

物体操作に適したワーキングメモリを持つ汎用人工知能アーキテクチャの検討

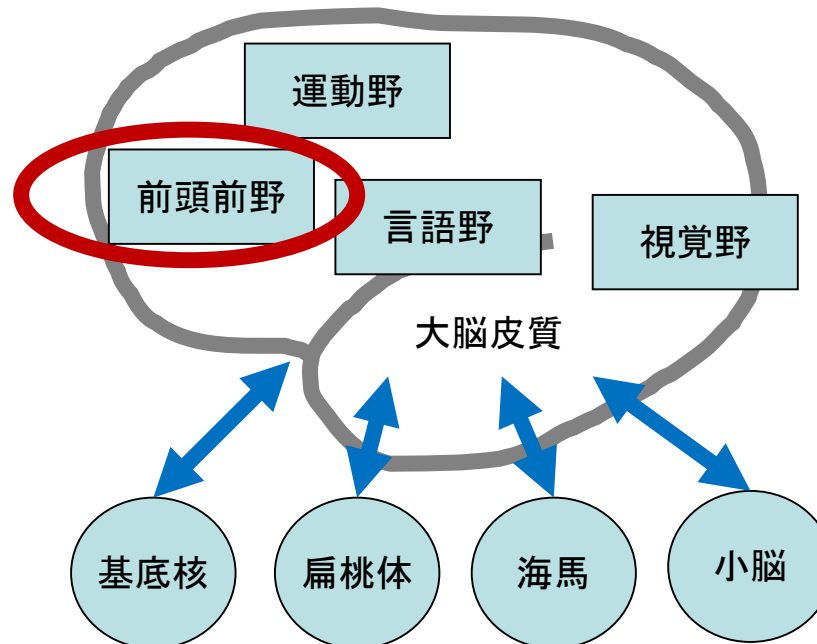
2020-11-20 汎用人工知能研究会

一杉裕志、中田秀基、高橋直人(産総研)、
佐野崇(東洋大学)

※ 忌憚ないコメントよろしく願います。

私の研究の中期的目標

- ヒトの脳の前頭前野周辺の計算論的モデルの構築
- 「前頭前野はヒトをヒトたらしめ、思考や創造性を担う脳の最
高中枢であると考えられている。」「前頭前野 - 脳科学辞典」
<http://bsd.neuroinf.jp/wiki/%E5%89%8D%E9%A0%AD%E5%89%8D%E9%87%8E>
- 作業仮説：前頭前野は教師なし学習と強化学習を用いた
プログラム合成システム



学習により獲得する「プログラム」のイメージ

行動ルール: スパースな整数値ベクトルのパターン、
状態行動対を抽象化したもの

プログラム:
100万個程度の
行動ルールの集合

(_,_,_,29,_,_,_,_,X,_,X,_,_,...)
(_,_,_,48,_,_,_,_,X,_,X,_,_,...)
(_,_,_,62,_,_,_,_,X,_,X,_,_,...)
...
(29,_,X,_,_,_,Y,_,_,_,12,_,_,Y,...)

↑ ↓ 参照・更新

ワーキングメモリ:
100万次元程度の
スパースな
整数値ベクトル

(0,0,0,29,0,0,0,0,94,0,0,0,0,0,0,0,...)

固定長の数値ベクトルだけで
複雑な行動ルールを表現可能か？

物体操作はヒトが日常的に行う知的作業

- 物体の操作
 - 食料の獲得・調理、道具の作成・使用
- 例：料理
 - 冷蔵庫からニンジンをとってきてまないたに載せて、包丁で切って、鍋に入れて、...
- ものをつかむ、動かすなどの個々の動作の制御は前運動野、手順を考えて実行するのは前頭前野。← 本研究の対象



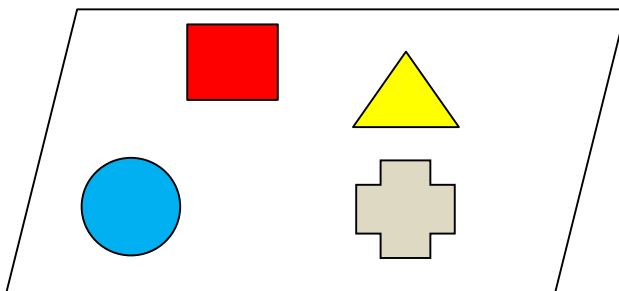
「かわいいフリー素材集いらすとや」
<https://www.irasutoya.com/>

関連研究

- 認知アーキテクチャ
 - Soar のオブジェクト, ACT-R のチャンク
 - 汎用的なデータ構造
- ニューラルネットを用いたプログラム合成システム
 - DNC(Differentiable Neural Computers) [Graves et al. 2016]
 - データ構造を扱えるようにメモリに工夫
- 深層学習を用いた表現学習
 - GQN(Generative Query Network) [Eslami et al. 2018]
 - 複数の物体から構成される画像を再構成
 - MONet(Multi-Object Network) [Burgess et al. 2019]
 - 複数の物体に逐次的に注意を向けて認識

特徴統合理論とオブジェクトファイル理論

- 視覚バイディング問題に関連する認知科学的理論
- 特徴統合理論[Treisman and Gelade 1980]
 - 視覚情報処理の初期段階で特徴(輝度、色、傾きなど)ごとに空間的に並列に処理された情報に対し、**逐次的に空間的注意を向ける**ことで物体ごとに特徴を結びつける。
- オブジェクトファイル理論[Kahneman et al. 1992]
 - 結びつけられた物体の属性がオブジェクトファイルとして短期記憶で保持される。
 - 実験により**オブジェクトファイルの数は4個程度**であると主張。



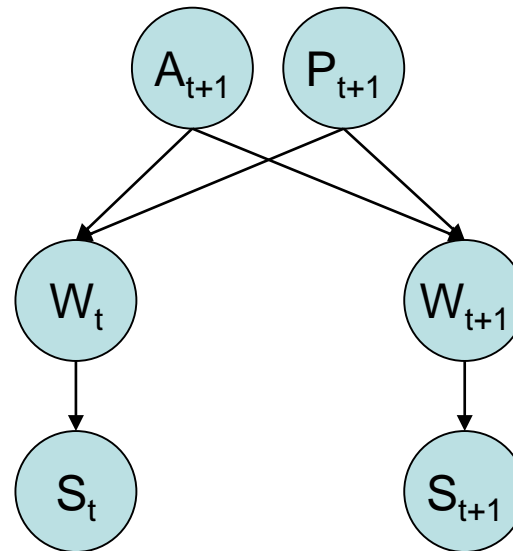
位置:左上、形:四角、色:赤
位置:右上、形:三角、色:黄
...

