２０１３年度複雑系プログラミング特論（担当：鈴木麗璽）

ガイダンス

概要

生命や社会現象の本質的な特徴を捉えたモデルを計算機上に創って動かして理解する構成論的アプローチは，複雑系科学における重要な研究手法の一つである． 本講義では，複雑系モデルの構築と実験および解析に必要な，プログラミングや計算機の活用に関する知識と技術を基礎から身につけることを目的とする．講義 に加えていくつかの基本的な複雑系モデルの実装を行う．

コンタクト

- 名前：鈴木麗璽（スズキレイジ）
- メール：reiji@nagoya-u.jp
- 居室：情科棟５１３
- 内線：４２５８
-

コンピュータの利用について

この授業ではpythonを実際に動かします．受講者の数によっては，SISスタジオのMacでは足りなくなる場合があります．その際は，可能な人にノート PCの持ち込みをお願いする場合がありますので，ご協力宜しくお願いします．

イントロダクション－複雑系とプログラミング－

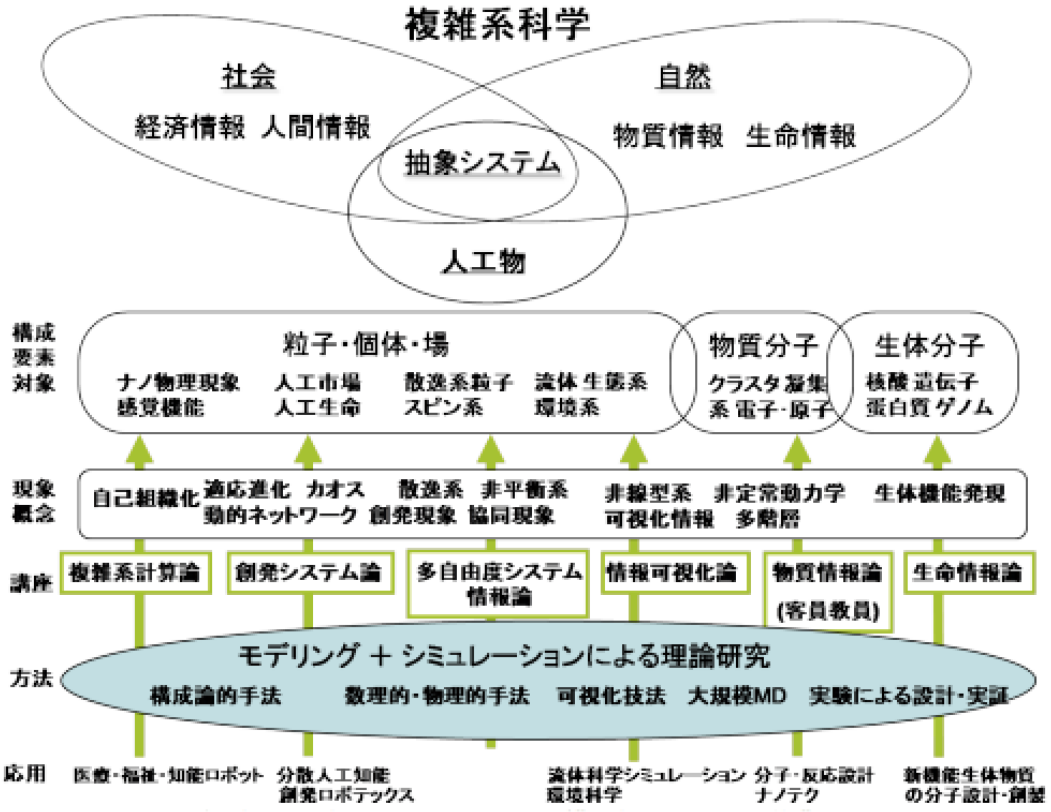
複雑系でなぜプログラミングか？

本専攻の学生さんなら，その理由はよく知っているところかもしれないけれど，自己紹介も兼ねて，（少なくとも私の考える）複雑系科学研究とプログラミングとの関係 についてお話します．

複雑系科学のめざすところ

本研究科複雑系科学専攻では，工学，物理学，有機化学，人類学，生物学，進化学，バイオインフォマティクス 等，実に様々な分野に関して研究が行われている．一見するとバラバラのよう にも見えるかもしれないけれど，専攻の名が指す通り複雑系という視点において共有するものがある．つまり，自然，人間，社会，物質，集団，生態など，扱 う対象は様々だし，方法論や手法は違うけれども，対象とする系の（広い意味での）複雑な側面をどう理解するか，また，どう役に

