

## 発表要旨

### 1. これまでの研究業績

- (a) Fully Automated Cyclic Planning for Large-Scale Manufacturing Domains ≈ 1. In ICAPS14.
- (b) Solving Large-Scale Planning Problems by Decomposition and Macro Generation ≈ 1. In ICAPS15.
- (c) Tiebreaking Strategies for A\* Search: How to Explore the Final Frontier ≈ 1. In AAAI16. (*JSAI* 学生奨励賞)
- (d) Tie-Breaking Strategies for Cost-Optimal Best First Search ≈ 1. Journal of Artificial Intelligence Research 58 (2017): 67-121.
- (e) Exploration Among and Within Plateaus in Greedy Best-First Search ≈ 1. In ICAPS17.
- (f) Efficient Optimal Search under Expensive Edge Cost Computation ≈ 2. In IJCAI17.
- (g) Classical Planning in Deep Latent Space: From Unlabeled Images to PDDL (and back) ≈ 1. In KEPS17.

### 2. 今後の研究計画 : *Neural-Symbolic Hybrid System* による 次世代 AI システムの研究開発

※ 1 Masataro Asai, Alex Fukunaga

※ 2 Masataro Asai, Akihiro Kishimoto, Adi Botea, Radu Marinescu, Elizabeth Daly M, and Spyros Kotoulas

## 1 研究テーマ: プランニング(自動行動計画)とは?

プランナ=抽象行動列を求める人工知能プログラム



### プランニングの応用実績

汎用性  
ゆえに  
様々な  
応用例

Deep Space 1  
(Muscettola et al. '98)  
宇宙船の自律航行



Mars Exploration Rover  
(Bresina et al. '05)  
火星探査ロボットの  
自律行動システム

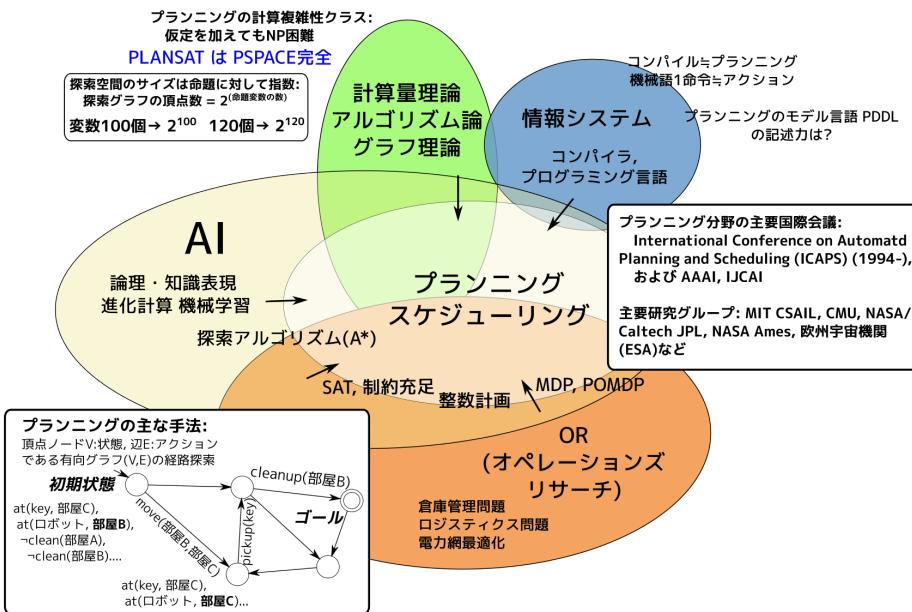


Core Security (セキュリティ企業)  
**CORE SECURITY**  
Thinking Ahead.

