2-Level Quasi-Planarity or How Caterpillars Climb (SPQR-)Trees

Giordano Da Lozzo, Giuseppe Di Battista, Fabrizio Frati, and Maurizio Patrignani
Roma Tre University, Rome, Italy
SODA21

王宏藝

Outline

- Introduce 3 linear equivalent problem
  - 2-level quasi-planar drawing
  - (2, 2)-track layout
  - Bipartite 2-page book embedding (B2BE)
- Proof B2BE is NP-complete.
- An linear algorithm of B2BE if we know half of the answer.

2-level quasi-planar drawing

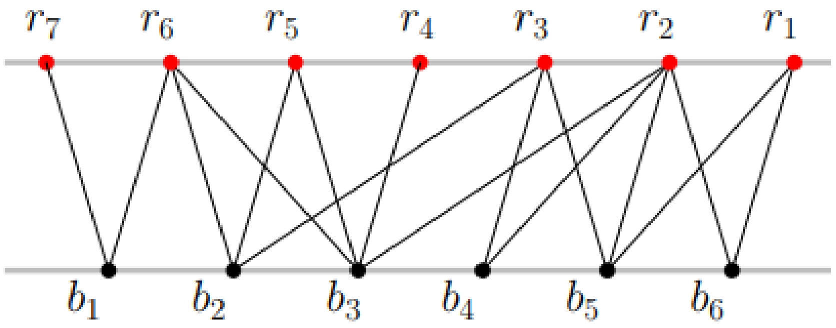
Bipartite

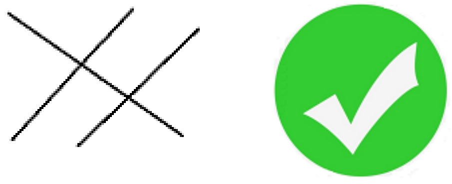
V_b/V_r lie along L_b/L_r

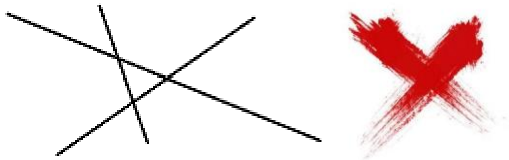
l_b and l_r are parallel

Each edge is drawn in the unbounded plane delimited by l_b and l_r

No three edges in E pairwise cross







(2, 2)-track layout

The vertexes are divided into 2 colors.

The edges are divided into 2 colors.

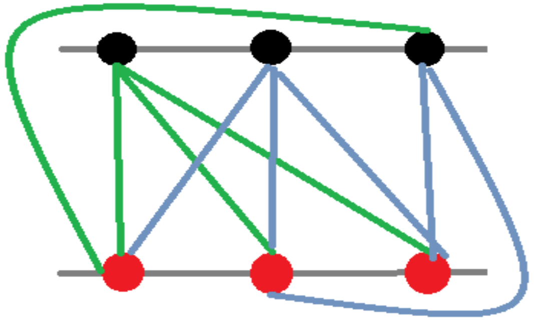
Each color class t has a total order ξ_t.

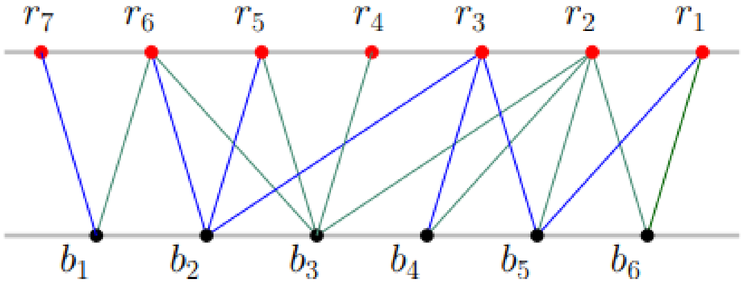
There are no two edges (a,b) and (x,y) with that:

(1) a and x , b and y have the same color

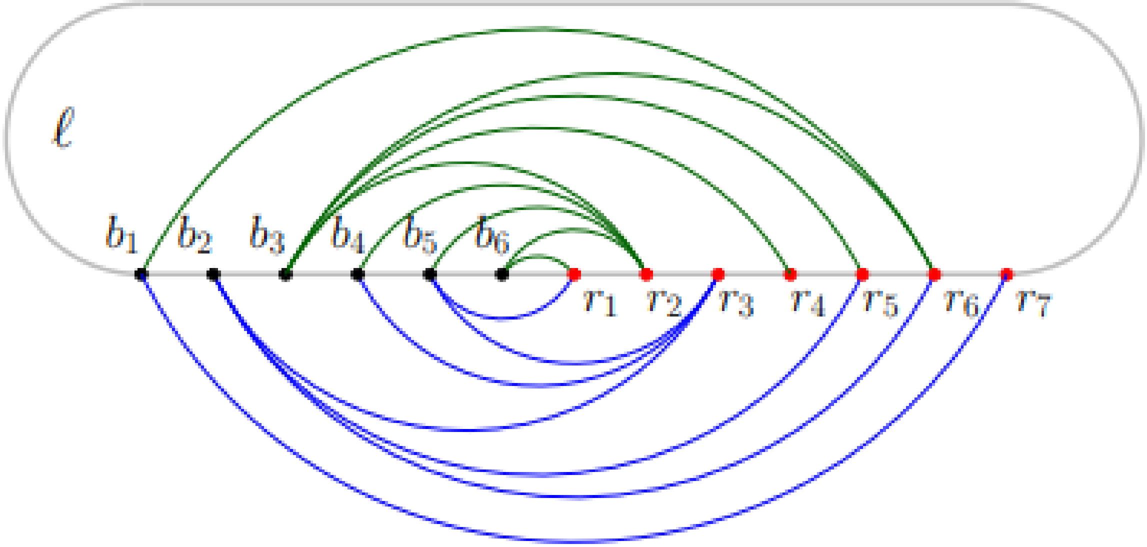
(2) (a,b) and (x,y) have the same color

(3) a $\prec \xi_{C(a)}$ x, and y $\prec \xi_{C(y)}$ b

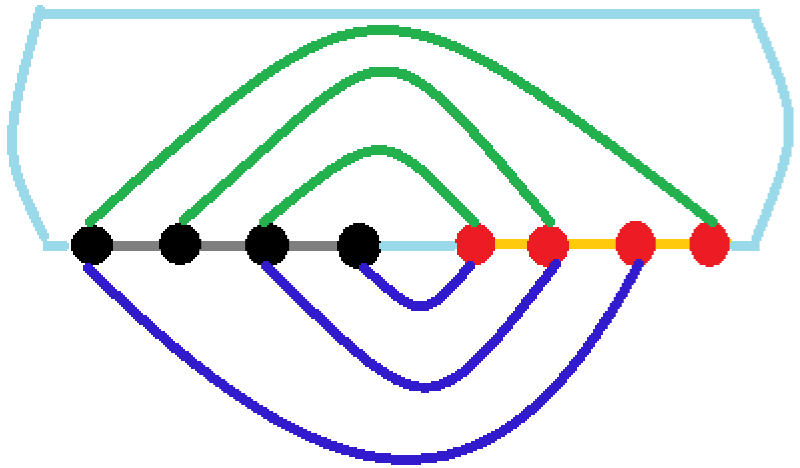




Bipartite 2-page book embedding



Bipartite 2-page book embedding



Rail/track

Black rail/track

Red rail/track

RB-connected track

Finding Rail are equivalence with B2BE.

Proof B2BE is NP-complete

NP: Given the order $(b_1,\ldots,b_m,r_1,\ldots r_p)$, the order can be verifiable in polynomial time.

(1) Connect (b_i,b_{i+1}), i=1, 2…, m-1

(2) Connect (r_j,r_{j+1}), j= 1, 2…, p-1

(3) Connect (b_m,r_1) and(b_1,r_p)

Connect the whole rail

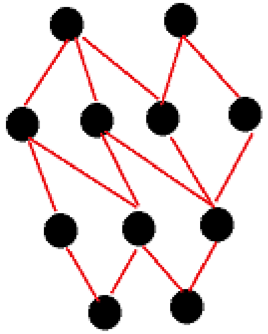
(4) Test the planarity

Above action can de done in linear time.

Hardness: We will use **Special Level Planarity** to prove.

Level Planarity

$H = (V, E)$ admits an embedding under three constraints.



(1) **Vertex Layers:** $V \rightarrow \{1, \ldots, k\}$

(2) **Edge Limit:** The edges only exist between adjacent layers

(3) **No Cross:**

There is a total order to each layer , (u,v) 、 (p,q) are two edges,
u 、 p are in **Layer_i** , v 、 q are in **Layer_{i+1}**,

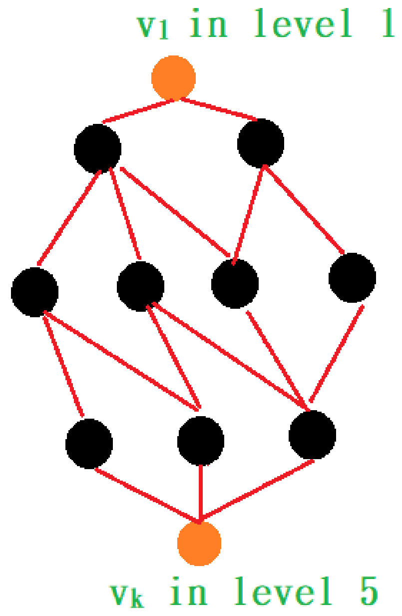
$Ord_i(u) < Ord_i(p)$ if and only if $Ord_{i+1}(v) < Ord_{i+1}(q)$

Special Level Planarity

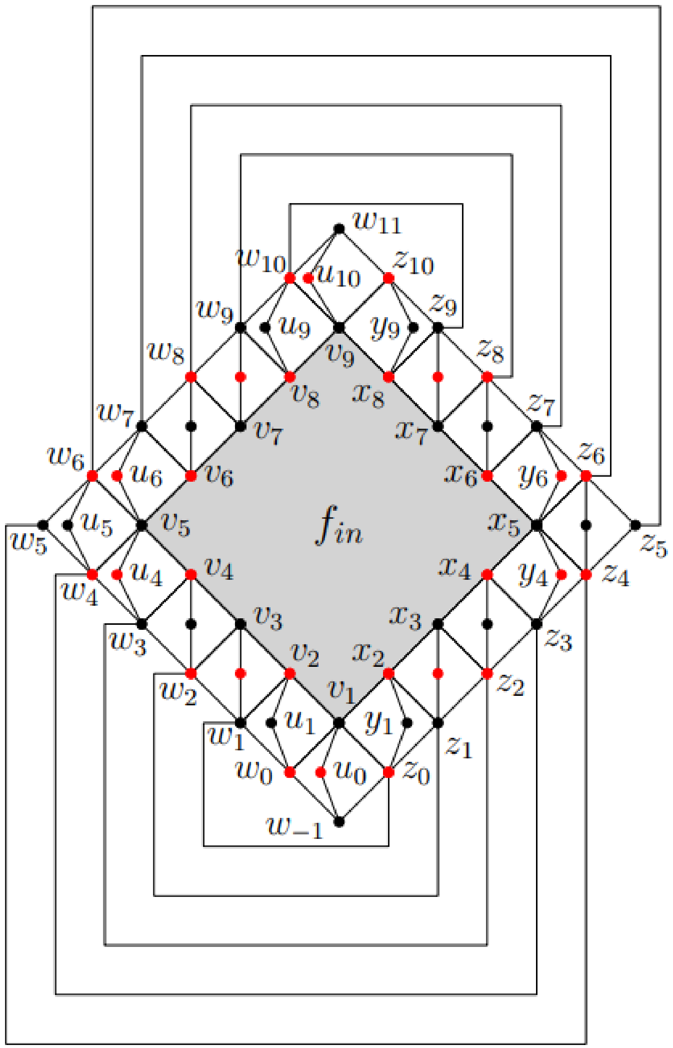
- Leveled Planarity is also NP-complete when the following two properties hold:

  (i) the input also specifies the number k of levels and two special vertices v_1 and v_k such that v_1 is the only vertex on level 1 and v_k is the only vertex on level k.

  (ii) k is constrained to be odd



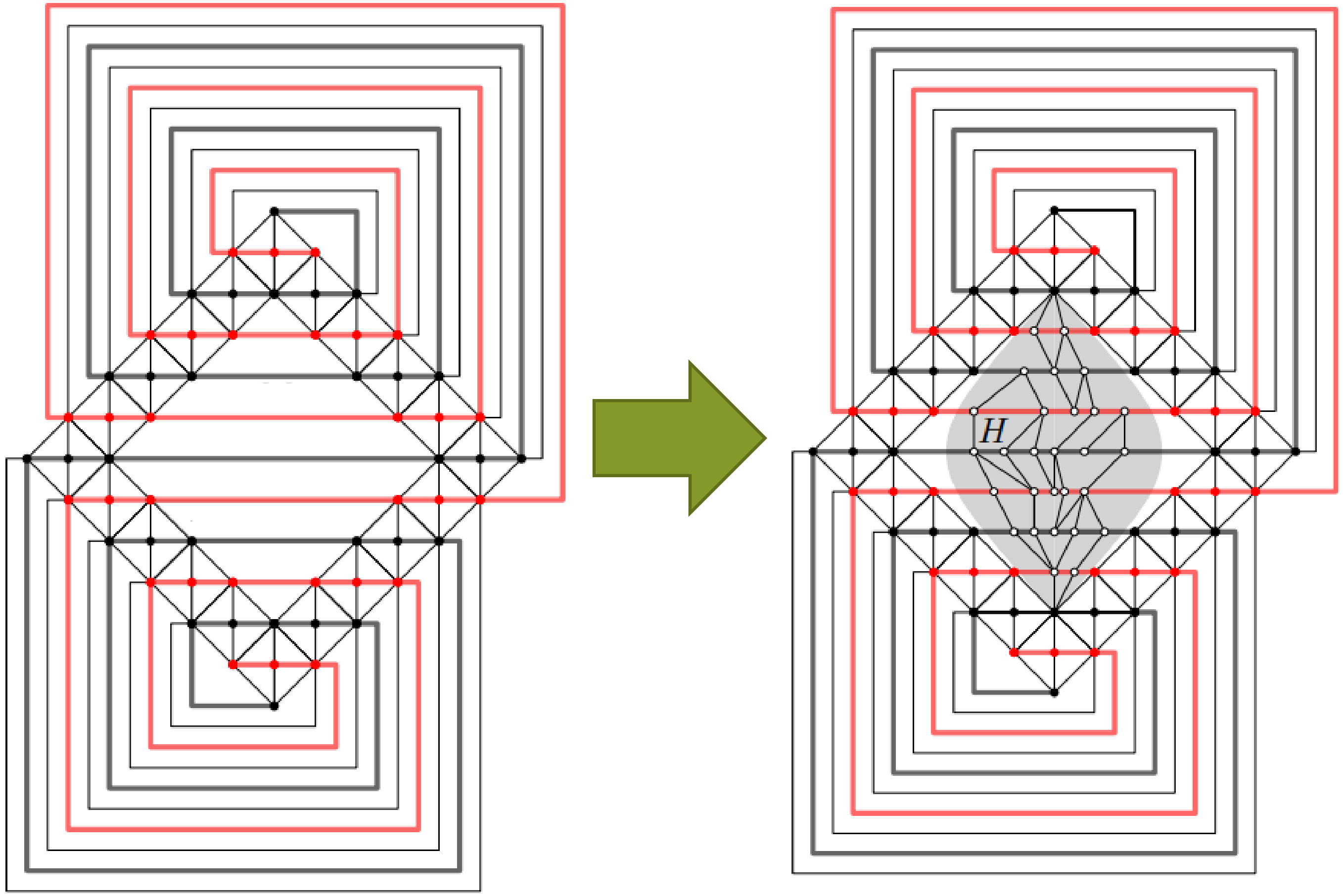
If H admits a special leveled planarity embedding if and only if $H+F_k$ admits a B2BE .



(=>)

F_k admits the Rail, we just only put each layer of H in the tracks of f_{in},

it will not break the planarity of F_k , then we get the Rail of $H+F_k$



(<=) Some track of F_k are fixed

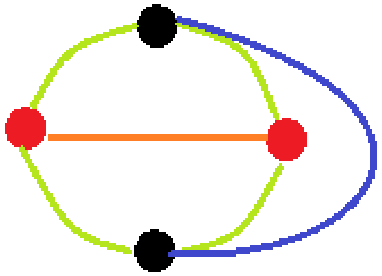
F_k is a bipartite graph which is triconnected

Lemma :

$G = (V_b, V_r, E)$, let c_f = (v1, v2, v3, v4) is a length-4 cycle of G ,then the Rail include:

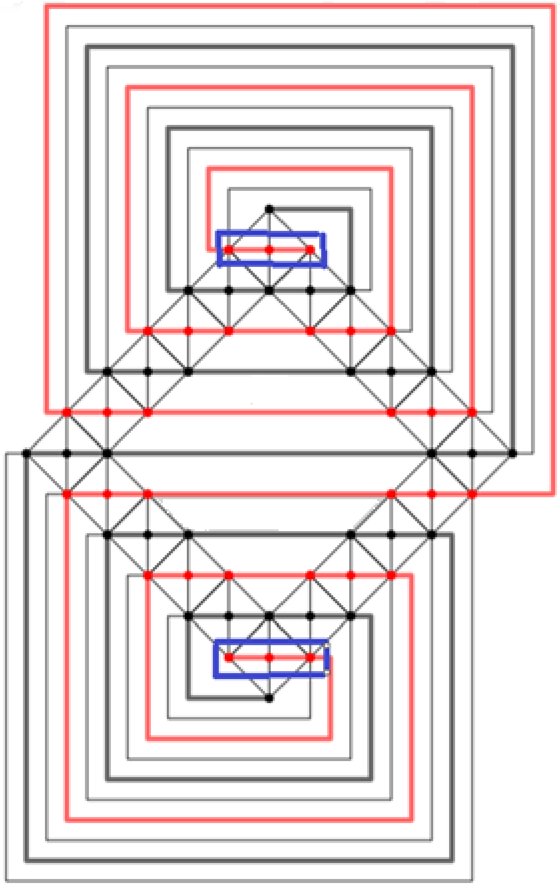
(a) At least one of $\{(v_1, v_3),(v_2, v_4)\} \in C$

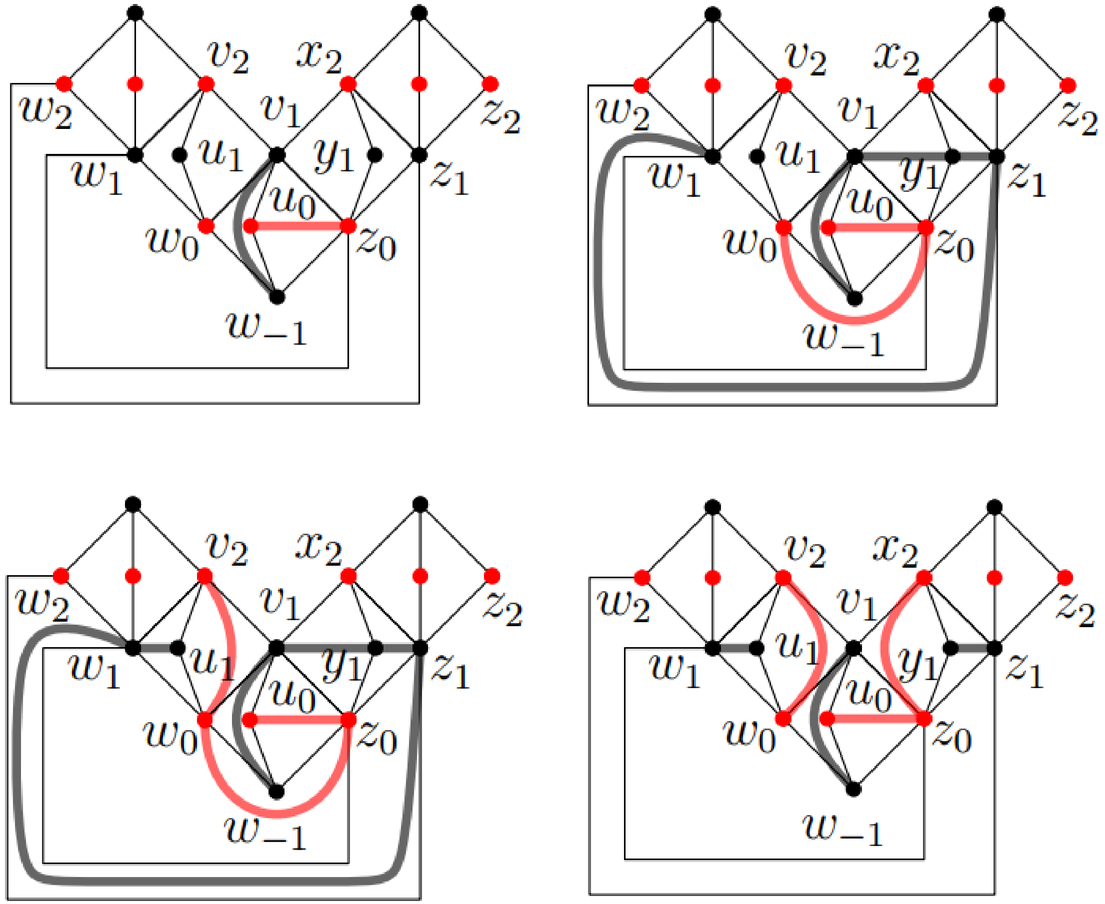
(b) In any planar embedding E of $G \cup E(C)$ either (v_1, v_3) or (v_2, v_4) lies inside c_f.



Lemma :

Any Rail C of the frame F_k contains the (w_0, u_0), (u_0, z_0), (w_{k+1}, u_{k+1}), and (u_{k+1}, z_{k+1})

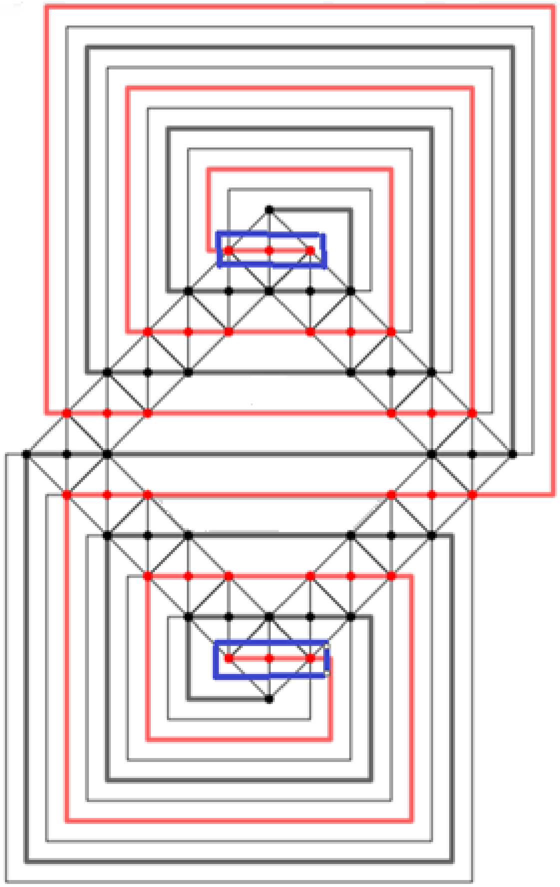


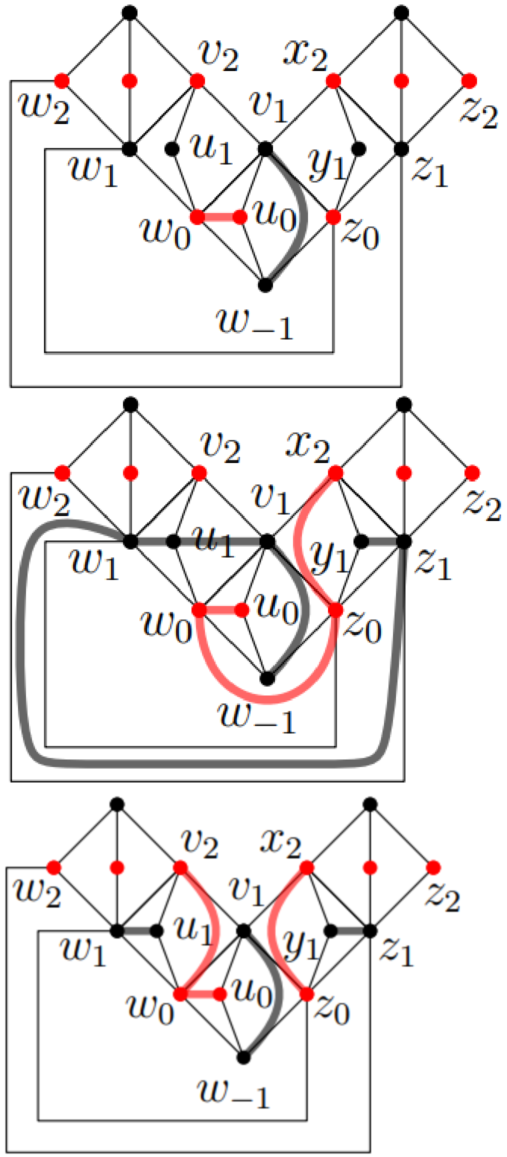


If only (u_0 , z_0) in the Rail

Lemma :

Any Rail C of the frame F_k contains the (w_0, u_0), (u_0, z_0), (w_{k+1}, u_{k+1}), and (u_{k+1}, z_{k+1})

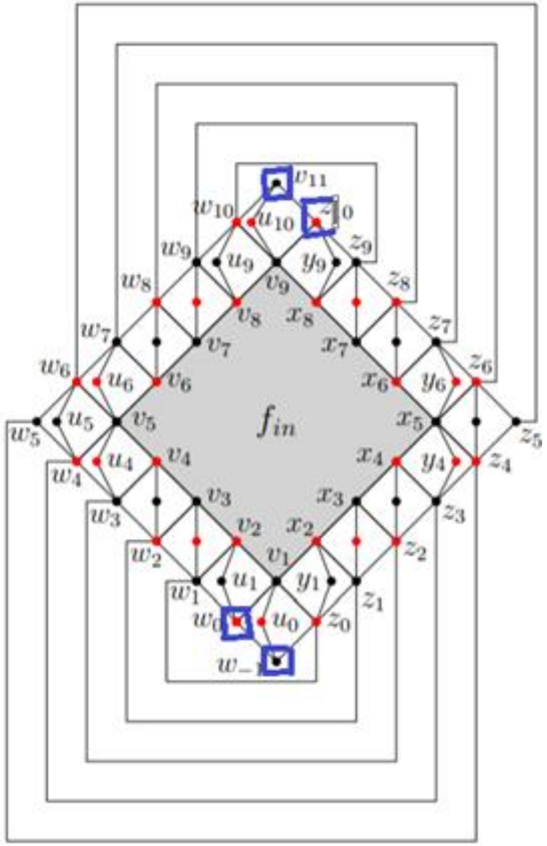


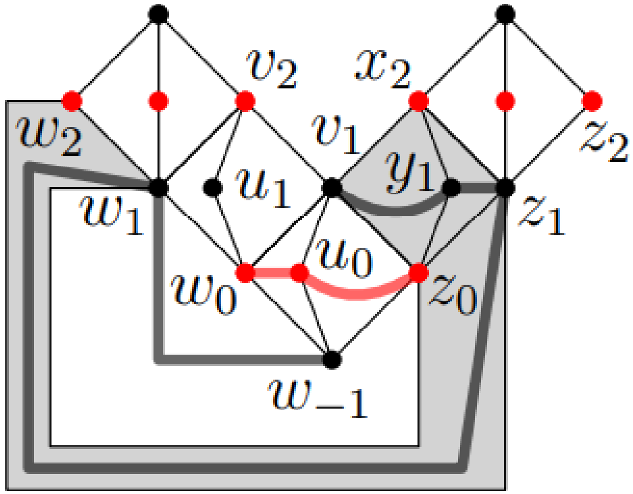


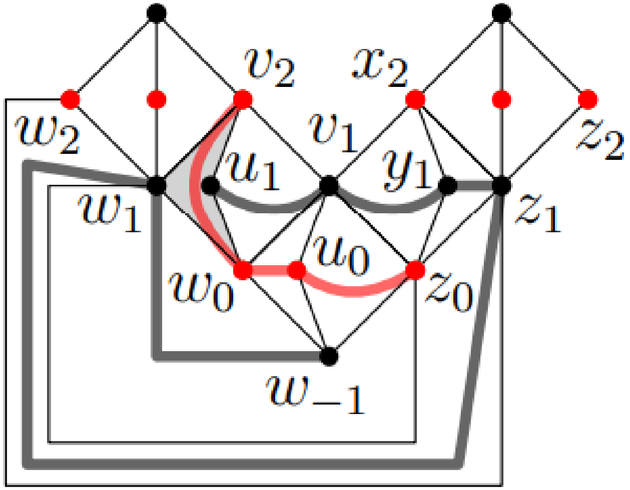
If only (w_0, u_0) in the Rail

Lemma :

w_0 and z_{k+1} are the red end-vertices of C, and the vertices w_{-1} and w_{k+2} are the black end-vertices of C

