

AIにデザインはできるか／Architectural Intelligence

○渡辺 誠*¹ 千葉貴史*²

キーワード：誘導都市 アルゴリズムック・デザイン 設計 情報 強化学習

何故「アルゴリズムック・デザイン」と呼ぶのか

「アルゴリズムック・デザイン」は、「要求される条件をより良く叶える設計を生成させるアルゴリズムを考案し、それを使って設計を得る方法」である。要は、「問題を解く方法＝手順を考え出す」、ということである。そう言うと、これはあらゆる課題に共通のことになる。なにも「アルゴリズムック・デザイン」などと言う必要はない。設計に限らず、企画・立案と呼ばれる行為は皆、問題を解く手順を考えだそうと奮闘している。どんな「設計」過程にも、その手順＝アルゴリズムは存在する。配置計画、法規制、避難計画、構造解析・・・等々、皆、それぞれのアルゴリズムを持つ。ではなぜことさら、「アルゴリズムック・デザイン」と呼ぶ必要があるのか。

「アルゴリズムック・デザイン」の目指すゴール

その理由のひとつは、目指す方向と、想定するゴール、にある。「アルゴリズムック・デザイン」の目指すものは、よりよい「設計」を得ること、である。ここでいう設計とは、部分的な技術のことではない。設計行為によってできあがる実体は、ひとつの建築、ひとつの街である。複数の選択肢を同時につくって、そこからいいものを残して後は壊す、というようなことは、建築や都市の場合、できない。幾多の要素技術を使い、ひとつの実体をつくるのが設計であるなら、そのひとつ＝要素の総体を、「良く」すること。配置、法、避難、構造、空間、形態、機能、施工・・・等々の要素技術・過程の総合結果としての、ただひとつの実体、が、よりよくなるようにあらかじめ設計できるような、方法、それが「アルゴリズムック・デザイン」である。

もちろん、それは言うは易く・・・である。各要素技術のアルゴリズムを書き出すことすら、容易ではない。ましてや、その総体である設計行為の、アルゴリズムなど、書き出せるわけではない。優れた設計者が頭の中で行っていることを外部化するなど、無理な話だ。

しかし、その無理な話しを、無理でなくしよう、というのがまた、「アルゴリズムック・デザイン」なのである。それは無謀な挑戦かもしれない。しかし、不可能と決まった訳でもない。

アルゴリズムと、プログラム

その、無理っぽい、大それた目的をかなえるために、幾多の試行が行われている。なんとかしてアルゴリズムを書き出すことができて、それを実行するにはほとんどの場合、

コンピュータプログラムが必要となる。アルゴリズムはできて、プログラムができなければ試行はできない。アルゴリズムとプログラムは密接に関わるが、この両者は同じではない。

例えば、「誘導都市」シリーズのひとつ、「ON DEMAND CITY」1994-2012- では、住宅やコンビニ等の機能単位を必要個数だけ任意の道路パターンの上にばら蒔き、じわじわと移動させてそれらすべての相互距離が望ましい設定値になるような配置を得ようとする。この最初の段階でのアルゴリズムは次のようになる。

「各要素機能間の望ましい距離の設定→計画上からの各所要個数の算定→街路パターンの生成→初期値ランダム配置→全要素間の相互距離の算定とスコアリング→要素をひとつ移動してスコアが向上していれば次の移動→スコアが向上していなければ移動をキャンセルして次の移動→一定回数の試行でスコアが向上しなくなれば停止」

しかしこのままではまだプログラムにはならない。街路発生アルゴリズム、スコアリングアルゴリズム、等、これをブレイクダウンしたアルゴリズムが必要である。下位に行くほど、次第にアルゴリズムはプログラム本体に近接していく。両者の関係は、グラデーショナルだ。ただし、その初期段階のアルゴリズムと成果品であるプログラムを比べれば、両者は明らかに異なる。それぞれに思想と技術があり、様式や趣向もある、別の領域の成果品なのだ。

