データベースシステム 116−44
（1998. 7. 9）

位相空間データモデル Universe での空間，時間，時空間データ表現

黒木進　牧之内顕文
九州大学大学院システム情報科学研究科
知能システム学専攻
812-8581 福岡市東区箱崎 6-10-1

概要

時間的に変化する空間データを時空間データという．従来，時間的に不連続に変化する時空間データモデルが研究されてきたが，現在では時間に関して連続的に変化する時空間データのためのデータベースモデルが求められている．この課題を解決するため，われわれは位相空間データモデル Universe を提案する．位相空間データモデル Universe は，時間データと空間データと時空間データを統一して表現・検索することを目的として，これら三種のデータを位相空間という形で抽象化する．そして Universe は抽象化された位相空間を表現・検索するためのデータ表現と演算子を提供する．提供されたデータ表現と演算子を説明すると同時に，時空間データベースへの応用について述べる．

Representation of Spatial, Temporal, and Spatio-Temporal Data in the Topological Space Data Model Universe

Susumu Kuroki, Akifumi Makinouchi

Department of Intelligent Systems
Graduate School of Information Science and Electrical Engineering
Kyushu University
6-10-1 Hakozaki, Higashi, Fukuoka 812-8581, Japan

Abstract

Spatio-temporal data are such spatial data as change over time. Data models for spatial databases have been studied and presented. In these models, no time-dimension is provided, so objects are supposed to change neither their shapes, nor their positions. Recently, spatial data models with time-dimension, called spatio-temporal data models, are requested for modelling real world where objects change their positions and shapes over time. This paper presents such a data model based on a mathematical notion to represent gradual changes of objects. This model, named Universe, transforms spatio-temporal data into a topological space. Universe supports a set of operations and predicates on the topological spaces so that users can retrieve any time-varying data in a unifom way. How to represent spatio-temporal data in Universe and how to use the operations provided by Universe for spatio-temporal queries are discussed in this paper.

1 はじめに

空間データベースにおいて最近関心を集めている研究課題の一つは，空間データが時間的に滑らかに変化していくプロセスをモデル化することである．この課題を解決するために，空間データの時間的変動を表現するデータ（時空間データと呼ぶ）を記録するためのデータモデルの研究が行なわれている [1, 3, 7, 8, 11, 12, 14, 15, 16, 18]．また，取り扱うデータの高次元化も課題となっており，一部の画像データベース [9] のように，画像を特徴付ける統計量によって高次元の図形化されたデータを取り扱う手法も盛んに行なわれいている．また，通常は図形とは考えられていないデータ，例えば従業員の給料の時間的変化を記述する時間データベースのモデリングに幾何学的な考えを導入し，図形として取り扱うことも考えられる．このように，空間データの滑らかな時間的変動や，より高次元の図形として表現されるデータや，従来は図形としては取り扱われなかったデータを表現する図形が取り扱えるように従来の空間データ表現を拡張することが必要とされている．この課題を解決するためにわれわれは位相空間データモデル Universe の開発に取り組んでいる [11] が，ここではこのモデル Universe で図形をどのように表現するか，図形に対する操作としてどのようなものを定義するかを述べる．また， Universe で用いられる図形表現の利害特失について論じる．同時に，このデータモデル Universe を用いた時空間データの表現と検索の例をあげる．

2 位相空間データモデル Universe

空間データ，時間データ，時空間データを統一的に表現・検索するために，われわれは位相空間という概念を用いる．位相空間を表現するためのデータモデル Universe で，どのようにこれらデータを表現するか，またどのような演算子をもちいて位相空間を検索するか説明する．

2.1 位相空間

位相空間とは三角形や四面体などの図形を抽象化した概念であり，点列 x_n が点 x に限りなく近づくかどうかや，そこで定義された実数値関数が連続かどうかを判定できるような構造をもった集合である．数学的には，位相の定められた集合 X を位相空間という [10]．位相とは開集合系 O によって X 上に決められる構造である．この開集合系 O によって点の近接性や関数の連続性を判定できる．また，位相空間は距離の概念を用いずに定義できるため，さまざまな空間の図形を統一的に取り扱うことができる．

空間データはユークリッド空間の図形を表現する．ユークリッド空間は位相空間であると同時にユークリッド距離を持っているため，位相空間の特殊なケースとして取り扱われる．例えば，ユークリッド空間に対して，ユークリッド距離を用いて位相空間の面積や体積といった特徴量を計算できる．同時に，そのユークリッド空間は位相空間でもあるので連結成分の数や穴の数などの特徴量も計算できる．

