Continuous Pattern Mining Using The FCPGrowth Algorithm In Trajectory Data Warehouses

Marcin Gorawski, Paweł Jureczek
Silesian University of Technology
Institute of Computer Science
e-mail: {Marcin.Gorawski, Pawel.Jureczek}@polsl.pl
HAIS, San Sebastián, Spain, June 23-25, 2010

Outline

- Motivations
- FCP-Tree
- FCPGrowth
- Results
- Conclusion
- Future work

Motivations

- great interest in mining behavior patterns
- efficient mining of frequent continuous patterns of mobile objects
- large dataset of long trajectories

Frequent continuous patterns

Definitions:

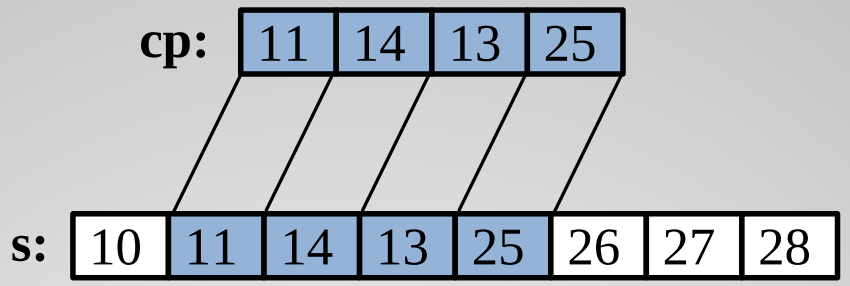
1) Given two sequences $X = \langle b_1 b_2 \ldots b_m \rangle$ and $Y = \langle a_1 a_2 \ldots a_n \rangle$, where $m \leq n$, the sequence X is a continuous subsequence of Y if there exists an integer i such that $b_1 = a_i$, $b_2 = a_{i+1}$, …, $b_m = a_{i+m-1}$, and if for any two elements b_i and b_j $(i \neq j)$ we have $b_i \neq b_j$.

2) $sup(s) = |\{s \mid s_i \in S \wedge s \subseteq s_i\}|/m$ where $S = \{s_1, s_2, \ldots s_m\}$

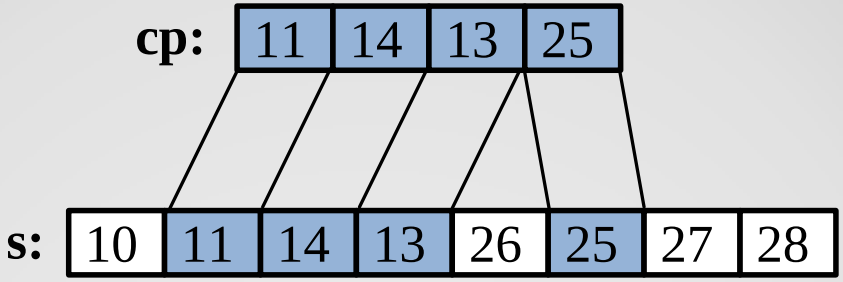
3) $sup(s) \geq min\_sup$

Continuous pattern - Example

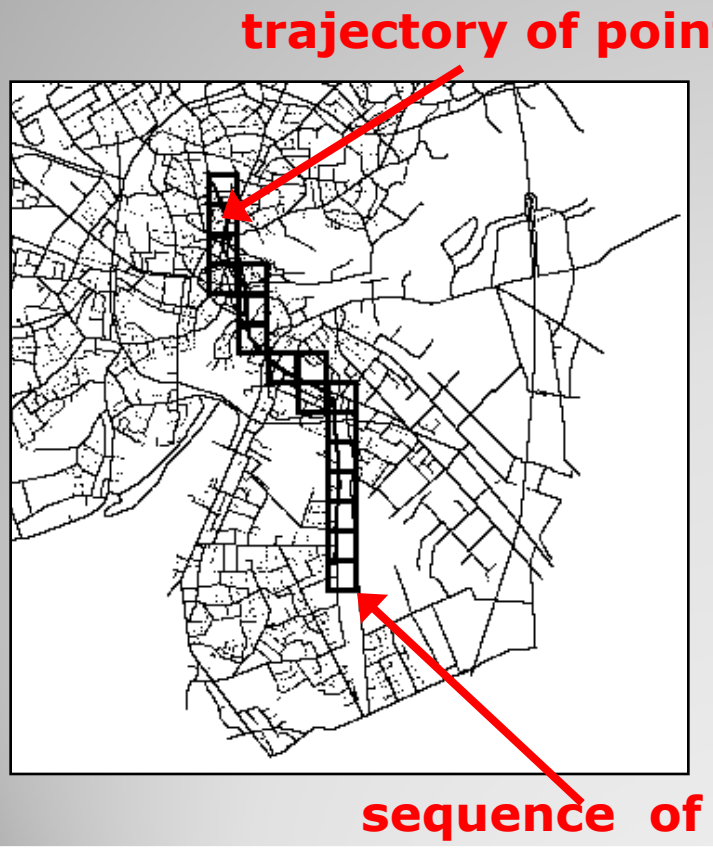
a)