ベイジアンネットでおブジェクトファイルを実現可能か？

- 前提：
 - ヒトは複数の物体の属性の組をおブジェクトファイルとしてワーキングメモリ上に保持する。 [Kahneman et al. 1992]
 - おブジェクトファイルの処理に大脳皮質が関与する。
[Prabhakaran et al. 2000] etc.
 - 大脳皮質はおそらくベイジアンネットである。 [Ichisugi 2007] etc.
- ベイジアンネットを使っておブジェクトファイルが実現可能ならず。

ワーキングメモリの生成モデル

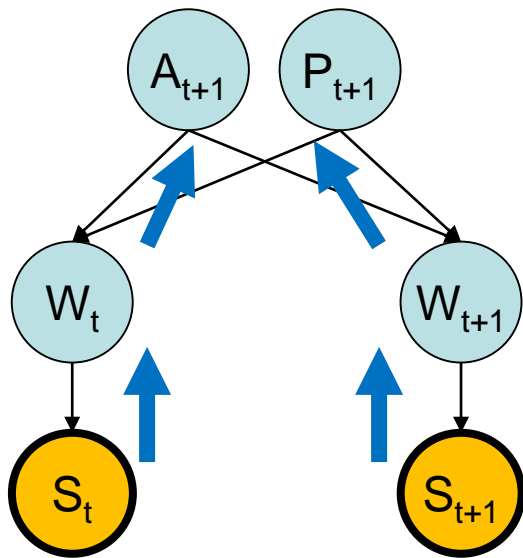
- A: 行動ルール(能動的な値の参照・更新)
- P: デフォルトの保持・忘却ルール
- W: ワーキングメモリ
- S: センサー入力



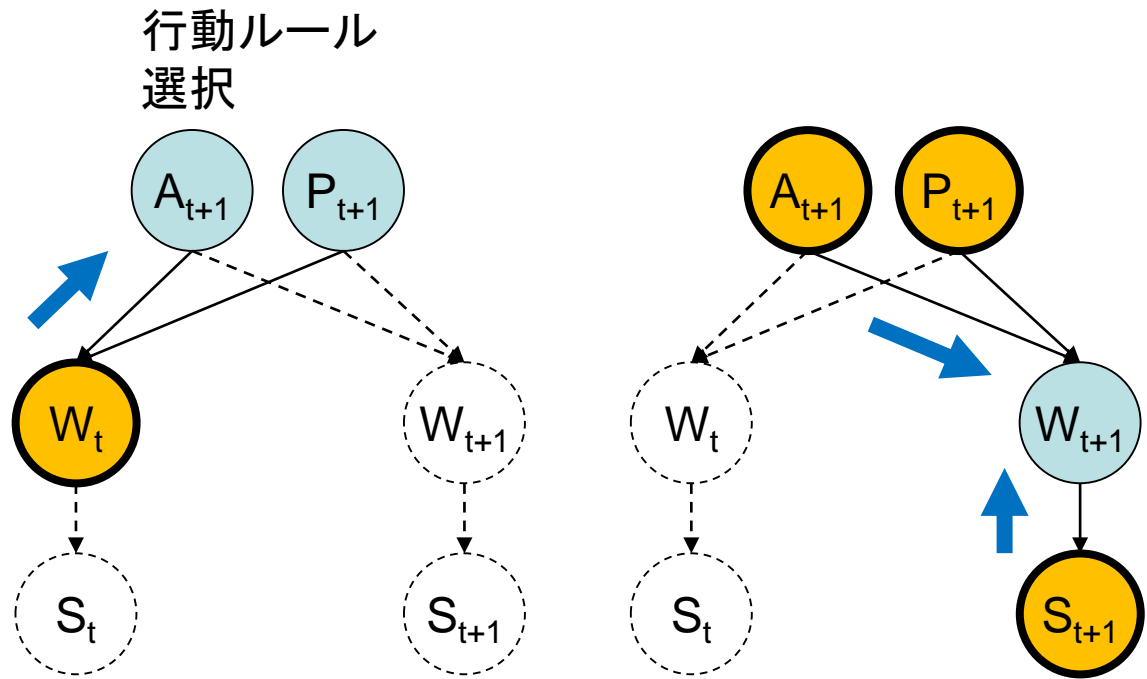
- ・ワーキングメモリの値はセンサー入力から推定される環境の状態を圧縮・抽象化した情報を表現。
- ・ノード A と P は、時刻 t と t+1 におけるワーキングメモリの間の関係を圧縮し、能動的因子と受動的因子に分解して表現。

