第4回AI若手の集い(人工知能学会主催)
MYCOM2003資料
2003年5月26日，27日

強化学習 Profit Sharing の今後の可能性

植村　渉，辰巳　昭治
大阪市立大学大学院工学研究科電子情報系専攻　知識情報処理工学研究室
〒558-8585　大阪府大阪市住吉区杉本町 3-3-138
wataru@kdel.info.eng.osaka-cu.ac.jp,tatsumi@info.eng.osaka-cu.ac.jp
http://www.kdel.info.eng.osaka-cu.ac.jp/~wataru

Abstract: 自律したエージェントの実現にはいくつもの問題が存在する．代表的な問題として「フレーム問題」と呼ばれる曖昧さの処理の問題がある．その問題に対して，著者が注目している方法として，強化学習法の 1 つである Profit Sharing 法を紹介する．この手法は学習解に最適性が保障されず，準最適解を学習する可能性があるが，学習にかかる時間が短いという特徴がある．この特徴は，フレーム問題の特徴と類似しているため，解決の手がかりを見出すことができないか検討する．また，Profit Sharing 法の現在の問題点として，学習できる行動系列の長さに制限がある．そこで，行動系列の長さに依存しないほう法を提案し，効果を迷路タスク走行問題にて確認する．
本論文では，まず，フレーム問題について紹介し，その特徴に類似する Profit Sharing 法を紹介する．そして，Profit Sharing 法の現在の問題点をあげ，解決策を提示する．

1. はじめに

これからの人工知能として，人間との対話や協調などを可能とするためには，物事を自分で判断する力が求められる．しかし，実現への道のりは険しく，様々な問題が存在する．一部には，実現の見通しはなく，夢物語だと断言され，「現代の錬金術」とも言われている[Dreyfus 65]．また，たとえ曖昧な思考が実現できたとしても，正確さを失ったコンピュータがどこまで必要とされるかは難しい問題である．これらの議論についてはグループディスカッションに期待し，本稿では保留する．

曖昧さの実現を阻む問題の一つとして，状態数の爆発問題がある．最適解を計算するためには全パターンを計算した上で見つけるのが一番確実である．しかし，莫大な計算時間がかかる．例えば，普通のゲームでも，五目並べで 10^{105} 通りの状態数，チェスで 10^{50} 通り，囲碁では 10^{172} 通りの状態数にも及ぶ[松原 95]ため，全状態数の計算は不可能である．現実問題を厳密に扱うと　さらに状態数が多くなる(というよりは，時間軸を状態に含めると同じ状態が存在しないため無限になる)．このような状態数の問題に対しては，必要のない状態を切り捨てて，必要な状態だけをいかに切り取るかがポイントとなる．「曖昧性」が要求されるといえる．この問題は具体的にモデル化されておらず「フレーム問題」として漠然と扱われている．本稿ではまずフレーム問題を取り上げ，その特徴を紹介する．

次に，この問題に対して強化学習法の一つである Profit Sharing 法の特徴に着目する．強化学習法とは試行錯誤を基にした学習方法である．Profit Sharing 法では，学習解に最適性が保障されず，準最適解を学習する可能性がある．しかし，解の質を落とす変わりに学習にかかる時間が短いという特徴がある．この特徴は，フレーム問題のキーとなる曖昧性と類似している．将来的にフレーム問題を解く鍵となるのではないかと筆者は着目しているが，現段階では関連研究は行われていない．Profit Sharing 法には現在学習結果に一貫性をもたせるため合理性定理[宮崎 96]が提案されている．しかし，この定理に従うと複雑な学習ができないという問題点があり，そのため複雑な問題環境には適用されなかった．本稿では，その問題点の改善のため新しい定理を提案し，複雑な問題環境でも学習できる Profit Sharing 法を提案する．

2. フレーム問題

現実社会でエージェントが自律するためには，曖昧な表現を理解し行動しなければならない．代表的な例題としてデネットの物語がある．詳しくは，[羽地 98]や[人工知能 web]に書かれている．問題環境は図 1 の通りである．ある部屋の中に，予備のバッテリーがワゴンにおいてある．しかし，このワゴンには時限爆弾も一緒に置いてある．ロボットは爆弾をどけて，ワゴンを押して部屋から出てこなければならない．最初のロボット R1 はワゴンを押すことが爆弾にどう影響するか考えなかったために，爆弾ごと動かしてしまい失敗する．次に演繹(deduce)できるロボット R1D1 を作る．このロボットはワゴンを動かすことで他にどう影響を与えるかを考え出したが，「他」 の対象物が多すぎたため時間がかかりすぎて爆弾は爆発してしまう．そこで R2D1 (robot-relevant-deducer：分別のある演繹ロボット) を開発した．R1D1 は影響の受けない対象を全て考えていた

