災害時の被害推定のための時空間類似シナリオ検索手法

林 秀樹[1,a)]　淺原 彰規[1]　菅谷 奈津子[2]　小川 祐一[2]　冨田 仁志[3]

概要：本報告では，被害状況把握に必要な災害状況を推定するため，事前に災害シミュレーション結果の時間的推移（時系列グリッドデータ）などのデータ群（時空間シナリオ）を蓄積したデータベースから，災害発生時に得られる観測データに類似する時空間シナリオを高速検索する時空間類似シナリオ検索手法を提案した．提案手法は，時空間インデックスを用いて，時空間類似シナリオ検索における時空間交差判定を効率的に行うことで，処理時間の短縮を図った．性能評価の結果から，提案手法は，時間属性のみまたは空間属性のみのインデックスを用いる従来手法と比べ，高速に処理できることを確認した．また，提案手法は，100 件の観測データと 500 億件規模の時系列グリッドデータとの時空間類似シナリオ検索を 30 秒程度で処理できることを確認し，実用への目標と考える 10 分以内で検索処理を完了できる見通しを得た．

1. はじめに

日本は，地震，火山活動が活発な環太平洋変動帯に位置し，地震の発生回数や活火山の分布数の割合は極めて高いものとなっている [1]．また，地理的，地形的，気象的諸条件から，台風，豪雨，豪雪等の自然災害が発生しやすい国土となっている．日本では，自然災害により多くの人命や財産が失われている．最近の大規模災害としては，2011 年 3 月の東日本大震災 [2] ，2014 年 8 月の広島市土砂災害 [3]，2014 年 9 月の御嶽山噴火 [4] が発生し，いずれの災害も甚大な被害をもたらした．一方で，南関東地域で M7 クラスの地震が発生する確率は 30 年間で 70%と推定されている [5]．首都直下地震が発生すると，建物倒壊や市街地延焼火災などにより，人的・物的被害が甚大となることが想定されている [6]．大規模災害からの人命や財産の保護は，日本の最重要課題である．

大規模災害が発生すると，救助・救急，医療など多岐にわたる応急活動を効果的に実施するため，国や地方公共団体は，被害情報の収集及び通信の確保を迅速に行う．国は都道府県に，都道府県は市区町村などに被害状況の提出を依頼し，依頼元に電子ファイルや FAX などで被害情報を提出する [7]．このように人手で被害情報を集約するため，大規模災害の被害状況を把握するには数日を要す．応急活動を効果的に実施するためには，被害の規模感を迅速に把握することが必要であり，被害状況を把握する時間を短縮することが課題と考える．

この課題に対し，大規模災害発生時には，電力網や通信網などライフラインが被害を受け，防災関係機関は断片的な情報しか得られない状況を想定し，その断片的な情報から被害推定を行うことが重要であると考える．特に，防災関係機関が，救助部隊の派遣先や救助活動の範囲など具体的な方針を立てる際，災害の状況を把握することは有用と考える．近年の災害シミュレーション技術の発展により，例えば，津波 [20]，火災延焼 [21]，豪雨 [8] などの災害の状況を事前に把握可能になってきた．

ここで，大規模災害の発生直後に，災害の規模感を把握するために，災害シミュレーションを実行することを考える．シミュレーションを実行するためには，入力条件が必要となるが，大規模災害発生時に入力条件を収集することが困難な場合がある．例えば，都市型大地震の発生後に火災状況を把握するため，火災延焼シミュレーションを実行する場合，出火点，風向，風速などが入力条件になるが，出火点の情報を集めることは困難と考える．一方，シミュレーションの入力条件がわかったとしても，シミュレーションが完了するまでに時間を要する場合がある．例えば，津波シミュレーションの場合，複雑な物理計算を含む津波伝播の数値計算を行う必要があり，一般に結果を得るまでに長い時間を要する．このような場合，災害の状況を把握するため，大規模災害の発生直後に災害シミュレーションを実行するのとは別の手段を考える必要がある．その手段として，災害発生前に様々な入力条件でシミュレーションを行い，各入力条件から得られた災害の時間的推移のデータ群（時空間シナリオ）をデータベースに蓄積し，災害時には，センサが観測した断片的な災害状況に類似する時空間シナリオを検索する手段が考えられる．類似する時空間シナリオに基づき，センサが存在しない地域やセンサが存在するが災害状況を収集できない地域の災害状況を推定する，また今後の災害状況の時間的推移を予測することが可能と考える．本報告では，センサの断片的な観測データに類似する時空間シナリオを検索することを時空間類似シナリオ検索と呼ぶ．

ここで，時空間類似シナリオ検索の課題について述べたい．まず，災害シミュレーションの結果は，空間が格子で分割され，各格子に物理量が関連付くグリッド形式の分布データとして一般に表現される．また，時間的推移を含むため，時系列データとなる．災害シミュレーションの結果は，時間的な粒度，時間幅，空間的な粒度（グリッドサイズ），空間範囲に依存するが，一般に大規模なデータとなる．さらに，時空間シナリオ検索では，データベースに多

[1] 株式会社 日立製作所 研究開発グループ システムイノベーションセンタ
[2] 株式会社 日立製作所 情報・通信システム社 IT プラットフォーム事業本部
[3] 株式会社 日立製作所 社会イノベーション事業推進本部 ソリューション・ビジネス推進本部
[a)] hideki.hayashi.xu@hitachi.com

数のシナリオを蓄積する必要があるため，データがより大規模化すると考える．例えば，人口 100 万人規模の政令指定都市または中核都市（約 300km^2）を対象とする津波シミュレーションの結果は，1 シナリオあたり約 5 億件規模となる．さらに，震源の深さ，地震の強さ，震災の位置をそれぞれ 10 段階にパターン化し，1,000 件のシナリオを想定すると，全体として 5,000 億件規模となる．従来技術では，RDBMS (Relational Database Management System) を用いて，利用者が指定した時間帯のデータ群を高速に検索する，あるいは利用者が指定したある地域のデータ群を高速に検索することは可能である．しかし，数千億件規模のシミュレーション結果を対象として，観測データと類似する時空間シナリオを検索する場合には，時間属性と空間属性の両方を条件とした検索が必要となるため，従来技術では，実用的な時間内で検索させることは困難と考える．大規模災害時に 30 分間隔で被害推定を更新する想定で，データ変換，加工，蓄積など前処理に 10 分，推定処理に該当する時空間類似シナリオ検索に 10 分，その結果の配信や表示に 10 分と想定する．そして，観測データは断片的になることから，数百件程度と考え，数千億件規模のシミュレーション結果に対し，時空間類似シナリオ検索を 10 分以内で完了させることが実用への目標と考える．

