Track Layout is Hard

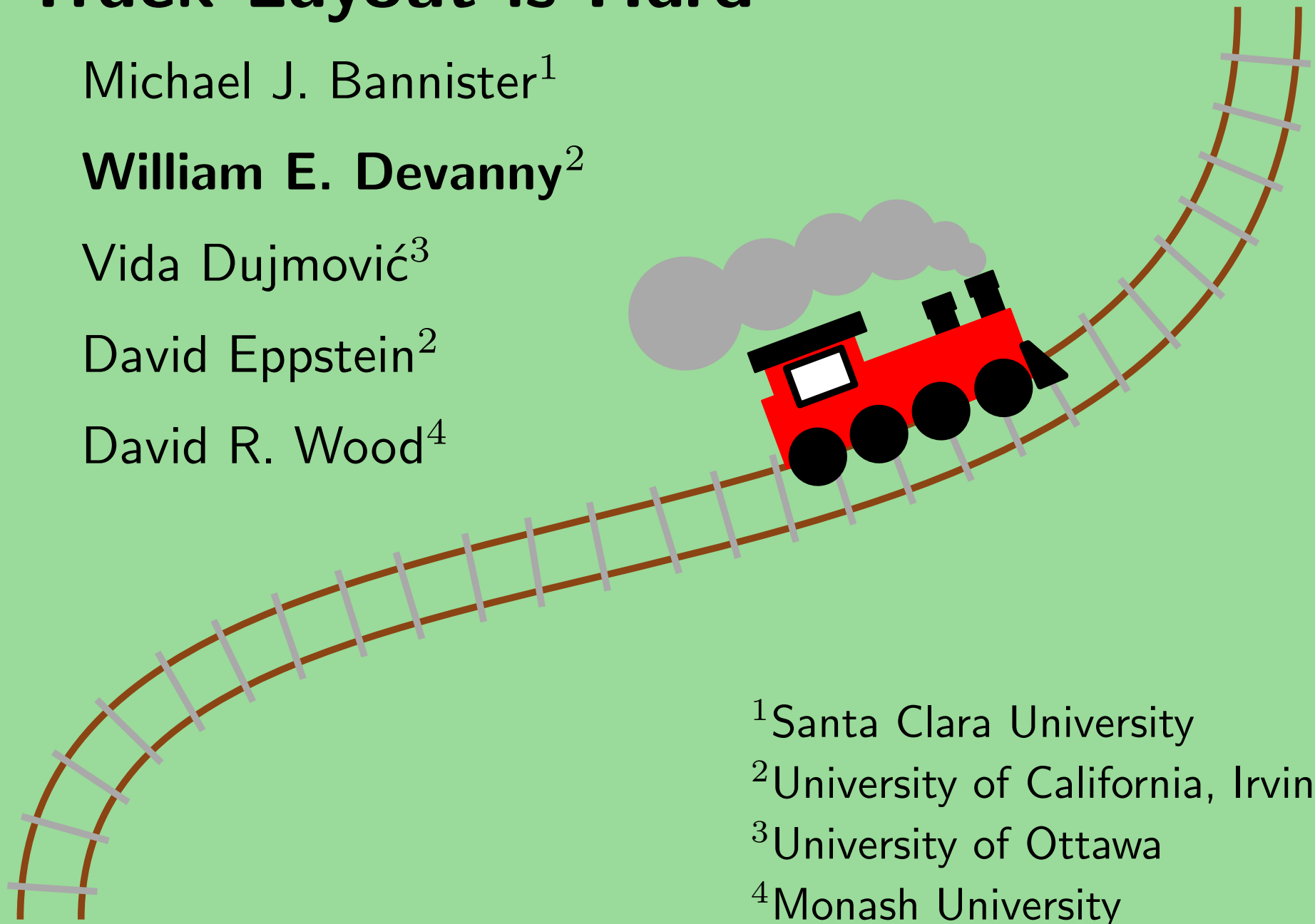
Michael J. Bannister[1]

William E. Devanny[2]

Vida Dujmović[3]

David Eppstein[2]

David R. Wood[4]