ため，このロボットはそれを省いて必要なものだけを考慮するようにした．しかし，影響の受けない対象を省く動作も対象物が多すぎたため時間がかかりすぎて，爆弾は爆発してしまう．R2D2 は物事を的確に判断しすばやく行動できるロボットである．つまり，R1D1 や R2D1 は，対象物を全て計算してしまうため計算時間が実時間で収まらなくなってしまっている．それに対して，R2D2 は自分に必要な事柄のみを対象として，計算を行う．

別の例を考えよう．二年前のサマースクールにて日本福祉大学の山羽先生のお話に登場したのだが，「手を動かしてコップを取る動作」を考える．手を動かす動作に関係する関節は肩と肘にある(手首は無視する)．次元数よりも自由度の方が大きいので，コップを取るための解は多数存在する．その中から人間は無意識の内に一つを選択して，行動を起こしている．このような動作も現在のロボットではかなり難しい．わざと自由度を落として，解を導くことで解決している．元の自由度のままでは有限時間内に解は見つからない．

これらの問題には，莫大な状態集合から解を素早く見つけ出さないといけない特徴がある．この時に求められる解として，最適解ではなく 少々質が落ちた無駄のある解でも問題がない．曖昧性という言葉で集約できるこの問題は，解決への道のりは未だに見えてこない．

3. 強化学習法

強化学習法は，試行錯誤を基にした学習方法である．つまり，エージェントが与えられた問題環境下で試行錯誤を繰り返し，よりよい解を自ら見つけてゆく方法である．ユーザはゴールの状態だけを記述すれば良い．エージェントがゴールに到達すると報酬 r を受け取る．エージェントはこの情報を用いて学習を行う．ここで，エージェントは問題環境に対する予備知識がないため，この報酬値の絶対的な意味はわからない．つまり，これより良い報酬が他に存在するかもしれないし，この報酬が一番良いかもしれない．ここで，強化学習独特の問題が存在する．より良い報酬を発見するためには環境を幅広く探索する必要があるが，その結果無駄な行動を多く含んでしまう．逆に，報酬獲得を優先すると，より良い解を発見できなくなってしまう．前者の環境同定型の強化学習はその性質上，環境がマルコフ性を満たしていないと性能を発揮できない．後者の報酬獲得優先の強化学習は，報酬を獲得する手順を強化するため，環境の緩やかな変化にも対応できる．Profit Sharing は後者の経験強化型の強化学習である．

3.1 経験強化型強化学習とフレーム問題の特徴点

環境同定型の強化学習は，環境を全て試行することで最適解を見つけ出す．最適性が保障されているため様々な研究がされている．しかし，複雑な環境の場合，全試行を実現するのに有限時間で終了しないため適用が難しい． それに対して経験強化型の強化学習は，報酬を獲得できる行動を優先して選択するため学習解に最適性は保障されない．そのかわり，学習の立ち上がり速度は速い．また，環境がマルコフ性を満たす必要がなく，一部の非マルコフ性環境でも動作する事が確認されている．

ここで，経験強化型の強化学習の特徴点と，フレーム問題の特徴点とを比較する．共通する特徴点として「全状態を試行しないで，解を出す」，「学習解には最適解の保障がない」が挙げられる．つまり，環境同定型

- 予備のバッテリーが置いてある．
- バッテリーはワゴンの上にあるが，そのワゴンには爆弾が仕掛けられている

図1 デネットの物語 (Dennett1984, p.147f)

の強化学習は，R1D1 や R2D1 のように全状態を試行するタイプであり，経験強化型の強化学習は，R2D2 のような全状態を試行しないで，解決できる解を優先して行動するタイプである．将来的にフレーム問題の解決となる鍵が Profit Sharing には含まれていると考えられる．ただし，両者間には大きく異なった点も存在する．例えば，フレーム問題では失敗は許されないため何度も試行錯誤ができない．また，Profit Sharing を適用する前提として，ロボットがワゴンや爆弾などの対象物を認識して記号化する必要もある．前者に関しては，ロボットの頭の中での試行錯誤(シミュレート)により解決することになる．後者に関しては該当分野の技術の前進を期待する．

