

Q と CORMAS の結合：多層マルチエージェントシミュレーションへの挑戦

Connecting Q and CORMAS: Toward Multi-Layer Multiagent Simulation

鳥居 大祐
Daisuke Torii

石田 亨
Toru Ishida

京都大学 社会情報学専攻
Department of Social Informatics, Kyoto University

For an integrated simulation such as natural environment affected by human society, it is indispensable to provide an integrated simulator that incorporates multiple computational models. We proposed a multi-layer socio-environmental simulation by layering social interaction scenario on environmental simulation. For this simulation, we connect two generic simulators. One is scenario description language *Q*, which is suitable for describing social interactions. Another is CORMAS, which models interactions between a natural environment and humans. The key idea is to realize a mapping between agents in different simulators. This integration becomes possible by the salient feature of *Q*: users can write scenarios for controlling legacy agents in other systems. Moreover, we find that controlling the flow of information between the two systems can create various types of simulations. We also confirm the capability of CORMAS/*Q*, in the well-known Fire-Fighter domain.

1. はじめに

地球の温暖化、酸性雨、森林火災、地震災害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。例えば、人と環境を含めた総合的な *Socio-Environmental Simulation* 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。MABS のためのシミュレータの中には、エージェントのモデリングに汎用性を保つため、特定の計算モデルを持たせないものもある。例えば、*Socio-Environmental Simulation* のためのシミュレータとして代表的なものに CORMAS [Bousquet 98] が挙げられる。CORMAS では、エージェントの社会的インタラクションを実現するために、一からモデルの構築を行う必要がある。

このような問題への対策の流れとしては二つ挙げられる。一つは、計算モデルの結合である。これは Doking [Axtell 96] と呼ばれ、二つの計算モデルを比較し妥当性を得る目的と結合された計算モデルの可能性を探る目的がある。もう一つは、マルチレイヤーアーキテクチャの構築である。特徴の異なる二つのシミュレータを結合して、目的に合わせたシミュレーションを柔軟に実現するものである。例えば、ロボカップシミュレータでは、サッカーサーバに対し、様々なエージェントモデルを結合するプロトコルが提供されている[Noda 01]。また、*Q* [Ishida 02b] は、既存エージェントに対し外付けで結合でき、エージェントの社会インタラクションを設計するための言語である。*Q* は、計算機の非専門家向けの記述言語であり、ドメインに依存せずあらゆるエージェントシステムに結合できる汎用性を持っている。これまでに、FreeWalk や Microsoft Agents と結合の実績がある[Ishida 02a, Murakami 03, Nakanishi 02]。

本研究では、実際に、自然環境と人間とのインタラクションの記述に適したマルチエージェントシミュレータ CORMAS に対し、社会的インタラクションの記述に適したマルチエージェントシナリオ記述言語 *Q* を結合し、多層 *Socio-Environmental Simulation* を構築する。CORMAS は、自然資源の管理のシミュレーションに実績がある[Bah 03, Perez 02]。自然環境のシミュレーションをセルラオートマトンモデルで実現し、エージェントは自然環境の状態に応じた行動を定義することができる。一方、シナ

リオ記述言語 *Q* では、複雑なプランやエージェント間のインタラクションを拡張状態遷移機械モデルで記述できる。*Q* は社会的インタラクションのシミュレーションに実績がある[Murakami 03, Nakanishi 02]。

本結合は、二つの計算モデルの結合であると共に、二つの汎用性を持った既存シミュレータの結合によるマルチレイヤーアーキテクチャの構築でもある。即ち、セルラオートマトンモデルと拡張状態遷移機械モデルの結合だけでなく、例えば、CORMAS 上で *Q* の特徴である IPC (表1参照) の利用や参加型シミュレーションを実現できる。本結合の実現には、レガシーなエージェントのシナリオが記述できるという *Q* の特徴が生かされている。本研究では、さらに、接続されたシステム間を流れる情報を Connection Description で制御することにより、両シミュレータの内部を変更することなく、柔軟にシミュレーションを操作できる手法を提案する。また、両者の接続には、以下の2つの課題を解決しなければならない。

機能分担

両シミュレータ内のエージェント間の対応付けとそれぞれの役割を定義する必要がある。CORMAS/*Q* の場合には、同一エージェントの CORMAS 側で自然環境の観察と行動を実行し、*Q* 側でエージェント間のインタラクションのシナリオを記述する。