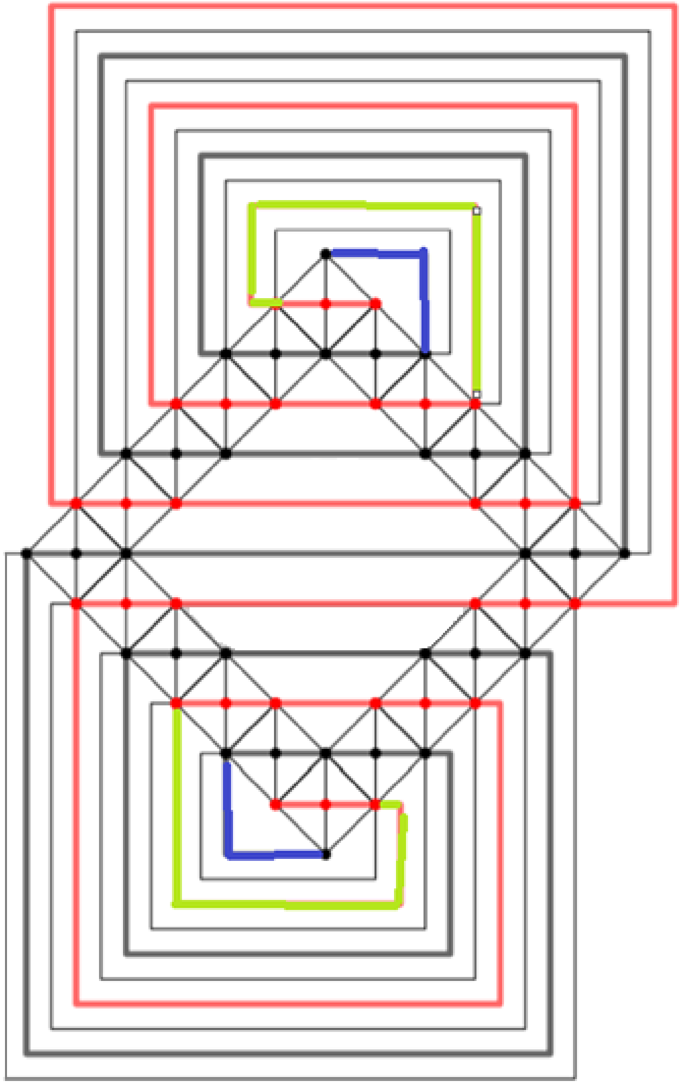






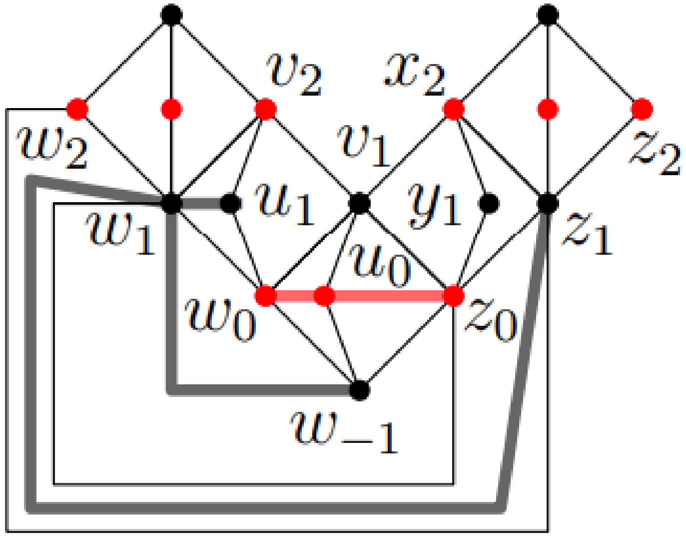
W_{-1} must be the end vertex

If z_0 is end vertex



Lemma :

C contains black tracks (w_{-1}, w_1) and (w_{k+2}, z_k), and red tracks (z_0, w_2) and (w_{k+1}, z_{k-1})

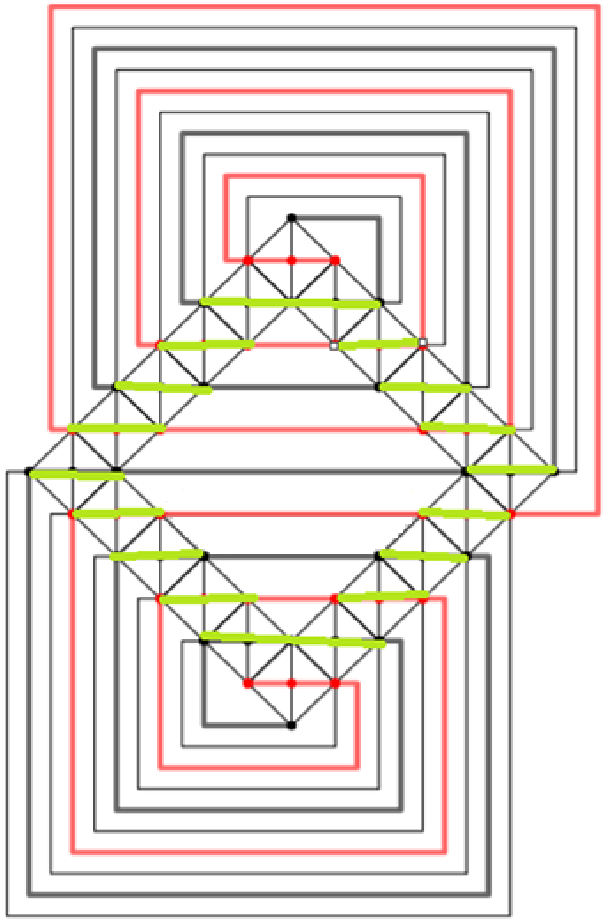


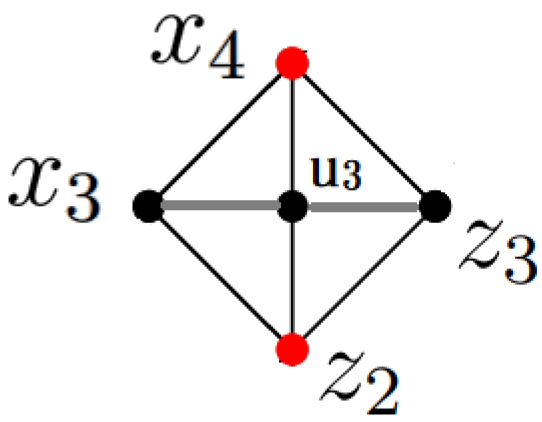
Because w_0 is the end-vertex,$(w_1,u_1) \in C$

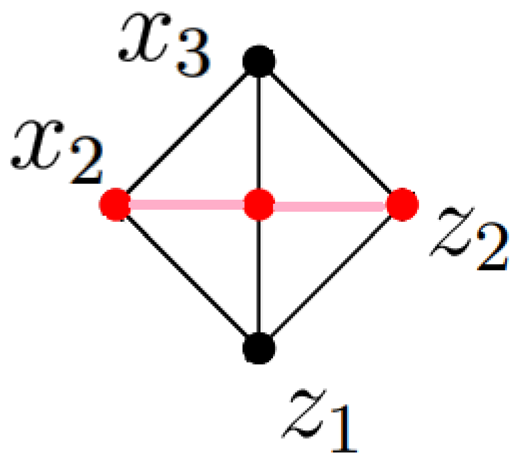
If (z_0,w_2) is not a track

Lemma :

C contains the tracks (w_i, u_i), (u_i, v_i), (x_i, y_i), (y_i, z_i), for $i = 1, \ldots, k$, where $x_1 = v_1$ and $x_k = v_k$.

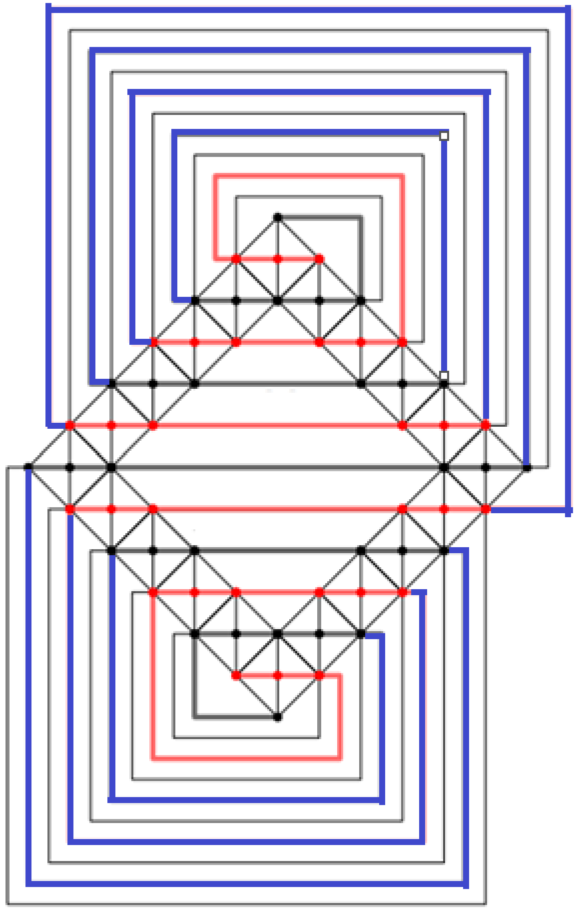


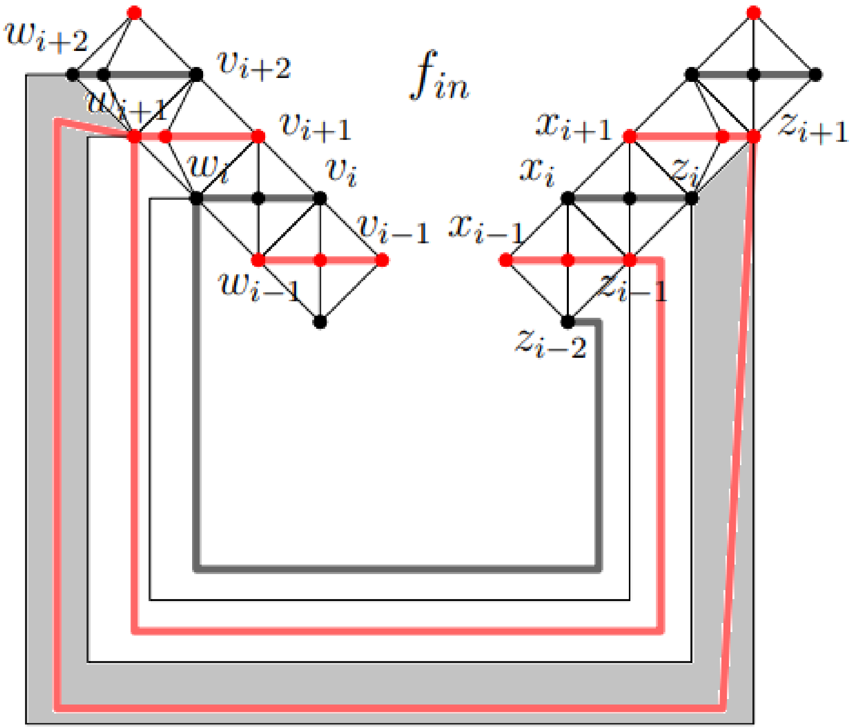


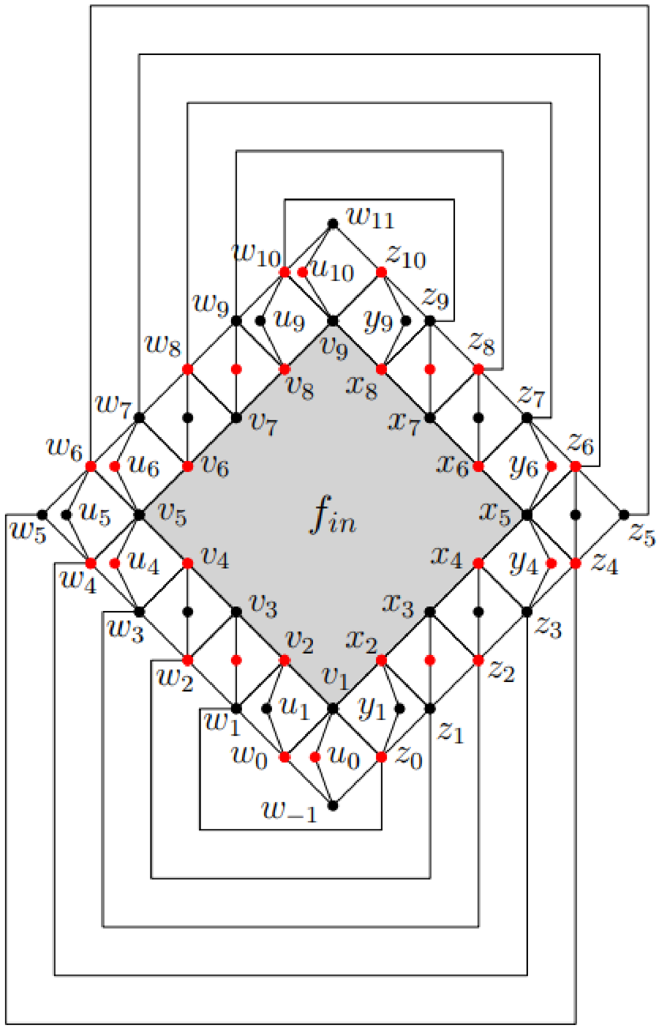


Lemma :

C contains the tracks (z_i , w_{i+2}), for i = 1, . . . , k





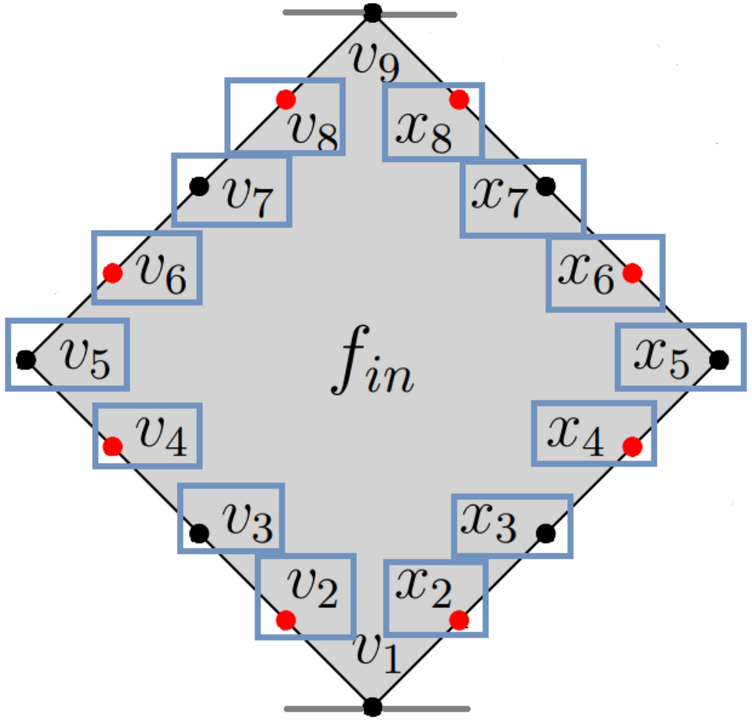


Lemma :

All the vertices in $V(H) \setminus \{v_1, v_k\}$ lie in the f_{in}.

proof. Because H is connected, and the face which v_1 share with v_k is f_{in}.

Define :



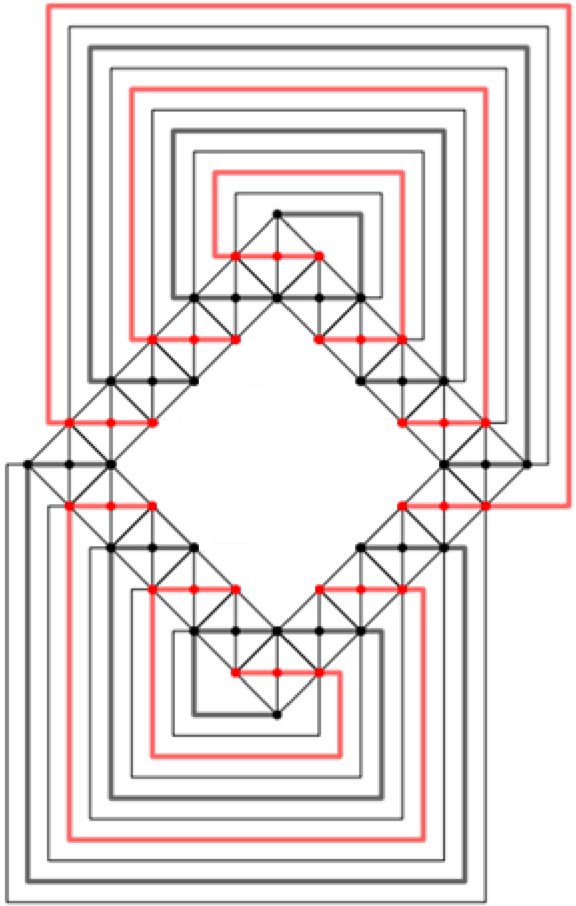
We define the vertexes incident to the f_{in} except for v_1 and v_k **track end**, and divide them into 4 groups :

(1) Left Black track end

(2) Right Black track end

(3) Left Red track end

(4) Right Red track end

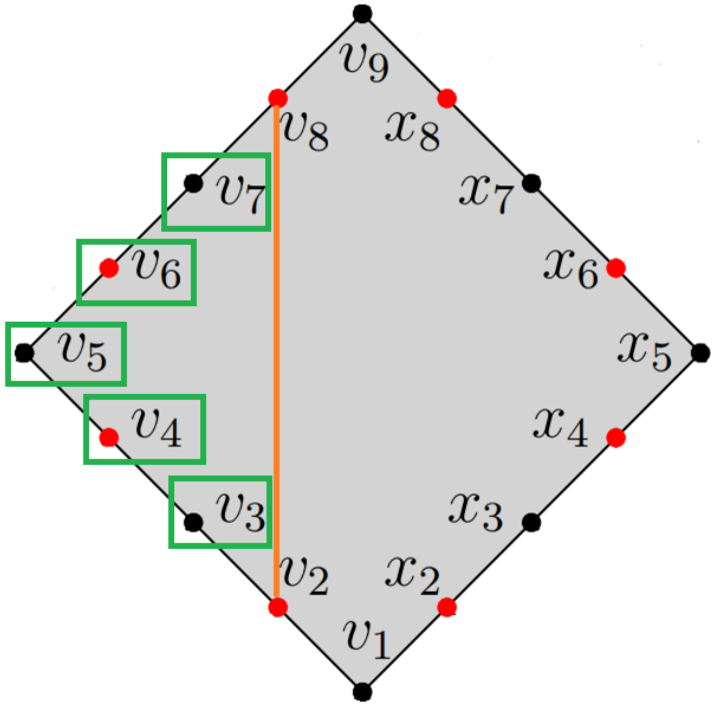


Lemma :

One side of the tracks incident to **track ends** are in the f_{in}.

proof. One side of the **track ends** are already fixed, and the **track ends** do not share any face with other vertex except for f_{in}.

The tracks in f_{in} will form some paths, we define the paths **internal paths**.

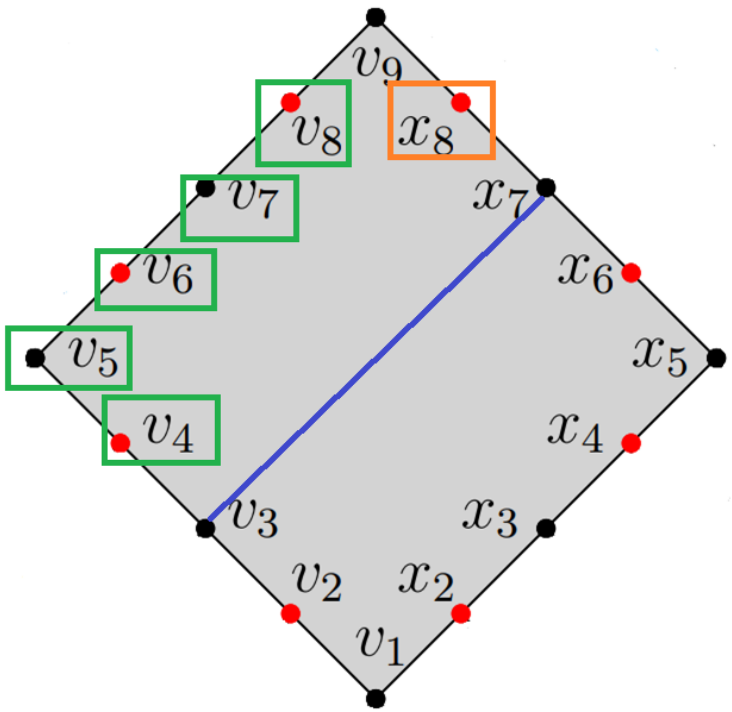


Lemma :

The **internal paths** can not between two same side track ends.

proof. By contradiction, if there is one internal path between the same side, that will separate two faces which are incident to **odd** track ends. So there will be isolated track end.

Example: There are still 5 track ends



Ex: The top part has

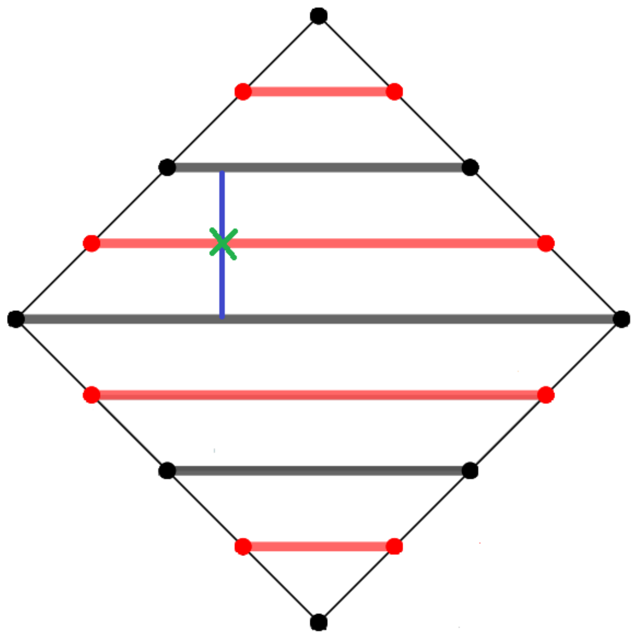
5 Left track ends

1 Right ends

Lemma :

The **internal paths** are all between v_i and x_i, for i=2,…k-1.

proof. By contradiction, if there is one internal path between v_j and x_k, $j \neq k$, then it will separate fin into **top** and **down** part, the number of left and right track ends will not be the same in both parts.



Ex: **If there is an edge between Layer 3 and 5，**

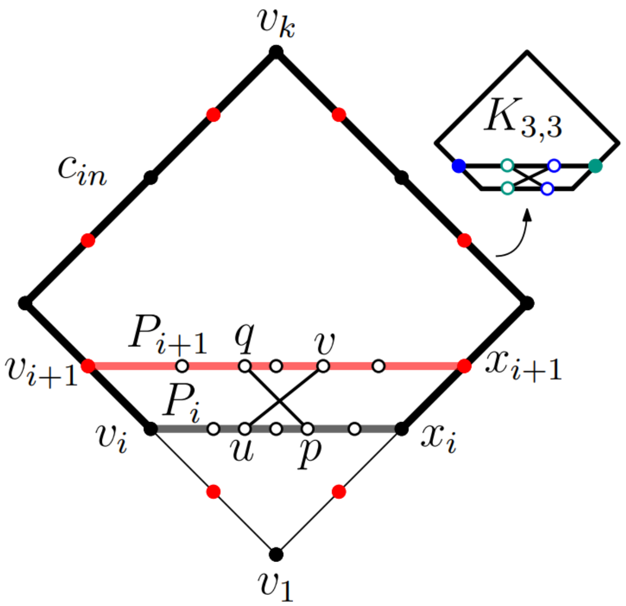
It will cross the path 4.

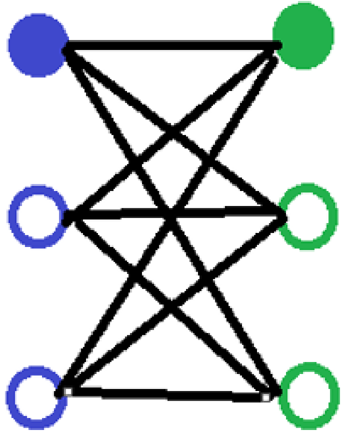
Lemma :

The vertexes in H are all on the internal path. **------------- Vertex Layers**

There are no edges between different layers **------------- Edge Limit**

proof. If this kind of edge exists, it will cross at least one internal path, then the planarity will be broken.





Lemma :

We define the order of each layer ,the righter vertex is bigger, and there are no (u,v),(p,q) ,{u 、p} {v 、q}are in the same layer u/q is lefter than p/v **-----No Cross**

proof.

By contradiction, if above situation exists, $H \cup F_k$ will contain the subdivision of $K_{3,3}$,

by Kuratowski's theorem, $H \cup F_k$ does not admits any planar embedding.

Theorem. H admits a special level planar embedding if and only if $H \cup F_k$ admits a B2BE.

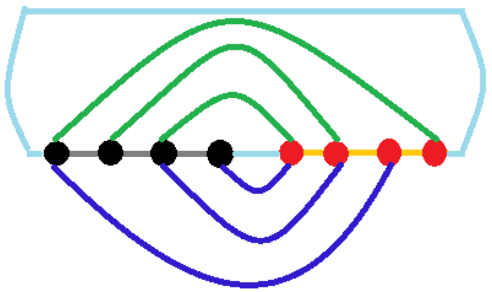
Theorem. B2BE is NP-complete

Theorem. 2-level quasi-planar drawing is NP-complete

Theorem. (2, 2)-track layout is NP-complete

Linear Algorithm for B2BEFO(Fixed Order)

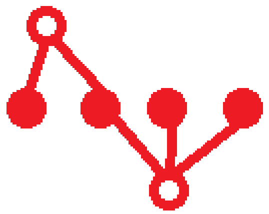
Strategy



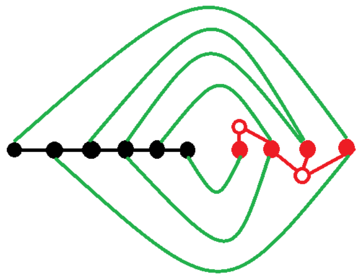
(1) Check whether G has a **Black rail** by testing the planarity of $G_B = G \cup B$, $B = \{ (b_1,b_2),\ldots(b_{m-1},b_m) \}$

(2) Check whether G_B has a **Red rail** and the two **RB-connected tracks** by a sufficient and necessary condition named **Caterpillar property**.