3.2 Profit Sharing 法

Profit Sharing の枠組みは次の通りである．報酬を獲得するまでの行動系列をエピソードと呼び，エピソード単位で強化を行う．獲得した報酬 r をエピソード内の各行動選択の評価値に分配する事で強化を行う．この分配関数を強化関数 f と言う．従来の研究では強化関数の設定方法は場当たり的であったが，1994 年に強化関数の定理「合理性定理[宮崎 94]」が提案されて以来，その定理に従った強化関数を用いるようになった．しかしこの定理は，0 への収束が速い関数(代表的な関数として等比減少関数)を要求する為，エピソードが長いと学習速度が極端に遅くなる傾向がある．本研究では，学習の効率を高めた強化関数の条件を提案する．

3.3 従来の強化関数

あるエピソードにおいて同一状態が二回以上存在し，それぞれで別の行動を選択しているとき，状態遷移を考えるとループが存在する．この時，ループへの行動選択よりもゴールへの行動選択を強化すべきである．このようなループを迂回系列と呼ぶとき，常に迂回系列上にあるルールを無効ルール それ以外のルールを有効ルールという．合理性定理は，エピソード内に無効ルールと有効ルールが存在するときに必ず有効ルールを強化するための条件式(1)を与えている．

$$L\sum_{j=i}^{W} f_j < f_{i-1} \qquad \forall i = 1,2,\ldots,W. \tag{1}$$

ここで，W はエピソードにある行動数，L は各行動で選択できる行動数の最大である．

また Profit Sharing では報酬を得ることで学習を行うため，この強化関数 f を用いて学習した結果，報酬へたどり着く必要がある．無限にルールを選択し続けるものをプランとし，単位行動当りの報酬の期待値が 0 でないプランを報酬プランとする時，式(1)を満たす強化関数が報酬プランを学習できることが証明されている．

3.4 提案手法

合理性定理は無効ルールの抑制を保障しているが，迂回系列中のルールは，迂回系列を抜けるための強化が必要であり，抑制の対象とする必要がない．本研究では迂回系列へ至るルールのみを抑制の対象とする．

エピソード内で，同一状態が複数存在し，異なるルールを選択した時，それらのルールに対して目標状態に一番近いルールを非迂回ルール，それ以外のルールを迂回ルールとする．迂回ルールを区切とし，エピソードのルール群を分割する．それぞれの分割したルール群を区間 Z とし，目標状態に近い順に z_1, z_2, …, z_n とする．一般的に区間内のルール群に優越はないので区間ごとに同一の強化値で強化する．区間に対する強化関数として区間強化関数 g を用いる．区間内の強化に減少関数を用いても問題はないが，次の区間の強化値を下回ってはいけない．ここでエピソードの i 番目のルールが有効ルールであり，i+1 番目のルールが無効ルールの時，この i+1 番目のルールは迂回ルールとなる．つまり，無効ルールが一つ以上連続するとき，先頭の無効ルールは必ず迂回ルールとなる．この時，次式(2)を満たす区間強化関数 g が迂回ルールを抑制できることになる．

$$L\sum_{j=i}^{n} g_j < g_{i-1} \qquad \forall i = 1,2,\ldots,n. \tag{2}$$

ここで，n は区間の数である．

以下に，このことを説明する．

(1) 迂回ルールの抑制

迂回系列への遷移があるエピソードを考える．迂回系列内にあるルールの数を n とし，非迂回ルールを用いて迂回系列から抜け出し，目標状態に至るまでのルールの数を ω，区間数を z とする．この時，迂回系列を抜け出る行動(非迂回ルール)の学習には f_ω を用い，迂回系列内の行動の学習には $f_{\omega+1}$, …, $f_{\omega+n}$ を用いる．

ここで，迂回ルールが学習時に一番強化されるのは，迂回系列内の行動強化に用いる強化値全てを用いて強化する時で，値は $\Sigma_{i=\omega+1}^{\omega+n} f_i$ である．この時，非迂回系列を g_{z-1} で強化し，迂回ルールを $\Sigma_{i=z}^{z+n} g_i$ で強化する．この状況で迂回ルールを抑制する必要がある．よって，次式(3)を満たす必要がある．

$$\sum_{i=z}^{z+n} g_j < g_{z-1} \tag{3}$$

ここで，非迂回ルールの数 L が複数の時を考える．一番選択確率の高い非迂回ルール A が，全非迂回ルールの中から選択される確率は 1/L 以上である．この非迂回ルール A が選択された後，最悪 L 回他の非迂回