b)



Spatio-temporal continuous sequence



Steps:

- determine grid resolution
- map the points of a given trajectory into regions of the sequence

FCP-Tree

- prefix tree
- compressed data structure
- header table

Building FCP-Tree – Example (1)

a) sequence database with minimum support ***minsup*=33,3%**

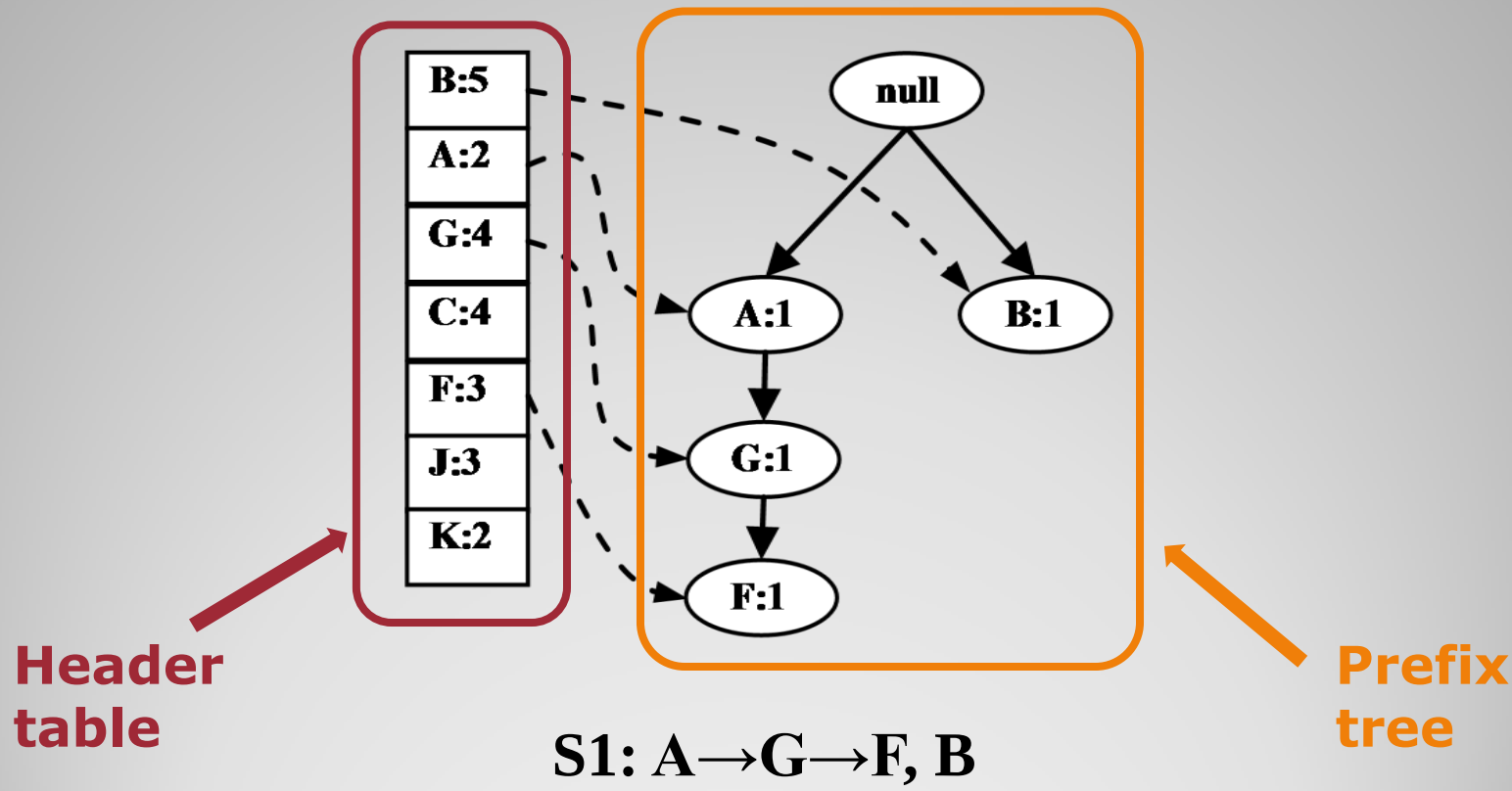
No	Input sequence
1	A→G→F→I→B
2	A→G→F→J→K
3	C→G→Z→B
4	B→G→F
5	B→C→J→K
6	B→C→J

