人　工　知　能　93−17
（１９９４．　３．　９）

取引履歴公開下での最適取引戦略
自律的エージェント社会の行動規範（１）

伊藤 昭　　矢野博之

郵政省通信総合研究所

自由な取引を行なうエージェントの社会において，公正な取引を保証する方法の一つとして，過去の取引履歴を公表することで，不正な取引に対する社会的制裁を実現する機構を検討する．具体的には，各エージェントには囚人のジレンマと同型の取引を，次々と相手を変えながら行なわせる．このとき，過去の取引（対戦）履歴が公表されるという条件下で，エージェントの採るべき最適な取引戦略を検討する．我々はこのような状況で利用できる代表的な取引アルゴリズムをいくつか取り上げ，それらが相互に協調／競合する振舞いを調べることで，各アルゴリズムの定性的性質を理解しようと試みる．また，この結果に基づき，エージェントが採るべき最適な取引戦略についての考察を行う．

Optimal Deal Strategies Under the Disclosure of the Contract History
The Behavioral Norm in the Society of Independent Agents (1)

Akira Ito　Hiroyuki Yano

Communications Research Laboratory

Iwaoka, Nishiku, Kobe, 651-24, Japan

The social sanction mechanism against unfair deals is investigated in a society of autonomous agents. The mechanism is realized by disclosing the contract histories of all the agents. To simulate the situation, each agent is made to engage in the deal equivalent to the "Prisoner's dilemma" problem repetitively, each time changing the other party of the deal. Optimal deal strategies are searched under the condition that the contract records will be disclosed and open to all the agents. Several deal algorithms are taken up, and their behaviors are investigated by matching them under various conditions. Based on the results, the condition for optimal deal strategies of the agents are discussed.

1 まえがき

自律したエージェントによって構成される社会では，各エージェントは自分の持つ行動規範に従って，他のエージェントと交渉しながら，自己の目的を追求する．このとき，エージェントが協調して行動することにより，単独で行動するよりも大きな利益を得ることのできる状況は多い．例えば，物々交換社会での「取引」を考えて見る．A，Bが「もの」O_A，O_Bをもっており，それぞれのA，Bにとっての価値E_A，E_Bが

$$E_A(O_A) < E_A(O_B),\ \text{かつ}$$
$$E_B(O_A) > E_B(O_B)$$

であれば，A，Bは相互にものを交換した方が利益になる．

このような推論は，取引においてお互いに相手を信頼できるときには有効であるが，一般的に自律したエージェントの社会においてこのような仮定はなり立たない．実際，A，Bどちらかが不正な取引をすれば短期的にはより大きな利益を手に入れることができる（例えば自分だけ物を受けとって，相手には自分の物を渡さない，または偽物を渡すなど）．従って，自己の利益を追求する自律エージェントは，有利と思えば取引において相手を裏切ることも可能である．

このような状況をモデル化したものとして，囚人のディレンマ[1]が知られている．これは，2人の参加者による非ゼロ和ゲームであり，次のように定式化される．参加者は，C(協調：Cooperate)，またはD(裏切り：Defect)のいずれかの手をだすことができ，双方の手の組合せにより表1のような得点を得る．

表1．囚人のディレンマ：得点表

A \ B	C	D
C	A:3 B:3	A:0 B:5
D	A:5 B:0	A:1 B:1

このゲームの特徴は，相手の手がどのようであっても，自分としてはCを出すよりもDを出す方が有利であること，しかしながら，双方がDを出し続けた時の得点（表1では，1点）や，互い違いにC，Dを出した時の平均得点（(0+5)/2=2.5点）よりも，双方が協調してCを出した時の得点の方が高いということである．従って，何らかの方法で協調する機構が備わっていれば，協調してCを出すのが最善戦略となる．しかしながら，そのような機構が初めからは与えられていない時に，エージェント同士がどのようにして協調を行なったら良いのかが課題となる．