時間管理

両シミュレータの内で進行する時間の整合性をとる必要がある。CORMAS/*Q* の場合には、CORMAS で時間を管理し、*Q* では一切の管理を行わない。

以下、本論文では CORMAS/*Q* を説明し、実際に行った Cohen らによる森林火災のシミュレーション[Cohen 89]を例に、多層マルチエージェントシミュレーションの有効性を確認する。

2. 多層マルチエージェントシミュレータ

2.1 アーキテクチャ

多層 *Socio-Environmental Simulation* は、環境シミュレーションに社会シミュレーションを結合することで実現される。このシミュレータは、社会シミュレーションと環境シミュレーションの二層より構成される。社会シミュレーションは、意志決定、交渉、協調過程のシミュレーションである。環境シミュレーションは、環境を観察し、環境に作用する工

エージェントと、環境内での変動の伝播のシミュレーションである。

社会的インタラクションと、環境における変動の伝播は以下のように実行される。まず、社会的インタラクション(例えばグループ意思決定)のシナリオは、社会シミュレーションの層で実行されるが、メッセージングなど実際の行為は、環境シミュレーションの側で対応するエージェントによって実行される。これは、現実世界における行為をすべて、環境シミュレーション側に集約し、両シミュレーションの同期の問題を解決するためである。即ち、社会シミュレーションの実行が、環境シミュレーションに比べ十分に速ければ、時間の管理は環境シミュレーションの側だけで行えばよい。

一方、環境における変動の波及は環境シミュレーション側で計算されるが、エージェントの制御は社会シミュレーションのシナリオに従う。例えば、エージェントがシナリオに従って自然環境に対し何らかの行為を実行すると、その影響の波及が計算される。エージェントはその結果を、シナリオに従って観察する。

2.2 Connection Control

異なるレイヤー間の結合箇所を流れる情報を制御することにより、既存のシミュレータに手を加えなくとも、多様なシミュレーションを実現できる。例えば、次の3種の情報が結合部分を流れる。

- a) 環境の観察
- b) 環境への行為
- c) 他のエージェントとのコミュニケーション

上記の情報に対し、結合部分で以下の操作が可能である。

- i) 情報を伝達しない (accuracy)
- ii) 情報の伝達を遅らせる (delay)

この操作を、シミュレーション利用者の視点から、organizational descriptionとして以下のように指定させる。

Communication Channel

組織内の情報伝達の正確性、速報性を指定する。Fire Fighterの例(図1参照)では、一人の fireboss の下で複数の fire fighter が働く。Fireboss と Fire Fighter の間の情報伝達、fire fighter 同士の情報伝達に関して指定することができる。例えば、fireboss から Fire Fighter への情報伝達の正確さ (accuracy) を 0.8、その遅延 (delay) を 3 と指定したとしよう。すると、fireboss から Fire Fighter へのメッセージは 5 回に 1 回は欠落し、到達したメッセ

ージは 3 単位時間遅延する。それに対し、Fire Fighter は現場からの情報伝達の困難さがあり、その正確さを 0.5 とするなどの設定が可能となる。

Efficiency of Information Gathering

組織の情報収集能力を指定する。Fire Fighter の例では、fireboss の情報収集は正確性においても速報性においても Fire Fighter に比べ、著しく劣るかもしれない。

Efficiency of Environmental Actions

組織の環境への行為の正確性、遅延を指定する。例えば、Fire Fighter の例では、現場の Fire Fighter の消火作業は、70%しか有効でないかもしれない。

重要なことは、結合するシミュレータの内部に手を加えなくとも、結合部分に connection description を与え、僅かの処理を加えるだけで、社会シミュレーションと環境シミュレーションの柔軟な結合が可能となることである。

3. Q と CORMAS の結合

3.1 Q と CORMAS

実際に社会シミュレータとして Q を用い、環境シミュレータとして CORMAS を用いて、前章のアーキテクチャに従って、両シミュレータの結合を試みる。結合されてできたシミュレータは、表 1 に示すように Q と CORMAS のそれぞれの特徴を生かすことができる。

まず、各エージェントの Q シナリオを用意する。シナリオにはエージェントが実行すべき環境の観察 (Cue)、環境への行為 (Action) が含まれている。同時に、シナリオには他のエージェントからのメッセージ受信 (Cue)、他のエージェントへのメッセージ送信 (Action) も記述される。Q のエージェント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。シナリオの実行中にキューとアクションの実行が必要となれば、その依頼を Q から CORMAS に伝え、その実行結果を CORMAS から Q に伝えなければならない。

