空間制約充足問題の定式化とその解法*

4J−2

市原尚久　本多一賀　大和田勇人　溝口文雄[†]

東京理科大学　理工学部[‡]　東京ガス(株)[§]

1 はじめに

レイアウト問題に代表される設計計画問題は人工知能の分野において, 制約充足問題と呼ばれており, 一般にNP完全であることから特定のアルゴリズムを持たない. これらの問題を解く為の幾つかのテクニックやアルゴリズムが提案されている.

レイアウト問題は与えられた制約条件の元で複数のオブジェクトの位置を求める問題であり, これは2次元空間上での間の関係を扱うために, 非常に複雑であり, これを解く為には一般に組み合わせ爆発が避けられない.

本論文ではこのようなレイアウト問題のあるクラスを制約充足問題の枠組で定式化を行ない, その解法について述べる.

2 空間制約充足問題の定式化

空間制約充足問題 (SCSP:Spatial-CSP) は2次元平面上の各オブジェクトの位置座標を幾つかの変数で与えて, それらの変数間に制約を与えて各オブジェクトの位置座標を決定する問題である. しかし, 通常のSCSPでは「任意の異なるオブジェクトAとBは重ならない」という制約があり, これは「AはBの(左 or 右 or 上 or 下)にある」というOr関係で定義されるので, 計算量は指数オーダーになる事が分かる.

我々は「制約は全て二項関係であり, 全てのオブジェクトが長方形で, 互いに重ならない」という制限をしたSCSPのクラス,S-SCSPを定義する.

定義 1 (S-SCSP) *S-SCSP:P* $= (D, X, Y, C_x, C_y)$ は以下のように与えられる.

- X, Yはそれぞれ変数 $x_{ij} (1 \leq i \neq j \leq n)$ と変数 $y_{ij} (1 \leq i \neq j \leq n)$ の集合であり, x_{ij}はオブジェクトi, jの二項位置関係, y_{ij}はオブジェクトi, jの二項隣接関係を意味し, 以下のような領域 **(domain)** を持つ.

$$x_{ij} \in \{l, r, a, b\}, y_{ij} \in \{adj, dis\}$$

- C_x, C_yは各変数集合X, Yに与えられた制約条件である.

又, 常に以下が成り立たなければならない.

$$x_{ij} = adj \Leftrightarrow x_{ji} = adj\ ,\ x_{ij} = dis \Leftrightarrow x_{ji} = dis$$

$$x_{ij} = l \Leftrightarrow x_{ji} = r\ ,\ x_{ij} = a \Leftrightarrow x_{ji} = b$$

ここで,$x_{ij} = l, r, a, b$はそれぞれ「オブジェクトiはオブジェクトjの(左(l) or 右(r) or 上(a) or 下(b))にある」という意味であり, $y_{ij} = adj, dis$は「オブジェクトiはオブジェクトjと(隣接している(adj) or 離れている(dis))」という意味である.

例えば,$x_{3,6} = l, y_{3,6} = adj$であれば「オブジェクト3はオブジェクト6の左側に隣接している」という意味である.

ここで,2つのオブジェクト$i, j \in D (i \neq j)$間の二項関係の変数集合を$R_{ij} = \{x_{ij}, y_{ij}, x_{ji}, y_{ji}\}$と記し, **二項ノード**と呼ぶ. 又,3つのオブジェクト$i, j, k \in D (i \neq j \neq k)$間の三項関係の変数集合を$T_{ijk} = \{R_{ij}, R_{jk}, R_{ik}\}$と記し, **三項ノード**と呼ぶ. このとき任意の二項関係には8通りの関係があるので, 各R_{ij}の領域の最大サイズは8であり, , 各T_{ijk}の領域の最大サイズは$8^3 = 512$である.

定義 2 (オペレータ) 逆方位に変換する関数オペレータOpと対角方位に変換する関数オペレータDiaを定義する.

$$Op(r) = l, Op(L) = r, Op(a) = b, Op(b) = a$$

$$Dia(l) = Dia(r) = \{a, b\}, Dia(a) = Dia(b) = \{l, r\}$$

ここで$x_{ij} = Op(x_{ji})$,$y_{ij} = y_{ji}$が常に成り立つ.

3 空間的局所無矛盾

例えば「AはBの右にあり,BはCの右にあり,CはAの右にある」という関係は空間的に矛盾であると言える. このような一般的な空間的矛盾関係を証明する式を定理1に示す.

---

*Formulating and Solving a Spatial Constraint Satisfaction Problem

†Naohisa Ichihara, Kazuyoshi Honda, Hayato Ohwada, Fumio Mizoguchi

‡Faculty of Sci. and Tech.,Science Univ. of Tokyo

§Tokyo Gas Co., Ltd.

