

A-1. 計画研究A01班の報告

伊藤 健洋

(計画研究A01班代表, 東北大学)

学術変革領域研究 (B) 2020年度～2022年度
組合せ遷移の展開に向けた計算機科学・工学・数学によるアプローチの融合

計画研究 A01班

計算機科学アプローチによる組合せ遷移の展開： アルゴリズムの自動生成に向けて



Combinatorial Reconfiguration

研究代表者	伊藤 健洋	(東北大学)
研究分担者	大舘 陽太	(名古屋大学)
研究分担者	小林 靖明	(北海道大学)
研究分担者	和佐 州洋	(法政大学)
研究分担者	山内 由紀子	(九州大学)

江藤 宏 さん
2021年10月～2022年9月まで
特任助教(研究)としてA01班に在籍

計画研究A01班の目標

A01班の大目標

組合せ遷移に対する「**アルゴリズム的メタ定理**」の構築

A01班の2年半の目標

伊藤

大館

小林

- (i) 論理とグラフ構造パラメータを用いたメタ定理の構築
- (ii) 近接分野のアルゴリズム手法を組合せ遷移へ導入

山内

和佐

B01班：
工学



＜大目標＞
組合せ遷移に対する
実装技術の構築と産業応用
＜2年半の目標＞

- (i) 3つの技法を用いて組合せ遷移の汎用ソルバー開発
- (ii) 配電制御システムを題材とした産業応用の実践

A01班：
計算機科学



＜大目標＞
組合せ遷移に対する
アルゴリズム的メタ定理の構築
＜2年半の目標＞

- (i) 論理とグラフ構造パラメータを用いたメタ定理の構築
- (ii) 近接分野のアルゴリズム手法を組合せ遷移へ導入

C01班：
数学



＜大目標＞
組合せ遷移に対する
数学理論の構築
＜2年半の目標＞

- (i) 組合せ遷移における数学活用事例の体系的収集
- (ii) 組合せ遷移の研究に資する数理手法の開発

A01班（計算機科学）での活動

- 2, 3週間ごとに約2時間のミーティングを継続開催
- 対面打合せは, コロナ禍と相性が悪かった (予定が第〇派に直撃)
- とはいえ, 領域開始早々 (2020年11, 12月) に対面打合せができたので, 目標と課題は共有できていた
 - このときに**新しい概念である「非対称な遷移」**について, 対面で打合せられたのは幸運だった
[中畑裕さん (当時学生, 現在B01班), 岩政勇仁さん (C01班) を交えて@京大]

学術雑誌 (査読有)	25
------------	----

国際会議 (査読有)	23
------------	----

プレプリント	17
--------	----

CORE Rank A*

AAAI 2021, 2022×2本, 2023

PODS 2022

SODA 2022

CORE Rank A

ESA 2022

ISAAC 2022

MFCS 2022×2本

STACS 2022

WG 2022



2022年9月札幌にて

アルゴリズム的メタ定理

[2002 – 2012]

- 計算困難性の結果が多数 (PSPACE完全性)
- 解空間グラフが連結となるための十分条件

[2013 – 現在]

- 到達性判定に対して, アルゴリズム手法が急速に整備
2年間で20本以上の論文@arXiv (査読なし)
→ 国際会議SODA, ICALP, STACS, ISAAC, SWAT, WADS, etc.
- 遷移の性質をきちんと捉えたアルゴリズム手法

事例研究の積み重ね (個別の問題のアルゴリズム開発) から,
アルゴリズム手法を「統一された設計技法」へと昇華したい
→ どういう問題であれば、この手法は有効なのか？

アルゴリズム的メタ定理 (Courcelleの定理を例に)



独立集合問題は、木幅定数グラフで解けますか？

動的計画法で解けます



じゃあ、支配集合問題は、木幅定数グラフで解ける？

動的計画法で解けます

フィードバック集合問題は、木幅定数グラフで解ける？

動的計画法で解けます



結局、どんな問題なら、木幅定数グラフで解けるの？

MSOで記述できる問題なら大丈夫ですよ！



MSOで記述できるかどうかだけチェックできれば、アルゴリズムの中身は知らなくても良いんだね！

アルゴリズム的メタ定理

[2002 – 2012]

- 計算困難性の結果が多数 (PSPACE完全性)
- 解空間グラフが連結となるための十分条件

[2013 – 現在]

- 到達性判定に対して, アルゴリズム手法が急速に整備
2年間で20本以上の論文@arXiv (査読なし)
→ 国際会議SODA, ICALP, STACS, ISAAC, SWAT, WADS, etc.
- 遷移の性質をきちんと捉えたアルゴリズム手法

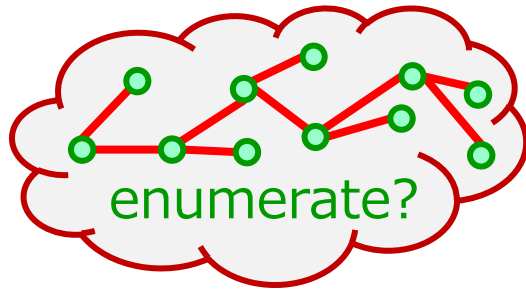
事例研究の積み重ね (個別の問題のアルゴリズム開発) から,
アルゴリズム手法を「統一された設計技法」へと昇華したい
→ どういう問題であれば、この手法は有効なのか？

組合せ遷移に対しては, 1つだけ知られている

A.E. Mouawad, N. Nishimura, V. Raman, M. Wrochna. Reconfiguration over tree decompositions. Proc. IPEC 2014, LNCS 8894, pp. 246-257 (2014)

組合せ遷移と近接分野 (1) : 列挙

列挙では、実行可能解を漏れなく重複なく全て出力したい



解空間

代表的手法：解に親子関係を導入
→ 解空間を連結 & 効率よく探索する

解空間の連結性は保証されるが、最短遷移は？

Katsuhisa Yamanaka, Takashi Horiyama, Kunihiro Wasa (A01班).

Optimal reconfiguration of optimal ladder lotteries.

Theoretical Computer Science 859, pp. 57-69 (2021)



「組合せ遷移」セミナーで、山中克久先生にお話し頂く

→ C01班（数学）の視点を導入して. . .

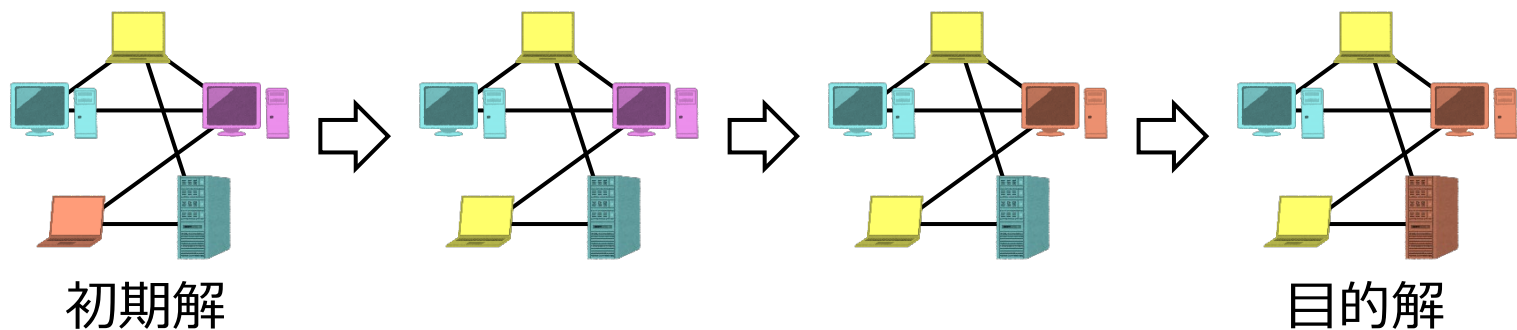
Yuta Nozaki (C01班), Kunihiro Wasa (A01班), Katsuhisa Yamanaka

Reconfiguration and Enumeration of Optimal Cyclic Ladder Lotteries

電子情報通信学会コンピュータシミュレーション研究会 2022年12月

組合せ遷移と近接分野 (2) : 分散計算

分散計算では“大局的な変化”を伴う遷移は現実的ではない
 →各プロセスが**局所変数を変更**することで遷移系列を生成



主に2つのタイプ

1. 同期式分散システムでの 遷移の**並列化**

- 点彩色遷移 [Bonamy et al. DISC 2018]
- 極大独立集合遷移

[Censor-Hillel&Rabie ICALP 2019]

• 全域木遷移

Siddharth Gupta, Manish Kumar, Shreyas Pai.
Distributed reconfiguration of spanning trees.

SSS 2022: 346-351

2. 非同期式分散システムでの **局所性**の下での遷移

- 全域木遷移

Yukiko Yamauchi, Naoyuki Kamiyama,
Yota Otachi. Distributed reconfiguration
of spanning trees. SSS 2021: 516-520

波及効果 (?)

A-2. アルゴリズム基盤： 組合せ遷移のアルゴリズム的メタ定理

大舘 陽太

(名古屋大学)

 Tatsuya Gima, Takehiro Ito, Yasuaki Kobayashi, Yota Otachi.

