

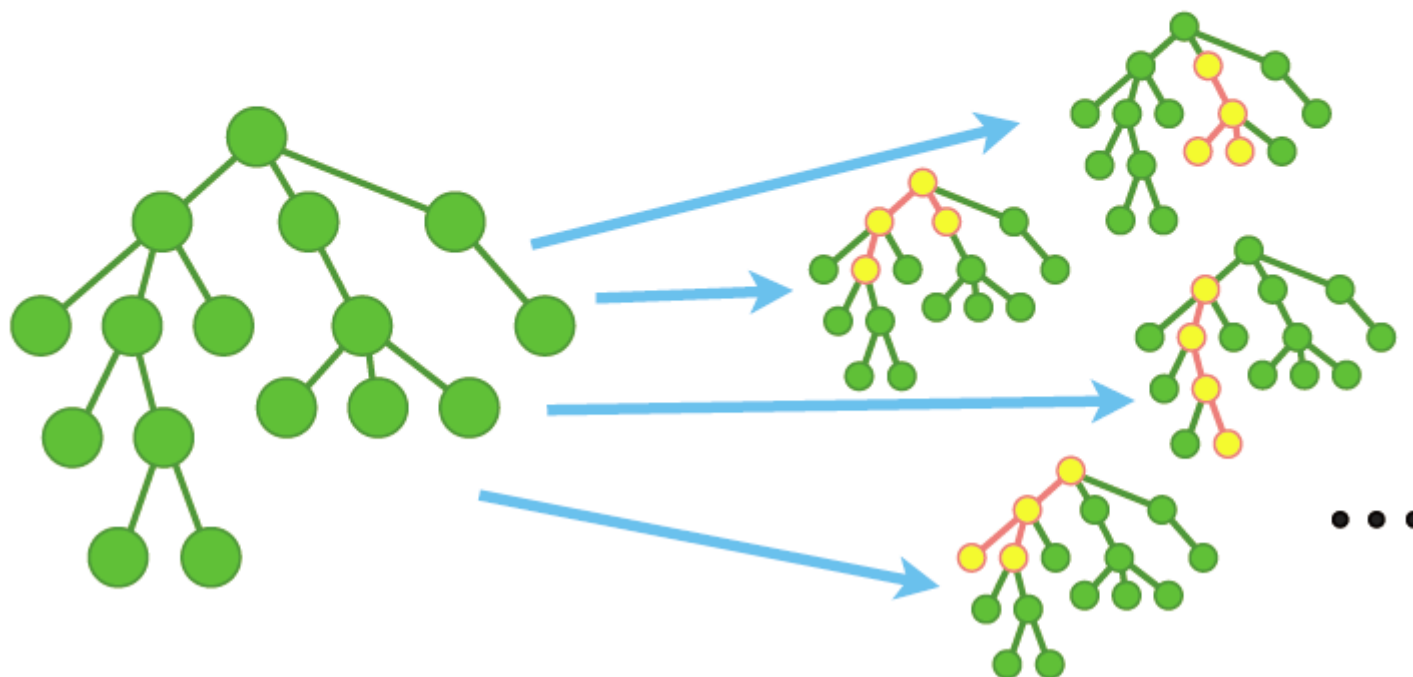
木に含まれる限定サイズ部分木の列挙



○和佐 州洋¹, 宇野 毅明², 有村 博紀¹ (1北大, 2NII)

k-部分木列挙問題

入力: 根付き木 $T=(V, E)$, 正整数 k
出力: T に含まれるすべての k -部分木





研究目的

- 効率よい部分木列挙アルゴリズムを開発する.