企業ネットワークをハッキングする  
ための手段を自動で生成する  
→穴探し、脆弱性を自動診断

手元にある問題をアクションとして  
論理の形に記述できれば  
目的達成の手段を自動で計算できる

## 2 プランニング(自動行動計画)分野の位置づけ



プランニング分野は、人工知能の専門分野という位置づけで、隣接するオペレーションズ・リサーチやアルゴリズム論などの分野の技術を利用してしています。特に、プランニング問題を解くのにはグラフ探索の技術が用いられます。

## 3 研究業績1: 査読付き学会論文 ICAPS14 (採択率33%)

**セル生産方式:** 工場生産方式の一種 (対義語: ライン生産)

ロボットアームなどを含む小規模セルが複数協調して製造



**目的:** 並列の製造プランを自動計画 + 製作時間を最小化

一台の組み立て問題を解くのはある程度複雑でも可能

**研究の対象**

✗ アクチュエータの角度制御 (例: 0: 0deg → 30deg, 1deg/sec)

◎ 行為の計画 (例: 部品2をアーム1で塗装, ネジ3を締め etc.)



→ しかも探索空間はその指數で増加  
解ける探索空間のサイズ:

以前:  $10^6 \rightarrow$  提案手法:  $10^{274}$

**問題点:** 握る対象の数が多くなる場合 (製品が4個程度以上の時)

最適解を返すソルバで直接解く → 探索空間が広大で探索不可

1つの製造手順を複数つなげて後処理最適化 → 生産時間が長い (makespan)

**手法:** 問題の繰り返し構造を「工場, セル生産, 製品」など前提なしに

**任意のプランニング問題** の論理構造から自動検出

検出した繰り返し構造から効率よい (生産時間の短い) プランを生成

**結果:** 1000台以上の製品を組み立てる問題を解いた

+ 生産時間を最大3割短縮(後処理最適化と比較)

新規性: 繰り返し構造を自動で抽出するというアイディア+抽出技術

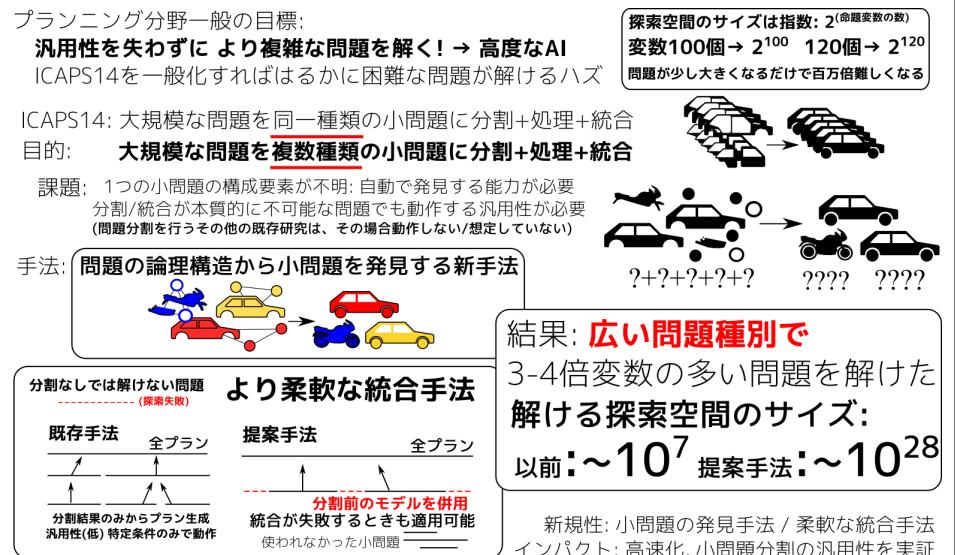
インパクト: 大きな高速化, プランニングの適用規模を広げた

株式会社IHIと共同研究(システムの先行評価として)