Algorithmic meta-theorems for combinatorial reconfiguration revisited.

The 30th European Symposium on Algorithms (ESA 2022).

September 5–9, 2022, in Potsdam, Germany (hybrid).

*儀間達也さん（名大 D1）が現地で発表．一部のスライドを流用．

A01 班の目標（の一つ）：組合せ遷移アルゴリズムの自動生成

- アルゴリズム開発の第一歩は「事例研究」
 - ▶ 個々の問題に対し，その問題に特化したアルゴリズムを構築
- 事例研究の積み上げにより，異なる問題に対する共通する手法の発見
 - ▶ 事例研究 → 統一されたアルゴリズム設計技法の研究 → **アルゴリズム的メタ定理**の発見
 - ▶ 問題の「記述」からアルゴリズムを自動生成する仕組み
- **組合せ遷移に対するアルゴリズム的メタ定理**を作りたい
 - ▶ 事例研究はこの 10 年でかなり進んでいる
 - ▶ 既に，あるアルゴリズム的メタ定理も示されているが [Mouawad et al. IPEC '14]，ある不満が…… → 今回の研究

組合せ遷移に対するアルゴリズム的メタ定理

知られていたこと

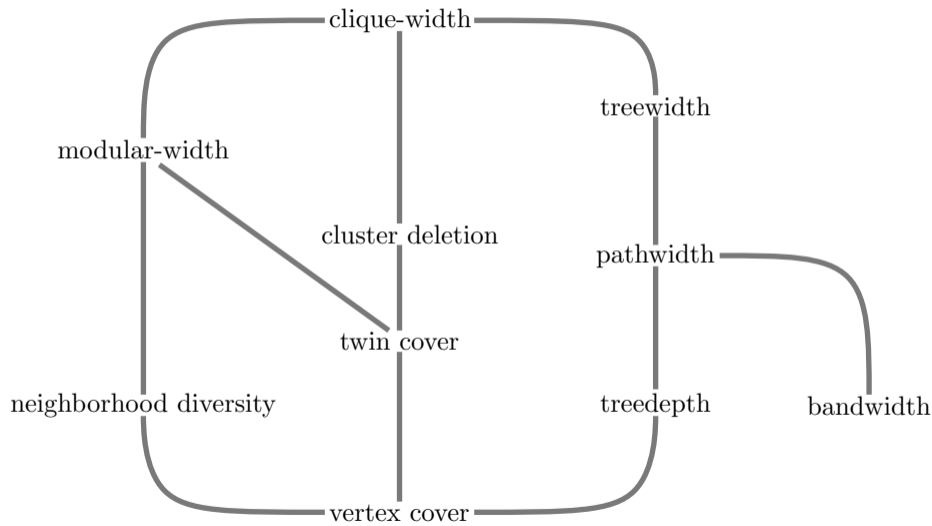
- 実行可能解がグラフ上の単項二階論理 (MSO) で書けるとき、
遷移列長とクリーク幅をパラメータとして FPT¹ [Mouawad et al. 2014]
 - ▶ MSO で様々な性質が記述できる (例: 独立集合, 支配集合, 連結性, 極大・極小性)
 - ▶ クリーク幅はとても一般的なパラメータ (少なくとも MSO の文脈では)
- 遷移列長を制限しないと, バンド幅定数のグラフに対しても
独立集合遷移が PSPACE 完全 [Wrochna 2018]
 - ▶ 「バンド幅定数」はかなりきつい制限 (なので, その後の発展がなかった?)

今回の研究 (扱うのはグラフの頂点部分集合の遷移のみ)

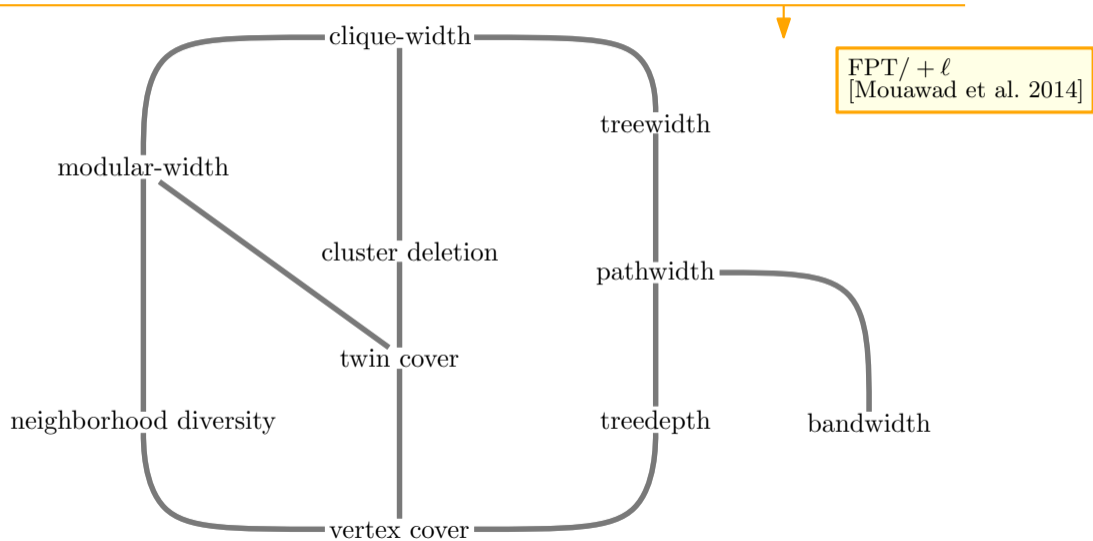
- 遷移列長を制限しない場合に対するアルゴリズム的メタ定理を作りたい
→ できた! (対象範囲には限りがあるが, 結構ギリギリまでいけている)
- 各結果に対するある意味での下界も与えた

¹FPT: 入力サイズ n , パラメータ k のとき, ある計算可能関数 f があり $f(k) \cdot n^{O(1)}$ 時間で解けること.

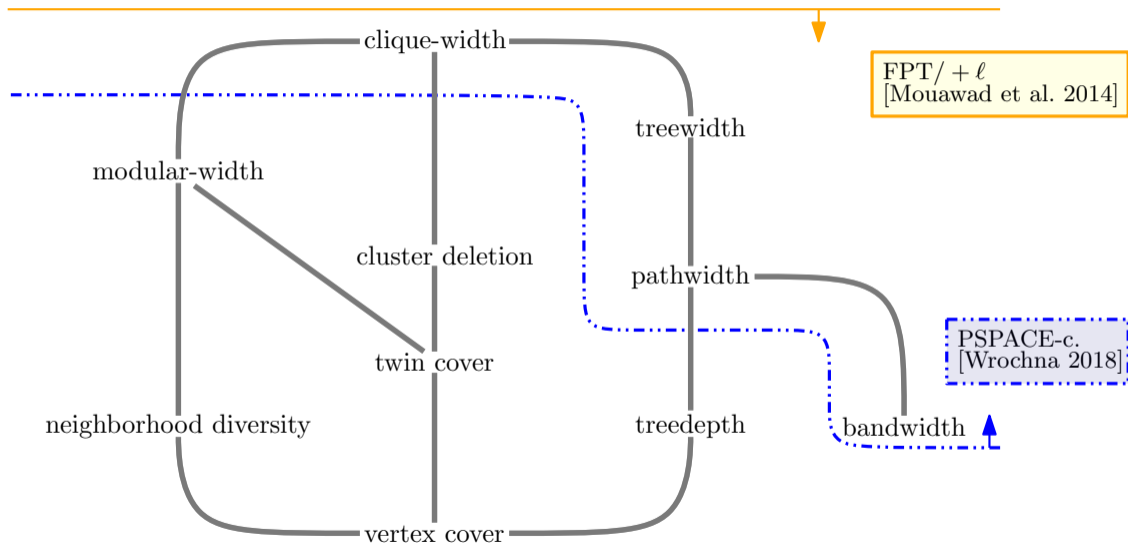
全体像: メタ組合せ遷移問題の計算量 (MSO₁, TJ)



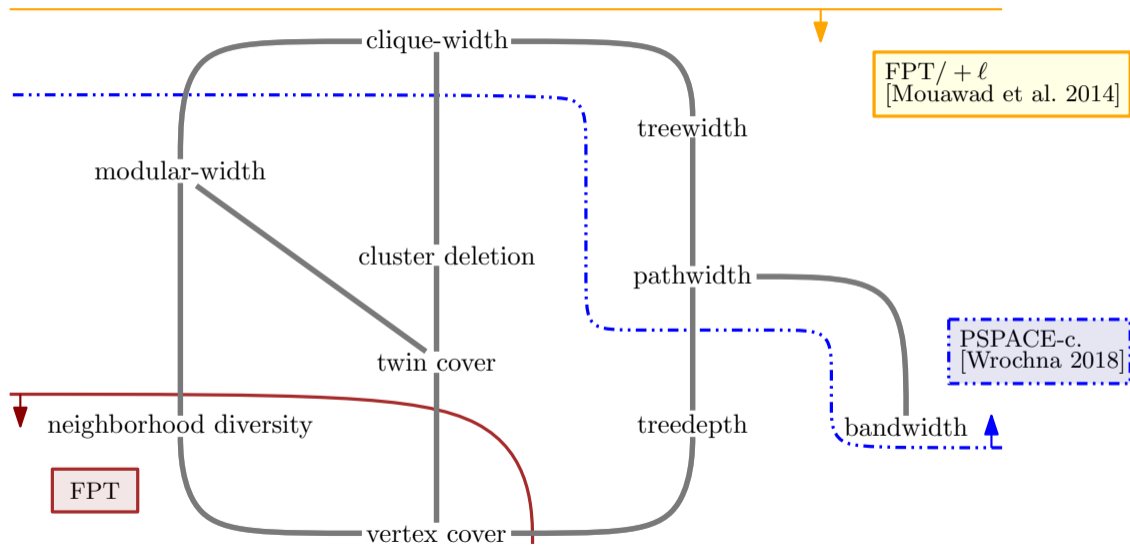
全体像: メタ組合せ遷移問題の計算量 (MSO₁, TJ)



