3U－3

分散協調型強化学習による マルチエージェントシステム

井田 通　上田 茂雄　阿部倫之　服部 進実
金沢工業大学 情報工学科

1 はじめに

ネットワーク内に存在するソフトウェアエージェントは、利用者の代理として働くための自律的能力と環境に対する適応学習能力を必要としている [4]。ここで、環境変動に対するリアルタイム性を考慮すると、行為を適用しながら環境状態との「ずれ」をインクリメンタルに補正していくリアクティブな推論システムが必要と考える。本稿では、リアクティブなルールと協調型強化学習によって環境変化に追従していくマルチエージェントシステムについて述べ、その評価プラットフォームを示す。

2 マルチエージェントシステム

2.1 エージェントの動作シナリオ

エージェントの動作シナリオは、組み替えの容易さとリアルタイム性を考慮し、リアクティブルール [4] で記述する。リアクティブルール（以下ルール）は、エージェントが観測した環境情報から直接判定できる条件部と、環境に対して直接適用できる行為部から成り立っている。条件部が満足したルール（発火ルール）は実行権を持つが、排他的（または非決定的）な関係を持つルールが存在する場合、ルールの評価値（付け値、bid）に比例した確率で選択実行する。この非決定的に形成される行為の系列がエピソードであり、付け値の学習結果に依存して適応的に形成される。十分な学習（トレーニング）によって付け値の変化が小さくなった場合、最大の付け値（highest bid）を持つ発火ルールを選択する。

2.2 強化学習

強化学習は、行為の評価値（報酬）に基づいて試行錯誤的に適用行為を洗練していく教師なしの機械学習であり、変化の予測が困難な問題に対して一定の適応能力が期待できる [4]。エージェントの学習目標は、動的に変化する環境下において、有効なエピソード（行為の系列）に対して継続的に報酬を与えることといえる。本システムでは環境変化を考慮して、学習途中においても報酬獲得の継続性が期待できる profit sharing 法 [2] と bucket brigade 法 [1] を用いている。付け値は、ルールの強度（strength）と支持度（support）から算出する。強度は分配された報酬によって変化し、支持度は条件部の成立に貢献した発火ルールの個数によって変化する。profit sharing 法では、報酬の分配規則（均等分配、等差・等比減少分配）に基づいてルールを直接的に強化するため、少ない試行回数で学習できる可能性を有している。また、bucket brigade 法は、現在の発火ルールと１ステップ過去の発火ルール間でのみ報酬のやり取りを行なうため、推論履歴を保持する必要が無くリアルタイム性が高い。

2.3 分散協調型強化学習

本システムでは、profit sharing 法と bucket brigade 法を協調型マルチエージェントシステムに適用できるように拡張を試みた。図１に強化学習エージェントにおける協調の枠組みを示す。エージェントの協調では、タスク通知などの明示的なメッセージの他に、自己の意図を

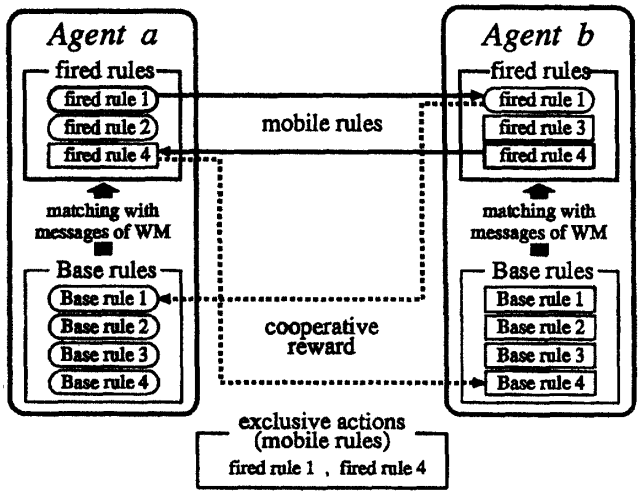

図 1: 分散協調モデル

Multi-Agent System based on the Distributed and Cooperative Reinforcement Learning
Tooru Ida, Shigeo Ueda, Noriyuki Abe, Shimmi Hattori
Kanazawa Institute of Technology
7-1 Ohgigaoka, Nonoichi, Ishikawa 921, Japan

表現した発火ルールを通知できる。これはエージェント間で事前に設定した排他行為が発生したとき、互いに相手の発火ルールを自己のエピソードに反映させることによって、相手の意図を共有し、適応的な協調の実現をはかるものである。また、取り込んだ発火ルールの貢献度を報酬（cooperative reward）として相手に通知することで、システム全体の均衡状態を維持（協調）する方向にエージェントの学習を進める。