「アルゴリズムック・デザイン」は、アルゴリズムとプログラムというふたつの領域を、並行して、あるいは相前後して、もしくは融合させながら、進展させることになる。

書き出せるものと、書き出せないもの

よりよい解答を得るためのアルゴリズムを書き出す、ことは容易ではないが不可能でもないとしたが、実は、通常の意味で書き出すことは殆ど不可能、という分野がある。

その逆、書き出せる、ものはどういふものか。「誘導都市」シリーズのひとつ「太陽神の都市」1994-2012- は、「反・太陽神の都市」と共に、集合住宅の受光エネルギーを最大・最少にする配置を得ようとするものであるが、ここで扱う光は、アルゴリズムの書き出しが可能である。直進する光の多い少ないは、定義が明快だ。

同様に、視線の絡み合わない、あるいは適度に絡みあう配置計画を得る「PrivaCity」2012- でも、視線・視界は定義可能で、アルゴリズムを書き出すことができる。こうした物理指標や、計測できるデザイン指標を使う場合

は、アルゴリズム化の可能性が高い。

では書き出せない類のものは何か。

そのひとつは、定義と計測が（目下の技術では）困難な分野、たとえばひとの感性に関わる領域である。

その最たる例は、美意識だ。「美しいかたちをつくるアルゴリズム」というものは、まだない。美しさ、は、定義できないからだ。

そんなことはない、多くの被験者のアンケートや脳機能の計測実験を繰り返せば、美しいかたちはみつかるはず、という意見もあるだろう。ではしかし、好きな異性の顔は、美しいと思える異性の顔は、あなたも隣の方も、同じだろうか。

アルゴリズムは、美しいかたち をつくれるか：

「流れのプログラム 2000」

定義できる価値基準を持つタイプの課題に対しては、それを解くアルゴリズムを書くことができる（可能性が高い）。

定義できない価値基準を持つタイプの課題に対しては、それを直接解くアルゴリズムを書くことはできない（可能性が高い）。

前者は、次にアルゴリズムのプログラム化に進む。もちろんこれは容易ではないが、とにかく進むべき道はある。

後者には、まず、最初に進むべき道が見当たらない。かつて、この後者のアルゴリズム化を試行したことがある。それは「流れのプログラム」2000 である。これはつくばエクスプレス柏の葉キャンパス駅を設計した際に、未だソフトウェア創造事業の支援を受けて開発試行したものである。ここでは、定義できない美しさ、を、対話を通じて得るアルゴリズムを考えた。ここではニューラルネットの教師付学習（FFNN：feedforward neural network）を用いた。これに GA を組み合わせて、形態発生と被験者によるスコアリングの繰り返しから、よりよいとスコアされる解答＝デザインを生成できるはず、という目論見であった。

プログラムは作動し、それなりの結果は得られたが、求めるレベルの解には至らなかった。プログラムが示すデザインは、たいして美しいとは思えなかったのである。

プログラムの解答は、ひとの感性により手が加えられて、建築は立派に完成した。ひととキカイのコラボレーションという意味では成果ともいえるが、プログラムとしては性能不十分、となった。

ここで美しいかたち、を目的にしたのは、美しいかたち、が手ではデザインできないから、ではもちろんない。手で描けば、美しいかたちをつくることは容易だ。そうでなければ、建築家を名乗れない。できないからキカイに頼る、のではなく、手で簡単にできることが、キカイには苦手なので、それをうまくやらせるキカイをつくってみたい、という意図からのことである。

その目的は、この時点の技術と試行では、果たせなかった。

→# 1 「流れのプログラム 2000」 参照

アルゴリズムは、美しいかたち をつくれるか、再び：

「流れのプログラム 2015-」

流れのプログラムの特徴は、何が美しいのか、という価値基準＝評価関数をしめさずに、結果を得ようとするところにある。それをプログラム自身が学習することによって達成しようとする。試行錯誤から技量を向上させる、という方法は、ヒトの成長過程の学習方法と（表面上は）似ている。こうした方法は機械学習と呼ばれて、昨今、各種の領域で精力的に利用開発が進んでいる（技法の呼称・定義には、曖昧性・多義性有）。機械学習の技法のひとつとして用いられるのが CNN (Convolutional Neural Network) である。これもニューラルネットであるが、ハードの能力向上とあいまって、以前の NN より実効性が高まったとみなせる。そこで CNN を用いて「流れのプログラム」を再開し、目下進行中である。