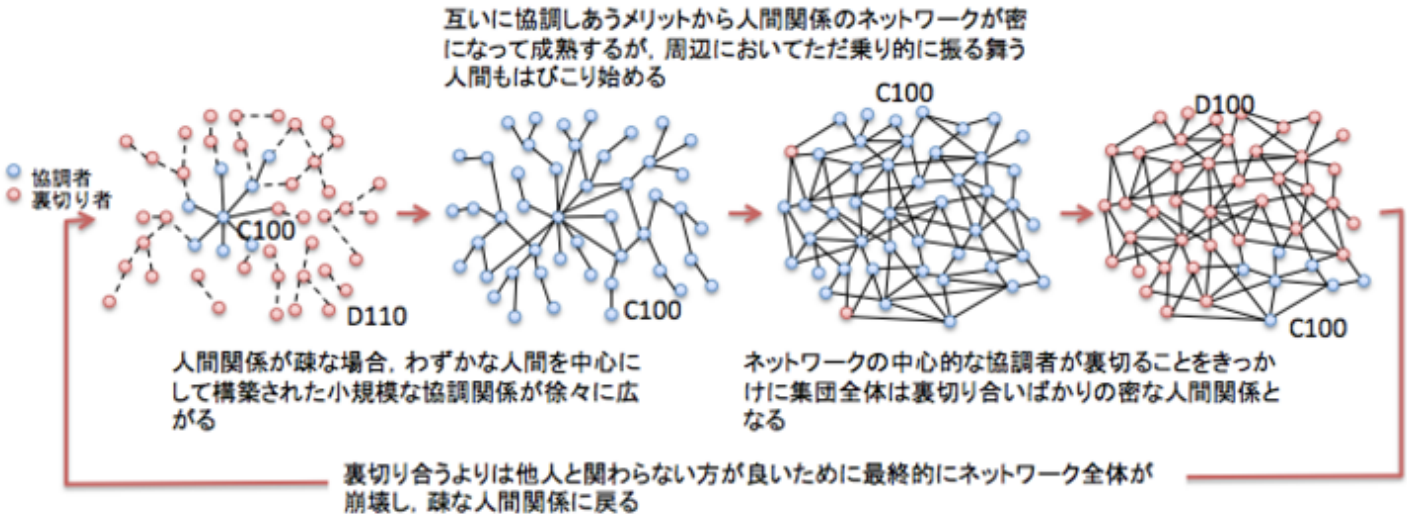
立てるかに取り組んでいる点 である．というより，複雑さの理解は，現代の最新の科学のほぼ全てにおいて本質だといえる．



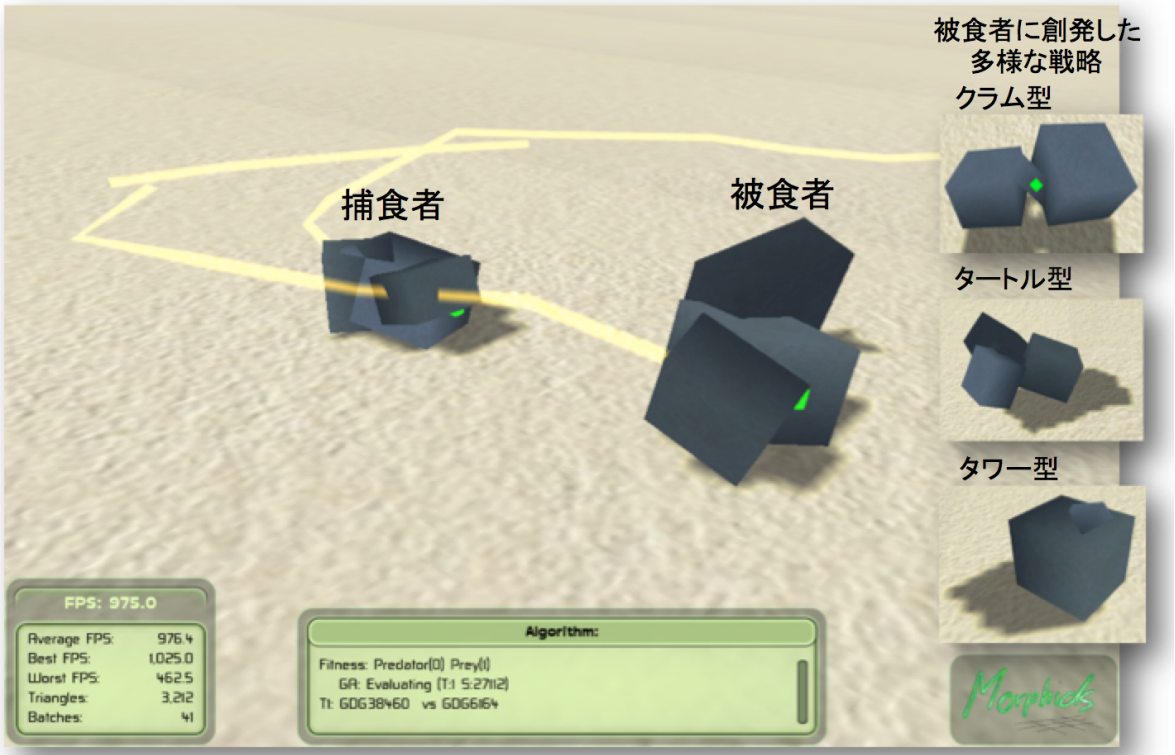
複雑系科学専攻Ｗｅｂページ”専攻の全体像”より

人工生命研究

- 計算機やロボット，分子などの人工メディアを用いて，創発的な振る舞いを”創って・動かして・観察する”ことを通して，生命に関する普遍的な 現象を理解し，応用する一連の試み・研究の流れ
- 特に，生物・社会・生命現象に関する普遍的な概念や概念間の関係を再構築し，応用.

