同期と非同期を許した群れパターン発見：
サイバーフィジカルマイニングに向けて

有村　博紀

北海道大学大学院情報科学研究科

耿暁亮(北大D2)，宇野毅明（NII）
との共同研究

動機：モビリティ，サイバーフィジカル

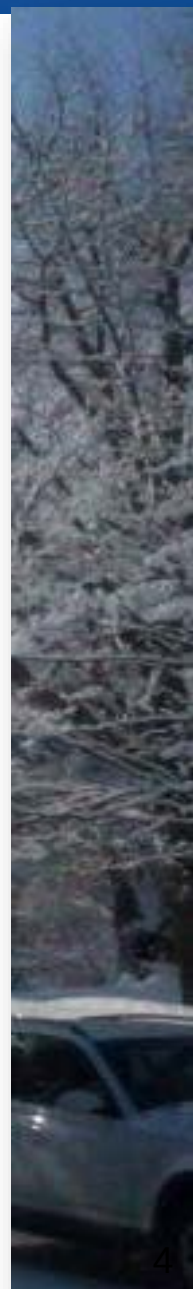
- CPS（サイバーフィジカルシステム）
- 人とモノのモビリティに関心
  - 移動を通じて人間の活動にアクセス
  - 社会活動の最適化（スマートXX）
  - 各種サービス・産業の基盤と媒体となる？
- 大量の移動体データ
  - プローブカー，歩行者，野生動物？
  - GPS, スマホ，WIFI，etc.
- どのような情報を取り出す？
  - 時空間における移動の解析・予測
  - 移動パターンの発見（「トラジェクトリパターン」）

なぜ時空間マイニング

- パターンマイニング
  - 複雑な構造　－　グラフマイニング
  - 複雑な同値性　－　Closed pattern mining
  - 応用可能性　－　？（統計モデリング，機械学習と関連
- 実世界を扱いたい
- 時空間データは，連続性や幾何的不変性があって，（難しくて）面白そうだ．他のものとの「紐付け」も必要そうだ．
- 以前した研究）
  - 先行研究：幾何グラフ（Kuramochi & Karipis）．
  - 2次元平面上の幾何グラフのマイニング（Arimura, Shimozono, DS'07）．並進，回転，拡大の不変性をあつかった．そのまま．．．
  - 津田先生が3次元空間かつロバストに！
- 要は面白そう

GCPSプロジェクト(H23～H27予定)

- **「グリーン・サイバー・フィジカル・システム基盤技術開発」**（代表：坂内先生->安達先生）
- NII，九大，北大，阪大の4拠点で
  - NII（安達淳）「IT統合基盤のCPS共通技術」
  - 九大（安浦寛人）「データ収集／解析技術と学研都市スマートシティ化への適用」
  - 北大（田中譲）「オープン・スマート・フェデレーション技術と**スマート除排雪への適用実証実験**」
  - 阪大（東野輝夫）「プラットフォーム技術と都市街区における行動」
- 北大は**「スマート除排雪への適用実証実験」**



動機：サイバーフィジカルシステム

- CPS（サイバーフィジカルシステム）
- 人とモノのモビリティに関心
  - 移動を通じて人間の活動にアクセス
  - 社会活動の最適化（スマートXX）
  - 各種サービス・産業の基盤と媒体となる？
- 大量の移動体データ
  - プローブカー，歩行者，野生動物？
  - GPS, スマホ，WIFI，etc.