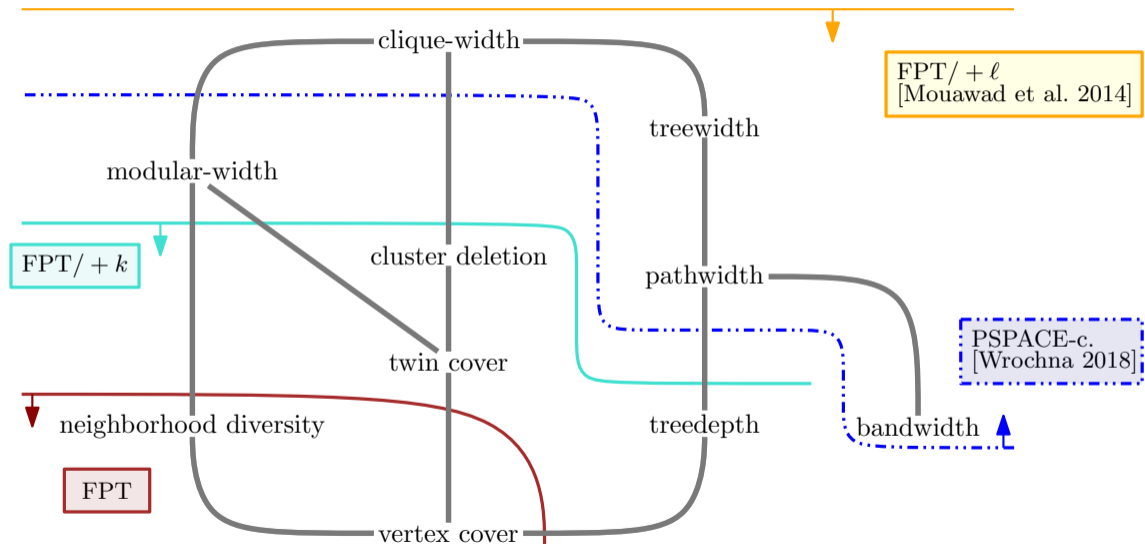
全体像: メタ組合せ遷移問題の計算量 (MSO₁, TJ)



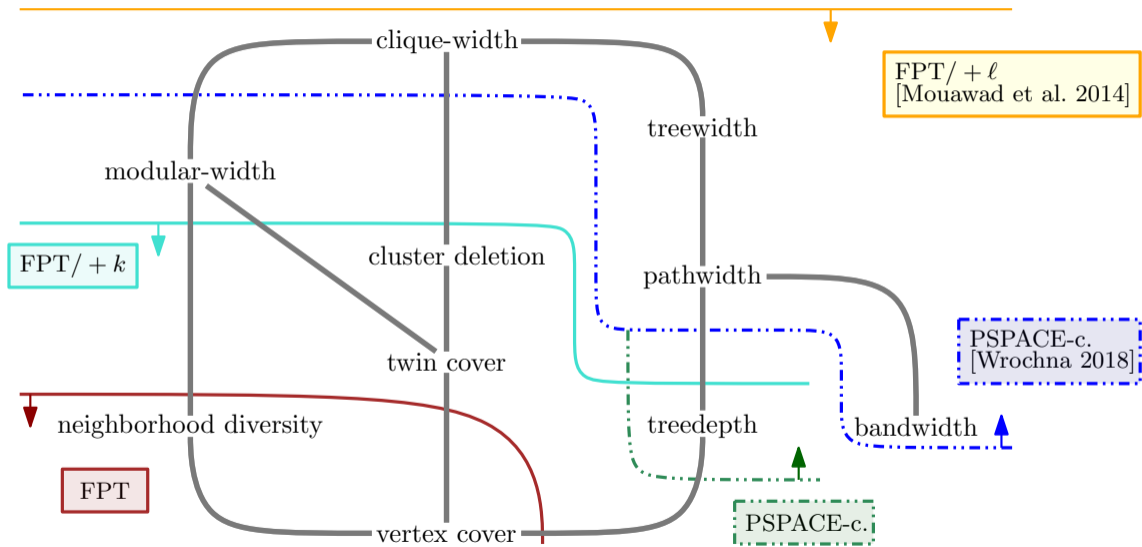
全体像: メタ組合せ遷移問題の計算量 (MSO₁, TJ)



全体像: メタ組合せ遷移問題の計算量 (MSO₁, TJ)



全体像: メタ組合せ遷移問題の計算量 (MSO₁, TJ)



グラフ上の単項二階論理 (Monadic Second-Order logic, MSO)

MSO 式の例: 「 X は独立集合である」を表す式 $\varphi(X)$

$$\varphi(X) = \neg(\exists u \exists v (u \in X \wedge v \in X \wedge E(u, v)))$$

グラフ上の MSO (MSO_1) で使えるもの

- 頂点・頂点集合を表す変数: $v_1, v_2, \dots, V_1, V_2, \dots$
- 変数に対する量化: $\exists v, \forall u, \exists W, \forall X$
- 否定, 論理和, 論理積: $\neg\varphi, \varphi \vee \psi, \varphi \wedge \psi$
- 等号・帰属関係: $u = v, v_i \in V_j$
- 2 頂点が隣接しているかを表す述語: $E(u, v)$

(辺や辺集合も扱える拡張は, MSO_2 とよばれる)

Courcelle の定理 (遷移でない問題に対するアルゴリズム的メタ定理)

(Courcelle はこの成果で 2022 EATCS-IPEC Nerode Prize を受賞)

MSO MODEL CHECKING

入力: グラフ G , MSO 式 φ .

質問: G は φ が表す性質を満たすか?

- φ が MSO_2 : FPT / G の木幅 + $|\varphi|$ [Courcelle 1990]
- φ が MSO_1 : FPT / G のクリーク幅 + $|\varphi|$ [Courcelle, Makowski, Rotics 2000]

気持ち (木幅版)

- 幅が小さい木分解を, ラベル付き木だと思う (ラベル数は幅の関数で抑えられる)
- 木分解が与えられたときの MSO 式チェックは, 対応するラベル付き木に対するある木オートマトンの実行で表せる. (ほぼ, ボトムアップ DP)
- 木分解が必要だが, 前処理として計算しておける [Bodlaender 1996]

Mouawad らのアルゴリズム的メタ定理 (遷移長がパラメータの一部)

遷移問題で、実行可能解がグラフ上の単項二階論理 (MSO) で書けるとき、
遷移列長とクリーク幅をパラメータとして FPT [Mouawad et al. 2014]

遷移列長 ℓ がパラメータなので、条件を満たす遷移列の存在が一つの MSO 式で書けてしまう

$$\Phi = \exists X_0 \exists X_1 \dots \exists X_\ell \left(X_0 = S \wedge X_\ell = S' \wedge \bigwedge \varphi(X_i) \wedge \bigwedge \text{one-step}(X_i, X_{i+1}) \right)$$

ここで、 $\text{one-step}(X_i, X_{i+1})$ は遷移の一步を表すルール. (« ℓ 以下» ならば «変化なし» も許す)

この仕組はとても柔軟

- 「MSO (MSO₁) とクリーク幅」を, 「MSO₂ と木幅」や「FO と nowhere dense グラフ」などに置き換えられる (ロジックとそれが FPT となるグラフクラスの組)
- 遷移の一步 $\text{one-step}(X_i, X_{i+1})$ も柔軟に書ける. (i に依存してもよい)
 - ▶ TJ/TS は簡単. TAR もサイズ制限を満たす追加・削除の列をすべて試せばよい

考える問題

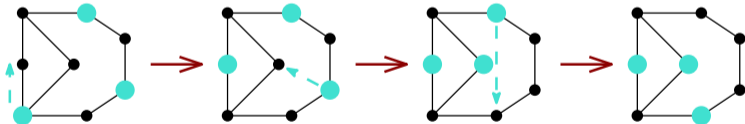
グラフに対する MSO で表せる頂点集合の遷移. 遷移ルールは Token Jumping.