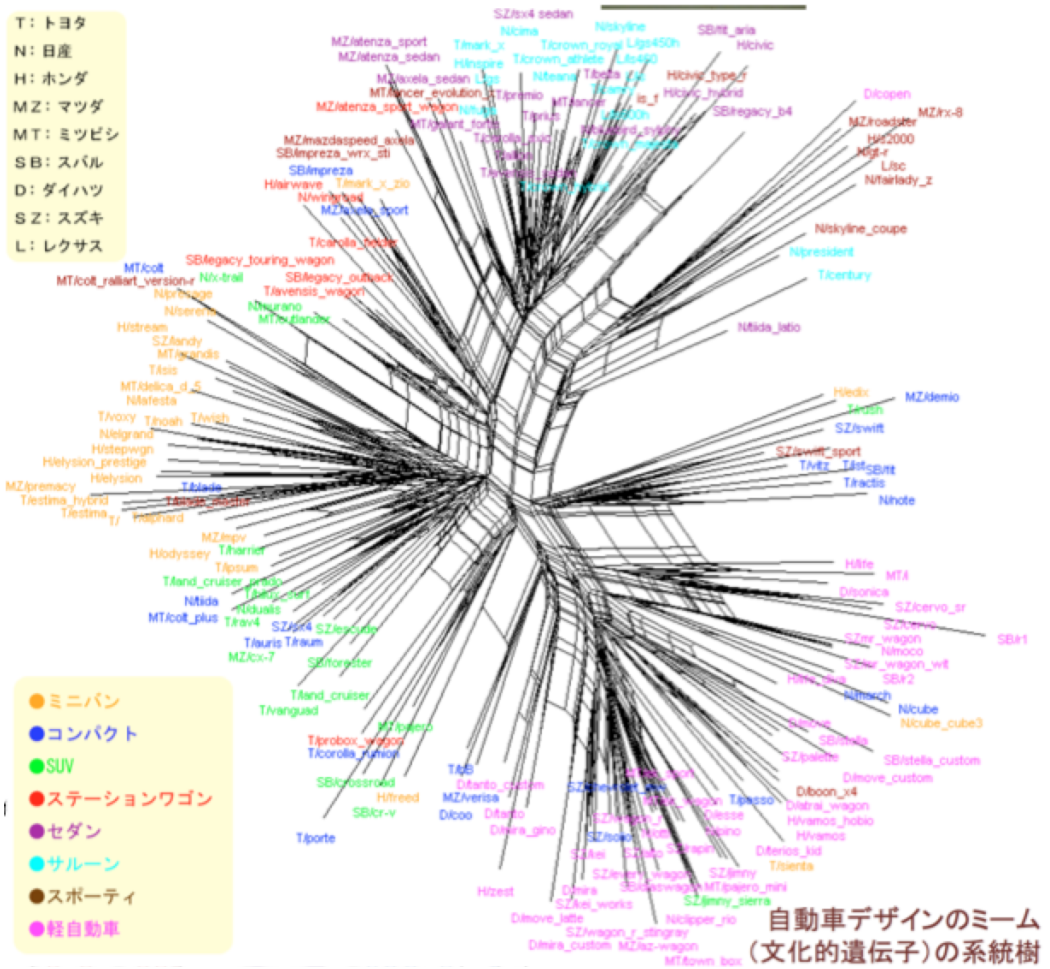


協調行動とネットワーク構造の共進化
Reiji Suzuki, Masanori Kato and Takaya Arita: "Cyclic Coevolution of Cooperative Behaviors and Network Structures", Physical Review E (Selected for Virtual Journal of Biological Physics Research, Vol. 15, Issue 5, March 1, 2008), 77(2), 021911 (7 pages) (2008).