- どのような情報を取り出す？
  - 時空間における移動の解析・予測
  - 移動パターンの発見（「トラジェクトリパターン」）

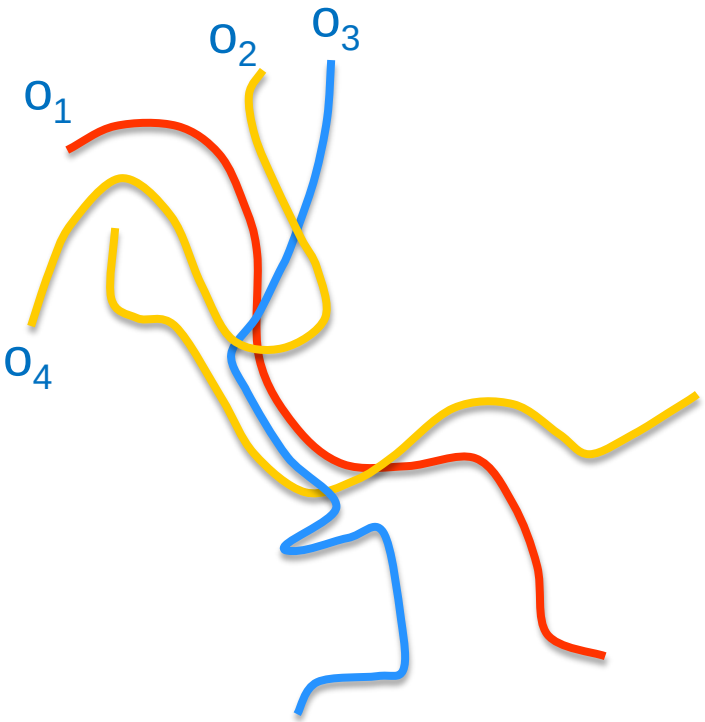
軌跡(トラジェクトリ)データ

- 時空間データはさまざまな定式化が可能

① 移動体(moving objects)の集合 $O = \{o_1,\ldots,o_n\}$
- 野生動物，歩行者，プローブカー
- 付加情報は仮定しない(属性ラベルなし)

② 時間 **T**
- 連続時間 **T** = R
- 離散時間 **T** = [0..T]. (等間隔)

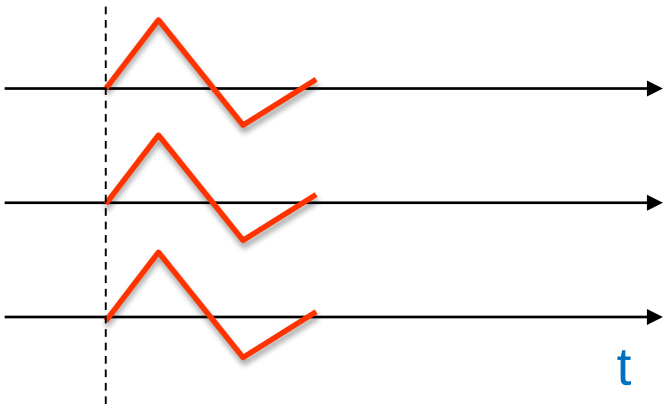
③ 空間 **S**
- 2次元連続空間 **S** = R^2
- 2次元のメッシュ **S** = $[0..u]^2$
- 道路ネットワーク **S** = (V, E)



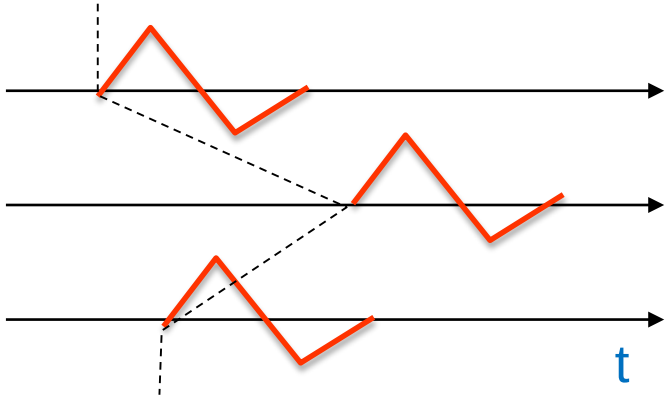
定義:「群れ」パターン

- 同期型と非同期型のパターン
  - 移動体の位置が，厳密に同じ時刻で同期するか（同期型），同期せず相対時刻の意味で近接するか（非同期型）の違い.
  - オリジナルの群れパターンは同期型.
  - 今回は，非同期型も導入する.

