Correspondence between PLCA and Maptree: Representations of a Space Configuration

Kazuko Takahashi

Abstract We discuss the correspondence of two qualitative spatial representations: PLCA and maptree. They can provide a topological configuration of a space with finer granularity by depicting the construction of a figure using points and lines. We define conversions between these two representations to show that they have the same granularity level of expression. We also investigate preservation of planarity on the conversions.

1 Introduction

In Qualitative Spatial Reasoning (QSR) [1, 3], image data are often represented using spatial relationships between objects projected in a two-dimensional (2D) Euclidean space. This means that the region occupied by one object may overlap that of another object. It is usually represented in the form of binary or ternary relations (e.g., [4]). There are other methods of representation that use incidence relations of elements, such as points or lines and their inclusions. *PLCA* was designed to represent the connection patterns of regions using incidence relations [6]. In PLCA, an area is defined as an element that does not overlap with each other. A *maptree* is a representation that is also based on an incidence relation [8, 9]. Its data structure is an extension of a combinatorial map corresponding to the embedding of disconnected graphs. A few studies have compared QSR systems using incidence relations, which can give a more granular level of representation than systems using binary relations.

The entire space is regarded as being divided by edges in the representations based on incidence relations. The treatment of a scene including disconnected components is crucial. Specifically, it is regarded as a composition of connected components, in which the locations of the components are expressed.

Kazuko Takahashi
Kwansei Gakuin University, 2-1, Gakuen, Sanda, 669-1337, JAPAN, e-mail: ktaka@kwansei.ac.jp

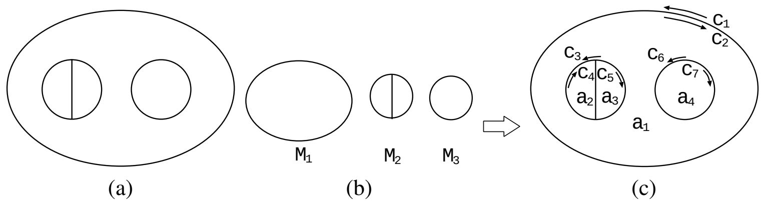


Fig. 1 Treatment of disconnected components

Consider the representations of the figure in Figure 1(a). There are three components: M_1, M_2, and M_3 (Figure 1(b)). In PLCA, these components are considered to be the areas with the interior. After embedding these on a closed surface, four disjoint areas are generated: a_1, a_2, a_3, and a_4. Of these, a_1 has two holes (Figure 1(c)). The border of an area may not be a Jordan curve, and one area may have multiple borders. Each edge in an embedding has two sides that confront the opposite areas, such as c_1 and c_2 or c_3 and c_4. By contrast, in maptree, these components are considered to be connected graphs, consisting only of strings. After embedding these on a closed surface, the complement in the surface of a connected graph is a collection of faces. Each edge in an embedding has two sides that confront the opposite areas, such as c_1 and c_2 or c_3 and c_4. The borders of the graphs are these edges.

This shows that the two representations reflect different recognitions of a scene, and it is interesting to consider their convertibility. Additionally, PLCA has several advantages. First, Coq proof assistant gives a constructive definition with its formal proof, and several properties of planarity are also proved formally [5]. Second, a transformation method from a PLCA expression to RCC, a representation using a binary relation, is constructed by adding an attribute to each area [6]. This transformation method can be extended to another fine-grained expression using a binary relation [2]. On the other hand, although maptree inherits the theoretical background of a combinatorial map, its properties have not been clarified sufficiently. It follows that, if we can convert maptree into PLCA, then the above advantages of PLCA can be applicable to maptree.

In this paper, we give a specific definition of the planar maptree, then define conversion rules between PLCA and maptree, and investigate preservation of planarity on these conversions.

This paper is organized as follows. We describe PLCA and maptree in Section 2 and Section 3, respectively. Then, we describe conversions between these two representations in Section 4. Finally, we present our conclusions in Section 5.

2 PLCA

A PLCA expression is defined as a five-tuple, $\langle P, L, C, A, om \rangle$. In PLCA, there are four kinds of object: points, lines, circuits, and areas. *A point* is the most primitive object and points are distinguishable from each other. *A line* represents segments between two points and is defined as a pair of points. Each line has a direction from the first to the second element of the pair. The inverse direction of a line l is denoted by $\bar{l}$ [1], and $\bar{\bar{l}} = l$. *A circuit* represents a closed outline and is defined as a list of lines. Each circuit is closed, that is, the first element of the first element l_0 and the second element of the last element l_n are the same. *An area* represents a region enclosed with circuits and is defined as a set of circuits. Additionally, we use a specific circuit in the outermost side of the figure, denoted by *outermost (om)*.

