

時変重みを持つ道路網上のモバイル k NN 問合せ

藤野 友輔[†] 上島 紳一[†]

[†] 関西大学総合情報学研究科 〒 569-1095 大阪府高槻市霊仙寺町 2-1-1

E-mail: †{k674609,ueshima}@kansai-u.ac.jp

あらまし 近年、空間ネットワーク上での空間探査を伴う問合せ処理の研究が進められている。これらの研究では、空間索引を用いて探査空間の刈込み手法を用いた効率的な問合せ処理方法が提案されている。しかし、通常、オブジェクトは移動し、ネットワークは時間に依存したトラフィックにも影響を受けるため、これらの空間索引を直接適用すると計算量が極端に増大し、現実的な時間では実行できない。このため、本稿では、モバイルオブジェクトを対象として、空間ネットワークを再構成して絞り込む問合せ処理法を提案する。提案手法では、各オブジェクトの移動範囲を設定して、オブジェクトまでの移動コストの上下限を予め算出し、質問点に対する近傍候補を絞り込む 2 種類のセルを生成して、時変ネットワーク上のオブジェクトに対して効率的な k NN 探査を可能にする手法を提案する。提案手法の評価として、道路ネットワークを摸して生成したネットワークへ適用して数値特性の確認を行ない、更に、数値地図を用いた現実の道路網での有効性を検証する。

キーワード k NN 探査, 移動体, 空間分割, 空間データベース, 時変重み付きエッジ, 数値地図, 道路

A Network Reconstruction Approach for Mobile k NN Queries on Time-dependent Road Networks

Yusuke FUJINO[†] and Shinichi UESHIMA[†]

[†] Graduate School of Informatics, Kansai University
2-1-1, Ryouzenji, Takatsuki, Osaka, 5691095 Japan

E-mail: †{k674609,ueshima}@kansai-u.ac.jp

Abstract Recently much attention has been focused on spatial query processing schemes over road networks. On static networks, the use of spatial indices, such as distance indices, regional adjacency indices based on space partitioning, provides efficient processing schemes that prune search space for spatial queries, such as k NN, Rk NN, etc. On the other hand, these indices are not directly applicable to practical road networks, where their traffics are time-dependent. To cope with this problem, a new scheme for spatial query processing is proposed for mobile objects by using network reconstruction. In our approach, assuming moving areas of mobile objects, upper/lower-bound moving costs to objects are calculated in their areas beforehand, and two types of cells are generated as neighborhoods of mobile objects. By use of this preprocessing, an efficient query processing algorithm is proposed against k NN over time-dependent networks. Numerical simulations are also provided for investigation of this scheme on both time-dependent road networks and digital road maps with simulated traffics as practical settings.

Key words k NN, mobile object, spatial partitioning, spatial database, time-dependent network, digital road map

1. はじめに

近年、オンラインマップサービスの進展や高性能な携帯型端末の普及により、位置情報を用いたサービスに注目が集まっている。この位置情報を用いたサービスの中で、最も需要が高い問合せは、道路網から病院やレストラン等のオブジェクトに対する k 近傍探査 (k NN) と、その目的地点への所要移動時間を

求める問合せである [1]。しかし k NN 探査処理に必要な道路網データは膨大であり、サーバーでの処理にて実行する端末数が増加するとサーバー自身への負荷も増加する。そこで空間ネットワーク上での空間探査を伴う問合せ処理の効率化に関する研究が進められてきた [2][3]。

これらの研究では空間ネットワーク上に存在するオブジェクトは固定されてモデル化されている場合が多く、モバイルオブ

ジェクトへの k NN 探査において適用出来ない．これはモバイルオブジェクトは位置が変化するものであり，これにより同一の質問地点であっても近傍が変化するためである．この例を図 1 へ示す．そこで，モバイルオブジェクトに対して空間索引を生成し k NN 探査を効率的に行なうための種々な手法が提案されている [4]．

本来現実世界でのネットワークでは，時刻によりトラフィックが変化するため，エッジが時変重みをもつ空間ネットワークである時間依存空間ネットワークとして考える必要がある．しかし時間依存空間ネットワークに対し単純に既存のアルゴリズムを適用させると，計算量が極端に増大し，現実的な時間では実行できない．

本稿では，現実の道路網上で問合せを想定し，時間依存空間ネットワーク上でモバイル k NN 問合せに焦点を絞る．Demiryurek らは時間依存ネットワーク上での k NN 探査手法 [5] を提案した．この Demiryurek らが提案した手法に対し，本提案手法を組込むことによりモバイルオブジェクトへの k NN 探査に対応させる．移動するオブジェクトまでの取り得る移動コストを予め算出し，その値を用い空間ネットワーク内の部分ネットワークの再構成を行なう．この再構成されたネットワークにより，近傍候補を絞り込む 2 種類のセルを空間索引として生成する．これにより時間依存空間ネットワーク上のモバイルオブジェクトに対し，生成された空間索引を用い効率的に k NN 探査を実行する手法を提案する．

更に道路ネットワークを摸して生成したネットワークへ提案手法を適用することにより数値特性の確認を行なう．また実際の道路網での有効性を確認するため，国土地理院発行の数値地図上にトラフィックをモデル化してネットワークを生成する．このネットワークに対し提案手法を適用し，定量的に評価する．更に，複数端末間の連携環境でのセルの生成時間と探査時間についても評価し，議論する．

2 章では時間依存ネットワークと対象となるモバイルオブジェクトの性質について述べる．3 章では部分ネットワークの再構成を述べ，4 章ではそれを用いた空間セル生成アルゴリズムについて述べる．5 章では時間依存空間ネットワーク上でモバイルオブジェクトへの k NN 問合せと，近傍地点までの経路探索アルゴリズムについて述べる．6 章と 7 章にて数値シミュレーション実験により手法の評価と検討を行う．

2. 準備

2.1 空間ネットワーク

[定義 1] 時間依存空間ネットワーク

道路網において，交差点をノード，道路片をエッジと見なし有向空間ネットワーク $G(V, E)$ (V : ノード集合, $E \subset V \times V$: エッジ集合) を考える．さらに，各エッジは時間に依存した，負でない重みを持つものとし，時間の関数により重みが定められる．

