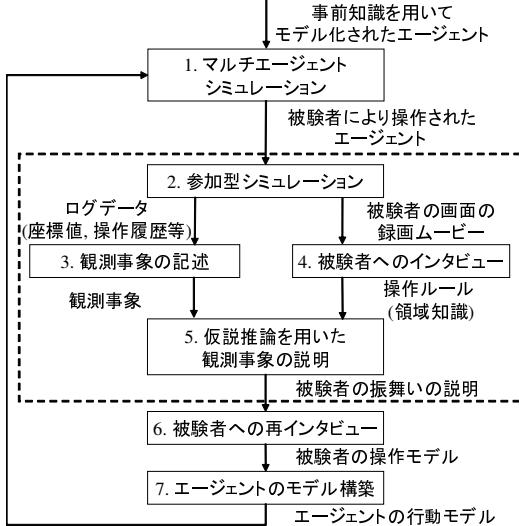


図 1 モデリングプロセス

ステップ 4：被験者へのインタビュー 被験者の操作画面を録画した動画を用いて、一部の被験者にインタビューを行い、領域知識となる操作ルールを獲得する。
ステップ 5：仮説推論を用いた観測事象の説明 領域知識と観測事象、そして利用した操作ルールに関する仮説を用いて、被験者の振舞いを説明する操作モデルの候補を仮説推論により獲得する。
ステップ 6：被験者への再インタビュー モデリング対象の被験者に対する質問を通して、生成された複数の説明から一意に説明を絞り込む。その説明に用いられた仮説から、被験者の操作モデルを獲得する。
ステップ 7：エージェントのモデル構築 獲得された操作モデルによって生成された振舞いの説明を専門家が分析し、エージェントの行動モデルを構築する。
 本論文では、特に点線で囲まれた、仮説推論による操作モデル獲得の部分に焦点を当てる。

3. 問題の形式化

本章では、被験者の操作モデル獲得に仮説推論を適用するために、領域知識と観測事象を形式的に定義する。

3.1 用語の定義

本研究では、被験者は、操作しているエージェントの一人称視点で観測された世界を基に、自分の操作モデルに従って次の操作を決定していると仮定する。被験者が観測した世界を S で表す。 S は、世界を記述したリテラルの連言からなる。特に、時刻 t における記述を S_t と表す。なお、ここでの時刻とは、被験者の観測した世界 S が変化するごとに 1 進む離散値とする。一方操作モデルは、モーデリング対象の被験者により実行される操作ルール集合 P と、操作ルール間の優先度に関する順序関係 \preceq の対 $\langle P, \preceq \rangle$ で表される。 P は、被験者へのインタビューから得

られた全操作ルールの集合 $Rules$ の部分集合であり、 \preceq は、直積集合 $Rules \times Rules$ の部分集合である。 $Rules$ 内の各操作ルールを $rule_i$ ($0 \leq i < j \leq |Rules|$) とするとき、 $\langle rule_i, rule_j \rangle \in \preceq$ のとき関係は $rule_i \preceq rule_j$ と記述することもできる。

このような被験者の操作モデル獲得に仮説推論を適用するには、被験者の操作ルールや操作の選択方法などの、常に成り立つ知識を領域知識 Σ として定義し、 Σ 内の各知識を σ_k ($0 \leq k \leq |\Sigma|$) と記述する。一方、各操作ルールが被験者により利用されたかどうか ($rule_i \in P$) や、どのルールが優先されるか ($rule_i \preceq rule_j$) といった、必ず成り立つとは限らない、他と矛盾する可能性のある知識は仮説とし、仮説の集合を H で表す。さらに、参加型シミュレーションの開始時刻 0 から終了時刻 end までの被験者の振舞いを観測事象 G として与える。最終的に仮説推論により導かれた解 h から、被験者の操作モデル $\langle P, \preceq \rangle$ を構築する。なお、 h は H の部分集合である。

3.2 領域知識

被験者の操作モデル獲得に仮説推論を適用する場合、領域知識は、被験者へのインタビューで獲得された操作ルール、被験者の操作の選択に関する知識、そして仮説によって生じる矛盾を防ぐための制約知識から構成される。

本研究では、操作ルールを条件-動作ルールとして記述する。被験者は、条件部の全てのリテラルが充足するときに、動作部の操作を行う。具体的な操作ルールの記述例を例 1 に示す。

例 1 (条件-動作ルールによる操作ルールの記述)。

$rule_1 : if Near(x, self), Noop(x), Noop(self) then Initiate(walk)$
 $rule_2 : if Near(x, self), Walk(x), Noop(self) then Initiate(walk)$
 $rule_3 : if Near(x, self), Noop(x), Noop(self) then Initiate(turn)$

$rule_1$ は、「操作しているエージェント (self) の停止中 (Noop) に、近く (Near) のあるエージェント x も立ち止っていれば (Noop)、そのエージェントを追い越すよう、歩かせる (Initiate(walk))」という操作ルールである。次に $rule_2$ は、「操作しているエージェントの停止中に、近くのあるエージェント x が歩いていれば、そのエージェントに付いて行くよう歩かせる」という操作ルールである。最後の $rule_3$ は、「操作しているエージェントの停止中に、近くのあるエージェント x も立ち止っていれば、周囲を見回せるよう体の向きを変える (Initiate(turn))」という操作ルールである。

次に、被験者の操作選択に関する知識について定義する。定義 1 (被験者の操作の可能性 1: 操作の選択 σ_1)。

被験者の観測した世界 S_t において適用可能 (Applicable) なルールの中で、優先度の最大な操作ルール $rule_i$ を被験者は利用する^{*2}。その結果、時刻 t において被験者は

^{*2} 本研究では理解を容易にするため単純な行動決定規則を利用したが、恒真であれば複雑なものでも領域知識に新たに追加することで比較的容易に拡張可能である。

$rule_i$ の動作部の操作を行う ($Do(action(rule_i))$) . なお , $Applicable$ と Do は , それぞれ操作ルールの条件部が充足したことと , 動作部の操作を行ったことを示す述語である . また , 関数 $action$ は , 操作ルールの動作部で被験者が起動したエージェントの機能を値として返す .

$$(\exists rule_i (rule_i \in P \wedge rule_i = \max_{\preceq} \{rule | Applicable(rule, S_t)\})) \Rightarrow Do(action(rule_i))$$

定義 2 (被験者の操作の可能性 2:操作の継続 σ_2) .

被験者は時刻 t よりも前に行つたある操作を , 時刻 t においても継続して行うことがある . このような被験者の操作を示す述語として $Continue$ を利用する .

ただし , 常に $Continue$ が成り立つとすると , 被験者の操作が Do であるか , $Continue$ であるかを説明することができない . そこで , 各時刻において , 被験者がとりうる操作は Do もしくは $Continue$ のどちらか一方とし , σ_1 と σ_2 は排他的な知識とする .

最後に , 仮説によって生じる矛盾を防ぐために制約知識を導入する . 今回用いる制約知識は , 被験者の操作ルールに関する以下のものである .

定義 3 (制約知識 σ_3) .

被験者の利用する操作ルール P のうち , 条件部の等しい全ての操作ルールは , その動作部の操作も等価である . なお関数 $condition$ は , 操作ルールの条件部であるリテラルの集合を値として返す .

$$\begin{aligned} \forall rule_i, rule_j (rule_i, rule_j \in P \wedge \\ (condition(rule_i) = condition(rule_j)) \\ \Rightarrow (action(rule_i) = action(rule_j))) \end{aligned}$$

3・3 観測事象の記述

観測事象 G は , 被験者の観測した世界 S が変化するごとに観測される . したがって , 時刻 t における観測事象を G_t とすると , 観測事象 G, G_t はそれぞれ次のように定義することができる .

定義 4 (観測事象 G) .

$$G \equiv (G_0 \wedge \dots \wedge G_t \wedge \dots \wedge G_{end})$$

定義 5 (時刻 t における観測事象 G_t) .

観測事象 G_t は , エージェントを通して , 被験者がどのような世界を観測し , エージェントをどう操作したかが記述される . 時刻 t における被験者の操作を A_t で表す . 具体的に A_t は , 被験者がある操作を行うことを意味する Do か , ある操作を継続することを意味する $Continue$ で表現されたリテラルのどちらかである .

$$G_t \equiv (S_t \Rightarrow A_t)$$

これらの観測事象はログデータから時系列に沿って獲得され , 述語論理で記述される . 仮想空間という性質上 , ログデータはエージェントの座標値や向き , 被験者のキー入力から構成される . 例えば , 参加型シミュレーションにおいて , ある被験者が操作するエージェント John が

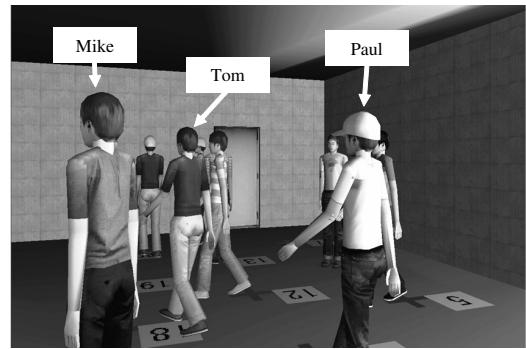


図 2 時刻 $T - 1$ におけるエージェント John の視界

歩き出した時刻を T とする . 時刻 $T - 1$ において , エージェント John の視界が図 2 で , かつ John が立ち止っており , 被験者が John を歩かせようと操作しているとき , G_{T-1} は次のように記述される .

例 2 (観測事象 G_{T-1} の記述) .

$$\begin{aligned} Near(Mike, John) \wedge Near(Paul, John) \wedge Far(Tom, John) \wedge \\ Noop(Mike) \wedge Walk(Paul) \wedge Walk(Tom) \wedge ToLeft(Paul) \wedge \\ ToLeft(Mike) \wedge Forward(Tom) \wedge InFrontOf(Paul, John) \wedge \\ InFrontOf(Mike, John) \wedge InFrontOf(Tom, John) \wedge \dots \wedge \\ Noop(John) \Rightarrow Do(walk) \end{aligned}$$

なお , 例 2 で用いた述語は次の 4 種類に分けられる .

距離 $Near(x, y), Far(x, y)$:

エージェント x はエージェント y の近く/遠くにいる .

位置関係 $InFrontOf(x, y)$:

被験者の操作するエージェントから見て , エージェント x はエージェント y の前にいる .

向き $Forward(x), ToLeft(x)$:

被験者の操作するエージェントから見て , エージェント x は前/左を向いている .

動作 $Noop(x), Walk(x)$:

エージェント x は立ち止って/歩いている .

4. 操作モデルの獲得

これまで述べてきた領域知識と観測事象を用いて , 仮説推論により被験者の操作モデルの獲得を試みる . 具体的には , 操作ルールと被験者の操作に関する領域知識 Σ と , 被験者の利用する操作ルールとそのルール間の優先順序に関する仮説 H を用いて , 観測事象 G を説明する . これにより以下の 3 つの条件を満たす解の仮説集合 h を求める . なお本研究では , h から導かれる , 優先度付きの操作ルール集合 $\langle P, \preceq \rangle$ が , 被験者の操作モデルを表す .

(1) $h \cup \Sigma \vdash G$

(2) $\Sigma \cup h$ が無矛盾 .

(3) h の部分集合が上記の条件を満さない .

定義 4 により G が時系列に沿って得られた観測事象 G_t の連言であることから , 1 番目の条件は , 証明可能性の推移則を適用して $h \cup \Sigma \vdash G_0, \dots, h \cup \Sigma \vdash G_t, \dots, h \cup \Sigma \vdash$

G_{end} と変換できる。さらに、定義 5により G_t は $S_t \Rightarrow A_t$ であるから、演繹定理を適用して、 $h \cup \Sigma \cup \{S_t\} \vdash A_t$ を示せば良い。例えば、例 2 で示した観測事象 G_{T-1} の証明は、 $h \cup \Sigma \cup \{S_{T-1}\} \vdash Do(walk)$ に変換される。この論理式の説明の構造を図 3 に示す。ただし、 $Rules$ は例 1 で示した $\{rule_1, rule_2, rule_3\}$ であり、一方、仮説集合 h として、 G_0, \dots, G_{T-2} の証明から $h_{T-2} = \{rule_3 \in P\}$ が得られていると仮定する。図 3 に示す説明構造の生成プロセスは次のとおりである。

- (1) $Do(walk)$ を証明するには、 σ_1 により、 $action(rule_i) = walk$, $rule_i \in P$, そして $rule_i = \max_{\leq} \{rule | Applicable(rule, S_{T-1})\}$ が全て真であることを示さなければならない。
- (2) $action(rule_i) = walk$ が成り立つために、動作部の操作が $Initiate(walk)$ である $rule_1, rule_2$ が $rule_i$ の候補となる。
- (3) $rule_i$ に $rule_1$ を代入した場合。
 - a $rule_1 \in P$ を証明するために、 H から $rule_1 \in P$ を選定する。しかし、これでは P に含まれる $rule_1$ と $rule_3$ の条件部が等しく、各ルールの動作部の操作が等価ではないので、 σ_3 が成り立たずバックトラックする。
- (4) $rule_i$ に $rule_2$ を代入した場合。
 - a $rule_2 \in P$ を証明するために、 H から $rule_2 \in P$ を選定する。
 - b $rule_2 = \max_{\leq} \{rule | Applicable(rule, S_{T-1})\}$ を証明するために、 H から $rule_1 \leq rule_2$ と $rule_3 \leq rule_2$ を選定する。
 - c $h_{T-1} = \{rule_2 \in P, rule_3 \in P, rule_1 \leq rule_2, rule_3 \leq rule_2\}$

$$S_{T-1} = \{Near(Mike, John) \wedge Near(Paul, John) \wedge \dots \wedge Noop(Mike) \wedge Walk(Paul) \wedge \dots \wedge Noop(John)\}$$

$$A_{T-1} = Do(walk), h_{T-2} = \{rule_3 \in P\}$$

$$Rules = \{rule_1 : \text{if } Near(x, self), Noop(x), Noop(self) \text{ then } Initiate(walk)\}$$

$$\{rule_2 : \text{if } Near(x, self), Walk(x), Noop(self) \text{ then } Initiate(walk)\}$$

$$\{rule_3 : \text{if } Near(x, self), Noop(x), Noop(self) \text{ then } Initiate(turn)\}$$

$$H = \{rule_1 \in P, rule_2 \in P, \dots, rule_1 \leq rule_2, rule_2 \leq rule_1, rule_3 \leq rule_2, \dots\}$$

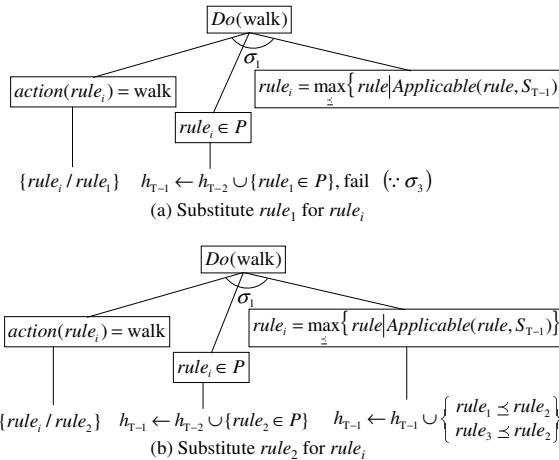


図 3 $h \cup \Sigma \cup \{S_{T-1}\} \vdash Do(walk)$ の説明の構造

Algorithm 1 OperationModel(t) return (P, \preceq)

```

1:  $t /*$  時刻 */
2:  $P /*$  被験者の利用する操作ルールの集合 ( $P \subseteq Rules$ ) */
3:  $\preceq /*$  操作ルールの優先度に関する順序関係 */
4:  $S_t /*$  時刻  $t$  における被験者の観測した世界 */
5:  $A_t /*$  時刻  $t$  における被験者の操作 */
6: if  $t < 0$  then
7:   return ( $\{\}, \{\}$ )
8: end if
9:  $(P, \preceq) \leftarrow OperationModel(t - 1)$ 
  /* 被験者の操作が時刻  $t - 1$  と同じ場合 */
10: if  $A_t = Continue$  then
11:   return ( $P, \preceq$ )
12: end if
  /* 時刻  $t$  で被験者が新しい操作を開始した場合 */
  /* 極小なモデルの構築のために  $P$  内の操作ルールから選択 */
13: if choose  $p \in \{rule | Applicable(rule, S_t)\}$ ,
     $A_t = Do(action(rule)) \cap P$  then
14:    $\forall r \in \{rule | Applicable(rule, S_t)\}$  に対し,  $p$  が  $r \preceq p$  となるように  $\preceq$  を更新する
15:   return ( $P, \preceq$ )
16: end if
  /*  $P$  以外の新しい操作ルールから選択 */
17: if choose  $p \in \{rule | Applicable(rule, S_t)\}$ ,
     $A_t = Do(action(rule)) \setminus P$  then
18:    $P \leftarrow P \cup \{p\}$ 
19:   if Inconsistent?( $P$ ) then
20:     fail
21:   end if
22:    $\forall r \in \{rule | Applicable(rule, S_t)\}$  に対し,  $p$  が  $r \preceq p$  となるように  $\preceq$  を更新する
23:   return ( $P, \preceq$ )
24: end if
25: fail

```

Algorithm 2 Inconsistent?(P) return boolean value

```

1: if  $\exists p_i, p_j (p_i, p_j \in P \wedge (condition(p_i) = condition(p_j) \wedge$ 
   $\neg (action(p_i) = action(p_j)))$  then
2:   return true
3: end if
4: return false

```

$rule_3 \leq rule_2$ } が得られる。

上記のプロセスを参加型シミュレーションの終了時刻まで続けることで、John を操作した被験者の操作モデルを獲得することができる。操作モデルを獲得する関数 OperationModel のアルゴリズムを Algorithm1 に示す。

関数 OperationModel の目的は、引数として与えられた時刻 t までの観測事象を矛盾なく説明する、被験者の極小な操作モデルを求めることがある。ここで矛盾とは、条件部が等しく動作部の異なる操作ルールが、被験者の利用する操作ルールの集合 P の中に存在することを意味する。したがって、本アルゴリズムでは、 P が更新されるたびに、Algorithm2 を用いて P 内に矛盾がないかを確認する（ステップ 19）。 P の要素は関数 choose を用いて非決定的に選択されるため、矛盾が発生した場合は、choose で選択されなかった他の候補が選択される。なお、choose で選択する候補が存在しなかった場合は、関数 choose は false を返す。

また、極小な操作モデルを求めるために、関数 OperationModel は、時刻 $t - 1$ で求めた P を用いて、時刻 t

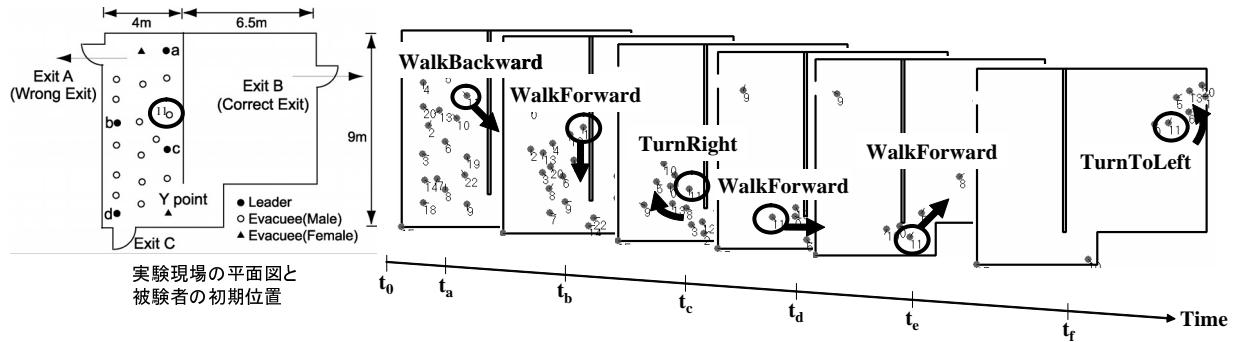


図4 エージェント11を操作した被験者の操作ログ(各時刻のスクリーンショット(鳥瞰図)は、参加型シミュレーション後のインタビューで、被験者から操作ルールを獲得した場面と対応する)

の被験者の操作の説明を試みる(ステップ13-16)。時刻 $t-1$ で求めた P だけでは、時刻 t の被験者の操作を説明できなかった場合に、新たな操作ルールを選択する(ステップ17-24)。万が一、全ての候補で説明できなかった場合はfailを行い、時刻 $t-1$ の P を再度求め直す。

5. 応用例

本章では、提案したモデリングプロセスを避難誘導のドメインに適用し、その有用性について述べる。

5.1 避難ドメイン

仮想空間における避難誘導訓練に必要な避難者エージェントを構築するために、本研究で提案したモデリングプロセスを、参加型シミュレーションで避難者役を担った被験者の操作モデルの獲得に適用した。ただし、ここで構築する避難者エージェント群は、制御不能なパニック状態にある一様なエージェントではなく、相手を優先させたり、周囲に同調したりするなど、ある程度理性の機能した多様な避難者エージェントを対象とする。これは、パニックが起こる前にいかにして速やかに正しい出口に避難者を誘導するかを訓練の課題としているためである。

具体的には、実世界で行われた避難誘導に関する統制実験[Sugiman 88]を例題として用いた。この実験の行われた部屋の構造と、避難者と誘導者の初期位置を図4の左側に示す。全ての被験者と誘導者が入室後出口C(ExitC)を閉め、避難の開始とともに出口A(ExitA)と出口B(ExitB)を開ける。出口Aは全ての避難者から認識可能だが、出口Bは最初の時点で誘導者しかその存在を知らない。避難者は誘導者によって出口Bに誘導される。

本研究では、まず実験の環境を三次元仮想空間プラットフォームであるFreeWalk[Nakanishi 04]上に再現した。次に、16体の避難者エージェントのうち12体を人間の操作するエージェントに置き換え、参加型シミュレーションを行った(図5)。参加型シミュレーションから得られたログデータに対し、提案手法を用いて避難者エージェントを操作した被験者の操作モデルを獲得する。なお、残りの4体の避難者エージェントと4体の誘導者エー

ジエントの振舞いは、シナリオ記述言語 Q で記述したシナリオにより制御した[Ishida 02, Murakami 03]。

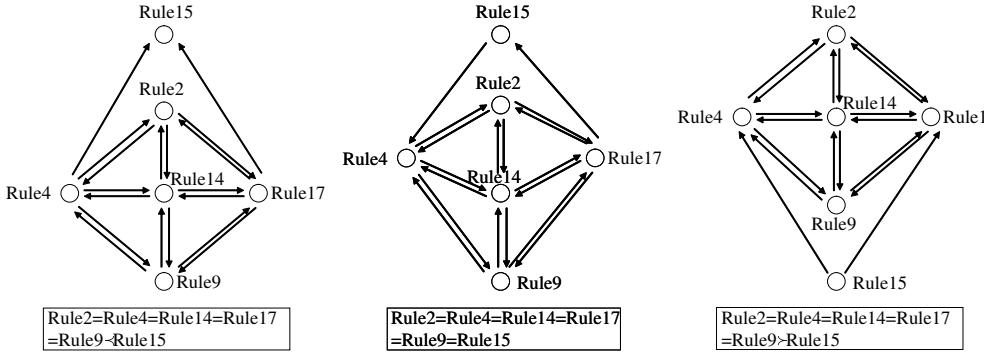
5.2 モデリング手法の適用

避難者エージェント11(図4の円で囲まれたエージェント)を操作した被験者の操作ログを用いて観測事象を記述し、仮説推論により観測事象の説明を試みた。なお、領域知識として用いた操作ルールは、6人の被験者へのインタビューで得られた以下の18個のルールである。

- Rule1: 動いている人の方向を向く。
- Rule2: 誘導者が見えなければ、周囲を見回したり、後ろに下がったりして誘導者を探す。
- Rule3: 大勢の人が向いている方に向かう。
- Rule4: 出口が見えないとき、周囲を見回す。
- Rule5: 誘導者を見かけたら、その誘導者に近づく。
- Rule6: 近くの誘導者に付いていく。
- Rule7: 混雑しているところに向かう。
- Rule8: 誘導者の動きを観測する。
- Rule9: 誘導者が向かった方向に向かう。
- Rule10: 大勢の人が向かった方向に向かう。
- Rule11: 歩いている前方に混雑があると立ち止まる。
- Rule12: 同じ方向に向かっている近くの人が歩けば、前に進む。
- Rule13: 同じ方向に向かっている近くの人が立ち止まれば、歩くのを止める。



図5 FreeWalkを用いた参加型シミュレーションの様子



Rule14: 前にいる人について行く .

Rule15: 歩いている前方に混雑があるとすり抜けて前に行こうとする .

Rule16: 歩行中に前の人気が立ち止まると追い抜く .

Rule17: 出口が見えれば、出口の方に向かう .

Rule18: 出口に向かう前に再度周囲を見回す .

避難者エージェント 11 の操作ログの説明から得られた操作ルールの集合を表 1 に示す . 各操作ルール集合は , 論理に基づいて獲得されているため , 観測された事象に対して無矛盾な説明を生成すること (正当性) は保証される . 一方 , その妥当性に関しては , 録画した画面を操作者に見せながら逐次収集した操作ルールと比較することで検証される . 今回は検証の結果 , 収集された操作ルールと P_1 が一致することが確認された . ただし , このような妥当性の検証を全被験者に対して行うのはコストが大きく現実的ではない .

そこで , 獲得された操作ルール集合を活用し妥当な操作ルール集合を獲得するために , 質問・応答機構を用いた絞り込み手法が有効と考えられる . これは , 操作ルールの集合を二手に分け , 一方の操作ルール集合でのみ説明できる , ログデータには無い新しい状況設定とその時のエージェントの操作を作成し , 操作者に質問するものである . 例えば , 表 1 の場合「出口に向かう誘導者を見たとき , 出口に向かうか . それとも周囲を見渡すか . 」と質問することで , その答えにより $\langle P_1, P_2 \rangle$ か $\langle P_3, P_4 \rangle$ のどちらかに絞り込むことが可能である . このような質問・応答を繰り返すことで , 最終的に一つの操作ルール集合に絞り込んでいくことができる . このように , 複数の操作ルール集合を獲得しようとも , 最終的に最も被験者のモデルに類似したものを選択できる絞り込みルールを , 自動的に生成することが今後の課題である . なお , 最終

的にどのモデルも残らなかった場合は , 領域知識の操作ルールの数が十分でなかったと考えられ , 再インタビューにより新しい操作ルールの獲得を試みる必要がある .

一方 , P_1 に含まれる操作ルール間の優先順序は , 図 6 に示すように 3 種類獲得された . 図 6 では , 操作ルール間の優先順序を矢印で示しており , その矢印の向きは優先順序の高い方を表す . なお , $\text{Rule1} \preceq \text{Rule2}$ かつ $\text{Rule2} \preceq \text{Rule1}$ の関係が成立つ場合は , $\text{Rule1} = \text{Rule2}$ と等価であり , 両方向に延びた矢印は優先順序が等しいことを意味する .