この問題は，Axelrod[1], Genesereth[2]などによって広範に調べられてきた[3]．特に，同じ相手と無限回の対戦をするという仮定の下での，さまざまな戦略の強さが調べられており，コンピュータ選手権もAxelrodらによって，これまでに2回行なわれている．そこでの条件は，同じ相手と無限回（不特定回）対戦を行なう「反復囚人のジレンマ」ゲームを，全参加者（アルゴリズム）によるリーグ形式で行なうというものであった．しかしながら，実社会では常に同じ相手と無限回の対戦を行なうわけではない．実際我々は，多くの場合一回限りの取引を行なうことで，社会的な生活を営んでいる．確かにAxelrodらも文献1で強調しているように，協調活動の出現は「今後とも同じ相手と引続き対戦する」という予測に基づいている．逆に，「一回限りの取引」であれば，双方の最善の戦略は「裏切り」であり，協調行動の出現は不可能である．事実，我々の回りにおいても，一回限りの取引を良いことに，詐欺や，インチキ，またそこまで行かなくとも悪徳商法の話題には限りがない．

人間社会においては，このような不正取引は法律などにより処罰され，損害賠償などによる回復のための制度が整備されている．しかしながら，法的な制度のみが不正取引を防止していると考えるのは誤りであろう．むしろ，実際の商社会においては，「商人は信用が第一」といわれる様に，信用情報の公開による社会的制裁機構が有効に働いているものと思われる．我々は，このような社会的制裁機構をモデル化するために，取引履歴の公開という条件下で，自律したエージェントが囚人のディレンマと同型の取引をおこなうとき，どのような取引戦略を採用すれば良いかを調べることとした．

以下，2では我々のモデルの定式化を行なう．3．では，様々な取引戦略アルゴリズムの導入を行なう．4．では，主として計算機シミュレーションにより，各戦略の性質を調べる．5．では，実験結果に基づく考察を行なう．

2 モデル

エージェントは，2次元空間上のの「世界」をランダムに歩き回っており，同時に同じ場所にいるエージェント同士は「取引」を行なうことができる．取引は囚人のジレンマと同型で，双方はC(Cooperate),

またはD(Defect)の手を出すことができる．参加者A，Bそれぞれが Cまたは Dの手を出したときの得点（利益）を表2に示す．表1との違いは，一回の取引の取引手数料を 2.5と考え，その分を全ての得点から引いたことである．エージェントがどの手を出すかは全くエージェントの自由である．ただし，そのときの双方が出した「手」は，公表されて（永遠に）記録として残される．また，このときの取引履歴は，以後どのエージェントも自由に参照することができる．

表2．取引収益/損失表

A \ B	C	D
C	A:0.5 B:0.5	A:-2.5 B:2.5
D	A:2.5 B:-2.5	A: -1.5 B: -1.5

エージェントは，最初に自己資産 $E = E_0$を持ち，取引のたびに得られた利益／損出は資産に加えられる．実は，取引手数料導入の目的は，この資産が急速に増大することを防ぐ為である．なお，エージェントの資産が一定以上になると，エージェントは自己の資産を分け与えて，子エージェントを作る（分家）．この時，子エージェントは親の行動規範を継承するが，取引履歴は継承しない．これは別の表現をすれば，誰も各エージェントの親エージェントを知る方法がないということである（悪人の子供でも，子供には罪がない）．

以上で述べたモデルの形式的定義を与える．

定義1 世界

世界は$N \times N$の格子からなる2次元平面である．格子位置を $(i,j),(0 \leq i,j \leq N-1)$ で表す．ただし (i,j) と $(i+N,j)$，(i,j) と $(i,j+N)$ とを同一視することで，境界のない閉空間を構成する．また，システムは離散時間を持ち，エージェントは単位時間に（高々）一回の取引を行なう．

定義2 エージェント

エージェントは次の内部状態を持つ：

(位置 L，資産 E，行動規範 A，取引履歴 H)