各ノード v は 2 次元での座標 $(v.x, v.y)$ を持つ．また，モバイルオブジェクト O_i と，それらの k 近傍を探索しているクエリ q の質問地点はこの G のノード上に存在している．

[定義 2] 時間に依存したエッジの重み

全てのエッジにおいて，時刻 t におけるエッジ e_i の重みを返す関数 $f(e_i, t)$ があり，各エッジはこの関数 f により，移動開始時刻に対応する移動時間が重みとして定められる．ここでは， $t = [0, 24]$ に対して， e_i の持つ移動時間の最大値を上限値として $e_{i,max}$ ，最小値を下限値として $e_{i,min}$ とする．これらは時間独立な 2 つの重みである．エッジの時間依存重みの遷移の例を図 2 に示す．

[定義 3] 上限移動時間と下限移動時間

G 上で出発地点を s ，到着地点を d とする時，各エッジの $e_{i,max}$ を用い s から d への合計移動コストが最小となる経路を通った際，その合計移動コストを上限移動時間 (UTT) とする．また各 $e_{i,min}$ を用い， s から d への合計移動コストが最小となる経路を通った際，その合計移動コストを下限移動時間 (LTT) とする．これら UTT, LTT を以下のように定義する．経路内の各移動コストが $e_{i,max}$ をとった際の s から d への合計移動コストが最小となるエッジ列をダイクストラ法にて求め， e_1, e_2, \dots, e_k とした時，

$$UTT(s, d) = \sum_{i=1}^k e_{i,max} \quad (1)$$

また，同じ様に経路内の各移動コストが $e_{i,min}$ をとった際の合計移動コストが最小となるエッジ列を e_1, e_2, \dots, e_k とした時，

$$LTT(s, d) = \sum_{i=1}^k e_{i,min} \quad (2)$$

となる． s から d までの合計移動コストが最小となる経路を通った際，その合計移動コストはどの時刻においても必ず $UTT(s, d)$ と $LTT(s, d)$ の間となる．この時， $e_{i,max}$ と $e_{i,min}$ は時間に対して独立であり，各ノードへの到着時刻に影響されない．

[定義 4] モバイルオブジェクト

モバイルオブジェクト O_i は自律的に移動し，移動先の状況に合わせて振舞いを変化させる．そこで， O_i が存在する位置は固定ではなく， G 内のノード集合 V 上を移動するものとする．この O_i が位置する可能性がある部分ネットワークを R_i と定義する．

R_i は予め定められており固定されているものとし， O_i は必ず R_i に含まれているノードのいずれかの上に存在するものとする．この時， O_i の位置 $(O_i.x, O_i.y)$ は他の O_i の位置に依存せず， R_i は重なり合ってもよい．これにより， R_i はそれぞれ独立に設定することが可能である．

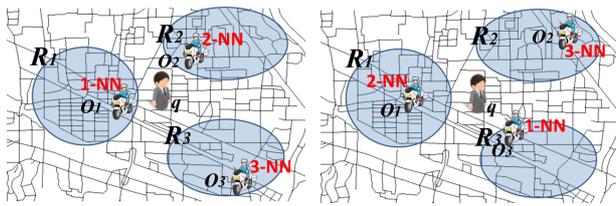
2.2 空間探索問題

[定義 5] 時間依存最小移動時間経路

エッジの重みが時間に依存する為， G では各ノードの出発時刻及び到着時間を考慮する必要がある．目的地までの移動コストは，各ノードへの到着時刻に応じた移動コストを用いて求めていく．

問合せの発行時刻を t_s とした時，出発地点 s から到着地点 d までの時間に依存する移動コスト $TT(s, d, t_s)$ は次の式の様に算出される．

$$TT(s, d, t_s) = \sum_{i=1}^k f(e_i, t_i) \quad (3)$$



(a) 12 時時点 (b) 14 時時点

図 1 近傍オブジェクトの変化

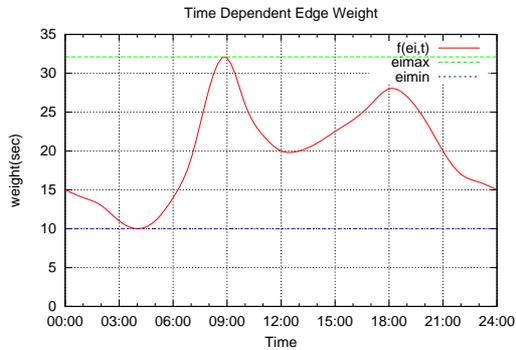


図 2 時間依存エッジ重み $f(e_i, t)$

$$t_1 = t_s, t_{i+1} = t_i + f(e_{i+1}, t_i). \quad (4)$$

ここでエッジ e_i は、到着時刻 t_i でのエッジの重みを用いて、合計移動コストが最小となる経路を辿るように展開される。これにより s から d まで展開されたエッジ列が時間依存最小移動時間経路となり、その際の合計移動コスト $TT(s, d, t_s)$ が時間依存な最短移動コストとなる。

[定義 6] 時間依存モバイルオブジェクト k NN 探索の問題

G において k NN 探索問題の結果は、 q の発行時刻に大きく影響される。これは、時刻により G 上のエッジの重みと、 O_i の位置関係が変化するためである。この q の発行時刻が結果に影響を与える例を図 1 に示す。

図 1 では、 q の発行地点が同じであっても出発時刻の違いにより近傍の O_i が変化している。これは、 O_i の位置の変化により q から O_i までの距離が変化することに加え、エッジの重みも変化しているためである。このため、もし q と O_i の位置が同じであっても、 q の発行時刻が違えば近傍の O_i が同一であるとは限らない。これらの要因より、モバイルオブジェクトが存在する時間依存ネットワークでは事前にオブジェクトまでの距離を求める方法は用いれない。

3. 部分ネットワークの再構成

ここでは、 G の部分ネットワーク R_i の再構成について述べる。

3.1 モバイルオブジェクトへの移動時間

G において、 R_i は固定である。これにより任意のノード p から O_i までの最小移動コストの上限値である WorstCost(WC) と下限値である BestCost(BC) を定義することが可能である。

