

リソース保存型人工生命システムにおけるデジタル生物の進化

松崎 周一[†], 鈴木 秀明^{††}, 小佐野 峰忠[†]

本稿では人工生命モデル「ティエラ」を基にした新しい生態シミュレータを提案している。このシステムの特徴はデジタル生物の構成要素となる物質的リソースが保存されている点である。我々は2種類のデジタル生物種を導入して捕食—被捕食の関係をモデル化し、システムの振舞いを観察した。実験では競合する2種の安定した共存関係と適応的な進化が示された。このような種の関係性はシステムの安定性と進化の多様性を生み出すうえで重要な意味を持っていると考えられる。

Evolution of Digital Organisms in a Resource-conservative Artificial Life System

SHUICHI MATSUZAKI,[†] HIDEAKI SUZUKI^{††} and MINETADA OSANO[†]

A new ecological simulator which is based on an architecture of the Tierra system is developed to analyse an effect of inter-species communications. The simulator involves conserved and limited resources which exist as a fundamental material for digital organisms. In an experimental system using two types of digital organisms that each acts as predator and prey, population of both organisms were maintained stabilizing and adapted to each other, whereas they deviated without competitor. The result suggests that communications between predator and prey play an important role in a system to establish stable pattern and evolutionary diversity.

1. はじめに

デジタル生物 (digital organisms) とは、自己複製、突然変異、自然選択、進化といった自然界のプロセスを仮想世界の中で経験するコンピュータプログラムである。近年、このデジタル生物を用いたシミュレーションによる生態系ダイナミクスの分析が注目されている。ティエラ (Tierra) はこれらの研究で最初の重要な成果をもたらしたデジタル生物のモデルである^{1),2)}。ティエラにおける進化シミュレーションでは寄生体と呼ばれる変異種が観察されている。寄生体とは他者の遺伝情報を利用して自己複製する能力を身に

つけている新しいタイプのデジタル生物である。この成果は、自然界とは異なる環境 (つまりコンピュータの仮想世界) でも自然界と同等のプロセスを経験する個体であれば同様の結果が導き出される可能性を提示した最初の例である。またその後のデジタル生物を用いた研究は、自然界の複雑なダイナミクスを分析するための新しいツールとして高く評価されてきている。これらの研究では、デジタル生物を用いたシミュレーションに基づいて、実際の生物進化の複雑性や多様性にも通じる独自の経験に基づく知見が述べられている³⁾⁻⁶⁾。しかし従来の研究の問題点の1つは、本来は生態系の構造に大きな影響を及ぼしていると考えられる、捕食—非捕食の種間関係が考慮されていなかったことである。

そこで本研究では、捕食者と被捕食者の2種類のデジタル生物種を用いてそれらの種間相互作用がシステムに及ぼす影響を分析した。この実験には我々が開発した新しい実験システムが用いられている。従来のモデルと比較して本モデルの特徴を端的に言えば「リソース保存性」である。Suzukiらが指摘しているように、ティエラをはじめとする従来のシステムでは物質的なリソース (すなわちデジタル生物を表現す

[†] 会津大学大学院コンピュータ理工学研究科
Graduate School of Computer Science and Engineering,
Aizu University

^{††} ATR ネットワーク情報学研究所
ATR Network Informatics Laboratories
現在、室蘭工業大学生命ソフトウェアラボトリー
Presently with Life-oriented Software Laboratory,
Muroran Institute of Technology
現在、情報通信研究機構未来 ICT 研究センター
Presently with Kobe Advanced ICT Research Center
(KARC), National Institute of Information and Communications Technology (NICT)

るビットデータ)が実質的に無限のものとして扱われていた⁷⁾。そのためデジタル生物は資源の奪い合いという自然界の基本プロセスを必要としなかった。これに対して我々のモデルにおけるリソースは有限であり、各個体の活動にはリソースを獲得するプロセスが追加されている。その結果、実験では種間競争による特徴的なシステムダイナミクスが現れている。以下ではまずデジタル生物の基本設計ならびに実験システムの特徴を説明する。次に我々の行った実験とそれによって得られた結果について詳述する。

