

セミリフレクション機能を持ったエージェントの開発

山崎 泰行*

竹内 郁雄†

概要

本研究では認知科学における内省と自己反映計算としてのリフレクションに着目し、セミリフレクションという手法を提案する。このセミリフレクション機能を動的な環境下において不完全な観測情報しか得られないRoboCup サッカーエージェントに実装し、自己監視、不具合に対する応急措置、修正要求の能力を持ったエージェントの開発を行った。結果として、セミリフレクション機能は、エージェントのパフォーマンスの向上と、エージェント開発における開発者の負担軽減に繋がった。セミリフレクションという手法が実世界的な環境下において頑健に動作する自律エージェントを構築するのに有効であることが確認できた。

Development of the Semi-reflective RoboCup soccer agents

Yasuyuki Yamazaki ‡

Ikuo Takeuchi §

Abstract

In this research, we propose a technique named semi-reflection. The semi-reflection function was implemented on RoboCup soccer agents which can obtain only incomplete information under a dynamic environment . We developed agents which have the ability of self-monitoring, quick self-correction and malfunction report that demands programmers to do debugging. The semi-reflection function improved the agent performance and reduced developer's burden of agent development. Experiments show that the technique of semi-reflection is effective to construct a robust autonomous agents which work under real world environment.

1 はじめに

社会全体の情報化、IT 化が進むにつれ、日常の様々な場面で情報処理システムが使用されるようになった。このように情報処理が対象とする問題

領域が拡大するのに伴い、人間、環境といった複雑に変化する対象を扱うためのシステムの重要性が増している。例として、ペットロボットやヒューマノイド型ロボットなどの対話的なロボットは実環境において利用者と接し、状況や文脈に応じて各利用者の望む通りに動作しなくてはならない。この時、設計者が予想もしなかったような状況や、不完全な観測情報からでもリアルタイムに適切な動作を決定する必要がある。しかし、実行時にあ

*電気通信大学電気通信学研究科情報工学専攻

†東京大学情報理工学系研究科創造情報学専攻

‡Department of Computer Science, Graduate School of Electro-Communications, The University of Electro-Communications

§Creative Informatics, Graduate School of Information Science and Technology, The University of Tokyo

りうる全ての可能性を的確に予測し、どんな状況においても適切な対応をすることのできる情報処理システムを実現することは容易ではない。従って、動的な環境のもとで不完全な観測情報しか得られない中でも、実時間で問題に対処できるような情報処理が期待されている。

2 背景と目的

環境への適応性の高いシステムを構築する手法としてリフレクションという概念がある。リフレクションとは自己の状態を感知したり変更することができる機構のことであり、リフレクションを導入したシステムではシステム自身の内部に隠蔽されたパラメータやプログラムコード自身を参照、変更することにより機能の修正、追加が行える。

また、認知科学においても、内省に基づいて自己修正を行うというメタ認知活動としてのリフレクションが近年注目されている。自分の行動や思考を観察、再検討し、その行動や思考を生み出した知識を再構成するリフレクションは、人間の学習における最も重要な活動の一つであるとされている [1]。

本研究では動的な環境に対して、適応性の高いシステムを構築するため、リフレクションの概念を援用する。研究プラットフォームとして RoboCup サッカーシミュレーションを用いる。

RoboCup サッカーシミュレーションは、サッカーサーバにエージェントを接続し、それぞれのエージェントがサーバとのデータのやりとりを行うことにより、サッカーのシミュレーションを行うというものである。個々のエージェントがサーバから得るデータは誤差を含む不完全情報であり、しかも周りの環境は複雑に変化する。エージェントはこの不完全情報と複雑な環境の中で実時間で適切な動作を決定しなければならない [5]。

3 研究方針

本研究ではセミリフレクションという手法を提案し、セミリフレクション機能を実現するための自己監視エージェントにより自己監視、不具合に対する応急措置、修正要求の能力を実装する。