パラメタ学習とワーキングメモリの更新

A: 行動ルール
P: 保持・忘却ルール
W: ワーキングメモリ
S: センサー入力



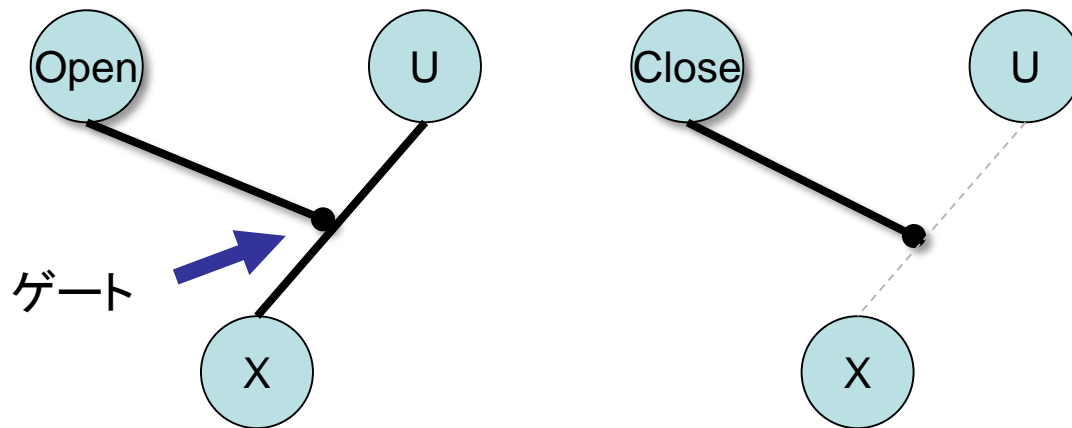
パラメタ学習



ワーキングメモリの更新

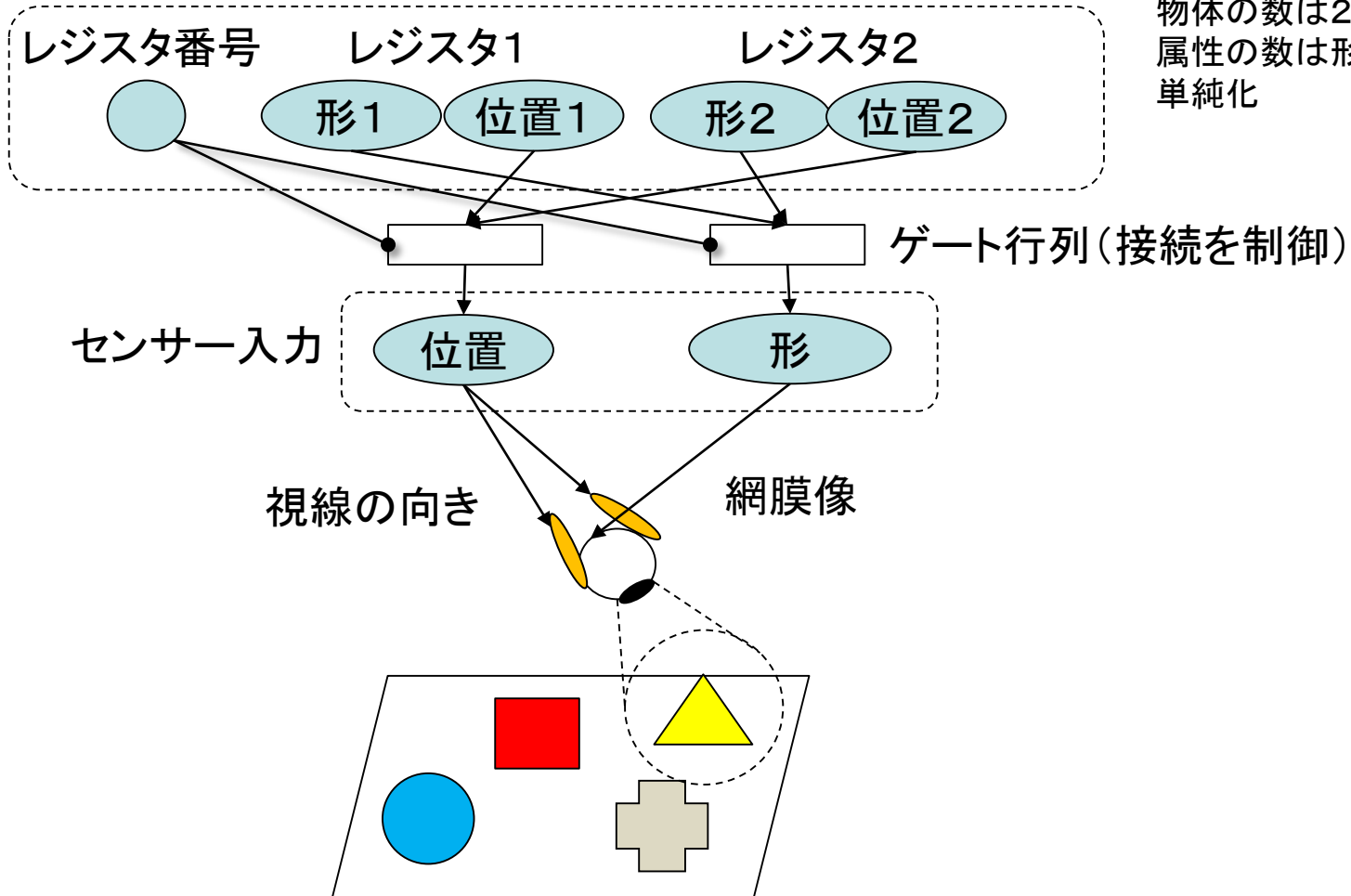
大脳皮質モデル BESOM [Ichisugi 2007]

- 条件付確率に制約を入れてパラメタ数の爆発をおさえたベイジアンネット。
- 最新版の BESOM は、ゲートにより他のノード間の接続を制御する機構を持つ。[一杉、第4回 汎用人工知能研究会, 2016], [一杉、第15回 汎用人工知能研究会, 2020]



オブジェクトファイルは BESOM を使った生成モデルとして表現可能

ワーキングメモリ



注:この絵では
物体の数は2個、
属性の数は形と位置のみに
単純化

物体の探索

レジスタ番号

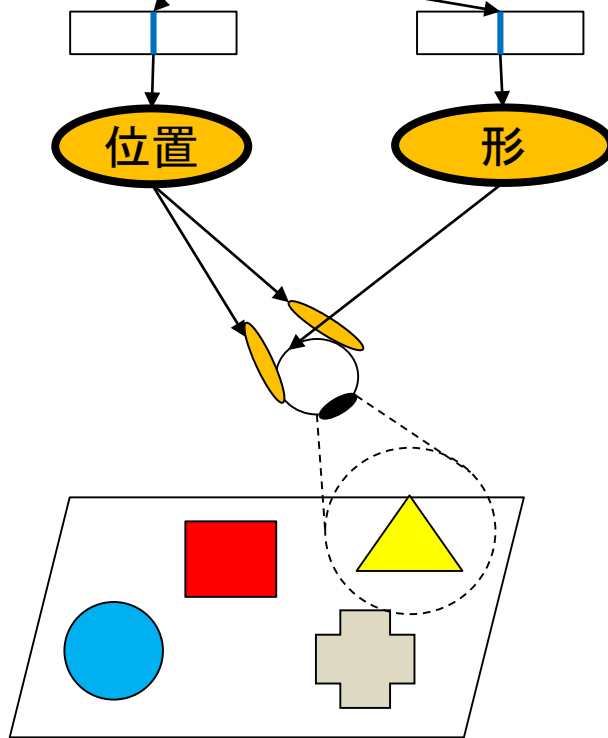
1

レジスタ1

三角 (81,13)