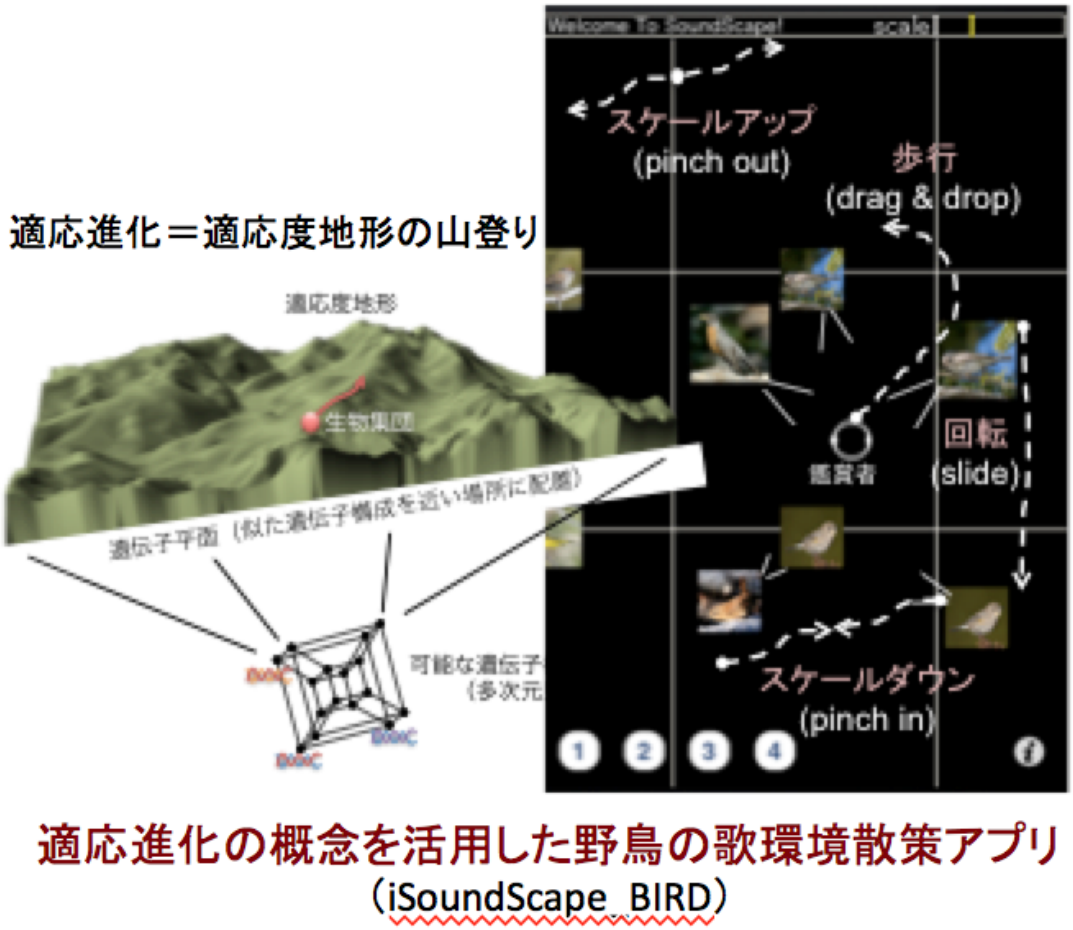


3次元仮想物理環境における被食者・捕食者の構造と行動の共進化

関連文献：Takashi Ito, Marcin L. Pilat, Reiji Suzuki and Takaya Arita: "ALife approach for body-behavior predator–prey coevolution: body first or behavior first?", Artificial Life and Robotics, DOI 10.1007/s10015-013-0096-y (2013).



前田実里，鈴木麗璽，有田隆也: "自動車のデザインの系統樹からみるミームの系統進化", 情報処理学会第71回全国大会論文集, pp. 2-357-358 (2009).



iPadで鳥の歌景観を進化（iSoundScape_BIRD）

Reiji Suzuki, Souichiro Yamaguchi, Martin L. Cody, Charles E. Taylor and Takaya Arita: "iSoundScape: Adaptive Walk on a Fitness Soundscape", Applications of Evolutionary Computation, LNCS 6625 (Proc. of the 9th European Event on Evolutionary and Biologically Inspired Music, Sound, Art and Design (evomusart2011)), pp. 404-413 (2011/04).

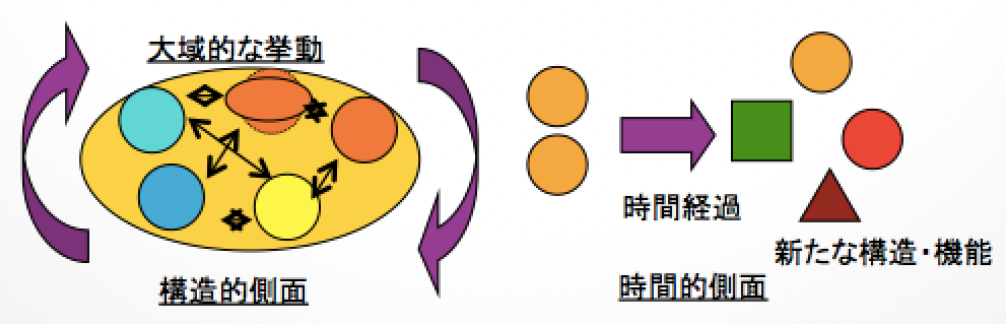
ここで考える複雑系

この講義では，特に，創発的な系，つまり，”単にこみいっている（complicated）のではなくて，ミクロレベルでは単純だけど，マクロレベルでは複雑な現象を示す（complex）系”に焦点を合わせ，その理解と応用を考える．

構造

> 多数のミクロで自律的構成要素が集まって系全体を構築し，その相互作用がマクロな系全体の振る舞いを決定し，また，マクロな振る舞いがミ クロ な挙動を制約する．自然界には，全体として現れる性質をうまく環境への適応に利用するものが多く見られる．アリのフェロモン用いた餌探しはその典型 例．

時間
: ミクロなプロセスの積み重ねの結果，マクロな時間スケールで見た時に何か新しい構造や適応性が発現する．生物集団における自然選択の積み重ねの 結果，新たな 種や形質，適応性が現れる．それがさらに新たなミクロな進化の土台になる．集団が進化することや生まれる構造そのものが前者の意味での創発現 象である場合もある．



なぜプログラミングか？

上記の意味での複雑な系は，従来の要素還元的，もしくは，解析的な理解が容易でない場合がある．もしくは，解析が（理論的には）可能であっても，実際に系を動 かして 挙動を観察することが，系の理解に至る早道につながる場合がある．

- 複雑な挙動そのものをトップダウンに記述しにくい．
- 単純化すると複雑さが抜け落ちてしまう．
- 多様な要素がありすぎできない．
- 解析的に答えが出しにくかったり，答えがあることはわかっていても実際の答えを見つけられない場合がある．