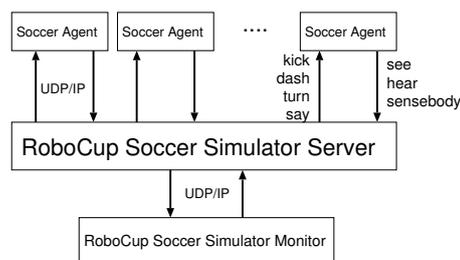


図 1: サッカーシミュレーションシステムの構成

3.1 セミリフレクション

セミリフレクションでは、エージェントの不審な挙動や異常行動を検出し、エージェントの動作を応急措置として表層的に修正するが、異常の原因であるプログラムそのものは変更しない。修正内容やその時の環境状態をログとして残すことで、修正したことを通知し、開発者にプログラム変更を促す。修正が表層的と言う点で弱いリフレクションであるといえる。しかし、複雑な環境で動作するエージェントの行動を修正する場合、想定外の状況に対応できるようにプログラムコード自体を変更するようリフレクションを実現することは困難である。セミリフレクションでは応急措置として確実な行動のみを準備することでその場の問題に即応し、異常報告によって開発者のデバッグの負担を軽くすることができる。また、この方法は一般的な学習によって環境に適応するという手法を用いるより実践向きかつ実時間問題向きである。

3.2 自己監視エージェント

本研究では異常検出、問題修正といった機能を自己監視エージェントとして実装する。自己監視エージェントは図 2 のように行動決定などを行うメインエージェントとは同一エージェント内にある別のエージェントとして動作し、自己をいわば他者の目で分析する役目を持っている。

自己監視エージェントは、メインエージェントが持っている環境認識、実行した行動とそれを実行するに至った経緯である行動の目的の履歴を、必要なだけ収集し、それらをもとに認識、行動、目的の評価を行う。評価により異常が検出された場合には、認識の修正や応急措置の行動をメインエー

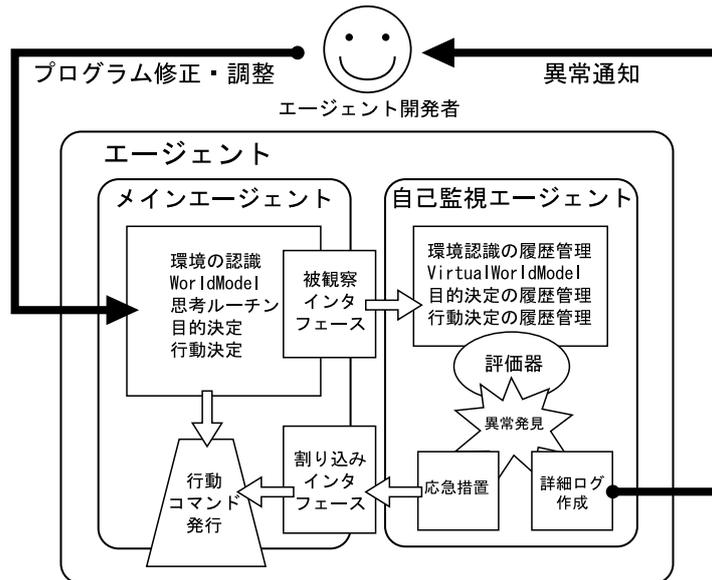


図 2: エージェント構成

エージェントに指示すると同時に、それらに関するログ等をエージェント開発者に通知する。

4 セミリフレクションの実現

本研究では自己監視エージェントによるセミリフレクション機能を RoboCup サッカーエージェントに実装した。

4.1 エージェントの内部構造

メインエージェントは大きく分けて、ワールドモデリング、プランニング、スキルの3つの処理部と、WorldModel と呼ばれるデータ構造を持つ。WorldModel はエージェントが持つ環境の認識であり、ボールや選手の位置や速度などの情報の総体である。ワールドモデリングはエージェントの認識を構築する処理であり、サッカーサーバから与えられる誤差を含んだセンサ情報から WorldModel を構築する処理と、環境の変化にあわせて WorldModel を更新する処理からなる。プランニングでは WorldModel から現在の戦術的状况を把握し、「パスをする」「ボールを取りに行く」などその状況で採るべき目的を決定する。ドリブルなどのようなスキルでは、ある目的を達成するために最適な行