→# 3 「流れのプログラム 2015-」 参照

学習プロトプログラム：Q-Gate

これと関係する技法に、Q ラーニングがある。Q ラーニングがどの程度有効性を持つのかを検証するために作成した、できるだけ単純なプログラムが、Q-Gate である。多数の粒子が複数のパターンで出現する環境で、上下ふたつの領域での粒子分布の割合が指定値になるように、中間にあるゲートを開け閉め操作する、というのがプログラムのしごとである。粒子が多く来るときはゲートの幅を狭く、少ないときは広くすればいいのだが、プログラムにはそれは教えられていない。いっとうすればいいかを、試行錯誤を通じて（プログラムが）自分で学んでいくのである。結果として、学習による能力向上は確認された。

→# 2 「誘導都市 Proto Program: Q-gate

強化学習による設計系プロトプログラム試行 2015」 参照

AI の波_浮き沈みの歴史

機械学習も広くは AI の一環とされる。AI=人工知能は別に新しいものではなく、1950年代から開発され、80年代にもブームが起きたこともある。例えば医療でのエキスパートシステムは、70年代に盛んに喧伝されたが、その後病院で病名診断時に医師がこれを使うのを見たことはない（画像診断等には使われている）。一度持ち上げておいて、次にあれはもうダメだ、というように落とすのは、メディアの常道である。しかし、研究は流行とは違う。今日またブームのような状況を示す機械学習や AI も、しばらくするとまた使えないと言われるかもしれない。そうした浮き沈みの激しい波を乗り越えて、次の階梯へと進む意思と力を持つことが、新しい領域の進展には必要だろう。

AI にデザインは可能か / Architectural Intelligence

現在のディープラーニングは、ビッグデータから有意な傾向を取り出すとか、群衆の映像から特定の個人を見つける、といった、関係性の発見には有効に使われつつある。しかし、デザインする、というような、いわゆる創造的行為にはまだ使えていない。制作に使えない、というわけではない。ニュースサイトで刻々と入るデータから随時ニューステキストをつくることはすでに欧米サイトで広く実用されている (AP 通信等)。ゴッホでもモネでもその画風を模した絵を描けるプログラムもある (Wolfram Research 社)。この延長上には、コルビジェ・マシンもできそう。そうしたいわば編集型のデザインも、デザインのひとつの道ではある。しかし、この方向で、編集を越えた創作、創造の域に達することが可能かは、定かではない。

AI は、いわばブラックボックスである。アルゴリズムック・デザインの非 AI 型のものでは、課題をクリアするためのアルゴリズム=ロジックを (プログラム以前に) ひとが考案作成する。しかし、AI では、ひとがつくるのは、AI のしくみであって、AI がどのように学習し何ができるようになるのか、は、AI 任せ、である。課題解決のアルゴリズムは、前述のように、定義できない価値基準に対しては有効な手立てを持たないことも多い。しかし、AI は、原理上は、そうした対象も扱うことができるはずだ。

そうした AI=Artificial Intelligence も包含して、「設計」に、検証可能な「知的方法」でたち向かおうとする営為全体に、ひとつの呼称を与えることができる。これまで別個に進められてきた、建築計画、環境工学、構造力学、そしてデザイン領域でのそうした試みを、アルゴリズムック・デザインを含めて、より広く・深く、AI=Architectural Intelligence と呼ぶことができよう。続きはまた、次回に。

2 : 誘導都市 Proto Program : Q-gate

強化学習による設計系プロトプログラム試行 2015

背景 :

