Approximation Algorithms

Lecture 9: An Approximation Scheme for EuclideanTSP

Part I: TravelingSalesmanProblem

Joachim Spoerhase

Winter 2021/22

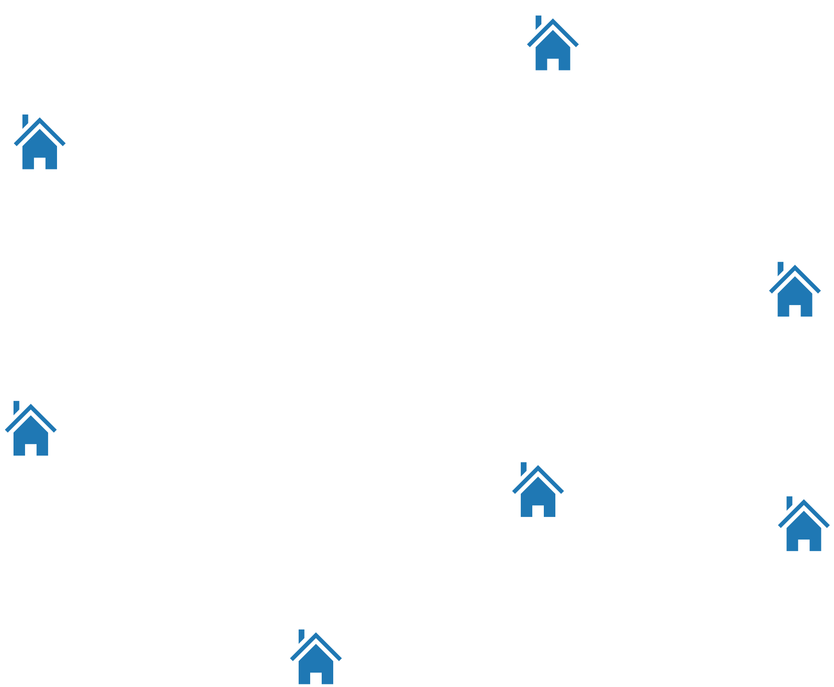
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

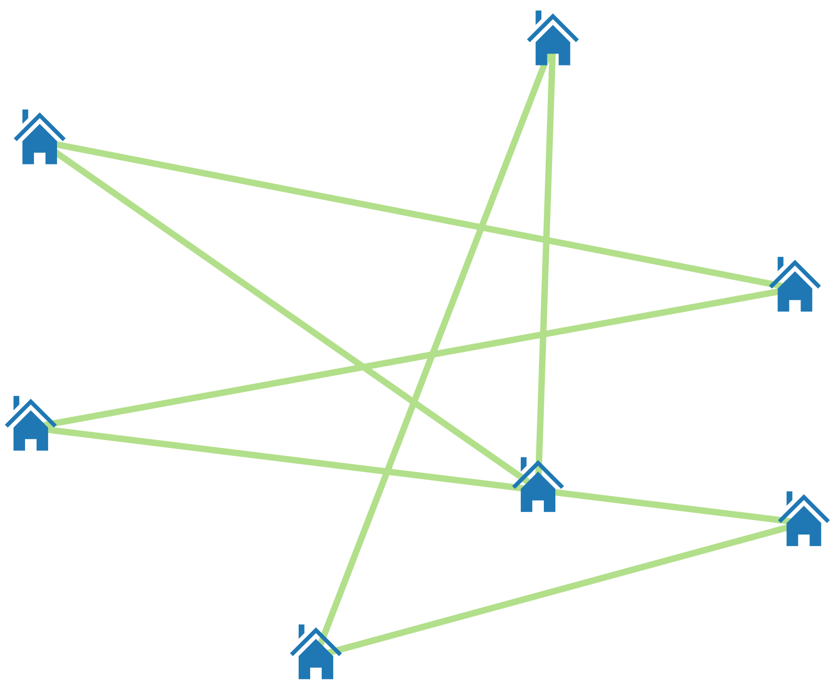


TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle)

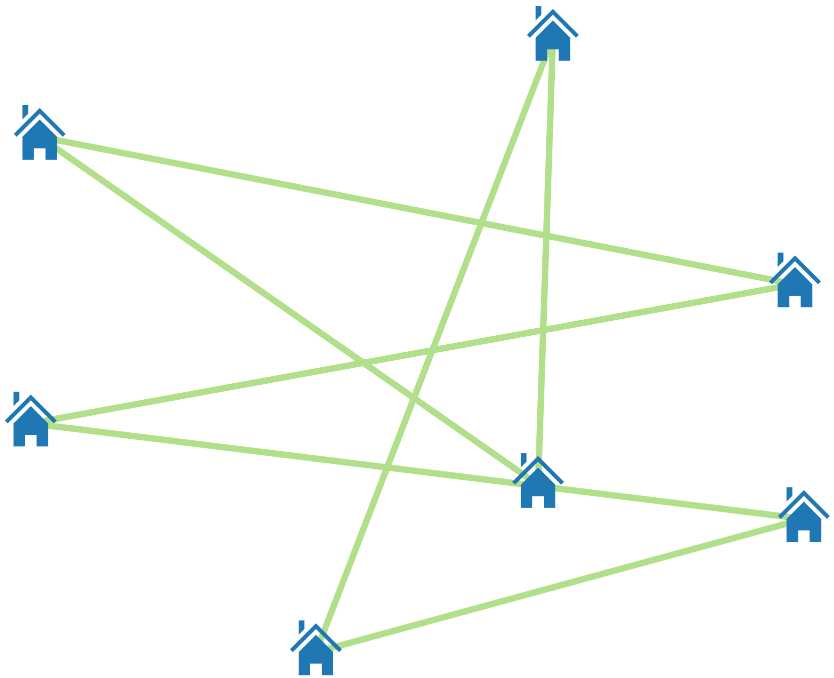


TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle)

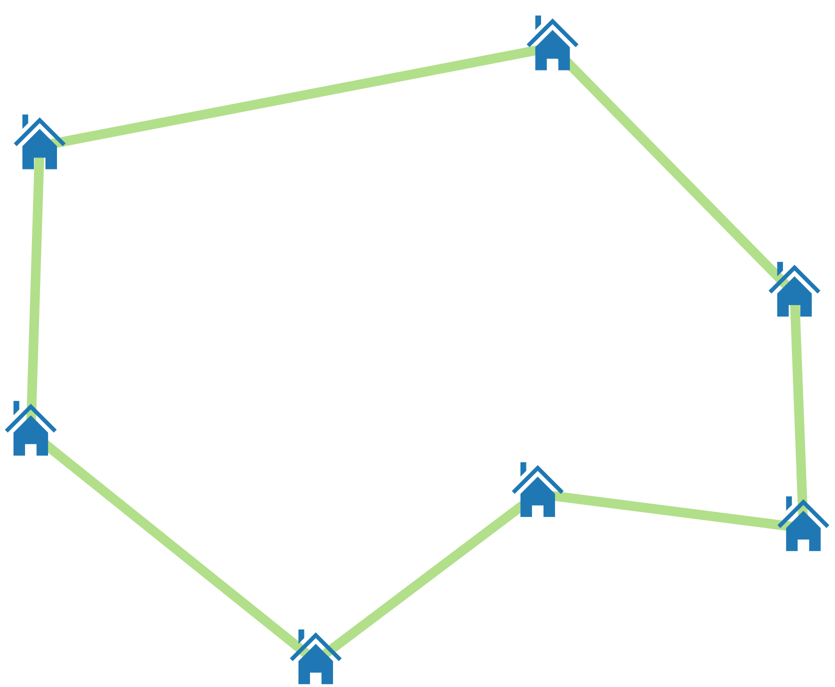


TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

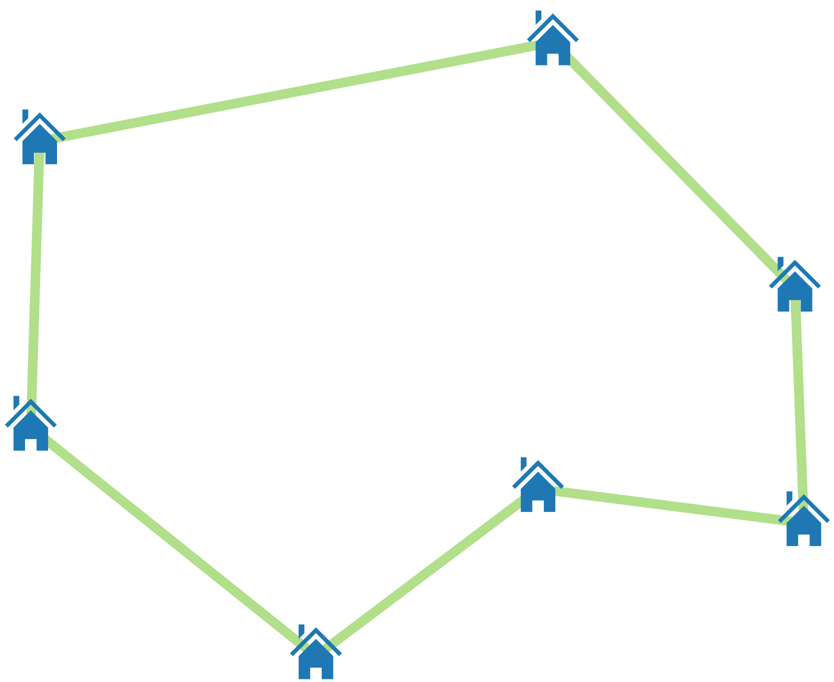


TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.



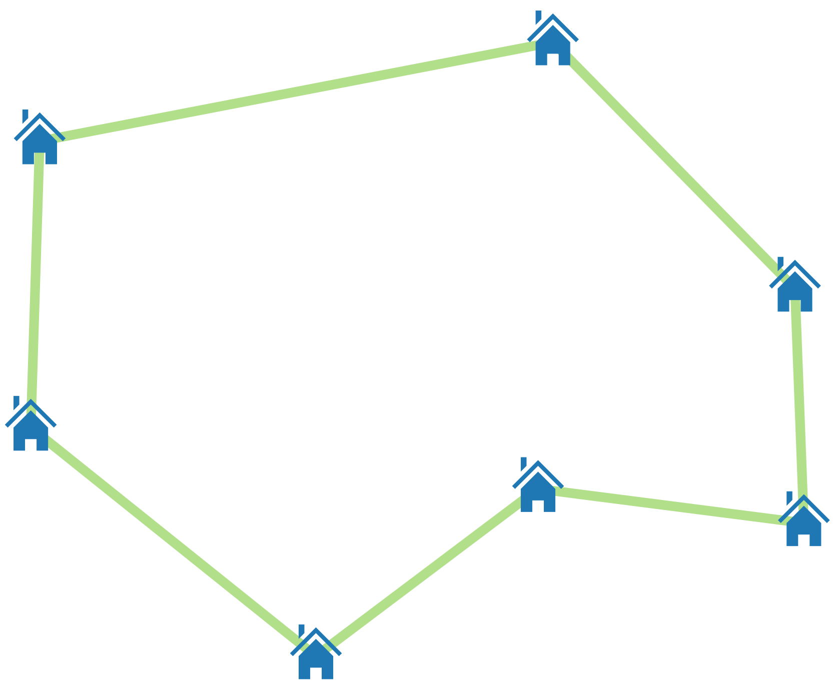
For every polynomial $p(n)$, TSP cannot be approximated within factor (unless P=NP).

TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.



For every polynomial $p(n)$, TSP cannot be approximated within factor $2^{p(n)}$ (unless P=NP).

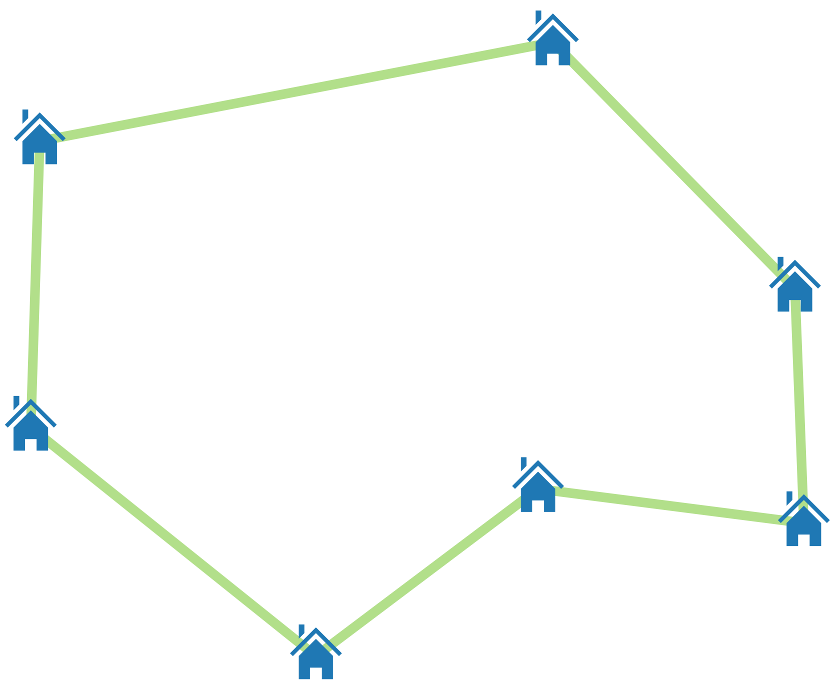
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

Distance between two points?



For every polynomial $p(n)$, TSP cannot be approximated within factor $2^{p(n)}$ (unless P=NP).

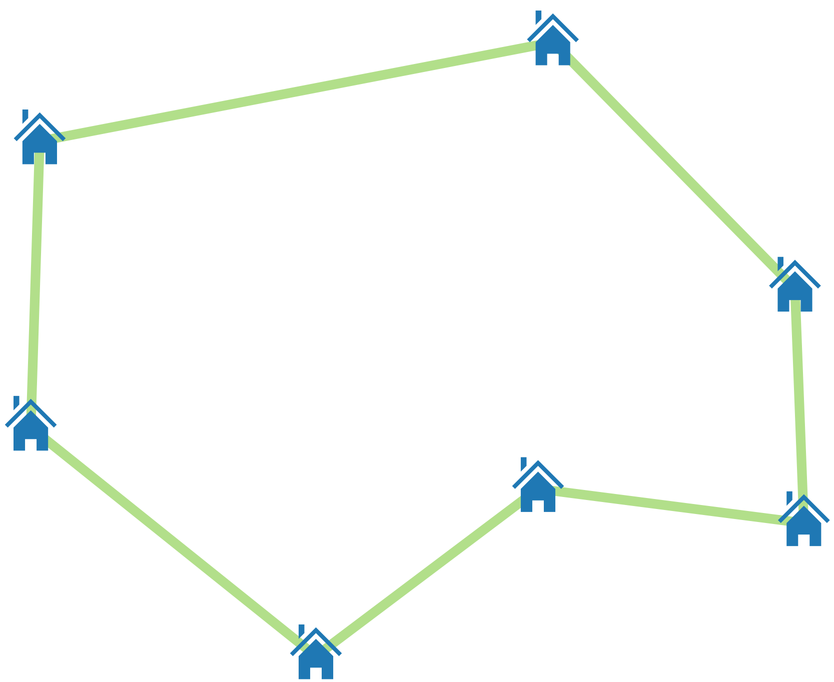
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

Distance between two points?



For every polynomial $p(n)$, TSP cannot be approximated within factor $2^{p(n)}$ (unless P=NP).

There is a $3/2$-approximation algorithm for MetricTSP.

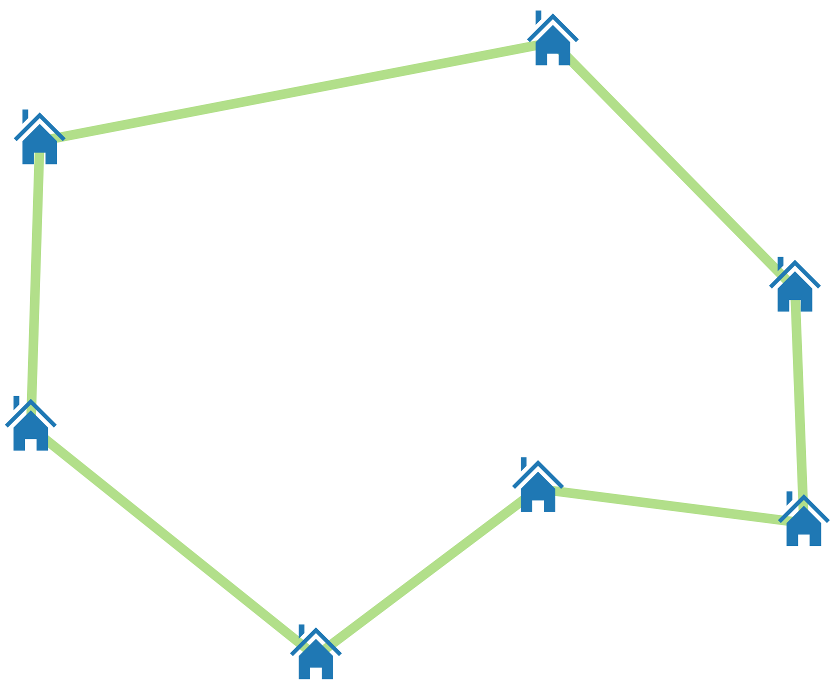
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

Distance between two points?



For every polynomial $p(n)$, TSP cannot be approximated within factor $2^{p(n)}$ (unless P=NP).

There is a $3/2$-approximation algorithm for MetricTSP.

MetricTSP cannot be approximated within factor $123/122$ (unless P=NP).

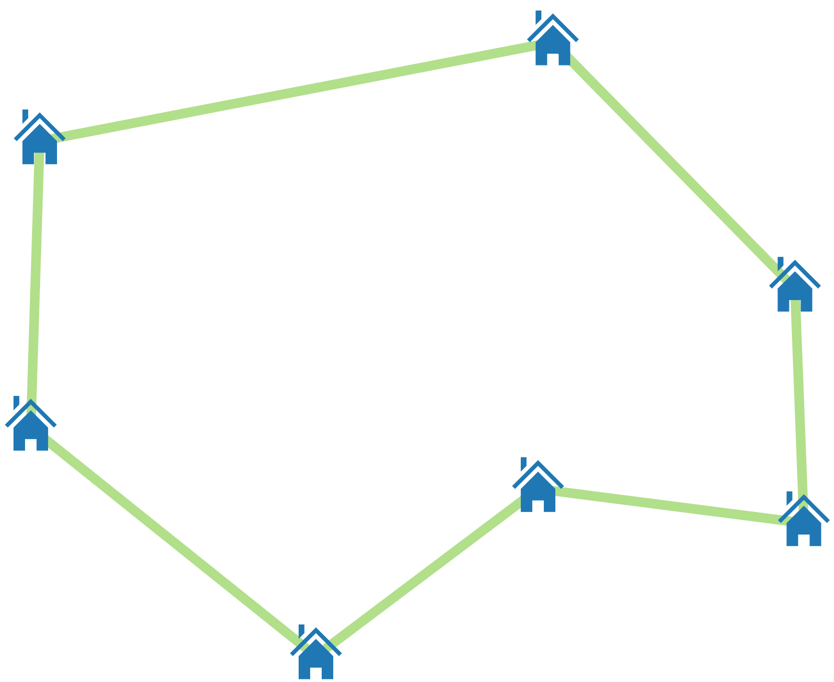
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



For every polynomial $p(n)$, TSP cannot be approximated within factor $2^{p(n)}$ (unless P=NP).

There is a $3/2$-approximation algorithm for MetricTSP.

MetricTSP cannot be approximated within factor $123/122$ (unless P=NP).

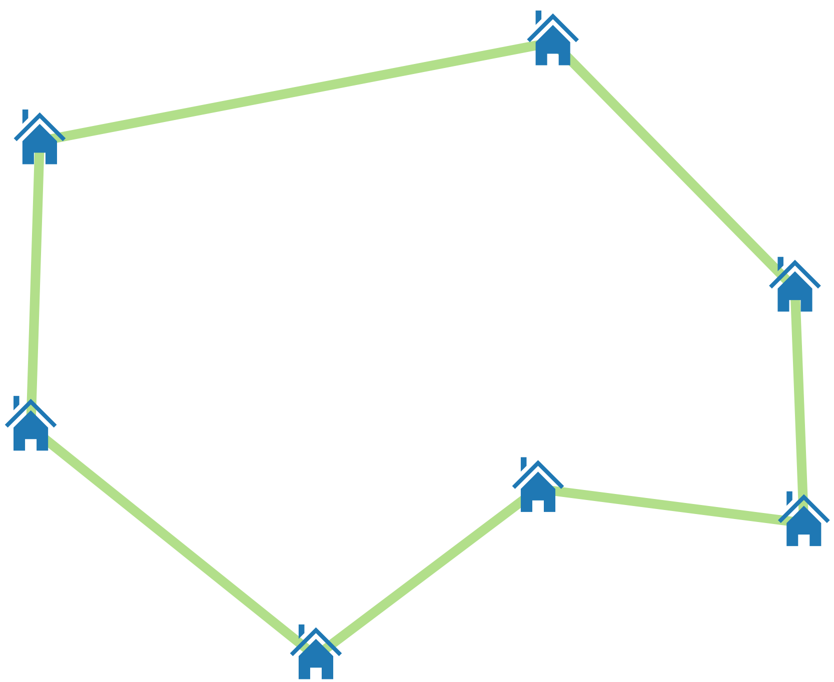
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

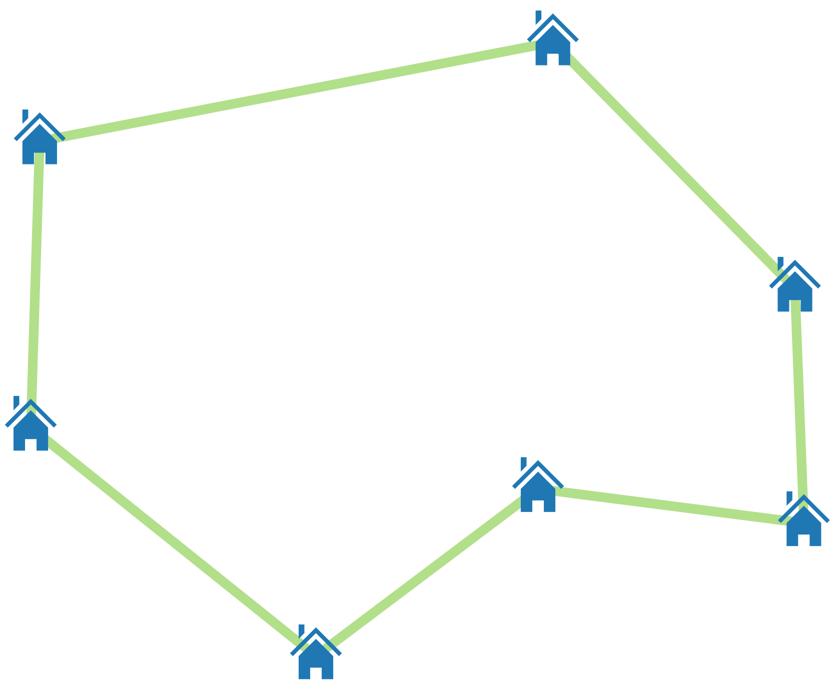
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

- Houses inside $(L \times L)$-square

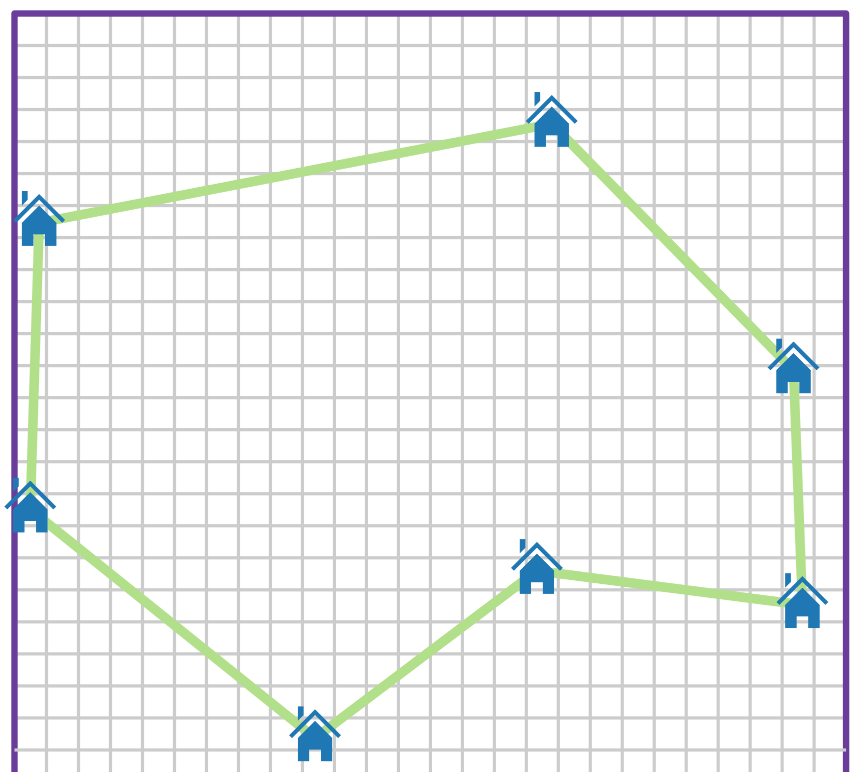
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

- Houses inside $(L \times L)$-square

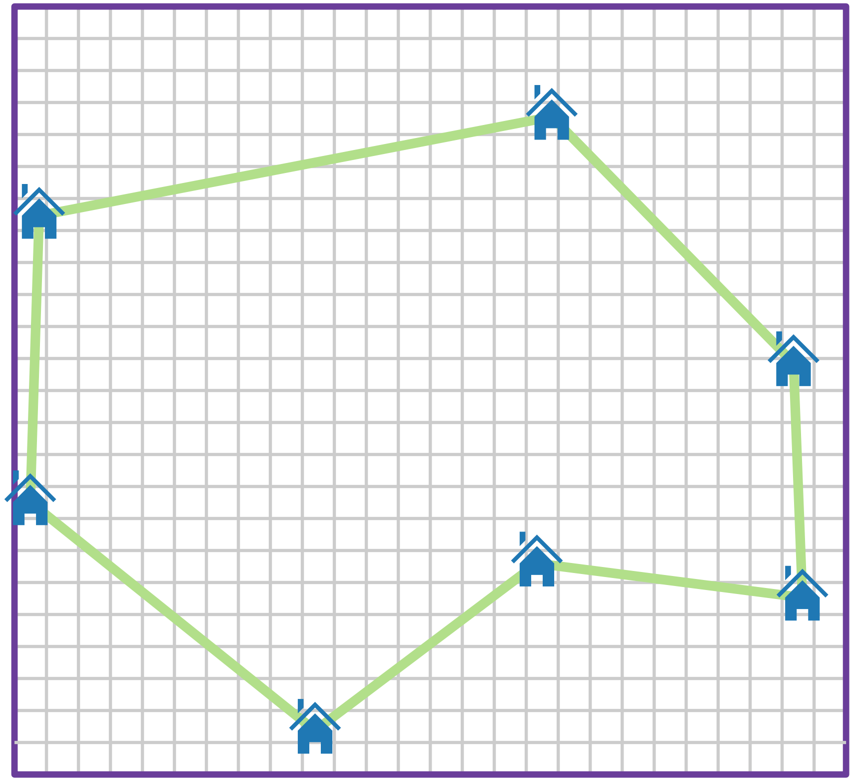
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

- Houses inside $(L \times L)$-square
- $L := 4n^2$

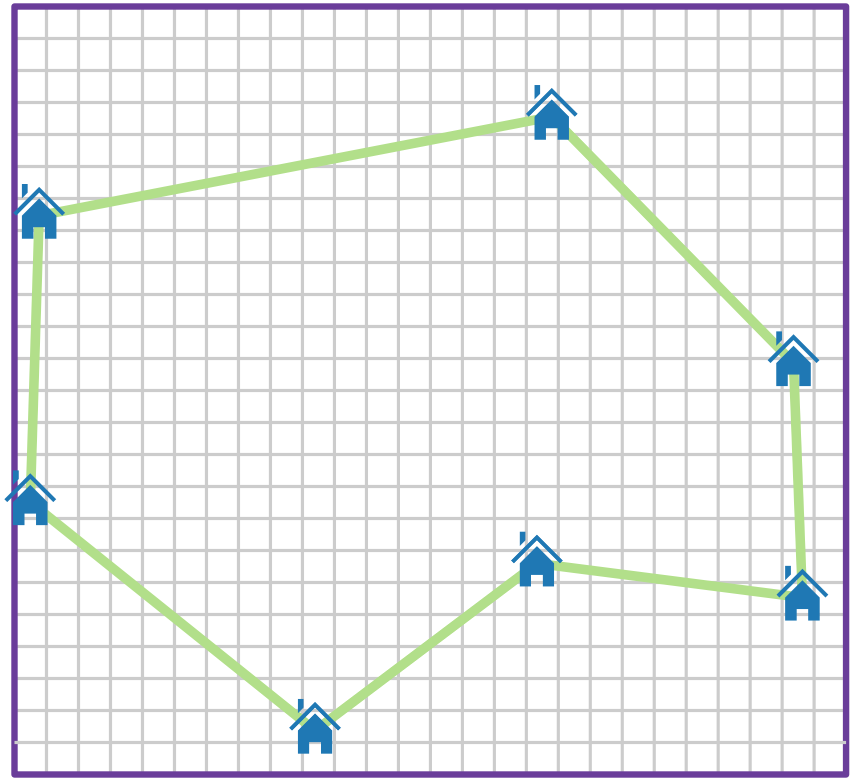
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

- Houses inside $(L \times L)$-square
- $L := 4n^2 = 2^k$;

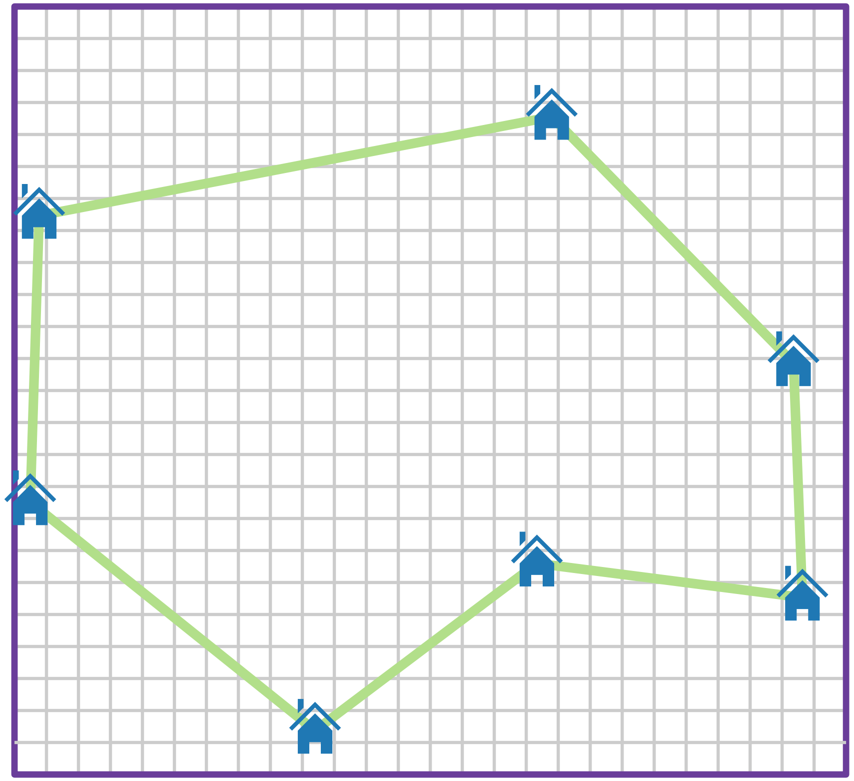
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

- Houses inside $(L \times L)$-square
- $L := 4n^2 = 2^k$; $k = 2 + 2\log_2 n$

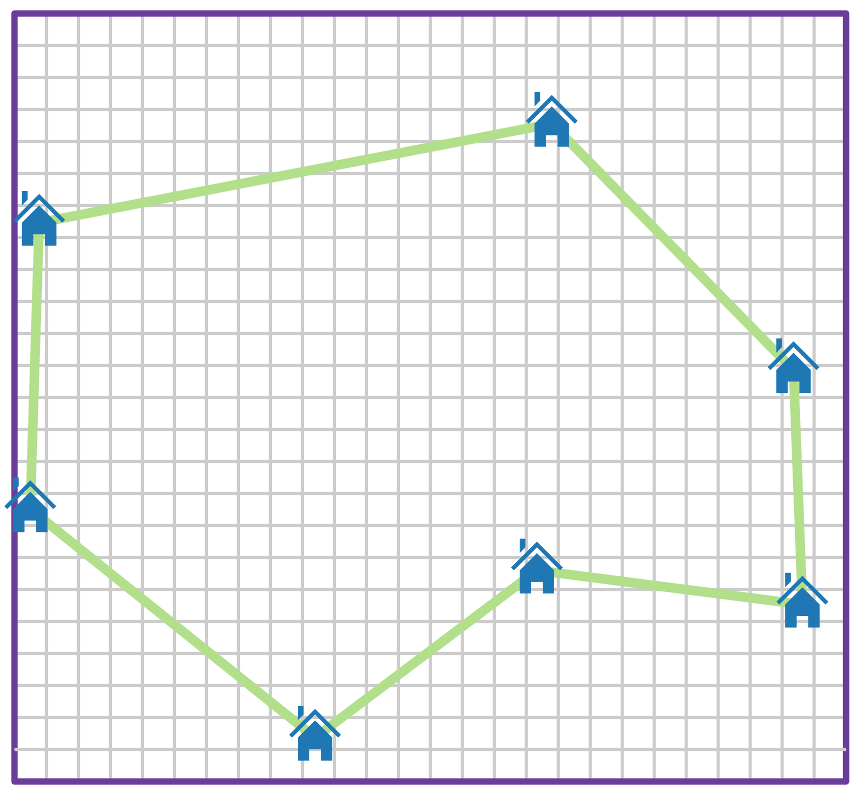
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