レジスタ2

_ _



眼球は目の前の物体の中からランダムに1つ選んだ方向を向く。

物体の探索

レジスタ番号

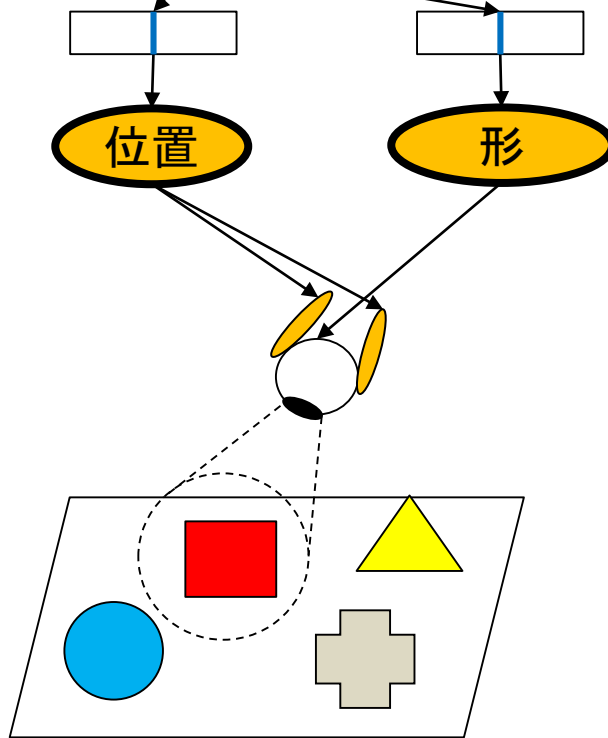
①

レジスタ1

四角 (23,18)

レジスタ2

_ _



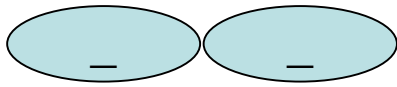
眼球は目の前の物体の中からランダムに1つ選んだ方向を向く。

物体の探索

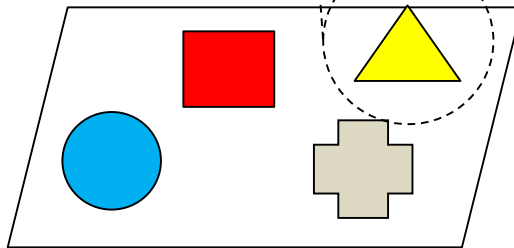
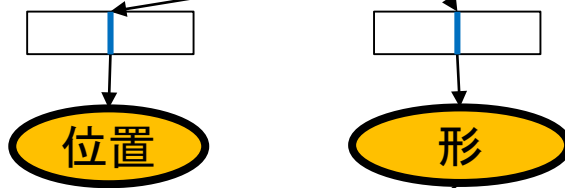
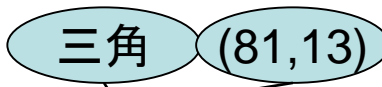
レジスタ番号

2

レジスタ1



レジスタ2



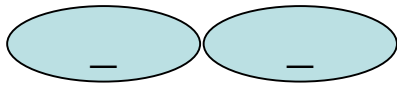
眼球は目の前の物体の中からランダムに1つ選んだ方向を向く。

物体の探索

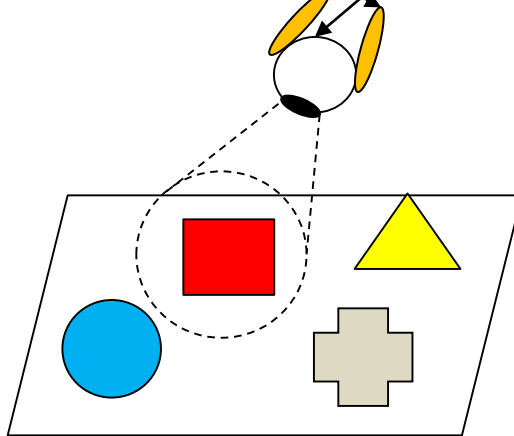
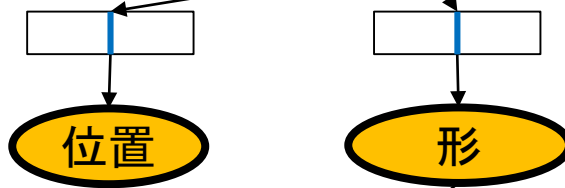
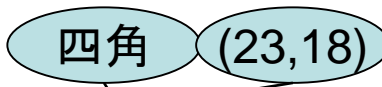
レジスタ番号