そこで，本報告では，大規模災害時の災害状況を推定するための高速な時空間類似シナリオ検索手法を提案する．提案手法では，時系列グリッドデータの時間属性値と空間属性値から時空間インデックスを事前に作成し，同インデックスを用いて，観測データの時刻と位置に該当する時系列グリッドデータの特性値を効率的に取得可能とすることで，時空間類似シナリオ検索の高速化を図る．提案手法は，RDBMS に時空間類似シナリオ検索を容易に実装できるように，RDBMS の内部を変更せずに，RDBMS の上で動作可能な手法とする．そのため，これまでに様々な時空間インデックスが提案されているが [24], [26]，提案手法は，代表的な時空間インデックスである B^x-tree[27] の考え方を応用する．B^x-tree は，標準的な RDBMS が備えている B$^+$-tree を利用し，指定した図形と移動体の位置との効率的な時空間交差判定を可能とする時空間インデックスである．提案手法では，時間属性も考慮した空間充填曲線を用いて，時系列グリッドデータの時間属性値と空間属性値を 1 次元の整数値（時空間区画番号）に変換し，時空間区画番号をキーとする B$^+$-tree を事前に作成する．時空間類似シナリオ検索の実行時には，観測データ群を入力とし，各観測データの時間属性値と空間属性値を 1 次元の整数値に変換し，その値を入力として，時系列グリッドデータの時空間インデックスを用いて，該当するシミュレーション結果の特性値を取得可能とする．そして，観測データ群の観測値と，観測データ群の時刻と位置に該当する時系列グリッドデータの特性値の相違（距離）を計算し，距離の小さい時空間シナリオを類似シナリオとして抽出する．本報告では，提案手法の有用性を示すため，一部に実際のシミュレーション結果を用いた性能評価の結果を示し，従来手法との比較や目標との関係について考察する．

以下では，2 章で時空間類似シナリオ検索の問題設定を明確にする．3 章で関連研究について述べ，本研究の課題を明確にする．4 章で提案手法について説明し，5 章で性能評価の結果を示す．最後に 6 章で本報告のまとめを述べる．

2. 時空間類似シナリオ検索

本章では，時空間類似シナリオ検索の問題設定について述べる．

2.1 時系列グリッドデータと観測データのスキーマ

災害の分布を時系列グリッドデータにより表現する．災害の分布は，災害シミュレーションの初期値の組合せの一つを一つのシナリオとし，複数のシナリオのシミュレーション結果であることを想定する．災害の分布は，シナリオ，時間，空間を入力すると，そのシナリオ，時間，空間に該当する特性値を返すデータとする．

この要件のもと，時系列グリッドデータのリレーションを $R_g(SID, I, G, V)$ と定義する．SID はシナリオ識別子，I は時間，G は空間，V は特性値を表す．シナリオ識別子 SID は，シミュレーションのシナリオごとに一意に割り当てられる識別子を表す．時間 I は期間とし，瞬間を表す始点 T_s と終点 T_e で表現される．空間 G はグリッドを構成するセルの位置と形状とし，四つの座標 (X_1, Y_1), (X_2, Y_2), (X_3, Y_3), (X_4, Y_4) で表現される．特性値は時間 I と空間 G に該当する災害の特性値とし，実数値，整数値，文字列値などで表現される．シミュレーションの種類によっては，複数の特性値が入る場合がある．特性値が n 個の場合，時系列グリッドデータのリレーションは，$R_g(SID, I, G, V_1, ..., V_n)$ となる．本報告では，災害の分布の特性値が一つの場合で説明するが，特性値が複数の場合でも拡張可能である．

一方，観測データは，センサが生成する観測値を表現する．観測データは，時間，空間を入力すると，その時間と空間に該当する観測値を返すデータとする．

この要件のもと，観測データのリレーションを $R_p(OID, I, G)$ と定義する．OID はセンサの識別子，I は時間，G は空間を表す．OID は，センサごとに一意に割り当てられる識別子である．時間 I は期間とし，瞬間を表す始点 T_s と終点 T_e で表現される．G はセンサの位置とし，一つの座標 (X, Y) で表現される．

2.2 時系列グリッドデータと観測データの距離

時系列グリッドデータと観測データの類似性を判定するための距離を定義する．ここでの距離とは，ある時間帯の観測データ群の観測値が，あるシナリオの時系列グリッドデータの特性値との相違を示す指標とする．データ集合の類似性を判定する距離の定義は様々存在するが，ここでは，代表的なユークリッド距離の考え方を用いる．時系列グリッドデータを $R_g(SID, I, G, V)$，観測データを $R_p(OID, I, G, V)$，観測データ群の時間帯を i，時系列グリッドデータのシナリオ識別子を sid とする場合，距離 D は式 (1) の通り定義される．

$$D(R_g, R_p, i, sid)^2 = \sum \begin{cases} (r_g.V - r_p.V)^2 \text{ if } intersect(r_g.I, r_p.I) \\ \qquad\qquad \wedge intersect(r_g.G, r_p.G) \\ 0 \text{ Otherwise} \end{cases}$$
$$\text{s.t. } r_g \in \sigma_{SID=sid}(R_g),\ r_p \in \sigma_{intersect(I,i)}(R_p) \quad (1)$$

ここで，$intersect$ は，引数となる二つの時間属性値または二つの空間属性値が交差する場合に真値を返す関数を表す．σ はリレーショナル代数の選択演算を表し，添え字は選択条件を示す．式 (1) では，時系列グリッドデータの時間属性値と観測データの時間属性値が交差し，かつ時系列グリッドデータの空間属性値と観測データの空間属性値

が交差する場合に，時系列グリッドデータの特性値と観測データの観測値の差を 2 乗し，同じシナリオ内で累積和を計算する．この値は，時系列グリッドデータの特性値と観測データの観測値が近い値ほど，小さな値となり，類似することを示す．

2.3 時空間類似シナリオ検索の定義

時空間類似シナリオ検索は，ある時間帯の観測データ群の観測値と類似する上位 k 件の時空間シナリオを抽出する．時系列グリッドデータを $R_g(SID, I, G, V)$，観測データを $R_p(OID, I, G, V)$，観測データ群の時間帯を i，結果として抽出するシナリオの件数を k とする場合，時空間類似シナリオ検索 $STSim$ を式 (2) の通り定義する．

$$\begin{aligned} STSim(R_g, R_p, i, k) = \{&sid \in S \mid |S| = k, \\ &D(R_g, R_p, i, sid) < D(R_g, R_p, i, sid'), \\ &sid' \in \Pi_{SID}(R_g) - S\} \end{aligned} \quad (2)$$