操作ルールの実行選択にこれら 3 種類の優先順序を用いると , どれも先ほどのインタビューで得られた操作ルールの実行順序と一致した . 具体的には , 図 4 の操作ログから算出された , 被験者の世界 $S_{t_a}, S_{t_b}, S_{t_c}, S_{t_d}, S_{t_e}, S_{t_f}$ において , 発火可能なルール $\{\text{Rule2}, \text{Rule14}, \text{Rule17}\}$, $\{\text{Rule4}, \text{Rule9}, \text{Rule14}\}$, $\{\text{Rule2}, \text{Rule4}\}$, $\{\text{Rule2}, \text{Rule4}, \text{Rule14}\}$, $\{\text{Rule9}, \text{Rule14}, \text{Rule17}\}$, $\{\text{Rule15}\}$ に対し , $\text{Rule2}, \text{Rule9}, \text{Rule4}, \text{Rule14}, \text{Rule17}, \text{Rule15}$ がそれぞれ選択され , インタビューにおいて各状態で得られた操作ルールと一致した . ただし , 図 6 に示した優先順序は , ほとんどのルールが同順位のため , 正しい実行順序を説明すると同時に他の実行順序の可能性も多く含み , 実行順序の適合度が低い . そこで , 優先順序を一意に定められるよう , 専門家の説明構造の分析による操作ルールの条件部の詳細化が必要である .

このように獲得された操作モデルは , 一見自明で事前知識による構築も可能であるように考えられる . ただし , 蓄積された事前知識は様々な人から様々なケースで収集された知識であり , 人のタイプによってどのような行動ルールを持ち得るかが明確に示されていなかった . 本研究は , 参加型シミュレーションで獲得された実際のデータに則し , ある状況設定の下で各被験者が利用した操作ルールの推定を可能にした点で有用と考えられる .

6. おわりに

エージェントを用いた仮想訓練システムをより現実的なものにするには , エージェントの振舞いを視覚的に忠実に再現するだけでなく , 多様なエージェントモデルを

表 1 エージェント 11 を操作した被験者の操作ルール集合

P	操作ルールの集合
P_1	$P = \{\text{Rule2}, \text{Rule4}, \text{Rule9}, \text{Rule14}, \text{Rule15}, \text{Rule17}\}$
P_2	$P = \{\text{Rule2}, \text{Rule4}, \text{Rule9}, \text{Rule14}, \text{Rule16}, \text{Rule17}\}$
P_3	$P = \{\text{Rule2}, \text{Rule4}, \text{Rule14}, \text{Rule15}, \text{Rule17}, \text{Rule18}\}$
P_4	$P = \{\text{Rule2}, \text{Rule4}, \text{Rule14}, \text{Rule16}, \text{Rule17}, \text{Rule18}\}$

構築することも必要である。本研究では、多様なエージェントモデルの構築に向けた、最初のステップとして、エージェントを操作した被験者の操作モデルを獲得することを目的として、以下の二つの課題に取り組んだ。

ユーザ固有のモデルの獲得 参加型シミュレーションのログデータを用いて各被験者の操作を説明することで、個々の被験者の操作モデルを獲得することが可能になった。説明に用いられた操作ルールの組合せとルール間の優先順序により、各被験者は特徴付けられる。

モデルの一貫性の保証 各操作ルールが、モデリング対象の被験者によって利用されたかどうかを仮説とし、仮説の矛盾のない選択を保証する仮説推論に基づいたモデリングプロセスを確立した。このプロセスにより、モデルの一貫性を保ちながら、多様な操作ルールを用いて被験者の操作モデルを構築することが可能になった。

謝 詞

京都大学の杉万俊夫教授、中西英之助手に多大な御協力を頂きましたことをここに感謝します。また、本研究は、日本学術振興会科学研究費基盤研究(A)(15200012, 2003-2005)の補助を受けました。なお、FreeWalk/Qは、科学技術振興機構、戦略的創造研究推進事業の「デジタルシティのユニバーサルデザインプロジェクト」により開発されました。

◇ 参考文献 ◇

- [Cassell 00] Cassell, J., Bickmore, T., Campbell, L., Vilhjalmsson, H., and Yan, H.: Human Conversation as a System Framework: Designing Embodied Conversational Agents, in Cassell, J., Prevost, S., Sullivan, J., and Churchill, E. eds., *Embodied Conversational Agents*, pp. 29-63, MIT Press (2000)
- [Cavazza 02] Cavazza, M., Charles, F., and Mead, S.: Interacting with Virtual Characters in Interactive Storytelling, in *Proc. of the First International Conference on Autonomous Agents and Multi-agent Systems*, pp. 318-325 (2002)
- [Ishida 02] Ishida, T.: Q: A Scenario Description Language for Interactive Agents, *IEEE Computer*, Vol. 35, No. 11, pp. 54-59 (2002)
- [Johnson 00] Johnson, W., Rickel, J., and Lester, J.: Animated Pedagogical Agents: Face-to-Face Interaction in Interactive Learning Environments, *International Journal of Artificial Intelligence in Education*, Vol. 11, pp. 47-78 (2000)
- [Lester 00] Lester, J., Towns, S., Callaway, C., Voerman, J., and FitzGerald, P.: Deictic and Emotive Communication in Animated Pedagogical Agents, in Cassell, J., Prevost, S., Sullivan, J., and Churchill, E. eds., *Embodied Conversational Agents*, pp. 123-154, MIT Press (2000)
- [Marsella 02] Marsella, S. and Gratch, J.: A Step Toward Irrationality: Using Emotion to Change Belief, in *Proc. of the First International Conference on Autonomous Agents and Multi-agent Systems*, pp. 334-341 (2002)
- [Murakami 03] Murakami, Y., Ishida, T., Kawasoe, T., and Hishiyama, R.: Scenario Description for Multi-Agent Simulation, in *Proc. of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 369-376 (2003)
- [Nakanishi 04] Nakanishi, H.: FreeWalk: A Social Interaction Platform for Group Behaviour in a Virtual Space, *International Journal of Human Computer Studies*, Vol. 60, No. 4, pp. 421-454 (2004)
- [Pelachaud 02] Pelachaud, C., Carofiglio, V., Carolis, B., Rossis, F., and Poggi, I.: Embodied Contextual Agent in Information Delivering Application, in *Proc. of the First International Conference on Autonomous Agents and Multiagent Systems*, pp. 758-765 (2002)
- [Poole 87] Poole, D.: Theorist: A Logical Reasoning System for Defaults and Diagnosis, in Cercone, N. and McCalla, G. eds., *The Knowledge Frontier*, Springer-Verlag (1987)
- [Rickel 99] Rickel, J. and Johnson, W.: Virtual Humans for Team Training in Virtual Reality, in *Proc. of the Ninth International Conference on Artificial Intelligence in Education*, pp. 578-585 (1999)
- [Sugiman 88] Sugiman, T. and Misumi, J.: Development of a New Evacuation Method for Emergencies: Control of Collective Behavior by Emergent Small Groups, *Journal of Applied Psychology*, Vol. 73, No. 1, pp. 3-10 (1988)
- [Sukthankar 04] Sukthankar, G., Mandel, M., Sycara, K., and Hodges, J.: Modeling Physical Capabilities of Humanoid Agents Using Motion Capture Data, in *Proc. of the Third International Conference on Autonomous Agents and Multi-agent Systems*, pp. 344-351 (2004)
- [Traum 02] Traum, D. and Rickel, J.: Embodied Agents for Multi-party Dialogue in Immersive Virtual Worlds, in *Proc. of the First International Conference on Autonomous Agents and Multi-agent Systems*, pp. 766-773 (2002)
- [Wray 03] Wray, R. and Laird, J.: Variability in Human Behavior Modeling for Military Simulations, in *Proc. of Behavior Representation in Modeling and Simulation Conference* (2003)

ence on Autonomous Agents and Multiagent Systems, pp. 369-376 (2003)

[Nakanishi 04] Nakanishi, H.: FreeWalk: A Social Interaction Platform for Group Behaviour in a Virtual Space, *International Journal of Human Computer Studies*, Vol. 60, No. 4, pp. 421-454 (2004)

[Pelachaud 02] Pelachaud, C., Carofiglio, V., Carolis, B., Rossis, F., and Poggi, I.: Embodied Contextual Agent in Information Delivering Application, in *Proc. of the First International Conference on Autonomous Agents and Multiagent Systems*, pp. 758-765 (2002)

[Poole 87] Poole, D.: Theorist: A Logical Reasoning System for Defaults and Diagnosis, in Cercone, N. and McCalla, G. eds., *The Knowledge Frontier*, Springer-Verlag (1987)

[Rickel 99] Rickel, J. and Johnson, W.: Virtual Humans for Team Training in Virtual Reality, in *Proc. of the Ninth International Conference on Artificial Intelligence in Education*, pp. 578-585 (1999)

[Sugiman 88] Sugiman, T. and Misumi, J.: Development of a New Evacuation Method for Emergencies: Control of Collective Behavior by Emergent Small Groups, *Journal of Applied Psychology*, Vol. 73, No. 1, pp. 3-10 (1988)

[Sukthankar 04] Sukthankar, G., Mandel, M., Sycara, K., and Hodges, J.: Modeling Physical Capabilities of Humanoid Agents Using Motion Capture Data, in *Proc. of the Third International Conference on Autonomous Agents and Multi-agent Systems*, pp. 344-351 (2004)

[Traum 02] Traum, D. and Rickel, J.: Embodied Agents for Multi-party Dialogue in Immersive Virtual Worlds, in *Proc. of the First International Conference on Autonomous Agents and Multi-agent Systems*, pp. 766-773 (2002)

[Wray 03] Wray, R. and Laird, J.: Variability in Human Behavior Modeling for Military Simulations, in *Proc. of Behavior Representation in Modeling and Simulation Conference* (2003)

〔担当委員：桜井 成一朗〕

2005年6月6日 受理

著者紹介

村上 陽平

2001年京都大学工学部情報学科卒。2003年同大学院社会情報学専攻修士課程修了。現在、同大学院社会情報学専攻博士課程に在学中。マルチエージェントの組織形成に興味を持つ。

杉本 悠樹

2003年京都大学工学部情報学科卒。2005年同大学院社会情報学専攻修士課程修了。同年、西日本電信電話株式会社入社。マルチエージェントシミュレーションに興味を持つ。

石田 亨(正会員)

1976年京都大学工学部情報工学科卒業、1978年同大学院修士課程修了。同年日本電信電話公社電気通信研究所入所。ミュンヘン工科大学、パリ第六大学、メリーランド大学客員教授など経験。工学博士。IEEE フェロー。情報処理学会フェロー。現在、京都大学大学院情報学研究科社会情報学専攻教授、上海交通大学客員教授。自律エージェントとマルチエージェントシステム、セマンティック Web 技術を取り組む。デジタルシティ、異文化コラボレーションプロジェクトを推進。

社会・環境シミュレーションにおけるインタラクション層の構築

Construction of Interaction Layer on Socio-Environmental Simulation

鳥居 大祐
Daisuke Torii

京都大学情報学研究科社会情報学専攻
Department of Social Informatics, Kyoto University
torii@kuis.kyoto-u.ac.jp, <http://www.lab7.kuis.kyoto-u.ac.jp/~torii/>

石田 亨
Toru Ishida

(同 上)
ishida@i.kyoto-u.ac.jp, <http://www.lab7.kuis.kyoto-u.ac.jp/~ishida/>

keywords: multiagent-based simulation, agent interaction, agent programming languages and environments

Summary

In this study, we propose a method to construct a system based on a legacy socio-environmental simulator which enables to design more realistic interaction models in socio-environmental simulations. First, to provide a computational model suitable for agent interactions, an interaction layer is constructed and connected from outside of a legacy socio-environmental simulator. Next, to configure the agents interacting ability, connection description for controlling the flow of information in the connection area is provided. As a concrete example, we realized an interaction layer by *Q* which is a scenario description language and connected it to CORMAS, a socio-environmental simulator. Finally, we discuss the capability of our method, using the system, in the Fire-Fighter domain.

1. はじめに

地球の温暖化、森林火災、地震災害、食料問題などは世界における環境・災害問題の大きなトピックであり、情報技術が果たす役割が期待されている。人と環境を含めた総合的な社会・環境シミュレーションが実現すれば、問題に対する政策や訓練などに反映することができる。

社会・環境シミュレーションでは、エージェントは他のエージェントとの交渉・協調や環境からの観測を元に自身の状態を変化させ、また、環境へ行動を行う。したがって、環境を表す計算モデルの他に、エージェント間やエージェントと環境間のインタラクションが重要な要素であり、シミュレーションを通じてそれらを十分検証・設計することが重要である。そのためには、インタラクションを扱う計算モデルを元に、様々なインタラクション記述を試せることが望ましい。

しかしながら、社会層・環境層のモデリングが可能な既存の社会・環境シミュレータの多くは、シミュレーションに便利なツールが様々用意され、利用価値の高いものであるが、エージェントのモデル記述については、その自由度を高めるために専用のAPIなどを用いた行動モデルを一般に記述するものであり、インタラクションを専用に扱う計算モデルを元にした記述を提供していない（図1参照）。したがって、既存の社会・環境シミュレータに

対し、インタラクション記述を与え、シミュレーションを行える環境を構築することが必要である。さらに、シミュレーションでの検証結果を現実の社会システムの設計などに反映させるためには、インタラクションをシナリオとして記述できるだけでは不十分であり、インタラクションに関わるエージェント個人の能力のばらつきや環境から受ける能力の制限を勘案することで、より現実に即したインタラクションを検証・設計できることが重要となる。

このような背景を受けて、本研究では、社会・環境シミュレーションにおいて、より現実に即したインタラクションモデルの検証・設計を行うことができるシミュレータを、既存の社会・環境シミュレータを利用して構築する手法を提供することを目的とする。のために、次のような接近法を取った。

- 1) インタラクションのみを独立に記述し、既存の社会・環境シミュレータでそれを利用するためには、エージェントがシナリオを解釈し、それに従って動作する機構が必要となる。しかし、既存の社会・環境シミュレータの拡張性を考えるとそれを内部に構築するのは現実的でない。そこで、シミュレータに大きな変更を加えず外部のインタラクション層にてシナリオを解釈しエージェントを制御するアーキテクチャを採用する。2) また、エージェントの個人能力を微調整するためには、既存の社会・

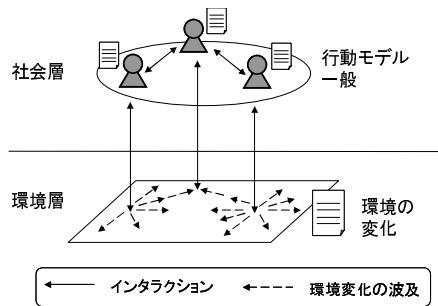


図1 従来の社会・環境シミュレータ

環境シミュレータの内部を変更する必要があるが、同様にその拡張性を勘案すると内部の変更は避けたい。そこで、本アーキテクチャにおけるインタラクション層と社会・環境シミュレータの間を流れる情報を制御することで、シミュレータの内部を変更せずより簡単に実現する手法(Connection Control)を提案する。

社会・環境シミュレーションの実現手法としては、ロボカップレスキー[Takahashi 02]やU-MART[Kita 03]が関連研究として挙げられる。環境としてそれぞれ災害現場や株取引市場を与え、そこにエージェントが参加するためのプロトコルを規定することで様々なエージェントの参加を可能としている。したがって、両者ともエージェントのインタラクション記述を既存の社会・環境シミュレータのエージェントに外部から与える本研究とはアーキテクチャの上で異なる。また、シナリオ記述言語Q[Ishida 02b]は、既存のエージェントに外付けでインタラクション記述を与えるものである。本研究では、より現実に即したインタラクションモデルを設計・検証するためのシミュレータを得るために、既存の社会・環境シミュレータを利活用するという観点から、Connection Controlの実現を含めた適切なインタラクション層の接続手法を考察・提案した点がその意義となる。

以下、本論文では、まず、提案手法について述べる。次に、自然資源管理を目的とした既存の社会・環境シミュレータであるCORMASと上述のQを結合したCORMAS/Qの実装を示す。最後に、森林火災のシミュレーション[Cohen 89]を行うことで、本手法の可能性を議論する。

2. 社会・環境シミュレーションにおけるインタラクション層の構築

2.1 アーキテクチャ

社会・環境シミュレータの外部にインタラクション層を構築したシミュレータのアーキテクチャを図2に示す。このシミュレータは、社会層と環境層の二層より構成される(図中では実線で区切られている)。社会層は、さらに二層よりなる。交渉や協調の過程の記述(インタラクション記述)を解釈し、解釈結果をエージェントに伝

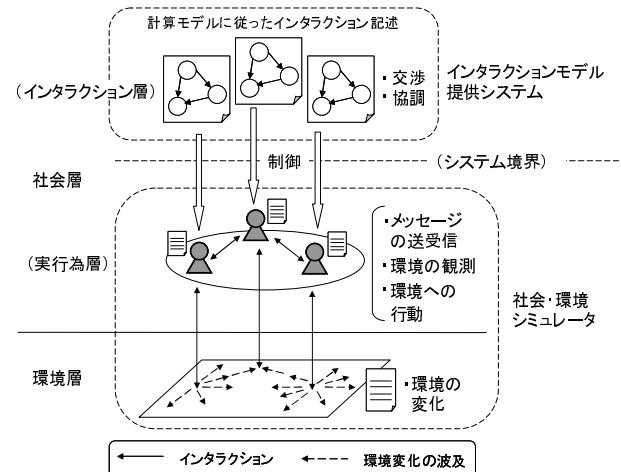


図2 インタラクション層が構築された社会・環境シミュレータ

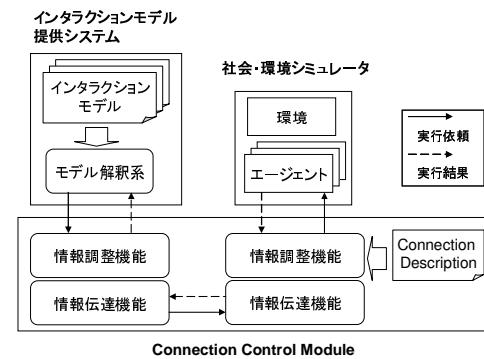


図3 システム構成

える層(Interaction Layer)と解釈結果を受け取ったエージェントにより実際のメッセージの伝達や環境の観察、環境への行動が行われる層(実行行為層)である。即ち、インタラクション層により実行行為層の制御が行われる。前者は、専用に用意されたインタラクションモデル提供システムにより、後者は、既存の社会・環境シミュレータにより記述が提供される。環境層の記述は社会・環境シミュレータにより提供され、エージェントや自然の作用による環境内での変動の伝播が記述される。

図2には、インタラクションモデルとエージェント間に対応付けが示されている。尚、エージェントの協調を表すインタラクションモデルは、複数のエージェントが関わるやり取りの手順として記述される場合もあるが、エージェントが外部からインタラクションを受け取り動作する本アーキテクチャでは、最終的にそれをエージェント一体の視点で分解した記述が割当てられなければならない。

2.2 システム構成

インタラクション記述は、その計算モデルに従って解釈されるモデル解釈系を必要とする。モデル解釈系は、インタラクション記述を解釈し、それをインタラクショ

ンとそれに対応する処理を付加する^{*1}。エージェントに渡る解釈結果が複雑なほど社会・環境シミュレータ内のエージェントにその処理の機構を追加しなくてはならず、その拡張性を考えると、解釈結果はインタラクション記述全体を解釈したものではなく、計算モデルに従ってなるべく細分化されたものが望ましい（状態遷移モデルならば、脚注1が一つの例となる）。したがって、解釈系は、実行依頼の出力後、エージェントの実行結果を入力として次の解釈結果を出力するものとなる。

一方、社会・環境シミュレータのエージェントは、実行依頼を受け取り、それに対応する実際の行動をシミュレーション上で行い、実行結果を返信する。そのためには、エージェントは、ユーザにより定義された実行為にあたるプログラムの他に、実行依頼を得て、それに従った実行為を行った結果を返す機能を持つ必要がある。尚、この機能を社会・環境シミュレータ内で実現できることが本アーキテクチャを構築する上で必須となる。

両システム間には、各システムの入出力を仲介する機能（Connection Control Module）が必要となる。これは、情報交換と情報調整の機能から構成される。情報交換機能は、異種システム間の情報のやり取りを実現するもので、ファイル共有やTCP通信など一般にプロセス間通信に用いる手法を利用できる。情報調整機能は、次章に述べるConnection Controlを実現するものである。

2・3 構成システムの動作例

本構成にしたがって、4章にて取り上げるFire Fighterモデルの一部より抜粋した例により、エージェント間インタラクションの実行の流れを示す（図4参照）。本例では、社会・環境シミュレータ上にFire FighterエージェントとFirebossエージェントが存在する。一方、インタラクション層では、各エージェント用に書かれたインタラクションシナリオをモデル解釈系が読み込み、社会・環境シミュレータ上のエージェントに対して指示を送る。

本例は、Fire Fighterが計画ルート上に火災を観測した後から始まる。まず、インタラクション層のモデル解釈系はシナリオを解釈し、Firebossへ修正ルート要求を送信する実行依頼をFire Fighterへ渡す。すると、Fire Fighterは社会・環境シミュレータ上で用意された通信機能を利用して実際にFirebossに要求を送信する。送信終了後、完了通知をインタラクション層に返すと、モデル解釈系はシナリオを読み進め、次に、Firebossから修正ルート受信の観測をするようFire Fighterに実行依頼を渡す。一方、Firebossは、Fire Fighterからの通信の有無を観察するようにインタラクション層から依頼されているので、Fire Fighterからの要求を受信し、修正ルート要求を受信したことを実行結果としてインタラクショ

^{*1} 例えば、状態遷移モデルで、ある状態から複数の状態への遷移の条件がそれぞれ記述されている場合、条件にあたる複数のインタラクションを平行に観測するという解釈がoutputとなる。

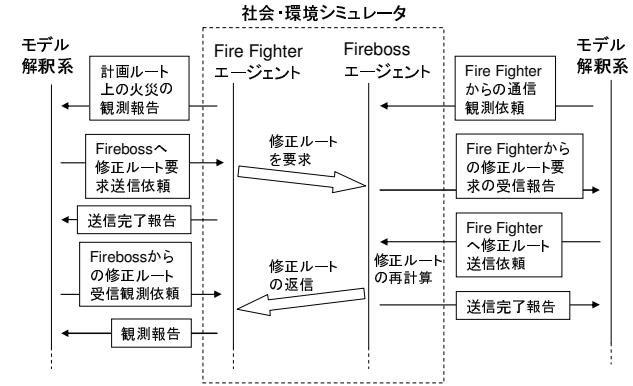


図4 Fire Fighterを例にしたインタラクション実行の流れ

ン層に返す。次に、Firebossはインタラクション層からFire Fighterへ修正ルートを送信するよう実行依頼を受け取る。すると、修正ルートの計算を行い、その結果をFire Fighterに送信する。送信後、送信完了報告をインタラクション層に返す。一方、インタラクション層の依頼によりその返信を待ち続けていたFire Fighterは修正ルート受信後、インタラクション層に受信を報告し、図4上の処理はすべて終了する。

3. Connection Control

3・1 社会・環境シミュレーションにおけるインタラクション

マルチエージェントと環境を想定する社会・環境シミュレーションにおいて、エージェントが関わるインタラクションは一般に以下の3種に分類でき、モデルはこれらの組み合わせにより記述される。1)他のエージェントとのコミュニケーション、2)環境の観察、3)環境への行為。インタラクションが記述される計算モデルに関わらず、インタラクション層と社会・環境シミュレータ間には、3種のインタラクションで構成される情報が流れる。

3・2 Connection Control

より現実に即したインタラクションモデル設計のために、システム間を流れるインタラクションを制御することにより、社会・環境シミュレータの内部を変更することなく、より簡単にシミュレーションを操作する手法（Connection Control）を提案する。これは、個人能力のばらつきや環境から受けける能力の制限をシミュレーションにおいて表現することに相当する。尚、本制御はインタラクション層における記述とは別の専用の記述（Connection Description）により行われるものである。

インタラクションに対し行う操作を2種類定め、インタラクション毎にConnection Descriptionにて以下のパラメータにて指定するようとする^{*2}。

^{*2} この2種類が可能な操作をすべて網羅しているとは言えない

表1 Connection Control の効果と Connection Description の指定例

インタラクションの種類	エージェント能力への効果	パラメータの指定内容	効果の例	例におけるConnection Descriptionでのdelayのタイミング指定
他エージェントとのコミュニケーション	情報伝達能力の微調整を行うことに相当。	accuracy は通信路上でメッセージが欠落する割合, delay は本来よりも情報伝達が余計にかかる時間。	accuracy は雑音などから時々会話が届かない状況, delay は会話内容が完全に伝達できるまでにかかる時間が通常よりも長くなる状況など。	対象となるコミュニケーションがまず開始される必要があり, また, 実行結果を解釈系に返すのを遅らせることで同等の効果を得られるので, 情報発信者/受信者ともに(b)。
環境の観察	環境から情報収集能力の微調整を行うことに相当。	accuracy は環境からの情報収集の成功率, delay は本来よりも情報収集が余計にかかる時間。	常に遠隔地にいる人が離れた対象を把握する場合において, accuracy は情報収集の時々の失敗や, delay は遅延のため常に少し過去の情報しか手に入らない状況など。	その時点での情報をまず取得する必要があり, また, 実行結果を解釈系に返すのを遅らせて同等の効果を得られるので(b)。
環境への行為	環境への行動能力の微調整を行うことに相当。	accuracy は環境への行為が成功する割合, delay は本来よりもその行為の完了に余計にかかる時間。	能力の高い人であるならすぐに行方が達成できるものでも, 人によっては失敗したり, 時間が余計にかかったりする状況など。	環境への行為は環境の状態を変化させるため, 実行前で伝達を止めておく必要があるので(a)。

- (1) 情報伝達の成功率 (accuracy)
- (2) 情報伝達の遅延時間 (delay)

成功率は%で指定できる。例えば, 80%とした場合, 接続部にて5回に1回情報が欠落する。また, 遅延時間は社会・環境シミュレータの時間概念の単位時間に基づいた数にて指定する。例えば, 離散時間であるなら自然数にて指定し, 3と指定された場合, 3単位時間情報の伝達を遅らせる。

accuracy は, 指定されたインタラクションを指定した確率で, エージェントへ伝えないことで実現される。delay に関しては, インタラクションの性質を考慮して望まれる効果を導くように, Connection Description にて処理のタイミングが以下のいずれかで指定される必要がある。

