

GPSの移動履歴と地図上経路の近似照合

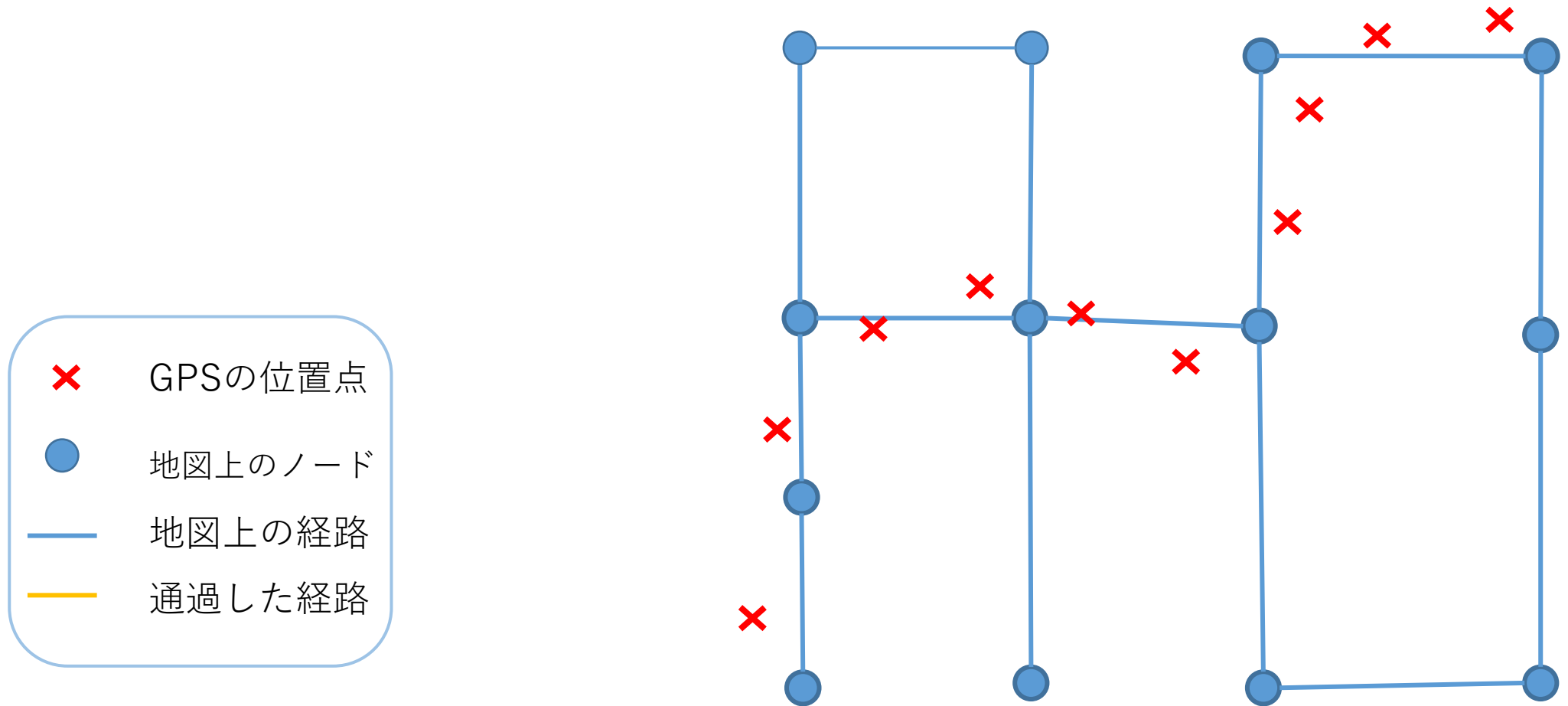
知能情報工学科 4年

182C1120 中島健斗

1. はじめに
2. 研究内容
3. 提案手法
4. 実験
 - ー 地図データ
 - ー 照合結果
5. 考察とまとめ

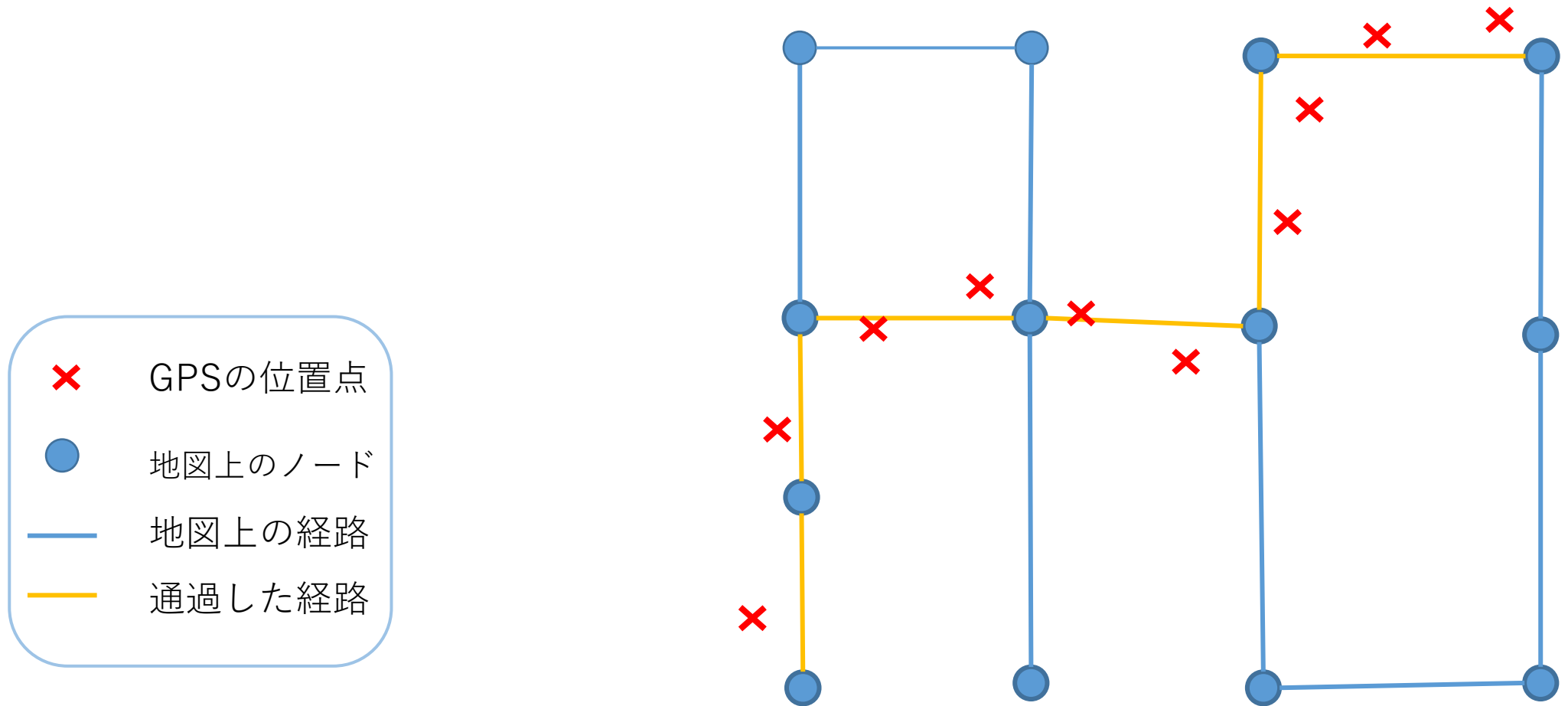
1 はじめに

GPSログと地図上経路の近似照合とは・・・



1 はじめに

GPSログと地図上経路の近似照合とは・・・



2 研究内容

研究内容

点列と線分列の照合による移動履歴と地図上経路のマッチング

問題の定義

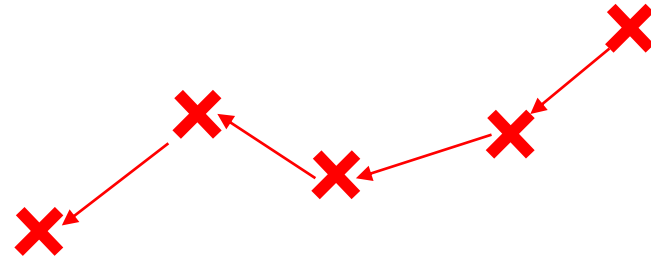
入力 移動履歴 $S = \langle (s_1, s_2), (s_2, s_3), \dots, (s_{m-1}, s_m) \rangle$,

