

## Octgridに基づく表編集アルゴリズム

吉住 寿洋<sup>†</sup> 塩野 康徳<sup>†</sup> 夜久 竹夫<sup>††</sup> 土田 賢省<sup>‡</sup>  
<sup>†</sup>東洋大学大学院 <sup>††</sup>日本大学 <sup>‡</sup>東洋大学

### 1 はじめに

表処理ソフトウェアは数多く存在し、用途も多岐に渡っている。しかし、これらのソフトウェアの編集操作ではユーザが予期しない動作を引き起こしたり、反応が非常に遅かったりする場合がしばしばある。そのため、効率的かつ信頼性のある表処理システムが望まれている。我々はこれまでに汎用な表のモデル化を行い[1, 2], それに基づく表処理システムの開発を行ってきた[3, 4, 5]. 本稿では、新たな表編集アルゴリズムの提案とそれらの実装について説明する。

### 2 準備

#### 2.1 表の形式化

ある表に対して周囲に周辺セルを持たせ形式化する。周辺セルは幅と高さの一方または、両方が0である(図1参照)。

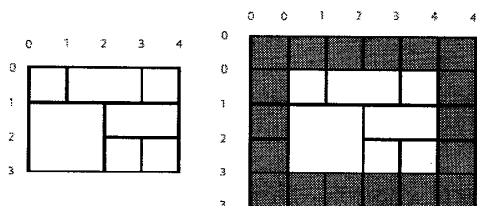


図1 周辺セルの例

#### 2.2 Octgrid

Octgrid は、表のグラフ表現である。セルをノードで表し、セルの壁の垂直/水平方向の位置が等しく最も近いセル同士の場合は、それらに対応するノードをエッジで結ぶ。図2に、不均一な大きさのセルを持つ表(矩形図)とそれに対応するoctgridの例を示す。

Octgrid は最大次数が8 などの特徴をもつ。

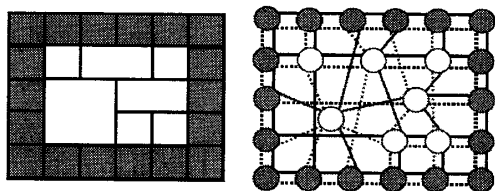


図2 表とそれに対応するoctgridの例

Table Edit Algorithms for Octgrid

<sup>†</sup> Toshihiro YOSHIZUMI(gz0700063@toyonet.toyo.ac.jp)

<sup>†</sup> Yasunori SHIONO(dz0600013@toyonet.toyo.ac.jp)

<sup>††</sup> Takeo YAKU(yaku@cs.chs.nihon-u.ac.jp)

<sup>‡</sup> Kensei TSUCHIDA(kensei@toyonet.toyo.ac.jp)

Open Information Systems, Toyo University Graduate School(<sup>†</sup>)

Department of Computer Science and System Analysis, Nihon

University (<sup>††</sup>)

Department of Information and Computer Sciences, Toyo University

(<sup>‡</sup>)

<sup>†, ‡</sup> 2100, Kujirai, Kawagoe-shi, Saitama, 350-8585, Japan

<sup>††</sup> 3-25-40, Sakurajosui, Setagaya-ku, Tokyo, 156-8550, Japan

### 3 表編集アルゴリズム

今回我々が提案するアルゴリズムは、表の分割と表の結合の二つである。*Horizontal Split Table 1-3* は、一つの表を上下二つの表に分割を、*Unify Tables 1-2* は、二つの表を一つの表に結合するアルゴリズムである。

#### 3.1 表の分割

表の分割では、一つの表を上下二つの表への分割を行う。我々は以下の三つの表分割アルゴリズムを提案する。

*Horizontal Split Table 1:* ユーザが指定した座標値でセルの分割を行い、表全体を分割する。

*Horizontal Split Table 2:* ユーザが指定したセルを含む行を削除し、削除された行を基準に表全体を分割する。

*Horizontal Split Table 3:* ユーザが指定した座標値よりも北壁が上にあるセルとそうでないセルとで表全体の分割を行う。欠けた部分を最少の数の空白のセルで補い全体を一つの矩形にする。

#### 3.2 表の結合

表の結合については以下の二つの表結合アルゴリズムを提案する。

*Unify Tables 1:* 結合する表の大きさを結合される表の大きさに変換してから二つを結合する。変換前と変換後の表は相似である。

*Unify Tables 2:* 二つの表を元の大きさのまま結合し、欠けた部分を空白のセルで補い全体を一つの矩形にする。

#### 3.3 アルゴリズム *Horizontal Split Table 2*

ここでは、前節のアルゴリズム *Horizontal Split Table 2* についてより詳しく述べる。

##### *Horizontal Split Table 2*

**入力:** Octgrid  $G$ ,  $G$  のノード  $v_d$  (セル  $d$  に対応)

**出力:** 削除した行の上下の octgrid (それぞれ  $G'$ ,  $G''$ )

**手法:**  $G$  のノードの集合  $V = \{v_1, v_2, \dots, v_n\}$ ,  $v_d$  の北壁と南壁の座標値をそれぞれ  $nw(v_d)$ ,  $sw(v_d)$ ,  $G$  の上側の周辺ノードの個数を  $k$  とする。

- $G$  のノード  $v_d$  に既存の行削除アルゴリズムを適用し、行に対応するノードを削除する。
- (1)で削除されずに残った各ノード  $v_i (1 \leq i \leq n)$  を octgrid  $G'$  (上側)か  $G''$  (下側)のいずれかのノードとして次

のように分類する。

- (a)  $nw(v_d) \geq sw(v_i)$  のとき,  $v_i$  を  $G'$  のノードとする。
- (b)  $sw(v_d) \leq nw(v_i)$  のとき,  $v_i$  を  $G''$  のノードとする。