注意しなければならないのは時間の管理である。一般には、Q と CORMAS で独立に時間を管理し、Connection Control Module がその同期を図る方法が考えられる。Q はイベントドリブ

表 1 Q and CORMAS

	Q	CORMAS
目的	<ul style="list-style-type: none"> 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。 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。 	<ul style="list-style-type: none"> 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。 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。
計算モデル	<ul style="list-style-type: none"> 拡張状態遷移機械によりエージェントのシナリオを表現。 	<ul style="list-style-type: none"> セルラオートマトンにより、自然環境の変動を表現。
シミュレーション記述の特徴	<ul style="list-style-type: none"> 利用者は各エージェントが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することができる。 インタラクションパターンカード (IPC) による計算機の非専門家によるシナリオ記述が可能。 一部のエージェントを人間に置き換えることにより、ユーザ参加型シミュレーションが可能。 	<ul style="list-style-type: none"> 各ステップ (単位時間) 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シミュレーションが実行できる。 モデリングツール (空間、エージェント、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 (コミュニケーション、統計情報) が提供されている。 MapInfo、ArcView などの GIS (Geographic Information Systems) からの空間マップをインポートして利用可能。
シミュレータの特徴	<ul style="list-style-type: none"> イベント駆動モデル。インタラクションにより生じるイベントを契機として、対応する処理を実行。 時間管理機能はなく、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。 	<ul style="list-style-type: none"> 離散事象モデル。各ステップ (単位時間) 毎に隣接するセル間での自然環境の伝播や、エージェントの行為を実行。 各ステップに時間を割り当てることにより、実時間でのシミュレーションが可能。
応用例	<ul style="list-style-type: none"> 避難誘導のシミュレーション 仮想空間上での社会心理学実験 	<ul style="list-style-type: none"> 水、森林、牧草地などの資源の再利用問題 農地の相互利用問題
実装言語	<ul style="list-style-type: none"> Scheme 	<ul style="list-style-type: none"> Smalltalk

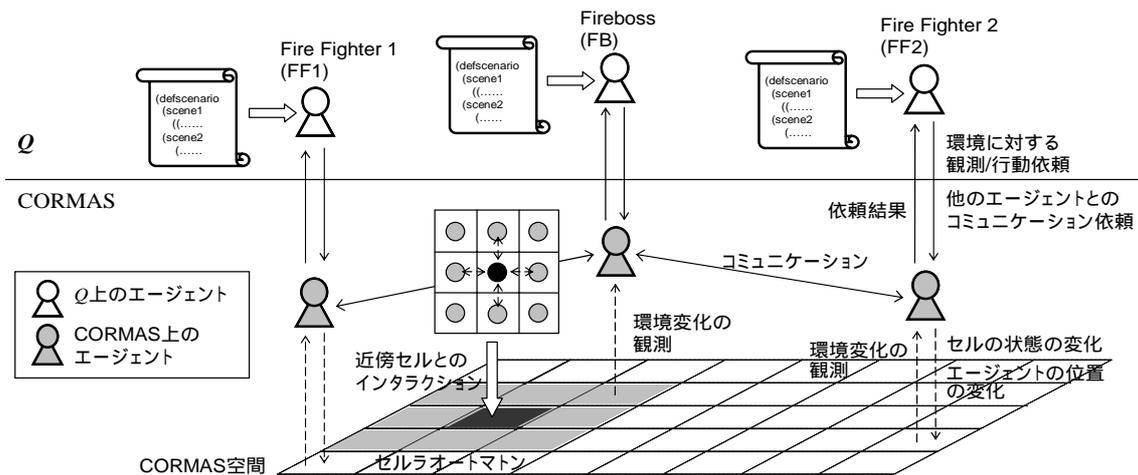


図1 CORMAS/Qによる Fire Fighter シミュレーション

ンでシナリオの実行が進んでいくが、CORMAS はステップ(単位時間)毎にシミュレーションが進行する。従って、両者の同期のためには制御が複雑となり、QとCORMASの内部にも手を入れる必要が生じる。そこで、CORMASにすべての時間管理を任せ、時間を要する観察や行為は全てQからCORMASに依頼することとした。即ち、Qでの処理はCORMASに比べ十分早く、CORMASの実行には遅延を生じさせないという仮定をおき、CORMAS上での時間管理にとって、Qを意識する必要がないようにした。