- (a) 実行依頼の伝達を指定時間遅らせる。
- (b) エージェントから解釈系への実行結果の伝達を指定時間遅らせる。

前節で述べた3種のインタラクションそれぞれについて, 本手法の効果と Connection Description の上記タイミングの指定例を表1に示す。

Connection Control が与える能力の制限は, シミュレーション全体に渡るものに限られる。例えば, 環境から受けれる能力の制限については, その環境から常に受けれる障害を表現するものであるので, ある特定の状況における制限(あまりに距離が離れている時にコミュニケーションに障害ができるなど)を表現したければ, 社会・環境シミュレータ内のモデル記述で実現することとなる。また, 本手法はエージェントの実行の前後で情報の制御を行うので, そのために表現力に制限がある。例えば, 火災の鎮火で火力に状態量がある場合, accuracy では鎮火作業を失敗したとしても火力が半分になる状況を, delay では火力を本来の倍の時間かけてゆっくりと状態量を0にする状況を, それぞれ表現することは難しい。

までも, これだけでも様々な効果をエージェントの能力の微調整に与えることができ, 様々な観点からインタラクションモデルを検証することができると言える。

3.3 Connection Control の実現

Connection Control を実現するためには, 実行依頼は, 逐次的に行うインタラクションを含んではいけない。なぜならば, 実行依頼に逐次的に行われるインタラクションが複数含まれていた場合(例: A の後に B を行う), 情報調整機能は, 2番目以降のインタラクションが開始されるタイミングを得られず, ユーザが意図する delay の処理を実現できないからである(したがって, A を依頼し, 完了後 B を依頼するようにする)。つまり, これを図3におけるどちらかの情報調整機能で実現しなくてはならない。

社会・環境シミュレータが離散時間による時間進行を行う場合, delay の処理のために, 社会・環境シミュレータの時間進行を Connection Control Module にて把握する必要がある。このことを考えると, Connection Description は, 社会・環境シミュレータ側の情報調整機能に付属して設置されなくてはならない(図3 参照のこと)。

尚, 本アーキテクチャにおいて, エージェントに対する入力は3・1節における3種のインタラクション, エージェントからの出力はその実行結果で規定され, さらに本節の要件を実現する限り, 3・2節で述べた効果の範囲はあるものの, Connection Control は社会・環境シミュレーションに一般に適用されるものである。

4. Q と CORMAS の接続

実際に社会・環境シミュレータであるCORMASに対して, インタラクション記述とその解釈系を持つQを用いて, 前章で述べた手法に従って, インタラクション層を構築する。本章では, Q と CORMAS の特徴について述べた後, 接続の実装を論じる。

4.1 Q と CORMAS

§ 1 Q

シナリオ記述言語 *Q*^{*3}[Ishida 02b] は、複数の（レガシーな）エージェント間のインタラクションシナリオを記述するための言語である（既存の）エージェントシステムに外付けで、エージェントの実行制御を行うことを目的としている。

シナリオ記述の背後にある計算モデルは拡張状態遷移機械である。拡張状態遷移機械はコミュニケーションプロトコルの記述に一般的に用いられている計算モデルであり、*Q* はそれを言語化したものと言える。

Q が提供するシミュレーション記述の特徴は、ユーザが各エージェントの持つセンシング機能（キュー）と実行機能（アクション）を利用してシナリオを記述できることである。キューとアクションは、*Q* 处理系によってではなく、既存のエージェントシステムによって実行される。*Q* を用いたシミュレーションシステムは、*Q* 言語処理系、*Q* コネクタ（レガシーな）エージェントシステムにより成る。*Q* コネクタは、*Q* 言語処理系の処理結果の出力形式を定義するものである。

シナリオ解釈はイベント駆動モデルにより行われる。インタラクションにより生じるイベントを契機として、対応する処理を行う。時間の管理はキューとアクションを実行するエージェントシステムに依存する。

Q 言語は、複雑な社会的インタラクションを記述するのに実績がある [Ishida 02a, Murakami 03]

§ 2 CORMAS

CORMAS^{*4} (COmmon pool Resources and Multi-Agents Systems)[Bousquet 98] は、自然資源を利用する個人あるいは集団間の協調過程のシミュレーションを目的とするシステムである。自然環境の変動と、自然環境を観察/実行するエージェントを定義できる。

シミュレーションの背後にある計算モデルは、セルラオートマトンである。空間を 2 次元のメッシュで表し、各セルの単位時間（ステップと呼ぶ）当たりの変化を記述でき、これを元にセル間の変動の伝播を計算できる。

シミュレーション記述の特徴は、ステップ毎に各セルやエージェントが実行すべき機能を記述することで、簡単にシミュレーションが実現できることである。モデリングツール（空間、エージェント、エージェント間コミュニケーション）、管理ツール（シミュレーション）、可視化ツール（エージェント間コミュニケーション、統計情報）が提供されている。また、GIS（Geographic Information Systems）からの空間マップを利用することもできる。

CORMAS は、農業分野において自然資源管理問題のシミュレーションに実績がある [Mathevet 03, Perez 02]。

*3 http://www.lab7.kuis.kyoto-u.ac.jp/Q/index_j.htm

*4 <http://cormas.cirad.fr/indexeng.htm>

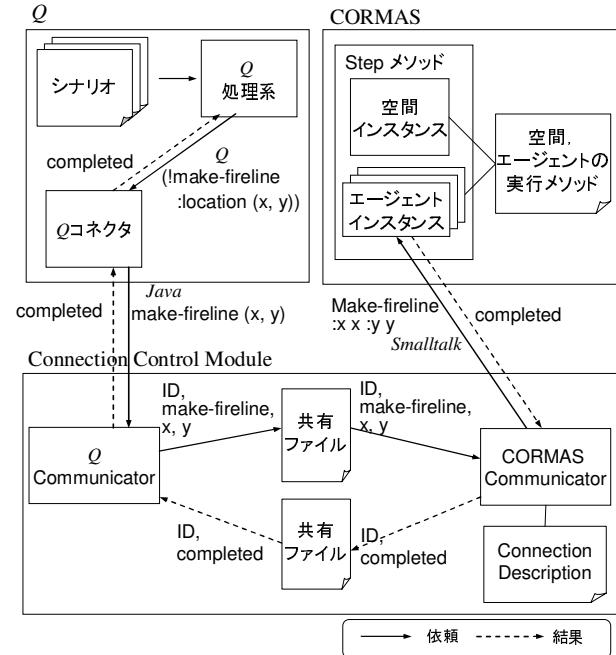


図 5 CORMAS/Q の実装

4.2 接続の実装

まず、各エージェントについて *Q* のシナリオを用意する。*Q* では、3.1 節に挙げた 3 種のインタラクションをキューとアクションという 2 つの機能で実現できる。即ち、他のエージェントとのコミュニケーションは、他のエージェントからのメッセージの受信（キュー）、他のエージェントへのメッセージの送信（アクション）に分けて記述される。また、環境の観察はキューで、環境への行為はアクションにより記述される。*Q* はシナリオの実行をするためのものであり、実際にキューとアクションを実行するのは CORMAS の対応するエージェントである。従って、*Q* のシナリオ中に現れるキューとアクションの実体は、CORMAS 側で定義されていなければならない。

図 3 における情報交換機能の実装には共有ファイルを用いた。共有ファイル方式は、TCP/IP や CORBA による接続と同様に、複数のシステムを接続する一般的な方法の一つである。今回、*Q* 言語処理系は Java 仮想機械上で^{*5}、CORMAS は SmallTalk 仮想機械上で動作しており、共に多様なプラットフォーム上で実行が可能である。この特徴を生かすためには、接続したシステムも OS 依存しない接続方式を選択することが望ましい。

共有ファイルを扱うため、*Q* 言語処理系側に *Q Communicator* を、CORMAS 側に *CORMAS Communicator* を用意した。*Q Communicator* は Java で、*CORMAS Communicator* は SmallTalk で実装されている。共有ファイルに書き込むフォーマットはあらかじめ決められており、*Q Communicator* は *Q* 言語処理系からのキューやアクションの実行依頼を共有ファイルに書き込み、COR-

*5 今回、*Q* は Kawa 版を使用した。

MAS側にQ側の実行要求を伝える役割と、実行結果を受け取ってQ言語処理系へ返す役割がある。CORMAS Communicatorは、エージェントから適宜呼び出されると、キューやアクションの実行依頼を受け取りエージェントへそれを渡す役割と、エージェントにより対応するCORMASメソッドを呼び出され実行された結果を共有ファイルに書き込む役割がある。

また、情報調整機能は次のように実現する。Q言語処理系は、3・3節で述べた条件を満たすインタラクションの集合を出力する^{*6}ので、特別な加工は必要ない。Connection Controlを実現するために、CORMAS CommunicatorにConnection Descriptionを与えた。情報の制御は、CORMAS Communicatorがエージェントプログラムを呼び出す際に、Connection Descriptionを参照して行われる。CORMASは離散時間モデルによりシミュレーション時刻を進行するため、遅延時間(delay)の実現のためには、CORMAS上での時間の進行をCORMAS Communicatorが把握する必要がある。

尚、本接続は、Qのこれまでの接続例であるFreeWalk[Nakanishi 04]やMSAgent[Ishida 02b]と比較して、情報調整機能部を構築してConnection Controlを実現したという点で新たな実装例を示している。

図5に、アクション'make-fireline'が実行された場合を例にしてこの様子を示す。

5. Fire Fighter モデル

5・1 問題

本研究での提案手法の可能性を議論するために、例題として、QとCORMASにより構築したシミュレータにより、Cohenらによって行われた火災シミュレーション[Cohen 89]を実装した。その目的は、様々な状況を想定した複数のシミュレーションを行うことで、適切なエージェント間のインタラクションを設計することである。

この問題では、火災の広がりを防ぐため、全体の状況を見ながら指示を出すFirebossと、その指示に従いながら公園内で行動するFire Fighterが存在する。Fire Fighterはブルドーザーであり、Firebossの指示に従いながら、火災を囲むように木を取り除いた道(fireline)を作ることにより、火災の広がりを防ごうとする。環境は、セルごとに川、道路、木、火などの状態を持っている。火災は、風の速度と向き、湿度、地形の特性などに従って広がっていく。仮に風が強ければ、川やfirelineなどの環境的境界を飛び越えて火が燃え広がっていく。

シミュレーションは公園内で火災が発生し、2体のFire Fighterが火災の位置と風向きなどから決定された集合ポイントに配置されるところから始まる。Firebossは、その時点での火災の状態から2体のFire Fighterが取

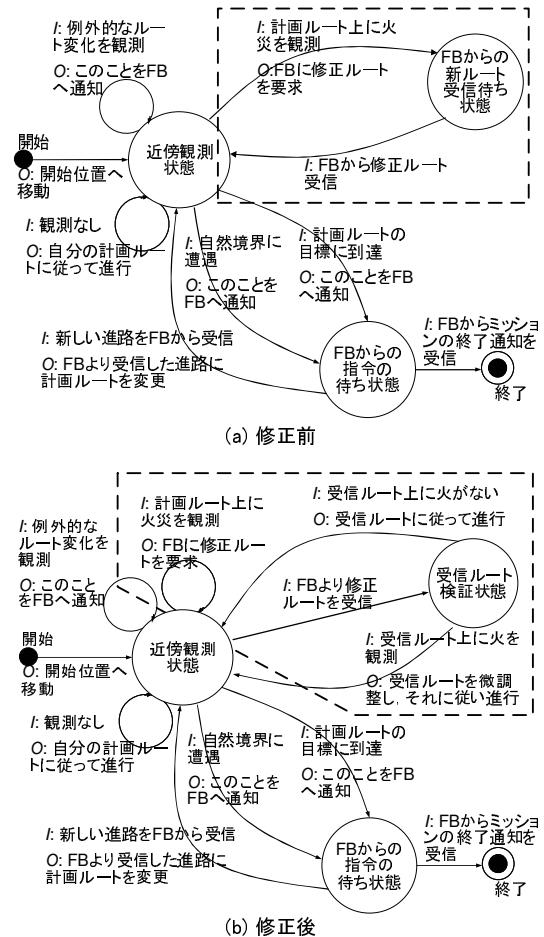


図6 Fire Fighter の修正前後のインタラクションモデル(状態遷移機械)

るべきルートを決定し指示を出す。各Fire Fighterは、指示されたルートを従って時計回りと半時計回りに動き、木を取り除いていく。Fire Fighterは、指示されたルート上に火災が広がっていたり、急激な環境の変化により、Fire Fighterに突如として火災が及んだ場合は、自分の判断やFirebossと連絡を取りながらルートの変更を行う。Firebossは、Fire Fighterからの要求があれば、対応した指示をFire Fighterに返信する。例えば、Fire Fighterから進行ルート上に火が広がっていたために修正ルートの要求があれば、エージェントの位置と環境全体の情報から作成したルートプランをFire Fighterに返信する。またFire Fighterは、川などの境界や他のエージェントによって作られたfirelineにぶつかった場合、次の行動をFirebossに問い合わせることもできる。Firebossはfirelineが完成されていると判断すれば、作業の終了を指示する。仮に、火災が他の場所に飛び火などして広がっていれば、別のルートを指示する。

このように、FirebossとFire Fighterのインタラクションによりシミュレーションが進行し、最終的に火災の進行が食い止められる。

^{*6} 正確には、条件を満たすようQコネクタで設定可能ということである。

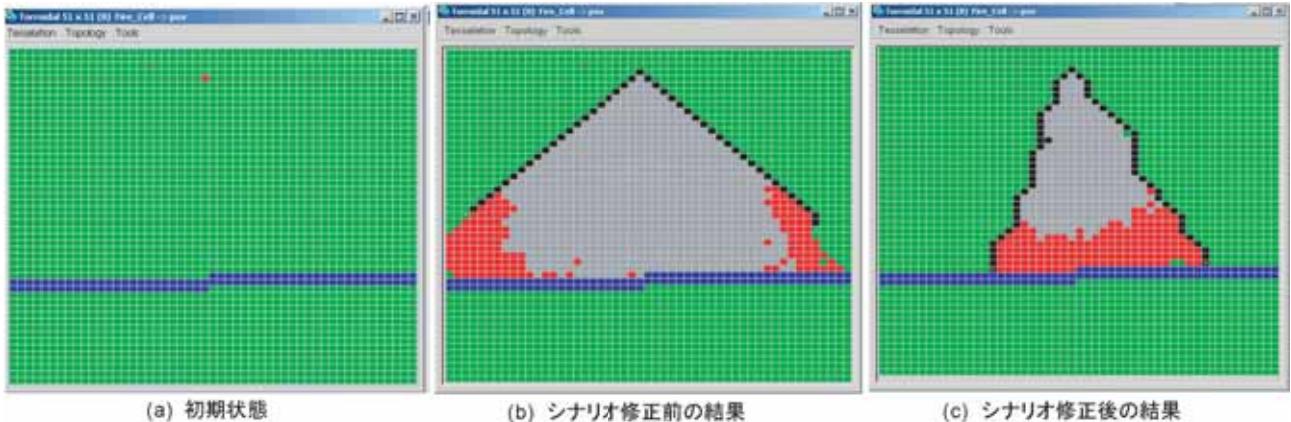


図 7 Fire Fighter シミュレーションのスクリーンショット: 森林は緑、川は水色、炎は赤、fireline は黒、エージェントは三角形で示されている

5・2 モ デ ル

CORMAS 上には、各セルに森林、空き地、川などの地形情報と風の向きと速度の天候情報を含む環境のモデルを構築し、火災が時刻と共に広がるシミュレーションが実現された。火災の伝播モデルはセルラオートマトンにより与えられる。一方、Fireboss と Fire Fighter に対応するエージェントを CORMAS に定義した。 Q シナリオでは、エージェント間の社会的インタラクションや、Fire Fighter と環境とのインタラクションのモデルが記述されている。また、CORMAS 側では、 Q 側からのインタラクションの依頼を受けて、他のエージェントとのコミュニケーション、環境変化の観測、セルの状態変化や移動などの行動を行う。

すでに述べた Fireboss (FB) と Fire Fighter (FF) のインタラクションは、状態遷移モデルで表現することができる（図 6 に Fire Fighter のモデルを示す）。シミュレーションでは、 Q 言語処理系は状態遷移図を直接的に処理できないので、状態遷移モデルが素直にコーディングされた Q シナリオが用いられる。

5・3 結 果

図 7 にシミュレーションのスクリーンショットを示す。空間には森林が広がり、一本の川が流れている。初期値として空間の上部中央に小さな火災を発生させ、北風により火災を拡大させていく。Fire Fighter の初期位置は、火災発生源の北側とした（図 7(a) 参照）。

ここでは、Connection Control を利用することで、インタラクションモデルの修正過程を示す（図 6 (a) から (b) へ。点線の囲みが修正箇所）。

まず、Fireboss から Fire Fighter への情報伝達が自然環境からの雑音により制限される状況を Connection Description により表現した。これは、Fireboss から Fire Fighter へのすべての通信に対して、情報伝達の遅延時間 (delay) を 3 (ステップ) にすることを行った。遅延は、Fireboss, Fire Fighter 共に、3・2 節における (b) のタイミングで行うよう指定した。すると、Fire Fighter の

進行が遅々として進まず、環境の画面下部が火でほとんど覆われてしまうこととなった（図 7(b)）。時間の進行と共に Fire Fighter の動きを中心に原因を追跡した結果、通信遅延のため、Fireboss からの修正ルートを完全に把握する頃には、火災が進行して修正ルート上に火災が及んでおり、再度、Fireboss へ修正ルートを要求しなければならないことが頻繁に起こっていることが分かった。

次に、Fireboss から受信した修正ルート上に火が及んでしまっている場合は、Fire Fighter が自分の周りの情報だけ進路をさらに修正できるようモデルを改良した（図 6(b)）。これにより、受信ルート上に火災が及んでいても自分でルートを微調整することにより、Fireboss との通信回数を抑えることができた。結果として、遅延がない時ほど火災の進行を抑えることができないものの、fireline と環境の境界（川）により火災を囲む状態を環境内に作ることができた（図 7(c)）。

5・4 議 論

本例題により、提案手法におけるアーキテクチャとシステム構成により構築した具体的な社会・環境シミュレータが問題なく動作したことを見せたと考える。

また、本例題である Fire Fighter のような環境と人が複雑に絡むドメインでは、複雑なインタラクションモデルが必要となり、提案手法が有効であると考える。 Q では、計算モデルとして状態遷移機械を用いることで、同時に複数の観測と観測に対応した行動を記述できるので、環境や司令部からの観測が同時に必要な本ドメインに適していると言える。 Q シナリオが与える記述性や可読性は、CORMAS 単独で達成するには労力を要する。なぜならば、状態遷移機械モデルに従った様々なシナリオを汎用的に処理するためには、CORMAS 上で Q 言語処理系と同様の機構を既存シミュレータの枠組みの中で工夫して組み込む必要があるからである。

さらに、Connection Control の使用例を示すことで、Connection Control という手法が、より現実に即したインタラクションモデルを設計する上で有効であることも

示せたと考える。本手法は、提案した接続手法により初めて実現されるものである。従来では、既存の社会・環境シミュレータの内部を変更しなくてはならず、本手法を適用することでエージェントの能力や環境からの制限を勘案したシミュレーションが容易に実現できる。注意しなくてはならないのは、本手法の効果への解釈をドメインの専門家と議論するなどして定義する必要があるということである。このことを考慮しつつ、その他のドメインについても適用することで、引き続き本手法の評価・精錬が必要となる。

6. おわりに

本研究は、社会・環境シミュレーションにおいて、より現実に即したインタラクションモデルの検証・設計を行うことができるシミュレータを、既存の社会・環境シミュレータを利用して構築する手法を提案した。本手法では、まず、ユーザがインタラクション設計に集中できるよう既存の社会・環境シミュレータの外部にインタラクション層を構築した。さらに、接続されたシステム間を流れる情報を制御することにより、エージェント個人の能力のばらつきや環境から受ける能力の制限を表現(Connection Control)することで、より現実に即したインタラクションモデルを検証・設計できるものとなっている。また、実際に、インタラクションモデルを拡張状態遷移機械モデルで提供する *Q* 言語処理系と社会・環境シミュレータである CORMAS を接続することで、具体的な実装法を示した。最後に、Cohen らによる Fire Fighter モデルを CORMAS/*Q* により実施することで、Connection Control によるインタラクションモデルの設計プロセスを示し、提案手法の可能性を議論した。

我々は今後、本論文で示した手法を実問題への適用すると共に、次のように CORMAS/*Q* の適用を検討している。*Q* は、シナリオ制御のエージェントを容易に人間が制御するアバターに置き換え可能なので、参加型シミュレーション（人間とソフトウェアエージェントが協力しながらシミュレーションを進める）に適している。また、*Q* はシナリオの記述に拡張状態遷移機械を用いているため、参加型シミュレーションの結果から、機械学習を用いてシナリオの改善を図ることも可能である。

謝辞

御助言を頂いたパリ第六大学の Alexis Drogoul 教授、本実装へ協力頂いた Stéphane Bonneaud 氏に感謝の意を表す。本研究は、日本学術振興会科学研究費 基盤研究(A)(15200012, 2003-2005)の助成金により行われた。

◇ 参考文献 ◇

- [Bousquet 98] Bousquet, F., Bakam, I., Proton, H. and Le Page, C. Cormas: common-pool resources and multi-agent

Systems. *Lecture Notes in Artificial Intelligence*, Vol. 1416, pp. 826-838, 1998.

[Cohen 89] Cohen, P. R., Greenberg M. L., Hart, D. M. and Howe, A. E. Trial by fire: Understanding the design requirements for agents in complex environments. *AI Magazine*, Vol. 10, No. 3, pp. 32-48, Fall 1989.

[Ishida 02a] Ishida, T. Digital City Kyoto: Social Information Infrastructure for Everyday Life. *Communications of the ACM (CACM)*, Vol. 45, No. 7, 76-81, 2002.

[Ishida 02b] Ishida, T. *Q*: A Scenario Description Language for Interactive Agents. *IEEE Computer*, Vol. 35, No. 11, pp. 54-59, 2002.

[Kita 03] Kita, H., Sato, H., Mori, N. and Ono, I. U-Mart System, Software for Open Experiments of Artificial Market. In *Proceedings of IEEE SIRA2003*, 1328-1333, 2003.

[Mathevet 03] Mathevet, R., Bousquet, F., Le Page, C. and Antona, M. Agent-based simulations of interactions between duck population, farming decisions and leasing of hunting rights in the camargue (southern france). *Ecological modelling*, Vol. 165, pp. 107-126.

[Murakami 03] Murakami, Y., Ishida, T., Kawasoe, T. and Hishiyama, R. Scenario Description for Multi-Agent Simulation. In *Proceedings of International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-03)*, pp. 369-376, 2003.

[Nakanishi 04] Nakanishi, H. and Ishida, T. FreeWalk/*Q*: Social Interaction Platform in Virtual Space. *ACM Symposium on Virtual Reality Software and Technology (VRST2004)*, pp. 97-104, 2004.

[Perez 02] Perez, P., Ardlie, N., Kuneepong, P., Dietrich, C., Merritt, W. S. CATCHCROP: modeling crop yield and water demand for integrated catchment assessment in Northern Thailand. *Environmental Modelling and Software*, Vol. 17, No. 3, pp. 251-259, 2002.

[Takahashi 02] Takahashi, T., Tadokoro, S., Ohta, M. and Ito, N. Agent Based Approach in Disaster Rescue Simulation, *RoboCup 2001: Robot Soccer World Cup V*, Lecture Notes in Artificial Intelligence, pp.102-111, 2002.

〔担当委員：栗原 聰〕

2005年6月3日 受理

著者紹介

鳥居 大祐(学生会員)

2001 年京都大学工学部情報学科卒業。2003 年同大学院社会情報学専攻修士課程修了。現在、同大学院社会情報学専攻博士課程に在学中。マルチエージェントシミュレーション、P2P システムに興味を持つ。

石田 亨(正会員)

1976 年京都大学工学部情報工学科卒業、1978 年同大学院修士課程修了。同年日本電信電話公社電気通信研究所入所。ミュンヘン工科大学、パリ第六大学、メリーランド大学客員教授など経験。工学博士。IEEE フェロー。情報処理学会フェロー。現在、京都大学大学院情報学研究科社会情報学専攻教授、上海交通大学客員教授。自律エージェントとマルチエージェントシステム、セマンティック Web 技術に取り組む。デジタルシティ、異文化コラボレーションプロジェクトを推進。

Disaster Evacuation Guide: Using a Massively Multiagent Server and GPS Mobile Phones

Yuu Nakajima, Hironori Shiina, Shohei Yamane, Toru Ishida

Kyoto University, Department of Social Informatics

{nkjm, shiina, yamane}@ai.soc.i.kyoto-u.ac.jp

ishida@i.kyoto-u.ac.jp

Hirofumi Yamaki

Nagoya University, Information Technology Center

yamaki@itc.nagoya-u.ac.jp

Abstract

The ubiquitous environment enables us to build systems that provide individual users with personalized navigation services in cities. In developing such a system, it is necessary to estimate the influence and the movement of its users and to get feedback. However, it is difficult to perform tests on such a system given the large number of human subjects involved and its scale, which matches that of a major city. One possible solution is proposed herein, the augmented experiment; it combines a multiagent simulation with a small-scale experiment performed with human subjects. In the experiment, the movements of agents that simulate users are shown to human subjects in order to give them the impression that the environment is populated with a large number of users. In this research, we build a evacuation guide system based on GPS-capable cellular phones and perform an augmented experiment wherein human subjects and evacuee agents are directed while the status of the simulation is passed to the human subjects. Interviews of the human subjects confirmed that the augmented experiment successfully gave the impression that a large number of users were present.

1. Introduction

Due to the popularity of and improvement in mobile phones, environments that support ubiquitous computing are spreading. In traditional information services, the user accesses the services desired via a terminal fixed in a room. However, in the ubiquitous environment, each user has his/her own portable device and accesses the services desired from any location via the wireless network. Because each person has his/her own

device such as a mobile phone, it is possible to show different information to each user. In addition, GPS (Global Positioning System) and RFID (Radio Frequency Identification) tags enable devices to get information of the location and the situation of the user. In such an environment, it is possible to provide services that suit the properties, the purpose, the location and the context of each user. Navigation in public spaces is one such service [7][9].

Current navigation services simply broadcast the same instructions over a large area, but what is needed is a system that can provide individualized instructions to each person. The use of GPS-capable cellular phones enables the provision of personalized navigation instructions to suit the user's characteristics, city-supplied evacuation targets, and the surrounding environment.

To know the behavior of such social systems, it is desirable to perform confirmation experiments with a large number of human subjects. Unfortunately, cost and safety concerns preclude such experiments from being performed on a realistic scale. The solution is the augmented experiment approach; multiagent simulations are combined with small-scale experiments in the real world [5][6].

Any large-scale evacuation guide system must be verified from the macro standpoint and the micro standpoint. The former is concerned with whether all citizens can take refuge efficiently. The latter addresses how citizens use the system. Verification will be accurate only when the human subjects feel as if their environment is populated with an adequate number of participants. Our goal is to devise a process for verifying city-wide navigation services. In this research, we confirm that the combination of multiagent simula-

tion and GPS-capable cellular phones can create a situation in which human subjects feel like they are participating with a large number of humans.

Section 2 explains augmented experiments. In Section 3, we describe the scenario description language and the massive multi-agent platform as the fundamental technology. Section 4 introduces our large-scale evacuation guide system for augmented experiments. In Section 5, we explain the setting of an initial experiment and its results.