ここで，S は時空間類似シナリオ検索の結果となるシナリオ識別子の集合で，$|S|$ は S の要素数を表す．Π はリレーショナル代数の射影演算を表し，添え字は射影条件を示す．この式では，時空間類似シナリオ検索は，時系列グリッドデータと観測データの距離関数の値が小さい（類似性の高い）上位 k 件のシナリオ識別子 sid を抽出することを表す．

時空間類似シナリオ検索では，全ての時空間シナリオを対象に，式 (1) の $intersect(r_g.I, r_p.I) \wedge intersect(r_g.G, r_p.G)$ で示す時系列グリッドデータと観測データ群の時空間交差判定を行う．時系列グリッドデータのシナリオや観測データの件数が増えると，この判定回数が多くなり，時空間類似シナリオ検索の処理時間が長くなると考える．

3. 関連研究

時空間類似シナリオ検索では，時空間交差判定が重要な技術となる．本章では，既存研究として，効率的な交差判定を実現する時間インデックス，空間インデックス，時空間インデックスについて説明する．そして，時空間類似シナリオ検索に従来技術を用いた場合の問題を示し，本研究の課題について述べる．

3.1 時間インデックス

時間データベースの研究分野において，データ集合間の時間交差判定を効率的に実現するための時間インデックスが提案されている．文献 [32] では，MVBT (Multiversion B^+-tree)[9] を用いた交差判定手法を提案し，性能評価により，この手法が，R*-tree[10] や B^+tree を用いた手法より高速に処理できることを示している．文献 [12] では，上限と下限からなる時間的な期間を示すデータを管理可能な RI (Relational Interval) -tree[22] を用いた交差判定手法が提案され，既存の RDBMS の上で動作可能な実装を示している．

3.2 空間インデックス

空間データベースの研究分野において，データ集合間の空間的な交差判定を効率的に実現するための空間インデックスが提案されている [13], [28]．空間的な交差判定は，点，線，多角形などの図形同士の重なりを判定する処理となる．空間的な交差判定の課題は，空間データベースに蓄積された大量の空間オブジェクトの中から，空間属性値が交差している空間オブジェクトを高速に抽出することにある．まず，階層的な平衡木のデータ構造では，R-tree[14] や R*-tree[10] が代表的である．これらの空間インデックスでは，オブジェクトをその空間属性値の最小外接矩形で管理し，階層的に入れ子になった相互に重なり合う最小外接矩形で空間を分割する．そして，B-tree を拡張したデータ構造となり，木構造の各ノードのエントリにまとまって最小外接矩形の情報が格納される．そのため，ページ単位でディスクと主記憶の間で入出力処理が行われるディスク型のデータベースシステムに有効なインデックスとされる．

次に，階層的な非平衡木のデータ構造では，四分木 [16] や k-d-tree[11] などが代表的である．これらの空間インデックスでは，空間的に近くのオブジェクトをまとめるアプローチで，空間を再帰的に分割する．これらの空間インデックスを用いると，交差判定の比較回数が削減されるため，CPU 処理時間の削減が課題となる主記憶型のデータベースに有用な空間インデックスとされる．

さらに，空間オブジェクトの空間属性値を 1 次元の整数値に変換してインデックスを作成するマッピングベースの空間インデックスが提案されている [15], [18], [33]．これらの空間インデックスでは，既存の RDBMS の内部を変更せずに，空間的な交差判定を効率的に実現することを目的に，標準的な RDBMS でサポートされている B^+-tree を利用する．空間オブジェクトの空間属性値を 1 次元の整数値に変換する際，Z-order 曲線や Hilbert 曲線などの空間充填曲線 [17] を用いる．そして，この 1 次元整数値をキーとして，B^+-tree を用いた空間インデックスを作成する．交差判定の実行時は，検索条件となる図形を 1 次元の整数値に変換し，B+-tree を参照して，検索対象を絞り込む．

3.3 時空間インデックス

時空間データベースの研究分野において，データ集合間の時空間交差判定を効率的に実現するため，時空間インデックスが提案されてきた [24], [26]．時空間的な交差判定は，時間的な範囲を示す期間と，空間的な範囲を示す図形の重なりを判定する．課題としては，時空間データベースに蓄積された大量の時空間オブジェクトの中から，時間属性値と空間属性値が交差している時空間オブジェクトを高速に抽出することである．

この課題を解決するため，初期の研究では，過去の時空間オブジェクトを検索対象とし，空間属性に時間属性を追加して，R-tree で管理する 3D-Rtree[30] や，時間的な変化をタイムスタンプ付きの R-tree で管理する HR-tree[25] などが提案されている．その後，将来の時空間オブジェクトを検索対象とし，R-tree に格納する外接矩形の時間発展を考慮する TPR-tree[29] が提案されている．TPR-tree を RDBMS に組み込む場合，RDBMS の内部を変更する必要があり，その工数が大きいとされる．一方，時空間オブジェクトの時間属性値と空間属性値を 1 次元の整数値に変換してインデックスを作成する B^x-tree[27] や B^{dual}-tree[31] などの時空間インデックスが提案されている．これらの時空間インデックスは，B^+-tree を用いるため，既存の RDBMS 上で動作し，高速に時空間的な交差判定を実現する．

3.4 本研究の課題

時空間類似シナリオ検索に，既存の RDBMS で実現されている時間インデックスと空間インデックスを用いる場合の各問題について述べる．

まず，時間インデックスを用いる場合について考える．この場合，時系列グリッドデータの時間属性に時間イン

デックスを作成する．そして，観測データの時刻をキーに，グリッドデータの時間属性のインデックスを参照し，観測データの時刻を含む時間属性をもつグリッドデータを検索する．その後，観測データの観測値と時系列グリッドデータの特性値の距離を計算し，類似性を判定する．例えば，シナリオが 1,000 件で，各シナリオが東西 500× 南北 500 グリッド（25 万グリッド），1 時間間隔で 24 時間分の時系列グリッドデータと，1 時間で 100 件の観測データ群の時空間類似シナリオ検索を行う場合を考える．この場合，時系列グリッドデータの時間インデックスを用いて，1 時間内のグリッドデータを抽出できるが，その件数が 25 億件（=観測データ 100 件 × シナリオ 1,000 件 × グリッド数 25 万件/時間）にも達する．その結果，25 億回のグリッドデータと観測データの空間的な交差判定を行う必要があり，時空間類似シナリオ検索の処理時間が長くなると考える．