エージェントはランダムに（単位時間当たり一コマの速度で）2次元空間上の格子点を動き回る．具体的には，単位時間後にエージェントが左右上下の隣接位置に移動する確率をそれぞれ r，元の位置に留まっている確率を $(1-4r)$ とする．同一の位置を占めるエージェントは（一度に1人の相手と）取引をする．ただし同一の場所に3個以上のエージェントがいるときには，ランダムに組合せを作って取引を行なう．(従って，同一場所にいうエージェントの数が奇数の場合には，取り残された一個は取引を行なうことができない)．エージェントは取引が不利だと思っても，取引を見送ることはできない．

定義3 取引

取引は，対戦するエージェントがそれぞれ C，または Dの手を出すことにより行なわれる．取引により得る利益／損害は表2に示されている．取引の履歴は公表され，以後どのエージェントからも参照可能となる．

なお，取引履歴の完全な形式は，時刻の逆順に並べられた取引記録：

(時刻 相手エージェント 自己の手 相手の手)

のリスト構造である．

定義4 行動規範

エージェントは，取引において相手の取引履歴と自己の行動規範 Aによって，次に出す手を決定する．行動規範は，相手のエージェントとその取引履歴情報とを入力として，自己のとるべき手（C，または D）を出力するアルゴリズムである．行動規範アルゴリズムは内部状態を持っていても良い．

定義5 分家（のれん分け）

資産が E_1以上になったエージェントは，自己の資産のうちから E_0を分け与えることで，新しいエージェントを生成する．新しいエージェントは親エージェントの位置，資産，行動規範を継承するが，取引履歴は継承しない．また，任意のエージェントの親エージェントを調べる方法は存在しない．

定義5 ノイズ

エージェントの出す手は時々「ノイズ」によって，反転させられることがある．すなわち，自己の行動規範従えば Cを出すべき時に Dを出したり，逆に Dを出すべき時に Cを出してしまったりする．この確率を，$e(0 \leq e < 1)$ で表す．

我々は以上のような状況設定において，エージェントが採るべき「最適な」行動規範を求めたいのであるが，そのためにはまず評価基準を明確にしておかなくてはならない．ここでいう「最適な」行動規範とは，様々な行動規範を持ったエージェントの社会の中で，自分の子エージェント（分家）を最も多く生成することのできる行動規範とする．

実世界においても絶対的に「最適な」種というものが存在しないのと同様に，このような意味で最適な行動規範はおそらく存在しないものと思われる．従って，以下では様々な行動規範アルゴリズムについて，

・アルゴリズムの1対1対決での相対的強さ，
・様々なアルゴリズムが混在する時の振舞い，
・小さな攪乱に対する各戦略のロバストネス，
・戦略アルゴリズムの簡単さ，

などを評価基準にして計算機シミュレーションにより実験的に調べていくこととする．

3 行動規範アルゴリズム

以下では，行動規範アルゴリズムの導入と分類を行なう．行動規範アルゴリズムの任務は，取引時に相手エージェントの過去の取引履歴から，相手の採用しているアルゴリズムや相手の次の手を推測し，自己の利益にとって最適な手を決めることである．ただし，自己の手は公表／記録されることを考えると，最善手を考えるに当たって，今回の取引による利益だけではなく，Dを出すことによる将来への影響などを合わせて考える必要がある．なお，当然のことながら，相手エージェントの採用しているアルゴリズムを直接的に知ることはできないものとする．

まず，アルゴリズムを協調アルゴリズムと，非協調アルゴリズムに分類する．協調アルゴリズムとは，協調アルゴリズムだけの社会では，常にCを出し続けるアルゴリズムである．協調アルゴリズムは，Axelrod らの分類では，「上品な」アルゴリズムに相当する（しかしながら，基礎となるモデルが異なるため，完全な対応は不可能である）．一方，協調アルゴリズムでないものは，非協調アルゴリズムである．なお，協調アルゴリズムも非協調アルゴリズムとの共存条件では，協調アルゴリズムに対してDを出すことがある．

次に，協調アルゴリズムを戦闘的アルゴリズムと，平和的アルゴリズムとに分ける．戦闘的アルゴリズムとは，非協調アルゴリズムに対しては，Dを出すことで非協調アルゴリズムを排斥しようとする機能を持つものである．これは，Axelrod らの分類では，「手応えのある」アルゴリズムと呼ばれている．なお，この分類は連続的であり，より戦闘的なものからより平和的なものへと，アルゴリズムは連続的に継っている．我々が主として興味あるものは，戦闘的な協調アルゴリズムである．

まず，基本となるいくつかのアルゴリズムを導入する．

(1) **無抵抗主義** 常にCを出し続ける (CCC).

