

The diagram illustrates the components of a computer system. A large yellow rectangle is labeled 'コンピュータ' (Computer). Inside this rectangle, there are two smaller boxes: a blue box labeled '電子回路(演算装置)' (Electronic circuit (Arithmetic device)) and a light blue box labeled 'プログラム' (Program). A line connects the two boxes. To the right of the yellow rectangle, there are two labels with arrows pointing to the components inside: 'ハードウェア' (Hardware) points to the '電子回路(演算装置)' box, and 'ソフトウェア' (Software) points to the 'プログラム' box. Below these, a separate box contains the text: 'プログラムに書かれた計算手順・戦略のことを「アルゴリズム」という。' (The calculation procedure and strategy written in the program are called 'Algorithm').

コンピュータ

電子回路(演算装置)

プログラム

ハードウェア

ソフトウェア

プログラムに書かれた
計算手順・戦略のことを
「アルゴリズム」という。

- コンピュータで様々な計算を行う際に、アルゴリズムの工夫によって、
何十倍、何百倍も計算時間が違ってくることもある。

1

計算時間の影響

- 速いか遅いかの違いだけと軽く考えてはいけません。
→ **とても重要!**
 - 3秒で終わるなら対話的に仕事ができるが、3分かかるとなると、ずっと座って待つてはいられない。
(web検索に1回3分かかったら今のインターネットはない)
 - 新作ゲームの設計がクリスマスに間に合うかどうかで、メーカーの売り上げが大きく影響を受ける。
企業の存続/倒産が左右されることも!
(従業員や家族の人生にまで影響が及ぶ)
 - ある計算に100年かかるとしたら、その人にとっては解けないのと同じ (人の寿命は有限)

2

計算時間を短縮するには

- スーパーコンピュータを使う？
 - 神戸に建設されたスパコン「京」は64万個以上の演算装置を並べて毎秒1京回(兆の1万倍)計算できる。
- ただし誰もが使えるものではない
 - 開発費: 約1000億円、 運用費: 約80億円/年(うち電気代20億円/年)
→ 国家的/世界的に重要な問題を解くために使われる。
- **アルゴリズム技術**
 - 与えられた問題を解くために、できるだけ少ない基本演算回数で済まそうとする工夫
 - 同じコンピュータでも、何百倍も速くなることがある。
 - または、もっと安価で低消費電力なコンピュータで済む。



京 K computer

©RIKEN

3

BDD(二分決定グラフ)

離散構造の最も基本的なモデルである「**論理関数**」の処理技法

1986年に画期的なBDD演算 (**Apply演算アルゴリズム**)を提案。
以後急速にBDD技術が発達。
(長期間、情報科学の全分野での最多引用文献となった)

Bryant (CMU)

BDD → AND → BDD
BDD → (BDD同士の論理演算) → BDD
(圧縮データ量にほぼ比例する計算時間)

近年のPC主記憶の大規模化により、BDDの適用範囲が拡大 (特に2000年以降)