- Houses inside $(L \times L)$-square
- $L := 4n^2 = 2^k$; $k = 2 + 2\log_2 n$
- integer coordinates

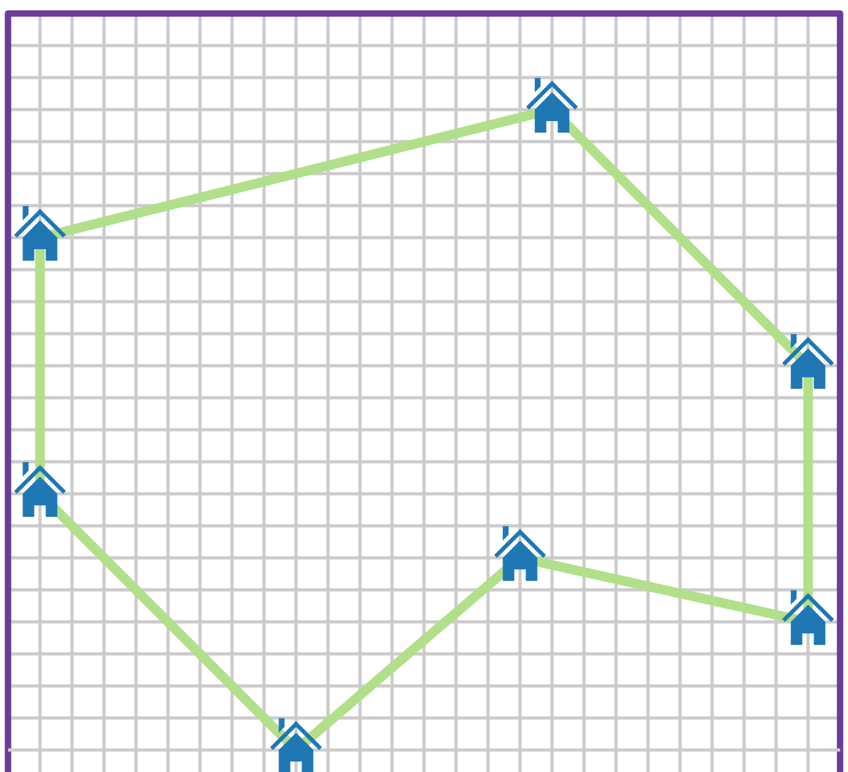
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

- Houses inside $(L \times L)$-square
- $L := 4n^2 = 2^k$; $k = 2 + 2\log_2 n$
- integer coordinates

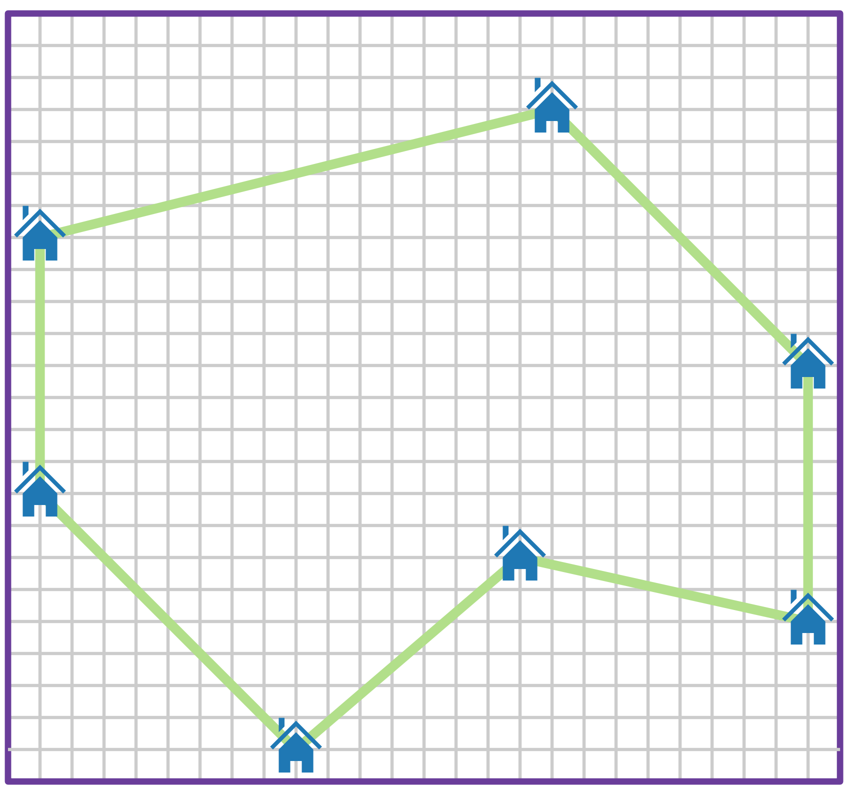
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

- Houses inside $(L \times L)$-square
- $L := 4n^2 = 2^k$; $k = 2 + 2\log_2 n$
- integer coordinates

("justification": homework)

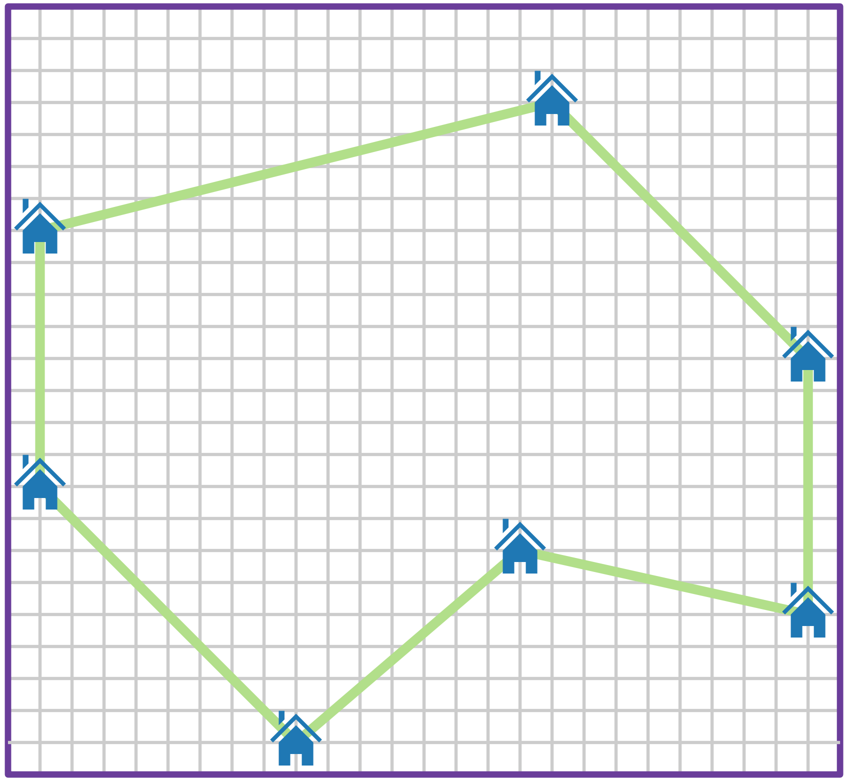
TravelingSalesmanProblem (TSP)

Question: What's the fastest way to deliver all parcels to their destination?

Given: A set of n houses (points) in $\mathbb{R}^2$.

Task: Find a tour (Hamiltonian cycle) of min. length.

The Salesman can fly $\Rightarrow$ Euclidean distance.



Simplifying Assumptions

- Houses inside $(L \times L)$-square
- $L := 4n^2 = 2^k$; $k = 2 + 2\log_2 n$
- integer coordinates

("justification": homework)

Goal: $(1+\varepsilon)$-approximation!