Problem MSO-RECONFIGURATION (MSO-R)

Input グラフ $G = (V, E)$, MSO 式 $\varphi(X)$, 頂点集合 $S, S' \subseteq V$ ($G \models \varphi(S)$, $G \models \varphi(S')$).

Question 頂点集合の列 $S_0 (= S), S_1, \dots, S_\ell (= S')$ で以下を満たすものはあるか?

- $G \models \varphi(S_i)$ for all i ,
- $|S_i \setminus S_{i+1}| = |S_{i+1} \setminus S_i| = 1$ for all i . (Token Jumping)



代表的な遷移ルール

- Token Jumping (TJ): 任意の頂点を任意の他の頂点と交換できる
- Token Sliding (TS): 任意の頂点を任意の隣接頂点と交換できる

MSO-RECONFIGURATION として表せる遷移問題の例

多くの問題が表せる。それぞれの問題の様々な亜種も表現できる。

- INDEPENDENT SET RECONFIGURATION

- ▶ VERTEX COVER RECONFIGURATION, CLIQUE RECONFIGURATION
- ▶ 亜種: maximal, distance d , induced matching, r -regular, H -free, ...

- DOMINATING SET RECONFIGURATION

- ▶ 亜種: minimal, connected, total, efficient, distance d , (σ, ρ) -domination の一部, ...

- FEEDBACK VERTEX SET RECONFIGURATION

- ▶ INDUCED FOREST RECONFIGURATION, INDUCED TREE RECONFIGURATION
- ▶ 亜種: minimal, connected, ...

- CHORDAL INDUCED SUBGRAPH RECONFIGURATION

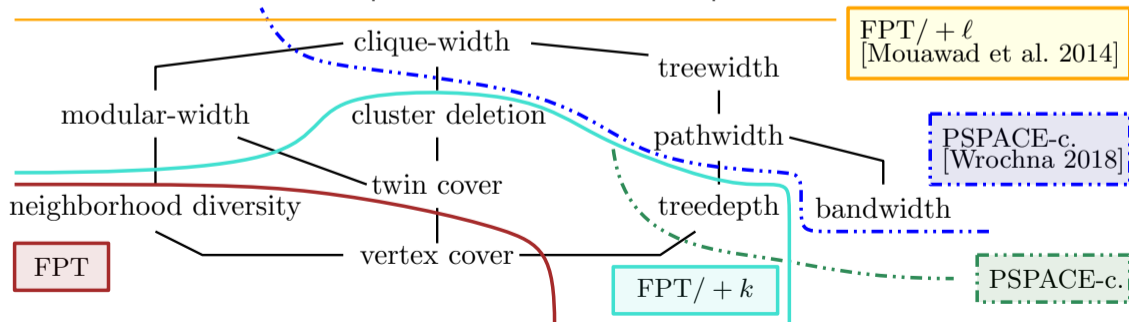
- ▶ 亜種: interval, proper interval, split, perfect, ...

- PLANAR INDUCED SUBGRAPH RECONFIGURATION

- ▶ \mathcal{H} -MINOR FREE INDUCED SUBGRAPH RECONFIGURATION

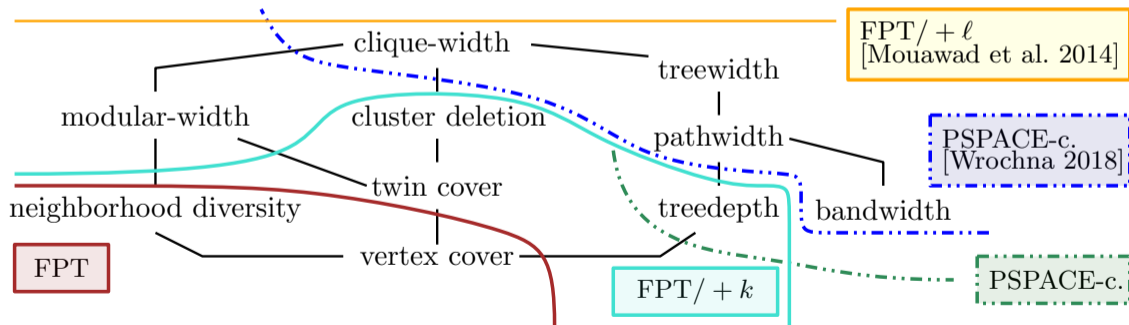
できたこと

- 1 MSO-R is FPT / neighborhood diversity + $|\varphi|$ (TS はできていない)
 - ▶ MSO_2 への拡張はできない ($E \neq NE$ の仮定の下で)
 - ▶ vertex cover なら MSO_2 までいける
- 2 MSO-R is FPT / treedepth + $k + |\varphi|$ (MSO_2 に拡張可. TS でも大丈夫)
 - ▶ (副産物) MSO-R is FPT / cluster deletion + $k + |\varphi|$
- 3 MSO-R is PSPACE-complete even on forests of depth 3.



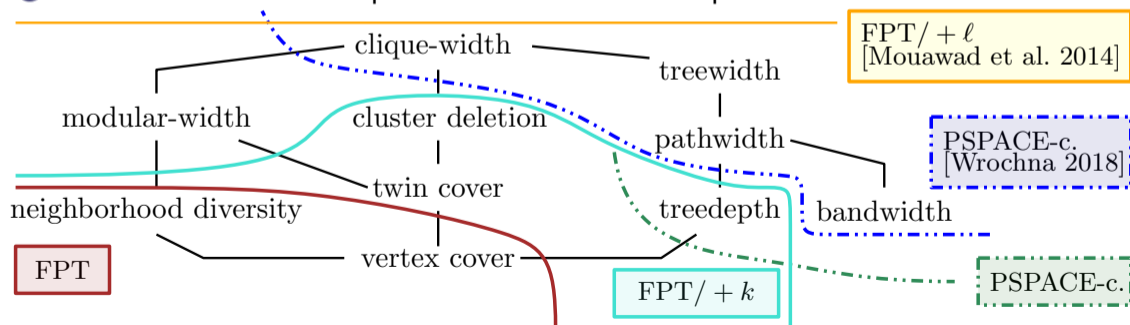
各パラメータについて

- **clique-width**: treewidth の一般化. ちょっと一般的すぎる
- **vertex cover**: treewidth の特殊可. ちょっと特殊すぎる
- **neighborhood diversity**, **twin cover**: vc の一般化. 密なグラフでも小さくなり得る
- **modular-width**: nd, tc の一般化. clique-width に近づくが, いろんな問題が解ける



まとめ

- ① MSO-R is FPT / neighborhood diversity + $|\varphi|$ (TS はできていない)
 - ▶ MSO_2 への拡張はできない ($E \neq NE$ の仮定の下で)
 - ▶ vertex cover なら MSO_2 までいける
- ② MSO-R is FPT / treedepth + $k + |\varphi|$ (MSO_2 に拡張可. TS でも大丈夫)
 - ▶ (副産物) MSO-R is FPT / cluster deletion + $k + |\varphi|$
- ③ MSO-R is PSPACE-complete even on forests of depth 3.



A-3. アルゴリズム基盤の隣接分野への展開

A-3.1. 組合せ遷移と列挙

和佐 州洋

(法政大学)

列挙問題

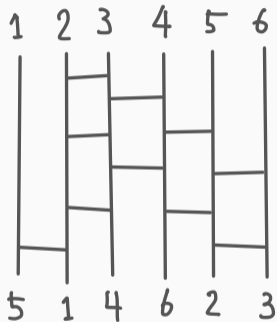
与えられた入力に対し，条件を満たす解を漏れなく重複なく出力する問題．

- ▶ 入力グラフに含まれる極大なクリークを列挙せよ．
- ▶ n 頂点の二分木を列挙せよ．
- ▶ SAT の充足解を列挙せよ．
- ▶ ...

今日の話

組合せ遷移と列挙が同じ遷移ルールのもとで考察されている阿弥陀籤の話

阿弥陀籤 (あるいは, Pseudoline arrangement)



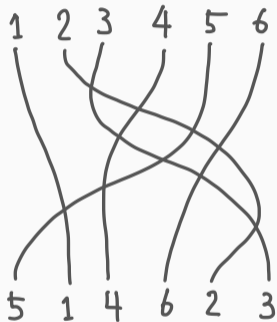
定義 (置換 π に対する阿弥陀籤)

阿弥陀籤 L は, n 本の縦線 (l_1, \dots, l_n) と 2 本の縦線の交差で表現される. このとき, L が置換

$\pi = (\pi_1, \dots, \pi_n)$ の阿弥陀籤であるとは,

- (1) 上端は恒等置換に対応
- (2) 各交差は, $[n]$ 中の二つの要素を交換する
- (3) 下端は π に対応

阿弥陀籤 (あるいは, Pseudoline arrangement)



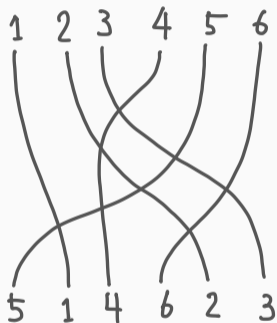
定義 (置換 π に対する阿弥陀籤)