3 評価プラットフォーム

本マルチエージェントシステムはCLOS（common lisp object system）を用いて作成した。エージェントの構成を図2に示す。エージェント内のルールは支持度に基づいてクラスタリングされており、支持度が閾値以下のルールはマッチングの対象から除外される。マッチング対象のルールを活性化ルールと呼び、エージェント内で活性化しているシナリオを表す。この活性化シナリオは学習の進行に伴い変化する。このマルチエージェントシステムをマルチサーバ負荷シミュレータと連動させて評価プラットフォームを実装した（図3）。マルチサーバ負荷シミュレータは、FORTRANとシミュレーション言語SLAM IIで作成している。クライアントの要求をサーバが受け付けるとそのサーバに対応したエージェントは、実行待ち時間や、損失率などから適応可能な行為を選択する。またエージェントは、サーバのバッファ使用率や画像データのフレーム損失率の変動状況を監視して、クライアントが要求したQoSと画面サイズの譲歩値に従ってそれらの値を適応的に変更する。これにより、サーバ負荷の均衡状態を維持すると共に、リクエストの平均待ち時間を下げるように強度と支持度の学習が進行する。

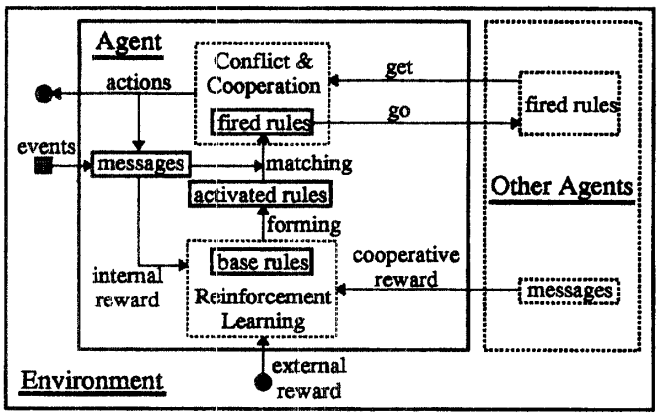

図 2: エージェントの構成

4 おわりに

本稿では、分散協調型強化学習によって環境変化に追従していくマルチエージェントシステムとその評価プラットフォームについて述べた。現在、リアクティブルールによるエージェントの動作シナリオを実装しており、今後、学習能力の評価を進めていく予定である。

参考文献

[1] J.H.Holland, K.J.Holyoak, R.E.Nisbett, P.R.Thagard, "Induction: Process of Inference, Learning, and Discovery",the MIT press(1986)

[2] Grefenstette,J.J.,"Credit Assignment in Rule Discovery Systems Based on Genetic Algorithms", Machine Learning,Vol.3,pp.225-245(1988)

[3] M.Tan,"Multi-agent Reinforcement Learning: Independent vs. Cooperative Agents", Proc.10th Int. Conf. on Machine Learning,pp.330-337(1993)

[4] 石田亨, 山田誠二, 他,"特集「エージェントの基礎と応用」", 人工知能学会誌,Vol.10,no.5,pp662-711(1995)

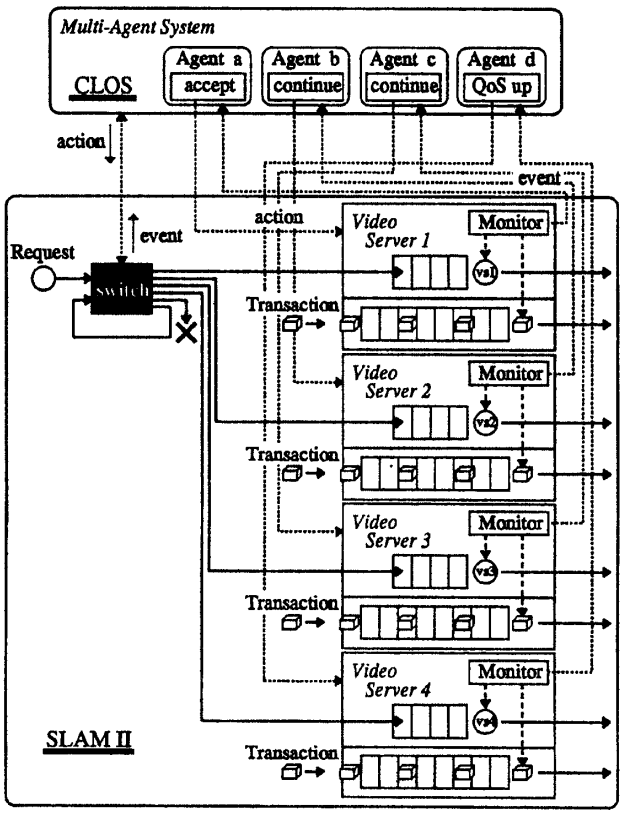

図 3: 評価プラットフォーム