動、または行動列を決定し、サーバーに送信する(走る、蹴るなどの)基本行動コマンドを決定する。

自己監視エージェントは図2のように単一エージェント内のミニエージェントとして独立した形で実装した。自己監視エージェントは、メインエージェントの WorldModel、プランニングによって決定される目的、スキルによって決定される行動を監視し、不適切な動作を発見すると、応急措置をメインエージェントに割り込んで指示すると同時に、それらに関するログ等をエージェント開発者に通知する。

4.2 メインエージェントの監視

我々が従来開発してきたエージェントには以下の問題があった。

- 開発者が想定していない状況での異常な動作
- 同じ動作の無意味な繰り返し
- 一貫性のない動作

これらの問題は「自分が何を知っていて、何を知らないか」という認識や、「自分がどんな目的で、現在の行動をとっているのか」という動作に対する内省(高次認知)が欠如していることから起こるものである。従って、エージェントの認識、行動、

目的を監視し、それらの履歴を評価することで異常検出を行った。

4.3 環境認識の修正

エージェントの環境認識を監視して、World-Model の持つ情報の修正を行った。ボールや選手の位置などの外部環境の情報は視覚に頼っており、従来のエージェントは見えていない選手を不明という扱いにしていた。そこで、自己監視エージェントによって、位置情報の履歴を利用し、視界外の相手選手の位置の推測を行った。

視界外の相手チームの選手の位置を推測するには、見失っている間に選手が「どこからどこに向かってどれだけ移動をしているか」を推測すればよい。そのためには、見えている間に選手がどのような動きをしているかを分析する必要がある。サッカーという競技はボールを中心に進行しているため、ボールの情報は選手の移動に大きく関係する。ボールの位置や移動量だけでなく、ボールがキープされているのか、ルーズボールなのかといったボールを中心とした状況も選手の位置の推測には有効な情報である。従って、選手の位置情報を、情報取得時のボールの位置や状況によって分類することにより、位置の推測を行った。

自己監視エージェントは位置推測によって仮想の WorldModel である VirtualWorldModel を構築し、それをメインエージェントの WorldModel に反映させることで環境認識の修正を行った¹。

4.4 動作の修正

ここでいう「動作」とは、エージェントの目的とそれを達成するための行動を総称する言葉である。メインエージェントは目的をたて、その目的を達成するための行動を毎サイクル実行する。目的はエージェントの持つ WorldModel などを参考に決定されるが、不完全要素や動的に変化する環境が原因となり、適切でない目的が選択されてしまうこともありうる。また、行動は目的よりも精

¹このような WorldModel の補正は本来内省ではなく、メインエージェントが行うべきものであるが、本研究では自己監視エージェントの機能として実装した。一般的な状況認識の修正に自己監視エージェントが有効である場合が予想されるからである。

密な情報が必要であるので、適切な決定はさらに難しい。本研究の異常検出で対象とした動作は以下の三つである。

- サッカーにおいて明らかに異常と判断できる目的の選択やエージェントのたてた目的にそぐわない行動
- 動作の無意味な繰り返し
- 前後の繋がりのない目的のふらふらとした変更

このような異常動作はメインエージェントのプランニングによって防ぐこともできそうであるが、すでに巨大化しているエージェントプログラムに手を加えて調整するのは多大な労力を必要とするばかりでなく、構造を複雑化することにもつながる。従って、自己監視エージェントとしてモジュール化した形で実装したほうが、見通しのよいプログラムになる。また、メインエージェントとは全く違ったロジックで動作を評価することで、客観的な異常検出ができる。

