Formalization of a Surface Subdivision Allowing a Region with Holes without Coordinates

June 29, 2016

Kazuko Takahashi, Sosuke Moriguchi, Mizuki Goto
Kwansei Gakuin University, Japan

1

Background

- A number of studies on symbolic treatment of geometric/topological properties
- Some of them use proof assistants on their formalization
  - Give a certified formalization, find drawbacks in the pen-and-paper proofs
- **Surface subdivision** is one of the topics of these studies

2

Surface subdivision

- Embedding of a surface with a finite numbers of connected regions
- Related topics:
  - Graph embedding, four color theorem, Jordan's simple curve theorem
- Data structures:
  - Doubly connected edge list (e.g. [1])
  - Hypermap (e.g.[2])

[1] de Berg, M. et al., O.:"Computational Geometry, Springer-Verlag (1997).
[2] Dufourd, J.-F. and Bertot, Y.: "Formal study of plane Delaunay Triangulation." in *ITP 2010*, pp.211-226, LNCS, vol.6172, Springer-Verlag (2010).

3

Surface subdivision

- Embedding of a surface with a finite numbers of co
- Re
  - s
- Da
  -
  - formalization using doubly connected edge list (e.g. [2])

Another formalization of a surface subdivision of the 2D plane

[1] Dufourd, J.~-F. and Bertot, Y.: "Formal study of plane Delaunay Triangulation." in *ITP 2010*, pp.211-226, LNCS, vol.6172, Springer-Verlag (2010).
[2] de Berg, M. et al., O.:"Computational Geometry, Springer-Verlag (1997).

4

Outline

1. The target issue
2. A PLCA expression
3. Encoding PLCA in Coq
4. Representation of subdivision using PLCA
5. Conclusion

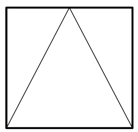
Outline

1. The target issue
2. A PLCA expression
3. Encoding PLCA in Coq
4. Representation of subdivision using PLCA
5. Conclusion

Surface subdivision (1)

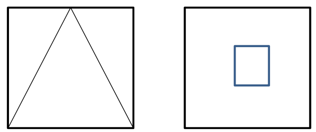
- Determining the locations of regions for a given set of points

without holes
(e.g. Delauney triangulation)



Surface subdivision (2)

- Determining the locations of regions for a given set of points

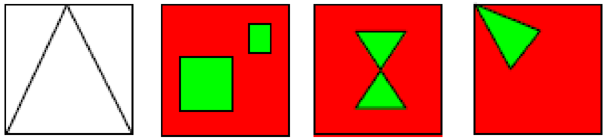


without holes
(e.g. Delauney triangulation)

+ with holes

- Disconnected components are regarded as regions embedded into the holes of another region

Examples of surface subdivisions



(a) (b) (c) (d)

a region with holes

a hole

(a) triangulation

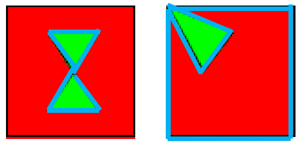
(b) a region with holes

(c) a region with holes connected with a point

(d) a region with a hole connected to itself with a point

9

Border of regions



(c) (d)

The border of a region is not always a Jordan curve

a simple closed curve without a self-loop

10

The target surface subdivision

- Consider a surface subdivision as a configuration in which both side of each line always belong to distinct regions

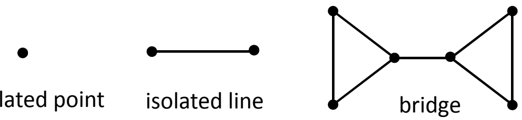
a representation of a figure showing its geometric or topological characteristics

- The configuration admits disconnected components

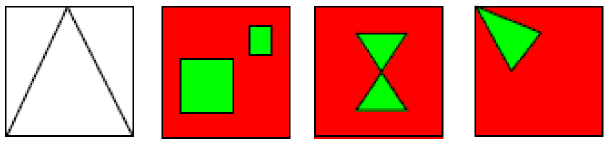
11

Allowed/disallowed configurations

Disallowed configurations



Allowed configurations



(a) (b) (c) (d)

The target surface subdivision

- Consider a surface subdivision as a configuration in which both side of each line always belong to distinct regions

  a representation of a figure showing its geometric or topological characteristics