Auxiliary graph A(E)



A(E)



A(E) ∪ G

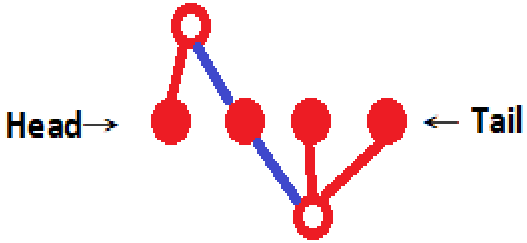
- The auxiliary graph A(E) consist:

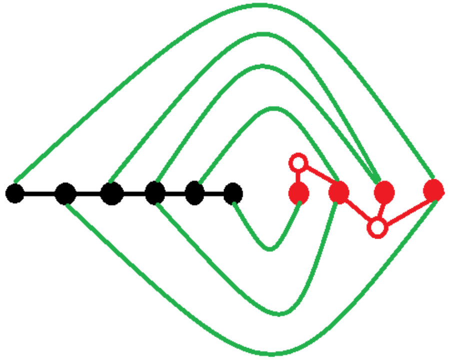
  (1) Every red vertex in G

  (2) A vertex for each face incident to at least two red vertex in E

  (3) An edge for each red vertex v of G incident to a red face f of E

Caterpillar property





- A embedding E has Caterpillar Property if and only if:

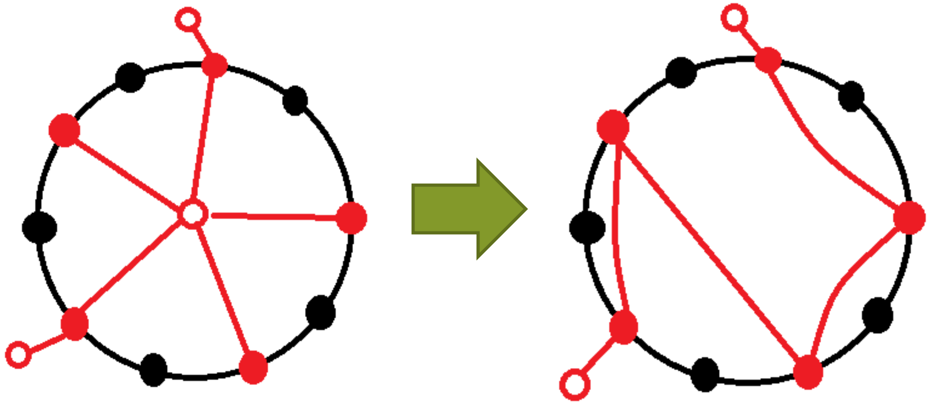
  (1) **Bones** :

  It's auxiliary graph A(E) is a caterpillar and it's backbone $(f_1, v_2, f_2, \ldots, v_k, f_k)$ spans all red faces

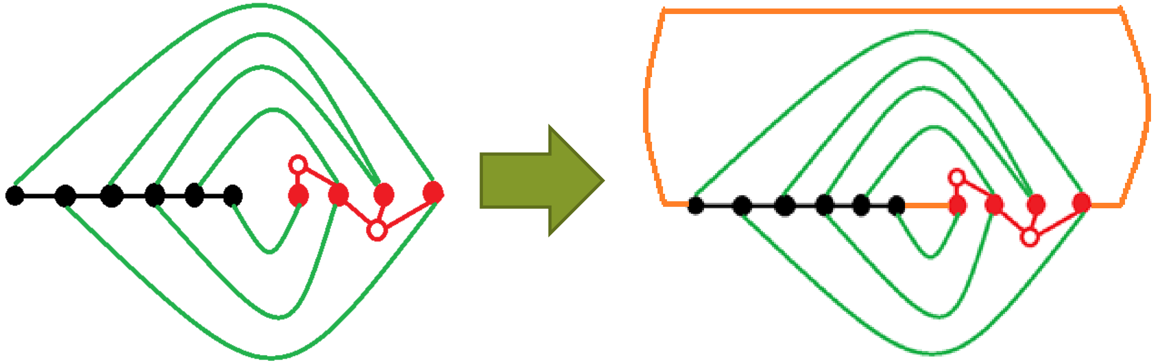
  (2) **Head and Tail** :

  $\exists$ two leaves vertex r_1 and r_p incident to f_1 、 f_k, respectively and share the same face to b_m 、b_1 , respectively in $E \cup A(E)$.

(<=) Transform the Caterpillar into Rail



Red rail



RB-connected tracks

(=>) G admits B2BE,then it has Caterpillar Property

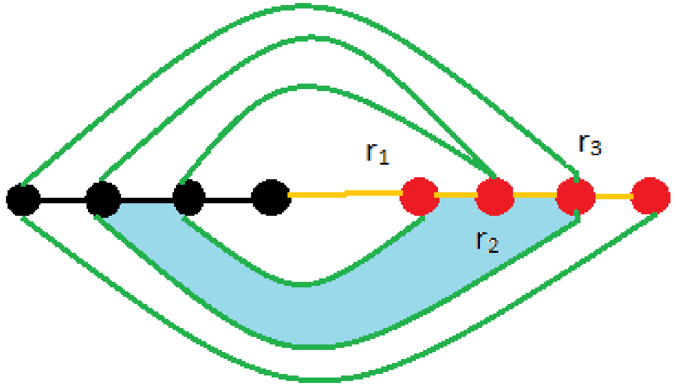
Claim : The red vertexes which have at least two edges in different pages are degree-2 in A(E), and the others are degree-1.

The degrees-2 red vertexes are in the bone, the degree-1 vertexes are the feet.

Lemma :

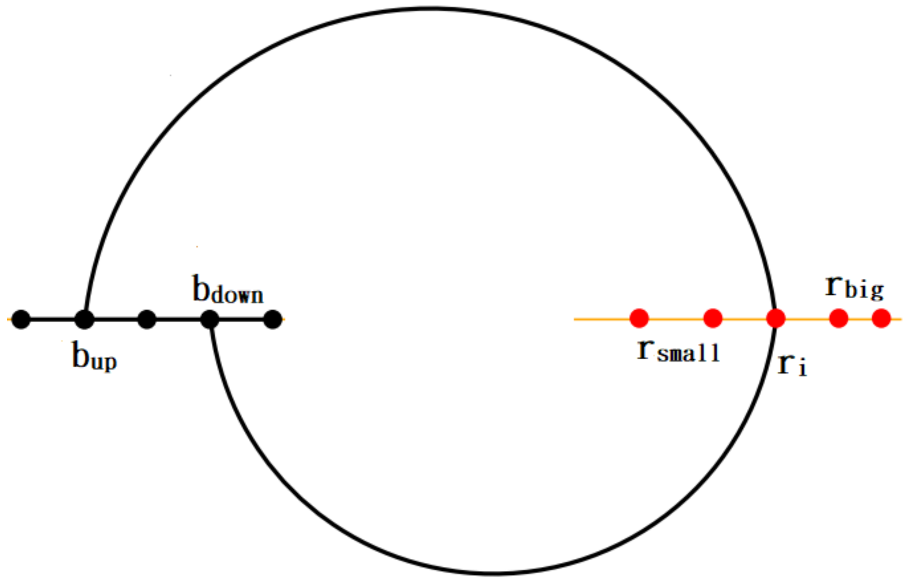
r_i is a red vertex , $\forall r_{small}$, r_{big}, small=1,2,…, i-1, big=i+1,…..,r_p , r_{small} does not share any face with r_{big} if and only if r_i has at least two edges in different sides.

(=>) If r_i does not have the edges in the both sides, then r_{i-1} and r_{i+1} can encounter each other in the other side, a contradiction.



All edges of r_2 are on the top page, then r_1 and r_3 will share the under page.

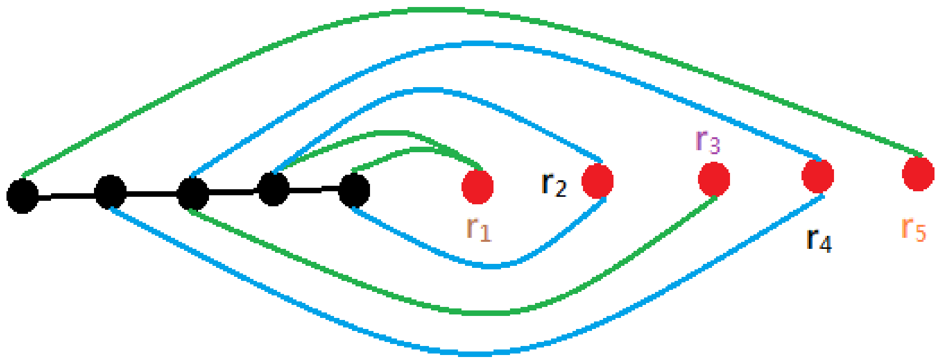
(<=) If r_i have at least two edges in different pages, let them be (r_i, b_{up}) 、 (r_i, b_{down}) , then the two edges and path (b_{up}, b_{down}) will be encased all the r_{small}.



All of the r_{small} will be separate from r_{big} by the two edges and the black path.

Observation:

The red vertexes are divided into several parts by the vertexes which have both side edges. The vertexes can only share face with the same group.

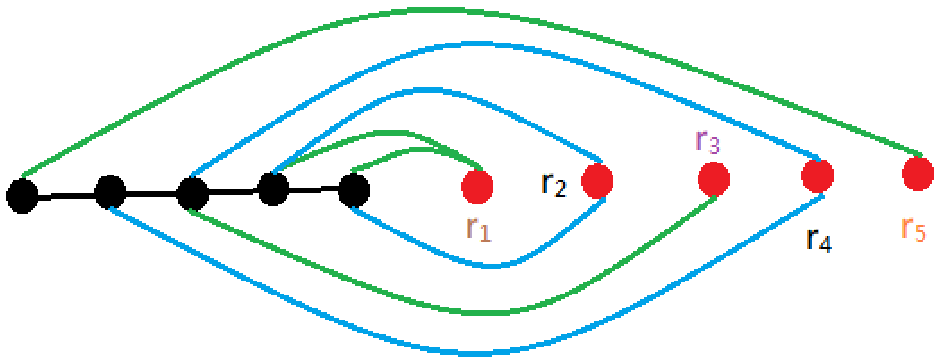


r_2 and r_4 both have 2 edges in the different edge.

The red vertexes are divided into three groups: $\{r_1,r_2\},\{r_2,r_3,r_4\},\{r_4,r_5\}$

Observation:

The red vertexes are divided into several parts by the vertexes which have both side edges. The vertexes can only share face with the same group.

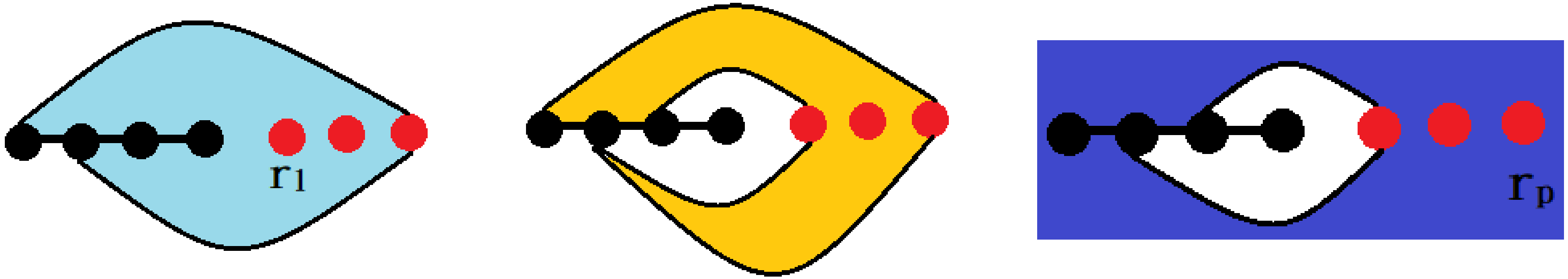


r_2 and r_4 both have 2 edges in the different edge.

The red vertexes are divided into three groups: $\{r_1,r_2\},\{r_2,r_3,r_4\},\{r_4,r_5\}$

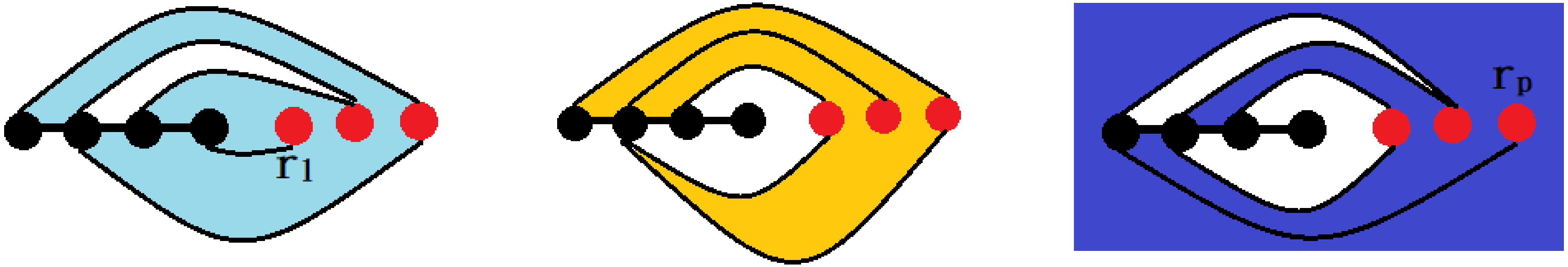
Observation(Each group):

After remove all edges of the group member except for the vertexes which have both side edges, there are only one red face shared by the group members.



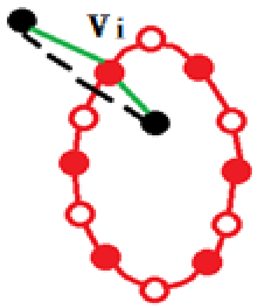
Observation(Each group):

Add the edges back, the new faces are only incident to to one red vertex ➡ There are not any red faces being generated.



By above claim and the two following features of A(E),we have a conclusion—— A(E) is a caterpillar.

(1) A(E) is acyclic



Find two black vertexes along the boundary in clockwise and counter-clockwise direction, then the black path will break the planarity

(2) A(E) is connected (Each red vertex is incident to the track incident to itself.)

The **Bones** are checked, and r_1/r_p share a face with b_m/b_1, the **Head and Tail** are checked, too.

B2BE with fixed order(B2BEFO)

G is a bipartite gragh admits a B2BE with the fixed order

$\pi_b = <b_1,\ldots,b_m>$ if and only if

(1) $<b_1,\ldots,b_m>$ is a Black rail

(2) Under π_b , exist a embedding E of G has Caterpillar Property

Augumented Block

To use the SPQR tree , we want to assume G is almost biconnected.

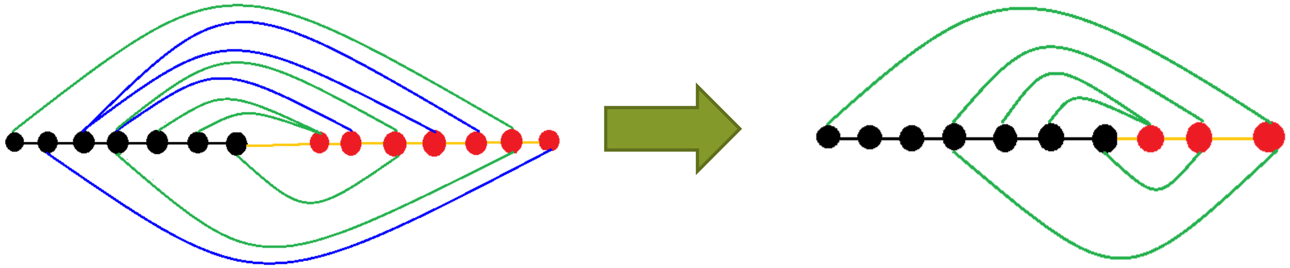
First ,we will introduce how to get the almost biconnected component.

Define:

Augumented Block: We call these almost biconnected components.

Solo red vertex: the red vertex whose degree is 1

Step 1: Remove all the solo red vertexes to get G_{new}



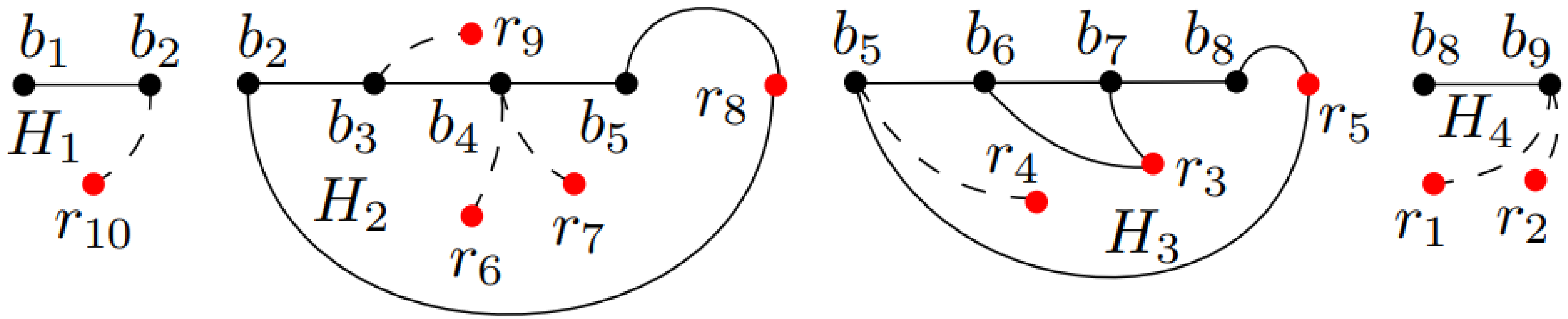
Step 2: Find the biconnected component of G_{new} .

Step 3: For each biconnected component , add the solo red vertexes incident to its black vertex in G.

Remove any red vertex can not disconnected G_{new} .

Because the black vertexes can touch each other through the **Black Rail**

=> All the cut vertexes in G_{new} are black.



The Augumented Blocks are like this graph

If G admits a B2BE iff all Augumented Blocks of G admits a B2BE.

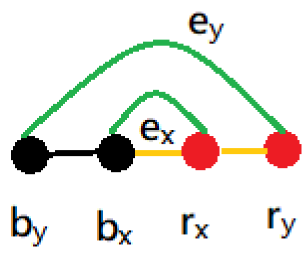
(=>) trivial

(<=)

If G_j means the Augumented Blocks of G, and admits a B2BEFO $<G_j , \pi_b^j >$.

(1) put with order $<\pi_b^1, \pi_b^2, . . . , \pi_b^q, \pi_r^q, \pi_r^{q-1}, . . . , \pi_r^1>$.

(2) put edges in the same page of Augumented Block

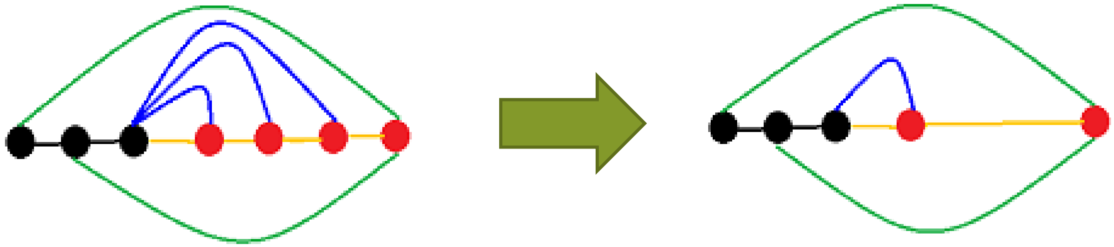


proof: By contradiction, there are two different edges $e_x=(b_x , r_x) \in Block_1$, $e_y=(b_y , r_y) \in Block_2$ will cross. Because we put $Block_1$ in inside. Then the order from left to right is $<b_y , b_x , r_x ,r_y >$,it will not cross, a contradiction.

Consider each Augumented Block

Each Augumented Block can construct its SPQR tree by removing all solo red vertexes. The method of solo red vertexes embedded into the SPQR tree will be introduced in the latter part.

All solo red vertexes incident to b_i can be cut off to only one solo red vertex because all of them can be put in the same face.



Linear time algorithm for B2BEFO problem

Step 1: Test whether existing a Black rail by testing planarity

Step 2: Find all Augumented Blocks

Step 3: Remove solo red vertexes and construct a SPQR tree

Step 4: Bottom-up traversal to check the Caterpillar Property and embed the solo red vertexes.

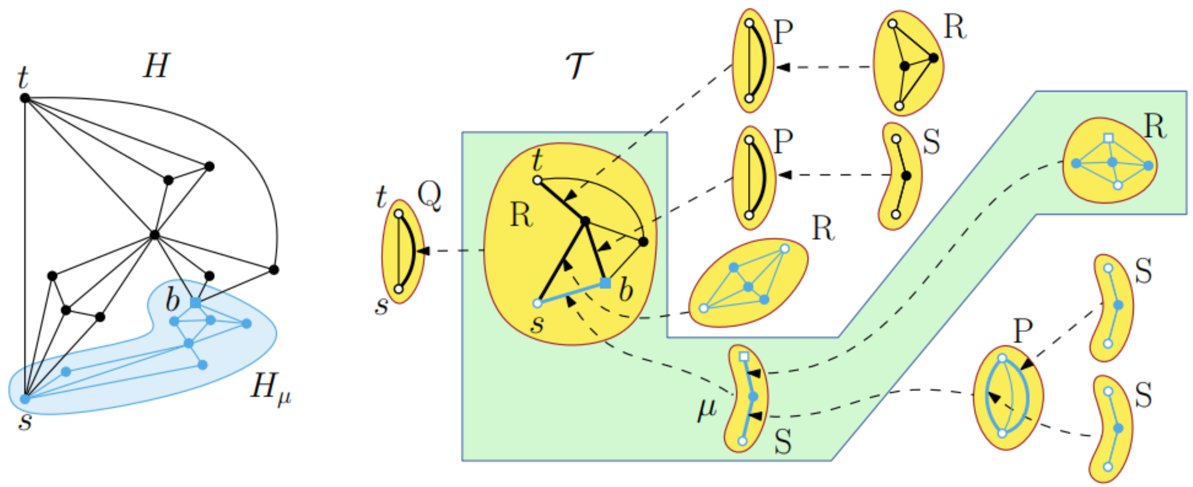
Step 5: If above succeed, construct a B2BEFO in linear time

SPQR tree

(1) **The reference is** **(**$\mathbf{b_1,b_2}$**)**

(2) The split pair (u , v) of each node μ is poles.

(3) The associated graph of each node μ which has virtual edges is skeleton, we denote its size by sk(μ).



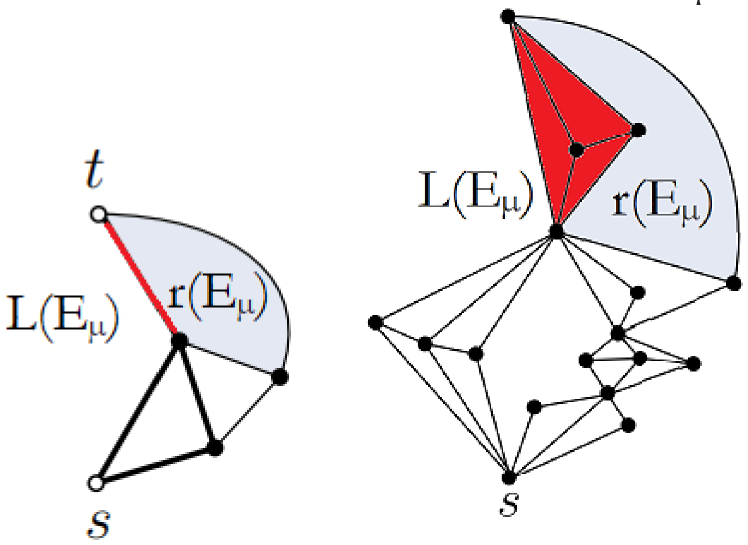
Special S node

To ease the description of our embedding algorithms, we use the slightly modified version of SPQR-trees ,where each S-node has exactly two children.

The SPQR-trees defined in this way can still be constructed in O(n) time.

Outer faces

Define: The virtual edge represented node μ will be incident to two faces, we call the left one $L(E_\mu)$, and the other one $r(E_\mu)$.



Initial idea

(1) All of the leaves are triconnected, so their embedding are unique.

(2) The non-leaf embedding are the permutations of the leaves.

Idea: We can list all of the possibility of each nodes which can be part of B2BE.

But the space and the time complexity will be exponential.