b) support for elements: **A:2, B:5, C:3, F:3, G:4, I:1, J:3, K:2, Z:1**

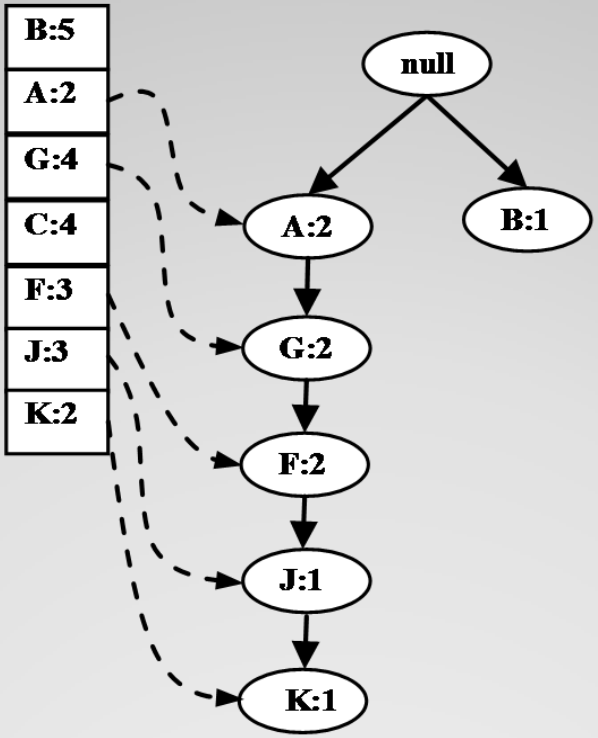
Building FCP-Tree – Example (2)

No	Input	Output
1	A→G→F→I→B	A→G→F, **B**
2	A→G→F→J→K	A→G→F→J→K
3	C→G→Z→B	C→G, **B**
4	B→G→F	B→G→F
5	B→C→J→K	B→C→J→K
6	B→C→J	B→C→J

Building FCP-Tree – Example (3)