- The configuration admits disconnected components

The idea

- Consider a surface subdivision as a configuration in which both side of each line always belong to distinct regions

  a representation of a figure showing its geometric or topological characteristics

- The configuration admits disconnected components

  ⇩

  PLCA planarity

The goal

- PLCA
  - Data structure in which all incidence relations between objects are stored [3]
  - Topological and geometric aspects can be distinguished
  - Not use coordinates
- Formalization with PLCA to represent a surface subdivision that allows disconnected components

[3] Takahashi, K. et al. "On embedding a qualitative representation in a two-dimensional plane." in *Spatial Cognition and Computation*, 8(1-2):4-26 (2008).

Outline

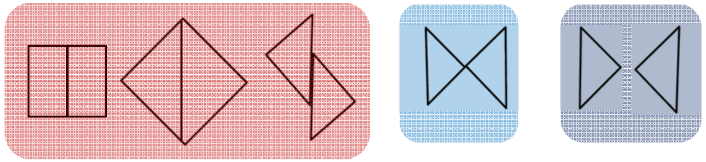
PLCA (1)

Symbolic representation for a spatial data focusing on connection patterns of regions

- Figures with the same connected patterns are regarded as the same

Originally designed to give a qualitative representation of a spatial object

- No numerical data

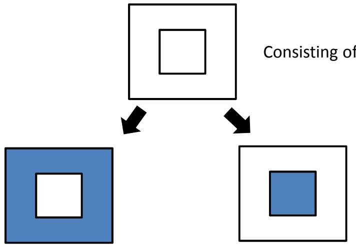


PLCA (2)

No pair of areas has a part in common

-- The entire figure is divided into disjoint areas

-- An area may have a hole



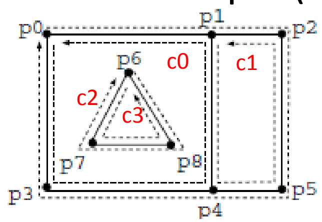
Data structure of PLCA

< P,L,C,A,o>

- P: a set of points
- L: a set of lines
  - a line is a pair of points (p_1,p_2), where each $p_i \in P$
  - each line l has a direction:
    $l^+=(p_1,p_2)$ ➔ $l^-=(p_2,p_1)$
- C: a set of circuits
  - a circuit is a list of lines [l_1,...,l_n] where each $l_i \in L$, $p_1=p_n$, where $l_i=(p_i,p_{i+1})$
- A: a set of areas
  - an area is a set of circuits {c_1,...,c_n} where each $c_i \in C$
- o: outermost, $o \in C$

Use **list** to implement most data structures

PLCA example (1)



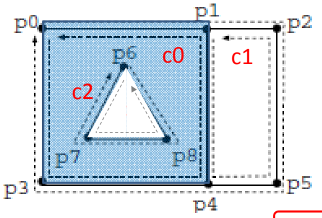
$$P = \{p_0, p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8\}$$
$$L = \{l_0, l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8, l_9\}$$
$$C = \{o, c_0, c_1, c_2, c_3\}$$
$$A = \{a_0, a_1, a_2\}$$

$$l_0 = (p_0, p_1)$$
$$l_1 = (p_1, p_2)$$
$$l_2 = (p_0, p_3)$$
$$l_3 = (p_1, p_4)$$
$$l_4 = (p_2, p_5)$$
$$l_5 = (p_3, p_4)$$
$$l_6 = (p_4, p_5)$$
$$l_7 = (p_6, p_7)$$
$$l_8 = (p_7, p_8)$$
$$l_9 = (p_8, p_6)$$

$$c_0 = [l_2^+, l_5^+, l_3^-, l_0^-]$$
$$c_1 = [l_3^+, l_6^+, l_4^-, l_1^-]$$
$$c_2 = [l_9^-, l_8^-, l_7^-]$$
$$c_3 = [l_7^+, l_8^+, l_9^+]$$
$$o = [l_0^+, l_1^+, l_4^+, l_6^-, l_5^-, l_2^-]$$

$$a_0 = \{c_0, c_2\}$$
$$a_1 = \{c_1\}$$
$$a_2 = \{c_3\}$$

PLCA example (2)



$P = \{p_0, p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8\}$
$L = \{l_0, l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8, l_9\}$
$C = \{o, c_0, c_1, c_2, c_3\}$
$A = \{a_0, a_1, a_2\}$