そんなときは実際に動かしてみるのが良い方法の一つ．

複雑な系をどう理解したいか

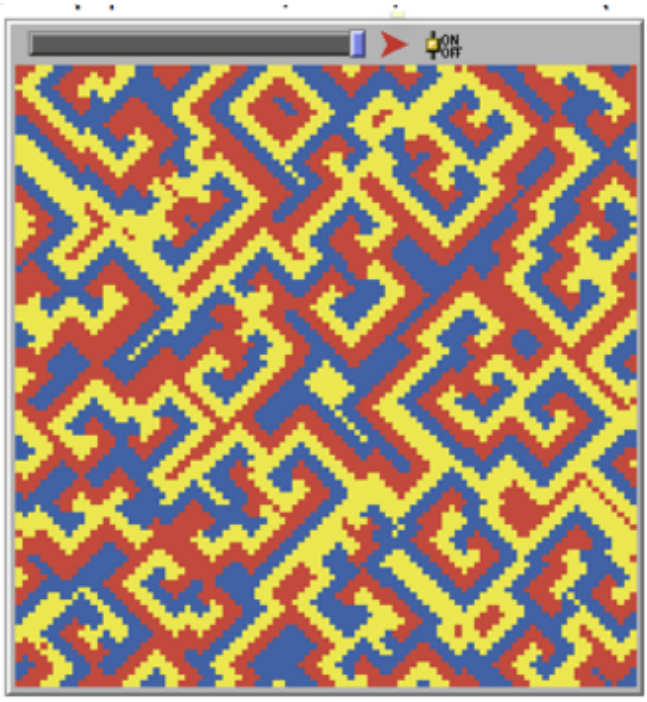
複雑な系を理解するといっても，興味の対象や議論の出発点よってその種類は異なる．

既知のマクロで複雑な系のミクロな仕組みを理解したい
: 例えばある細菌が動く仕組みを理解したいとき，具体的な全体としての挙動は既にわかっているとすると，知りたいのはそれを動かしているミクロなルール．

ミクロなルールから生まれうるマクロな複雑な挙動を理解したい
: 例えばじゃんけんのような単純な相互作用のルールがあったとき，系全体としてどんな挙動が生じるかを知りたいとき．ルールはわかっている，もしくは，仮定しているの で，知りたいのはそれに従って動かした結果生じうる挙動の理論的可能性

- イベント会場での混雑と情報提供
- ジャンケンがつくるパターン

http://www.alife.cs.is.nagoya-u.ac.jp/~reiji/aw/index.html
じゃんけん＋近所を真似

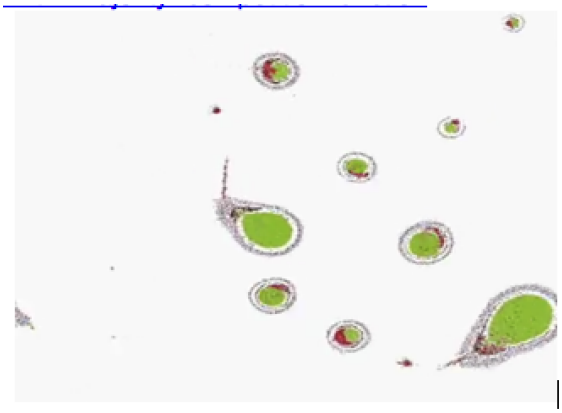


（参考）３種の大腸菌の繁殖ダイナミクス：Kerr, B., Riley, M. A., Feldman, M. W. and Bohannan, B. J. M.: Local dispersal promotes biodiversity in a real-life game of rock–paper–scissors, Nature, 418: 171-174 (2002).

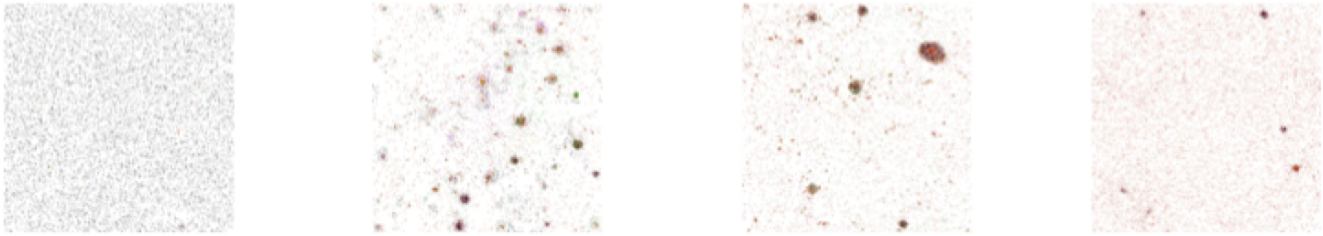
その中間もある

でも実際，ミクロとマクロ，いずれか片方がはっきりわかっている訳では無いことがある．そのような場合，両者の立場を行ったり来たりしなが ら，ミクロとマクロの関係の理解を深めていく．後者によって初めてマクロな現象の発生が明らかになり，その仕組みを解き明かすために新たなミ クロなルール を仮定したり，不要そうなルールを取り除いたり，新たな条件を仮定したりして，系に対する理解を深めていく．

- Swarm chemistry
  Swarm Chemistry Simulator Applet: Version 1.2.0
  Evolutionary Swarm Chemistry: Result with "majority" competition function