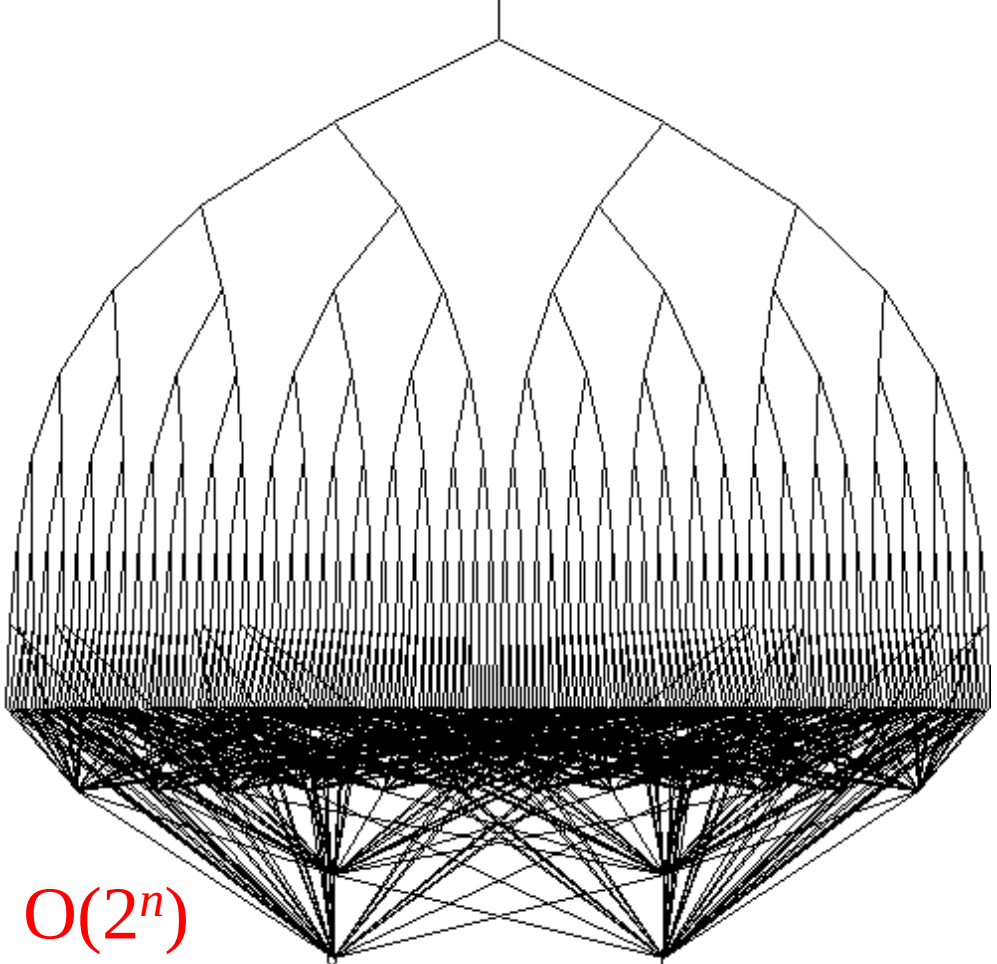
簡約化 (圧縮)

(BDD) (場合分け二分木)

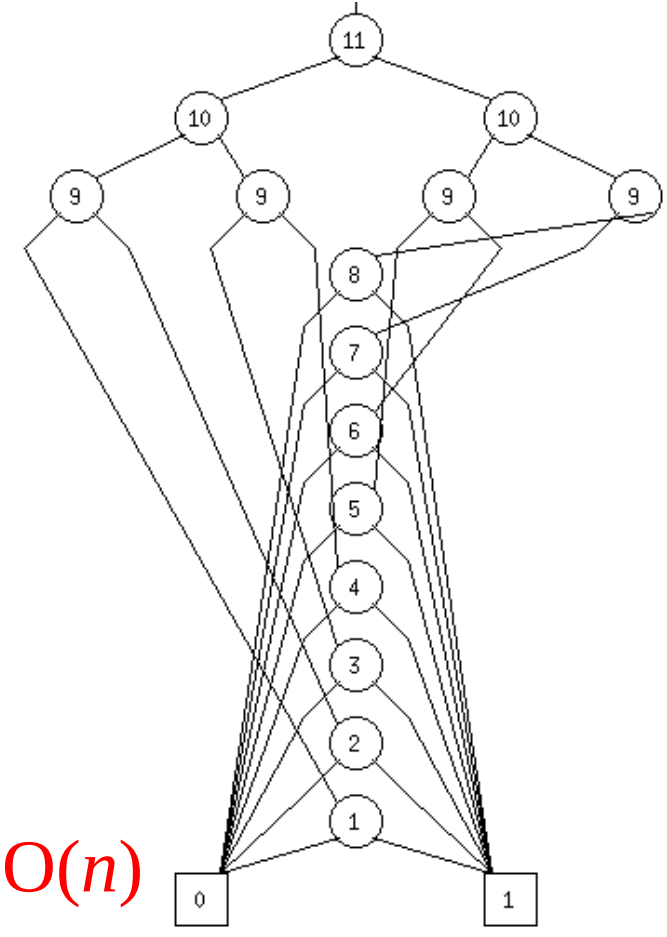
- 場合分け二分木グラフを簡約化(データ圧縮)
- 多くの実用的な論理データをコンパクトかつ一意に表現。(数十～数百倍以上の圧縮率が得られる例も)