自己監視エージェントはメインエージェントの動作履歴を見直し、評価することで異常検出を行う。動作履歴とは行動内容や目的内容の記録である。自己監視エージェントの評価によって異常と判断された場合には、メインエージェントの動作決定ルーチンに割り込みをし、自己監視エージェントの決定した修正行動をメインエージェントに強制的にとらせることで、動作の修正を行う。ここでの修正はプログラムコードなどの修正ではなく、具体的な応急措置行動をとらせることによる修正である。

応急措置の流れは以下の通りである。

1. メインエージェントが目的と行動を決定
2. 自己監視エージェントによる応急措置
 - 2.1 目的の妥当性を評価
 - 2.2 妥当性の低い目的の繰り返しの検出
 - 2.3 一貫性の有無を検出
 - 2.4 異常が検出されたなら、目的を変更
 - 2.5 行動の妥当性を評価
 - 2.6 妥当性の低い行動の繰り返しの検出
 - 2.7 異常が検出されたなら、行動を変更
3. 応急措置に従いメインエージェントの目的と行動を修正

4.4.1 異常動作の検出

行動が現在の目的を達成するためのものとして妥当であるか、目的がサッカーのルール、知識、戦略の3点から妥当であるかどうかを評価することで異常検出を行うために行動、目的の妥当性を表す値として妥当値を導入した。また、妥当性を評価するのに行動、目的における最低条件と理想値というものを定義した。メインエージェントの動作が最低条件を満たしているかどうか、理想値に近いものであるかどうかを評価することで、妥当値の計算を行う。妥当値が0であれば、それは完全な異常行動であり、100に近ければ近いほど妥当性の高い行動であるとした。

目的がパスである場合の妥当値計算の例を示す。パスにおける最低条件は

ルール パスの目標位置がオフサイドではない
知識 パスの目標位置がフィールド内である
戦略 パスの目標位置が近すぎない(自分のキック範囲内ではない)

とし、これらの条件を満たしていなければ妥当値を0とした。パスにおける理想値は

理想値1 ボール位置からパスの目標位置までの距離がフィールド幅の1/3程度である

理想値2 パスの目標方向が相手ゴール方向である

とした。この理想値とこれから実行しようとする目的の内容を比較することで妥当値の計算を行う。例として、ボールを保持しているエージェントが後方にいる味方選手にパスを出すという目的を持ち、ボール位置とパスの目標位置との距離が25.0m、パスの目標方向と相手ゴール方向との差が170.0°である時の妥当値は

$$best_pass_dist = 17.5 \quad max_pass_dir = 180.0$$

$$diff_dist = \min(25.0, 2 \times best_pass_dist) - best_pass_dist$$

$$理想値1 = 100 \times \left(1 - \frac{diff_dist}{best_pass_dist}\right) = 57$$

$$理想値2 = 100 \times \left(1 - \frac{\min(170.0, max_pass_dir)}{max_pass_dir}\right)$$

$$= 6$$

$$妥当値 = (57 + 6) \div 2 = 32$$

と計算する。

4.4.2 動作の無意味な繰り返しの検出

ドリブルはキックとダッシュの繰り返しであるので、繰り返すこと自体が異常であるわけではない。しかし、繰り返されると異常と判断できる動作もある。前節で述べたように動作の妥当性を評価した場合、妥当性の低い動作が検出される。単発の妥当性の低い動作は異常とは言いきれないが、そのような動作が何度も繰り返されることは異常であるといえる。例えば、目的がパスである時、即座にパス方向にボールを蹴りださず、より蹴りやすい位置にボールをコントロールすることがある。この行動が何サイクルも続いた場合、目的であるパスがいつまでも達成できず、異常であるといえる。