Approximation Algorithms

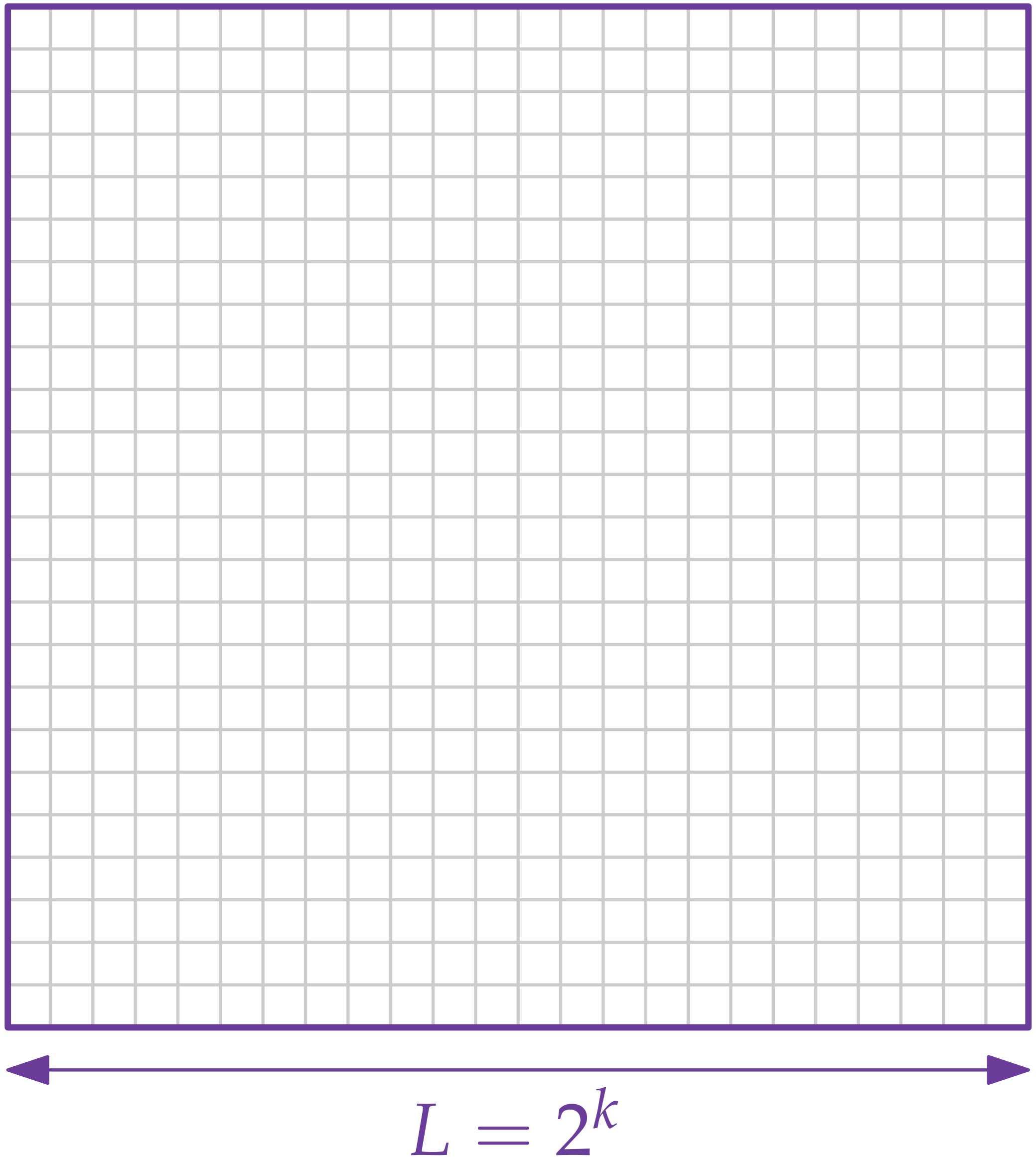
Lecture 9:
PTAS for EuclideanTSP

Part II:
Dissection

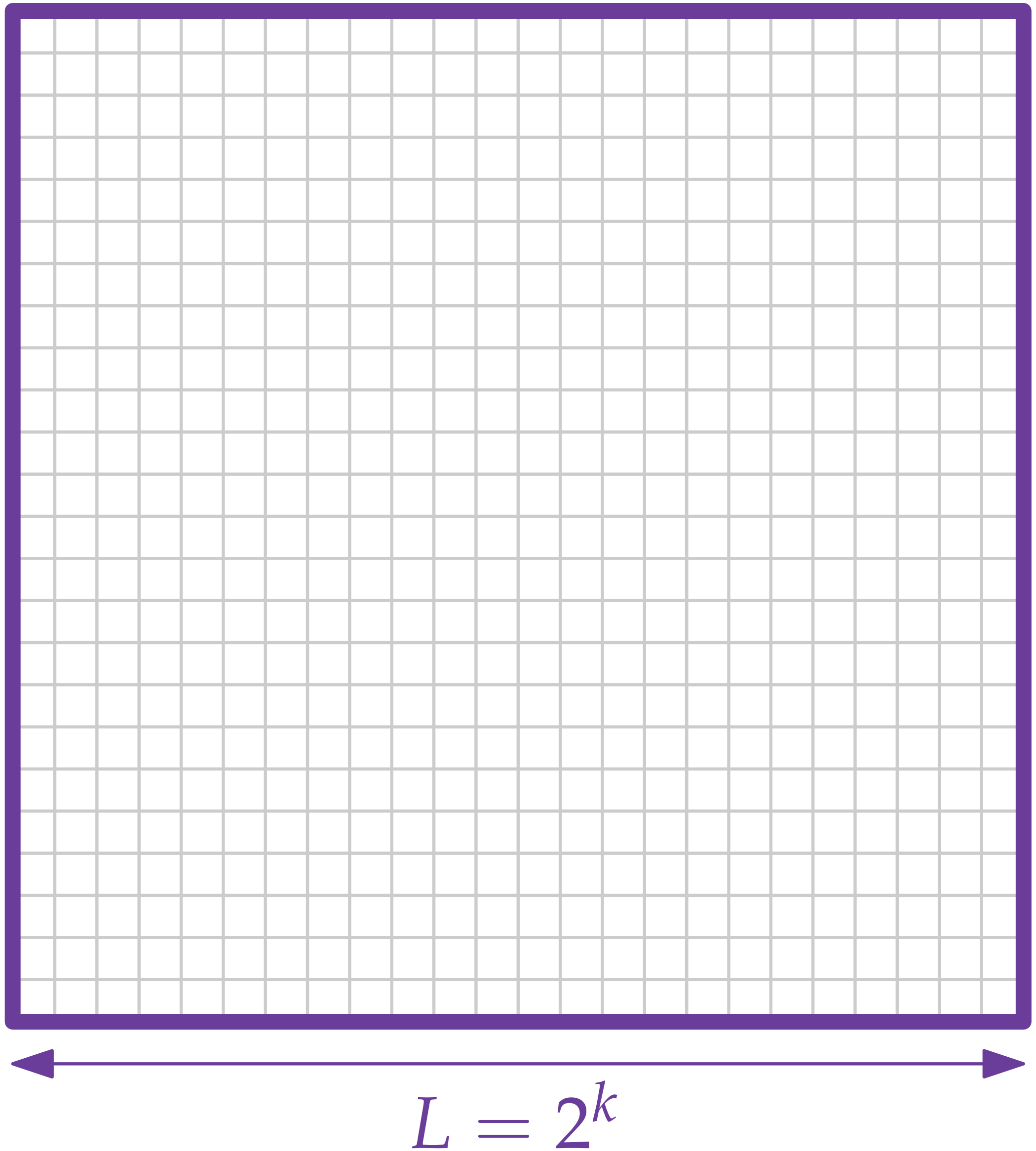
Joachim Spoerhase

Winter 2021/22

Basic Dissection

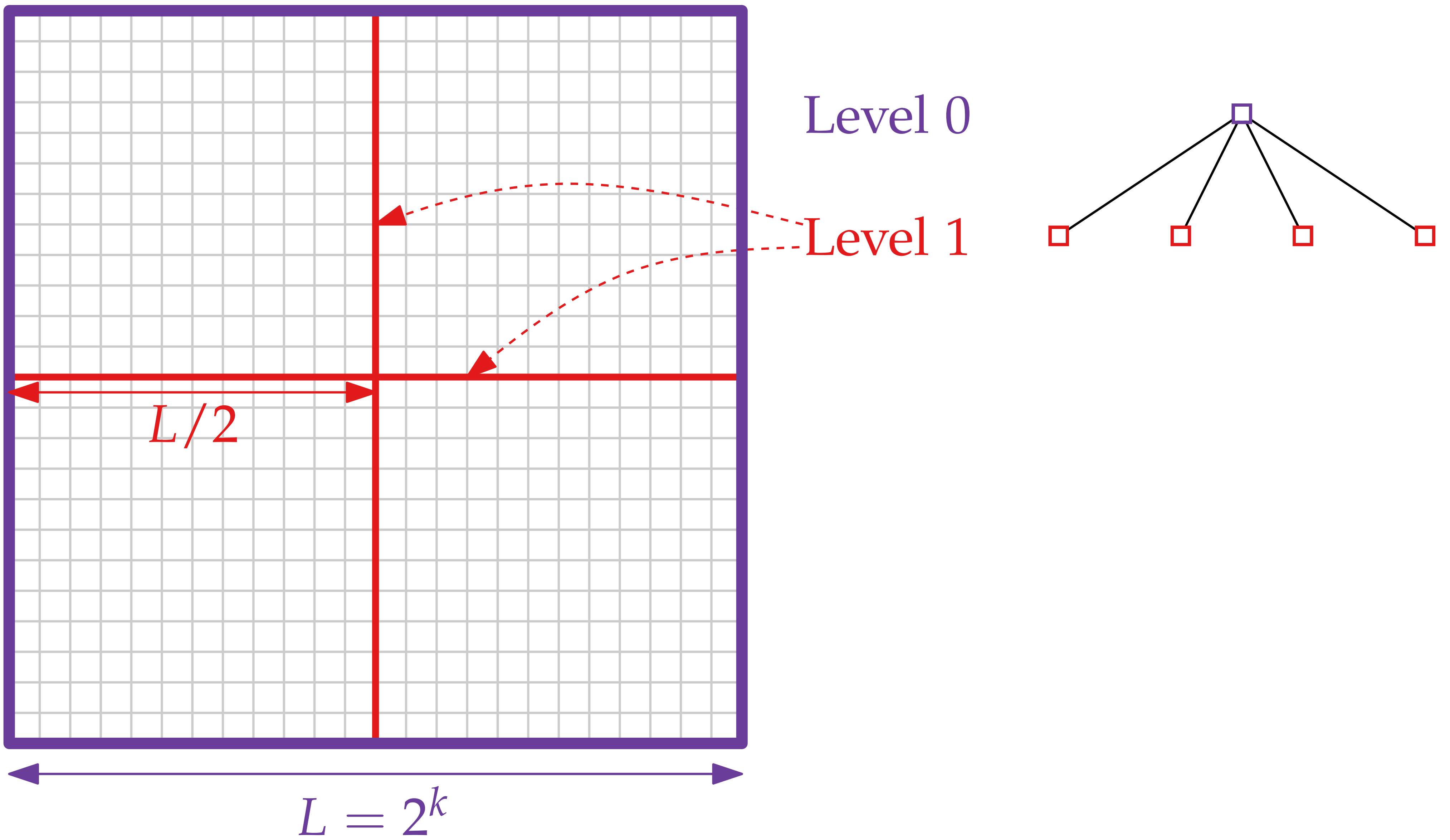


Basic Dissection

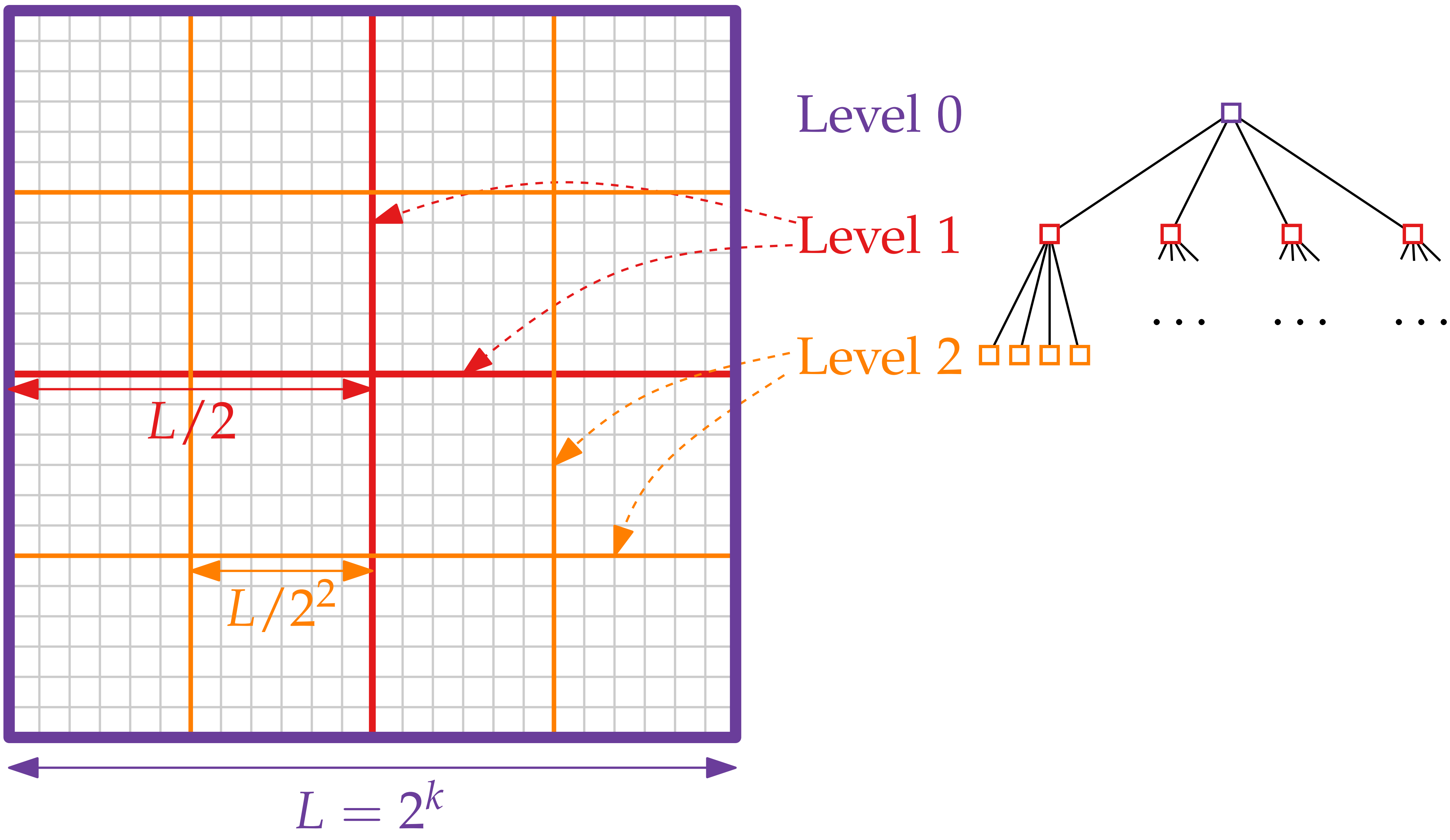


Level 0 □

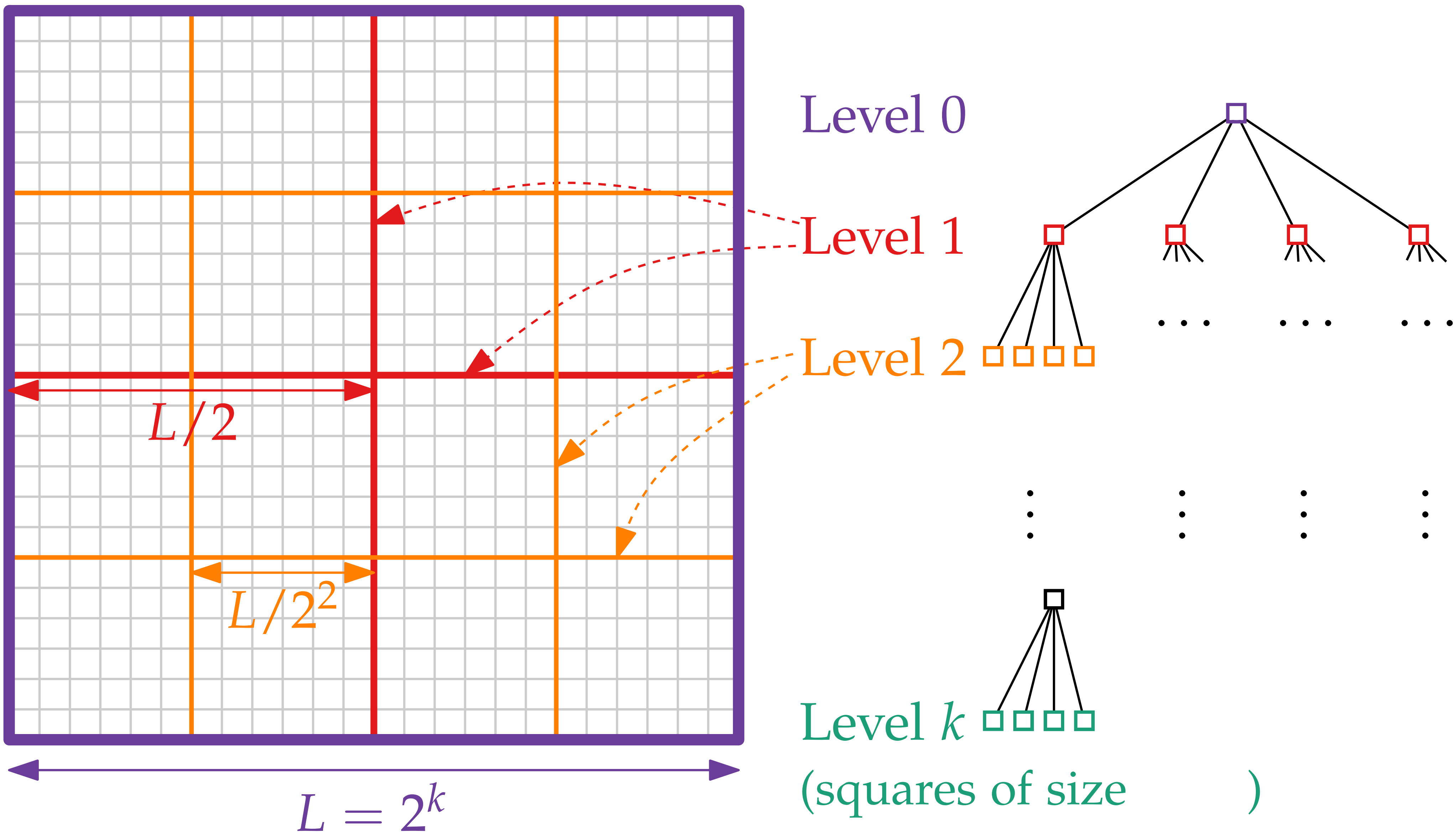
Basic Dissection



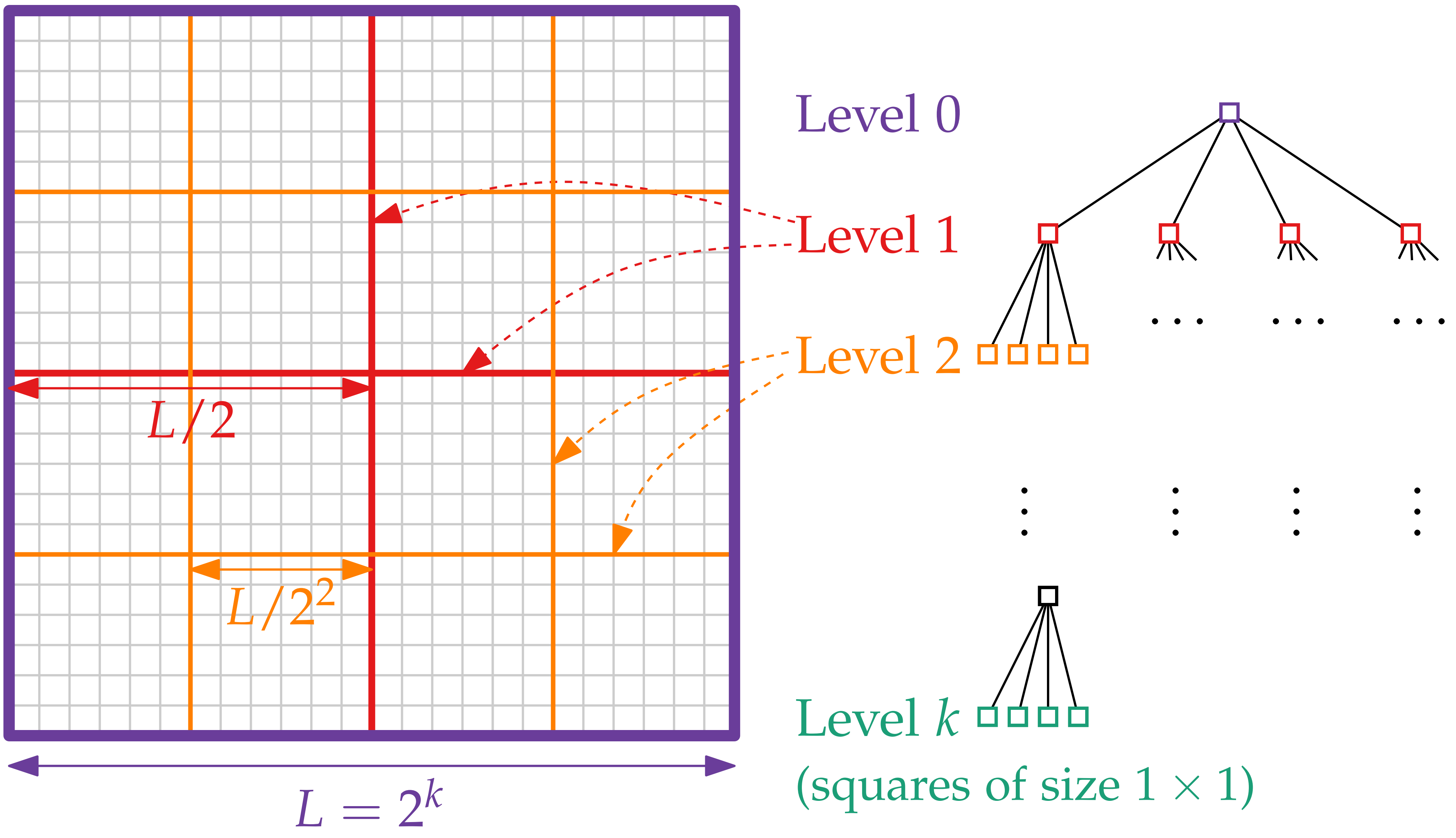
Basic Dissection



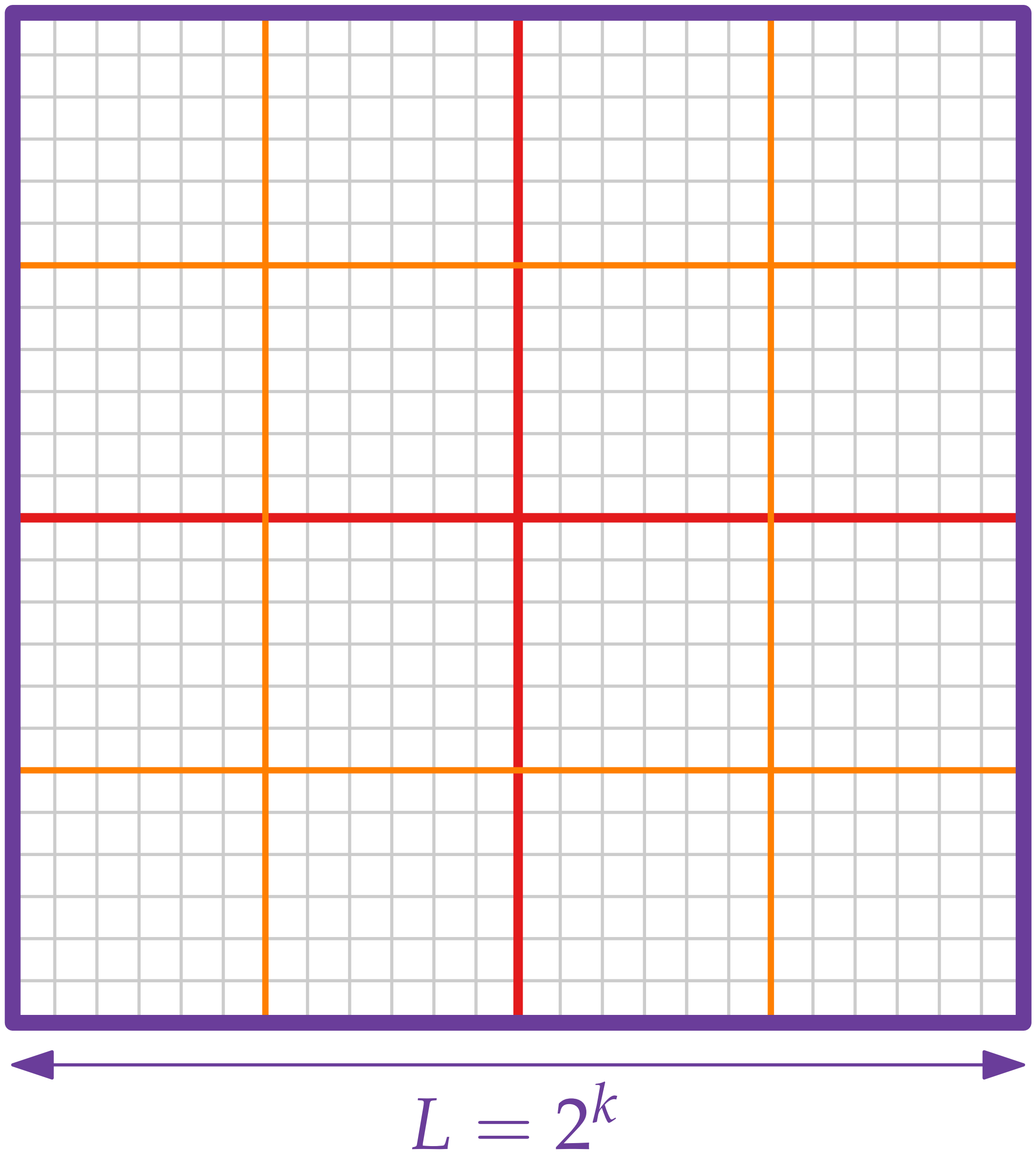
Basic Dissection



Basic Dissection

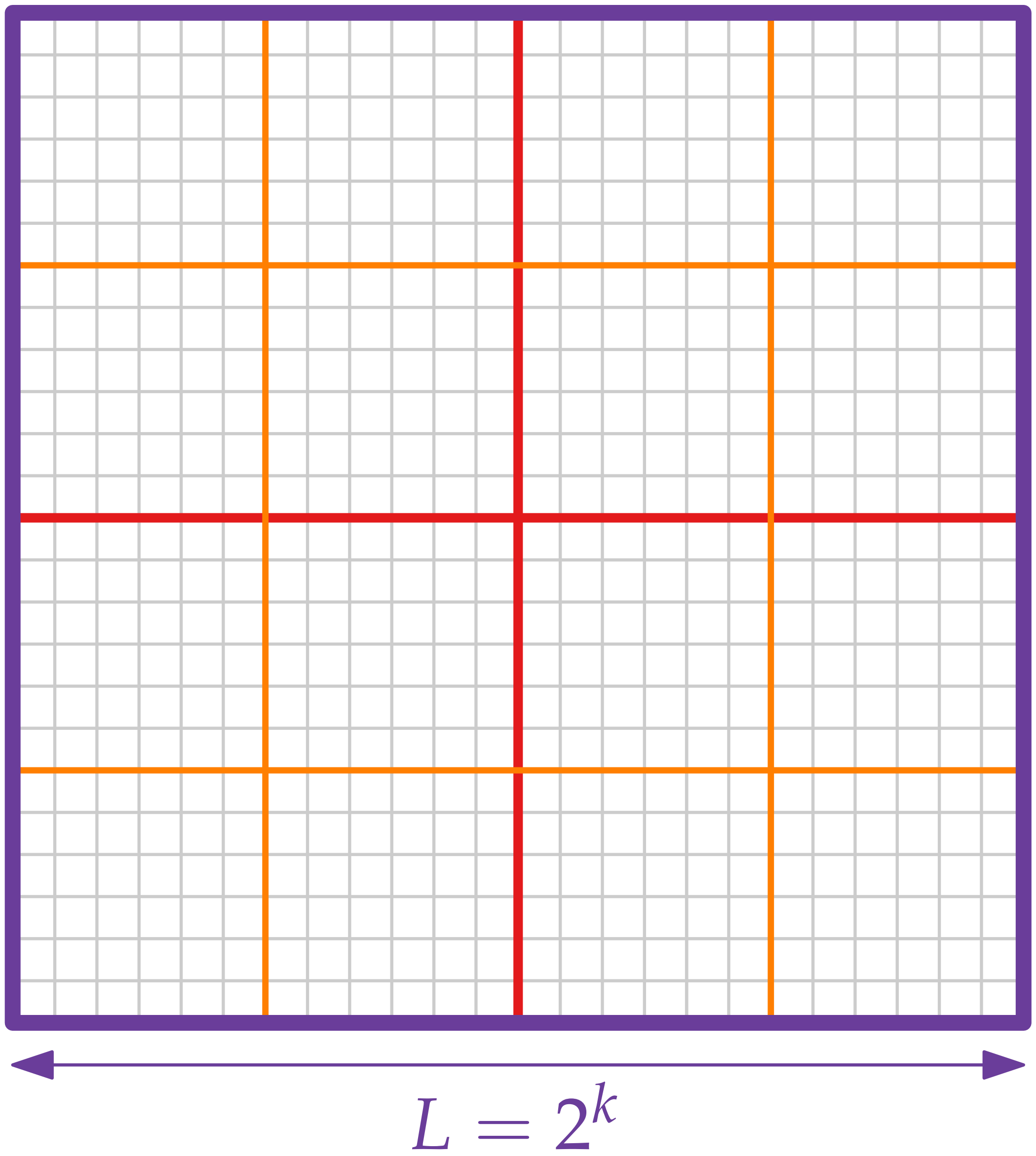


Portals



- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

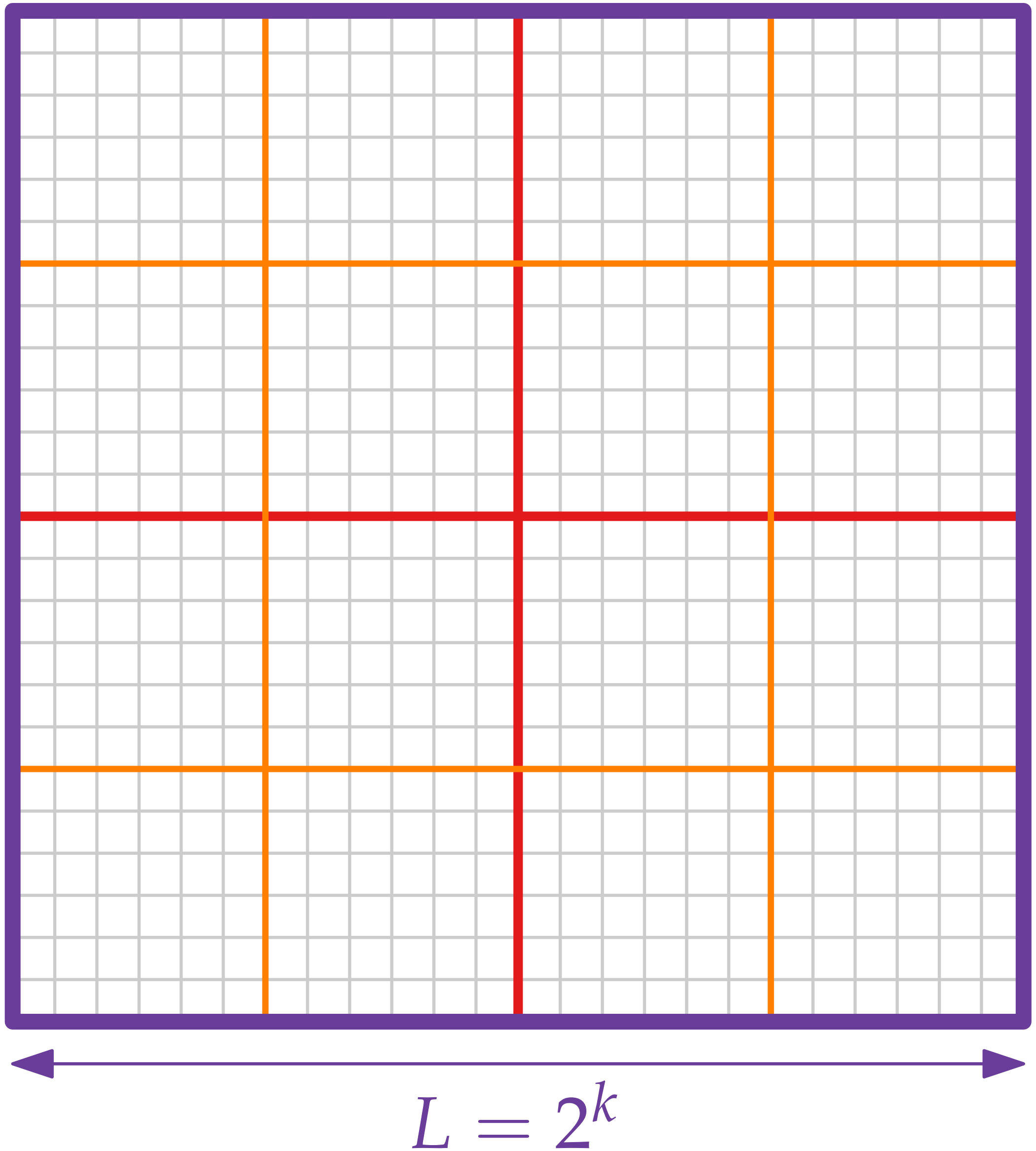
Portals



- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$

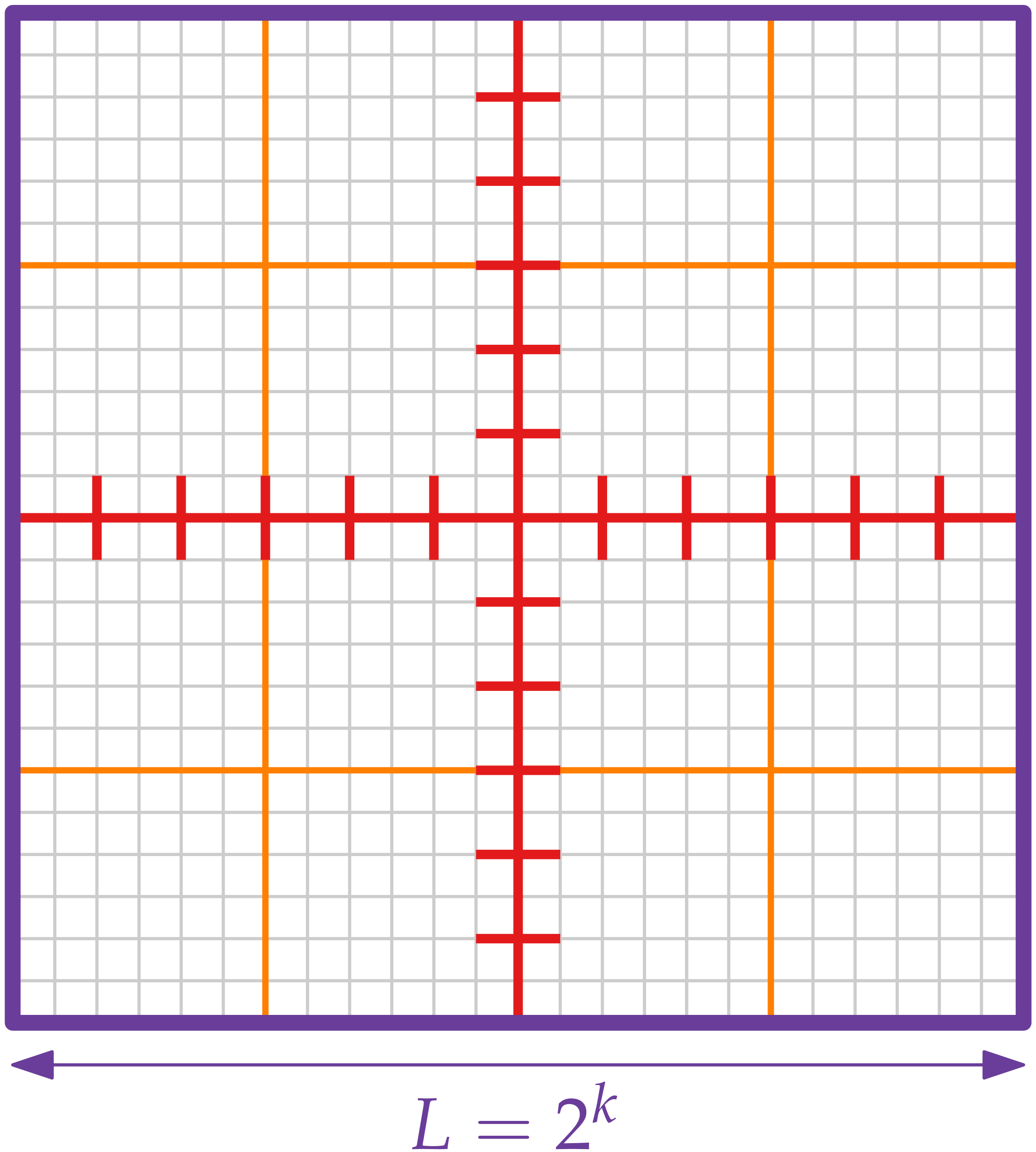
Portals



- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

Portals

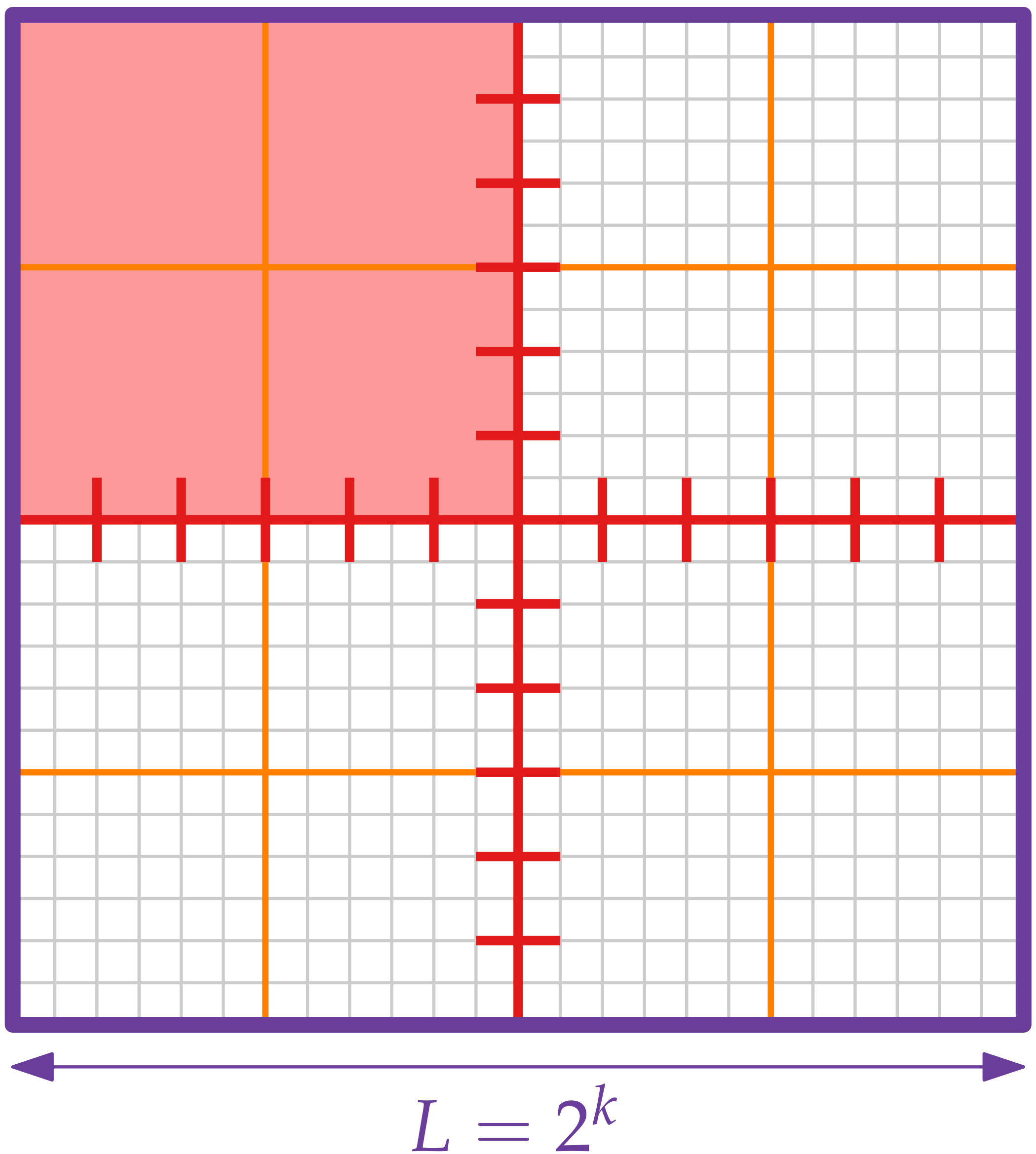


- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

- **Portals** on level-i-line with distance $L/(2^i m)$

Portals



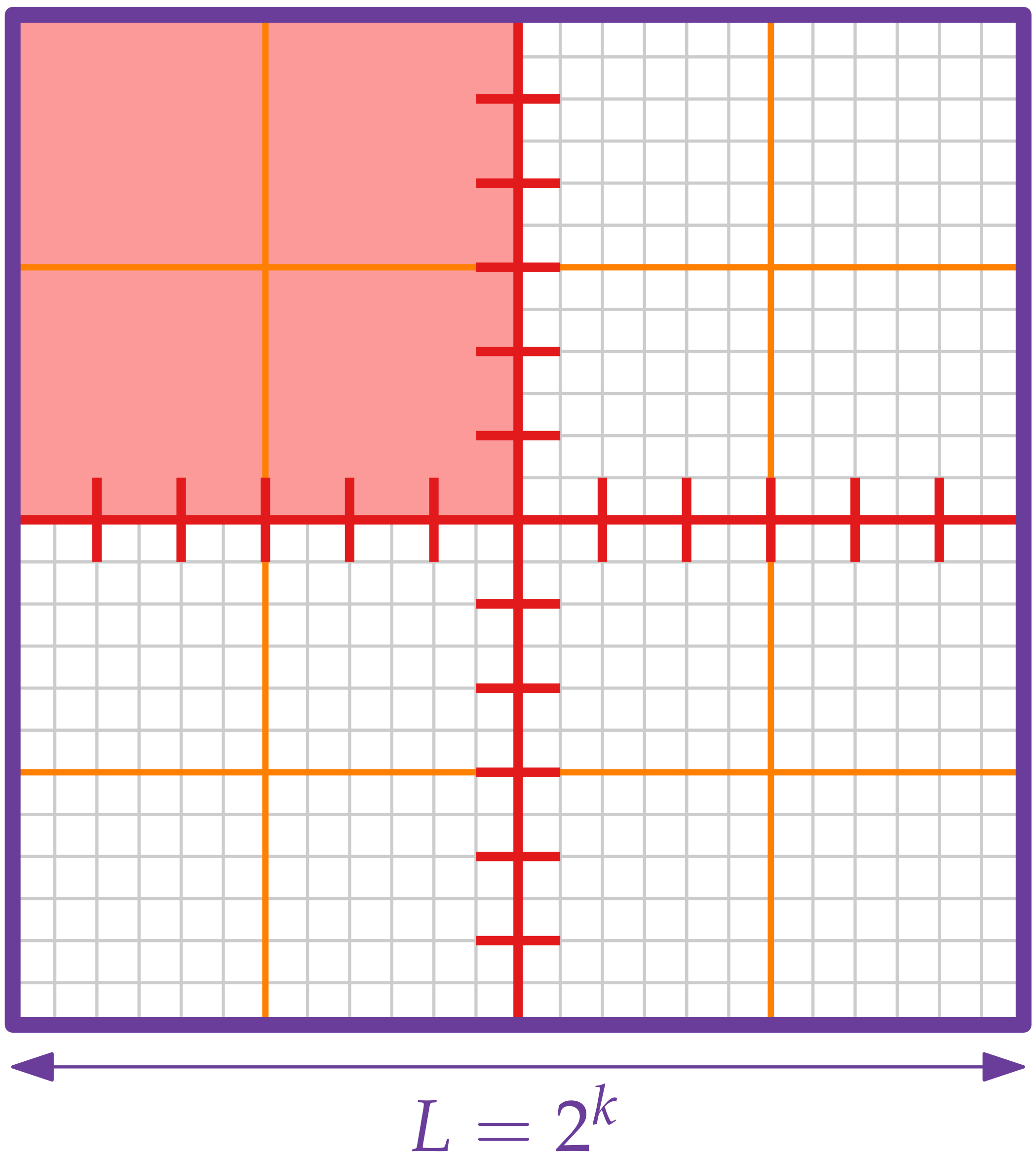
- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

- **Portals** on level-i-line with distance $L/(2^i m)$

- Level-i-square: size

Portals



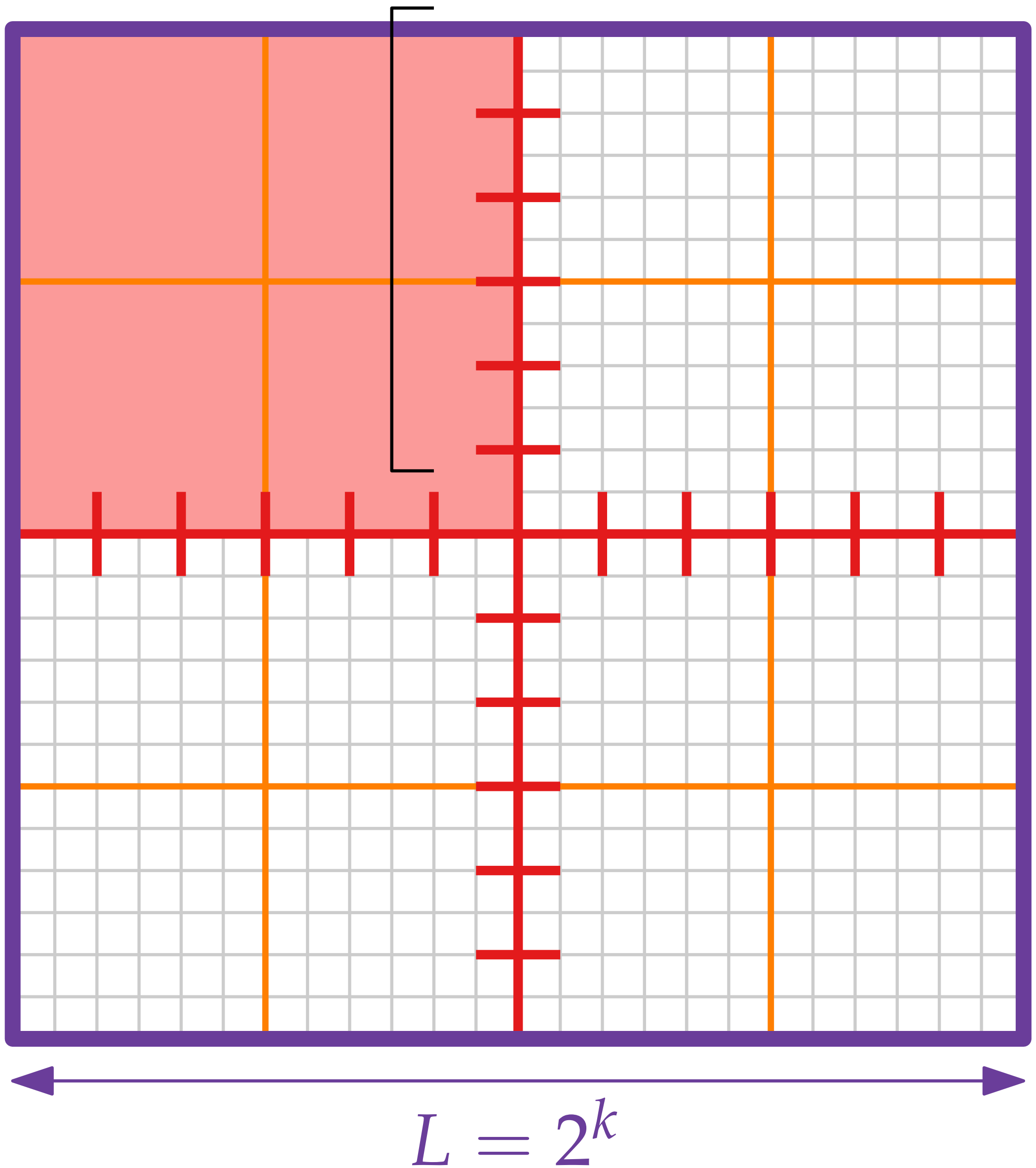
- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

- **Portals** on level-i-line with distance $L/(2^i m)$

- Level-i-square: size $L/2^i \times L/2^i$

Portals



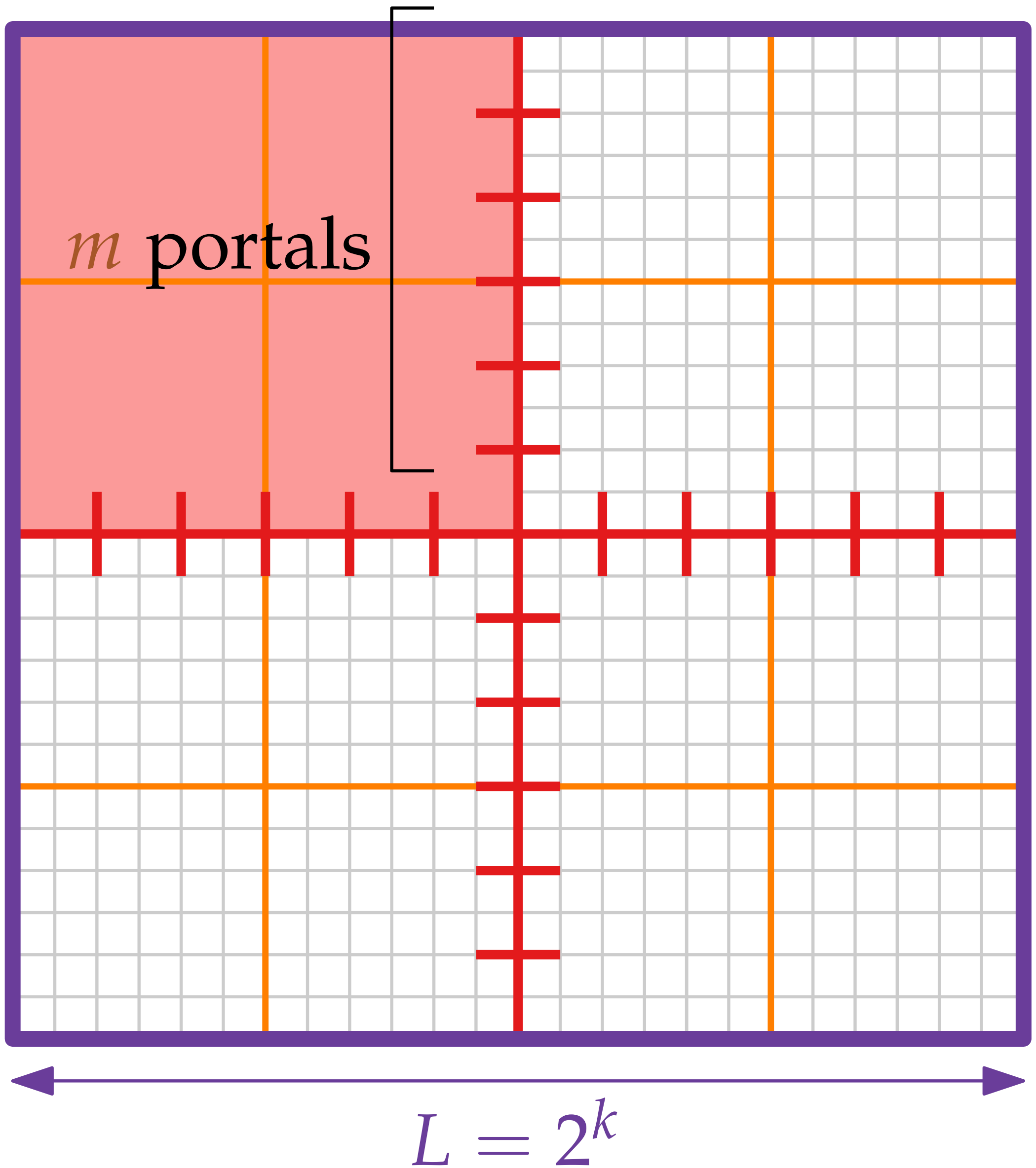
- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

- **Portals** on level-i-line with distance $L/(2^i m)$

- Level-i-square: size $L/2^i \times L/2^i$

Portals



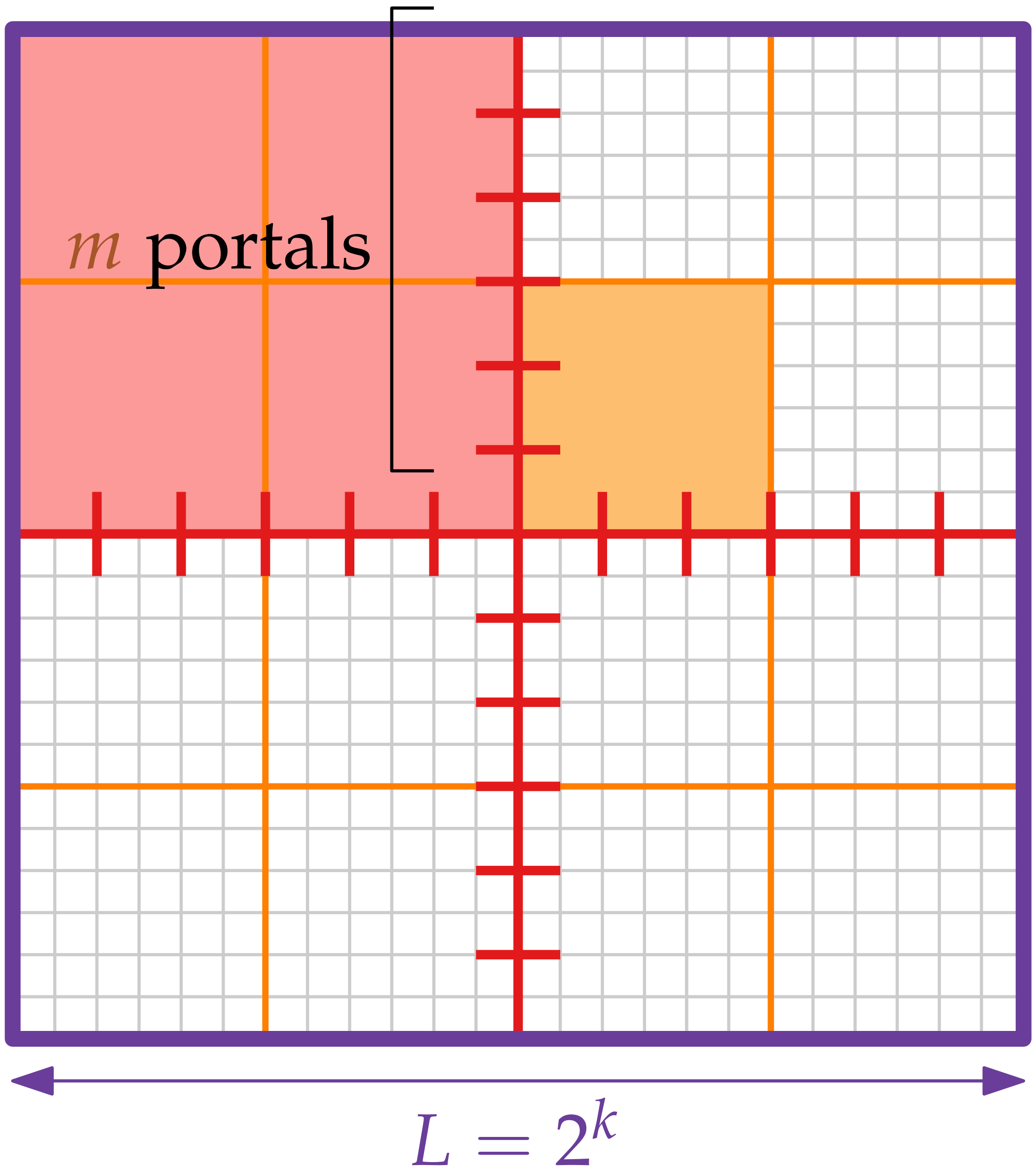
- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

- **Portals** on level-i-line with distance $L/(2^i m)$

- Level-i-square: size $L/2^i \times L/2^i$

Portals



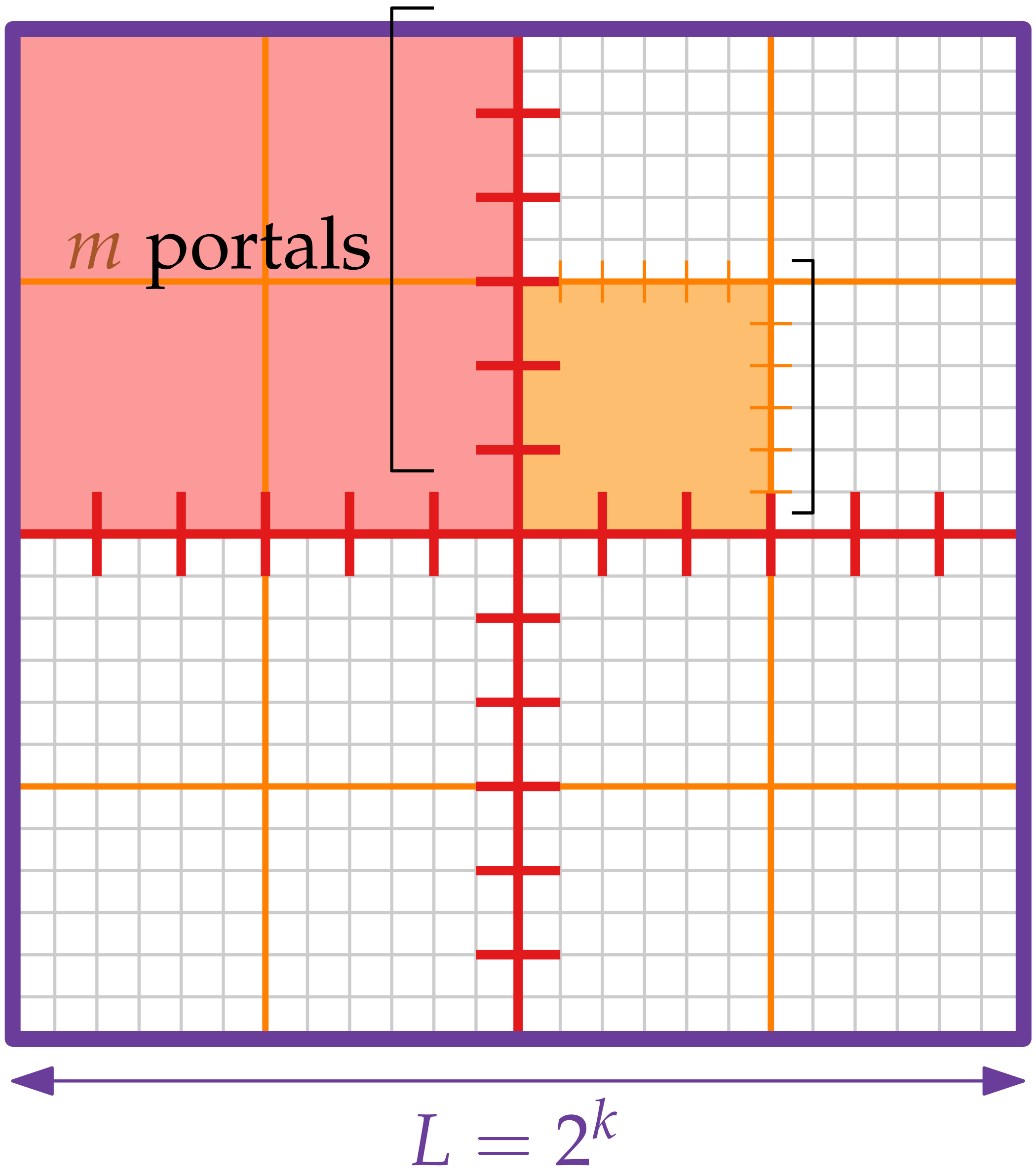
- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