2

レジスタ1

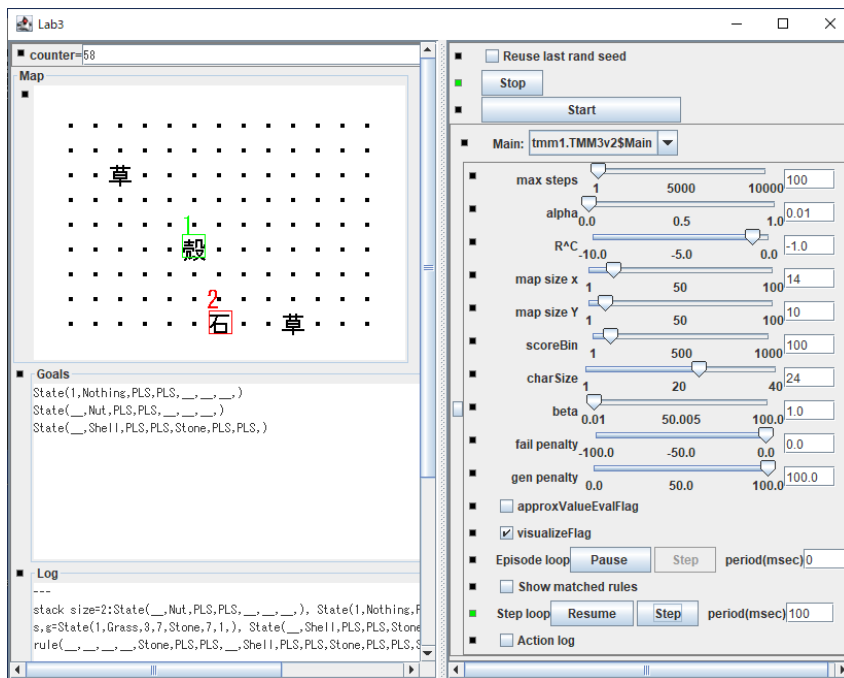


レジスタ2



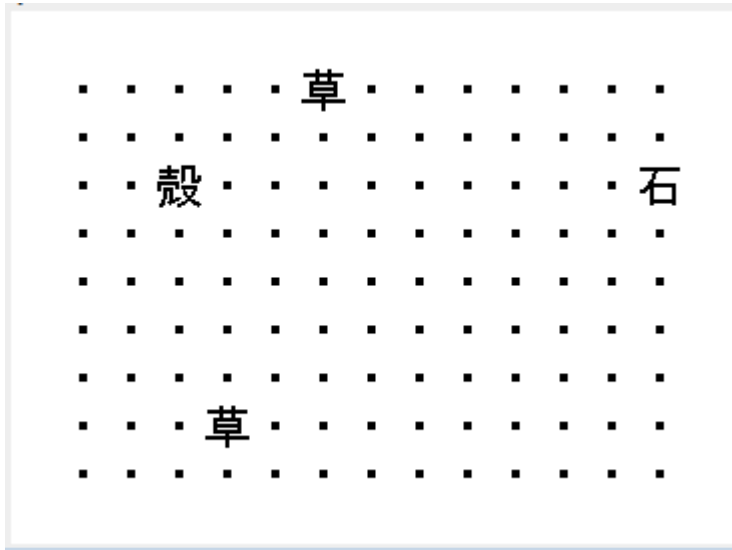
眼球は目の前の物体の中からランダムに1つ選んだ方向を向く。

提案モデルを使った認知アーキテクチャ のプロトタイプシステム



- 目的: 行動ルールの自律的獲得の際に原理的な問題がないかどうかの予備的な検討。
- ワーキングメモリに関しては**提案モデルに忠実な実装はしていない**。
 - 必要最低限の振る舞いを直接作り込み。
 - メモリの忘却はない。
- 基本構造はプロダクションシステム。
- システムの中核にサブルーチンの再帰呼び出しが可能な強化学習 RGoal [Ichisugi et al. 2019]を用いる。

殻に石をぶつけて実を食べるタスク



「石」を「殻」にぶつけると「実」になる。
「実」は食べられる。
「実」を食べると「空」になる。

エージェントがとれるプリミティブな行動：
物体2を物体1にぶつける: MoveO2toO1
物体1を食べる: EatO1

環境の初期状態の例

このタスクを解くプログラム（行動ルール集合）

ワーキングメモリの状態	サブゴールの状態	行動
1: rule((_, _ _ _ _ _ _),	(_, 空,+,+, _ _ _),	call(_, 実,+,+, _ _ _))
2: rule((_, 実,+,+, _ _ _),	(_, 空,+,+, _ _ _),	EatO1)
3: rule((_, _ _ _ _ _ _),	(_, 実,+,+, _ _ _),	call(_, 殻,+,+, 石,+,+))
4: rule((_, 殻,+,+, 石,+,+),	(_, 実,+,+, _ _ _),	MoveO2toO1)
5: rule((_, _ _ _ _ _ _),	(_, 殻,+,+, 石,+,+),	set(2, _ _ _ _ _ _)
6: rule((_, _ _ _ _ 石,+,+),	(_, 殻,+,+, 石,+,+),	set(1, _ _ _ _ _ _)

ワーキングメモリとサブゴールの状態は

7次元の整数ベクトル: (レジスタ番号, 形1,x1,y1, 形2,x2,y2)

“+” は 0 (unknown) 以外の値にマッチ

“_” は 0 を含む任意の値にマッチ

プログラムの実行: ワーキングメモリとサブゴールの値にマッチする「最も特殊なルール」を1つ選択して実行することを繰り返す。

プログラムの実行の流れ

時間

1: rule((_, _ _ _ _), (_ 空,+,+, _ _ _), call(_ 実,+,+, _ _ _))
3: rule((_, _ _ _ _), (_ 実,+,+, _ _ _), call(_ 殻,+,+, 石,+,+))
5: rule((_, _ _ _ _), (_ 殻,+,+, 石,+,+), set(2, _ _ _ _))
...
5: rule((_, _ _ _ _), (_ 殻,+,+, 石,+,+), set(2, _ _ _ _))
6: rule((_, _ _ _ 石,+,+), (_ 殻,+,+, 石,+,+), set(1, _ _ _ _))
...
6: rule((_, _ _ _ 石,+,+), (_ 殻,+,+, 石,+,+), set(1, _ _ _ _))
4: rule((_, 殻,+,+, 石,+,+), (_ 実,+,+, _ _ _), MoveO2toO1)
2: rule((_, 実,+,+, _ _ _), (_ 空,+,+, _ _ _), EatO1)
タスク終了

視線を向けた物体の情報をレジスタ2に入れる。
「石」が見つかったら次に進む。

視線を向けた物体の情報をレジスタ1に入れる。
「殻」が見つかったら次に進む。

考察

1: rule((_ _ _ _ _ _),	(_ 空,+,+, _ _ _),	call(_ 実,+,+, _ _ _))
2: rule((_ 実,+,+, _ _ _),	(_ 空,+,+, _ _ _),	EatO1)
3: rule((_ _ _ _ _ _),	(_ 実,+,+, _ _ _),	call(_ 殻,+,+, 石,+,+))
4: rule((_ 殻,+,+, 石,+,+),	(_ 実,+,+, _ _ _),	MoveO2toO1)
5: rule((_ _ _ _ _ _),	(_ 殻,+,+, 石,+,+),	set(2, _ _ _ _ _))
6: rule((_ _ _ _ 石,+,+),	(_ 殻,+,+, 石,+,+),	set(1, _ _ _ _ _))

- プログラムは非常に簡潔、汎用性が高い形で記述。
- 理由：
 - プログラムが物体の具体的な位置に依存しない。
 - 関係のない物体(「草」など)はデフォルトで無視。
 - ランダムな眼球運動が「ハードウェア」側で実装されている。
- 記憶コストが小さく、おそらく学習もしやすい。
 - 脳のモデルとしてもAGIアーキテクチャとしても有望。

強い制限があっても汎用性は損なわれない

- ワーキングメモリが保持できるのは定数個の物体の定数個の属性の組のみ。
 - 制限のおかげで自律的行動ルール獲得がおそらく可能。
- 物体そのものは脳内にはではなく環境にある。
→ **適宜物体を探して視線を向けることで、事実上、無制限の個数の物体を操作することができる。** 環境=チューリングマシンのテープ
- エージェントが環境モデルを獲得済みならば、脳内である程度多くの物体を操作するシミュレーションができる。
モデルベース強化学習 ← 提案アーキテクチャとは直交した機構

