【VLDB2013勉強会】

Session 12 : Spatial and Text

担当:姜(名古屋大学)

Direction-Preserving Trajectory Simplification

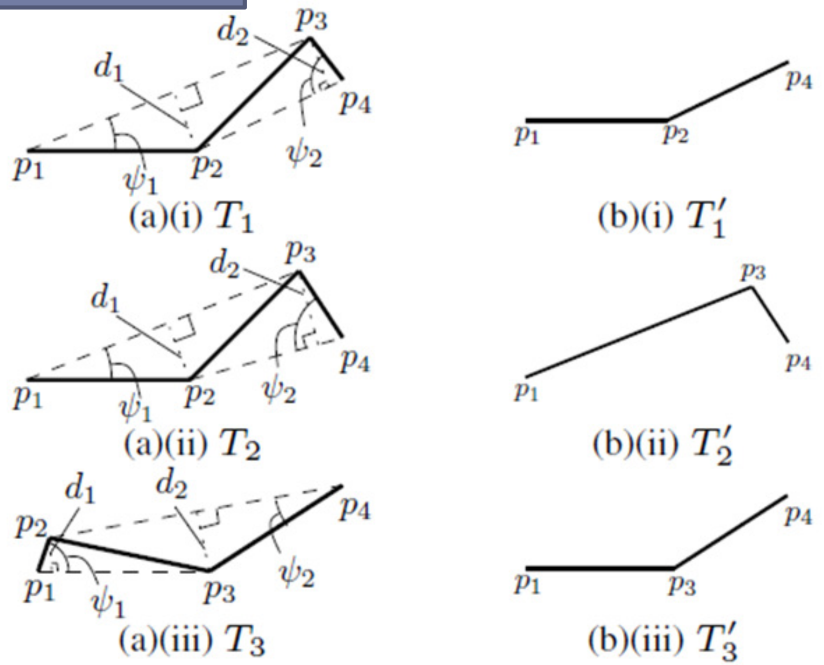
Cheng Long, Raymond Chi-Wing Wong (HKUST)

DPTS → 方向保存型の軌跡の簡略化

- **既存研究：位置保存型の簡略化**
  - **問題点**

    簡略化される前、軌跡が類似

    簡略化された後、軌跡が類似しない



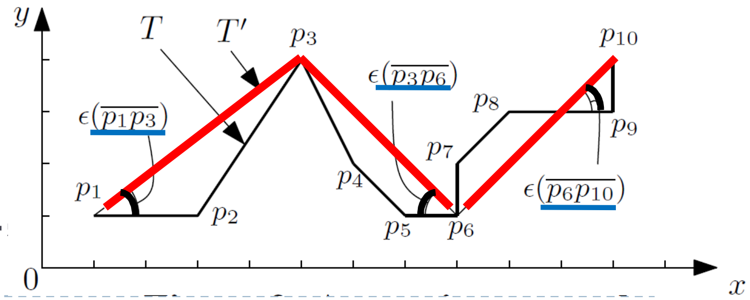
- **前提**
  - 方向の許容誤差 ϵt

    $\epsilon(T') = \max_{1 \le k < m} \epsilon(\overline{p_{s_k} p_{s_{k+1}}})$

    例： T' = (p_1, p_3, p_6, p_{10})

    $\epsilon(\overline{p_1 p_3})\ \epsilon(\overline{p_3 p_6})\ \epsilon(\overline{p_6 p_{10}})$

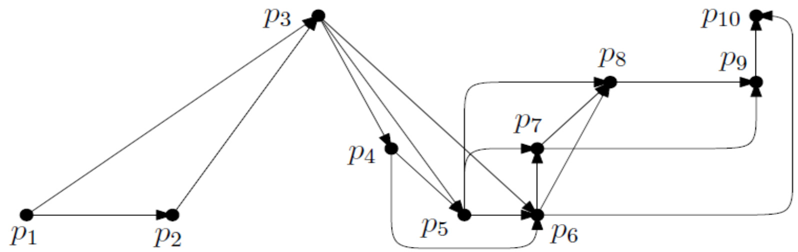
    ϵ(T')= max {0.644, 0.785, 0.785} = 0.785



- **問題定義**
  - Input: 軌跡データT, 許容誤差 ϵt
  - Output: 最小サイズのϵt−簡略化のT→T

提案手法:SPアルゴリズム

- **Step1: グラフ$G_{\epsilon t}$(V,E)を構築**
  - T中のpositionを一つ一つVに入れる
  - $\epsilon(p_i p_j) \leq \epsilon t$の場合、エッジ$(p_i p_j)$を作成



- **Step2:$G_{\epsilon t}$上でp_1からp_nまでの最短経路を発見**
  - 経路のエッジの数は経路の長さとなる
  - 最短経路は最小サイズの簡略化された軌跡となる
- **Step3:最短経路を用いてDPTSを生成**

例:
$\epsilon t=\pi/4=0.785$,
$T'=(p1,p3,p6,p10)$

時間複雑度

- Step1: $\epsilon(p_i p_j) \leq \epsilon t$　誤差計算$O(n^2)$、比較演算$O(n)$、全体が$O(n^3)$になる
- Step2: BFS　O(|V|) = $O(n)$, O(|E|) = $O(n^2)$, O(|V|+|E|) = $O(n^2)$
- Step3:　最短経路をDPTSに変換　$O(n)$

$O(n^3)$

空間複雑度

- $G_{\epsilon t}$(V,E)を維持するために、$O(|V|+|E|)$ の空間は必要となる

性能強化の技術

①実際の改善：$G_{\epsilon t}$ではなく、BFSで必要なエッジだけを構築

- L-length集合H_L（起点からの距離がL）と未訪問position集合を維持
- H_{L-1}中のp_i, U中のp_j, $\epsilon(p_i\ p_j) \leq \epsilon t$且つ$p_j$が$p_n$ではない場合 pjをUから$H_L$に移動