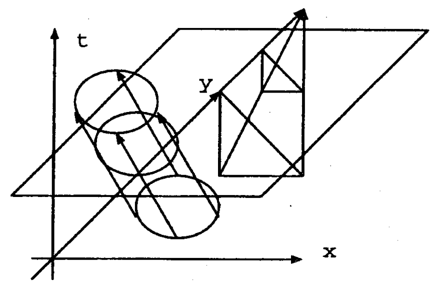

図 1: Space-Time Approach による空間データの時間変化の表現

位相空間は図形という概念を抽象化したものであるので，ユークリッド空間に存在する図形だけでなく，仮想的な図形を使って表される概念も位相空間を用いて表現することができる．時空間データの表現がその例である．例えば，三次元ユークリッド空間を並進運動している三次元図形 A を表現する時に，ユークリッド空間と時間との直積からなる四次元空間を定義する．その四次元空間での領域を用いて三次元図形 A の通過した領域を時空間データとして表現する方法がある．運動する物体を位相空間として表現するこの方法は， Space-Time Approach[6, 2] と呼ばれ，コンピュータアニメーションやロボティックスで用いられる．この方法で定義される領域は実在しない仮想的な図形で，この仮想的な図形を用いて三次元図形 A の位置と形状の時間的変化を記述できる (図 1)．但し，この四次元空間それ自体に距離を定義することはできず，時間軸においてや，時刻一定を表す等時刻面においては距離を定義できる．

このことは時間データについても成り立つ．時間データとは，ここでは時間的に変化する非空間データである．例えばある地点の温度の変遷は時間データである．先に述べた Space-Time Approach を用いれば，温度の時間的変遷は時間と温度の直積空間において「折れ線」という位相図形に変換される．時間と温度の直積空間においても，先の時空間データの場合と同様，温度差や時間間隔を計算できる．しかし，この位相空間全体に関する合理的な距離を定義することはできない．

このように，地図データや CAD データは，ユークリッド空間という特殊な位相空間としてモデル化される．時空間データや時間データも Space-Time Approach によって仮想的な図形として空間データと同様に位相空間として表現できる．それゆえ，従来の空間データ表現法を位相空間も取り扱えるように拡張すれば統一的なデータモデルを定義できる．

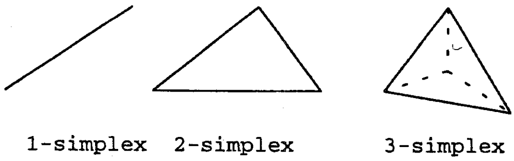

図 2: 単体の例

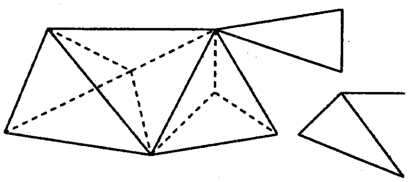

図 3: 単体的複体の例

2.2 位相空間の表現

位相空間 [10] を表現するため，組合せ構造を持つ基本的図形として単体 (simplex) を定義する．N 次元数空間 R^N の中に一般的な位置にある n+1 個の点 $a_0, a_1, ..., a_n$[2] をとる．このとき，$\lambda_0 + \lambda_1 + ... + \lambda_n = 1, 0 \leq \lambda_i (0 \leq i \leq n)$ のような n+1 個の実数 $\lambda_0, \lambda_1, ..., \lambda_n$ によって，$\lambda_0 a_0 + \lambda_1 a_1 + ... + \lambda_n a_n$ で表される点全体からなる凸集合を単体と呼び，例えば σ と表記する．特に次元を明記する時は n 単体と呼ばれる（図 2）．この定義から，0- 単体は点，1- 単体は線分，2- 単体は三角形，3- 単体は四面体となる．

また，この式で $\lambda_i = 0$ となる点の全体は，a_i を除いた n 個の点を頂点とする単体である．これは単体 σ の辺単体と呼ばれる．辺単体の辺単体もまた辺単体であると定義する．四面体の境界をなす三角形は 3- 単体の辺単体 (face) である．同時に，その三角形の辺単体となっている線分もまた，四面体の辺単体である．

位相空間は必ずしも単体とは限らないので，位相空間を表現するためには基本的図形である単体を組合せて，より複雑な図形を定義する必要がある．そこで，基本的図形である単体を元とする集合を考える．このとき，単体的複体 (simplicial complex) K は次の条件を満たす R^N 中の有限個の単体を元とする集合である．

$\sigma, \tau \in K \Rightarrow$

$\sigma \cap \tau$は空集合，またはσとτの両方の辺単体

単体的複体の例を図 3に示す．

単体的複体 K とは単体の有限集合であり，その集合の元を $\sigma_1, \sigma_2, ..., \sigma_r$ とする．このとき，R^N の中で，$|K| = \sigma_1 \cup \sigma_2 \cup ... \cup \sigma_r$ によって図形 $|K|$ を定義する．これを単体的複体 K の多面体という．逆に，位相空間 $|K|$ は単体的複体 K によって複数の単体に分割され，組合せ的な構造を与えられる．言い換えると，単体の組合せ構造を持ったものが単体的複体 K であり，そのような構造を取り払って全体を一つの点集合とみなしたものが多面体 $|K|$ である．ある多面体に与えられる組合せ構造は一意には決まらないが，元となっている単体すべてを点集合とみて和集合をとれば，その和集合は分割の種類にはよらない．