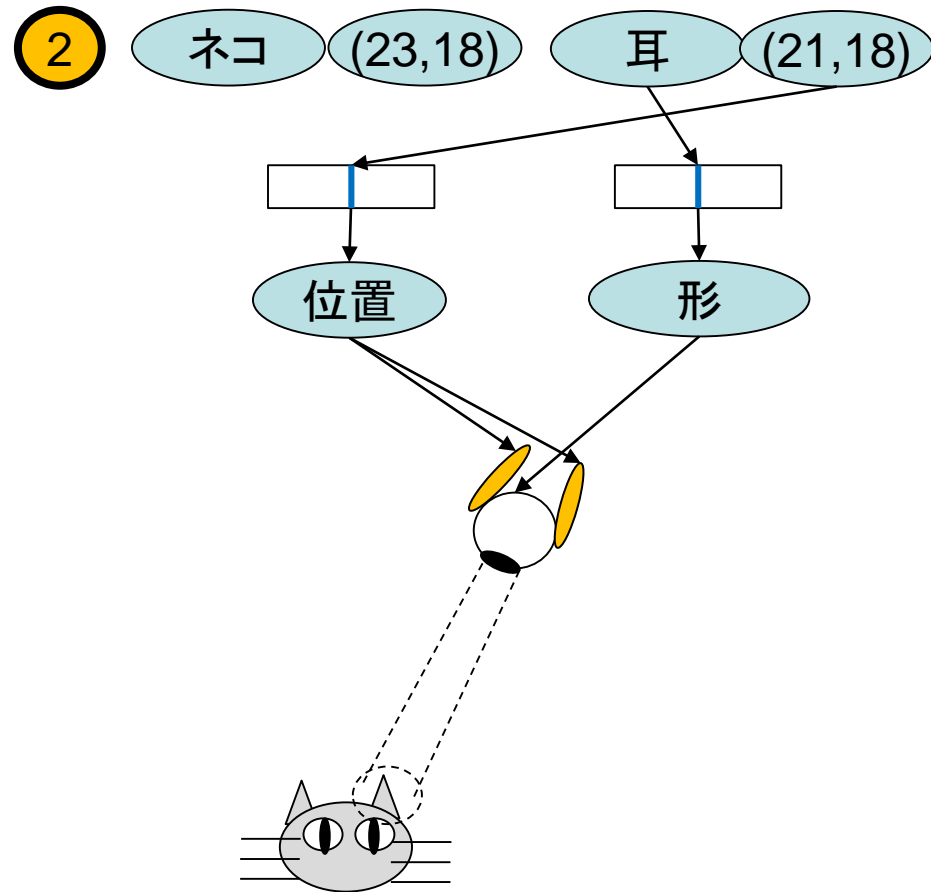
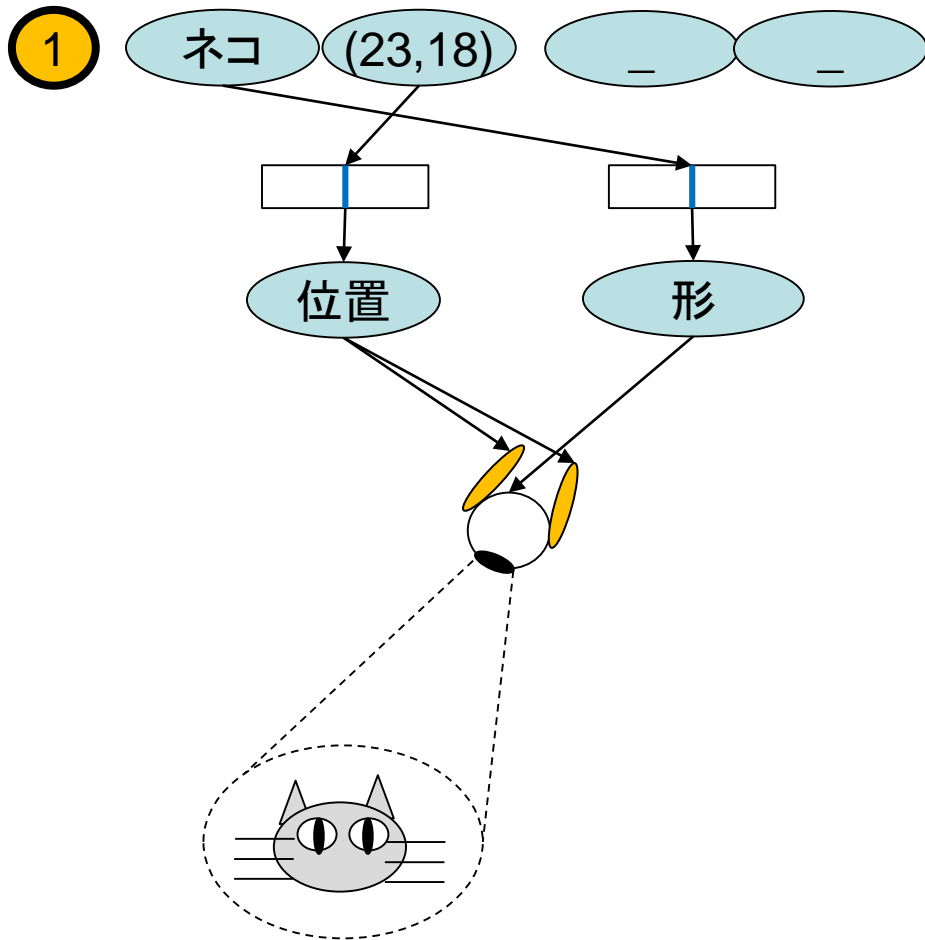
エージェントの視点からは、物体は
(シミュレーションされた)環境の中にある。