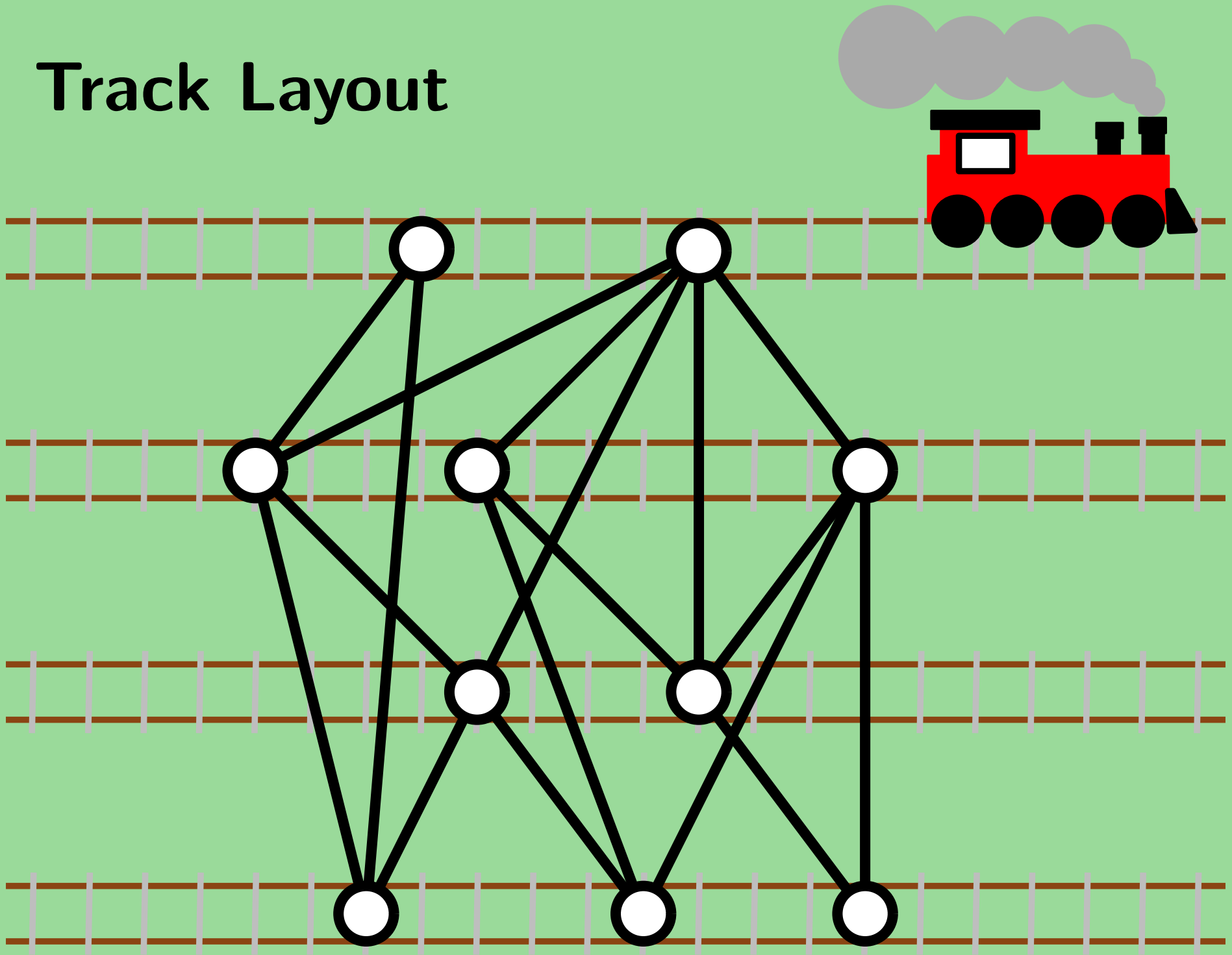
[1]Santa Clara University

[2]University of California, Irvine

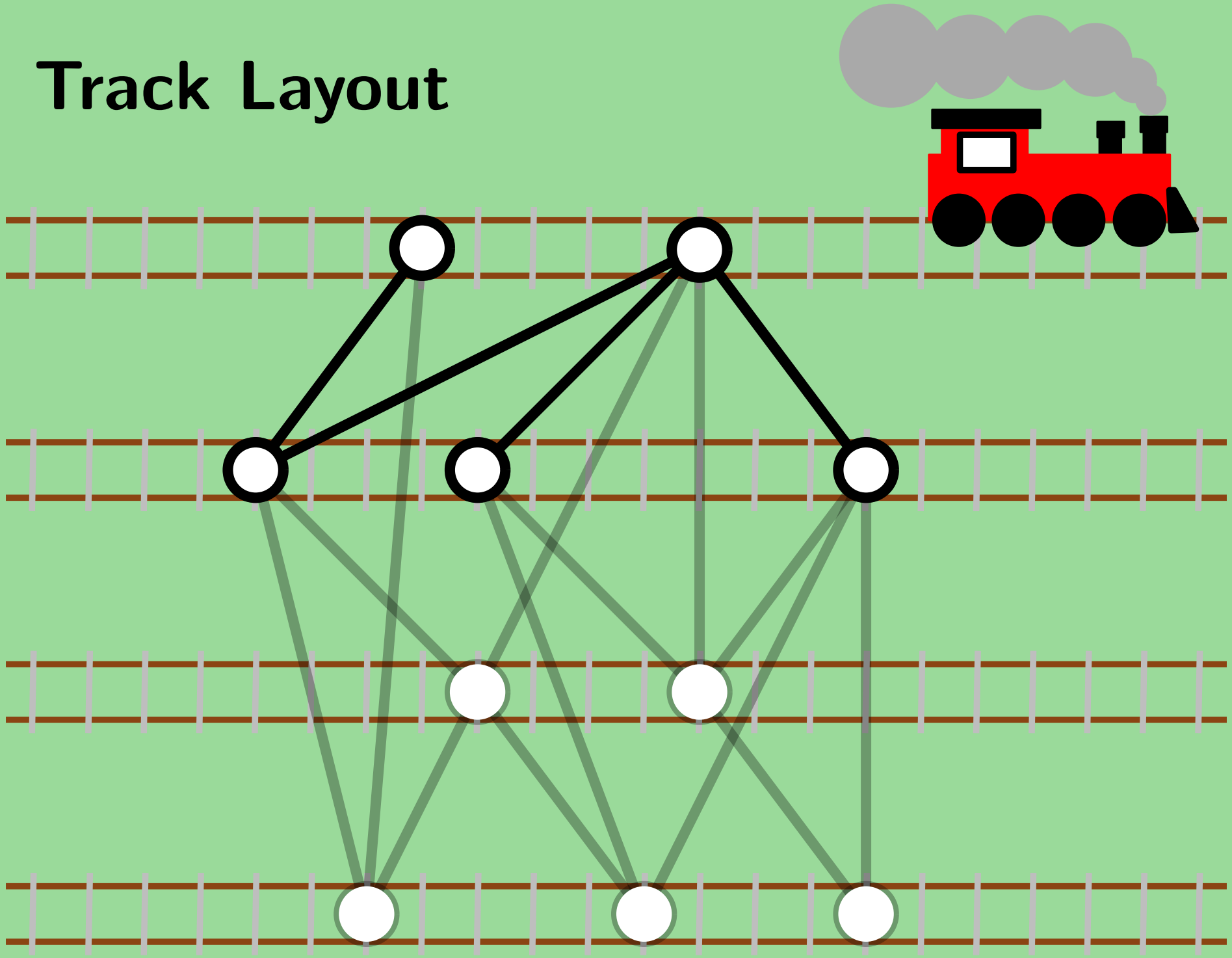
[3]University of Ottawa

[4]Monash University

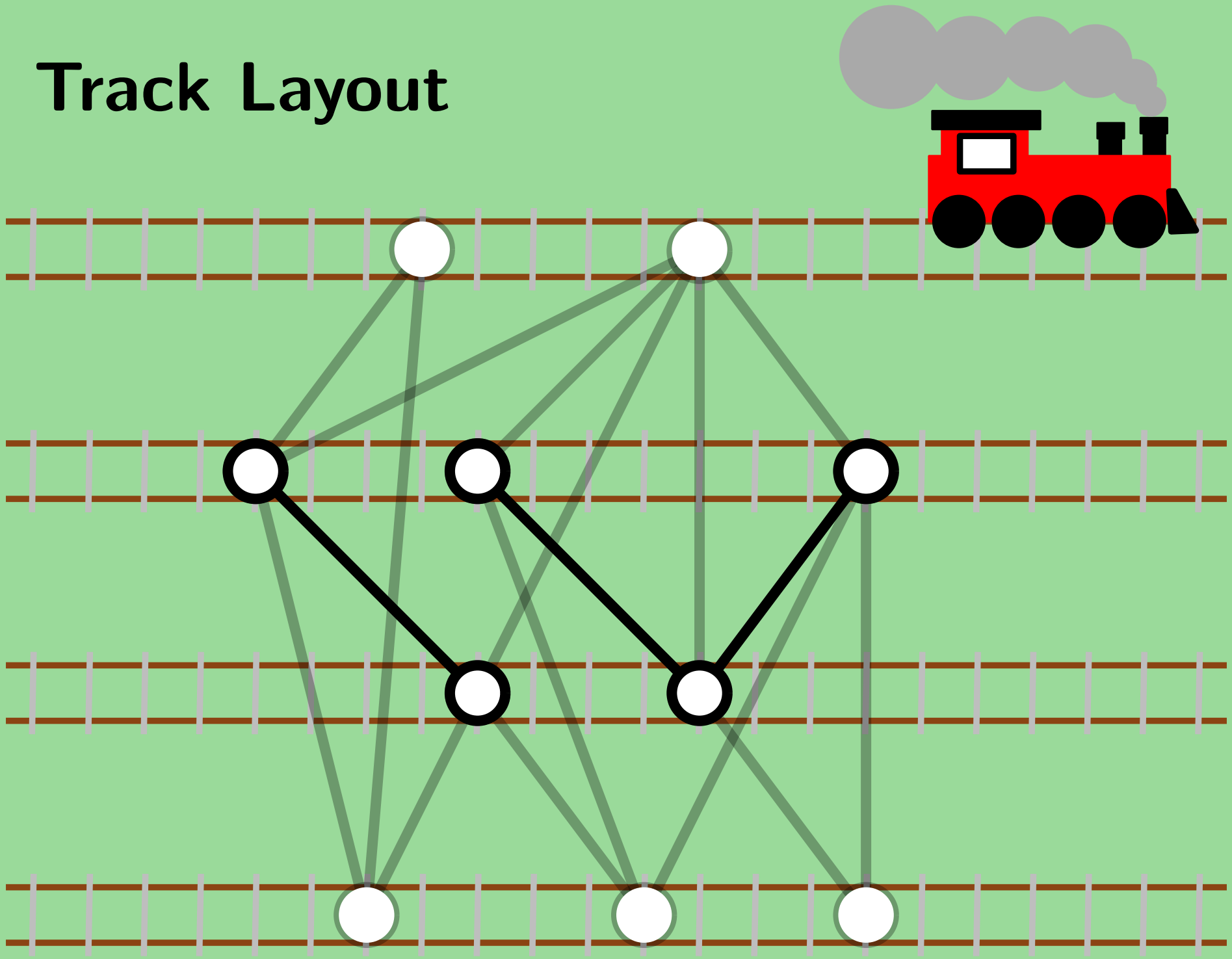
Track Layout



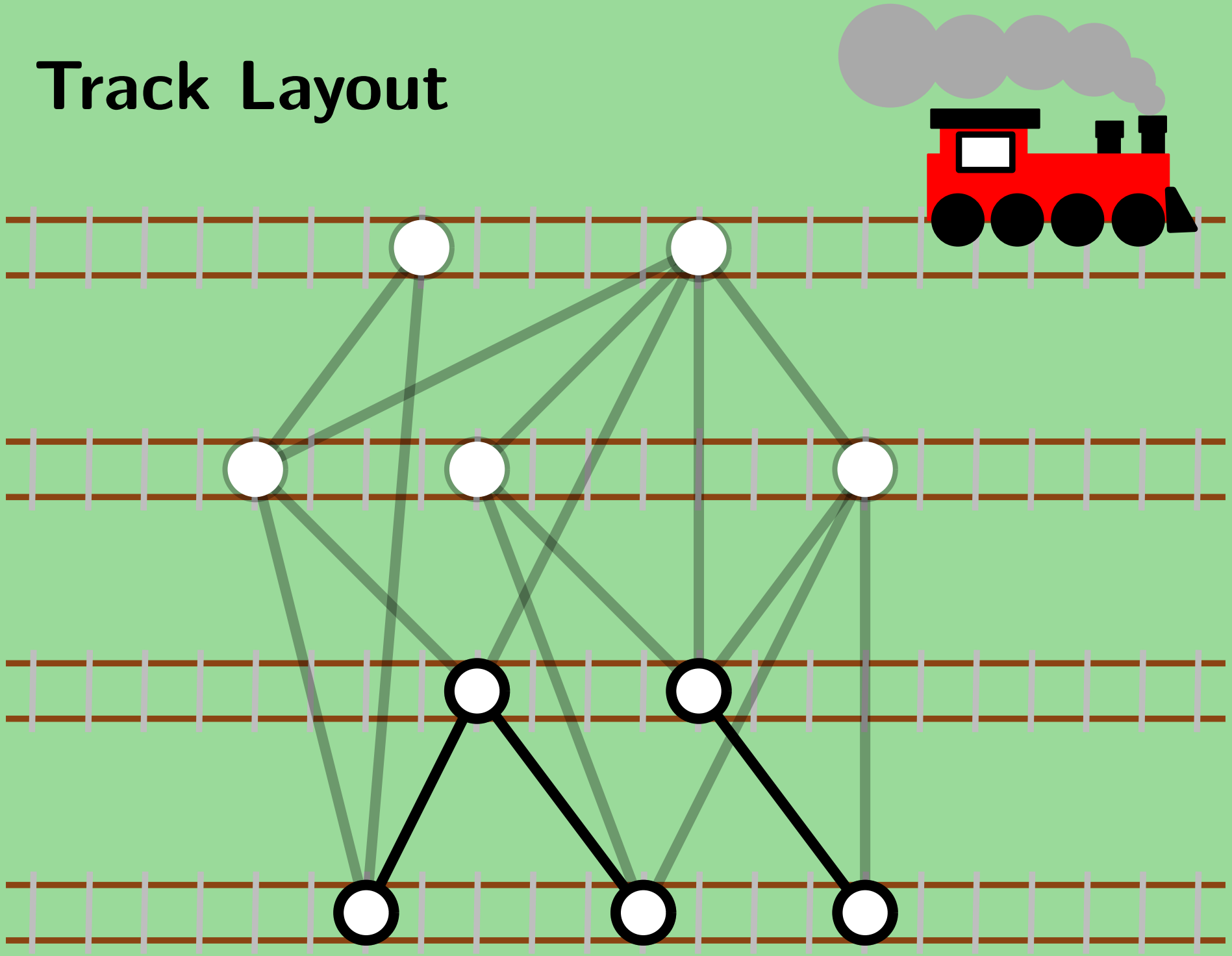
Track Layout



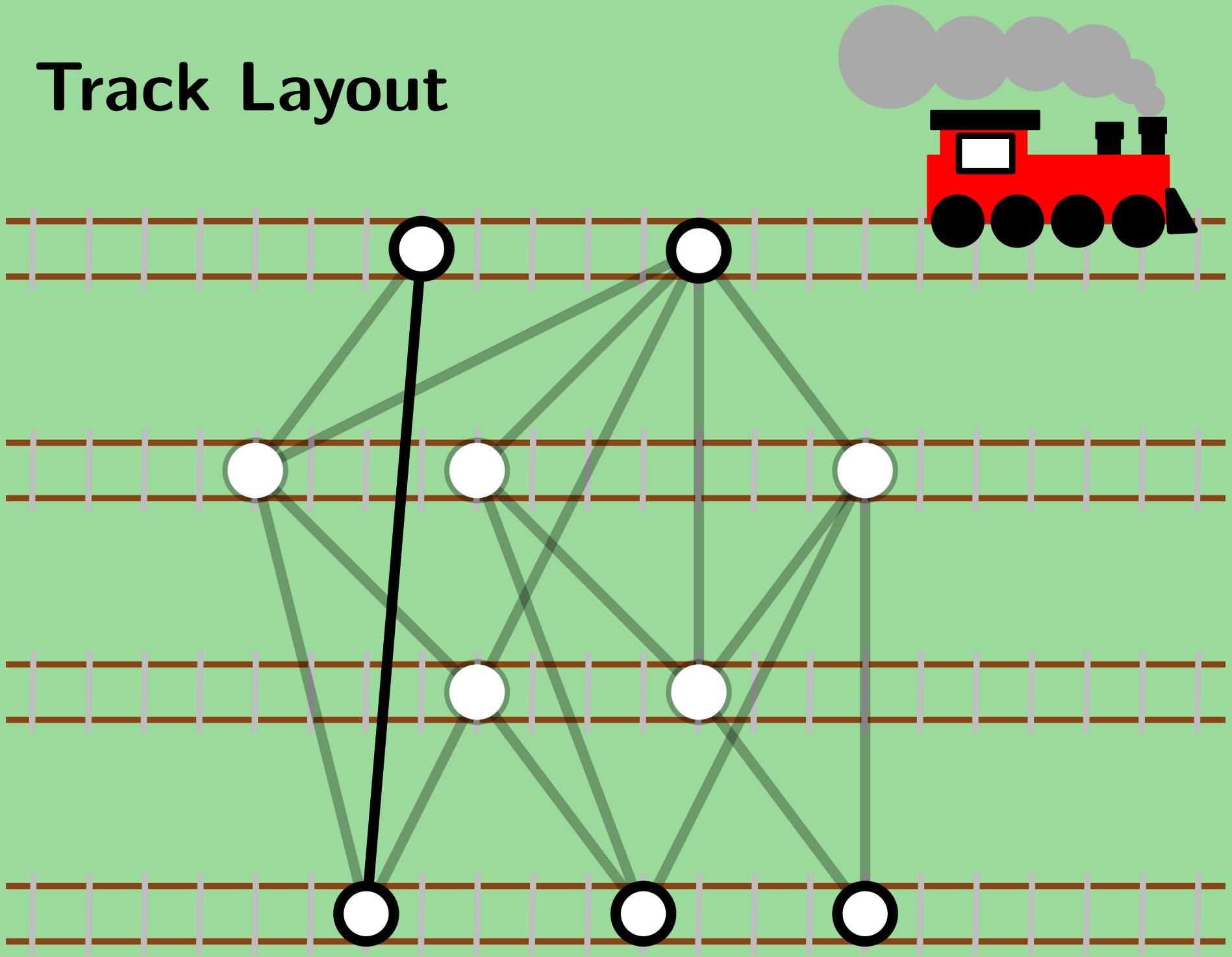
Track Layout



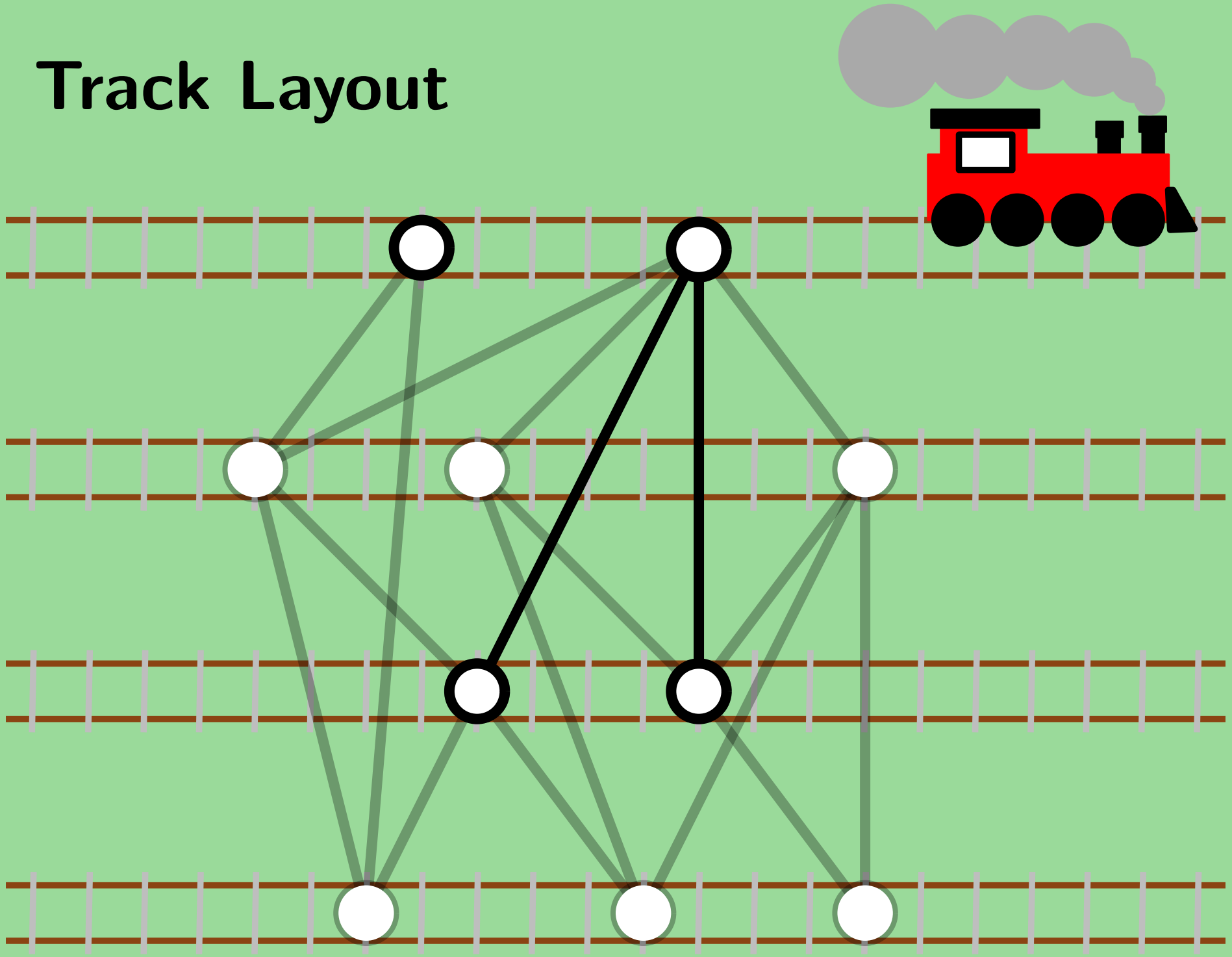
Track Layout



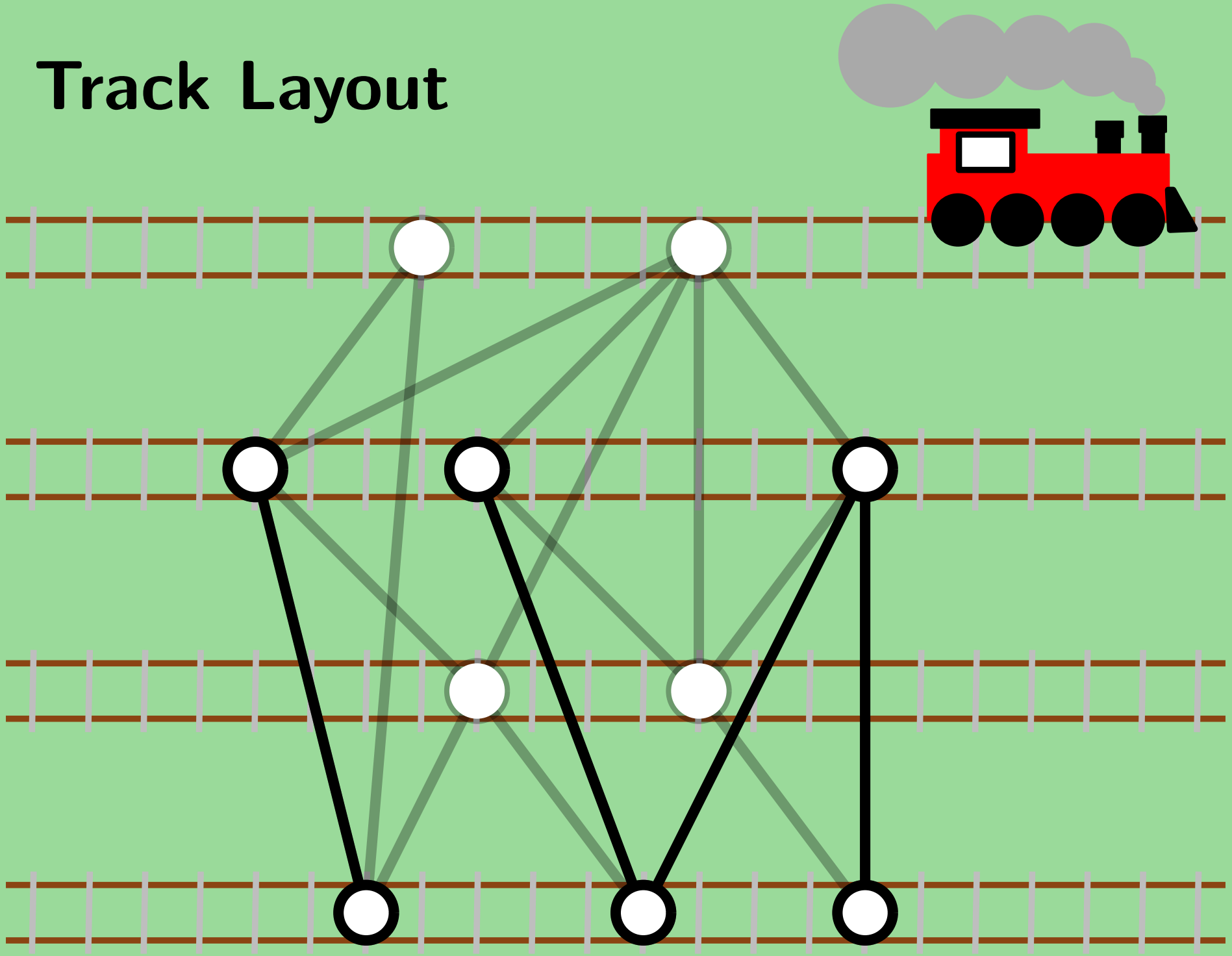
Track Layout



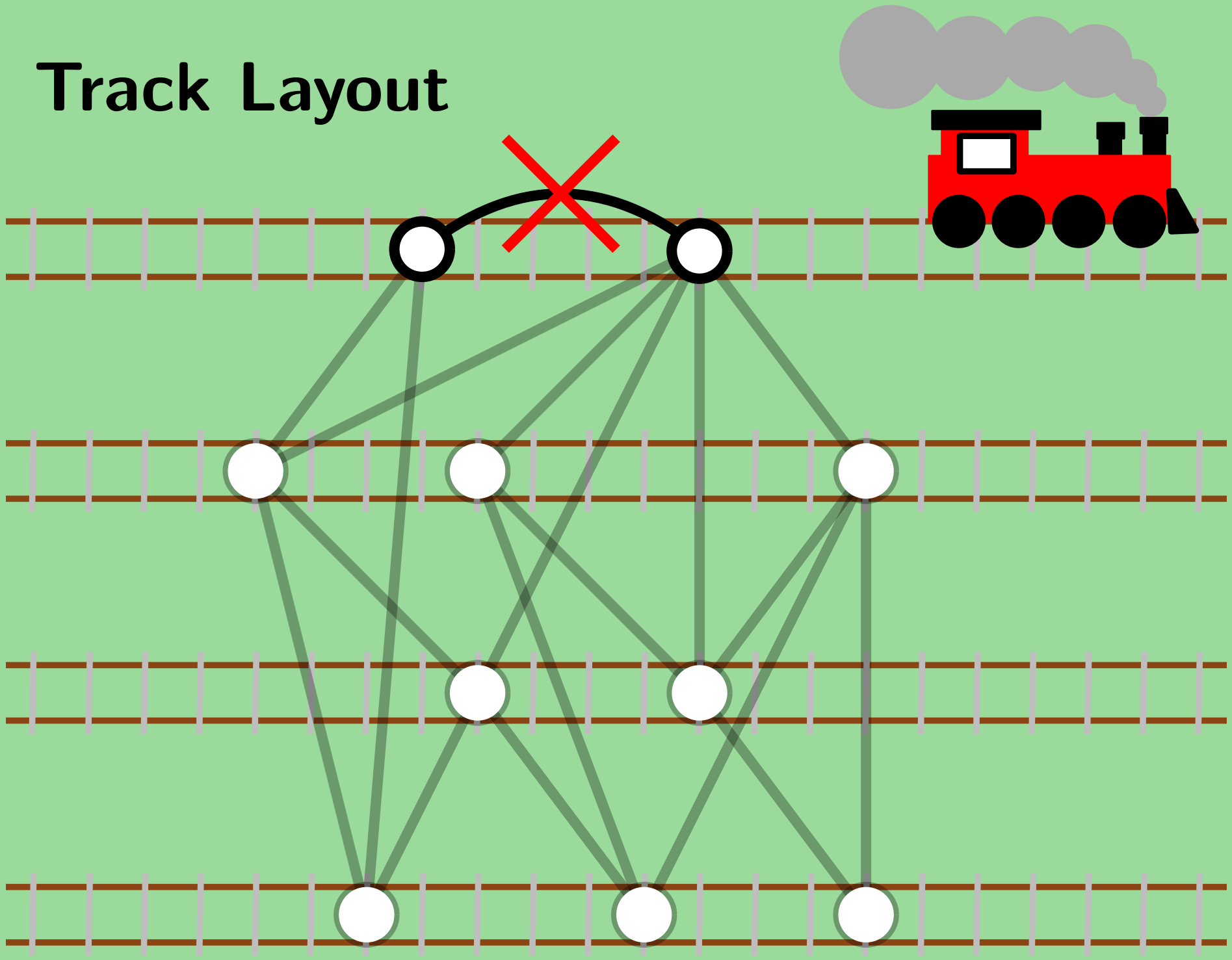
Track Layout



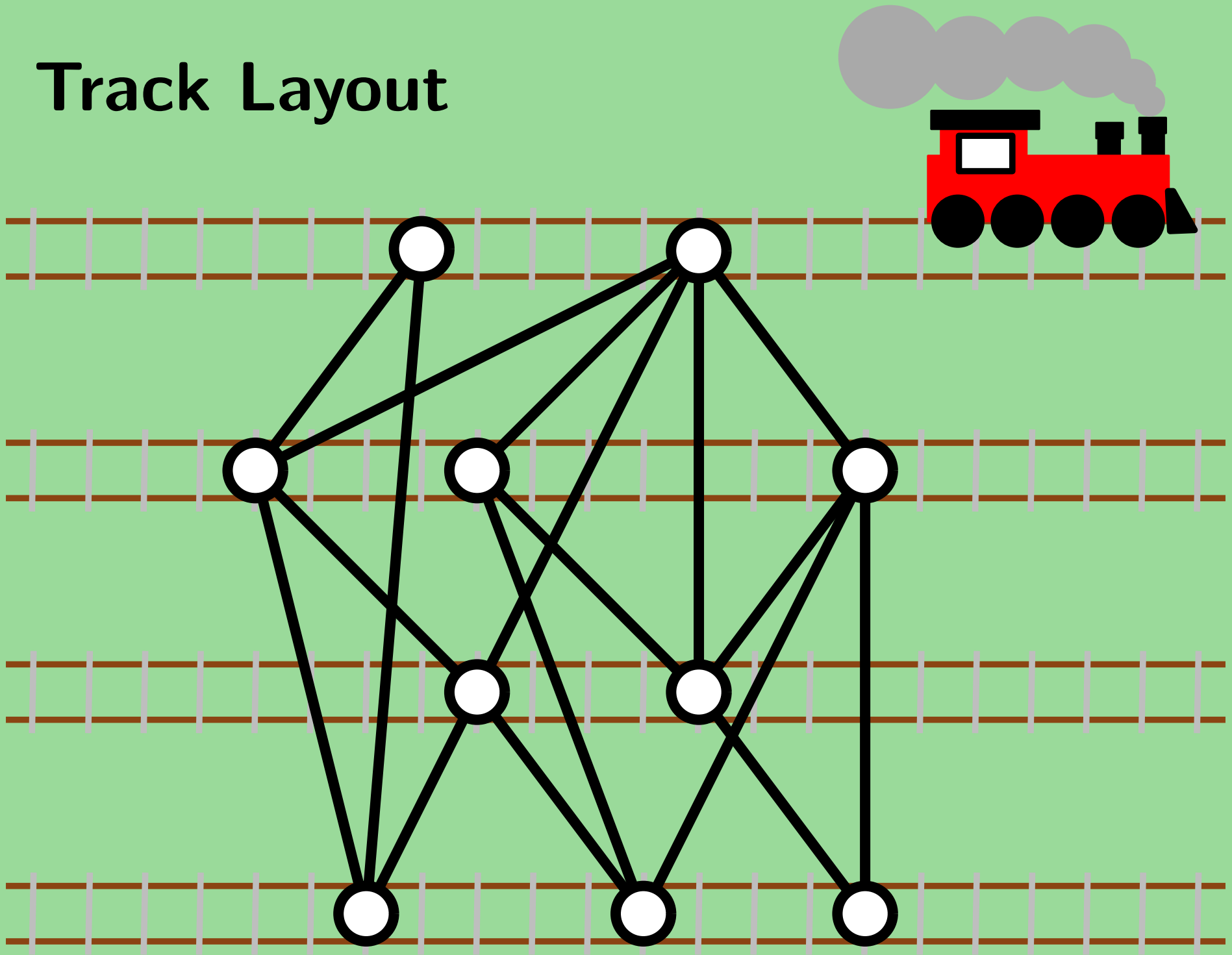
Track Layout



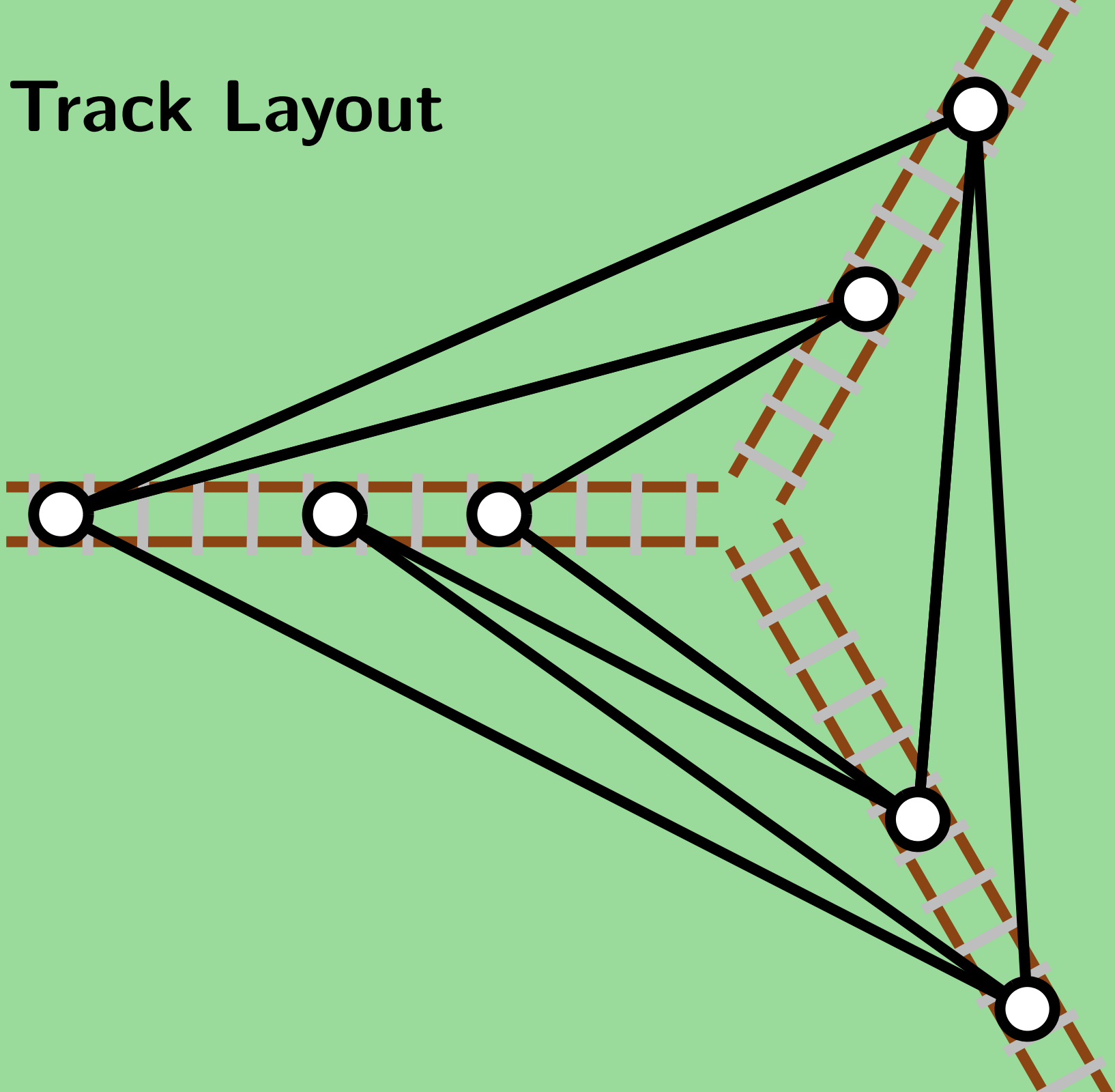
Track Layout



Track Layout



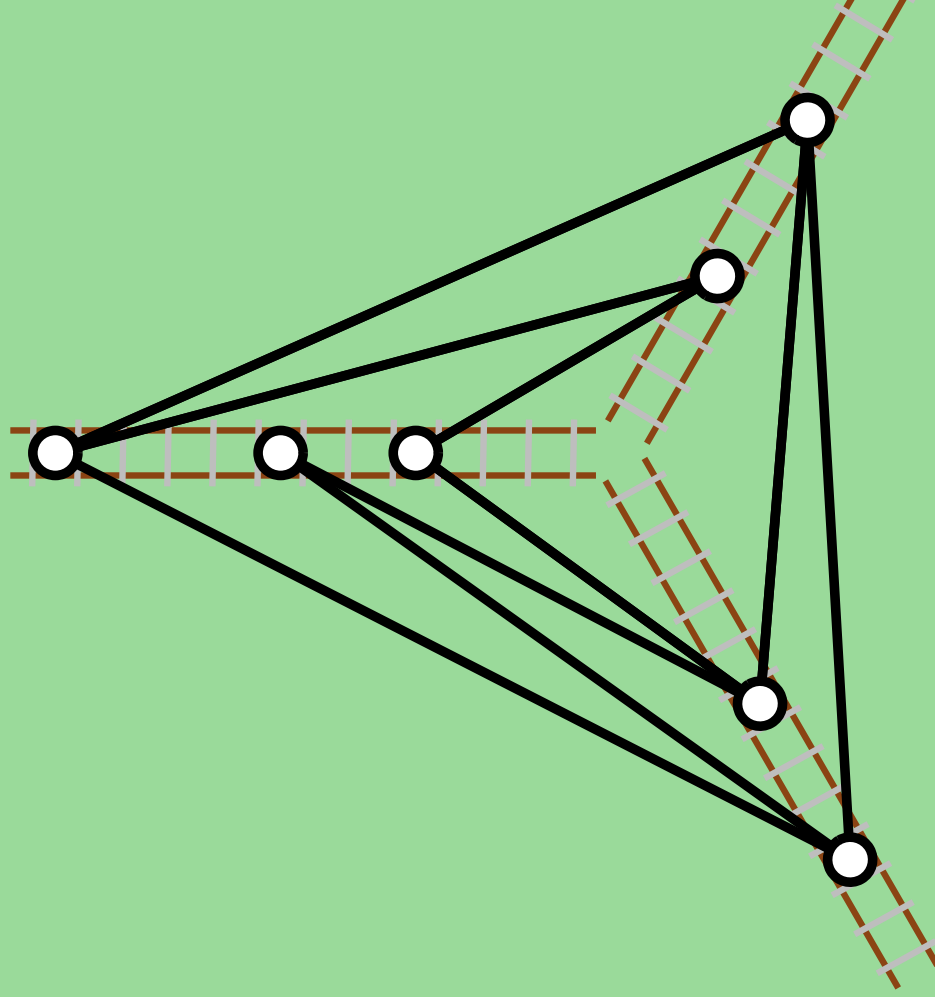
Track Layout



Track Layout

Track layouts find applications in minimizing the volume of 3D drawings

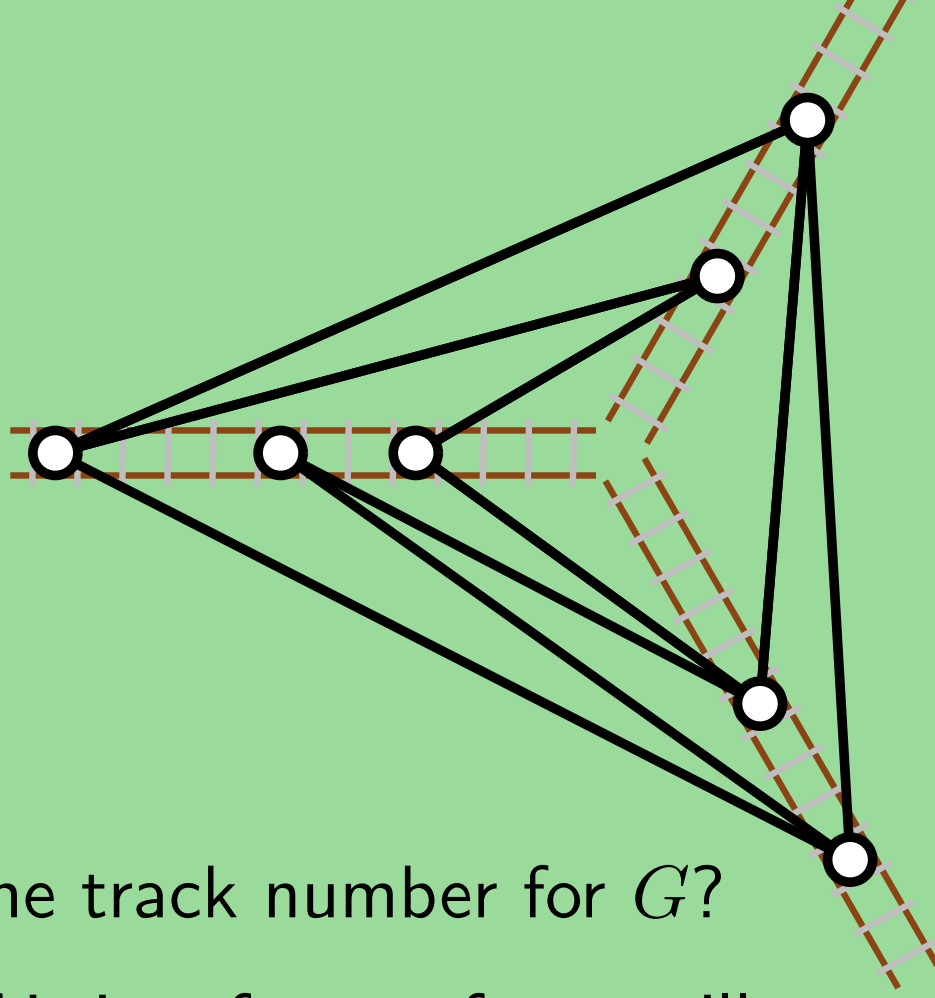
[Di Giacomo, 2004; Di Giacomo and Meijer, 2004; Dujmović and Wood, 2004; Felsner et al, 2003; Hasunuma, 2004]



Track Layout

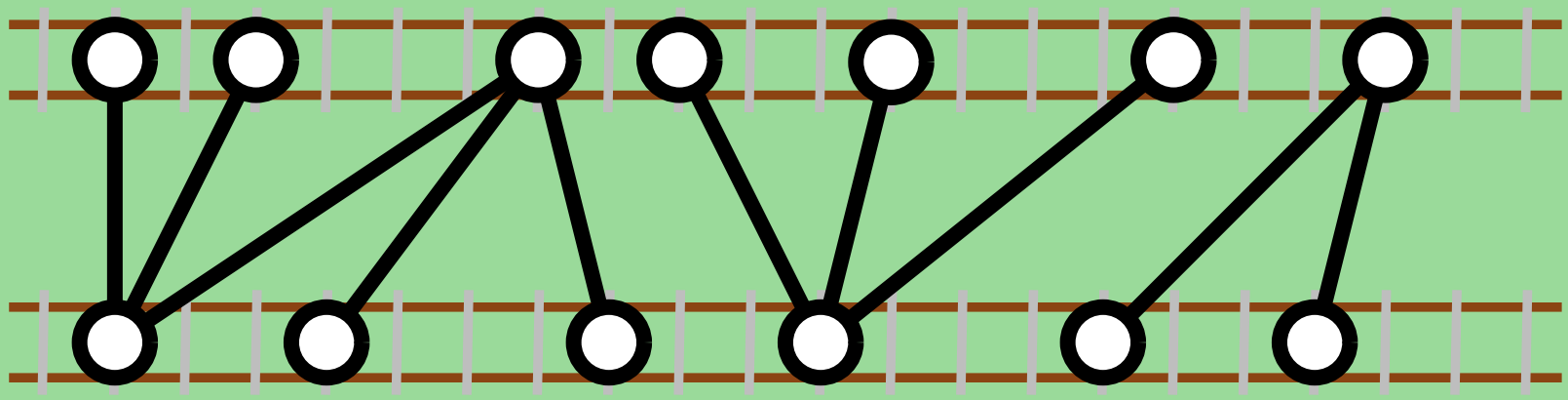
Track layouts find applications in minimizing the volume of 3D drawings

[Di Giacomo, 2004; Di Giacomo and Meijer, 2004; Dujmović and Wood, 2004; Felsner et al, 2003; Hasunuma, 2004]



Given a graph G can we compute the track number for G?

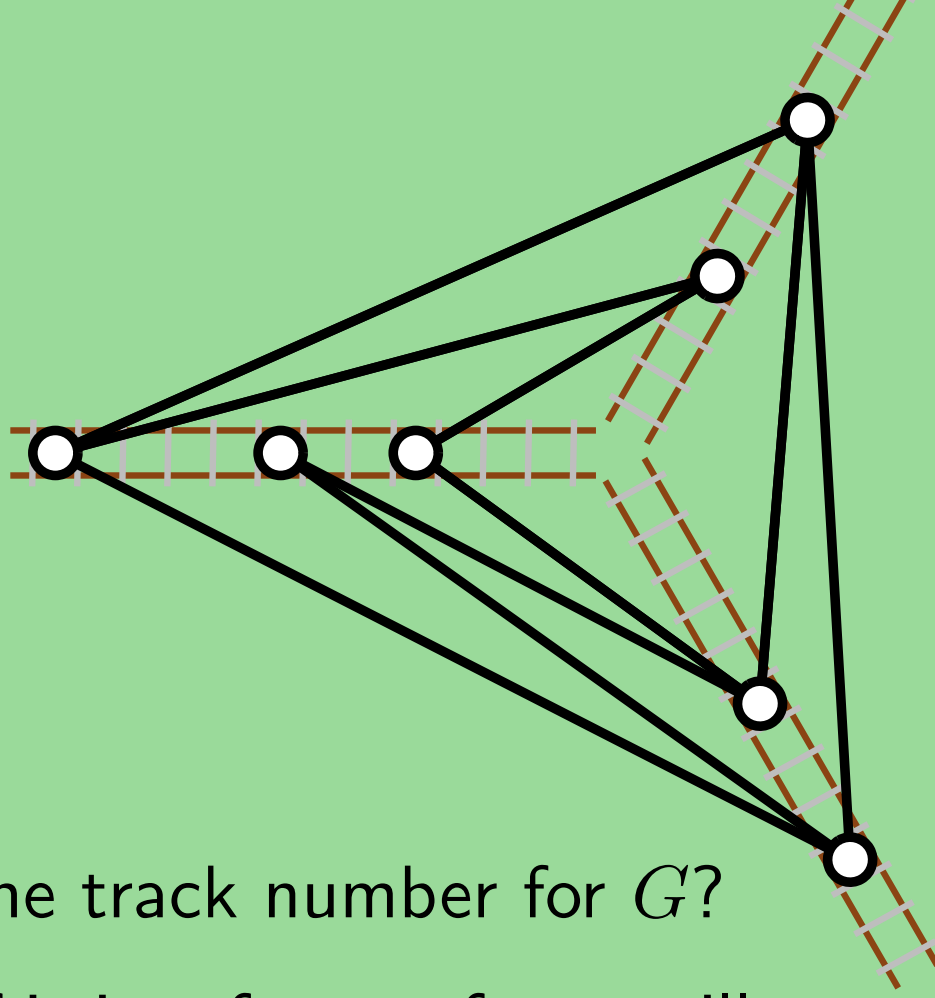
A graph is 2-track if and only if it is a forest of caterpillars



Track Layout

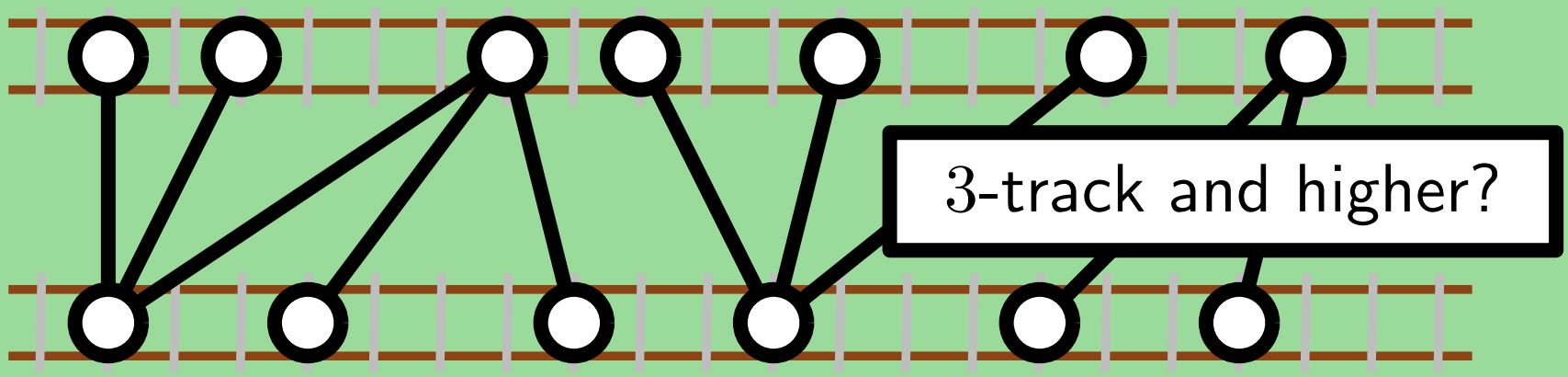
Track layouts find applications in minimizing the volume of 3D drawings

[Di Giacomo, 2004; Di Giacomo and Meijer, 2004; Dujmović and Wood, 2004; Felsner et al, 2003; Hasunuma, 2004]



Given a graph G can we compute the track number for G?