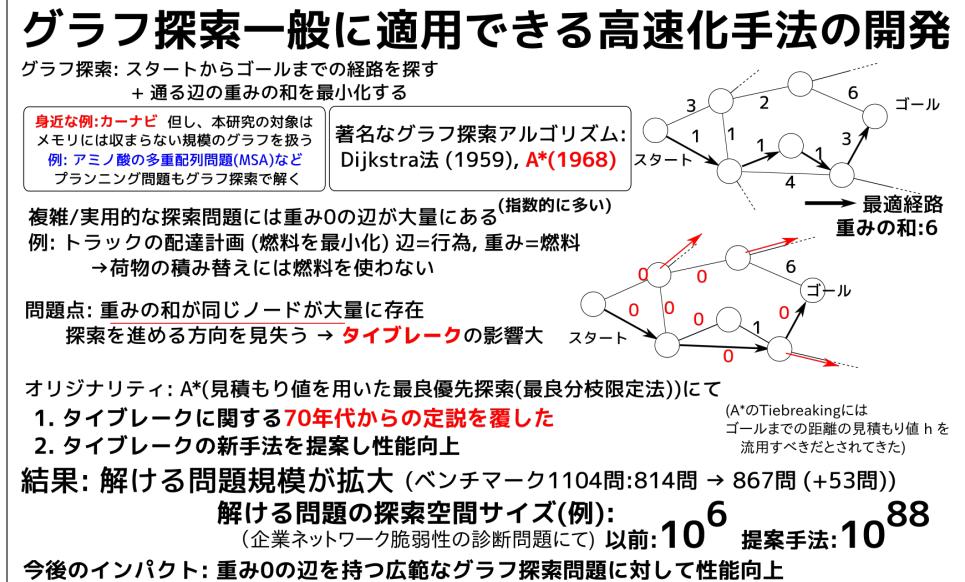
研究業績に移ります。ここでは、大規模なプランニング問題を解くために問題設定・ドメインによらず汎用に繰り返し構造を抽出する方法を開発しました。元の問題を繰り返し一周分の小問題に分割して解くことで、高速化と3割の生産時間短縮を達成しました。

大切なのは汎用性です。生産問題に限らず、掃除にも宇宙船にも同じ実行バイナリが使えます。

## 4 研究業績2：査読付き学会論文 ICAPS15 (採択率33%)



## 5 研究業績3：査読付き学会論文 AAAI16 (採択率26%)



続いて二、三本目の業績は、先ほどの手法で得られるのは1種類の小問題だけでしたが、これを複数種類の小問題に拡張しました。結果、より様々な問題で高速化を達成しました。ここまで汎用に小問題分割を適用した研究は、分野では始めてです。

最後に、申請後に行った研究が、難関国際学会 AAAI に採択されました。研究内容は、コスト 0 の辺を含むグラフを扱うグラフ探索アルゴリズム一般に適用できる内容で、非常に大きなインパクトを持つことが考えられます。

## 6 研究業績4：査読付き論文誌 JAIR (採択率12%)

### AAAI16の一般化：タイプブレーク≡非最適探索

問題点: 重みの和が同じノードが大量に存在 (ゼロコスト問題)

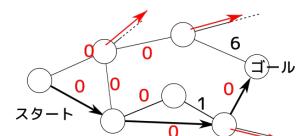
探索を進める方向を見失う → タイブレークの影響大

ノードの下界関数  $h$  の種類

許容的: 常に真のコストより低く見積もる → 最適解

非許容的: 非最適解を見つてしまふ ← 従来は使われない

Unit cost 関数: 辺コストを全て1で代用, 非許容的関数の例



主な発見: タイブレークとは 許容的関数  $h$  で作ったプラトーの中で非最適探索を行うことである

新規性: 最適探索で非許容的関数を効果的に使う方法を初めて示した

性能評価:

ゼロコスト問題のタイブレークに

非許容的 Unit cost 関数を使う

→ 指数的に高速化 (844問 → 906問)

→ AAAI16と同時に高速 (→ 911問)



理論的成果:

無限グラフでのA\*の完全性を

様々なタイブレーク法ごとに証明

→ 非最適探索の理論的結果を援用

**非最適探索用の全技術を最適探索に援用可能  
(従来は非互換と考えられてきた)  
→ 今後のさらなる発展が期待できる**

## 7 研究業績5：学会論文 ICAPS17 (採択率33%)

### 非最適探索の均一化手法の理解と改善

非最適探索: GBFS (貪欲最良優先探索) + 非許容的下界関数  $h$

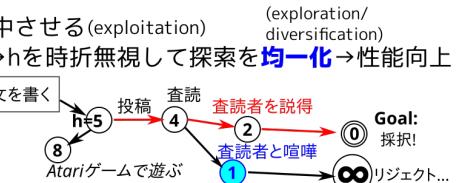
下界関数  $h$  を常に信頼して(貪欲)探索を集中させる(exploitation)

