Qualitative Spatial Representation Based on Connection Pattern and Convexity*

Shou KUMOKAWA and **Kazuko TAKAHASHI**
School of Science and Technonogy
Kwansei Gakuin University
2-1, Gakuen, Sanda, 669-1337, JAPAN

Abstract

We present an extended PLCA which can represent a convexity of an object qualitatively. PLCA is based on the simple components: $points(P)$, $lines(L)$, $circuits(C)$ and $areas(A)$, and the entire figure is represented as a combination of these components. The entire space is considered to be divided into disjoint regions, and the connection patterns of regions can be distinguished. We extend PLCA by utilizing a convex-hull of each area to give a qualitative shape representation. We formalize our method, present an algorithm to generate the symbolic expression from the given figure, and discuss the properties that should be satisfied by this expression. Our goal is to represent not only the shape of the outer circuit of a single region, but that of the boundaries between regions.

1. Introduction

Qualitative Spatial Reasoning(QSR) is a method that treats images or figures qualitatively, by extracting the information necessary for a user's purpose such as mereological relationships, relative positional relationships, relative size between regions (Cohn and Hazarika 2001; Stock 1997). In QSR systems, figures are represented not numerically but symbolically, so that the amount of data and computation can be reduced.

RCC (Randell and Cui 1992) is a logical theory that considers a space as a set of regions, in which the entire figure is represented in the form of a set of binary relations of regions. 9-intersection in another method which uses a matrix to show the relationships of objects (Egenhofer 1991; Egenhofer and Franzosa 1991; 1995). PLCA is a framework for qualitative spatial reasoning (Takahashi and Sumitomo 2005). It is based on the simple components: $points(P)$, $lines(L)$, $circuits(C)$ and $areas(A)$, focusing on the connection patterns of regions. Pairs of areas, circuits or lines never cross. Intuitively, the entire space is divided into disjoint regions. Consider, for example, the figure shown in Figure 1. It can be explicitly represented that two objects are touched by two points in PLCA, while only the property that two objects have the common part is represented in the other QSR methods.

*This research was supported by KAKENHI19500134.

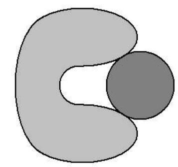


Figure 1: Objects touched by two points

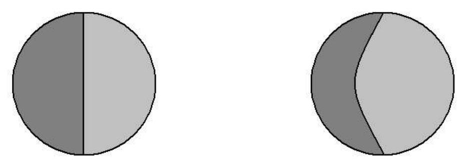


Figure 2: The same figures

However, only the characteristics of the connection patterns can be represented and there is no information on shapes. For example, two figures in Figure 2 are regarded as the same one, since both of which show two objects that are connected with a line.

Shape representation is necessary in many fields: recognizing maps or geologic changes, designing and building objects, using Geographic Information Systems (GIS).

Qualitative shape representation are studied in several works in QSRs. However, most of them focused on the shape of a single object, and they do not handle the shapes of multiple objects connected with each other (Figure 3). For example, consider the shapes of the boundaries of countries. Most of boundaries of European countries are curved while those of African countries are straight. We have to express not only the shape of a single boundary, but also the connection manner of these boundaries. In this paper, we treat these problems, and give a solution.

We extend PLCA so that it has the information of convex-hull for each area. The convexity of each object is represented using its convex-hull. We extract the difference part between the area and its convex-hull as concavity, and take recursively convex-hull of this part. This approach can ex-

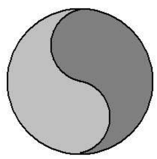


Figure 3: Figures including multiple objects

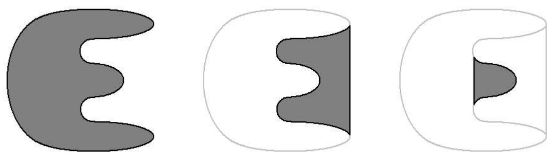


Figure 4: Hierarchical representation of convexity

press the shapes of regions on the detailed level (Figure 4). Our goal is to represent not only the shape of the boundaries of a single object, but that of the boundaries among objects and represent the convexity of areas.

We present an algorithm for generating a PLCA+ expression from a given figure in a two-dimensional plane, and discuss the properties the expression satisfy. We also discuss the expressive power of PLCA+.

This paper is organized as follows. In section 2, we present the formal definition of PLCA+, an extended PLCA expression, and the conditions that are to be satisfied. In section 3, we describe an algorithm from a given figure in a two-dimensional plane to generate the symbolic representation, and show the properties to be satisfied. In section 4, we compare our approach with the other works, and discusses the ability of PLCA+. Finally, in section 5, we show the conclusion.