同期型



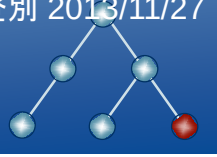
非同期型



定義:「群れ」パターン

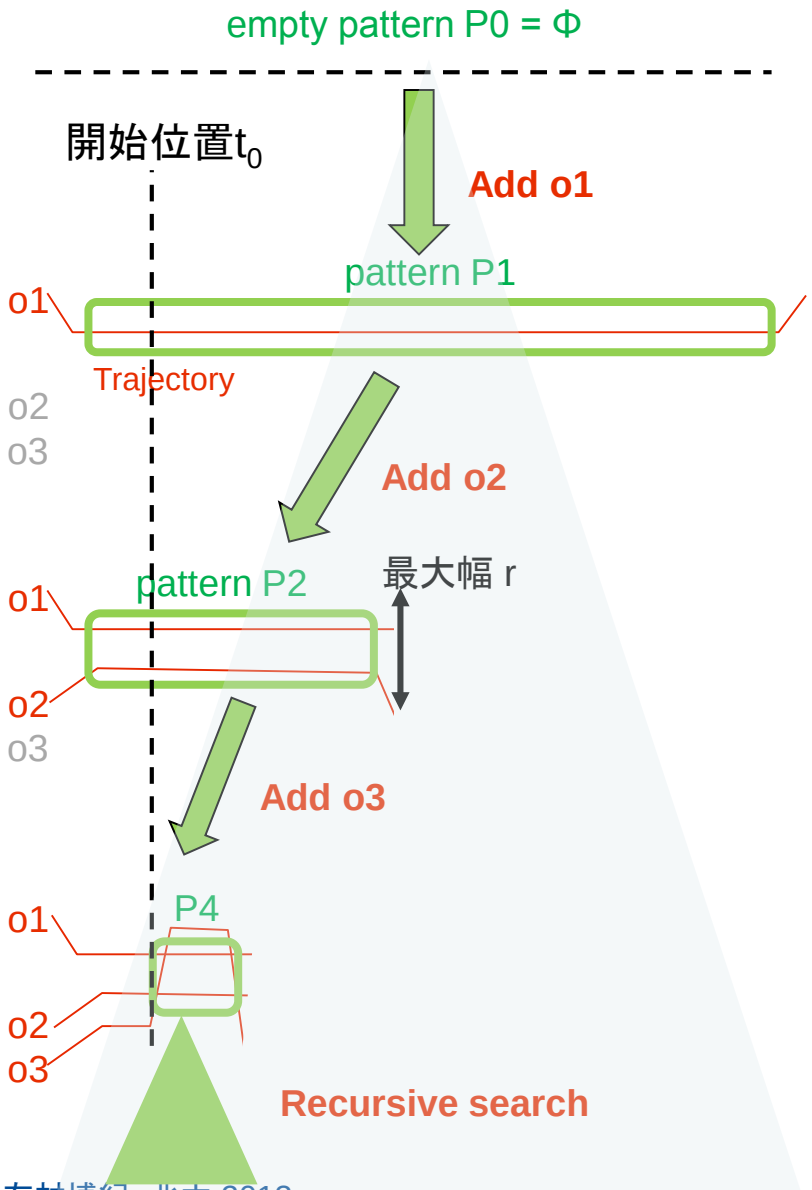
- 長さ極大群れパターン
  - 群れパターン P = (X, A)で，同じ移動体集合Xに対して，最大の幅 r の制約を保ったまま，区間 A=[beg, end] がそれ以上左右に広げられないようなもの. （右極大群れパターン＝右拡張だけを考えたもの）
- 極大と最大
  - 最小移動体数 m と最大幅 r の制約のもとで，長さ「最大」の群れパターンの発見は，一つでもNP完全 (Gudmundsson他, 2006)
  - 極大な一つは簡単にみつけられる.
- Q: すべての極大群れパターンの列挙が出力多項式時間（多項式遅延）でできるか？