BDD簡約化の効果

- 特定の問題では、指数関数的な圧縮効果が得られる。
 - 例題に依存するが、多くの実用的な問題で効果がある。



$O(2^n)$



$O(n)$

論理関数と組合せ集合

論理関数:
 $F = (a \wedge b \wedge c) \vee (\sim a \wedge b \wedge c)$

組合せ集合:
 $F = \{ab, ac, c\}$

→ ab (買った物客の購入品)
→ c
→ ac

■ 組合せ集合と論理関数の演算は対応関係がある。

- Union of sets → logical OR
- Intersection of sets → logical AND
- Complement set → logical NOT

a	b	c	F
0	0	0	0
1	0	0	0
0	1	0	0
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	0
1	1	1	0



ZDD(ゼロサプレス型BDD)による集合の表現

- 「組合せ集合」を効率的に表現するためのBDDの改良
 - 湊が世界で初めて考案し命名(1993年)
 - 通常と異なる簡約化規則を考案。
 - 疎な集合の族を扱う場合に著しい効果が得られる。
(例：商店の陳列アイテム数に比べて1顧客の購入点数は極めて少ない。)

通常のBDD

ZDDの簡約化

- ZDDはBDDの改良技術として現在、世界的に広く使われている。
 - 最近では、データマイニング分野に应用されて、画期的な有効性が示されている。(数百倍のデータ圧縮率・数十倍の処理高速化)
 - 他にも応用例は増えつつある。

日本科学未来館での成果展示

- 日本科学未来館(東京・お台場)で
我々の研究プロジェクトの展示
「**フカシギの数え方**」を開催

- 2012年8月1日～2013年4月25日
(期間中、夏休み冬休み1回ずつ)
- 小中高生・一般市民向けに
研究内容をわかりやすく展示
- 期間中に23万人が来場
- 北大総合博物館や札幌市青少年
科学館でも再展示を実施

Miraikan

at University of Tokyo Museum, Ordinatio, The 13th Exhibition
"The Art of 10th - Interdisciplinary Ventures -"