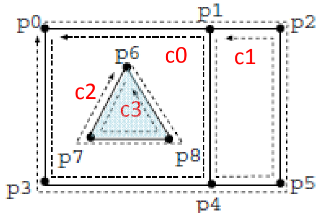
$l_0 = (p_0, p_1)$
$l_1 = (p_1, p_2)$
$l_2 = (p_0, p_3)$
$l_3 = (p_1, p_4)$
$l_4 = (p_2, p_5)$
$l_5 = (p_3, p_4)$
$l_6 = (p_4, p_5)$
$l_7 = (p_6, p_7)$
$l_8 = (p_7, p_8)$
$l_9 = (p_8, p_6)$

$c_0 = [l_2^+, l_5^+, l_3^-, l_0^-]$
$c_1 = [l_3^+, l_6^+, l_4^-, l_1^-]$
$c_2 = [l_9^-, l_8^-, l_7^-]$
$c_3 = [l_7^+, l_8^+, l_9^+]$
$o = [l_0^+, l_1^+, l_4^+, l_6^-, l_5^-, l_2^-]$

$a_0 = \{c_0, c_2\}$
$a_1 = \{c_1\}$
$a_2 = \{c_3\}$

21

PLCA example (3)



$P = \{p_0, p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8\}$
$L = \{l_0, l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8, l_9\}$
$C = \{o, c_0, c_1, c_2, c_3\}$
$A = \{a_0, a_1, a_2\}$

$l_0 = (p_0, p_1)$
$l_1 = (p_1, p_2)$
$l_2 = (p_0, p_3)$
$l_3 = (p_1, p_4)$
$l_4 = (p_2, p_5)$
$l_5 = (p_3, p_4)$
$l_6 = (p_4, p_5)$
$l_7 = (p_6, p_7)$
$l_8 = (p_7, p_8)$
$l_9 = (p_8, p_6)$

$c_0 = [l_2^+, l_5^+, l_3^-, l_0^-]$
$c_1 = [l_3^+, l_6^+, l_4^-, l_1^-]$
$c_2 = [l_9^-, l_8^-, l_7^-]$
$c_3 = [l_7^+, l_8^+, l_9^+]$
$o = [l_0^+, l_1^+, l_4^+, l_6^-, l_5^-, l_2^-]$

$a_0 = \{c_0, c_2\}$
$a_1 = \{c_1\}$
$a_2 = \{c_3\}$

22

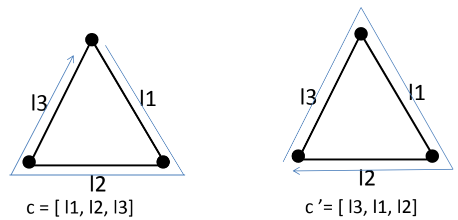
Equivalence on PLCA expressions (1)

Sensitive on the equivalence of PLCA expressions in its implementation

23

Equivalence on PLCA expressions (2)

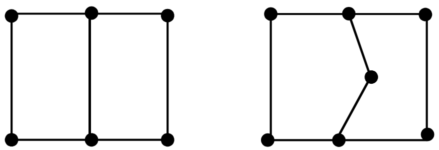
Different expressions may stand for the same object in a symbolic expression



24

Equivalence on PLCA expressions (3)

The same subdivision should be handled differently in the symbolic expression



Equivalence on PLCA expressions (4)

Equivalence relation over PLCA expressions

P, L, C and A are equivalent to their permutations, respectively

Each element of L is equivalent to its inverse

Each element of C is equivalent to its rotation

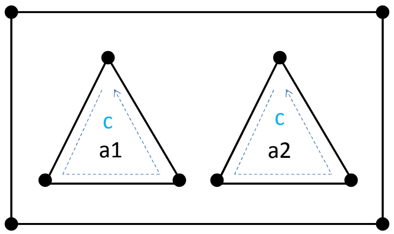
Each element of A is equivalent to its permutation

Consistent PLCA (1)

To permit only reasonable expressions

- putting topological constraint
- avoiding duplication in a list

a1 = { c }, a2 = { c }



Consistent PLCA (2)