- **Portals** on level-i-line with distance $L/(2^i m)$

- Level-i-square: size $L/2^i \times L/2^i$

Portals



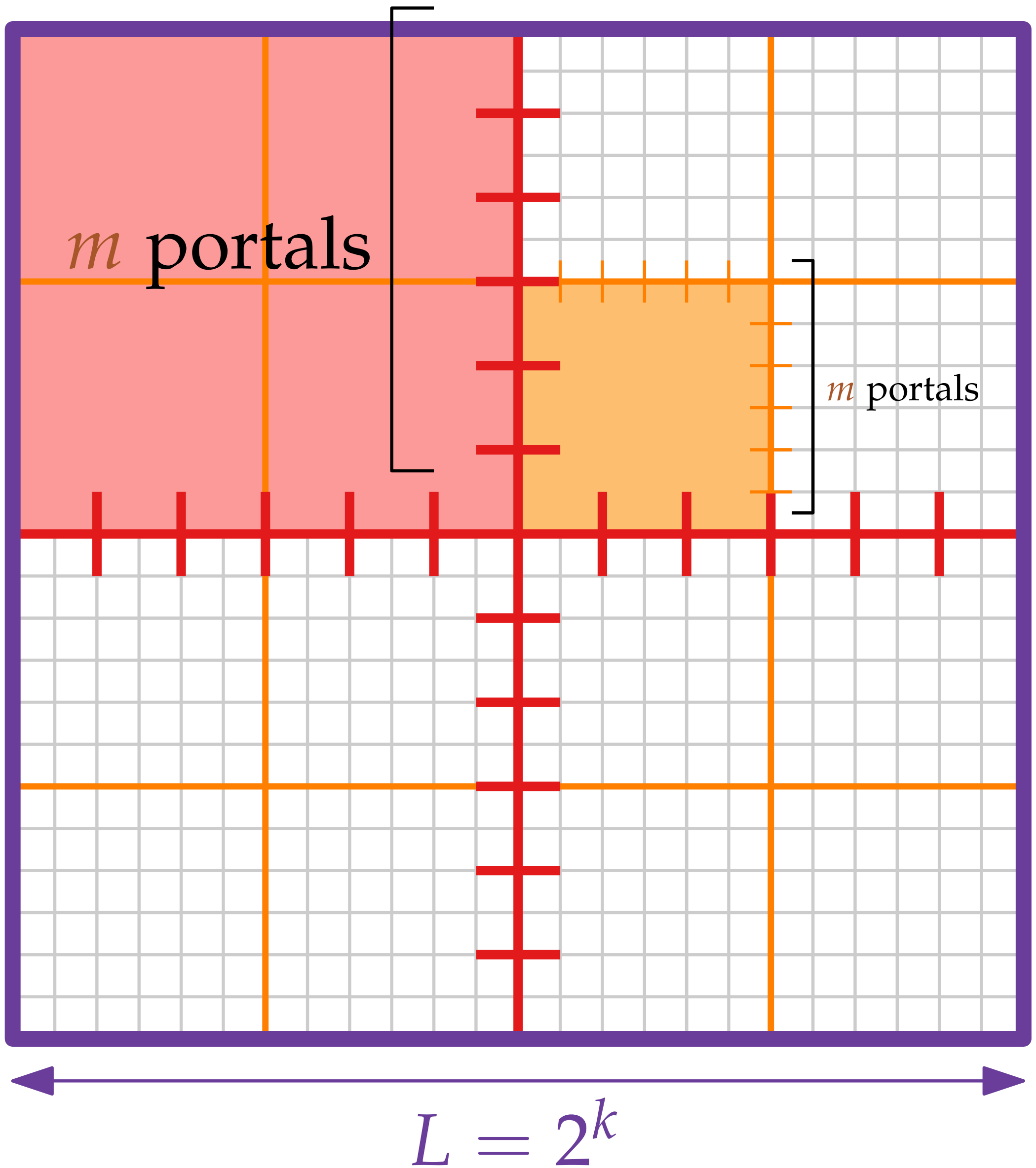
- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

- **Portals** on level-i-line with distance $L/(2^i m)$

- Level-i-square: size $L/2^i \times L/2^i$

Portals



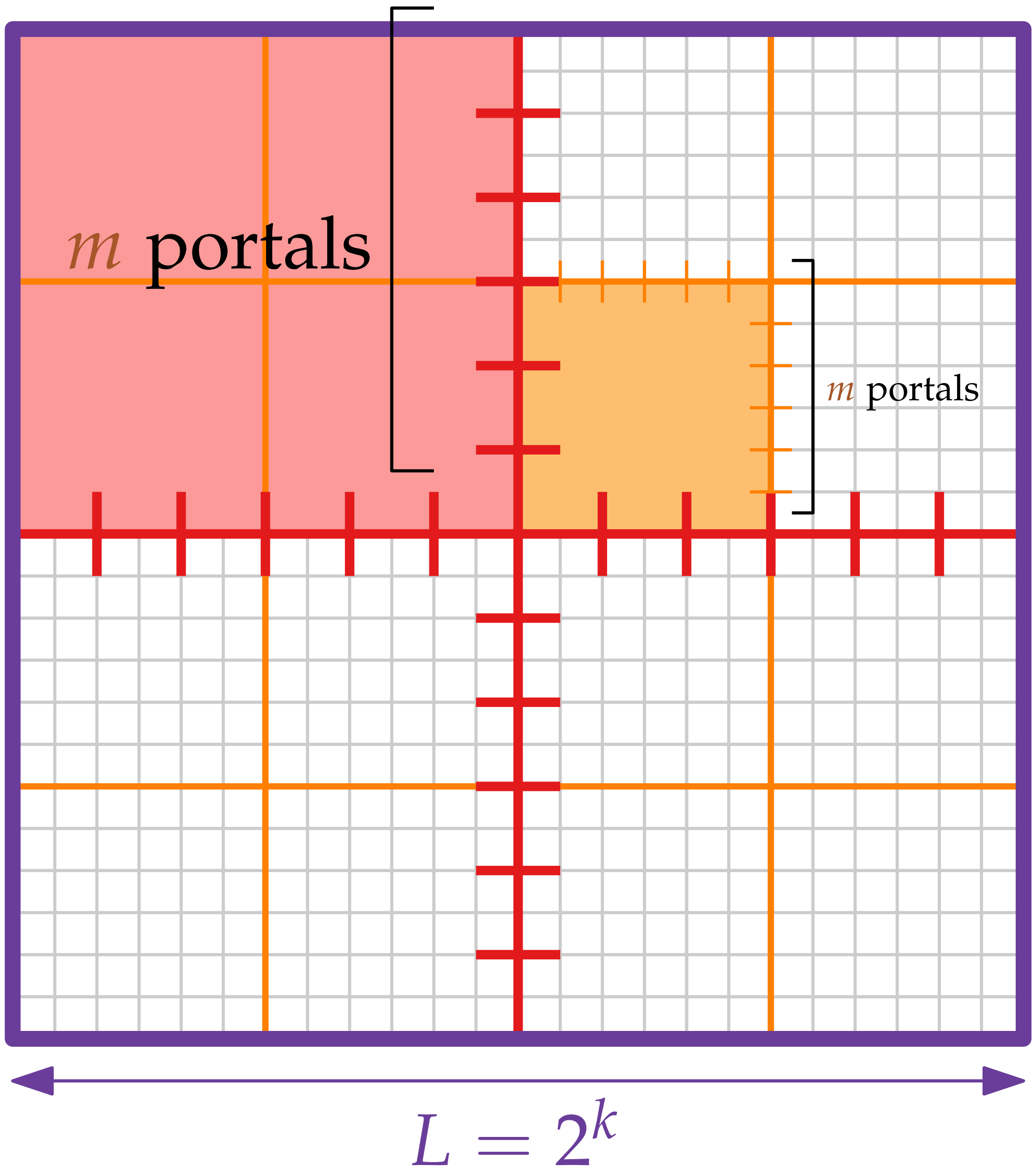
- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

- **Portals** on level-i-line with distance $L/(2^i m)$

- Level-i-square: size $L/2^i \times L/2^i$

Portals



- m power of two in interval $[k/\varepsilon, 2k/\varepsilon]$

  $k = 2 + 2\log_2 n$
  $\Rightarrow m = O((\log n)/\varepsilon)$

- **Portals** on level-i-line with distance $L/(2^i m)$

- Level-i-square: size $L/2^i \times L/2^i$

- Level-i-square has at most $4m$ portals on its boundary.