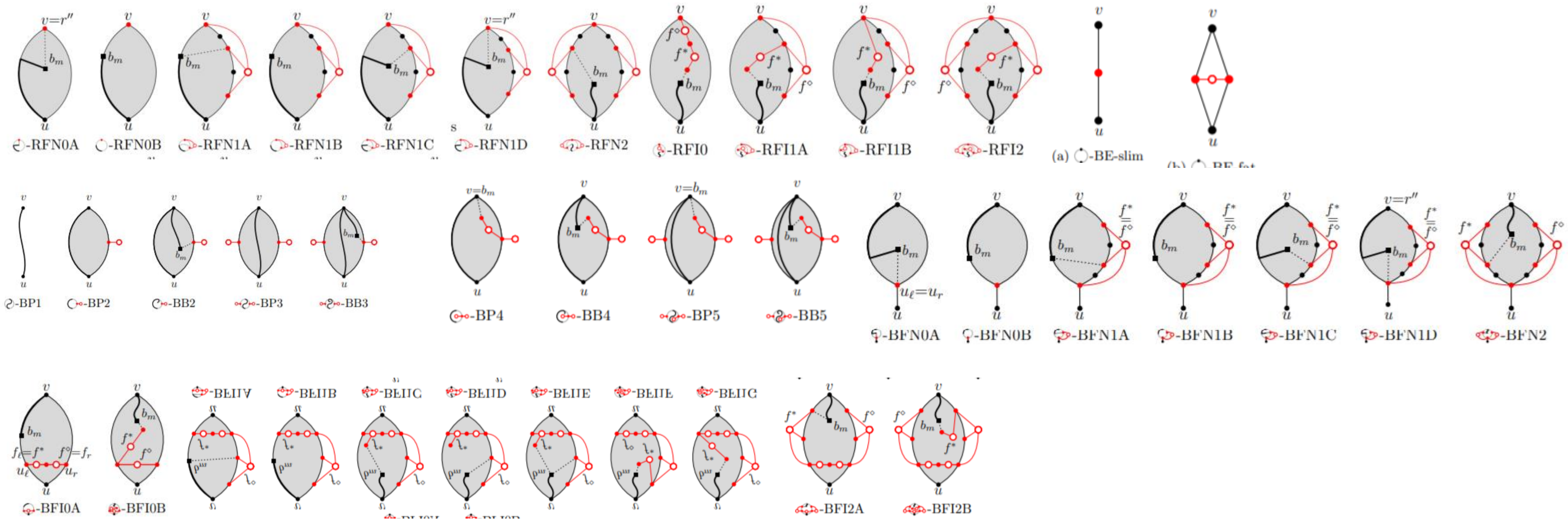
Main method to achieve linear time

Design a classification of all possible parts of B2BE on a node to achieve some characteristics:

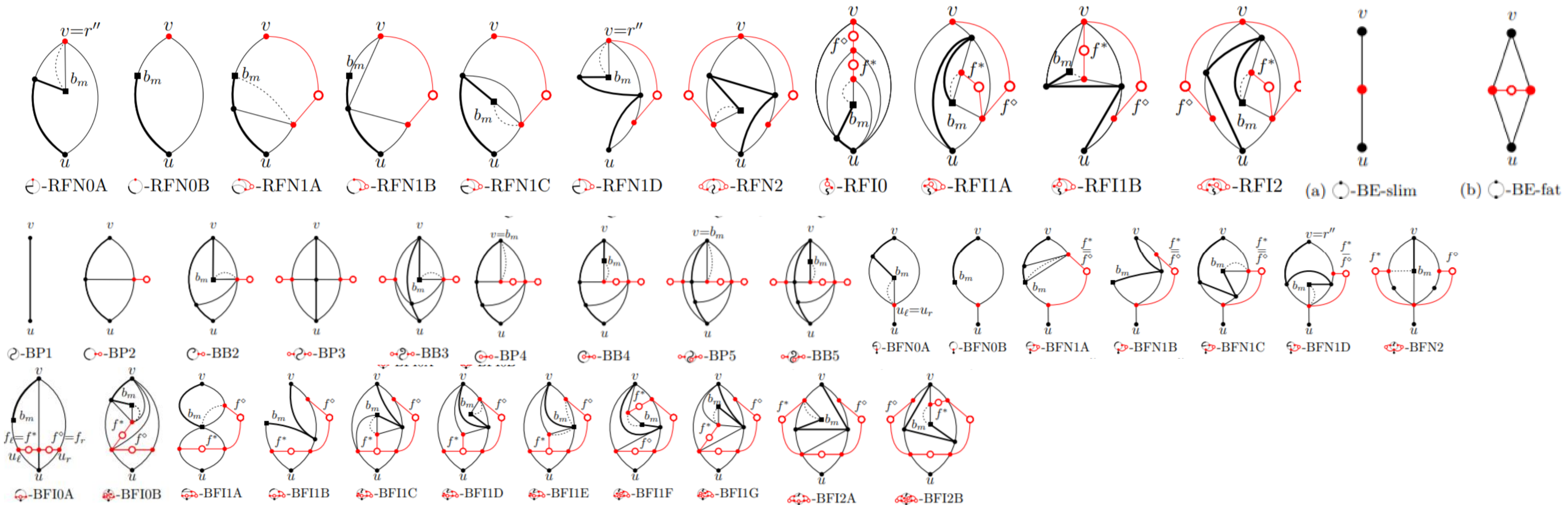
(1) If v is the child of μ ,the type of μ will not be influence the type of μ.

(2) All type the child of the root may be , we can use it to construct the complete B2BE.

Types(up to a flip)



Examples



-RFN0A
-RFN0B
-RFN1A
-RFN1B
-RFN1C
-RFN1D
-RFN2
-RFI0
-RFI1A
-RFI1B
-RFI2
(a) -BE-slim
(b) -BE-fat
-BP1
-BP2
-BB2
-BP3
-BB3
-BP4
-BB4
-BP5
-BB5
-BFN0A
-BFN0B
-BFN1A
-BFN1B
-BFN1C
-BFN1D
-BFN2
-BFI0A
-BFI0B
-BFI1A
-BFI1B
-BFI1C
-BFI1D
-BFI1E
-BFI1F
-BFI1G
-BFI2A
-BFI2B

Bottom up traversal

Bottom up traversal checking can be finished in linear time and finite space:

(1) Use replace graph similar to Thévenin's theorem (Replace the virtual edges by the examples which sizes are O(1))

(2) Only record piece of information for one type of one node

(3) Each node can be checked and put solo red vertex in O(sk(μ))

(4) Construct the embedding by the piece of embedding recorded before.

Define: If exist an embedding of node v is one of above type, we said that v is useful, and this type is exposed in v.

In latter part, we need to prove:

(1) If the child of root is useful, then G admits a B2BE.

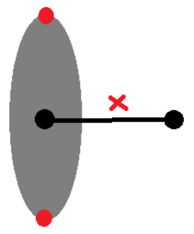
(2) If G admits a B2BE,then each node v is useful.

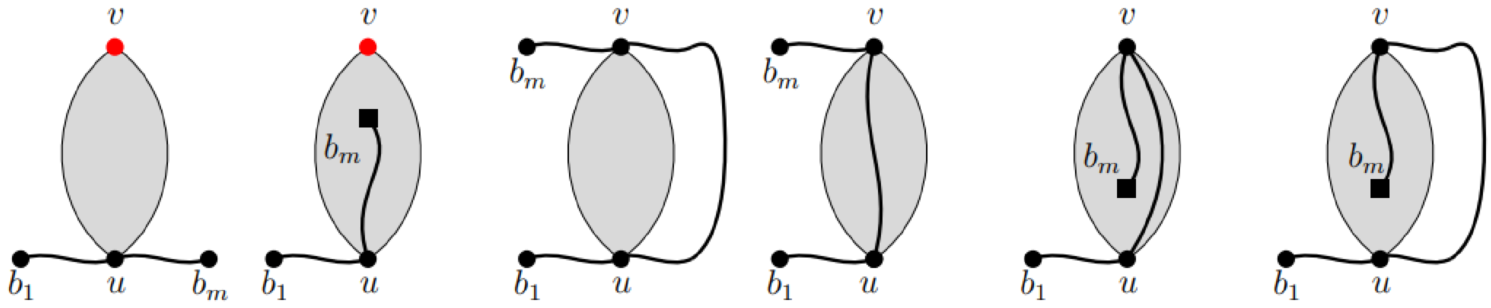
(3) v is μ's child, if we change v by replace graph in u , it will not influence the type of μ.

(4) $\forall$ v in SPQR tree can be test in skeleton(v) (The number of the permutation is limited in a small number).

Node Classification Roughly

- There is no node in T whose poles are both red.
- The reference of the tree is (b_1,b_2), so $(b_1,\ldots,u) \notin H_\mu$
- If one of $(u,\ldots,v)/(v,\ldots,b_m) \in H_\mu$, then whole path $(u,\ldots,v)/(v,\ldots,b_m) \in H_\mu$





(a) RE (b) RF (c) BE (d) BP (e) BB (f) BF

Observation:

A useful node should be a part of caterpillar , each node should offer an interface(export) to connect the whole caterpillar.

There are only three possibility of the export : the two outer faces and the red pole.

We still need to confirm b_1/b_m can find suitable r_p/r_1

Observation2 :

So, We use three Big features to subdivision the nodes:

(F1) the number of caterpillars $A(E_\mu)$ consists of

------ To know the role which μ play.

(F2) whether E_μ has at least one internal red face f or not

------ To know the caterpillar just touch $A(E_\mu)$ or deep into.

(F3) the number of outer faces of E_μ that belong to $A(E_\mu)$

------ To see how many export can be used.

Observation 3:

Beside of above features, sometimes μ has red pole ,then we have

(F4) the degree of the red pole of in $A(E_\mu)$

(F5) Whether b_m shares a face with the red pole

------ To see whether the red pole can be an export.

(F6) Whether b_m shares a face with one non-pole red vertex

------ To know whether a non-pole red vertex can play the role of r_1

Observation 4:

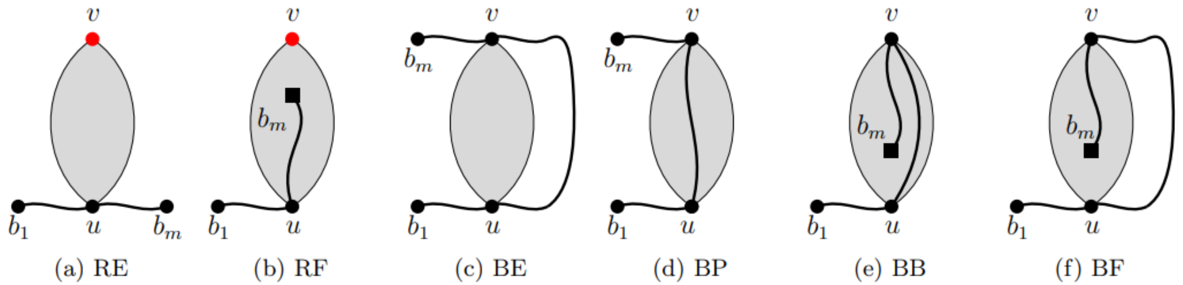
(F7) If b_m belongs to H_μ ,whether it is an internal vertex.

If b_m is an internal vertex, there should be an suitable r_1 belongs to H_μ .

(F8) If b_m is not an internal vertex, whether there are suitable r_1 belongs to H_μ and where is its position in the caterpillar.

------ Handle the head and tail of Caterpillar (b_1 is less complicated)

Some Lemmas



(1) Let μ be a nonroot node of T such that $b_1 \in H_\mu$. Then, b_1 is a pole of μ, and μ is of type either -BE, or -BF, or -RE.

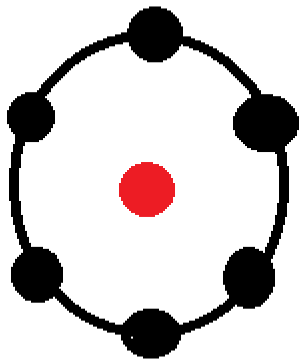
(=>) b_1 will not be internal vertex, so we can find an suitable r_p latter.

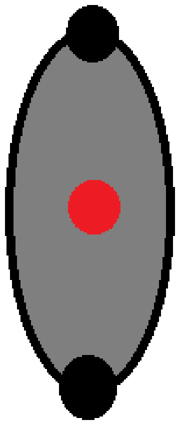
We will show the r_p exists if the child of the root is useful.

We will show this in the section of constructing the complete B2BE by the child of the root.

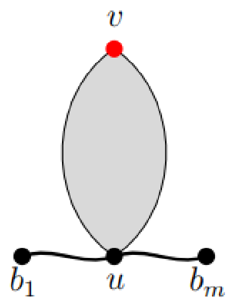
(2) The graph $A(E_\mu)$ is composed of at most two caterpillars. The caterpillar has only 3 export (pole 、$L(E_\mu)$ 、$r(E_\mu)$), if these three are red at the same time ,then there is only one caterpillar.

(3) If E_μ contains an internal red vertex, then it also contains an internal red face and a red vertex incident to the outer face.





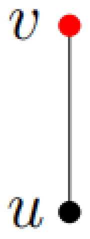
RE-type



If μ is of type -RE, then H_μ has a unique embedding.

Proof.

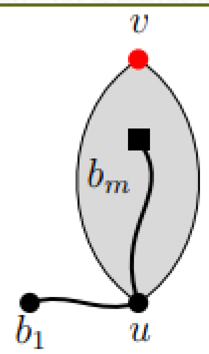
u is the only black vertex in μ , so the poles can not be separating pair.



=> (u,v) is an edge

=> H_u has unique embedding

RF-type



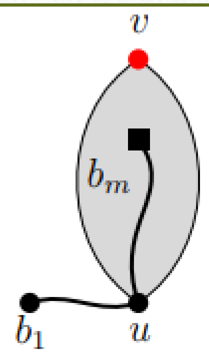
RF-type has only one caterpillar.

Further, if $A(E_\mu)$ contains both $L(E_\mu)$ and $r(E_\mu)$, then the path between $L(E_\mu)$ and $r(E_\mu)$ in $A(E_\mu)$ is $(L(E_\mu), v, r(E_\mu))$.

Proof.

Because the red pole is connected to the other two exports($L(E_\mu)$ 、 $r(E_\mu)$),so if there are two caterpillars, their exports must be $L(E_\mu)$ and $r(E_\mu)$, but the red pole connect the $L(E_\mu)$ and $r(E_\mu)$,so they belong to the same caterpillar, a contradiciton.

RF-type



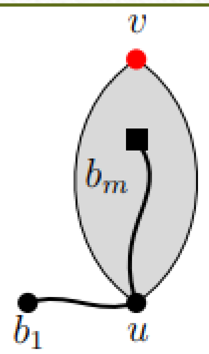
(F1) RF-type must have only one caterpillar

(F2) E_μ can have an internal red face f or not.

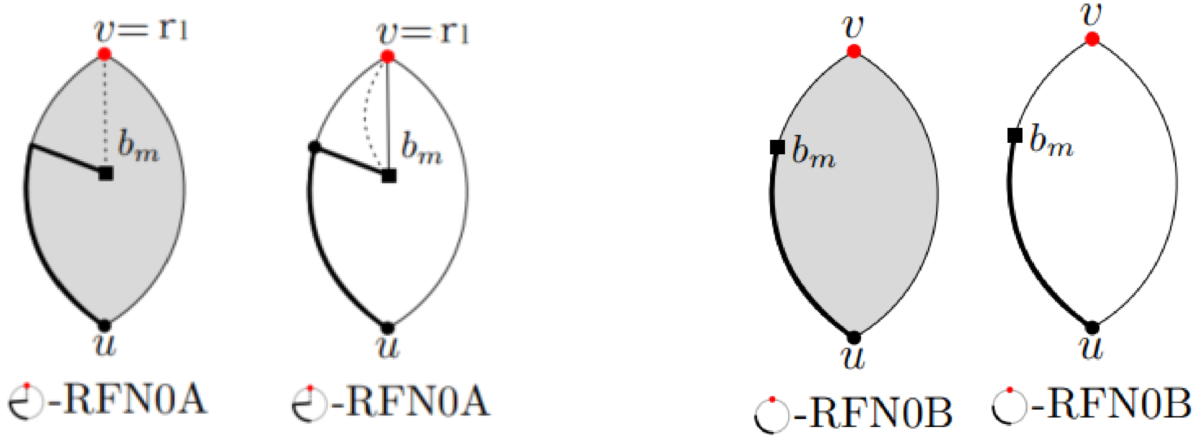
(F3) E_μ can have 0, 1, or 2 outer faces which are red.

So, there may be six types, -RFN0, -RFN1, -RFN2, -RFI0, -RFI1, and -RFI2.

RFN-type and only v is red



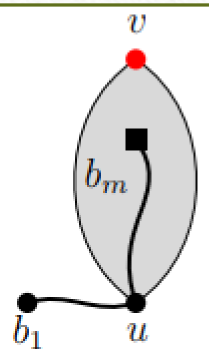
Suppose that E_μ does not contain any internal red face. If H_μ contains no red vertex different from v , then there are no outer faces are red.



b_m is an internal vertex

b_m is not an internal vertex

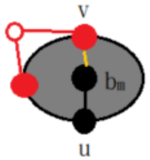
RFN-type and not only v is red



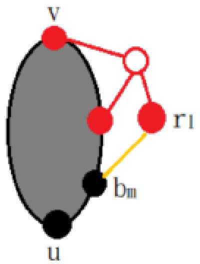
Suppose that E_μ does not contain any internal red face. If H_μ contains at least one red vertex different from v, then $r(E_\mu)$ or $L(E_\mu)$ is an end-vertex of the backbone of $A(E)$.

Proof. Because b_m is a non-pole vertex of H_μ ,so r_1 has three probabilities:

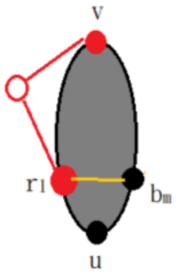
(3) $r_1 = v$



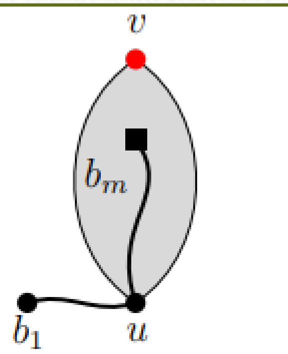
(1) r_1 is not in H_μ and incident to $L(E_\mu)$ or $r(E_\mu)$



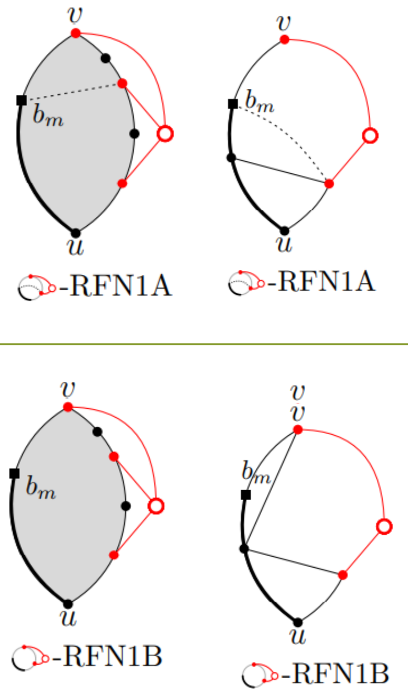
(2) r_1 is a non-pole vertex of H_μ incident to $r(E_\mu)$ or $L(E_\mu)$

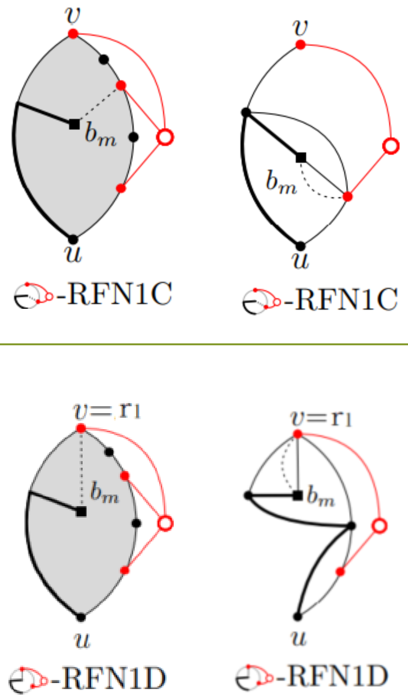


RFN-type and not only v is red

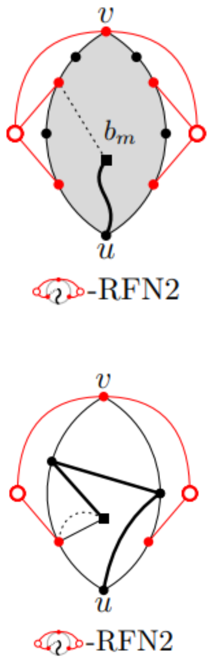


Let $r(E_\mu)$ is red





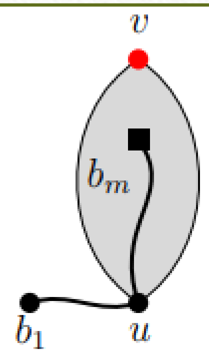
At least one red vertex different from v incident to b_m



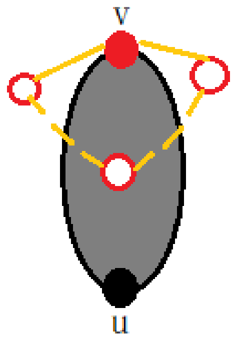
The only red vertex incident to b_m is v

b_m is incident to $L(E_\mu)$ | b_m is not incident to $L(E_\mu)$

RFI-type



Suppose that E_μ contains an internal red face, and let f_h and f_t be the end-vertices of the backbone B_μ of the caterpillar $A(E_\mu)$.

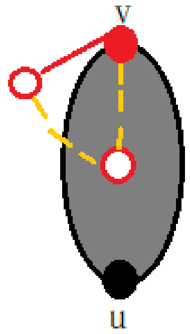


Lemma :

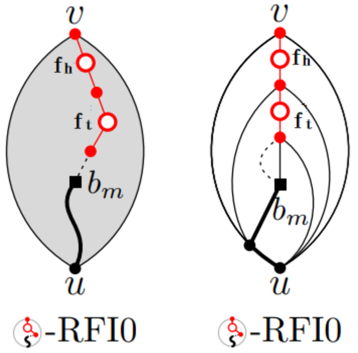
At least one internal face is the end-vertex of the whole backbone

proof.

By contradiction, no any internal red face is the end of $A(E)$,then it will form a cycle , a contradiction. We call the internal face f_t.



RFI0-type



Lemma :

If E_μ is of type -RFI0, then f_h corresponds to an internal face of E_μ incident to the red pole v, which is a leaf of $A(E_\mu)$

($f_h = f_t$ is possible).

proof. Because we suppose f_t is the end of $A(E)$ and $L(E_\mu)$ 、 $r(E_\mu)$ are not red, so the only export is the red pole.

RFI1-type

v is not in backbone of $A(E_\mu)$ v is in backbone of $A(E_\mu)$

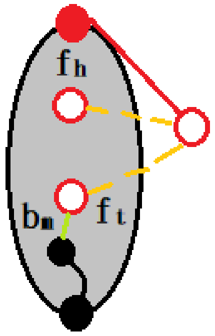
-RFI1A -RFI1A -RFI1B -RFI1B

Lemma :

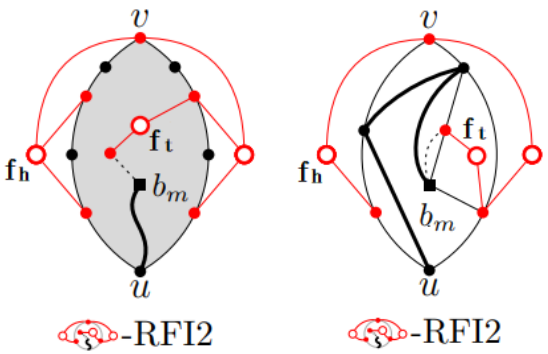
If E_μ is of type -RFI1, then f_h corresponds to an outer face of E_μ.

proof.

By contradiction, if the red outer face of E_μ is not f_h, then f_h is an internal face, so f_h is an end-vertex of the backbone of A(E), so $b_1 \in H_\mu$, a contradiction to the RF-type.



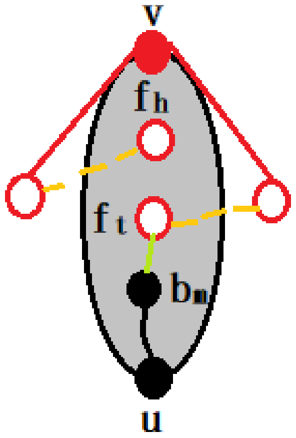
RFI2-type



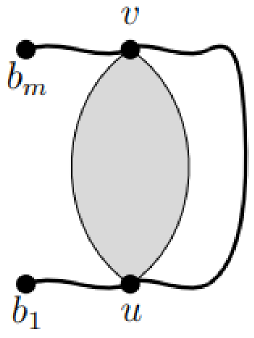
Lemma :

If E_μ is of type –RFI2, then f_h corresponds to an outer face of E_μ.

proof. Analogous to RFI1, if f_h is internal, then it will be an end-vertex of the backbone of A(E), a contradiction.



BE-type



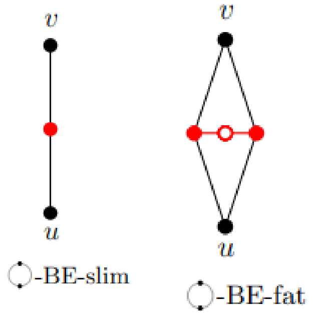
Because (u,v) is not in H_μ , so the poles of μ is a separating pair

The black vertexes in μ are only u and v

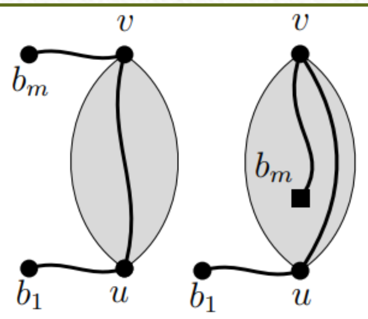
$\Rightarrow$ The other vertexes in μ are all red

$\Rightarrow$ The graph $H_\mu \backslash$solo red vertexes consists of a set of length-2 paths between the poles of μ.

$\Rightarrow$ If μ is of type -BE, then H_μ has a unique embedding (up to the permutations of the red vertexes).



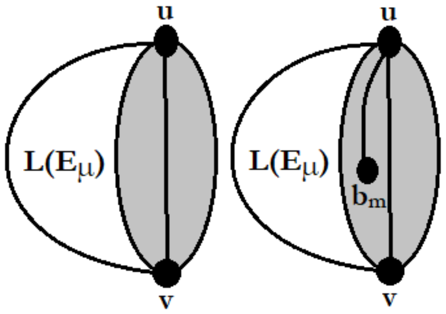
BP-type and BB-type



(d) BP (e) BB

Suppose that μ is of type BP or BB. Then in any embedding E of G, at least one red vertex of H_μ is incident to $L(E_\mu)$ and at least one red vertex of H_μ is incident to $r(E_\mu)$.

proof. By contradiction, if no red vertex not in H_μ incident to $L(E_\mu)/r(E_\mu)$,it will form a black cycle.



$\Rightarrow$ If one red vertex of H_μ incident to $L(E_\mu)/r(E_\mu)$,

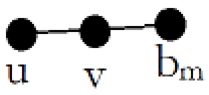
$A(E_\mu)$ contain $L(E_\mu)/r(E_\mu)$

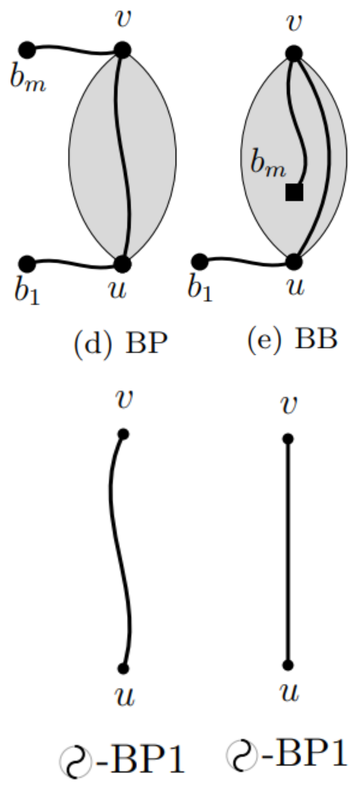
So, the outer faces of BB and BP, we mark them red if they have at least one red vertex incident to it.

BP-type and BB-type with zero caterpillar

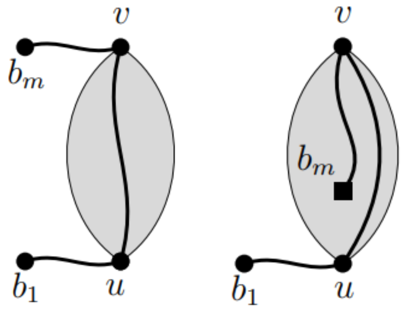
(1) BP-type and and BB-type are very similar expect for BB-type has at least one red vertex(otherwise it is BP type).

Proof: The vertex in H_μ can go to the both poles without through the another.



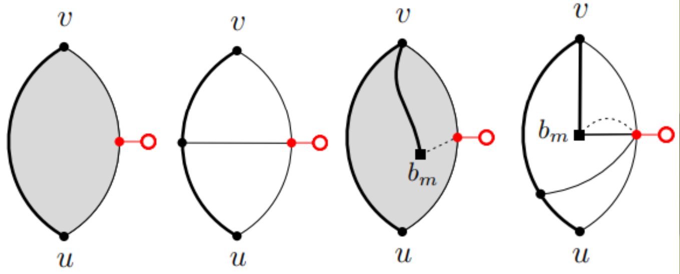


BP-type and BB-type with one caterpillar and no internal face



(d) BP (e) BB

(2) If $A(E_\mu)$ consists of one caterpillar and no internal faces, then its backbone is only $L(E_\mu)$ or $r(E_\mu)$.