モデルの将来の拡張

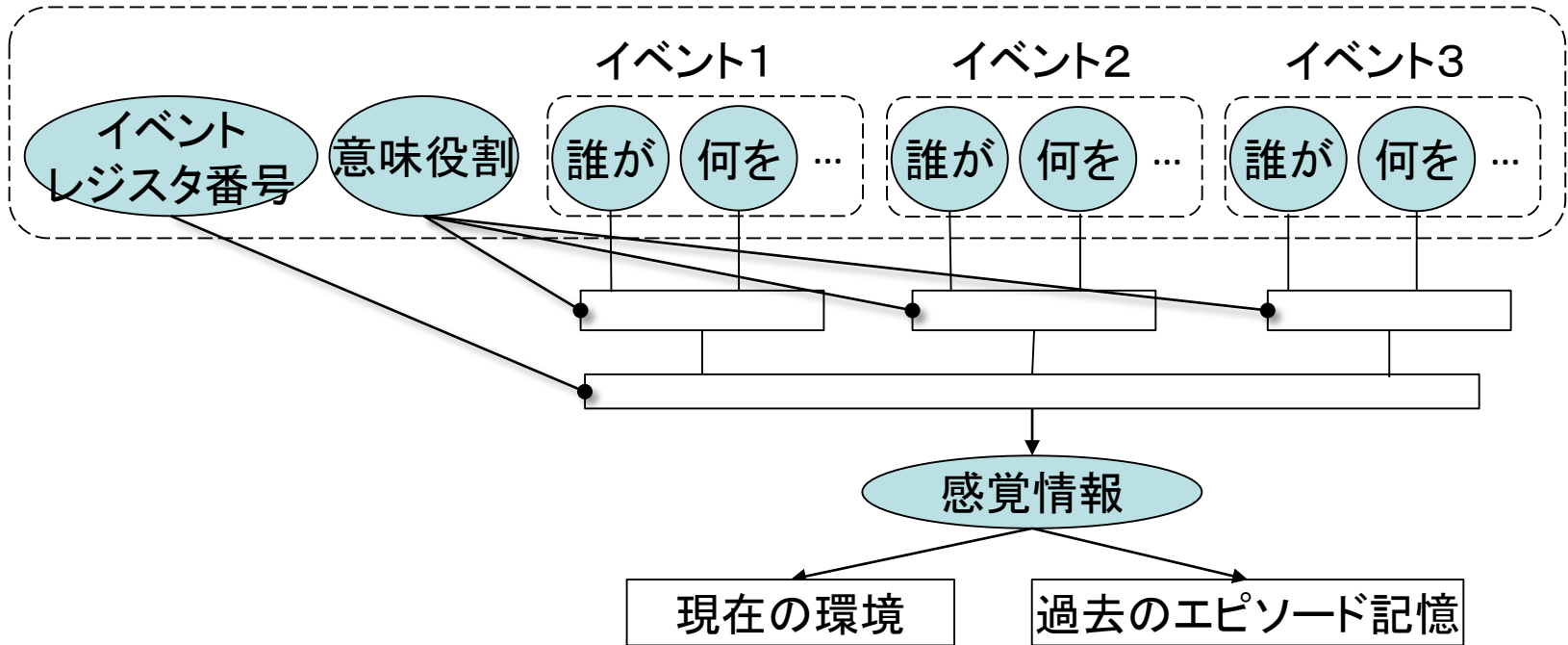
- ヒトのような知能を実現するためには、ワーキングメモリの構造をさらに拡張する必要がある。
- 物体間の関係の表現。
 - 「物体1は部品として物体2を持つ。」
 - 「物体1の上に物体2がある。」
 - 「物体1と物体2は似ている。」
- イベントの間関係の表現。
 - 「きのう戸棚にチョコレートがあったのを見たのに、いま戸棚にチョコレートがない。」
- 文の意味表現。
 - 記号 if, not, every, some, may, must などを使った、センサー入力に直接結びつかない情報の表現。

物体1は部品として物体2を持つ(has-a)



複数のイベント

ワーキングメモリ



イベント:「いつどこで誰が誰に何をどうした」という情報

文の意味表現

- 言語を使った対話や思考を行うために必要。
- おそらくヒト以外にはできない知的作業。
- 文は if, not, every, some, may, must などを使った、**センサー入力に直接結びつかない情報**を表現できる。
- 文の「意味」は文を解釈するプログラムで決まる。
(Cf. パースの記号論)
プログラムは教師なし学習と強化学習で経験から獲得可能。
→ 実装は将来の課題。