Approximation Algorithms

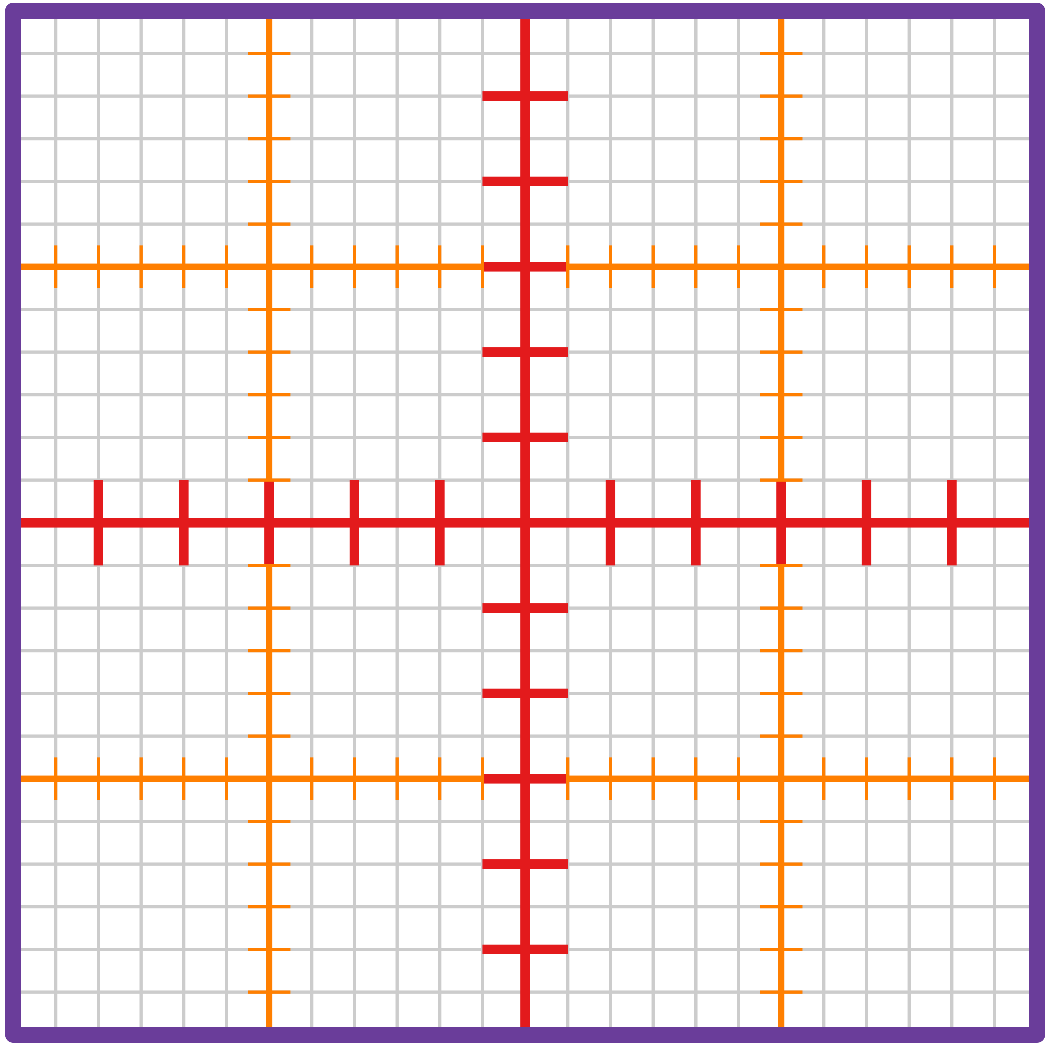
Lecture 9:
PTAS for EuclideanTSP

Part III:
Well Behaved Tours

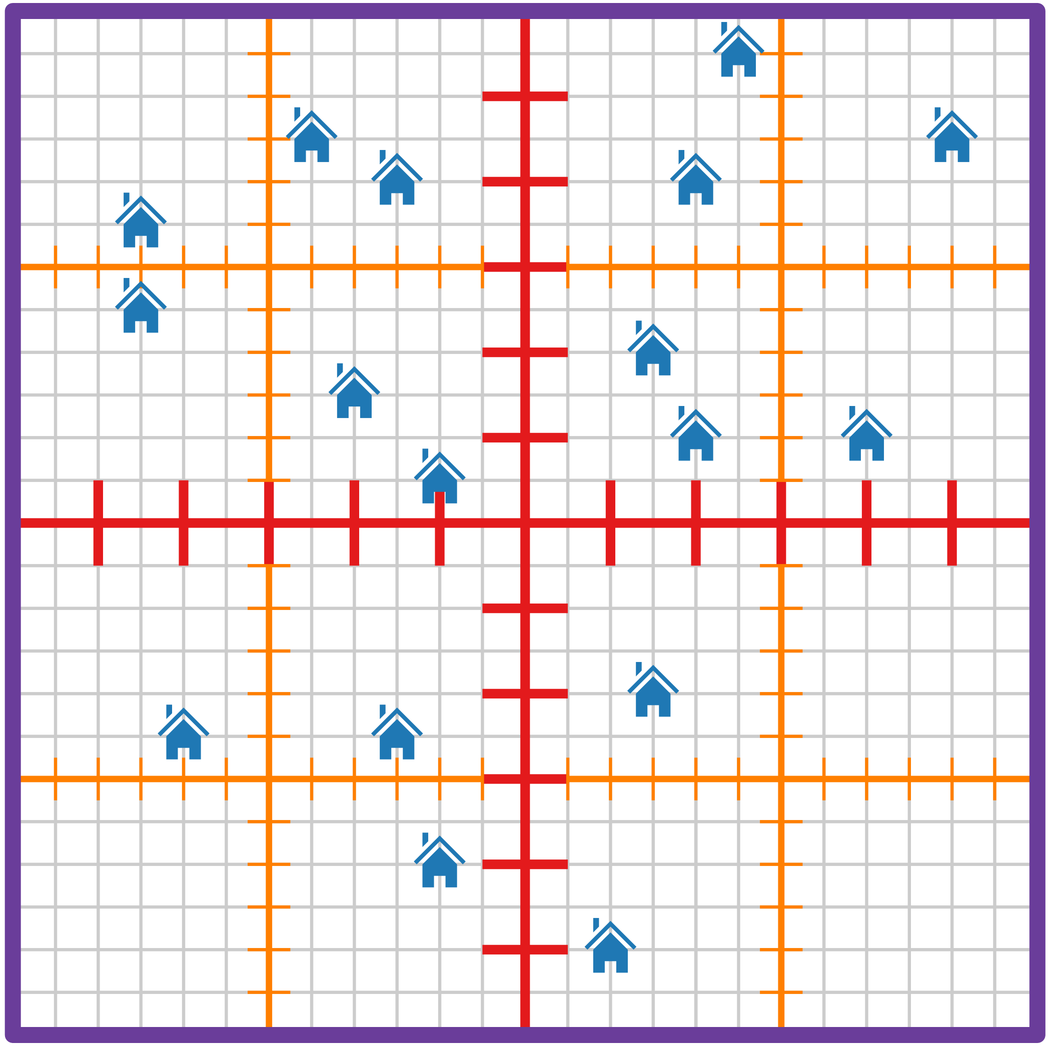
Joachim Spoerhase

Winter 2021/22

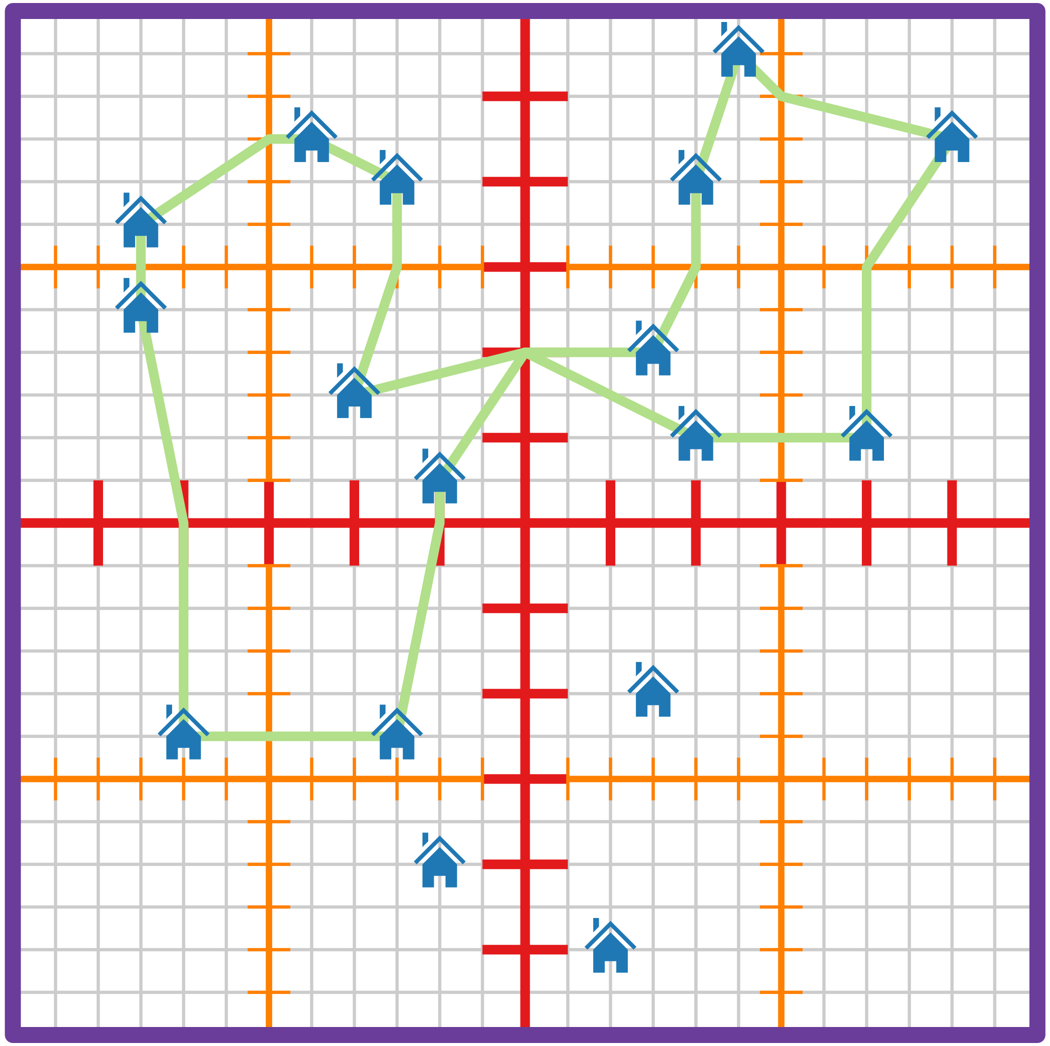
Well Behaved Tours



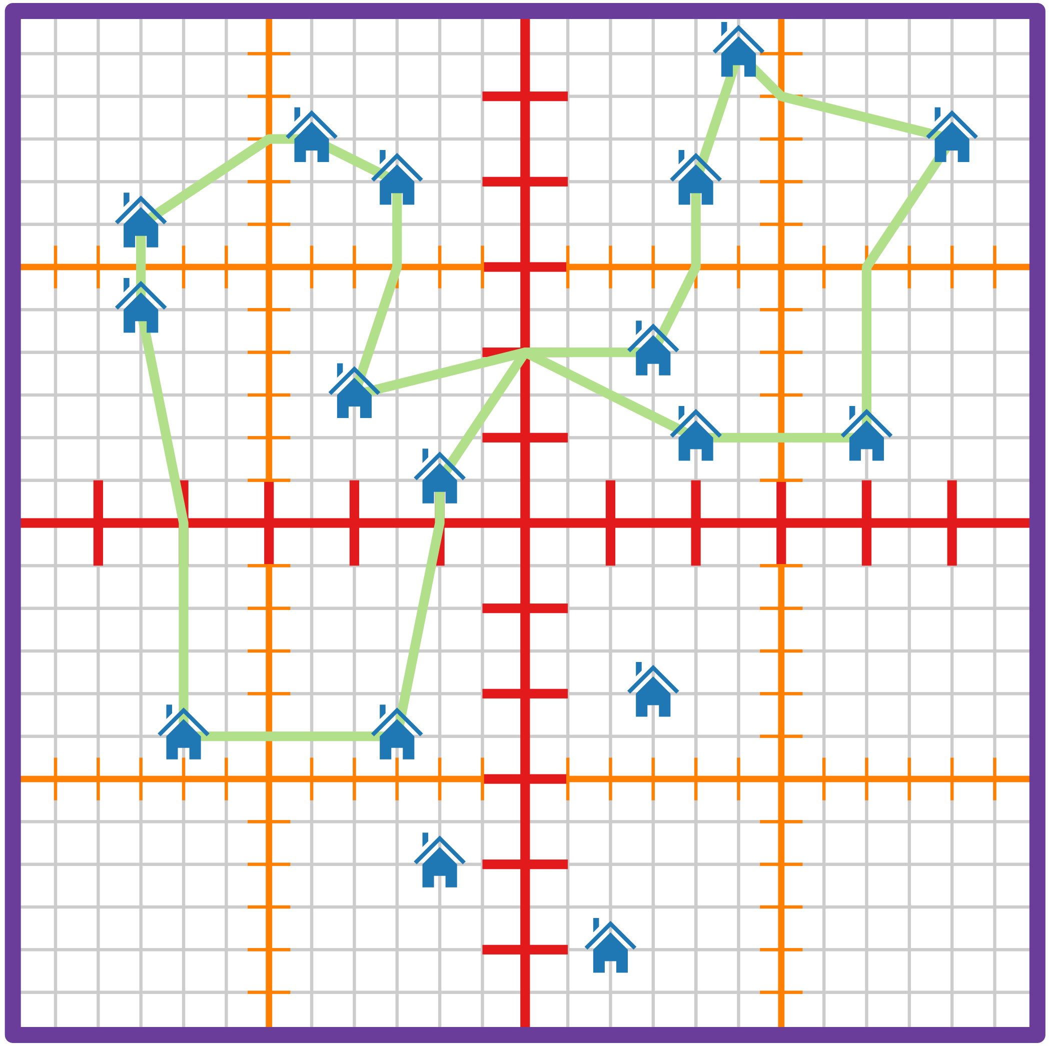
Well Behaved Tours



Well Behaved Tours

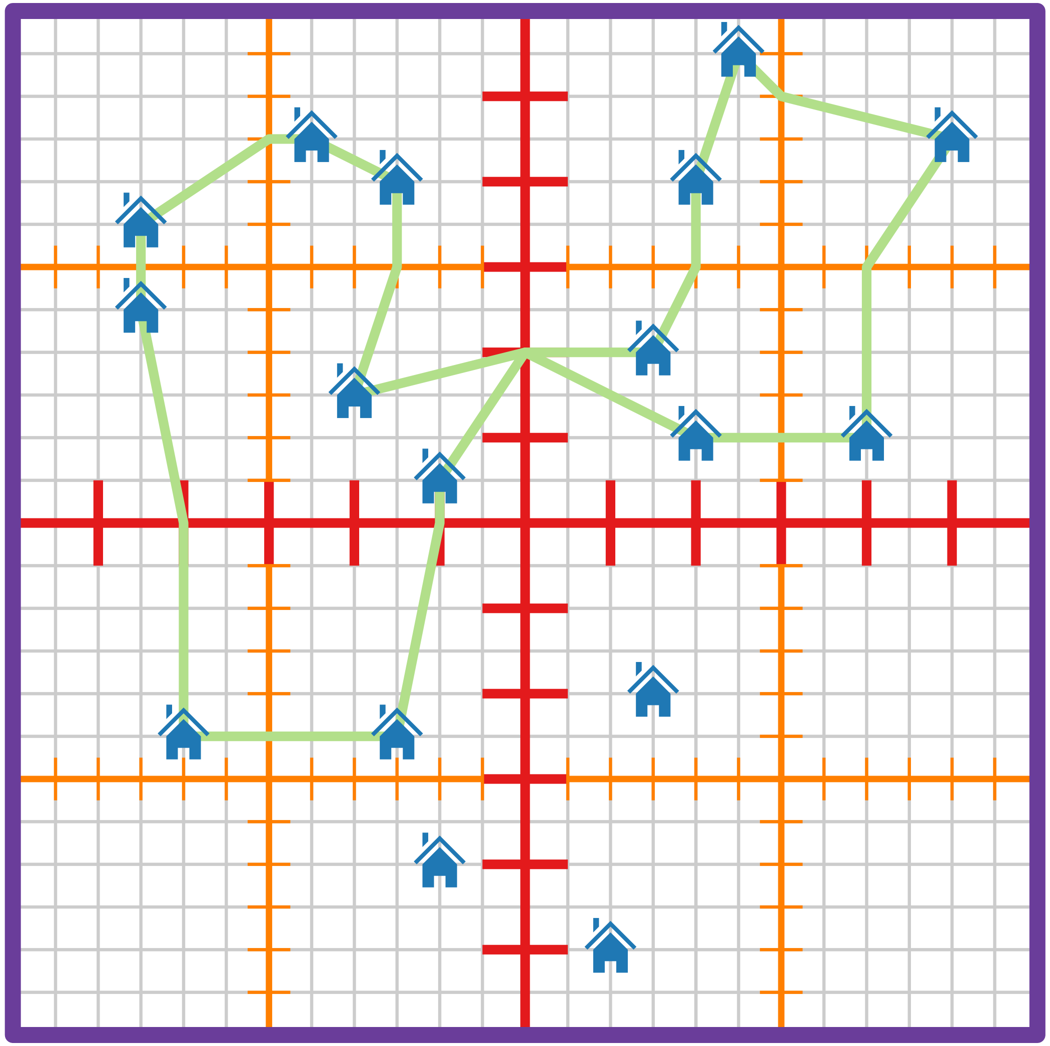


Well Behaved Tours



A tour is *well behaved* if

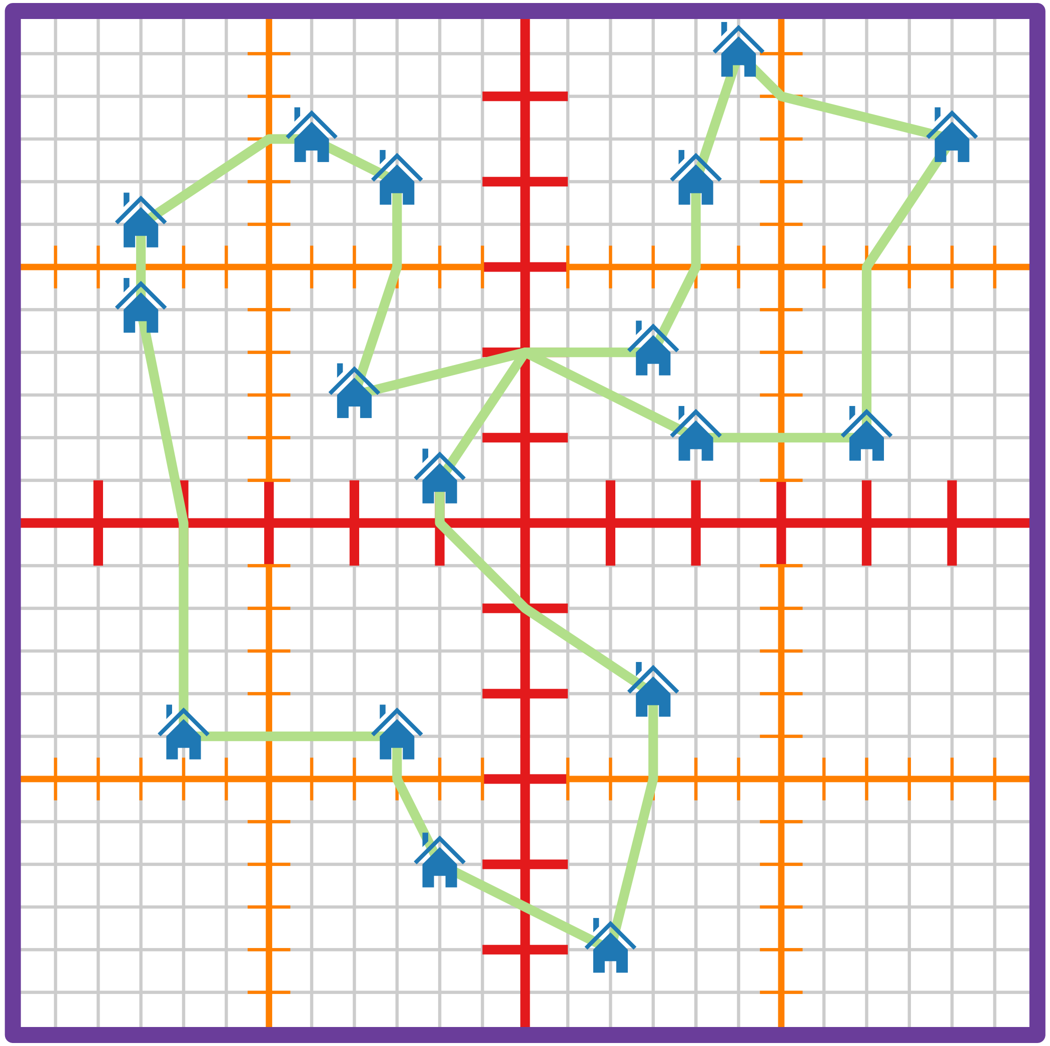
Well Behaved Tours



A tour is *well behaved* if

- it involves all houses

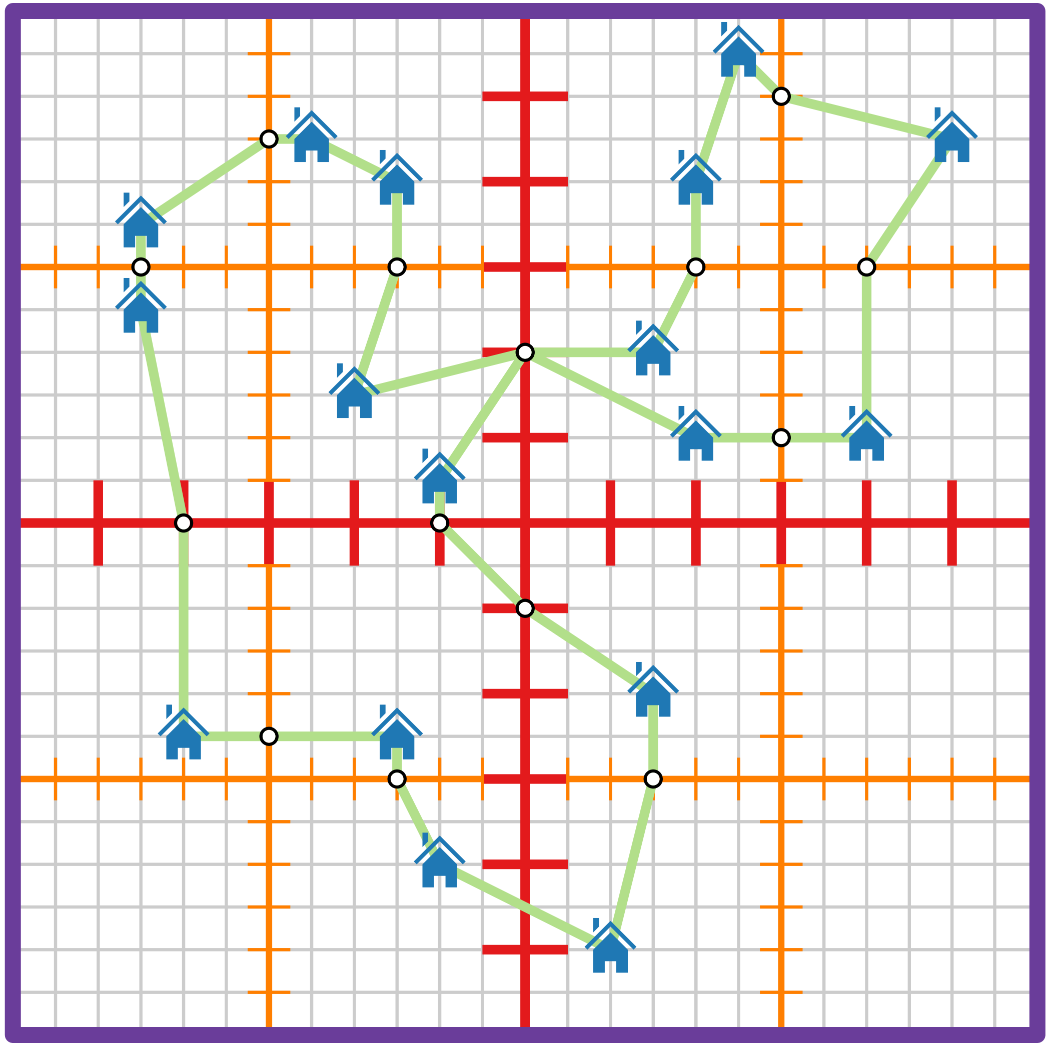
Well Behaved Tours



A tour is *well behaved* if

- it involves all houses

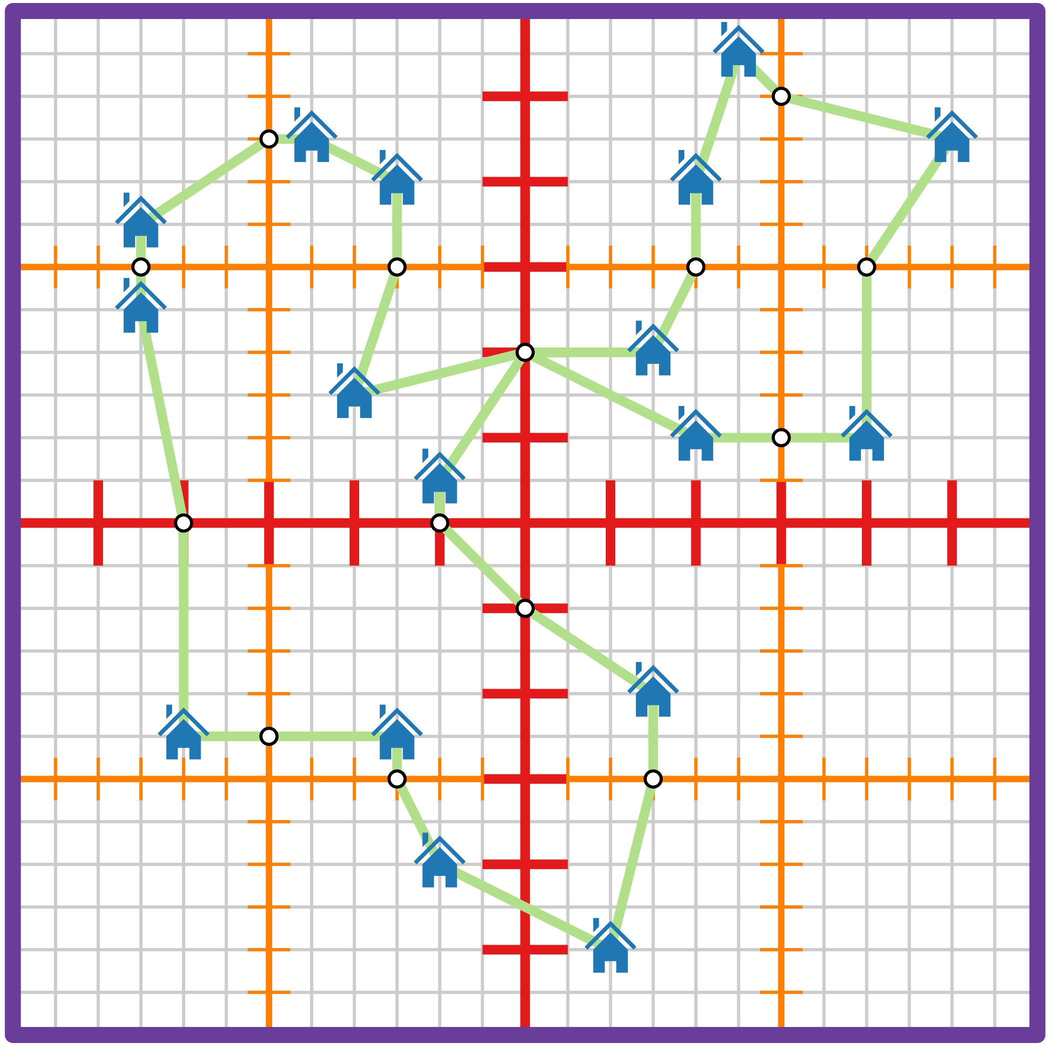
Well Behaved Tours



A tour is *well behaved* if

- it involves all houses and a subset of the portals,

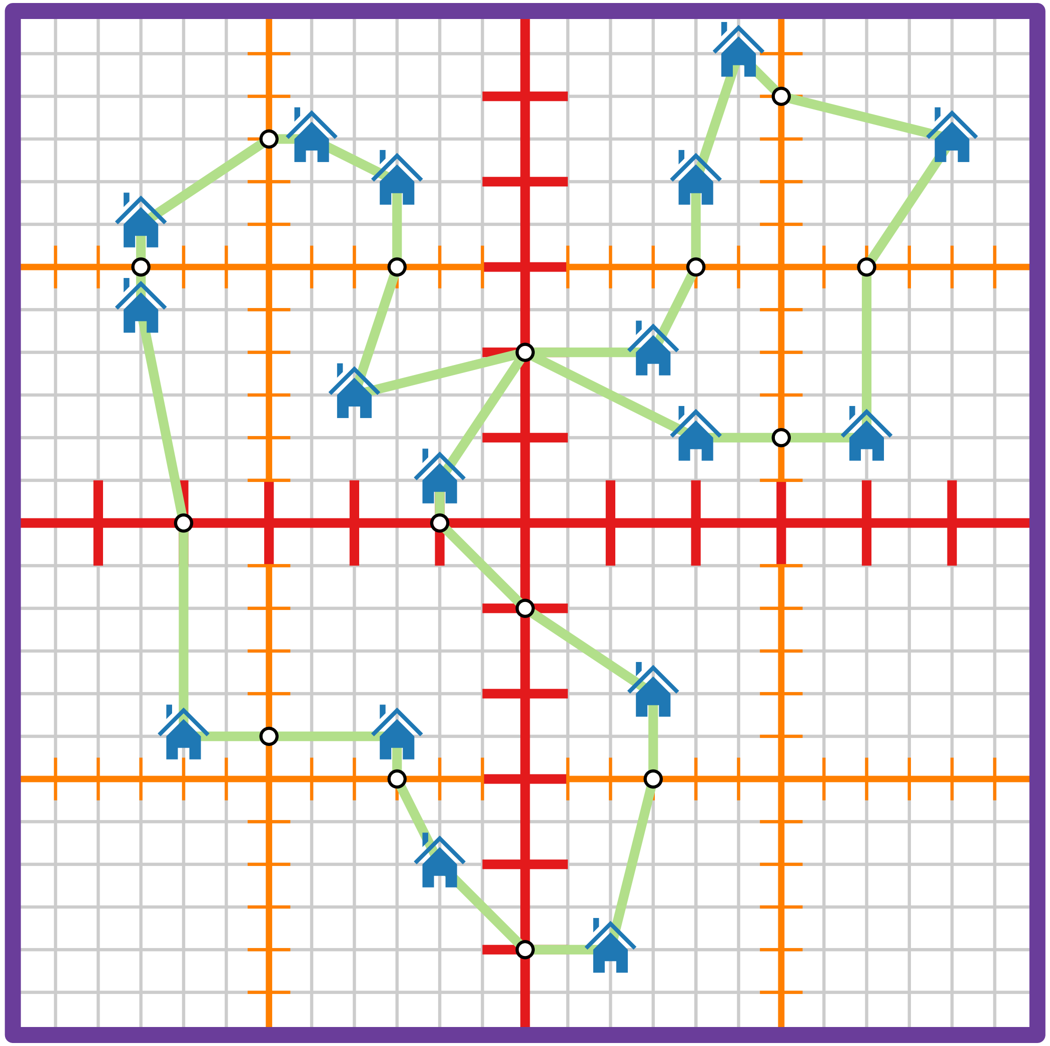
Well Behaved Tours



A tour is *well behaved* if

- it involves all houses and a subset of the portals,
- no edge of the tour crosses a line of the basic dissection,

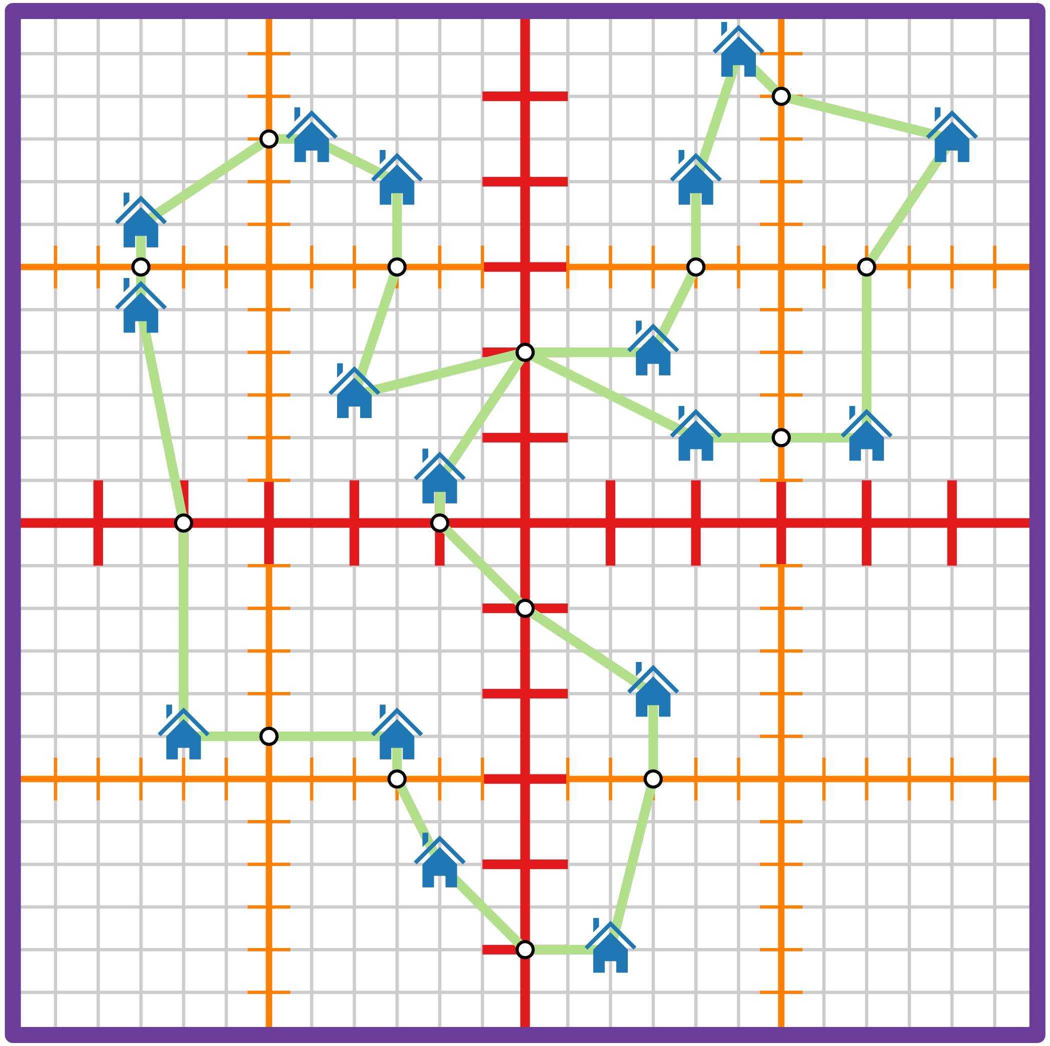
Well Behaved Tours



A tour is *well behaved* if

- it involves all houses and a subset of the portals,
- no edge of the tour crosses a line of the basic dissection,

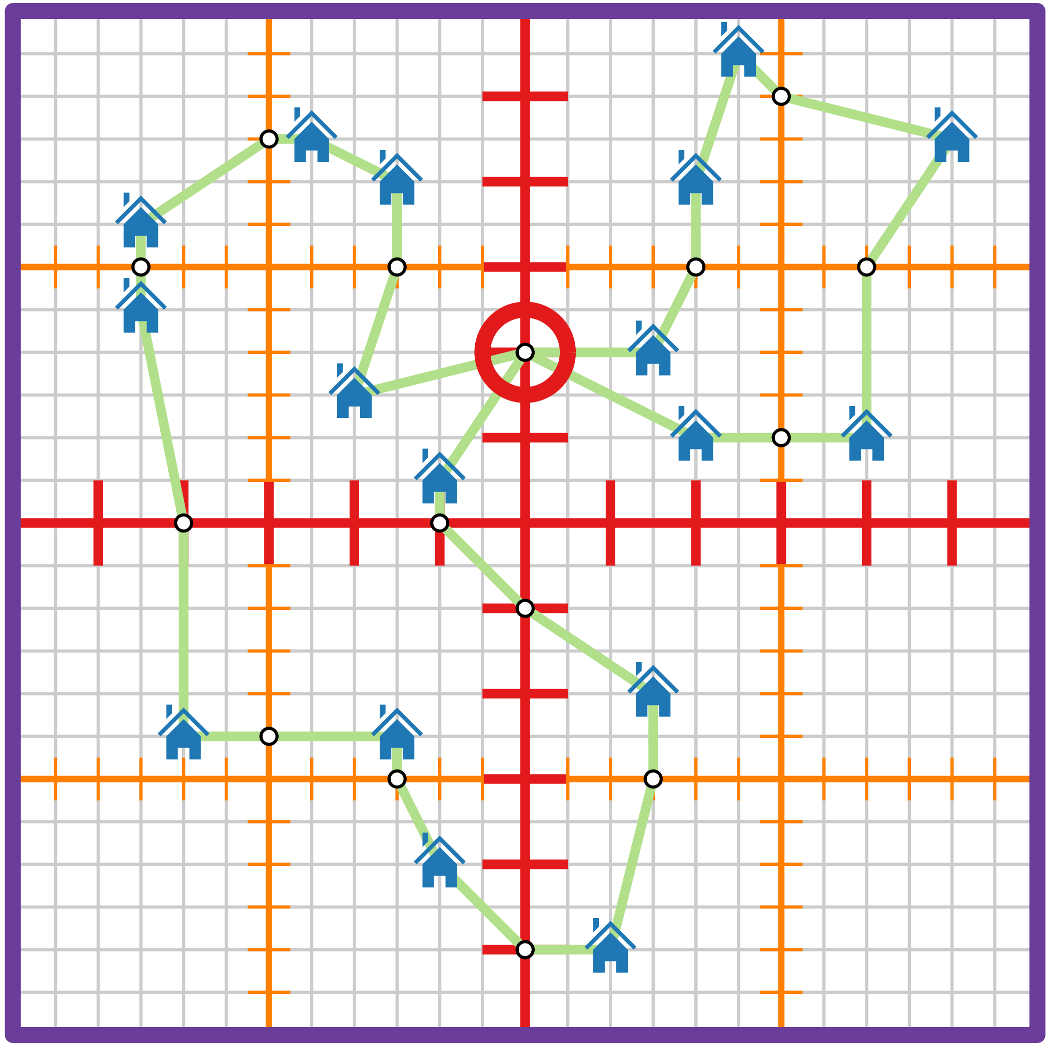
Well Behaved Tours



A tour is *well behaved* if

- it involves all houses and a subset of the portals,
- no edge of the tour crosses a line of the basic dissection,
- it is crossing-free.

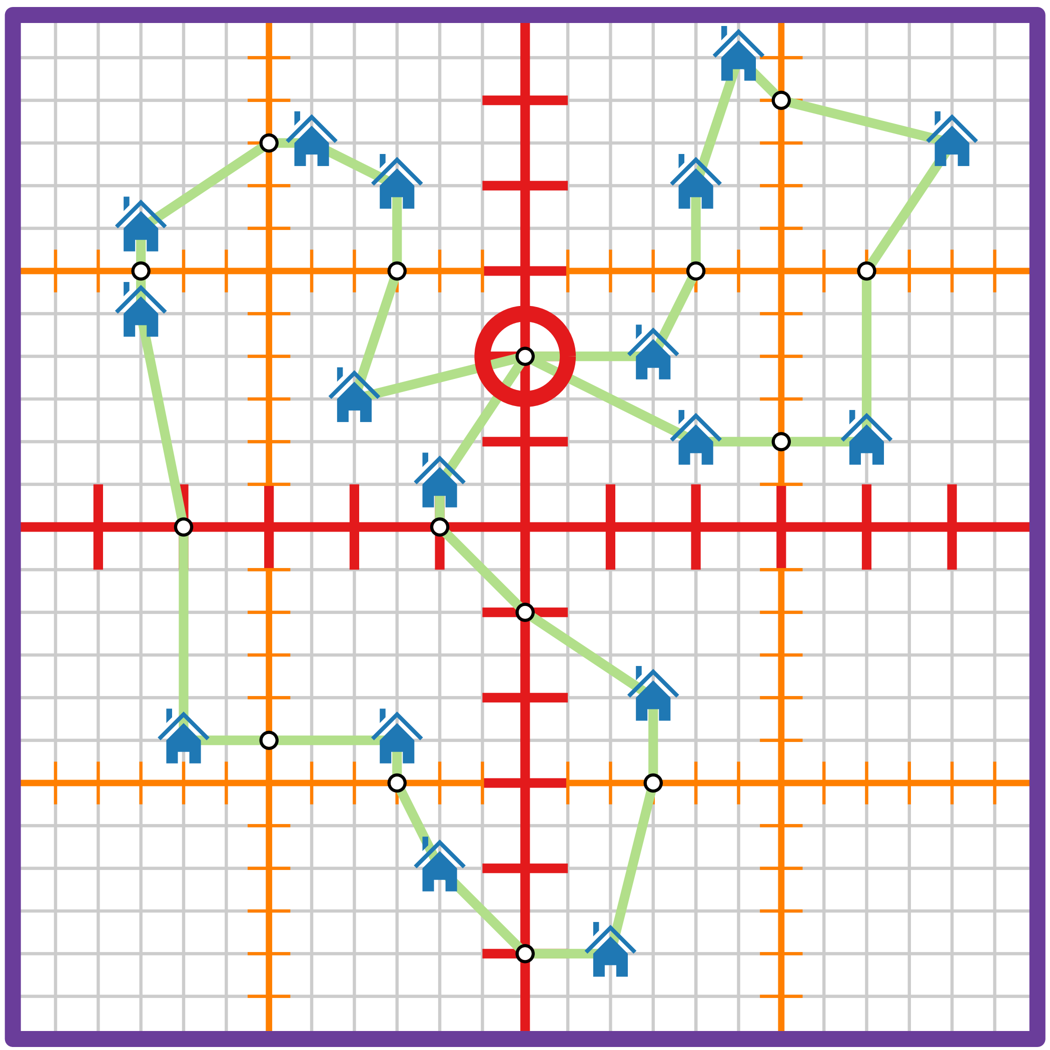
Well Behaved Tours



A tour is *well behaved* if

- it involves all houses and a subset of the portals,
- no edge of the tour crosses a line of the basic dissection,
- it is crossing-free.

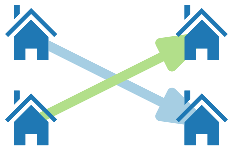
Well Behaved Tours



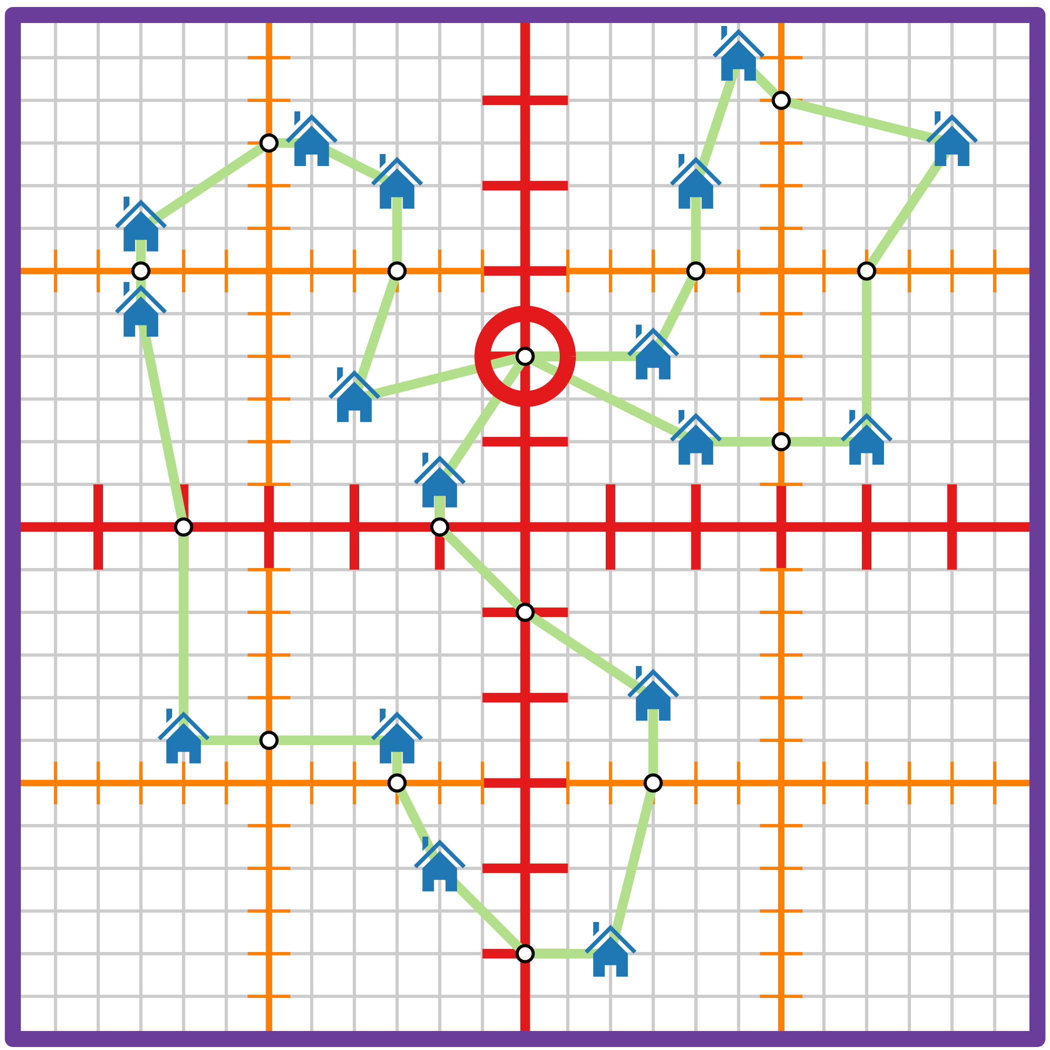
A tour is *well behaved* if

- it involves all houses and a subset of the portals,
- no edge of the tour crosses a line of the basic dissection,
- it is crossing-free.

Crossing



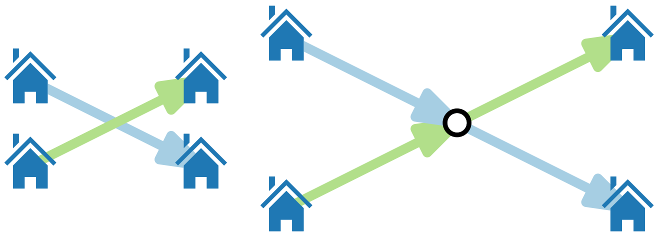
Well Behaved Tours



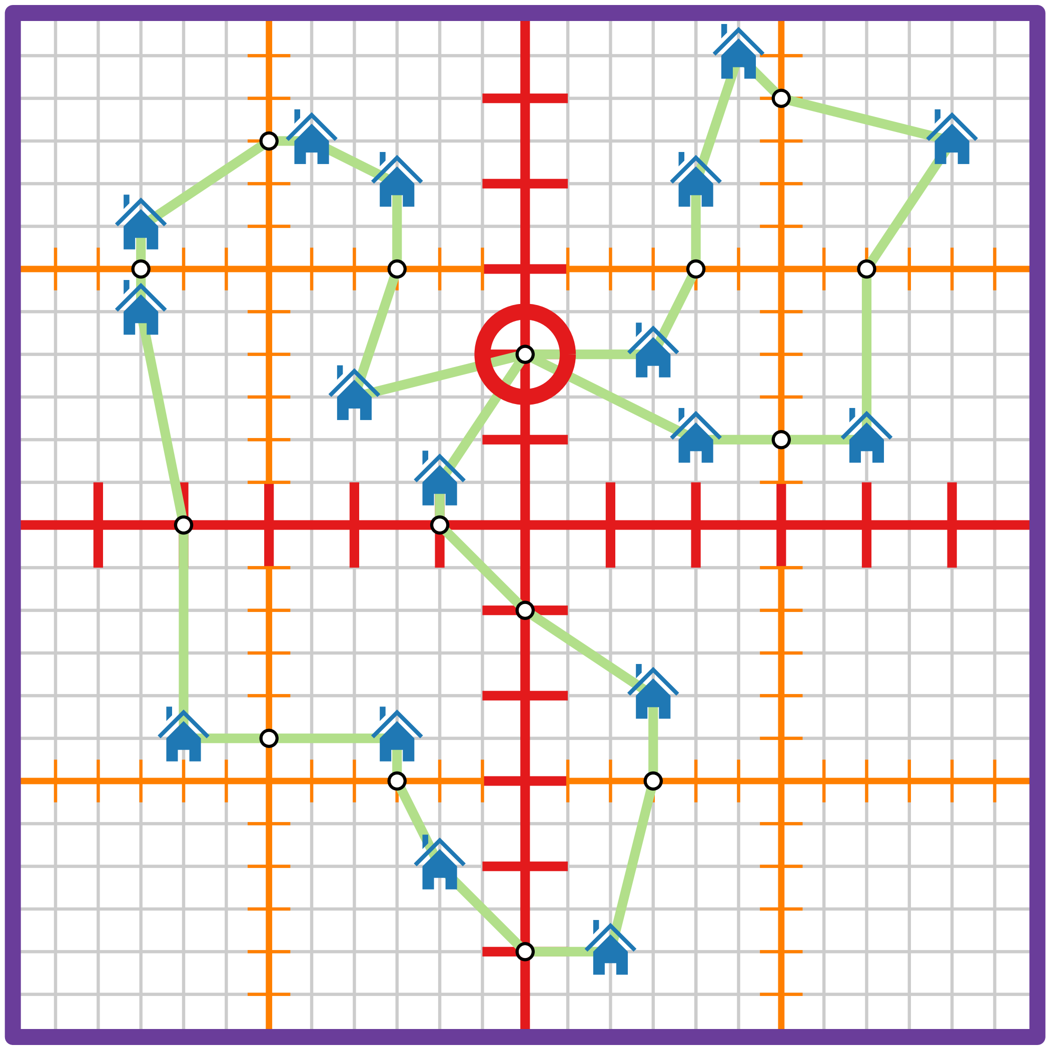
A tour is *well behaved* if

- it involves all houses and a subset of the portals,
- no edge of the tour crosses a line of the basic dissection,
- it is crossing-free.

Crossing

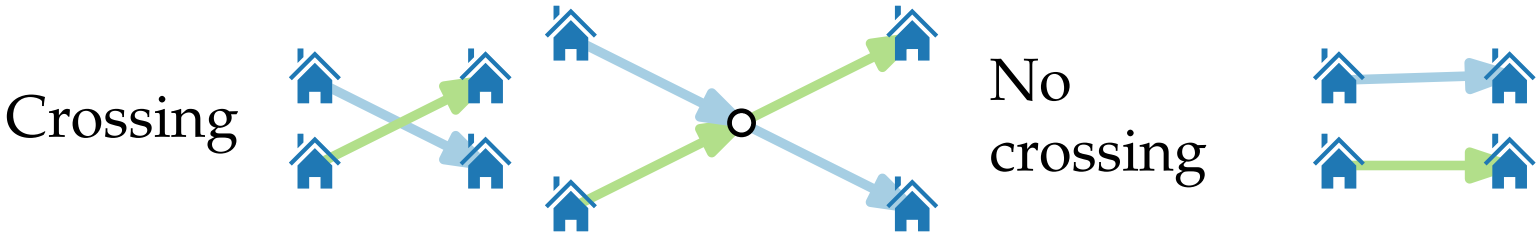


Well Behaved Tours

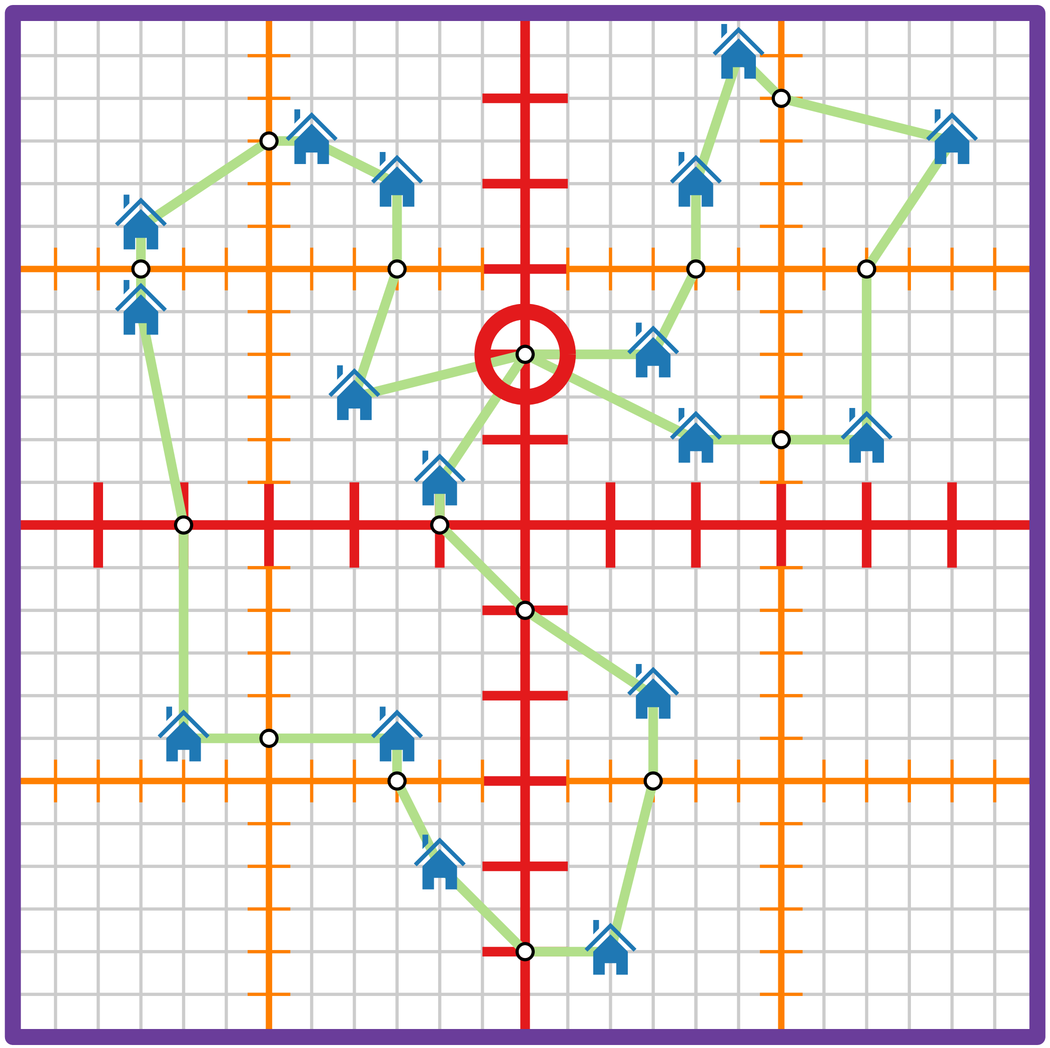


A tour is *well behaved* if

- it involves all houses and a subset of the portals,
- no edge of the tour crosses a line of the basic dissection,
- it is crossing-free.



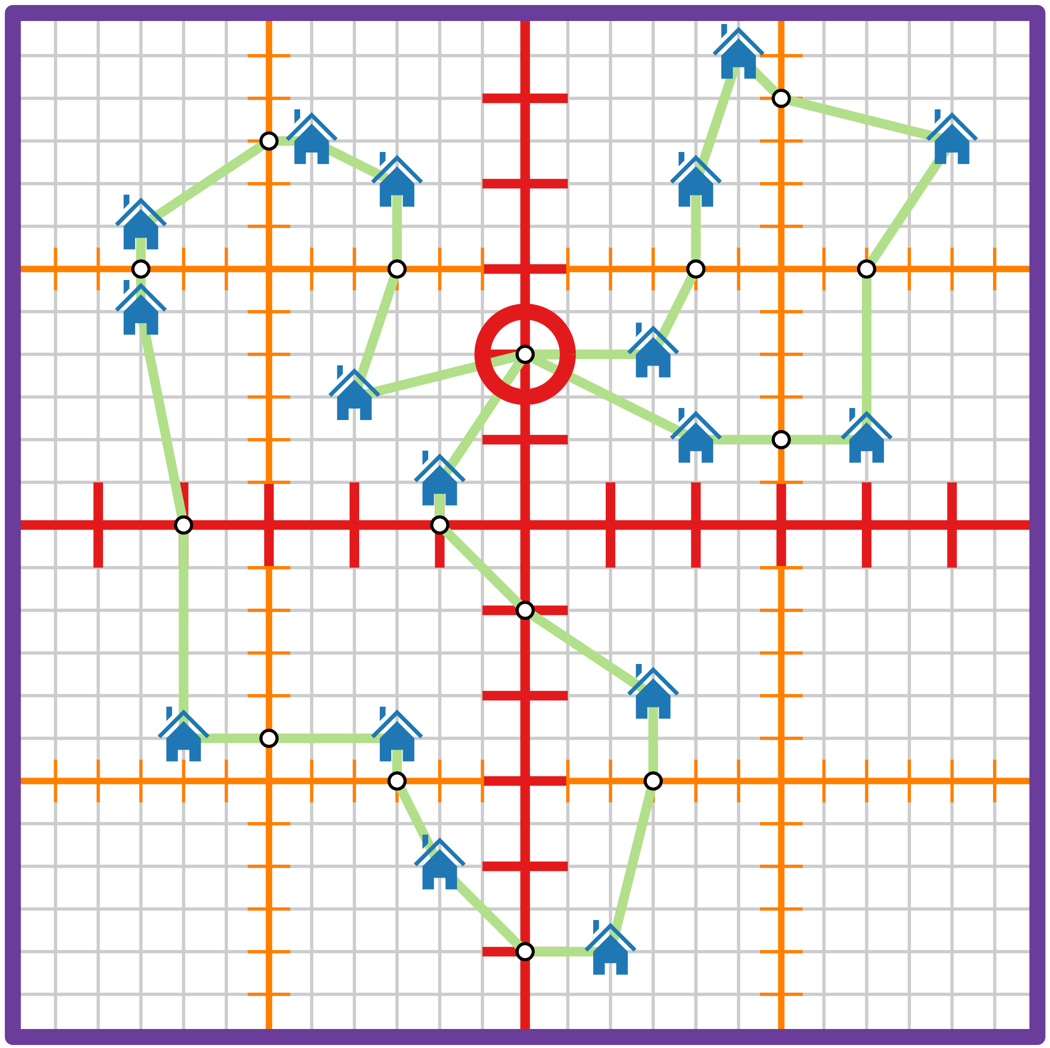
Well Behaved Tours



A tour is *well behaved* if

- it involves all houses and a subset of the portals,
- no edge of the tour crosses a line of the basic dissection,
- it is crossing-free.