A graph is 2-track if and only if it is a forest of caterpillars



Leveled planarity

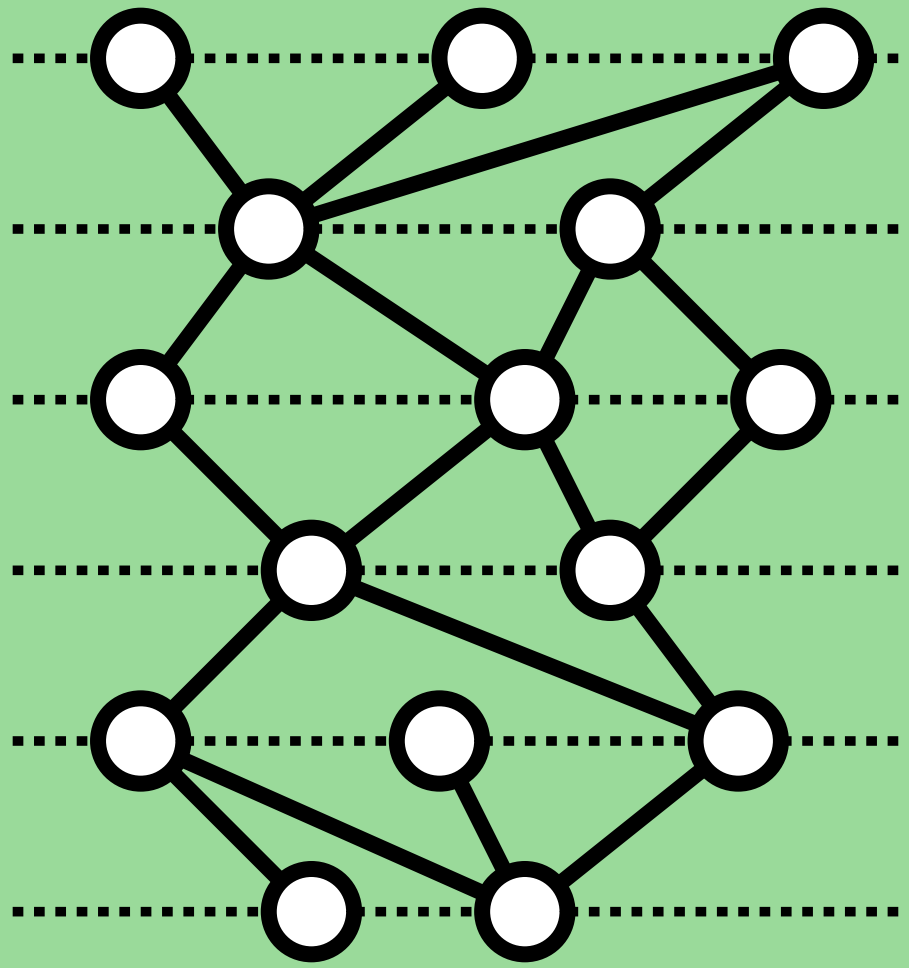
Perfect Sugiyama-style layered drawing

No edge crossing

No dummy vertices

Recognition is NP-complete

[Heath and Rosenberg, 1992]



Leveled planarity

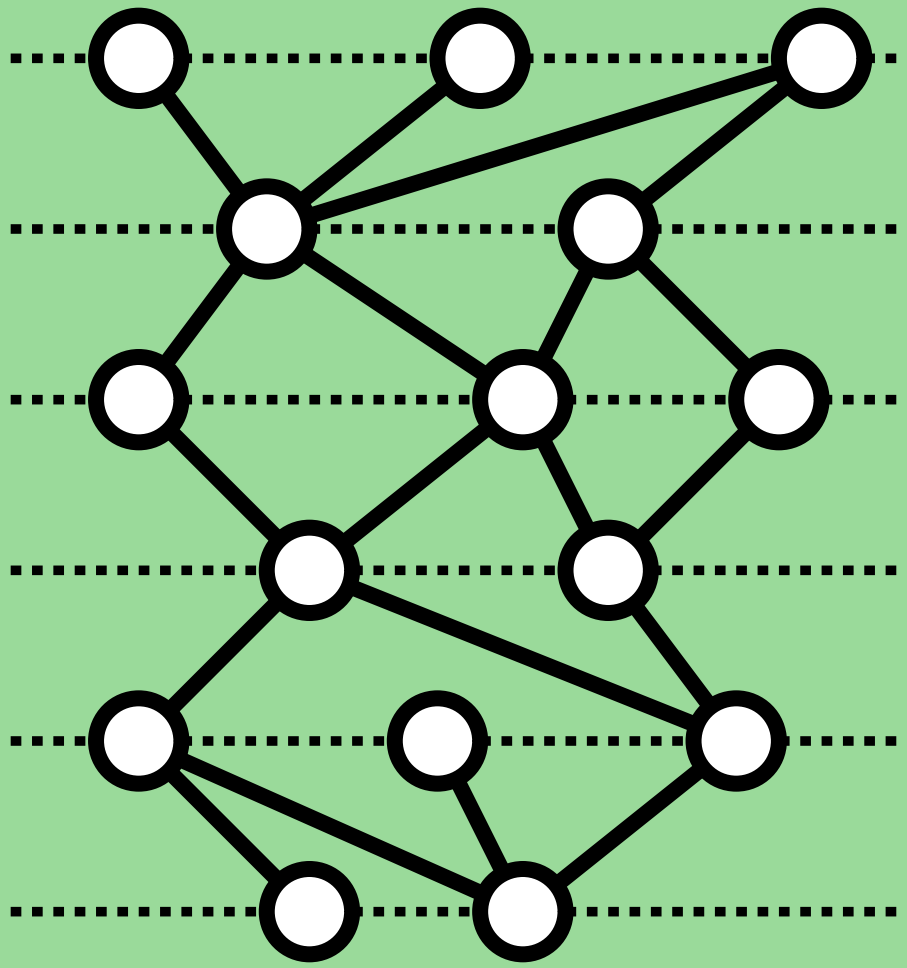
Perfect Sugiyama-style layered drawing

No edge crossing

No dummy vertices

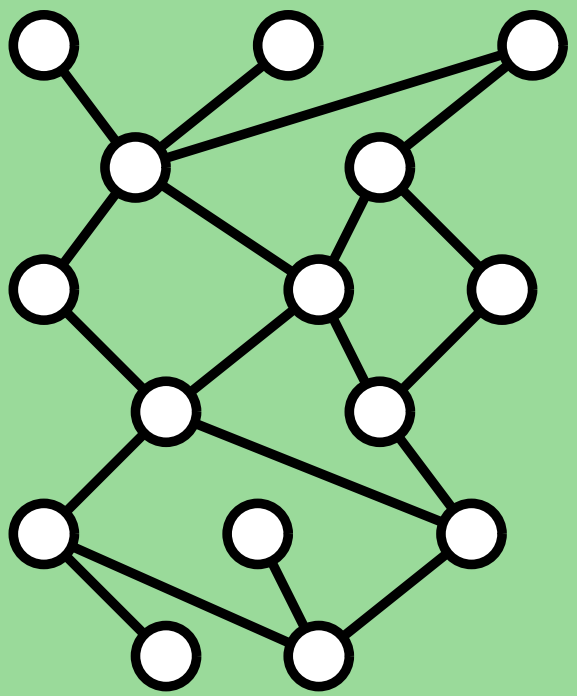
Recognition is NP-complete

[Heath and Rosenberg, 1992]

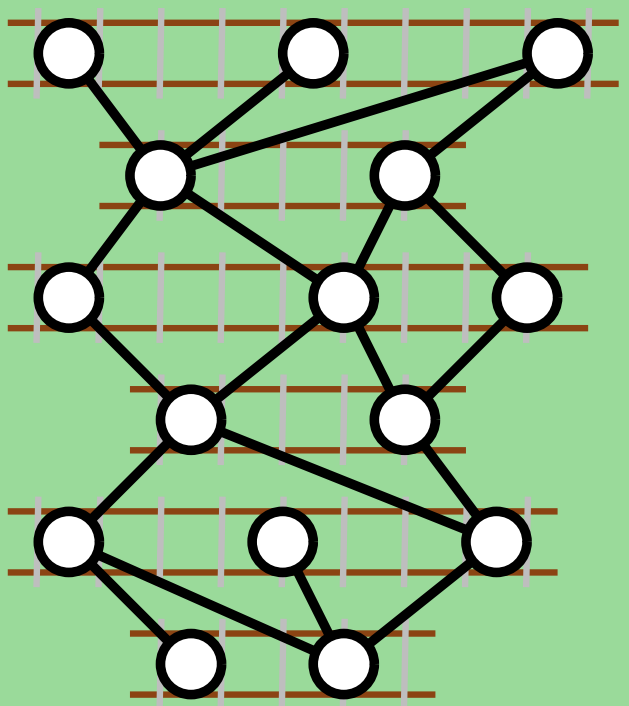


Will show leveled planar graphs are the same as bipartite 3-track graphs

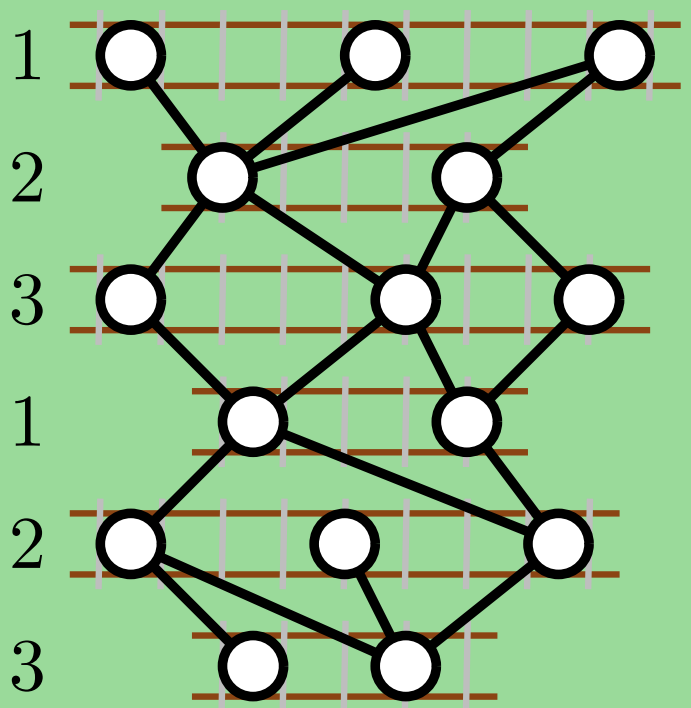
Lemma: Every leveled planar graph has a 3-track layout.



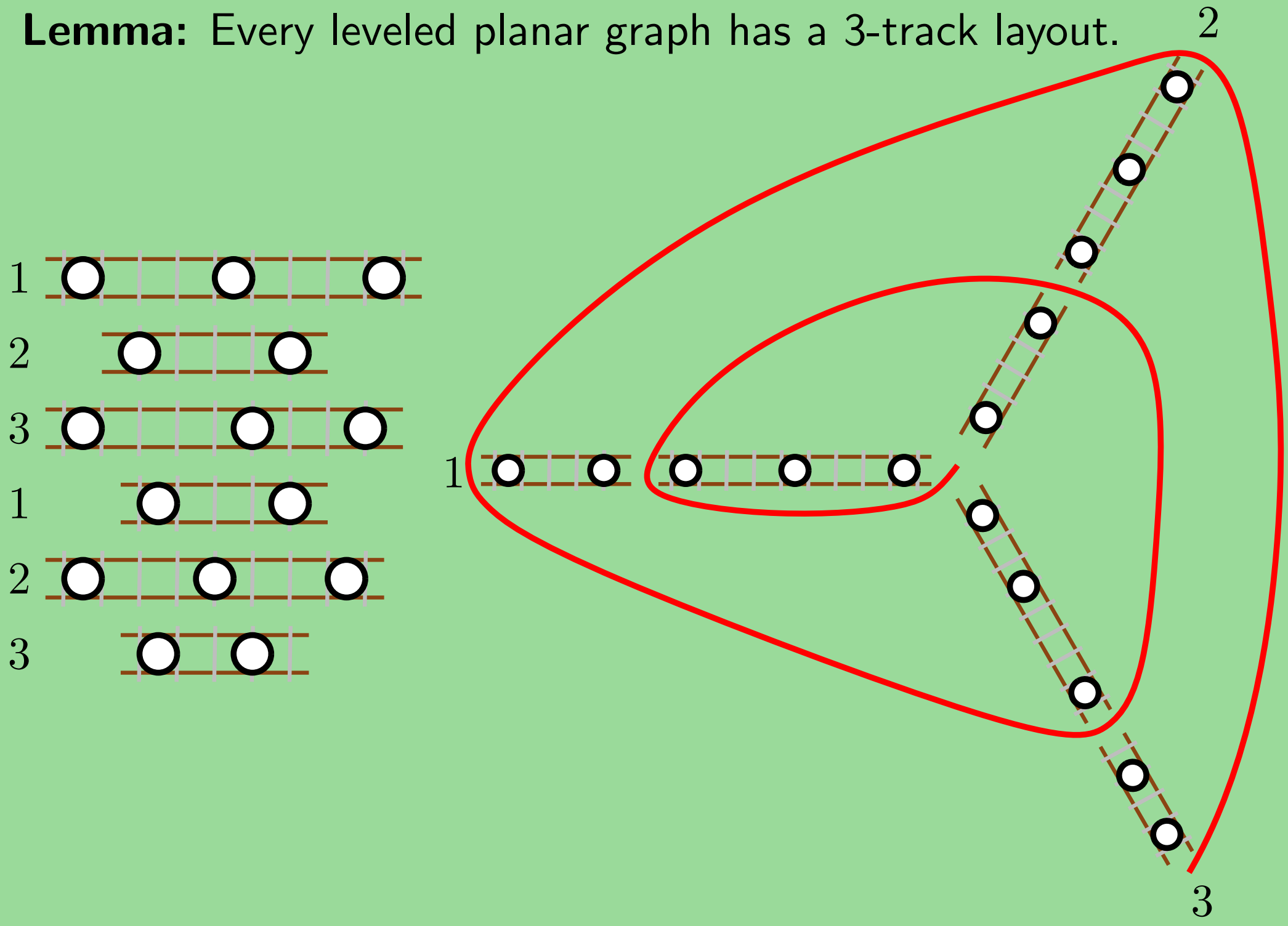
Lemma: Every leveled planar graph has a 3-track layout.



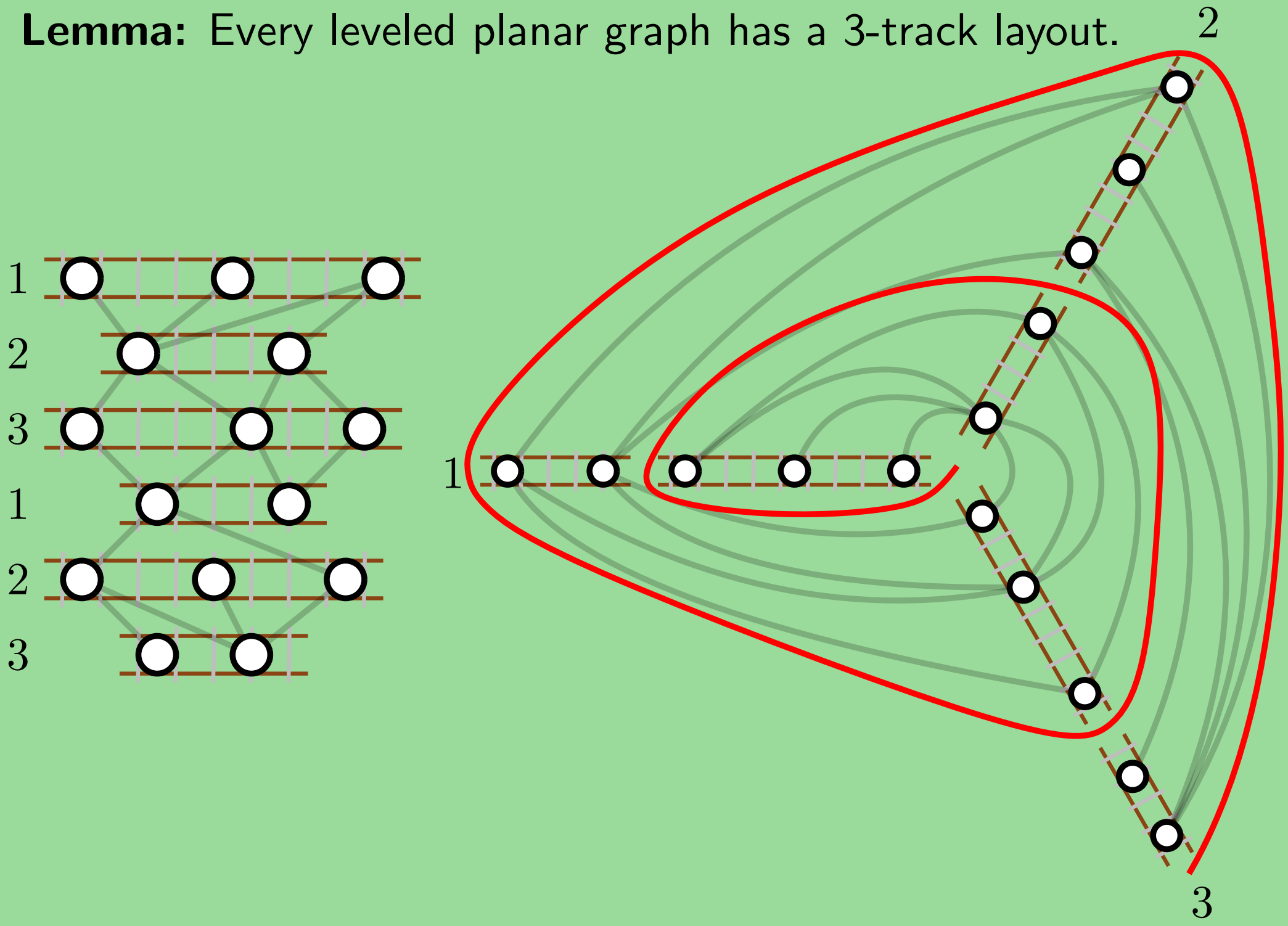
Lemma: Every leveled planar graph has a 3-track layout.



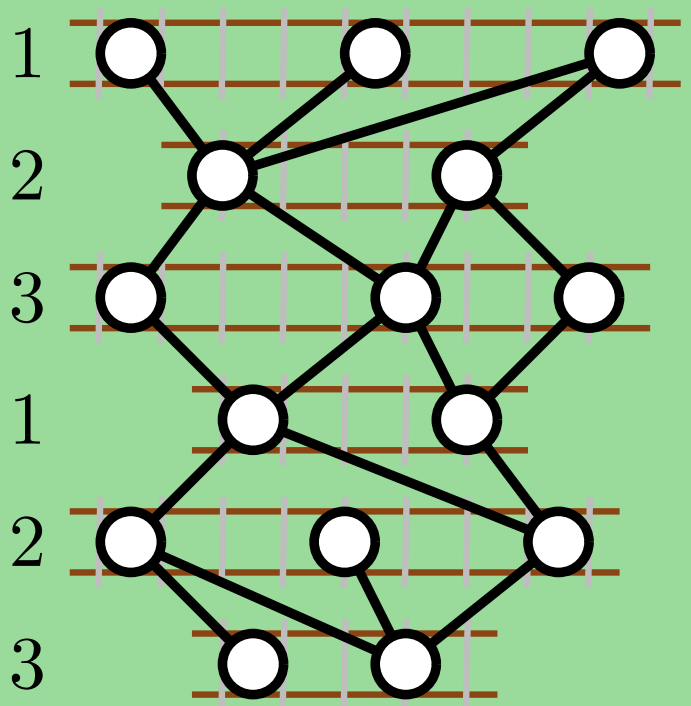
Lemma: Every leveled planar graph has a 3-track layout.

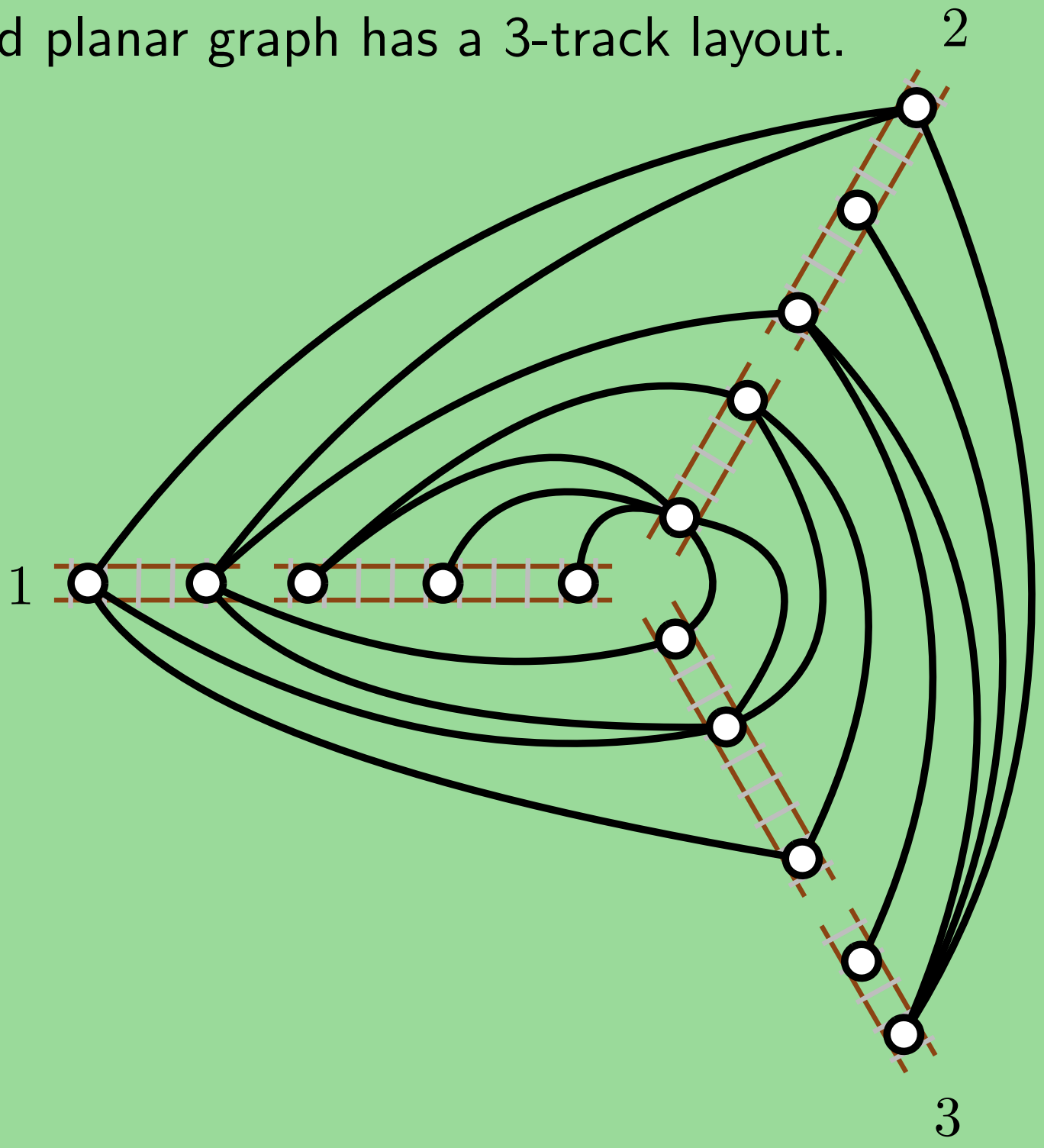


Lemma: Every leveled planar graph has a 3-track layout.



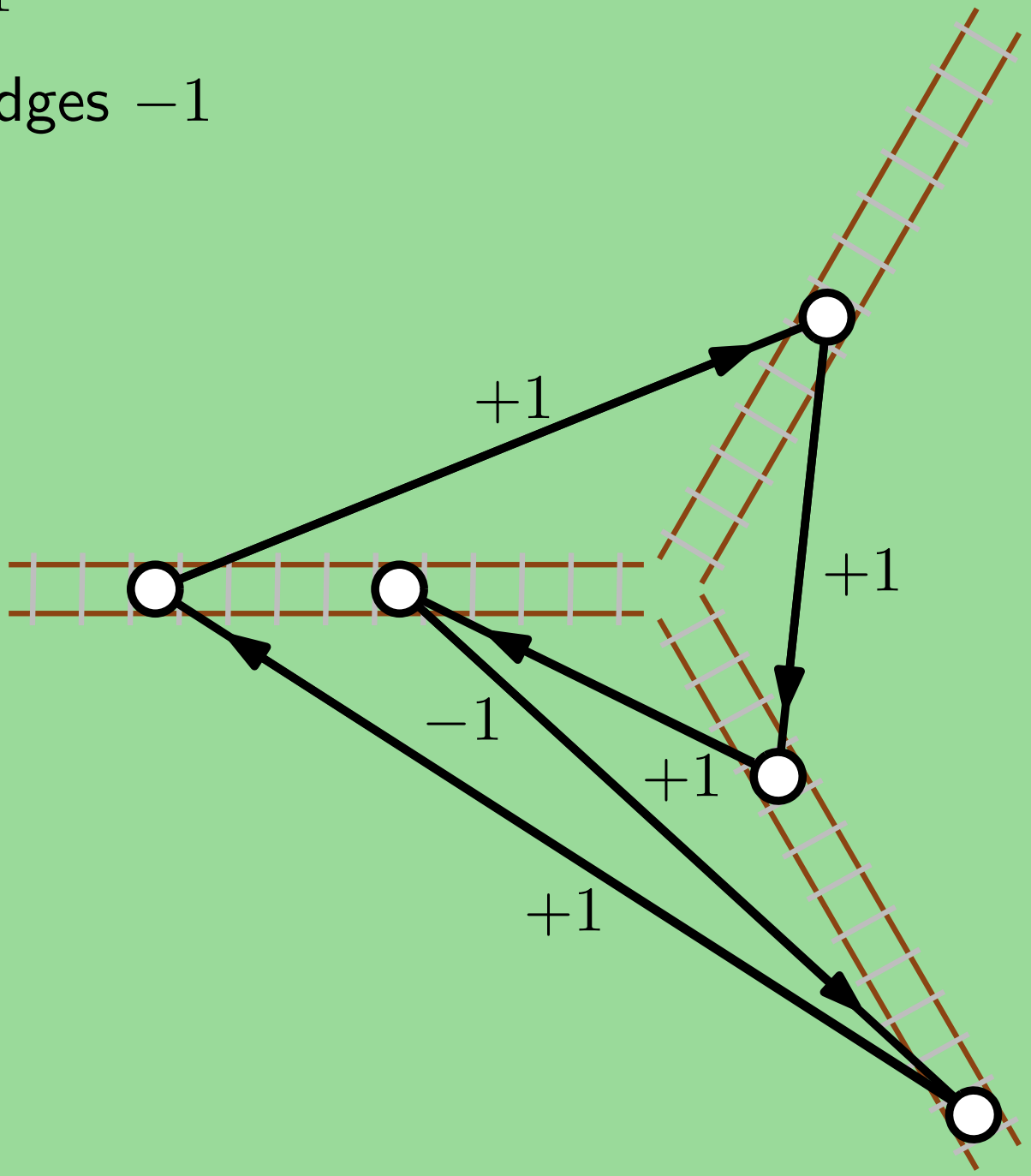
Lemma: Every leveled planar graph has a 3-track layout.