2. Augmented Experiment

A city-wide evacuation guide system is a large-scale social information service where a large number of users interact with each other. Testing the system requires analyzing the movement of crowds and the usability of the system. These analyses need human subjects who feel as if they are participating together with a large number of humans. Unfortunately, real-world experiments with many people are too expensive and rather dangerous.

The Augmented Experiment was proposed to test large-scale ubiquitous computing systems [6]. A real-world experiment with a small number of human subjects is enhanced by a large-scale multiagent simulation.

Figure 1 illustrates how an augmented experiment for a city-wide navigation service is realized. The augmented experiment system lets human subjects perceive the results of the simulation performed in the virtual space concurrently with the real-world experiment. The human subjects are made to feel that they are participating with many other humans. This approach yields data comparable to that by performing real-world tests with a great many human subjects. In addition, comparing the movements of the human subjects with those of the agents can enable us to refine the user models of the agents.

The augmented experiment for the navigation service proceeds as follows. The real-world experiment and the simulation in a virtual space are executed concurrently. A user agent in the simulation sends its location to a guide agent, who gives navigation instructions to the agent. The user agent determines his behavior considering the instructions and the agent's situation in the virtual space. The human subject, on the other hand, sends his location to the navigation service and the subject's position is projected into the virtual space. The guide agent for the subject sends navigation instructions and the status of the virtual space including positions of other evacuees to the subject.

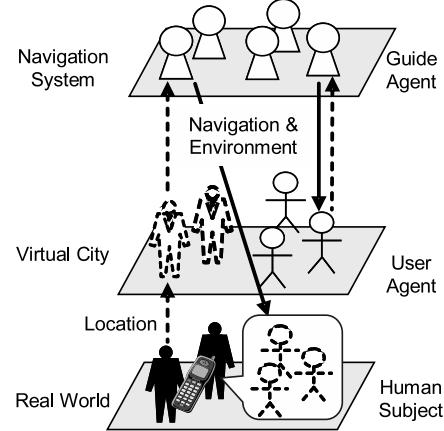


Figure 1. Example of Augmented Experiment for Navigation Service

Participatory simulations are another method that allows the actions of human subjects in a virtual space to be extracted [3]. In a participatory simulation, some agents are replaced by human-controlled avatars. A participatory simulation is performed in virtual space, and the avatars are controlled by human subjects sitting in front of their computers. Participatory simulations are particularly useful, but sometimes fail to give valid results. To understand how users really respond to socially embedded systems, real world experiments are often required.

3. Fundamental Technologies

We have combined the scenario description language *Q* [4] with a large-scale agent server Caribbean [12] to build a platform for large-scale multiagent simulations. Below, we describe the two technologies and the massively multiagent platform that we developed by combining them.

3.1. Scenario Description Language *Q*

*Q*¹ is an interaction design language that describes how an agent should behave and interact with its environment including humans and other agents. For details see [4]. In modeling human actions, it has been shown that the *Q* approach, describing the interaction protocol by scenarios, is more effective than alternative agent description methods that simply describe the appearance of human beings [10].

¹ *Q* is available from http://www.ai.soc.i.kyoto-u.ac.jp/Q/index_e.htm

The features of the *Q* language are summarized as follows.

- **Cues and Actions**

An event that triggers an interaction is called a cue. Cues are used to request agents to observe their environment. A cue has no impact on the external world. Cues keep waiting for the event specified until the observation is completed successfully. Actions, on the other hand, are used to request agents to change their environment. Cue descriptions begin with “?” while action descriptions begin with “!”.

- **Scenarios**

Guarded commands are introduced for the case wherein we need to observe multiple cues in parallel. A guarded command combines cues and actions. After one of the cues becomes true, the corresponding action is performed. A scenario is used for describing state transitions, where each state is defined as a guarded command.

- **Agents and Avatars**

Agents, avatars and a crowd of agents can be defined. An agent is defined by a scenario that specifies what the agent is to do. Avatars are controlled by humans so they do not need any scenario. However, avatars can have scenarios if it is necessary to constrain their behavior.

In addition, a tool called Interaction Pattern Card (IPC) is introduced into *Q* to support scenario descriptions. Even computer novices can easily describe scenarios by using this tool.

3.2. Agent Server Caribbean

Caribbean² is a large-scale agent server implemented in Java. Caribbean manages agents as objects. There are two types of objects in Caribbean: service objects and event driven objects. Objects in Caribbean communicate with each other using the Caribbean messaging facility. Service objects can be run at any time and are used for implementing modules such as databases with common information which are accessed frequently. In contrast, event driven objects run only when they receive messages from other objects. The Caribbean scheduler allocates threads to event driven objects based on messages. Most modules in a Caribbean system are implemented as this type of object.

² Caribbean is available form <http://www.alphaworks.ibm.com/tech/caribbean>

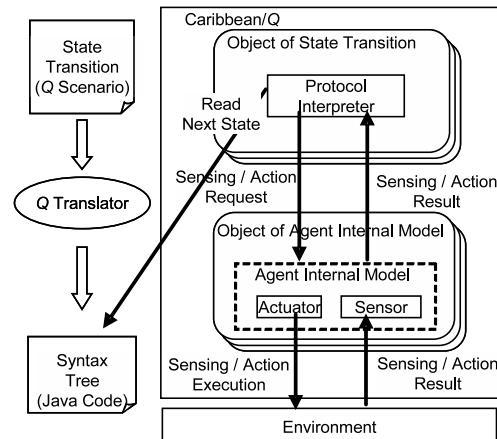


Figure 2. Overview of Massively Multiagent Platform Caribbean/*Q*

Using the naive method of assigning one thread to each object and running them concurrently limits the system to only several hundred objects. Caribbean does not run all objects concurrently, and assigns threads only to those event driven objects that have to work. This makes it possible to run a much larger number of objects.

Caribbean limits the number of the objects that can call on memory and controls the consumption of memory by shuttling objects between memory and external storage. When the number of objects that call on memory exceeds a limit, Caribbean moves inactive objects to external storage. When another object sends a message to an object in external storage, the object is restored to memory and the message is processed. This swapping means that Caribbean can manage many more agents than would fit directly into memory.

3.3. Massively Multiagent Platform Caribbean/*Q*

We developed a large-scale multiagent Caribbean/*Q* system, by combining scenario description language *Q* and large-scale agent server Caribbean. A key point is that it realizes the separation of protocol design and agent development and scalability. Figure 2 outlines the system.

A *Q* scenario describes an interaction protocol between an agent and the outer world. A protocol is given as a state transition description. The conventional processor of *Q* language, which is implemented in Scheme, cannot control enough agents to realize the massive

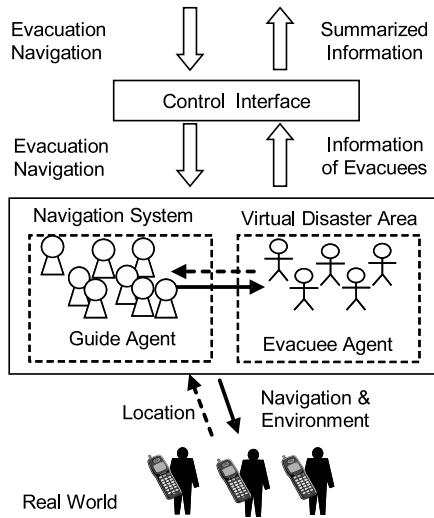


Figure 3. Overview of Evacuee in Augmented Experiment for Large-Scale Evacuation Guide System

scale needed. Therefore, we developed a new processor of *Q* language that runs on the agent server Caribbean.

In Caribbean/*Q*, the *Q* translator takes a *Q* scenario as input, and converts it into a syntax tree that is read by the state machine object in Caribbean. The state machine executes the converted syntax tree stepwise, by which the protocol given in *Q* is executed. The scalability of Caribbean is thus fully exploited by importing the *Q* processing system as an event driven object in Caribbean.

MACE3J [1], MadKit [2], Robocup Rescue [8] are other large-scale multiagent platforms. These platforms provide no explicit separation between protocol description and agent development. In contrast, the Caribbean/*Q* architecture is aimed at separating protocol design from agent development, which enables the experts of different domains to cooperatively and efficiently develop large-scale multiagent systems.

4. Large-Scale Evacuation Guide System for Augmented Experiment

We produced a large-scale evacuation guide system for executing as a good test of the augmented experiment approach. Figure 3 depicts the structure of the system.

The system commander assigns an evacuation destination and evacuation direction through the control interface shown in Figure 4. The commander issues high level instructions to the guide

agents using a map and the guide agents, assigned to the evacuees on a one-to-one basis, provide individual navigation instructions. Guide agents provide information. These instructions are sent to the evacuees via GPS-capable cellular phones.

Guide agents navigate corresponding users in the real world. Each guide agent navigates its user with consideration of the user's surrounding environment. Guide agents provide the movement. The positions and movements of other evacuees with drawing it are shown on the navigation map. In this way, human subjects can feel as if they are participating with a large number of humans.

4.1. Control Interface

Transcendent communication is proposed as the method for navigation in public spaces [11]. We implemented the control interface based on this architecture. In transcendent communication, the distribution of evacuees in the real space is reproduced on the virtual space as human figures that mirror the positions of evacuees; the positions of the subjects are acquired by sensors.

By running multiagent simulations, the commander can develop and optimize evacuation instructions as if he were interacting with a large number of human subjects. The commander views the virtual space via transcendent communication and treats real evacuees and virtual evacuees in the same way.

The current state of the virtual space is displayed as a birds-eye view on the monitor of the control center, so that the commander can grasp how evacuees are moving in the real world. In addition, the commander can instruct individual evacuees by clicking on the human figures on the screen. The system passes the instructions to the appropriate evacuees via their registered phone numbers or e-mail addresses. The use of transcendent communication makes it possible to grasp the situations of all evacuees while providing local navigation instructions as needed.

In a large scale disaster, the commander should not be trying to provide detailed instructions to each evacuee. The solution is for the commander to group evacuees in certain areas and send rough instructions to them as shown in Figure 4. However, such rough instructions can't provide enough information to the evacuees. Therefore, the guide agents acquire the rough instructions from the commander and send personalized navigation to their users by interpreting it as shown in Figure 5.

The commander tells the guide agents the direction to be taken to evacuate through the control interface.

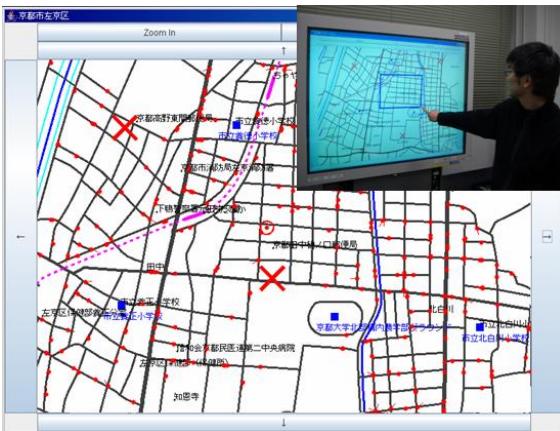


Figure 4. Transcendence Interface for Commander

The interface provides a complete map of the disaster area so that the commander can discern the current locations of almost all evacuees. The commander can also assign evacuation sites, set places of shelters, and record the information about dangers such as fires.

This system places geographical information of the disaster area into the virtual space by accessing a database holding numerical maps (1/25000) issued by the Geographical Survey Institute. Evacuation guides and disaster situations that are entered through the control interface are recorded in this database at regular intervals.

4.2. Massively Multiagent System

Agents on Caribbean/*Q* act as event-driven objects. Caribbean/*Q* can operate both guide agents and simulation agents simultaneously. Both guide agents and simulation agents are implemented as extensions of the event-driven object of Caribbean and agent behaviors are controlled via *Q* scenarios. They get information of dangerous sites and shelters from the same environment database.

Guide Agents:

Guide agents provide individual navigation instructions to each evacuee using information of the evacuee's position, navigation targets set by the commander, and the environmental situation.

A guide agent sends a surrounding map centered on the user's location upon receiving location information from the user's GPS-capable cellular phone. The map shows locations of dangerous sites such as fires, shelters to evacuate to, and the direction of safety are de-



Figure 5. Evacuee in Augmented Experiment and GPS-capable Cellular Phone

scribed. The user can send his location and get a new map whenever he wants to.

An agent is instructed on the evacuation direction by the control center. The agent retrieves shelters around the user, and selects a destination according to the ordered direction and distance between the user and each shelter. If the destination is changed, the agent notifies the user. If there is a person who needs help, his location is given to neighboring evacuees.

Evacuee Agent:

The positions and movements of other evacuees are provided to the human subjects through their navigation maps. The positions of the human subjects are captured by GPS and plotted in the virtual space. Evacuee agents determine their behavior after consideration of this data.

Evacuee agents behave as navigated users and act in the virtual space according to their scenario. An evacuee agent requests its guide agent to send a navigation map and uses the map in determining its movements. The scenario of an evacuation agent is shown in Figure 6.

Human users move in the real space following the maps sent by their guide agents and naturally cannot break the laws of physics. The agents, on the other hand, must be provided with geographic data to prevent them from ignoring reality. In the current system, simulation agents can move only along "roads". Road data is prepared for them in the virtual space. This data is composed of intersections (nodes) and segments (edges) connecting intersections. Taking account of shelters, dangerous sites, and the given direction, a simulation agent evacuates along the fixed roads on the virtual space.

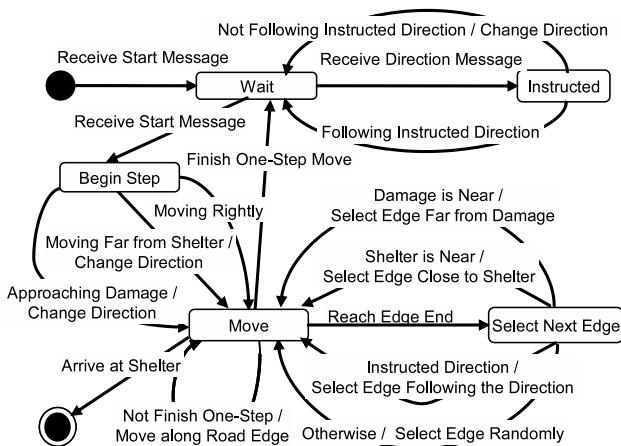


Figure 6. Scenario of Evacuation Agents

One cycle of an evacuee agent is as follows. First, the evacuee agent decides his goal. The agent checks if the current heading direction has become incorrect due to the emergence of dangerous sites or the disappearance of shelters. If the heading direction should be changed, the agent turns to new direction. Next, the agent starts to move along a road segment. If it reaches an intersection, it selects the next intersection and moves along the segment to the intersection. When it selects the next intersection, it obeys rules such as not to approach dangerous sites, to select an intersection close to a shelter, to follow the direction set by the control center.

In this experiment, all guide agents are controlled with the same scenario. However, by creating various scenarios, it is possible to develop a system that navigates users in various scenes or to conduct simulations with different properties and goals.

5. Augmented Experiment: Evacuation Guide

We executed an augmented experiment for the evacuation guide system described in Section 4. This experiment was conducted in order to test how human subjects responded to the simulated crowd.

5.1. Setting of Augmented Experiment

The experiment was conducted around Kyoto University. The area had a side length of 4 km. About 10 human subjects (13 in the first phase, 10 in the second phase) and 3,000 simulation agents were evacuated together. 5 shelters were dispersed throughout the area in advance.

The system was accessed via web browsers of current mobile phones. Each mobile phone determined its location with GPS and sent it to the system every minute automatically. In addition, human subjects could send their location manually whenever they needed a new map. Upon receiving location information, a guide agent would send a map that showed dangerous sites, shelters, and the direction to be followed its user.

One person played the role of the commander. He viewed the virtual space displayed on the monitor in the control room, and instructed both guide agents and simulation agents on the directions to be taken.

Although the human subjects were told about the virtual refugees before the experiment, they were given no instructions about responding to them such as following them or avoiding them.

The experiment proceeded as follows. When a disaster warning was issued for the area around Kyoto University, the commander started the evacuation guide. The commander grasped the disaster scene through the transcendence interface, selected groups of agents, and give them rough directions for safe shelter or evacuation. The guide agents received the location information from the GPS-capable cellular phones and sent navigation maps to the users. When a second disaster occurred, it was displayed to the commander. The commander changed the navigation instructions the evacuees impacted by the second disaster. The commander could also provide personal navigation instructions to evacuees via e-mail.

5.2. Result of Augmented Experiment

Our purpose was to discern if human subjects would participate in the navigation experiment as if they are in the midst of a large crowd. In this experiment, each navigation map showed the directions to shelters, locations of dangerous sites, and the positions and movements of simulation agents.

In the experiment, the subjects were made aware of the presence of the simulation agents via the maps displayed on their mobile phones. After the experiment, five human subjects were asked how they felt about this approach. All of them said that they had formed their evacuation plans after considering the virtual refugees. While some of them followed the virtual refugees, other avoided them. This shows that even a simple graphic representation notification of the virtual evacuees can influence human behavior.

One purpose of the augmented experiment is to refine the models of simulation agents based on the models of human beings extracted from data captured dur-

ing the experiment. Since the human subjects were dispersed widely across the test area, the data that could be captured was limited to just the time and location at which each subject requested a new map. This was supplemented by replies to questionnaires and interviews performed after the experiment. More detailed behavior of human subjects can be collected from small real-world experiments and participatory simulations. In small real-world experiments, the behavior of human subjects can be recorded by cameras. In participatory simulations, behavior can be reproduced from the histories of avatars.

6. Conclusion

To develop a navigation service for humans with mobile terminals, it is necessary to test the system with a large number of human subjects. However, any experiment in the real world with many human is too expensive and rather dangerous. Thus we took the augmented experiment approach which uses a multiagent simulation to expand a real-world experiment with a few subjects.

We produced a city-wide evacuation guide system assuming the use of GPS-capable cellular phones. In this system, each user receives updated maps and instructions from his/her own guide agent. We subjected this system to an augmented experiment.

In the augmented experiment, the positions and movements of simulated users was shown on the navigation maps sent to each subject. Interviews of the human subjects confirmed that the system successfully gave the impression to the human subjects that they were participating with a large number of users. The results of this experiment give some indication of the possibility of using augmented experiments to refine city-wide navigation services.

Future works included determining how to extract models of human subjects from the results of augmented experiments and how to use the models to enhance simulation accuracy.

Acknowledgment

We would like to thank Mr. Gaku Yamamoto and Mr. Hideki Tai at IBM Japan Tokyo Research Laboratory, and Mr. Akinari Yamamoto at Mathematical Systems Inc., for their various help. This work was supported by a JSPS Grant-in-Aid for Scientific Research (18200009), a Grant-in-Aid for JSPS Fellows and the Strategic Information and Communications R&D Promotion Programme.

References

- [1] L. Gasser and K. Kakugawa. Mace3j: fast flexible distributed simulation of large, large-grain multi-agent systems. In *AAMAS-02*, pages 745–752, New York, NY, USA, 2002. ACM Press.
- [2] O. Gutknecht and J. Ferber. The madkit agent platform architecture. In *Infrastructure for Agents, Multi-Agent Systems, and Scalable Multi-Agent Systems*, volume 1887, pages 48–55. Springer, June 3–7 2001.
- [3] P. Guyot, A. Drogoul, and C. Lemaitre. Using emergence in participatory simulations to design multi-agent systems. In *AAMAS-05*, pages 199–203, New York, NY, USA, 2005. ACM Press.
- [4] T. Ishida. Q: A scenario description language for interactive agents. *Computer*, 35(11):42–47, 2002.
- [5] T. Ishida. Society-centered design for socially embedded multiagent systems. In *CIA-04*, volume 3191, pages 16–29, 2004.
- [6] T. Ishida, Y. Nakajima, Y. Murakami, and H. Nakanishi. Augmented experiment: Participatory design empowered by multiagent simulation. In *IJCAI-07*, 2007.
- [7] L. Jin and T. Miyazawa. Mrm server: a context-aware and location-based mobile e-commerce server. In *WMC-02*, volume 2532, pages 33–39, New York, NY, USA, 2002. ACM Press.
- [8] H. Kitano, S. Tadokor, H. Noda, I. Matsubara, T. Takahashi, A. Shinjou, and S. Shimada. Robocup rescue: search and rescue in large-scale disasters as a domain for autonomous agents research. In *SMC-99*, volume VI, pages 739–743, Tokyo, Oct. 1999.
- [9] T. Koyanagi, Y. Kobayashi, S. Miyagi, and G. Yamamoto. Agent server for a location-aware personalized notification service. In *MMAS-04*, volume 3446, pages 224–238. Springer, 2004.
- [10] Y. Murakami, Y. Sugimoto, and T. Ishida. Modeling human behavior for virtual training systems. *AAAI-05*, pages 127–132, 2005.
- [11] H. Nakanishi, S. Koizumi, T. Ishida, and H. Ito. Transcendent communication: location-based guidance for large-scale public spaces. In *CHI-04*, pages 655–662, New York, NY, USA, 2004. ACM Press.
- [12] G. Yamamoto and H. Tai. Performance evaluation of an agent server capable of hosting large numbers of agents. In *AGENTS-01*, pages 363–369, New York, NY, USA, 2001. ACM Press.

Augmented Experiment: Participatory Design with Multiagent Simulation

Toru Ishida[†], Yuu Nakajima[†], Yohei Murakami[‡], Hideyuki Nakanishi[§]

[†]Department of Social Informatics, Kyoto University

[‡]The National Institute of Information and Communications Technology

[§]Department of Adaptive Machine Systems, Osaka University

ishida@i.kyoto-u.ac.jp, nkjm@ai.soc.i.kyoto-u.ac.jp, yohei@nict.go.jp, nakanishi@ams.eng.osaka-u.ac.jp

Abstract

To test large scale socially embedded systems, this paper proposes a multiagent-based participatory design that consists of two steps; 1) *participatory simulation*, where scenario-guided agents and human-controlled avatars coexist in a shared virtual space and jointly perform simulations, and the extension of the participatory simulation into the 2) *augmented experiment*, where an experiment is performed in real space by human subjects enhanced by a large scale multiagent simulation. The augmented experiment, proposed in this paper, consist of 1) various sensors to collect the real world activities of human subjects and project them into the virtual space, 2) multiagent simulations to simulate human activities in the virtual space, and 3) communication channels to inform simulation status to human subjects in the real space. To create agent and interaction models incrementally from the participatory design process, we propose the participatory design loop that uses deductive machine learning technologies. Indoor and outdoor augmented experiments have been actually conducted in the city of Kyoto. Both experiments were intended to test new disaster evacuation systems based on mobile phones.

1 Introduction

Ubiquitous/pervasive computing systems in the public space often interact with anonymous users. We propose to apply multiagent technologies to the design of those systems that are to be embedded in society.

Participatory simulations have been already studied intensively for modeling human societies [Drogoul *et al.*, 2002, Guyot *et al.*, 2005]. For designing socially embedded systems, however, simulations in virtual space cannot reproduce the reality of the actual application environment. The goal of this paper is to introduce real world experiments to the process of participatory design, so as to bridge the gap between participatory simulations and services in operation. We propose multiagent-based participatory design methodologies (*participatory design* hereafter) to test socially embedded systems in the following steps.

1. Describe interactions between human users and socially embedded systems so as to define the expected user behaviors with socially embedded systems (*interaction model* hereafter).
2. Perform a multiagent-based simulation by modeling users under the given interaction scenarios (*agent model* hereafter). The simulation takes place in virtual space and involves a large number of simulated users. Results of the simulation can predict how the entire system would work in society.
3. Replace some of the simulated users by human-controlled avatars to perform a multiagent-based participatory simulation (*participatory simulation* hereafter). The simulation is performed in virtual space, and the avatars are controlled by human subjects sitting in front of their personal computers.
4. Perform experiments in real space to try out the entire system with human subjects. Since the number of subjects is often limited, the experiment should be augmented by a large number of simulated users in virtual space. We called this the multiagent-based augmented experiment (*augmented experiment* hereafter).

To learn from the participatory design process, this paper separates agent models from interaction models: the former covers the beliefs, desires, intentions, and emotions of human users, and the latter covers protocols, methods, rules, or laws that guide users when interacting with the socially embedded systems. We use extended finite state automata for describing interaction models, while production rules are used to describe agent models for approximating users. We view interaction models as behavioral guidelines of human users playing with socially embedded systems; users keep autonomy within the given guidelines. The question is whether or not users will follow the guidelines in an actual service environment. In other words, the effectiveness of interaction models depends on the agent models, which include user personalities such as deliberative and reactive.

In this paper, we call the descriptions of interaction models *scenarios*. We use scenario description language Q, which can simultaneously interpret scenarios for a large number of agents [Ishida, 2002b]. Q has been already connected to FreeWalk [Nakanishi *et al.*, 2004b] for virtual city experi-

ments [Ishida, 2002a], Cormas for a natural resource simulation [Bousquet *et al.*, 1998], and Caribbean [Yamamoto *et al.*, 1999] for massively multiagent simulations [Ishida *et al.*, 2005]. Q, FreeWalk, and Caribbean have been used to conduct indoor and outdoor disaster evacuation experiments. Those real-scale experiments confirm the feasibility and usefulness of augmented experiments.

2 Participatory Simulation

In our participatory design methodology, we first conduct multiagent-based simulations. Figure 1(a) illustrates how a multiagent-based simulation is realized. The scenario processor interprets interaction models and requests agents in virtual space to perform sensing and acting functions. Note that, since agents are autonomous and have their own agent models, though agents receive instruction according to the scenarios, there is no guarantee that they will behave as requested.

We can easily extend multiagent-based simulations to yield participatory simulations by replacing some of the scenario-guided agents with human-controlled avatars. Figure 1(b) illustrates how human subjects and agents can cooperatively perform a simulation. Just as with video games, human subjects can join the simulation by controlling avatars via joy sticks, mice, or other input devices. To analyze simulation results, we monitor the entire process of the simulation by visualizing the virtual space. In addition to video taping the virtual space, we record how the human subjects control their avatars. Recording human behavior is useful for analyzing the simulation results and for improving the agent and interaction models.

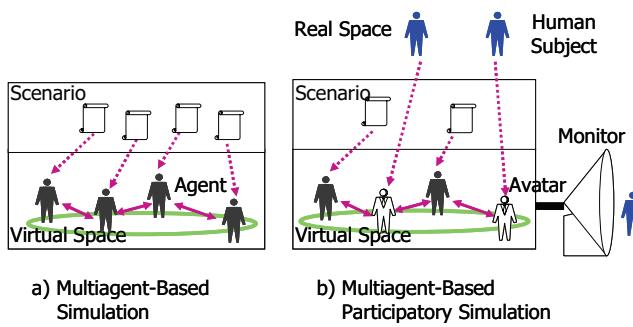


Figure 1. Participatory Simulation

In summary, a participatory simulation consists of 1) agents for modeling users, 2) avatars to represent human subjects, 3) scenarios for modeling interactions, 4) human subjects to control avatars, 5) virtual space to represent real space, and 6) a monitor to visualize simulations ongoing in the virtual space.