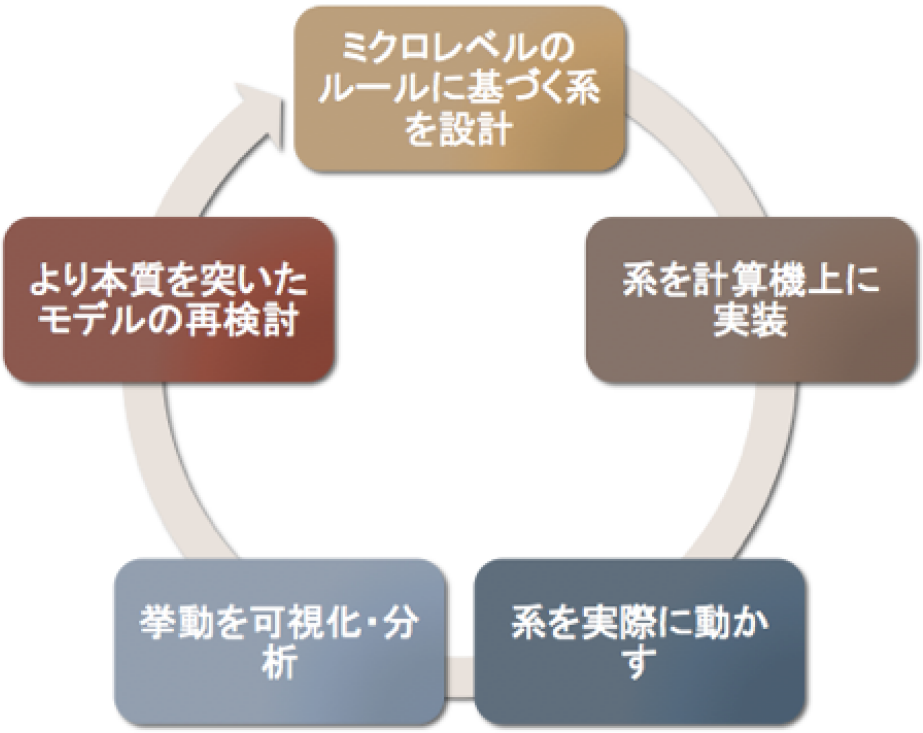
"Hiroki Sayama, Seeking open-ended evolution in Swarm Chemistry, Proceedings of the Third IEEE Symposium on Artificial Life (IEEE ALIFE 2011), Paris, France, 2011, IEEE, pp.186-193."



小林賢吾, 鈴木麗璽, 有田隆也, 佐山弘樹: "物理的に相互作用する粒子系を 用いたミーム伝搬モデル", 第25回自律分散システム・シンポジウム資料, pp. 169-174 (2013).

中間を行ったり来たり＝構成論的アプローチに 基づく複雑系の理解

モデル化，実装，実行，分析，再検討のループ



プログラミングに基づくモデルの実行は，単に設計したモデルを試行するだけの役割でなくて，系を理解するためのループを成り立たせるために必要な要 素．ある種思考の過程の一部．この意味で，プログラミングは複雑系科学において重要なスキル．

なぜ今プログラミングによるアプローチか？

1. 計算機の飛躍的な進歩：プログラムを高速に実行できる環境が手軽に手に入るようになった．
2. プログラミング言語（環境）の進歩： より手軽 に，プログラミングができるようになった．

大学院入試でプログラミングの問題を選択した人の多くは，Ｃ言語を勉強したり，Javaを学んだりしたことがあるはず．でもそれ以上にわかりやすさ，お手軽さを目指した言語がある．その名もPython．計算機の進歩のおかげで，プログラムの速さだけでなくて，つくりやすさでも恩恵を

もっと受けられる は ず.

http://rigaux.org/language-study/diagram.html

超高級言語Python

計算機資源の有効利用よりも，扱う側の容易さを念頭に設計されたスクリプト形式の言語．C言語はこの意味での対極（計算機にとって軽量（高速，省メモ リ））．構成論的ループの多くの段階において貢献しうる.

- 1990年代初頭にグイド・ファンロッサムによってオランダで作成され，ライブラリ等を追加しながら継続的に発展．今回使う環境は Ver.2.x系統（最新は3.x）
- 当初から，プログラミングの専門家でない人が，プログラムを読みやすく，かつ，書きやすいように配慮
    - 模擬コードにできるだけ忠実にプログラムできるようシンプルなコーディングが可能
- かつCやJavaよりも高級な言語
    - 宣言，型の変換，メモリの確保など，従来の高級計算機言語特有の煩わしさが無い．脱ポインタ
- 完全なオブジェクト指向言語
- どんなプラットフォームでも動く
- 数値計算や可視化など，ライブラリが豊富に整備されている．「巨人の方にのって」
- オープンソースで基本的にフリー
- 実は結構使われている
    - 欧米のプログラミング教育，Google，ニューヨーク証券取引所

本講義の目的

というわけで，この講義では,

- 複雑系のモデリングに興味のある人,
- プログラミングにこれまでそんなに触れたことが無いけど興味がある人
- C言語やJavaでプログラミングの基本はもう知ってるけど，新しい言語も楽しみたい人
- 実験結果の整理など，研究に役立つプログラミングの活用法を知りたい人

が，構成論的アプローチ，つまり,

- 複雑系に関する簡単なモデルを設計すること
- モデルを実装するプログラムを書くこと
- その結果を可視化すること
- その結果を分析すること

を，

- 超高級言語Pythonを使ってできるだけ効率的に

行うことを通して，

- 複雑系プログラミングの面白さを味わうこと

を目的とします．

講義の進め方

- トピックに関する講義とPythonを使った実習を行います
- 各テーマ講義と説明と実習を半々くらいにしたい（けど・・・）
- 適宜ミニレポートを出す予定
- いつもこの部屋（ＳＩＳスタジオ）でやります

テーマ（予定）

- 生態
  - 個体群動態
    - 変数，数値，繰り返し，関数定義
- 可視化
  - グラフ作成パッケージ
    - matplotlib
- 複雑系
  - セルオートマトン
    - リスト，条件分岐，etc.
- 社会
  - エージェントベースモデル
    - オブジェクトの定義と利用
- 工学応用
  - 遺伝的アルゴリズム