2. モデル

我々は2種類の異なるタイプのデジタル生物(以下では生物とする)を用いた生態シミュレータを提案する。このシミュレータがたどる基本プロセスは「自己複製 → 突然変異 → 選択」という自然現象である。各生物は機械語命令によって記述された自己複製プログラムである。生物のプログラムの命令実行にはそれぞれ均等にCPU時間が割り当てられる。またこの仮想世界には1つのメモリ空間があり、生物つまり自己複製プログラムのコードはすべてここに書き込まれている。自己複製とは生物が自分と同じプログラムをメモリ空間の異なる場所にコピーしていく作業である。突然変異とは、自己複製するプログラムの一部がランダムに低い確率で異なるパターンに変化することである。また選択とは、一定のルールに基づいて不適切な生物をシステムから削除する操作である。これらの基本設計の多くはティエラのデザイン^{1),2)}を継承している。本章では主に本モデルで新たに追加もしくは変更された部分について述べる。

2.1 自己複製

生物の自己複製活動は、まずコピーに必要なリソースを獲得する作業から始まる。ここでのリソースの最小単位とは情報の最小単位、すなわち1ビットである。シミュレーションにおいてシステムに存在するリソースを追加、削除することは禁止されている。自己複製において、生物はデータをコピーするために必要なリソースをメモリ空間から取得して再利用している。メモリ空間にあるリソースは、シンボルとよばれる6ビットの集合体として扱われる。シンボルはそのビットパターンによって、命令語として扱われるパターンとそれ以外のパターン(非命令語)の2種類に分類される。生物がシンボルを獲得する方法は生物の種類(植物種と動物種)によってそれぞれ異なる(2.1.1および2.1.2項を参照)。必要なリソースを獲得すると、生物は他の生物が書き込まれていない空間にコピーを

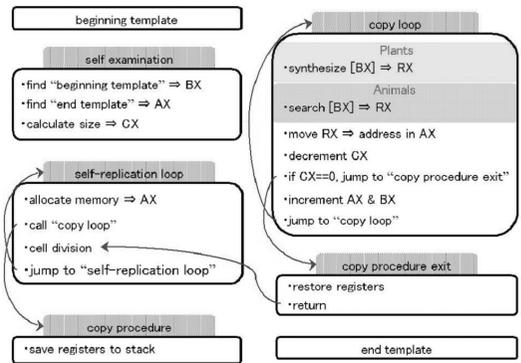


図1 自己複製プログラムの概要

Fig. 1 Abbreviated program of digital organisms.

書き込むことができる。メモリ空間上のアドレスにすでに他のプログラムが書き込まれているかどうかは、フラグを用いて自動的に識別されている。自己複製プログラムと実行手順の概要を図1に示す。自己複製処理の本体である“copy loop”の一連の動作では、まず必要なシンボルをメモリ空間から取得して自己のレジスタRXへいったん格納する。次にそのシンボルを複製先のメモリ空間へ移動させる。なお、本モデルで用いられる自己複製プログラムの先頭と最後には膜テンプレートと呼ばれる特殊な命令語のシーケンスがある。これは、捕食関係において他の生物がこの生物(プログラム)を同類と見なすかどうかを判定するために用いられるパターンである(2.1.2項を参照)。

2.1.1 植物種

植物種は、メモリ空間にある非命令語のシンボルを変換して命令語を生成し自己複製に利用する。植物先祖はメモリ空間からランダムに2つの非命令語を取得して、その一方のシンボルをコピーする命令語のシンボルに変換する。たとえば、取得した2つの非命令語が000001ならびに010100、またこの生物がコピーする命令語のパターンが001101だとする。このモデルではリソースが保存されているので2つの非命令語は001101ならびに000000のシンボルに変換される。このプログラムの実行手順を図2に示す。図中の1~7が示す処理は以下のとおり。