地図上のノード $T = \langle t_1, t_2, t_3, \dots, t_n \rangle$

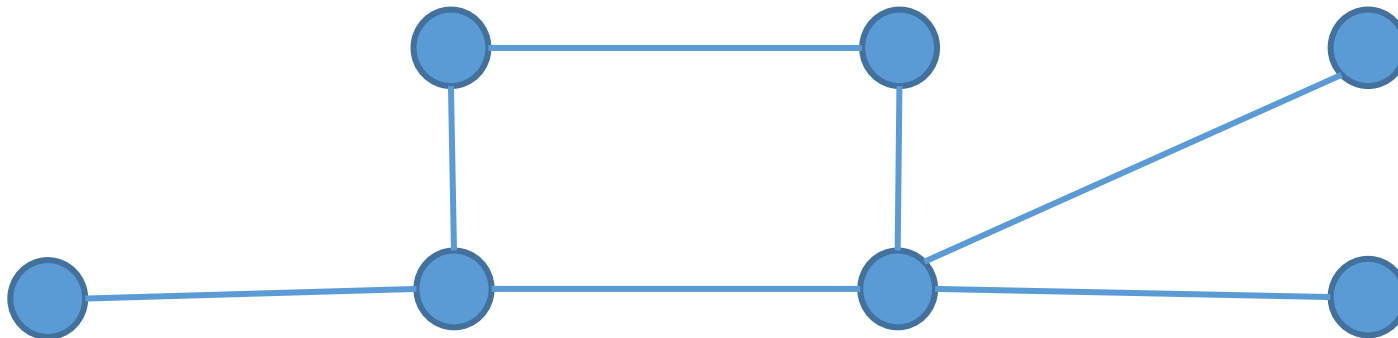
出力 実際に通過した地図上の共通経路 T' (ただし, $T' \in T$)