成績評価

レポート提出状況とその内容で評価します．

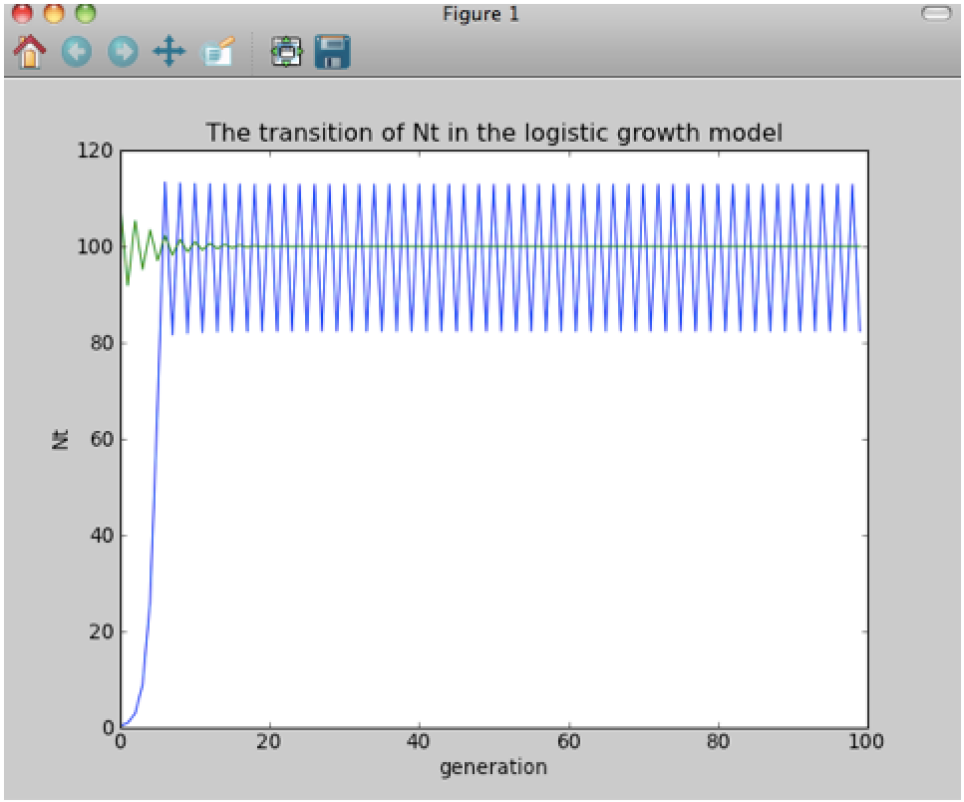
ＳＩＳラボでのPython環境―Enthought Python Distribution (EPD) Free―

- Pythonと関連する様々なライブラリを利用するためのパッケージ
- 無料．EPDの サイト（EPD Free） に行くと無料でダウンロードが可能
- 可視化ライブラリのmatplotlibやGUIのTkinterをはじめとして多くの有用なライブラリが細かい設定なしに利用可能

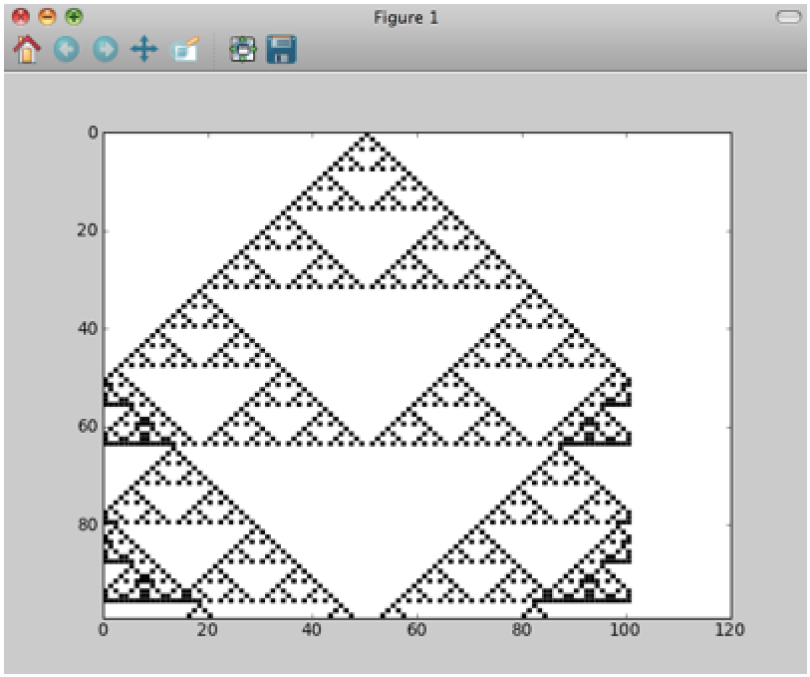
- Pythonシェル（IDLE）が付属
- この部屋のmacにはインストール済み

※重要：EPD FreeはEnthought canopy expressに置き換えられた模様なので，個人のPCにインストールする場合はさしあたりこちらをつかってみてください．機能は 授業で使う範囲ではEPD Freeと変わらないと思います．うまく動かない場合は，EPDFreeもダウンロードできるので，そちらを適宜利用してください．