Definition 3 (consistent PLCA). *Let $\langle P, L, C, A, o \rangle$ be a PLCA expression. The PLCA expression is said to be a* consistent PLCA expression *iff it satisfies all of the following conditions.*

- *For each point $p \in P$, there exists $l \in L$ such that $p \in l$.*
- *For each line $l \in L$, all points in l are in P.*
- *For each line $l \in L$, there exist circuits $c, c' \in C$ such that $l^+ \in c$ and $l^- \in c'$[3].*
- *For each circuit $c \in C$, each line in c or its inverse line is in L.*
- *For each circuit $c \in C$ except for o, there exists an area $a \in A$ such that $c \in a$.*
- *For each area $a \in A$, all circuits in a are in C.*
- *For any area $a \in A$, $o \notin a$.*
- *Each point $p \in P$ appears only once in P.*
- *Each line $l^+ \in L$ appears only once in L and $l^- \notin L$.*
- *Each circuit $c \in C$ appears only once in C and any rotated circuits other than c do not appear in C.*
- *Each area $a \in A$ appears only once in A and any equivalent area other than a does not appear in A.*

Consistent PLCA (3)

There does not exist more than one area that contains the same circuit

- For each point $p \in P$, there exists $l \in L$ such that $p \in l$.
- For each line $l \in L$, all points in l are in P.
- For each line $l \in L$, there exist circuits $c, c' \in C$ such that $l^+ \in c$ and $l^- \in c'$[3].
- For each circuit $c \in C$, each line in c or its inverse line is in L.
- For each circuit $c \in C$ except for o, there exists an area $a \in A$ such that $c \in a$.
- For each area $a \in A$, all circuits in a are in C.
- For any area $a \in A$, $o \notin a$.
- Each point $p \in P$ appears only once in P.
- Each line $l^+ \in L$ appears only once in L and $l^- \notin L$.
- Each circuit $c \in C$ appears only once in C and any rotated circuits other than c do not appear in C.
- Each area $a \in A$ appears only once in A and any equivalent area other than a does not appear in A.

29

Outline

1. The target issue
2. A PLCA expression
3. Encoding PLCA in Coq
4. Representation of subdivision using PLCA
5. Conclusion

30

Computational model of PLCA

- Inductively construct a PLCA expression using four constructors
- Intuitively,
  - Each constructor divides an existing area
- Symbolically,
  - It deletes the existing one area and adds two new areas, and reconfigures all the related objects

31

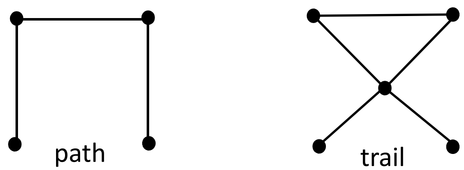
Data structure: path and trail

- A list of lines [$(p_0,p_1),(p_1,p_2),\ldots,(p_{n-1},p_n)$] (n>0)
- path: $p_0,\ldots,p_n$ are distinct

  used as a divider of a circuit to make a new area
- trail: $(p_0,p_1),\ldots,(p_{n-1},p_n)$ are distinct

  p0,…,pn may not be distinct

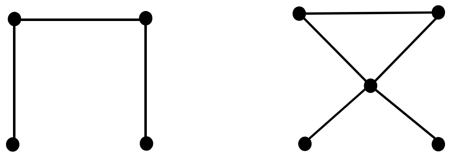
  used to show a property of a segment of a circuit



32

Closed path and closed trail (1)

Make a closed path/trail by adding a line connecting their start and end points



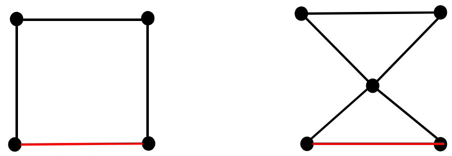
33

Closed path and closed trail (2)

Make a closed path/trail by adding a line connecting their start and end points

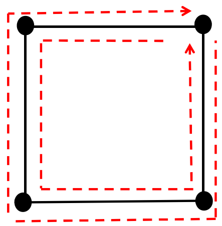
↓

circuit



34

Example of closed path and closed trail (1)

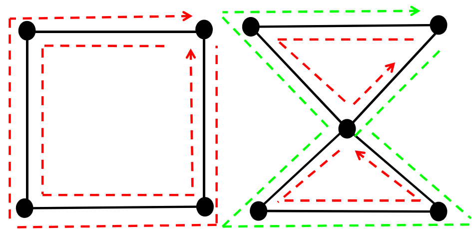


(a)