ルールが選択される場合を考える．迂回ルールを強化した強化量よりも非迂回ルール A の強化量が大きい必要がある．以上をまとめると次の定理を得る．

[定理 1]迂回ルールの抑制

任意の迂回ルールが抑制される必要十分条件は，下記の不等式(4)が成立することである．

$$L\sum_{j=i}^{n} g_i < g_{i-1} \qquad \forall i = 1,2,\ldots,n. \tag{4}$$

ここで，n は区間の数，L は迂回ルールの状態にある非迂回ルールの数である．L はその状態で選択できる行動数とすることで十分である．以後式(4)を迂回ルール抑制条件と呼ぶ．

(2) 報酬プランの獲得

迂回ルール抑制条件を満たす強化関数を用いて学習を行った際，報酬を得られないプランに陥るかどうか検討する．

[定理 2]報酬プランの獲得

強化関数が迂回ルール抑制条件を満たせば，報酬プランを必ず獲得できる．
(証明は付録 A)

(3) 学習に必要とする時間

迂回ルール抑制条件を満たす強化関数を用いて，学習する時の学習時間と環境の大きさの関係について考える．条件を満たす関数を用いて学習すると，その学習対象となる行動系列ではループが抑制される．行動系列にループがない場合は区間数が 1 であるため，定数を用いて強化することが許され，強化関数に環境の大きさが含まれない．Profit Sharing は目標状態に到着して初めて学習が行われるため，学習の立ち上がりの時間は学習初期のランダムウォークに起因する．

例えば，各状態でゴールに近づく解と変化しない解が 1:1 の割合の時，ゴールに到達する確率が 50%となる試行回数は，ゴールまでのステップ数の倍を必要とし，ゴールまでの距離に比例する．つまり，学習時間は環境の解の割合に影響し，解の割合が 1/2 の場合は環境の大きさに比例した時間がかかる．

(4) 従来の手法との比較

従来の合理性定理と本定理の関数クラスの範囲を図 1(a)に示す．本定理は抑制する対象を限定したため，扱える関数の種類が多くなっている．また抑制対象を図 1(b)に示す．有効ルールの存在しない状態での無効ルールの抑制動作が異なる．有効ルールの存在しない状態とは，常に迂回系列上にある状態である．しかし，その状態を起点と考えると，迂回系列から抜け出るためのルールが存在し，そのルールは学習すべきであり抑制する対象ではない．つまり，本定理は迂回系列に陥っても回復するための学習ができ，学習効率がよいことがわかる．

3.5 実験と結果

迂回ルール抑制条件に従った強化関数を用いた学習の効果を，迷路を用いたシミュレーション実験にて確認する．実験環境は図 2 に示す迷路走行タスク[Sutton 98]を用いた．始点(S)から終点(G)までの経路を学習する問題である．乱数系列を変えた実験を 100 回行い，その平均値を実験値とする．最適解の値は，報酬までの最短経路 14 ステップより，10/14　0.714 である．選択できる行動数 S=4 のため，Profit Sharing の強化関数は公比 1/4 の等比減少関数を用い，提案手法の区間強化関数も公比 1/4 の等比減少関数を用いた．また，環境を複雑にした時の性能比較のため，迷路の縦横の長さをそれぞれ 2, 3 倍にした迷路の実験を行った．

結果は図 3，図 4 である．従来の手法では，強化関数として常に減少する関数を用いたが，本提案手法では必要なときだけ減少する関数となるため，その改善効果が学習速度の違いとしてあらわれることが確認できる．