Label clockwise edges $+1$

Label counterclockwise edges -1



Label clockwise edges $+1$

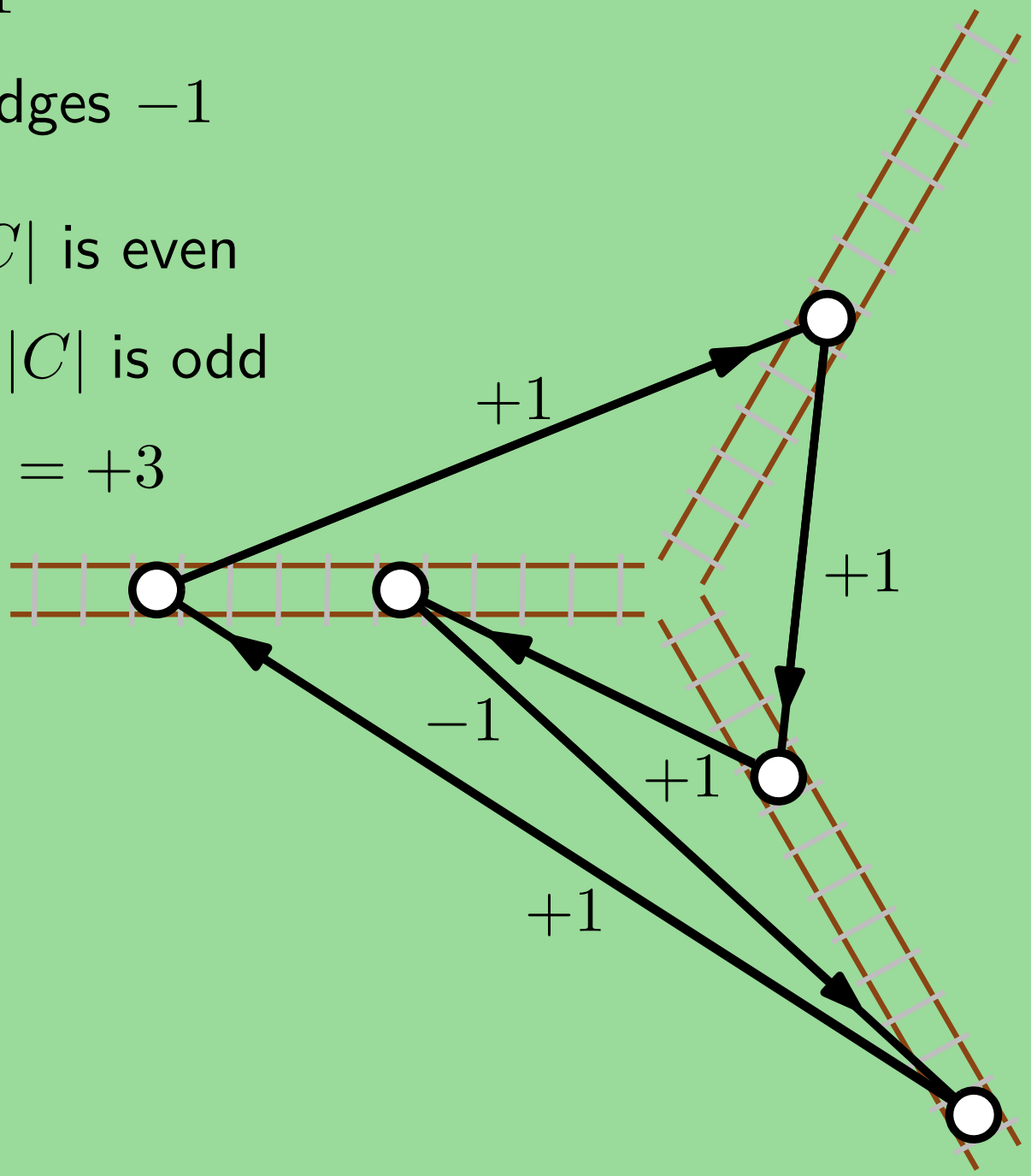
Label counterclockwise edges -1

Lemma:

$$\sum_{e \in C} \text{label}(e) = \begin{cases} 0 \text{ if } |C| \text{ is even} \\ \pm 3 \text{ if } |C| \text{ is odd} \end{cases}$$

$$+1 + 1 + 1 - 1 + 1 = +3$$

Proof:



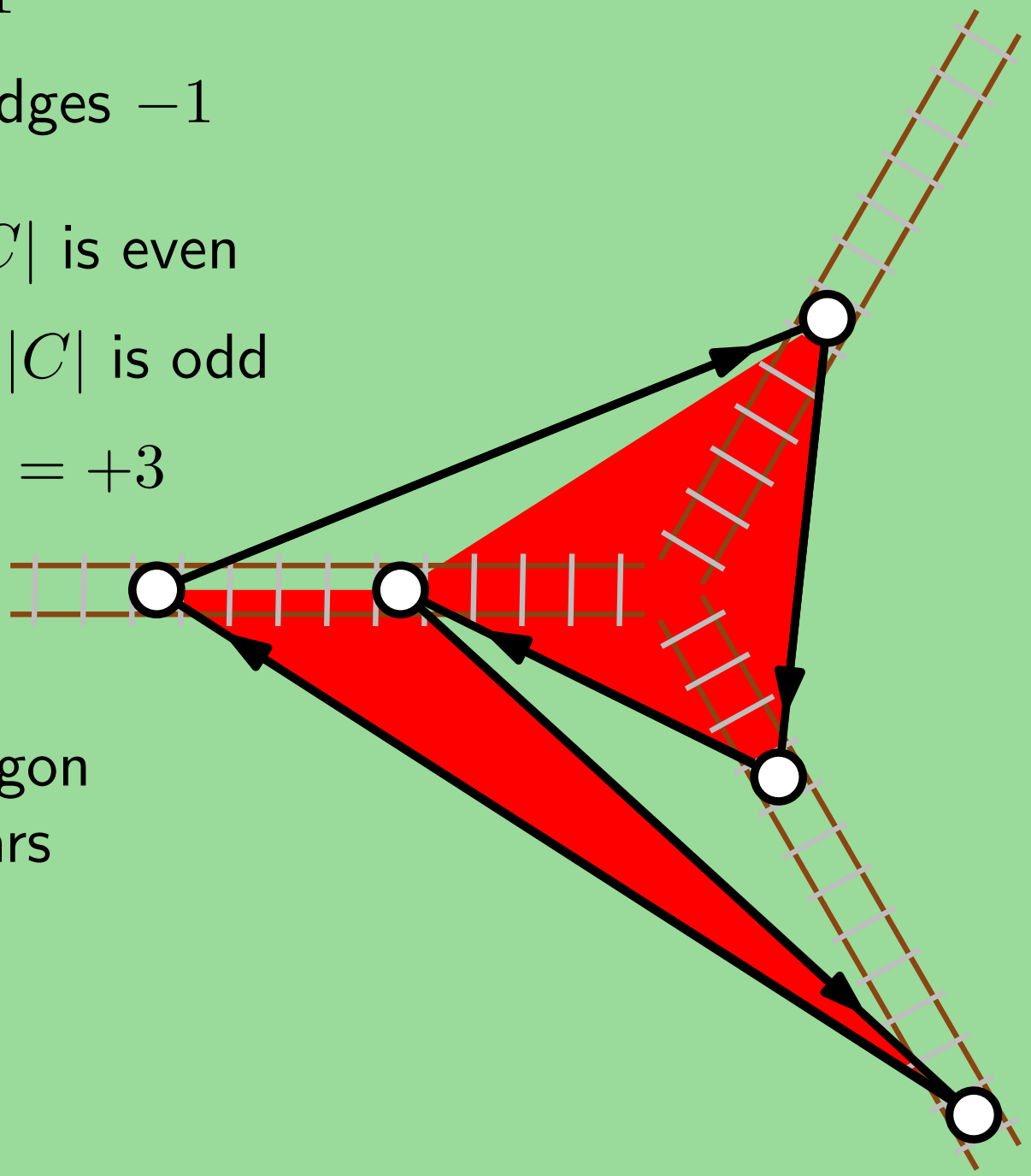
Label clockwise edges $+1$

Label counterclockwise edges -1

Lemma:

$$\sum_{e \in C} \text{label}(e) = \begin{cases} 0 \text{ if } |C| \text{ is even} \\ \pm 3 \text{ if } |C| \text{ is odd} \end{cases}$$

$$+1 + 1 + 1 - 1 + 1 = +3$$



Proof:

The cycle forms a polygon
and has at least two ears

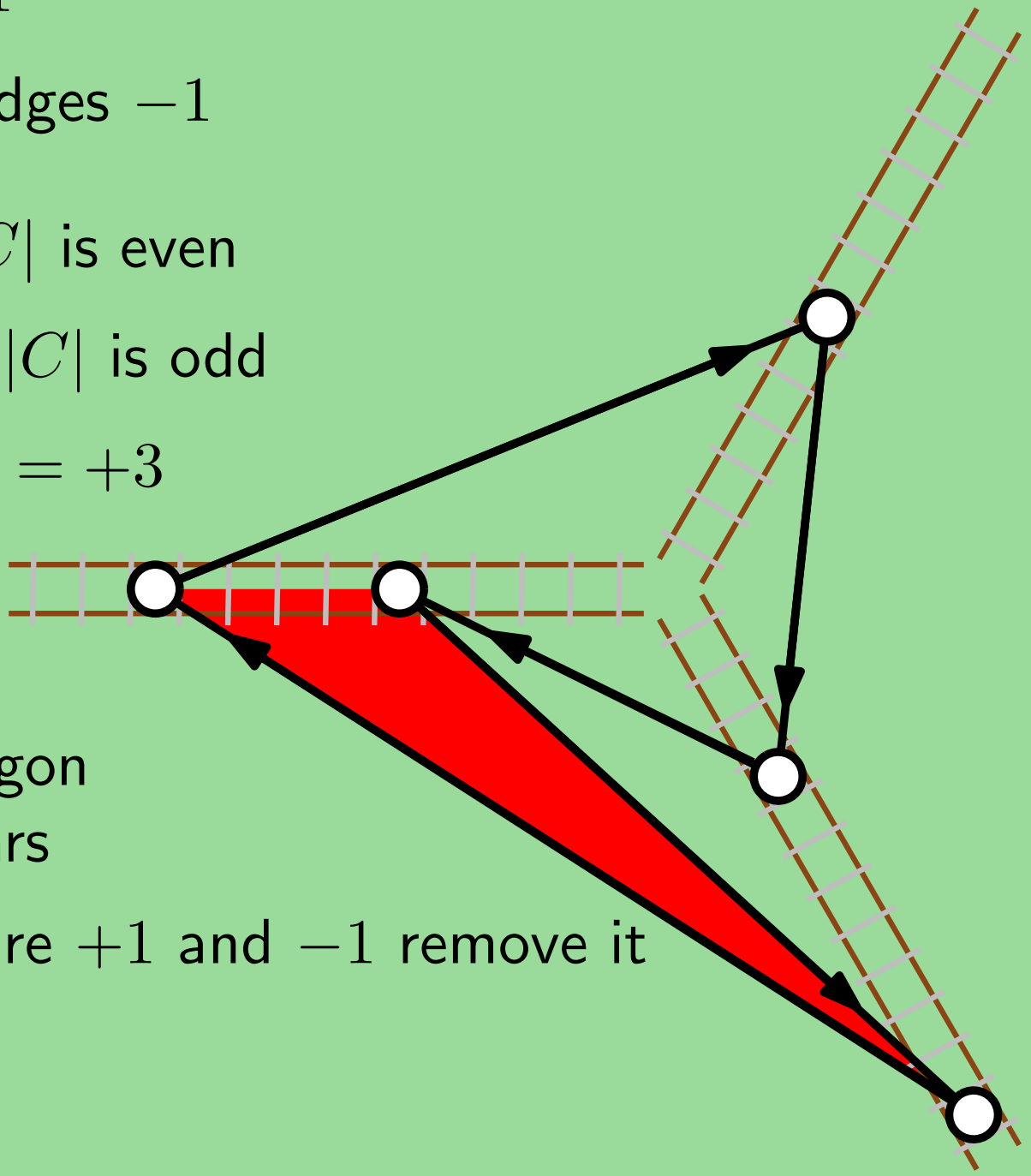
Label clockwise edges $+1$

Label counterclockwise edges -1

Lemma:

$$\sum_{e \in C} \text{label}(e) = \begin{cases} 0 \text{ if } |C| \text{ is even} \\ \pm 3 \text{ if } |C| \text{ is odd} \end{cases}$$

$$+1 + 1 + 1 - 1 + 1 = +3$$



Proof:

The cycle forms a polygon
and has at least two ears

If the edges of an ear are $+1$ and -1 remove it

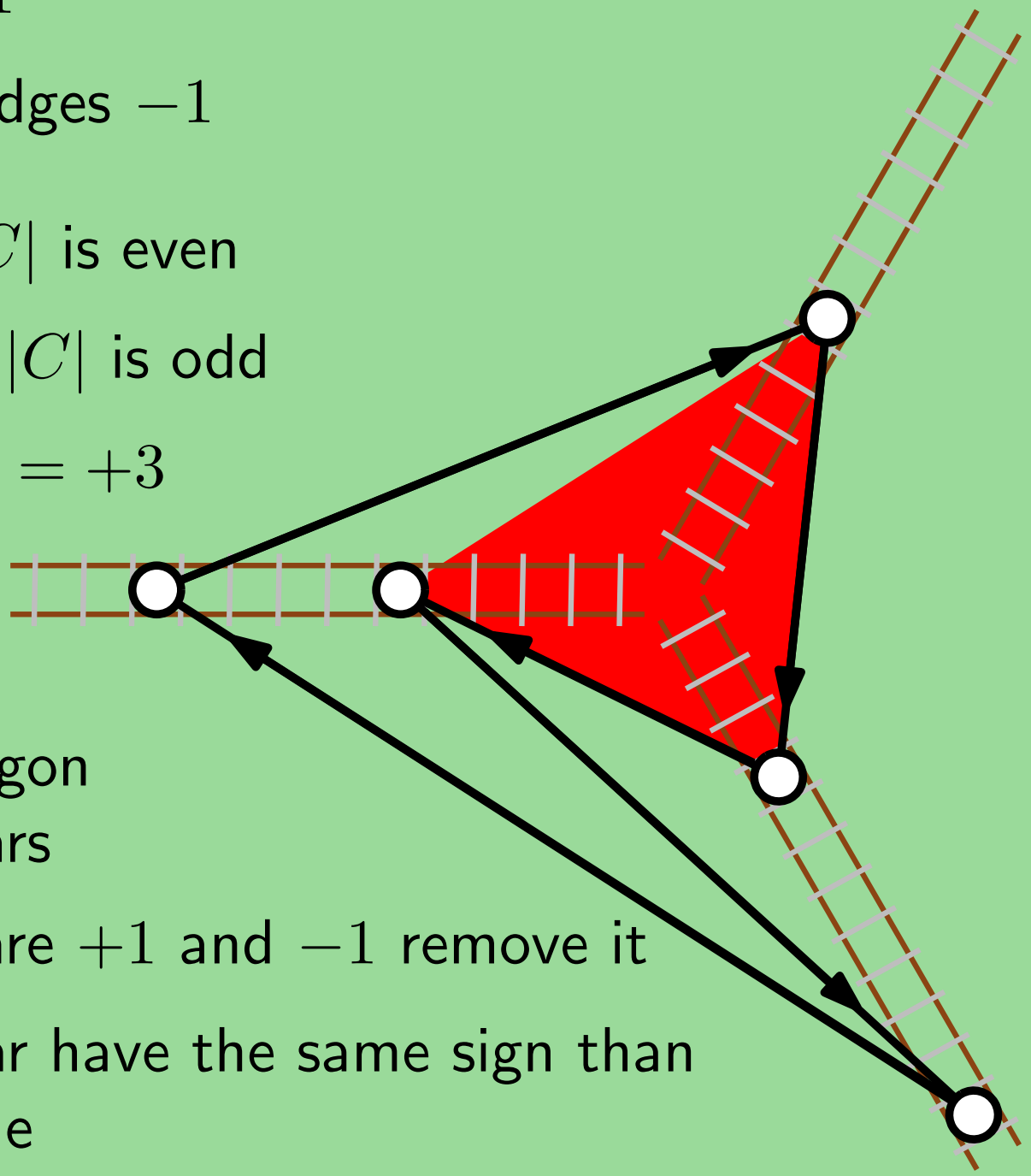
Label clockwise edges $+1$

Label counterclockwise edges -1

Lemma:

$$\sum_{e \in C} \text{label}(e) = \begin{cases} 0 \text{ if } |C| \text{ is even} \\ \pm 3 \text{ if } |C| \text{ is odd} \end{cases}$$

$$+1 + 1 + 1 - 1 + 1 = +3$$



Proof:

The cycle forms a polygon and has at least two ears

If the edges of an ear are $+1$ and -1 remove it

If the edges of every ear have the same sign than the polygon is a triangle

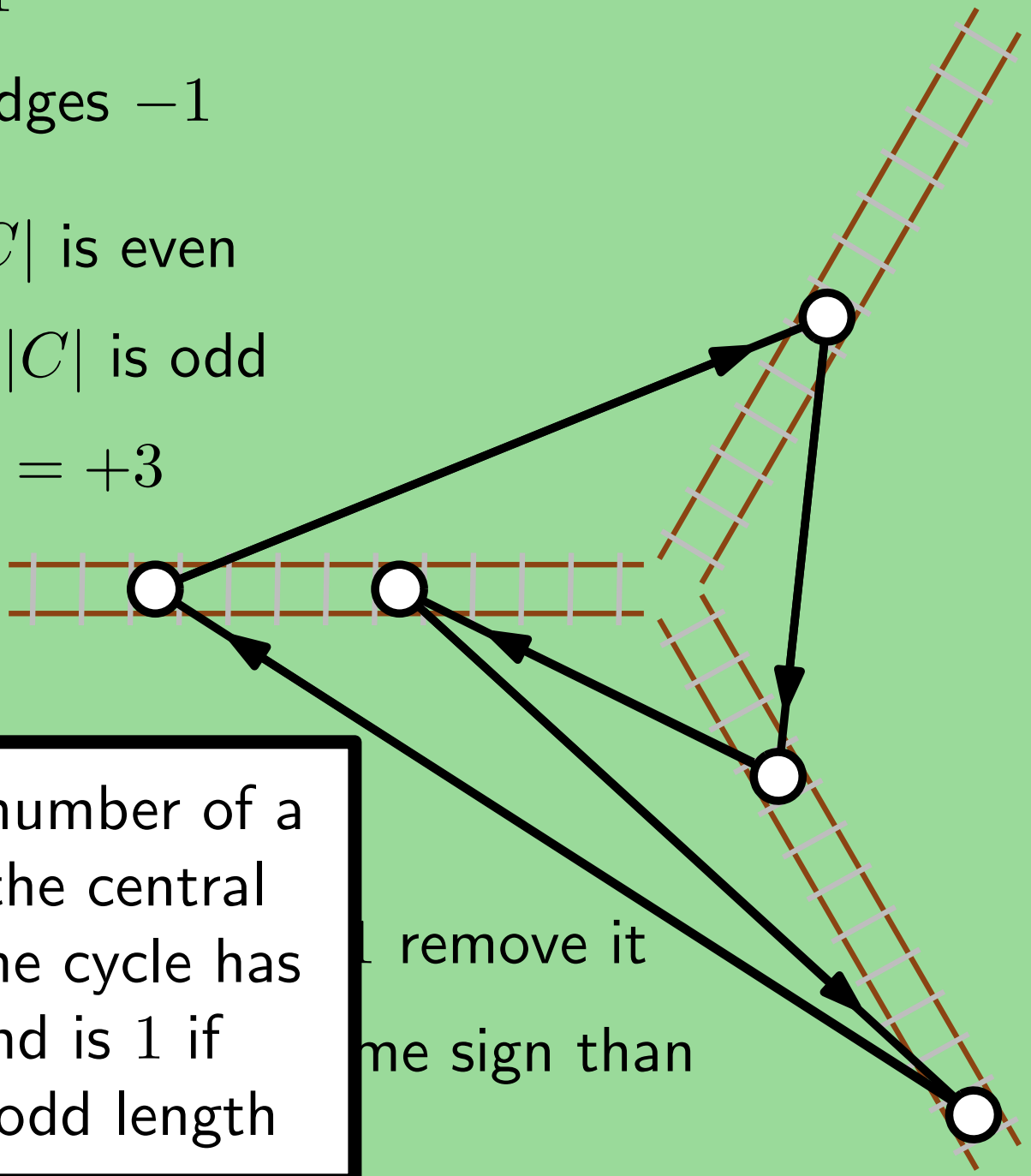
Label clockwise edges $+1$

Label counterclockwise edges -1

Lemma:

$$\sum_{e \in C} \text{label}(e) = \begin{cases} 0 \text{ if } |C| \text{ is even} \\ \pm 3 \text{ if } |C| \text{ is odd} \end{cases}$$

$$+1+1+1-1+1=+3$$



Proof:

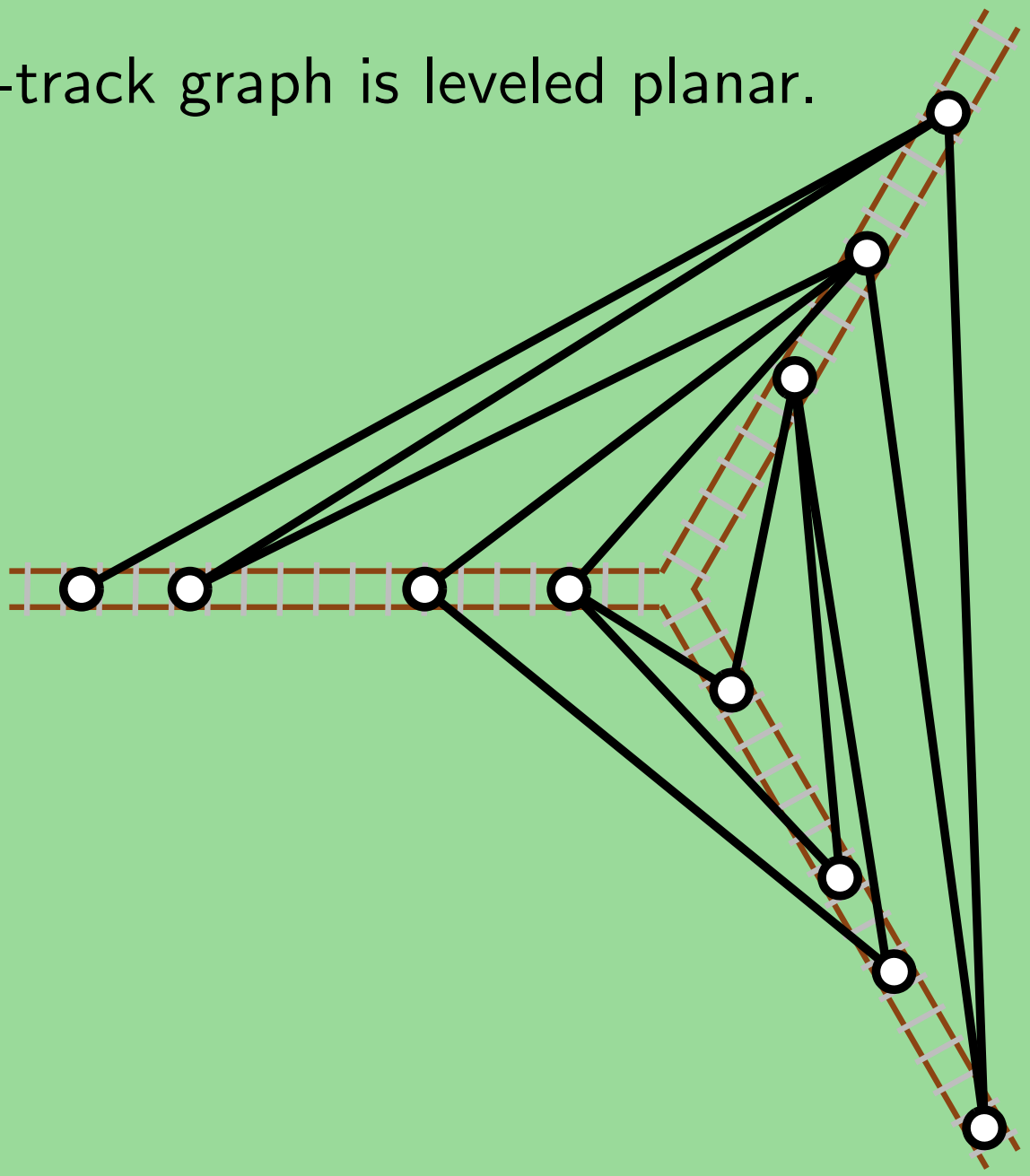
The cyc
and has

If the e remove it

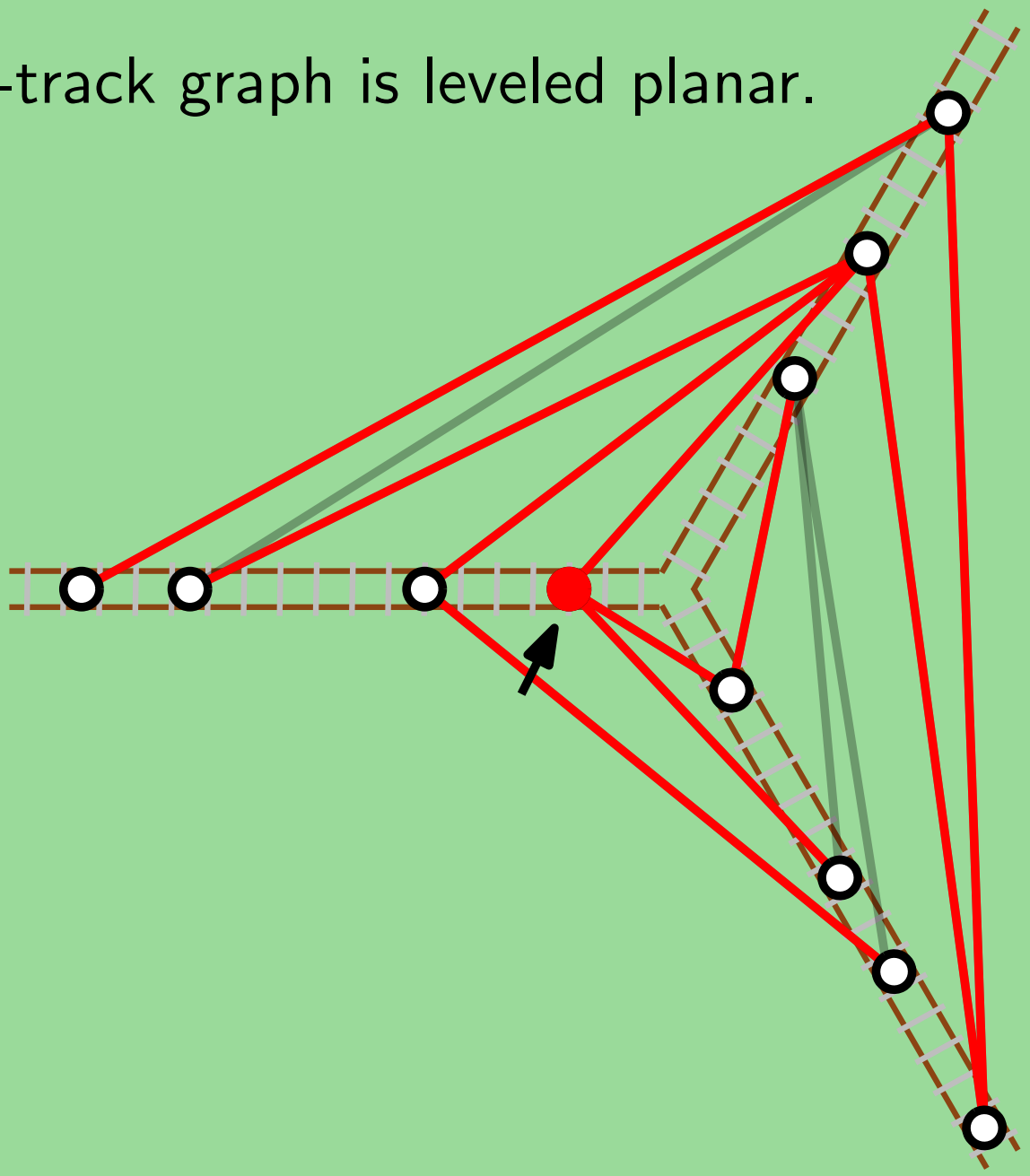
If the e me sign than
the poly

The winding number of a cycle around the central point is 0 if the cycle has even length and is 1 if the cycle has odd length