建築領域でのデザイン系プログラムの目的は、多くの場合、[「環境」状況と、応答「行動」の選択肢、が用意され、「評価」基準が明示された場で、指定された「状態観測」情報を利用して、評価基準をよりよく満たす (=得点=報酬 の高い)「状態」を得る「行動」を選択すること]、といえる。それは建築領域に限らず、設計行為一般に通ずることでもある。さらに単純化すればそれは、「入力と評価」に対応する「的確な出力」を得ることである。このように定義すると、その定義からは、対象の特殊性や付帯条件を極力排除した、単純なプログラム、を想定することができる。それを (設計系) プログラムの汎用「モデルプログラム」、あるいは「プロトプログラム」と呼ぶこともできよう。本研究は、このプロトプログラムを、特定の技法を用いて、探求するものである。

誘導都市 : (略)

目的 :

上記の研究を通じて、共通する課題 (=プログラム化が困難な面) も明らかになっている。そのひとつは、「何がいかに予め決められない=評価関数を定められない」場合であり、もうひとつが、「どうしたらよくなるか分からない=高い報酬を得る行動を予め定められない」場合である。このうち、特に後者への対応方法を、本研究は探索する。

方法 :

上記の検討を行う「方法」に、本研究では強化学習 (Reinforcement Learning) を使用する。強化学習は、「このようなときはこうすべし」という教師データは与えられずに、「これはよい」、という指標 (=報酬) のみが与えられ、試行錯誤でその報酬を得る「行動」を獲得していく方法であり、機械学習のいち分野である。

特性 :

強化学習を用いる意味は、即応性と学習性にある。前述のようにプログラムは一般に環境条件、評価関数、および得点を高めるための操作、が指示されて有効に作動する。このうち、「得点を高めるための操作」が指示されていない条件下で、その操作を試行錯誤で「学習」するのが強化学習である。さらに、環境条件は随時変動するため、学習を随時継続することが必要になる。逆にいえば、この即応性と学習性を必要としない課題では、強化学習を使う意味はない。こうした特性は、ロボットのような自立行動時に適しているが、前述の「目的」達成の可能性として、設計領域でのその有効性を探索することを、ここでは求める。

設定 : (略) 評価 : (略)

展開 :

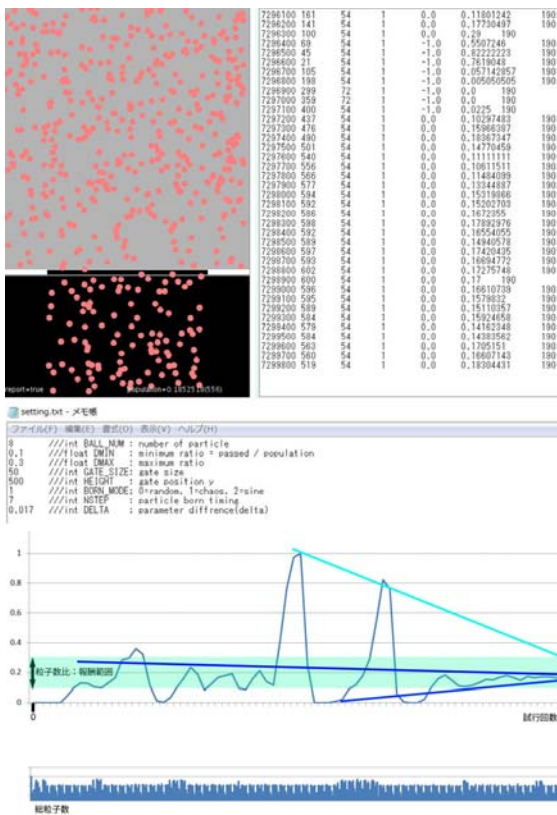
1-複数項目の処理 : 現在のプログラムでは、扱う項目は粒子の数と運動のみである。これを複数の項目を関係させて処理することで、後述するように対象範囲を拡張することが想定される。2-学習方式の深化 : 例えば「空白の次は→多数の波が来ることが多い」傾向にあるとすれば「空白の次は→開口を狭めておく→得点が高い」ことになる。これはいわば記憶の蓄積による学習といえる。しかし現在のプログラムはその学習能力を (十分には) 持たない。現在のプログラムは、「多数の粒子が来る→開口を狭めると得点が高い」という因果関係を学習した段階である。次の段階では、状況観測情報を蓄積してこのような記憶・推論による判断/行動を可能にすること、が考えられる。