次に，空間インデックスを用いる場合について考える．この場合，時系列グリッドデータの空間属性に空間インデックスを作成する．そして，観測データの位置をキーに，グリッドデータの空間属性のインデックスを参照し，観測データの位置を含む空間属性をもつグリッドデータを検索する．その後，観測データの観測値とグリッドデータの特性値の距離を計算し，類似性を判定する．先述の時間インデックスを用いる場合と同様の例で，空間インデックスを用いる場合の処理を考える．この場合，最悪のケースとして，100 件の観測データの位置が全て異なるグリッドデータの空間属性に含まれる場合，100 件の観測データの位置を含む 100 件のグリッドデータを絞り込むことができるが，その件数が 2,400 万件（=観測データ 100 件 × シナリオ数 1,000×24 時間）にも達する．そのため，2,400 万回のグリッドデータと観測データの時間的な交差判定（観測データの時刻がグリッドデータの期間に含まれるのかの判定）を行う必要があり，時空間類似シナリオ検索の処理時間が長くなると考える．

以上より，時空間類似シナリオ検索の処理時間を短縮することを課題とし，時系列グリッドデータと観測データ群の時空間交差判定を効率的に実現する手法を考える．

4. 提案手法

本章では，センサの断片的な観測データ群に類似する時空間シナリオを高速に検索する手法について述べる．

4.1 アプローチ

代表的な時空間インデックスとして，B^x-tree[27] が提案されている．B^x-tree[27] では，移動体の時刻と位置を 1 次元の整数値に変換し，その値をキーに B$^+$-tree を作成して，指定した図形と移動体の位置との交差判定を効率的に行う．提案手法は，その考えを応用し，時系列グリッドデータと観測データの時間属性値と空間属性値を 1 次元の整数値に変換し，その値をキーに B$^+$-tree を作成し，同 B$^+$-tree を用いて，時空間交差判定を効率的に行う手法とする．具体的には，時間と空間それぞれを一定の区画（時空間区画）に分割し，時系列グリッドデータと観測データを時空間区画に割り当てる．時空間類似シナリオ検索で時空間交差判定を行う場合には，時系列グリッドデータと観測データを時空間区画でつき合わせ，その後，詳細な判定を行う．そのため，提案手法では，時空間区画に番号（時空間区画番号）を割り当て，時系列グリッドデータと観測データのリレーションの各レコードに時空間区画番号を関連付ける．提案手法は，時空間区画への番号付与，時系列グリッドデータへの時空間区画番号の関連付け，観測データへの時空間区画番号の関連付け，時空間区画番号を用いた時系列グリッドデータと観測データの時空間交差判定，類似性判定から構成される．以下，それぞれの処理について述べる．

4.2 時空間区画への番号付与

提案手法では，事前に時間と空間をそれぞれ一定範囲の区画に分割する．それぞれを時間区画，空間区画と呼ぶ．

時間区画については，開始時刻を t_{min}，区画範囲を Δt とするとき，時間区画 $tp_i(i$：時間区画番号，$i = 0, 1, \cdots)$ は，$tp_i = [t_i, t_{i+1}]$（但し，$t_0 = t_{min}, t_{i+1} = t_i + \Delta t$）と定義される．空間区画については，$xy$ 平面で考えると，x と y の各区画の組合せで構成される．x の区画 xp_j（j：x 区画番号，$j = 0, 1, \cdots$）は，x の最小値を x_{min}，x の区画範囲を Δx とするとき，$xp_j = [x_j, x_{j+1}]$（但し，$x_0 = x_{min}, x_{i+1} = x_i + \Delta t$）と定義される．$y$ の区画 yp_k（k：y 区画番号，$k = 0, 1, \cdots$）は，y の最小値を y_{min}，y の区画範囲を Δy とするとき，$yp_k = [y_k, y_{k+1}]$（但し，$y_0 = y_{min}, y_{k+1} = y_k + \Delta y$）と定義される．その上で，空間区画 sp_{jk}（$j = 0, 1, \cdots, k = 0, 1, \cdots$） は，$sp_{jk} = \{xp_j, yp_k\} = \{[x_j, x_{j+1}], [y_k, y_{k+1}]\}$ と定義される．以上より，時空間区画 $stp_{ijk}(i = 0, 1, \cdots, j = 0, 1, \cdots, k = 0, 1, \cdots)$は，時間区画と空間区画の組合せで構成されるため，$stp_{ijk} = \{tp_i, xp_j, yp_k\} = \{[t_i, t_{i+1}], [x_j, x_{j+1}], [y_k, y_{k+1}]\}$ と定義される．

提案手法では，時空間区画に番号（時空間区画番号 stc）を割り当てる．時空間区画番号は，時空間類似シナリオ検索における時系列グリッドデータと観測データの時空間的な交差判定でつき合わせるときのキーとなる．時空間類似シナリオ検索の処理を想定すると，データアクセスの時空間的な近接性があると考えられるため，時間的・空間的に近い関係にあるデータに，近い時空間区画番号を割り当てることが有用と考える．

その考えのもと，まず，同じ期間の時空間区画に近い番号を割り当てるため，時空間区画番号を N ビットとする場合，その上位 n ビットを時間区画番号，残り $N - n$ ビットを空間区画番号に割り当てる．

例えば，時空間区画番号が 6 ビットの整数値で，時間区画番号に 2 ビット，空間区画番号に 4 ビットを割り当てるとする．時間の開始時刻 t_{min} を 0，区画範囲 Δt を 100 とする場合，$\{tp_0, tp_1, tp_2, tp_3\} = \{[0, 100], [100, 200], [200, 300], [300, 400]\}$ となる．時間区画 tp_0 の場合，時間区画番号の二進数表現は $(00)_2$ となり，時空間区画番号の最小値は $0(=(000000)_2)$，最大値は $15(=(001111)_2)$ となる．時間区画 tp_1 の場合，時間区画番号の二進数表現は $(01)_2$ となり，時空間区画番号の最小値は $16(=(010000)_2)$，最大値は $31(=(011111)_2)$ となる．

空間区画番号の割り当てには，空間充填曲線を用いる．空間充填曲線は様々あり，その影響を検証することも考え，空間を平易に辿る方法（Normal 法），Z-order 曲線で辿る方法（Z-order 法），Hilbert 曲線で辿る方法（Hilbert 法）のそれぞれで検討する．なお，提案手法の空間充填曲線の違いによる性能の差異については，5 章の性能評価で考察する．

図 1 に，Normal 法を用いた場合の時空間区画番号の割り当てについて示す．Normal 法では，x と y の最小値の空間区画を始点に，y 区画番号は一定で，x 区画番号が大きくなる方向に空間区画を辿る．x 区画番号が最大になる