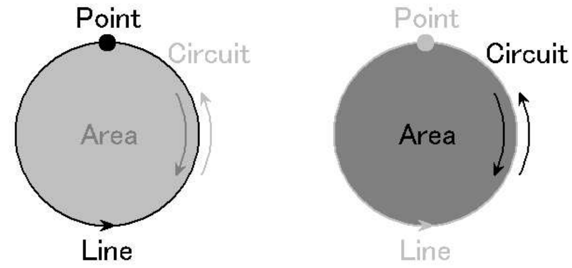
2. Definition of Extended PLCA

Definition of Classes

PLCA has four basic components: $points(P)$, $lines(L)$, $circuits(C)$ and $areas(A)$ (Figure 5). We add a new component $subPLCA$ to represent the shape of convexity of an area.

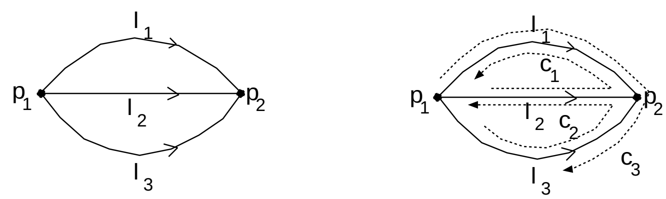
Point is defined as a primitive class.

Line is defined as a class that satisfies the following condition: for an arbitrary instance l of $Line$, $l.ps$ is a pair $[p_1, p_2]$ where $p_1, p_2 \in Point$. A line has an inherent orientation. When $l.ps = [p_1, p_2]$, l^+ and l^- mean $[p_1, p_2]$ and $[p_2, p_1]$, respectively. l^* denotes either l^+ or l^-, and l^*_{re} denotes the line with the inverse direction of the direction of l^*. Intuitively, a line is the edge connecting two (not always different) points. No two lines are allowed to cross. Note that multiple lines may have the same pair of points. In Fig. 6(a), the arrows denote the orientation of the lines. All of the lines $l_1.ps, l_2.ps$ and $l_3.ps$ are defined to be $[p_1, p_2]$, but they are distinguished by the circuits to which they belong.



(a) Point and Line (b) Circuit and Area

Figure 5: PLCA components



(a) (b)

Figure 6: Multiple lines with the same definition and the associated circuits

In this paper, we assume that each line in the figure is a curved one, although PLCA permits a straight line.

Circuit is defined as a class that satisfies the following condition: for an arbitrary instance c of $Circuit$, $c.ls$ is a sequence $[l_1^*, \ldots, l_n^*]$ where $l_1, \ldots, l_n \in Line(n \geq 1)$, $l_i.ps = [p_i, p_{i+1}](1 \leq i \leq n)$ and $p_{n+1} = p_1$. $[l_1^*, \ldots, l_n^*]$ and $[l_j^*, \ldots, l_n^*, l_1^*, \ldots, l_{j-1}^*]$ denote the same circuit for any j $(1 \leq j \leq n)$. In Fig. 6(b), we have three circuits: $c_1.ls = \{l_1^-, l_2^+\}$, $c_2.ls = \{l_2^-, l_3^+\}$, $c_3.ls = \{l_3^-, l_1^+\}$.

For $c_1, c_2 \in Circuit$, we introduce two new predicates lc and pc to denote that two circuits share line(s) and point(s), respectively. $lc(c_1, c_2)$ is $true$ iff there exists $l \in Line$ such that $(l^+ \in c_1.ls) \wedge (l^- \in c_2.ls)$. $pc(c_1, c_2)$ is $true$ iff there exists $p \in Point$ such that $(p \in l_1.ps) \wedge (p \in l_2.ps) \wedge (l_1^* \in c_1.ls) \wedge (l_2^* \in c_2.ls)$. A circuit is the boundary between an area and its adjacent areas viewed from the side of that area.

Area is defined as a class that satisfies the following condition: for an arbitrary instance a of $Area$, $a.cs$ is a set $\{c_1, \ldots, c_n\}$ where $c_1, \ldots, c_n \in Circuit(n \geq 1)$, and $\forall c_i, c_j \in a.cs; (i \neq j) \rightarrow (\neg pc(c_i, c_j) \wedge \neg lc(c_i, c_j))$. Intuitively, an area is a connected region which consists of exactly one piece. No two areas are allowed to cross. The final condition means that any pair of circuits that belong to the same area cannot share a point or a line. For areaa a_1 and a_2, if there exist circuits c_1 and c_2 such that $c_1 \in a_1.circuits$ and $c_2 \in a_2.cs$, respectively, and $lc(c_1, c_2)$ holds, then a_1 and a_2 are said to be *line-connected*.