同期型アルゴリズム1: FPM Basic



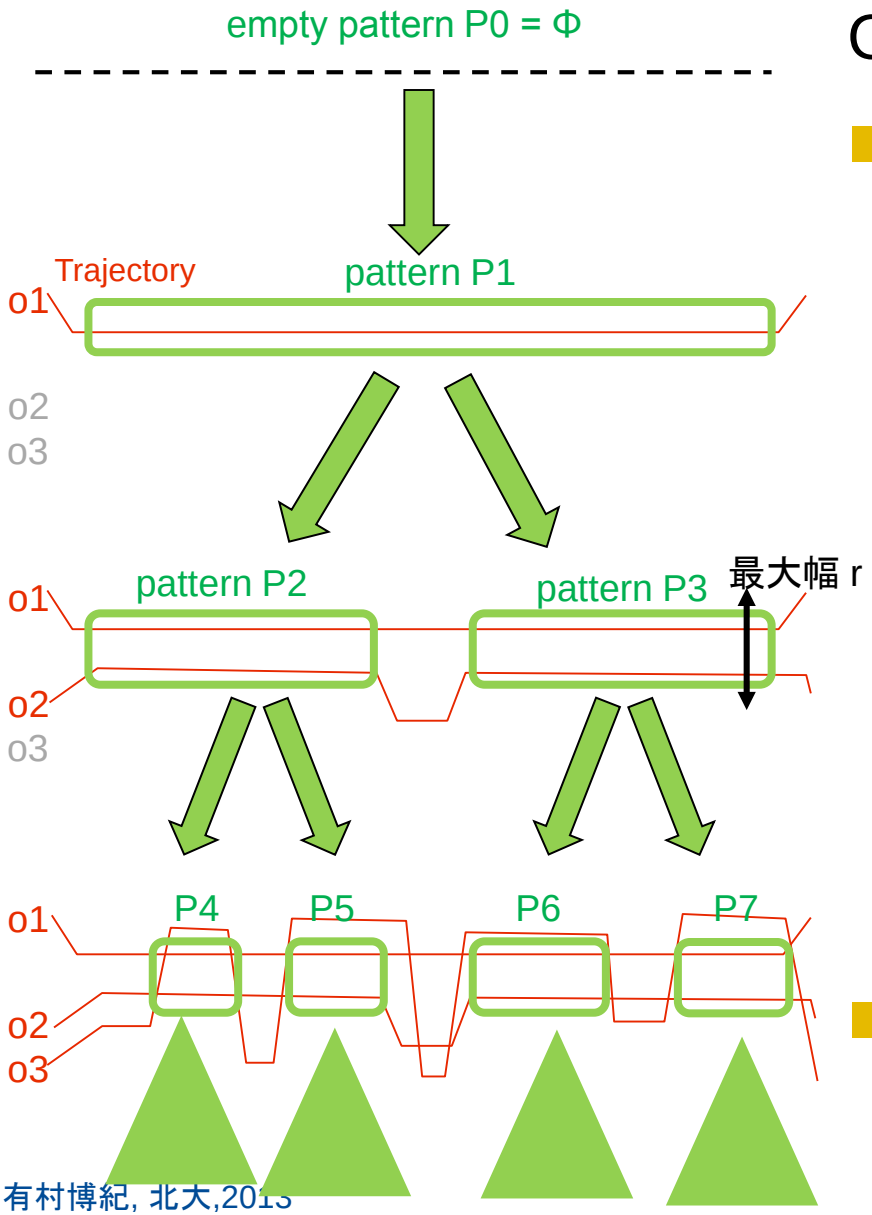
- 与えられた軌跡データベース中のすべての長さ極大群れパターン P = (X, A = [beg, end]) を列挙
- 基本アイディア：リジェクト法
  - 各開始時刻begに対して，集合 X を空集合から列挙.
  - 各Xとbegに対して，最大幅を守ったまま，区間を右へできるだけ延長する（右拡張）. 開始時刻begが左へ1ステップ延長できなければ長さ極大なので出力.
  - 区間Aの長さが制約をみたすなら，新しい移動体IDをXに追加して，再帰的に探索をする.
- 計算時間：
  - すべての長さ極大群れパターン P = (X, A = [beg, end]) を1個あたり，多項式時間($O(n^2T^2)$時間)ですべて列挙する
  - ただし，O(T)が余計！（Tは最大の時間軸なので大きい）