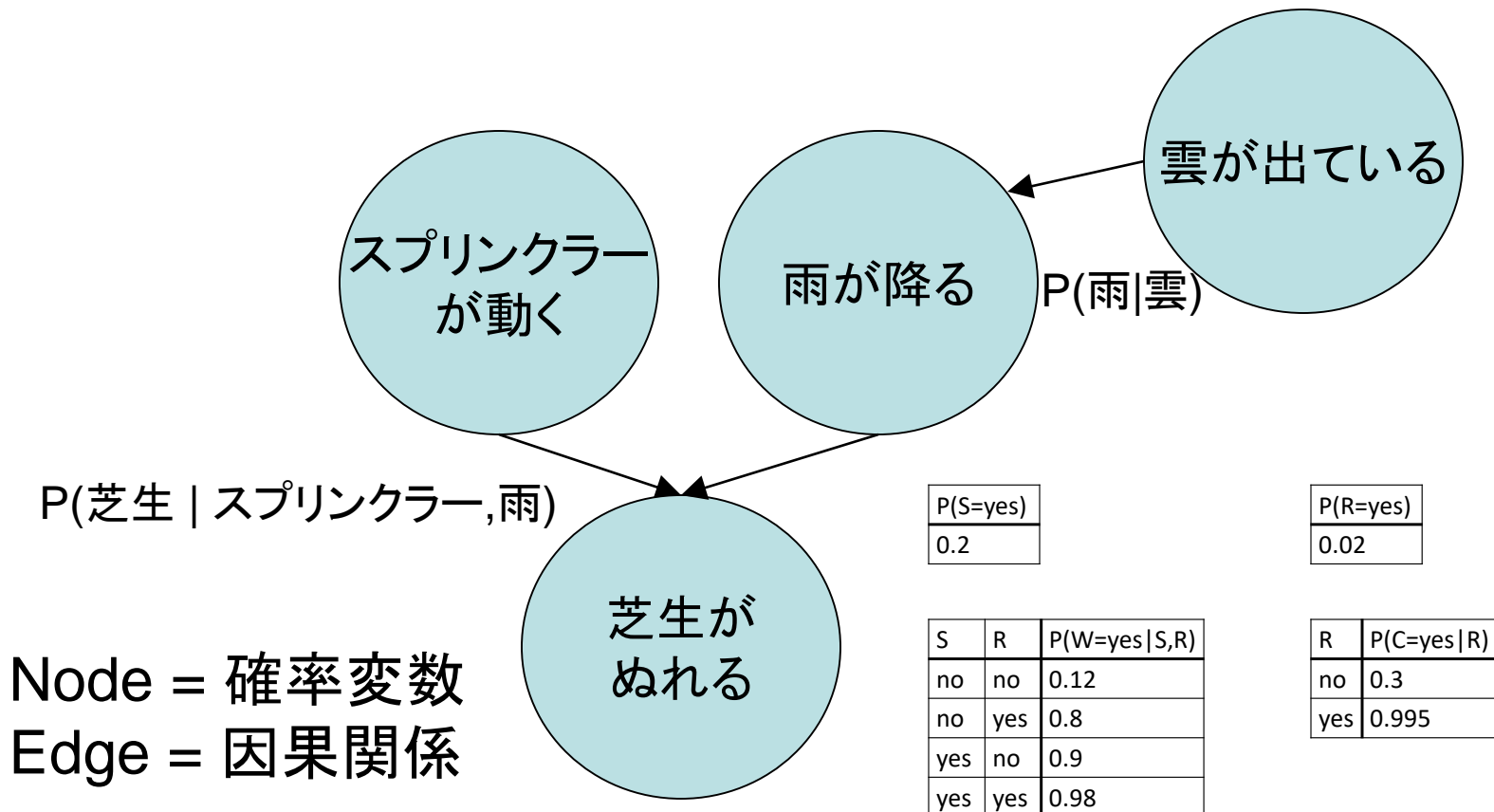
まとめ

- オブジェクトファイル理論を参考にしたワーキングメモリの機構をベイジアンネットを用いた生成モデルとして定式化した。
- 認知アーキテクチャのプロトタイプシステムを実装。
- ワーキングメモリを扱う行動ルールを経験から獲得する機構に関して理論的見通しが立った。実装は今後の課題。
- ワーキングメモリの構造のさらなる拡張について検討した。
 - 物体間の関係、イベント間の関係、言語の意味
- 本研究により、前頭前野のモデルとなるプログラム合成システムにおいて、データ構造の扱いに関する具体化の方向性が見えた。

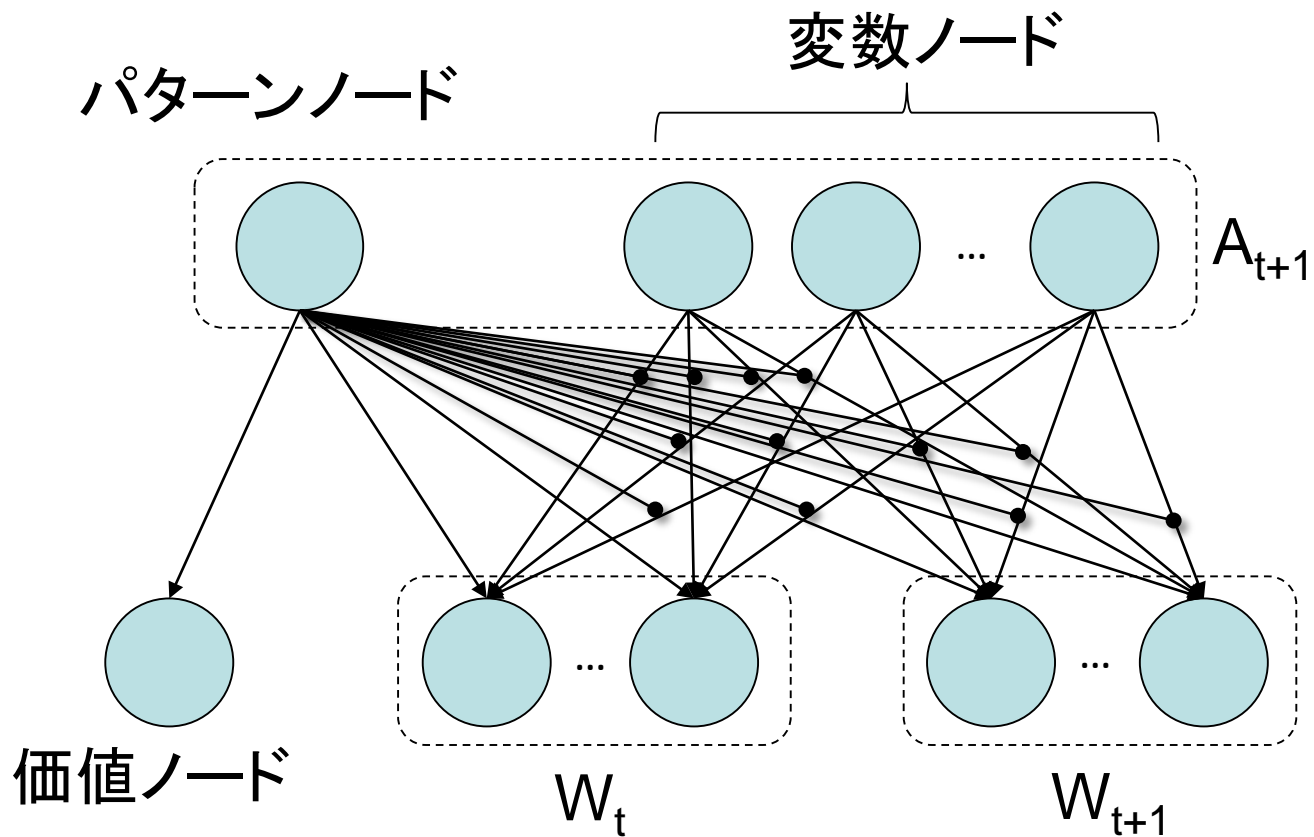
こういうアプローチの研究にご関心のある方、ご協力いたします。
過去の関連論文：<https://staff.aist.go.jp/y-ichisugi/j-index.html>

ベイジアンネットワーク [Pearl 1988] とは

- 確率変数の間の因果関係をグラフで**効率的**に表す**知識表現**の技術。



行動ルールを学習する回路



参照: [一杉 他 第15回 汎用人工知能研究会, 2020]

プログラム＝アルゴリズム＋データ構造

- アルゴリズムを表現するための機構：
 - メモリの参照・更新、条件分岐、I/O、サブルーチン呼び出し、例外処理の機構
 - これまでの RGoal を使った研究で実現の見込みが立っている。
- データ構造を表現する機構は？
 - 固定長のフラットなベクトルに符号化可能で、かつ複雑な情報処理が可能な「データ構造」が脳内にあるはず。それを明らかにするのが本研究の動機。