→ 非許容的  $h$  は常に正しいとは限らない →  $h$  を時折無視して探索を均一化 → 性能向上

問題点: アドホック手法が多い

→ どう組み合わせればよいか解らない

目的: → クリーンに評価しなおす



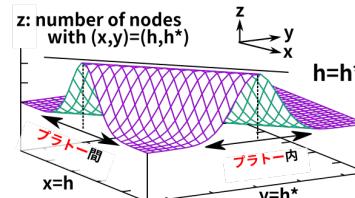
発見1. 2つの直行した均一化方法がある。

理由: 真の値  $h^*$  からの  $h$  の誤差は二次元的

→ プラトー間均一化と プラトー内均一化

→ 同じ均一化手法を二通りに利用出来る

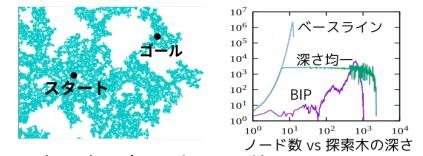
→ 直行だから同時利用してOK! → 性能向上



発見2. ボンドパーコレーション(BIP)

フラクタルを使った均一化手法

→ 性能向上の理由: 探索木の幅を削減



制限時間内で解けた問題数: 192 → 237.7  
参考: [Xie et. al. 2014] 深さ均一化 223.9

1. 未知の均一化アルゴリズムに汎用に適用できる法則
2. フラクタルを用いてより良い探索が出来る可能性

## 8 研究業績6：査読付き学会論文 IJCAI17 (採択率25%)

### 辺コスト動的計算が必要なグラフ探索の高速化

(IBM Research Ireland での研究)

- 問題点: 一回の辺コストを求める計算(外部ソルバ)が重い
  - 全コストを求めるのが現実的でない
  - A\*探索で必要になった辺コスト $c_a$ のみを動的に計算
  - それでもなお実行時間の90%を辺コスト計算に使用

例1: 辺コスト(都市間距離)が未知のTSP

都市数Nに対し $O(N^2)$ 回経路探索ソルバを呼び出す

例2: MWRP(複数ワーカー乗り換え問題):

[地図上の全場所] $^2 \times$ [全出発時刻] 個の二点間乗り換え

→全ての辺コストを求めるのは現実的でない

→コストを求めた後の問題自体もNP困難



解決法:  $c_a$ の下界  $c_h$ で代用、必要になるまで $c_a$ 計算を遅延

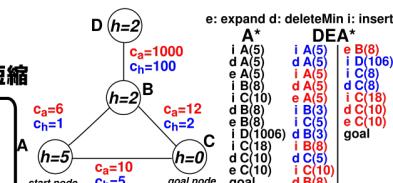
$c_h$ : 外部ソルバの下界閾値、 $c_a$ より高速に計算可能

同じノードを  $c_h/c_a$  で二回展開・キューに挿入

→多くの $c_a$ を計算せずにすむ → 探索時間を最大5倍短縮

多くの実問題は階層構造を持つ

既存外部ソルバを用いて汎用に適用可能



## 9 研究業績7：査読付きワークショップ KEPS (採択率60%)

### 深層学習(非記号型AI)と行動計画(記号型AI)の融合

画像で示された問題を

「パネル」「9マス」「動く」などの情報は一切与えられない問題を解いた例も与えられない

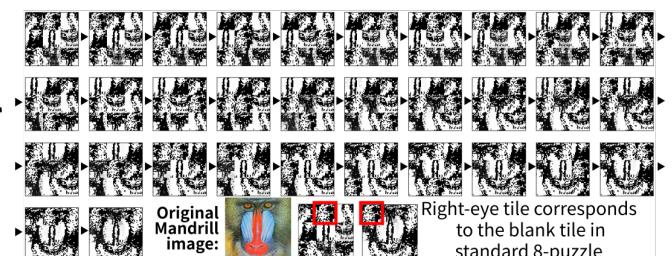
Initial state image  
Variational AutoEncoder  
Gumbel-Softmax を用いたVAEで記号的命題表現に自動的に変換し記号的プランニングソルバで解く