We conducted a multiagent-based simulation in the disaster evacuation domain. To establish the process of refining agent models and interaction models, we simulated the precisely controlled experiments conducted by Sugiman [Sugiman *et al.*, 1998]. He developed a simple environment with human subjects to determine the effectiveness of two evacuation methods (i.e., interaction models): the *Follow-direction*

method and the *Follow-me method*. In the former, the leader shouts out evacuation instructions and eventually moves toward the exit. In the latter, the leader tells a few of the nearest evacuees to follow him and actually proceeds to the exit without verbalizing the direction of the exit.

A participatory simulation was performed in which we replaced twelve out of twenty scenario-guided agents by human-controlled avatars. The virtual space was designed so that the human subjects could not distinguish agents from avatars. After the simulation, we applied deductive machine learning technology with domain knowledge to refine the agent models from the logs of the participatory simulation, and successfully modified the agent models and original evacuation scenarios (i.e., interaction models).

Once an accurate model is acquired from participatory simulations, it becomes possible to simulate experiments that cannot be conducted in real space. For example, Sugiman's experiment was performed using two or four leaders, but we can vary this number to clarify the relation between the number of leaders and the time required for evacuation. Moreover, we can explore an effective combination of the *Follow-me* and *Follow-direction* methods. Given accurate and explainable agent models, even though the models can only capture a part of the variations possible in human behavior, participatory simulations must be seen as useful in educating or training people. People can be provided with a learning environment wherein evacuations or other drills can be experienced.

3 Augmented Experiment

Participatory simulations are particularly useful when conducting controlled experiments: they make it easy to prepare the application environments for testing, and user behavior can be recorded for later analysis. However, it is sometimes fails to provide enough reality to allow the testing of ubiquitous/pervasive computing environments.

To understand how users behave with socially embedded systems, real-world experiments are often required. In the case of video phones, for example, since user behavior in everyday life is not easy to simulate, it is essential to observe how individual users employ their video phones in real space. In ubiquitous/pervasive computing, however, because a large number of electronic devices will be embedded in public spaces like railway stations and are in continuous service, it is desired but often impossible to conduct experiments with a large number of human subjects. Augmented experiments offer the ability to use just a small number of human subjects to perform an experiment that can be enhanced by large scale multiagent-based simulations.

Figure 2 illustrates how augmented experiments are realized. Figure 2(a) represents a real world experiment, where human subjects communicate and perform an experiment in real space. Figure 2(b) shows how we introduce a virtual space into the real world experiment. The sensors in the real space captured the behavior of human subjects for reproduction in the virtual space. The sensors can be cameras, RFIDs, or GPS depending on the environment. By using the virtual space, we can monitor the entire experiment from various

viewpoints. Furthermore, we can communicate with the human subjects in the real space through their avatars in the virtual space. Transcendent communication is a new monitoring interface, where a visually simulated public space provides a more flexible and consistent view than regular surveillance systems [Nakanishi *et al.*, 2004a].

Figure 2(c) illustrates an augmented experiment. In parallel with a real world experiment, a large scale multiagent-based simulation is conducted in virtual space: the experiment is augmented by the simulation. To provide social reality to the human subjects, scenario-guided extras are placed around the subjects. In contrast to the participatory simulations, the human extras do not control avatars: the human extras in the real space are controlled by the scenario-guided agents. For example, we can use human extras acting as evacuation leaders to conduct disaster evacuation drills. In contrast to participatory simulations where human subjects sense and act in virtual space, augmented experiments allow subjects to sense and act in real space. Conducting an augmented experiment is possible if the real space is equipped with enough sensors.

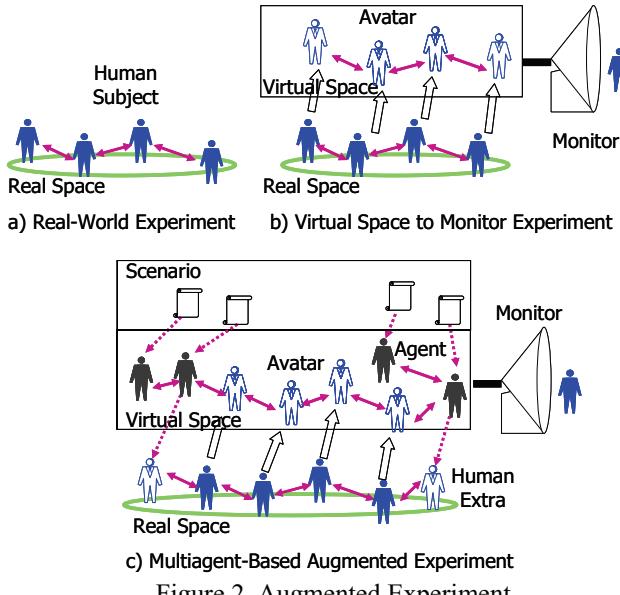


Figure 2. Augmented Experiment

In summary, an augmented experiment consists of 1) agents for modeling users, 2) avatars to represent human subjects, 3) scenarios for modeling interactions, 4) human subjects to act experiments, 5) virtual space to represent real space, 6) a monitor to visualize the experiment in real space enhanced by simulations in virtual space, 7) sensors to reproduce human subjects in virtual space as avatars, 8) communication channels between real and virtual spaces, and 9) human extras to represent agents in virtual space for interacting with human subjects.

The issues on augmented experiments are as follows.

1. Seamless connections between virtual and real spaces:
The difficulty with augmented experiments is to provide the human subjects with a sufficient level of reality. To guide human subjects in a timely fashion, the subjects

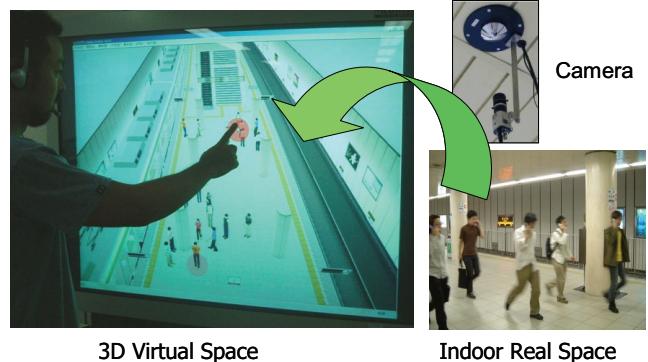
must receive sensing information through the communication channels or from the corresponding human extras. To ensure the subjects naturally behave in real space, the selection of sensor devices and communication media becomes crucial. Visual, auditory and textual interactions are to be organized appropriately.

2. Learning from data obtained in virtual and real spaces:
Learning issues are similar to those in participatory simulations, but their evaluation is more difficult, since the behaviors of human subjects in real space are harder to analyze than those in participatory simulations. Though we can monitor the experiment via virtual space, video capture of the human subjects in real space is also indispensable for analyzing the entire experiment.

To confirm the feasibility and usefulness of augmented experiments and to determine their future issues, we conducted indoor and outdoor evacuation experiments in real space augmented by a large scale multiagent-based simulation.

4 Indoor Experiment

Everyday, more than 300,000 passengers pass through Kyoto station, the main railway station in Kyoto. In this station, using Q [Ishida, 2002b] and FreeWalk [Nakanishi *et al.*, 2004b], we installed a disaster evacuation system that tracks passengers to help their navigation based on their current positions. Beyond conventional navigation systems, which announce route information using public loudspeakers, our system sends instructions to individuals using mobile phones. Augmented experiments are required for testing the evacuation system, because there is no other way to conduct experiments with enough reality.



3D Virtual Space Indoor Real Space

Figure 3. Indoor Experiment

The augmented experiment was designed as follows. As the sensors, we placed twenty eight cameras in Kyoto station, and captured the movements of passengers in real time. The cameras are specially designed to detect passengers' behavior but not personal features. Figure 3 shows the omnidirectional cameras placed in the station. As the virtual space, we used FreeWalk, a three dimensional virtual city system and used it to reproduce the passengers' behavior. We implemented a monitor based on transcendent communication [Nakanishi *et al.*, 2004a]. Figure 3 includes a snapshot of the monitoring system; human subjects on the station platform are projected

onto *avatars* in virtual space. A bird's-eye view of the real space is reproduced on the screen of the control center so that evacuation leaders in the center can easily monitor the experiment. As the *communication channels*, the leader can point at particular passengers on the screen, and talk to them through their mobile phones. When the monitor detects pointing operations, a wireless connection is immediately activated between the control center and the indicated passengers.

A multiagent-based simulation with a large number of *agents* controlled by evacuation *scenarios* was performed in parallel with the experiment in real space. See-through head-mounted displays are not suitable for presenting the simulation of augmented experiments, since it is unsafe to mask the views of passengers. As described above, since we used mobile phones, small and low-resolution images of three dimensional virtual spaces are difficult to understand. Instead of displaying visual simulations, the mobile phones displayed symbols that directly represent what human subjects need to understand. We did not use *human extras* in this experiment.

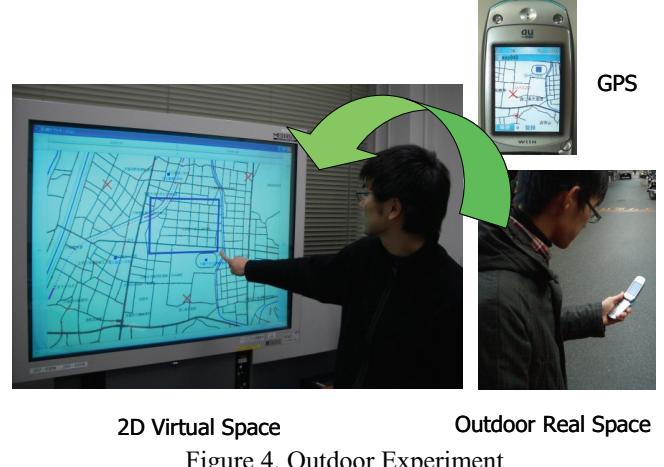
We conducted the indoor augmented experiment in February 2005. From this experience, we learned that insufficient visual reality prevented the human subjects from recognizing the crowd of agents around the staircase. It appears that the usability of an augmented experiment depends significantly on the user interface for interacting with the agents.

5 Outdoor Experiment

One year after the indoor experiment, we implemented a large-scale outdoor evacuation system for emergency situations using Q [Ishida, 2002b] and Caribbean [Yamamoto *et al.*, 1999], and conducted an augmented experiment in January 2006. This system architecture is close to that of our indoor experiment, but the *sensors* were GPS devices instead of omnidirectional cameras, and as the *virtual space*, we used a two dimensional map instead of a three dimensional virtual city system. The virtual space is displayed on the *monitor* screen of the control center in a birds-eye view, so that the leader could grasp how evacuees were behaving in the experiment. Evacuees on a screen consisted of a small number of *avatars* representing the *human subjects* in real space, and a large number of *agents* controlled by evacuation *scenarios* in virtual space. In the actual experiment, ten to thirteen humans and three thousand agents per each time undertook the augmented experiment.

The location of human subjects in real space was projected into virtual space based on their positions acquired by GPS. This map showed fires, blocks routes, and safe areas in real time. The human subjects could always get the latest map by sending their location. The evacuation leaders assigned evacuation destinations and directions to each evacuee through the monitor screen shown in Figure 4. The leader issued high level instructions to the evacuees using the map, and precise navigation instructions were automatically generated for each evacuee. Dragging operations indicating a rectangular area enabled the leader to broadcast an announcement to a group of evacuees.

As in the indoor experiments, the evacuation leader activated *communication channels* to particular evacuees by pointing at the evacuees on the screen. The leader could talk to the particular evacuees and provide customized navigation. We used *human extras* as evacuation leaders in this experiment to check the evacuation routes taken by human subjects in real space.



2D Virtual Space

Outdoor Real Space

Figure 4. Outdoor Experiment

The human subjects could grasp the experiment from the screen of their mobile phones. The map around the place the subject was standing was displayed together with information of fires, blocked routes, and safe areas, just as on the monitor in the control center. The moves of other evacuees were also displayed on their mobile phones. From this experience, we learned that a map can be an excellent interface between human subjects and agents as well as evacuation leaders. Unlike the indoor experiment, since route selection is the main issue in the outdoor experiment, the human subjects did not have much difficulty in imagining the disaster situation. However, if the number of agents was high in the outdoor experiment, then virtual crowding would become a problem. Human extras playing evacuation leaders can increase the reality of the experiment.

6 Learning from Experiments

Multiagent-based simulations use *production rules* to describe agent models for approximating human users, and *extended finite state automata* to describe interaction models among users and socially embedded systems. By analyzing the logs of participatory simulations or augmented experiments, we can refine both agent and interaction models. The learning process of multiagent-based participatory design consists of two phases: the *normal design loop* and the *participatory design loop*.

At first, scenario writers create agent and interaction models and verify their design by multiagent-based simulations. This process is called the *normal design loop* and its detailed steps can be found in a previous work [Murakami *et al.*, 2003]. After verifying the agent and interaction models, participatory simulations and augmented experiments are performed. The results of which may be different from those

of multiagent-based simulations, because human behaviors are often more variable than those of agents modeled by scenario writers. If this is the case, we first refine the agent models to better express the variation seen in human behavior. We then perform the multiagent-based simulations again with the new agent models. If the new simulations succeed to express various human behaviors but fail to produce the expected results, we then refine the interaction models; i.e. modify protocols, methods, rules, and/or laws. After the improved interaction model is confirmed by multiagent-based simulations, participatory simulation and augmented experiment take place again. This loop, called the *participatory design loop*, is illustrated in Figure 5.

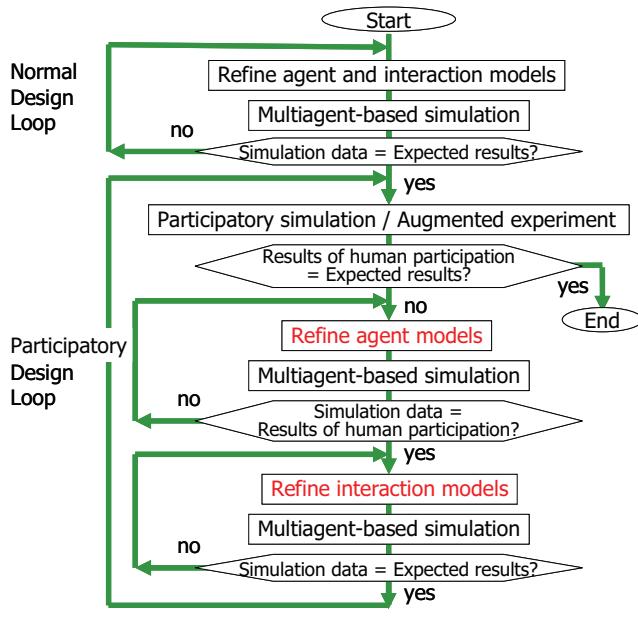


Figure 5. Participatory Design Loop

It would seem, at first glance, that inductive machine learning technologies could be applied to refine agent and interaction models. However, it is extremely rare to have a sufficient number of human subjects to gather enough data for inductive learning: deductive methods with domain knowledge are more feasible [Torii *et al.*, 2006]. To refine agent models, given that the training examples will be limited, explanation-based learning is a reasonable approach. Though variations exist in human behavioral rules such as “escape from crowds” and “follow crowds,” observing the behavior of human-controlled avatars can create a plausible explanation that selects one of the two rules.

By combining machine learning technologies with interviews of human subjects, agent models have been successfully refined in several domains [Murakami *et al.*, 2005]. To refine interaction models, if we have enough candidate protocols, we can try them, and select the one that best fits. If we need to invent a new model, however, there is no available technology that can create such an interaction model. The visualization of simulations and experiments can, however, support scenario writers in inventing new models.

7 Related Work

There are two types of multiagent-based simulations and they have different goals; a) *analytic multiagent-based simulations* often with simple agent and interaction models and b) *synthetic multiagent-based simulations* sometimes with complex agent and interaction models. Analytic simulations have been used to model social systems [Epstein and Axtel, 1996]. Here, the KISS principle (Keep It Simple, Stupid) is often applied [Axelrod, 1997]. The KISS principle states that modeling should be simple even though the observed phenomenon is complex, and that complexity should be a result of repeated interaction among agents and their environment. Hence, agents are often modeled with limited parameters. This approach is effective in the analysis of macro-micro links, the relationship between the macro properties of the entire system and the micro properties of the agents constituting the system.

On the other hand, synthetic simulations are used to produce realistic situations. Agent and interaction models can be complex reflecting the real world to make the simulation as realistic as possible. This approach is used in an early stage of system development [Noda and Stone, 2003], and in education or training [Rickel and Johnson, 2000]. Since our purpose is to design socially embedded systems, our simulation is synthetic: our agent and interaction models can be complex to reproduce realistic situations; after a participatory simulation, an augmented experiment is performed to understand user behavior in a more realistic environment.

Our work appears to follow various multiagent-based simulation papers including the RoboCup soccer simulator [Noda and Stone, 2003], but is clearly unique due to its focus on participatory design methodology. Drogoul proposed a methodological process for developing multiagent-based simulations, and introduced the idea of the participatory design of simulations [Drogoul *et al.*, 2002], while we are proposing the participatory design for socially embedded systems. Bousquet applied role games to the modeling of social systems [Bousquet *et al.*, 2002]. This paper, however, applies multiagent-based simulation to software system design, not social system analysis.

We propose the use of participatory methodology including simulations and experiments for designing large-scale socially embedded systems. This methodology differs from the Gaia methodology [Zambonelli *et al.*, 2003] or Opera [Vázquez-Salceda *et al.*, 2005], which focus on an early stage of designing multiagent systems mainly for enterprise applications [Jennings, 2000]. Our methodology can be applied to designing ubiquitous/pervasive computing systems, but the resulting systems are not necessarily multiagent-based, though multiagent systems are often natural solutions of socially embedded systems.

8 Conclusion

The waterfall model has been used in software development for a long time. Given the increase in human-computer interaction, however, it has become essential to use the user-centered design approach when creating usable and

accessible interactive systems. In future, however, millions of electronic devices with computing facilities will be connected with each other in ad hoc ways, and should behave coherently. It is natural to say we need a new methodology for designing ubiquitous/pervasive computing systems.

In this paper, we proposed the multiagent-based participatory design methodology for ubiquitous/pervasive computing systems, which are to be embedded in human society. Though the concept of multiagent-based participatory design is open-ended, we include the following two components.

We call multiagent-based simulations *participatory simulations* when they include human-controlled avatars and scenario-guided agents. We can monitor the entire process of a participatory simulation by visualizing the virtual space. We call real-world experiments as *augmented experiments* when they are augmented by large scale multiagent-based simulations. In ubiquitous/pervasive computing, because a large number of electronic devices will be embedded in public spaces and in continuous use, it is often impossible to conduct experiments with a large number of human subjects. Augmented experiments enable us to perform such trials with a small number of human subjects.

By introducing augmented experiments and actually trying out indoor and outdoor experiments in the city of Kyoto, we learned the following lessons. Interaction among the human subjects in the real space and agents in the virtual space should be designed carefully. If there exists a medium like geographical map, which human subjects use daily to grasp situations, it is a good idea to adopt them as the interface between the real and virtual spaces. Human extras controlled by scenario-guided agents are also effective to increasing the reality, if they have a clear role such as evacuation leaders.

References

- [Axelrod, 1997] R. Axelrod. *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*. Princeton University Press, pp. 4-5, 1997.
- [Bousquet *et al.*, 1998] F. Bousquet, I. Bakam, H. Proton and C. Le Page. Cormas: Common-Pool Resources and Multiagent Systems. *LNAI 1416*, pp. 826-838, 1998.
- [Bousquet *et al.*, 2002] F. Bousquet, O. Barreteau, P. Aquino, M. Etienne, S. Boissau, S. Aubert, C. Le Page, D. Babin, and J. C. Castella. Multi-Agent Systems and Role Games: Collective Learning Processes for Ecosystem Management. M. Janssen Ed. *Complexity and Ecosystem Management*, Edward Elgar Publishers, 2002.
- [Drogoul *et al.*, 2002] A. Drogoul, D. Vanbergue and T. Meurisse. Multiagent-Based Simulation: Where are the Agents? *LNAI 2581*, pp. 1-15, 2002.
- [Epstein and Axtel, 1996] J. M. Epstein and R. L. Axtell. *Growing Artificial Societies: Social Science from the Bottom Up*. MIT Press, 1996.
- [Guyot *et al.*, 2005] P. Guyot, A. Drogoul and C. Lemaître. Using Emergence in Participatory Simulations to Design Multi-Agent Systems. *AAMAS-05*, pp. 199-203, 2005.
- [Ishida, 2002a] T. Ishida. Digital City Kyoto: Social Information Infrastructure for Everyday Life. *Communications of the ACM (CACM)*, Vol. 45, No. 7, pp. 76-81, 2002.
- [Ishida, 2002b] T. Ishida. Q: A Scenario Description Language for Interactive Agents. *IEEE Computer*, Vol. 35, No. 11, pp. 54-59, 2002.
- [Ishida *et al.*, 2005] T. Ishida, L. Gasser and H. Nakashima Eds. *Massively Multi-Agent Systems I. LNAI*, 3446, Springer-Verlag, 2005.
- [Jennings, 2000] N. R. Jennings. On Agent-Based Software Engineering. *Artificial Intelligence*, Vol. 177, No. 2, pp. 277-296, 2000.
- [Murakami *et al.*, 2003] Y. Murakami, T. Ishida, T. Kawasoe and R. Hishiyama. Scenario Description for Multi-Agent Simulation. *AAMAS-03*, pp. 369-376, 2003.
- [Murakami *et al.*, 2005] Y. Murakami, Y. Sugimoto and T. Ishida. Modeling Human Behavior for Virtual Training Systems. *AAAI-05*, pp. 127-132, 2005.
- [Nakanishi *et al.*, 2004a] H. Nakanishi, S. Koizumi, T. Ishida and H. Ito. Transcendent Communication: Location-Based Guidance for Large-Scale Public Spaces. *CHI-04*, pp. 655-662, 2004.
- [Nakanishi *et al.*, 2004b] H. Nakanishi and T. Ishida. Free-Walk/Q: Social Interaction Platform in Virtual Space. *VRST-04*, pp. 97-104, 2004.
- [Noda and Stone, 2003] I. Noda and P. Stone. The RoboCup Soccer Server and CMUnited Clients: Implemented Infrastructure for MAS Research. *Autonomous Agents and Multi-Agent Systems*, Vol. 7, No. 1-2, pp. 101-120, 2003.
- [Rickel and Johnson, 2000] J. Rickel and W. L. Johnson. Task-Oriented Collaboration with Embodied Agents in Virtual Worlds. I.J. Cassell, J. Sullivan, S. Prevost, and E. Churchill Eds. *Embodied Conversational Agents*, pp. 95-122, MIT Press, 2000.
- [Sugiman *et al.*, 1998] T. Sugiman and J. Misumi. Development of a New Evacuation Method for Emergencies. *Journal of Applied Psychology*, Vol. 73, No. 1, pp. 3-10, 1988.
- [Torii *et al.*, 2006] D. Torii, T. Ishida and F. Bousquet. Modeling Agents and Interactions in Agricultural Economics. *AAMAS-06*, pp. 81-88, 2006.
- [Vázquez-Salceda *et al.*, 2005] J. Vázquez-Salceda, V. Dignum and F. Dignum. Organizing Multiagent Systems. *Autonomous Agents and Multi-Agent Systems*, Vol. 11, No. 3, pp. 307-360, 2005
- [Yamamoto *et al.*, 1999] G. Yamamoto and Y. Nakamura. Architecture and Performance Evaluation of a Massive Multi-Agent System. *Agents-99*, pp. 319-325, 1999.
- [Zambonelli *et al.*, 2003] F. Zambonelli, N. R. Jennings and M. Wooldridge. Developing Multiagent Systems: The Gaia Methodology. *ACM Transactions on Software Engineering and Methodology*. Vol. 12, No. 3, pp. 317-370, 2003.

META-LEVEL CONTROL ARCHITECTURE FOR MASSIVELY MULTIAGENT SIMULATIONS

Shohei Yamane

Department of Social Informatics
Kyoto University

Toru Ishida

Department of Social Informatics
Kyoto University

ABSTRACT

Various situations in a massively multi-agent simulation will emerge in a simulation or the period of the simulation will become too long. These situations cause problems for system operators in that each action scenario becomes too complex to maintain and a simulation costs very long time. Therefore, flexible control of the simulation, such as changing simulation speed and switching agents' action scenarios, is required. We propose a meta-scenario description language and a meta-level control architecture. The meta-scenario description language describes how to control simulations and agents based on an extended finite state machine. Meta-level control architecture achieves control on the basis of meta-scenarios provided by a meta-scenario interpreter, which controls interpreters of agents' action scenarios and the simulation environment. In addition, our proposed architecture does not lose scalability of massively multi-agent systems for some applications.

1 INTRODUCTION

The use of high-performance computers and distributed computing environments enable massively multi-agent systems (MMAS) to manage tens of thousands or millions of agents under discussion (Ishida 2005). Traffic simulation is a typical application of MMAS (Balmer 2004). Using MMAS, we can perform city-wide traffic simulations and disaster simulations. Various situations will emerge in such large-scale multi-agent simulation because of the various behaviors of the different agents. In addition, a simulation needs a considerable amount of time because it takes a long time to fully understand the impact of a certain event on the whole simulation. These cases require flexible control of runtime. We mention the following two examples.

First, in large-scale multi-agent simulations, the action scenarios of the agents is very complicated. Generating agent trips is essential for traffic simulations. A method using an OD-matrix (Balmer 2004) and a method using a genetic algorithm (Charypar 2003) are proposed as trip generation methods. Each trip becomes very complicated because

various situations can emerge in a large scale simulation. We think it is efficient to simplify each action scenario by preparing a certain action scenario for a certain situation and changing them on the basis of the situation.

Second, to construct more realistic agent models through simulations, it is effective to perform participatory simulations, in which some agents are operated by humans, and construct agent models from the resulting log data (Murakami 2005; Guyot 2005). Participatory simulations are often real-time simulations because the operators are actually making the decisions. However, real-time simulations are ineffective if the simulation period is very long. For this reason, flexible time management, such as making the simulation speed equal real-time only while the operators are making decisions, is required.

To achieve flexible control of large-scale multi-agent simulations, we propose a meta-scenario in which a language is used to describe control of simulations and action scenarios. There are two issues to achieve control based on a meta-scenario.

1. Definition of control functions

We need to define control functions required to describe changing action scenarios and time management.

2. Development of a control architecture

We need a control architecture for simulation control on the basis of meta-scenario descriptions. We will implement this architecture on a MMAS so as not to lose scalability of MMAS.

Section 2 contains the scenario description language that describes the action scenarios of the agents. Section 3 contains an explanation of the meta-scenario. Section 4 contains an explanation of the meta-level control architecture to enable simulation control based on the meta-scenario descriptions. Section 5 contains an application of the meta-scenarios for participatory simulations. Section 6 contains an evaluation of the scalability for the meta-level architecture.

