

ICDE2015勉強会

Research 16: Stream

担当：杉浦 健人（名大）

PIE: Approximate Interleaving Event Matching over Sequences

Zheng Li, Tingjian Ge (University of Massachusetts)

目的

複合イベント処理における**イベントの同時生起**及び**イベント生起時刻の曖昧さ**を考慮したい

- ▶ 現実のイベントには同時多発的な生起や信頼できないタイムスタンプなどが存在

提案

並列パターン及び**イベントの転置**を許したイベント照合

- ▶ PIE: a**P**roximate **I**nterleaving **E**vent matching

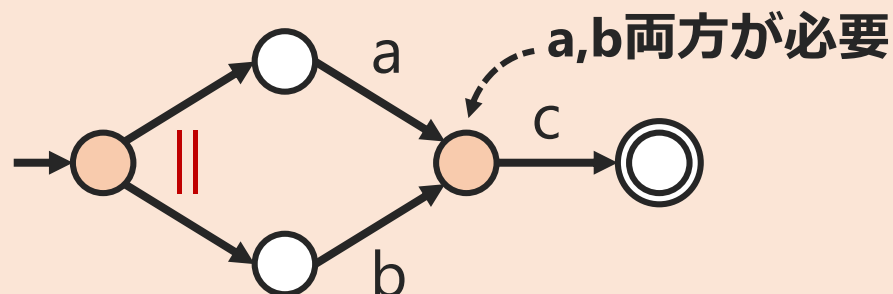
PIE: approximate interleaving matching

Parallel finite automaton による並列パターン“||”の受理

例 : Pattern = (a || b) c

受理 abc, bac, など

破棄 ac, bc, など



閾値 τ に基づくイベントの転置及び見逃しの許可

例 : Pattern = (a || b) c, $\tau = 1$

受理 abc, bac, **acb**, **bca**, など イベントの転置 = $1 \leq \tau$

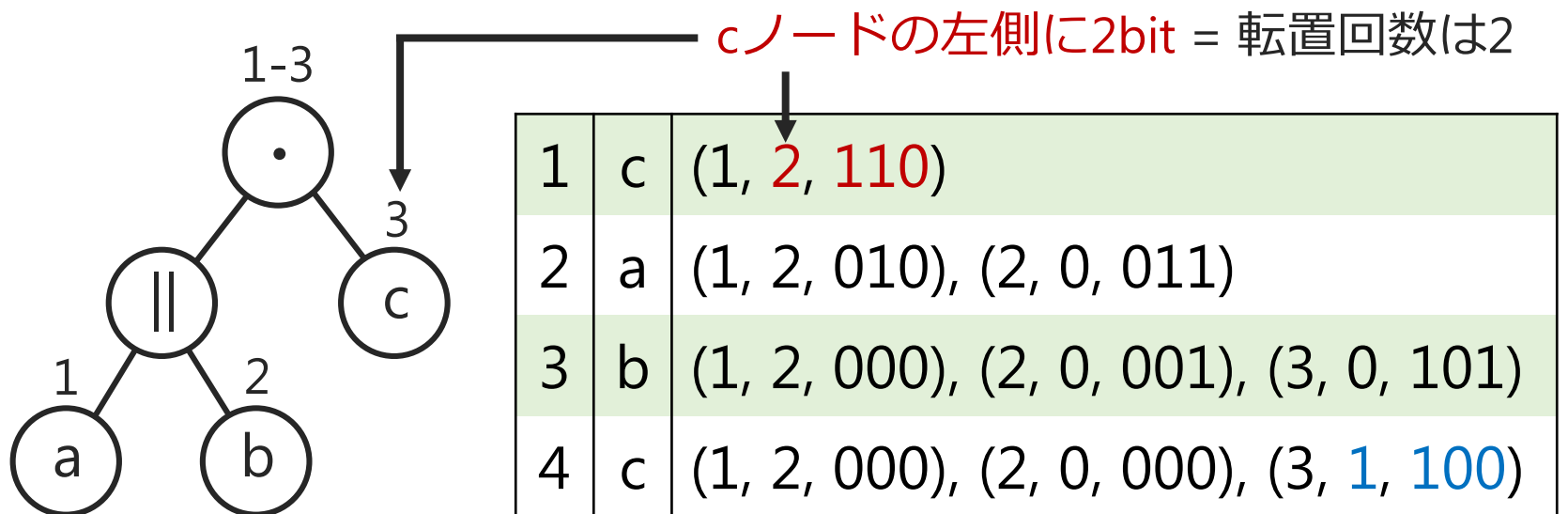
破棄 ac, bc, など イベントの転置 + 見逃し = $2 > \tau$

解法の概要

照合の進行度を**ビットマップで管理** (0=match, 1=unmatch)

▶ パターンをビットマップ木で表し転置回数の検出に利用

例 : Pattern = (a || b) c, input = c a b c



※ (開始時刻, 転置回数, ビットマップ)