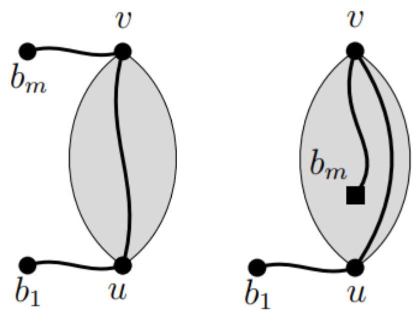
-BP2 -BP2 -BB2 -BB2

proof. By contradiction, if $A(E_\mu)$ does not contain $L(E_\mu)$ and $r(E_\mu)$, then there are no export, then $A(E_\mu)=A(E)$, but $b_1 \notin H_\mu$, a contradiction.

$A(E_\mu)$ does not contain both $L(E_\mu)$ and $r(E_\mu)$ because the path (u,v) separates $L(E_\mu)$ and $r(E_\mu)$.

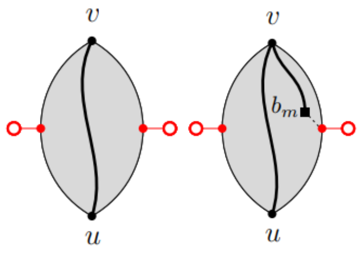
$A(E_\mu)$ does not contain any internal red face

BP-type and BB-type with two caterpillars and no internal face



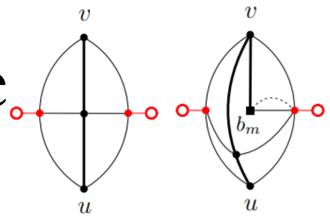
(d) BP (e) BB

(3) If $A(E_\mu)$ consists of two caterpillars and no internal face, then the backbone of one caterpillar starts at $L(E_\mu)$ and the other one starts at $r(E_\mu)$;



-BP3 -BB3

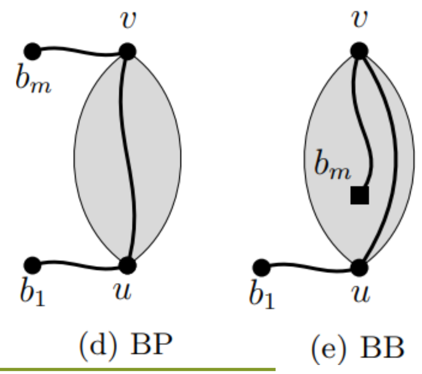
The backbone of both two caterpillars is a single vertex.



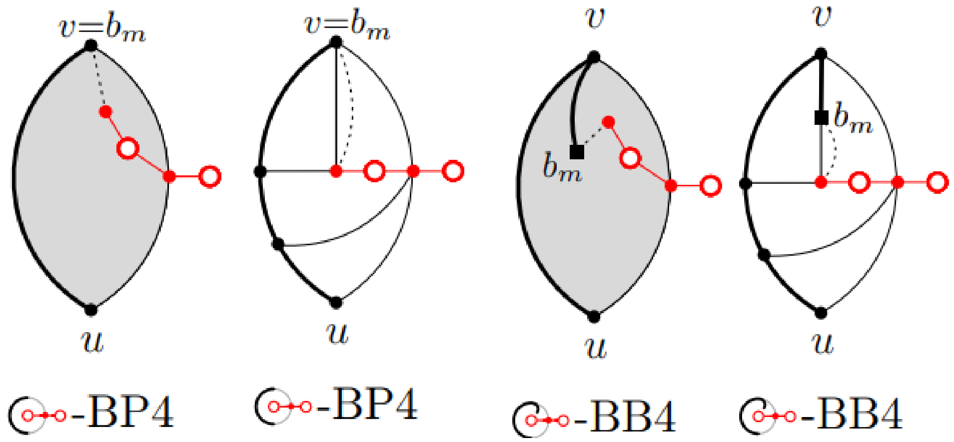
-BP3 -BB3

proof. Because there are no internal red face and there are two caterpillars.

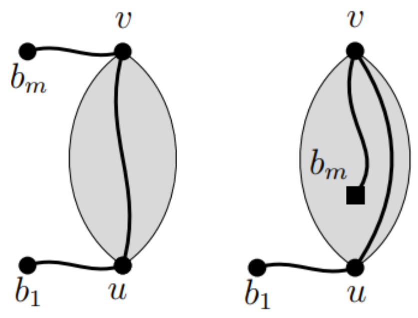
BP-type and BB-type with one caterpillar and at least one internal face



(4) If $A(E_\mu)$ consists of one caterpillar and at least one internal face, then the backbone of one caterpillar starts at $L(E_\mu)$ or $r(E_\mu)$ and one internal face is the end-vertex of the backbone.



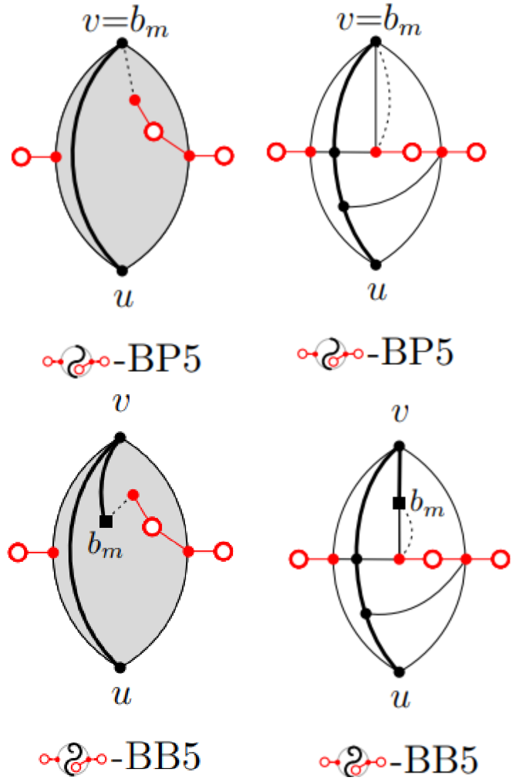
BP-type and BB-type with two caterpillars sand at least one internal face



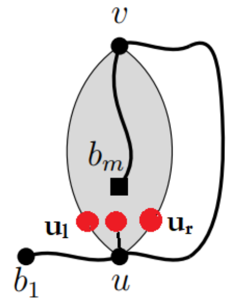
(d) BP (e) BB

(5) If one of the caterpillars composing $A(E_\mu)$, say C, contains a vertex corresponding to an internal face of E_μ, then b_m belongs to H_μ

($b_m = v$ if μ is of type -BP, while $b_m \notin \{u, v\}$ if μ is of type -BB);

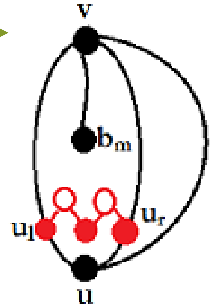


BF-type



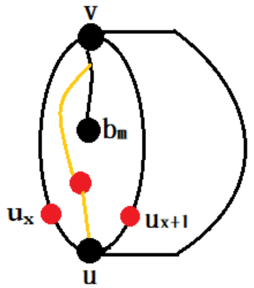
(f) BF

There exist a path Q_u in $A(E_\mu)$ between u_l and u_r ,Q_u passes through all the internal faces of E_μ incident to u.



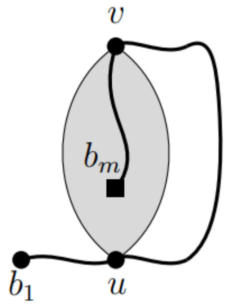
proof. By contradiction,$\{u_l=u_1,u_2,\ldots, u_r\}$are the red vertexes incident to u , if $\exists$ u_x and u_{x+1} do not share the same face ,then exsits a red vertex y separates u_x and u_{x+1},

but if it will separate u_x and u_{x+1} , then y is incident to u , a contradiction.



$\Rightarrow$ μ has at most one caterpillar.

BF-type

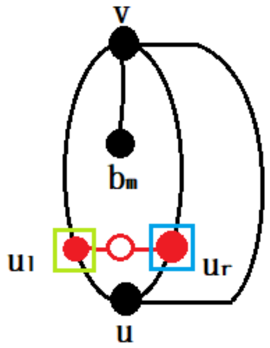


Suppose that E_μ contains a red face and let f_h and f_t be the end-vertices of the backbone B_μ.

(1) b_m shares a face with a leaf of $A(E_\mu)$ adjacent to one of f_h and f_t ,say f_t .

(2) f_h is either an outer face of E_μ or is an internal face of E_μ incident to u_l/u_r; in the latter case, u_l (resp. u_r) is a leaf of $A(E_\mu)$.

proof. Because $b_1 \notin H_\mu$, so one side of Q_u is going to b_m, the other side is going to an export to b_1(outer face or internal face incident to u_l/u_r)

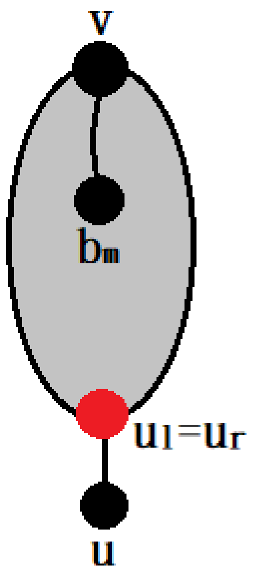


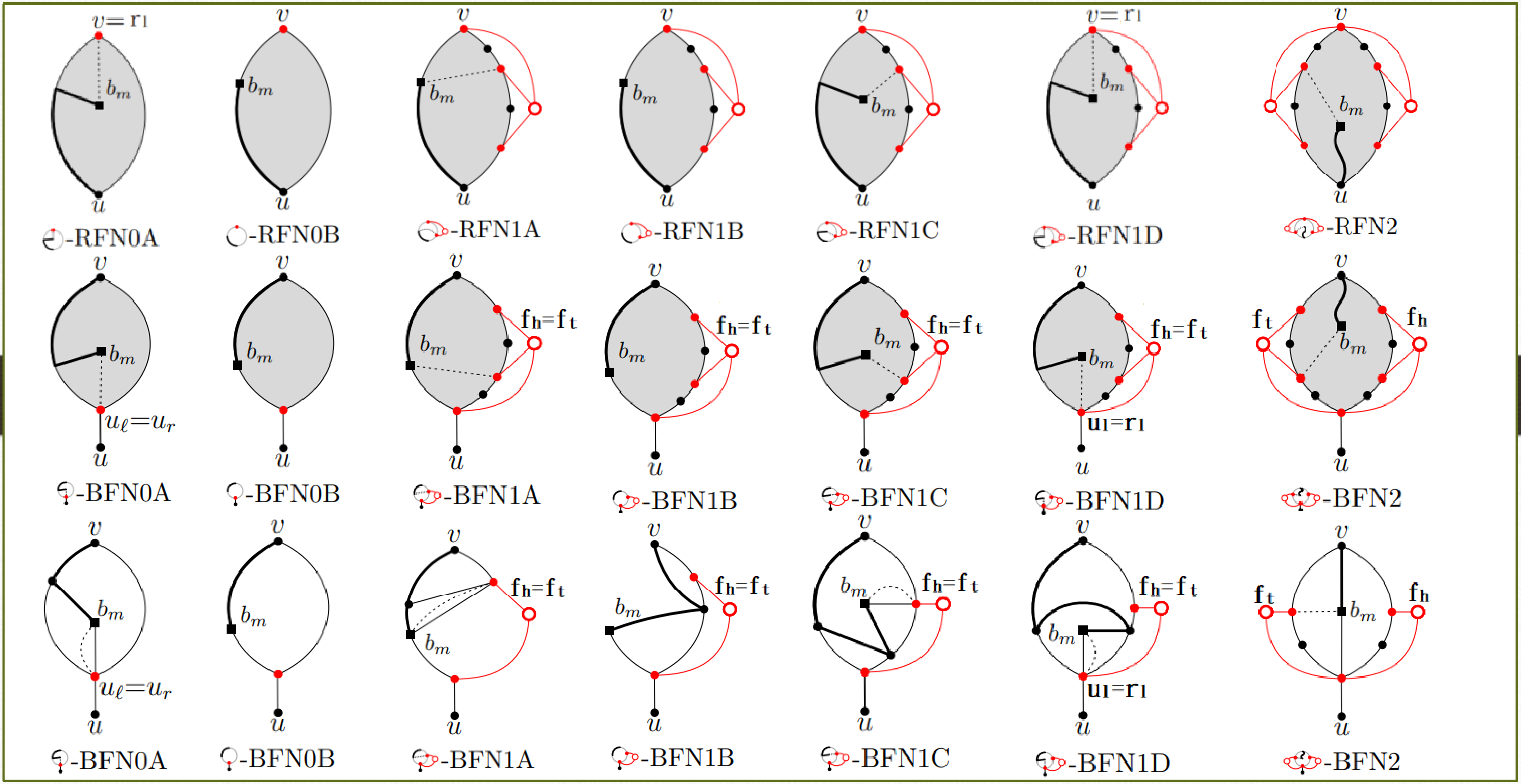
BFN-type

Because BFN-type does not have any internal faces, so the Q_u is a single vertex $u_l = u_r$.

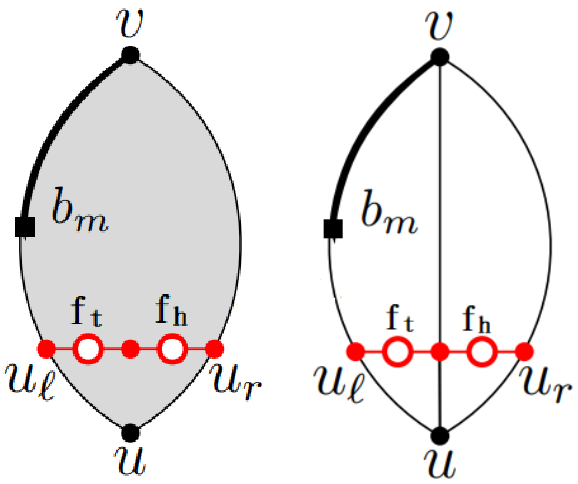
The u_l is a cut vertex, so there is a child of μ whose poles are u_l and v, and it's type is RF-type.

So we classify the BFN-type according to the type of the node whose poles are u_l and v.

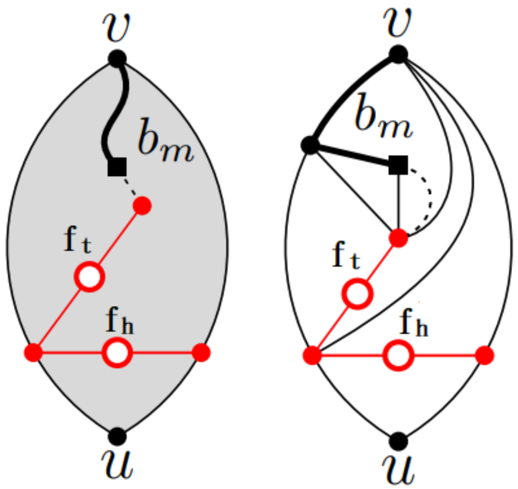



-RFN0A
-RFN0B
-RFN1A
-RFN1B
-RFN1C
-RFN1D
-RFN2
-BFN0A
-BFN0B
-BFN1A
-BFN1B
-BFN1C
-BFN1D
-BFN2
-BFN0A
-BFN0B
-BFN1A
-BFN1B
-BFN1C
-BFN1D
-BFN2

BFI0-type



-BFI0A -BFI0A



-BFI0B -BFI0B

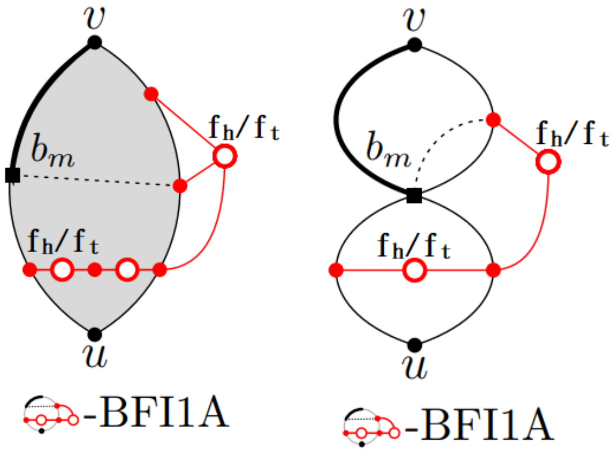
(1) b_m is incident to the outer face.
(2) The backbone of $A(E_\mu)=Q_u$

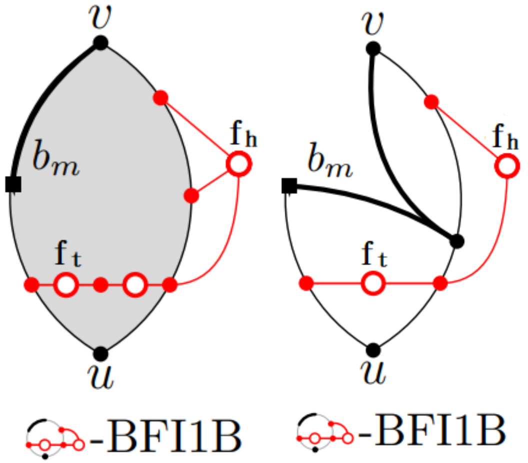
Otherwise

BFI1-type

Suppose r(E_μ) is red and B_μ does not contain any other internal red vertex except for Q_u and assume b_m is incident to L(E_μ),then we separate two types.

b_m shares a face with a leaf of A(E_μ) incident to r(E_μ)



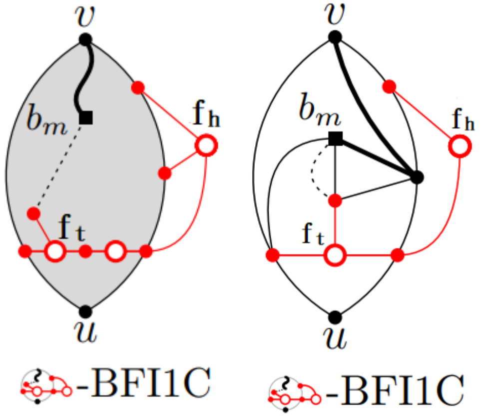


b_m does not share a face with a leaf of A(E_μ) incident to r(E_μ)

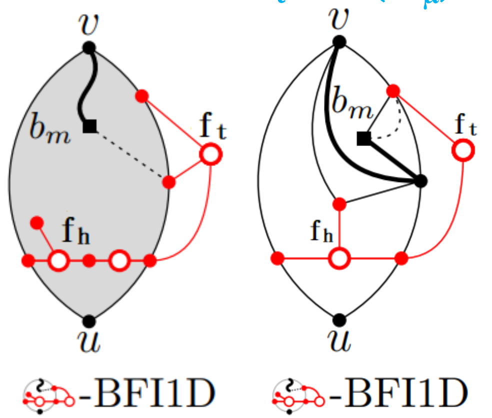
BFI1-type

Suppose $r(E_\mu)$ is red and B_μ does not contain any other internal red vertex except for Q_u and assume b_m is not incident to $L(E_\mu)$,then we separate three types.

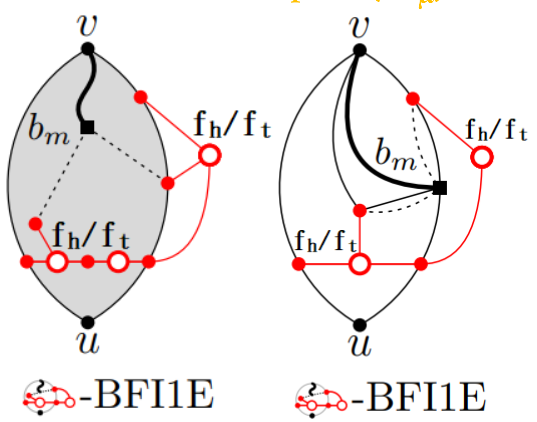
b_m only shares a face with a leaf of $A(E_\mu)$ incident to $f_t = f_l$



b_m only shares a face with a leaf of $A(E_\mu)$ incident to $f_t = r(E_\mu)$

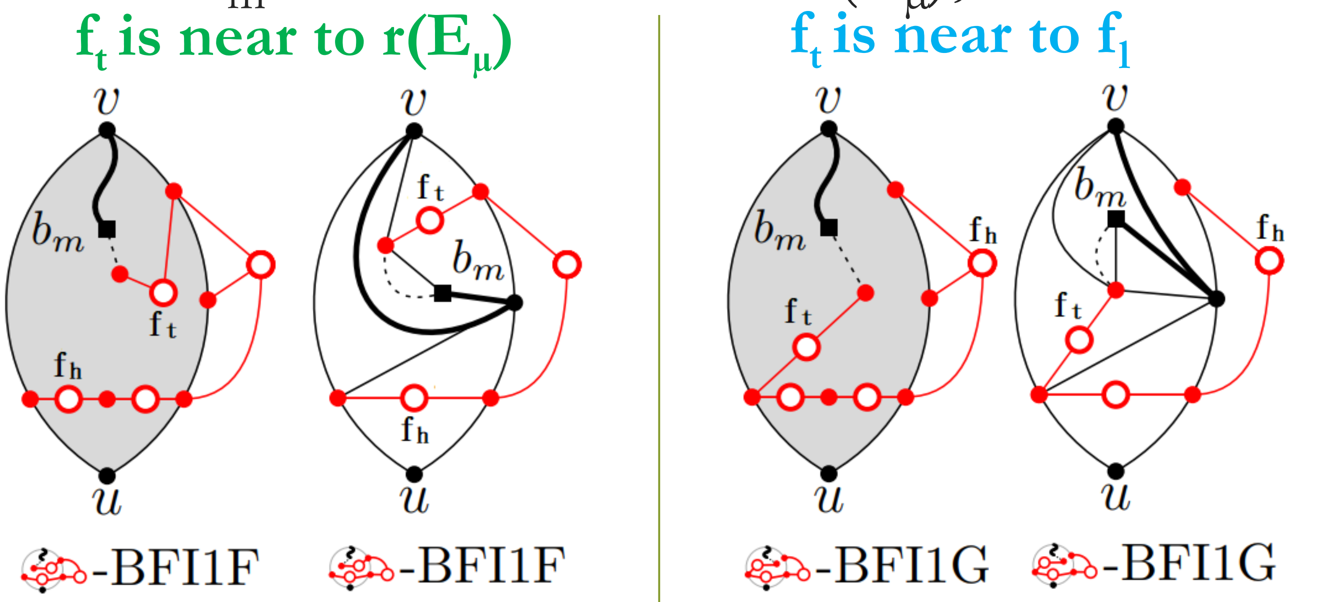


b_m both shares a face with a leaf of $A(E_\mu)$ incident to $f_l/r\ (E_\mu)$



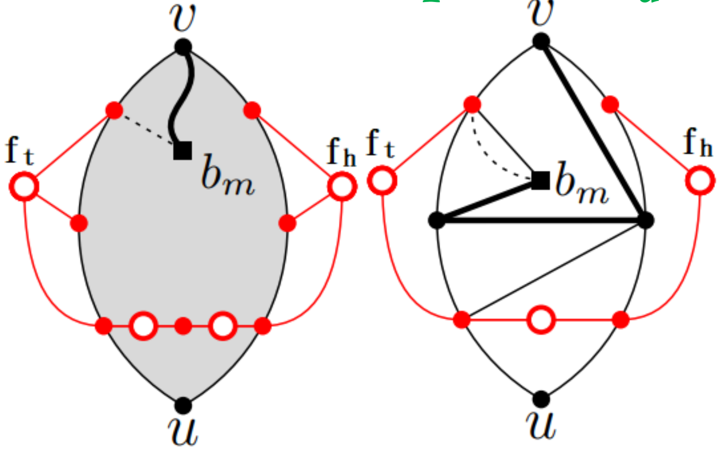
BFI1-type

Suppose $r(E_\mu)$ is red and B_μ contains any other internal red vertex except for Q_u and assume b_m is not incident to $L(E_\mu)$,then we separate two types.



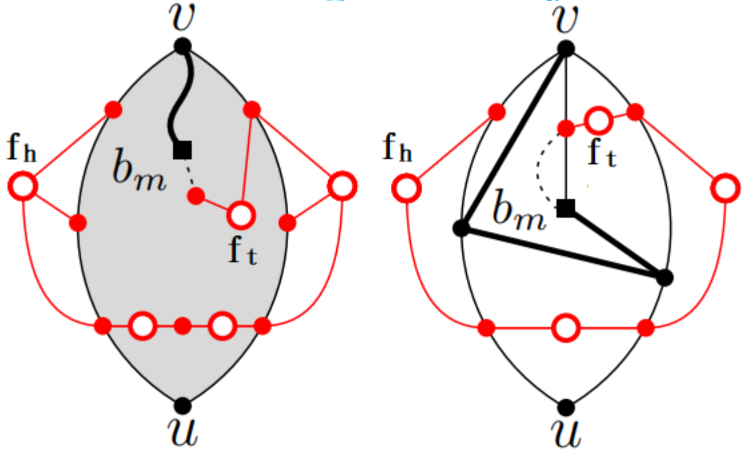
BFI2-type

There are no internal red face except for Q_u



-BFI2A -BFI2A

There are internal red face except for Q_u



-BFI2B -BFI2B

The Correctness proof of the replacement graph

The replacement graph

The proof strategy is if we change one virtual edge to replacement graph with the same **filp** , then the type/flip of this node will not change. We can derive this lemma to all virtual edges.

We will show the replacement graph will not influence these little features after using them to subdivide the types and verify the three big features at the last.

The big type of the node will not change

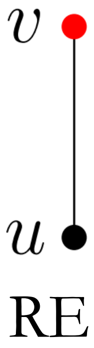
Let E_m is E_μ change one virtual edge to replacement graph

(1) The poles of E_m is the same as the E_μ

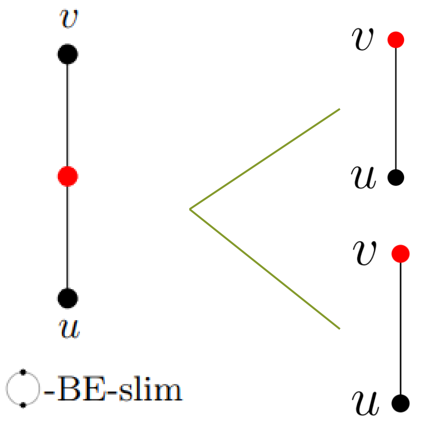
(2) If the child of E_μ include part of the black path, the replacement of it will include a black path which can play the same role, so the existance of the E_m's black path will not change.

(3) Whether b_m is a pole or not will not change.

The subdivision of BE-slim and RE-type

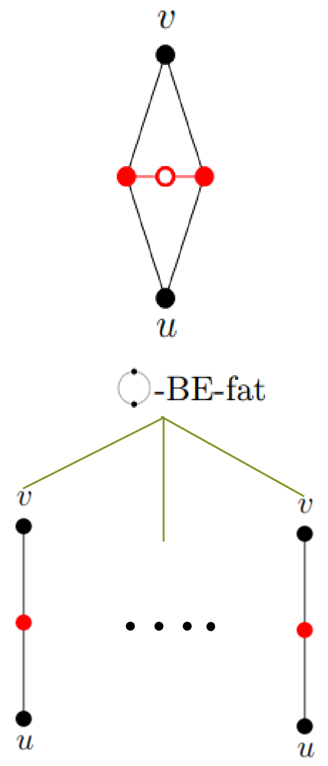


Because RE-type is a single edge and there are no children of it, so the replacement graph will not influence it.



The BE-slim must be an S node and the children of it are two RE-node, so after using replacement graph, the graph of this node will not change.

The subdivision of BE-fat



The BE-fat are all P-node whose son are all BE- slim.
The replacement graph of BE-slim are the same as the original graph, so the replacement graph will not influence the result of BE-fat type nodes.