ノードの数を減らしてから元のサイズに戻すNN  
入力画像と出力画像が一致するように学習  
かつ、真ん中の層は0/1値: 何らかの命題の真偽値に対応

結果を画像に戻して  
プランを返却する  
システム

LatPlan

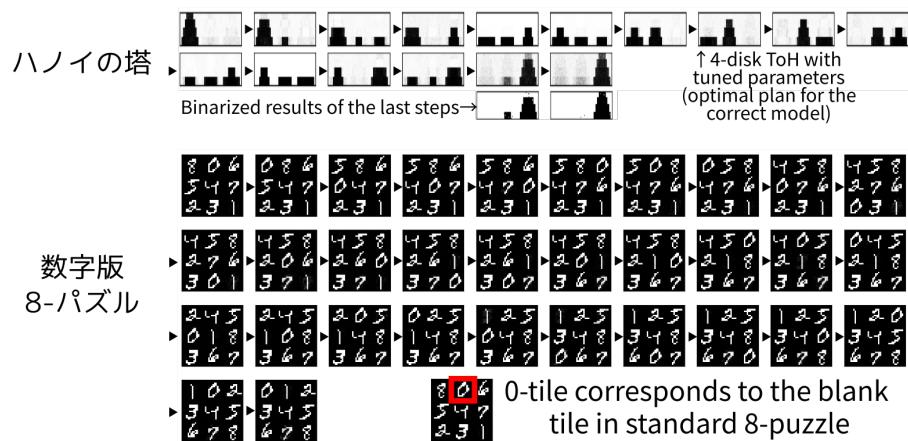


Original Mandrill image:  
Right-eye tile corresponds to the blank tile in standard 8-puzzle