2 SCENARIO DESCRIPTION LANGUAGE

In this paper, we chose scenario description language *Q* (http://www.lab7.kuis.kyoto-u.ac.jp/Q/index_e.html) (Ishida 2002) to describes the action scenarios of the agents. The reason for our selection is outlined in this section.

2.1 Specification

A scenario description describes how an agent should behave and interact with its environment involving humans and other agents. A scenario is represented as an extended finite state machine. The language facilities are explained below.

1. *Cue/Action*. Cues are used to request agents to observe their environment and trigger interaction. Cues are not permitted to have any side effects. Whereas, actions are used to request agents to change their environment. These cues and actions are defined by users, and how the agent that receives a request behaves depends on the implementation of the agent system.
2. *Guarded Command/Scenario*. Guarded commands are used in situations that require the simultaneous observing of multiple cues. A guarded command combines cues and actions. If a cue becomes true, the corresponding action is performed. A scenario is used for describing state transitions, and the state transitions are done by expressions. The scenario defines each state as a guarded command.

```
(defscenario overtake
  (&pattern ($x #f))
  (left-lane-normal
    ((?carExists :position "ahead"
      :speed "slow" :car $x)
     (!changeLane :lane "right")
     (!overtake :car $x)
     (go right-lane-normal))
    ((?carExists :position "ahead"
      :speed "fast" :car $x)
     (!changeSpeed :speed "fast")
     (go left-lane-fast)))
  (right-lane-normal
    ...)
  (left-lane-fast
    ...)
  ...)
```

The scenario in the above examples defines the states as left-lane-normal, right-lane-normal, and left-lane-fast. In left-lane-normal, the agent changes its lane and overtakes a car when there is a car driving slowly ahead of the agent, or the agent drives faster when there is a car driving fast

ahead of it.

3. *Agent/Avatar*. Agents and avatars are defined by specifying their names. Assigning a scenario to an agent, the agent begins to follow the scenario. Whereas, an avatar is operated by a human. However, we can assign scenarios to an avatar like an agent. By assigning a scenario to an avatar, we can let the avatar behave automatically while the human is not actively making decisions.

2.2 Scenario Processor

A scenario processor interprets a scenario description and requests the corresponding agent to execute cues, actions, and guarded commands. When the cues, actions, and guarded commands are executed, the agent returns the result to the scenario processor. The scenario processor decides the next command on the basis of this result, and then requests the agent to execute the new command. Scenario processors are separated from agent systems. This means that a meta-scenario processor can control the action scenarios of the agents by controlling the scenario processors.

3 META-SCENARIO

The meta-scenario that describes simulation control is explained in this section.

3.1 Definition of Functions

When the operator of an avatar finishes making a decision in a participatory simulation, the decision is not directly reflected in the simulation environment but is reflected in the avatar's internal state in most cases. Therefore, to observe the operator's decision making, the scenario processing state of the avatar must be observed. To change scenarios on the basis of simulation situations, we have to define functions, such as observation of simulation environment, and change scenarios assigned to agents. In addition, we have to define functions that change simulation environment to change simulation speed.

We propose a meta-scenario language with the control functions mentioned above. We used an extended finite state machine to describe the meta-scenario because simulation situations can be represented by states and controls can be represented by state transitions.

Q, as explained in section 2, is also based on extended finite state machine model. Therefore, we used cues, actions, and guarded commands to describe the meta-scenario descriptions as well as the scenario descriptions.

A summary of cues and actions of the meta-scenario description language is listed in Table 1. The commands

Table 1: Cues and Actions of Meta-Scenario Description Language

Scenario processing control	
?observeAction	observe execution of action
?observeCue	observe execution of cue
?observeTransition	observe execution of state transition
?observeScenario	observe execution of scenario
!runCue	request agent to execute cue
!runAction	request agent to execute action
!releaseScenario	release scenario from agent
!assignScenario	assign scenario to agent
Simulation control	
?observeEnvironment	observe simulation environment
!getEnvironment	get environmental information
!setEnvironment	set environmental information
!createAgent	create agent
!deleteAgent	delete agents
!createCrowd	create crowd of agents
!createAvatar	create avatar
!startSimulation	start simulation
!stopSimulation	stop simulation
observe/get information from real-world	
?observeMetaEnvironment	observe information from real-world
!getMetaEnvironment	get information from real-world

in this table that start with ? are cues and that start with ! are actions.

3.2 Examples of Meta-Scenario

The meta-scenario description language can describe various simulation controls. For example, in a traffic simulation, some agents may prefer to take a detour to avoid a traffic jam, whereas others may not. This situation requires two scenarios to be written for each agent. One scenario involves taking the usual route, and the other scenario involves taking a detour. We then write a meta-scenario to change the agents scenario if the agent, who prefers to avoid traffic jams, experiences a traffic jam on their usual route.

For another meta-scenario, we can write a meta-scenario to describe a simulation scenario. That is, the meta-scenario describes events in the simulation, including the creating agents, the deleting agents, the occurrence of traffic jams, and the occurrence of disasters.

Figure 1 shows an example of a meta-scenario when an agent’s scenario is changed because of a traffic jam. As shown in Figure 1, each state in the meta-scenario is represented by a circle, and each transition is represented by an arrow. In addition, a cue that triggers a transition and

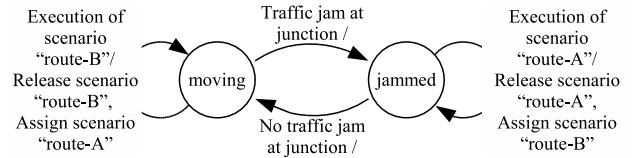


Figure 1: State Machine of Meta-Scenario that Changes Agents’ Scenarios

an action involved by the transition are indicated in form of (cue/action) – the lies on the appropriate state transition arrow. In this example, the meta-scenario assigns a scenario, “route-A”, (take a usual route) when the traffic is moving and assigns a different scenario, “route-B”, (take a detour) when a traffic jam occurs.

Figure 2 shows a meta-scenario description of the meta-scenario shown in Figure 1; the cues and actions listed in Table 1 are used.

4 ARCHITECTURE FOR SCENARIO PROCESING CONTROL

The meta-level control architecture to achieve the meta-scenario processor is explained in this section. The control flow for scenario processing is also explained.

Meta-level architecture has been described elsewhere (Gao 2002). Each agent has its own meta-layer to debug scenarios (Gao 2002). In this paper, a meta-level controls an entire simulation, including the environment and agents.

4.1 Meta-Level Control Architecture

Figure 3 shows the meta-level control architecture we designed.

Scenario Description A scenario description outlines an agent’s behavior using the language facilities explained in Section 2.

Scenario Interpreter A scenario interpreter corresponds to an agent in an agent system and interprets a scenario description. The result of this processing is a request to the agent to trigger an observation or to perform an action.

Meta-scenario description A meta-scenario describes the control and observation of scenario processing, such as observation and changing the command executed by the scenario interpreter. The meta-scenario also describes the setting of and acquisition of environmental information of the agent system, such as changing simulation speed.

Meta-scenario interpreter A meta-scenario interpreter interprets the meta-scenario descriptions. A meta-scenario requests the scenario interpreter to ob-

```
(defmetascenario avoid-jam
  (&pattern (agent #f))
  (moving
    ; observe traffic jam
    (?observeEnvironment
      :name "traffic jam"
      :value #t)
    (go jammed)
    ; observe execution of 'route-B',
    ; and set variable 'agent' to
    ; the agent
    (?observeScenario :name 'route-B
      :agent agent)
    ; release 'route-B' from agent
    (!releaseScenario :name route-B
      :agent agent)
    ; assign 'route-A' to agent
    (!assignScenario :name route-A
      :agent agent)
    (go moving))
  (jammed
    ; observe the traffic is moving
    (?observeEnvironment
      :name "traffic jam"
      :value #f)
    (go moving))
    ; observe execution of 'route-A'
    ; and set variable 'agent' to
    ; the agent
    (?observeScenario :name 'route-A
      :agent agent)
    ; release 'route-A' from agent
    (!releaseScenario :name 'route-A
      :agent agent)
    ; assign 'route-B' to agent
    (!assignScenario :name 'route-B
      :agent agent)
    (go jammed)))

```

Figure 2: Meta-Scenario Description that Changes Agents Scenarios

serve the scenario description processing, requests an agent to trigger an observation or perform an action, and requests the agent system to set/get environmental information.

Agent System An agent system receives and processes all the requests for the agents.

One meta-scenario interpreter in our meta-level control architecture controls all scenario interpreters. As an alternative architecture, we can consider that an individual meta-scenario interpreter controls an individual scenario interpreter. This alternative architecture has an advantage in controlling scenario interpreters because each scenario interpreter has its own meta-scenario interpreter. However, meta-scenario interpreters in such an architecture need to coordinate to control the simulation environment. For example, when a meta-scenario interpreter is performing the simulation at speed slow while another is it performing fast, they need to communicate to determine what the writer really wants. The cost of coordination will be high, and the result may be wrong.

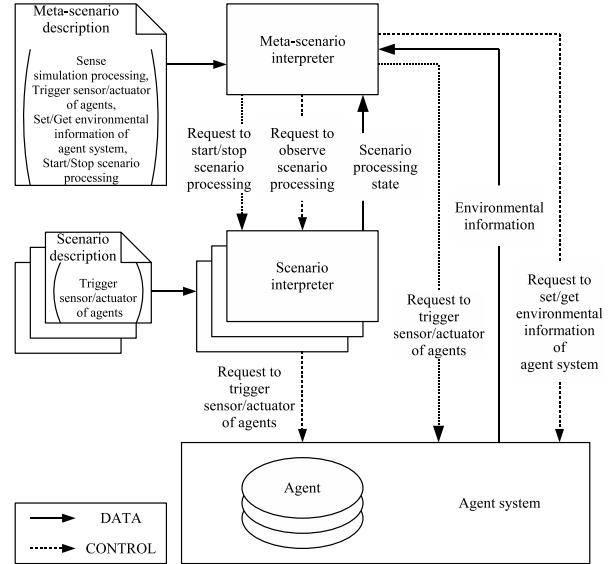


Figure 3: Meta-Level Control Architecture

We can assign more than one meta-scenario to one simulation. For example, using a meta-scenario to change the simulation speed and a meta-scenario to change scenarios enables time-management and scenario switching. Unfortunately, coordination becomes a problem when multiple meta-scenario interpreters are run.

4.2 Flow of Meta-Scenario Execution

We will explain the flow in meta-scenario processing for changing the actions of a scenario. The changing of actions is described using `?observeAction` and `!runAction` commands. We described an example that changes the scenario when a traffic jam occurs in section 3.2. In this subsection, we assume that one action can describe a route an agent take (e.g., `(!takeRoute :route route-A)`) is an action to request the agents to take route-A). Figure 4 shows a sequence diagram in which the meta-scenario change the action from a usual route to a detour.

1. The meta-scenario interpreter reads the meta-scenario description and determines the next command to be executed. The meta-scenario interpreter in this example requests all scenario interpreters to observe the action taken for the usual route.
2. The scenario interpreter reads its scenario description and decides the next command to be executed.
3. The scenario interpreter tests whether the command determined in step 2. is observed by the meta-scenario interpreter (i.e. the action taken is the usual route).

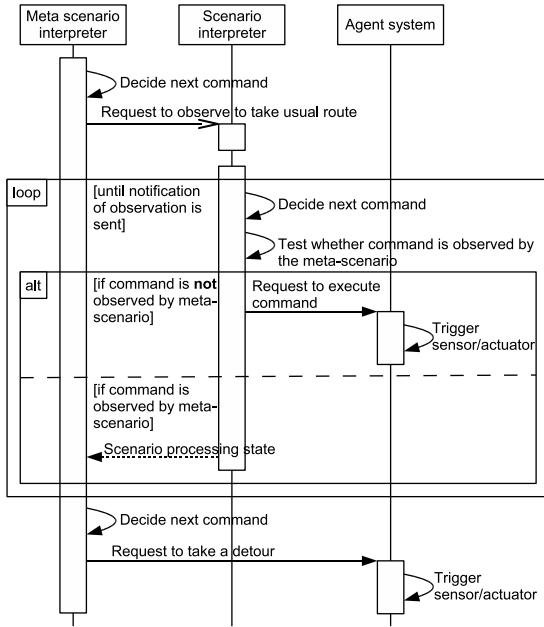


Figure 4: Sequence Diagram of Scenario Execution Control

4. If the command is not observed, the scenario interpreter requests the agent to execute sensor/actuator and returns to step 2.
5. If the command is observed, the scenario interpreter sends a scenario processing state to the meta-scenario interpreter.
6. The meta-scenario interpreter receives the observation result, reads the meta-scenario description, and determines the next command to execute.
7. In the example shown in Figure 4, the next command is to request the agent to take the usual route to execute an action that takes a detour.

The costs associated with implementing meta-level control architecture are expensive, therefore we have to reduce the extra cost of meta-level control. The extra cost is mainly caused by observation/control of scenario interpreters. If a meta-scenario observes and controls all the processes of each scenario interpreter, we can achieve more flexible control. However, the overhead with implementing such a system becomes quite high because message transfer between the meta-scenario interpreter and the scenario interpreter occurs at each step of the scenario interpreters. Each scenario interpreter in the proposed implementation contains its own scenario controller. The meta-scenario requests the scenario controllers to observe and control the scenario processing, the scenario controller observes and controls the scenario interpreter, and the scenario interpreter then sends the results to the meta-scenario interpreter. In this manner, message transfer between the meta-scenario interpreter and the sce-

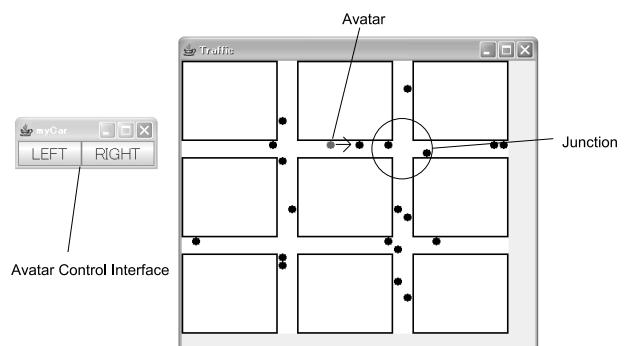


Figure 5: GUI of Simulation

nario interpreter occurs only when observation or control is done.

5 APPLICATION TO A PARTICIPATORY SIMULATION

An example of using a meta-scenario in a participatory simulation is described in this section.

Some agents, called avatars, are operated by humans in participatory simulations. The simulation speed of a participatory simulation must be slowed (or raised) to real-time speed while the operators are making decisions. With the meta-scenario, time-management like this is achieved by controlling the simulations on the basis of observations of the avatars and simulations.

5.1 The Simulation Performed in This Section

A graphical user interface of the simulation is shown in Figure 5. The right window shows the road grid. The left window shows the user interface for the operator of the avatar. In this simulation, many agents are driving on a road grid, with one avatar driving straight along a road. The human operating the avatar is assumed to make a decision when the avatar approaches the next junction (indicated by the circle shown in Figure 5). The operator makes his decision and clicks the "LEFT" or "RIGHT" button as desired.

5.2 A Scenario for an Avatar

The state transition diagram of the avatar scenario is shown in Figure 6. In the absence of any direct order from the operator, the avatar keeps driving straight if there is no car in front of it; if there is such a car, the avatar stops.

As shown in Figure 6, the scenario switches to the turn-left state when the operator clicks the "LEFT" button, alternatively the scenario switches to the turn-right state when the operator clicks the "RIGHT" button. Regardless of when the turn-left/turn-right state is

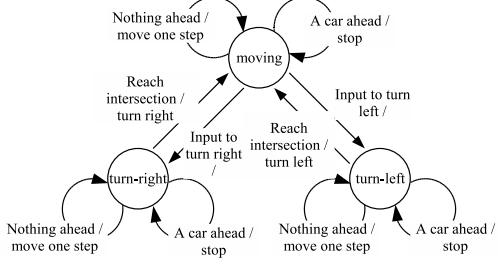


Figure 6: Scenario for Avatars

entered, the avatar keeps going straight until it reaches the intersection where it can make the selected turn. Once the turn is completed, the scenario transits to the moving state.

Assigning a scenario in this manner to an avatar can automate its behavior. This frees the operator from having to make all decisions, like stopping the avatar to avoid a collision.

5.3 A Meta-Scenario of Time-Management

In this simulation, we assumed that the operator of the avatar needs to make a decision only when the avatar approaches a junction. Figure 7 shows a meta-scenario when the simulation speed is slowed as the avatar approaches a particular junction and when the simulation speed is increased once the operator has finished making his decision (by clicking one of the buttons). The state transition diagram of this meta-scenario is shown in Figure 8.

This meta-scenario runs in the following way. First, the simulation is initialized in the initial state and transits to the on-road state. In the on-road state, when the avatar is far from the junction, the simulation speed is faster than real-time. When the avatar approaches the junction, the simulation speed is slowed, and the meta-scenario state transits to the near-junction state. If the operator has made a decision, the state of the avatar scenario transits to the appropriate state. The operator's decision is detected by using a ?observeTransition command in the near-junction state. Once the decision is detected, the simulation speed is increased. To prevent the near-junction state from being unnecessarily reentered (which would trigger a slow down in simulation speed), the meta-scenario waits some time after the avatar leaves the junction in the within-junction state.

As shown in this example, the operators' decision making is not always directly reflected in the simulation environment. However, the decision should be reflected in the avatars' internal states. The meta-scenario control architecture can achieve time management by utilizing scenario processing, such as state transitions of scenarios. This example shows that monitoring and controlling the scenario processing of agents at the meta-level enables simple and

```

(defmetascenario meta-traffic ()
  (initial
    (#t
      (!createCrowd :name 'Cars :population 20)
      (!createAvatar :name 'myCar)
      (!assignScenario :name 'car :agent Cars)
      (!assignScenario :name 'car-avatar
        :agent myCar)
      (!startSimulation) (go on-road)
      ; observe myCar to approach the junction(2,1)
      (?observeEnvironment :name 'onIntersection
        :args (list myCar 2 1))
      ; slow simulation speed
      (!setEnvironment :name 'sim-speed
        :args "slow")
      (go near-junction)))
    (near-junction
      ; observe myCar to leave the junction(2,1)
      (?observeEnvironment :name 'outOfIntersection
        :args (list myCar 2 1))
      ; raise simulation speed
      (!setEnvironment :name 'sim-speed
        :args "fast")
      (go on-road)
      ; observe myCar to transit to 'turn-right'
      ; in the scenario for the avatar 'car-avatar'
      (?observeTransition :scenario 'car-avatar
        :scene 'turn-right
        :agent myCar)
      ; raise simulation speed
      (!setEnvironment :name 'sim-speed
        :args "fast")
      (go within-junction)
      ; observe myCar to transit to 'turn-left'
      ; in the scenario for the avatar 'car-avatar'
      (?observeTransition :scenario 'car-avatar
        :scene 'turn-left
        :agent myCar)
      ; raise simulation speed
      (!setEnvironment :name 'sim-speed
        :args "fast")
      (go within-junction))
    (within-junction
      ; observe myCar to leave the junction(2,1)
      (?observeEnvironment :name 'outOfIntersection
        :args (list myCar 2 1))
      (go on-road))))
  
```

Figure 7: Meta-Scenario Description of Time-Management

flexible control of agents and simulations based on observations from simulation processing, including the agents' behavior.

6 EVALUATION

We showed how to control simulations by using the meta-scenario in the previous sections. The aim of a meta-scenario is to enable flexible control of large-scale simulations. As stated in Section 4, scalability is also an important issue. Unfortunately, the system used in Section 5 is a prototype, and we have not developed a system with enough scalability. We have developed a system that can manage some hundreds of thousands of scenario interpreters using Caribbean (Yamamoto 2001), which was developed by IBM. Therefore, we can create a MMAS with a meta-scenario if the cost of meta-level control is low.

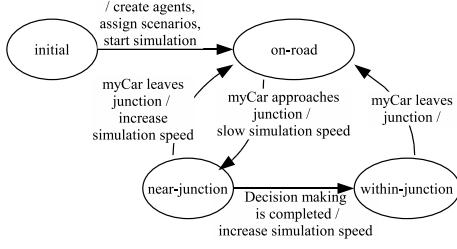


Figure 8: State Machine of Meta-Scenario for Time-Management

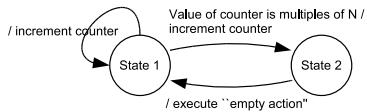


Figure 9: Scenario for Evaluation

6.1 Experiment

As explained in Section 4, the extra cost of meta-level control is mainly caused by observation/control of scenario interpreters. We created the following test to evaluate the overheads.

- The number of agents is 100.
- Each agent had a counter in its own memory.
- An agent scenario used is shown in Figure 9.
- The simulation ran for in one minute, and the sum of the counters of all the agents was measured.
- We performed two simulations. One was controlled by the meta-scenario and another was not. Then we compared them. The meta-scenario observed an “empty action”, as shown in Figure 9.

The scenario in this test was very simple. The scenarios for real applications will be more complex, and the cost of executing each cue and action will be higher. However, if a (meta-)scenario is very complex, the calculation cost of (meta-)scenario processing will not increase very much. This is because the fundamental process of a (meta-)scenario interpreter is not related to the size of the (meta-)scenario. Whereas, the cost to execute each cue or action differs among applications. The cost of the cues and actions in the test we performed was very small. This test was almost the worst case. In real applications, the ratio of the meta-level control cost to the overall cost would be smaller.

6.2 Result

Results are listed in Table 2. The number in each cell in this table means the sum of the overall counters, and the parenthetic values are the number of observations by

Table 2: Evaluation Results

Rate of observed action	Total number of actions with meta-scenario	Total number of actions without meta-scenario
1	20238(337.30)	30361
0.1	34277(57.13)	34708
0.01	33934(5.66)	35183

a meta-scenario per second. The results listed in Table 2 show that when the observation frequency is low, (in the case $N = 10$ and $N = 100$) a little decrease in performance is observed. This is because no message was sent while no observations were made a by a meta-scenario (execution of “empty action”) during our implementation. When the frequency is high ($N = 1$), some overhead was observed (performance becomes about 2/3) because the meta-scenario had to process the observation results.

These results suggest that if the number of target observations by a meta-scenario is small, the influence of the meta-scenario on performance is low. For example, only avatars are observed in participatory simulations. When applications change scenarios, many messages are sent at the time of changing. Other than at this moment, however, most of the simulation is performed with little overhead. As a result, this overhead has little influence on the simulation performance as a whole. Therefore, the architecture shown in Figure 3 is appropriate for the applications, such as participatory simulations and traffic simulations, we considered.

7 CONCLUSION

To perform traffic simulations and participatory simulations in large scale, flexible runtime control of simulations is required. We accomplished the following two goals.

1. To develop a meta-scenario description language
We defined functions required to describe observe and control simulations and scenario processing.
2. To develop meta-level control architecture
A meta-scenario interpreter in our proposed architecture controls the scenario interpreters that are separate from the agents system.

Using examples, we showed a meta-scenario can describe time management and scenario changing. We also showed this architecture provides scalability in applications, such as participatory simulations and traffic simulations.

REFERENCES

- Balmer, M., N. Cetin, K. Nagel, and B. Raney. 2004. Towards truly agent-based traffic and mobility simula-

- tions. International Joint Conference on Autonomous Agents and Multiagent Systems. 60–67.
- Balmer, M., and M. Rieser. 2004. Generating daily activity chains from origin-destination matrices. Transportation Research Board 84th Annual Meeting.
- Charypar, D., and K. Nagel. 2003. Generating complete all-day activity plans with genetic algorithms. presented at the 10th International Conference on Travel Behaviour Research.
- Gao, Z., T. Kawasoe, A. Yamamoto, and T. Ishida. 2002. Meta-level architecture for executing multi-agent scenarios. Pacific Rim International Workshop on Multi-Agent Systems. 163–177.
- Guyot, P., A. Drogoul, and C. Lemaitre. 2005. Using emergence in participatory simulations to design multi-agent systems. Proceedings of The Fourth International Joint Conference on Autonomous Agents and Multi-Agent Systems. 199–203.
- Ishida, T. 2002. Q: A scenario description language for interactive agents. IEEE Computer **35** 42–47.
- Ishida, T., L. Gasser, and H. Nakashima, eds. 2005. Massively Multi-Agent Systems I. Lecture Notes in Artificial Intelligence, 3446. Springer-Verlag.
- Murakami, Y., Y. Sugimoto, and T. Ishida. 2005 Modeling human behavior for virtual training systems. The Twentieth National Conference on Artificial Intelligence. 127–132.
- Yamamoto, G., and H. Tai. 2001. Performance evaluation of an agent server capable of hosting large numbers of agents. In: International Conference on Autonomous Agents. 363–369.

AUTHOR BIOGRAPHIES

SHOHEI YAMANE is a master student of the Department of Social Informatics in Kyoto University. In 2005 he graduated from Graduate school of Engineering in Kyoto University. His e-mail address is [⟨yamane@ai.soc.i.kyoto-u.ac.jp⟩](mailto:yamane@ai.soc.i.kyoto-u.ac.jp), and his web page is [⟨www.ai.soc.i.kyoto-u.ac.jp/~yamane/index_e.html⟩](http://www.ai.soc.i.kyoto-u.ac.jp/~yamane/index_e.html).

TORU ISHIDA is a professor of Kyoto University, IEEE fellow, and a guest professor of Shanghai Jiao Tong University. He has been working on autonomous agents and multiagent systems for more than fifteen years. He is currently working on massively multi-agent systems and semantic Web services. He initiated the Digital City Kyoto, the intercultural collaboration experiments (ICE) with Chinese, Korean, Malaysian colleagues, and the Language Grid project. He is an associate editor of Kluwer Journal on Autonomous Agents and Multi-Agent Systems and Elsevier Journal on Web Semantics. His e-mail address is [⟨ishida@i.kyoto-u.ac.jp⟩](mailto:ishida@i.kyoto-u.ac.jp), and his web page is [⟨www.ai.soc.i.kyoto-u.ac.jp/~ishida/⟩](http://www.ai.soc.i.kyoto-u.ac.jp/~ishida/).

Modeling Agents and Interactions in Agricultural Economics

Daisuke Torii

Department of Social
Informatics, Kyoto University
Kyoto, 606-8501, Japan
torii@kuis.kyoto-u.ac.jp

Toru Ishida

Department of Social
Informatics, Kyoto University
Kyoto, 606-8501, Japan
ishida@i.kyoto-u.ac.jp

Francois Bousquet

CIRAD
GREEN Research Unit,
Cirad-Tera, Montpellier, France
bousquet@cirad.fr

ABSTRACT

When multiagent simulations are used for consensus building among stakeholders, it is important not only that the domain experts can deeply understand stakeholders' actual behavior but also that the stakeholders can feel the simulation result as their solution. To this end, we propose a modeling methodology which combines several techniques with the participatory method which takes stakeholders into the modeling process using role playing games (RPG).