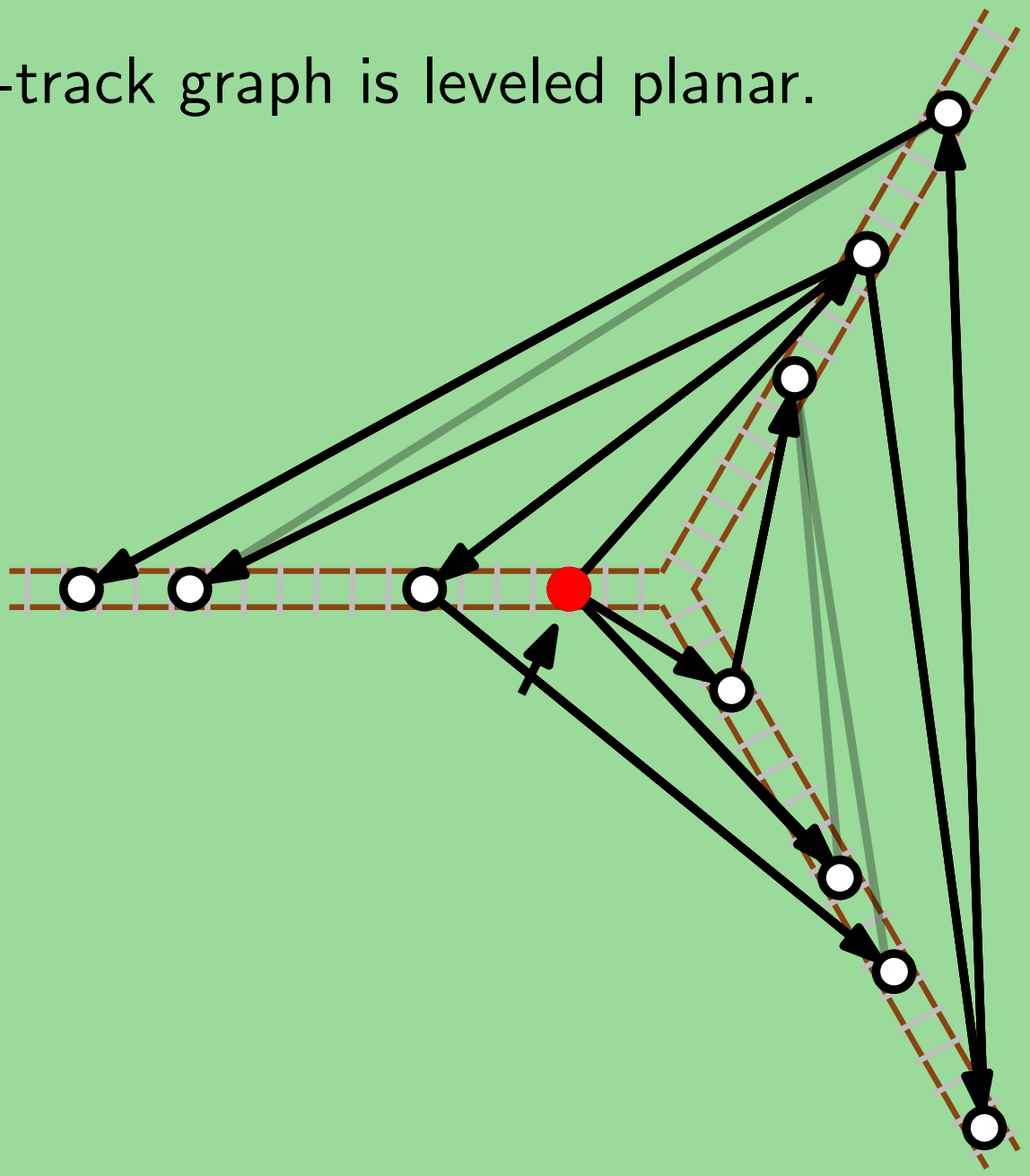
Lemma: Every bipartite 3-track graph is leveled planar.



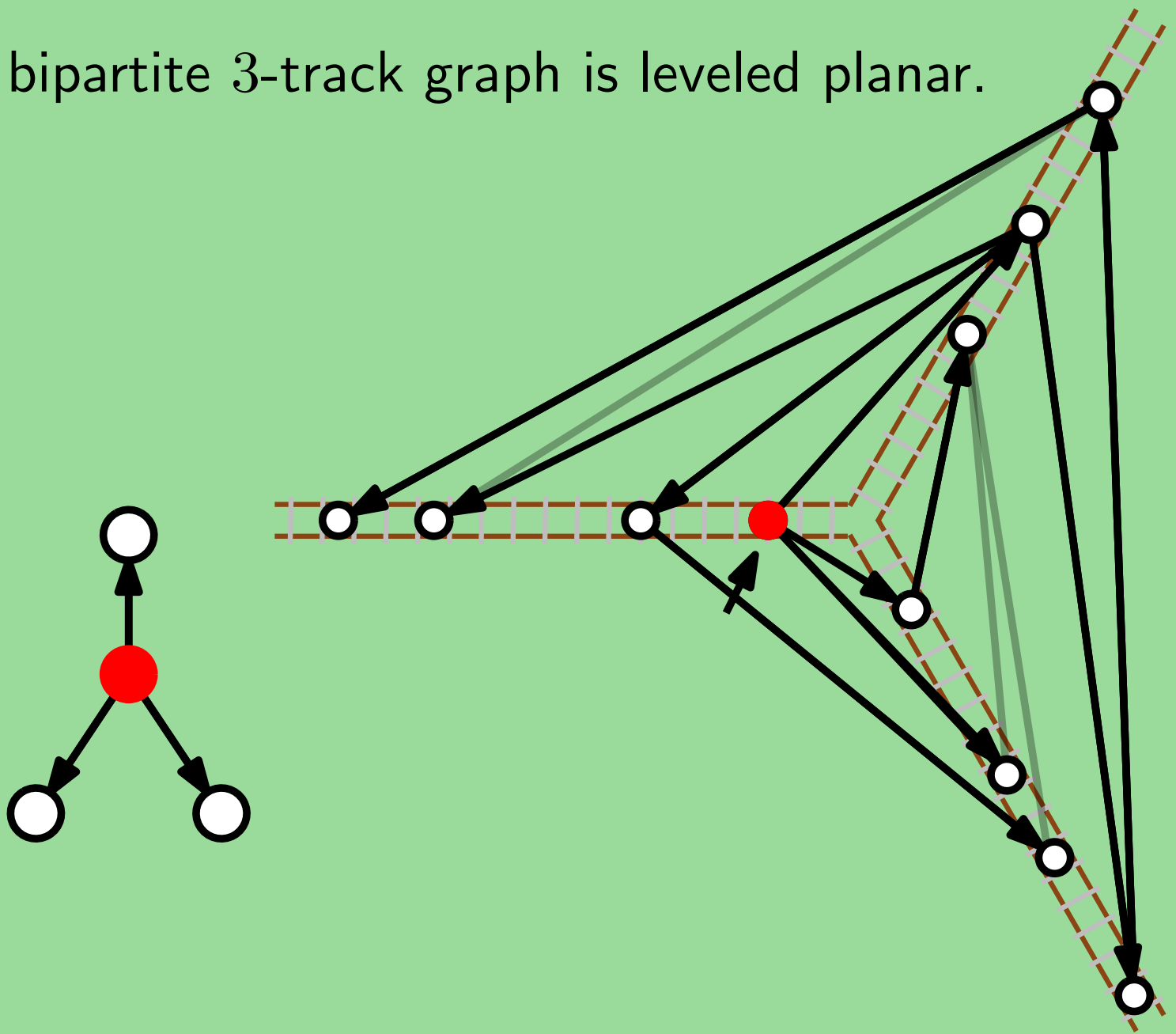
Lemma: Every bipartite 3-track graph is leveled planar.



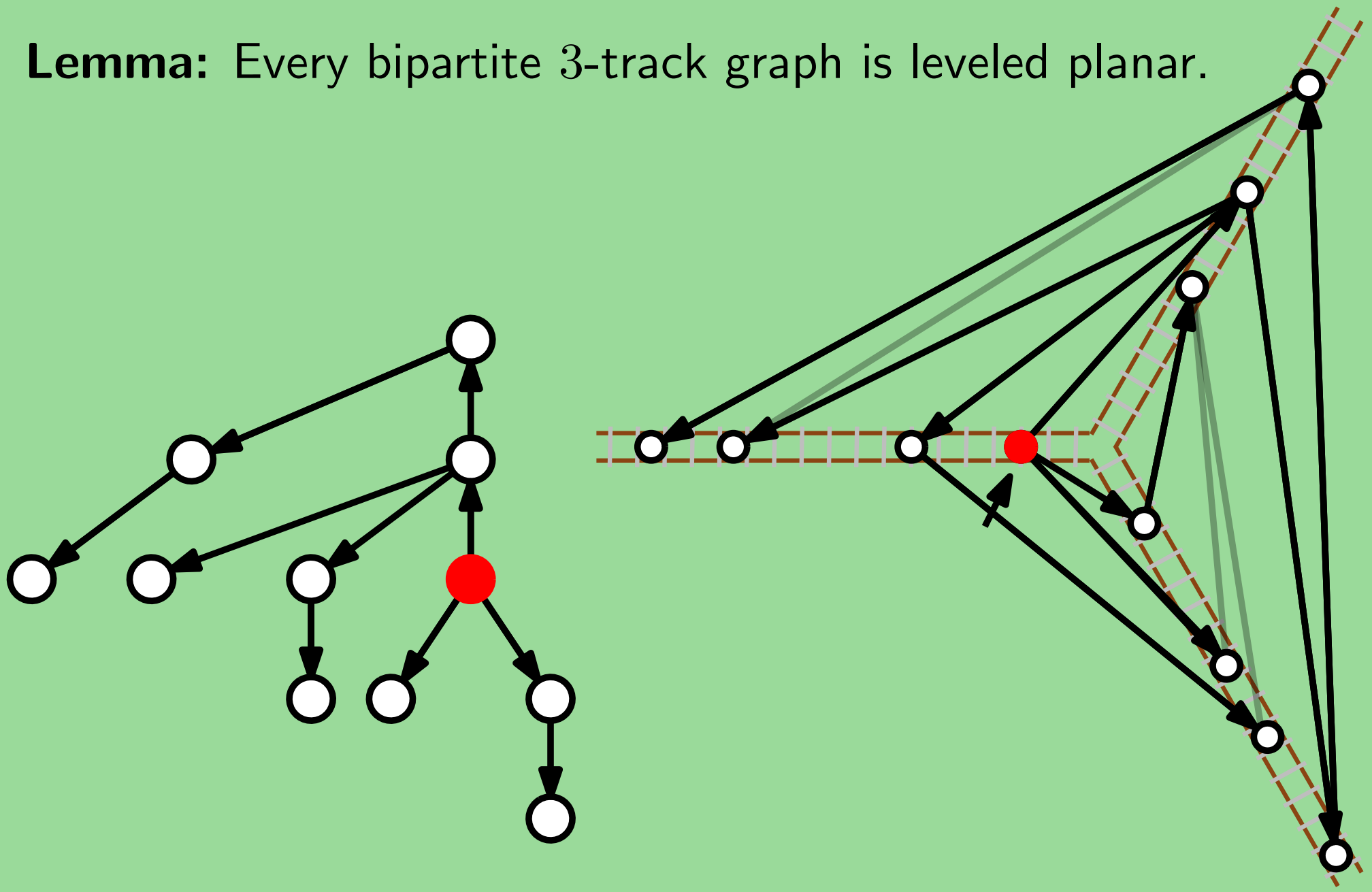
Lemma: Every bipartite 3-track graph is leveled planar.



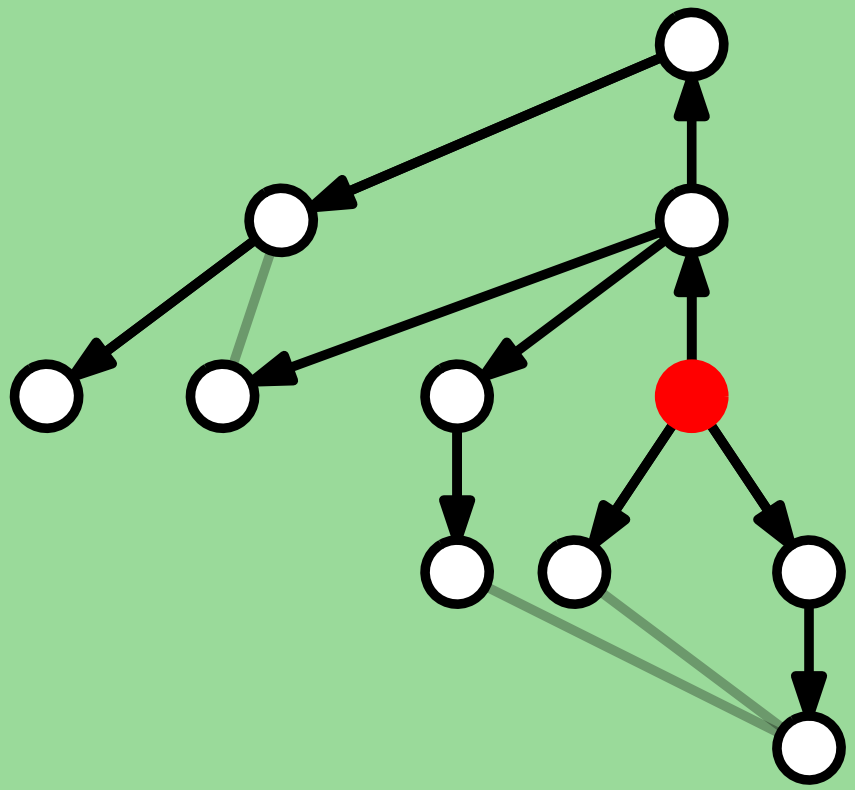
Lemma: Every bipartite 3-track graph is leveled planar.

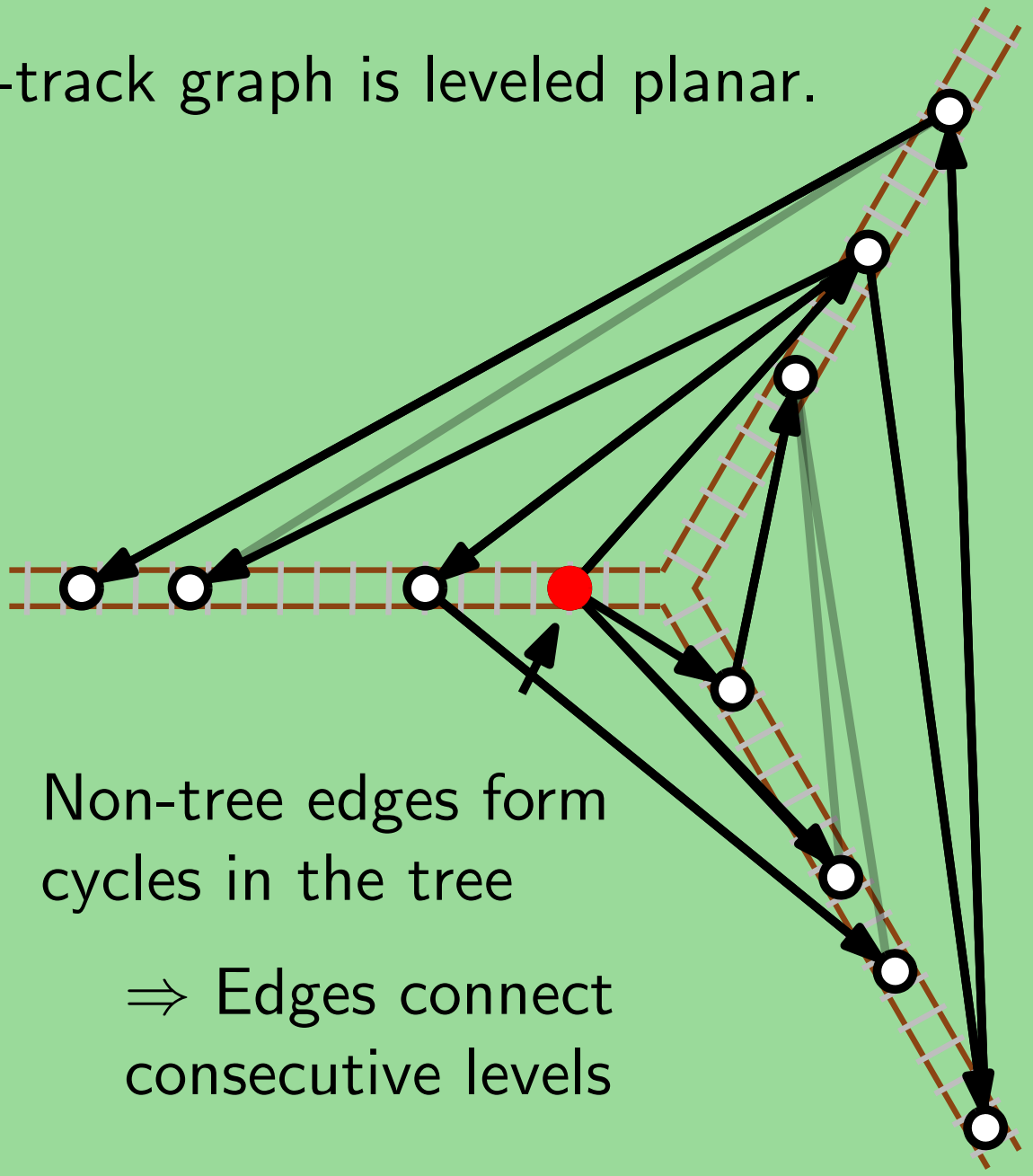


Lemma: Every bipartite 3-track graph is leveled planar.



Lemma: Every bipartite 3-track graph is leveled planar.

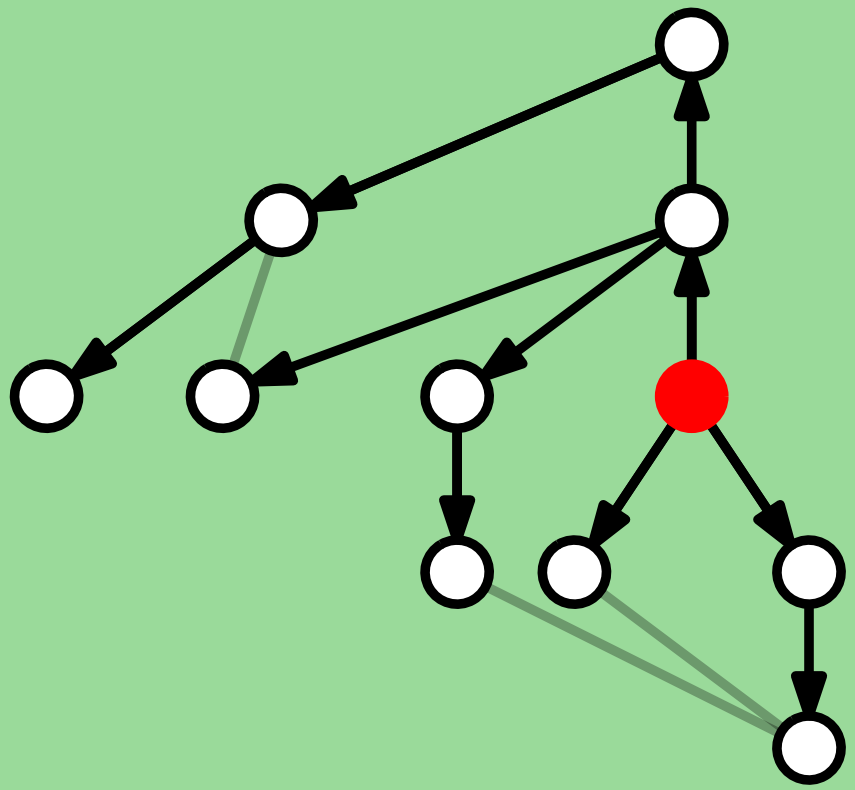


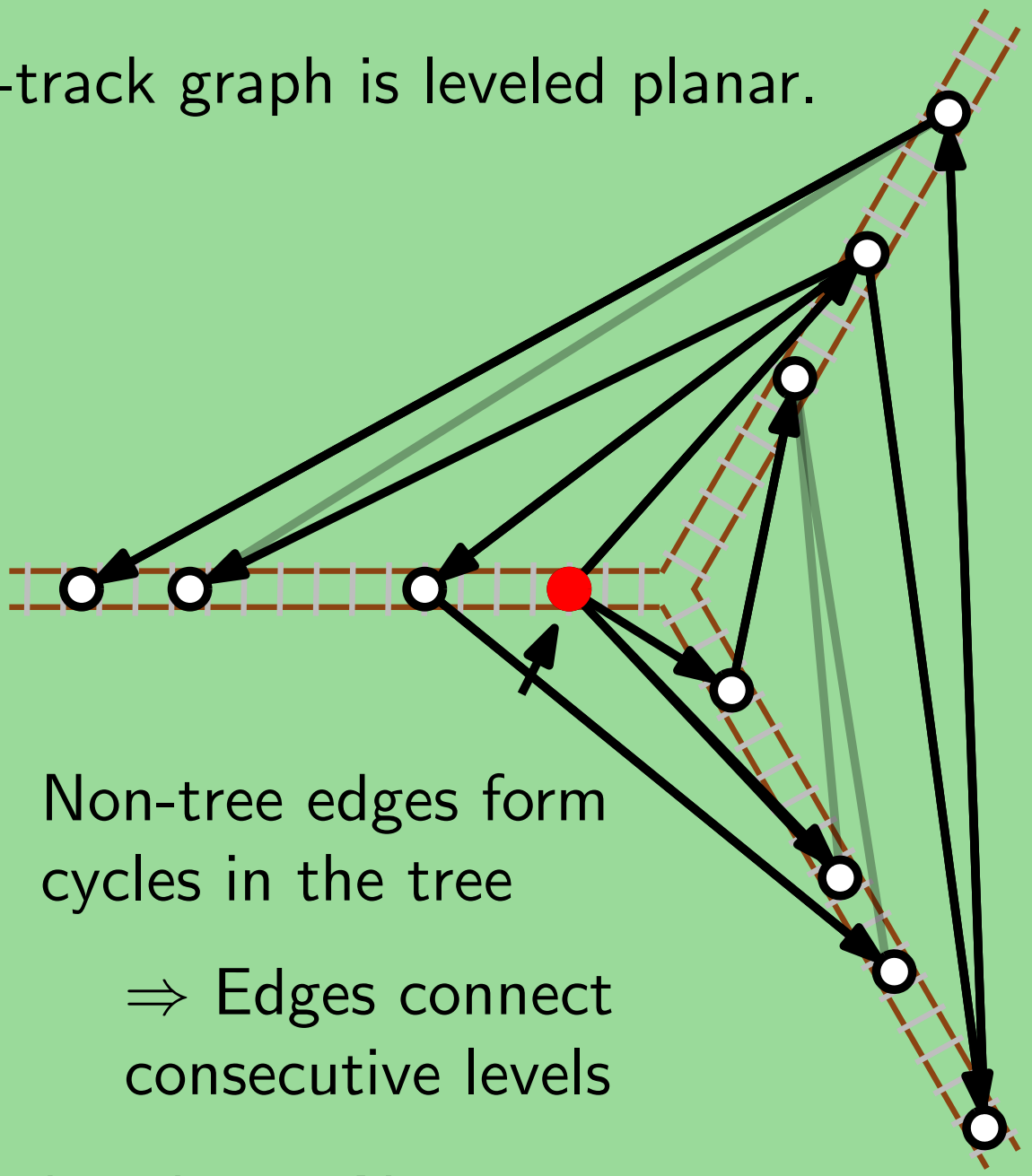


Non-tree edges form cycles in the tree

$\Rightarrow$ Edges connect consecutive levels

Lemma: Every bipartite 3-track graph is leveled planar.