Well Behaved Tours

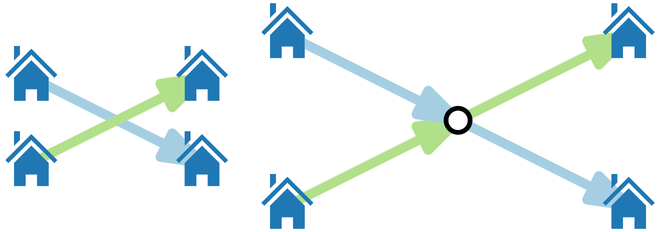


A tour is *well behaved* if

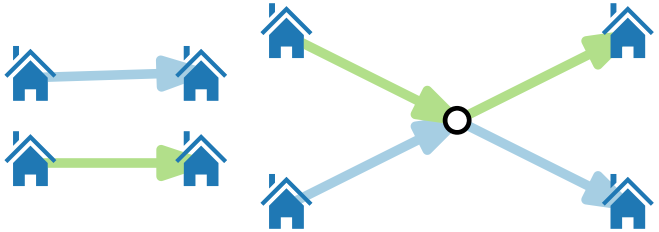
- it involves all houses and a subset of the portals,
- no edge of the tour crosses a line of the basic dissection,
- it is crossing-free.

W.l.o.g. (homework):
No portal visited more than twice

Crossing



No crossing



Computing a Well Behaved Tour

Computing a Well Behaved Tour

Lemma. An optimal well behaved tour can be computed in $2^{O(m)} = n^{O(1/\varepsilon)}$ time.

Computing a Well Behaved Tour

Lemma. An optimal well behaved tour can be computed in $2^{O(m)} = n^{O(1/\varepsilon)}$ time.

Sketch.

Computing a Well Behaved Tour

Lemma. An optimal well behaved tour can be computed in $2^{O(m)} = n^{O(1/\varepsilon)}$ time.

Sketch.
- Dynamic Programming!

Computing a Well Behaved Tour

Lemma. An optimal well behaved tour can be computed in $2^{O(m)} = n^{O(1/\varepsilon)}$ time.

Sketch.

- Dynamic Programming!
- Compute sub-structure of an optimal tour for each square in the dissection tree.

Computing a Well Behaved Tour

Lemma. An optimal well behaved tour can be computed in $2^{O(m)} = n^{O(1/\varepsilon)}$ time.

Sketch.

- Dynamic Programming!
- Compute sub-structure of an optimal tour for each square in the dissection tree.
- These solutions can be efficiently propagated bottom-up through the dissection tree.

Approximation Algorithms

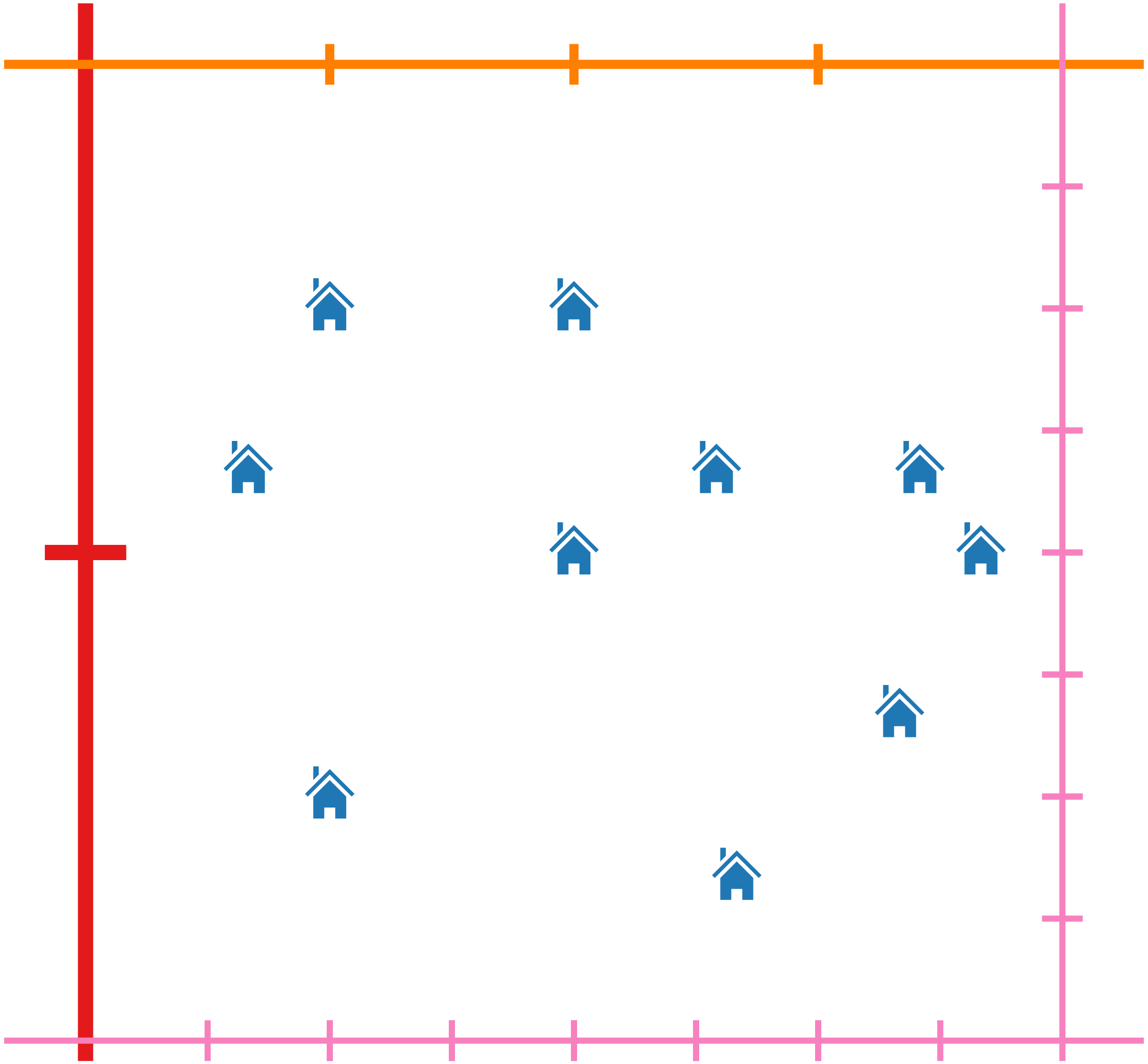
Lecture 9:
PTAS for EuclideanTSP

Part IV:
Dynamic Program

Joachim Spoerhase

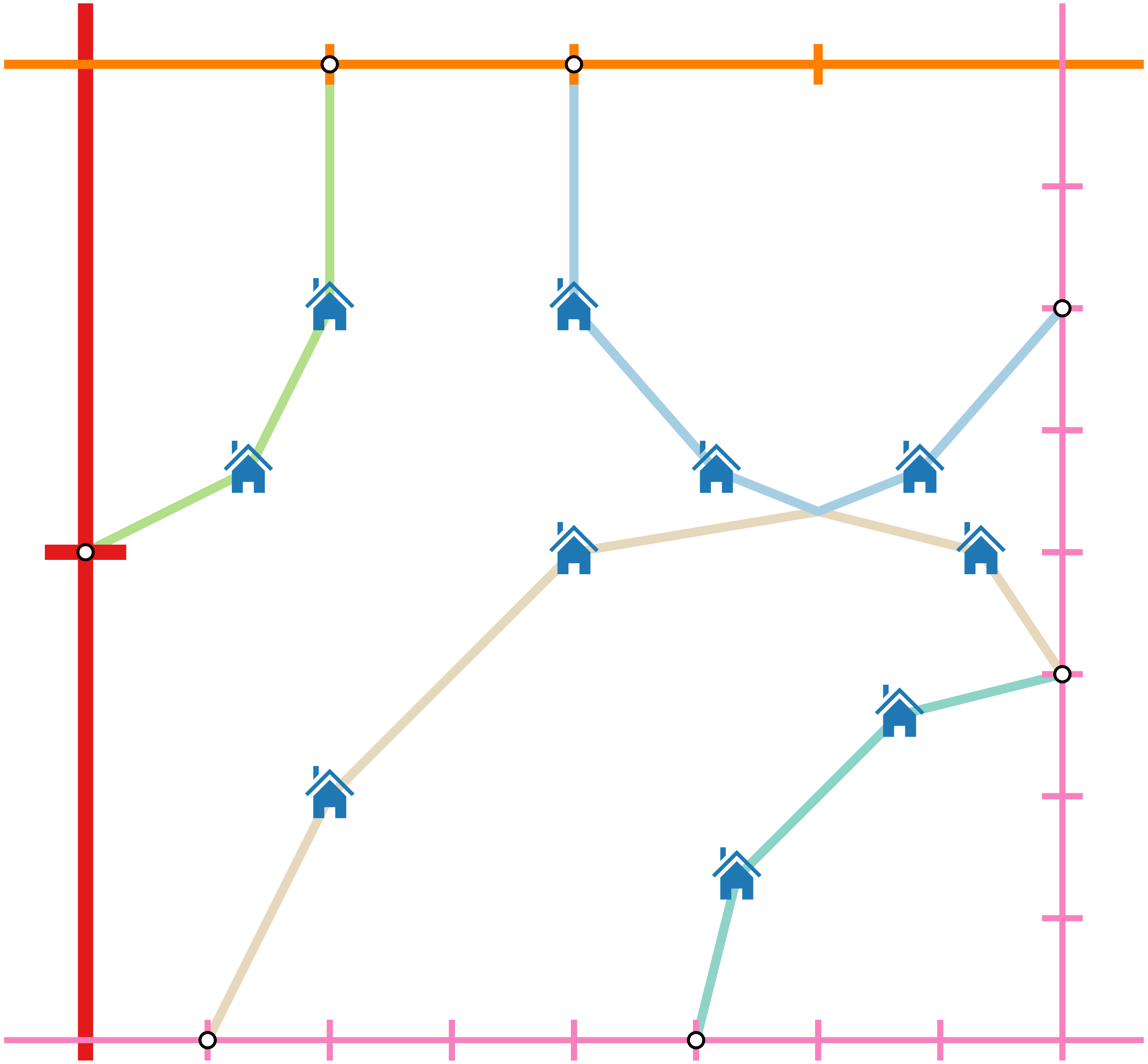
Winter 2021/22

Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

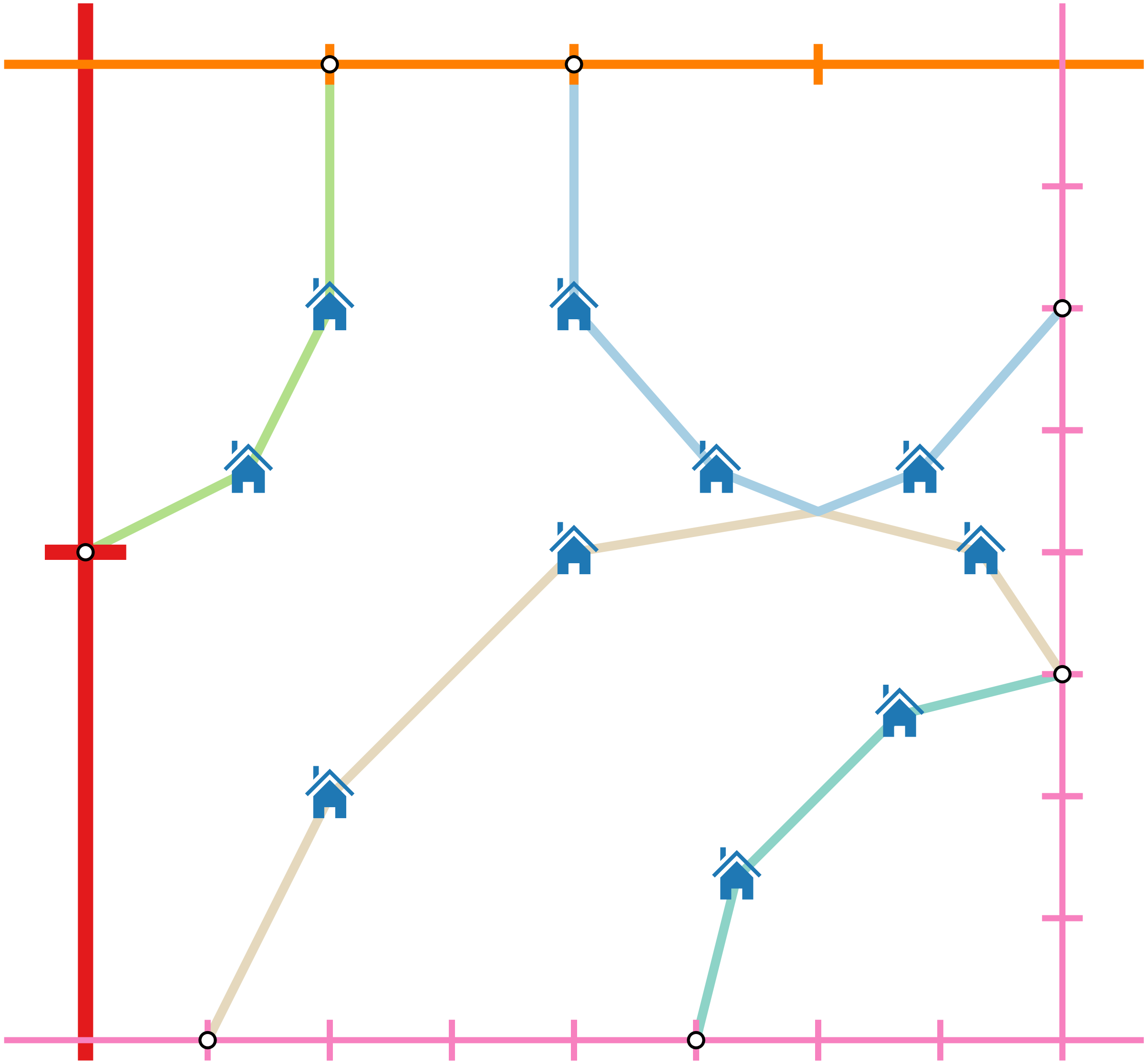
Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q

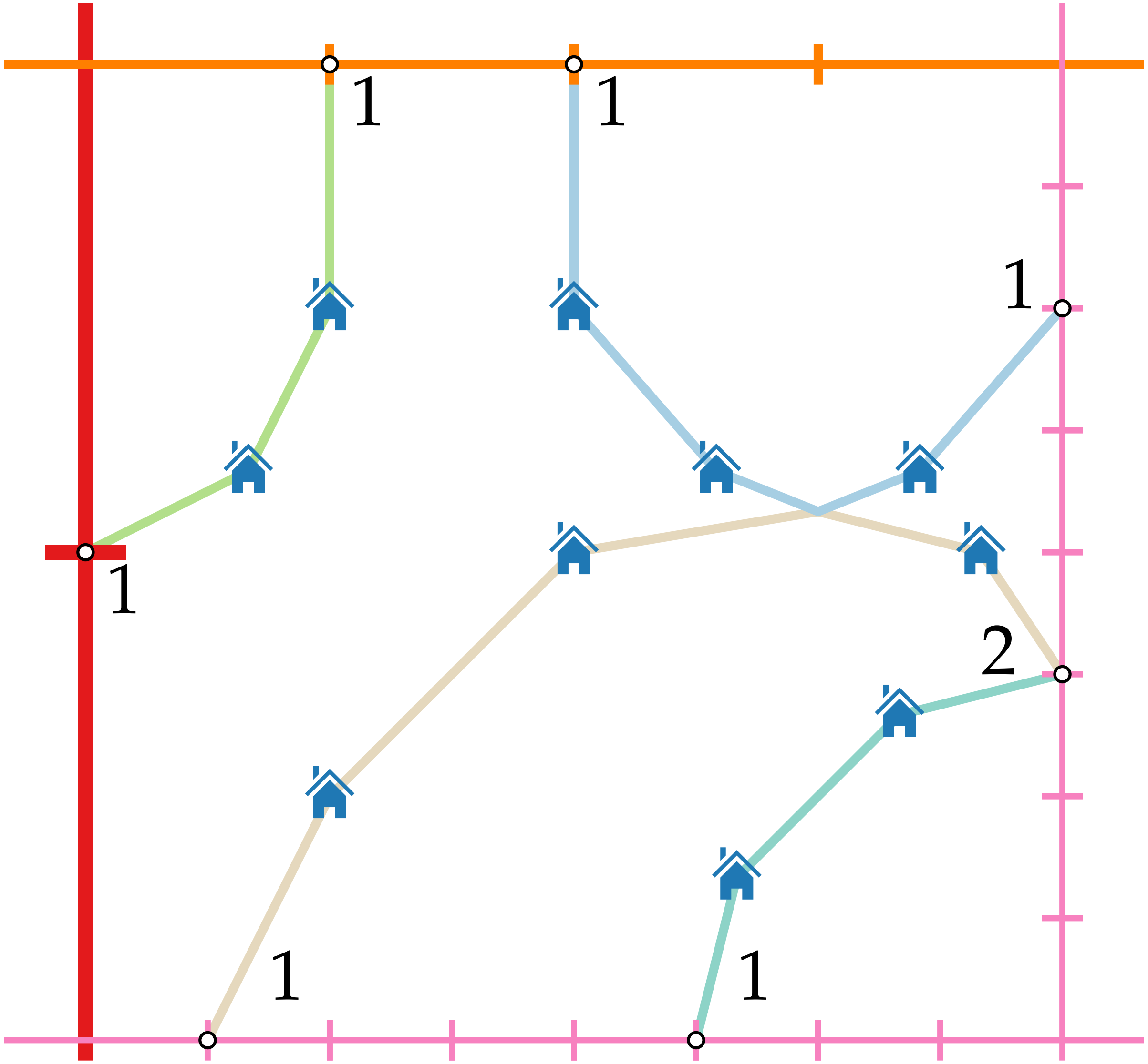
Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover

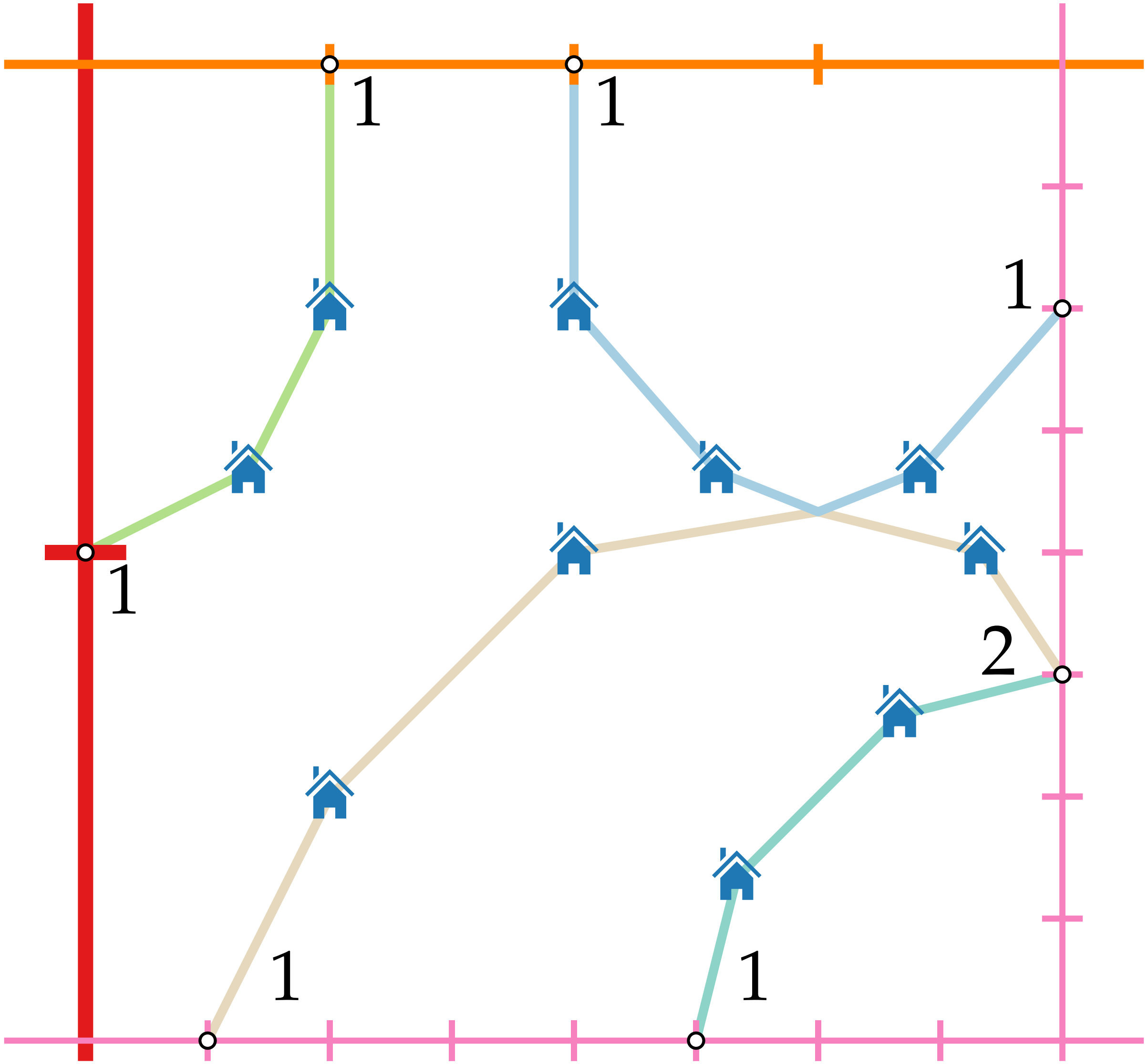
Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover

Dynamic Program (I)

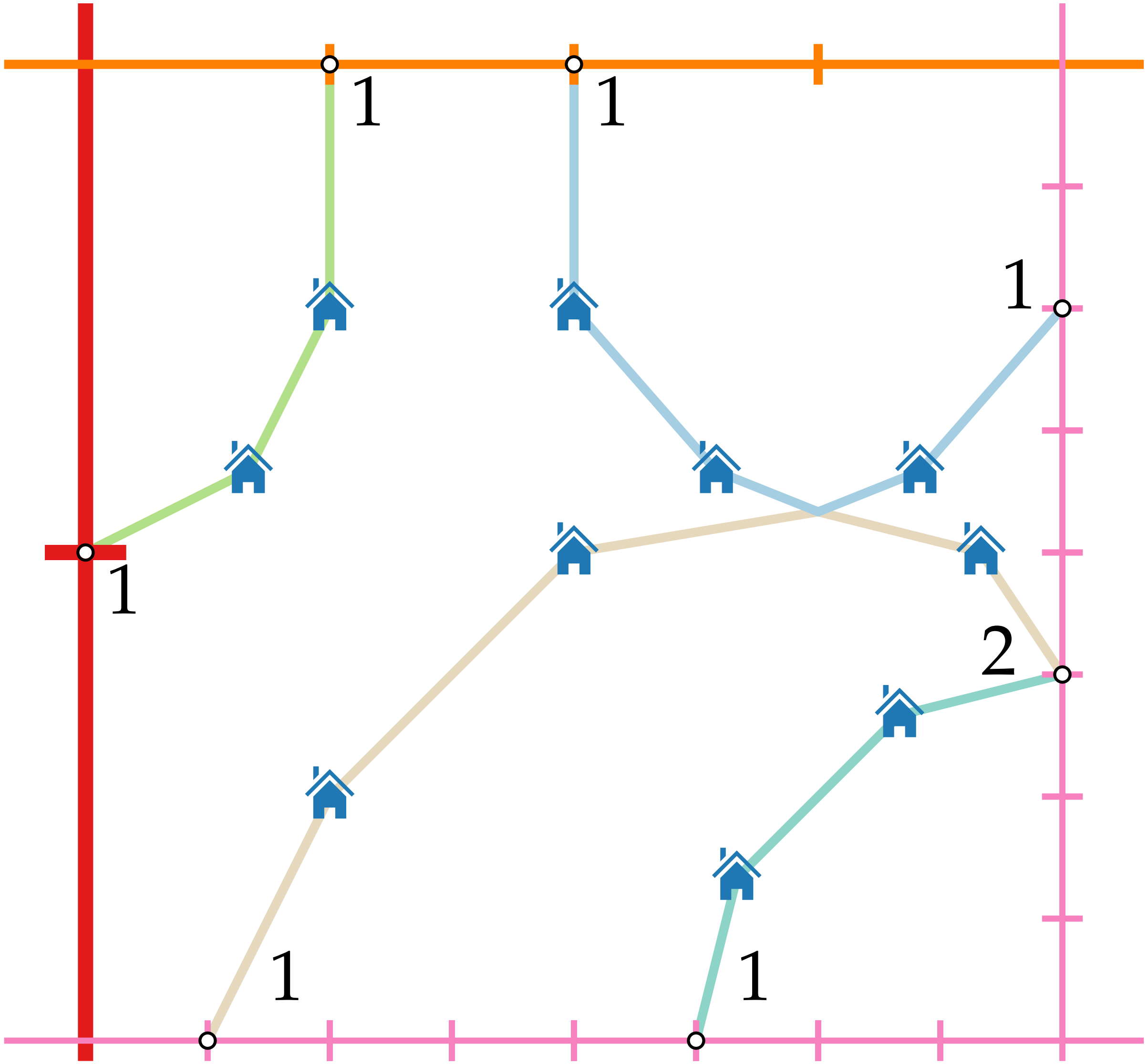


Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover

$\Rightarrow$ max. possibilities

Dynamic Program (I)

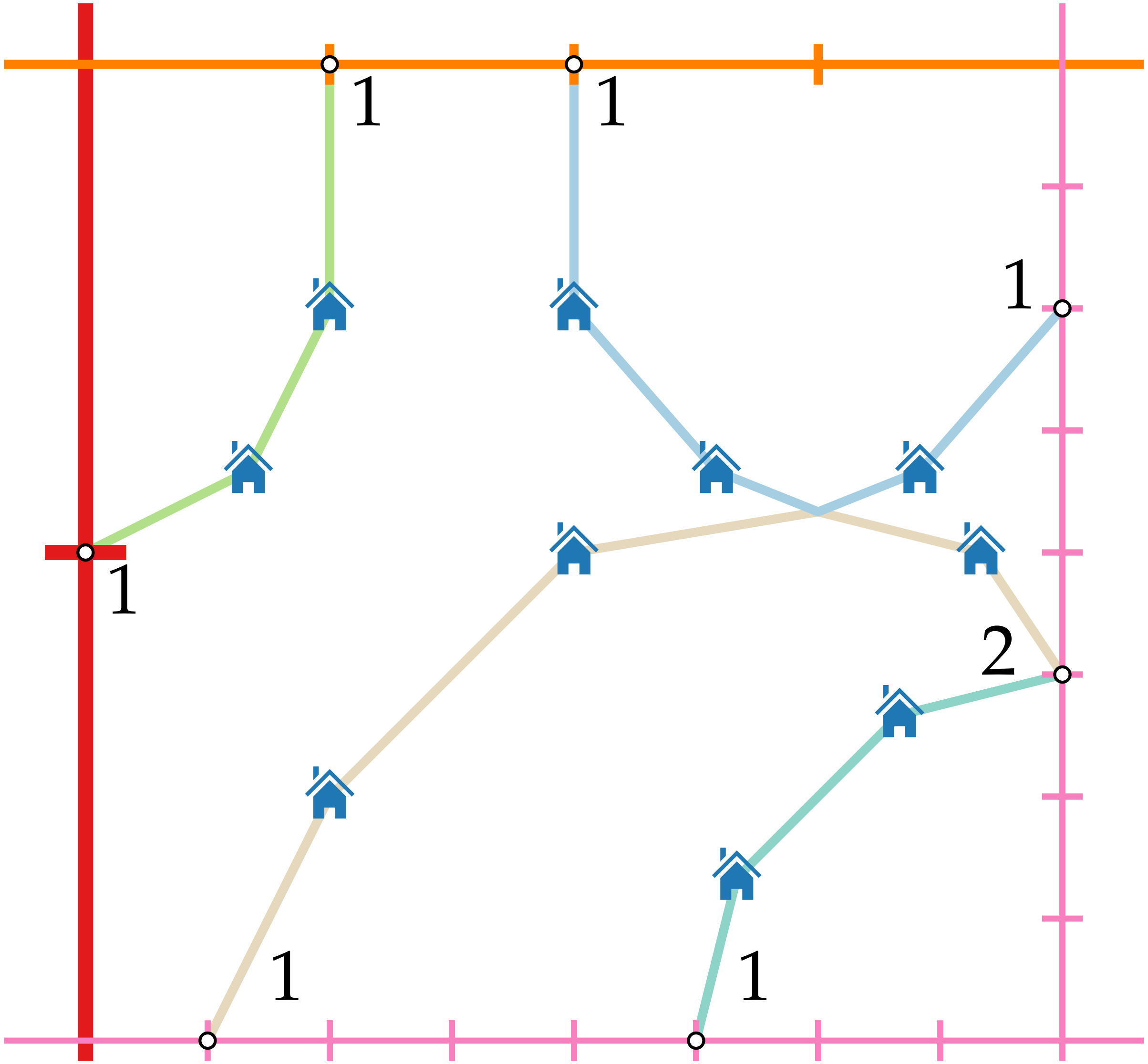


Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover

$\Rightarrow$ max. $3^{4m} =$ possibilities

Dynamic Program (I)



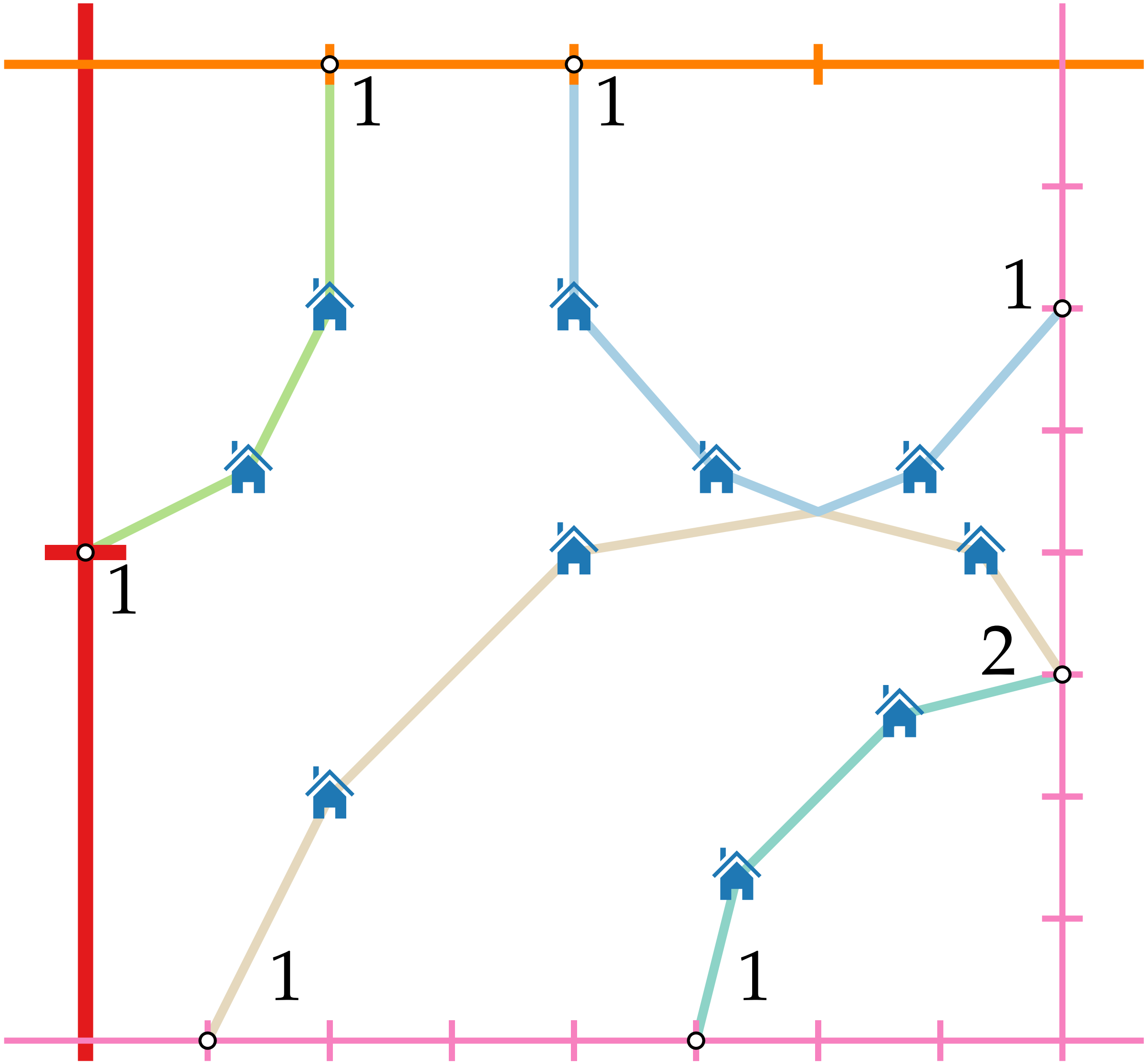
Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover

$\Rightarrow$ max. $3^{4m} =$ possibilities

$m = O((\log n)/\varepsilon)$

Dynamic Program (I)