主結果

- **定数遅延時間アルゴリズムを開発した.**

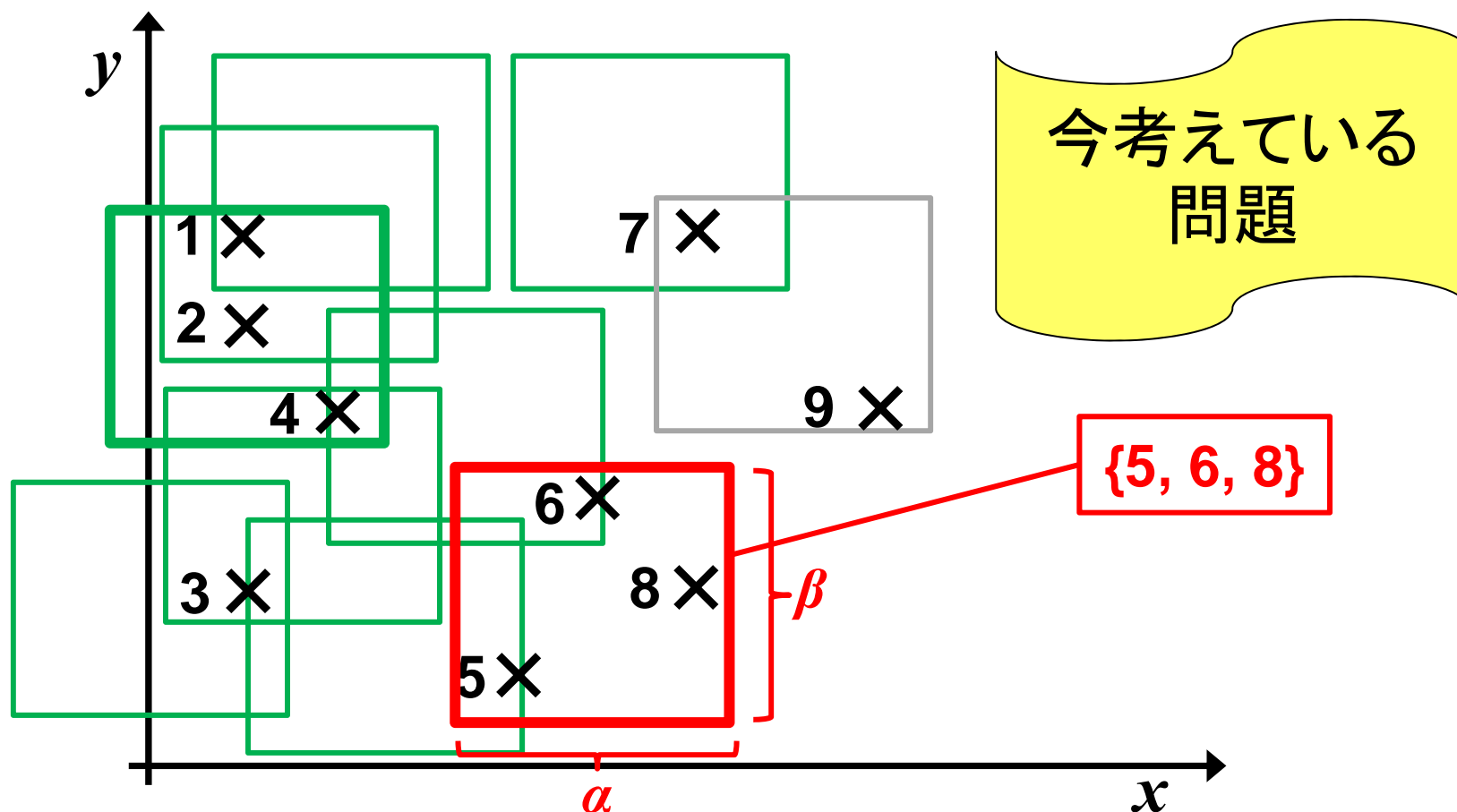
	一般的アルゴリズム [FGR2011]	提案 アルゴリズム
入力データ	一般のグラフ	木
列挙時間(1個あたり)	$O(k)$ 時間	$O(1)$時間
空間計算量	$O(m)$ 領域	$O(m)$ 領域

m : グラフ中の辺数
[FGR2011]: Ferreira, Grossi, Rizzi, *Output-sensitive Listing of Bounded-size Trees in Undirected Graphs*, Proc. ESA2011, 2011.

To appear as: Kunihiro Wasa, Takeaki Uno, and Hiroki Arimura, "Constant Time Enumeration of Bounded-Size Subtrees in Trees and Its Application," Proc. 18th Annual International Computing and Combinatorics Conference (COCOON'12), August 2012.