このような妥当性の低い動作の繰り返しを検出するため、妥当性の低い動作を記録し、記録された動作とこれから実行しようとしている動作の妥当値や状況を比較することで繰り返し検出を行った。また、動作の記録を行う領域を短期、中期、長期記憶の3種類用意することで、様々な周期で発生する繰り返しの検出を可能にした。短期記憶では15サイクル間、中期記憶では300サイクル間、長期記憶では1試合の間における全ての動作を検出対象とした²。

4.4.3 一貫性のない動作の検出

一貫性のない動作とは、目的が1サイクル毎に変化することで、どの目的も達成できないままエージェントがその場で右往左往してしまうことである。環境が動的に変化し、しかも不完全情報しか手に入らない状況では毎サイクル毎の判断にブレがでてきてしまうことはある程度さげられない。従って、目的の履歴に基づいて以下のような動作の検出を行った。

- 目的そのものが短期間に複数回変更される
- 目的の目標方向が短期間に複数回変更される

目的そのものの変更とは、あるサイクルではシュートを選択したが、次のサイクルではドリブル、次にはパスを選択するといったような変更のことを

²1 試合は6000サイクルである。

指す。目的の目標方向の変更とは、あるサイクルでは前方にドリブルを選択したが、次のサイクルでは後方にドリブル、次には右方にドリブルを選択するといったような変更のことを指す。

4.4.4 応急措置

妥当値が0の場合は明らかな異常動作である。妥当値が0でなく閾値以下の場合には即座に異常とは判断せず、繰り返し検出の対象とする。その動作が繰り返しであるなら、妥当値に繰り返し回数分の減点係数を掛ける。これによって、繰り返し回数が増えれば増えるほど妥当値が減点されることになる。妥当値が閾値以下であれば異常と判断し、応急措置を行う。また、一貫性のない動作が検出された場合にも応急措置を行う。閾値や減点係数は実験によって経験的に定めた。繰り返しの検出対象とするかどうかの閾値は最大値100の半分である50とし、異常と判断するかどうかの閾値はその半分の25とした。減点係数は短期的繰り返しであれば0.85、中期的繰り返しであれば0.7、長期的繰り返しであれば0.5とした。

応急措置では、簡単な修正用候補を用意し、行動や目的を妥当性の高いものや優先度の高いものに修正する。

応急措置における動作の修正案はどれも最適なものとはいえない。しかし、異常動作が検出された場合、それ以外の動作をエージェントにとらせることが重要であり、ある程度妥当性の高い動作に変更できればよしとする。また、応急措置はその場しのぎの修正であるので、最適な行動決定のため多くの計算コストを消費するよりは、単純な計算である程度の妥当性をもつ動作に変更したほうが良い。

4.5 異常な動作の開発者への通知

修正はどれも応急措置的なものであり、表層的な修正である。試合が終わり、次の試合になれば再び前と同じような異常動作が検出され、応急措置で対応することになる。エージェントが異常な動作をしないようにするには、プログラムそのものを変更しなくてはならない。

エージェント開発を行うにあたって、不審な挙動を発見し、それを修正することは非常に労力のいる仕事である。実際、エージェントの不審な挙動を発見するには、開発者自身がつぶさに試合を観察する必要があり、また、見た目には気が付かない異常がエージェントの動作に影響を及ぼしている可能性もある。

従って、開発者による、不審な挙動の発見や原因究明の負担を軽減するため、

- 位置推測で導出した相手エージェントの位置情報
- 毎サイクルの行動・目的とその目標位置や妥当値
- 異常動作の種類や修正前後の行動・目的とその目標位置、妥当値といった応急措置の内容