2 研究内容

移動履歴S → 時系列順に昇順な有向グラフ



地図上のノードT → 向きが存在しない無向グラフ



3 提案手法

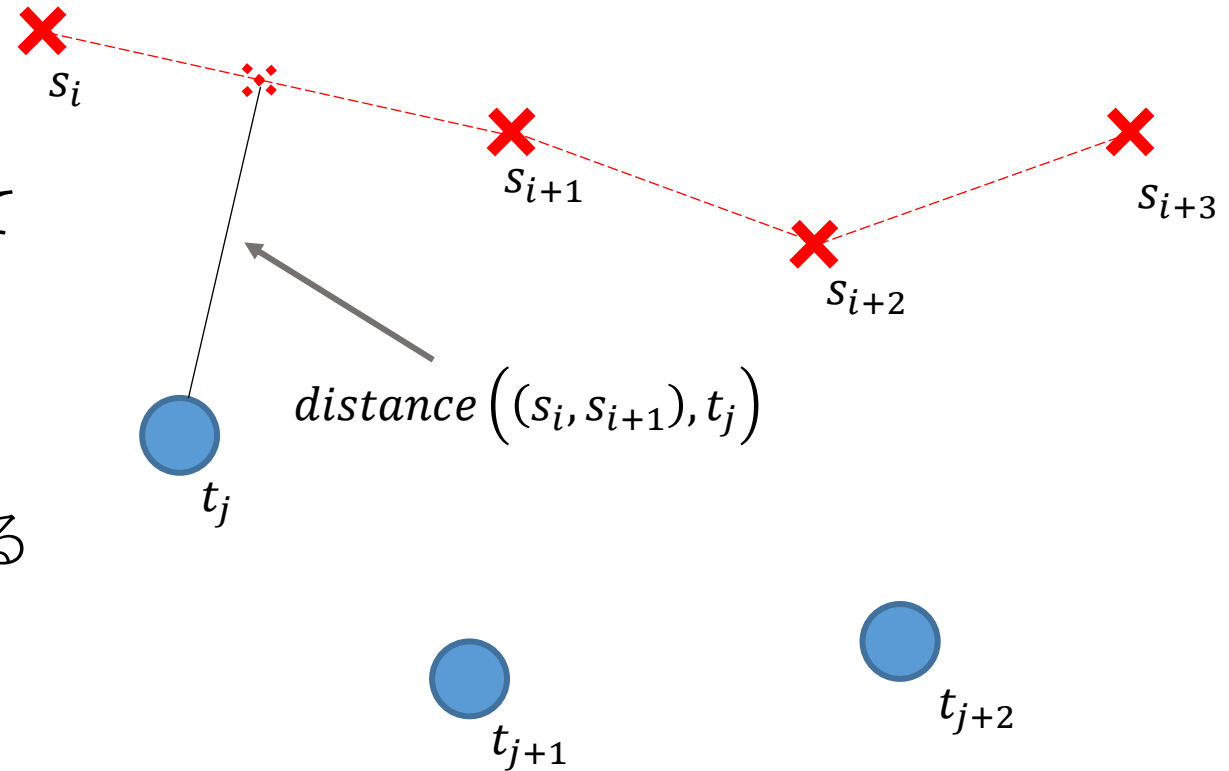
- 移動履歴を線分, 地図上のノードを点とする

- 同一視してよい距離の最大値を δ として

全ての (s_i, s_{i+1}) に対して

$distance((s_i, s_{i+1}), t_j) \leq \delta$ の判定をする

真のとき1, 偽のとき0を返す述語関数



3 提案手法

$$S = \langle (s_1, s_2), (s_2, s_3), (s_3, s_4), (s_4, s_5), (s_5, s_6) \rangle$$

$$T = \langle t_1, t_2, t_3, t_4, t_5 \rangle$$

	s_1	s_2	s_3	s_4	s_5	s_6
t_1	0	0	0	0	1	0
t_2	0	1	0	0	0	0
t_3	0	0	1	0	0	0
t_4	0	0	0	0	0	0
t_5	1	0	0	0	0	0