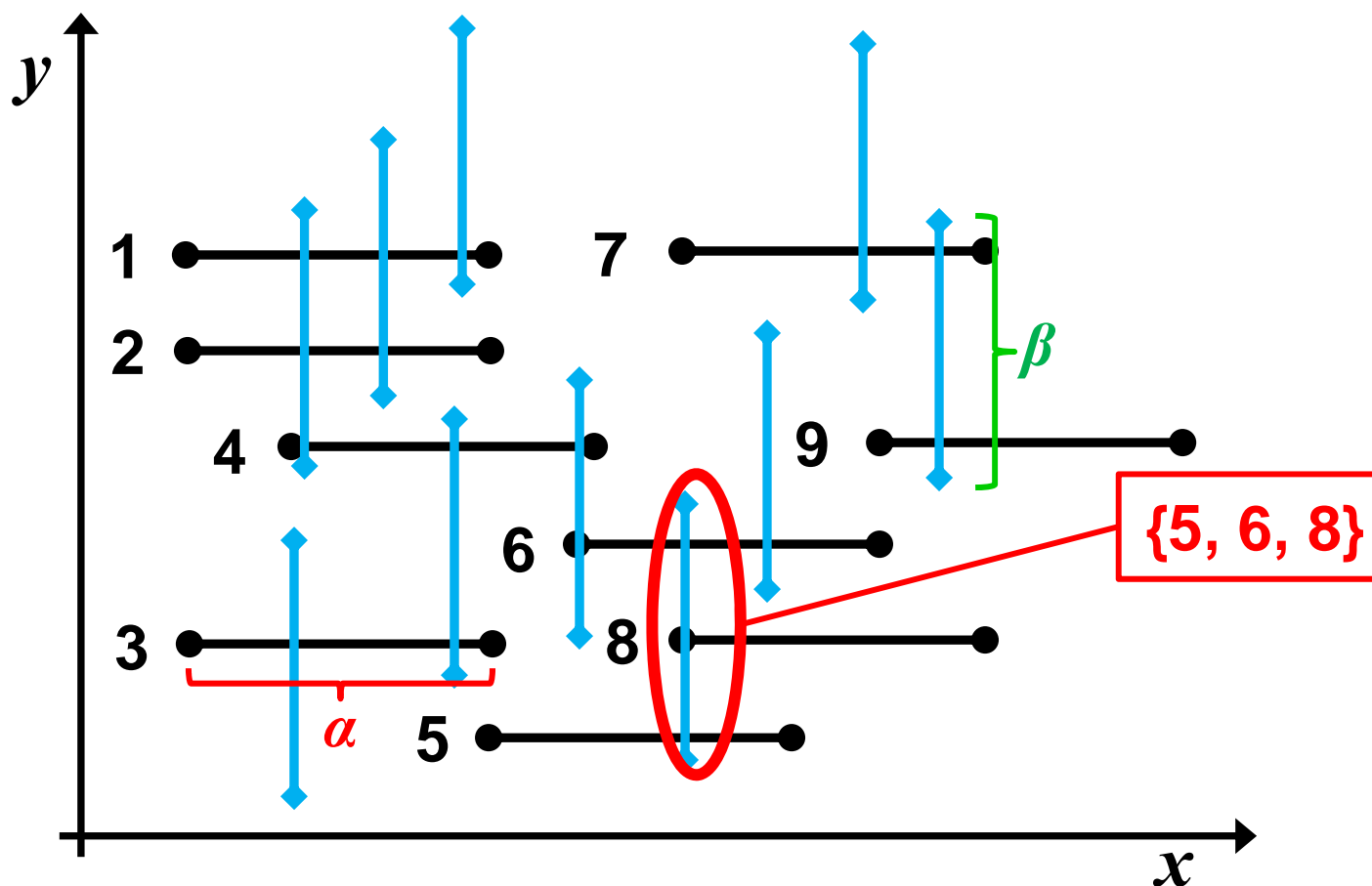


「限定サイズ長方形による点集合列挙」
平面上のN個の点の集合が与えられたとき、指定されたサイズの長方形で囲まれる異なる点集合を列挙せよ





等価な問題（「限定長さ線分の交点列挙」）：
与えられた同じ長さの水平線分の集合に対して，指定された長さの垂直線分と交わる水平線分の番号の集合を列挙せよ





■ 解の個数 s の上限と下限は？

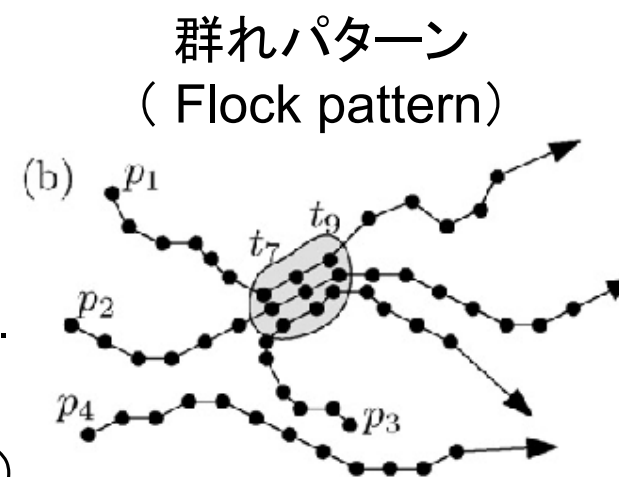
- サイズ固定の長方形の問題: $s = O(N^2)$

■ 計算時間

- 自明なアルゴリズム: $O(N^2)$ 時間と領域.
- 出力線形かつ低メモリのアルゴリズムを作れるか？

■ 発展

- 極大な部分集合だけを列挙する.
- 物体の「サイズ」を任意にして列挙する.
- K 個以上の点を含む部分集合を列挙する.
- 一般の K 次元空間への拡張
- データマイニングへの応用(群れパターン)
 - 狭い領域に、多くの個体があつまったところを見つける問題