阿弥陀籤 L は, n 本の縦線 (l_1, \dots, l_n) と 2 本の縦線の交差で表現される. このとき, L が置換

$\pi = (\pi_1, \dots, \pi_n)$ の阿弥陀籤であるとは,

- (1) 上端は恒等置換に対応
- (2) 各交差は, $[n]$ 中の二つの要素を交換する
- (3) 下端は π に対応

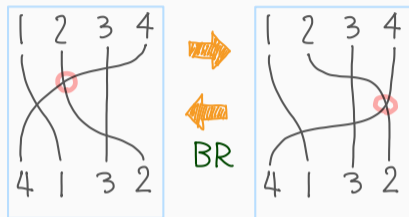
最適な阿弥陀籤



定義 (置換 π に対する最適な阿弥陀籤)

ある阿弥陀籤 L が**最適**であるとは、 L が π に対する阿弥陀籤であり、かつ、より少ない交差の阿弥陀籤が存在しない時をいう。

阿弥陀籤に対する操作：Braid relation



Braid relation (BR)：3つの縦線 i, j, k に対して、 $(j, k), (i, k), (i, j)$ と連続して交差が現れる時、 $(i, j), (i, k), (j, k)$ に交差の順序を変更する操作。

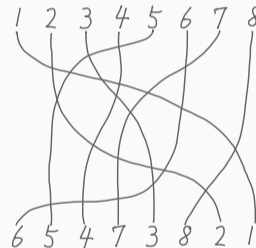
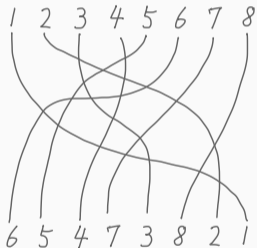
補題

任意の2つ最適な阿弥陀籤 L, L' に対して、 L に BR を繰り返し適切に適用することで L' を得ることができる。

阿弥陀籤における問い

組合せ遷移

BRのもと，最適な阿弥陀籤 L から別の最適な阿弥陀籤 L' を最小手数で得るアルゴリズムは？



列挙

全ての最適な阿弥陀籤を得るアルゴリズムは？

数え上げ

逆順列に対する最適な阿弥陀籤の数 B_n の上下界 [1]

$$\blacktriangleright 2^{0.2053n^2} \leq B_n \leq 2^{0.6571n^2}$$

Braid relation を用いた列挙（置換は固定）

最適な阿弥陀籤

- ▶ ひとつあたり定数時間で列挙可能 [5]

k 本からなる阿弥陀籤

- ▶ ひとつあたり定数時間で列挙可能 [4]

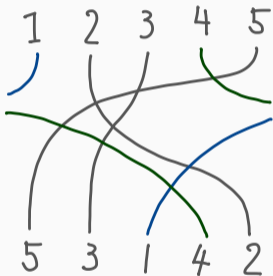
Braid relation のもとでの遷移

最適な阿弥陀籤の最適遷移列

- ▶ 線形時間アルゴリズムが存在 [3]

円筒阿弥陀籤への拡張

円筒阿弥陀籤への拡張 (野崎先生, 山中先生との共同研究)



円筒阿弥陀籤

阿弥陀籤を円筒の側面に描画したもの。

- ▶ 1番目と n 番目が隣り合っているような阿弥陀籤。
- ▶ 通常の阿弥陀籤のような“端”はない。

最適な円筒阿弥陀籤

交差の本数を減らすことのできない円筒阿弥陀籤。

ほぼ最適な円筒阿弥陀籤

各縦線において、交差がたかだか一つしかないような円筒阿弥陀籤。

どのような交差が阿弥陀籤上に現れるか？

阿弥陀籤 ある置換に対する最適な阿弥陀籤は全て、交差の集合が同じ.

円筒阿弥陀籤 ある置換に対する最適な阿弥陀籤でありうる交差の集合は様々.

最適な円筒阿弥陀籤の列挙 = \bigsqcup 交差集合を固定した列挙
ありうる交差の集合

変位ベクトル (Displacement vector)

縦線の左右方向の移動量. $\sum_{i \in [n]} x_i = 0$ かつ $i + x_i \equiv \pi_i \pmod{n}$ を満たすベクトル $x = (x_1, x_2, \dots, x_n) \in \mathbb{Z}^n$ を π の変位ベクトルと呼ぶ.

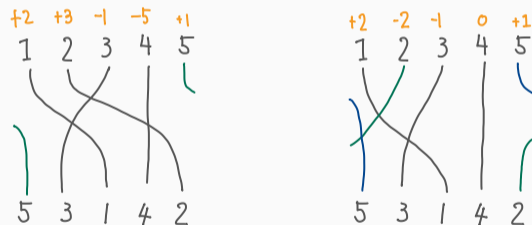


図 1: 一つの置換を表す 2 つ円筒阿弥陀籤は, 変位ベクトルが異なる場合がある

変位ベクトルに関する観察

ここで重要な観察は次：

置換 π に対して一つ変位ベクトルを決めると、 π に対応する円筒阿弥陀を実現する交差の集合が定まる。

さらに、Jerrum [2] によって次のことが示されている：

置換 π に対する変位ベクトルは max-min contraction によって、列挙できる。

max-min contraction

直感的には、最も移動量が多い左回りの縦線と右回りの縦線を選び、移動する向きを逆にする操作。

定理

任意の置換に対し、最適な円筒阿弥陀籤を一つあたり多項式時間で列挙できる。

これを示すために、次の2つを示した。

定理

置換と変位ベクトルを固定したとき、ほぼ最適な円筒阿弥陀籤は一つあたり多項式時間で列挙可能。

定理

任意の置換に対し、最適な変位ベクトルを一つあたり定数時間で列挙可能。

- ▶ さらに、最適な円筒阿弥陀籤の交差個数が最大となる置換の同定。

- ▶ 阿弥陀籤に対して，組合せ遷移と列挙という観点から研究した．
- ▶ 近年の列挙アルゴリズムは遷移的な視点で列挙していることが多い．
 - これまでは，割と表現しにくかったことが遷移という言葉でうまく説明できるようになった．
 - 遷移ルールを巧妙に考えだすことが列挙の典型テクニク．
- ▶ 一方，遷移的な問題に対し，列挙的な観点での研究はまだまだよくわかっていない．
 - 特に，列挙が難しい問題と組合せ遷移との関係，遷移グラフの直径と列挙の関係など...

- [1] Stefan Felsner and Pavel Valtr.
Coding and counting arrangements of pseudolines.
Discrete & Computational Geometry, 46:405–416, 2011.
- [2] Mark R. Jerrum.
The complexity of finding minimum-length generator sequence.
Theoretical Computer Science, 36:265–289, 1985.
- [3] Katsuhisa Yamanaka, Takashi Horiyama, and Kunihiro Wasa.
Optimal reconfiguration of optimal ladder lotteries.
Theoretical Computer Science, 859:57–69, 2021.
[doi:10.1016/j.tcs.2021.01.009](https://doi.org/10.1016/j.tcs.2021.01.009).

- [4] Katsuhisa Yamanaka and Shin-Ichi Nakano.
Efficient enumeration of all ladder lotteries with k bars.
IEICE Transactions, 97-A(6):1163–1170, 2014.
- [5] Katsuhisa Yamanaka, Shin-ichi Nakano, Y. Matsui, R. Uehara, and K. Nakada.
Efficient enumeration of all ladder lotteries and its application.
Theoretical Computer Science, 411:1714–1722, 2010.

A-3. アルゴリズム基盤の隣接分野への展開

A-3.2. 組合せ遷移と分散計算

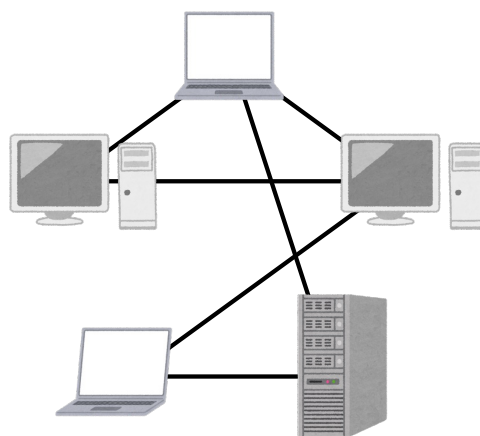
山内 由紀子

(九州大学)

組合せ遷移と分散計算

2

- 分散システム
 - ▣ 多数の計算機(プロセス)が通信しながら協調動作
 - ▣ **局所性**
 - 各計算機はシステム全体の状況がわからない
 - ▣ **非同期性・並列性**
 - 計算機がバラバラのタイミングで通信, 計算