- 1: 複製する自分の命令語をチェックする。
- 2と3: メモリ空間で利用されていない2つの非命令語をそれぞれランダムに取得する。
- 4: それらの非命令語に含まれるビット(リソース)を用いて、複製する命令語と同じ6ビットパターン(Productシンボル)を生成する。
- 5: 上の処理で使われなかった6ビットをランダムに組み合わせてもう1つの6ビットパターン(By-product

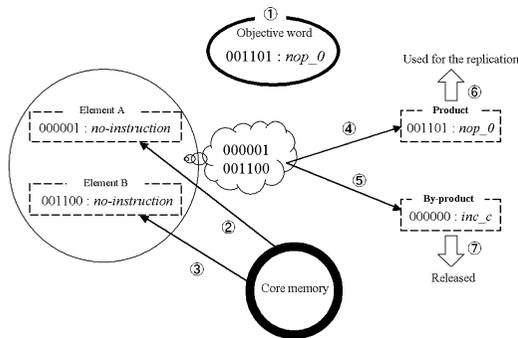


図 2 植物種の自己複製プロセス
Fig.2 Self-replication process of plants.

シンボル) を生成する。

6 : Product シンボルを自己複製先に割り当てられているメモリ空間に書き込む。

7 : By-product シンボルをメモリ空間のランダムに選択した場所へ書き込む。

2.1.2 動物種

動物種は、自分がコピーする命令語と同一のシンボルを直接取得して自己複製に利用する。この処理では自分から前後 350 以内のメモリ空間が探索の対象となる。動物種はここで見つけたシンボルを基本的に獲得することができる。ただしそのシンボルが他の生物の一部だった場合(それによって相手の生物はプログラムの一部が失われてしまうので)それが自分と近い生物だと判断すれば取得しない。この判定はその動物における「CPU 状態」によって行われている。CPU 状態は 2 値の変数 (true/false) であり、この値が false の生物はたとえシンボルを見つけても取得しない。探索開始時点で生物の CPU 状態は false にセットされている。探索中に「テンプレートマッチ」が起きた場合にこの値が切り替わる。テンプレートマッチはティエラで用いられている手法^{1),2)}を基にしている。テンプレート照合の元となるパターンはその生物の自己複製プログラムの中にあるサーチテンプレート(4 命令語からなるテンプレート)である。このサーチテンプレートと照合する対象は、各プログラムの先頭と最後にあるテンプレート(膜テンプレート)である。ここでのテンプレートマッチが従来の手法と異なるのは、相補的なパターンどうしだけでなく同一なパターンどうしであってマッシュしたと見なされる点である。たとえばサーチテンプレートが “NOP1, NOP0, NOP1, NOP0” の場合にマッシュする膜テンプレートは “NOP1, NOP0, NOP1, NOP0” もしくは “NOP0, NOP1, NOP0, NOP1” である。ただし、テンプレ

表 1 テンプレートマッチのルール
Table 1 Rules of the template matching.

	Identical matching	Complementary matching
Forward search	False	True
Backward search	False	True

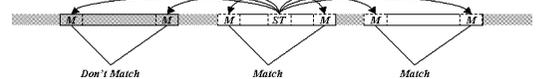


図 3 動物の自己複製における個体間相互作用
Fig.3 Inter-creature communication in animals self-replication.

トマッチによって CPU 状態が切り替わるパターンは、マッシュした種類(同一か相補か)ならびにサーチした方向によって決まっている(表 1)。図 3 は動物の自己複製活動における他の個体との相互作用パターンを示している。中央にある動物の自己複製プログラムがメモリ空間を探索しようとした場合、色付けされた空間にあるシンボルのみを取得する。この動物の CPU 状態は、探索中に自己のサーチ・テンプレート(ST)が膜テンプレート(M)と照合することで切り替わる。この図では右側の生物プログラム膜テンプレートとマッシュしている。このように動物は自分と同等の膜テンプレートを持っている他の生物のリソースは取得しない。すなわち膜テンプレートのパターンが種を判別する要素になっていることが分かる。