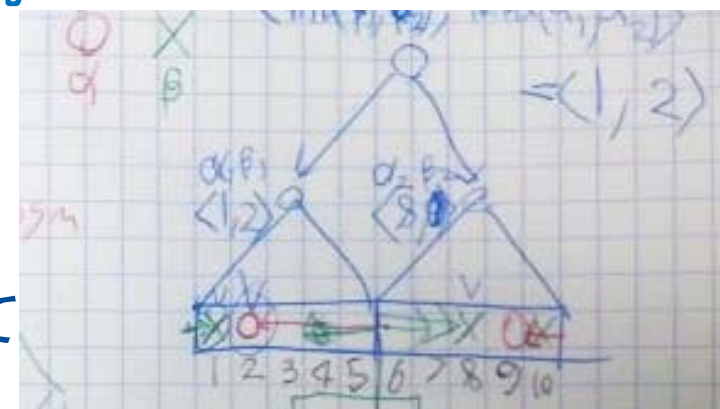
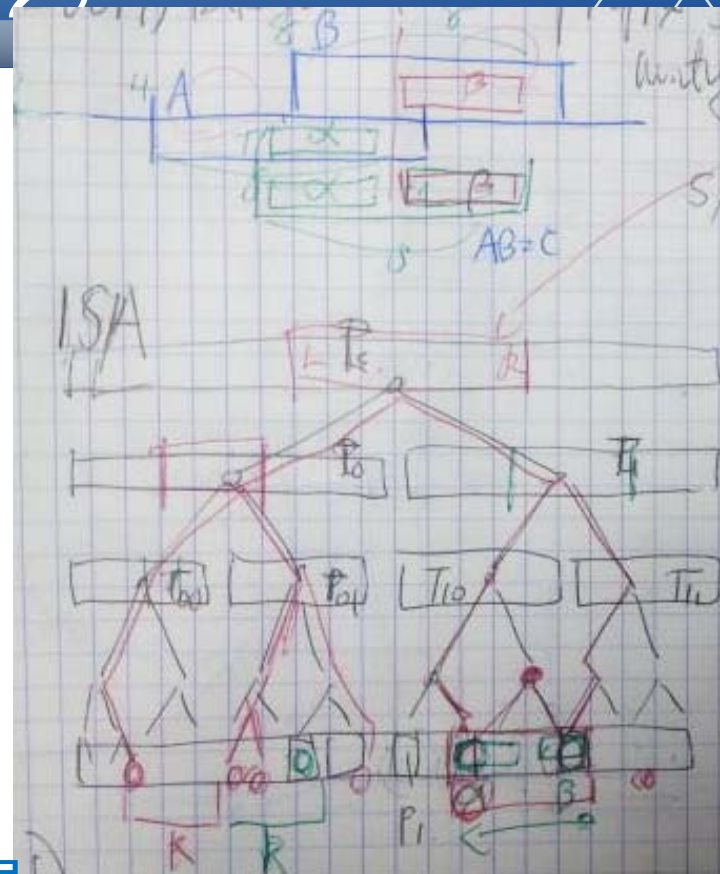
～ 津田先生と伝住君と以前に議論

Prefix-Suffix文字列マッチング

問題: prefix α とsuffix β のテキスト中の距離 k 以下の出現をすべて見つける.

1. Suffix Array (またはBWT)を用いて, α と β のランク区間をもとめる.
2. LSA上で, α と β のランク区間を同時に探索する. 可能な枝はすべてたどる.
3. 探索が幅 $2k$ のブロックに入ったら展開をやめて, キョリが k 以内の α と β の出現位置を見つける.

- メモリと時間のトレードオフ
- 平均時間解析?
- 別案として, α と β のポジションを求めてマージする方法(系列BDD).





～ 簡潔データ構造による圧縮符号化テキスト上の接尾辞木構築の高速化 (packed string)

- 符号数 K で総ビット長 N の圧縮符号化テキストの接尾辞木を作る.
- 現在は $O(N)$ 時間と $O(K + \|\Sigma\|)$ 語の領域 (Uemura, Arimura, CPM'11)
- 目標: $O((N \log w)/w + K)$ 時間と $O(K + \|\Sigma\|)$ 語の領域にする.
 - 符号木を簡潔トライで模倣: 一語あたり $O(\log w)$ 時間.
 - 接尾辞木の巡回は, パトリシア木を模倣する: 分岐一つは定数時間. 枝文字列の検査は, 枝一つあたり $O(\text{len}/w)$ 時間.
 - 接尾辞リンクの巡回の解析.

Binary Huffman code automaton