通信ネットワーク

組合せ遷移と分散計算

3

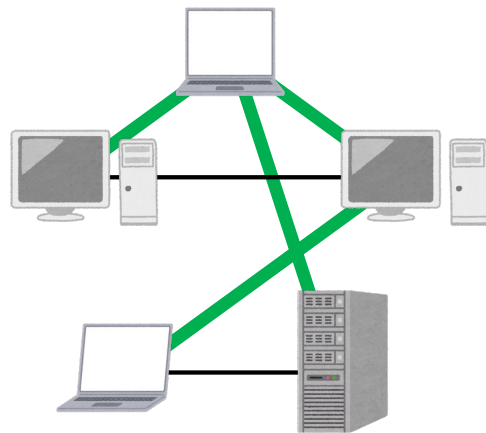
遷移問題を分散システム上で考えると、
制約が増えるのか、緩和されるのか？

□ 局所性

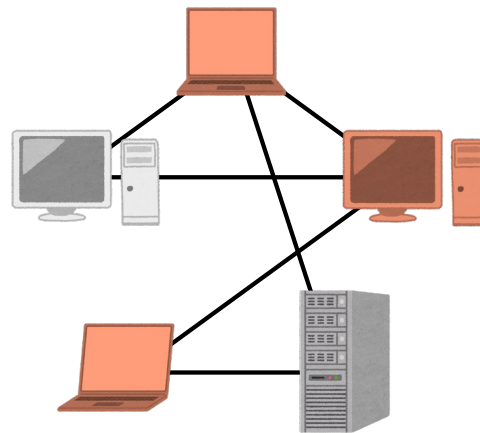
- 各計算機はシステム全体の状況がわからない

□ 非同期性・並列性

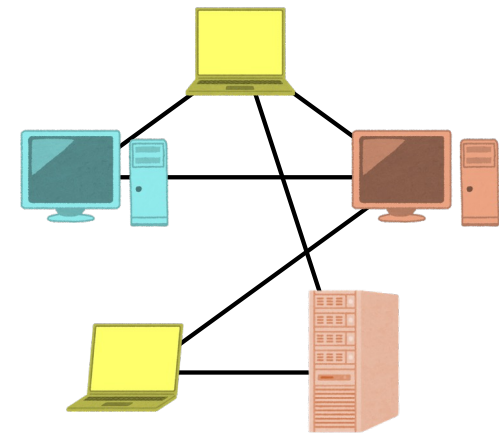
- 計算機がバラバラのタイミングで通信，計算



全域木



頂点被覆

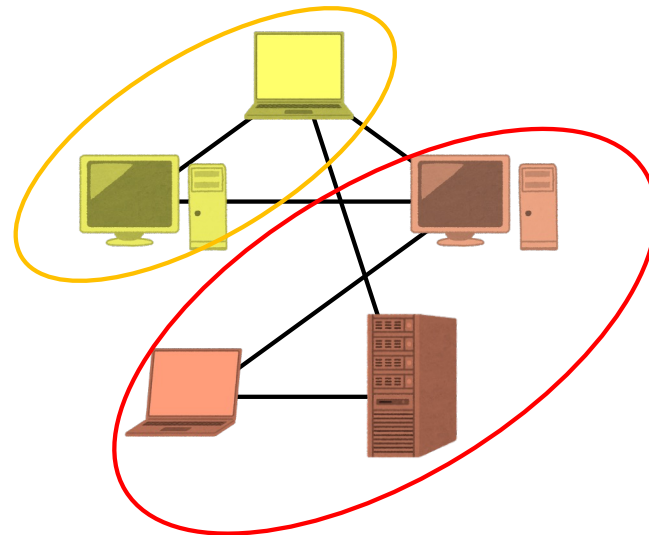


頂点彩色

分散システム

4

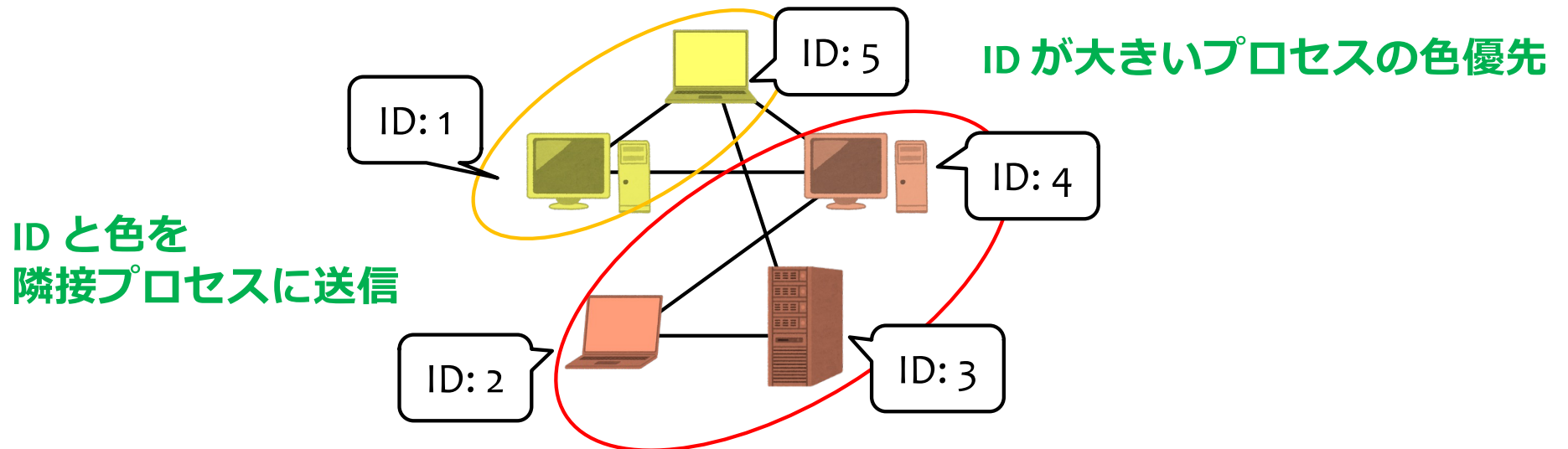
- 各プロセスは与えられた分散アルゴリズムをもとに
 - ▣ **受信**：隣接プロセスからメッセージを受信
 - ▣ **内部計算**：局所変数の更新，送信メッセージの決定
 - ▣ **送信**：隣接プロセスへメッセージを送信
- 例．頂点彩色
 - ▣ 各プロセスが自身の“色”を管理
 - ▣ 隣接プロセスに同じ色の頂点があれば色を変更



分散システム

5

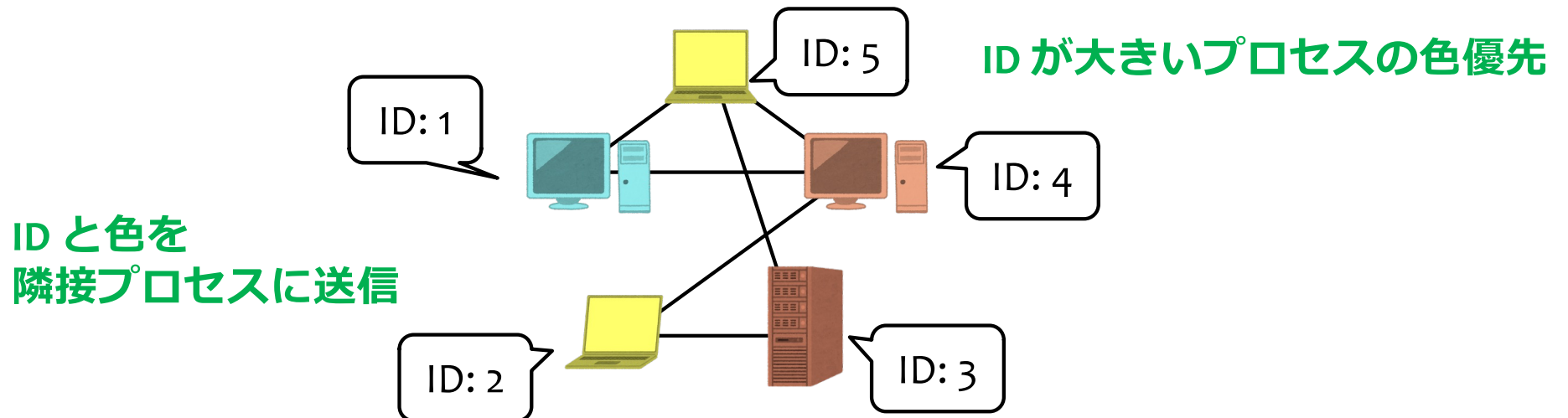
- 各プロセスは与えられた分散アルゴリズムをもとに
 - ▣ **受信**：隣接プロセスからメッセージを受信
 - ▣ **内部計算**：局所変数の更新，送信メッセージの決定
 - ▣ **送信**：隣接プロセスへメッセージを送信
- 例．頂点彩色
 - ▣ 各プロセスが自身の“色”を管理
 - ▣ 隣接プロセスに同じ色の頂点があれば色を変更