2.2 リソースの性質

シミュレーションの実行中に 100 step を超える期間にわたって利用されなかったシンボルは自動的に非命令語に変換される。本モデルの仮想世界では、命令語は有機的、非命令語は無機的な性格を持った物質と見なせる。そのためこの操作は自然界における異化(有機化合物が無機化合物に分解される作用)に近いプロセスである。

また、デジタル生物にとって利用可能なリソースが、リソース全体の 20%未満まで減少した場合にはそれ以上のリソースを生物が利用することができない。このルールは、自己複製に必要なメモリ空きを維持するためのものである。もし生物が自由に増殖し続けてしまうと、自己複製するためのメモリ空間が不足してシミュレーションが停止する可能性がある。

2.3 デジタル生物の死

生物が死亡する(CPU 時間が割り当てられなくなり、プログラムが実行不能になる)条件は次の 2 つのケースが考えられている。まず CPU が長期間(3,000 steps 以上)にわたって命令実行に失敗したときであ

る．これは主にプログラム異常によって命令ポインタが自分のプログラム以外のメモリ空間を指している場合に発生する．また，100 steps にわたって一度も自己複製しなかった生物は 10 分の 1 の確率で死亡する．これはプログラム異常もしくはリソース不足によって，必要なリソースを取得できない状態が続いている場合に発生する．

3. 実験

3.1 2 種類のデジタル生物種を用いたシステム
シミュレーションの初期状態では，上述した 2 種類のデジタル生物種の先祖と 65,000 のシンボルがメモリ空間に書き込まれている．図 4 はそれぞれの種の個体数変化を示している．実験を通して植物種の個体数は動物種の個体数を上回っていた．また両者の個体数はつねにわずかな振動を繰り返しているが，定性的には一定の安定状態を維持する傾向が見られた．

図 5 は，生物の膜テンプレートならびにサーチテンプレートの変化を示している．上述のように，これらのテンプレートは動物が他者のリソースを取得するためのテンプレートマッチに用いられている．図 5 (A)，(B) は，それぞれ膜テンプレートとサーチテンプレートのパターンについて集団のエントロピーを算出した結果である．1 つの生物種のエントロピーは次の式から求めた．

$$H = - \sum_i^n P_i \log P_i \quad (1)$$

ここで P_i はテンプレート i がそのときの群集が持っているテンプレート全体に占める割合である． H の値が大きいほど集団内のテンプレートが多様であることを意味する．膜テンプレートにおけるエントロピーは両種ともほぼ一定の値を維持する傾向が見られ

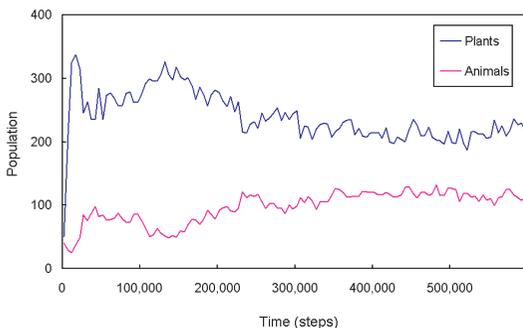


図 4 2 種類のデジタル生物集団の個体数変化

Fig. 4 Transition in the number of digital organisms in two groups.

た．また動物種が持っているサーチテンプレートのエントロピーも初期状態では低い値だったが，実験が進むにつれてより高い値を維持する傾向が見られた．これらの点から群集は突然変異を通して一定の多様なテンプレートを維持する傾向が見られた．次に，我々は膜テンプレートの量的な変化を表すための指標である Matched Template Number (MTN) を導入した．MTN とは，ある膜テンプレートがマッチ可能な 4 命令語のテンプレートの数を表す値である．この値は膜テンプレートを構成する命令語の長さが増加・減少するに従って大きく変化するため，群集内の平均的な膜