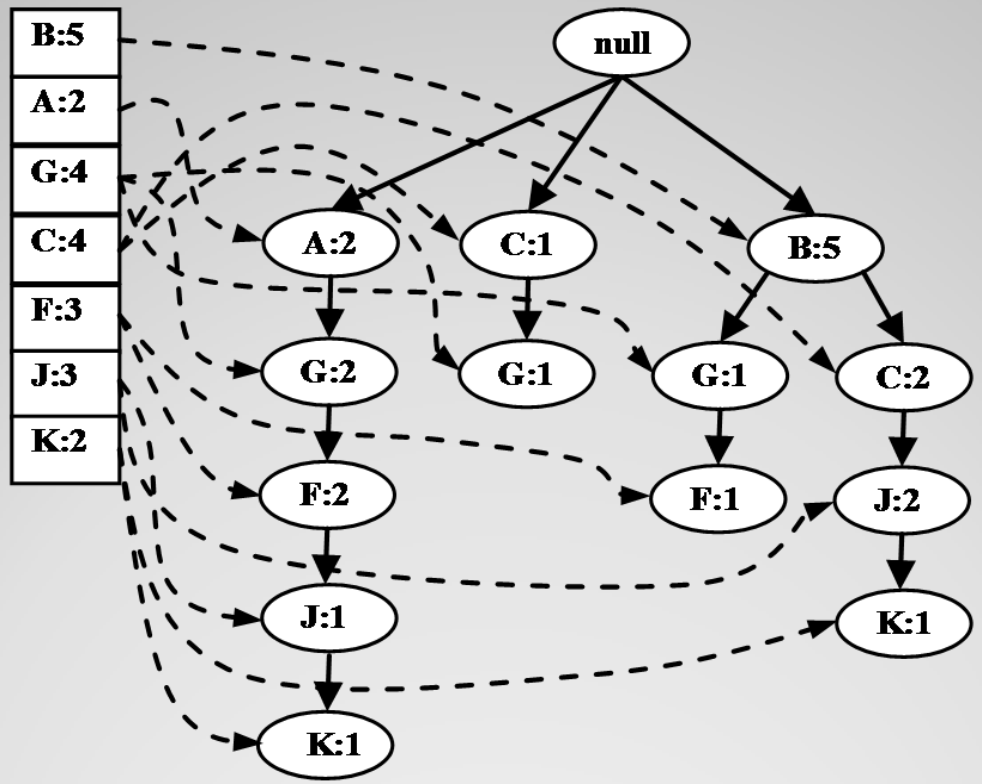


Building FCP-Tree – Example (4)



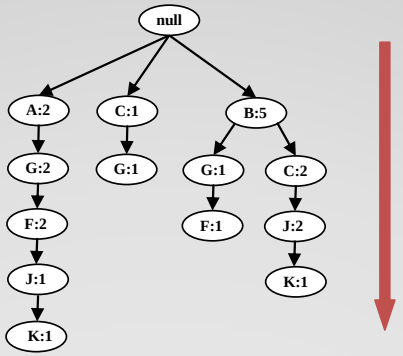
S2: A→G→F→J→K

Building FCP-Tree – Example (5)