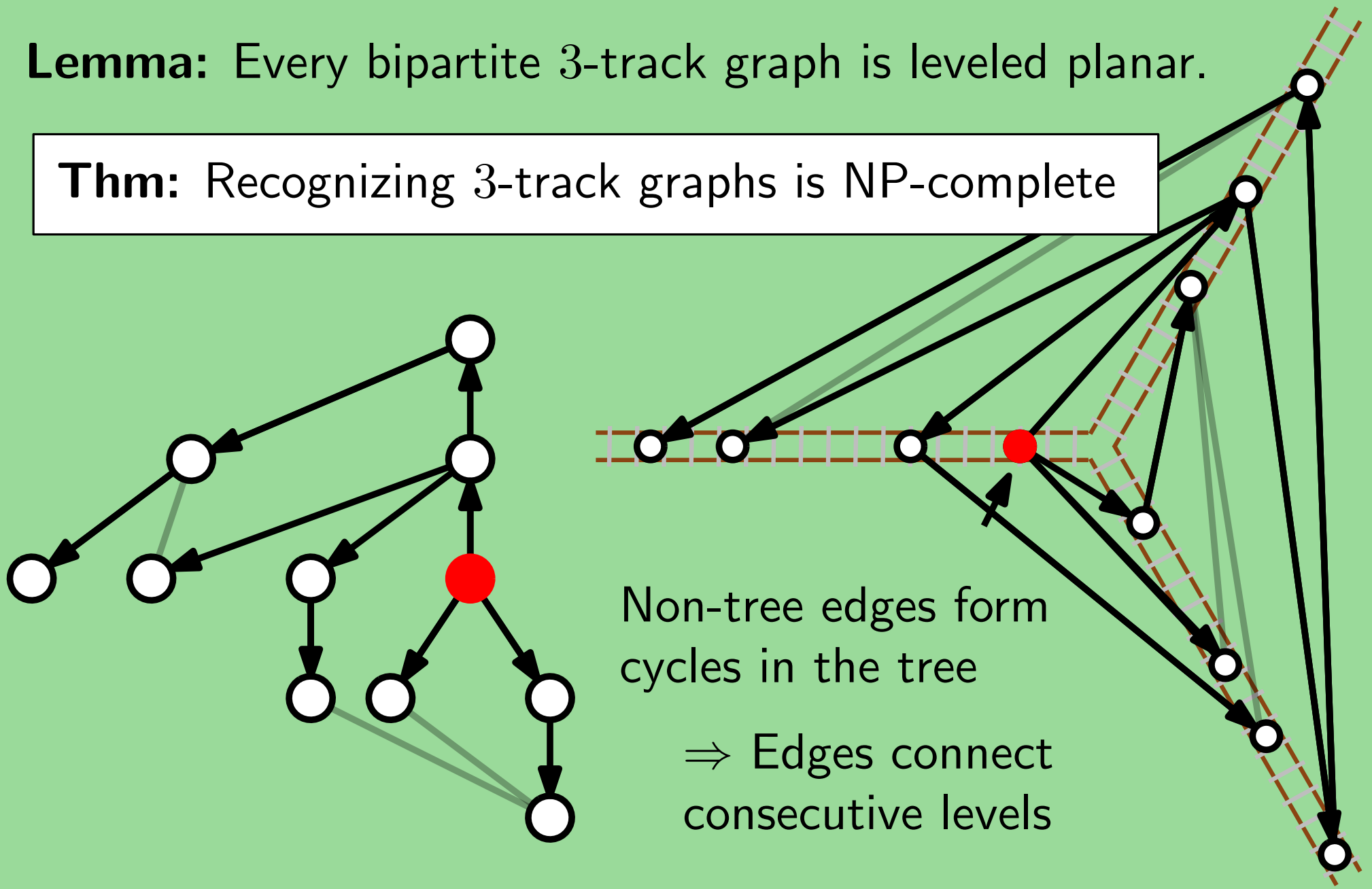
Non-tree edges form cycles in the tree

$\Rightarrow$ Edges connect consecutive levels

Order vertices by their track order $\Rightarrow$ No crossings

Lemma: Every bipartite 3-track graph is leveled planar.

Thm: Recognizing 3-track graphs is NP-complete



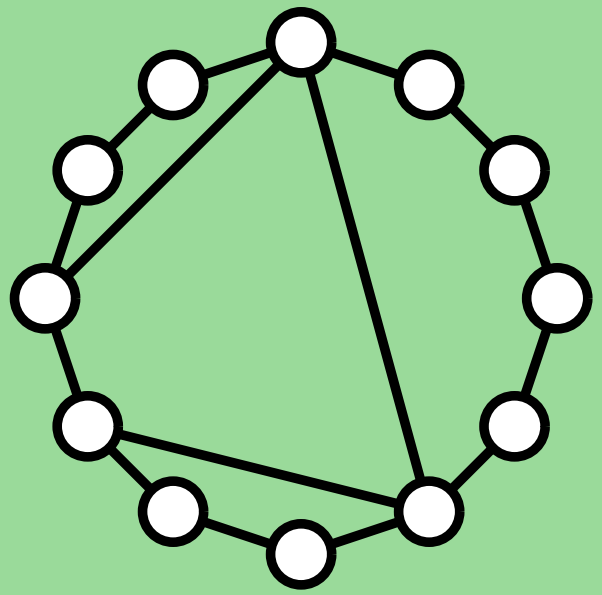
Non-tree edges form cycles in the tree

$\Rightarrow$ Edges connect consecutive levels

Order vertices by their track order $\Rightarrow$ No crossings

Which 3-track graphs can we recognize and layout 3-track?

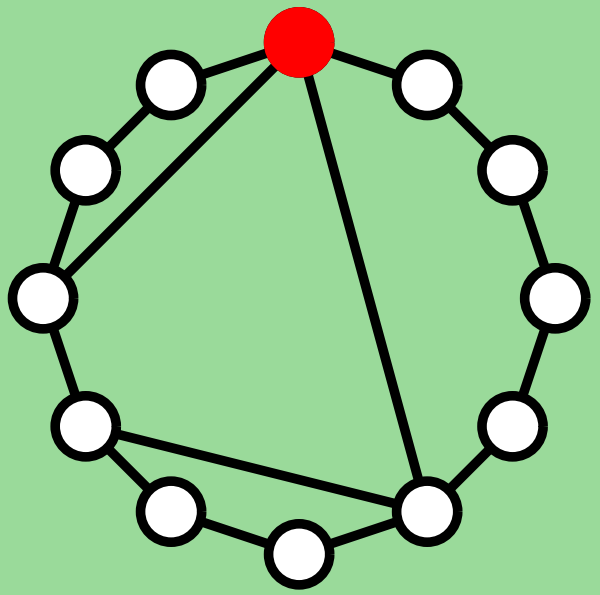
Bipartite Outerplanar Graphs

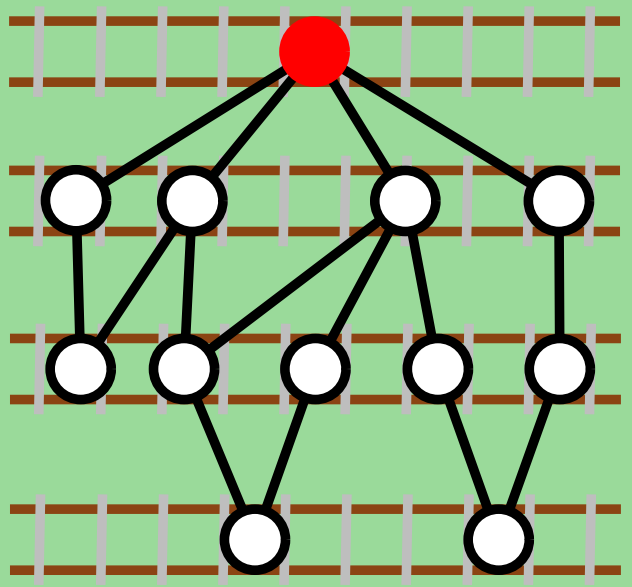


Which 3-track graphs can we recognize and layout 3-track?

Bipartite Outerplanar Graphs

Pick a vertex and perform a breadth first layering

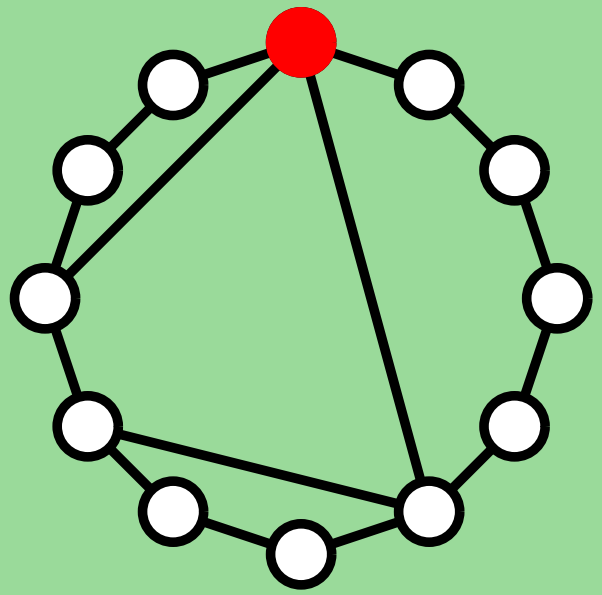


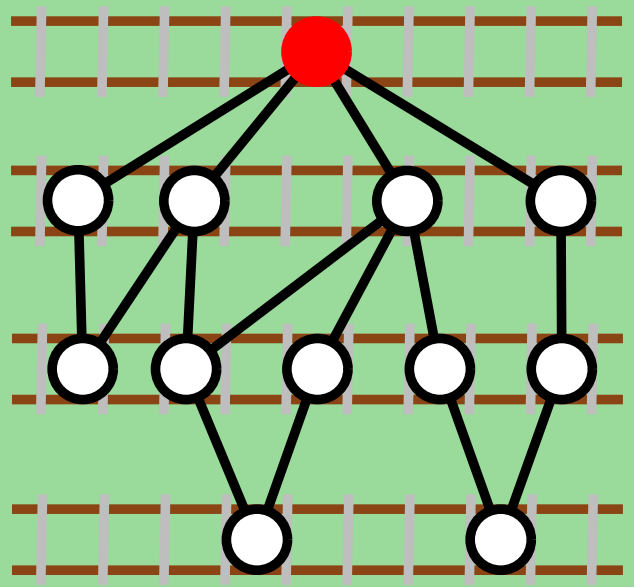


Which 3-track graphs can we recognize and layout 3-track?

Bipartite Outerplanar Graphs

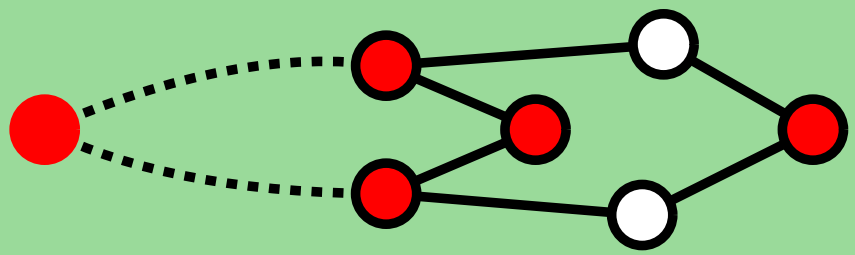
Pick a vertex and perform a breadth first layering





How to layout vertices with a level?

Each face cycle has a closest vertex to the chosen vertex

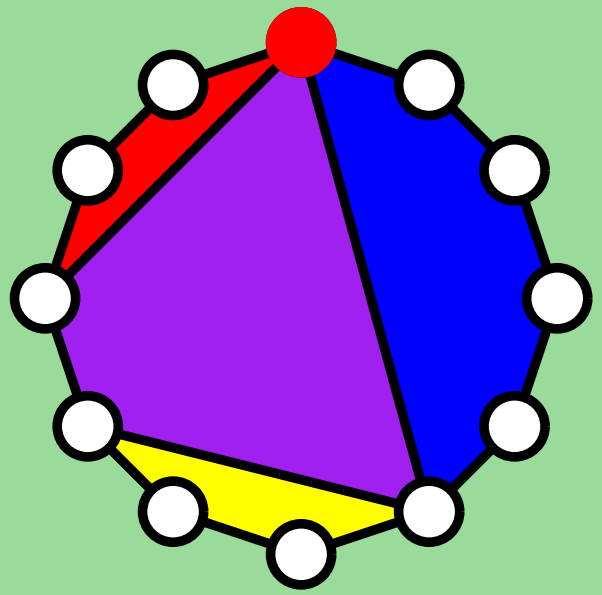


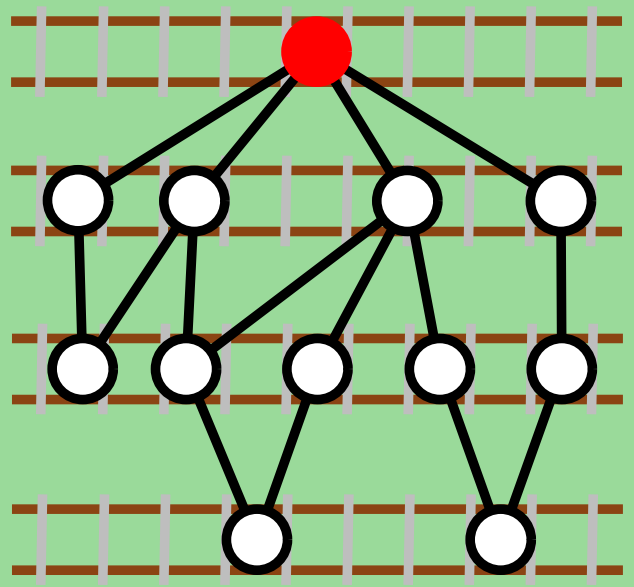
Otherwise there is a $K_{2,3}$ subdivision

Which 3-track graphs can we recognize and layout 3-track?

Bipartite Outerplanar Graphs

Pick a vertex and perform a breadth first layering





How to layout vertices with a level?

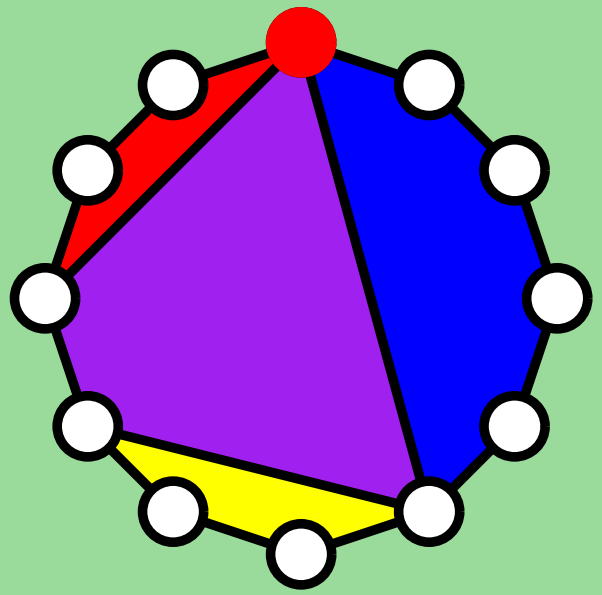
Each face cycle has a closest vertex to the chosen vertex

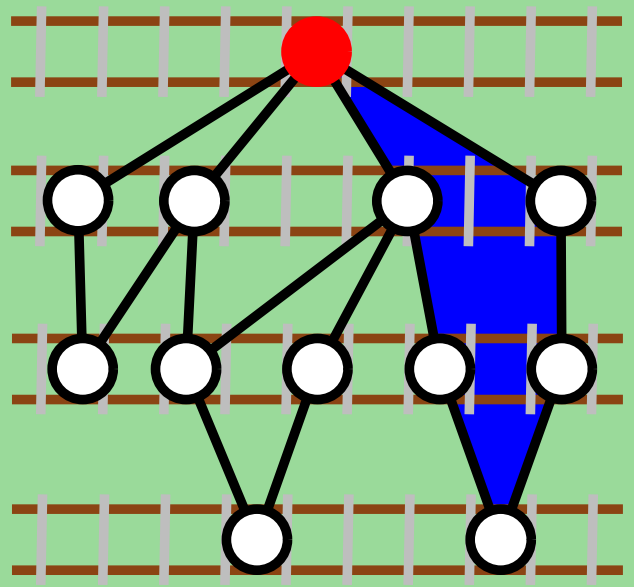
Order vertices of each face cycle independently

Which 3-track graphs can we recognize and layout 3-track?

Bipartite Outerplanar Graphs

Pick a vertex and perform a breadth first layering





How to layout vertices with a level?

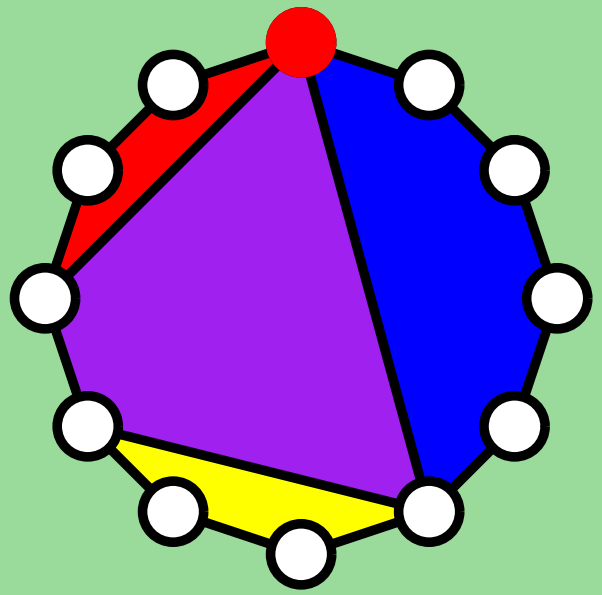
Each face cycle has a closest vertex to the chosen vertex

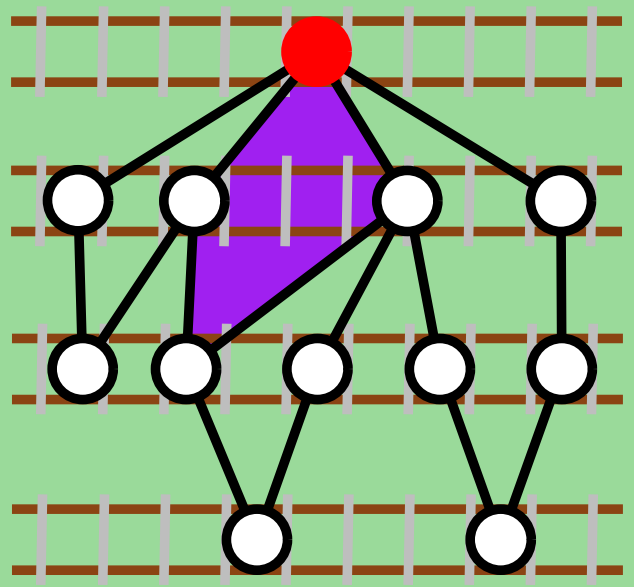
Order vertices of each face cycle independently

Which 3-track graphs can we recognize and layout 3-track?

Bipartite Outerplanar Graphs

Pick a vertex and perform a breadth first layering





How to layout vertices with a level?

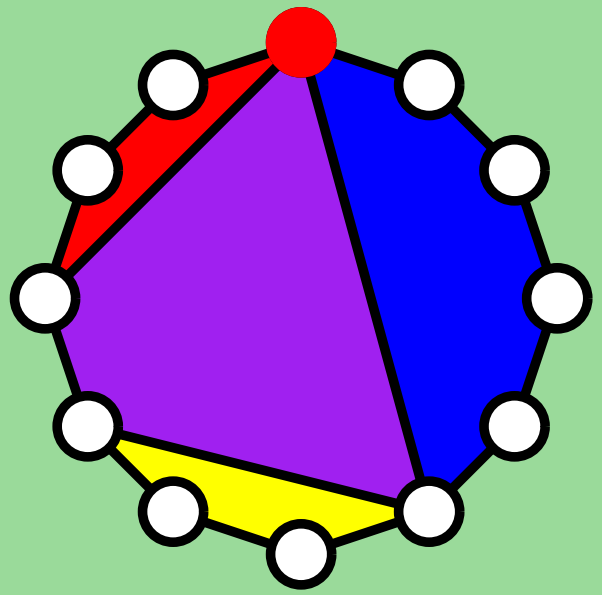
Each face cycle has a closest vertex to the chosen vertex

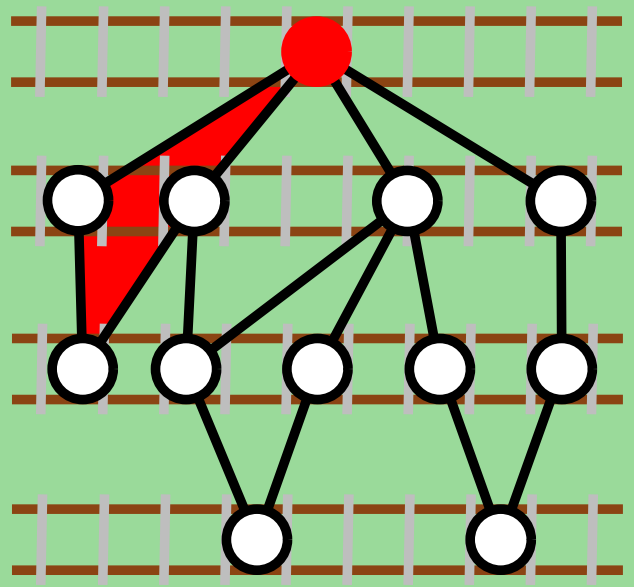
Order vertices of each face cycle independently

Which 3-track graphs can we recognize and layout 3-track?

Bipartite Outerplanar Graphs

Pick a vertex and perform a breadth first layering





How to layout vertices with a level?

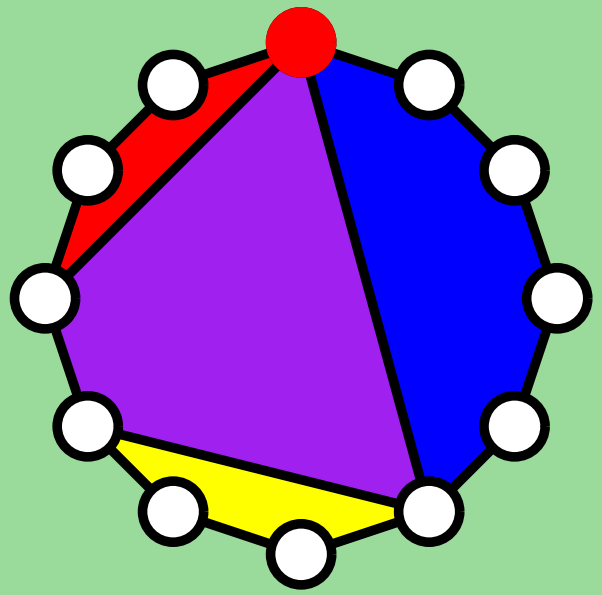
Each face cycle has a closest vertex to the chosen vertex

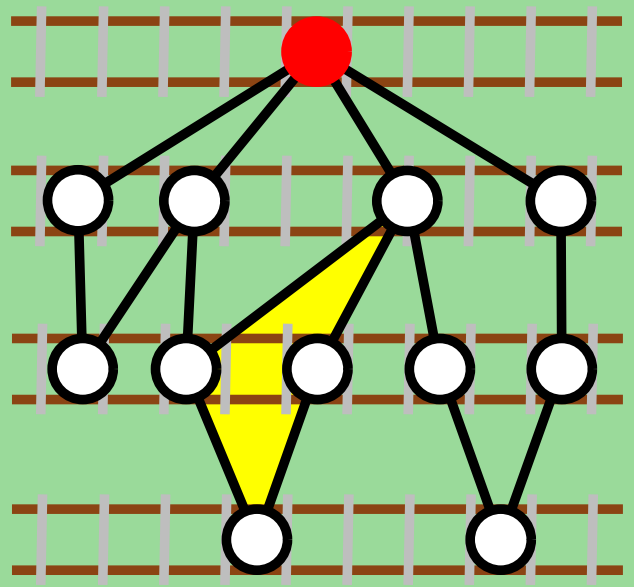
Order vertices of each face cycle independently

Which 3-track graphs can we recognize and layout 3-track?

Bipartite Outerplanar Graphs

Pick a vertex and perform a breadth first layering





How to layout vertices with a level?

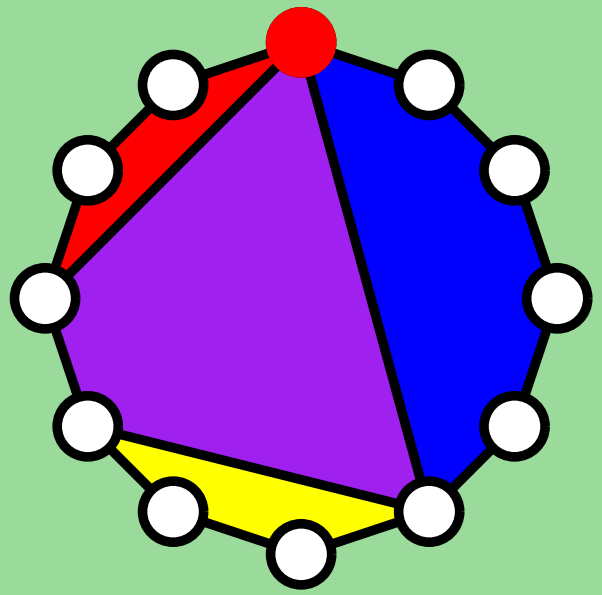
Each face cycle has a closest vertex to the chosen vertex

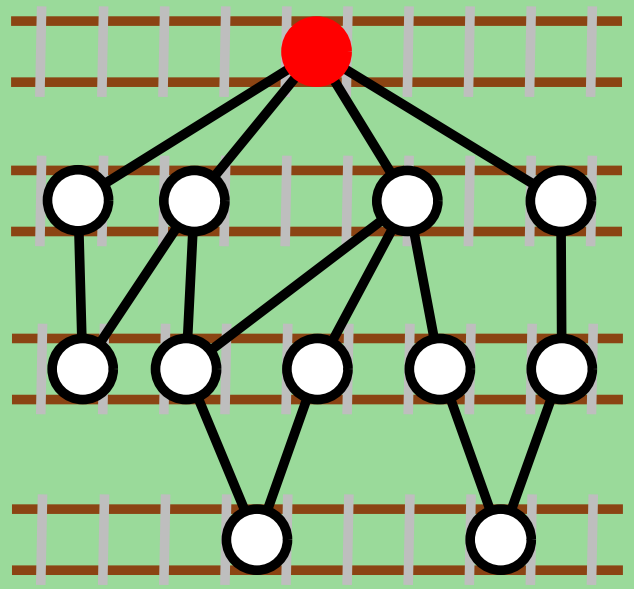
Order vertices of each face cycle independently

Which 3-track graphs can we recognize and layout 3-track?