CORMASでは、ステップごとに、空間的な計算の伝播と、エージェントの観察/行為が計算される。例えば、Qのシナリオから依頼された行為が実行されると、その環境への波及は以降のステップでCORMASにより計算され、Qのシナリオから依頼された観測により、以降のシナリオの実行にフィードバックが生じる。

3.2 Connection Control Module

QとCORMASを接続するために、Q言語処理系側にQConnectorを、CORMAS側にCORMASConnectorを用意した。QConnectorはCORMAS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。CORMASConnectorでは、キューやアクションの実行要求を受け取り、対応するCORMASメソッドを呼び出す。また、その実行結果をQ側に返信する役割がある。

4. Fire Fighter モデル

4.1 問題

QとCORMASの接続の有効性を確認する例題として、Cohenらによって行われたYellowstone National Parkの火災シミュレーション[Cohen 89]を実装した。

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、平地、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、地形の特性などによって広がっていく。

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ことから始まる。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、指示されたルートに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れば、ルートを自ら変更する。また、急激な環境の変化により、Fire Fighterに突如として火

災が及ぶことがある。その場合は、行動プランに関係なく、瞬時にその場所から退去する。このようなルートの変更はFirebossに報告され、Firebossは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。そのプランと比較して、Fire Fighterのルート変更が適切なものでないと判断されれば、Fire Fighterは、現在のルートプランをFirebossのルートプランに変更する。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。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。

4.2 シナリオ

図1にこの問題のCORMAS/Qでの実装を示す。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。一方、FirebossとFire Fighterに対応するエージェントをQとCORMAS双方に定義し、QシナリオでCORMAS内のエージェントの行動を制御している。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が記述されている。

すでに述べたFirebossとFire Fighterの役割は、状態遷移図で表現することができる(図2参照)。また、FirebossのQシナリオを図3に示す。Qシナリオは基本的に状態遷移図を素直にコーディングしたものである。4.1の問題説明と図2の状態遷移図、図3のシナリオ記述の対応は簡単に確認できると思われる。

4.3 結果

図4にシミュレーションのスクリーンショットを示す。初期値として空間の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。CORMASのマップ(図4参照)は、firelineと環境の境界(川)により火災を囲み、シナリオが意図したように火災の広がりを食い止めることができたことを示している。このことは、エージェントの行動プランを表したシナリオの適切さと、多層マルチエージェントシミュレータの妥当性が示されたと考えられる。

5. おわりに

本論文では、総合的なシミュレーションシステムを構築する手法を示した。具体的には、拡張状態遷移機械を計算モデルとしてイベントドリブン方式のQ言語処理系と、セルラオートマトンを計算モデルとしてステップ実行方式のCORMASを結合した。結合されたシミュレーションシステムにより、社会シミュレーションと環境シミュレーションが可能となる。

結合を可能としたのは、Q言語処理系がレガシーなエージェントのシナリオを記述し、その実行を制御するという特徴を持っていたことに負うところが大きい。時間を要するエージェントの行為を全てCORMASに任せ、時間管理を一元化し、結合された