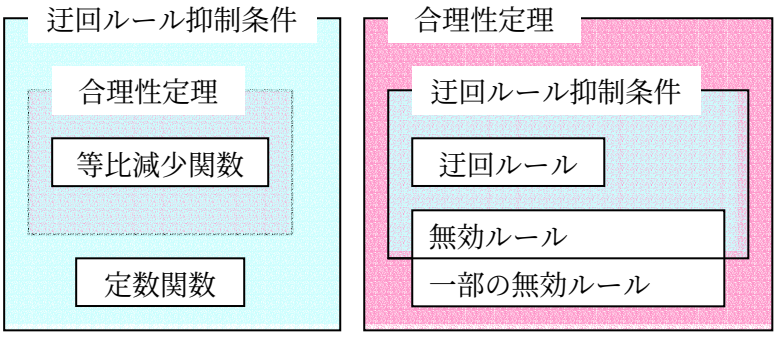

図 1 (a) 定理に従う関数の範囲　(b) 抑制対象

5	11	14	20	26	31	37		G
4	10		19	25	30	36		45
S	9		18	24	29	35		44
3	8		17	23	28	34	40	43
2	7	13	16	22		33	39	42
1	6	12	15	21	27	32	38	41

図 2 実験に用いた迷路環境

環境を複雑にすると従来の手法では学習の効果が確認できないが，本手法では効果を確認できる．収束値の 90%の値に達する行動選択回数は，二倍迷路の時で約 22000 回，三倍迷路で約 45600 回である．状態数の増加に対して収束までの時間が線形的な増加量を示していることが図 5 でわかる．迂回ルール抑制条件を満たす強化関数を用いて迷路走行タスクを学習すると，学習時間は環境の大きさに比例し，安定して学習できることがわかる．

3.6 考察

強化学習 Profit Sharing を用いて学習する際，従来は無効ルールを抑制するために等比減少関数を強化関数に用いた．しかし，強化関数の 0 への収束速度が速いため，環境が複雑になると学習効率が悪くなることがしばしばであった．

本研究では，抑制する対象を無効ルールの先頭である迂回ルールのみに絞る方法を提案した．報酬プラン獲得の条件を満たしながら，強化関数の 0 への収束速度を改善でき，環境の複雑さに影響を受けにくい学習が実現した．

本研究により環境の複雑さの制限がなくなり，今まで状態数が大きすぎて Profit Sharing を導入できなかった問題への適用が今後期待される．

4. おわりに

自律するエージェントを実現するための問題の一つとして，フレーム問題がある．本研究では，エージェントの思考時の曖昧さの問題として捉え，類似する特徴を持つ Profit Sharing 法を紹介した．この手法は学習解に最適性が保障されず，準最適解を学習する可能性がある．そのかわり，学習にかかる時間が短いという特徴がある．この手法の問題点は，学習できるステップ数に制限があることである．そこで，ステップ数に依存しない方法を提案し，効果をシミュレーション実験にて確認した．

5. 参考文献

[Dreyfus 65] Dreyfus,"argues against the possibility of AI". :すみません．論文手に入れていません(´･ω･`)ｼｮﾎﾞｰﾝ

[Grefenstette 88] Grefenstette, J.J., "Credit Assignment in Rule Discovery Systems Based on Genetic Algorithms," Machine Learning, Vol.3, pp.225-245(1988).

[宮崎 94] 宮崎 和光, 山村 雅幸, 小林 重信, "強化学習における報酬割当ての理論的考察" , 人工知能誌, Vol.9, No.4, pp.580-587(1994).