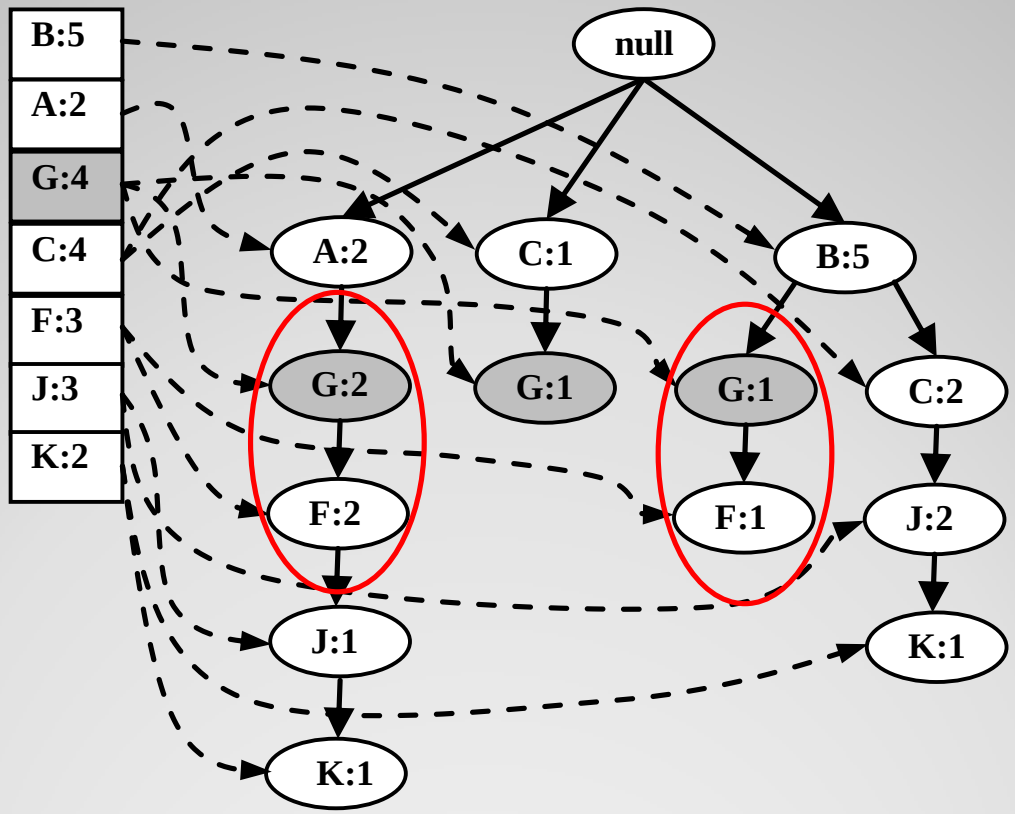
FCPGrowth

- top-down approach



- divide and conquer
- no intermediate subtrees

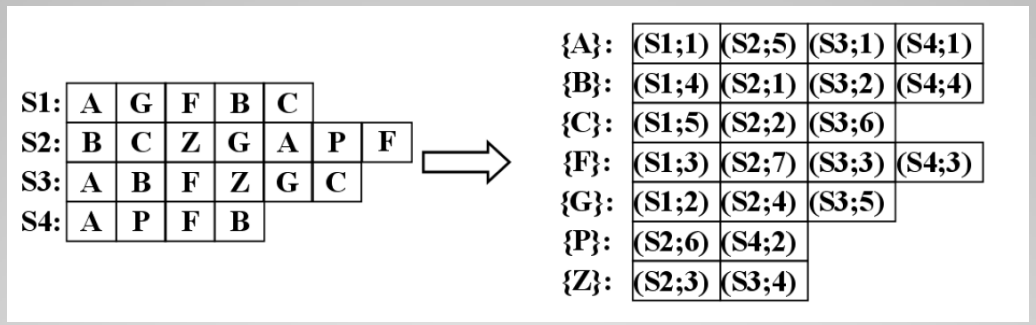
FCPGrowth – Example

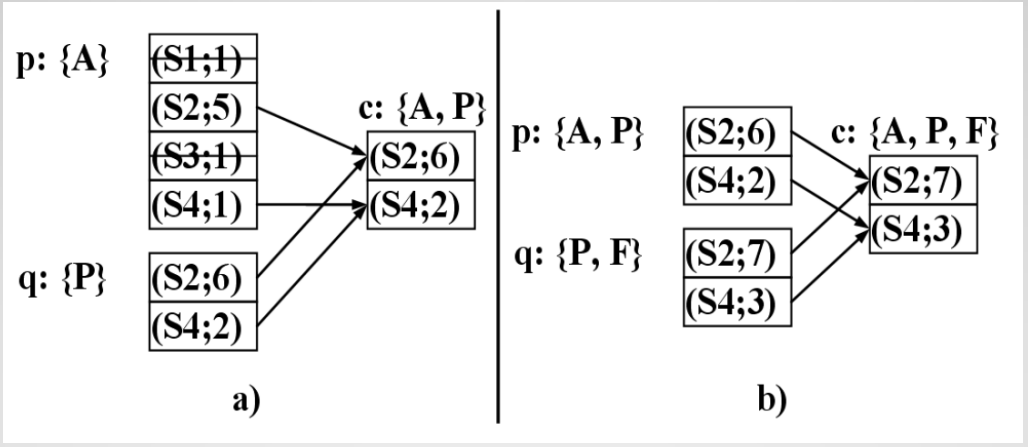


Experimental results

- Brinkhoff's network-based generator
- No. of Sequences: 10 – 50k
- minsup: 0.1 – 0.5%
- Avg. length: ~14.3
- Unique elements: 550 ~ 630