Window-Chained Longest Common Subsequence: Common Event Matching in Sequences

Chunyao Song, Tingjian Ge (University of Massachusetts)

背景・目的

Longest Common Subsequence (LCS) には時間の制約が無い

例：

s_1	a	b	d	c	e	f	g
s_2	g	f	e	d	a	b	c

➔ **LCS = a b c**

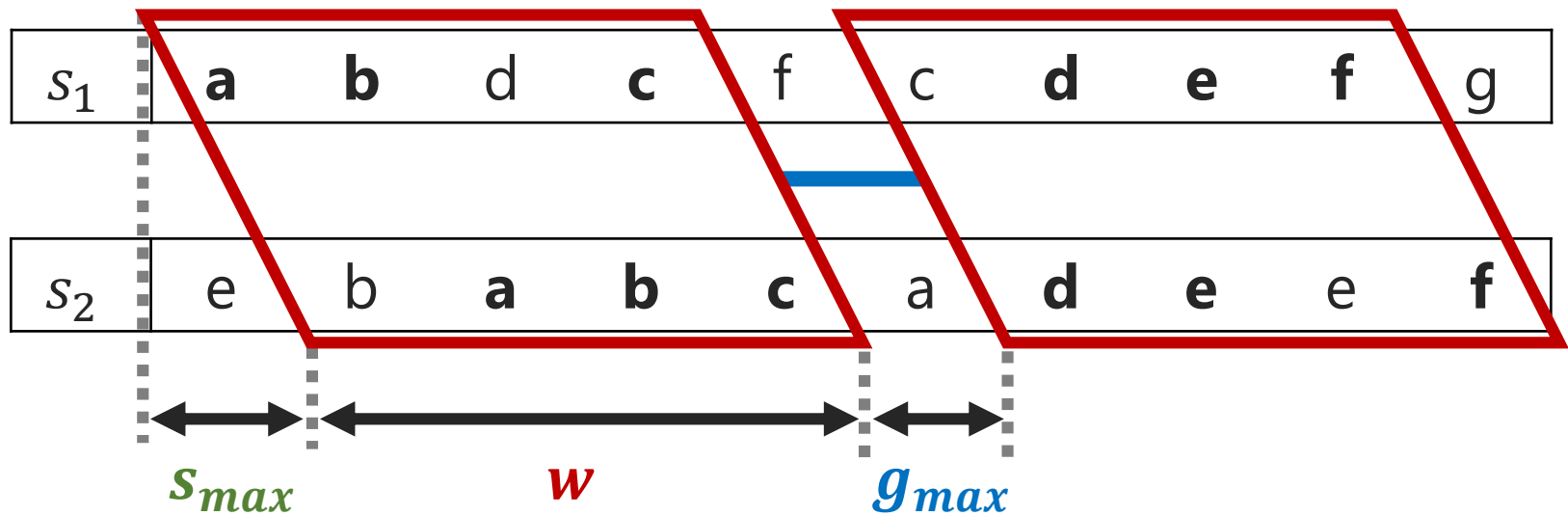
現実のイベントは**時間と密接に関係**

- ▶ LCSをそのまま適用しても有益な知識は検出できない
- ▶ **時間に関する制約を考慮**したLCSを提案
 - ▶ Window-Chained Longest Common Subsequence

Window-Chained Longest Common Subsequence

時間に関する三つの制約

1. LCSはある**ウィンドウ w** の中に存在
2. ウィンドウは g_{max} **以下の間隔で連なる**
3. シーケンス間のウィンドウの**ズレは s_{max} 以下**



解法のアイデア

2つの部分問題に分割

1. LCSが存在する全ウィンドウの検出

▶ 動的計画法 or 空間インデックスの利用

	a1	b2	c3	c4	d5
a1	(1,1,1)	(1,1,1)	(1,1,1)		
a2	(1,2,1)	(1,2,1)	(1,2,1)		
b3	(1,2,1)	(1,2,2) (2,3,1)	(1,2,2) (2,3,1)	(2,3,1)	
c4	(1,2,1)	(1,2,2) (2,3,1)	(1,2,3) (2,3,2) (3,4,1)	(2,3,2) (4,4,1)	(4,4,1)
d5		(2,3,1)	(2,3,2) (3,4,1)	(2,3,2) (4,4,1)	(4,4,2) (5,5,1)

	a1	c2	a3	b4	c5
b1				*	
a2	*		*		
c3		*			*
b4				*	
a5	*		*		

一致する箇所を
インデックスで
検出し, そこを
起点にウィンド
ウを生成

2. 一致長が最大となるウィンドウの組合せを探索

解法のアイデア

2つの部分問題に分割

1. LCSが存在する全ウィンドウの検出

2. 一致長が最大となるウィンドウの組合せを探索

▶ ウィンドウの集合をグラフに変換し経路探索

■ ウィンドウを頂点とし, 連結可能なペアに辺を引く