JST ERATO Yukio Ikenoue Science Visualization Project
JST ERATO 定規館長科学可視化プロジェクト

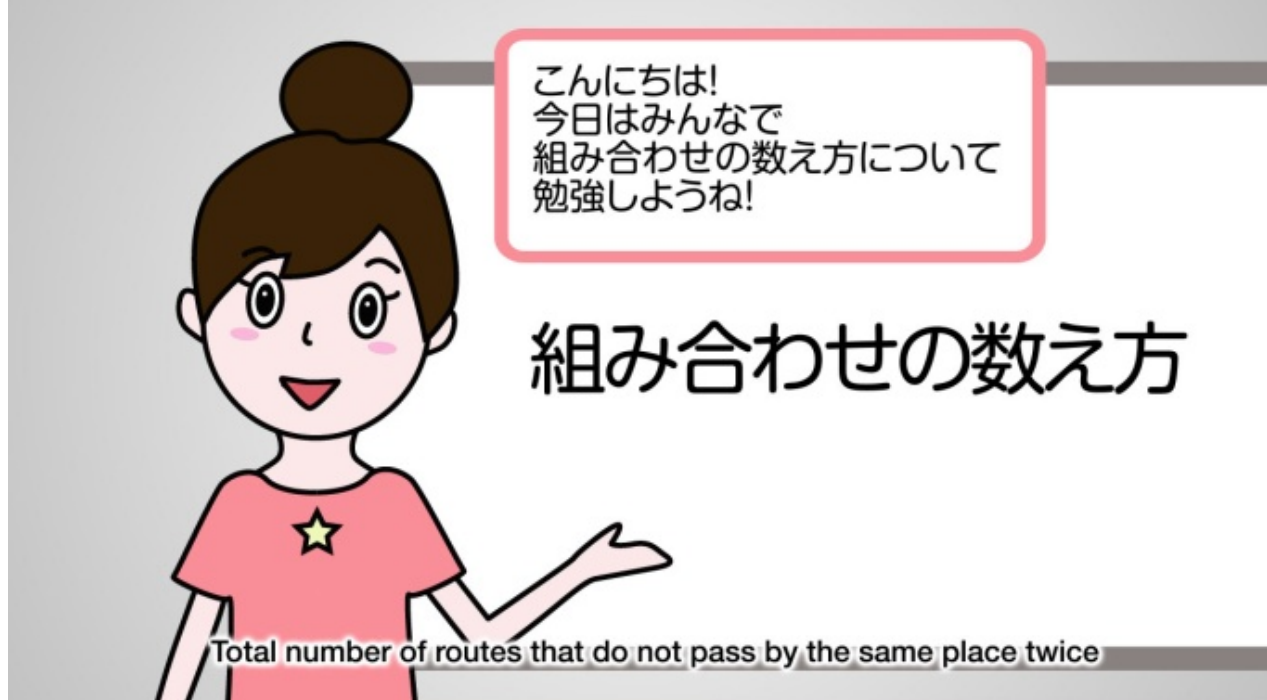
「の」「フ」「数」「え」「方」の「フカシギの数え方」第1回開催分

2012. 01
(wed.)

2013. 02
/ 25
(mon.)

アニメーション動画の反響

- 「フカシギの数え方」で、本プロジェクトの企画・監修によるアニメーション動画を製作
- **YouTube再生回数が現在230万回超**
(非公式サイトを合わせると500万回以上)
- サイエンス系コンテンツでは極めて異例の大ヒットに



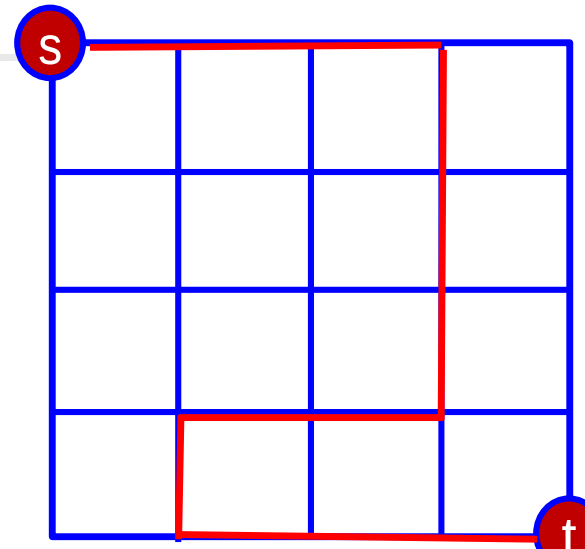
組み合わせの数え方

Total number of routes that do not pass by the same place twice

9

「おねえさんの問題」の豆知識

- 最短経路の数え上げは簡単
($\rightarrow {}_{2n}C_n$; 高校で習う問題)
- 遠回りを許すと突然難しくなる。
 - 簡単な計算式は見つかっていない。
 - 湊が考案した「ZDD」と呼ばれる技法を使って
高速に数え上げる方法が、最近開発されている。
 - アニメ動画で、おねえさんが25万年かかった計算結果の
数字が出ているため、「公式を教えて」というコメント多数。
 $\rightarrow$ 「どうも公式ないらしいよ」
 $\rightarrow$ 「ZDDというのを使いたいよ」



10

[illegible]