分散システム

6

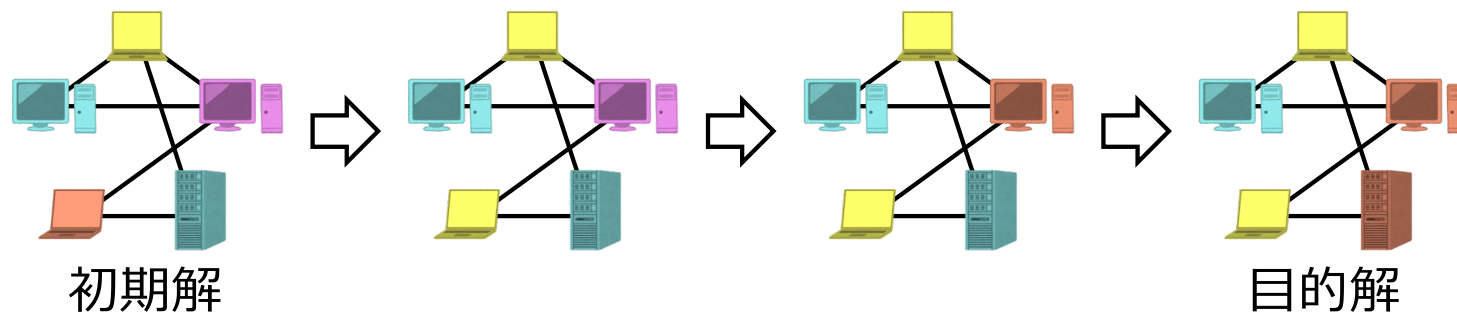
- 各プロセスは与えられた分散アルゴリズムをもとに
 - ▣ **受信**：隣接プロセスからメッセージを受信
 - ▣ **内部計算**：局所変数の更新，送信メッセージの決定
 - ▣ **送信**：隣接プロセスへメッセージを送信
- 例．頂点彩色
 - ▣ 各プロセスが自身の“色”を管理
 - ▣ 隣接プロセスに同じ色の頂点があれば色を変更



分散遷移問題

7

- 各プロセスが局所変数を変更することで遷移系列を生成



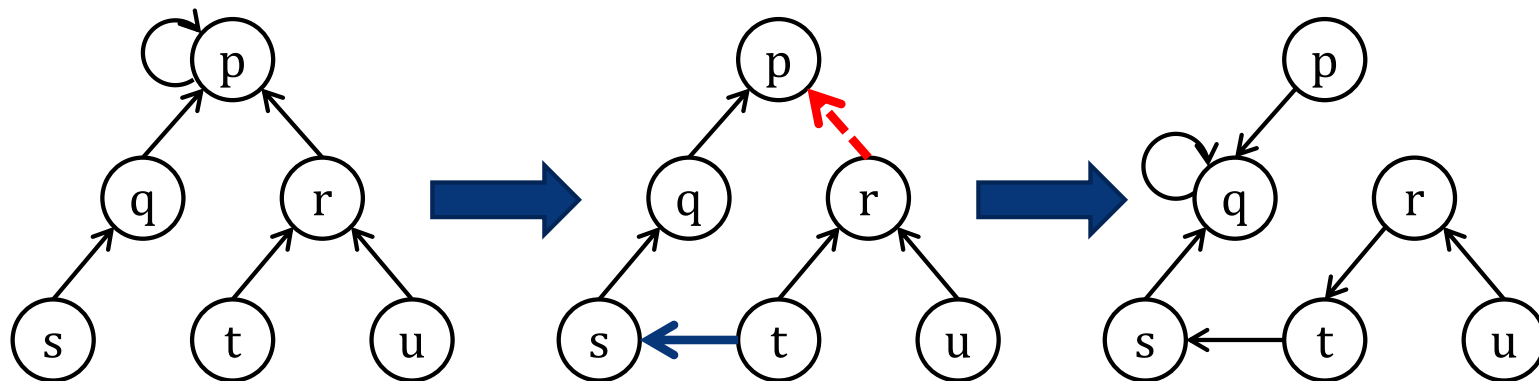
- 主に2つのタイプの既存研究
 - **同期式**分散システムでの遷移の**並列化**
 - プロセスが同期して通信, 内部計算
 - 頂点彩色 [Bonamy et al., 2018], 独立集合 [Censor-Hillel and Rabie, 2019], 全域木 [Gupta et al., 2022]
 - **非同期式**分散システムでの**局所性**の下での遷移
 - プロセスの動作タイミングがバラバラ
 - 全域木 [Y. et al., SSS 2021]

全域木の分散遷移問題 (非同期式)

Y. Yamauchi, N. Kamiyama, and Y. Otachi, Distributed reconfiguration of spanning trees. SSS 2021, pp. 516--520.

8

- 各プロセスは初期木と目的木の接続辺を与えられる
- 遷移のルール
 - ▣ 1つの頂点に接続する辺のメンバーシップの変更のみ可
 - ▣ どの時点でも全域木(連結かつサイクルなし)
- 全域木の逐次遷移はマトロイドの基交換公理より明らか



全域木の分散遷移問題 (非同期式)

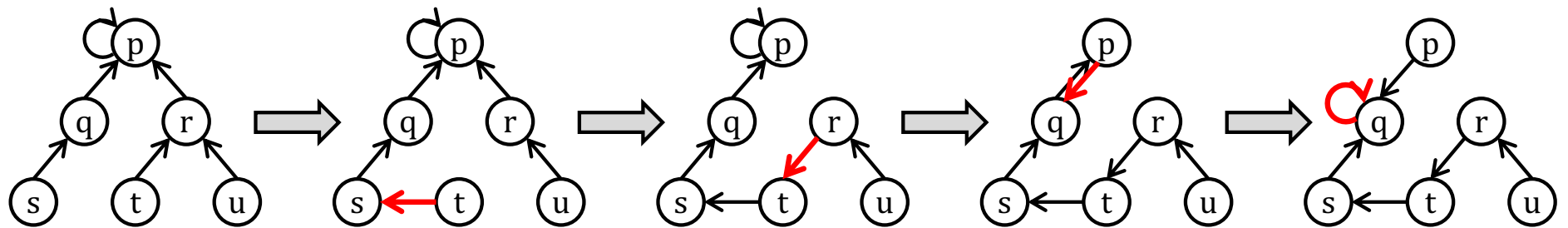
Y. Yamauchi, N. Kamiyama, and Y. Otachi, Distributed reconfiguration of spanning trees. SSS 2021, pp. 516--520.

9

- 各プロセスは初期木と目的木の接続辺を与えられる
- 遷移のルール
 - ▣ 1つの頂点に接続する辺のメンバーシップの変更のみ可
 - ▣ どの時点でも全域木(連結かつサイクルなし)

本研究の成果

局所性の制約の下でも、木の向きづけを行えば遷移可能

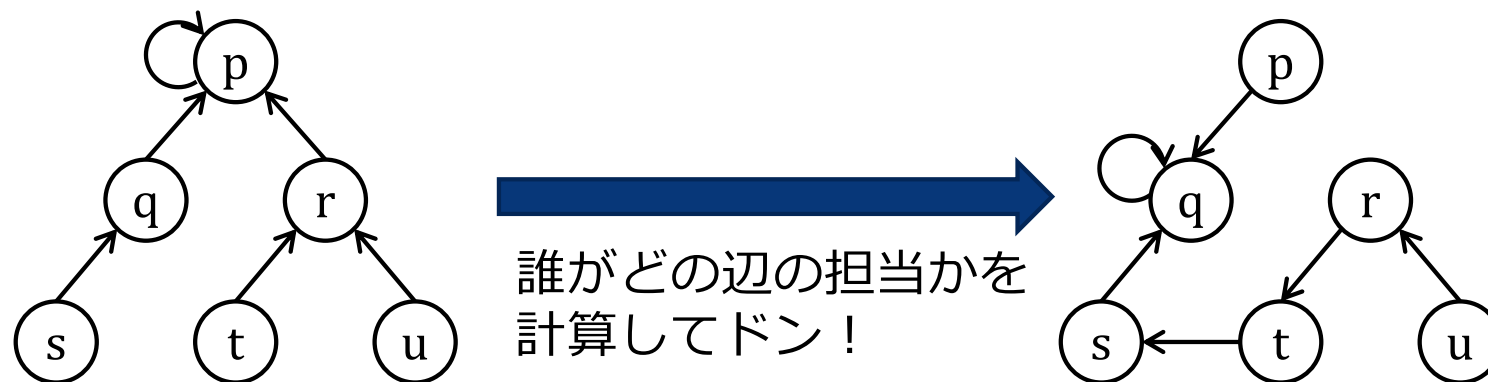


全域木の分散遷移問題 (同期式)