拡張 :

本プログラムでの条件は、外力としての粒子の運動であるが、その粒子と運動に別の「意味」を付与する (=読み替える) ことで機能を拡張し、プロトプログラムを実施プログラムに適用することが考えられる。例えば、1-変動する自然環境条件の選択的受容例 : 可動開口部、ルーバ一類の角度、ガラスの透過・反射率、等を変化させて、望

ましい状態の、光・熱・温度、湿度、気流、新鮮空気、等を得る。 2-変動する自然環境条件への形態応答例：気流の状態により外部形状を変化させる、音響の状態により内部空間の形状を変化させる、参加者の数・行動により空間の広がりを変化させる、等。 3-変動する社会的状況への選択的応答例：Twitter等のSNSやビッグデータ上の言葉・意見の量と分布の推移を、粒子の運動・状態に「読み替える」ことで、社会状況・活動に応じて変化する建築、への展開も可能性がある。(本稿は部分再録)

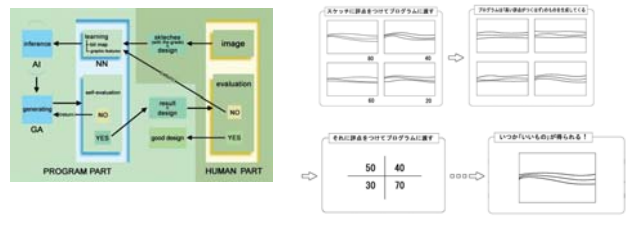
下図：実行時メイン画面例：ゲートの開閉操作を学習中
 下下図：状態記述欄：左から、実行回数、粒子総数、状態1、2、要求条件の満足度、ゲート通過後の粒子数比、ゲート開口幅



上図：実行結果の時系列グラフ：実行回数を経るに従い、粒子数比が期待値に収束していることから、学習が有効に作用していると判断される。

1 : 流れのプログラム 2000

ここでは、建築はうまく行ったがプログラムの性能は不十分、であった。その原因を特定することは困難だ。なにしろ、NNは何をどう判断しているのか人知では理解が難しい。また、一本の線は、その僅かな揺らぎで良し悪しの判断が変わる。そして、まあよい、という線の無数のバリエーションの中に、とてもよい、という線が潜在するとは限らない。創造は飛躍の中にあり、その飛躍がいつ起きてどこに着地するのか、その予測は、この時点ではできなかった。



上図 2 点：
 「流れのプログラム」の「流れ」
 左図：完成した「TX 柏の葉キャンパス駅」2004/5

3 : 流れのプログラム 2015-

「よいものを生成する」には、「よいものを見分けられる」ことが前提となる。そこで第一段階として、教師データとを与えて、それに類するものを対象データから検出できるか、の試行を行った。機能評価を明瞭にするために、対象はできるだけ単純で作為を無くし、一定範囲内での点群の分布を使った。その点群の分布画像に、良し悪しの採点をして、これを教師データとする。現在の試行で検出は概ね可能となっている (Google 検索程度?)。目下、その次の展開を検討中である。



上図：教師データ
 右図：結果例：学習後のプログラムによる対象データの分類。教師データ 2 (右上からの対角連続をよしとする) に対応する、選択。

[参考文献]
 1) 「建築は、柔らかな科学に近づく」(渡辺誠・建築資料研究社)
 2) 「アルゴリズムック・デザイン 実行系」(渡辺誠・丸善出版)

*1 渡辺 誠/アーキテツ オフィス代表
 東京都立大学 都市生活学部 教授
 *2 建築ピボット代表