For example, a PLCA expression for Figure 1(c) is shown below.

$\langle P, L, C, A, c_1 \rangle$	$l_1 = (p_1, p_1)$	$c_1 = [l_1]$	$a_1 = \{c_2, c_3, c_6\}$
$P = \{p_1, p_2, p_3, p_4\}$	$l_2 = (p_2, p_3)$	$c_2 = [\overline{l_1}]$	$a_2 = \{c_4\}$
$C = \{c_1, c_2, c_3, c_4, c_5, c_6, c_7\}$	$l_3 = (p_3, p_2)$	$c_3 = [l_2, l_4]$	$a_3 = \{c_5\}$
$L = \{l_1, l_2, l_3, l_4, l_5\}$	$l_4 = (p_2, p_3)$	$c_4 = [\overline{l_2}, \overline{l_3}]$	$a_4 = \{c_7\}$
$A = \{a_1, a_2, a_3, a_4\}$	$l_5 = (p_4, p_4)$	$c_5 = [l_3, \overline{l_4}]$	
		$c_6 = [l_5]$	$c_7 = [\overline{l_5}]$

A PLCA expression is too permissive to find a corresponding topological space. For example, if there exists more than one area that contains the same circuit, such an expression does not make sense. Thus, we set a restriction on this data structure and define a consistent PLCA. We consider planarity only for a consistent PLCA.

Definition 1 (planar PLCA). If a consistent PLCA satisfies PLCA-constraints, PLCA-connectedness, and PLCA-euler, then it is said to be *a planar PLCA*.

PLCA-constraint is a condition stipulating that only straight lines are allowed and that there is no isolated point, no bridge between points nor isolated lines. PLCA-connectedness guarantees that no objects are separated, including the *outermost*. That is, each object is traceable from the *outermost*. Without this condition, some elements may be independent from the others, and we would not know where to embed them. PLCA-euler guarantees that a PLCA expression can be embedded in a 2D space so that the orientation of each circuit can be defined correctly.

A planar PLCA expression provides a surface subdivision of a 2D space. Here, we consider a surface subdivision as a configuration in which both sides of each line always belong to distinct areas.

3 Maptree

A combinatorial map is a representation of an embedding of a connected graph. In a combinatorial map, *a dart (or a half-line)*, is defined as a primitive, and other elements are defined as a permutation of a set of darts.

Let A be a finite set. We call any bijective function $\phi : A \to A$ *a permutation* of A. For $a_1, \ldots, a_n \in A$ and a permutation ϕ, if $a_2 = \phi a_1$, $a_3 = \phi a_2$, $\ldots$ $a_1 = \phi a_n$, we call

[1] We use this notation to coincide with the one in maptree. Although, in PLCA, l^+ and l^- are used to show the directions of a line.

$(a_1 \cdots a_n)$ *a cycle*. Then, any permutation is written as a collection of cycles. Let Φ be a collection of permutations of A. Φ is *transitive* if for any elements $x, y \in A$, we can transform x to y by a sequence of permutations from Φ. For a dart δ, we call a pair of δ and $\overline{\delta}$ *a complementary pair*, and $\overline{\overline{\delta}} = \delta$.

Definition 2 (combinatorial map). *A combinatorial map* $M\langle S, \alpha, \tau\rangle$ consists of:

1. A finite set $S = \{\delta_1, \ldots, \delta_n, \overline{\delta_1}, \ldots, \overline{\delta_n}\}$ of complementary pairs of darts.
2. A permutation α of S where $\alpha = \phi_1 \cdots \phi_m$. ($\phi_1, \ldots, \phi_m$ are called α-*cycles*.)
3. A permutation τ of S where $\tau = (\delta_1 \overline{\delta_1})(\delta_2 \overline{\delta_2}) \cdots (\delta_n \overline{\delta_n})$.

subject to the constraint that the collection of permutations $\{\tau, \alpha\}$ is transitive.

Given a combinatorial map M with α-cycles $\phi_1, \ldots, \phi_m$, *p-star associated with* M is an edge-labelled tree with a central black node from which edges connect to white nodes, the i-th edge being labelled with ϕ_i ($1 \le i \le m$). *A bw-tree* is a colored tree with the nodes colored black or white, subject to the condition that no two adjacent nodes have the same color. White nodes correspond to faces (regions).