## 9.1 研究業績7：査読付きワークショップ KEPS (採択率 60%)

### LatPlan: 同じシステムで全く異なる問題を解ける

問題ごとにニューラルネットの学習は必要



## 9.2 研究業績7：査読付きワークショップ KEPS (採択率 60%)

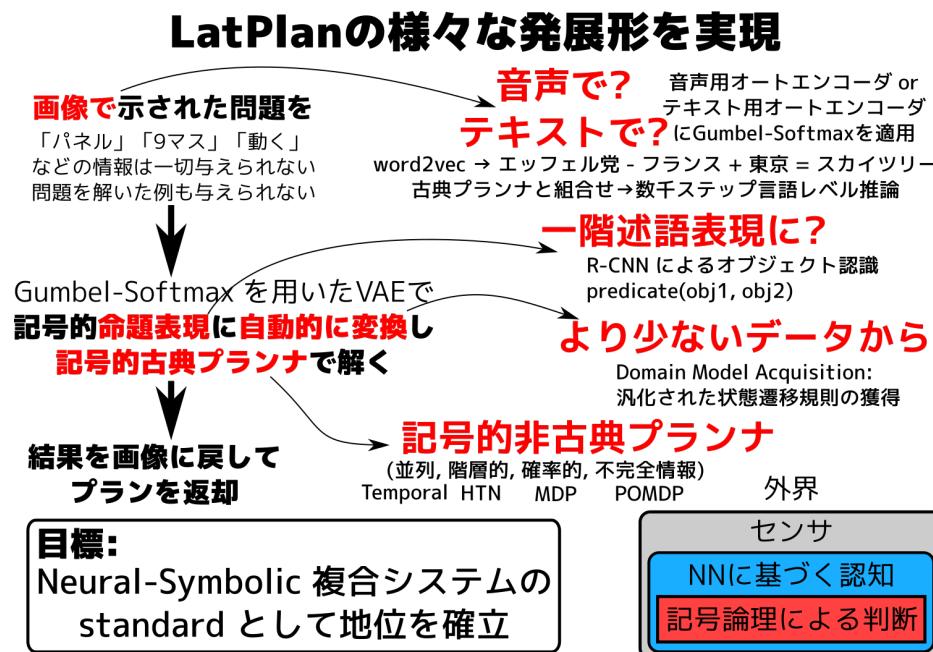
### 強化学習とは異なり、優れた理論的性質

アルゴリズムの完全性: 解が存在するときには必ず解を発見する  
解の最適性: 理論的性質のわかっている下界関数を使うことで保証

	入力・認識	意思決定
DQN	画像・ニューラル (事前知識なし)	Greedy+NN強化学習 =反射的エージェント
AlphaGo	ハードコード ("石", "マス目")	記号的 MCTS+NN強化学習 vs. イ・セドル 白74手 大幅に読み違え
LatPlan	画像・ニューラル (事前知識なし)	A*+許容的下界関数 (完全性+解の最適性)

ニューラル認識と記号的意思決定(理論保証付き)を持つシステム  
認識の誤差(NNに起因)はあっても意思決定の誤りは無い

## 10 今後の研究計画



## 11 まとめ

1. 難関国際会議 (33%) Fully Automated Cyclic Planning for Large-Scale Manufacturing Domains. In ICAPS2014.
  - (a) 任意の問題から 1 種類の繰り返し構造を自動で検出
  - (b) 工場での製造スケジューリング (x1000 高速化, 探索空間  $10^6 \rightarrow 10^{274}$ )

以上です、ありがとうございました。

2. 難関国際会議 (33%) Solving Large-Scale Planning Problems by Decomposition and Macro Generation. In ICAPS2015.
  - (a) 複数の繰り返し構造をより柔軟・汎用に組み合わせる手法
  - (b) ベンチマークセット全体で高速化 (x3-4 高速化, 探索空間  $10^7 \rightarrow 10^{28}$ )
3. 難関国際会議 (26%) Tiebreaking Strategies for A\* Search: How to Explore the Final Frontier. In AAAI-2016. (JSAI 学生奨励賞)
  - (a) コストゼロの辺がグラフ探索に引き起こす問題を解決
  - (b) コストゼロの辺を含むグラフ探索全てに影響 (探索空間  $10^6 \rightarrow 10^{88}$ )
4. 難関論文誌 (12%) Tie-Breaking Strategies for Cost-Optimal Best First Search. Journal of Artificial Intelligence Research 58 (2017): 67-121.
  - (a) (3.) に加え タイブレーキング と 非最適コスト探索の関連性を指摘
  - (b) さらに性能向上
5. 難関国際会議 (33%) Exploration Among and Within Plateaus in Greedy Best-First Search. In ICAPS2017.
  - (a) 非最適コスト探索をフラクタルを用いて改善
  - (b) プラトー内均一化とプラトー間均一化の直交性を実証
6. 難関国際会議 (25%) Efficient Optimal Search under Expensive Edge Cost Computation. In IJCAI-2017.
  - (a) 辺コストの動的計算が必要な問題に対して高速な最適アルゴリズム DEA\*
7. 国際ワークショップ (60%) Classical Planning in Deep Latent Space: From Unlabeled Images to PDDL (and back). Knowledge Engineering for Planning and Scheduling (KEPS) Workshop
  - (a) 画像から命題を自動生成して記号的 AI で組合せ最適化問題を解き、画像で出力するシステム

## 12 付録 古典プランニング問題とは(定義)

アクション集合 A, オブジェクト集合 O, 初期状態 I, ゴール G

状態 := 真である命題の集合

アクション  $a \in A : \{ \text{pre}(a), \text{add}(a), \text{del}(a), \text{cost}(a) \}$

ただし、 $\text{pre}(a)$ : 前提条件,  $\text{add}(a)$ : 追加効果,  $\text{del}(a)$ : 削除効果,  $\text{cost}(a)$ : アクションの適用コスト