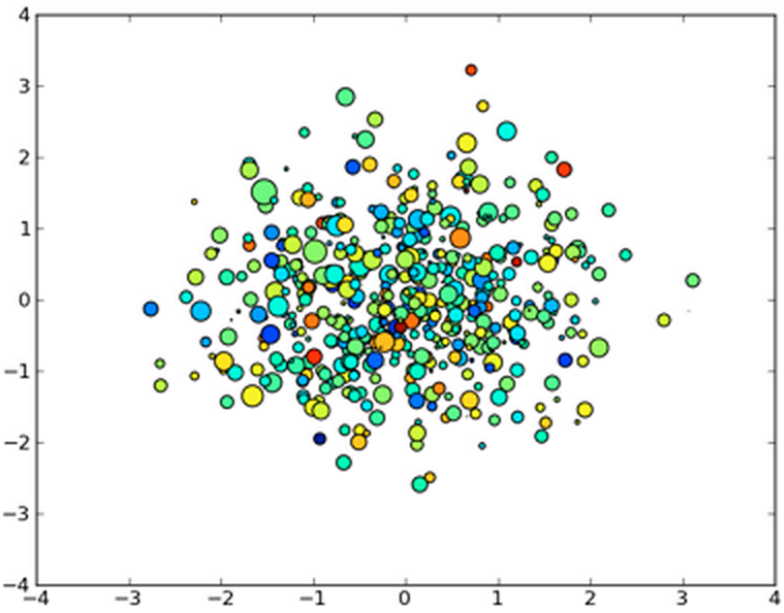
スナップショット



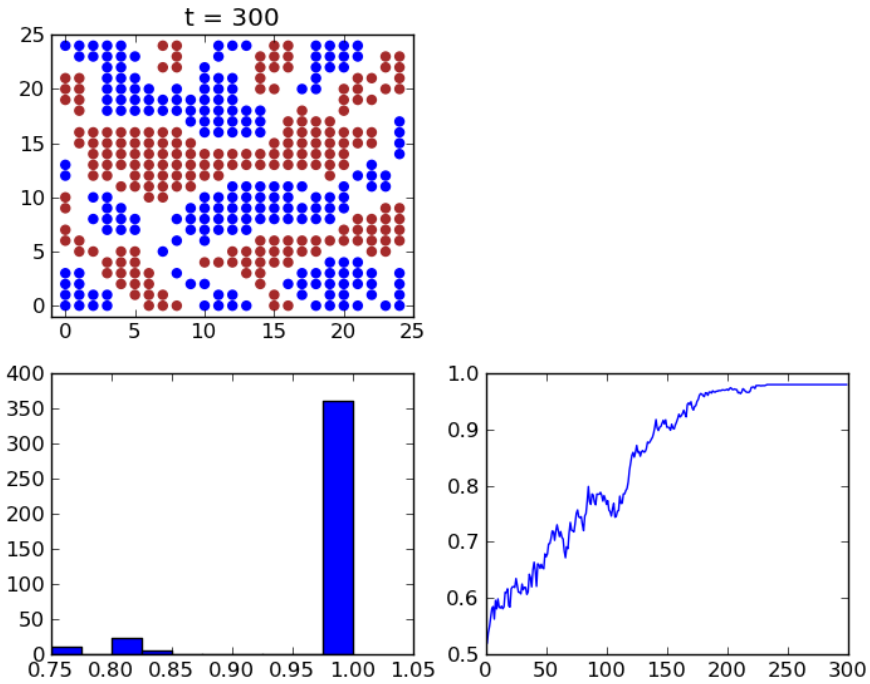
個体群動態モデル



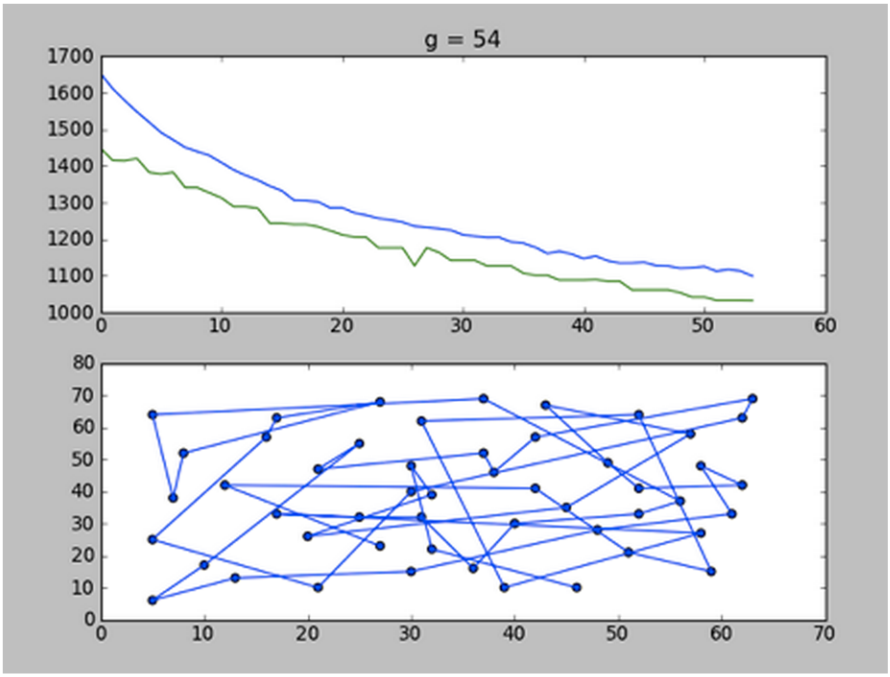
セルオートマトン



Matplotlibの活用



エージェントベースモデル



遺伝的アルゴリズムと巡回セールスマン問題

etc.