[2]ベクトル $a_1 - a_0, a_2 - a_0, ..., a_n - a_0$が一次独立

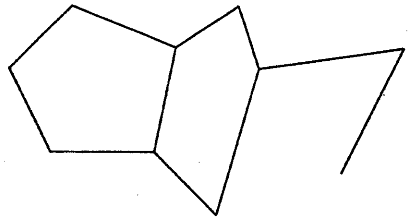

図 4: 凸胞複体の例

多面体だけを扱うとすると，位相空間として例えば球面や球体といった基本的な図形を取り扱うことができない．そのときは，多面体を非多面体にうつす同位相写像をうまく選べば非多面体である位相空間を単体分割できる．この単体分割を使えば，位相空間に関する位相的な特徴量は，単体分割されてできた単体的複体によって計算できる．それゆえ，位相的な特徴量を計算するには多面体だけ考え，それを表現する単体的複体を取り扱うことができれば必要十分である．

また，位相空間がユークリッド空間のとき，単体的複体によって表現された多面体を使えば位相空間に関する計量的な特徴量の近似値を計算できる．これは位相空間の種類を問わず普遍的に適用できる方法である．使用する単体の数を増やしてゆけばいくらでも近似の精度を上げることができる．そのため，位相空間として多面体を考えてそれを単体的複体を用いて表現すれば，計量的な特徴量の計算には実用的に十分である．

そのため，位相空間データモデル Universe では，位相空間として多面体だけを考える．

2.3 単体的複体と凸胞複体の比較

多面体 $|K|$ を表現する方法として，単体的複体を用いる方法を説明したが，これ以外の表現法には，たとえば凸胞複体 (cell complex)[10] を用いる方法がある．いま，十分多数の半閉空間の積集合を凸胞体 (cell) と呼ぶ．この凸胞体に対しても，単体に対して辺単体を定義したように辺凸胞体を定義できる．つまり辺凸胞体は凸胞体を定義する閉半空間の境界上の点全体が作る集合である．この時，凸胞複体（図 4）とは有限個の凸胞体の集合 K であり，次の条件を満たす．

$\sigma, \tau \in K \Rightarrow$

$\sigma \cap \tau$は空集合，またはσとτの両方の辺凸胞体

単体的複体表現と，凸胞複体表現のいずれによっても，多面体を表現することができる．その際の利害得失は以下のようになる．

2.3.1 データ量

多面体を表現する場合，単体的複体を用いても凸胞複体を用いても多面体の頂点数に変わりはない．それゆえ，多面体の位相を記述するデータ量を比較することによって計算機内部での多面体の表現に必要なデータ量の大小を比較する．単体も凸胞体もどちらも凸集合であるから，辺の接続関係は一意に決まるため理論的には頂点集合で表現することができる．そうすると多面体を単体または凸胞体に分解した時の要素数によってデータ量はほぼ決まるが，これは凸胞体を用いた方が要素数の少ない分解が得られる場合が多い．そのためデータ量の面では凸胞複体を用いた表現の方が有利である．

2.3.2 操作の複雑さ

多面体に関する操作として，例えば境界をとる演算を考える．この演算は多面体の境界を構成する辺単体または辺凸胞体の集合を求めることに等しい．これは多面体を構成する単体や凸胞体に向きを与え，それぞれの要素の辺をとり，それらに重みをつけて集計したものとして与えられる．しかし，このとき重みを計算するのは単体を用いた方が簡単であり，またアルゴリズムも容易である．

2.3.3 図形に関する性質

単体と凸胞体はいずれも凸図形であるという性質を持っている．単体の場合，この性質は実際的な局面，つまり浮動小数点表示された数で座標が表現された場合でも，任意の2頂点が等しくなければ常に成り立つ．また，単体は辺単体を構成するための最小数の点で定義されるため，同一の辺単体に含まれる頂点は必ず同一平面上にある．一方，凸胞体の場合は頂点位置の表現誤差によって凸胞体は必ずしも凸でなくなることがある．同様の理由から，凸胞体に含まれる辺凸胞体の頂点は理想的な条件下では同一の面上にあるが，浮動小数点数で表現された場合同一の面上にあることは必ずしも保証されない．例えば，図5では本来凸多角形であるべき点線で表示された5角形は頂点座標の表現誤差によって凸でもなければ（下側の点線が実線に移る），同一平面にも存在しなくなる（上側の点線が実線に移る）．このように凸胞複体を用いて多面体を表現すると位相的情報と計量的な情報（凸性など）の食い違いが起こる可能性があるが，単体的複体によって多面体を表現すれば，位相的情報と計量的な情報（凸性など）の食い違いは起こらない．