離散構造処理系プロジェクトの基本構想

■ 様々な工学的応用を持つ基盤技術として
「離散構造処理系」に着目し、研究開発を行う

システム設計自動化

データマイニングと知識発見

大規模システム故障解析

機械学習と自動分類

制約充足問題

生命情報科学

web情報解析

工学的応用
 → 社会への
 影響大

離散構造処理系

本研究の
 対象領域

性能向上
 (10倍～
 100倍以上)

集合論

記号論理

帰納的証明

組合せ論

グラフ論理

確率論

数学的
 概念構造

超高速グラフ列挙アルゴリズムの公開

- プロジェクトの成果として
超高速グラフ列挙索引化ソフト
「Graphillion」を開発・公開
 - Pythonをベースとした
ライブラリパッケージとして提供
- Graphillionの活用法と内部の
アルゴリズム技法をまとめた
専門書を出版
 - プロジェクト関係者13名の共同執筆
- 2015年4月8日発売開始
 - 予約好調につき
発売前に重版増刷決定

1

オープンソフトウェア: “Graphillion.org”

- ZDDを用いた超高速グラフ列挙のツールボックス。
– Pythonベースのライブラリとして提供



The screenshot shows the GitHub repository page for `takemaru/graphillion`. The repository is described as a "Fast, lightweight graphset operation library". It has 363 commits, 7 branches, 4 releases, and 2 contributors. The commit history table shows the latest commit is "Merge branch 'v0.95c'" by `takemaru`, dated 11 days ago. Below the table, there are links to `cmake` and `doc` files, and a link to "make figures smaller".

US map in Knuth-book

US map in Knuth-book

(133)

15

パス列挙問題 実験結果

日本地図グラフ

北海道から鹿児島までの全パス

	頂点数	計算時間	BDDノード数	パスの数
日本地図	47	0.01秒	951	1.4×10^{10}
2重化	94	248.72秒	18,971,787	5.0×10^{44}

5039760385115189594214594926092397238616064 通り

(= 503正9760澗3851溝1518積9594杼2145垓9492京6092兆3972億3861万6064)

16

条件付きパス(サイクル)の列挙

ERATO MINATO ZDD Project

17

鉄道ナビゲーションシステム Ekillion

go! tokyo 1131203 1131204 weight=stations order=dsc topk=10 inc=exc= outmode=1 access 10441

大蔵 東京 福岡 新潟

始点駅 御茶ノ水・JR中央

終点駅 水戸橋・JR中央

順位付け 昇降ヤ 昇降▲

表示件数 10

ルートに含め希望

ルートに含めなし

Longest self-avoiding cycles (343 stations)

6,482,787 self-avoiding paths From Ochanomizu to Suidobashi

トップ10 / ルート条件数9452787

川→10000001 駅数:344 駅数:343

川→10000002 駅数:344 駅数:343

川→10000003 駅数:344 駅数:343

川→10000004 駅数:344 駅数:343

川→10000005 駅数:344 駅数:343

川→10000006 駅数:344 駅数:343

川→10000007 駅数:344 駅数:343

川→10000008 駅数:344 駅数:343

川→10000009 駅数:344 駅数:343

川→10000010 駅数:344 駅数:343

電力網への応用 (2012年2月プレスリリース)

林 泰弘 教授(早稲田大)との共同研究

電力系スマートグリッド業界のリーダ的存在 (経産省スマートハウス標準化検討会座長、他多数の要職)