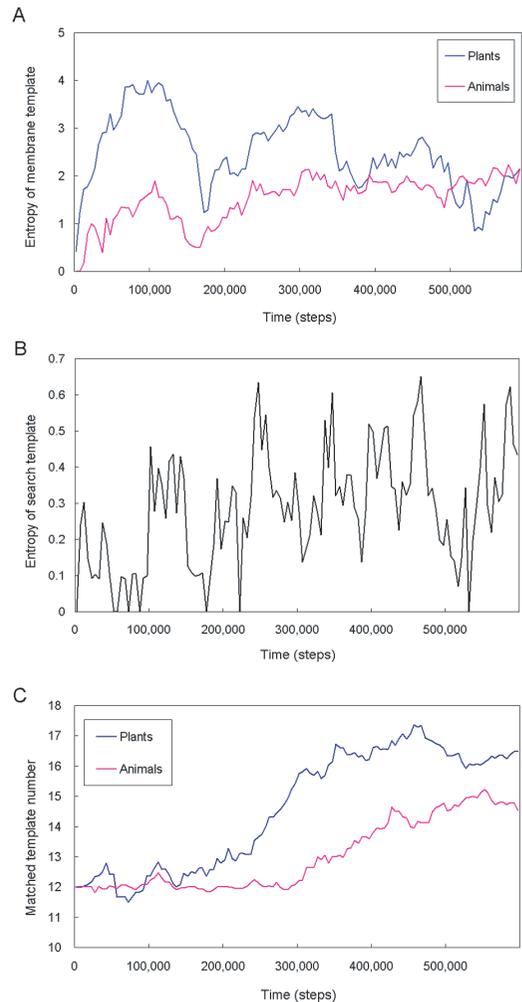


図 5 植物種ならびに動物種におけるテンプレートパターンの変化．A：膜テンプレートのエントロピー．B：サーチテンプレートのエントロピー．C：MTN の平均値

Fig. 5 Transition in template patterns in the group of plants and animals. A: The entropy of the membrane template. B: The entropy of the search template. C: The average of matched template number (MTN).

テンプレートの長さを測るための1つの指標といえる。例として長さが5の膜テンプレート“NOP0, NOP1, NOP0, NOP0, NOP0”を仮定すると、マッチ可能な4命令語のテンプレートは次の4種類になる。

NOP0, NOP1, NOP0, NOP0
 NOP1, NOP0, NOP0, NOP0
 NOP1, NOP0, NOP1, NOP1
 NOP0, NOP1, NOP1, NOP1

すなわちこの膜テンプレートのMTNは4である。重要な点として、MTNが高くなるほどテンプレートマッチが起こる可能性が高くなり、その結果他の動物からプログラムの一部を奪われる可能性が低くなる。しかし一方でプログラムが冗長になるために自己複製にかかるコストが高くなる。図5(C)はMTNを全個体について調べ、集団平均をとった結果である。シミュレーションでは生物集団におけるMTNの平均値は増加し続ける傾向が見られた。

3.2 一方のデジタル生物種を除いたシステム

上述した2種類の先祖のうち、初期状態で一方だけを用いた場合の実験結果を示す。

3.2.1 動物種を除いた実験

植物種だけを用いた実験システムでは、捕食による死が起こらないために個体の適応度にかかわらず生き延びる可能性が非常に高くなった。その結果これまでの条件に基づくシミュレーションでは世代交代がほとんど見られなかった。そこで、突然変異によってプログラムの長さが異常(10以下または500以上)になったものや、正常な自己複製を行えない個体(非命令語を含んでいるものなど)を逐次チェックして削除しながら実験を進めた。その結果、植物種のMTNは世代交代を通して通常の実験のときとは逆の方向へ進化する傾向が見られた(図6)。しかし一方で、同一の実験でも途中から動物種を追加すると(図6の破線)MTNが増加する方向に転じた。すなわち、植物種のMTNの変化は動物種の存在によってまったく逆の傾向を示した。