(2) **しっぺ返し戦略** 相手の直前の手をお返しする (TFT).

(3) **ランダム戦略** 取引相手に関係なく，1/2の確率でCとDとを選ぶ (RAN).

(4) **完全搾取戦略** 常にDを出し続ける (DDD).

(1)(2)が協調戦略に入れられる．また，(2)は，戦闘的戦略でもある．しかしながら，戦闘的戦略として，(2)はあまりうまく動かない．それは，取引相手が直前にDを出していたとしても，それがa)相手を搾取するつもりでなされたのか，b)相手が裏切りを出すと思って自己防御のためにそうしたのか，c)相手の前回の裏切りを凝らしめるためになされたのか，の判別がつかないからである．そういったことを気にせずにしっぺ返しを行なうと，今度は自分が次の取引時にしっぺ返しを受けることになり，最終的にはしっぺ返しの嵐を引き起こすことになる．

とりあえず，b)の危険を回避するために，相手の直前の取引履歴を参照するとき，(時刻 相手エージェント 自己の手 相手の手)=(* * D D) となるものがあれば，その記録を読み飛ばして，さらにその前の記録を参照するものとする．ただし，ここで * はワイルドカード don't care であり，(* * D D) は，自分も相手もDを出したことを表す．これは，次の取引相手の直前の手を調べて，(* * D D) であった場合には，本当はどうしてDを出したのかはわからないが，好意的に「正当防衛としてDを出した」ものと解釈して，その取引のことは許して(忘れて)，もう一つ前の取引を調べることを意味する．そこで，このように変更された戦略の修正版ができる．

(5) **しっぺ返し戦略II** (* * D D) を除いて，相手の直前の手をお返しする (TT3).

つぎは，c)を取り込むことを考える．ただし，相手の直前の相手がその前にDを出していたからといって，それを単純に「凝らしめ」だと見做すわけにはいかない．一つの方法は，「自分が相手の立場にいたらやはりDを出していた」という状況であれば，これを許して（忘れて）あげよう，というものである．ただし，この単純そうに見える戦略は本質的に再帰アルゴリズムであり，場合によっては際限なく履歴を参照する可能性のある，計算時間消費型アルゴリズムである．(我々のモデルのように，歴史に始まりがあれば，そこで停止することは確実であるが)．ともかく，このようにして相手の最後の取引履歴を調べることで，次のアルゴリズムを得る．

(6) **自省戦略** 相手の履歴から，(* * D D) と，「自分でも同じ状況ではDを出す」と思われるものを除いて，相手の直前の手を出す (REF).

このアルゴリズムは，少し寛容過ぎる．相手が，Cを出しそうな相手にCを出していたからといって，単純に仲間だと判断するわけにはいかない．もう少し判定を厳しくして，直前の履歴が双方ともCであった時には，さらにもう一つ前の履歴を参照することにして，次の修正バージョンを得る．

(7) 自省戦略 II (**DD), (**CC), および「自分でも同じ状況ではDを出す」と思われるものを除いて，相手の直前の手を出す (RF1).

これまでのアルゴリズムは，相手の最後の取引記録のみを利用するというものであった．それに対して，これまでの全取引履歴を調べて，相手が協調に値するかどうかを決める「厳格な」アルゴリズムが考えられる．

(8) 自己中心的戦略 相手の全ての取引記録に対して，一回でも「自分だったらCを出す状況でDを出し」ていれば，Dを出す (SLC).