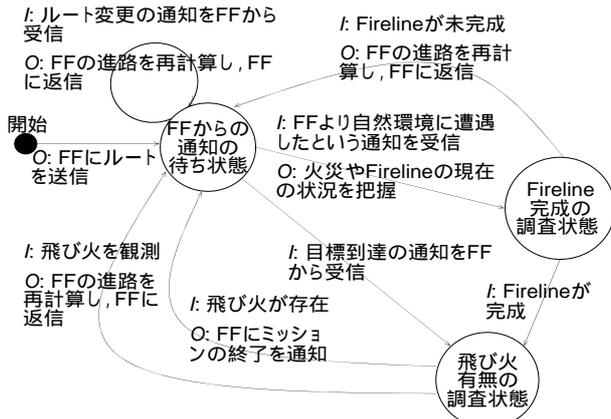


図2 Firebossの状態遷移図

```
(defscenario fireboss
  (&pattern ($FF_name "") ($FF_info "") ($env_info "") ($route ""))
  (Waiting-FF-report
    ((?route_change_report :from $FF_name :info $FF_info)
     (!get_env_info :info $env_info)
     (!calculate_route :info $FF_info :info $env_info :result $route)
     (!send_route :to $FF_name :result $route)
     (go Waiting-FF-report))
    ((?natural_boundary_report :from $FF_name :info $FF_info)
     (!get_env_info :info $env_info) (go Investigating-firelines))
    ((?goal_arrival_report :from $FF_name :info $FF_info)
     (go Searching-other-fire)))
  (Investigating-firelines
    ((?firelines_encircled :info $env_info) (go Searching-other-fire))
    ((?firelines_not_encircled :info $env_info)
     (!calculate_route :info $FF_info :info $env_info :result $route)
     (!send_route :to $FF_name :result $route)
     (go Waiting-FF-report)))
  (Searching-other-fire
    ((?no_more_fire :info $env_info)
     (!send_mission-complete :to $FF_name)
     (go Watching-FF-report))
    ((?other_fire :info $env_info)
     (!calculate_route :info $FF_info :info $env_info :result $route)
     (!send_route :to $FF_name :result $route)
     (go Waiting-FF-report))))
```

図3 FirebossのQシナリオ

システムが複雑になることを避けている。さらに、結合部分に connection description に従った情報の伝達制御を埋め込むことにより、多様なシミュレーションを可能とすることに成功している。さらに CORMAS/Q により、Cohen らによる森林火災のシミュレーションモデルを実装し、モデルが意図する結果を得ることで、システムの動作確認を行った。

我々は IRRI (International Rice research Institute) Thailand Office と CIRAD の共同プロジェクトに協力して、タイにおける環境シミュレーションに、この CORMAS/Q の適用を検討し始めている。Q は、シナリオで制御された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ることができるので、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。今後は、さらに共同研究を進め、実問題への適用を図っていく予定である。

謝辞

本研究はパリ第六大学との共同プロジェクトの一環として行われた。御助言を頂いたパリ第六大学の Alexis Drogoul 教授、本実装において協力いただいた Stephane Bonneaud 氏に感謝の意を表す。

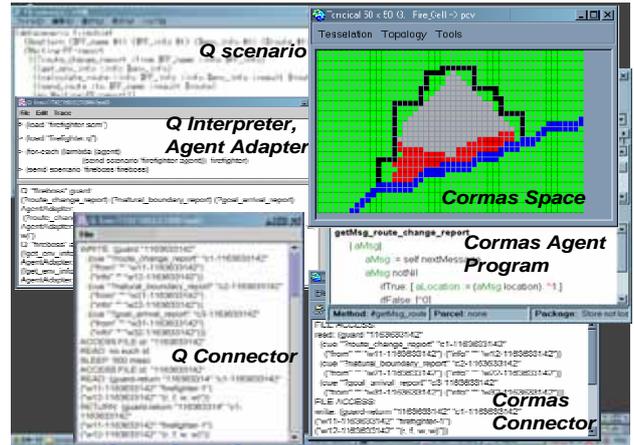


図4 Fire Fighter シミュレーションの様子

参考文献

[Axtell 96] Axtell, R. L, Epstein, J. M. and Cohen, M. D. Aligning Simulation Models: A Case Study and Results. *Computational and Mathematical Organization Theory*, vol. 1, 123-141, 1996.

[Bah 03] Bah, A., Touré, I. and Le Page, Ch. An Agent-Based Model tool for multi-agent simulation for Understanding the Multiple Uses of space Land and Resources around a Drilling Sites in the Sahel. *In Proceedings of Modsim 2003 International Congress on Modeling and Simulation*, 1060-1065, 2003.

[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* 1416, 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, 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, 54-59, 2002.

[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)*, 369-376, 2003.

[Nakanishi 03] Nakanishi, H., Nakazawa, S., Ishida, T., Takanashi, K. and Isbister, K. Can Software Agents Influence Human Relations? Balance Theory in Agent-mediated Communities. *In Proceedings of International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-03)*, 717-724, 2003.

[Noda 01] Noda, I. and Stone, P., RoboCup Soccer Server and CMUnited: Implemented Infrastructure for MAS Research, Infrastructure for Agents. *Multi-Agent Systems, and Scalable Multi-Agent Systems*, Wagner, T., Rana, O., ed., 94-101, 2001.

[Perez 02] Perez, P. and Ardlie, N., Kunepong, 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, 251-259, 2002.