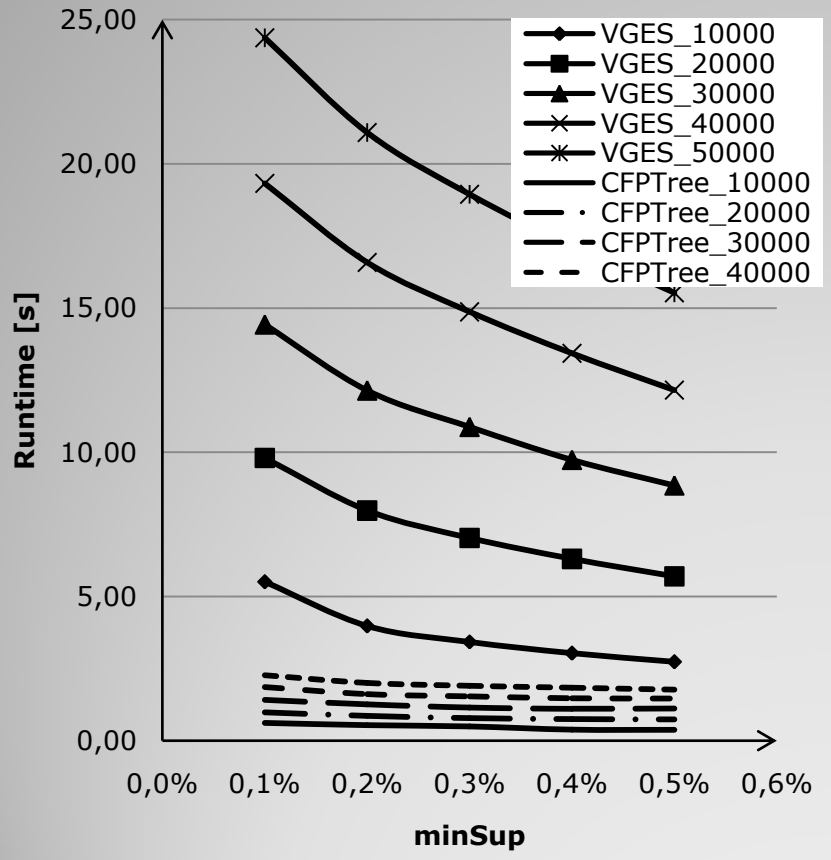
VAES (Vertical Approach for Exact Search)



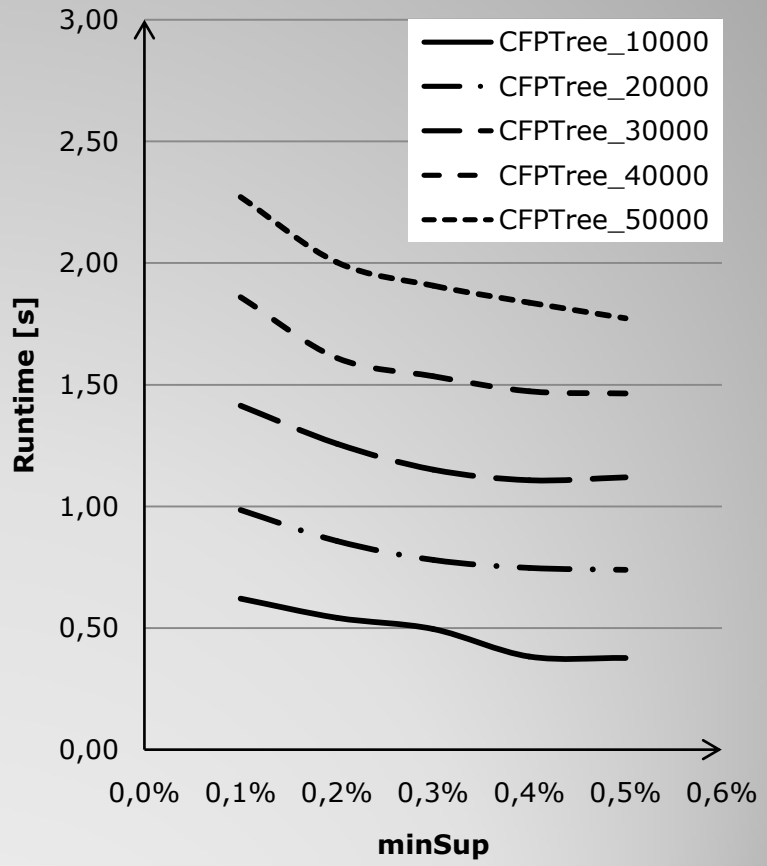


***minsup*=2**

Experiments



a)



b)

Conclusion

- promising approach for compressing continuous sequences
- algorithm for mining continuous patterns
- experiments

Future work

- analysis of the tree size and complexity
- maximal and closed pattern extensions
- real data experiments

Thank you for your attention!