Definition 3 (maptree). Let M be a finite collection of combinatorial maps. *A maptree* T_M is an edge-labelled bw-tree formed by the merging of p-stars at white nodes.

Here, we introduce a formal definition of a planar maptree, which is not explicitly described in [9].

Let T_M be a maptree for $M = (M_1, \ldots, M_k)$ $(1 \le i \le k)$ where $M_i\langle S_i, \alpha_i, \tau_i\rangle$. We consider the region assignment function *reg* from a set of α-cycles in M to a set of regions. That is, for each α-cycle ϕ, we associate the region Δ denoted by $reg(\phi) = \Delta$. For α-cycles ϕ_i of α_i and ϕ_j of α_j, if $reg(\phi_i) = reg(\phi_j)$, then it is said that M_i *and* M_j *share a region*. Henceforth, the term 'maptree' denotes the maptree associated with a region assignment function.

Definition 4 (planar maptree). A maptree T_M for $M = (M_1, \ldots, M_k)$ is said to be *a planar maptree* if the following conditions are satisfied:

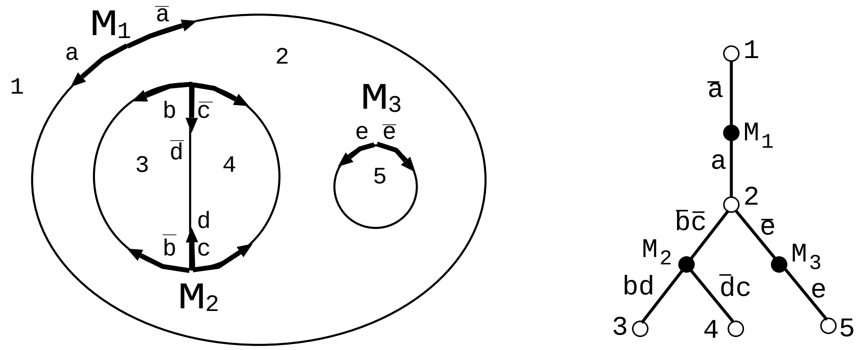
1. Exactly one root node Δ_{root} for T_M exists.
2. For ϕ, ϕ' $(\phi \neq \phi')$, if they are in the same permutation α, then $reg(\phi) \neq reg(\phi')$.
3. When $k \ge 2$, for each M_i $(1 \le i \le k)$, there exists an α-cycle ϕ such that $reg(\phi) = \Delta(\neq \Delta_{root})$ and that Δ is shared with M_j $(1 \le j \le k,\ i \neq j)$.
4. When $k \ge 2$, for any pair of M_i and M_j $(1 \le i, j \le k,\ i \neq j)$, they share at most one region.

For example, the planar maptree corresponding to an embedding of a disconnected graph in Figure 2(a) is represented as follows and depicted in Figure 2(b).

It consists of three combinatorial maps M_1, M_2, and M_3:

$M_1\langle\{a, \overline{a}\}, (a)(\overline{a}), (a\overline{a})\rangle$
$M_2\langle\{b, c, d, \overline{b}, \overline{c}, \overline{d}\}, (\overline{b}\overline{c})(bd)(\overline{d}c), (b\overline{b})(c\overline{c})(d\overline{d})\rangle$
$M_3\langle\{e, \overline{e}\}, (e)(\overline{e}), (e\overline{e})\rangle$

The region assignment function is defined as follows: $reg((\overline{a})) = 1 = \Delta_{root}$, $reg((a)) = reg((\overline{b}\overline{c})) = reg((\overline{e})) = 2$, $reg((bd)) = 3$, $reg((\overline{d}c)) = 4$ and $reg((e)) = 5$.



(a) (b)

Fig. 2 An embedding of a disconnected graph and the corresponding maptree.

4 Conversions between PLCA and Maptree

We show the conversion from a planar maptree to a PLCA expression.

Let T_M be a planar maptree for $M = (M_1, \ldots, M_k)$ $(1 \leq i \leq k)$, where $M_i\langle S_i, \alpha_i, \tau_i\rangle$ is a combinatorial map, and let *reg* be a region assignment function. First, we calculate $\beta_i = \tau_i \alpha_i^{-1}$ for each i. We bijectively map each dart, α-cycle, β-cycle, and region in T_M to elements in PLCA, and make a set of P, L, C, and A. Then, we generate their incidence relations.