同期型アルゴリズム1: FPM Basic



- 計算方法:
  - **初期）** 開始位置t_0を決める. **空軌跡**Φからスタート.
  - **帰納）**親パターン (X, A)をもらい, 新しい軌跡o_iを1本足す. 幅rの最長部分区間 B ⊆ A を求める
  - テスト:Bの左端がt0に一致するならば, P = (X∪{o_i}, B)を長さ極大の子パターンとして出力.
  - 以上を再帰的にくりかえす.
- 定理（計算時間）:
  1個あたり, $O(n^2T^2)$時間ですべて列挙（多項式遅延）
- 問題点: ただし, O(T)が余計！

同期型アルゴリズム2:まとめ分割型



O(T)を落として，無駄なく列挙したい

- 計算方法:
  - **初期）空軌跡Φ**
  - **帰納）長さ極大な親パターン (X, A)をもらう.**
  - これに新しい軌跡を1本足す(一意に)．まとめて幅rの部分区間 $A_1, \ldots, A_h$を求める
  - すると$(X, A_1), \ldots, (X, A_h)$が，それぞれ長さ極大の子パターンに.
  - 以上を再帰的にくりかえす.
- 定理(計算時間):
  1個あたり，多項式時間($O(n^2T)$時間)ですべて列挙可能！

EXP1: Comparison of Sync & Async

- Comparing the #patterns and cputime for **FPMsync** and **FPMasync** algorithms

#patterns found	sync patterns	async patterns
true patterns	40	40
FPMsync	40	0
FPMasync	41	43

時間がズレたパターンが正しく見つかっている

total time	sync patterns	async patterns
FPMsync	0.640 sec	0.640 sec
FPMasync	0.733 sec	0.686 sec

with geo-index

Setting

- Area 100.0 x 100.0
- 400 trajectories of length 100 generated by random walk with step 1.0 and angle ±90deg
- 5 inplanted copies of each of 40 random patterns of length 10 within width 1.0x1.0
- Mining with max width 1.0x1.0, min length 10, and frequency at least 5.