We assume that there exists a circuit in the outermost extremity of the figure called $om(outermost)$. This means that the target figure is drawn in a finite space, and the space can be divided into a number of areas that do not overlap

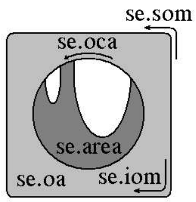


Figure 7: Important components of convex-hull

with each other.

SubPLCA is defined as a class that satisfies the following condition: for an arbitrary instance se of $SubPLCA$, it has the following components:

definition 1 *(SubPLCA)*

$$se.ps = \{p_0, p_1, \cdots, p_{n-1}\} \quad where\ p_0, p_1, \cdots, p_{n-1} \in \textit{Point}$$
$$se.ls = \{l_0, l_1, \cdots, l_{n-1}\} \quad where\ l_0, l_1, \cdots, l_{n-1} \in \textit{Line}$$
$$se.cs = \{c_0, c_1, \cdots, c_{n-1}\} \quad where\ c_0, c_1, \cdots, c_{n-1} \in \textit{Circuit}$$
$$se.as = \{a_0, a_1, \cdots, a_{n-1}\} \quad where\ a_0, a_1, \cdots, a_{n-1} \in \textit{Area}$$
$$se.area = a \qquad where\ a \in se.as$$
$$se.som = c \qquad where\ c \in se.cs$$

We call this expression SubPLCA of Area a.

Intuitively, a SubPLCA se is an expression for a restricted frame in which the extracted area $se.area$ from the source figure is pasted. There exists a circuit in the outermost extremity of the frame called $som(suboutermost)$.

We also define three components of se. $se.iom$ is the inner circuit of the frame, $se.ocs$ is the outer circuit of the convex-hull of the extracted area, and $se.oa$ which is said to be *background area* is the outside of the convex-hull of the extracted area in the frame. The correspondence of these components and the figure is shown in Figure 7, and their formal definitions are shown below.

definition 2 *(the inner circuit of the frame) $se.iom$ is such Circuit c that satisfies:*

$$\begin{array}{ll} & \forall l^* \in se.som.ls(l^*_{re} \in c.ls) \\ \wedge & \forall l^* \in c.ls(l^*_{re} \in se.som.ls) \\ \wedge & c \in se.cs \end{array}$$

It means that for each line l^* that belongs to the subouter-most circuit, the opposite direction of l^* belongs to $se.iom$, and vise versa.

definition 3 *(the outer circuit of the convex-hull of the extracted area) $se.oca$ is such Circuit c that satisfies:*

$$c \in se.oa.cs \wedge c \neq se.iom$$

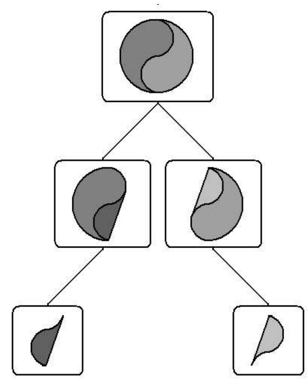


Figure 8: A tree structure of $e^+.ses$

definition 4 *(the background area) $se.oa$ is such Area a that satisfies:*

$$\begin{array}{ll} & se.iom \in a.cs \\ \wedge & |a.cs| = 2 \\ \wedge & a \in se.as \\ \wedge & a \neq se.area \end{array}$$

It means that the circuits belonging to the background area are only the inner circuit of the frame and the outer circuit of the convex-hull of the extracted area.

Tree Structure of SubPLCAs If the source figure contains n areas, then n number of $se's$ are defined independently. Moreover, if a concave part of the source figure again has a concavity, we use hierarchical representation to show its shape. It means that each SubPLCA is a PLCA+ expression which includes SubPLCA, recursively. As a result, for a PLCA+ expression e^+, $e^+.ses$ has a tree structure (Figure 8). For a SubPLCA se, if $se.area = a \wedge se.as = \{a, a_1, a_2, \cdots, a_n\}$, Area a is said to be *a parent Area* of Area $a_1, a_2, \cdots, a_n$, and $a_1, a_2, \cdots, a_n$ are said to be *child Areas* of Area a.