3.  $G'$  のノードと  $G''$  のノードを結んでいる全てのエッジを削除する。

4.  $G'$  を次のように作成する。

- (a)  $G'$  に  $v_d$  と等しい北壁を持つ周辺ノードを  $k$  個追加する。
- (b) 追加したノードとそれと等しい東壁(西壁)を持つノードの中で最も下にあるノードとエッジで結ぶ。
- (c)  $G'$  に追加したノードのセルの壁の座標値に対応する属性値を設定する。

5.  $G''$  を次のように作成する。

- (a)  $G''$  に  $v_d$  と等しい南壁を持つ周辺ノードを  $k$  個追加する。
- (b) 追加したノードとそれと等しい東壁(西壁)を持つノードの中で最も上にあるノードとエッジで結ぶ。
- (c)  $G''$  の全てのノードのセルの壁の座標値に対応する属性値を設定する。

6. 上側のグラフを  $G'$ , 下側のグラフを  $G''$  として出力する。

このアルゴリズムの計算量は以下のように予想される。

[予想]

『アルゴリズム *Horizontal Split Table 2* の計算量は  $O(n)$  と予想される。ただし,  $n$  はノードの個数である。』

#### 4 実装

上記のアルゴリズムを実装し, RD Table Editor に組み込んだ。RD Table Editor は H10-Code を基にした表編集エディタである。ここで, H10-Code とは octgrid による表のモデルを反映した XML ドキュメント形式である。RD Table Editor では左側に表, 右側にその octgrid が表示される。図3 は分割を行う前の表と octgrid が示された画面である。色のついたセルを基準にアルゴリズム *Horizontal Split Table 2* を適用する。図4 に分割した表の上側の表を, 図5 に分割した表の下側の表を示す。

#### 5 おわりに

今回は, octgrid に基づくアルゴリズム *Horizontal Split Table 1-3, Unify Tables 1-2* の設計と実装, 評価を行った。また, アルゴリズム *Horizontal Split Table 1-3* を応用し, アルゴリズム *Vertical Split Table 1-3* も実装することが可能である。

今後は, 今回提案したアルゴリズムの改良及び, 表の分割・結合以外の表編集アルゴリズムの提案, 評価を行っていく予定である。

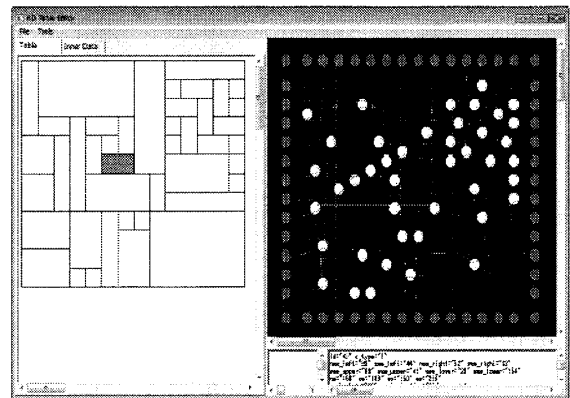


図3 アルゴリズムの適用前

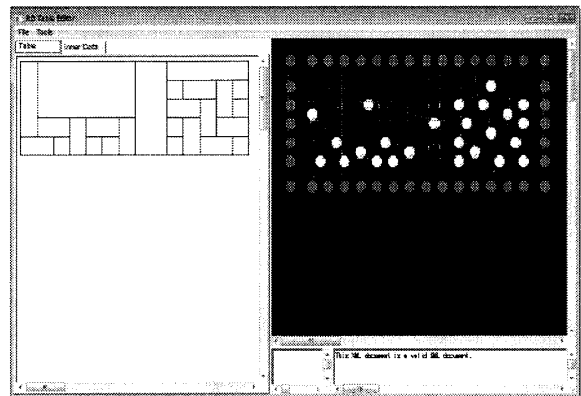


図4 アルゴリズムを適用した結果(上側)

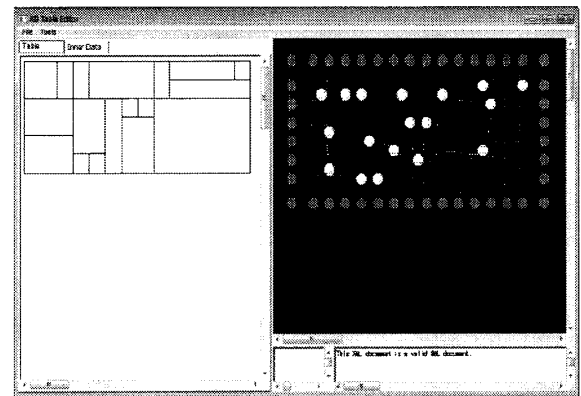


図5 アルゴリズムを適用した結果(下側)

#### 参考文献

- [1] T. Motohashi, et al., "Attribute Graphs and Their Algorithms for Table Interface," TECHNICAL REPORT OF IEICE, SS2002-5(2002-5).
- [2] T. Motohashi, K. Tsuchida, and T. Yaku, "Octgrid: A data structure of the rectangular dissection for beam oriented transformation," <http://www.waap.gr.jp/waap-rr/waap-rr-06-005/index.html>
- [3] Y. Shiono et al., "XML Representation Based on Octgrid for Table Processing", AICT2006, pp.225-232, August 8, 2006.
- [4] Y. Shiono, K. Tsuchida, and T. Yaku, "H10-Code", <http://www.waap.gr.jp/waap-rr/waap-rr-06-006/H10-Code.html>
- [5] 吉住寿洋, 塩野康徳, 山澤聡, 土田賢省, "Octgrid に対するグラフアルゴリズム", 日本応用数理学会 2008 年度年会講演予稿集, pp.175-176.