2.3.4 Universe で用いた表現

以上の考察から，計算機内部でのデータ表現に要する記憶領域の観点からは凸胞複体を用いるのが便利である．一方，多面体に関する操作の単純さと現実の計算機環境での表現の安全性からは単体的複体を用いるのがよいとわかった．多面体に対する操作が単純であることと多面体を表現する時に位相的なデータと計量データの間に矛盾が起こることを回避できるため，データモデル Universe においては単体的複体を用いて多面体を表現することにした．

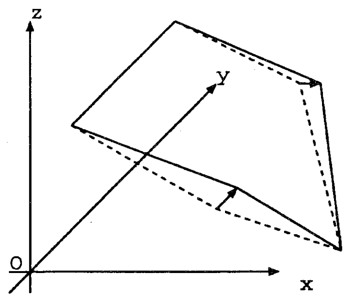

図 5: 位相的な情報と計量的なデータが食い違う例

2.4 演算子

ここでは位相空間に関する演算子として，このデータモデル Universe の提供するもの（表1）を説明する．同時に，空間データや時空間データ，時間データに関する検索に演算子がどのように使われるかを説明する．その際，位相空間としては多面体を考える．それと同時に，その多面体は単体的複体によって表現されるとともに，その表現に基づいた組合せ構造が与えられている．

2.4.1 集合演算子

集合演算子は，位相空間の和 (union)，差 (difference)，積 (intersection)（図6）を求めるものである．これらの演算子は例えば，空間データベースにおいてはしばしば使用され，その意味や使用法はすでにあきらかになっている[5]．例えば，2つの空間データの積はそれらの重なり部分を表している．そこで，ここでは時空間データにこの演算子を適用するとどのようなことができるのか説明する．

まず，二つの位相空間 $|A|$ と $|B|$ の和集合を計算する演

表 1: 演算子の一覧．位相空間は単体的複体によって表現されている．

演算子	入力 (1)	入力 (2)	出力
Intersection	位相空間	位相空間	位相空間
Union	位相空間	位相空間	位相空間
Difference	位相空間	位相空間	位相空間
Section	位相空間	位相空間	位相空間
Projection	位相空間	位相空間	位相空間
intersect	位相空間	位相空間	真または偽
contain	位相空間	位相空間	真または偽
meet	位相空間	位相空間	真または偽
disjoint	位相空間	位相空間	真または偽
equal	位相空間	位相空間	真または偽

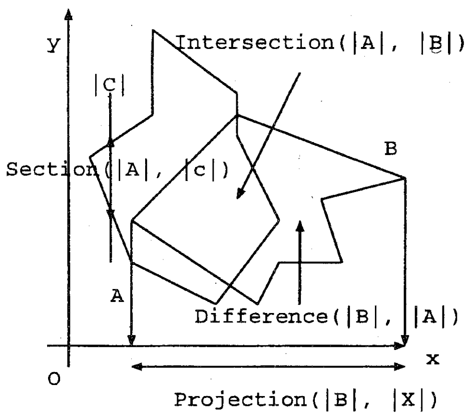

図 6: 集合演算子

算子 $Union(|A|,|B|)$ を考える．位相空間を表現する単体的複体は有限個の単体の和集合であるから，二つの位相空間 $|A|$ と $|B|$ が共通部分を持たなければ，2 つの単体的複体を構成する単体の集合もまた単体的複体となる．その結果出来上がった新たな単体的複体の表現する位相空間は，二つの別々の位相空間が併合されてできた位相空間を表す．その意味で，この演算子は二つの時空間データの併合を決める時空間演算子 $Merge(|A|,|B|)$[7] を抽象化している．二つの位相空間 $|A|$ と $|B|$ が共通部分を持つとすると，和集合演算によってその二つの時空間データは互いに「食い込むようにして」併合されることを意味する．

次に，二つの位相空間 $|A|$ と $|B|$ の差集合を計算する演算子 $Difference(|A|,|B|)$ を考える．この演算子は先の和集合演算子の逆であると考えられる．この場合，時空間データに関する差集合演算子は，時空間データ $|A|$ から時空間データ $|B|$ が脱落するプロセスをモデル化すると同時に，時空間に関する分裂演算子 $Split(|A|,|B|)$[7] を抽象化している．二つの時空間データが共通部分を持つとすると，その二つの時空間データは互いに『引きちぎれるようにして』分裂してできた一方の位相空間を意味する．

最後に，二つの位相空間 $|A|$ と $|B|$ の積集合を計算する演算子 $Intersection(|A|,|B|)$ を考える．この演算子は先の差集合と組になっており，$Difference(|A|,|B|)$ によって時空間データ $|A|$ から脱落した時空間データの片割れである．