つまり、 O_i が R_i 上のどのような位置にいても、 p から O_i への最小移動コストは必ずその WC と BC の間の値となる。任意の時刻 t_s での任意のノード p から O_i への WC 及び BC を次のように定義する。

$$WC(p, O_i) \leq TT(p, O_i, t_s) \leq BC(p, O_i). \quad (5)$$

そこで、 WC と BC を求め、 R_i の再構成をおこなうことにより、 R_i 内の O_i の位置推定を大規模計算に置き換える。次に述べる WC と BC の算出とそれを用いた部分ネットワークの再構成処理は各々の R_i 毎に独立しており、並列処理が可能である。

3.2 WC と BC の算出

WC は p から O_i までの最小移動コストが一番大きくなる状況の値であり、また BC は p から O_i までの最小移動コストが一番小さくなる状況の値である。 R_i に含まれる任意のノード r からオブジェクト O_i までの WC 及び BC は次のようになる。

$$\begin{aligned} BC(r, O_i) &= \min\{LTT(r, x)\} \\ WC(r, O_i) &= \max\{UTT(r, x)\} \end{aligned} \quad (6)$$

where $x \in R_i$.

また、 R_i の境界ノード集合を B_i とした時、任意の時刻 t_s での R_i に含まれない任意のノード n から O_i までの時間依存な最短移動コスト $TT(n, O_i, t_s)$ は次のように定義することができる。

$$\begin{aligned} TT(n, O_i, t_s) &= \min\{TT(n, x, t_s) \\ &\quad + TT(x, O_i, t_s + TT(n, x, t_s))\} \end{aligned} \quad (7)$$

where $x \in B_i$.

これにより時間に依存しない値である UTT 及び LTT を用いると n の BC 及び WC は次のように求めることができる。

$$\begin{aligned} BC(n, O_i) &= \min\{LTT(n, x) + BC(x, O_i)\} \\ WC(n, O_i) &= \min\{UTT(n, x) + WC(x, O_i)\} \end{aligned} \quad (8)$$

where $x \in B_i$.

これらにより、事前に R_i に含まれるの全てのノードに対して WC 及び BC を求めておく事によって G 上の全てのノードの WC 及び BC を容易に求めることが可能になることが解る。そこで、 R_i に含まれるノードの WC, BC を求めるべく R_i に含まれる任意のノード n_1, n_2 に対して G 上で全点对最短経路問題 [7] を解く。これにより求められた移動コストの最大値がそのノードから O_i への WC となる。 BC はそのノード上に O_i が存在する場合となるので、常に 0 となる。

3.3 再構成

WC と BC を用い R_i の再構成をおこない、 R_i の上位層となるネットワーク \hat{R}_i を生成する。初めに R_i に含まれるノード集合と、 R_i 内部から外部へ向かうエッジ集合を \hat{R}_i へと複写する。そして、 O_i に対応する仮のモバイルオブジェクト \hat{O}_i を \hat{R}_i 上の任意の位置に生成する。この \hat{O}_i から R_i に含まれる全てのノードに対して、有向エッジ e_i を生成する。この時、 e_i は宛先ノードの WC と BC の二つの値を重みとして持っている可変の重み付きエッジとする。

この例を図 3 に示す。図 3(a) は元となる R_i を示しており、図 3(b) では、その R_i を元に再構成をおこなった上位層ネットワーク \hat{R}_i を示している。

4. 二種類のセル生成

ここでは、 \hat{R}_i と式 (1) 式 (2) を用いて、 G 上でのモバイルオ

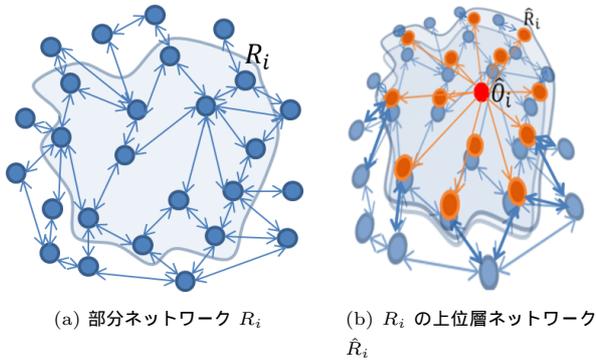


図3 部分ネットワークの再構成

プロジェクトへの k NN 問合せを効率化するために生成する二種類のセルについて述べる．

4.1 必ず一番近くなるセル

G 上にオブジェクト集合 $O = \{O_1, O_2, O_3, \dots\}$ が与えられた時, O_i が必ず一番近くなる空間セル $C(O_i)$ を

$$C(O_i) = \{V | WC(V, O_i) \leq BC(V, O_j), i \neq j\} \quad (9)$$

と定義する．

$C(O_i)$ は, O_i に対応する仮のモバイルオブジェクト \hat{O}_i から並列ダイクストラ法 [6] を展開することにより生成される．この際に生成対象となる \hat{O}_i からの展開時には, WC または UTT を用い展開し, 残りの他の \hat{O}_i からの展開には BC または LTT を用い展開する．この時, 上位層ネットワーク \hat{R}_i のノードと, その直下に存在する下位層ネットワークのノードは展開コストを共有するものとする．これにより生成対象オブジェクトの \hat{O}_i から展開された領域が $C(O_i)$ となる．

この処理を各モバイルオブジェクトに対し繰り返し, 必ず一番近くなるセル $C(O_i)$ を生成する． $C(O_i)$ は, O_i が任意の時刻で 1-NN となるノードの集合である． $C(O_1), C(O_2), \dots, C(O_n)$ は互いに排他的である．

5 つのモバイルオブジェクトに対して $C(O_i)$ の生成を行った例を図 4(a) に示す．

q の質問地点がこの $C(O_i)$ 内に位置する場合は, 常に所属する O_i が 1-NN となる．この場合には, 高コストである時間依存最小移動時間経路の計算を省略し, 1-NN を求めることが可能となる．