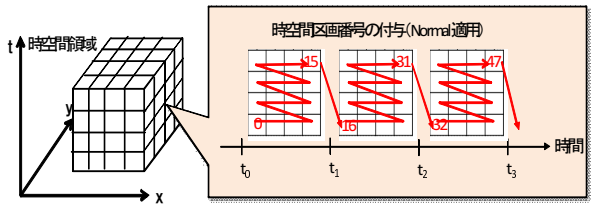


図 1 Normal 法を用いた時空間区画領域への番号割り当て

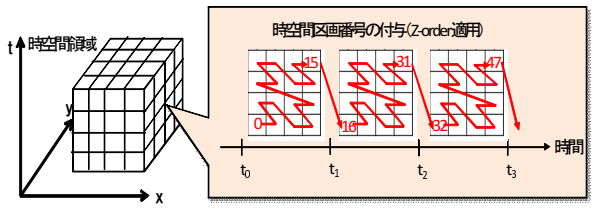


図 2 Z-order 法を用いた時空間区画領域への番号割り当て

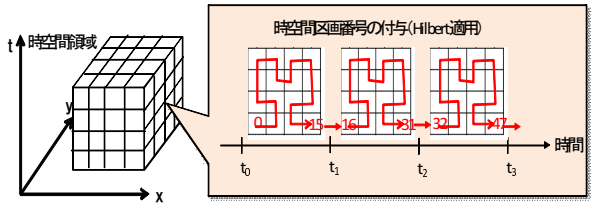


図 3 Hilbert 法を用いた時空間区画領域への番号割り当て

と，y 区画番号が大きくなる方向に一つ進み，y 区画番号は一定で，x 区画番号が大きくなる方向に空間区画を辿る．この巡回を x 区画番号と y 区画番号の両方が最大となる区画に達するまで繰り返す．Normal 法の空間区画番号は，x 区画番号の最大値を j_{max}，x 区画番号を j，y 区画番号 k とすると，$j_{max} \times k + j$ となる．例えば，x 区画 [100,200]（x 区画番号：1），y 区画 [200,300]（y 区画番号：2）の空間区画番号は，9（$= 4 \times 2 + 1$）となる．その上で，時間区画 [100,200]（時間区画番号：1）の場合，時空間区画番号は 25($= (0110001)_2$) となる．

図 2 に，Z-order 法を用いた場合の時空間区画番号の割り当てについて示す．Z-order 法では，Z 字で繰り返しながら一筆書きに辿る．Z-order 法による空間区画番号は，x と y の最小値の空間区画を始点とする場合，x 区画番号と y 区画番号をそれぞれ 2 進数で表現し，y の上位ビットと x の上位ビットを交互に並べ，10 進数に変換することで求められる．例えば，x 区画 [100,200]（x 区画番号：1）と y 区画 [200,300]（y 区画番号：2）から構成される空間区画番号は，それぞれを二進数で表現すると，$(01)_2$ と $(10)_2$ となり，y の上位ビットと x の上位ビットから順に交互に並べると，$(1001)_2$ となり，9($= (1001)_2$) となる．時間区画 [100,200] の場合，時空間区画番号は 25($= (011001)_2$) となる．

図 3 に，Hilbert 法を用いた場合の時空間区画番号の割り当てについて示す．Hilbert 法を用いた空間区画番号の計算方法については，文献 [23] を参考とする．例えば，x と y の最小値の空間区画を始点とする場合，x 区画 [100,200]（x 区画番号：1）と y 区画 [200,300]（y 区画番号：2）から構成される空間区画番号は，7 となる．時間区画 [100,200] の場合，時空間区画番号は 23($= (010111)_2$）となる．

4.3 時系列グリッドデータと時空間区画の関連付け

時系列グリッドデータは，その時間と空間に交差する時空間区画に関連付けられる．具体的には，時系列グリッドデータのリレーション R_g のレコード r_g が，$intersect(r_g.I, tp_i) \wedge intersect(r_g.G, sp_{jk})$ を満たす時空間区画 stp_{ijk} の時空間区画番号 stc_{ijk} と関連付けられる．時系列グリッドデータは，複数の時空間区画に関連付けられる場合がある．そのため，提案手法では，R_g を二つのリレーションに分割し，時系列グリッドデータのリレーション $R_g(GridID, SID, I, G, V)$ と，時系列グリッドデータと時空間区画番号を関連付けたリレーション $R_{gstc}(GridID, STC)$ とする．$GridID$ は時系列グリッドデータの識別子を表す．STC は時空間区画番号を示し，この属性値をキーに B$^+$-tree で時空間インデックスを作成する．R_g と R_{gstc} は，$GridID$ で関連付ける．

例えば，グリッドデータの期間 I が [100,200]，空間が G={(150,150),(250,150),(250,250),(150,250)} を頂点とするセルの場合，グリッドデータの期間と空間は，時間区画が [100,200]，空間区画が x[100,200]y[100,200] と x[200,300]y[100,200] と交差する．Z-order 法を用いる場合，このグリッドデータは時空間区画番号 19，22，25，28 と関連付けられる．

4.4 観測データと時空間区画の関連付け

観測データは，その時間と空間に交差する時空間区画に関連付けられる．具体的には，観測データのリレーション R_p のレコード r_p が，$intersect(r_p.I, tp_i) \wedge intersect(r_p.G, sp_{jk})$ を満たす時空間区画 stp_{ijk} の時空間区画番号 stc_{ijk} と関連付けられる．観測データは，一つの時空間区画に関連付けられる．提案手法では，R_p に時空間区画番号 STC を追加し，$R_p(OID, I, G, STC)$ とする．STC の属性値をキーに B$^+$-tree で時空間インデックスを作成する．

例えば，観測データの期間 I が [100,110]，空間 G が (150,250) の場合，観測データの期間と空間は，時間区画 [100,200]，空間区画 x[100,200]y[200,300] と交差する．Z-order 法を用いた場合，このポイントデータは時空間区画番号 25 と関連付けられる．

4.5 時空間交差判定

提案手法では，観測データの時空間区画番号をキー，時系列グリッドデータの時空間区画番号のインデックスを用いて，時系列グリッドデータと観測データを時空間区画番号でつき合わせる．つき合わせ方法は，時系列グリッドデータと観測データに関連付けられる時空間区画の定義により異なる．ベースラインとして，時系列グリッドデータと観測データの時空間区画の定義が同じ場合（時間区画，x 区画，y 区画の定義が同じ）を考える．この場合，時系列グリッドデータと観測データの時空間類似シナリオ検索の時空間交差判定の条件に，$r_g.STC = r_p.STC$ を追加する．