類似度テーブル

- $dp(i, j)$ に $distance((s_i, s_{i+1}), t_j) \leq \delta$ を加算
類似度テーブルのマスを埋めていく

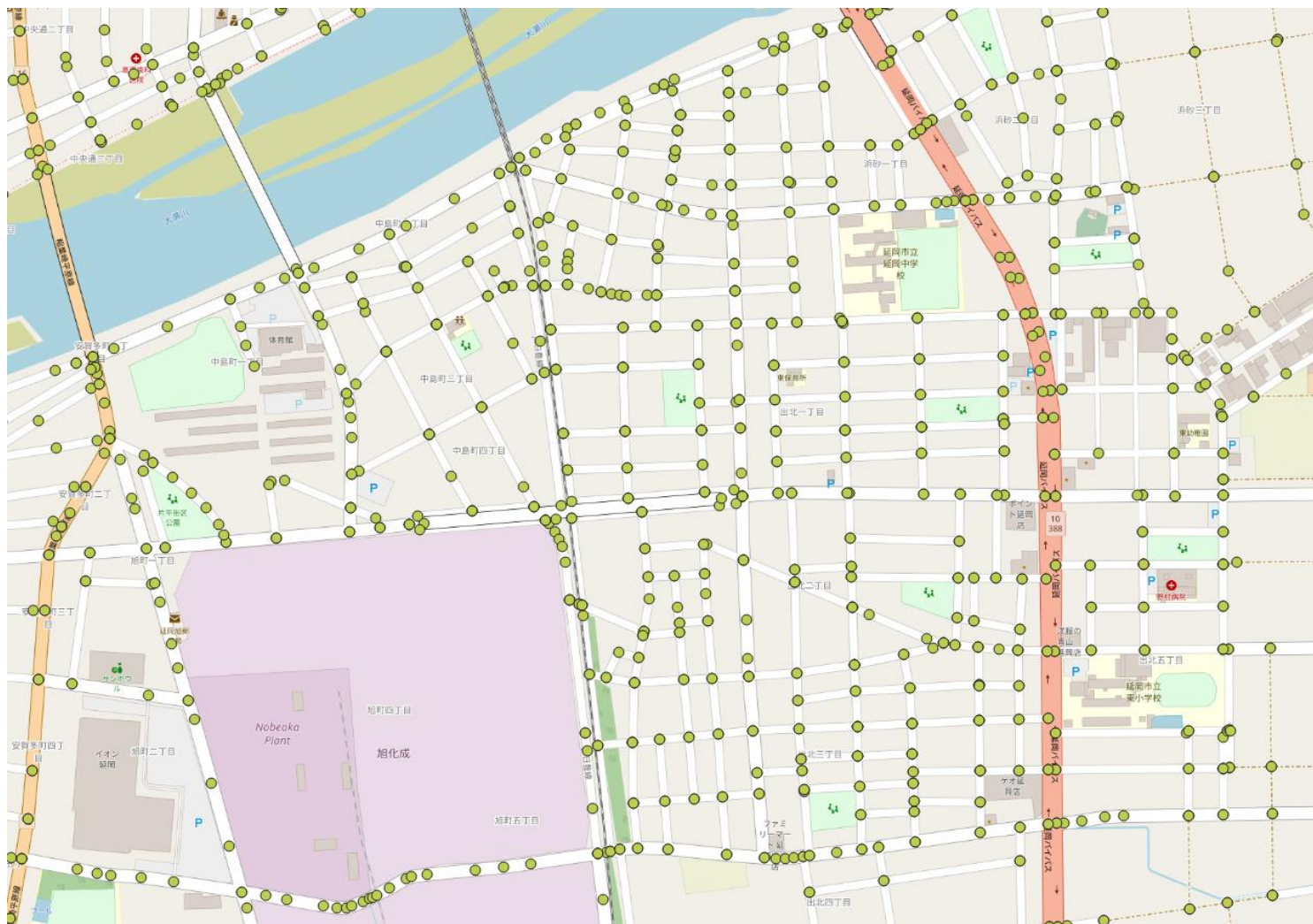
類似度テーブルを埋める計算量は $O(mn^2)$

- $dp(i, j) = 1$ のとき地図上の位置点 t_j を抽出し
共通経路 T' を構成する

バックトラックによる計算量は $O(mn)$