4.2 一番近くなる可能性があるセル

G 上にオブジェクト集合 $O = \{O_1, O_2, O_3, \dots\}$ が与えられた時, O_i が一番近くなる可能性がある空間セル $\hat{C}(O_i)$ を

$$\hat{C}(O_i) = \{V | BC(V, O_i) \leq WC(V, O_j), i \neq j\} \quad (10)$$

と定義する．

$\hat{C}(O_i)$ は, O_i に対応する仮のモバイルオブジェクト \hat{O}_i から並列ダイクストラ法を展開することにより生成される．この際に生成対象となる \hat{O}_i からの展開時には, BC または LTT を用い展開し, 残りの他の \hat{O}_i からの展開には WC または UTT を用い展開する．この時, 上位層ネットワーク \hat{R}_i のノードと, その直下に存在する下位層ネットワークのノードは展開コストを共有するものとする．これにより生成対象オブジェクトの \hat{O}_i から展開された領域が $\hat{C}(O_i)$ となる．

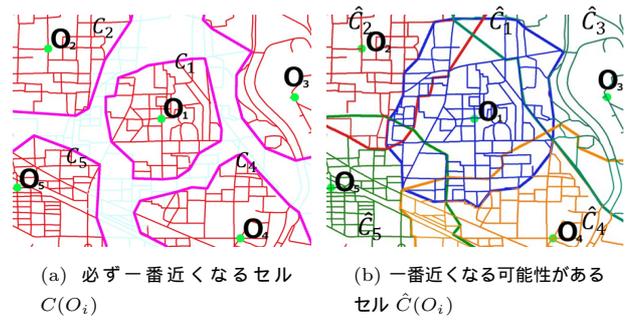


図4 二種類のセル

この処理を各モバイルオブジェクトに対し繰り返し, 一番近くなる可能性があるセル $\hat{C}(O_i)$ を生成する．これは, $C(O_i)$ 生成時の WC と BC , UTT と LTT を入れ替えたものであると言える． $\hat{C}(O_i)$ は, O_i が任意の時刻で 1-NN となる可能性があるノードの集合である．言い換えれば, $\hat{C}(O_i)$ 外のいかなるノードも 1-NN として O_i を持つ可能性はない． $\hat{C}(O_1), \hat{C}(O_2), \dots, \hat{C}(O_n)$ はそれぞれ他の一つ以上の $\hat{C}(O_i)$ と重なりあう．

5 つのモバイルオブジェクトに対して $\hat{C}(O_i)$ の生成を行った例を図 4(b) に示す．

q の質問地点が複数の $\hat{C}(O_i)$ 内に含まれている場合は, それら複数の $\hat{C}(O_i)$ に属しているオブジェクト集合に対して, 出発時刻に依存した移動時間を求める必要がある．

5. 時間依存モバイル k NN 問合せ処理

5.1 時間依存 A* アルゴリズム

時間依存モバイル k NN 問合せ処理では, 時間依存最小移動時間経路とその移動コストを求める為に, A* アルゴリズムを拡張し用いる． G における時間依存最小移動時間経路は, 各ノードへの到着時刻でのエッジの重みを用いて展開していくことにより求めることができる．しかし, 大規模な G 上では全てのノードを展開するのは現実的では無く, 計算領域を刈り込む必要がある．そこで, A* アルゴリズムのヒューリスティックな関数を使うことにより, アクセスする必要があるノードの刈りこみを行う．

エッジの長さがコストと考えられる通常の空間ネットワークでは, ユークリッド距離をヒューリスティック値として使うことができる．しかし, G では, 全ての可能な出発時刻において, 決して実際の値が下回らない値をヒューリスティック値として選択することが必要である．この単純な値はユークリッド距離を全領域内の最大上限速度で割ったものである．しかし, この方法では刈り込み量は微量となり, あまり効果を望めない．そこで, オブジェクトへの時間依存最小移動時間経路は必ずそのオブジェクトの属する $\hat{C}(O_i)$ から出ない性質を利用する．

これにより分母を $\hat{C}(O_i)$ 内のみでの最大速度とした値をヒューリスティック値とすることができ, ノードの展開範囲を大幅に刈り込みことができる．質問地点が宛先オブジェクトの $\hat{C}(O_i)$ 内に位置しない場合においては, 宛先オブジェクトが含まれる $\hat{C}(O_i)$ まで, 隣接する $\hat{C}(O_i)$ を展開していく．これらの展開された $\hat{C}(O_i)$ 内での最大速度を分母とした値をヒューリ

スティック値とし、出発時刻に依存した時間依存最小移動時間経路の探査とその移動コストの計算を行う。

5.2 時間依存モバイル k NN 問合せ処理

時間依存モバイル k NN 問合せ処理では、初めに生成した 2 種類のセル $C(O_i)$, $\hat{C}(O_i)$ を用い、 $1-NN$ を見つけ、そこから、 $k-1$ 個の近傍を探すように $\hat{C}(O_i)$ を展開してゆく。

まず初めに、 q の質問地点の $1-NN$ を識別する為に、 $C(O_i)$ または $\hat{C}(O_i)$ を使用する。 q の位置が与えられると、初めにそのノードが位置している $C(O_i)$ を探す。 $C(O_i)$ 内に位置しているのであれば、その $C(O_i)$ に属している O_i が、この q の $1-NN$ となる。どの $C(O_i)$ にも位置していなければ、位置している複数の $\hat{C}(O_i)$ から $1-NN$ を探すことになる。従い、この場合では、 $1-NN$ を決定するために、 q が位置する $\hat{C}(O_i)$ に属している O_i に対しクエリ発行時点での位置を問合せ、それらに対して時間依存な最小移動コストを求める。この際に前述の時間依存 A^* アルゴリズムを用いる。これにより、求めた移動時間が最小となるオブジェクトが $1-NN$ となる。

残りの NN を見つけるには、隣接している $\hat{C}(O_i)$ を展開していくことにより行なう。残りの NN は、既に確定した NN が属している $\hat{C}(O_i)$ に隣接している $\hat{C}(O_i)$ に存在することが保証されている。そこで、隣接している $\hat{C}(O_i)$ を展開し、属している各々の O_i のクエリ発行時点での位置を問合せ。 q からそれらの地点に対して出発時刻に依存した最小移動コストを求め、この移動コストが一番小さい O_i が次の NN となる。これを k 個満たすまで繰り返すことにより時間依存モバイル k NN 問合せ処理を行なう。