3.2.2 植物種を除いた実験

動物種だけを用いた実験システムでは、メモリ空間にある命令語のシンボルが消費しつくされると自己複製が見られなくなった。我々は動物が命令語を探索するメモリ空間の範囲を1,500まで拡張し、さらにシステムにある命令語を非命令語に変更するルール(2.2.1項)を排除するなどの操作によって動物種の生存により有利な環境を設定した。その結果、最初是一定の自己複製が行われたものの途中からやはり自己複製がまったく行われなくなった(図7)。すなわち、シ

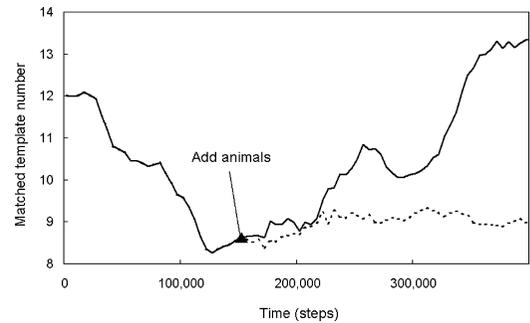


図6 動物種を除いたシステムにおけるMTNの平均値
 Fig. 6 Transition in the average of MTN in the system without animals.

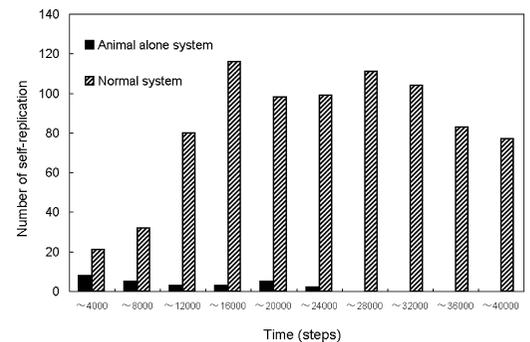


図7 植物種を除いた実験ならびに2種を用いた実験における自己複製回数(4,000 stepsごとに集計)
 Fig. 7 The number of self-replications in the system without plants and the system with plants and animals aggregated every 4,000 steps.

ステムから植物種を除くと動物種の活動も維持できなくなった。

3.3 単純複製体を追加したシステム

ティエラでは通常の生物よりも非常に速く増殖することができるような種(単純複製体)を与えると進化の多様性が失われる。ティエラと異なる人工生命システムでは、リソースを有限にした場合に従来よりも進化の多様性が向上する実験結果が報告されている⁹⁾。そこで、本モデルで進化の多様性が向上したことを調べるためにティエラとの比較実験を行った。実験では2つのシステムに同様の単純複製体を挿入した後の個体数変化を観察した(図8)。単純複製体は20個の命令語だけで作られた自己複製プログラムである。単純複製体の自己複製にかかる時間は通常の生物の約6分の1である。ティエラで単純複製体を挿入した場合その個体数は最大限まで増加し、それ以外のデジタル生物は失われた。しかし本モデルでは、単純複製体の個体数はほぼ一定の状態にとどまり、既存の生物種の個体数も維持されていた。

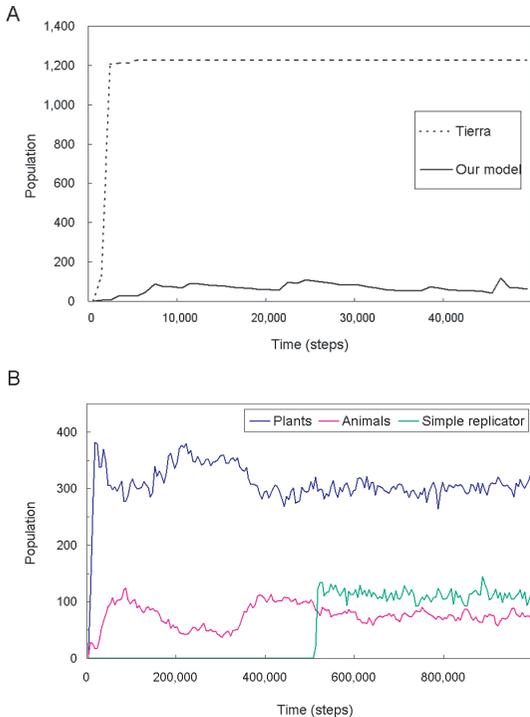


図 8 単純複製体を追加したシステムの変化．A：ティエラと本モデルにおける単純複製体の個体数．B：本モデルにおける植物種，動物種，単純複製体それぞれの個体数

Fig. 8 Transition in experimental systems in which simple replicators are inserted. A: the number of simple replicators in Tierra and our model. B: the number of plants, animals and simple replicators in our model.