(9) 利己的戦略 相手の全ての取引記録に対して，一回でも「その時自分が出すであろう手と同じ手を出していない」場合があれば，Dを出す (SLF).

(8)は自分よりも「良心的」でないアルゴリズムを排斥しようとする．一方(9)は，自分とは少しでも異なったアルゴリズムを持つものを排斥しようとする．(8)(9)は，全ての取引履歴についての検査を行なうため，極めて計算資源を消費するアルゴリズムである．実際このアルゴリズムは，平均的な履歴の長さに対して，指数関数的な計算時間を要求する．なお，(6)〜(9)は共に，仲間かどうかの判定に自己のアルゴリズムを適用することで，決して「同士打ち（同じアルゴリズムを持っている相手にDを出す)」をしないアルゴリズムとなっている．

4 シミュレーション結果

3．で導入したアルゴリズムを用いて，様々な取引戦略の性質を計算機シミュレーションで実験した．用いられたパラメタは，以下の通りである．

2次元空間の大きさ：$N = 6$ (6x6の2次元格子)．
格子点移動の早さ：$r = 1/5$
（前後左右に移動する確率はそれぞれ1/5)
ノイズ：4.1では $e = 0.1$，それ以外では $e = 0$.
資産:初期資産 $E_0 = 20$，分家のための資産 $E_1 = 40$
（資産が40になると二つに分裂する）

4.1 単一種環境

まず，全てのエージェントが単一のアルゴリズムを共有している場合を考える．もちろん，協調的アルゴリズムでは全てのエージェントがCを出すため，平均得点が0.5点となり，単純に成長する．五分五分でCとDを出すRANは少しづつ減少し，DDDは急速に自滅する．

単一種環境で興味があるのは，システムに何らかの「ノイズ」がある時である．我々は，CとDとを相互に誤って出してしまう確率 $e = 0.1$ を与えて，単一種環境での系の振舞いを調べた．その結果を表3に示す．結果は定性的に，成長，停滞（明確な成長，自滅をせず，ランダムウォーク的な振舞いをする)，自滅に分類される．また，状況を時系列的に表現したものを図1に示す．予想されたことではあるが，厳格なアルゴリズムをもったSLFは同士打ちを始めて自滅し，またTFT，TT3，REFの成績も芳しくない．

表3．ノイズ下での各アルゴリズムの振舞い

CCC	成長		REF	成長
DDD	自滅		RF1	成長
RAN	自滅		SLC	停滞
TFT	自滅		SLF	自滅
TT3	停滞			

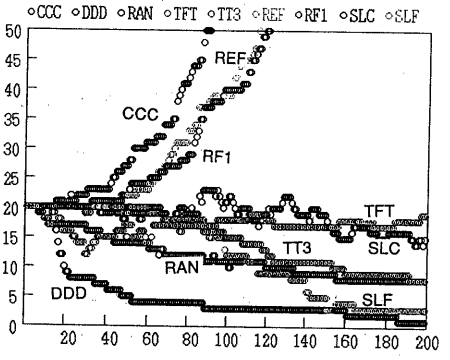


図1 ノイズ下での各アルゴリズムの振舞い

4.2 一対一対戦

次に，それぞれのアルゴリズムについて一対一（各エージェント20個づつ）の対戦を行なわせて，相対的な「強さ」を推定した．この場合にも，協調

的アルゴリズム同士ではCを出し続けるだけである．従って，ここで興味があるのは，各協調アルゴリズムと非協調アルゴリズムとの対戦である．実験結果を表4に示す．また例として，RAN-TT3の対戦の様子を図2に示す．表中,「勝利」は相手の非協調アルゴリズムを駆逐したことを，また「敗退」は逆に自分が駆逐されたことを，また「共倒れ」は両者がお互いに滅ぼしあったことを表す．

表4．各アルゴリズムの一対一対戦

	DDD	RAN
CCC	敗退	敗退
TFT	共倒れ	共倒れ
TT3	勝利	共倒れ／勝利
REF	勝利	勝利
RF1	勝利	勝利
SLC	勝利	勝利
SLF	勝利	勝利