S. Gupta, M. Kumar, and S. Pai, Distributed reconfiguration of spanning trees, SSS 2022.

10

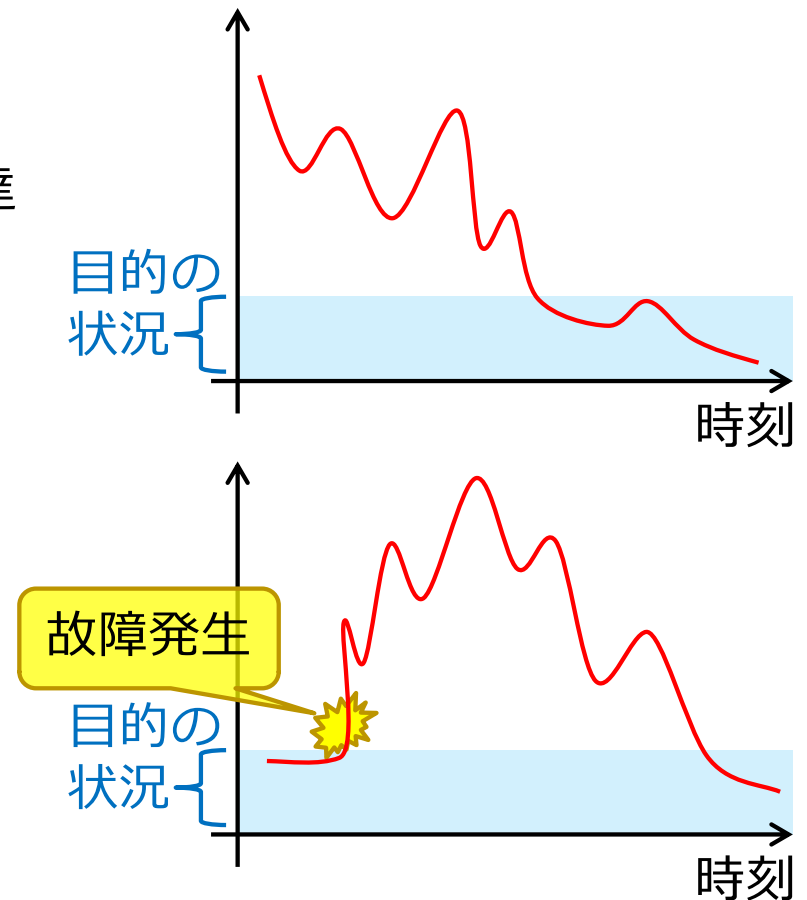
- 各プロセス p は初期木と目的木の接続辺を与えられる
 - ▣ 親ポインターの変更スケジュールを計算
- 遷移のルール
 - ▣ 1つの頂点に接続する辺で同時に高々 k 個の辺の追加削除
 - ▣ どの時点でも全域木
- ワンショット(長さ1)のスケジュール
 - ▣ $k = 1$ でスケジュールの計算に $\Omega(n)$ ラウンド
 - ▣ $k = 2$ でスケジュールの計算に $\Omega(\log n)$ ラウンド



頑健な分散システムの設計パラダイム

11

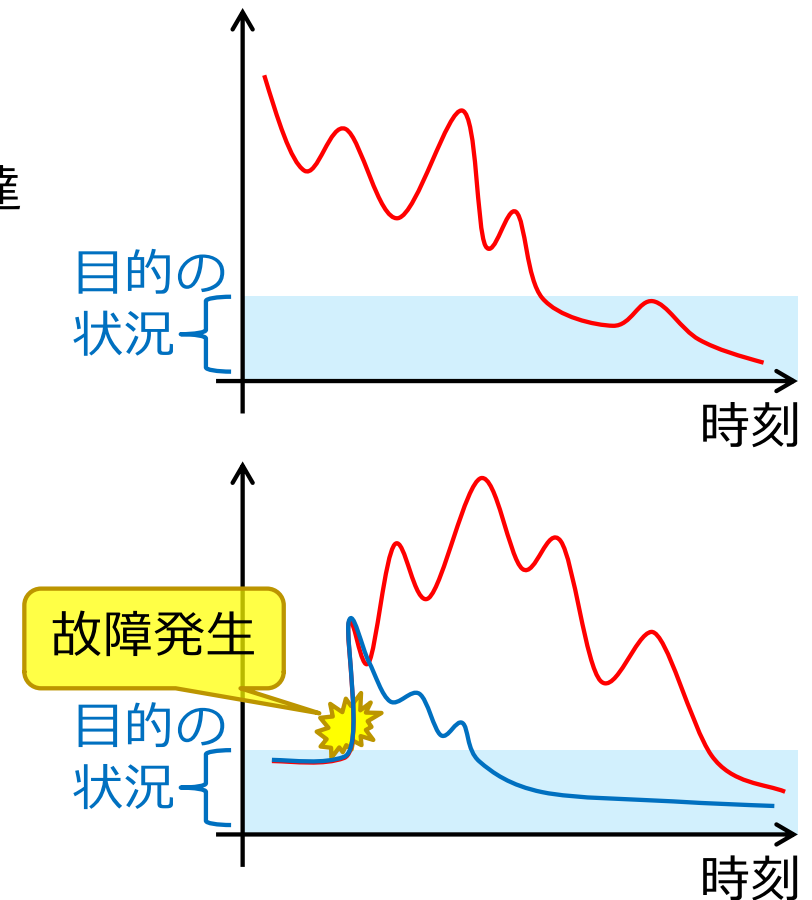
- **自己安定アルゴリズム** [Dijkstra, 1974]
 - ▣ 任意のシステム状況から開始しても、いずれ条件を満たすシステム状況に到達
 - ▣ 初期化不要のネットワークプロトコル
- **故障封じ込めアルゴリズム** [Ghosh et al., 1996]
 - ▣ 少数の故障(局所変数の書換え)が起これども、故障の影響を拡げずに復旧
- **分散遷移**
 - ▣ 故障の後の復旧途中にも安全性を保証(フェールセーフの一種)



頑健な分散システムの設計パラダイム

12

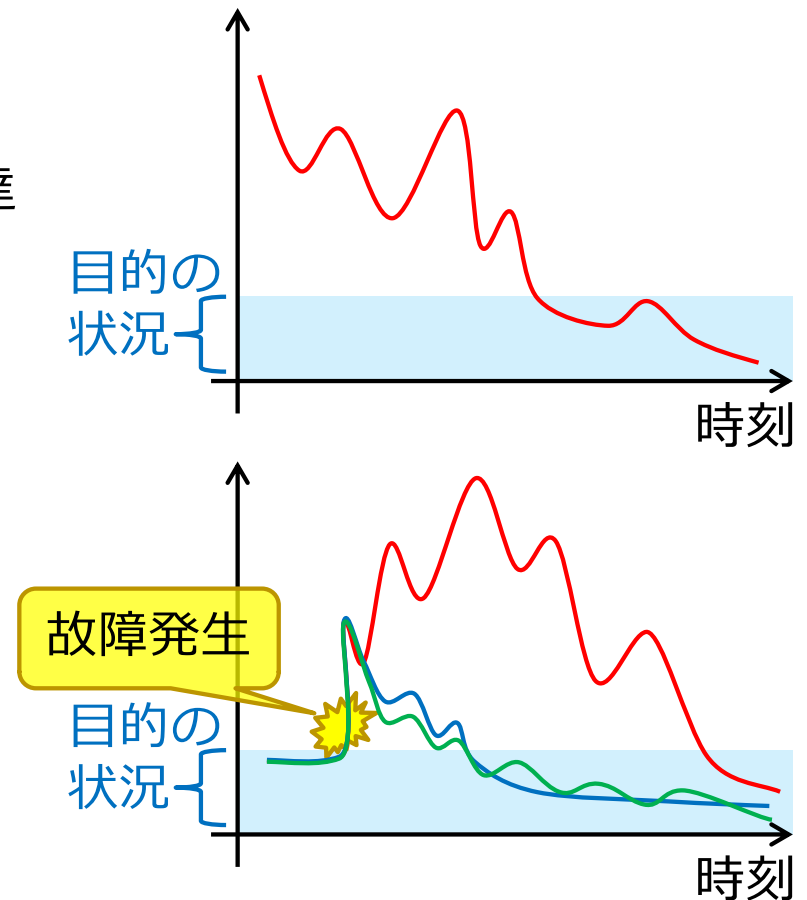
- **自己安定アルゴリズム** [Dijkstra, 1974]
 - ▣ 任意のシステム状況から開始しても、いずれ条件を満たすシステム状況に到達
 - ▣ 初期化不要のネットワークプロトコル
- **故障封じ込めアルゴリズム** [Ghosh et al., 1996]
 - ▣ 少数の故障(局所変数の書換え)が起これども、故障の影響を拡げずに復旧
- **分散遷移**
 - ▣ 故障の後の復旧途中にも安全性を保証 (フェールセーフの一種)



頑健な分散システムの設計パラダイム

13

- **自己安定アルゴリズム** [Dijkstra, 1974]
 - ▣ 任意のシステム状況から開始しても、いずれ条件を満たすシステム状況に到達
 - ▣ 初期化不要のネットワークプロトコル
- **故障封じ込めアルゴリズム** [Ghosh et al., 1996]
 - ▣ 少数の故障(局所変数の書換え)が起こっても、故障の影響を拡げずに復旧
- **分散遷移**
 - ▣ 故障の後の復旧途中にも安全性を保証 (フェールセーフの一種)



まとめと今後の展望

14

- まとめ
 - ▣ 分散システムでの遷移問題
 - 並列化による逐次遷移問題の一般化
 - 局所性を考慮した逐次遷移問題への制約付加
 - ▣ 同期式分散システムでの頂点彩色, MIS, 全域木の分散遷移を紹介
 - ▣ 本研究では非同期式分散システムでの全域木の分散遷移を提案

- 今後の展望
 - ▣ 分散システム特有の離散構造の遷移問題
 - 合意形成, リーダー選挙, コータリー, モバイルロボット
 - ▣ 頑健性保証のための新たな設計パラダイム
 - ▣ 分散最適化
 - 許容解の系列を経て最適解へ