4. 考 察

4.1 群集の安定性と進化の多様性

本モデルでは与えられた 2 つの生物種の個体数が互いに安定し、それと同時に互いに進化を続けた。しかし 1 つの種だけがいる環境ではこのようなパターンは生まれなかった。これは実際の生態系¹⁰⁾と同様に、食物関係による種間相互作用が群集の安定性や進化の複雑性に寄与していると考えられる。それぞれのデジタル生物種の振舞いは、システムにある 2 種類のシンボルに関して次のような役割を担っている。

植物種：非命令語から新たに命令語を生産する。

動物種：既存の命令語を消費する。

必要なリソースを獲得するという点からこれらの種は互いに依存関係にある。このように異なる個体集団の間に存在する関係性がシステムの安定性をつくりだす 1 つの要因になっていると考えられる。

また、自然界で捕食関係にある生物種は互いに適応を繰り返す改良進化を続けていると考えられている。

このような進化は主に捕食関係を有利にする特定の機能の中に見られる。これと同様のプロセスは本モデルでも観察された。すなわち生物は膜テンプレートのマッチパターンを増加させることによって他者から捕食される危険性を減らしている。これが種間の適応進化であることを裏づける結果として、植物種だけのシステムでは逆に膜テンプレートのマッチパターンが減少していた。マッチパターンの多様性はその膜テンプレートのコードの長さにはほぼ比例している。すなわち、植物種は捕食者がいない環境では余分なテンプレートのコードを取り除くことによって自己複製にかかる時間とリソースを節約する方向へ進化したと考えられる。

4.2 リソース保存性の意義

3.3 節で述べた実験結果は、リソース保存性がシステムの多様性維持に貢献していることを示している。従来のデジタル生物のシミュレーションではリソース不足に陥ることがないので、短時間でより多く自己複製を繰り返し実行できる種が最終的に選択される傾向にあった。しかし本モデルでは、それぞれの種に必要なリソース（シンボル）の豊富さや、それを獲得する能力によっても自己複製の効率は変化する。リソースが有限なシステムでは同じパターンの消費のみを繰り返す群集は徐々に必要なリソースが不足することから、複数の異なる種の共存が促進されたと考えられる。

4.3 Avida との比較

Avida はティエラを基に設計された代表的な生態シミュレータである¹¹⁾。Avida では、デジタル生物どうしの相互作用を実現する 2 種類の環境が新たに導入されている。まず生物は隣接している他者（Avida の生物は 2 次元格子空間にいる）を削除しそこに自己複製したプログラムを書き込む。また生物は与えられた「タスク」を実行することでより多くの CPU 時間をボーナスとして得ることができる。これらの相互作用は生物どうしが互いの生存に直接的あるいは間接的に関わっているものの、このような個体間の関係性を変化させうる適応進化は存在しない。本モデルでは一方の生物種のパターンを相手に応じて変化させることで両者の関係性が変化する。

5. む す び

本稿では、デジタル生物を用いて捕食—被捕食の関係をデザインした生態シミュレータを提案した。それを用いて、有限なリソースをめぐる種間相互作用がシステムにどのような影響を及ぼしているかを分析した。その結果このような生態構造がシステムの安定性ならびに多様性を維持するうえで重要な役割を担って