このように、 $C(O_i)$ と $\hat{C}(O_i)$ を用い NN の候補を絞ることにより、 O_i の位置問合せと、出発時刻に依存した最小移動コストを求める回数を減らし、効率的に NN を見つけることができる。これにより、計算量とアクセス数の削減を行うことが可能となる。

6. 生成データに対する数値シミュレーション評価

6.1 対象データ

プログラムにより道路地図を摸して自動的に生成したネットワークを用い、本稿で述べた G におけるモバイル k NN 探索と時間依存最小移動時間経路探索の性能評価と検討を行う。

対象データとして、モバイルオブジェクト数を固定しノード数を変化させたネットワークと、ノード数を固定しモバイルオブジェクト数を変化させたネットワークを生成する。この際にモバイルオブジェクトの移動範囲は、ランダムに決定された出発地点から最短移動経路に沿って移動した際に 1.2km 以内であるノードを移動範囲としている。

また各ネットワークは、エッジの重みが固定の通常ネットワークと、エッジの重みが時間依存な時間依存ネットワークの二種類を生成する。それらにネットワークに対し提案手法を適用し、特性の確認を行う。

通常ネットワークではエッジの距離をエッジの重みとする。時間依存ネットワークでは、距離を 30~50 の間のランダムな速度で割った値を重みとして 5 つ作成し、その重みを 24 時間

	ノード数が変化	オブジェクト数が変化
ノード数	500~1500 (500 単位で増加)	7500
エッジ数	ノード数 \times 2	15000
オブジェクト数	7	2~60 (2 単位で増加)
エッジの重み	距離または距離/30~50 のランダムな速度	
オブジェクト移動範囲	片道 1.2km 圏内	

表 1 生成ネットワークの詳細

周期になるよう繋ぎ合わせる事により、時間に依存したエッジの重みとする。この生成データの詳細を表 1 に示す。

本実験では、FX-8150@3600MHz の 8 コア CPU、16GB メインメモリといったスペックの PC 上にてプログラムを実行し、データの測定を行う。また分散処理評価の際には、CPU の 1 コアを一つの端末と見立て処理を分散させ評価を取る。本稿で述べたアルゴリズムを実装したプログラムは C++ 言語にて作成したネイティブプログラムとなっており、分散処理可能な部分のみがマルチスレッドとなっている。

6.2 評価方法

ここでは、自動的に生成された通常道路ネットワークと、時間依存ネットワークの二種類のネットワーク上にてセルの生成と k NN 探査の評価を行う。

セルの生成に関しては、部分ネットワークの再構成とそれを用いた二種類のセル生成を合わせた実行時間をセルの生成速度として評価する。また、処理の分散の効果についても評価を行う。セルの生成時間評価では、10 回のセル生成を繰り返し、その実行時間の中央値を評価する。また、処理の分散の評価では同じ環境にて、CPU の使用コア数を変更し評価する。

k NN 探索に関しては、質問地点となるノードをランダムに選出し、その地点から $1-NN, 3-NN, 5-NN$ の探索と、それら NN までの最小移動時間経路の探索を行なう。この時、時間依存ネットワーク上での探索開始時刻はランダムな時刻を設定する。また、モバイルオブジェクトの位置は、クエリ発行時に毎回移動範囲内のノードをランダムに選出し決定するものとする。これらの処理を 30 回行い、その実行時間の中央値を評価する。

6.3 データ特性によるセルの生成速度への影響

データの特性によるセルの生成速度への影響を評価する。通常ネットワーク並びに時間依存ネットワーク上での、ノード数に対するセルの生成時間の変化を図 5 へ示す。また、モバイルオブジェクト数に対するセルの生成時間の変化を図 6 へ示す。

図 5 の結果からセルの生成コストは、ノード数の増加に伴い指数関数的に増加することが見て取れる。これはノード数の増加に伴いセル生成時に行うパラレルダイクストラ処理の展開数が指数関数的に増加するためである。また図 6 の結果からセルの生成コストはモバイルオブジェクト数に比例して増加することが見て取れる。これは、モバイルオブジェクトの増加に伴い、部分ネットワークの再構成と、セル生成時のパラレルダイクストラ処理を行う回数が比例して増加するためである。これら二つの結果から、時間依存ネットワークと通常ネットワークはほぼ同コストにてセル生成処理が可能なが見て取れる。これは、部分ネットワークの再構成及びセルの生成は時間依存ネットワークであっても、時間に依存しない重みである UTT と LTT を用い処理するためである。また、これらの結果から