上述の拡張とし，時系列グリッドデータと観測データの時間区画の定義が異なる場合を考える．例えば，時系列グリッドデータがシミュレーション結果の場合，期間が，あるイベントの発生時刻を起点とし，そこからの経過時刻で表現されることが多い．観測データの時間は，観測した絶対時刻で表現されるものと考える．この場合，時系列グリッドデータと観測データのいずれかの時間区画をシフトさせ，合わせることで，対応可能と考える．時区間類似シナリオ検索の時空間交差判定の条件に，$r_g.STC = r_p.STC + STC\_OFFSET$

表 1 性能評価に用いた時系列グリッドデータ

パラメータ	値
シナリオ数	50, 100, 200, 400, 10,000
時間間隔	1 時間
全体の期間	24 時間
グリッド数	504 × 480
グリッドデータ数 [1/時間]	241,920
全グリッドデータ数	290,304,000 (50), 580,608,000 (100), 1,161,216,000 (200), 2,322,432,000 (400), 58,060,800,000 (10,000)

を追加する．時空間区画番号は，上位 n ビットを時間区画番号に割り当てるため，時空間区画番号にオフセットを加えることで，時系列グリッドデータと観測データを時空間区画番号でつき合わせ可能である．

その後，交差判定を詳細に行う．時系列グリッドデータと観測データの時間属性値と空間属性値を参照し，$intersect(r_g.I, r_p.I) \wedge intersect(r_g.G, r_p.G)$ の真偽を判定する．

4.6 類似性判定

時系列グリッドデータと観測データの時空間交差判定の結果が真になった場合に，2.2 節で定義した式 (1) を用いて，距離計算を行う．そして，全てのシナリオの距離が求まったところで，距離の値が小さい（類似性の高い）上位 k 件のシナリオ識別子を抽出する．

5. 性能評価

本章では，提案手法の有用性を示すために実施した性能評価の結果と考察について述べる．

5.1 評価環境

測定環境は，RDBMS を備えた DB サーバ（CPU(2.4GHz × 10 コア)×4 個，メモリ 384GB）と，データやインデックスを格納するストレージ（SAS(1.2TB)×111 個，RAID5（4D+1P）×21 ＋ 6 スペア）を 8Gbps のファイバケーブル 4 本で接続した構成とした．RDBMS を用いて，提案手法と従来手法を実装した．

表 1 に性能評価に用いた時系列グリッドデータを示す．時系列グリッドデータには，シミュレーション結果の一例として，気象庁の数値予報 GPV(Grid Point Value) のうち，MSM(Meso Scale Model) 雨量データを用いた（京都大学 生存圏研究所より入手）．MSM(Meso Scale Model) 雨量データ 1 日分を一つのシナリオとみなし，シナリオ数が 50，100，200，400 のデータはそのまま用いた．シナリオ数が 10,000 のデータは，データ量が大きいため，シナリオ数が 400 のデータをもとに疑似的に生成した．観測データは疑似的に生成し，1 時間ごとに 10 件，100 件，1,000 件とし，それぞれ異なるグリッド内で収集されるものとした．なお，表 1 の全グリッドデータ数の括弧内の値は該当するシナリオ件数を示す．

提案手法と従来手法では，それぞれ 4 章と 2 章で述べたスキーマに従い，データベースにデータを格納した．提案手法の時空間区画に関するパラメータ設定は，表 2 の通りとした．従来手法は，時系列グリッドデータの時間属性のみにインデックスを構築する場合と，空間属性のみにインデックスを構築する場合とした．両方の場合ともそれぞれ B^+tree を用いてインデックスを構築した．空間属性のインデックスは，測定環境の RDBMS が R-tree など空間インデックスをサポートしていないため，B^+tree を用いたが，どちらの場合も 3.4 節で述べた問題に直面することから，提案手法との比較に用いた．時空間類似シナリオ検索は，1 時間の観測データ群に対し，ある 1 時間の時系列グリッドデータとの距離を計算し，類似性の高い上位 10 件のシナリオを抽出する検索とした．性能評価の指標は，時空間類似シナリオ検索の処理時間とし，クエリが RDBMS に発行されてから，結果が返されるまでの時間とした．

表 2 提案手法の時空間区画の設定

パラメータ	時系列グリッドデータ	観測データ
Δt	1 時間	1 時間
Δx	5 km	5 km
Δy	5 km	5 km
N	64 bits	64 bits
n	32 bits	32 bits

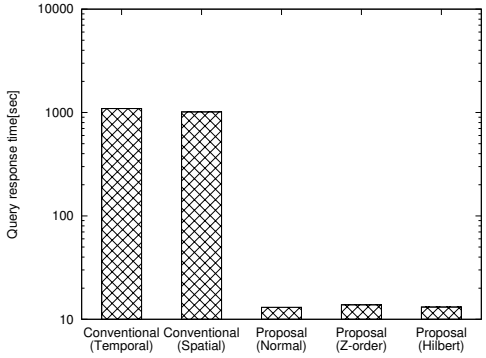


図 4 従来手法との比較

5.2 従来手法との比較

提案手法と従来手法の比較は，小規模なデータ（時系列グリッドデータのシナリオ数が 50，観測データの件数が 10 件）を用いて実施した．図 4 に，評価結果を示す．本図の横軸の”Conventional(Temporal)“は従来手法で時間属性のインデックスを用いた場合，“Conventional(Spatial)” は従来手法で空間属性のインデックスを用いた場合，“Proposal(Normal/Z-order/Hilbert)“は，提案手法でそれぞれ空間充填曲線に Normal 法，Z-order 法，Hilbert 法を用いた場合を示す．本図の縦軸は対数目盛で時空間類似シナリオ検索の処理時間を表す．

本結果から，提案手法は，従来手法より，時空間類似シナリオ検索を高速に処理できることがわかった．提案手法による時空間インデックスを用いた効率的な時空間交差判定の有用性を確認した．また，提案手法の空間充填曲線の違いによる性能の差異は確認できなかった．空間充填曲線の適用は，時空間的に近いデータをまとめてディスクアクセスする効果を見込んだが，測定環境が複数のディスクを用いて，並列にディスクアクセスするため，その効果が小さいものと考える．

本結果により，従来手法と比べた場合の提案手法の有用性を確認できたため，以降では，提案手法に絞り，性能評価の結果と考察について述べる．

5.3 観測データ件数の影響

図 5(a)(b)(c)(d) に，時系列グリッドデータのシナリオ数が 50，100，200，400 の場合の提案手法の検索処理時間を示す．各図の横軸は観測データ数で，1 時間あたり 10 件，100 件，1,000 件の場合の結果を示す．縦軸は時空間類似シナリオ検索の処理時間を示す．