積集合演算子の変種も存在する．それは断面演算子（section）である．断面演算子 $Section(|A|,|B|)$ とは，面 $|B|$ による $|A|$ の断面を求めるものである．断面演算子が時空間データに適用された時，この面 $|B|$ が例えば等時刻面であれば，その時刻における時空間データの位置と形が返される．また，その面 $|B|$ が空間的な制約だけによって決まるものである場合，位相空間 $|A|$ を $|B|$ に制限してえられる領域の時間変化が得られる．例えば地形データの時間変化 $|A|$ と標高 100 メートルを表す面 $|B|$ が与えられると，$|B|$ による $|A|$ の断面 $Section(|A|,|B|)$ は標高 100 メートルの地域の時間変化を表している．このように，断面演算子によってある領域に焦点をあて，そこでのデータの時間変化が得られる．

それ以外に，位相空間 $|A|$ を部分空間 $|B|$ に射影する演算子も存在する．例えば $|B|$ が時間軸とすると，射影演算子 $Projection(|A|,|B|)$ は，位相空間 $|A|$ の存在する時区間を与える．また，$|B|$ が等時刻面であるとすると，射影演算子 $Projection(|A|,|B|)$ は，位相空間 $|A|$ として表現された時空間データが通過した領域を与える．例えば，位相空間 $|A|$ が降水量が 1mm 以上の領域の時間変化を表すとすると，射影演算子 $Projection(|A|,|B|)$ によって降水量が 1mm 以上の領域の全体が得られる．

2.4.2 位相演算子

二つの位相空間に関する位相的な関係を記述するための演算子を位相演算子という．現在このモデル Universe では，位相演算子として位相述語が定義されている．位相述語とは，その述語が規定する二つの位相空間の位相的な関係，例えば共通部分を持つかどうかという関係が成り立つかどうかを判定し，成り立っていれば真を，そうでなければ偽を返す演算子である．二つの位相空間に関する位相述語は $intersect(|A|,|B|)$（二つの位相空間が共通部分を持つ時にだけ真），$disjoint(|A|,|B|)$（二つの位相空間が共通部分を持たない時にだけ真）の二つに分けられる．位相述語 $intersect(|A|,|B|)$ の変種として $contain(|A|,|B|)$ や，$meet(|A|,|B|)$ が与えられる．集合演算子の時と同様，ここでも時空間データに関して，その意味と利用法を説明する．

まず，$disjoint(|A|,|B|)$ という述語が真となる場合を考える．そのような場合は，二つの位相空間 $|A|$, $|B|$ が共通部分を持たない時であるが，これは二つの時空間データがそれぞれの生存期間のすべての時刻にわたって空間的に接点を持たないことを意味する．この述語は例えば，一回も衝突しない（すべての時刻にわたって接点を持たない）時空間データの組を求めるのに使用する．具体例としては，例えば分子運動をモデル化し衝突しなかったものの組をとりだそうする時にこの時空間述語を利用して検索条件を指定できる．

時空間述語 $intersect(|A|,|B|)$ は，二つの時空間データ $|A|$ と $|B|$ がそれぞれの生存期間の共通部分を表す時区間のどこかの時点で一度でも交わることがあれば真，そうでなければ偽となる述語である．この時空間述語は先の時空間述語 $disjoint(|A|,|B|)$ と時空間の意味でも否定になっている．この時空間述語を用いるとロボティックスやコンピュータアニメーションで用いられる衝突検出の条件を，$intersect(|A|,|B|)$ と記述できる．

これら二つ以外の時空間述語には，$contain(|A|,|B|)$ や，$meet(|A|,|B|)$ や，$equal(|A|,|B|)$ がある．これらの時空間述語も上の二つの時空間述語と同様に解釈することができる．時空間述語の $contain(|A|,|B|)$ と $equal(|A|,|B|)$ は二つの時空間データの生存する期間のすべての時点にわたって，同名の空間述語 $contain(|A|,|B|)$ と $equal(|A|,|B|)$ が成り立つことと解釈される．時空間述語 $meet(|A|,|B|)$ は，それぞれの生存期間の共通期間のある時刻に同名の空間述語が成立することと解釈される．

また，時空間述語 $equal(|A|,|B|)$ については，二つの位相空間 $|A|$ と $|B|$ が点集合として等しいかどうかを調べる述語と，多面体を単体分割した結果得られる単体的複体表現の一致を問題にする述語を定義することができる．通常は点集合として二つの多面体が一致するかどうかを判定する述語 $equal(|A|,|B|$ を用いれば十分である．というのも，単体的複体で表現された多面体の形状は単体分割の結果には依存しないからである．