There are two types of model required to simulate a social system as a multiagent system: agents (internal models) and interactions. Hence, we considered a modeling method according to each character. In modeling an agent (e.g. decision making) which is implicit in human, the identification of the model greatly depends on the modeler's ability. Therefore we propose a modeling method wherein classification learning creates an alternative model from RPG log data for validating the domain experts' hypothesis. On the other hand, in modeling interactions (e.g. negotiation) which are emerged outside of human, it is rather important to show and capture continuously appeared interactions. Therefore we propose a modeling method with participatory simulation where a stakeholder participates as an avatar and agents act as the other stakeholders in order to deeply understand the stakeholders' interactions. Our methodology was effective to give the domain experts a deeper understanding through a real case study of agricultural economics in the northeast of Thailand [17].

Categories and Subject Descriptors

I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence—*Multiagent Systems*

General Terms

Design

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS'06 May 8–12 2006, Hakodate, Hokkaido, Japan.
Copyright 2006 ACM 1-59593-303-4/06/0005 ...\$5.00.

Keywords

multiagent simulation, participatory simulation, machine learning, agent interaction, modeling methodology

1. INTRODUCTION

Multiagent simulation has been used to analyze and design social systems (traffic [1], disaster management [6][11], market [14] etc.). The general modeling process is that a model created by the domain knowledge is adjusted to output the same kind of results as the real data. However, when using multiagent simulations for consensus building among stakeholders, it is important not only that the domain experts can deeply understand stakeholders' actual behavior but also that the stakeholders can feel the simulation result as their solution. In this paper, we aim to provide a modeling methodology to satisfy these points.

The participatory method by social scientists is suitable for this purpose [3][4]. The key idea is that stakeholders can understand the simulation result as their solution because they share the modeling process with the domain experts and can believe that the model reflects their actual behavior. A role playing game (RPG) where stakeholders play on a board representing their environment is used to share the modeling process and to well understand their actual thinking and behavior. "Participation" is the key aspect of our modeling methodology.

To simulate a social system as a multiagent system, two types of model are required: agents (internal models) and interactions. For example, in an economic system in agriculture described in this paper, decision making models and negotiation models of farmers and seed suppliers are required. Hence, we have to consider a modeling method according to each character.

In modeling an agent (e.g. decision making) which is implicit in human, the identification of the model greatly depends on the modeler's ability. Therefore, a method for model validation is very important. On the other hand, in modeling interactions (e.g. negotiation) which are emerged outside of human, it is rather important to show and capture continuously appeared interactions. RPG using a board in the participatory method is not enough to express such dynamic process as interactions, and it is important to give a method for this purpose.

In this study, to give solutions of each issue above, the following technological methods are applied in the notion of the participatory method.

Modeling agents with classification learning

A hypothesis of domain experts is validated by a model which classification learning creates from RPG log data. The merits are 1) classification learning creates an objective model without influence of the modeler's ability because the creation is based on the algorithm, and 2) classification learning is more robust than human in performance even when the number of elements to be compared is large. What is important here is to consider a method that classification learning creates a model which logically explains the stakeholders' behavior even if the data available is sparse (it is difficult to gather enough data for classification learning because of the cost of RPG).

Modeling interactions with participatory simulation

In participatory simulations, stakeholders participate as avatars and agents act as the other stakeholders. The merits of using this kind of simulation are 1) it is easy to express and record every interaction on a computer, and 2) domain experts can concentrate on a specific stakeholder because agents act as the other stakeholders. What is important here is to create an environment that supports the domain expert in designing agent-avatar interaction so as to elucidate the behavior of the stakeholder.

Our modeling methodology was actually evaluated in an actual case study of agricultural economics in the northeast of Thailand in co-project of IRRI¹ and CIRAD² [17].

The following section will first explain the related works. Section 3 will explain the agricultural economics used in our evaluation. Sections 4 and 5 will explain the modeling method of agents with classification learning and interactions with participatory simulations, respectively, using our example.

2. RELATED WORKS

Two studies have used machine learning with participatory simulation for modeling. 1) One applied machine learning to the logs of participatory simulations for developing diverse agent models in a virtual training system [12]. 2) The other allowed agents to learn effective negotiation strategies for the participatory simulations of a virtual market [9]. In these two studies, participatory simulation is not used to share the experience of domain experts and stakeholders. Moreover, machine learning is not used for validating a hypothesis of domain experts.

We know of two studies which use participatory simulation. 1) One developed a web forum of agents and stakeholders for holding a continuous RPG free from the constraints of time and place [4]. 2) The other used participatory simulations in which only humans participated as avatars so as to eliminate a drawback with the RPG approach in which roles of participants are fixed and emergence of new roles cannot be handled [5]. In the first study, participatory simulation is intended to create a place for continuous meeting of stakeholders and this differs from our use of participatory simulation: to exploit interaction models of stakeholders. The second also mentions design processes of participatory

simulation, but they are not for constructing an environment where agents and avatar participate.

3. AGRICULTURAL ECONOMICS OF NORTHEASTERN THAILAND

In this research, we applied our proposed processes to the agricultural economics of northeastern Thailand in a collaborative research project with IRRI and CIRAD [17].

The background is as follows. In the northeast of Thailand, the Thai government distributes a few different rice varieties. The problem is that the organization of this distribution is not always efficient: farmers do not always get the seeds they want while the seeds of some varieties are produced in excess. Also, it seems desirable that this distribution should not reduce the use of local varieties. The problem is how to develop a system that can both deliver good quality seeds and also conserve rice biodiversity. Researchers have been studying two subjects: 1) understand farmers' needs concerning rice varieties and the selection model of rice varieties and 2) to identify the problems of the current delivery system of rice seed by acquiring a farmer's selection model of seed suppliers and a flow model of rice seeds among the seed suppliers and farmers.

To realize this economic system in multiagent simulation, we need to have a) decision making model of farmers for rice varieties, b) decision making model of farmers for seed suppliers, c) negotiation model of suppliers for rice seeds and d) negotiation model of farmers for rice seeds. In this study, we will apply our modeling process of decision making and negotiation to b) and c), respectively.

4. MODELING AGENTS WITH CLASSIFICATION LEARNING

In this section, we will discuss the issues and the solutions in using classification learning to model agents (internal models) and explain the processes involved with an example: decision making of farmers for seed suppliers. The decision making model is represented in a decision tree and we selected the well-known decision tree learning algorithm "C4.5" for classification learning [13].

4.1 Issues

The characteristics as a classification learning problem are as follows:

1. The number of features tends to be large because the experts take into account all elements that they think might be related to the classification. However, the more features that are included, the more data sets tend to be needed to assure reasonable learning performance [10]. Unfortunately, it is difficult to gather enough data because the cost of RPG is high (an RPG session usually engages 12-15 players for 2 days).
2. Noise data can affect the learning result because the amount of data may be insufficient.
3. The classification knowledge gained through learning should logically explain the real behavior of stakeholders and satisfy the experts.

¹International Rice Research Institute, <http://www.irri.org/>

²Centre de Coopération Internationale en Recherche Agronomique pour le Développement, <http://www.cirad.fr/>

4.2 Approaches

In our approach for these issues, expert knowledge is effectively used for obtaining a model even from insufficient amount of data sets. Here, the unneeded features are winnowed in two ways:

Feature Selection Method

The wrapper approach³ [10] is used for eliciting a model with higher classification accuracy for unknown data; that is, a model with more generality. This method identifies a feature subset that maximizes the prediction accuracy as determined by cross validation⁴, by eliminating irrelevant features from the initial feature set through hill-climbing search. In this method, however, when the quantity of data sets is insufficient, there is a possibility that some features needed for adequately reproducing reality will be eliminated due to noise data. Therefore, in our approach, the features contained in the hypothesis (in a decision tree) are regarded as being candidates to adequately explain reality and retained in the process (e.g. five features in figure 2 are not eliminated in this process). These features are always contained in the feature subset, but the learning algorithm selects a part of features from the subset, so the important features are not always contained in the learning result. This means that the wrapper approach can winnow the features to prevent the learning from selecting irrelevant features which do not ensure generality.

Visualization Method

In order to obtain a result which logically explains the stakeholders' behavior, the learning model is refined by eliminating irrelevant features contained in it and applying the classification learning again. Such conditions are selected by the domain expert. Here, It is important to consider an evaluation method that gives domain experts intuitive understanding of the model and excludes their bias to the model. To this end, the environment and participants of the RPG are reproduced on a computer using a graphical interface and computer agents. The expert is shown the results made by the agents who have the learning model and the RPG log data itself (the expert is not told which is from the RPG log data). Then he puts comments to each result. After this, the model itself is directly shown to the expert. He selects irrelevant features, comparing his comments to the visualized result with the model.

There are three merits on this method. First, a visualized

³A method using a searching algorithm that identifies a feature subset which gives the best performance of a model outputted from an inductive learning. In many cases, cross validation is used to estimate the performance and hill-climbing is used to search a feature subset with the best estimated performance. In hill-climbing, there are two searching direction: one is backward elimination which begins at the full set of features and eliminates one feature which makes the estimated performance the best. Another is forward selection which begins at the empty set of features and adds one features which makes the estimated performance the best.

⁴Data sets are divided into n groups without overlapping. A machine learning program is applied to the n-1 groups and the result is tested by one unused group. The classification accuracy of the test data is used as criterion for the performance estimation. Switching the group used for the test data, this process is repeated n times. We assigned n to 10 which is the frequency generally used.

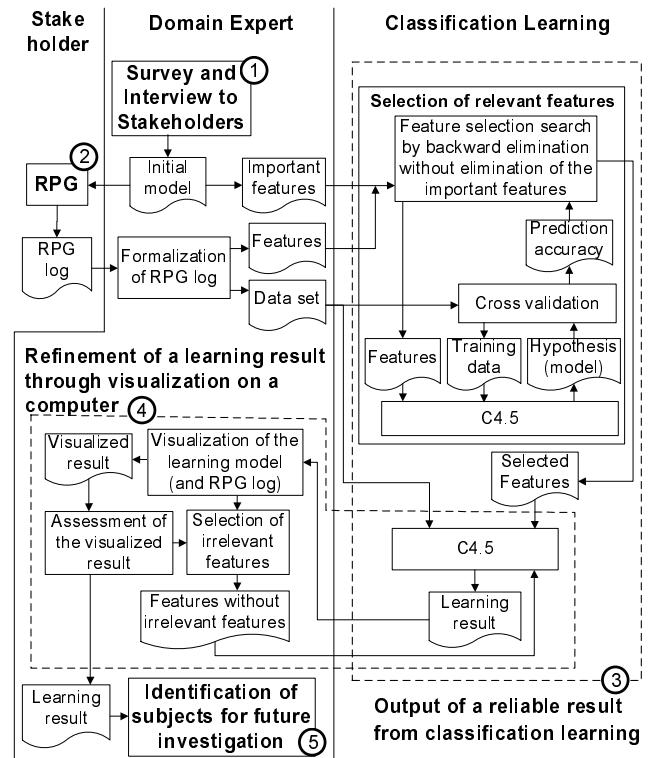


Figure 1: Modeling process for agents with classification learning

method helps the expert to find apparently strange results and eliminate irrelevant features. Second, the learning result can be neutrality evaluated, while the expert tends to criticize the learning result when the model is shown from the beginning because he is obsessed with his hypothesis. Third, he can scrutinize the learning result because a model is re-evaluated based on his comments to the visualized result. The reason why RPG log data is also shown to the expert is to convince him of validity of the learning result (he can accept a part of the learning result when the output of it resembles the one from the RPG log even if it does not support his hypothesis.).

4.3 Modeling Process: a Real Example

We constructed five steps in which the approaches above are combined (Figure 1 shows the detail of this process); 1) Survey and Interview to Stakeholders, 2) RPG, 3) Output of a reliable result from classification learning, 4) Improving the result by expert knowledge and classification learning, 5) Identification of subjects for further investigation. This process will be explained with the results gained by applying it to a real example (see [15] for the detail explanation of the decision making model).

STEP1: Survey and Interview to Stakeholders

Domain experts create an initial model (decision tree) from relevant literature and interviews with stakeholders. Important features are extracted from the initial model. In our example, five features were extracted for the feature selection method (Figure 2).

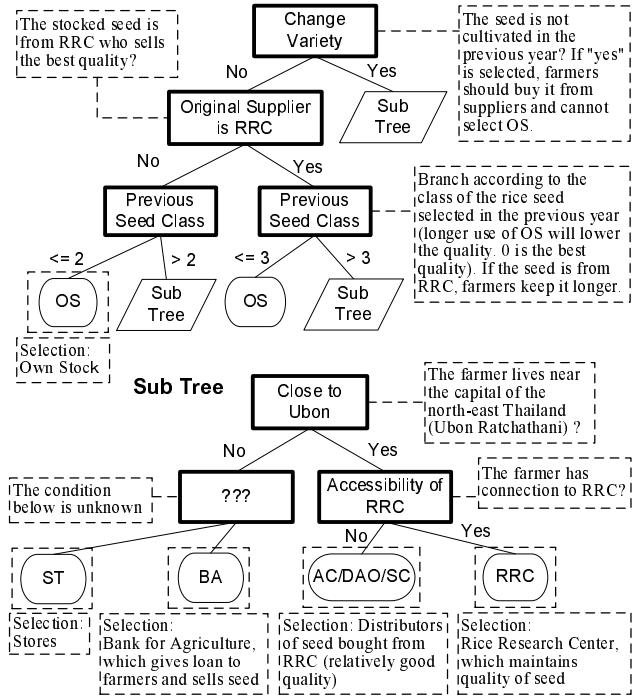


Figure 2: The initial decision tree model about farmers' selection of a rice supplier: The rectangles surrounded by bold lines are features. Five kinds of feature except “???” are the important one.

STEP2: RPG

RPG sessions using a board that represents the stakeholders' environment are organized. Log data for each stakeholder is recorded.

In our example, two RPG sessions were held. A board representing farm land was created (Figure 3 (a)). Totally, we got log data of 25 stakeholders.

STEP3: Output of a reliable result from classification learning

Learning data is created by transforming the RPG log data into a format that the learning algorithm can understand. Next, irrelevant features are eliminated by the feature selection method (mentioned in section 4.2). In this process, important features identified in STEP1 are not eliminated. Finally, the machine learning outputs a model from the feature subset and the data set.

In our example, after formatting the log data, we had 16 features and 80 data sets. Next, the feature selection method was applied. Here, five of the features identified in STEP1 were not eliminated. Finally, the feature selection method eliminated 3 features and we got the first result from the remaining 13 features.

STEP4: Refinement of the learning result through visualization on a computer

The environment and participants are reproduced on a computer using a graphical interface and computer agents. In this evaluation, the experts are shown two types of results: results from the learning model and RPG data. After this,

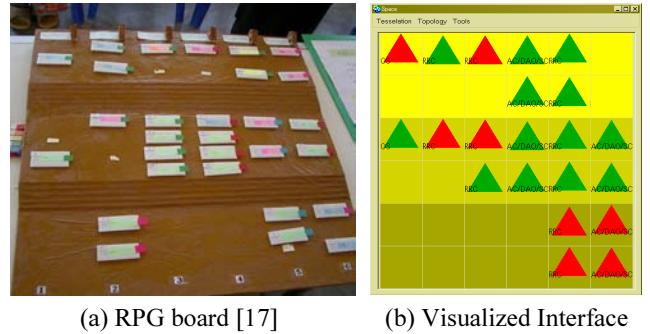


Figure 3: Interfaces for modeling: These are examples used in a case study of rice production in the northeast of Thailand. Both show upper, middle, and lower lands.

the model is directly shown to the experts. When there are irrelevant classification conditions in the model, the expert eliminates that feature which consists of the conditions and C4.5 will be applied with the resulting feature subset. This process is repeated until the expert is satisfied with the learning result.

We implemented the first learning model in CORMAS [2] (Figure 3 (b)) and evaluated the model by showing the results to the expert. At each step (equivalent to one year), the expert carefully observed the results and commented on the decision making of each agent.

After this, the first learning result was evaluated by directly showing the decision tree to the expert and an irrelevant condition was pointed by the expert (one feature was selected here and was eliminated from the feature set). The expert understood the reasons for the phenomena seen in the visualization. Classification learning was applied again with the reduced set of features. After applying the same processes in STEP4, the expert thought the learning result (Figure 4) is logical to explain the decision making of the stakeholders in RPG and was satisfied with it, then we went to the next step.

STEP5: Identification of subjects for further investigation

By comparing the experts' hypothesis (Figure 2) with the learning result, the experts investigated their hypothesis and find subjects for investigating in the future. These will be discussed among the domain experts or investigated by repeating the process from STEP1.

In our example, the expert found that some parts supported his hypothesis: a) A condition “Close to Ubon = yes” leads a branch by “Accessibility of RRC”, b) The bigger value of “Previous Seed Class” leads not OS but RRC or AC/DAO/SC. Meanwhile, other parts did not support it and the following subjects were identified: a) The value of “Previous Seed Class” (2 or 3 in the hypothesis while 0 in the result), b) The importance of two features which contains in the hypothesis but do not contain in the result (“Change Variety” and “Original Seed Supplier is RRC”). c) The possibility of the other important features (there should exist several features for the decision of OS except “Close to Ubon = No”).

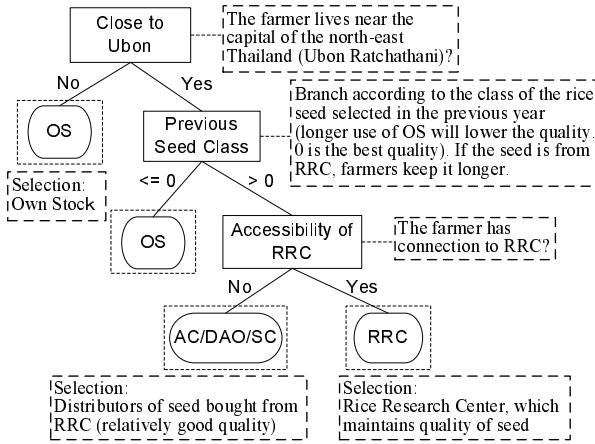


Figure 4: The final learning result

4.4 Evaluation

We can explain the effectiveness of our approach in the following way: first, the feature selection method reduced the cost of the expert refining the learning model. In our example, when the expert eliminates features without the feature selection method, it is necessary for the expert to iterate the refinement process three times, but when using the feature selection method, just one round of refinement was enough, because the feature selection method eliminated three features. In our approach, considering the preparation costs of the visualization method, it was desirable for the expert to reduce the refinement process. Moreover, we confirmed the effectiveness of our proposed method in which the features the expert thinks important are not eliminated in the feature selection method process. In our experiments, important features like these were often eliminated when this method was not used, and the experts could not explain the results as a decision making model of the stakeholders.

Second, we consider that using visualization to evaluate the learning model was useful for gaining a better understanding of the characteristics of the model and for facilitating the collaboration with the expert. The expert was able to make various comments about the decision making of agents who used the learning model because the interface had the same characteristics as the RPG. This is effective because it leads to more comments than when the learning model was shown directly to the expert, an approach that was tried in the early stage of our research.

5. MODELING INTERACTIONS WITH PARTICIPATORY SIMULATION

In this section, we will discuss the issues and the solutions in using participatory simulation to model interactions and explain the processes involved with an example: negotiation of seed suppliers. The negotiation model is represented in a state automaton and we implemented our participatory simulation in CORMAS/Q [16].

5.1 Issues and Approaches

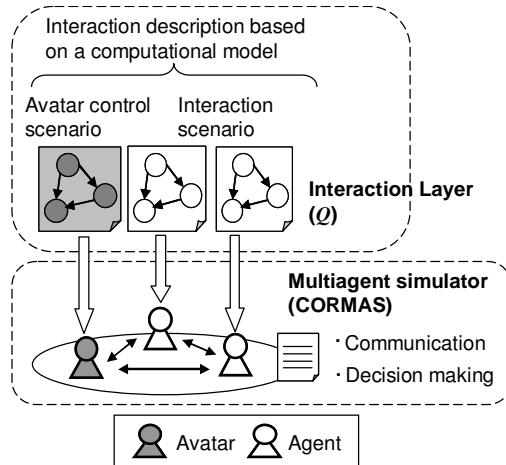


Figure 5: Agent Control from Interaction Layer

Issues

For coaxing the actual behavior from stakeholders, it is important to design how the stakeholder controls an avatar and how agents interact with the avatar. Therefore, it is indispensable to give a development environment where domain experts are able to focus on designing interactions and the interaction design is directly used in system implementation. To this end, we developed a simulation system that uses the system architecture described in the next section.

Agent Control from Interaction Layer

We use a system architecture shown in Figure 5 where an interaction layer constructed outside of a simulation system controls the agents in the system[7][16]. This architecture makes it possible to use descriptions based on a computational model that makes it convenient to write interactions. Therefore, users can focus on interaction design.

As an actual simulation system based on this architecture, we combined the scenario description language Q [7] with the multiagent simulator CORMAS [2] which is often used in research based on the participatory method [3] (the system is called CORMAS/ Q [16]). The computational model behind a Q scenario is an extended finite state automaton, which is commonly used for describing communication protocols. By using Q , users can directly create scenario descriptions from extended finite state automata. In Q scenarios, we can use sensing functions (cues) and action functions (actions). Scenarios are interpreted by Q interpreter, while cues and actions are executed by CORMAS. Cues/actions become the interface between domain experts and system developers, which enables collaboration in the design of interactions.

5.2 Interaction Description

For realizing participatory simulations where agents and avatars interact, it is necessary to assume the three kinds of interaction descriptions shown in Table 1. These descriptions are designed according to the modeling process shown in the next section. Interaction protocols are very important because they define the activity of avatars (stakeholders) in the simulation world. An interaction scenario regulates an agent's behavior about interactions (e.g. a negotiation

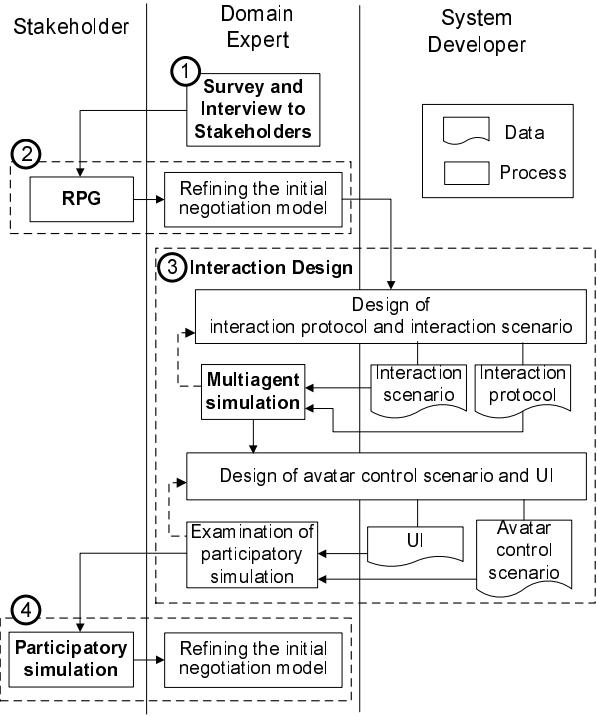


Figure 6: Modeling process for interactions with participatory simulation

model for agents who access the other agents and avatars). The avatar control scenario is an extended scenario of the interaction protocol. There are four elements in this scenario: 1) control of user interfaces (UI) (display etc.), 2) observation of input by a user through UI, 3) sensing from other agents, 4) action with other agents; this is triggered by input to the UI by a user.

Table 1: Interaction Descriptions

Classification	Definition of interaction description
Interaction protocol	Procedures which should be abided to negotiate with another agent or avatar
Interaction scenario (negotiation model)	Interaction procedures (e.g. access priority of trading or condition for completion of the deal) are described within the framework of the interaction protocol
Avatar control scenario	Control procedures of UI and an avatar along the interaction protocol

5.3 Modeling Process: a Real Example

The following four steps aim to refine an interaction model of a stakeholder and show a process to design three interaction descriptions shown in Table 1; 1) Survey and Interview to Stakeholders, 2) RPG, 3) Interaction Design, 4) Participatory Simulation (see Figure 6).

In this real example, we aimed to refine a negotiation model of *SC* (*Seed Center*) who is one of the important suppliers in northeastern Thailand. There are four types

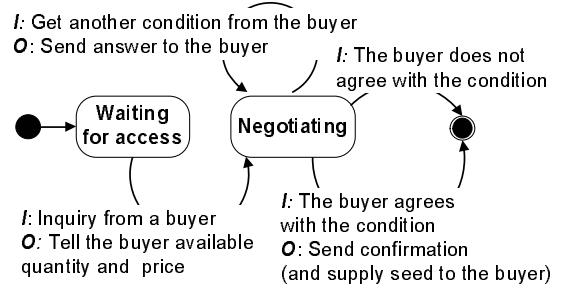


Figure 7: Example of interaction protocol (for sellers): Conditions are exchanged between the seller and the buyer in enough times to agree and the negotiation is settled if the buyer agrees/disagrees with the condition.

of supplier except *SC* and they are realized as agents in the participatory simulation (they are shown as *A*, *B*, *C* and *D* for descriptive purpose.).

STEP1: Survey and Interview to Stakeholders

Domain experts create an initial model from relevant literature and interviews to stakeholders. (In our example, the initial model was very abstract which just specified a contact list of each supplier).

STEP2: RPG

RPG based on STEP1 is held with stakeholders. In the process, RPG is very important because RPG is a good place for communication between domain experts and stakeholders unfamiliar with the research process and sometimes unfamiliar with computer systems. Through RPG, behavioral model and the reason behind the behavior are exposed. The initial model is improved by knowledge acquired through RPG.

The refined model of *SC* in our example is that rice seed is supplied according to a priority order (*A*, *B*, *C* and *D*)

STEP3: Interaction Design

An interaction protocol and interaction scenarios are described based on observation in RPG. What is important here is that several kinds of interaction scenarios are created for each agent to observe reaction of a stakeholder in participatory simulation. These descriptions are designed by a computational model of the interaction description through collaboration of a domain expert and a system developer. Finally, these descriptions are implemented in a computational language and tested through several runs of multiagent simulation. The processes of design and simulation are repeated until a domain expert is satisfied.

Next, UI and an avatar control scenario are designed by the domain expert and system developer for participatory simulation. In the avatar control scenario, the control procedures of UI are considered based on the interaction protocol. Next, UI and the avatar control scenario are implemented by the system developer and the participatory simulation system is tested by the domain expert. The processes above are repeated until the domain expert is satisfied.