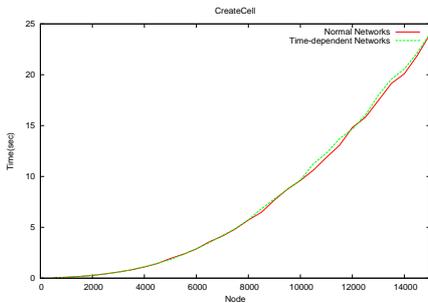


図 5 生成ネットワークにおける
ノード数に対するセルの生成時間

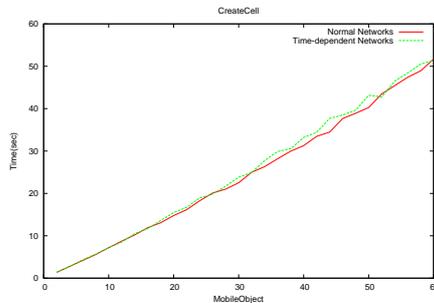


図 6 生成ネットワークにおける
オブジェクト数に対するセルの生成時間

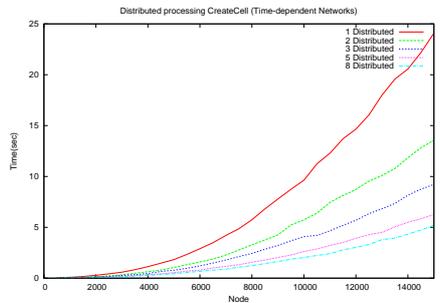


図 7 時間依存ネットワークにおける
ノード数に対するセルの分散生成時間

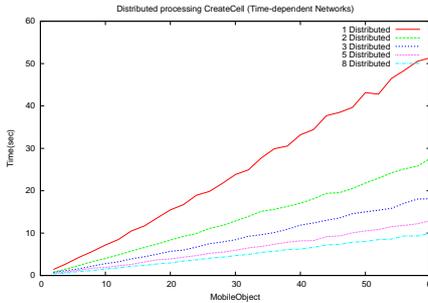


図 8 時間依存ネットワークにおける
オブジェクト数に対するセルの分散生成時間

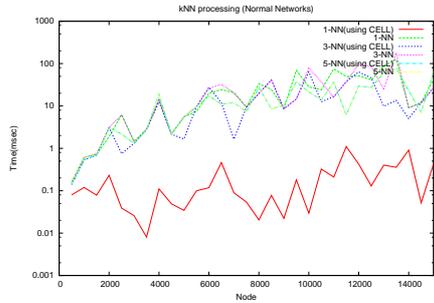


図 9 通常ネットワークにおける
ノード数に対する k NN 探査時間

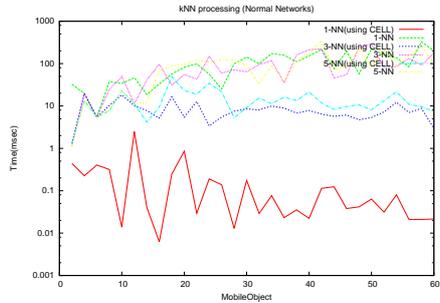


図 10 通常ネットワークにおける
オブジェクト数に対する k NN 探査時間

セルの生成コストは非常に大きいことが見て取れる。しかし、実行はセル作成時の一度のみであるため、運用時への影響は小さい。また、ノード情報更新時のセル再構成に関しては、比較的小さいコストにて再構成可能である。これは、更新があったノードに接続されているノードが所属しているセルと、それに隣接しているセルに対してのみ再構成の処理を行えばよいためである。

6.4 分散処理によるセルの生成速度への影響

部分ネットワークの再構成及び二種類のセル生成は、それぞれの領域が独立して生成することが可能であり、処理を分散させることが可能である。そこで、セルの生成処理を分散させることによる、セルの生成速度への影響を評価する。

時間依存ネットワーク上でのノード数に対する分散数 1,2,3,5,8 での実行速度の変化を図 7 へ、モバイルオブジェクト数に対する実行速度の変化を図 8 へ示す。今回通常ネットワークと時間依存ネットワークでの結果はほぼ同位であったため、通常ネットワークでの結果は省略する。これらの結果から、ノード数の増加やモバイルオブジェクト数の増加に拘わらず処理の分散化が有用であることが確認できる。これはモバイルオブジェクト一つの部分ネットワークの再構成及びセル生成の処理が完全に独立して処理可能なためである。

しかし処理コストが元から小さい状況では、分散処理の有効性は僅かしかみられない。これは、分散により削減される処理コストと、分散処理を行う際の処理コストの差が小さいためであると考えられる。そのため、端末間で分散処理を行うための処理コストが高いような状況であれば、分散処理を行うことで処理コストが余計にかかってしまう可能性もある。これらの理由から、セルの生成コストが大きいであろうネットワークでのセル生成の際は分散処理を行い、処理コストが僅かな場合は分

散処理をさせないようにすることが望ましいと考えられる。

6.5 データ特性による k NN 探査速度への影響

生成されたセルを用い k NN 探査を行なった場合と、セルを用いず時間依存 A*探査のみを用いて k NN 探査を行なった場合との処理時間の比較を行う。この比較により、セルの利用による実行速度への影響の検証を行う。

通常ネットワーク上において 1-NN,3-NN,5NN を求めた際のノード数に対する実行時間の結果を図 9 へ、モバイルオブジェクト数に対する実行時間の結果を図 10 へ示す。また時間依存ネットワーク上にて 1-NN,3-NN,5NN を求めた際のノード数に対する実行時間の結果を図 11 へ、モバイルオブジェクト数に対する実行時間の結果を 12 へ示す。

この結果から、セルを用いた場合は、セルを用いない場合に対し、全体的に低い処理コストにて、 k NN 探査を行える事が確認できた。またセルを用いない場合では、求める NN の数に拘わらずノード数やモバイルオブジェクト数の増加に合わせ処理時間も増加しているが、セルを用いる場合では求める NN の数により変化することが見て取れる。これは、セルを用いない場合は全てのモバイルオブジェクトに対して最小移動コストを求める必要があるのに対し、セルを用いる場合は NN の候補を絞ることができるためである。特に 1-NN を求める際は、候補となるオブジェクトが少ない傾向にあるため、際立って低い処理コストにて実行できている。また、これらの結果からノードの増加時よりもモバイルオブジェクト数の増加時の方がより効果を発揮していることが確認できる。これは、モバイルオブジェクト数が増えるほど全体に対する NN 候補となるオブジェクト数の割合が小さくなるためである。

通常ネットワークに比べ、時間依存ネットワークでは時変エッジ重みの影響により各々の \hat{C}_i が大きくなる。このため重なり

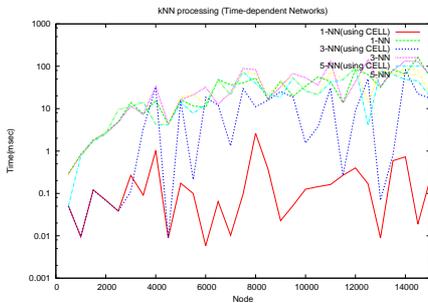


図 11 時間依存ネットワークにおける
ノード数に対する k NN 探索時間

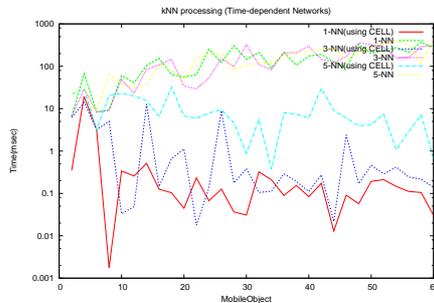


図 12 時間依存ネットワークにおける
オブジェクト数に対する k NN 探索時間

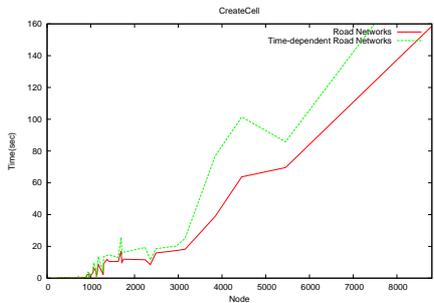


図 13 道路ネットワークにおける
ノード数に対するセルの生成時間

幅員	最大平均移動速度	最小平均移動速度
25.0m 以上	60km/h	50km/h
13.0m 以上 25.0 未満	60km/h	45km/h
5.5m 以上 13.0m 未満	50km/h	30km/h
3.0m 以上 5.5m 未満	40km/h	15km/h