- - - - closed path
- - - - closed trail

35

Example of closed path and closed trail (2)

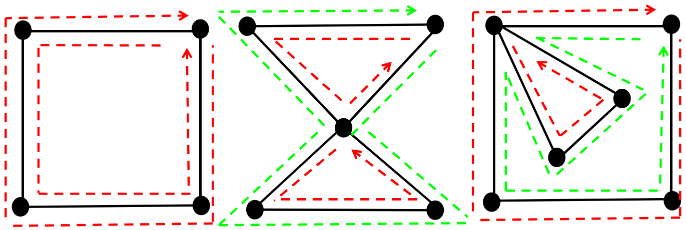


(a) (b)

- - - - closed path
- - - - closed trail

36

Example of closed path and closed trail (3)



(a) (b) (c)

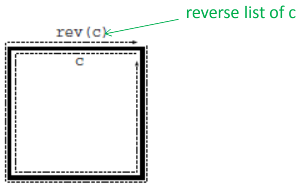
- - - - closed path
- - - - closed trail

Construction of a PLCA expression

- Using four constructors
  - single_loop
  - add_loop
  - add_inpath
  - add_inline

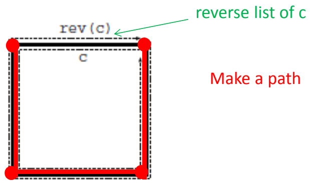
Constructor: single_loop

Make a Jordan curve as an outermost



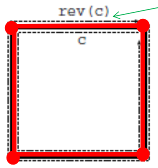
Constructor: single_loop

Make a Jordan curve as an outermost



Constructor: single_loop

Make a Jordan curve as an outermost

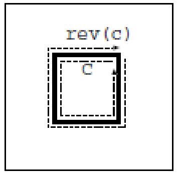


reverse list of c

Make a path
And connect its start and end points

Constructor: add_loop

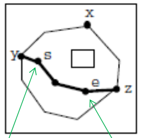
Add a Jordan curve in the specified area



Similar construction with that of single_loop

Constructor: add_inpath

Divide an area by cutting a specific circuit
At least two lines are added

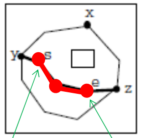


start point

end point

Constructor: add_inpath

Divide an area by cutting a specific circuit
At least two lines are added



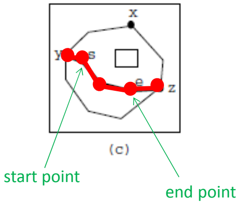
Make a path

start point

end point

Constructor: add_inpath

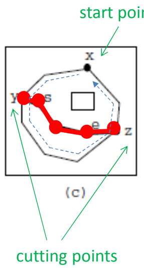
Divide an area by cutting a specific circuit
At least two lines are added



Make a path
And connect its start and end point
to the specified points in the same
existing circuit, respectively

Constructor: add_inpath

Divide an area by cutting a specific circuit
At least two lines are added

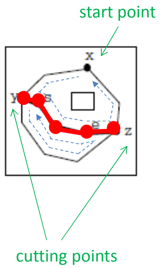


Make a path
And connect its start and end point
to the specified points in the same
existing circuit, respectively

Circuit has a cyclic structure

Constructor: add_inpath

Divide an area by cutting a specific circuit
At least two lines are added



Make a path
And connect its start and end point
to the specified points in the same
existing circuit

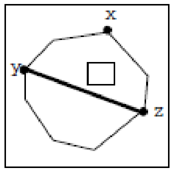
Circuit has a cyclic structure

⬇

Reconfiguration of a circuit should begin with
the original start point

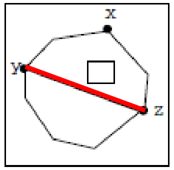
Constructor: add_inline

Make a short cut between two points of an existing circuit
Only one line is added



Constructor: add_inline

Make a short cut between two points of an existing circuit
Only one line is added



Make a line between the specified points
in the same existing circuit

Construction of a PLCA expression

- Using four constructors
  - single_loop
  - add_loop
  - add_inpath
  - add_inline
- Inductive PLCA (IPLCA)