In our example, through discussion between a domain expert and a system developer, an interaction protocol and

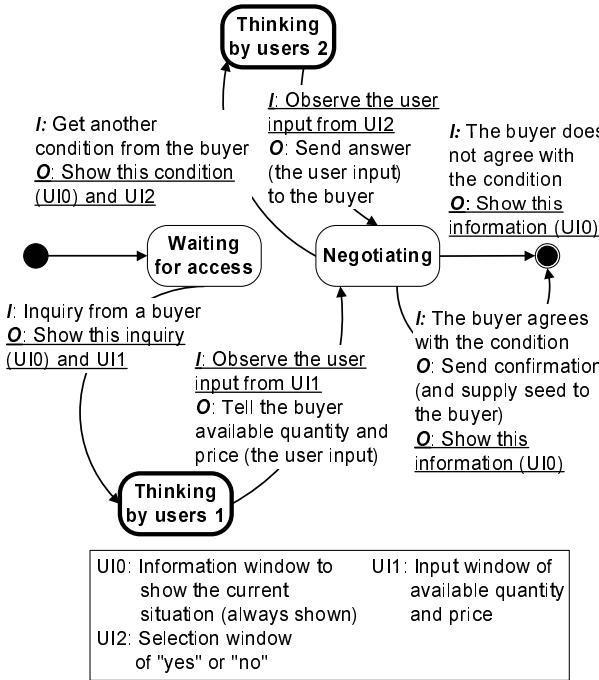


Figure 8: Example of avatar control scenario (for *Seed Center*): the interaction protocol shown in figure 7 is extended by putting controls concerning to UI (states surrounded by a bold line and conditions/actions with underlines)

interaction scenarios for each agent were designed by a state transition model which is a computational model of Q . The interaction protocol is shown in Figure 7. Three kinds of interaction scenarios are also prepared: a) access in a fixed order, b) access in a cheap price order and c) access in a short distance order. Next, the domain expert and system developer defined cues/actions of Q and the system developer implemented corresponding behavior of the cues/actions in CORMAS (this part of design process are described in [11]).

UI for our participatory simulation was not one big panel with all functions but small panels with only necessary functions at each scene, which are opened and closed according to the avatar control scenario. As in the interaction scenario, the avatar control scenario was designed by state transition diagram (Figure 8). Then, cues/actions are designed and corresponded to the control of UI or avatar behavior in CORMAS.

STEP4: Participatory Simulation

Participatory simulation with stakeholder is held. The domain expert refines the interaction scenario by analyzing the log data and knowledge acquired through conversation with the stakeholder.

In our example, the refining process of the model is as follows. The result after several times of participatory simulation was that supplier B was always selected first then the selection of A , C and D was not fixed. This was observed every time, though we changed the interaction scenario given to the agents in every run of simulation. After the domain expert recognizes this phenomenon, he tried one

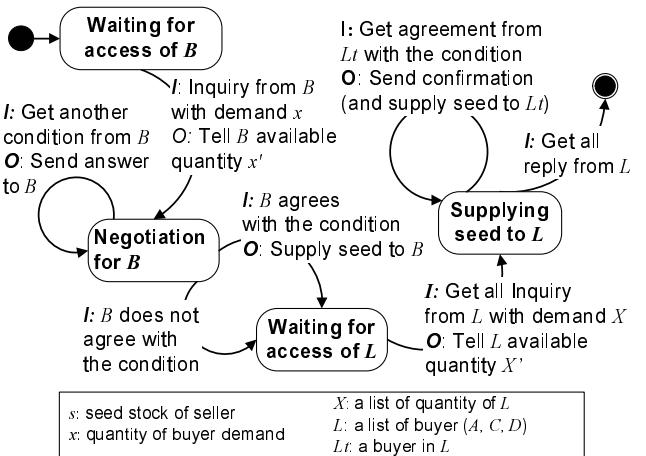


Figure 9: Refined negotiation model of *Seed Center* (*SC*) after Participatory Simulation: B is first chosen then A , C and D are equally distributed. If the stock remains, the other buyers will be supplied.

more around of simulation while he gave questions to the stakeholder. From the conversation with the stakeholder, the domain expert could refine the model as shown in Figure 9.

5.4 Discussion

We confirmed the merits to use participatory simulation. 1) The domain expert could focus on observing specific stakeholders (only *SC* in our example) because agents took roles instead of the stakeholders in RPG and a computer program automatically conducts the simulation. 2) The domain expert could give questions according to the progress of simulation. The stakeholder could monitor the simulation progress and see the impact of his own decision, which enabled him to give more concrete answer.

We consider that this participatory simulation operated as intended. This is because the system architecture could give the environment for the domain expert to focus on interaction descriptions and our proposed processes of designing three kinds of interaction descriptions were effective to reflect his knowledge acquired through surveys and RPG to the simulation world.

6. CONCLUSION

In this study, we proposed a novel methodology of modeling agents (decision making) and interactions (negotiation) in multiagent simulations for consensus building among stakeholders. This methodology combines several techniques with the participatory method, which takes stakeholders into the modeling process. We tackled the following two problems.

Modeling agents with classification learning

A modeling method wherein classification learning is applied to RPG log data was established for validation of domain experts' hypothesis (agent's internal model). The key ideas are 1) a feature selection method for enhancing reliability of the learning result by classification learning, and 2) visualization of the learning result on a computer to promote

understanding of domain experts in refining the learning result. The modeling process with these methods was applied to farmers' decision model of seed suppliers. As a result, the domain expert's hypothesis was validated and several subjects for further investigation were found.

Modeling interactions with participatory simulation

A modeling method with participatory simulation was established for deep understanding of stakeholders' interactions (e.g. negotiation). The key ideas are 1) a system architecture where interaction layer controls agents gives an environment for the domain expert to focus on interaction descriptions, and 2) three kinds of interaction descriptions (interaction protocol, interaction scenario, avatar control scenario) are defined and designed through the modeling process. Actually, a participatory simulation was developed on CORMAS/Q [16] and the domain expert could get deeper understanding of a negotiation model among seed suppliers.

Our modeling methodology has the following two important features for multiagent simulations to build consensus among stakeholders. First, RPG and participatory simulation give shared experience of modeling with domain experts, which is important for stakeholders to find their solutions in the simulation results. Second, the modeling method with classification learning and participatory simulation enables domain experts to deeply understand and model stakeholders' actual behavior in the system. Indeed, the impact was confirmed in the real case study of agricultural economics in the northeast of Thailand [17].

In this study, we could use a decision tree and a state automaton to represent decision making and negotiation, respectively. In the other cases, however, we might find or develop suitable models, the learning algorithms, systems providing the interaction description. This will be the future works, which also give new seeds of artificial intelligence and multiagent research. Another future work is use of new technologies for user participation. For example, *Augmented experiment* [8] can give an environment where stakeholders take part in the simulation world from their real work space, which enables domain experts to understand more actual behavior of stakeholders.

7. ACKNOWLEDGMENTS

This research was supported by the young researcher development program of the JSPS 21st century COE program "Informatics Research Center for Development of Knowledge Society Infrastructure". We would like to thank Dr. Guy Trébuil and Dr. Chirawat Vejpas who gave valuable comments as a domain expert, and farmers and seed suppliers who joined our experiment in Ubon Ratchathani.

8. REFERENCES

- [1] M. Balmer, N. Cetin, K. Nagel, and B. Raney. Towards truly agent-based traffic and mobility simulations. In *Proc. Int. Joint Conf. Autonomous Agents and Multiagent Systems (AAMAS-04)*, pages 60–67, 2004.
- [2] F. Bousquet, I. Bakam, H. Proton, and C. L. Page. Cormas: common-pool resources and multi-agent systems. volume 1416 of *LNAI*, pages 826–838. Springer-Verlag, 1998.
- [3] F. Bousquet, O. Barreteau, P. d'Aquino, M. Etienne, S. Boissau, S. Aubert, C. L. Page, D. Babin, and J. Castella. Multi-agent systems and role games: an approach for ecosystem co-management. In *Complexity and Ecosystem Management: The Theory and Practice of Multi-agent Approaches*, pages 248–285. Edward Elgar Publishers, 2002.
- [4] N. Gilbert, S. Maltby, and T. Asakawa. Participatory simulations for developing scenarios in environmental resource management. In *Proc. Third Workshop on Agent-Based Simulation*, pages 67–72, 2002.
- [5] P. Guyot, A. Drogoul, and C. Lemaitre. Using emergence in participatory simulations to design multiagent systems. In *Proc. Int. Joint Conf. Autonomous Agents and Multiagent Systems (AAMAS-05)*, pages 199–203, 2005.
- [6] D. Helbing, I. Farkas, and T. Vicsek. Simulating dynamical features of escape panic. *Nature*, 407(6803):487–490, 2000.
- [7] T. Ishida. Q: A scenario description language for interactive agents. *IEEE Computer*, 35(10):42–47, 2002.
- [8] T. Ishida. Multiagent simulation meets the real world. In *Proc. Int. Joint Conf. Autonomous Agents and Multiagent Systems (AAMAS-06)*, 2006.
- [9] H. Kita, H. Sato, N. Mori, and I. Ono. U-mart system, software for open experiments of artificial market. In *Proc. of IEEE Int. Symposium on Computational Intelligence in Robotics and Automation*, pages 1328–1333, 2003.
- [10] R. Kohavi and G. H. John. Wrappers for feature selection. *Artificial Intelligence*, 97:273–324, 1997.
- [11] Y. Murakami, T. Ishida, T. Kawasoe, and R. Hishiyama. Scenario description for multi-agent simulation. In *Proc. Int. Joint Conf. Autonomous Agents and Multiagent Systems (AAMAS-03)*, pages 369–376, 2003.
- [12] Y. Murakami, Y. Sugimoto, and T. Ishida. Modeling human behavior for virtual training systems. In *Proc. National Conf. Artificial Intelligence (AAAI-05)*, pages 127–132, 2005.
- [13] J. R. Quinlan. Induction for decision trees. *Machine Learning*, 1(1):81–106, 1986.
- [14] L. B. Said, T. Bouron, and A. Drogoul. Agent-based interaction analysis of consumer behavior. In *Proc. Int. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS-02)*, pages 184–190, 2002.
- [15] D. Torii, F. Bousquet, T. Ishida, G. Trébuil, and C. Vejpas. Using classification learning in companion modeling. In *Proc. of the 8th Pacific Rim Int. Workshop on Multi-Agents (PRIMA-05)*, pages 327–341, 2005.
- [16] D. Torii, T. Ishida, S. Bonneaud, and A. Drogoul. Layering social interaction scenarios on environmental simulation. In *Multi-Agent and Multi-Agent-Based Simulation*, volume 3415 of *LNCS*, pages 78–88. Springer-Verlag, 2005.
- [17] C. Vejpas, F. Bousquet, W. Naivinit, G. Trébuil, and N. Srisombat. Participatory modeling for managing rainfed lowland rice variety and seed systems in lower northeast thailand. In *Mekong Rice Conference*, 2004.

インタラクションを投影するデジタルシティ

中西英之 [大阪大学准教授] 石田亨 [京都大学教授]

010

Special Feature

デジタルシティプロジェクト

デジタルシティとは実在する都市のメタファーに基づいて、地域情報空間を設計する試みである^[1]。われわれは、NTTコミュニケーション科学基礎研究所オープンラボ(1998-2000)、デジタルシティ京都実験フォーラム(1999-2001)において、デジタルシティ京都の構築を試みている。

デジタルシティ京都の活動を始めた当時は、インターネットは世界に向けて情報発信するためのメディアであるというのが一般的の認識であった。これに対しデジタルシティ京都は、市民と訪問者のインタラクション(相互作用)の場となることを目的とした^[2]。地域に関するウェブサイトを半自動的に抽出し、位置を割り出し、地図上で閲覧可能とすることにより、実際の都市の活動を地図上に描き出そうとした。京都の場合、観光スポットは山際に、商店は都市の中央に集中している様子が生き生きと浮かび上がった。さらに、京都市中心部を模した3次元仮想都市を地元商店街と共に開発し、インターネットを通じて訪問客を呼び込むシステムを作りあげた。このようにデジタルシティ京都は、都市の活動を描き出し、市民と訪問客のインタラクションの場となることを目標とした研究活動であった。

デジタルシティは世界各地でさまざまに試みられたが、やがて大規模な商用ポータルに押され始める。デジタルシティ京都も例外ではなく、市民参加の都市のポータルという当初の目標は断念せざるを得なくなつた。そのため、科学技術振興機構CRESTデジタルシティプロジェクト(2000-2005)は、仮想都市の新たな利用を目指すものとなつた。当時は、コンピュータ上で都市空間を再現するといえば、グラフィックスのリアルさの追求が主な関心事であった。それに対してわれわれの仮想都市空間は、インタラクションのリアルさを追求した。すなわち、都市空間のなかに置かれた人間集団の行動を、それを生み出

す集団内のインタラクションを含めて、実在する都市空間を投影したVR(Virtual Reality)空間のなかに再現しようとした。

インタラクションを投影する技術

仮想都市シミュレータの実現に向けてわれわれは、社会的エージェントという技術を開発した^[3]。エージェントとは自律的に環境のなかで行動する主体を表す抽象概念であり、社会的エージェントとは、VR空間のなかで自律的に活動するキャラクター(仮想人間)である。

人間集団の空間的な行動をモデル化する方法としては、抽象度の高いものから順に、流体モデル、粒子(particle)モデル、エージェントモデルなどがある。流体モデルでは人間集団の行動を、気体や液体の流れのような物理現象としてシミュレートする。粒子モデルも同様に人間集団の行動を物理現象としてシミュレートするが、人間集団をひとまとめにしてその行動を流体の流れとして計算するのではなく、個々の人間の行動をひとつの粒子の動きとして計算する。これによって、出入口に対して扇状に群がる行動などが再現可能となる。

エージェントモデルでは個々の人間は粒子のように物理法則に従って動くのではなく、人間が行う状況判断を模擬するさまざまなルールに従って行動する。これによって、道に迷ったときに自分の近くにいる他者について行く、といったような内部状態に依存した行動の再現が可能となる。われわれはこのようなエージェントよりもさらに抽象度が低い社会的エージェントを用いて、シミュレーションをVRとして人間のユーザーが体験できるようにした。

通常のエージェントとは異なる社会的エージェントの特徴のひとつは、アニメーションを用いて自分の行動を視覚的にユーザーに提示する点である^[図1]。これによってエージェントとユーザーの非言語(nonverbal)コミュニケーション

1973年生まれ/京都大学卒業/同大学院修了/博士(情報学)

2002年情報処理学会坂井記念特別賞、2004年テレコムシステム技術賞、2006年科学技術分野の文部科学大臣表彰科学技術賞受賞

いしだ・とおる

自律エージェントとマルチエージェントシステム研究における貢献でIEEE、情報処理学会フェロー/1998年京都大学で社会情報学専攻の設立に参加、

2002年にはInternational Conference on Autonomous Agents and Multiagent Systems(AAMAS)の第一回大会委員長を務める

デジタルシティ、言語グリッド、異文化コラボレーションなどの社会との接点での研究活動を展開。

ションが可能となる。例えば、ユーザーが操作するキャラクター(アバターと呼ばれる)に社会的エージェントが近づいた場合、それは仮想的な対人距離(interpersonal distance)の変化として提示される。すなわち、アバターの視界を3次元グラフィックスとして表示しているコンピュータの画面に、手前に向かって接近してくるキャラクターのアニメーションが表示される。この社会的エージェントの行動はユーザーの行動に影響を与え、さらにそのユーザーの行動が社会的エージェントの行動に影響を与えるというループが生じる。このようなHuman-In-the-Loop型のシミュレーションは参加型(participatory)シミュレーションと呼ばれる。社会的エージェントは、キャラクターのアニメーションを用いて非言語コミュニケーションにおけるアリアリティを伴った参加型シミュレーションを可能にする技術といえる。社会的エージェントと通常のエージェントには、キャラクターアニメーションを用いるという行動表現の違いがあるだけではなく、行動制御にも重要な違いがある。ユーザーを含めずエージェントのみを用いるシミュレーションの場合、行動制御の機構はひとつだけでよい。

これに対して参加型シミュレーションの場合は、エージェント用の制御機構とアバター用の制御機構が必要となる。オンラインゲームのような仮想空間の場合はユーザーが操作するキャラクターとコンピュータが操作するキャラクターはあらかじめ決まっている。例えば、モンスターのキャラクターをユーザーが操作することはない。よって、各キャラクターはどちらかの制御機構で動かせばよい。

これとは異なり、仮想都市シミュレータではユーザーが自由に操作するキャラクターを選択できるようにする必要があった。そこですべてのキャラクターを、キーボードとマウスを使って操作することも、行動ルールを与えて自律的に行動させることも可能にする制御機構を実装した。

避難訓練への応用

仮想都市シミュレータの応用先として、われわれがまず思いついたのは公共施設における避難訓練であった。利用者が限定されている学校やオフィスとは異なり、駅のような公共施設では利用者が集まって避難訓練

図1|社会的エージェントを用いたシミュレーションの例



を実施するということは不可能に近い。そこで、京都駅を模した3次元仮想都市を構築し[図2]、そのなかで参加型の避難シミュレーションを行った。

避難シミュレーションの構築にあたっては、エージェントの設計が大きな課題であった。エージェントの行動は流体モデルなどに比べるとずっと複雑であるため、シミュレーションの妥当性が問題となる。そこで、実際に地下室のなかで行われた避難訓練の実験[4]における避難者役の被験者の行動が再現されるようエージェントを設計した。具体的には、周囲にいる他者(他のエージェントもしくはアバター)の行動を真似る行動ルールを導入しないと避難行動を再現できないことがわかった。

また、仮想都市シミュレータ上で参加型避難シミュレーションを実施し、その際にユーザーがアバターを操作した履歴からエージェントの行動ルールを抽出する研究を行った。避難実験中、ユーザーの視界を映し出している画面を録画し、避難実験後のインタビューでそれを被験者に見せながら、どのようなルールでアバターを操作したのか聞き出した。そして、シミュレーション中に発生する多様な状況においてエージェントがどの行動ルールを使うのかを決める選択ルールを、仮説推論という分析方法によって抽出した[5]。

参加型避難シミュレーションを何度も行ってみた結果、避難訓練への応用にあたってはVR以外の体験方法も併用すると訓練の効果

が増すことがわかった。そのひとつは、避難の様子を俯瞰的に眺める非没入型の体験方法の併用である。訓練効果を調べる実験を行ったところ、アバターの一人称視点で避難シミュレーションを体験するだけの場合と、その前にまず避難シミュレーションを仮想空間の上方に配置された鳥瞰視点から眺める場合とでは、後者のグループの訓練効果の方が有意に高いという結果を得た。

もうひとつの併用すべき体験方法はAR(Augmented Reality)である。仮想京都駅におけるVR体験と、本物の京都駅におけるAR体験を比較する実験を行った。AR体験では、被験者の周囲が他の避難者(エージェント)でどれくらい混雑しているのかという情報を、仮想都市シミュレータが被験者の所持する携帯電話にメールで送信した。前進できないほどの混雑であることをメールが告げた場合には、前進可能になったことを告げるメールを受け取るまで立ち止まらなければならない、という条件のもとで実際に京都駅のなかで被験者をゴール地点に向かわせた。このような身体性を伴う体験はアバターを操作する非身体的な体験に比べて、前に進めない間の不安感などをよりリアルに与えることができた。

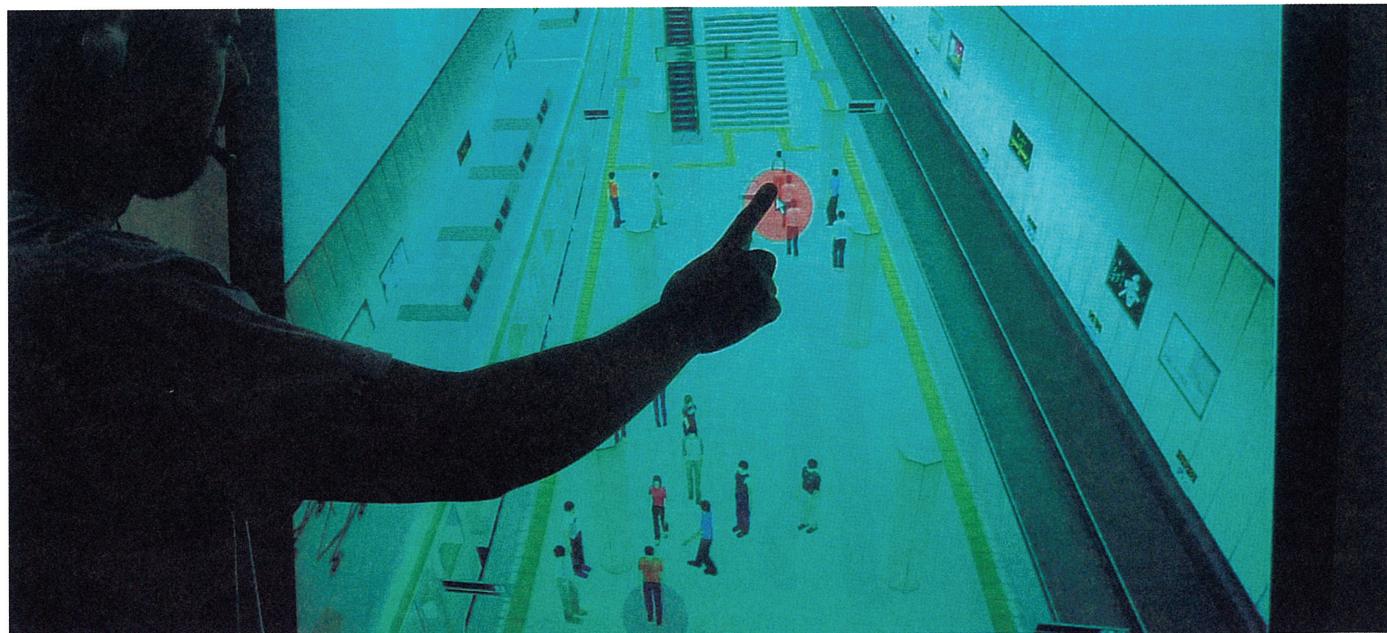
避難誘導への応用

仮想都市シミュレータを避難訓練に用いる場合、避難群集の行動はあらかじめ仮想都市に

図2|京都駅を模した3次元仮想都市を構築



図3|仮想都市シミュレータを避難訓練に用いた例



埋め込まれており、これはオフラインのインタラクション投影であるといえる。それに対して以下に説明する避難誘導への応用は、オンラインのインタラクション投影といえる。

現在、一般的に駅などで避難誘導に用いられる設備は監視カメラと構内放送用スピーカーである。管制室にいる誘導者は現場の状況をおおまかにしかとらえることができず、また全体に対して一斉に同じ情報を流すことしかできない。そのため、細かい誘導は現場にいる係員に頼ることになる。

そこで、われわれは仮想都市シミュレータを拡張して、管制室からの細かい誘導を可能にするシステムを開発した。この誘導システムでは誘導者が眺めている3次元仮想都市のなかに、現場の避難者の様子がアニメーションで表示される[図3]。もし、避難者が持っている携帯電話の番号が改札通過時のICカード読み取りなどを介してシステムに登録されていれば、その避難者に対応するエージェントをポインティング操作で指定して、メールを送ったり音声通話を開始したりすることができるようになっている。

このように実世界にいる人々に対して外側からのアクセスを可能にするユーザーインターフェースのことを、われわれは超越インターフェースと呼んでいる。このシステムのプロトタイプを2005年に京都駅の地下鉄駅舎に実装した。避難者の現在位置を検出するセンサーとしてカメラ28台を天井に取りつけた[写真]。

監視カメラの映像と比較して、仮想都市シミュレータの描画する仮想の避難現場の方が優れている点がいくつかある。第一に、現場の映像をインタラクティブにすることができます。監視カメラの映像はインタラクティブではなく、そこに映っている人や物に直接アクセスすることはできない。第二に、自由な視点から現場を眺めることができる。監視カメラの設置位置には物理的な制約がある。例えば、プラットホーム全体を眺められる視点の位置は、実際には天井よりもずっと上方であ

る場合がほとんどである。第三に、断片的な現場の情報をひとつのグラフィックスに統合して可視化することができる。通常、管制室には監視カメラの映像を映し出すためのモニターが数多く並んでいる。仮想都市シミュレータのシミュレーション機能が最も活かされるのはこの第三の点においてである。断片的なセンサー情報を統合することだけでなく、センサーで得られる不完全な情報を補完するのにも用いることができる。例えば、ある避難者の現在位置がセンサーで検出できなくなったときに、それまでの移動履歴からそれ以後の避難経路をシミュレートし、システムの画面上では引き続き表示し続ける、などの使い方である。

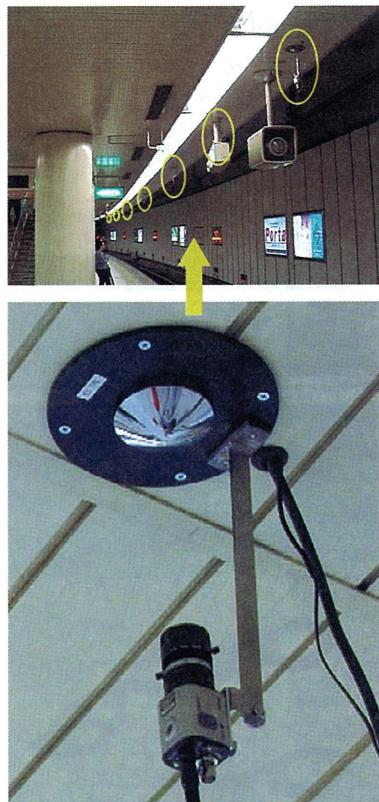
今後の展望

上記の超越インターフェースは実世界での人々のインタラクションをセンサーでとらえ、仮想空間のエージェント群に投影し、モニター上に可視化するものである。これによって観察者は、超越的な視点でモニターを通じて実世界の人々とコミュニケーションができる。内在的な視点に終始したこれまでの自律エージェント研究とは異なる発想である。

地下鉄京都駅の実験では要素技術としてカメラや3次元仮想空間を用いたが、他の要素技術を用いても超越インターフェースを実現することはできる。実際に大学院生を中心となり、GPSと2次元地図を用いた屋外での避難誘導実験を2006年に実施している。

技術的な課題はエージェントのモデリングと、100万エージェント単位の大規模なシミュレーション技術の開発である。とくに前者は参加型シミュレーションの基本技術となるものであり、被験者の行動ログやインタビュー結果から被験者の意思決定モデルを学習する必要がある。ゲーミング、機械学習などさまざまな手法を駆使したモデリング手法の確立が求められている。

避難者の現在位置を検出するセンサー



参考文献

- 1 石田亭/「デジタルシティの現状」/『情報処理』/Vol.41/No.2/pp.163-168/2000
- 2 安岡美佳、石田亭、Alessandro Aurigi/「ワールドデジタルシティ:社会を映す鏡」/『社会情報学会誌』/Vol.17/No.1/pp.25-38/2005
- 3 中西英之、小泉智史、石黒浩、石田亭/「市民参加による避難シミュレーションに向けて」/『人工知能学会誌』/Vol.18/No.6/pp.643-648/2003
- 4 杉万俊夫/「避難誘導法のアクションリサーチ」/『応用心理学講座 第3巻 自然災害の行動科学』/pp.110-122/福村出版/1988
- 5 石田亭、鳥居大祐、村上陽平、寺野隆雄/「社会シミュレーションと参加型デザイン」/『情報処理』/Vol.48/No.3/pp.271-277/2007