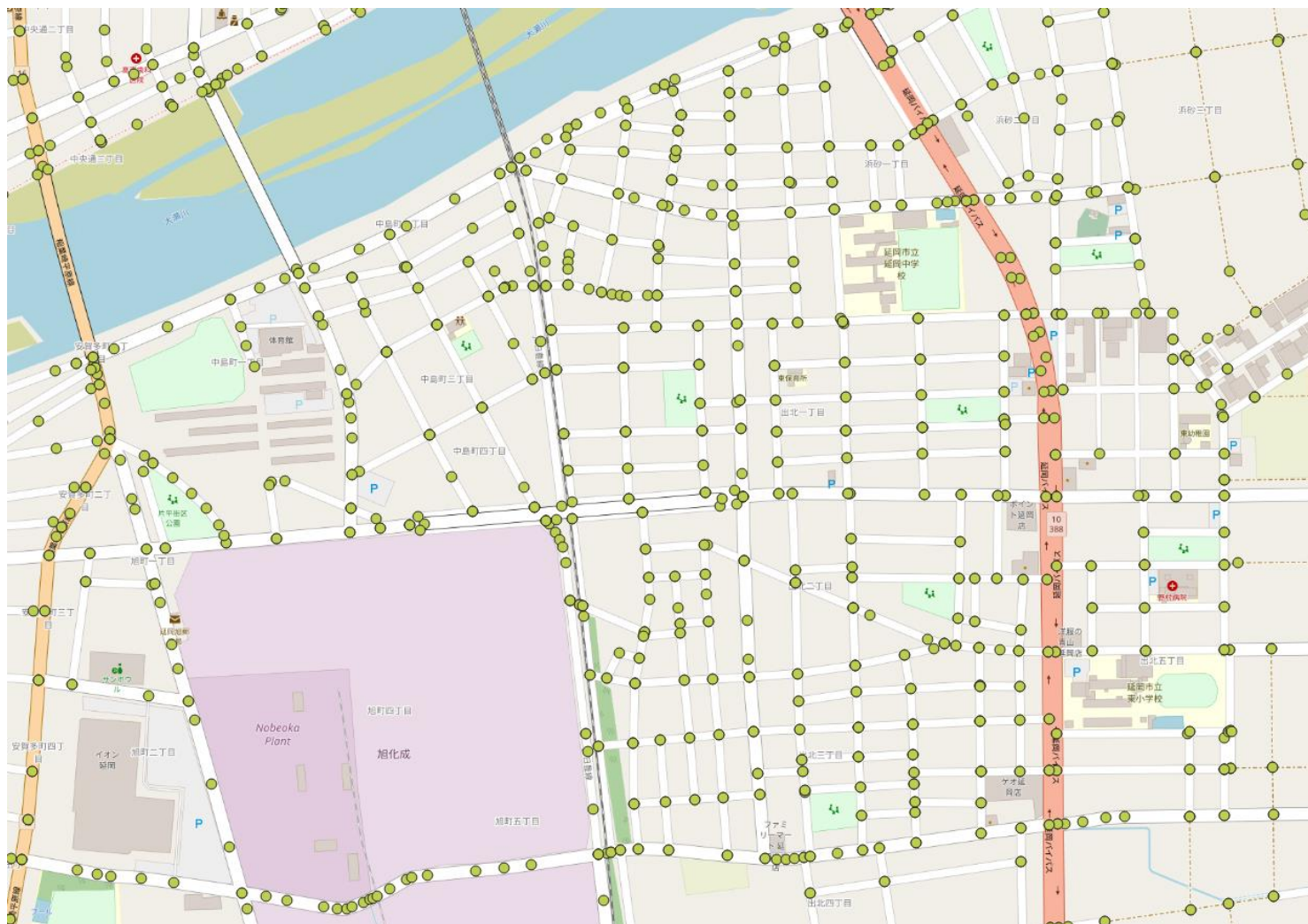
4 実験 ―地図データの用意

①OpenStreetMap上の地図データから道路上の位置点を抽出する



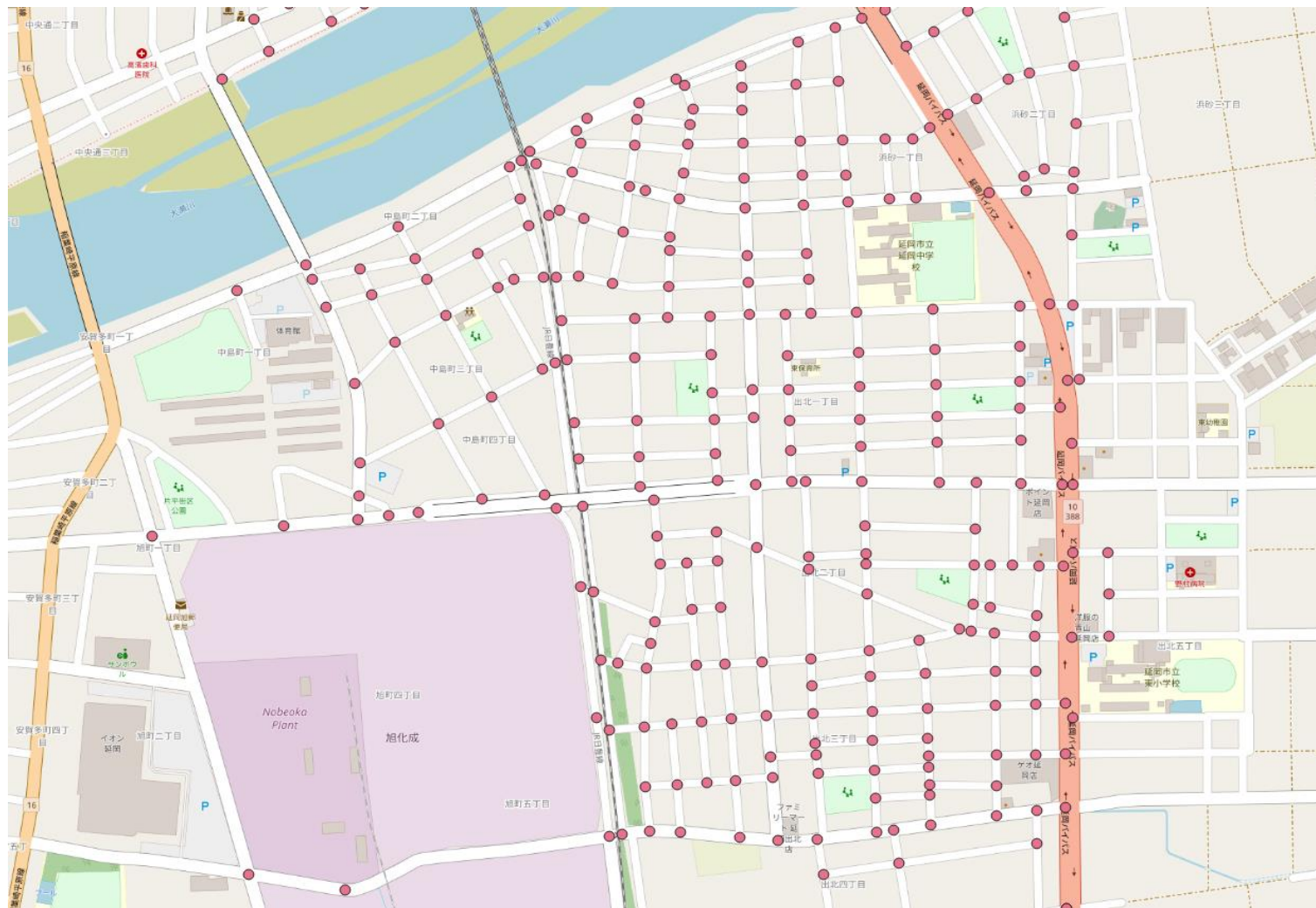
4 実験 ―地図データの用意

②highwayが共有するnodeのみを抜き出す



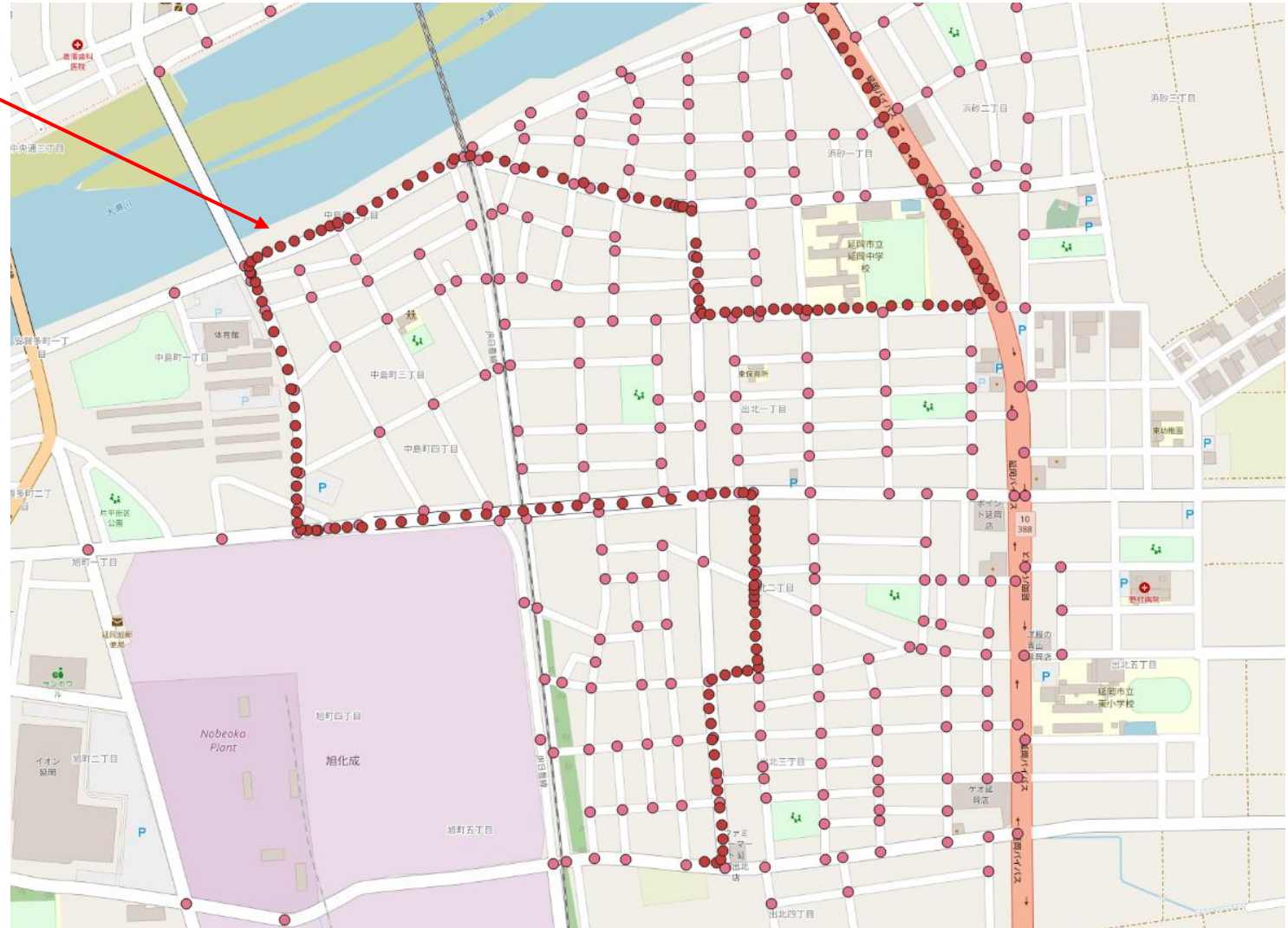
4 実験 ―地図データの用意

③地図上のnode