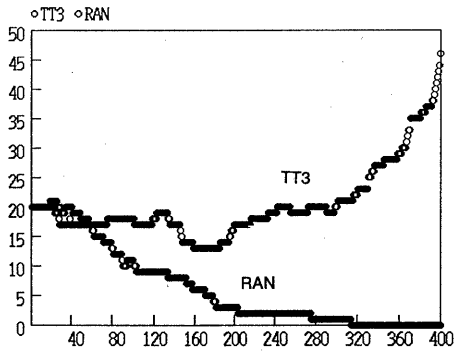


図2 RAN-TT3戦の振舞い

4.3 競合環境1 (CCC-DDD/RAN-XXX)

非協調アルゴリズムを含む3種以上のアルゴリズムの競合環境は，興味深い．非協調的アルゴリズムの存在の下では，戦闘的アルゴリズムはDを出して戦わなければならない．この時，相手の対戦履歴だけからでは，敵と味方の区別が難しくなる．表5には，戦闘的協調アルゴリズムがCCC，DDD（またはRAN）との競合環境で，どのような振舞いをするかを示した．表中，DDDはCCC-DDD-XXXの対戦を，RANはCCC-RAN-XXXの対戦を表す．

表5．CCC-DDD/RAN-XXXでの振舞い

XXX	DDD	RAN
TFT	共倒れ	敗退 (RAN 自滅)
TT3	勝利	敗退 (RAN 自滅)
REF	勝利	敗退 (RAN 自滅)
RF1	勝利	勝利 (CCC と共存)
SLC	勝利	勝利 (CCC と共存)
SLF	勝利	勝利

CCC-DDD-XXX戦では，早い段階でCCCはDDDに駆逐されてしまう．その後，TFTではDDDと共倒れとなり消滅するが，それ以外のアルゴリズムではDDDを駆逐するのに成功する．

CCC-RAN-XXX戦では，TFT/TT3/REFアルゴリズムはランダムアルゴリズムRANをうまく検出できず，敗退する．一方，勝利したRANはお互いに裏切りあって自滅する．RF1，SLCでは，RANを駆逐した後CCCと共存して，成長する．これに反して，SLFではCCCをも駆逐して単独で成長する．例として，CCC-RAN-TT3戦の様子を図3に示す．

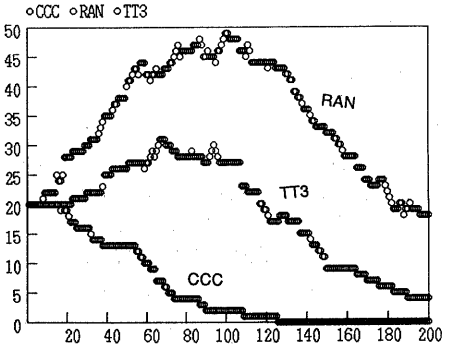


図3 CCC-RAN-TT3戦の振舞い

4.4 競合環境2 (RAN-XXX-YYY)

次には，非協調アルゴリズムの存在下で，異種アルゴリズムがどのように協調するかを見るため，RANと2種の戦闘的協調アルゴリズムとの競合環境を作ってみる．その結果をまとめると，次のようになる．いずれの場合にも，RANは他の2種の戦闘的協調アルゴリズムにより急速に駆逐される．

しかしながら，RANとの戦いで明らかになった各アルゴリズムの振舞い方の違いから，SLCは自分より悪い手を出すアルゴリズムを，SLFでは自分と異なる手を出すアルゴリズムを駆逐しようとする．

それに対して，TFT，TT3，REF，RF1 は有効な対抗策を持っていないため，SLC，SLF に駆逐されてしまう．一方，TFT，TT3，REF，RF1 同士では共存可能であり，お互いに協調して成長する．例として，RAN-TT3-REF，RAN-RF1-SLC の振舞いを図 4，図 5 に示す．