PLCA+ Expression

PLCA+ expression is defined as a class that satisfies the following condition: for an arbitrary instance e^+ of PLCA+ expression, $e^+.ps$, $e^+.ls$, $e^+.cs$, $e^+.as$ and $e^+.ses$ are sets of Points, Lines, Circuits, Areas and subPLCA expressions, respectively, and $e^+.om \in e^+.cs$ is the outermost circuits of the whole figure.

definition 5 *(element) (i) Let p, l, c and a be Point, Line, Circuit and Area, respectively. If $p \in l.ps$, then p is said to be* en element of l. *If $l \in c.ls$, then l is said to be* en element of c. *If $c \in a.ls$, then c is said to be* en element of a. *(ii) Let o_1, o_2 and o_3 are either Point, Line, Circuit or Area. If o_1 is an element of o_2 and o_2 is an element of o_3, then o_1 is an element of o_3.*

Consistency of PLCA+

definition 6 *(consistency) A PLCA+ expression e^+ is said to be* consistent *iff the following constraints are satisfied.*

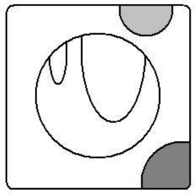


Figure 9: A figure corresponding to such SubPLCA that does not satisfy the uniqueness of $se.iom$

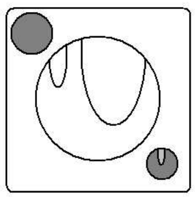


Figure 10: A figure corresponding to such SubPLCA that does not satisfy the uniqueness of $se.oa$

Constraint on Point-Line Each Point belongs to some line. Each Point in $l.ps$ should belong to $e^+.ps$ where l belongs to $e^+.ls$. For each SubPLCA, the same constraints are put.

Constraint on Line-Circuit Each Line belongs to exactly two distinct Circuits. Each Line in $c.ls$ should belong to $e^+.ls$ where c belongs to $e^+.cs$. For each SubPLCA, the same constraints are put.

Constraint on Circuit-Area For any $Circuit$ other than $outermost$ and $suboutermost$ belongs to exactly one Area. Each Circuit in $a.cs$ should belong to $e^+.cs$ where a belongs to $e^+.as$. For each SubPLCA, the same constraints are put.

Due to these three constraints, neither isolated lines nor points are allowed.

Constraint on SubPLCA There exist the unique $se.area$, $se.som$, $se.iom$, $se.oca$ and $se.oa$ for each subPLCA. Moreover, the extracted area $se.area$ and the background area $se.oa$ should be line-connected.

The uniqueness of $se.iom$ eliminates the case that is shown in Figure 9. The uniqueness of $se.oa$ eliminates the case that is shown in Figure 9. The line-connectedness of $se.area$ and $se.oa$ eliminates the case that is shown in Figure 11.

Planarity of PLCA+

We have investigated the planarity condition for PLCA expression (Takahashi and Sumitomo 2008). For a PLCA+ expression, since each SubPLCA se and e^+ are also regarded as PLCA expressions, they satisfy this condition.

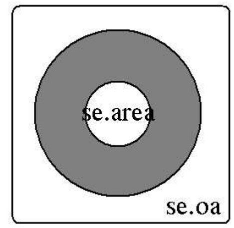


Figure 11: A figure corresponding to such SubPLCA that does not satisfy the line-connectedness between $se.area$ and $se.oa$

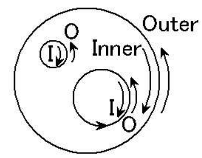


Figure 12: Inner Circuit and Outer Circuit

$$
\begin{aligned}
&|e^+.ps| - |e^+.ls| - |e^+.cs| + 2|e^+.as| = 0\\
&\forall se \in e^+.ses\big(\\
&\quad |se.ps| - |se.ls| - |se.cs| + 2|se.as| = 0\big)
\end{aligned}
$$

In addition, there are more constraints for planarity, between Area and SubPLCA, and between SubPLCAs.

Inner and Outer Before presenting the conditions, we introduce the concept of Inner and Outer.

First, we define the predicates ioc(is Outer Circuit) and iic(is Inner Circuit) for a Circuit c as follows.

definition 7 *(Inner/Outer Circuit)*

$$
ioc(c) = \begin{cases}
 & c = e^+.om \\
 & \exists se \in e^+.ses(c = se.som) \\
true & \forall l^* \in c.ls\big(l^*_{re} \in c'.ls \wedge \neg ioc(c')\big) \\
 & \quad \forall a \in \{a | c \in a.cs\}\Big(\\
 & \qquad \exists c' \in a.cs \setminus \{c\}\big(\neg ioc(c')\big)\Big) \\
false & otherwise
\end{cases}
$$

$$
iic(c) = \begin{cases} true & \neg ioc(c) \\ false & otherwise \end{cases}
$$

Intuitively, Outer Circuit means the Circuit which encircles the outside of an Area, while Inner Circuit means the one which encircles the inside (Figure 12). For a consistent PLCA expression, it is decidable whether a Circuit is Inner Circuit or Outer Circuit (Takahashi and Sumitomo 2008).

Next, we define the predicate io(Inner Object) for components o_1 and o_2 as follows.