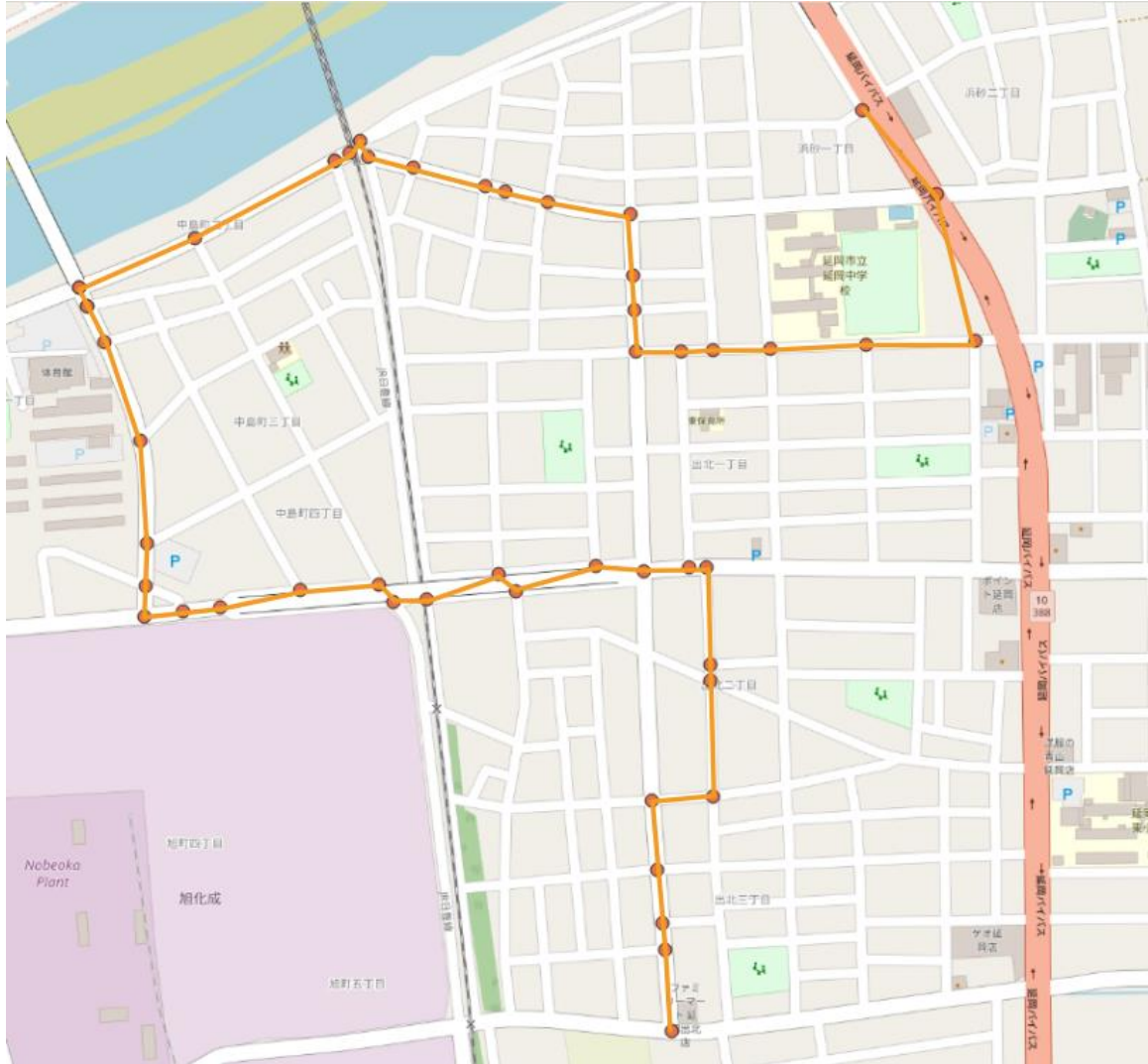


4 実験 —地図データの用意

移動履歷 $S = \langle (s_0, s_1), (s_1, s_2), \dots \dots, (s_{n-2}, s_{n-1}) \rangle$



4 實驗 — 照合結果



5 考察

- ・ より大きいデータでの処理時間を調べるため
縦10.431km, 横7.998km, ノード数2,841の地図データに対して照合実験を行った

移動距離 [km]	実行時間 [ms]
2.37	642.6
2.84	657.3
3.14	864.5
4.15	1222.1

更に地図データを飯塚市全域(213.96km²)(ノード数7,018)まで範囲を広げると

→ $O(mn^2)$ の為, およそ7,455.3ms程かかると予想される

5 考察

課題と改善点

- ・ ノードを全て抜き出している → $O(mn^2)$ なので多くの処理時間がかかる
- ・ 地図面積に応じてノードを除外すれば、処理時間を短縮できる

ご清聴ありがとうございました