表 2 幅員による平均移動速度

合うセルが多くなり候補となる NN が多くなる．その影響により時間依存ネットワークでは通常ネットワークに比べ、処理コストの削減比率が小さくなっていることも確認できる．

7. 数値地図に対する数値シミュレーション評価

7.1 対象データと評価方法

実際の数値地図データを用い、本稿で述べた G におけるモバイル k NN 探索と最小移動時間経路探索の性能評価と検討を行う．対象データとして、国土道路地理院にて刊行されている「数値地図 25000 大阪」と「数値地図 25000 京都」に含まれている道路網データ [8] のうち特徴的な 49 個のデータを用い、比較を行う．この道路網データの交差点をノード、道路片をエッジとみなし利用する．

巡回中の警察官への k NN 探索を想定し、警察官をモバイルオブジェクトとする．モバイルオブジェクトは道路網データに含まれる一つの警察関連施設につき一つ作成するものとする．またモバイルオブジェクトの移動範囲は実際の大阪府警察の管轄範囲 [9] を参考に、警察関連施設を出発地点とし最短移動経路に沿って移動した際に 1.2km 以内であるノードを移動範囲とした．

本章では、エッジの重みが固定である通常の道路ネットワークと、エッジの重みが時間に依存する時間依存道路ネットワークの二種類のネットワーク上にてセルの生成と k NN 探索の評価を行う．通常の道路ネットワークではエッジの距離を重みとして使用し、時間依存道路ネットワークでは次の時間に依存するエッジの重みを使用する．今回使用する数値地図データには道路片の交雑状況は入っていないため、実際の道路状況を想定し、道路片の幅員データから 4 パターンの最大平均移動速度と最小平均移動速度を設定する．幅員が大きい道路片は、制限速度が高いことから、最大平均移動速度を大きくした．また、幅員が大きいほど混雑時の影響が小さいと考えられるため、最大平均移動速度と最小平均移動速度の差を小さくした．幅員が小さい道路片は、制限速度が低いことから、最大平均移動速度を小さくした．また、幅員が小さいほど混雑時の影響が大きいと考えられるため、最大平均移動速度と最小平均移動速度の差を

大きくした．これらの規則に則り作成した 4 パターンの最大平均移動速度と最小平均移動速度の組み合わせを表 2 に示す．

これを基にしたランダムな速度でエッジの距離の割った値を重みとして 5 つ作成しする．その重みを 24 時間周期になるよう繋ぎ合わせる事により、時間に依存したエッジの重みとする．その他評価方法及び実験環境は前述のものと同じである．

7.2 データ特性によるセルの生成速度への影響

実道路網データの特性によるセルの生成速度への影響を評価する．通常道路ネットワーク並びに時間依存道路ネットワーク上でのノード数に対するセルの生成時間の変化を図 13 へ示す．また、モバイルオブジェクト数に対するセルの生成時間の変化を図 14 へ示す．

これら結果からセルの生成コストは、基本的にノード数またはモバイルオブジェクト数のどちらか一方でも大きい時に増加することが見て取れる．しかし、必ずしもノード数やモバイルオブジェクト数の増加に伴い生成コストが大きくなるとは言えない．これは、実道路網ではノード数が大きくともモバイルオブジェクト数が小さかったり、モバイルオブジェクト数が大きくともノード数が小さかったりすることが多々あるためである．実際に図 13, 図 14 では、生成ネットワークへの適用時の結果である図 5, 図 6 と異なり、激しい起伏が多々見られる．

7.3 分散処理によるセルの生成速度への影響

本提案手法は移動範囲の被覆の許容など、実世界での利用を想定し設計している．これにより実道路網への適用時においても処理を分散させることが可能である．そこで、セルの生成処理を分散させることによる、実行時間への影響を評価する．

時間依存道路ネットワーク上でのノード数に対する分散数 1,2,3,5,8 の実行速度の変化を図 15 へ、モバイルオブジェクト数に対する実行速度の変化を図 16 へ示す．

これらの結果から、実道路網への適用時においても、処理の分散化により確実に生成時間を削減できていることが確認できる．また紙面の都合上図を割愛したが、生成ネットワークへの適用時と異なり、時間依存道路ネットワークでは通常道路ネットワークに比べ全体の処理コストが増加していることが見て取れた．これは実道路網では、生成ネットワークに比べエッジの繋がりが複雑なため、時間依存ネットワークにおいて細かい識別処理が多く入るためである．