Input:a Set of 3-ary relations T_{ijk}
Output:a Solution S
1).Let Q be a set of $T_{ijk}, i \neq j \neq k$).
2).Let $S = \{\}$.
3).Do while $Q \neq \{\}$ (Loop with Backtrack).
3a). select and delete Minimum size T from Q
3b). $labeling(T)$.
3c). $propagate(T, Q)$.
3d). Let G be a set of any ground variable $T' \in Q$,
3e). $Q \leftarrow Q - G$,$S \leftarrow S \cup G$

図 1: アルゴリズム

定理 1 (空間的矛盾証明式) *3* つのオブジェクト $i, j, k \in D(i \neq j \neq k)$ について以下の式のいずれか常に成り立つとき**空間的矛盾**となり, 成り立たないとき, **空間的局所無矛盾**となる.

$$y_{ik} = adj \quad \wedge \quad x_{ij} = x_{jk} \qquad ...(1)$$

$$x_{ik} = Op(x_{jk}) \quad \wedge \quad x_{ij} = x_{jk} \qquad ...(2)$$

$$y_{ij} = adj \quad \wedge \quad x_{ik} = Op(x_{jk}) \quad \wedge \quad x_{ij} \in Dia(x_{jk}) \quad ...(3)$$

$$y_{jk} = adj \quad \wedge \quad x_{ik} = Op(x_{ij}) \quad \wedge \quad x_{ij} \in Dia(x_{jk}) \quad ...(4)$$

$$y_{ij} = adj \quad \wedge \quad x_{ik} = x_{ij} \quad \wedge \quad x_{i,j} = Op(x_{j,k}) \qquad ...(5)$$

$$y_{jk} = adj \quad \wedge \quad x_{ik} = x_{jk} \quad \wedge \quad x_{i,j} = Op(x_{j,k}) \qquad ...(6)$$

ここで, 変数 x_{ij}, y_{ij} に領域のある要素を割り当てることをラベリングと呼び, S-SCSP 問題:Pを解く事は X, Y における全ての変数を空間的に矛盾なくラベリングすることであり, この解を $s = Sol(P)$ と記し, 全解集合を $S = \{s | s = Sol(P)\}$ と記述する. このとき, 任意の $s \in S$ の示す各オブジェクトは互いに重なる領域を持たない.

4 S-SCSP の解法

4.1 アルゴリズム

定理 1 を使って, 三項ノード T_{ijk}を局所無矛盾にすると領域サイズは 512 から 336 になる. 又, ある $R \in T_{ijk}$が局所無矛盾になってラベリングされ他の $R', R''(\neq R) \in T_{ijk}$ がラベリングされてないとき, T_{ijk}の領域サイズは 36 に縮小する.

このような特徴を使って S-SCSP を解くアルゴリズムを図 1に示す.

最初に三項ノード Tの集合を Q として入力する. ステップ 3a は Q における領域サイズが最小であるような $T \in Q$ を選び,Q から削除する. $labeling(T)$ は Tを空間的局所無矛盾になるようにラベリングする関数であり, $propagate(T, Q)$ は, ラベリングされた Tに含まれる同じ Rを含む他の全ての T'に Rを伝搬させる関数である.

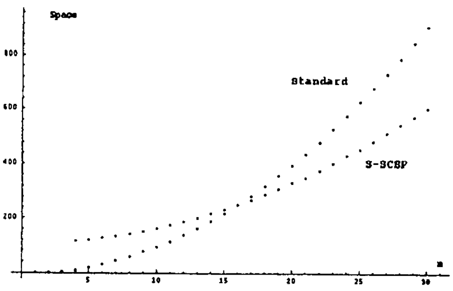

図 2: 探索空間

ステップ 3 全体はバックトラックを用いて実行されるが, そのステップ 3c,3d により制約伝播とすでにラベリングされた Tを削除していくのでステップ 3 のステップ数 Sは $S \leq |Q|$ であり, n によって単調増加し, 計算時間に大きく影響を与える.

4.2 探索空間の比較

実験的にオブジェクトの数 n を 4 から 30 まで増やしていき図 1のアルゴリズムの探索空間をシュミレートして測定した. その結果が図 2にある. 本アルゴリズムではステップ数が多いが枝の数が少ないために n が 17 以上になる従来よりも効率の良い計算をすることになる.

5 おわりに

本論文で示したアルゴリズムは S-SCSP における空間的局所無矛盾を保証する規則を使う事で従来に比べて高速に解を求めることができるが, 依然として指数オーダーが避けられていない. 又, ここではオブジェクトの大きさが考慮されていないが, 本論文で示したような定性的アプローチにこのような「大きさ」を扱うと, 定理 1 などの局所無矛盾性が成り立たなくなるのである.

現在, 我々はこれらの枠組みを用いた応用システムとして住宅間取り自動設計システムを開発中である.

参考文献

[1] Alan K.Mackworth, Consistency in Networks of Relations, *Artificial Intelligence* 8(1977) ,pp99-118.

[2] 本多, 市原, 大和田, 溝口. 自動配置システムの為の空間配置問題の分類とその解法について, 第 49 回情報処理学会全国大会.