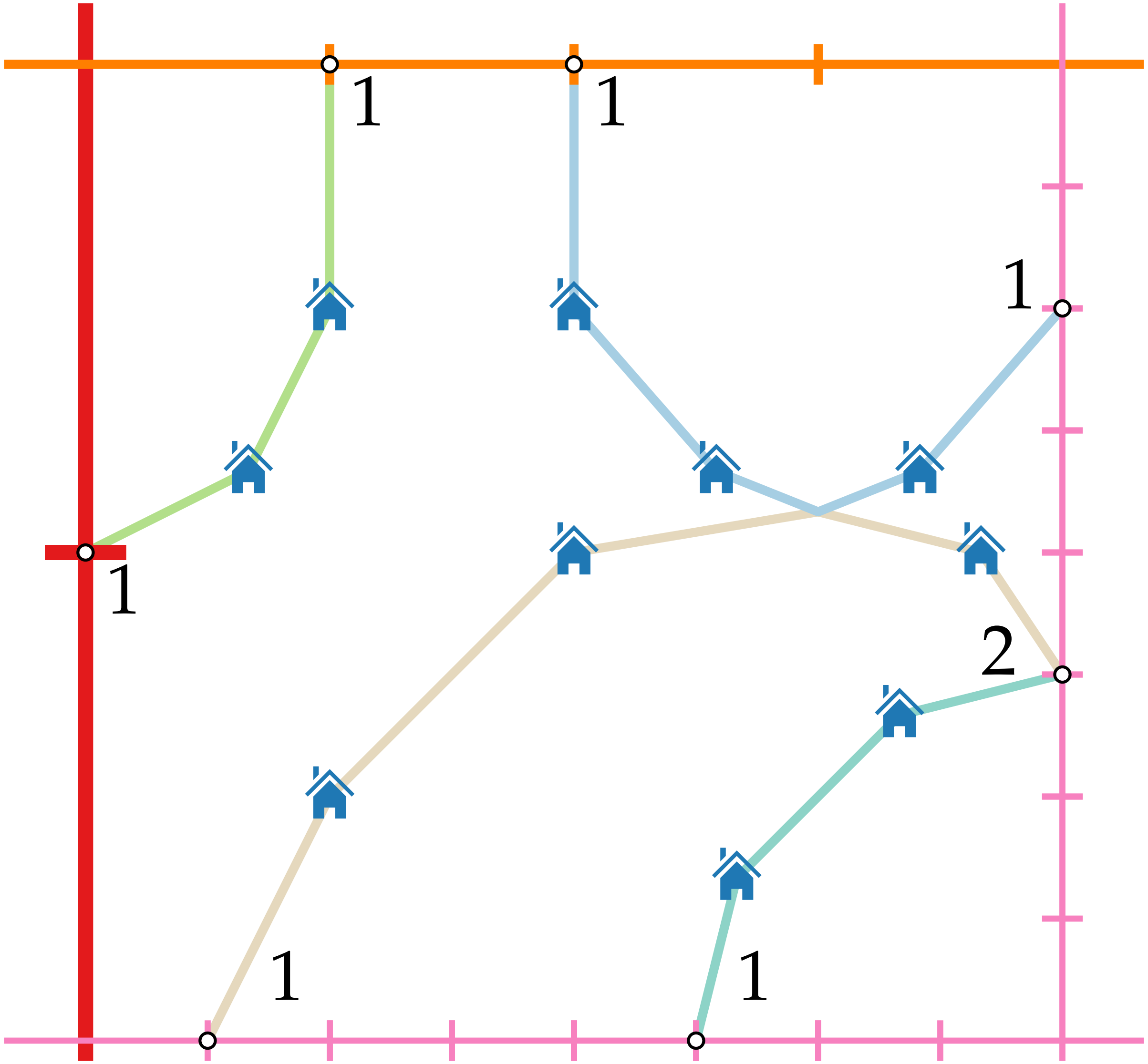
Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover

$\Rightarrow$ max. $3^{4m} = 3^{O((\log n)/\varepsilon)}$ possibilities

$m = O((\log n)/\varepsilon)$

Dynamic Program (I)



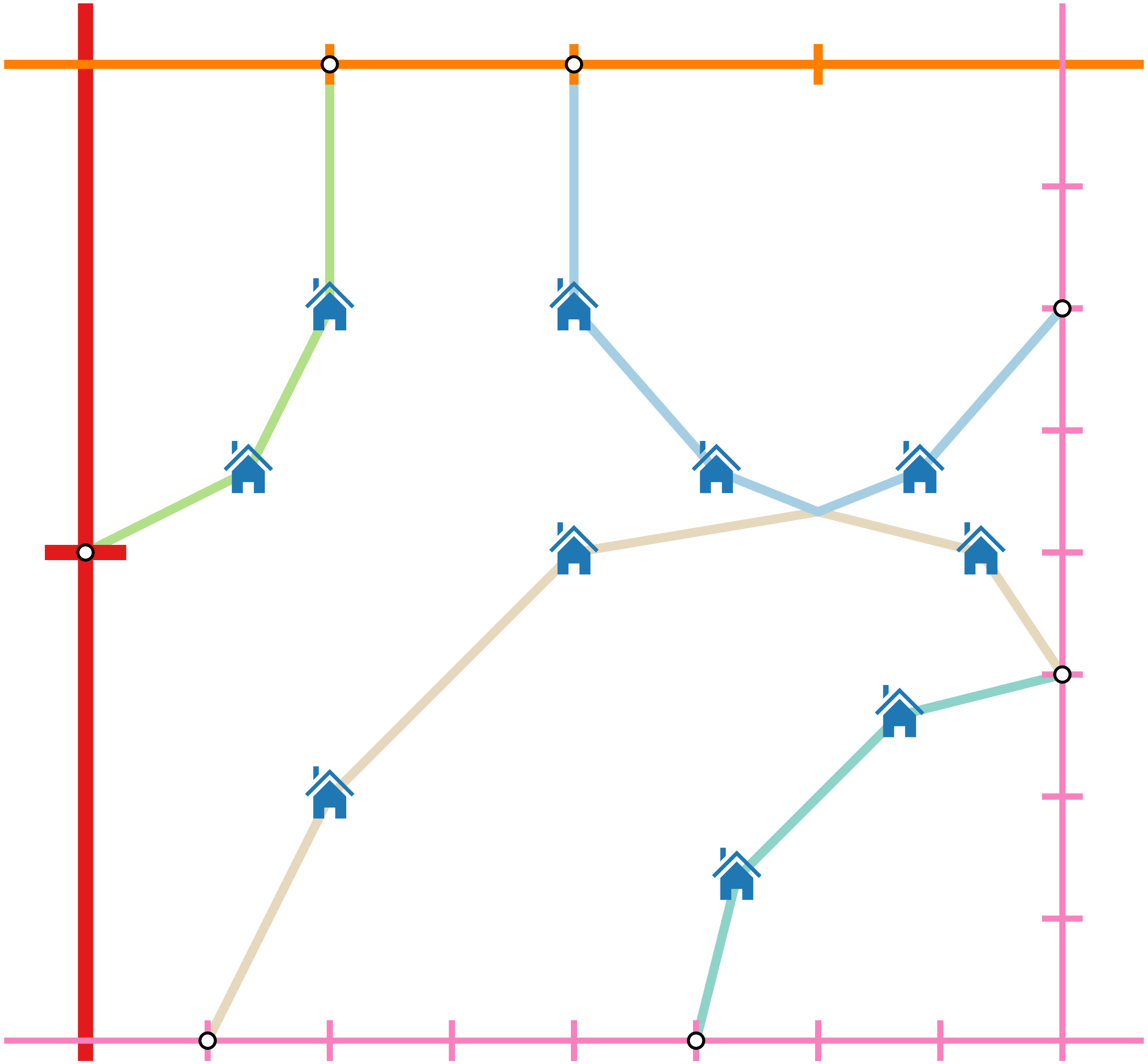
Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover

$\Rightarrow$ max. $3^{4m} = 3^{O((\log n)/\varepsilon)} = n^{O(1/\varepsilon)}$ possibilities

$m = O((\log n)/\varepsilon)$

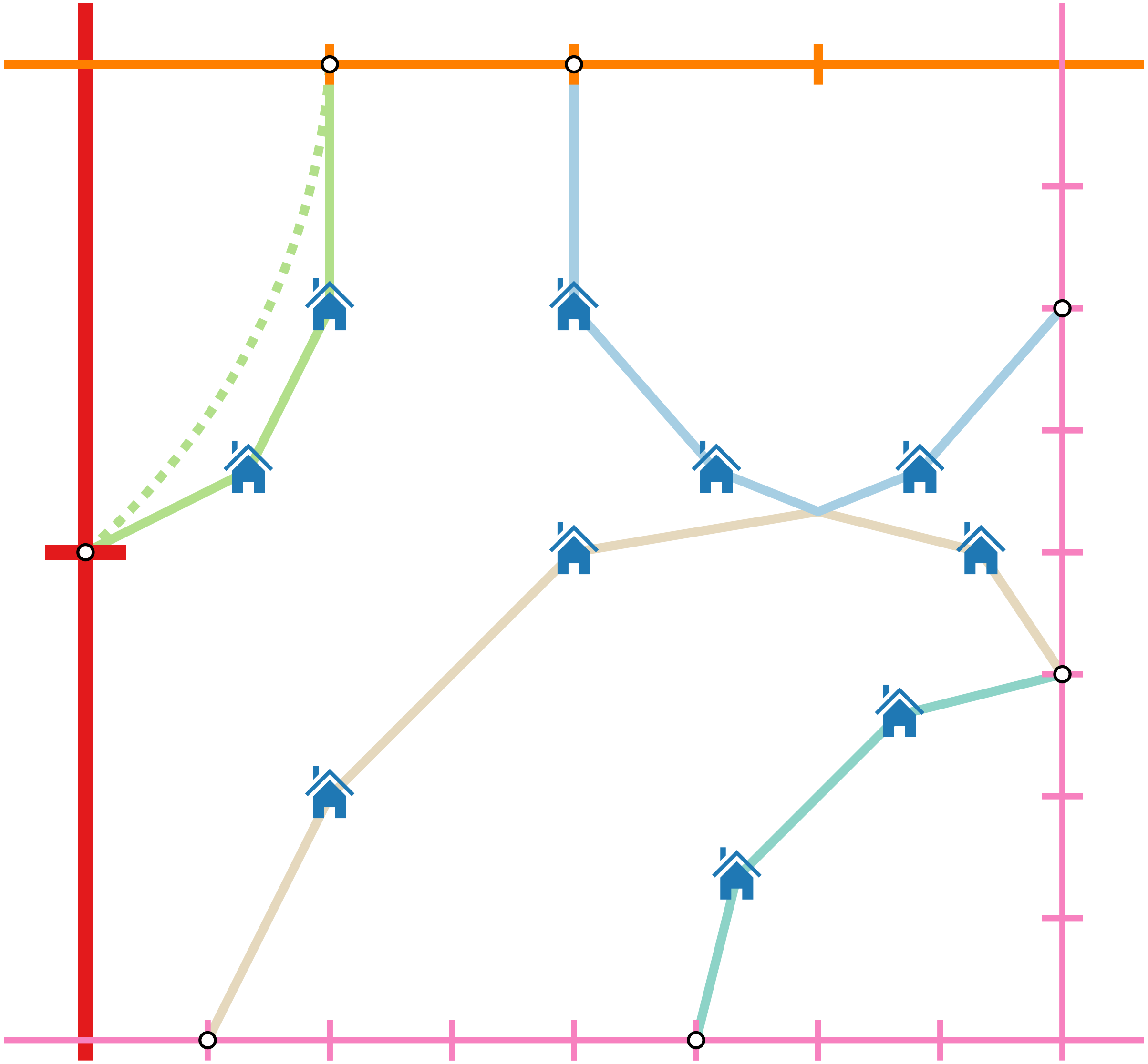
Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals

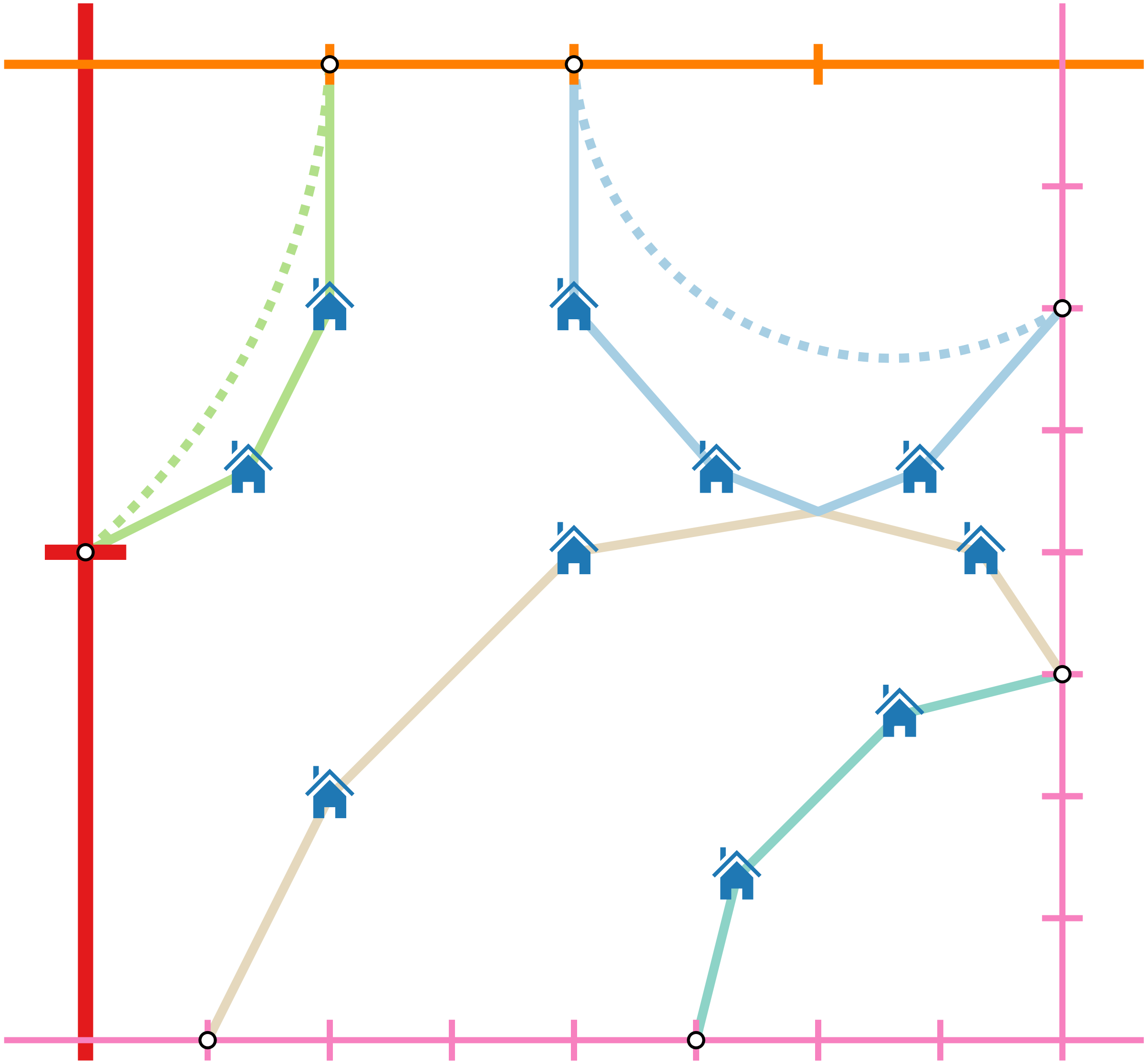
Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals

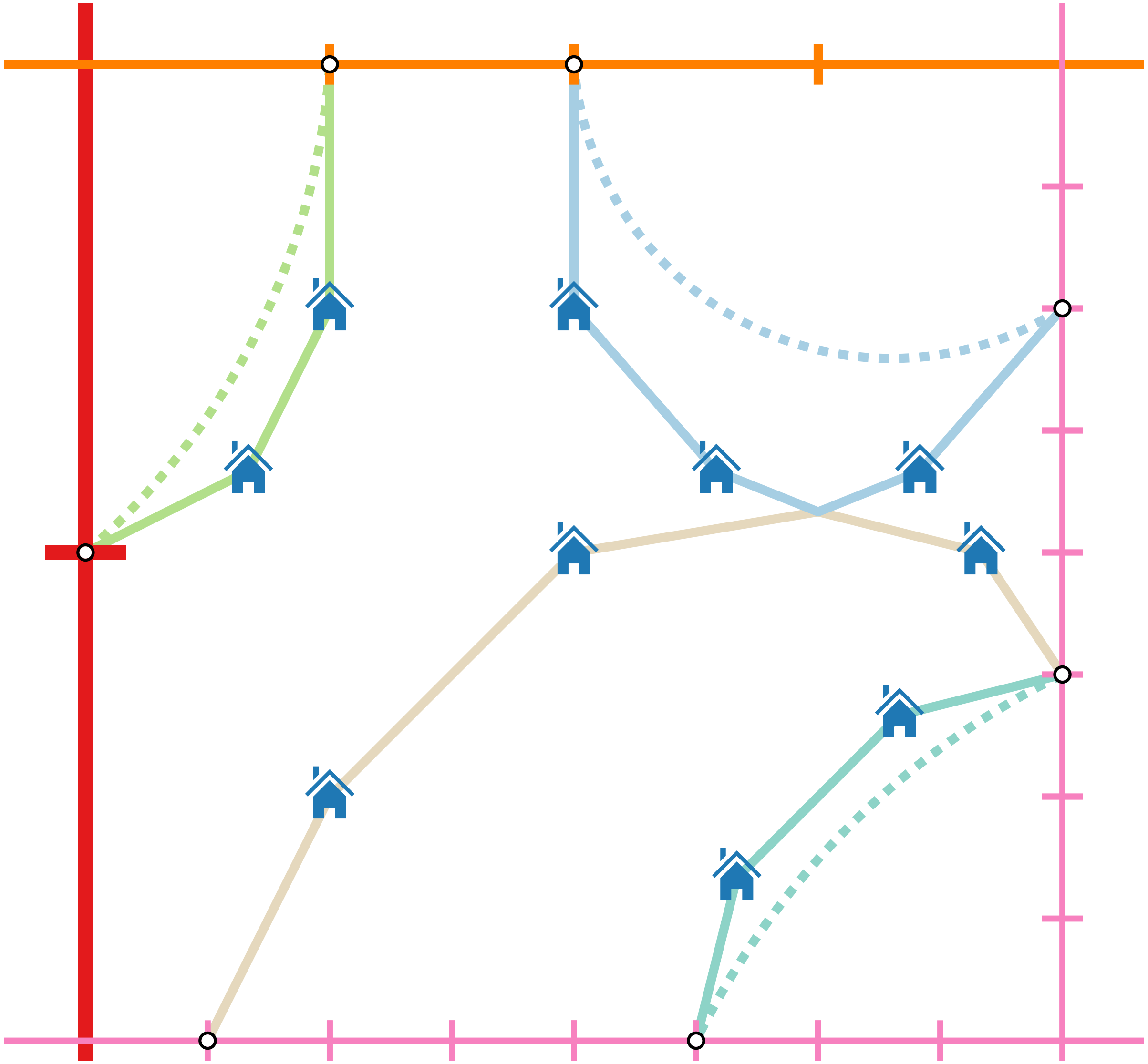
Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals

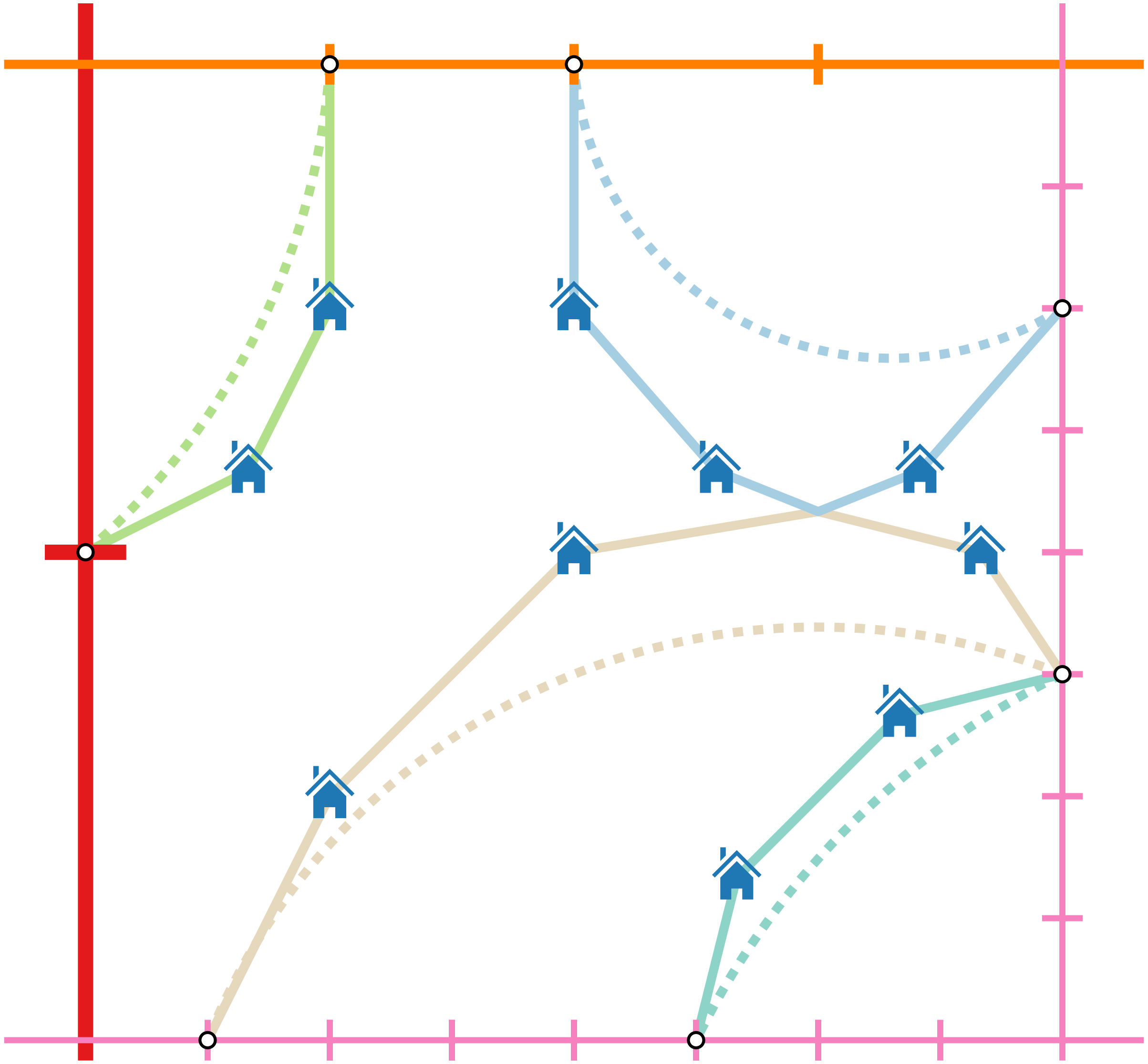
Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals

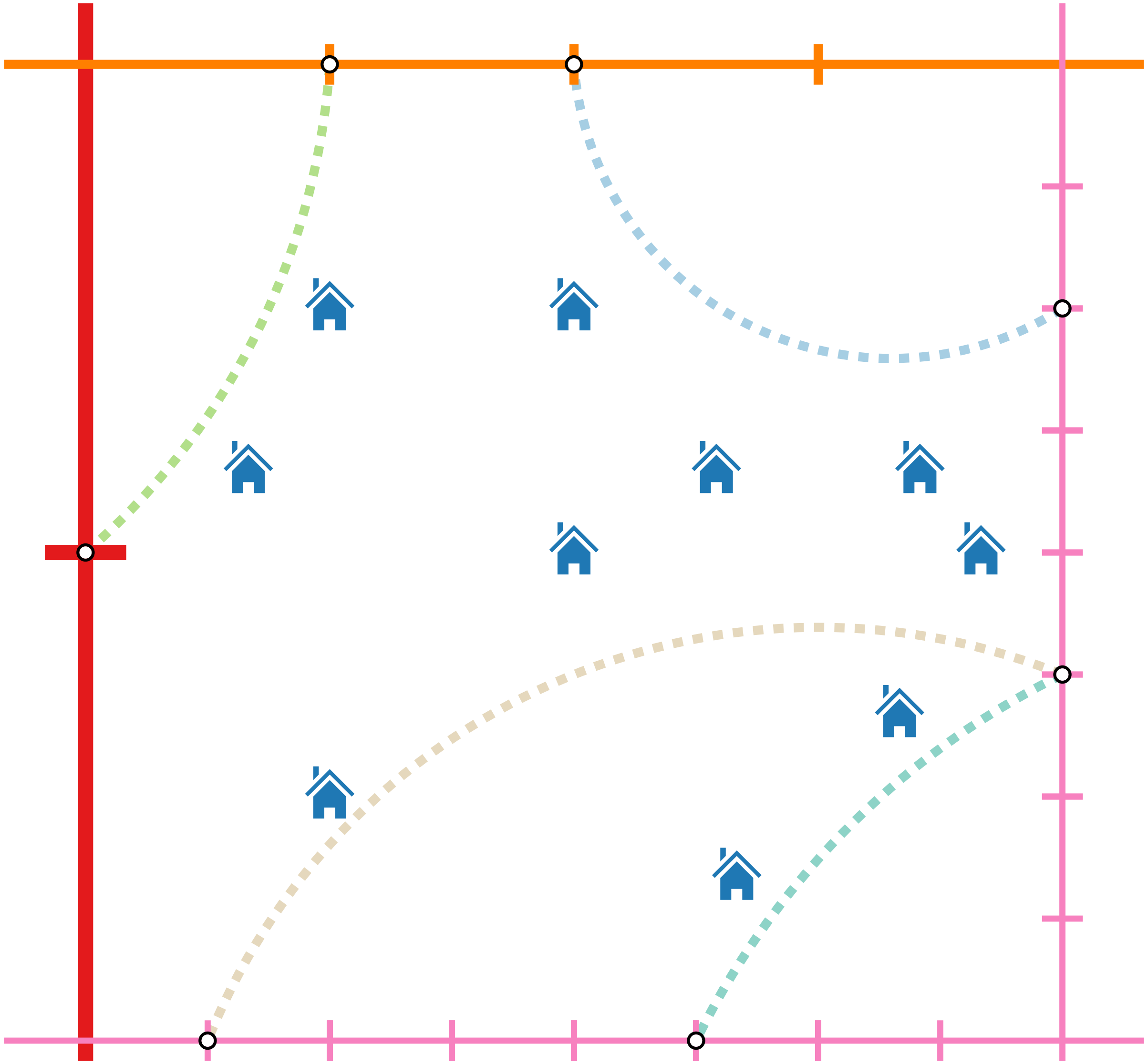
Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals

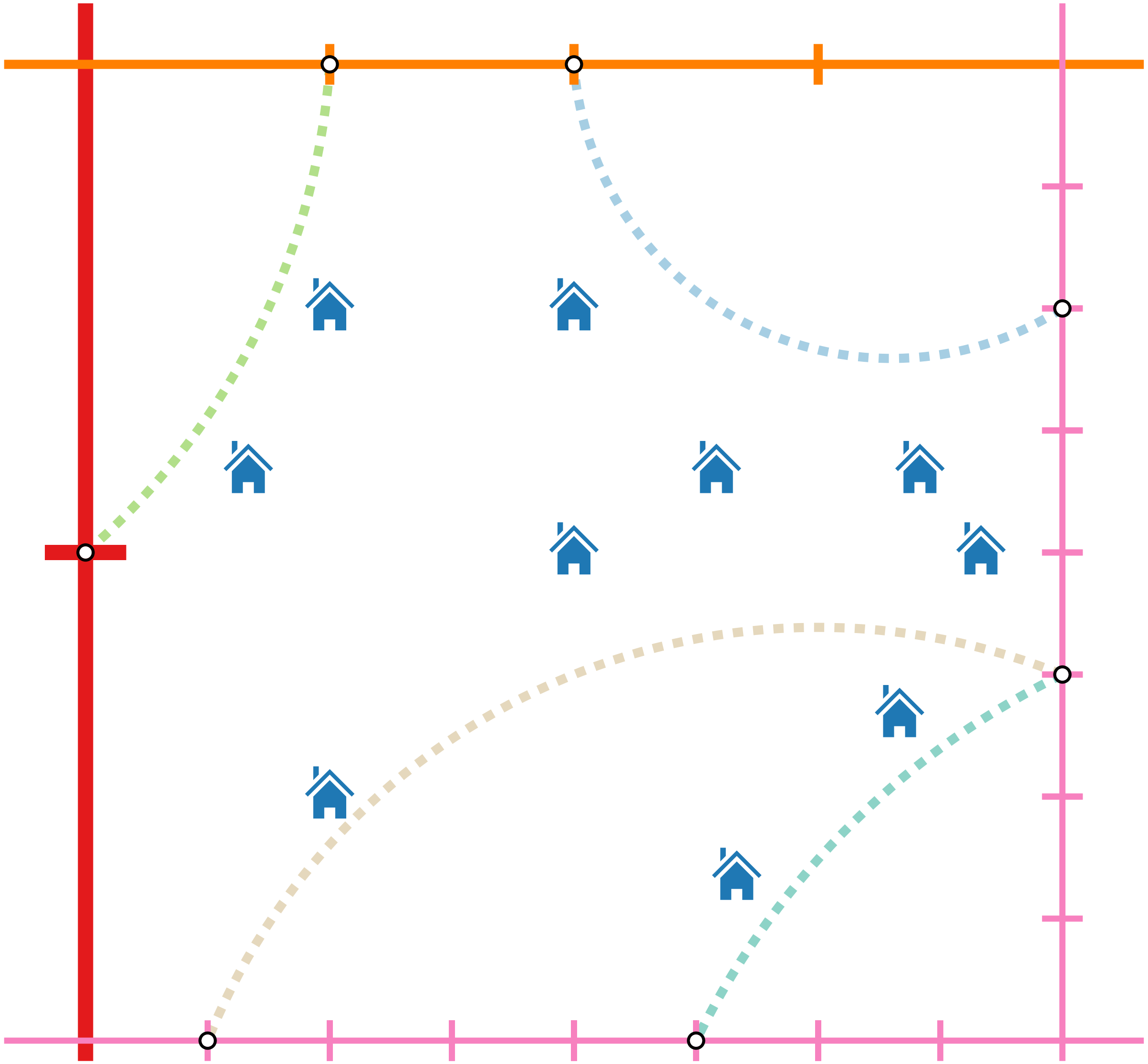
Dynamic Program (I)



Each well behaved tour induces the following in each square Q of the dissection:

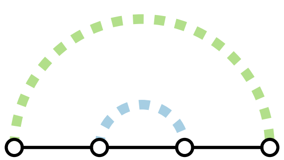
- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals

Dynamic Program (I)

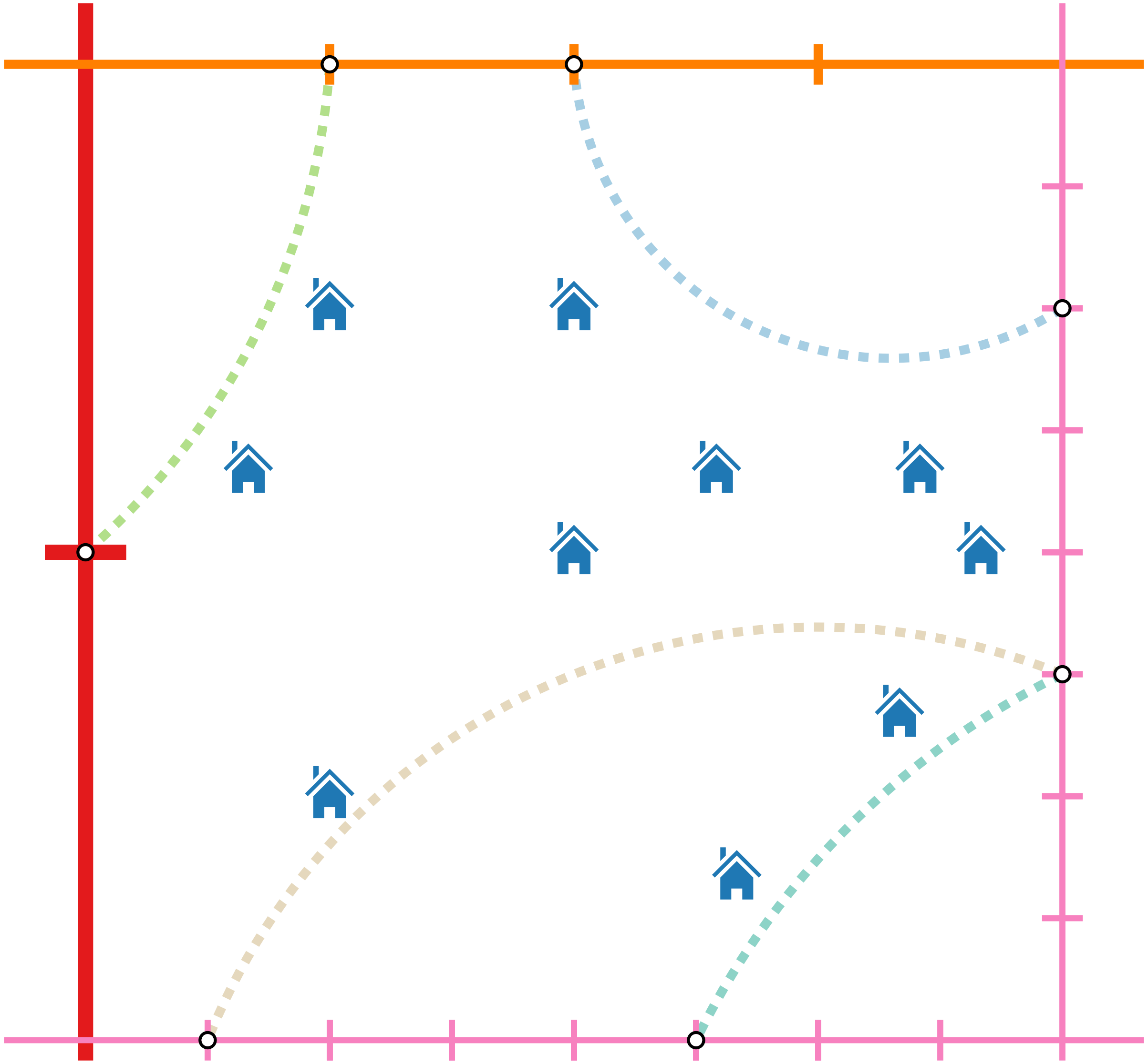


Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals

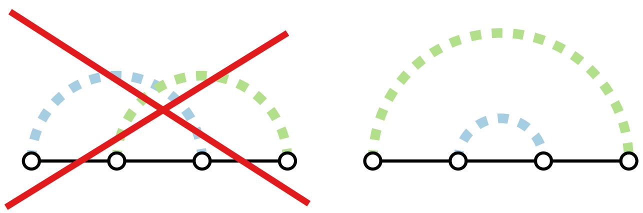


Dynamic Program (I)