Bipartite Outerplanar Graphs

Pick a vertex and perform a breadth first layering





How to layout vertices with a level?

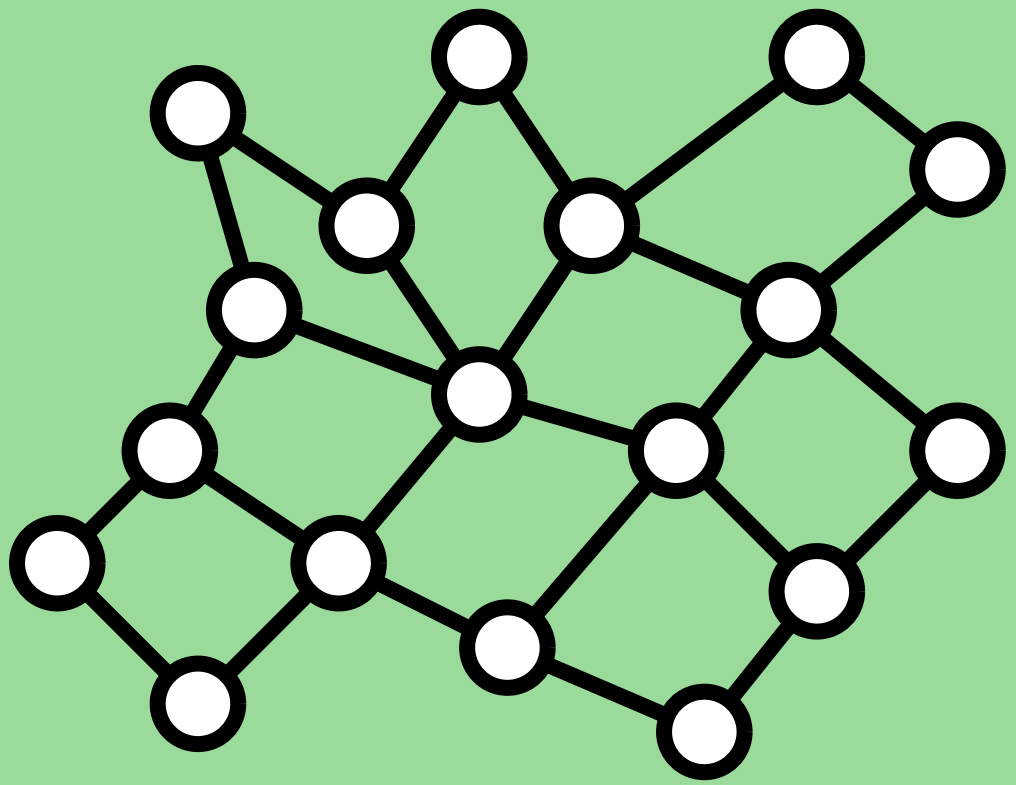
Each face cycle has a closest vertex to the chosen vertex

Order vertices of each face cycle independently

Because every face cycle has a valid layout, the entire graph has a valid layout [Abel et al, 2014]

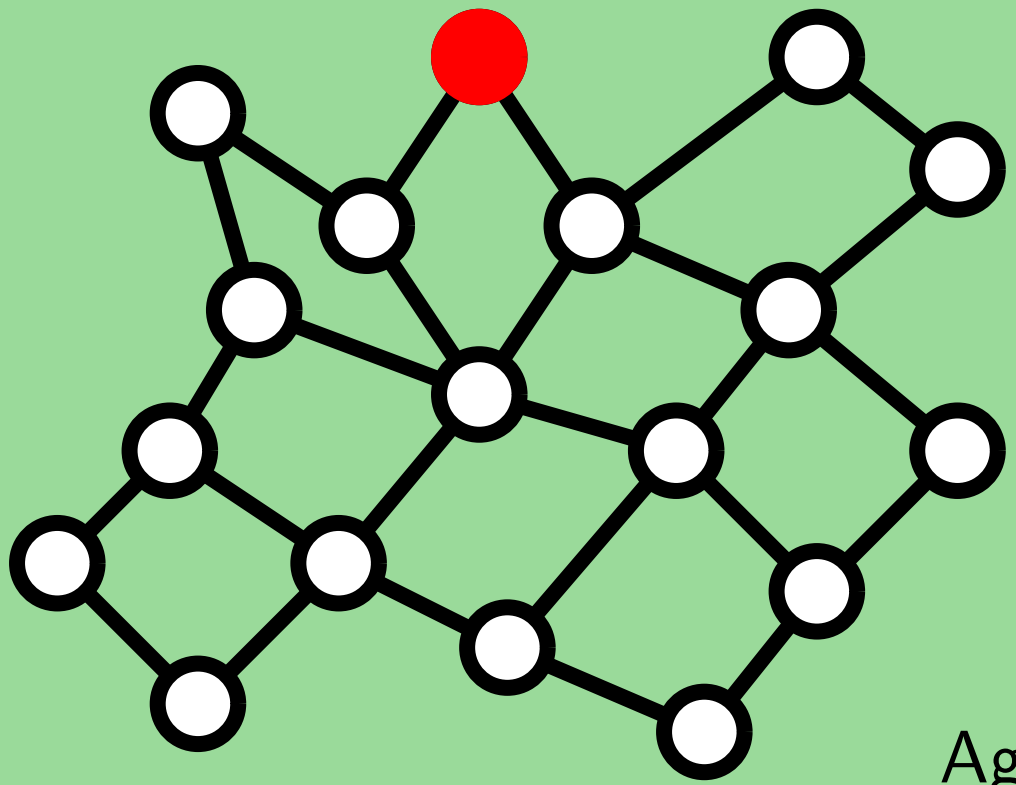
Squaregraphs

Every bounded face is a 4-cycle and every interior vertex has at least 4 neighbors



Squaregraphs

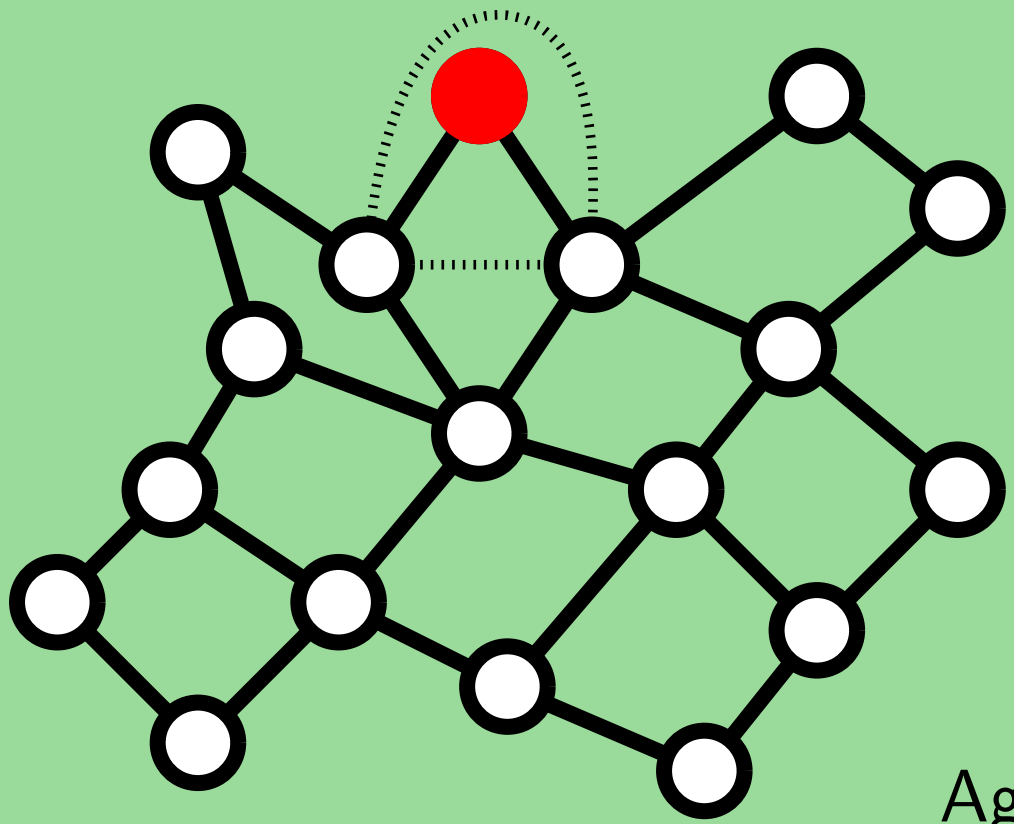
Every bounded face is a 4-cycle and every interior vertex has at least 4 neighbors



Again pick a vertex and perform a breadth first layering

Squaregraphs

Every bounded face is a 4-cycle and every interior vertex has at least 4 neighbors

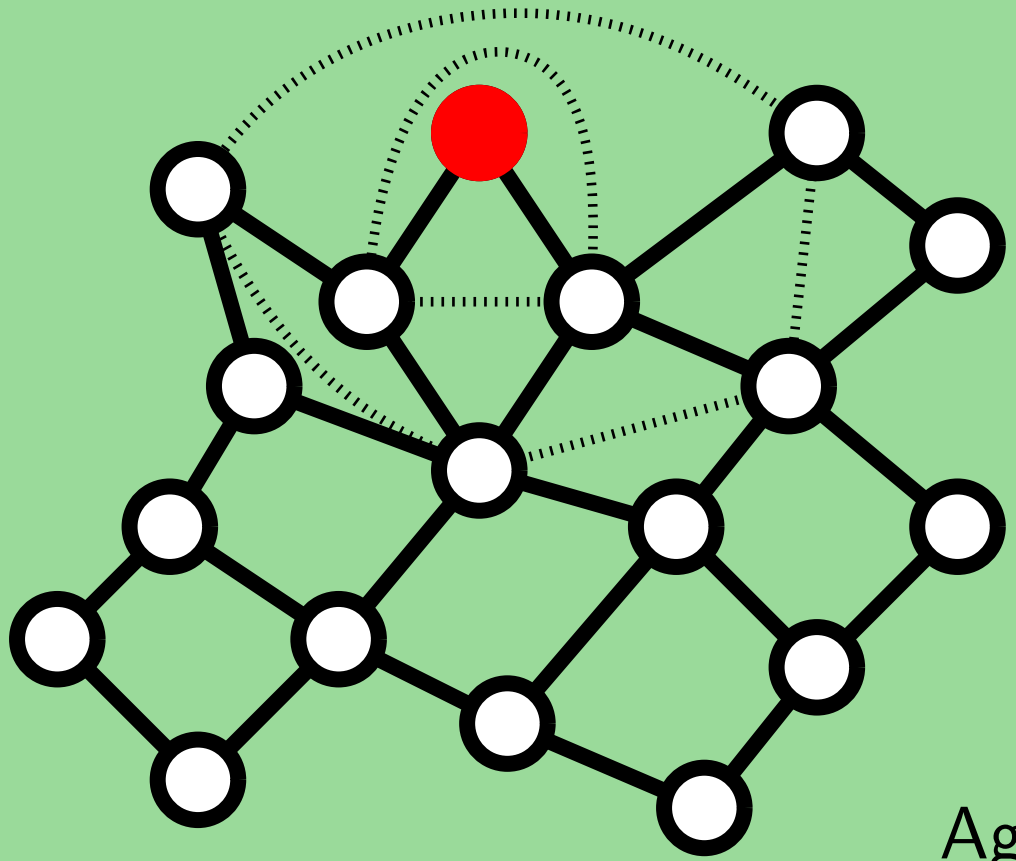


Again pick a vertex and perform a breadth first layering

Use [Bandelt et al, 2010] for ordering layers

Squaregraphs

Every bounded face is a 4-cycle and every interior vertex has at least 4 neighbors

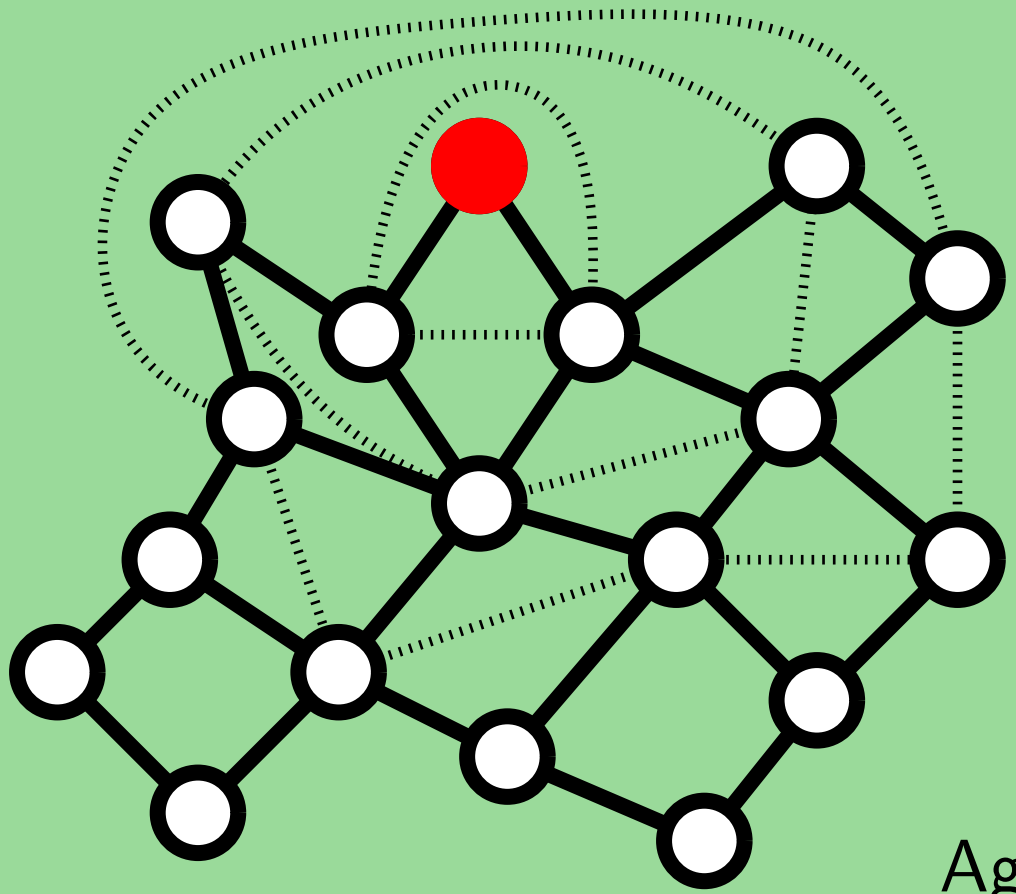


Again pick a vertex and perform a breadth first layering

Use [Bandelt et al, 2010] for ordering layers

Squaregraphs

Every bounded face is a 4-cycle and every interior vertex has at least 4 neighbors

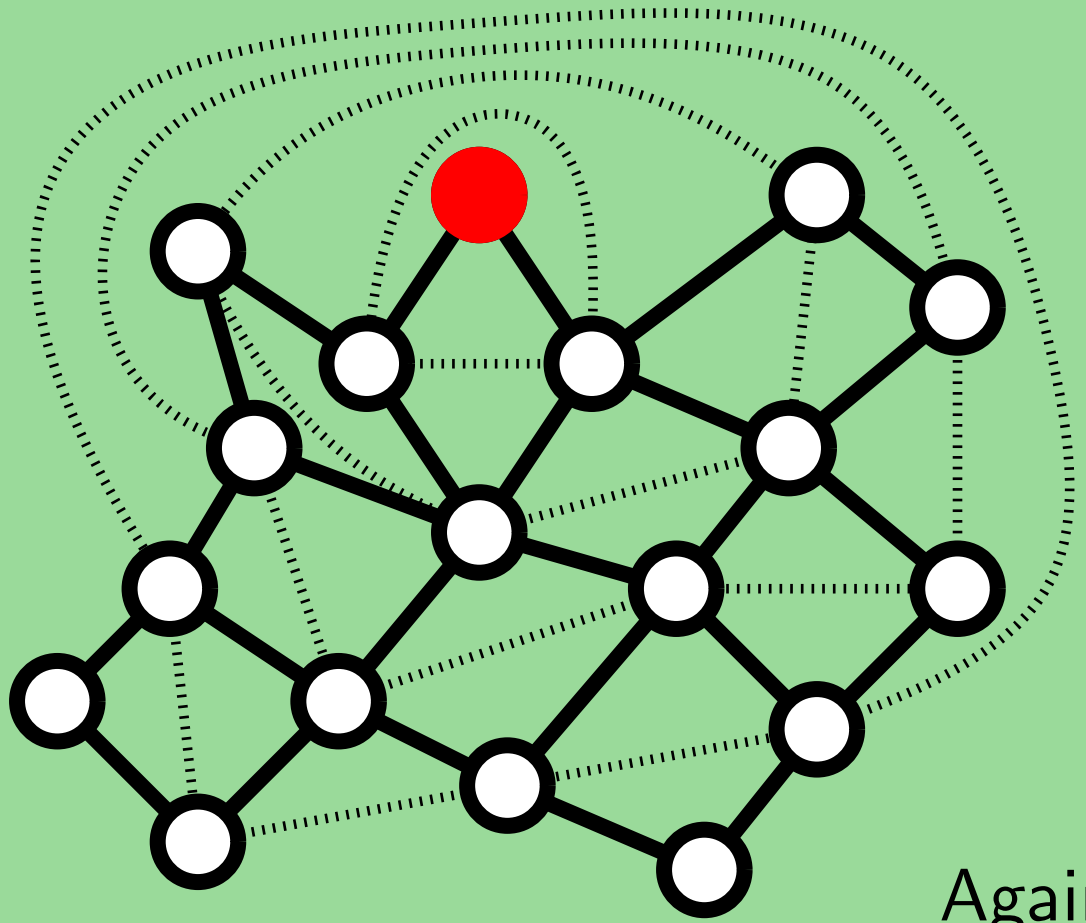


Again pick a vertex and perform a breadth first layering

Use [Bandelt et al, 2010] for ordering layers

Squaregraphs

Every bounded face is a 4-cycle and every interior vertex has at least 4 neighbors

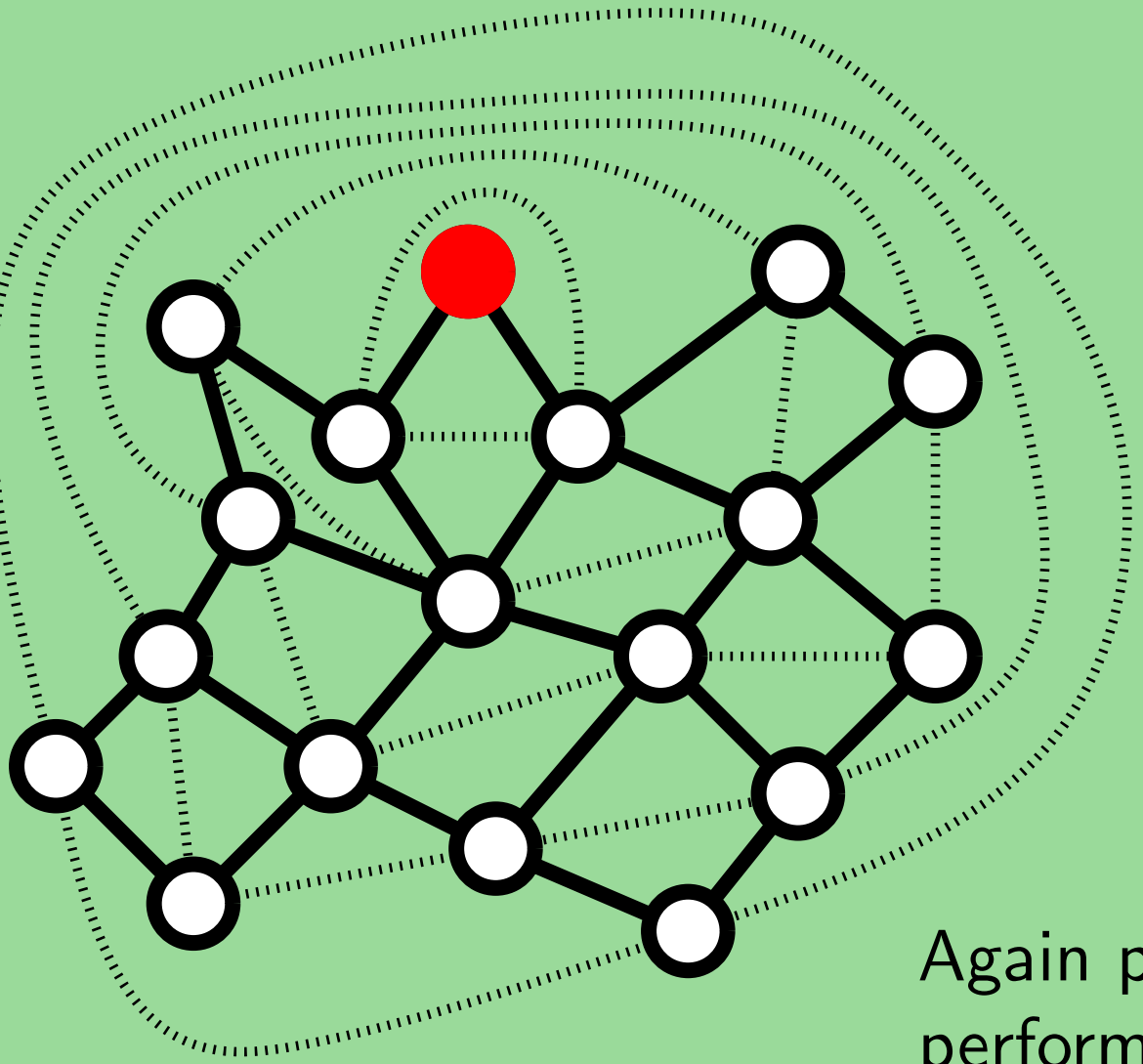


Again pick a vertex and perform a breadth first layering

Use [Bandelt et al, 2010] for ordering layers

Squaregraphs

Every bounded face is a 4-cycle and every interior vertex has at least 4 neighbors

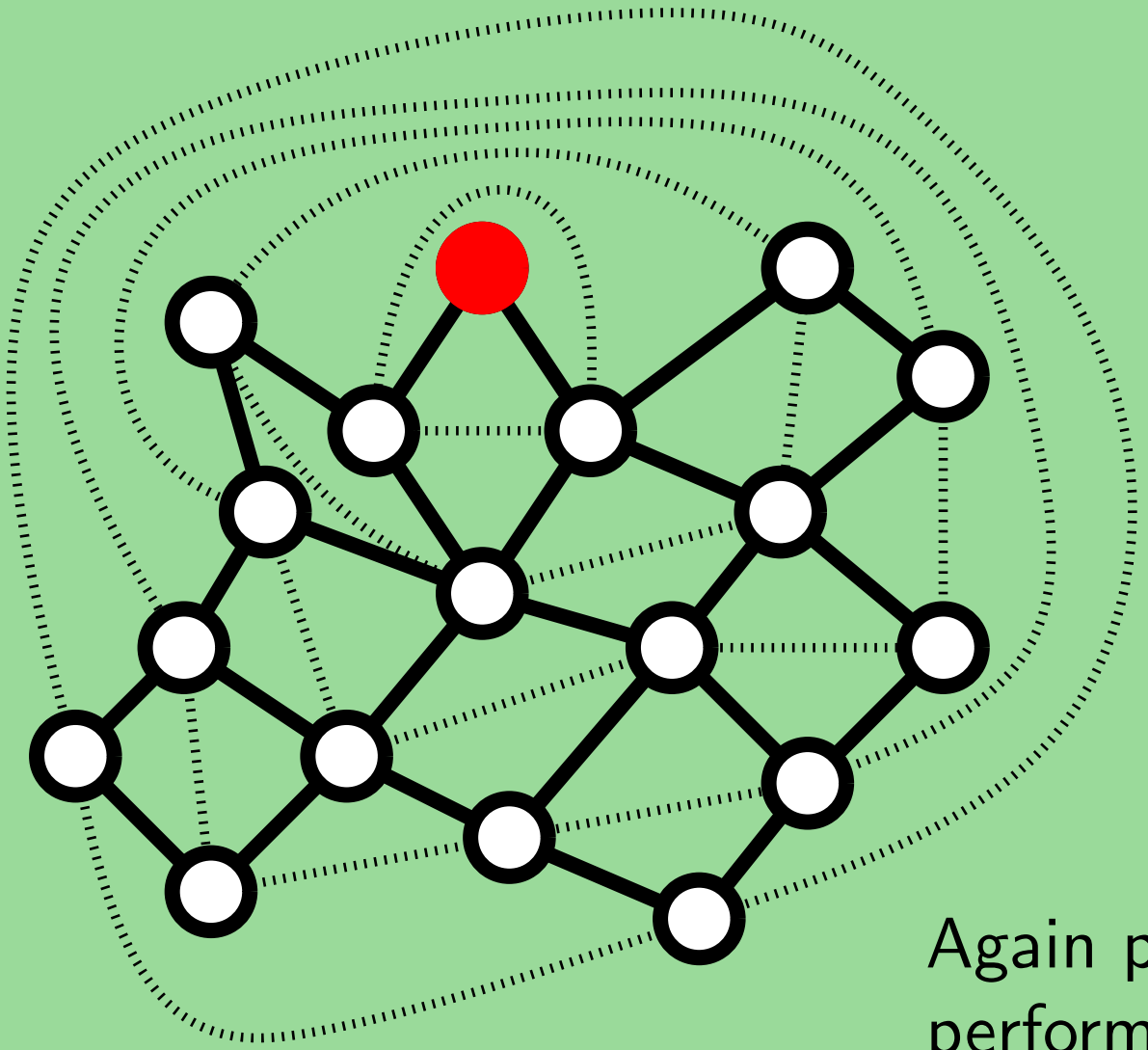


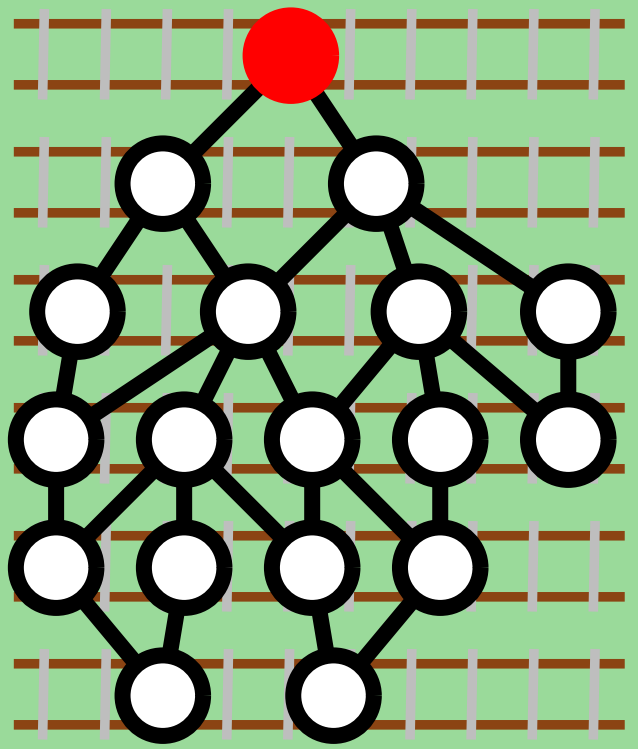
Again pick a vertex and perform a breadth first layering