これ以外の時空間的な位相述語も定義することができる．そのような述語は集合演算子と位相演算子を用いて定義できる．例えば二つの時空間データを表す多面体が，ある時刻において空間的に分離していることがあるようなものを検索するとき，検索条件を指定する時空間述語として，あらたに $disjoint(|A|,|B|)\_for\_sometime$[16] を定義することも可能である．この述語は例えば

$$
\begin{aligned}
& disjoint(|A|,|B|)\_for\_sometime \\
& = disjoint(|A|,|B|) \\
& \vee equal(Intersection(|A|,|B|),A) \neq A \\
& \wedge equal(Intersection(|A|,|B|),A) \neq B
\end{aligned}
$$

と展開できるので，これにしたがって新たに定義すれば良い．

2.4.3 距離演算子

この演算子は，空間データや時間データを表す位相空間に適用できる．適用を受けるデータが空間データであれば例えばユークリッド距離に基づいて計算が行なわれる．時間データについても時区間の長さを用いて演算を行なうことができる．たとえば，空間データ $|A|$ に対して空間データ $|B|$ がある距離 d 以内にあるかどうかを出力する空間述語 $within\_distance(d,|A|,|B|)$ は例えば二つの多面体の間の距離をはかることで真偽判定される．

時空間データを表す位相空間 $|A|$ と $|B|$ についても，それぞれに対して断面演算子を用いてある特定の時刻における空間データを取り出したり，射影演算子を用いて位相空間の生存時区間を取り出すことができれば比較を行なうことができる．例えばある時空間データ $|A|$ と同時期に存在する時空間データ $|B|$ を取り出すためには，時空間述語と同名の時間述語 $intersect()$[3] を用いて，$|A|$ と $|B|$ を時間軸に射影する．それらの射影が共通部分を持つかどうかを $intersect(Projection(|A|,|T|),Projection(|B|,|T|))$ が真となるかどうかで決めて，真となる時空間データを取り出せば良い（ただし，$|T|$ は時間軸を表す位相空間）．また，生存期間がユーザから与えられた期間 $Interval$ より長い時空間データ $|A|$ を取り出す時には，時間データに関して時区間の長さを取り出す演算子 $duration()$ を用いて，$duration(Projection(|A|,|T|)) \geq Interval$ が真になるような時空間データ $|A|$ を取り出せば良い．

また，ある指定された特定の時刻にある時空間データからある距離 d 以内にある時空間データを取り出すには，二つの空間データの距離をはかるための空間演算子 distance() 定義する．この演算子を用いると，Section() 演算子によって切り出されたある時刻での空間データに対して，述語

[3] $overlap()$ とも書かれる．

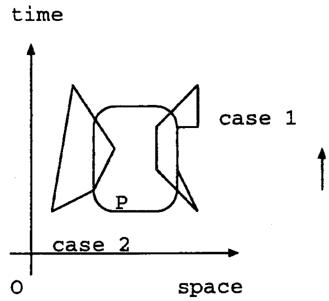

図 7: オブジェクト o と多角形 P の位相空間における関係

$$distance(Section(|A|,|H|),Section(|B|,|H|)) \leq d$$

が真となる時空間データを取り出すことで条件を指定できる（ただし，$|H|$ は指定された時刻の等時刻面）．

3 時空間データ検索への応用

位相空間データモデル Universe を用いて表現された時空間データに対して，どのように時空間検索を記述するかを述べる．ここでは，Sistla ら [16] の時空間問い合わせの例題を用いて時間，空間，時空間条件の部分を抽出し説明する[4]．

まず最初の問い合わせは，「ある将来の状態において，多角形 P の内側に入るすべてのオブジェクト o（the object that is contained in the polygon P eventually）」を求める問い合わせである．検索されるオブジェクト o は，「ある将来の時刻に o が P に含まれる」が成り立てば良い．議論を単純化するために，すべてのデータが基準となる時刻以降に存在するように定義されていて，将来にしかデータは存在しないと仮定する．そのとき，o と P の関係を位相空間で表現すると図 7のようになる．

上記の検索条件が成り立つためには o と P の時間変化を表す位相空間 $|o|$ と $|P|$ が空でない共通部分 $|I|$ とすると，$|I| = Intersection(|o|,|P|)$ を持つことが必要である．そのとき，共通部分 $|I|$ は o と P の共通部分の時間変化を記述していて，ある時刻 t_1 で o が P に含まれるとすると，時刻 $t = t_1$ を表す等時刻面による $|I|$ と $|o|$ は一致する．よって，$|o|$ から $|I|$ を取り去ることによって得られる差集合 $|D| = Difference(|o|,|P|)$ を作ると，空間的に o が P に含まれている時区間には $|D|$ は点をもたない．そのため $|D|$ と $|o|$ を時間軸に射影することによって得られる時区間 $|T_D| = Projection(|D|,|T|)$ と $|T_o| = Projection(|o|,|T|)$ は異なる．つまり前者は後者の真部分集合になっている（$|T_D| \subset |T_o|$）．逆に，このとき時区間 $|T_c| = Difference(|T_o|,|T_D|$ にわたって空間的には $Difference(o,P) = \emptyset$ となる．これはすなわち時区間 $Difference(|T_o|,|T_D|)$ にわたって