The subdivision of BB and BP-type

The all type of BB and BP-type are dicided by (F1)~(F3) and whether b_m belongs to H_μ.
We will prove the three features of H_μ will not be influenced latter.

Common Small Features

Whether b_m is an internal vertex.

If b_m is an internal vertex of μ.(b_m still not an internal vertex is trivial)

Let v is one child of μ which includes b_m .

Assume b_m is an internal vertex of v, then after replacing, it still an internal vertex of v **=> b_m is still an internal vertex of μ.**

Assume b_m is not internal vertex of v, then after replacing, it still an incident to at least an outer face, these outer face are the faces on sk(μ) and they are not the outer faces of μ(otherwise b_m is not an internal vertex of μ). **=> b_m is still an internal vertex of μ.**

Whether a face belongs to sk(μ) is red

Let the face incident to the virtual edge f:

If there are more than two/one red vertex in the virtual edge incident to f, there are still more than two/one red vertex in the virtual edge incident to f after replacing.

(1) There are more than two red vertex incident to f if and only if there are more than two red vertex incident to f.

(2) There is one red vertex incident to f if and only if there is one red vertex incident to f.

Common Big Features

There are an internal face in E_μ if and only if there are an internal face in E_m---------- (F3)

If there is at least one internal red face not in sk(μ), otherwise all of them are still red in above proof.

Let v is the child includes of it, then after replacing, there is at least one internal red face in v. => At least one internal red face in μ.

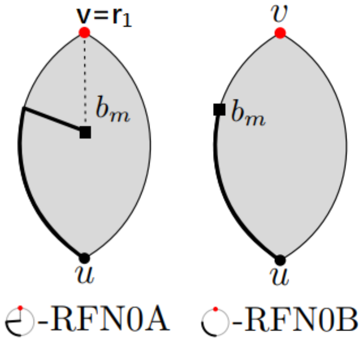
The outer face is red in E_μ if and only if it is red in E_m---------- (F2)

The outer face belongs to sk(μ), so by the proof in above section, the replacement graph will not influence whether it is red .

The proof of (F1)Number of Caterpillars is put in the last in this part.

The subdivision of RF -type

RFN0-type

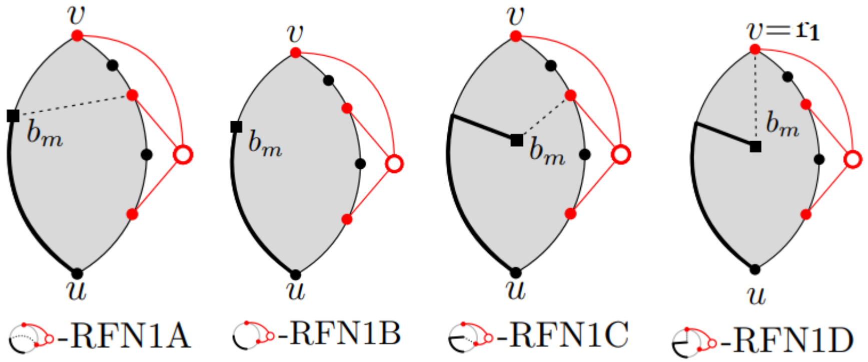


b_m is an internal vertex of E_m if and only of it is an internal vertex of E_μ . **-------Be proved in above section**

b_m will share a face with the pole v after replacing

proof. We only consider the child v includes of b_m and the red pole, the red pole is a pole of v, then b_m still shares a face with the red pole after replacing. **-------By (F5) Whether b_m shares a face with the red pole**

RFN1-type



b_m is an internal vertex of E_m if and only of it is an internal vertex of E_μ .

-------Be proved in above section

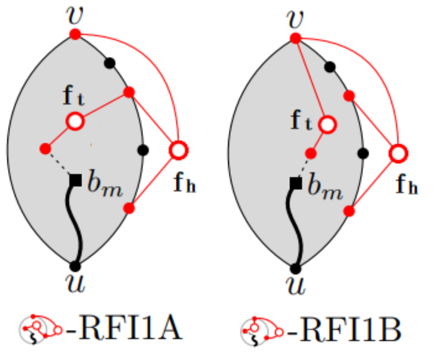
b_m shares a face with non-pole red vertex of E_m if and only of b_m shares a face with non-pole red vertex of E_μ.

proof. Consider the child v includes of b_m and non-pole red vertex x.

If x is a red pole of v **-------Be proved by (F5) Whether b_m shares a face with the red pole**

If x is not a pole of v**-------Be proved by (F6) Whether b_m shares a face with one non-pole red vertex.**

RFI-type



(1) The $r(E_m)$ is red if and only if $r(E_\mu)$ is red.

(2) There are internal faces in E_μ if and only if there are internal faces in E_m.

(3) The internal faces should connect to the red pole or $r(E_m)$.

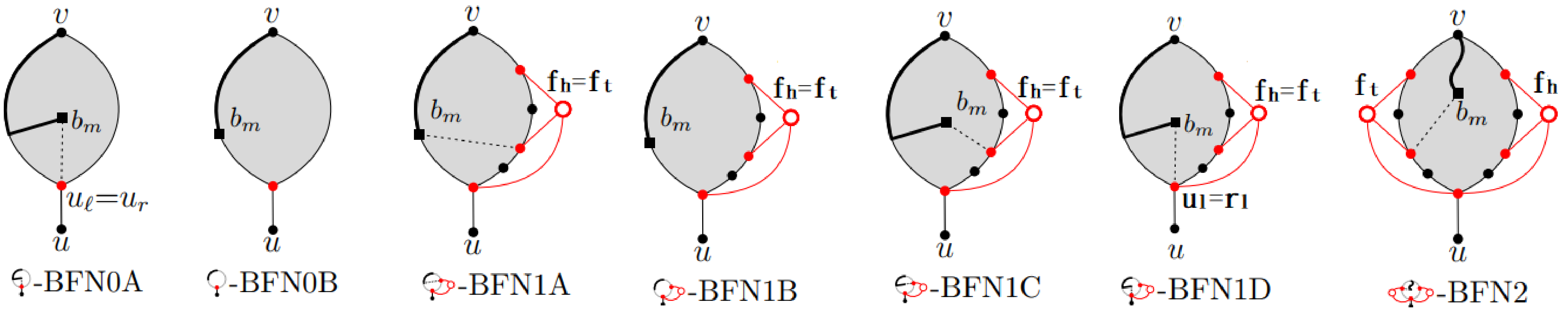
(4) Color of the faces incident to red pole and in sk(μ) will not change.

(5) The internal red face incident to red pole and belongs to one virtual edge will not be influenced.--------**(F4) the degree of the red pole of in $A(E_\mu)$**

The subdivision of BF -type

BFN-type

The subdivision of the BFN-type are decided by the child which pole is (u_l=u_r , v), the subdivision of this RFN-type does not change by the replacement graph.



Lemma : Face $f_l(E_\mu)/f_r(E_\mu)$ is an end-vertex of a backbone of $A(E_\mu)$ if and only if $f_l(E_m)/f_r(E_m)$ is an end-vertex of a backbone of $A(E_m)$.

proof. If $f_l(E_\mu)/f_r(E_\mu)$ is a face of $sk(\mu)$,then it keep red.

If $f_l(E_\mu)/f_r(E_\mu)$ is an internal face of the child of μ, then it goes to the export through u_l/u_r after replacing.

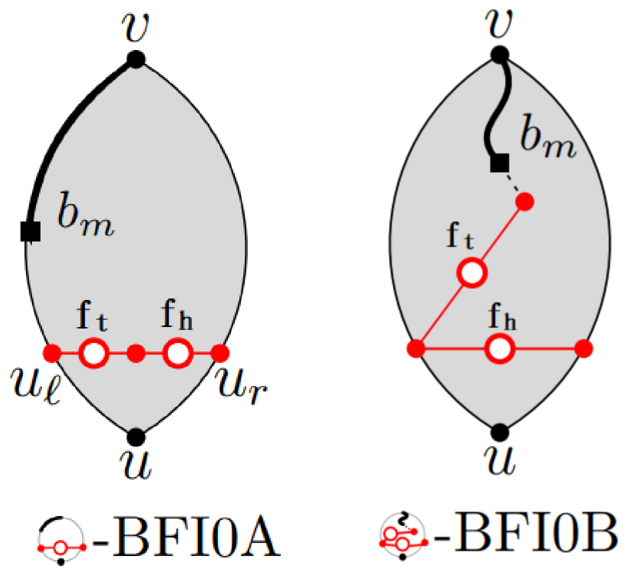
Lemma : The end-vertex of a backbone of $A(E_\mu)$ that is adjacent to the leaf r_1 of $A(E_\mu)$ that shares a face with b_m in E_μ is $L(E_\mu)$, $r(E_\mu)$, $f_l(E_\mu)$, $f_r(E_\mu)$, or is none of such faces, if and only if leaf r_1 shares a face with b_m is the correspond face in $E_{m.}$

The face b_m shared with r_1 is red and is one of the four faces if it is the end-vertex of $A(E_\mu)$, we already proved that they are still the end-vertex of $A(E_m)$ and b_m is still incident to it.

If it is not red and is one of the four faces, then r_1 is the only red vertex incident to this face, so it will be incident to this face in E_m

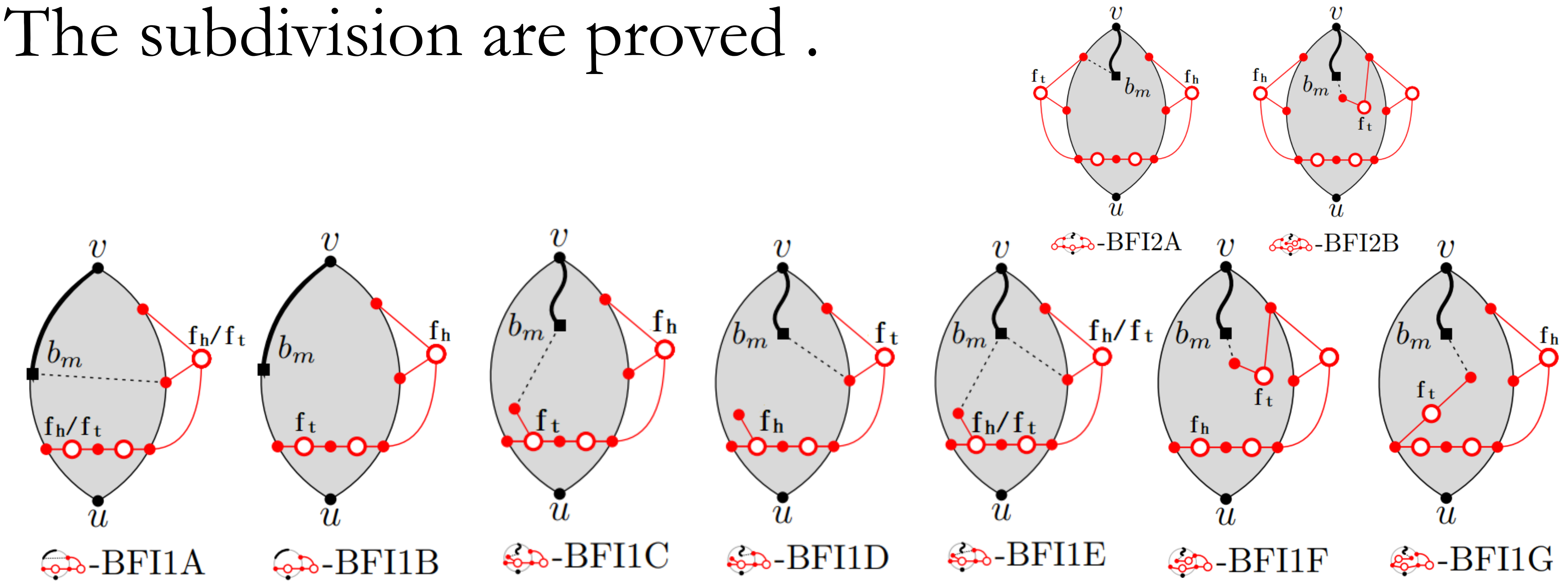
BFI0-type

Whether b_m an internal vertex will not be influenced by the above proof.



BFI1 and BFI2-type

The subdivision are proved .



(F1) Number of Caterpillars

(1) The BE、RE are the same after replacing.

(2) The BB、BP :Number of caterpillars are the same with the number of the outer faces.

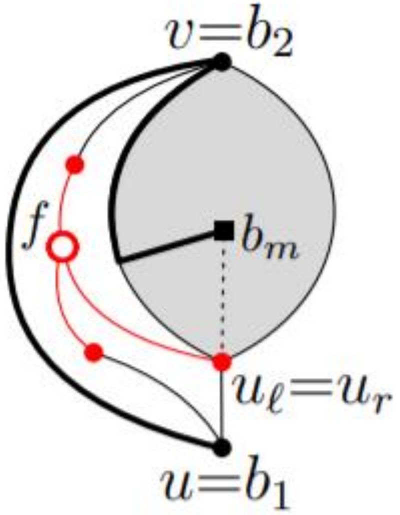
(3) RF: The degree of the red pole are the same.

(4) BF: There are at least one vertex in Q_u after replacing(otherwise u is disconnected).

Use the child of the root to construct B2BE

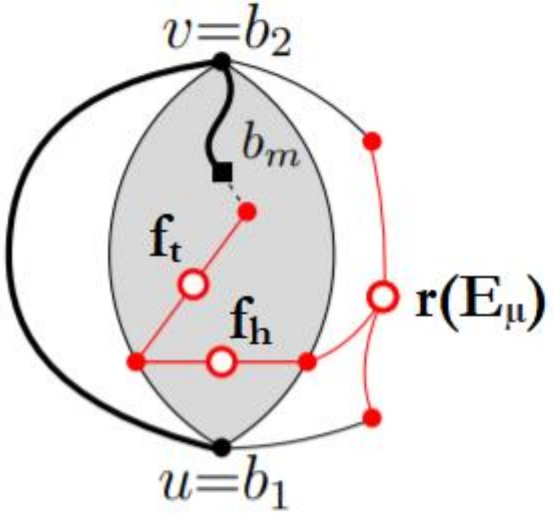
Case 1: E_μ is -BFN0

- There exists an solo red vertex incident to u and an solo red vertex incident to v. (At least three red vertex)
- $u_l = u_r$
- Put these two solo red vertexes and (u , v) into the same face f, f ∈ {L(E_μ), r(E_μ)}
- There exists a leaf incident to the end-vertexes and r_1



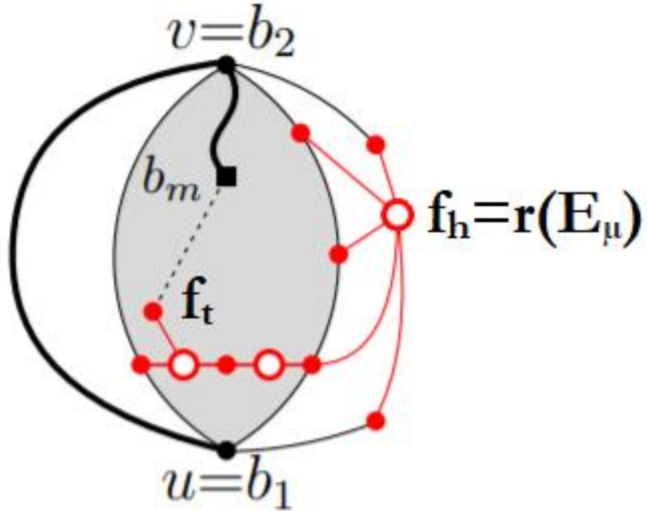
Case 2: E_μ is -BFI0

- b_m shares a face with a leaf of $A(E_\mu)$ adjacent to f_t
- f_h is an internal face of E_μ incident to $u_r(u_l)$, and u_r is a leaf of $A(E_\mu)$.
- If exist some solo red vertexes, put them into the face different with (u,v)
- There exists a leaf incident to the end- vertexes and r_1



Case 3: Others

- At least one of $L(E_\mu)$ or $r(E_\mu)$ is red, say $r(E_\mu)$
- Put all solo red vertexes into $r(E_\mu)$
- There exists a leaf incident to the end- vertexes and r_1



How to embed solo red vertex

If G has Caterpillar Property if and only if it has Ordered Property.

Ordered Property : A embedding E of H has Caterpillar Property , for every solo red vertex (u , r), it satisfies the following property :

These vertexes are embedded in one face of the highest node which includes of u.

Proof of Ordered Property

Observation: The pole of the highest node which includes of u is not u.

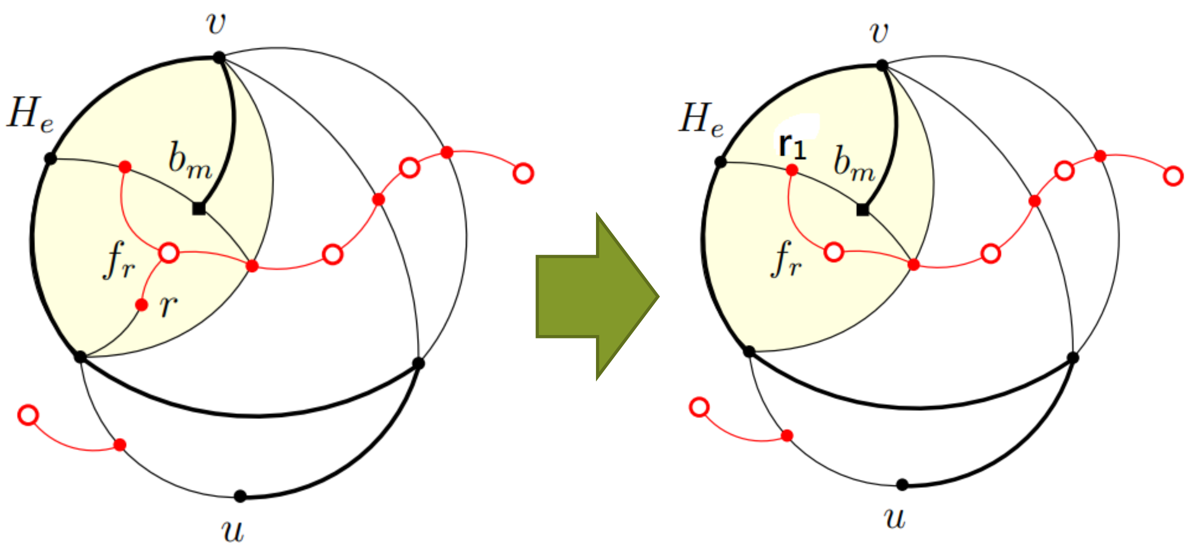
That is the all solo red vertexes incident to the pole are in $L(E_\mu)/r(E_\mu)$

(1) Removing all the solo red vertexes will not influence the Caterpillar Property of G.

(2)Add them back to G by Ordered Property will not influence the Caterpillar Property .

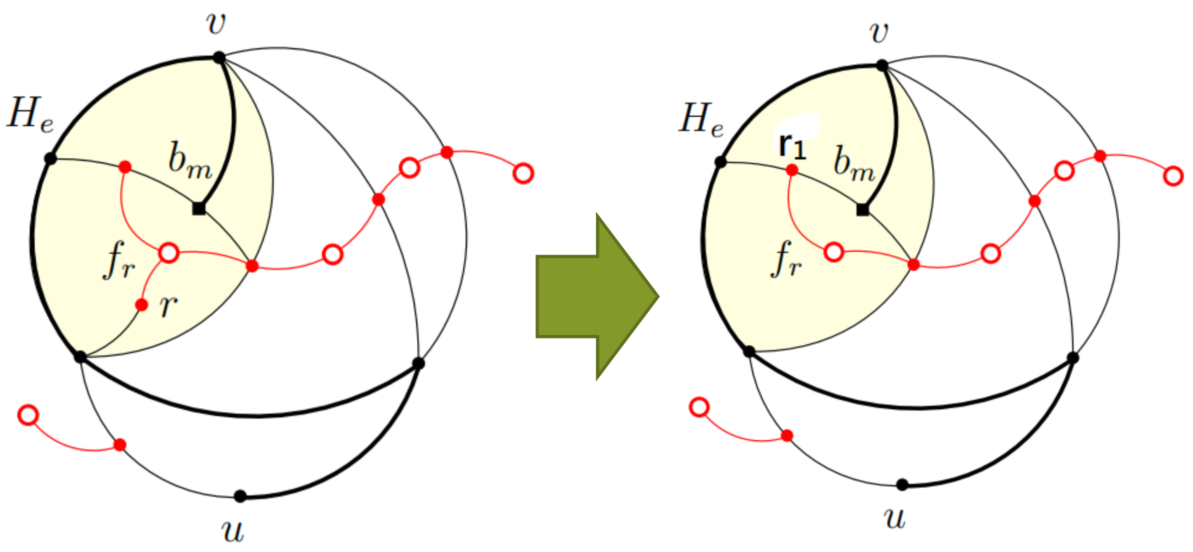
Case 1. f_r remains red after the removal of r

Let r is a solo red vertex of e and incident to the pole , and the parent of e is the highest node includes b , f_r is an internal node of e. After removing r, there is still a suitable r_1 (Because b_m is incident to f_r)



Case 2. f_r is not red after the removal of r

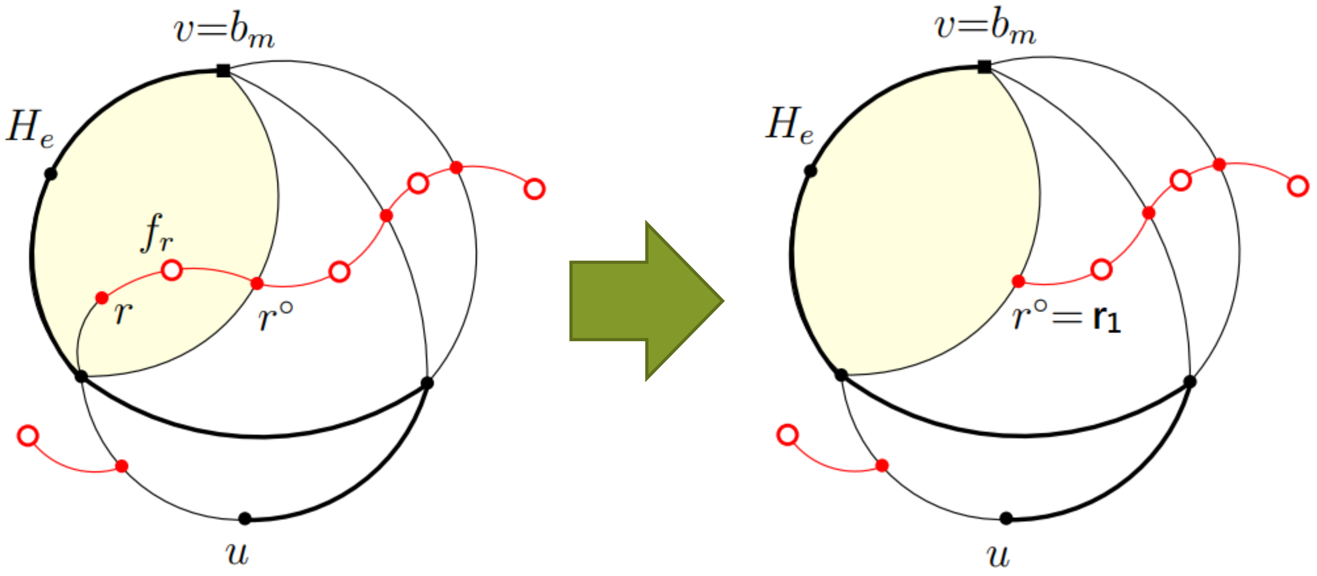
Let r is a solo red vertex of e and incident to the pole , and the parent of e is the highest node includes b , f_r is an internal node of e. After removing r, there is still a suitable r_1 / r_p(Because b_m is incident to f_r)



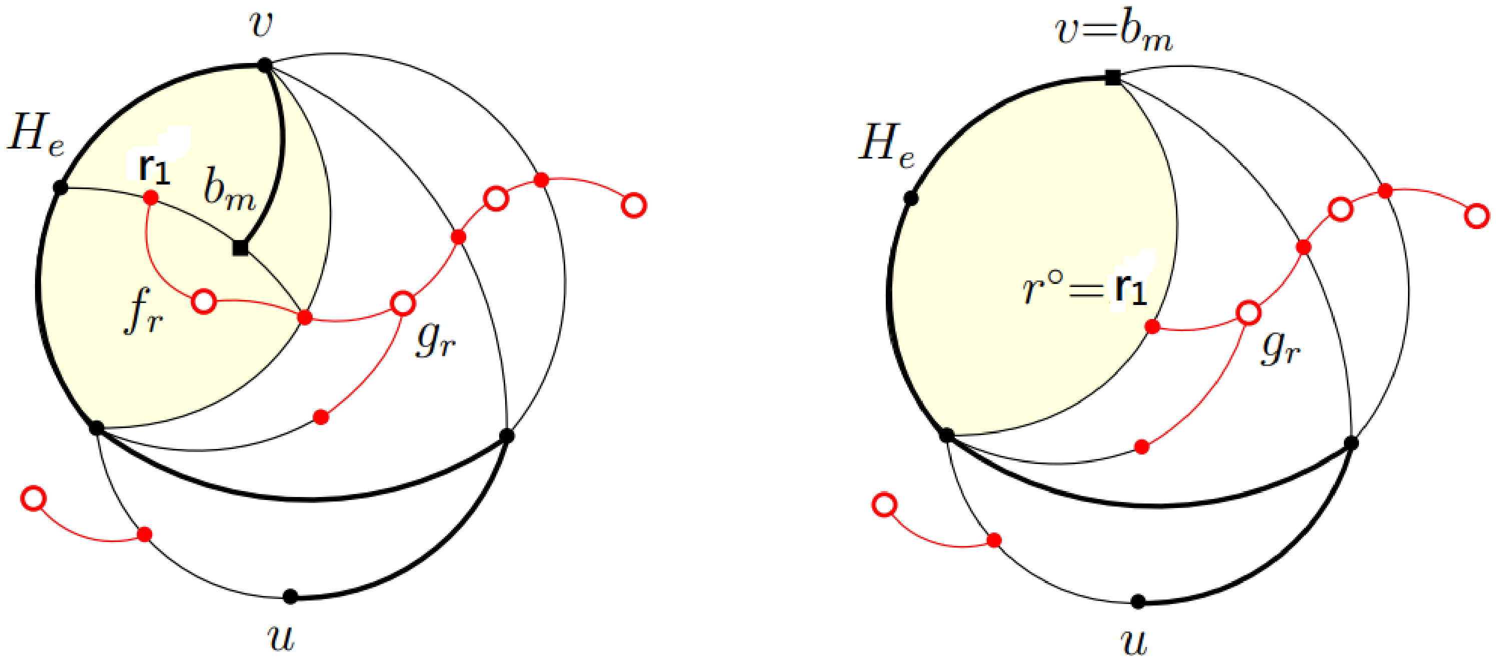
Case 2 f_r is not red after the removal of r

f_r is an end-vertex of the backbone ,and r∈ $\{r_1, r_p\}$,

After removing, the else red vertex incident to f_r is the new r_1/r_p

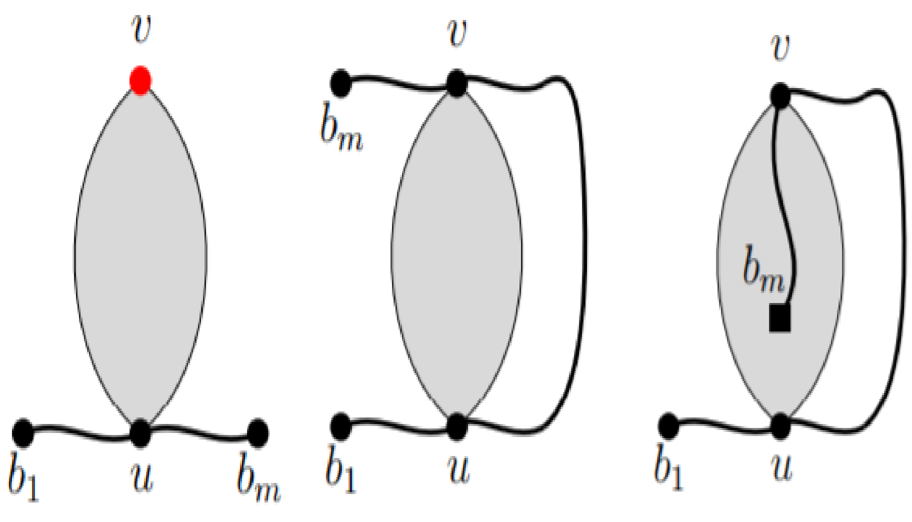


If one of $\{g_l , g_r\}$ is red



None of $\{g_l , g_r\}$ is red

Case 1: $H \setminus H_e$ contains no red vertex



(a) RE (c) BE (f) BF

- Situation 1 : $b_1 \in H_e$
- b_1 is a pole of e and e is of type either -BE, or -BF, or –RE
- f_r is an internal red face of E_e=> e is not of type -RE.