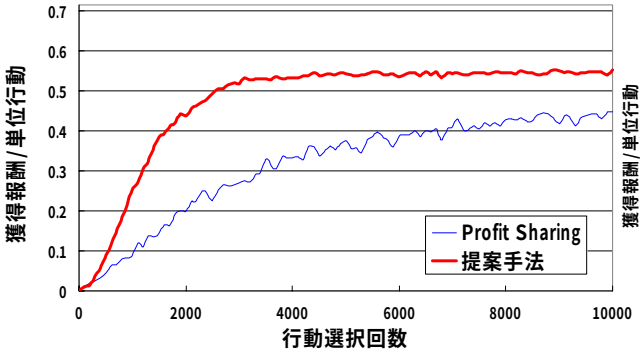

図 3 迷路走行タスク実験結果

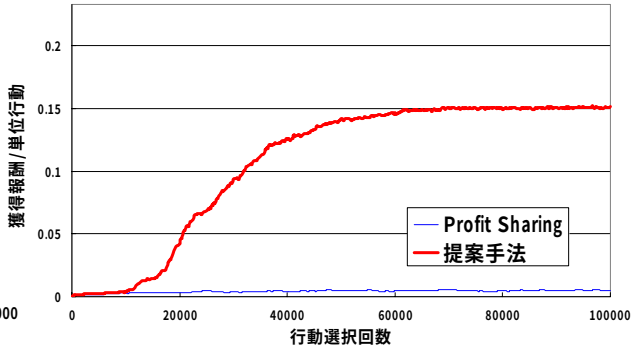

図 4 縦横三倍の迷路における実験結果

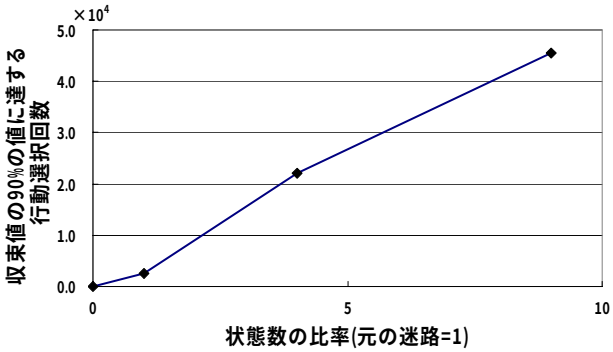

図 5 収束速度と状態数の関係

[松原 95] 松原 仁, “最近のゲームプログラミング研究の動向”, 人工知能誌, Vol.10, No.6, pp.835-845(1995)
[Sutton 98] Richard S.Sutton and Andrew G.barto, “Reinforcement Learning”, The MIT Press.(1988)
[羽地 98] 羽地 亮“「フレーム問題」の解消”, 京都大学哲学研究室, http://www.bun.kyoto-u.ac.jp/phil/pros/01/haji.html
[人工知能 web] 人工知能学会 web ページ, “人工知能の話題”, http://www.ai-gakkai.or.jp/jsai/whatsai/AItopics1.html

6. 付録:定理 2 の証明

[宮崎 94]の証明と同様の環境を用いる. 迂回ルール抑制条件を満足する強化関数を用いて学習し, その学習結果に従って行動した時, 報酬プランとならない場合を考える. 報酬プラントならない場合は, 報酬を伴わないループに陥る時である. このようなループは 2 個以上のエピソードから構成される. ここで有効ルール数 L=2 かつループから脱出できる行動(x_o,y_o)のある状態を x, y の二ヵ所とする(図 6 参照)が, それ以外の場合もまったく同様である. 状態 x, y 間でループを構成するためにはループの出口となる状態において次の不等式群が成り立つ必要がある.

$$\Delta\omega\vec{x}_o<\Delta\omega\vec{x}_i \quad (A.1)$$

$$\Delta\omega\vec{y}_o<\Delta\omega\vec{y}_i \quad (A.2)$$

$\vec{x}_o$, $\vec{y}_o$はそのループから出て行くルール, $\vec{x}_i$, $\vec{y}_i$はそのループ内に戻るルールである. また, Δ はそのルールに加算される強化値の総和を表す. 迂回ルール抑制条件を満足し, かつ$\vec{x}_i$を含むエピソードが$\vec{x}_o$を含んでいたとすると,

$$\Delta\omega\vec{x}_o>\Delta\omega\vec{x}_i \quad (A.3)$$

となり, ループが構成できない. よって, $\vec{x}_i$を含むエピソードは$\vec{x}_o$以外のルール つまり$\vec{y}_o$を使ってループの外へ出る必要がある. $\vec{y}_i$に関しても同様である. 次の不等式群が成り立つ.

$$\Delta\omega\vec{x}_i \square \Delta\omega\vec{y}_o \quad (A.4)$$

$$\Delta\omega\vec{y}_i \square \Delta\omega\vec{x}_o \quad (A.5)$$

等号成立はそれぞれ同一区間にある場合

式(A.1), 式(A.2), 式(A.4)と式(A.5)より, 次の不等式が得られる.

$$\Delta\omega\vec{x}_o+\Delta\omega\vec{y}_i \square \Delta\omega\vec{x}_o+\Delta\omega\vec{y}_o<\Delta\omega\vec{x}_o+\Delta\omega\vec{y}_i \quad (A.6)$$

この不等式を満たす解は存在しない. ゆえに, 迂回ルール抑制条件を満たす区間強化関数を用いた強化関数では, 必ず報酬プランが獲得される.

(証明終了)

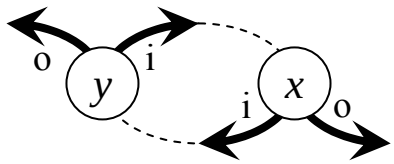

図 6 報酬プラン獲得の証明の環境