②複雑度の改善：$\epsilon(p_i p_j) \leq \epsilon t$

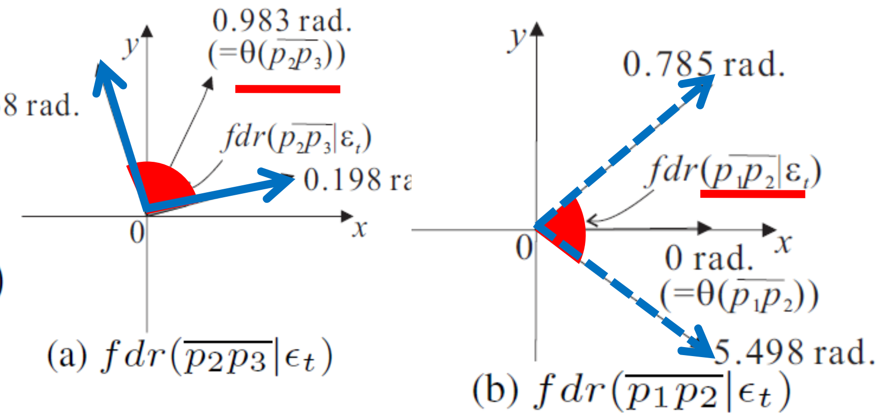
- **概念：**実行可能な方向範囲　→fdr

$$fdr(T[i,j]|\epsilon_t) = \cap_{i \leq h < j} fdr(\overline{p_h p_{h+1}}|\epsilon_t)$$

例：fdr(p_1,p_2)=|5.498, 0.785|

fdr(p_2,p_3)=|0.198, 1.768|

fdr(T|1,3|)=(p1,p2)∩(p2,p3) =|0.198,0.785|

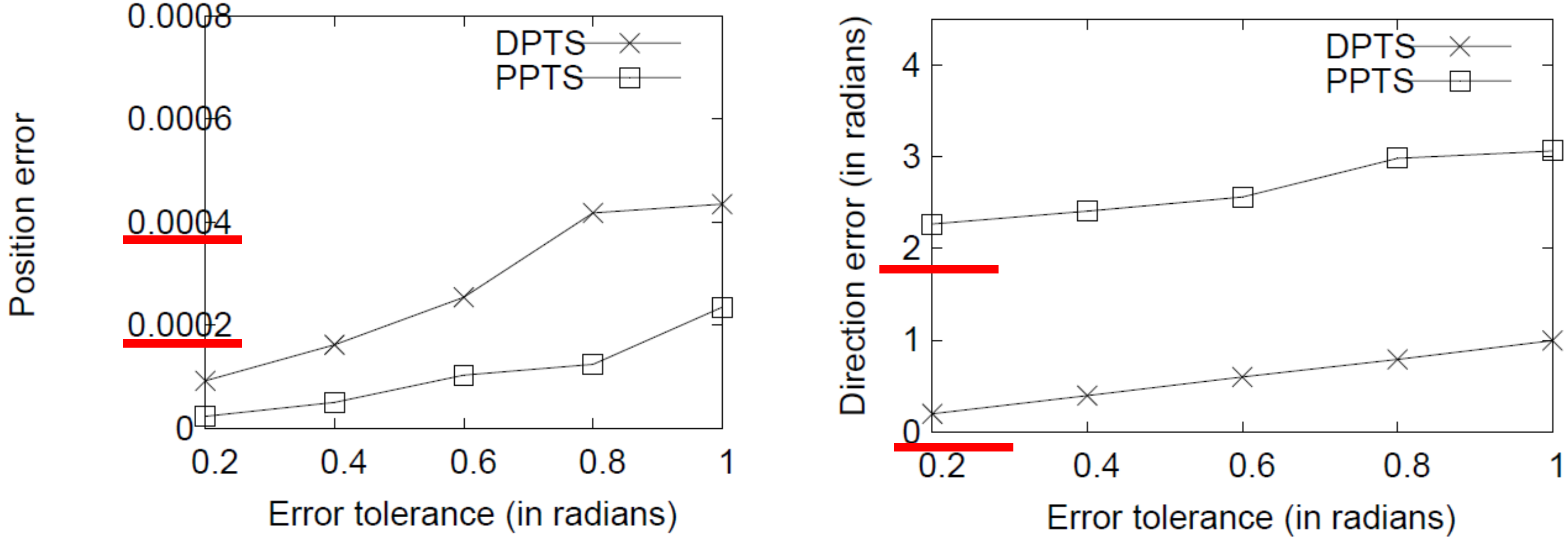


- **定理：**$\epsilon(p_i\ p_j) \leq \epsilon t \Leftrightarrow \theta(p_i\ p_j)$ *in* fdr(T[i, j])
  - fdr断片数の最大値はC、計算時間は$O(C)$
  - $\epsilon(p_i p_j) \leq \epsilon t$の計算時間は$O(n) \rightarrow O(C)$

	時間複雑度	空間複雑度
①	$O(n^3)$	$O(n)$
②	$O(C \cdot n^2)$	$O(C \cdot n + \|V\| + \|E\|)$
①+②	$O(C \cdot n^2)$	$O(C \cdot n)$

実験評価

- DPTS(方向保存型)vs PPTS(位置保存型)
  - **位置誤差の差は小さいが、方向誤差の差は大きい**



(a) Position error　(b) Direction error

- 結論
  - DPTSはある程度位置情報を保存できる
  - PPTSは方向情報を保存できない