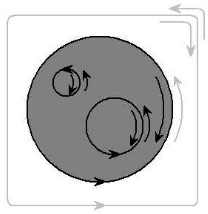


Figure 13: Inner Objects in an Area

definition 8 *(Inner Object)*

$$io(o_1, o_2) = \left\{ \begin{array}{l} o_1 \in As \wedge o_2 \in o_1.cs \\ o_1 \in Cs \wedge o_2 \in o_1.ls \\ o_1 \in Ls \wedge o_2 \in o_1.ps \\ iic(o_1) \wedge o_1 \in o_2.cs \\ \quad ioc(o_1) \wedge l^* \in o_1.ls \\ \qquad \wedge \exists c \in o_2.cs(l^*_{re} \in c.ls) \\ io(o_1, o_3) \wedge io(o_3, o_2) \end{array} \right.$$

Let o_1 and o_2 are either Point, Line, Circuit or Area. Intuitively, $io(o_1, o_2)$ is $true$ if o_2 is in the inside of o_1, and $false$, otherwise (Figure 13).

Constraint on Area-SubPLCA For each Area a other than the background area in an se, there exists an Area in some se' whose parent Area is a. There is no child Area of background areas. These constraints are formalized as follows.

Let $As_{se.oa} = \bigcup_{se \in e^+.ses} se.oa$.

$$\forall a \in As \setminus As_{se.oa} \Big(\big|\{se | se.area = a\}\big| = 1 \Big)$$

$$\forall a \in As_{se.oa} \left(\begin{array}{l} \big|\{se | a \in se.as\}\big| = 1 \\ \wedge a \notin \bigcup_{se \in e^+.ses} se.area \\ \wedge a \notin e^+.as \end{array} \right)$$

An Area a and all of its Inner Objects are included in the inside of the background area of the SubPLCA of a. This constraint is formalized as follows.

$$se.area = a \quad (1)$$

$$se.ps \supseteq \left\{ p \,\middle|\, \begin{array}{l} \big(io(a, p) \wedge p \in Ps\big) \\ \vee \exists l \in \{l | l^* \in se.som.ls\}(p \in l.ps) \end{array} \right\} \quad (2)$$

$$se.ls \supseteq \Big\{ l \Big| \big(io(a, l) \wedge l \in Ls\big) \vee l^* \in se.som.ls \Big\} \quad (3)$$

$$se.cs \supseteq \left\{ c \,\middle|\, \begin{array}{ll} & io(a, c) \wedge c \in Cs \\ \vee & c = se.som \\ \vee & c = se.iom \\ \vee & c = se.oca \end{array} \right\} \quad (4)$$

$$se.as \supseteq \left\{ a' \,\middle|\, \begin{array}{ll} & io(a, a') \wedge a' \in As \\ \vee & c = se.oa \\ \vee & c = se.area \end{array} \right\}$$

Constraint on SubPLCA-SubPLCA Consider the SubPLCAs of areas which are line-connected. The line shared by these areas should not connect to the each background area of the SubPLCAs (Figure 14). It reflects the fact that when the areas are line-connected, the one is convex and the other is concave.

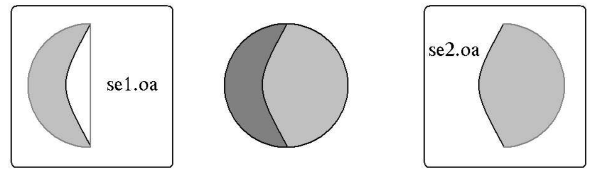


Figure 14: Convexity of Line and constraints SubPLCAs

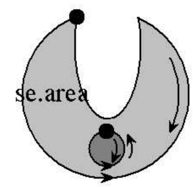


Figure 15: Inside of Area a

$$\forall l^* \in se_1.oca.ls(l^*_{re} \notin se_2.oca.ls)$$
$$(se_1, se_2 \in e^+.ses \wedge se_1 \neq se_2)$$

3. Generation of PLCA+

Making PLCA+ from PLCA and Figure

For a given figure F in a two-dimensional plane, we have already described the generation of PLCA expression e for F (Takahashi and Sumitomo 2008). Here, we describe the generation of PLCA+ expression e^+ from F and e. Note that A and A' denote the part in the figure F corresponding to the expression a and the convex-hull of A. In this algorithm, for each area in F, we prepare the frame for its SubPLCA, make an expression corresponding to the inside of A and that corresponding to the background in the frame, and combine these expressions. If the extracted area has a concave part, this process is recursively repeated.

We show the outline of the algorithm.

Initially, $e^+.ps, e^+.ls, e^+.cs, e^+.as$ are set to be $\{\}$.

function : $generate(F, e)$

(a) Set $e^+.ps = e^+.ps \cup e.ps$, $e^+.ls = e^+.ls \cup e.ls$, $e^+.cs = e^+.cps \cup e.cs$ and $e^+.as = e^+.as \cup e.as$.