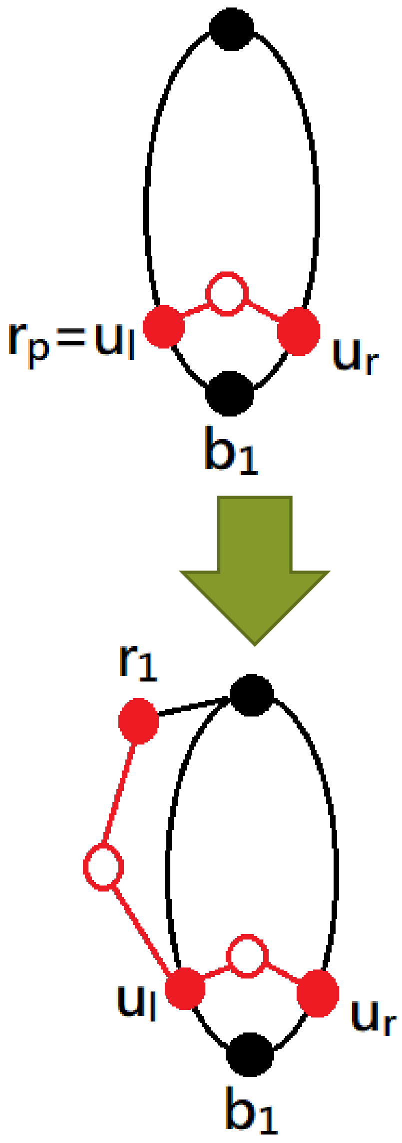
(1) All the neighbors of b_1 in H_e are red. Let u_l and u_r be the neighbors of b_1 incident to g_l and g_r.

(2) No red vertex different from u_l and u_r is incident to g_l and g_r

(3) $r_p = u_l$ or $r_p = u_r$, say $r_p = u_l$

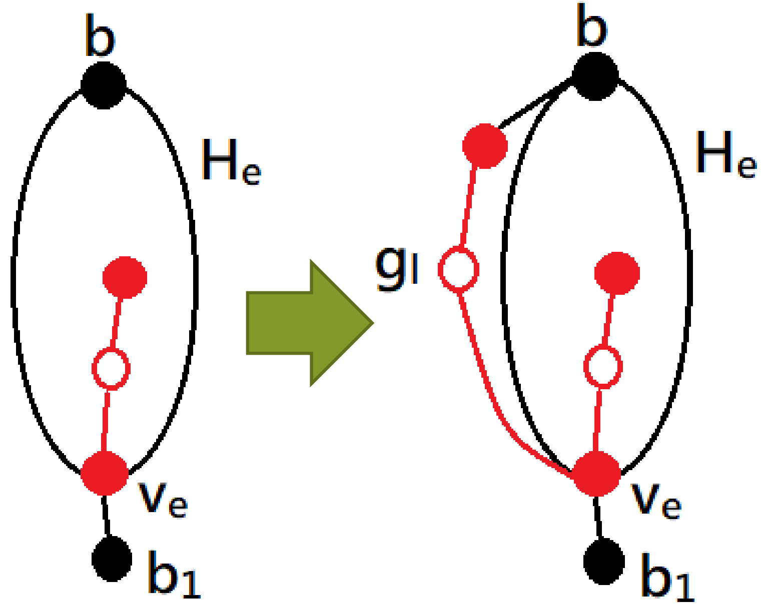
(4) u_l is a leaf of A(E) adjacent to an end-vertex of the backbone of A(E)

(5) Placing r and (b, r) inside g_l



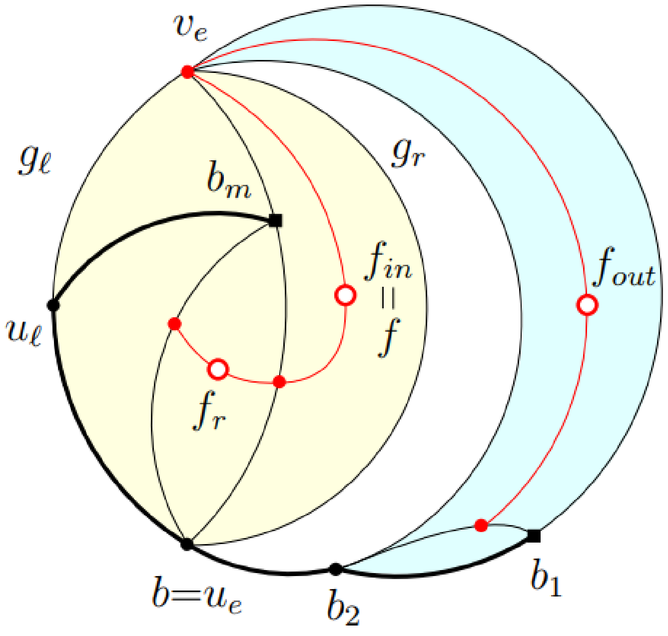
Case 1: $H \setminus H_e$ contains no red vertex

- Situation 2 : $b_1 \notin H_e$
- b_1 has at least one red neighbor in H
- b_1 is adjacent to a red pole v_e of e
- v_e is a leaf of A(E) adjacent to an end-vertex of its backbone (v_e is incident to b_1 otherwise H\solo red vertexes is not biconnected).
- Put (b,r) into g_l or g_r



Case 2: $H \setminus H_e$ contains at least one red vertex

- v_e of e different from b is red (otherwise no export)
- e is type -RF, since e is not type –RE because f_r is red
- v_e belongs to the backbone of A(E), and it is incident to a red face f_{in} internal to E_e and to a red face f_{out} not belonging to E_e

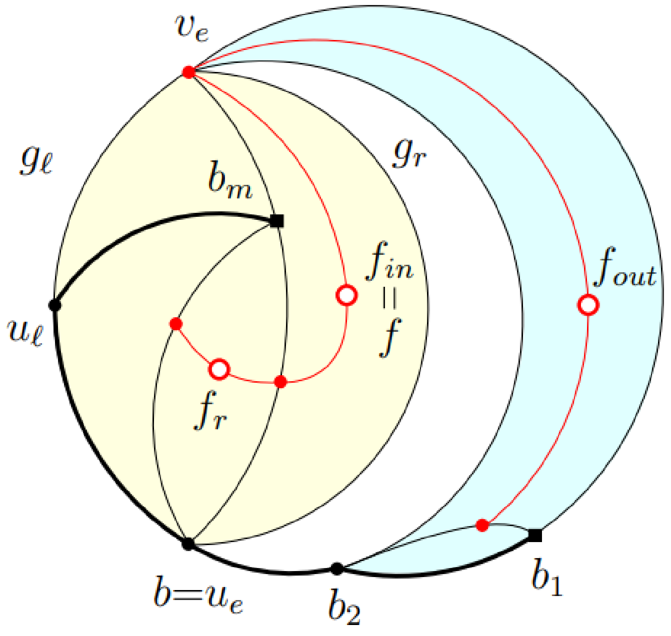


Case 2: $H \setminus H_e$ contains at least one red vertex

- $b = u_e$ is incident to f_r, there exists at least an internal face f_r of E_e that is incident to u_e

  $\Rightarrow u_l \neq u_r$
- One of $\{u_l, u_r\}$ is red . (Because RF-type)

  $\Rightarrow$coincide with v_e
- H_e contains the edge ($b=u_e$, v_e)



f is red in E

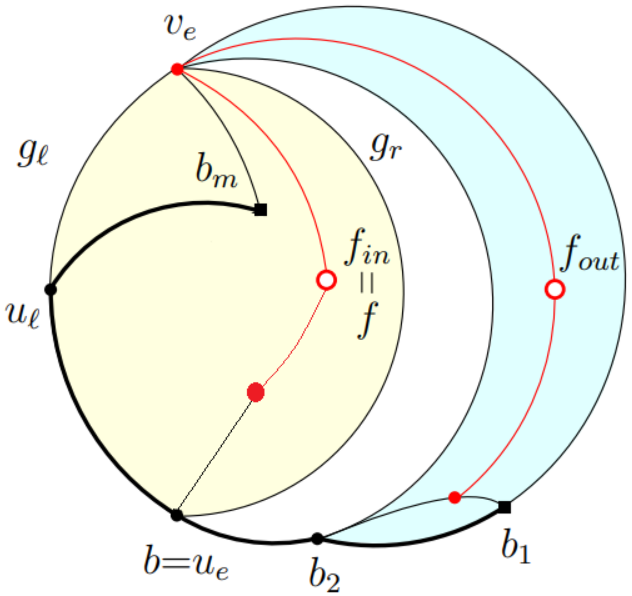
f_r is between u_l and (u_e, v_e)

Case 1: $f = f_r$

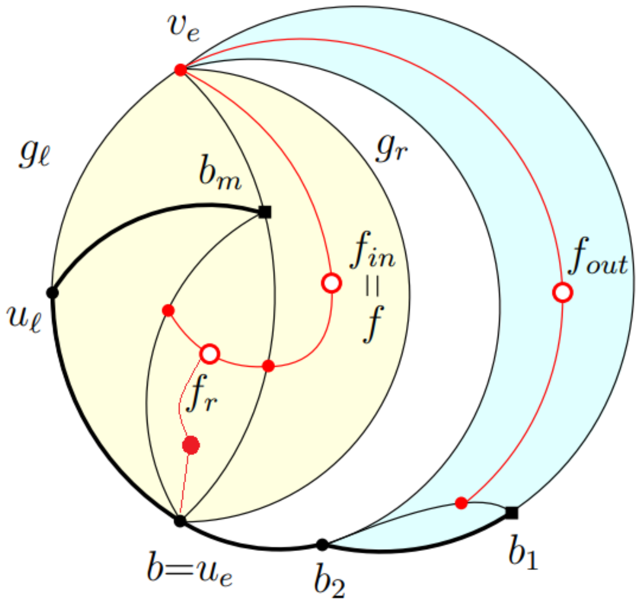
Case2 :

f_r is serarated from (u_e, v_e),

then there exists a red vertex divide them into f_r and f

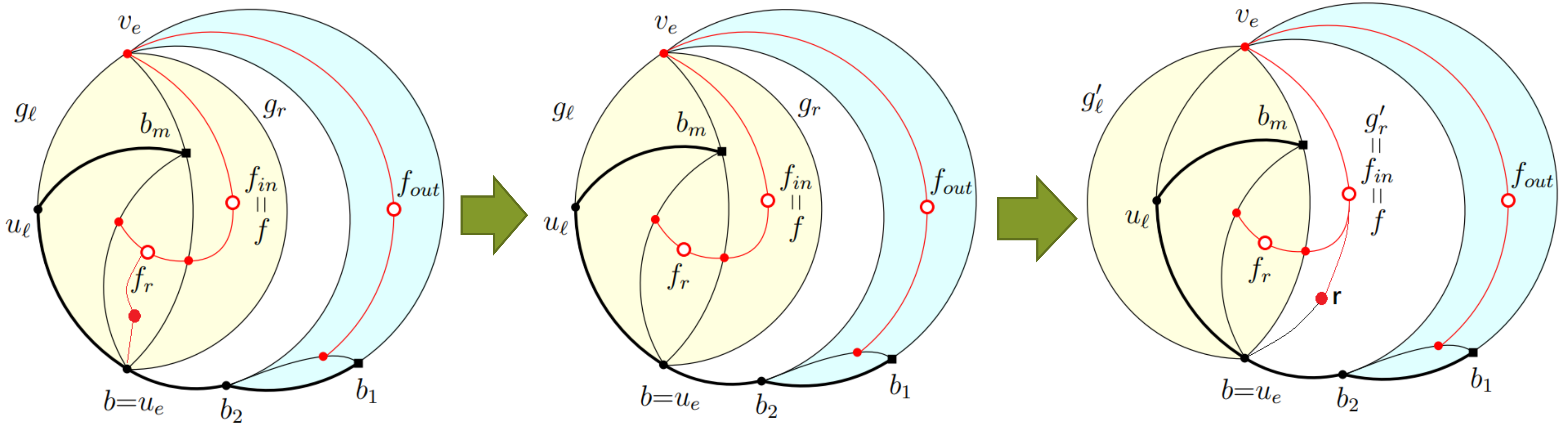


Case 1

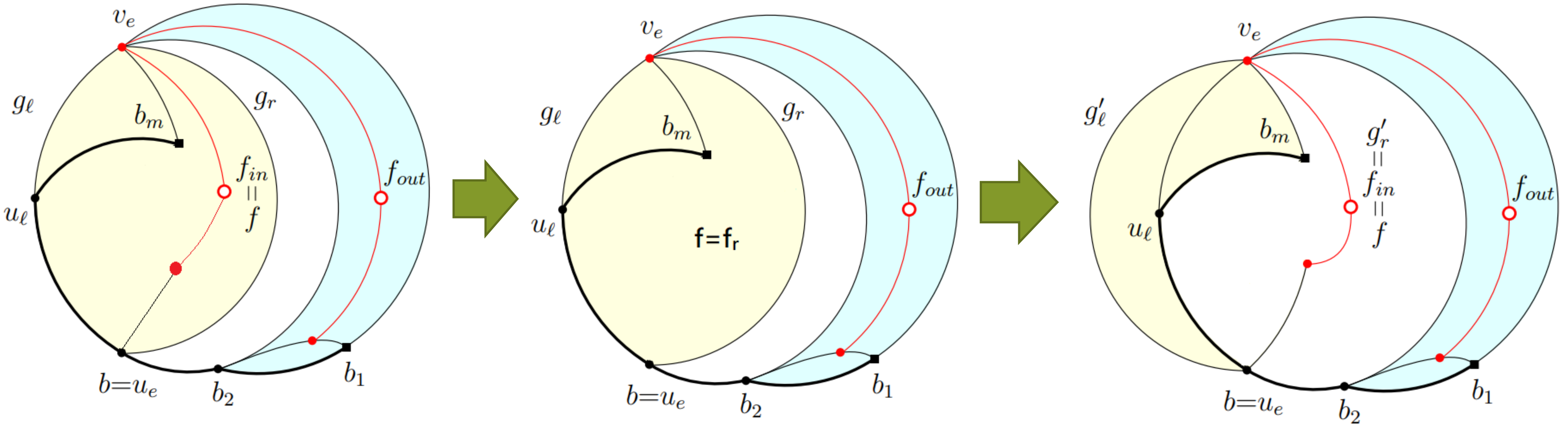


Case 2

f is still red after removing the solo red vertex



f is not red after removing the solo red vertex



Handle solo red vertex

By Ordered Property, when we climb the tree, we can embed the solo red vertex r when μ is the highest node of b.

Proof that we can check each node in sk(μ)

Review:

By the replacement graph which is like Thévenin's theorem, we can check a graph in O(1), and the possibility type of each node is O(1).

The only obstacle to linear algorithm is the permutation of the skeleton(μ) may be **exponential** of the sk(μ).

So we will show that the permutation of the skeleton(μ) is finite, and the virtual edges may have more than 2 types are only a small part.

Q-node

Each Q node is a single edge, so we can check it in O(1) time.

The Q node is not the highest node of its pole.

=>No solo red vertex are embedded.

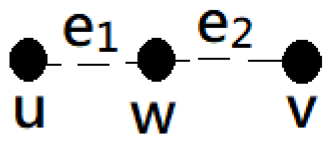
S-node

Lemma :

Each type of S-node can be checked in O(1) time.

proof.

The SPQR tree version is that each S-node has only 2 virtual edges, so we can check each type of the node in O(1) time.

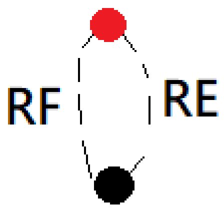


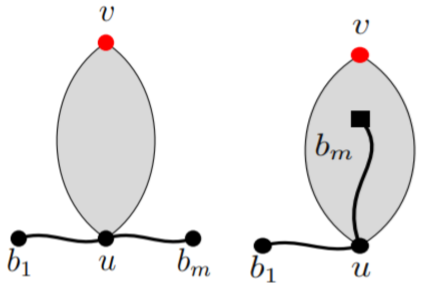
μ is only the highest node of w, we only need to consider the type of e_1/e_2, and embed the solo red vertex incident to w in $L(E_\mu)/r(E_\mu)$ for each type, it can be done in O(1).

P-node

Lemma :

If μ is a P-node such that v_{μ} is red, then the type of μ is -RF and it has exactly two children.
One of them is of type -RE and the other is of type –RF.





(a) RE (b) RF

We can list all of the possibility in O(1) and we do not need to consider any solo red vertexes.

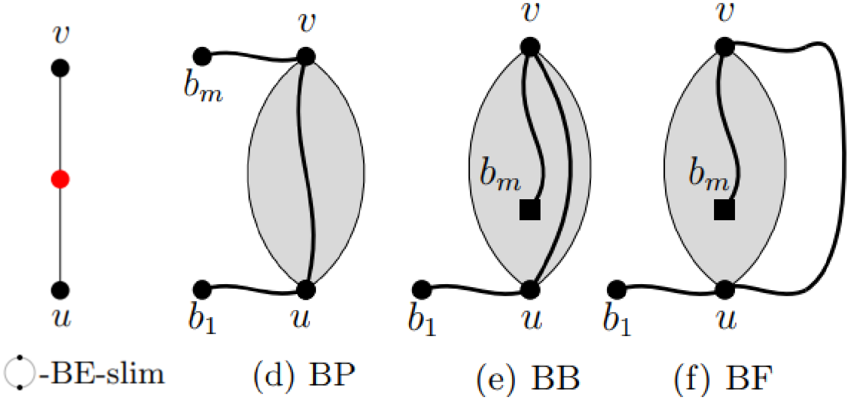
Lemma:

If μ is a P-node such that v_μ is black, then μ has at most two children that are not of type -BE slim. Each of them is of type -BP, -BB, or -BF. If there exist exactly two such children, then one is of type -BP and the other is of type -BF.

proof.

(BB 、BP),(BB 、BF)

can not exist at the same time

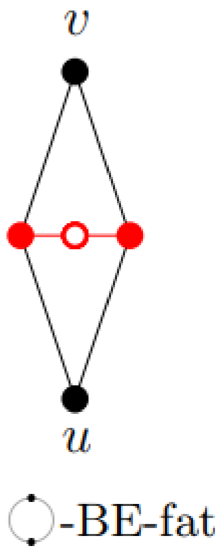


Permutation of the virtual edges

Let $e_1, \ldots, e_k$ be the virtual edges of sk(μ), there are h of them are not type BE-slim

- h=0 => μ is BE-fat
- h=1 => a BE, e_1, b BE , a ≤1, b=k-a-1
- h=2 => a BE, e_1, b BE ,e_2, c BE, a≤1, b ≤1, c=k-a-2

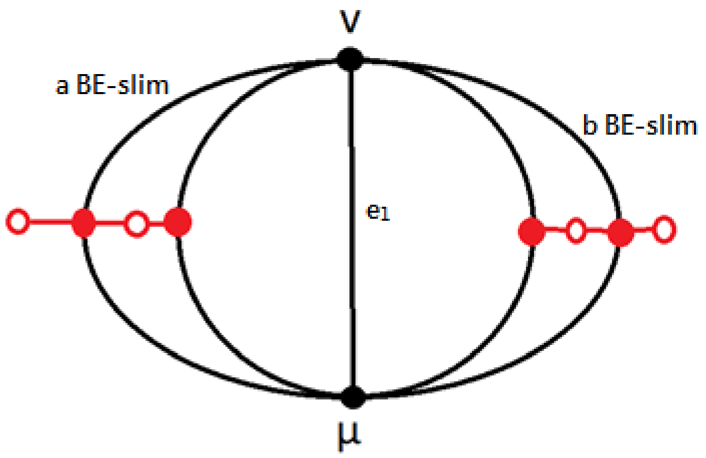
If h=0, μ is the type-BE-fat, the embedding of μ is unique, the solo red vertex can only be embedded in $L(E_\mu)$ 、 $r(E_\mu)$, so it can be handled in $O(sk(\mu))$.



If $h=1$,e_1 and μ is type BB or BP

By definition, $L(E_\mu)$ and $r(E_\mu)$ are red, we set $a \leq b$, then we have $a \leq 1$:

proof.



For a contradiction, let $a>1, b>1$

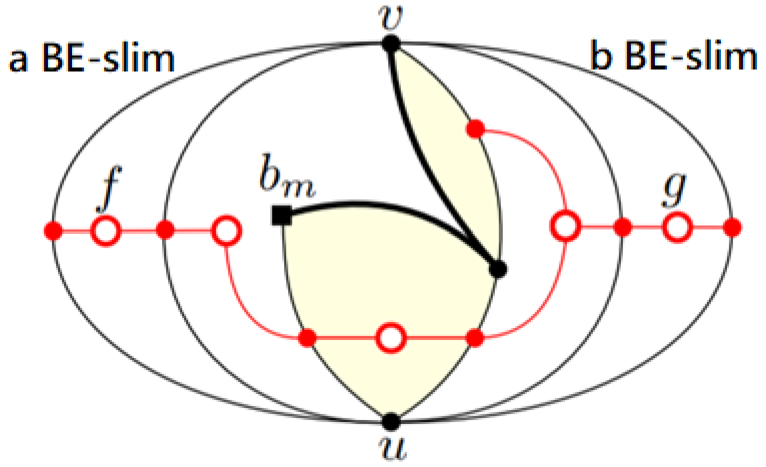
$\Rightarrow$ Exists two caterpillar has internal red face

$\Rightarrow$ Since $b_1 \notin H_\mu$(type is BB or BP) ,a contradiction.

In this case,we can check the node in $O(sk(\mu))$

If $h=1$,e_1 and μ is type BF

By definition, $L(E_\mu)$ and $r(E_\mu)$ are not red, we set $a \leq b$, then we have $a \leq 1$:



proof.

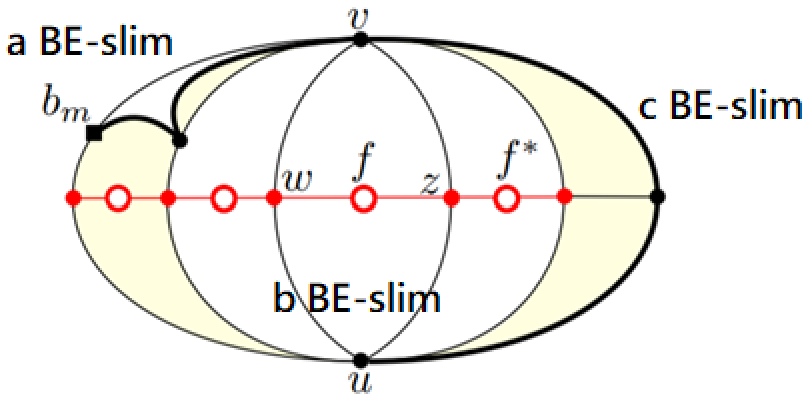
For a contradiction,

f、g is the end-vertex of $A(E_\mu)$.

b_m share no face with leaf incident to f or g, a contradiciton.

In this case ,we can check the node in $O(sk(\mu))$.

If h=2 , one of $\{e_1,e_2\}$ is BP, and the other one is BF, and thus μ is of type –BB, we have a ≤ 1, and b ≤ 1
a≤1 is similar to the case in which h = 1



proof. For a contradiction, let b>1
⇒ an outer face of E_μ as an end-vertex
⇒ let f* be the other end-vertex of B
⇒ No leaf that is adjacent to f* in $A(E_\mu)$ and that shares a face with b_m

Lemma :

Suppose that μ is a P-node such that v_μ is black.
We can check each type in $O(sk(\mu))$.

R-node

Definition. A face f of S_μ is original red if it get more than 2 points(The following can get one point):

(1)It is incident to a red vertex of sk(μ)

(2)It is incident to a virtual edges of type -BE or -BF

(3)It is an outer face, and the type of μ is -BP or -BB.

Lemma (Reddened conditions):

Let f be a non-original red face of sk(μ), we say f is Reddened when:

(1) Put a solo red vertex into it

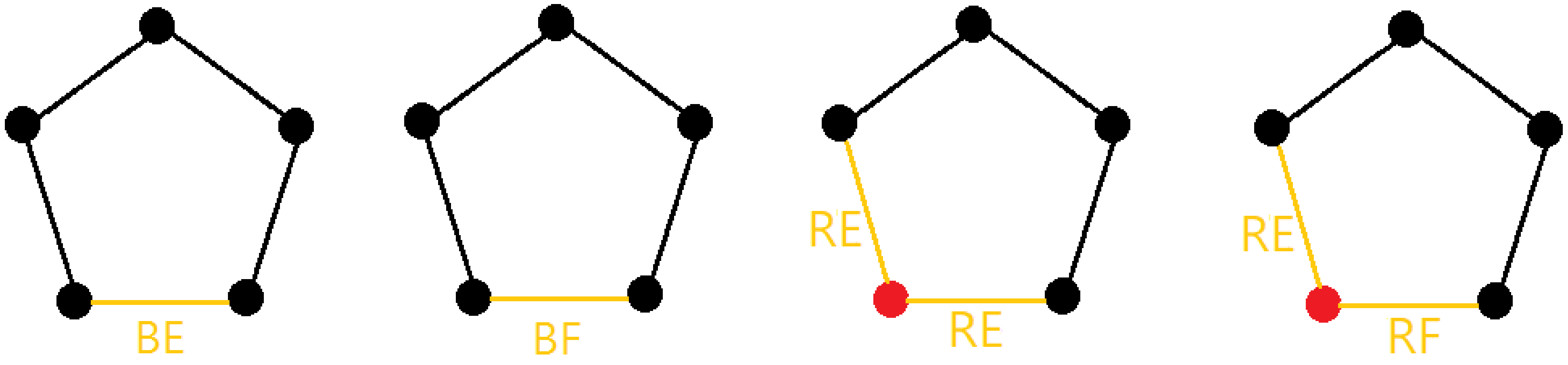
(2) One virtual edge incident to it is T_2 or is T_1 and the flip that the red outer face corresponding to f

proof.

There are at least one virtual edge incident to f is not BB or BP(otherwise form a black cycle), so f get at least one point, it will be Reddened if and only if get another point (one of the two condition holds).