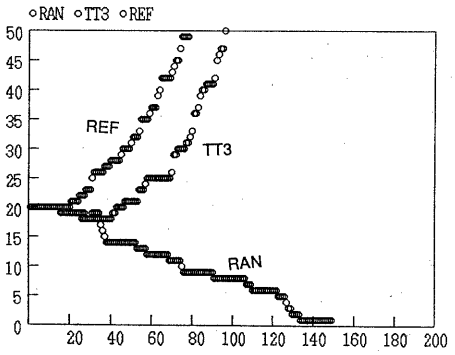


図 4 RAN-TT3-REF 戦の振舞い

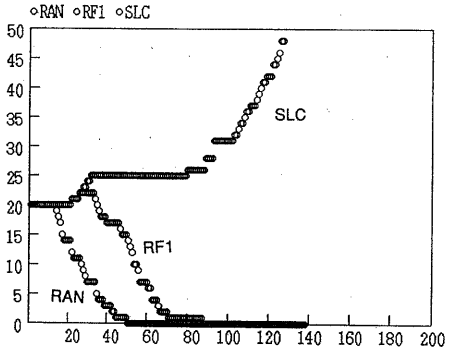
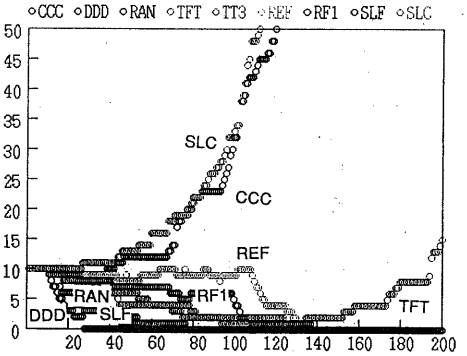


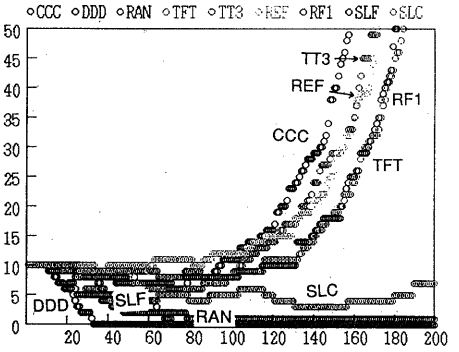
図 5 RAN-RF1-SLC 戦の振舞い

4.5 全種競合

最後に，ここで取り上げた全てのアルゴリズムを一斉に競合させてみる．何回かの試行の結果得られた，定性的性質は次の通りである．全ての場合について，非協調アルゴリズム (DDD，RAN) は急速に駆逐される．次に，排他的な SLF が皆に戦いを挑んで，駆逐されてしまう．TFT/TT3/REF/RF1 グループは相互に協調できるが，SLC とは協調できない．CCC はいずれのグループとも協調できる．



(1) SLC 勝利



(3) TFT/TT3/REF/RF1 勝利

図 6 全種競合

結局，安定解は以下のいずれかになる．

(1) SLC が勝利して，CCC を除く他のアルゴリズムを駆逐する．

(2) TFT/TT3/REF/RF1 グループが勝利して，SLC を駆逐する．

(3) CCC が先に成長することで，SLC と TFT/TT3/REF/RF1 グループが休戦状態となり，お互いに成長する．

実際には，ランダム揺らぎによりどちらが先に成長するかで，最終的に (1)(2)(3) のどの状態に落ち着くかが決まる．(1)(3) それぞれの例を図 6 に示す．

5 考察

我々は，いくつかのアルゴリズムについて，シミュレーションによりその性質を調べてみた．一つの興味ある結果は，「しっぺ返し」戦略がそのままの

形では，Axelrod らの行なった反復囚人のジレンマ[1] の時のようには，うまく動かないということである．最大の理由は，一旦何らかの理由でしっぺ返しが始まってしまうと，たとえ「しっぺ返し」だけの集団であっても，ロメオとジュリエットの悲劇のように，際限のないしっぺ返しの嵐を逃れる方法がないということにある．