(b) Set $e^+.ses = \{\}$ and $Areas = e^+.as$.

(c) Repeat **(d)** until $Areas = \{\}$.

(d) Pick up an arbitrary Area a from $Areas$, and proceed the followings.

(d1) The inside of the Area Each element of a is added to $se.ps$, $se.ls$, $se.cs$ and $se.as$, depending on its class. And set $se.area = a$ (Figure 15).

(d2) The outside of the Area Make a SubPLCA expression se which consists of the only one area a' with one Points, one Line, and two Circuits $se.som$ and $se.iom$ (Figure 16).

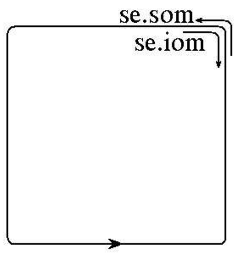


Figure 16: The background in the frame

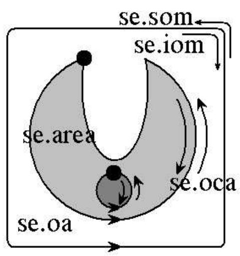


Figure 17: Combining

- **(d3) Combining expressions** Make a new Circuit expression corresponding to the circuit that encircles the outer part of Area a, and add it both to $se.cs$ and to $se.oa.cs$ (Figure 17).
- **(d4) Generating expression for concavity** If A' is not fully occupied by A in F, create the concave part of A by comparing A and A' in F. In this process, Line-division and Area-generation operators (Sumitomo and Takahashi 2007) are used (Figure 18). Otherwise, do nothing.
- **(d5) Updating Areas** Add this se to $e^+.ses$, and add all Areas in $se.areas$ other than $se.area$ and $se.oa$ to $Areas$.

Judgment of Line Convexity

For a given PLCA+ expression, assume that a and a' are the expressions corresponding to the Area A and its convex-hull A' in the figure. If A and A' are matched, that is, A' is fully occupied by A, then A is said to be *convex*, otherwise, it is said to be *concave*.

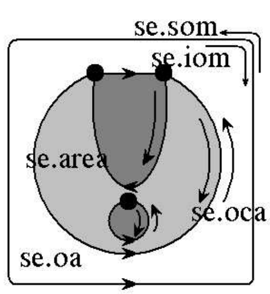


Figure 18: Generation of concavity

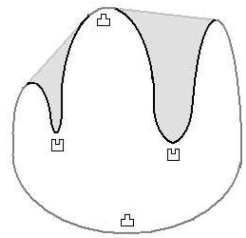


Figure 19: Convexity of Line

The Line expression corresponding to the part that is matched when A and A' are piled is said to be *convex*; otherwise, it is said to be *concave* (Figure 19). Note that the convexity of a Line is determined by viewing from the inside of A and it is inverted by viewing from the outside. For a directed Line l^* which is an element of an Area a, $convex(l^*)$ denotes that Line l is convex from the side of an Area a; and $concave(l^*)$ denotes that Line l is concave from the side of an Area a.

Each line in the figure is a curved one. Therefore, the Line in $se.oca$ is concave, since $se.oca$ corresponds to the outer circuit of the convex-hull. Thus, the following properties hold.

$$convex(l^*) \leftrightarrow concave(l^*_{re})$$
$$\forall se \in e^+.ses\Big(\forall l^* \in se.oca.ls\big(concave(l^*)\big)\Big)$$

An Algorithm for Judging the Convexity of Line We show an algorithm for determining the convexity of Line which is not included in the outermost or sub-outermosts.

function : $getConvexity(l^*)$

Consider the SubPLCA se such that $l^* \in se.ls$ holds.

(a) If $l^* \in se.oca.ls$, then $concave(l^*)$.

(b) If $l^* \notin se.oca.ls$, consider $getConvexity(l^*_{re})$.

- **(b1)** If $convex(l^*_{re})$ is obtained as the result of $getConvexity(l^*_{re})$, then $concave(l^*)$.
- **(b2)** If $concave(l^*_{re})$ is obtained as the result of $getConvexity(l^*_{re})$, then $convex(l^*)$.

The convexity of each Line is decidable by this algorithm (see Appendix).

4. Discussion

There are several works which studied a qualitative shape representation.