[4] 簡単のため，時間，空間，時空間に関係しない条件は省略した．

オブジェクトoが多角形Pに含まれることになる．そこで，位相空間 $|P|$ が位相空間 $|o|$ をいつかは含むときにだけ真となる上位の位相述語 $contain\_eventually(|P|,|o|)$ を，

$contain\_eventually(|P|,|o|)$
$= intersect(|P|,|o|)$
$\wedge |D_{oP}| = Difference(|o|,|P|)$
$\wedge Equal(Projection(|D_{oP}|,|T|)), Projection(|o|,|T|)) =$ "$False$"

と定義する．そうするとこの問い合わせは例えば以下のように記述される．

Select o
Where $contain\_eventually(|P|,|o|)$

2番目の問い合わせは，「1番目の問い合わせの条件を満たすもののうちで，Pに含まれる時間が2単位時間であるオブジェクトo」をすべて求めるものである．この問い合わせで追加された条件である「Pに含まれる時間が2単位時間である」という条件は，前のパラグラフで定義した時区間 $|T_c|$ の長さが2単位時間であるということになる．そのため時間データに固有の演算子 *duration()* を与えられた時間データの長さを与える演算子として定義すれと同時に，位相空間 $|o|$ が位相空間 $|P|$ に含まれる時区間を与える上位の演算子 $duration\_for\_contain(|P|,|o|)$ を

$duration\_for\_contain(|P|,|o|)$
$= duration(Difference(|T_{DoP}|,|T_o|))$
但し，
$|T_{DoP}| = Projection(Difference(|o|,|P|),|T|)$
$|T_o| = Projection(|Difference(|o|,|P|))$

と定義する．これを用いて「Pに含まれる時間が2単位時間である」という条件は

$duration\_for\_contain(|P|,|o|) =$ "2"

と表現される．そのため，2番目の問い合わせは次のように記述される．

Select o
Where $contain\_eventually(|o|,|P|)$
$\wedge duration\_for\_contain(|P|,|o|) =$ "2"'

3つめの問い合わせは，2つめの問い合わせに加えて「多角形Pに含まれなくなった時間から5単位時間以内に別の多角形Qにオブジェクトoが含まれる」という条件が追加されたものである．このとき多角形Qの時間変化を表す多面体を $|Q|$ と定義する．あらたに加わった条件のうち「別の多角形Qにオブジェクトoが含まれる」という部分は1つめの問い合わせの条件と同じである．そこで「多角形Pに含まれなくなった時間から5単位時間以内に」という部分をどのように書き下すかが問題となる．

これを書き下すためには，2つめの問い合わせのときと同様，オブジェクトoが多角形P, Qに含まれている時区間を求めてそれらの時間的な関係を調べれば良い．オブジェクトoが多角形Qに含まれる時区間 $|T_{cQ}$ は上と同様に，

$|T_{cQ}| = Projection(Differnece(|Q|,|o|),|T|)$

と書ける．ここで時区間に対してその最大値と最小値を与える上位の演算子をそれぞれ，$max\_of\_duration()$ と $min\_of\_duration()$ を定義すると，「多角形Pに含まれなくなった時間から5単位時間以内に別の多角形Qにオブジェクトoが含まれる」という条件は，

$contain\_eventually(|Q|,|o|)$
$\wedge max\_of\_duration(|T_{cQ}|) - min\_of\_duration(|T_o|)$
$\leq$ "5"

となる．よってこの問い合わせは以下のように記述される．

Select o
Where $contain\_eventually(|o|,|P|)$
$\wedge duration\_for\_contain(|P|,|o|) =$ "2"
$\wedge contain\_eventually(|Q|,|o|)$
$\wedge max\_of\_duration(|T_{cQ}|) - min\_of\_duration(|T_o|)$
$\leq$ "5"

Sistlaらのあげた例題は上記の三つであるが，これらははいずれもUniverseデータモデルの演算子を用いて記述されることがわかった．それらの例題以外にも，時間，時空間問い合わせの例題[17]を，Universeデータモデルの演算子を用いて記述できることを確認した．

4 おわりに

単体的複体を用いた位相空間表現を用いた位相空間データモデルUniverseを提案した．位相空間は図形の概念を一般化したもので，地図やCADにあらわれる図形のみならず，非空間的なデータを図形化して得られる仮想的な図形についても全く同じ形で表現・格納することができる．そのためこのデータモデルUniverseは幅広い適用領域を持っている．例えば時空間データをSpace-Time Approachを用いて位相空間に変換することで，このモデルUniverseで時空間データを取り扱うことができる．Space-Time Approachを用いて時空間データを位相空間表現することがこのモデルの特徴である．