などのエージェントの内部状態をログとして残す機能を実装した。

5 評価

評価実験としてセミリフレクション機能を実装したエージェントYowAI05、実装していないYowAI04と以下の3チームと対戦を行った。

BrainStormers RoboCup2005の優勝チーム
TokyoTechSFC RoboCup2005 3位の国内トップチーム

CHELLO JapanOpen2004でベスト8のチーム
YowAI05とYowAI04で各チーム相手に30試合行い、評価した。

YowAI05が1試合の間に行った応急措置の回数の平均値をまとめたものを表1に示す。以下は表1の項目の説明である。

全異常行動、全異常目的 異常と判断された全ての行動、目的に対する応急措置の回数（繰り返しも含む）

行動、目的の繰り返し 妥当性の低い行動、目的の繰り返しに対する応急措置の回数

一貫性のない目的 一貫性がないと判断された全ての目的に対する応急措置の回数

全異常行動、行動の繰り返しに対する応急措置の原因はドリブル、パスのスキルにおいて最適な

表 1: 動作の応急措置

応急措置を行った異常動作	対戦チーム		
	vsCHELLO	vsTTSFC	vsBRST
全異常行動	26.41	25.89	20.41
行動の繰り返し	6.04	6.58	5.2
全異常目的	140.43	93.44	113.3
目的の繰り返し	2.51	2.94	3.24
一貫性のない目的	1.55	2.27	1.74

(TTSFC=TokyoTechSFC BRST=BrainStormers)

行動決定ができていないことが原因であった。全異常目的の応急措置回数が多いのは、ボールを取りに行くというスキルにおいて目標位置がフィールド外になることが原因であった。妥当性の低い目的、一貫性のない目的に対する応急措置回数は相手チームの強さに応じて増えている。これは相手チームの守備によって妥当性の高い目的決定をすることができなくなっているためであった。

各対戦相手との対戦成績と7つの評価項目を表2に示す。評価項目の値は30試合の平均値である。

CHELLO、TokyoTechSFCとの対戦では、わずかに勝率が上昇し、敗率が減少している。TokyoTechSFCとの対戦における勝ち数の変化が大きいことから接戦で勝つことのできる確率が上昇したといえる。しかし、BrainStormersとの対戦で結果が変化するほどの向上ではない。

全体的にYowAI05はYowAI04に比べ、ボール支配率が上回っている。これは行動の応急措置によって無駄な行動が減ったことで、ドリブルやパスのミスが減り、確実な行動が増えたためであると考えられる。ドリブルとパスの確実性の向上がボール支配率の向上に繋がった

応急措置の効果として、YowAI05がフィールド外にボール蹴りだすことが減り、相手チームのキックイン回数が減少した。また、ボール支配率の高いCHELLO、TokyoTechSFCとの対戦ではオフサイド回数が減少した。

以上のことからセミリフレクション機能によりYowAI05はYowAI04よりも動作が安定したことで、わずかではあるが、確かにパフォーマンスの向上があった。

表 2: エージェントの性能評価

評価項目	CHELLO	
	YowAI05	YowAI04
成績(勝-分-敗)	26-2-2	25-1-4
得点	4.7	3.8
失点	1.5	1.8
ドリブル	58.6	56.9
オフサイド	3.0	6.8
相手のキックイン	4.8	7.3
ボール支配率	45.7	42.5
パスミス	28.1	41.1
評価項目	TokyoTechSFC	
	YowAI05	YowAI04
成績(勝-分-敗)	14-12-4	9-15-6
得点	0.8	0.5
失点	0.2	0.3
ドリブル	56.0	47.1
オフサイド	1.5	1.6
相手のキックイン	5.8	9.1
ボール支配率	43.1	38.1
パスミス	24.6	30.0
評価項目	BrainStormers	
	YowAI05	YowAI04
成績(勝-分-敗)	0-0-30	0-0-30
得点	0.2	0.2
失点	5.5	5.7
ドリブル	35.8	31.9
オフサイド	0.4	0.1
相手のキックイン	8.0	11.0
ボール支配率	28.5	24.8
パスミス	24.7	28.1