In some works, logic based approach is taken, that is, the relationship of the regions are represented using predicates. Gotts provided a qualitative representation for a shape of a region in RCC framework (Gotts 1994). He used a predicate that stands for a connected relation and showed that various types of qualitative shape references can be represented using the Clark's C operator in the first order logic. Cohn proposed a symbolic representation for the shapes of

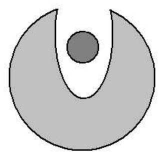


Figure 20: Geometric inside

figures (Cohn 1995). He extended RCC to represent the difference of shapes of regions in the first-order logic. He considered the convexity of a region, and represented the difference of the original region and its convex-hull as the concavity. He represented the subtle qualitative shape difference using the relative positional relationships of the regions appeared as the concave parts. Moreover, he used the hierarchical treatment of the region to represent the complicated shapes. These works show the expressive power of RCC or C operator, however, a new predicate and axioms should be defined every time a new distinction is introduced, and there are no discussion on the well-definedness. Pratt investigates the shape representation in an algebraic manner (Pratt 1999). In PLCA+, we also use the convex-hull and hierarchical treatment of the information on convexity. However, we adopt the representation in a kind of object oriented manner, instead of predicates. Moreover, they handled a shape of the single object, and not referred to the connection of the objects. For example, the connection of two objects with the concave part shown in Figure 3 cannot be represented in their methods, while it can be represented in PLCA+.

Another approach for qualitative shape representation is the one that focused on the shape of the lines between regions. In (Museros and Escrig 2004), a line is divided into several segments, and the properties such as qualitative shape, angle or size of the segments are represented. In this method, lots of information is necessary even for a single segment. In (Nedas and Egenhofer 2004), a line is also divided into segments, and the relationships of these segments are represented. In (Schlieder 1996), a shape of the line is represented by positional ordering of the points on the line. In these methods, the position on which the point is set is difficult, and lots of redundancy appears depending on the positions. Moreover, it is impossible to represent the location of the objects shown in Figure 20 called the geometric inside by using the information added on lines. That is the reason why we use the convex-hull in PLCA+. It is possible to represent it in PLCA+ by a little extension, although current PLCA+ does not handle such a case.

5. Conclusion

We have proposed a qualitative spatial representation PLCA+, an extended PLCA to handle the qualitative shape representation. It is based on the convex-hull. We formalized the definition of PLCA+, gave an algorithm from a figure to generate the PLCA+ expression.

PLCA provides a symbolic expression for figures in a two-dimensional plane representing the connection patterns of regions using the simple components of Point, Line, Circuit and Area, and PLCA+ can represent the convexity of regions in addition. We can reason about the convexity of a single region, and the connectivity of multiple regions with the information on convexity.

We have also shown the properties that should be satisfied by the PLCA+ expression generated from the figure. These conditions are considered to be necessary and sufficient conditions of the planarity of PLCA+ expression. In future, we are considering to prove this property.

Extension of PLCA+ is also under consideration. We would like to treat the figures using straight lines, and also treat the other relationships of regions including geometric inside.

References

A. G. Cohn. A hierarchical representation of qualitative shape based on connection and convexity. In *Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science (COSIT95)*, Springer-Verlag, pages 311–326, 1995.

A. G. Cohn and S. M. Hazarika. Qualitative spatial representation and reasoning: an overview. *Fundamental Informaticae*, 46(1-1):1–29, 2001.

M. J. Egenhofer. Reasoning about Binary Topological Relations. *Second Symposium on Large Spatial Databases*, pages 143–160, Springer-Verlag, 1991.

M. Egenhofer and R. Franzosa. Point-Set Topological Relations. *International Journal of Geographical Information Systems 5*, 1991.

M. Egenhofer and R. Franzosa. On the equivalence of topological relations. *International Journal of Geographical Information Systems*, 9(2):133–152, 1995.

N. Guarino, S. Borgo and C. Masolo. A pointless theory of space based on strong connection and congruence. In *Proceedings of the Fifth International Conference on Principles of Knowledge Representation and Reasoning (KR96)*, pages 220–229. Morgan Kaufmann Publishers, 1996.

N. M. Gotts, How far can we 'C'? defining a doughnut using connection alone. In *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning (KR94)*, pages 246–257, Morgan Kaufmann Publishers, 1994.

D. Randell, Z. Cui, , and A. Cohn. A spatial logic based on regions and connection. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR92)*, pages 165–176. Morgan Kaufmann Publishers, 1992.

L. Museros and M .T .Escrig. A qualitative theory for shape representation and matching. In *18th International Workshop on Qualitative Reasoning*. 2004.

K. Nedas and M. Egenhofer Splitting ratios: Metric details of topological line-line relations. In *Seventeenth International Florida Conference on Artificial Intelligence (FLAIRS04)*, 2004.

I. Pratt. Qualitative spatial representation languages with convexity *Journal of Spatial Cognition and Computation*, 1:181–204, 1999.

D. Randell, Z. Cui and A. Cohn. A spatial logic based on regions and connection. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning (KR92)*, pages 165–176. Morgan Kaufmann Publishers, 1992.