EXP2: Speed-up by Geo-index

- Cpu time of FPMasync algorithm without/with geometric index

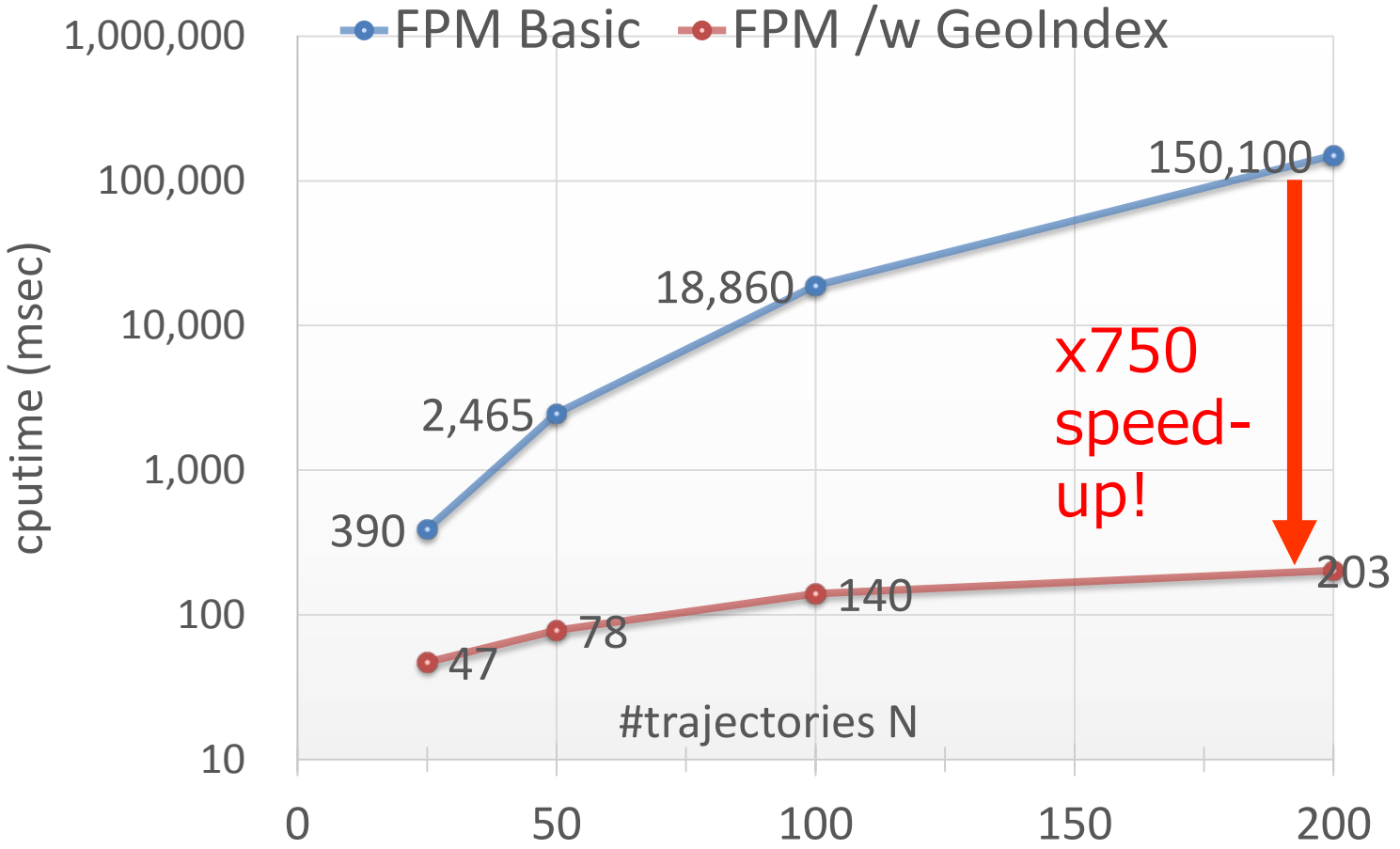
Setting: N = 25 to 200 trajectories of length 40 in which N/10 random patterns x 5 copies are implanted. Other parameters are same to EXP1:

#trajectories N	25	50	100	200
#patterns	2	5	10	20
time /wo geo (sec)	0.39	2.465	18.86	150.1
time /w geo (sec)	0.047	0.078	0.14	0.203