6 異常通知機能の評価と考察

エージェント開発における異常通知機能の有用性について述べる。

異常通知機能によって、異常動作の発見の手間が大幅に減った。以前は全ての試合を実際に見なければそのような動作の発見はできなかったが、異常や応急措置の内容をログとしてを残すことにより、試合を自動実行しておき、後でログに残されている異常箇所を見直すことで発見ができるようになった。

また、異常通知機能により以前は発見できなかったようなバグを発見することができた。例えば、ボールを取りに行くというスキルにおいて移動目標位置がフィールド外になってしまうという異常が発見された。このような異常は試合を観察しているだけでは気が付かないものであり、エージェントの目的や目標位置が出力されたログを見直さ

ないと発見できないものである。

7 関連研究

Kennedy、Sloman [2] らは Treasure ゲームを題材とし、主に正しい行動パターンとエージェントの行動を比較することでリフレクションを行っている。Oehlmann、Edwards、Sleeman [3] らは実験によって知識を獲得し、実験と現状との比較、類推によって異常を発見するシステムを構築した。両者とも RoboCup サッカーシミュレーションのように複雑に変化するような環境を扱ってはならず、エージェントの行動も単純である。

Guessoum [4] はエージェント同士が監視しあうことで、異常を検知するという手法を採っている。RoboCup サッカーシミュレーションではエージェントの視界や通信環境に制限があり、他のエージェントの異常を視覚により検出し、それを通信で伝えることは難しい。

8 まとめ

本研究では認知科学における内省と自己反映計算としてのリフレクションに着目し、セミリフレクションというモジュール性の高い手法を提案した。セミリフレクション機能では、単一エージェント内のミニエージェントである自己監視エージェントが、動作決定を行うメインエージェントの不審な挙動を発見した際に応急措置的動作に切り換えるという手法をとる。応急措置が行われたら、修正内容やその時の環境状態をログとして残すことで、開発者にプログラムの変更を促す。このセミリフレクション機能を RoboCup サッカーエージェントに実装し、自己監視、不具合に対する応急措置、修正要求の能力を持ったエージェントの開発を行った。

結果として、セミリフレクション機能による異常検出や応急措置がエージェントの動作の安定性に繋がり、さらに異常通知機能が開発者のエージェント開発の手助けとなった。セミリフレクションという手法が動的な環境下において頑健に動作する自律エージェントを構築するのに有効であることが確認できた。

9 今後の課題

本研究では単一エージェント内のミニエージェントとして自己監視エージェントを実装した。そのため真の意味での客観的な評価とはいえず、エージェントそのものに異常があった場合には検出できない。外部のエージェントによる監視も併用し、統合することができれば、単一エージェントでは検出できないようなエージェントの異常のほか、チームプレイの異常の検出も可能になるであろう。

今回実装したセミリフレクション機能は我々が開発してきた RoboCup サッカーエージェントに特化したものである。自己監視エージェントを可能な限り抽象化・汎用化し、他の RoboCup サッカーエージェントにも適用可能にすることや、自己監視エージェントを RoboCup サッカーに依存しない一般的な内省エージェントのアーキテクチャに基づいて実装することが課題として挙げられる。

参考文献

- [1] John A. Barnden: Metaphor, Self-Reflection and the Nature of Mind, Euroconference on Consciousness and the Imagination, 2002.
- [2] Catriona M. Kennedy and Aaron Sloman: Autonomous Recovery from Hostile Code Insertion using Reflection, Cognitive Systems Research, pp.89-117, 2003.
- [3] Ruediger Oehlmann, Pete Edwards, Derek Sleeman: Introspection Planning: Representing Metacognitive Experience, AAAI Press, 102-110, 1995.
- [4] Zahia Guessoum: Monitoring and Organizational-Level Adaptation of Multi-Agent Systems, published in Proc. 3rd Int'l Conf. AAMAS04, ACM Press, 2004.
- [5] Mao Chen, Ehsan Foroughi, Fredrik Heintz, et al: SoccerServer Manual Ver7.07, 2002