J. Renz. *Qualitative Spatial Reasoning with Topological Information*. LNAI-2293. Springer Verlag, 2002.

C. Schlieder. Qualitative shape representation. In A. Frank(Ed.), *Spatial Conceptual Models for Geographic Objects with Undetermined Boundaries*, pages 123–140. London:Taylor & Francis, 1996.

O. Stock(ed.), editor. *Spatial and Temporal Reasoning*. Kluwer Academic Press, 1997.

T. Sumitomo and K. Takahashi. The qualitative treatment of spatial data. In *International Journal on Artificial Intelligent Tools* 16(4), pages 661–682, 2007.

K. Takahashi and T. Sumitomo. A framework for qualitative spatial reasoning based on the connection patterns of regions. In *IJCAI-05 Workshop on Spatial and Temporal Reasoning*, pages 57–62, 2005.

K. Takahashi and T. Sumitomo. On Embedding a qualitative representation in a two-dimensional plane. *Spatial Congnition and Computation*, to appear.

Appendix. Decidability of Line Convexity

For a consistent planar PLCA+ expression e^+, let Ls_{se} be $Ls \setminus \bigcup_{se \in e^+.ses}\{l|l^* \in se.som.ls\}$, which is equivalent to $\bigcup_{se \in e^+.ses} se.ls \setminus \{l|l^* \in se.som.ls\}$.

We show that the convexity of each Line in each SubPLCA is decidable. We prove this by the induction on the tree structure of SubPLCA shown in Section 2.

lemma 1 *The convexity of each Line in SubPLCA se at the leaf node is decidable.*

$$\forall l^* \in Ls_{leafnode}\big(convex(l^*) \wedge concave(l^*_{re})\big).$$

Proof)

In this case,

$$se.as = \{se.oa, se.area\}$$

holds, since the SubPLCA of $se.area$ is equivalent to se itself and there is no child node.

The number of the other components of se are determined since it is consistent and planer. Therefore, the PLCA+ expression for se is as follows (Figure 21):

$$\begin{array}{ll}
se.ps = \{p_1, p_2\} & l_1.ps = [p_1, p_1] \\
se.ls = \{l_1, l_2\} & l_2.ps = [p_2, p_2] \\
se.cs = \{c_1, c_2, c_3, c_4\} & c_1.ls = [l_1^+] \\
se.as = \{a_1, a_2\} & c_2.ls = [l_1^-] \\
se.area = a_2 & c_3.ls = [l_2^+] \\
se.som = c_1 & c_4.ls = [l_2^-] \\
se.iom = c_2 & a_1.cs = \{c_2, c_3\} \\
se.oa = a_1 & a_2.cs = \{c_4\} \\
se.oca = c_3 &
\end{array}$$

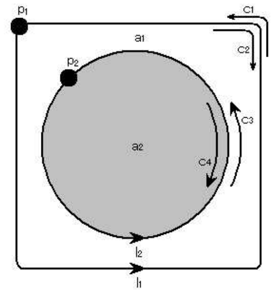


Figure 21: The SubPLCA for a leaf node

In this case, it is sufficient to determine the convexity of the directed Lines of l_2^+ and l_2^-.

$concave(l_2^+)$ holds since $l_2^+ \in se.oca.ls$. Therefore, $convex(l_2^-)$ holds. Thus, the lemma holds.
Q.E.D.

lemma 2 *Assume that the convexity of each line in all the SubPLCAs $se_i(1 \le i \le n)$ which are the SubPLCAs of Areas in $se.as$ is decidable. Then, the convexity of each line in SubPLCA se is decidable.*

Proof)

Let $Ls_{internal} = se.ls \setminus \{l|l^* \in se.som.ls\}$. It is sufficient to prove that

$$\forall l^* \in Ls_{internal}\big(convex(l^*) \wedge concave(l^*_{re})\big)$$

$Ls_{internal}$ can be divided into three subsets: the directed Lines in $se.oca$, the directed Lines belonging to the Circuit in $se.area.cs$, the directed Lines belonging to the Circuit in $se_i.cs$.

$\forall l^* \in se.oca.ls\big(concave(l^*)\big)$ holds and the convexity of each Line in se_i is decidable from the induction hypothesis. The shape of the Line belonging to the circuit in $se.area.cs$ is determined by the definition of line convexity. It means that the information of the convexity of all Lines in $Ls_{internal}$ is obtained. Moreover, the convexity of each Line is decidable, since e^+ is a consistent planer expression. Therefore, the lemma holds.
Q.E.D.

theorem 1 *For a consistent planar PLCA+ expression, the convexity of each Line in the expression is decidable.*

Proof) The theorem holds from the above two lemmas.
Q.E.D.