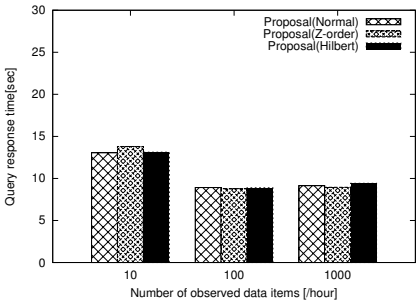
本図の結果からシナリオ数が同じ場合，観測データの件数が増えても，時空間類似シナリオ検索の処理時間があまり変わらないことを確認した．時空間類似シナリオ検索の処理内容を踏まえると，観測データの件数が増えれば，時空間交差判定の回数が増えるため，処理時間が長くなると考えられるが，この観測データの件数の範囲では，処理時間にあまり影響を与えないという結果となった．また，シナリオ件数と観測データの件数が同じ場合，空間充填曲線の違いによる大きな差異は確認できなかった．図 4 の考察と同様で，測定環境による影響と考えられる．さらに，観測データの件数が同じ場合で，シナリオ数が増えても，検索処理時間の大きな差異は見られなかった．この場合も時空間交差判定の回数が増えるため，検索処理時間が長くなることが想定されるが，このシナリオ件数の範囲では，処理時間にあまり影響を与えないという結果となった．

5.4 時系列グリッドデータのシナリオ数による影響

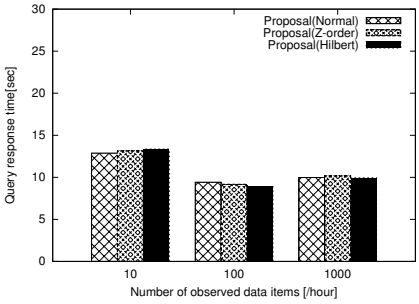
観測データの件数を 100 件とし，時系列グリッドデータのシナリオ件数を 10,000 件まで増やし，提案手法の時空間類似シナリオ検索の処理時間への影響を調べた．図 6 に，その測定結果を示す．本図の横軸はシナリオ件数，縦軸は時空間類似シナリオ検索の処理時間を示す．

本図の結果から，提案手法のいずれの場合も，時系列グリッドデータのシナリオ件数を 10,000 件まで増やすと，検索処理時間が長くなることを確認できた．これは，シナリオ件数が増えると，時空間類似シナリオ検索の時空間交差判定の回数が増えるためと考えられる．また，提案手法の空間充填曲線の違いによる検索処理時間の大きな差異は見られなかった．図 4 の考察と同様で，測定環境による影響と考えられる．

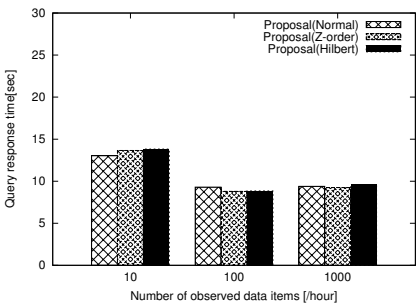
ここで，提案手法の実用性を考察するため，目標との関係について述べる．1 章で述べた通り，目標として，断片的な観測データ 100 件と，数千億件規模の時系列グリッドデータとの時空間類似シナリオ検索の処理時間を 10 分以内と考えている．図 4 の結果から，100 件の観測データと 500 億件規模の時系列グリッドデータとの時空間類似検索の処理時間は 30 秒程度であった．ここから，5,000 億件規模まで時系列グリッドデータの件数を増加させた場合，検索処理時間が線形的に増加したとしても 160 秒程度となるため，目標を達成できる見通しを得たと考える．



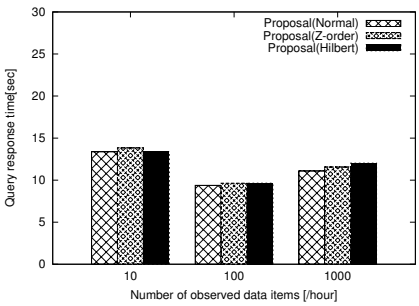
(a) シナリオ数: 50



(b) シナリオ数: 100



(c) シナリオ数: 200



(d) シナリオ数: 400

図 5 観測データ件数ごとの検索処理時間

6. おわりに

本研究では，被害状況把握に必要な災害状況を推定するため，事前に災害シミュレーション結果の時間的推移（時系列グリッドデータ）などのデータ群（時空間シナリオ）を蓄積したデータベースから，災害発生時に得られる観測データに類似する時空間シナリオを高速検索する時空間類似シナリオ検索手法について提案した．提案手法は，時空間インデックスを用いて，時空間類似シナリオ検索における時空間交差判定を効率的に行うことで，検索処理時間の短縮を図ることを特徴とした．性能評価の結果から，時間属性のみまたは空間属性のみのインデックスを用いる従来手法と比べ，高速に検索できることを確認した．また，提案手法は，100 件の観測データと 500 億件規模の時系列グリッドデータとの時空間類似検索を 30 秒程度で処理できることを確認し，実用時に 10 分以内で検索処理を完了できる見通しを得た．

今後の課題としては，具体的なシナリオの中で提案手法を適用し，その有用性を確認することが挙げられる．

謝辞 本技術は，総務省の「G 空間プラットフォームにおけるリアルタイム情報の利活用技術に関する研究開発」による委託を受けて実施した研究開発による成果である．

性能評価で用いた気象庁 GPV データは，京都大学生存圏研究所のアーカイブを利用している．ここに謝意を表す．

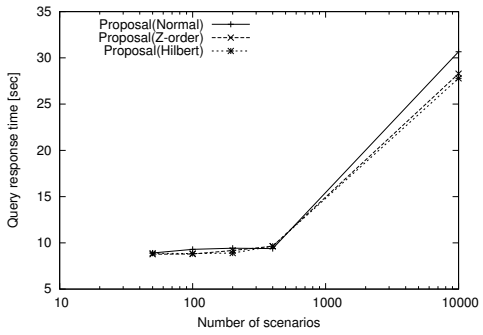


図 6 グリッドデータの大規模化に対する影響

参考文献

[1] 内閣府: 日本の災害対策,
入手先 ⟨http://www.bousai.go.jp/1info/pdf/saigaipamphlet_je.pdf⟩ (2015).

[2] 緊急災害対策本部: 平成 23 年 (2011 年) 東北地方太平洋沖地震 (東日本大震災) について,
入手先 ⟨http://www.bousai.go.jp/2011daishinsai/pdf/torimatome20150309.pdf⟩ (2015).