Use [Bandelt et al, 2010] for ordering layers

Squaregraphs

Every bounded face is a 4-cycle and every interior vertex has at least 4 neighbors



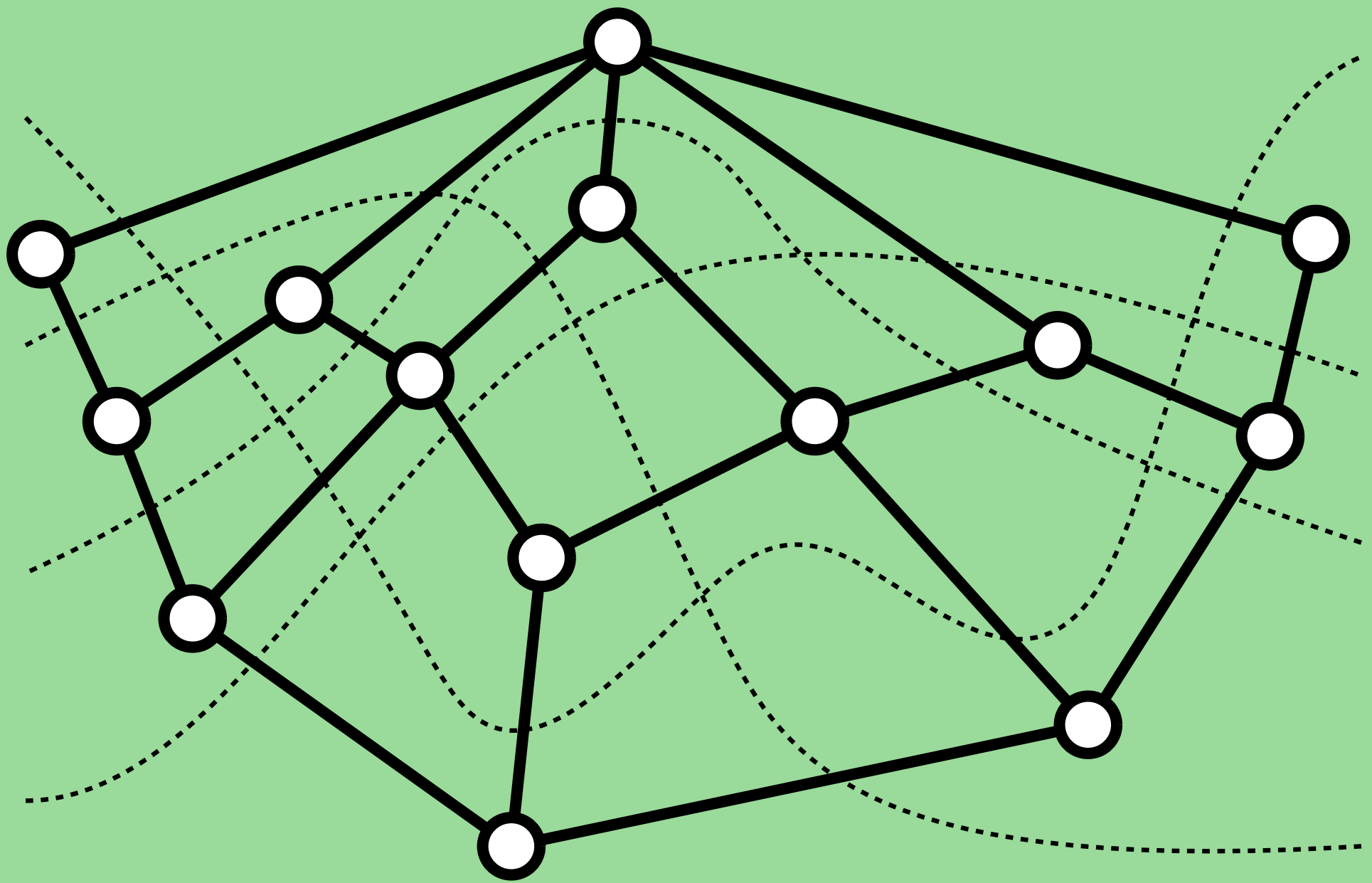


Again pick a vertex and perform a breadth first layering

Use [Bandelt et al, 2010] for ordering layers

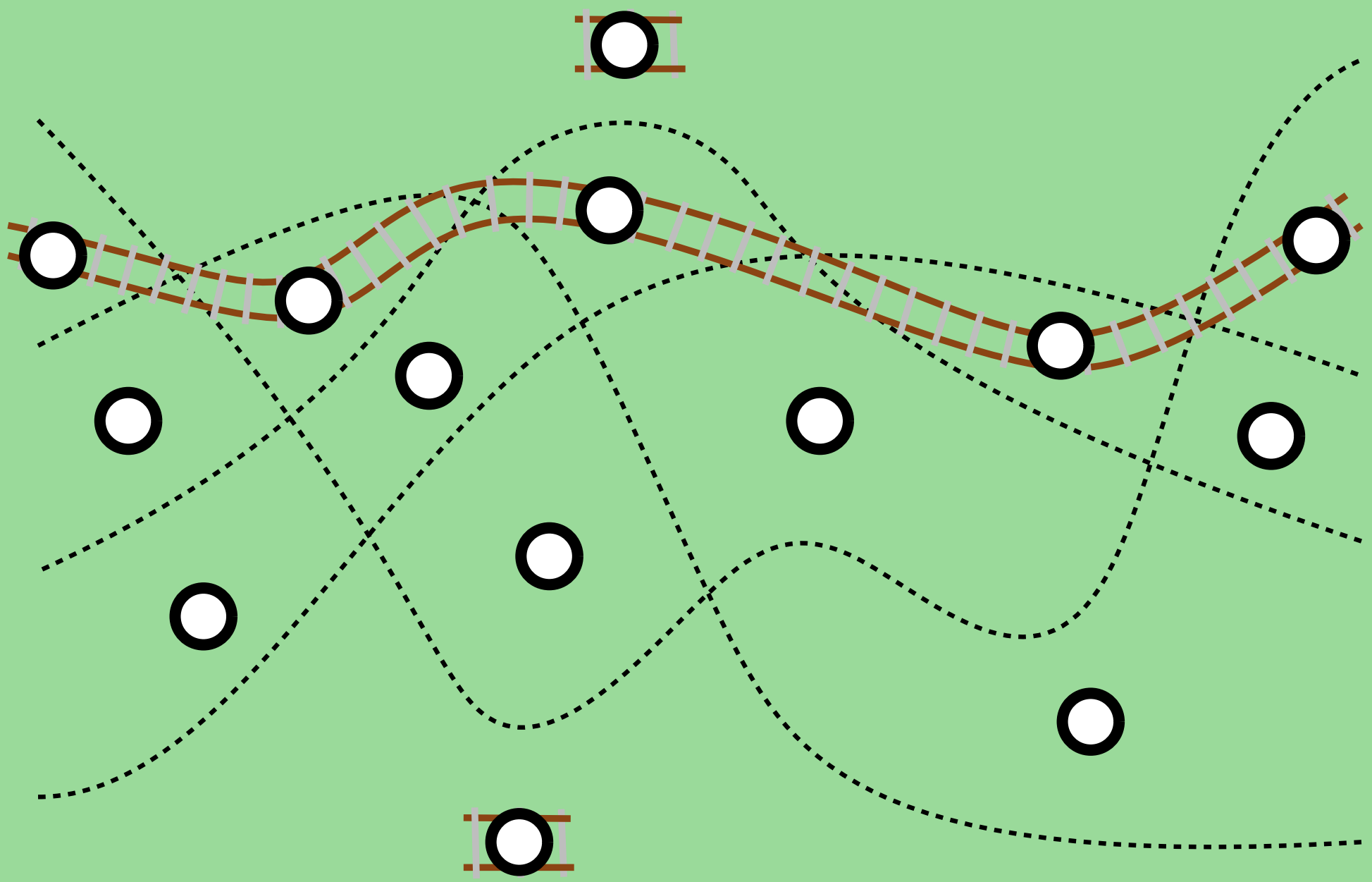
Dual graphs of x-monotone curves

where each curves projection covers the entire x-axis



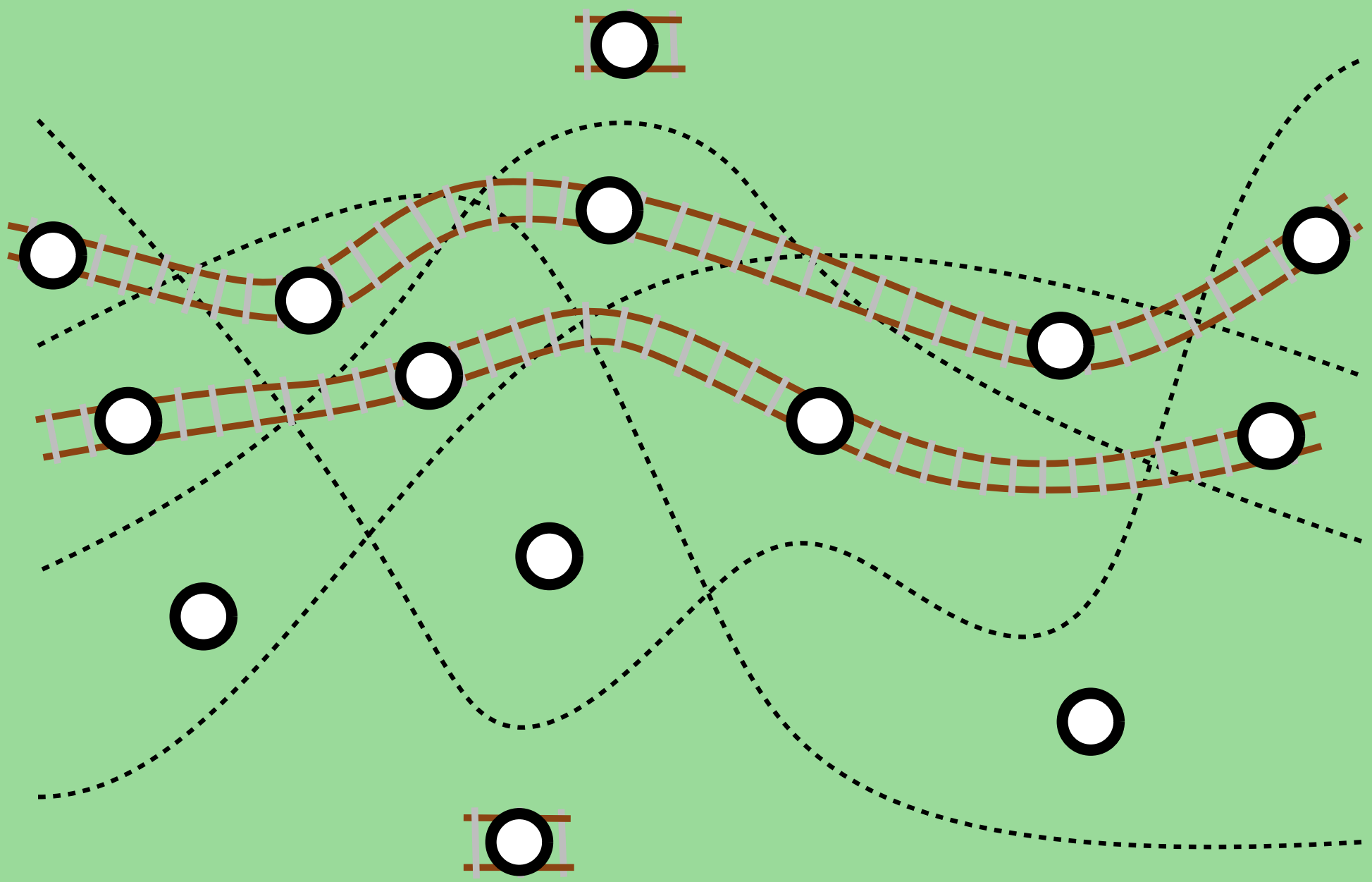
Dual graphs of x-monotone curves

where each curves projection covers the entire x-axis



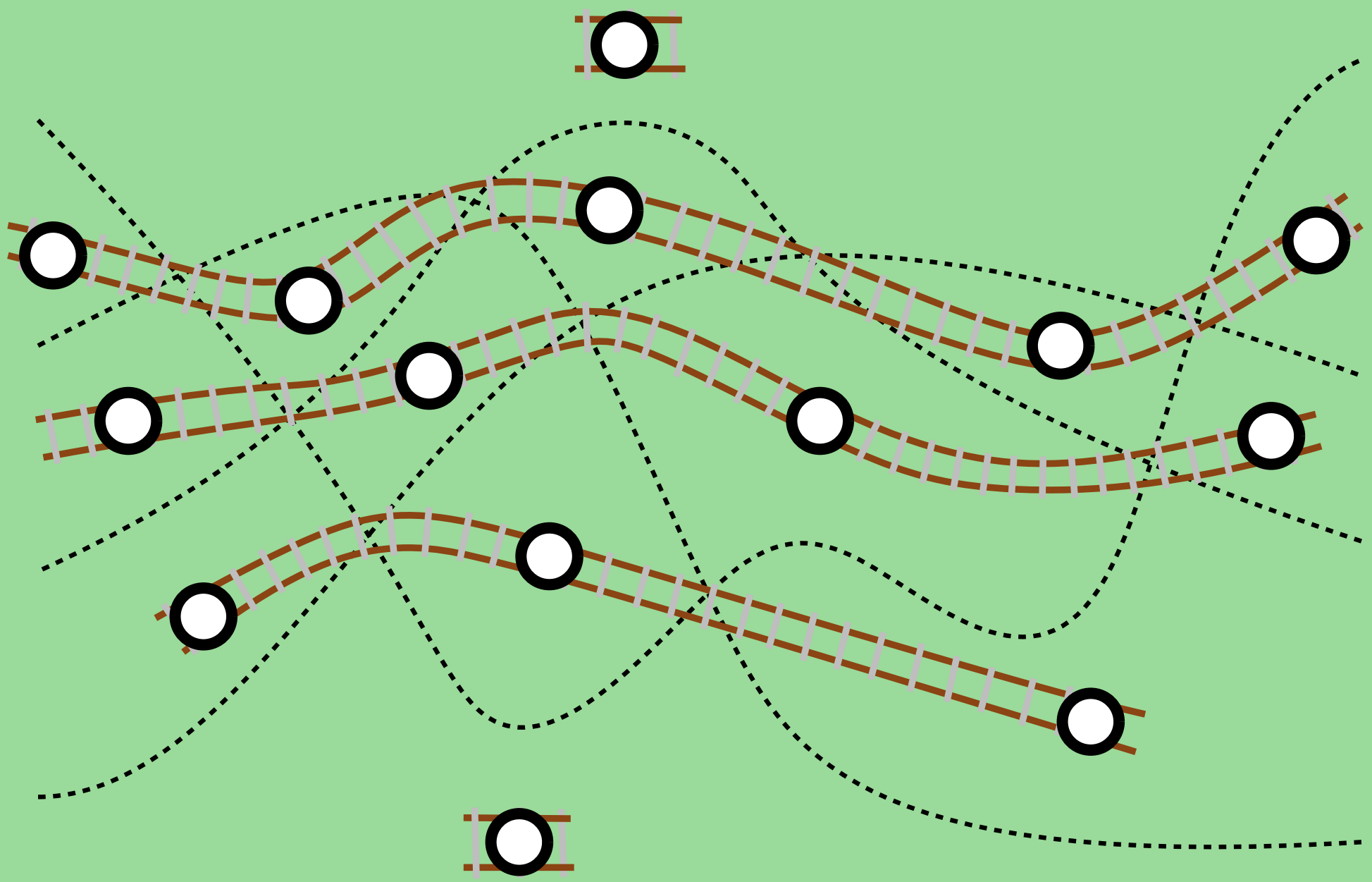
Dual graphs of x-monotone curves

where each curves projection covers the entire x-axis



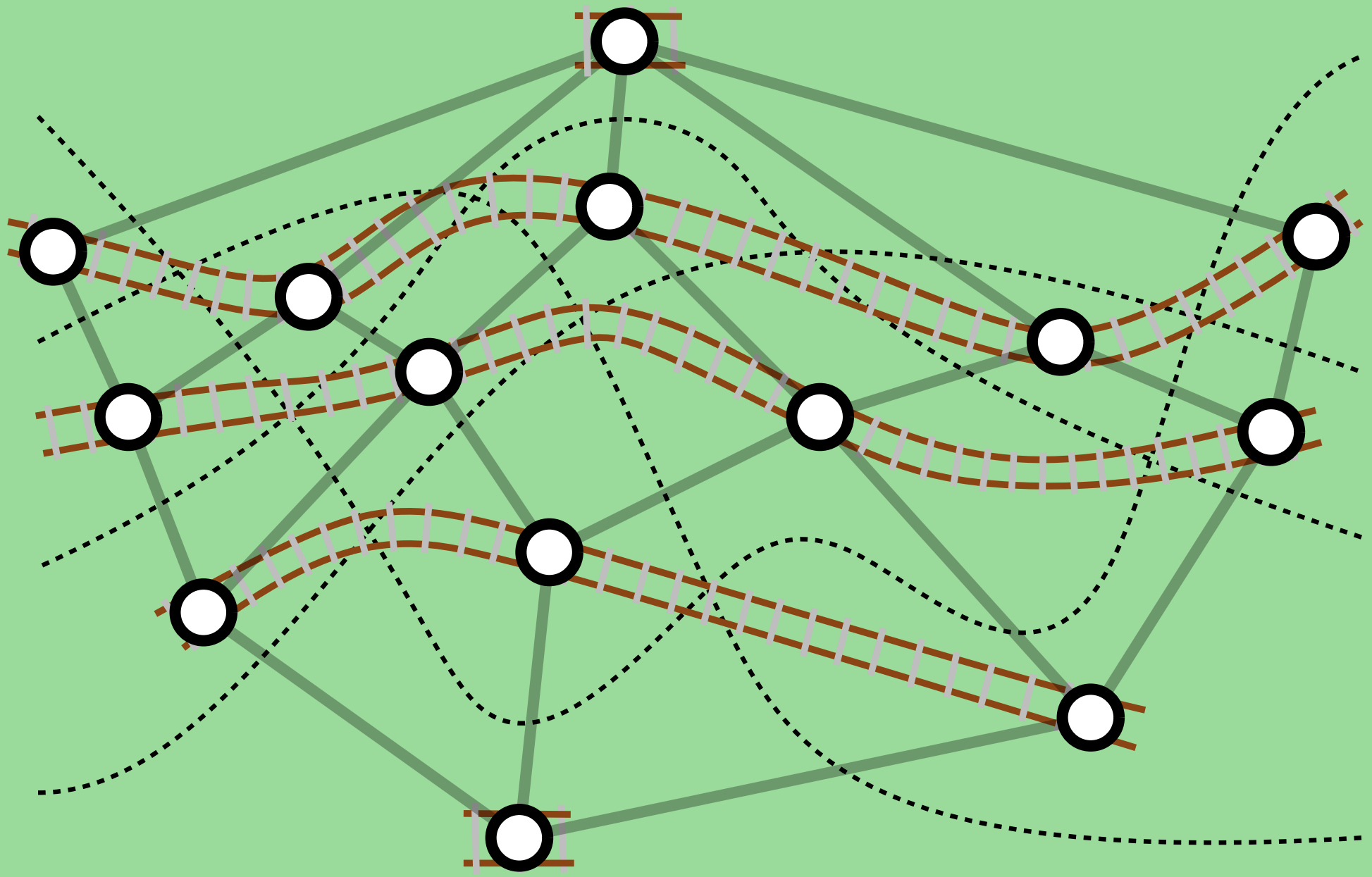
Dual graphs of x-monotone curves

where each curves projection covers the entire x-axis



Dual graphs of x-monotone curves

where each curves projection covers the entire x-axis



Summary

Recognizing 3-track graphs is NP-complete

-holds for larger track number

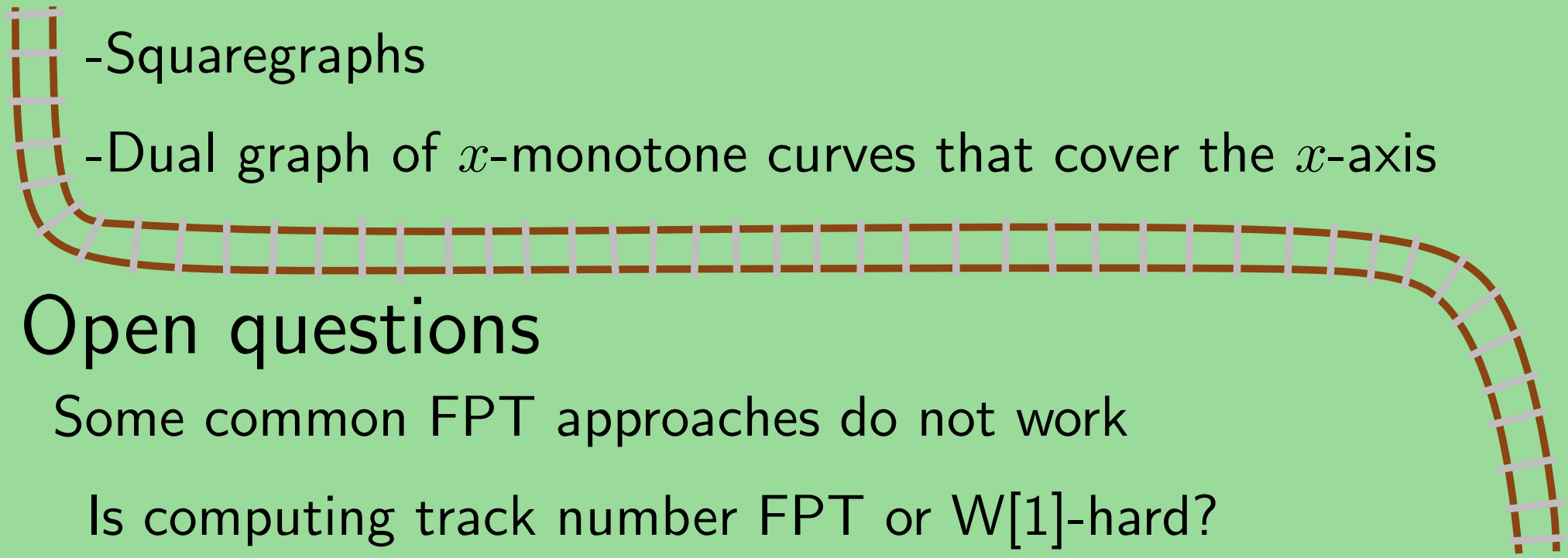
Computing the track number for a graph is NP-hard

3-track layouts for special graph classes

-Bipartite outerplanar graphs

-Squaregraphs

-Dual graph of x-monotone curves that cover the x-axis

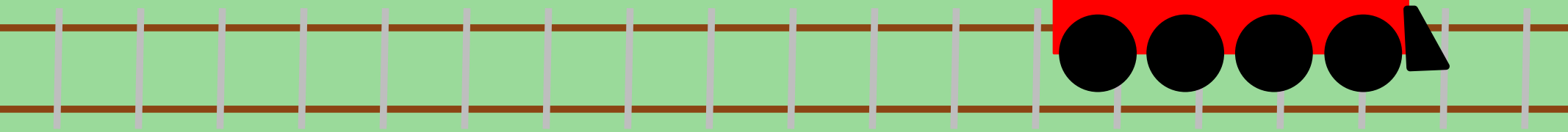


Open questions

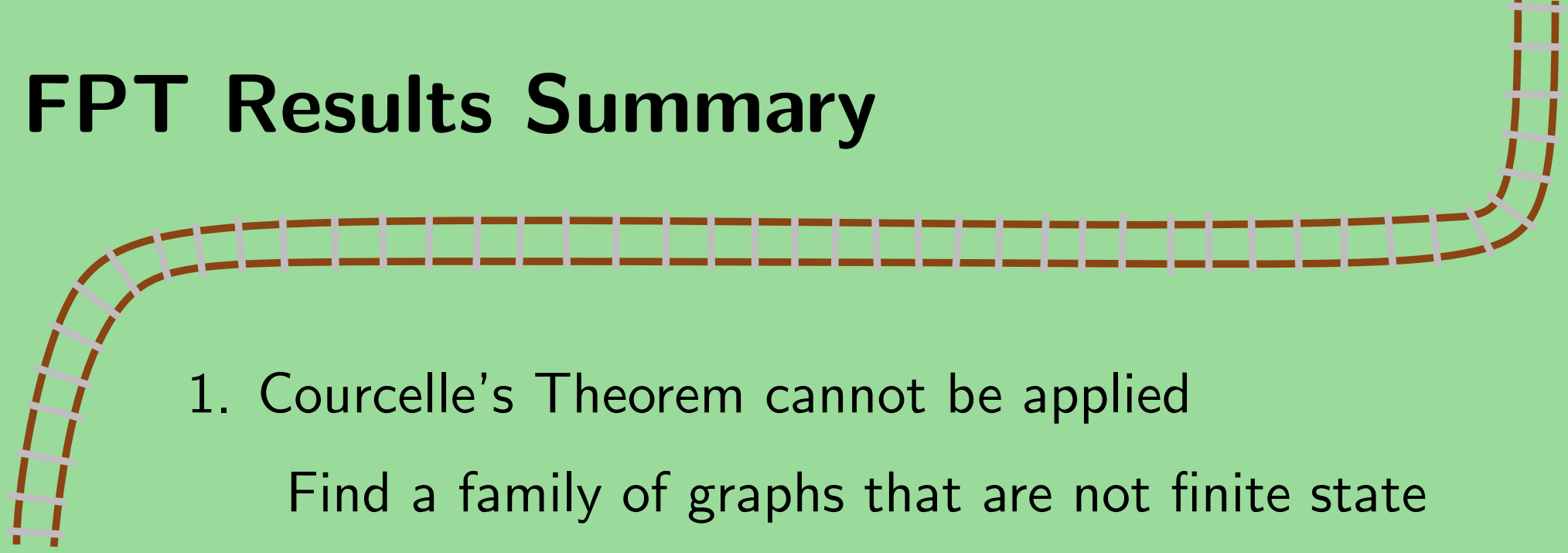
Some common FPT approaches do not work

Is computing track number FPT or W[1]-hard?

Thank you!



FPT Results Summary



1. Courcelle's Theorem cannot be applied

   Find a family of graphs that are not finite state

2. 2-core kernelization for k-almost-tree will not work

   Found pairs of graphs with the same 2-core and different track number

3. Track number is non-uniformly FPT in tree depth

   Using forbidden subgraph testing