電力網最適化の研究で1990年代より湊と協力関係

大震災後、より緊急性の高い研究課題に

今後長期的に不足する電力を自然エネルギーで補うために必須の電力網解析・制御技術を支援

情報科学の研究者集団として我が国の苦境を克服するためできる限り貢献したい。

ERATOプロジェクトでの取り組みを加速

電力網のスイッチ制御

各区域が必ず1ヶ所の変電所設備に接続

停電しない

異なる変電所系統をつながない

電流が多過ぎると電線が焼ける

遠くに伝えると電圧が下がる

膨大な数の構成が存在

右のような14スイッチの小さな例題でも16384通りの組合せの中から210通りを探し出す問題

標準的な配電網の例では468個のスイッチがあり、10の140乗の組合せから探し出す問題に

電力網のスイッチ制御

ZDDを使った新しい高速アルゴリズムを開発。

標準的な電力網モデル (スイッチ468個) でグラフの制約と電氣的制約を共に満たす解を全て求め、圧縮して表現することに成功

電力の損失を最小にする組合せを求めることに成功

圧縮データ: 約110万個 (779MB) 実行時間: 数十分

解の個数: 約10⁶³ (2136那由他8201阿僧祇3834恒河沙8532極9116載8261正2214澗8049溝560積9817杼8392垓4438京5235兆3981億8952万1540) 通り

電力網のほかにも様々な応用がある

避難所の配置問題

道路、鉄道、ガス、水道、通信

選挙の区割り問題

建物のフロアプラン

避難所割り当て問題

京大建築系・加藤直樹先生他との共同研究

京都市上京区の避難所割り当て問題に適用

オペレーションズリサーチの国際会議 (ISORA 2013) で発表

Figure 13: Number of population in each cell

Figure 14: The optimal pattern with respect to Criterion 1.

Figure 15: The optimal pattern with respect to Criterion 2.

Figure 16: One of the pareto 22

ZDDによる区割り列挙

統計的疫学調査におけるホットスポットの抽出

石岡先生、栗原先生 (岡山大)、水田先生 (北大)、川原先生 (NAIST)

日本の都道府県の連結ブロックが何通りあるか → 明治の都道府県設置以来、初めての結果

(独)統計センターから公式データとして公開

計算時間 0.346359秒 ZDDノード数: 12076 解総数: 約1098億通り

北大・NAIST・岡山大・統計センターから研究成果を報道発表 (2016年9月2日)

共同通信社からニュース配信され、多くの地方紙web版に掲載

統計連合大会で本成果に関する企画セッションを開催

東京新聞 9/2朝刊

国際的に有名なホットスポット例題への適用

米国N.Carolina 州内100都市の乳児突然死亡率

連結ブロックを全列挙する ZDDの構築に成功

The mortality rate per 1,000 live births

0.00 2.00 3.00 4.00

0.00 1.00 3.00 4.00

1.00 2.00 3.00 4.00

(a)

(b)

(c)

選挙区割りへの応用

川原先生、堀田先生 (文教大) 他との共同研究

一票の格差最小だけではなく地理的・社会的制約条件があるため、列挙して解の分布を示すことに社会的意味がある

茨城県の例: 頂点数41、辺数87、分割数7

解の個数は11893998242846個

その中から格差が1.4倍以下のものを抜き出すと 25730669個 (1925.21秒で計算)

おねえさんのLINEスタンプ

2015年10月より公開中

120円で40ボタンが無期限で使い放題

未来館で初の公式LINEスタンプ (JSTでも初)

「フカシギの数え方」オリジナルクリアファイルの商品化

2016年1月より未来館ショップで販売開始

A4版 税込270円 (ライセンス料の一部はJST公益事業に活用される)

初版900部を完売し、8月以降追加増刷

未来館ショップのクリアファイルで首位の人気

法則 105

まとめ

アルゴリズム技術の社会的な重要性

速いか遅いかの違いだけと軽く考えてはいけない。

与えられた問題を解くために、できるだけ少ない基本演算回数で済まそうとする工夫

同じコンピュータでも、何百倍も速くなることがある。

または、もっと安価で低消費電力なコンピュータで済む。

研究成果展示: 「フカシギの数え方」

「ZDD」と呼ばれる技法を使って超高速に処理。

社会の様々な問題に幅広く応用されている。

ウェブ情報へのリンク

<http://www.lab2.kuis.kyoto-u.ac.jp/>

「京大 コンピュータアルゴリズム」でネット検索すると、研究室のホームページが出ます。

Minato Lab

Group of Computer Algorithms

トップページ

メンバー

研究

研究室の活動

学生募集

リンク集

言語切り替え (Language Switcher)

English (United States)

日本語

コンピュータアルゴリズム