いることが示唆された。またリソースの有限性は群集の多様性維持に間接的な影響を及ぼしていることが考察された。

謝辞 第2著者の研究は、情報通信研究機構の研究委託および同志社大学学術フロンティア共同研究プロジェクトの援助のもとに実施したものである。

参 考 文 献

- 1) Ray, T.S.: Evolution, Ecology and Optimization of Digital Organisms, *Santa Fe Institute working paper*, 92-08-042 (1992).
- 2) Ray, T.S.: An Evolutionary Approach to Synthetic Biology: Zen and the Art of Creating Life, *Artificial Life*, Vol.1, No.1/2, pp.195-226 (1994).
- 3) Wilke, C.O., Wang, J.L., Ofria, C., Lenski, R.E. and Adami, C.: Evolution of digital organisms at high mutation rates leads to survival of the flattest, *Nature*, Vol.412, No.6844, pp.331-333 (2001).
- 4) Yedid, G. and Bell, G.: Macroevolution simulated with autonomously replicating computer programs, *Nature*, Vol.420, No.6917, pp.756-757 (2002).
- 5) Lenski, R.E., Ofria, C., Pennock, R.T. and Adami, C.: The evolutionary origin of complex features, *Nature*, Vol.423, No.6936, pp.139-144 (2003).
- 6) Chow, S.S., Wilke, C.O., Ofria, C., Lenski, R.E. and Adami, C.: Adaptive radiation from resource competition in digital organisms, *Science*, Vol.305, No.5680, pp.84-86 (2004).
- 7) Suzuki, H., Ono, N. and Yuta, K.: Several Necessary Conditions for the Evolution of Complex Forms of Life in an Artificial Environment, *Artificial Life*, Vol.9, No.2, pp.153-174 (2003).
- 8) Alberts, B., Bray, D., Lewis, J., Raff, M., Roberts, K. and Watson, D.: *Molecular Biology of the Cell*, Garland Publishing (1994).
- 9) Naveh, B., Sipper, M., Lancet, D. and Shenhav, B.: Lipidia: An artificial chemistry of self-replicating assemblies of lipid-like molecules, *Artificial Life IX: Proc. 9th International Conference on the Simulation and Synthesis of Living Systems*, Pollack, J., Bedau, M., Husbands, P., Ikegami, T. and Watson, R.A. (Eds), pp.466-471 (2004).

10) 巖佐 庸: 数理生物学入門—生物社会のダイナミクスを探る, HBJ 出版局 (1990).

11) Ofria, C. and Wilke, C.O.: Avida: A Software Platform for Research in Computational Evolutionary Biology, *Artificial Life*, Vol.10, No.2, pp.191-229 (2004).

(平成 17 年 2 月 8 日受付)

(平成 17 年 4 月 12 日再受付)

(平成 17 年 5 月 30 日再々受付)

(平成 17 年 7 月 8 日採録)



松崎 周一

昭和 50 年生。平成 10 年会津大学コンピュータ理工学部コンピュータソフトウェア学科卒業。平成 17 年会津大学大学院コンピュータ理工学研究科博士後期課程修了。博士(コンピュータ理工学)。同年より室蘭工業大学生命ソフトウェアラボラトリー研究員。人工生命ならびに感性工学の研究に従事。



鈴木 秀明(正会員)

昭和 31 年生。昭和 57 年東京大学大学院理学系研究科相関理化学専門課程修士課程修了(株)本田技術研究所アシスタント・チーフ・エンジニア, ATR 人間情報通信研究所主任研究員, ATR ネットワーク情報学研究所主任研究員等を経て,平成 17 年より(独)情報通信研究機構未来 ICT 研究センター専攻研究員。人工化学の方法論を用いた分子反応システムの計算機内デザインとその工学応用の研究に従事。理学および情報学博士。情報処理学会 MPS 研究会, 日本進化学会, International Society for Artificial Life 各会員。SICE 知能工学部会委員。



小佐野峰忠

昭和 20 年生。東京理科大学理学部応用物理学学科卒業。工学博士(東京大学)。現在,会津大学大学院コンピュータ理工学研究科教授。知識工学および人工生命を研究。