それを避けるために，我々は TFT から TT3, REF と改良版アルゴリズムを作ってきた．REF は，相手が協調できる仲間であるかどうかを判定するために，自己のアルゴリズムを再帰的に使うことで，同士打ちを回避する．しかしながら，このような考えを押し進めていくと，自分と全く同じ振舞いをしないものは仲間ではないという排他的アルゴリズム SLF に行き着く．

排他的アルゴリズム SLF は，他のアルゴリズムと同数で渡りあった時には極めて強いが，全てのアルゴリズムを敵に回してしまうため，全種競合実験では敗退してしまう．そこで，2 つの改良版が考案された．一つは，SLF の排他性を緩和した SLC，もう一つは TFT/TT3/REF との協調関係を壊さない範囲で仲間判定を厳しくした RF1 である．その結果，4.5 で示した (1)〜(3) という安定状態が生じることとなった．

なお，(2) で SLC と TFT/TT3/REF/RF1 との平和共存が可能となったのは，早い段階で非協調アルゴリズムが駆逐されたことで CCC が成長し，世界が見かけ上「平和」となったからである．もし再び，何らかの理由で非協調アルゴリズムが侵入するようなことがあると，SLC と TFT/TT3/REF/RF1 とは，お互いに相手を駆逐しようとして戦うことになる．

結局，このような状況下で「最適な」アルゴリズムというものは，どうも存在しないようである．実際，あまり強く「仲間性」を検査すると排他的になってしまい，多くのアルゴリズムを敵に回すことになる．逆に寛容であり過ぎると，相手に付け込まれてしまう．この辺りは，人間社会の協調のあり方とも似て，興味深いものがある．

6 まとめ

我々の研究が，これまでの囚人のジレンマの研究や協調の理論と大きく異なるのは，エージェントが同一の相手と無限回（不特定回）取引を行なうのではなく，一回限りの取引を次々と相手を変えながら行なうという設定にある．もちろん，このような状況では，一般に裏切り（D）が最適戦略となってしまう．そこで，我々は裏切りに対する社会的制裁機構として，取引履歴の公表を採り入れることにした．これは，実社会の取引においても，不公正な取引に対する制裁として，「信用」，「評判」がそれなりに有効に作用していることをモデル化したものになっている．

また，最適な取引戦略を探索するために，いくつかの取引アルゴリズムを作成し，それらを相互に対戦させることで，各アルゴリズムの性質を調査し，最適な取引戦略の条件を探ってみた．その結果，自己のアルゴリズムを用いて相手の行動を推測し，それほど寛容でもなく，それほど排他的でもない SF1, SLC の近くに，最適なアルゴリズムがあるのではないかという予測を得た．

自律エージェントの行動規範の問題は，これまで多くの場合，まず様々な「規則」を導入し，エージェントは必ずその規則を守るものだという仮定が採られていた．しかしながら，人間社会を見てもわかるように，自律性は規則をも破る自由をエージェントに与える．工学的に見ても，どうしても必要な時には規則を無視して，社会的により大きな利益を追求すべき応用は多い（例えば，急病人を運ぶためにスピード違反をする）．もちろん，規則違反を無条件で放置していては社会が成り立たないわけで，ここで提案した，情報公開（取引履歴公開）による社会的制裁というアプローチが有効な領域は多いと思われる．

参考文献

[1] R. Axelrod, The Evolution of Cooperation, Basic Books Inc., 1984.

[2] M. R. Genesereth, M. L. Ginsberg, and J. S. Rosenschein: “Cooperation Without Communication”, in IAAA-86, IAAA, 1986.

[3] 大沢英一，沼岡千里，石田亨：「サーベイ：分散人工知能小問題集」，中島秀之編，マルチエージェントと協調計算 I，近代科学社．

[4] K. Lindgren: “Evolutionary Phenomena in simple Dynamics, C. G. Langton (eds.), Artificial Life II, pp.295-312, Addison Wesley, 1992.

[5] J. S. Rosenshein and M. R. Genesereth: “Deals Among Rational Agents”, IJCAI’85, pp.91-99, 1985.