状態 s に対するアクション a の適用:  $\text{pre}(a) \subseteq s$  の時に適用可能で、

$$a(s) = (s \cup \text{add}(a)) / \text{del}(a)$$

終了判定:  $s \supseteq G$  ならば ゴール達成

## 13 付録 古典プランニングを研究する意義は?

### "古典"プランニングを研究する意義は?

古典プランニングは

様々な拡張のベースとなる 最小限のフレームワーク

実験は古典プランニングでやった というただそれだけ

← 古典プランニングでは評価がやりやすい

- ・単純なモデル

- ・詳細が調べつくされている

- ・何が性能向上の理由なのか切り分けやすい(差分が取りやすい)

直接の実用的な  
意義もある



様々な拡張の「基礎」となる基盤技術

古典プランニングで成功した技術は  
他の問題にも使える!

類似のシナリオ:SATソルバ  
同じく基礎的なモデル



## 14 付録 AIの倫理について

- 研究内容は漠然とした「AI」のうち グラフ探索の研究
- 善悪の判断はそれ自体は行わない
- 値値判断は与えられる入力の中にある = 使用者の価値観を反映する
- 悪用の問題はある。しかし、自分としては、災害救助ロボットなど、人道的な応用を目指している

## 15 付録 ディープラーニングとどう違うのか

- 機械学習を埋め込むことは可能だ
- が、求められる推論の複雑さが根本的に違う → 独立した分野

ニューラルネット, DL, 強化学習

- 入力: 現在のデータ、過去の履歴、報酬 etc..      プランニング出力:
  - **10ステップ, 100ステップ先の未来を先読みした行動計画**
- 出力:
  - 次の**1ステップ**のアクション選択ポリシー(強化学習)
  - 固定長の分類結果(画像認識)
  - ある意味状況に応じて脊髄反射なエージェント

ただしプランニングに学習機を埋め込むことは可能(実例複数あり)

DLから見て、プランニングはアプリケーション

プランニングから見て、DLはツール

両者の組み合わせは機会があればやってみたい

### 15.1 付録 ディープラーニング関連

趣味の一環で、Common Lisp から直接使える GPGPU のライブラリを作成中(DL を作ってみるため)

- OpenCLベース
- Lispの文法を直接OpenCL Cに変換し実行するトランスレータ
- OpenCLのメモリ管理をLisp GCに埋め込み

## 16 付録 第五世代コンピュータとの違いは?

第五世代コンピュータ: 並列推論機械(Prologベース, ハードウェア, OS)

根本的なソフトウェア技術、探索技術が未発達だった

第五世代	現在
後方全探索+バックトラック	前方ヒューリスティック探索
Prologベース	C/C++で高度に最適化されたプログラム State packing, 決定木, mutex...

今はベンチマーク問題 **1104問** のうち 5 分で **800問** 前後解ける

仮に当時のソフトウェアを現在のハードウェアで動かしたとしても、**100問**も解けないだろう

## 17 付録 Explicit Graph と Implicit Graph との違い

カーナビ、ソーシャルグラフなど: Explicit Graph Search  
グラフ全体がメモリ (~数ペタバイト) または二次記憶 (~数ゼタバイト) に収まる

参考: 2012 年の全世界のデジタルデータ: 数ゼタバイト ( $1ZB = 10^{21}$  バイト)

AI and Web の分野など

プランニングにおける探索グラフ : Implicit Graph Search

地球上に存在する全計算資源を集めて二次記憶に入らない

グラフのノード数は状態変数に対して指数的に増加

動的に必要な量のみメモリ確保をしないと問題が解けない

探索空間サイズの例:

$3 \times 3 \times 3$  のルービックキューブ:  $4.32 \times 10^{19} = 4$  エクサバイト

$4 \times 4 \times 4$  のルービックキューブ:  $7.40 \times 10^{45} \sim 10^{24}$  ゼタバイト

$5 \times 5 \times 5$  のルービックキューブ:  $2.83 \times 10^{74}$

## 18 付録 似たような研究は誰がやっていますか どこでやられていますか

国内ですか、国外ですか？

YES NO

## 19 付録 プランニングはマイナーで大したことのない分野?

日本にプランニングの研究室がない ≠ 世界で研究室がない

ICAPS, SoCS: 例年 150 人-200 人の参加者を集めており大変盛況,

AAAI, IJCAI: プランニングに関する論文は例年数十本採択 Proceedings の一つの章をなす

JAIR, AIJ: 論文誌でもプランニングの論文は多い (JAIR Volume 54: 12 本中 2 本がプランニング論文)