Let $\mathscr{S}_S, \mathscr{S}_\alpha, \mathscr{S}_\beta$, and $\mathscr{S}_R$ be sets of darts, α-cycles, β-cycles, and regions appearing in T_M, respectively. The notation $x \mapsto y$ denotes that an element x is mapped to y.

1. Make sets of P, L, C, and A.
   We bijectively map each β-cycle to a point, dart to a line, and α-cycle to a circuit in PLCA. For a dart $\delta \in \mathscr{S}_S$, if $\delta \mapsto l$, then $\overline{\delta} \mapsto \overline{l}$. For a region, $\Delta_{root} \mapsto \perp$, because the external region of the *outermost* does not exist in PLCA; otherwise, it is mapped to an area in PLCA.
2. Generate incidence relations.
   - For $\delta \in \mathscr{S}_S$, $\delta \mapsto l$; if δ is in ρ, $\overline{\delta}$ is in ρ', and $\rho \mapsto p, \rho' \mapsto p'$, then $l = (p, p')$.
   - For $\phi \in \mathscr{S}_\alpha$, $\phi \mapsto c$; if $\phi = (\delta_1 \cdots \delta_s)$ and $\delta_j \mapsto l_j$ $(1 \leq j \leq s)$, then $c = \{l_1, \ldots, l_s\}$.
   - For $\Delta \in \mathscr{S}_R$, $\Delta \mapsto a$; if $reg(\phi_j) = \Delta, \phi_j \mapsto c_j$, then $a = \{c_1, \ldots, c_t\}$, where t is the number of ϕ_j that satisfies $reg(\phi_j) = \Delta$.
   - For Δ_{root}, if $reg(\phi) = \Delta_{root}, \phi \mapsto c$, then $om = c$.

For the obtained PLCA expression, PLCA-consistency, PLCA-connectedness and PLCA-euler clearly hold from the planarity of a maptree. However, PLCA-constraints is not satisfied, because a planar maptree admits isolated lines and bridges as well as multiple edges connecting the same pair of points. Therefore, we need a condition so that it provides a surface subdivision of a 2D space.

Proposition 1. *Let T_M be a planar maptree. If each α-cycle ϕ in T_M satisfies the following conditions, then T_M provides a surface subdivision of a 2D space: (i) there does not exist δ such that δ and $\overline{\delta}$ are both included in ϕ, and (ii) $|\phi| \geq 3$.*

We can similarly define a conversion rule from a planar PLCA expression to a maptree. In this case, the crucial point is that the mapped data are correctly divided into a set of combinatorial maps. We have proved that the obtained one is a planar maptree.

5 Conclusion

We have discussed the correspondence of two qualitative spatial representations based on incidence relations, PLCA and maptree. We have defined conversions between these two representations and clarified the condition that a planar maptree provides a surface subdivision of a 2D space. The main contribution of this paper is to relate the two representations that reflect different recognitions of a scene: area-based and string-based. We have implemented in Prolog prototypes of the conversion programs in both directions.

The proofs are done manually, and we will provide a strict proof using proof assistants in future. We also want to show the relationship between representations using incidence relations and binary relations.

References

1. Cohn, A. and Renz, J.: *Handbook of Knowledge Representation*, chapter 13. Qualitative Spatial Reasoning, pp. 551-596. Elsevier (2007).
2. Lewis, J. A., Dube, M. P. and Egenhofer, M. J.: The topology of spatial scenes in $\mathbb{R}$. *COSIT13*, pp. 495-515, LNCS 8116, Springer (2013).
3. Ligozat, G.: *Qualitative Spatial and Temporal Reasoning*, Wiley (2013).
4. Randell, D., Cui, C. and Cohn, A.: A spatial logic based on regions and "Connection". *KR1992*, pp.165-176 (1992).
5. Takahashi, K., Moriguchi, S. and Goto, M.: Formalization of a surface subdivision allowing a region with holes without coordinates. *ADG2016*, pp.190-207 (2016).
6. Takahashi, K. and Sumitomo, T.: The qualitative treatment of spatial data. *International Journal on Artificial Intelligent Tools*, 16(4):661-682 (2007).
7. Takahashi, K., Sumitomo, T. and Takeuti, I.: On embedding a qualitative representation in a two-dimensional plane. *Spatial Cognition and Computation*, 8(1-2):4-26 (2008).
8. Worboys, M.: The maptree: a fine-grained representation of space. *GIScience 2012*, pp.298-310, LNCS 7478, Springer (2012).
9. Worboys, M.: Using maptrees to characterize topological change. *COSIT13*, pp.74-90, LNCS 8116, Springer (2013).