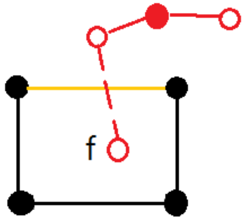
Observation:

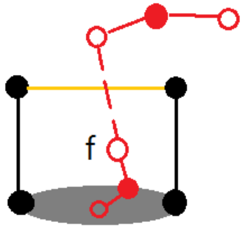
All non-original red face of sk(μ) are bounded by almost all BB and BP，then there are only few possibility of other virtual edges:



Lemma:

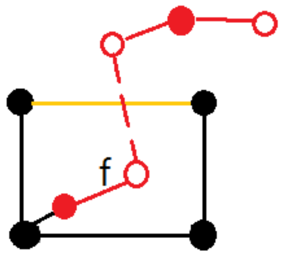
If f is an internal red face is Reddened in sk(μ), then the end-vertex of B is one of {f , Internal face of the virtual edge incident to f}



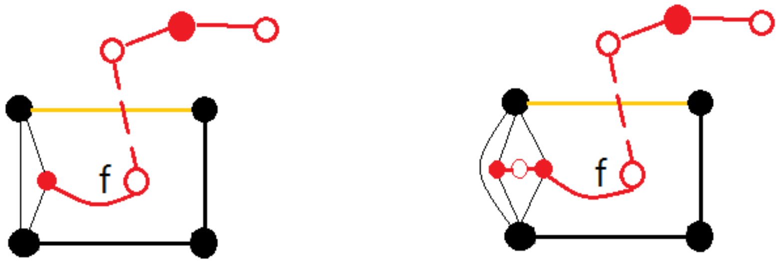


proof. By contradiction, we assume the both side of f will not be block block in f or the virtual edge incident to f. Then we have:

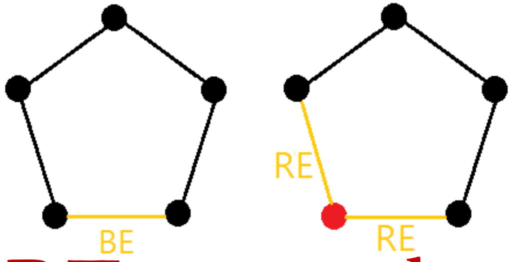
(1) The second point is not from the solo red vertex (otherwise f is the end-vertex)



(2) The second point is not from the BB or BP type virtual edge, otherwise the end-vertex will in this edge or f.



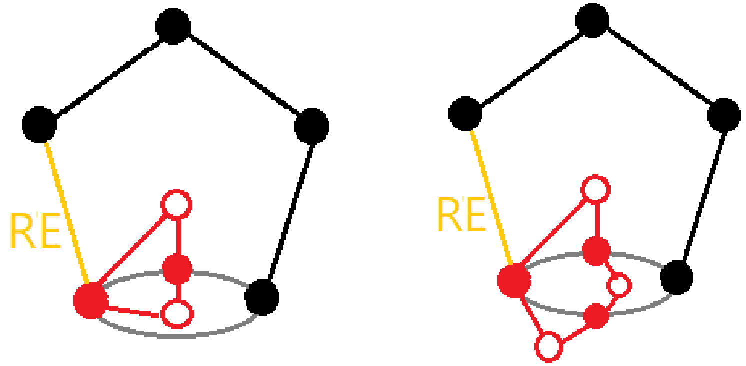
(3)We can not get 2 points from the single BE-type edge without solo red vertexes.



(4)We can not get 2 points from two RE-type edges without solo red vertexes.

Then we consider the RF -edge gets two points by itself.

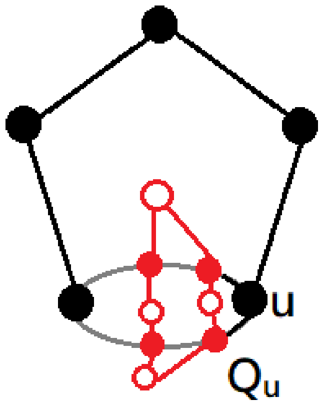
If the caterpillar does not block in f or the virtual edge incident to f, then it form a cycle .



(5)We can not get 2 points from a single RF-edge.

Finally, we consider the BF -edge gets two points by itself.

It forms a cycle, too.(Because the Q_u)



(6)We can not get 2 points from a single BF-edge.

We prove the Lemma.

Lemma:

If f is an internal red face is Reddened in sk(μ), then the end-vertex of B is one of

{f , Internal face of the virtual edge incident to f}

Lemma:

There exists at most **one** internal non-original red face of sk(μ) that is Reddened in E_{μ}.

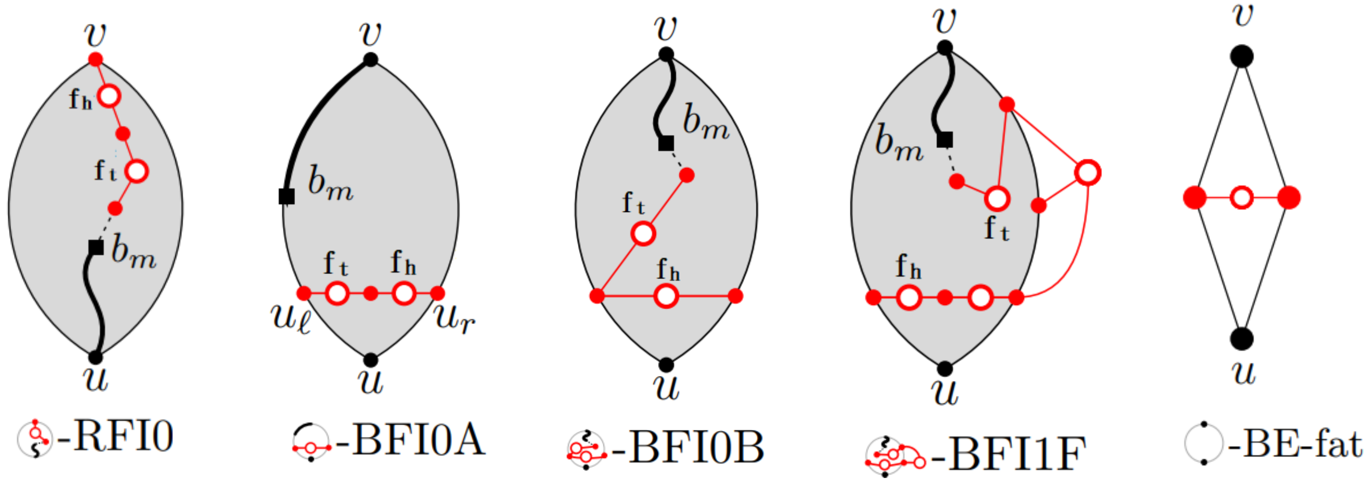
Furthermore, such a face is incident to a virtual edge e of sk(μ) such that b_m belongs to H_e.

proof.

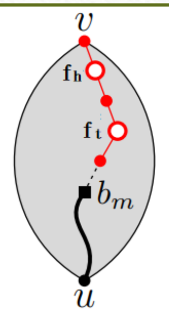
If there are two Reddened face in sk(μ),then the two end-vertex of the backbone of $A(E_{\mu})$ are both internal red face.

The only embedding types for which two end-vertices of the backbones of $A(E_\mu)$ are internal faces of E_μ are -RFI0, -BFI0, -BFI1F, and -BE fat.

BE fat is excluded since μ is an R-node.



If E_μ is -RFI0

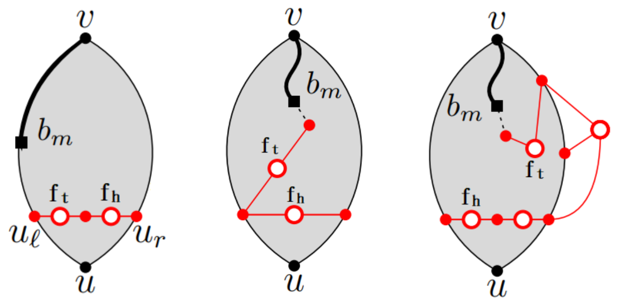


Since μ is an R-node, we have that u_μ has at least two neighbors in H_μ, exactly one of which is black.

=> there exists a red neighbor of u_μ incident to an outer face of E_μ , it coincide with v_μ, otherwise one of the outer faces of E_μ would be red.

=> The existence of the edge (u_μ, v_μ) implies that μ is a P-node, a contradiction.

If E_μ is -BF



-BFI0A -BFI0B -BFI1F

Since μ is type –BF, $A(E_\mu)$ is a single caterpillar.

Since the end-vertex of B_μ are both internal faces of E_μ, we have that at least one of them, say f_h , is incident to u_μ.

f_t and f are also incident to u_μ in E_μ and sk(μ), respectively.

u_μ does not have any black neighbor in H_μ, and thus the two virtual edges incident to u_μ and incident to f are of type either -RF, or -RE, or -BF, or -BE, a contradiction of f is non-original red.

Since μ is an R-node, its type is either -BP, or -BB, or -BF, or -RF.

(1) If μ is -BP, or -BB, or –RF, then $b_1 \notin H_\mu$.

(2) If μ is -BF, the end-vertex if in the b_m side.

A face is incident to a virtual edge e of sk(μ) such that b_m belongs to H_e.

Lemma 54.

There exists at most one outer non-original red face f of sk(μ) that is Reddened.

proof.

(1) μ is not RE or BE because it's R-node .

☯-BP1

(2) If μ is BB or BP because it's not BP1, so one of the outer faces is original red.

(3) If μ is RF ,then the edge between the pole (u , v) does not exists in sk(μ), otherwise μ is a P-node.

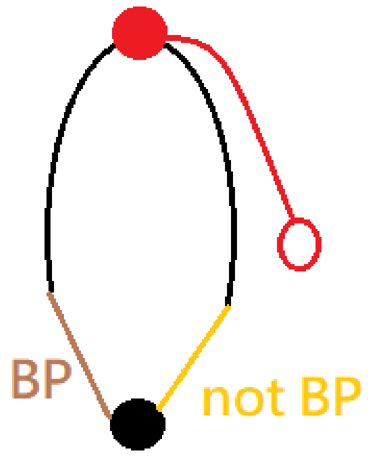
The degree of u is at least 2 in sk(μ) because μ is R-node.

Only one of them can be BP.

=> The other is BE、BF、RE、RF

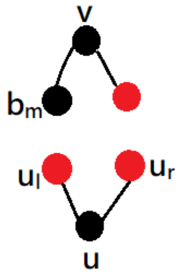
=> One of the outer face is incident to two red vertex or the red pole and one BE or BF in sk(μ).

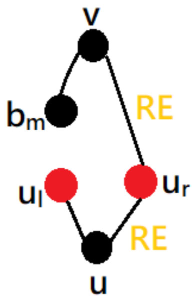
=> At least one of the outer face is original-red.



Suppose that the type of μ is -BF.
Since μ is an R-node, it has two distinct red neighbors u_l and u_r that are incident to $L(E_\mu)$ and $r(E_\mu)$.

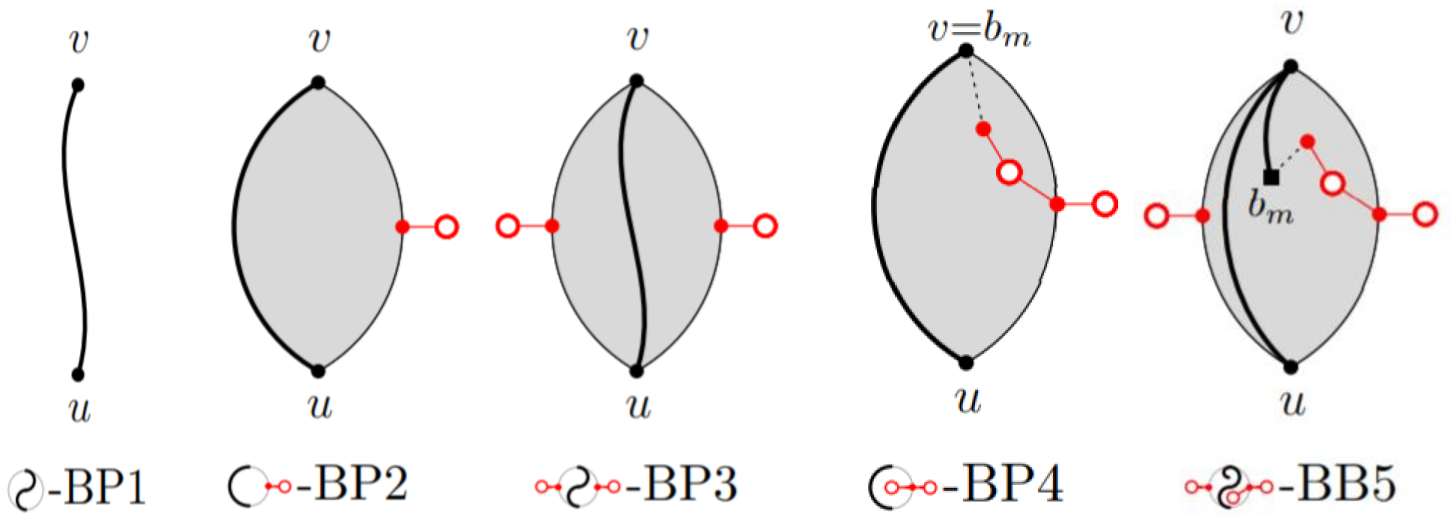
The pole v of μ has only one black neighbor in H_μ.
One edge incident to v is connect to a red vertex in sk(μ), say e.
(1) If e is not (v , u_r), then $L(E_\mu)$ is original red.
(2) If e is (v , u_r), then $L(E_\mu)$ can not be Reddened
without the solo red vertexes incident to the poles,
but they are not decided in μ.



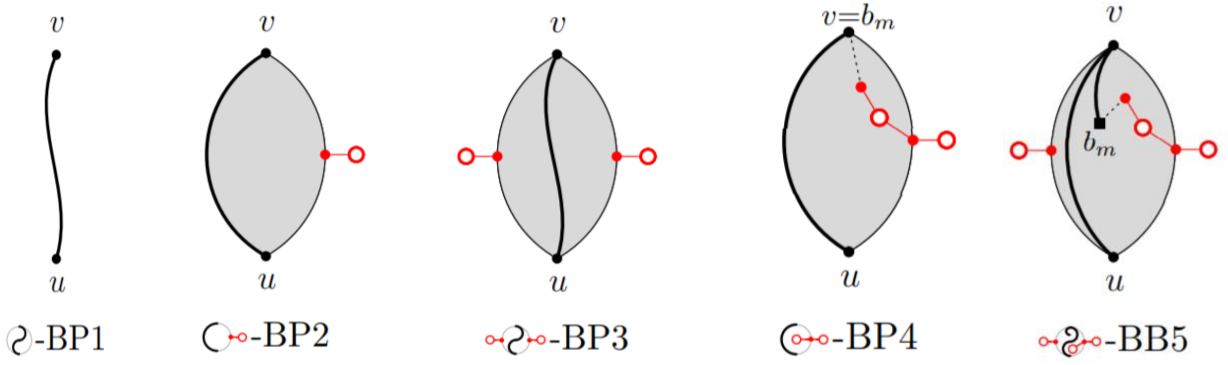


Case 1: b_m does not belong to H_μ.

1. the type of μ is -BP;
2. the type of E_μ is either -BP2 or -BP3;
3. sk(μ) contains no internal original red face;
4. the virtual edges of sk(μ) are of type -RE, -BE slim, or –BP(otherwise there is an internal red face)



5. for each type -BP virtual edge e of sk(μ), the type of E_e is either -BP1 or -BP2;
6. for each virtual edge e such that the type of E_e is -BE slim or -BP2, we have that e is incident to an outer face Also, if the type of E_e is -BP2, then E_e is the flip that put the red face in the outer face.
7. each solo red vertex incident to a vertex of sk(μ) is incident to an outer face of E_μ.



Check in $O(|sk(\mu)|)$ time whether one of the following conditions is fulfilled.

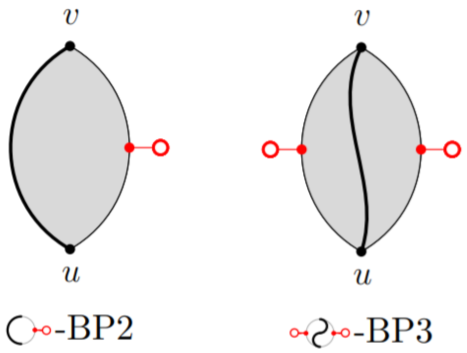
1. $sk(\mu)$ contains an internal original-red face
2. there exists at least one virtual edge e of $sk(\mu)$ that is not of type -RE, -BE slim, or -BP1
3. $sk(\mu)$ contains a type -BP virtual edge whose pertinent graph can not admit the type -BP1 or -BP2
4. $sk(\mu)$ contains a virtual edge not incident to an outer face of S_μ whose pertinent graph only admits type -BP2
5. there exists a solo red vertex that is incident to an internal vertex of $sk(\mu)$

Handle the case:

(1) After finishing the above check , we check the correctness of this node.

(2) We can know the type of this node by check the outer face only. Because the possibility of this node is only BP2、BP3.

(3) All action can be done in sk(μ).



Case 2: b_m is a vertex of $sk(\mu)$

1. the type of μ is -BP if $b_m = v_\mu$ is a pole of μ, and is either -RF, or -BF, or -BB, otherwise;
2. the virtual edges of sk(μ) are of type -RE, -BE slim, -BE fat, or -BP;

   proof.

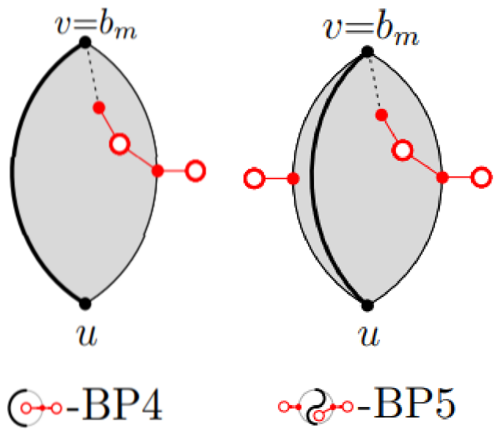
   They can not be the type of BB、BF、RF because b_m belongs to sk(μ)

3. There exists at most one type -BP virtual edge e of sk(μ) such that the type of E_e is -BP4 or -BP5;

Further, there exists a single virtual edge e_m of type -BP incident to b_m;

Finally, if a virtual edge e exists such that the type of E_e is -BP4 or -BP5, then $e = e_m$.

proof. There is only one BP edge incident to b_m, and BP4、BP5 is incident to b_m



In this case, we do not need to do a preliminary test. Because the above observation is from its structure, not from the embedding, so they are guaranteed to hold.

We divided the remaining work into three parts:

(i) Handle the Reddend Case
(ii) Handle all BP virtual edges
(iii) Handle other virtual edges

Lemma:

There exists at most one internal non-original red face of sk(μ) that may be reddened and such a face has to be incident to b_m. Such a face must be one of the two faces incident to e_m.

proof.

Consider any face f incident to b_m and not bounded by e_m;

The two virtual edges that lie along the boundary of f and that are incident to b_m are of type -RE or –BE.

=>f is original red.

(1) At most one of the outer faces of $sk(\mu)$, say $r(E_{\mu})$, is non-original red and is allowed to be reddened .

(2) Let the two face incident e_m f_m 、g_m

(3) We consider the six pairs (x, y), with $x \in \{\emptyset, f_m, g_m\}$ and $y \in \{\emptyset, r(S_{\mu})\}$.

We only show one of the most difficult case in which

(i) $x, y \neq \emptyset$

(ii) $x \neq y$, and

(iii) both x and y are non-original red

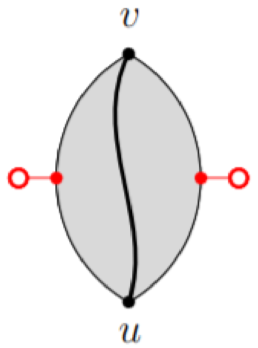
Phase 1 :Satisfying the pair (x,y)

Compute the set S_x (resp. S_y) that contains all the type -BP virtual edges incident to x (resp. to y) that are not of type -BP1, and all the black vertices of sk(μ) incident to x (resp. to y) that are incident to a solo red vertex.

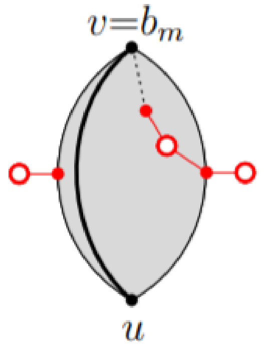
If S_x (resp. S_y) is empty, then we abort the construction for the pair (x, y)

If $S_x = S_y$ and $|S_x| = |S_y| = 1$

If the only element in $S_x = S_y$ is a black vertex or a virtual edge that -BP3 or -BP5 is not exposed, then we abort the construction for the pair (x, y).



-BP3



-BP5

Otherwise, S_x contains exactly one virtual edge e_i , which -BP3 or -BP5 is exposed. If $e_i \neq e_m$, then -BP3 is exposed in e_i.

We set $type_i$ = BP3 and we arbitrarily set the flip.

If $e_i = e_m$, then we make up to four independent choices by setting $type_i$ -BP3 or -BP5, and by setting the flip arbitrarily.

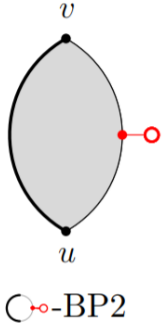
In all the other cases (that is, if $|S_x| \geq 1$, $|S_y| \geq 1$, and it holds that $|S_x| \geq 2$, or $|S_y| \geq 2$, or $S_x \neq S_y$), we can find two distinct elements, one in S_x and one in S_y.

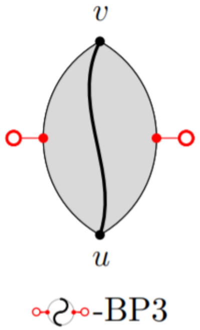
We use these elements to ensure that both x and y are reddened.

If this element is a vertex w, then we select x to be the designated face f_w of w.

Otherwise, this element is a virtual edge e_i , and we proceed as follows.

If this element is a virtual edge e_i , Suppose that $e_i \neq e_m$.
If -BP2 is exposed in e_i , then we set $type_i$ = BP2 and
We choose the flip that put the red face side into the
Face which we want to let it reddened.
Otherwise, -BP3 is exposed in e_i , if the face z of sk(μ)
incident to e_i different from x is either original red or
coincides with y, then we set $type_i$ = BP3 and we arbitrarily
set the flip.





Suppose now that $e_i = e_m$. Let $z \in \{f_m, g_m\}$ be the face of S_μ incident to e_i different from x.

If z is original red or z = y, then we make up to six independent choices by selecting $type_i$ as each of the exposed types in e_i :

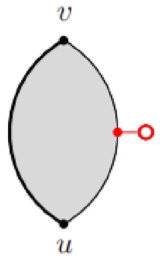
If $t_i \in \{BP2, BP4\} \subseteq T_1$, then we select only one flip for x_i .

If $t_i \in \{BP3, BP5\} \subseteq T_2$, then we select both possible flips for x_i).

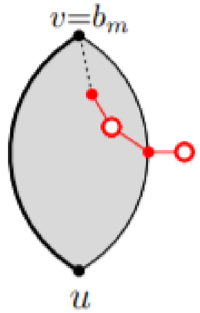
If z is non-original red and z≠y.

If the set of exposed types in e_i contains neither -BP2 nor -BP4, we abort the construction of the embedding configurations for the pair (x, y).

Otherwise, we make up to two independent choices by setting $type_i$ as each of the exposed types for e_i among BP2 and BP4, and by setting the flip that put the red face side in x.



-BP2



-BP4

Phase 2 :Handling the remaining type -BP virtual edges.

Actually ,we do this phase first to reduce the possibility of the pair(x,y)

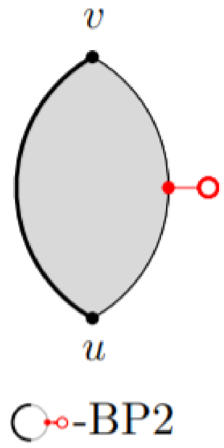
Let e_i be any virtual edge of sk(μ) of type -BP that is not of type -BP1.

Let f_i and g_i be the two faces of S_μ that are incident to e_i.

If both f_i and g_i are non-original red faces different from x and y, then we abort the construction of the embedding configurations for the pair (x, y).

If exactly one of f_i and g_i , is a non-original red face different from x and y, then we distinguish the case in which $e_i \neq e_m$ from the one in which $e_i = e_m$.

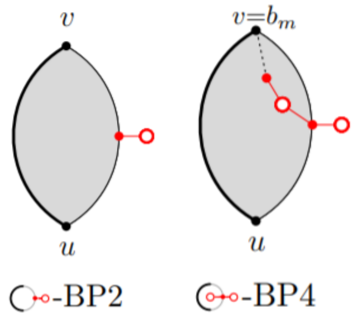
If $e_i \neq e_m$, we check whether the type -BP2 is exposed in e_i .
If not, we abort the construction of the embedding configurations for the pair (x, y).
Otherwise, we set $type_i$ = BP2 and we set the flip that put the red face into the original-red face.



If $e_i = e_m$, we check whether the type -BP2 or -BP4 is exposed in e_i .

If not, we abort the construction of the embedding configurations for the pair (x, y).

Otherwise, we make up to two independent choices by setting t_i as each of the admissible types for e_i among -BP2 and -BP4, and put the red face side in the face which is x or y or original red.



Handling the remaining virtual edges.

Handle the case:

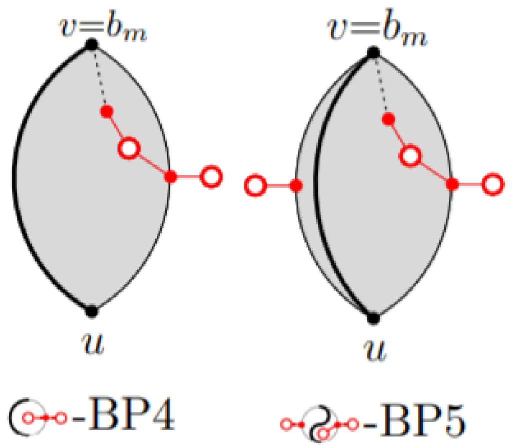
Finally, for each virtual edge e_i of sk(μ) that is of type -RE, -BE slim, -BE fat, or -BP1,we can embed them in arbitrary flip.

The number of pair (x,y) is O(1) size.

Each pair can be check in O(sk(μ)) time.

Case 3: b_m belongs to H_μ but not to $sk(\mu)$

1. the type of μ is either -RF, or -BF, or -BB;
2. the type of e_m is either -RF, or -BF, or -BB;
3. the virtual edges of $sk(\mu)$ different from e_m are of type -RE, -BE slim, -BE fat, or -BP;
4. there exists no type -BP virtual edge e of $sk(\mu)$ such that the type of E_e is -BP4 or -BP5.



The algorithm to compute a constant-size set of embedding configurations for μ is similar to the one for the case in which b_m is a vertex of $sk(\mu)$ (Case 2).

(i) Satisify (x,y)
(ii) Handle all BP virtual edges
(iii) Handle other virtual edges

Handle the case:

The pair (x,y) is at most 6 combination

$x=\{f_m, g_m, \emptyset\}$,f_m 、 g_m are the both side of e_m

$y=\{r(E_\mu),\emptyset\}$, $r(E_\mu)$ is the non-original red outer face if it exists

All actions are similar to Case2,we can done all of them in $O(sk(\mu))$.

The topic can be discussed

I rank the importance of the topics:

(1) The computational complexity of Quasi-planar

(2) A algorithm for Quasi-planar

(3) The caterpillar characteristic in different graph

(4) The Thevenin's theorem in the Brute-force

Conclusion

(1) Prove a problem which is NP-complete and linear equivalence with other two NP-complete problem, so this problem is the core of this area.

(2) Offers an linear algorithm to this problem given the order of the black vertex.