九州工業大学 情報工学部 知能情報工学科 2012 年度 卒業論文 概要

所 属 部 門	知能数理学	指 導 教 員	下薗 真一　准教授
学 生 番 号	09231006	氏　　名	石井　祥悟
論 文 題 目	境界矩形拡張列の格子配置によるグリッドグラフ描画アルゴリズム		

1　はじめに

グラフを視覚化する際, グラフノード間に一定以上の空間を確保したいケースがある. この目的のために, 多項式時間で動作するグラフノードの二次元点集合近似照合による格子状配置アルゴリズムが提案された [1] が, 時間・空間ともに計算量の次数が高く, 現実的なサイズのグラフに直接利用することはできなかった.

そこで本研究では, 二次元近似点照合を格子点へのマッチングに限定し, 格子上への編集距離が最小となる境界矩形拡張列に近似することで, 時間・空間ともに $O(n^2)$ で動作するグリッド配置アルゴリズムを提案する.

2　境界矩形拡張列の格子配置とその解法

二次元点集合を $S = \{(x,\ y) \mid x,\ y \in \mathbb{Z}^{+2}\}$ とする. S の点を全て含む最小の軸平行な矩形を S の境界矩形という. 真に包含関係にある S の部分集合の列 $S_1 \subset \ldots \subset S_n = S$ の境界矩形の列 $R_1, \ldots, R_n$ を境界矩形拡張列という.

格子 $\mathrm{Grid}(d) = \{(dx,\ dy) \mid x,\ y \in \mathbb{N}\}(d \in \mathbb{Z})$ 上への境界矩形拡張列の配置とその編集距離を, 次のように定義する:

1. 空の点集合の境界矩形は格子配置でき, 編集距離 0 である.
2. R_k が格子配置されているならば, S_k に含まれない S_{k+1} の点 p が 1 個追加された境界矩形 R_{k+1} は, p を $\mathrm{Grid}(d)$ 上に配置する平行移動 $\mathrm{t}(p)$ で格子配置でき, 編集距離は $\mathrm{t}(p)$ と R_k の編集距離の和である.

S の任意の境界矩形拡張列は格子配置可能である. ここでは計算量を低い次数でおさえるために, 原点 $o = (0,0)$ から右上方向に拡張される境界矩形拡張列のみ考えることにする. 編集距離が最小である境界矩形拡張列を, 動的計画法を用いて求める.

グラフノードのグリッド配置は, 編集距離が最小となる境界矩形拡張列の, DP 表をバックトレースすることで得られる.

3　実験結果

点集合のグリッド配置アルゴリズムを C++(MinGW32 gcc-4.6.2) で実装し, 実行時間と出力のグリッド配置を確認した.

図 1 と図 2 は, サイズ 100 の点集合に対して, 実行した入力と出力である. サイズが小さいため, 実行時間は 1msec 以下であった.

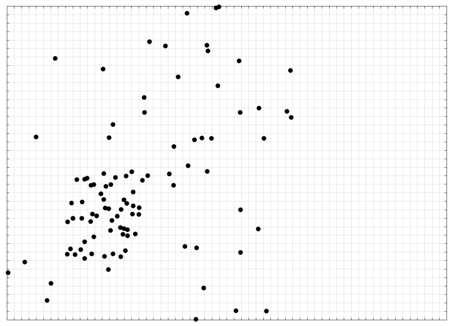

図 1: Size100 のグラフ

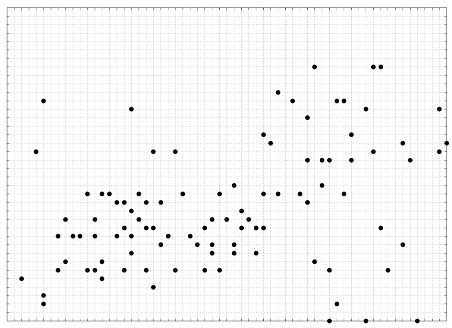

図 2: 矩形拡張列による格子配置

また, 入力頂点数が 10000 までの点集合に対して, 実行時間を計測した. サイズ 1000 のときは 0.093sec (標準), サイズ 8000 のときは 6.037sec (標準) で終了した. サイズ 8000 以降は実験環境ではヒープ領域が不足し, DP 表を生成することができなかった.

4　おわりに

グリッド幅に対し, 入力点集合の分布が密な場合は, 配置は線形に伸長される. 不必要な拡張を抑えるために, 冗長な矩形拡張列を畳み込む処理などを検討している.

参考文献

[1] 吉田英聡, 二次元点集合近似照合によるグラフの格子状配置アルゴリズム, 電子情報通信学会総合大会講演論文集 2008 年 情報・システム (1), "S-17"-"S-18", 2008-03-05