x750 speed-up at N = 200

EXP2: Speed-up by Geo-index

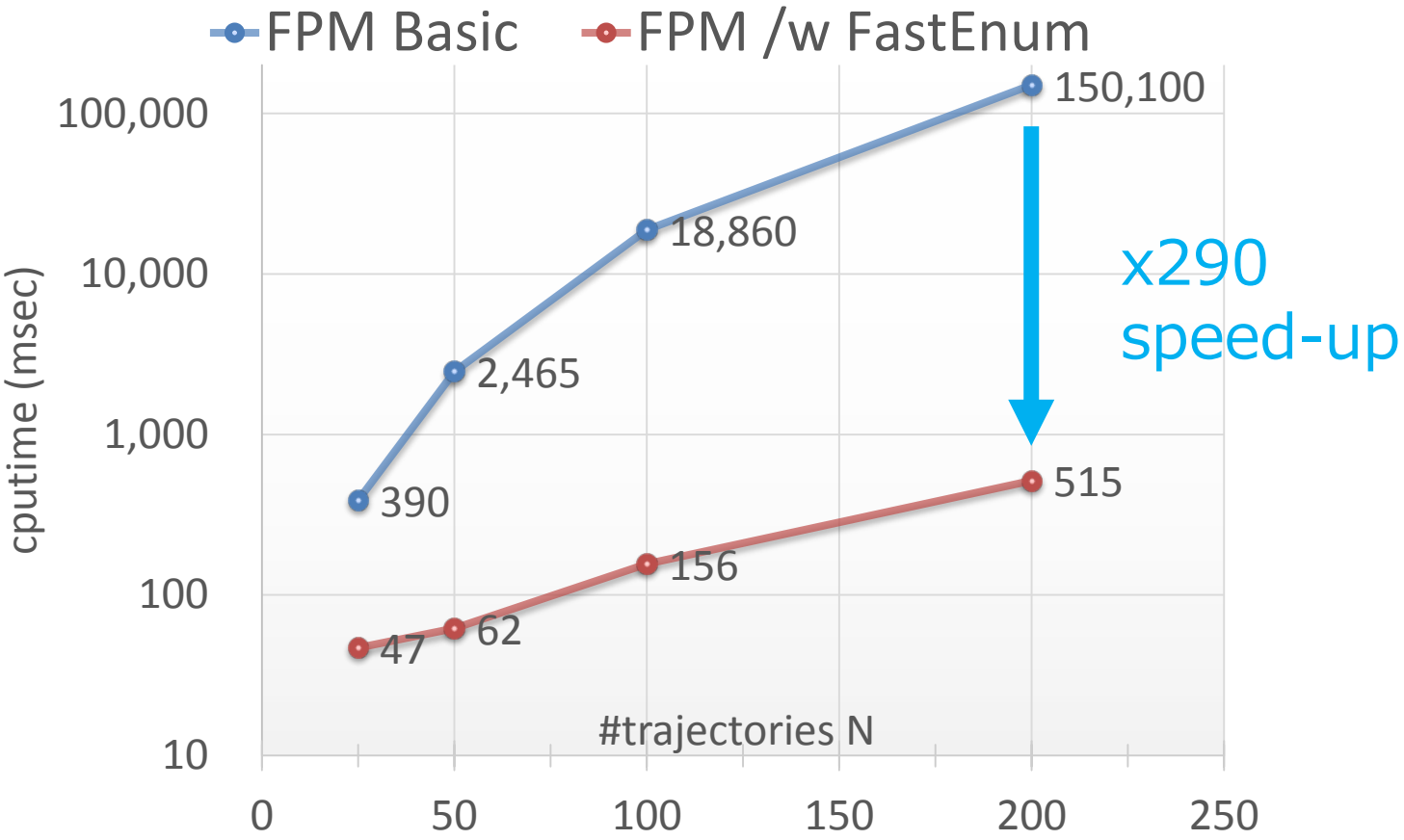
- CPU time of FPM algorithm without/with geometric index



Setting: N = 25 to 200 trajectories of length 40 in which N/10 random patterns x 5 copies are implanted. Other parameters are same to EXP1:

EXP3: Enumeration strategy

- Comparison of FPM Basic and FPM FastEnum (Extend and Split algorithm)



Setting: N trajectories of length 40. Other parameters are same to EXP2:

まとめ

- CPS，モビリティ，時空間マイニング
- 軌跡データ(トラジェクトリデータ)
- 群れパターンマイニング：同期型と非同期型
- 多項式遅延の列挙アルゴリズム
  - FPM Basic(リジェクト法)：$O(n^2T^2)$遅延アルゴリズム
  - FPM Fast. $O(n^2T)$遅延アルゴリズム.
  - 幾何索引(レンジクエリ)を用いた実際的な高速化手法
- 実験　ー　人工データで概ねよさそう(動きそう)
- 今後の課題
  - 疎な群れパターン. ● 軌跡のためのカーネル？
  - 道路ネットワーク上のマイニング・索引・圧縮