7.4 データ特性による k NN 探索速度への影響

実道路網において、生成されたセルを用い k NN 探索を行なっ

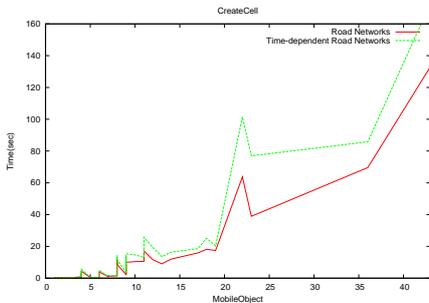


図 14 道路ネットワークにおける
オブジェクト数に対するセルの生成時間

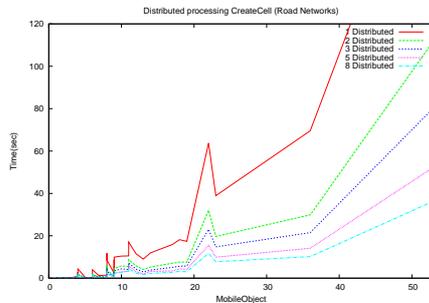


図 15 通常道路ネットワークにおける
オブジェクト数に対するセルの分散生成時間

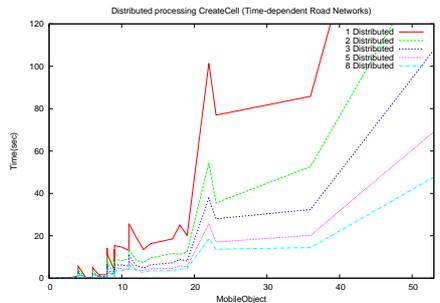


図 16 時間依存道路ネットワークにおける
オブジェクト数に対するセルの分散生成時間

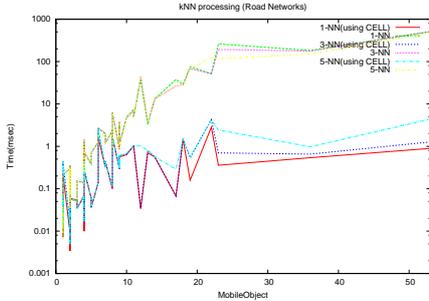


図 17 通常道路ネットワークにおける
オブジェクト数に対する k NN 探索時間

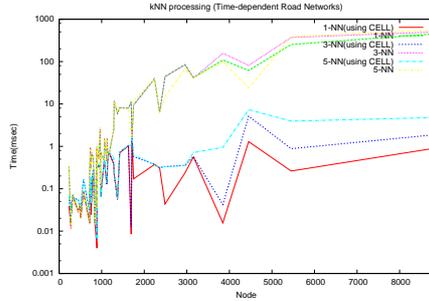


図 18 時間依存道路ネットワークにおける
ノード数に対する k NN 探索時間

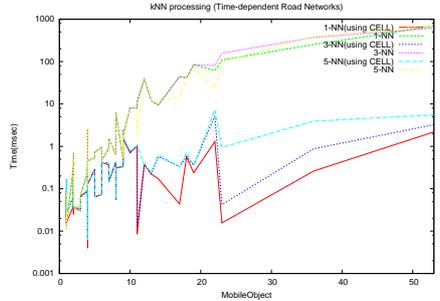


図 19 時間依存道路ネットワークにおける
オブジェクト数に対する k NN 探索時間

た場合と、セルを用いず時間依存 A*探索のみを用いて k NN 探索を行った場合との処理時間の比較を行う。この比較により、セルの利用による実行速度への影響の検証を行う。

通常道路ネットワーク上において 1-NN, 3-NN, 5NN を求めた際のモバイルオブジェクト数に対する実行時間の結果を図 17 へ示す。また、時間依存道路ネットワーク上にて 1-NN, 3-NN, 5NN を求めた際のノード数に対する実行時間の結果を図 18 へ、モバイルオブジェクト数に対する実行時間の結果を図 19 へ示す。

この結果から、実道路網データへの適用時においても、セルを用いることの有効性が確認できた。更に両者のネットワークにおいて生成ネットワークへの適用時よりも処理の削減比率が大きい。これは、実道路網は生成ネットワークに比べエッジの複雑性が増し、それにより目的地への移動コストの計算量が増加しているためである。これによりセルを用いた NN 候補の絞り込みが大幅な処理コストの削減に繋がっている。

8. おわりに

本稿では、時変重みを持つ道路網上でのモバイル k NN 問合せ処理手法の提案と評価を行なった。評価では生成したネットワークへの適用にて本提案手法の特性を確認し、実際の道路網データへの適用により有用性を確認した。これにより、提案手法の有効性が見て取れた。

提案手法により生成されたセルを用いることで、 k NN 探索の実行時間を大幅に削減することが可能である。セルの生成は非常に処理コストがかかることが確認された。しかし、問合せ回数が多いような状況では、セルを利用することで探索処理コストの削減量がセル生成コストを上回り有用である。また、セルの生成処理を分散させることにより、生成時間が大幅に削減されることが確認できている。

今回ネットワークとして道路網データを用い実験を行ったが、ノードと正の重みをもつエッジで構成されたネットワークであれば、本提案手法を適用することが可能である。道路ネットワークに限らず様々な分野のネットワーク上にて本手法を適用することにより、新たな可能性を見出すことができる。

文 献

- [1] Zhao, G. Xuan, K. and Taniar, D.: "Path k NN query processing in Mobile Systems", IEEE Transactions on Industrial Electronics, Vol.99, pp.1-8 (2011).
- [2] Zhao, G. Xuan, K. Rahayu, W. Taniar, D. Safar, M. Gavrilova, M.L. Srinivasan, B.: "Voronoi-Based Continuous k Nearest Neighbor Search in Mobile Navigation", IEEE Transactions on Industrial Electronics, Vol.58, pp.2247-2257 (2011).
- [3] Kambara, T. Tamaki, K. Muranaka, T. Ueshima, S.: "Spatial Queries Processing in Autonomous Mobile System Environment" IEEE 17th International Conference on Parallel and Distributed Systems(ICPADS), pp.770-775 (2011)
- [4] Li, G. Fan, P. Li, Y.: "An Efficient Technique for Continuous K-Nearest Neighbor Query Processing on Moving Objects in a Road Network" IEEE 10th International Conference on Computer and Information Technology(CIT 2010), pp.627-634 (2010)
- [5] Demiryurek, U. Banaei-Kashani, F. and Shahabi, C.: "Efficient K-Nearest Neighbor Search in Time-Dependent Spatial Networks", Proc. 21st International Conference on Database and Expert Systems Applications(DEXA2010), pp.432-449 (2010)
- [6] Graf, M. and Winter, S.: "Network Voronoi Diagrams", Proc. Österreichische Zeitschrift für Vermessung und Geoinformation, Vol.91 No.3, pp.166-176 (2003).
- [7] Cormen, Thomas H. Leiserson, Charles E. Rivest, Ronald L.: "Introduction to Algorithms (first edition ed.)", MIT Press and McGraw-Hill, pp.558-565, pp.570-576 (1990)
- [8] 国土地理院, 数値地図(空間データ基盤), <http://www.gsi.go.jp/>
- [9] 大阪府警察, 管轄区域について, <http://www.police.pref.osaka.jp/>