[3] 非常災害対策本部: 8 月 19 日からの大雨による広島県の被害状況等について,
入手先 ⟨http://www.bousai.go.jp/updates/h260819ooame/pdf/h260819ooame36.pdf⟩ (2014).

[4] 非常災害対策本部: 御嶽山の噴火状況等について,
入手先 ⟨http://www.bousai.go.jp/updates/h26ontakesan/pdf/h26ontakesan44.pdf⟩ (2014).

[5] 地震調査研究推進本部 地震調査委員会：相模トラフ沿いの地震活動の長期評価（第二版）について,
入手先 ⟨http://www.jishin.go.jp/main/chousa/14apr_sagami/index.htm⟩ (2014).

[6] 内閣府 中央防災会議 首都直下地震対策検討ワーキンググループ：首都直下地震の被害想定と対策について（最終報告）,
入手先 ⟨http://www.bousai.go.jp/jishin/syuto/taisaku_wg/pdf/syuto_wg_report.pdf⟩ (2013).

[7] 内閣府 中央防災会議 首都直下地震対策検討ワーキンググループ：発災時における政府の情報収集・集約の現状と今後の対応について,
入手先 ⟨http://www.bousai.go.jp/jishin/syuto/taisaku_wg/5/pdf/1.pdf⟩ (2012).

[8] 気象庁: 数値予報とは,
入手先 ⟨http://www.jma.go.jp/jma/kishou/know/whitep/1-3-1.html⟩

[9] Becker, B., Gschwind, S., Ohler, T., Seeger, B., and Widmayer, P.: An asymptotically optimal multiversion b-tree, *VLDB Journal*, vol.5, no.4, pp.264-275 (1996).

[10] Beckmann, N., Kriegel, H., Schneider R., and Seeger, B.: The R*-tree: an efficient and robust access method for points and rectangles, *Proc.of ACM SIGMOD'90*, pp.322-332 (1990).

[11] Bentley, J.L.: Multidimensional binary search trees used for associative searching, *Commun. ACM*, vol.18, Issue 9, pp.509-517 (1975).

[12] Enderle, J., Hampel, M., and Seidl, T.: Joining interval data in relational databases, *Proc of ACM SIGMOD'04*, pp.683-694 (2004).

[13] Gaede, V. and Gunther, O.: Multidimensional access methods, *ACM Computing Surveys*, vol. 30, issue 2, pp.170-231 (1998).

[14] Guttman, A.: R-trees: a dynamic index structure for spatial searching, *Proc. of ACM SIGMOD'98*, pp.47-58 (1984).

[15] Hayashi, H., Tanizaki, M., Sato, A., Kimura, K., Kajiyama, H., and Irie, M.: Spatial search processing in embedded devices, *Proc. of ACM GIS'09*, pp.516-519 (2009).

[16] Hjaltason, G.R., and Samet, H.: Speeding up construction of PMR quadtree-based spatial indexes, *VLDB Journal*, vol.11, issue 2, pp.109-137 (2002).

[17] Jagadish, H.: Linear clustering objects with multiple attributes, *Proc. of ACM SIGMOD'90*, pp.332-342 (1990).

[18] Jagadish, H., Ooi, B.C., Tan, K.-L., Yu, C., and Zhang, R.: iDistance: an adaptive B^+-tree based indexing method for nearest neighbor search, *ACM Trans. Database Syst.* vol.30, no.2, pp.364-397 (2005).

[19] Jensen, C.S., Lin, D., and Ooi, B.C.: Query and update efficient B^+tree based indexing of moving objects, *Proc. of VLDB'04*, pp.768-779 (2004).

[20] 越村俊一, 村嶋陽一, 日野亮太, 太田雄策, 小林宏明, 撫佐昭裕, 鈴木崇之, 井上拓也: リアルタイム津波浸水・被害推定，東京大学空間情報科学研究センター次世代社会基盤情報寄付研究部門 第 10 回公開シンポジウム,
入手先 ⟨http://i.csis.u-tokyo.ac.jp/event/20150127/index.files/150127_csisi_10_11.pdf⟩ (2015).

[21] 加藤孝明, 程 洪, 亜力坤玉素甫, 山口 亮, 名取晶子: 建物単体データを用いた全スケール対応・出火確率統合型の地震火災リスクの評価手法の構築, 地域安全学会論文集, No.8，pp.279-288 (2006).

[22] Kriegel, H.-P., Potke, M., and Seidl, T.: Managing intervals efficiently in object-relational databases, *Proc of VLDB'00*, pp.407-418 (2000).

[23] Lawder, J.K and King, P.J.H.: Querying multidimensional data indexed using the hilbert space-filling curve, *SIGMOD Record*, vol. 30, issue 1, pp.19-24 (2001).

[24] Mokbel, M.F., Ghanem, T.M., and Aref, W.G.: Spatio-temporal access methods, *IEEE Data Eng. Bull.*, vol.26, no.2, pp.40-49 (2003).

[25] Nascimento, M.A., and Silva. J.R.O.: Towards historical R-trees, *Proc. of the ACM Symp. on Applied Computing (SAC'98)*, pp.235-240 (1998).

[26] Nguyen-Dinh, L.-V., Aref, W.G., and Mokbel, M.F.: Spatio-temporal access methods: part 2 (2003 - 2010)," *IEEE Data Eng. Bull.*, vol.33, no.2, pp.46-55 (2010).

[27] Jensen, C.S., Lin, D., and Ooi, B.C.: Query and update efficient B^+tree based indexing of moving objects, *Proc. of VLDB'04*, pp.768-779 (2004).

[28] Samet, H.: Foundations of multidimensional and metric data structures, *organ Kaufmann* (2006).

[29] Saltenis, S., Jensen, C.S., Leutenegger, S., and Lopez, M.: Indexing the positions of continuously moving objects, *Proc of ACM SIGMOD'00*, pp.331-342 (2000).

[30] Theodoridis, Y., Vazirgiannis, M., and Sellis, T.: Spatio-temporal indexing for large multimedia applications, *Proc. of IEEE Conf. on Multimedia Computing and Systems (ICMCS'96)* pp.441-448 (1996).

[31] Yiu, M., Tao, Y. and Mamoulis, N.: The B^{dual}-tree: indexing moving objects by space filling curves in the dual space, *VLDB Journal*, vol.17, no. 3, pp.379-400 (2008).

[32] Zhang, D., Tsotras, V.J., and Seeger, B.: Efficient temporal join processing using indices, *Proc of IEEE ICDE'02*, pp.103-113 (2002).

[33] Zhang, R., Qi, J., Stradling, M., and Huang, J.: Towards a painless index for spatial objects, *ACM Trans. Database Syst.*, vol.39, no.3, pp.19:1-19:42 (2014).