単体的複体を用いて位相空間を表現することによって表現のためのデータ量は増える．その一方で単体的複体を用いて表現することにより，構成要素である単体は常に凸であることが保証され，位相的な情報と計量的な情報の食い違いを減らすことができる．同じことは凸胞複体を用いても行なうことができる．しかし，現実の計算機環境においては実数の浮動小数点表示のために凸胞体の凸性を保証することができないため，単体的複体を用いた．

また，単体的複体を用いて表現された位相空間に関する演算子を定義した．時空間データを位相空間として表現した時に，これら演算子の意味するところを説明すると同時にどのように使用するか述べた．

今後の課題としては，実際に位相空間データを作成し，位相空間を表現するのに必要な計算機内部でのデータ量を見積もると同時に，演算子の実行にかかる計算量を評価することがあげられる．また，われわれはここで定義した演算子の実装を行っている[13]が，演算子の計算のためにわれわれが設計・実装したアルゴリズムが破綻なく終了するかどうか確認する必要がある．また，このモデルを用いて実際にアプリケーションを構築することも課題である．

謝辞

この研究の一部は文部省平成10年度科学研究費補助金重点領域研究（課題番号08244105）の補助を受けている．

参考文献

[1] Aritsugi, M., Tagashira, T., Amagasa, T., and Kanamori, Y.: An Approach to Spatio-Temporal Queries – Interval-Based Contents Representation of Images –, *Proc. Int. Conf. on Database and Expert Systems Applications* pp. 202–213, (1997).

[2] Cameron, S.: Collision Detection by Four-Dimensional Intersection Testing, *IEEE Trans. Robotics and Automation* , Vol. 6, No. 3, pp. 291–302, (1990).

[3] Del Bimbo, A., Vicario, E., and Zingoni, D. : Symbolic Descrition and Visual Querying of Image Sequences Using Spatio-Temporal Logic, *IEEE Trans. Knowledge and Data Engineering*, Vol. 7, No. 4, pp. 609–622, (1995).

[4] Egenhofer, M., J., Frank, A., U., and Jackson, J. P.: A Topological Data Model for Spatial Databases, *Proc. 1st Symp. Spatial Databases*, pp. 271–286 (1989).

[5] Egenhofer, M., J., Franzosa, R.: On the Equivalence of Topological Relations, *Int. Journal of Geographical Information Systems* , Vol. 9, No. 2, pp. 133–152 (1995).

[6] Glassner, A. S.: Spacetime Ray Tracing for Animation, *IEEE Trans. Computer Graphics and Applications*, Vol. 8, No. 2, pp.

[7] Hornsby, K. and Egenhofer, M. J.: Qualitative Representaion of Change, *Spatial Information Theory* (Hirtle, S., and Frank, A. U.(ed.)), Springer, pp. 15–33 (1997).

[8] Kämpke, T.: Storing and Retrieving Changes in a Sequence of Polygons, *Int. Journal of Geographical Information Systems*, Vol. 8, No. 6, pp. 493–513, (1994).

[9] 加藤俊一: 画像の内容検索電子美術館への応用, 情報処理学会誌 , Vol. 33, No. 5, pp. – (1992)

[10] 河田敬義 (編): 位相幾何学, 岩波書店 (1965).

[11] Kuroki, S., Ishizuka, K., and Makinouchi, A.: Towards a Spatio-Temporal OQL for the Four Dimensional Spatial Database System Hawks, *Proc. 8th Int. Workshop on Database and Expert Systems Applications* , pp. 142–147 (1997).

[12] Masunaga, Y.: The Block-World Data Model for the Realization of Three-Dimensional Virtual Work Space, *Proc. Int. Symposium on Digital Media Information Base*, pp. 1–10, (1997).

[13] 尾下真樹, 黒木進, 牧之内顕文: d- 多面体の集合演算アルゴリズム, 情報処理学会論文誌, (1998) (投稿中).

[14] Peuquet, D.J., and Duan, N.: An Event-Based Spatiotemporal Data Model (ESTDM) for Temporal Analysis of Geographical Data, *Int. Journal of Geographical Information Systems*, Vol. 9, No. 1, pp. 7–24, (1995).

[15] Raafat, H., Yang, Z., and Gauther, D.: Relational Spatial Topologies for Historical Geographical Information, *Int. Journal of Geographical Information Systems*, Vol. 8, No. 2, pp. 163–173, (1994).

[16] Sistla, A. P., Wolfson, O., Chamberlain, S., and Dao, S.: Modeling and Querying Moving Objects, *Proc. 13th Int. Conf. Data Engineering*, pp. 422–432 (1997).

[17] Tsotras, V., Jensen, C., Snodgrass, R.: An Extensible Notation for Spatiotemporal Index Queries, *SIGMOD Record* , Vol. 27, No. 1, pp. 47–53 (1998).

[18] Worboys, M.: A Unified Model for Spatial and Temporal Information, *The Computer Journal*, Vol. 37, No. 1, (1994).