  A PLCA expression constructed in this way

Outline

1. The target issue
2. A PLCA expression
3. Encoding PLCA in Coq
4. Representation of subdivision using PLCA
5. Conclusion

IPLCA and surface subdivision

To show:

IPLCA represents a surface subdivision of a finite 2D plane that allows a region with holes

⇩

Represent the conditions for a subdivision

PLCA planarity

PLCA planarity

- The conditions for a surface subdivision
- Consists of three properties
  - PLCA-constraints
  - PLCA-connectedness
  - PLCA-euler

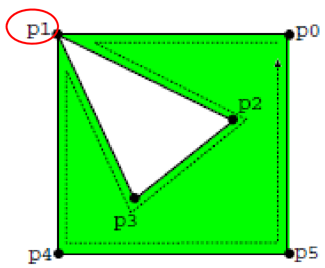
53

PLCA-constraints

- Only the target figure can be expressed
  - No isolated points
  - No bridge between points
  - A finite plane encircled with a Jordan curve
  - Appropriate treatment of a duplicated point on a circuit

54

Treatment of a duplicated point



55

PLCA-connectedness

- No objects including outermost are separated
  - Holding on different types of P,L,C,A
  - Conversion from these data types to a common data type and introduction of a new incidence function

56

PLCA-euler

- Can be embedded in a 2D plane
  - An equality of numbers of the objects P,L,C,A and easily proved

57

Planar PLCA

A planar PLCA

A consistent PLCA expression that satisfies PLCA-constraints, PLCA-connectedness and PLCA-euler

58

Theorems on planarity

Theorem 1 [planarity]

Inductive PLCA is a planar PLCA.

Theorem 2 [planarity preservation]

Let *e,e'* be equivalent PLCA expressions.

If *e* is a planer PLCA, then *e'* is also a planar PLCA.

59

Formal definition of a subdivision with PLCA

For each line,

if it is not included in the outermost,

- then distinct two areas are connected by the line

otherwise,

- there exists only one area that is connected with the outside of the figure by the line

61

Theorem on subdivision

Theorem 3 [subdivision]

A planar PLCA is a subdivision of a finite 2D plane allowing a region with holes


```
Theorem PLCAsubdivision_area :
forall(P : list Point)(L : list Line)(C : list Circuit)(A : list Area)
  (o : Circuit),
 PLCA_planar P L C A o ->
 forall (l : Line),
  In l L
  -> ~LIn l o
  -> exists (c c' : Circuit),
   In c C
   /\ In c' C
   /\ In l c
   /\ In (reverse l) c'
   /\ exists (a a' : Area),
       In a A
       /\ In a' A
       /\ In c a
       /\ In c' a'
       /\ a <> a'.
```

subdivision: case1

```
Theorem PLCAsubdivision_outermost :
forall(P : list Point)(L : list Line)(C : list Circuit)(A : list Area)
  (o : Circuit),
 PLCA_planar P L C A o ->
 forall (l : Line),
  LIn l o
  -> exists (c : Circuit),
   In c C
   /\ LIn l c
   /\ o <> c
   /\ exists (a : Area),
       In a A
       /\ In c a.
```

subdivision: case2

Theorem on subdivision

Theorem 3 [subdivision]

A planar PLCA is a subdivision of a finite 2D plane allowing a region with holes

IPLCA represents a surface subdivision of a finite 2D plane that allows a region with holes

Theorem on subdivision

Theorem 3 [subdivision]

A planar PLCA is a subdivision of a finite 2D plane allowing a region with holes

IPLCA represents a surface subdivision of a finite 2D plane that allows a region with holes

Surface subdivision is formalized with PLCA and proved using Coq

Outline

1. The target issue
2. A PLCA expression
3. Encoding PLCA in Coq
4. Representation of subdivision using PLCA
5. Conclusion

66

Conclusion

- Show a symbolic expression with PLCA for a surface subdivision of a finite 2D plane that allows holes in a region
- All formalization and proofs are performed using a proof assistant Coq (about 12,800 lines)
- Provide a computational model of PLCA
  - Give a certified proof
  - Relate QSR to software science

67

Future works

- To revise implementation in the form not using list
- To implement high level computational geometry algorithms in PLCA and prove them
- To extend the approach to handle more general surface with an arbitrary genus

68

Thank you!

69