主な研究室:

MIT CSAIL (Brian Williams), Carnegie-Mellon,

NASA (NASA Ames および NASA/Caltech JPL のそれぞれに 20 名以上の研究者), 欧州宇宙機関 (ESA)

指導教員は NASA JPL AI Lab の元メンバー

## 20 付録 汎用性を失わずに解く?

**No Free Lunch 定理:** 最適化アルゴリズムの性能は 全問題の平均を取れば 全て同じ

Q. NFL 定理のもとで「汎用性を失わずに高速に解く」というのは不可能?

A. NFL 定理は確かにそのように主張するが、プランニング分野の意味する「汎用性」は

人間にとて有意義な問題の集合 における汎用性である。

全プランニング問題の集合  $\supseteq$  人間にとて有意義な問題の集合

従って、全問題の平均を取れば という前提が成り立たない。

## 21 付録 その研究は...

重要度	評価	オリジナリティ	21.1 過去と未来で誰が似たようなことをやっているか	動機
ACP	発表した 難関学会	ループの概念を検出	はるかに巨大な問題 この専門分野をやっている人は少ないん <small>で規模問題</small> 少し離れているが最も似ている研究という	産業応用 人間プログラミング
CAP	難関学会 発表した いくつか質問された メールやり取り	問題分割手法 柔軟な統合手法	SAT ソルバの研究の人はいる – 田中先生 大規模な問題 推論系 <small>それまでの分割系の手法より</small> logic and reasoning Lemma Reusing for SAT based Planning and Scheduling ICAPS 10 年ぐらい前 NII の井上勝海 CSP でプランニング	産業応用 混ざった問題
AAAI16	三人中 二人の査読者に絶賛された	同コストのノードの分類	神戸大学田沼先生 SAT 70 年代からの定説を覆す 九大 犀川誠一先生 A*AS 性能改善 自分の指導教官がアメリカから飛んできた研究者なので、もともと日本にいなかった研究者だった スライド 2 を見る	広範なグラフ探索問題 下界以外で改進 基礎技術 CSP
過去全体			ERATO の人は同じくグラフ探索をやっていますが、 explicit/implicit の違いがあります。瓦林先生秋葉さん	
未来全体			Katsutoshi Hirayama, Kobe university (constraint optimization, CSP, distributed CSP)	
過去未来			Toru Ishida (Kyoto U. stopped working in search a long time ago, but for a while, he was the leader in A*-related search methods in Japan, coauthored some papers with Korf).	グラフ探索の改善 これがわかる
対称性	どちらも underinvestigated, もっとしらべるべき			産業応用 これがわかる
問題分割	とされている			どちらも綺麗
融合				産業応用

## 22 付録 学会論文の位置づけ

## 23 HTN と ICAPS-15 の違い

HTNは人間が問題分割を行う人件費を考えると非常にコストパフォーマンスが悪い→自動で問題分割

## 24 付録 Q&A

- Q. 探索空間の比較について、なぜ「以前」の数字がスライドによって変わるの?
- A. 論文の中で使った実験設定が異なるからです。

## 25 付録 今後のキャリアパスは?

研究者押しで押しまくる大学あるいは企業の研究者迷うことなくパッと答える

## 26 付録 今後の研究計画が、インパクトの少ない今までの延長のように見えるんだけど…

## 27 付録 アドホックな解法に見えるんだけど…

## 28 付録 失敗しそうなんだけど…

## 29 付録?? とにかくまとはずれな質問

引き出しを探るタイプの質問

上席研究員主任研究員

senior engineer

4段階

下 associate XX senior principal 上

PI principal investigator

time sink