論文の内容の要旨

論文題目：ラムダゲーム：メタゲームへのアプローチ
(The Lambda Game System: an approach to a meta-game)

氏名：舛本 現

ゲームにおける戦略の open-end な進化は，人工生命の研究において重要なテーマであり，これまでも様々な研究がなされてきた．これらの研究でその基礎となっているのはゲーム理論であるが，そこではゲームとは一般的にルールが固定されており，各プレイヤーはそのルールが規定する戦略 (選択肢) のなかでどの戦略が自分の利得を最大にするかを考え，選択し，それに応じた利得を得るものとして扱われている．

しかしルールが固定され，そのなかでの最適な行動が存在するのなら open-end な時間発展など不可能であり，あとに残るのは，本当は最適な戦略があるのだが人間の推論能力や計算能力が限られているからそこに到達しえない，という限定合理性の問題だけになってしまう．そこで多くの open-end な時間発展を扱う研究では，非常に限定されたかたちではあるが，戦略の空間に openness を導入した．それはある時は記憶戦略における記憶の長さであり，また各個体のつくる相手のモデルの決まらなさであったりした．

しかし他方にはルールにこそ戦略を open-end に発展させる源があるという見方がある．これはつまりルールが open であることこそが戦略の open-endedness を生み出すという立場である．例えば，社会におけるルールというのは多くの場合，自然言語で記述されており，自然言語であればもちろん解釈は開かれているから，現実の社会における戦略は多様でありえると考える．ルールの openness を導入したモデルでは，例えば Howard の meta-game theory のように囚人のジレンマにおいて「あなたが協調するなら私も協調し，あなたが裏切るなら私も裏切る」といったルールに言及するメタ戦略を (one-shot のゲー

ムでも) 許すような枠組を用いたり，あるいは Hofstadter が紹介した Nomic というゲームのようにすべてのルールを明示的に自然言語で書き，それを更新していくゲーム (「これらのルールを書き換えよ」というルール) を考えたりすることが行われる.

ではルールの openness というのはどのようにして可能になるのであろうか. ルールは戦略の定義域を規定してそれらの組に対する利得を決定し，各プレイヤーはルールの定義域の範囲内の戦略で利得の最適化を図る. このような点において，ここで考えているゲームにおけるルールと戦略というのは，関数と変数のような関係にあるといえよう. 通常のゲーム理論の枠組みでは，変数 (戦略) は決して関数 (ルール) の規定する定義域の外に出ることはできないし，関数 (ルール) も定義域の外の入力 (戦略) に対しては出力 (得点) を与えることはできない.

一方，ここで考えたいルールが開かれているゲームというのは，いわば遊び (プレイ) と呼ぶべきもので，そこでは定義域が開かれているので，ルールは定義域の外の入力に対しても出力を返さなくてはいけない. ここで要求されているのは，定義域を限定せずにどんな入力に対してもとりあえずは出力を返す体系である. そのためには定義域のクラスが関数自体のクラスと同じくらい広く，関数と変数，すなわちルールと戦略，が同じ format で書かれている type-free なものでなければいけない.

そこで本研究ではゲームを記述するために λ 計算を導入する. λ 計算は type-free な計算体系であり，このことはすなわち全ての λ 式が関数と変数のどちらにもなれることを意味している. また λ 計算ではゲームの利得表を記述するのに必要な真偽値や if 文，自然数を表わすことができる.

具体的なゲームとしては各プレイヤーがそれぞれ「協調 (=C)」と「裏切り (=D)」という 2 つの戦略をもつ one-shot の囚人のジレンマゲームを考える. このゲームを構成する要素である「ルール」,「プレイヤーの戦略」,「利得」をすべて λ 式で表すことによって上で述べたような type-free なゲームを構成することができる.

ここでゲームのルールに求められるのは，2 人のプレイヤーが出したこれら 2 つの戦略の組に対して利得となる自然数を返すことである. そこでここでは戦略として Barendregt が導入した真偽値の表現 ($\mathbf{T} = \lambda xy.x, \mathbf{F} = \lambda xy.y$) を用い，自然数の表現として以下のように再帰的に定義される Barendregt 数を用いた. $\mathbf{0} = \lambda x.x$, $\mathbf{n+1} = \lambda x.x\mathbf{F}\mathbf{n}$. 真偽値と自然数が定義されたので，ルールとしての game master の λ 式 (=$\mathbf{G}$) は 2 重の if 文として以下のように構成することができる. $\mathbf{G} = \lambda x.x(\lambda y.y\,\mathbf{3}\,\mathbf{5})(\lambda y.y\,\mathbf{0}\,\mathbf{1})$ (ただしここで $\mathbf{0}, \mathbf{1}, \mathbf{3}, \mathbf{5}$ は Barendregt 数である).

以上のようなゲームの定式化のもとでは，ゲームのルール (=$\mathbf{G}$)，得点，戦略 (=$\mathbf{T}, \mathbf{F}$) がすべて λ 式で表現されている. しかし前述したように λ 計算は type-free な体系なので，ここで定義されたゲームのルール (=$\mathbf{G}$) は元々想定されていた戦略である $\mathbf{T}, \mathbf{F}$ 以外の任意の λ 式も入力として受けつけることができる.

そこで本研究では，まず計算機シミュレーションによりランダムに生成した λ 式を戦略として game master に代入したときにどのような反応が起こるのか，game master は数を返すのか，もし数を返すのならはそれは元のゲームのマトリクスの得点とどのような関係にあるのかを調べた.

その結果として，元々のゲームのルールが用意した元々の利得表にないような利得を獲得する多くの λ 式の戦略がみつかった. すなわちこれらの λ 式が $\mathbf{G}$ に代入されると元の

game master には 0，1，3，5 の各得点しか書かれていないにもかかわらず，それ以外の得点 (例えば 2 点や 6 点) を利得として返す.

これらの戦略は game master の λ 式の if 文の内容 (“if x then A else B” の A や B) を参照することで if 文そのものと干渉を起こし，ゲームのルールを出し抜くことを可能にしている．すなわち自分がこれから代入される λ 式がどのような得点の選択肢を持っているかを判断して異なる if 文には異なる対応をすることによって高得点をとっていることが明らかになった.

さらにいくつかの戦略はルールである λ 式の if 文から「足す 1(successor)」「引く 1(predecessor)」という関数を合成して，元の if 文に書かれている数字に対して操作をほどこしてから自分の得点にするということによって高い利得を得たり，相手の得点を計算するときには定数関数，つまり入力に関係なく一定の数を返す関数を合成することによって相手の得点を低く抑えるような振舞いもみられた.

さらに λ 式の集団を用意して上で定義した λ ゲームでの利得を適応度とする環境で進化させたときにどのような戦略の集団が生まれるかを調べた．ここでは進化オペレータとしては λ 式に遺伝的プログラミング (GP) と同様の手法で変異と交叉をおこなった.

その結果として，ある程度 λ 式の複雑さが増すと，ランダムに生成された λ 式よりもさらに高得点を得る戦略が生まれた．これらの戦略は「足す 1(successor)」という関数を複数個組み合わせることにより高い利得を得ている．またあるタイプの戦略が他の戦略の侵入に対してある程度ロバストであることがわかった.