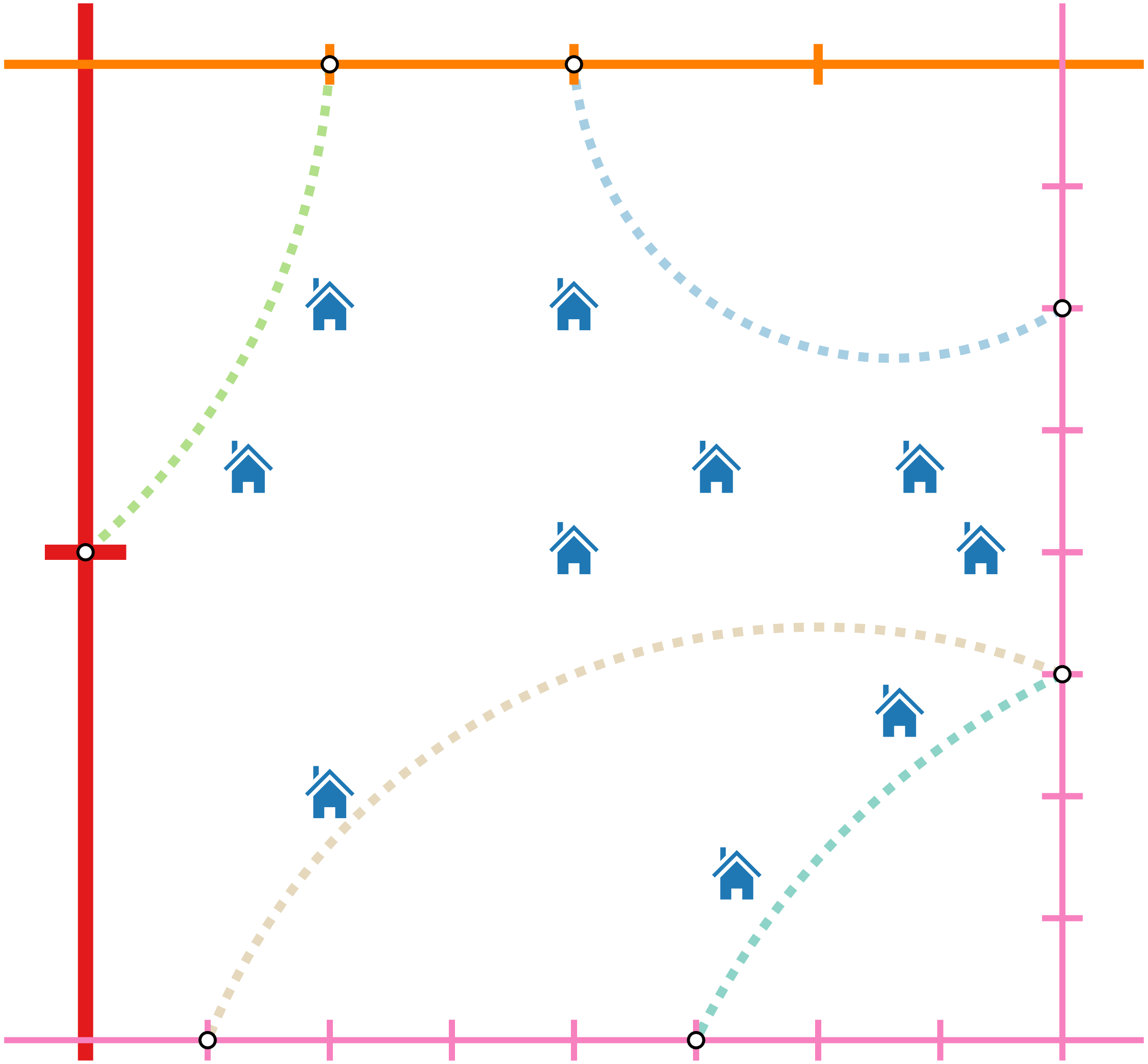


Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals

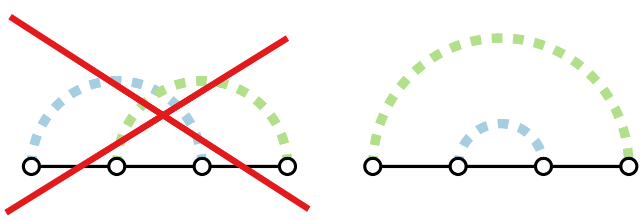


Dynamic Program (I)



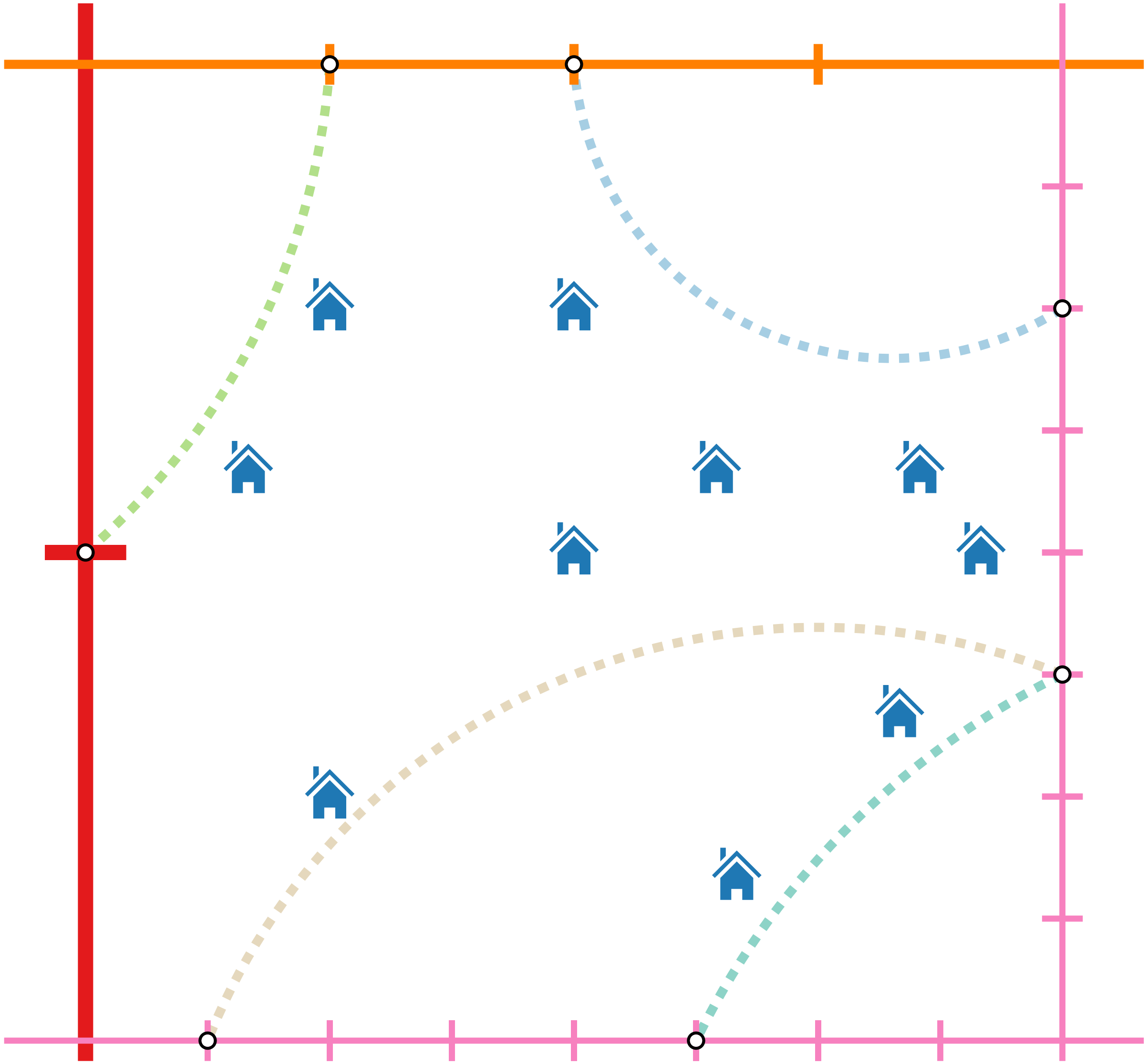
Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals



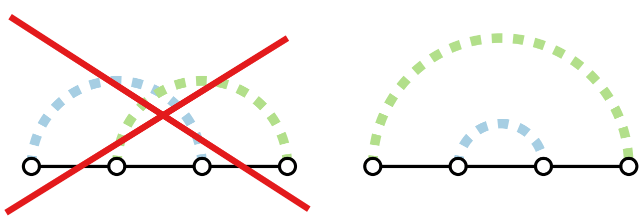
$\Rightarrow$ max.

Dynamic Program (I)



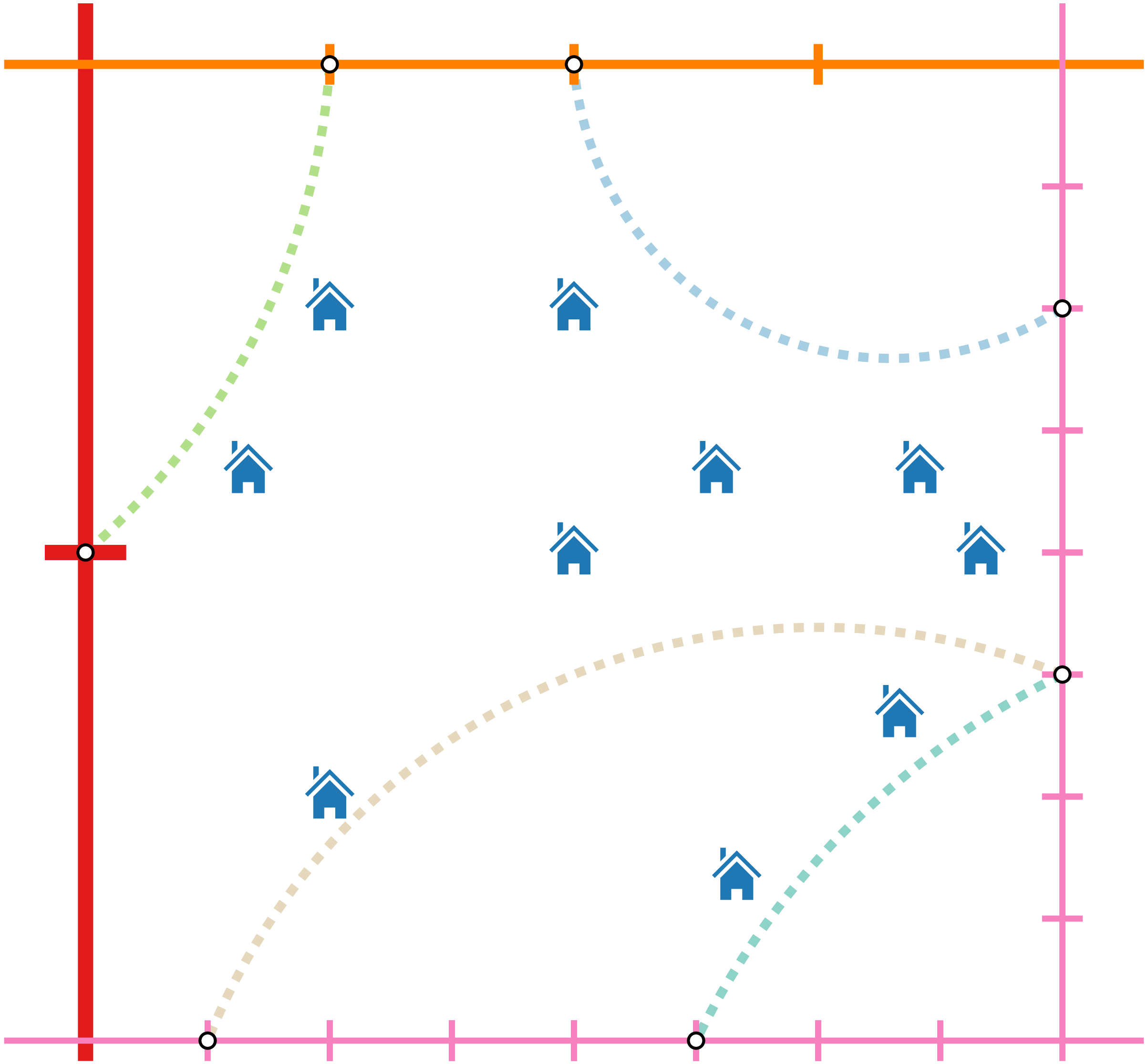
Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals



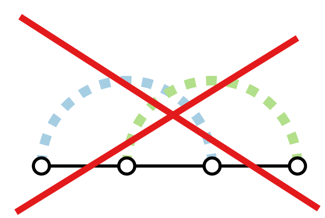
$\Rightarrow$ max. $\underbrace{}_{\text{\#visit vectors}}$

Dynamic Program (I)



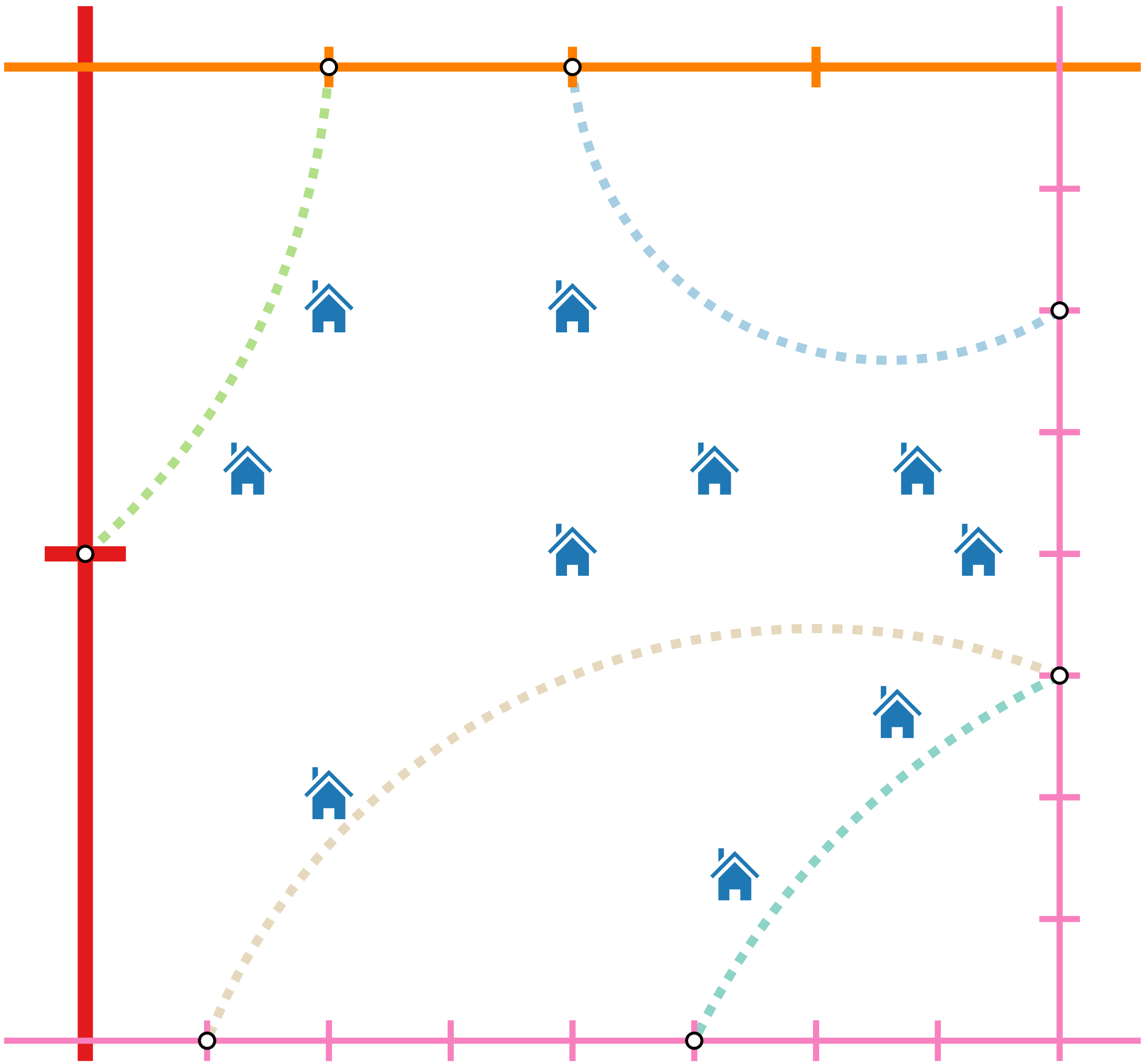
Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals



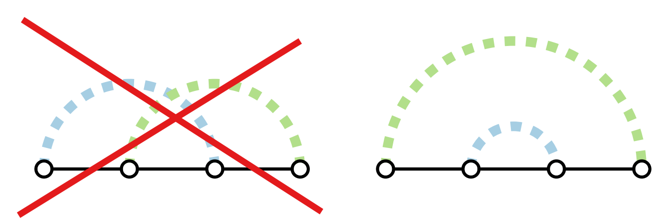
$$\Rightarrow \text{max.} \underbrace{\qquad\qquad}_{\text{\#visit vectors}} \times \underbrace{\qquad\qquad}_{\text{\#real. pairings}}$$

Dynamic Program (I)



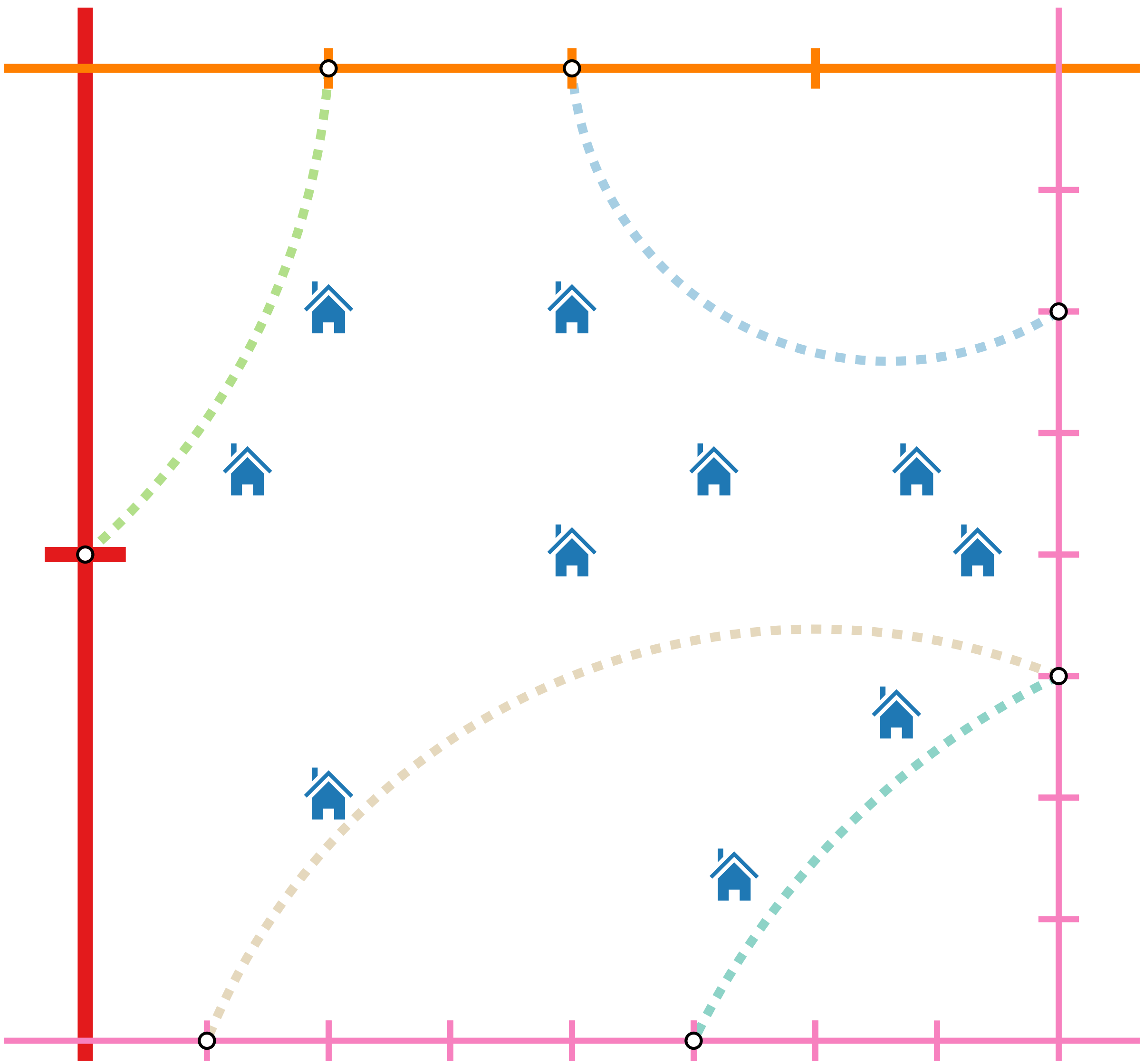
Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals



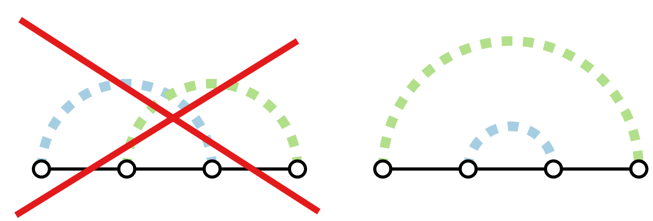
$$\Rightarrow \text{max. } \underbrace{n^{O(1/\varepsilon)}}_{\text{\#visit vectors}} \times \underbrace{\qquad}_{\text{\#real. pairings}}$$

Dynamic Program (I)



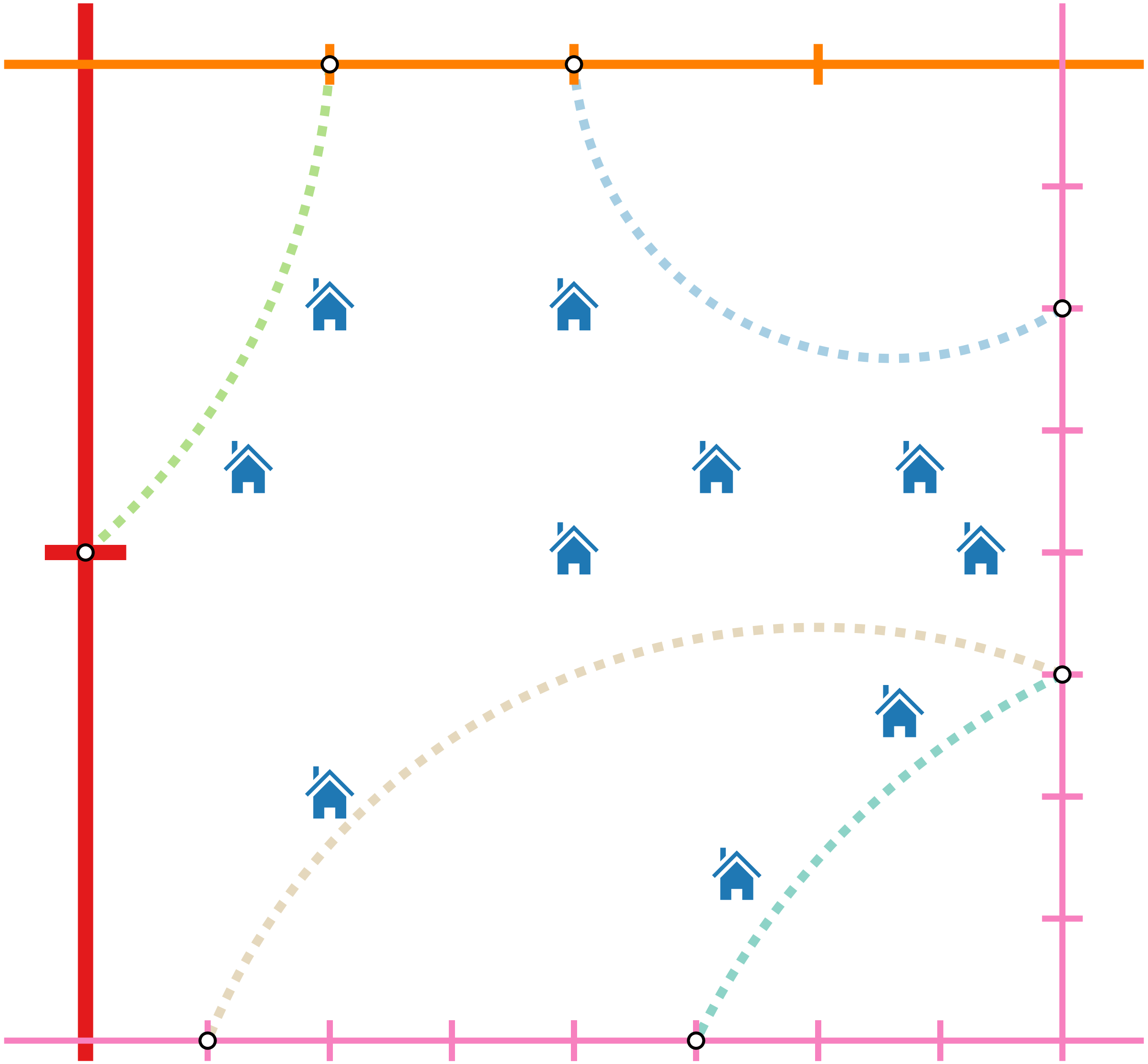
Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals



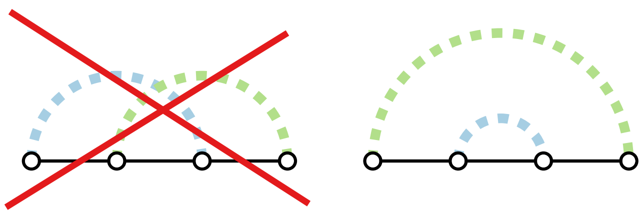
$$\Rightarrow \text{max.} \underbrace{n^{O(1/\varepsilon)}}_{\text{\#visit vectors}} \times \underbrace{2^{O(m)}}_{\text{\#real. pairings}}$$

Dynamic Program (I)



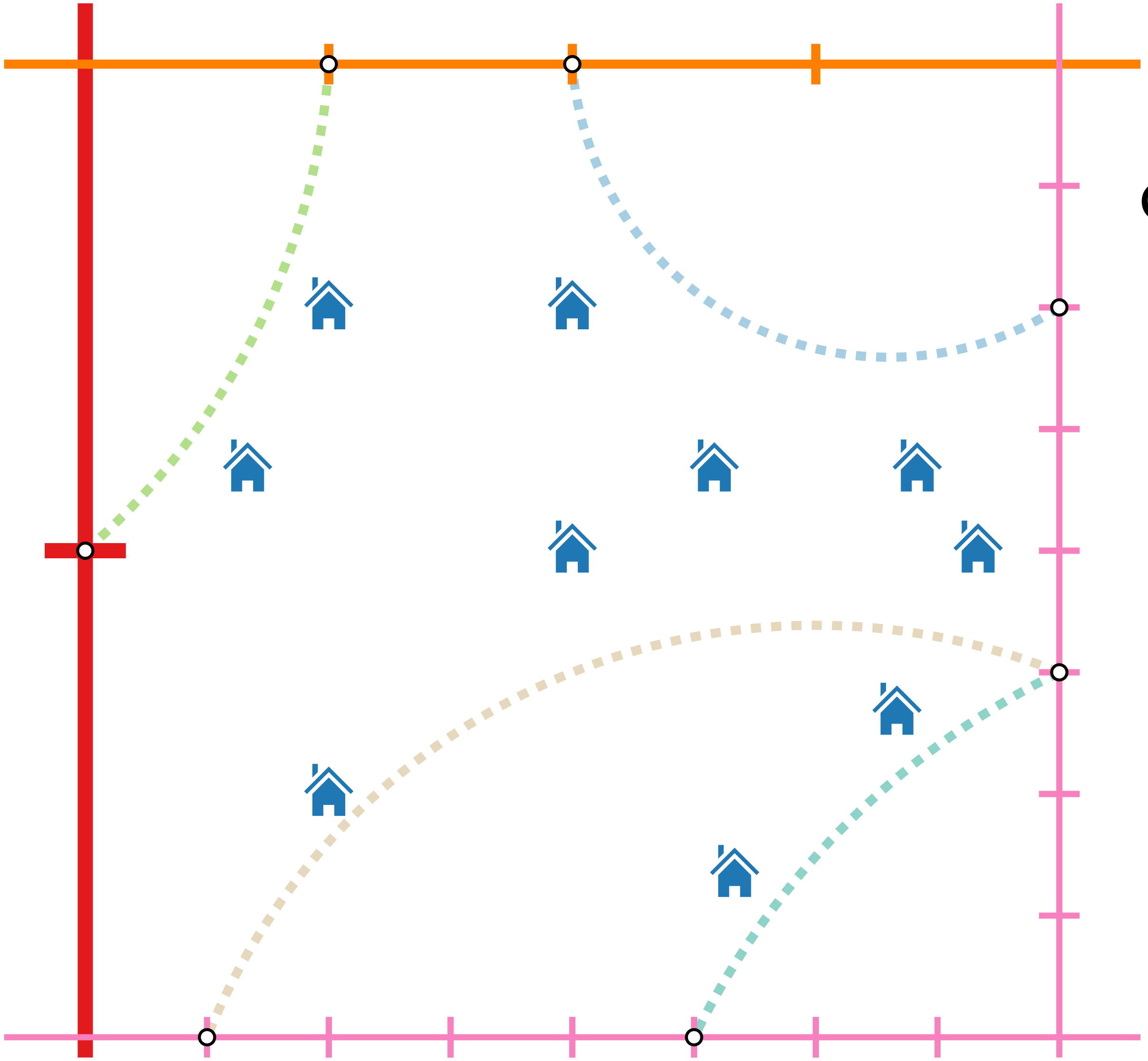
Each well behaved tour induces the following in each square Q of the dissection:

- A path cover of the houses in Q
- Each portal of Q is visited 0,1 or 2 times by this path cover
- A **crossing-free pairing** of the visited portals



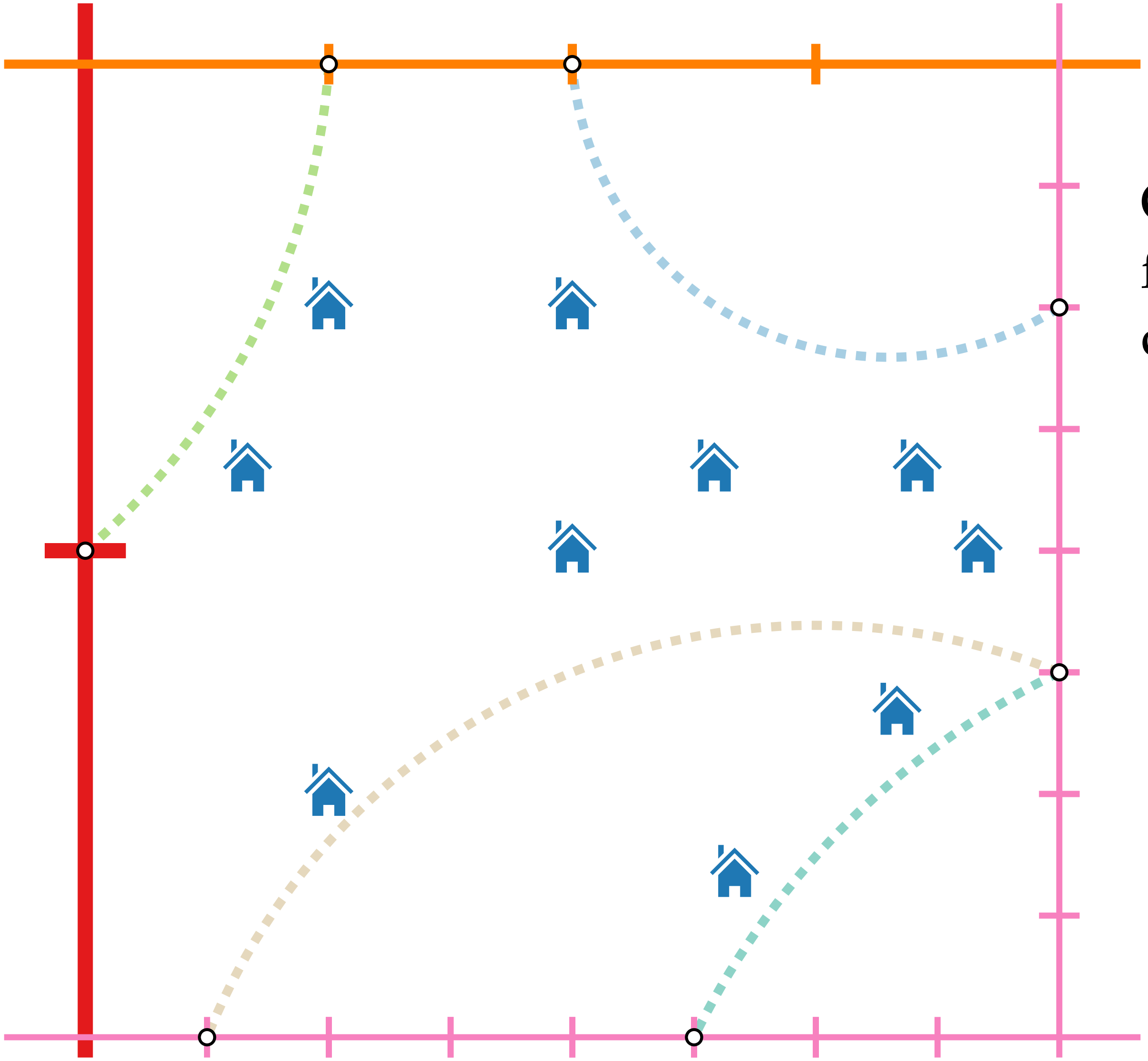
$$\Rightarrow \text{max. } \underbrace{n^{O(1/\varepsilon)}}_{\text{\#visit vectors}} \times \underbrace{2^{O(m)}}_{\text{\#real. pairings}} = n^{O(1/\varepsilon)} \text{ crossing-free pairings}$$

Dynamic Program (II)



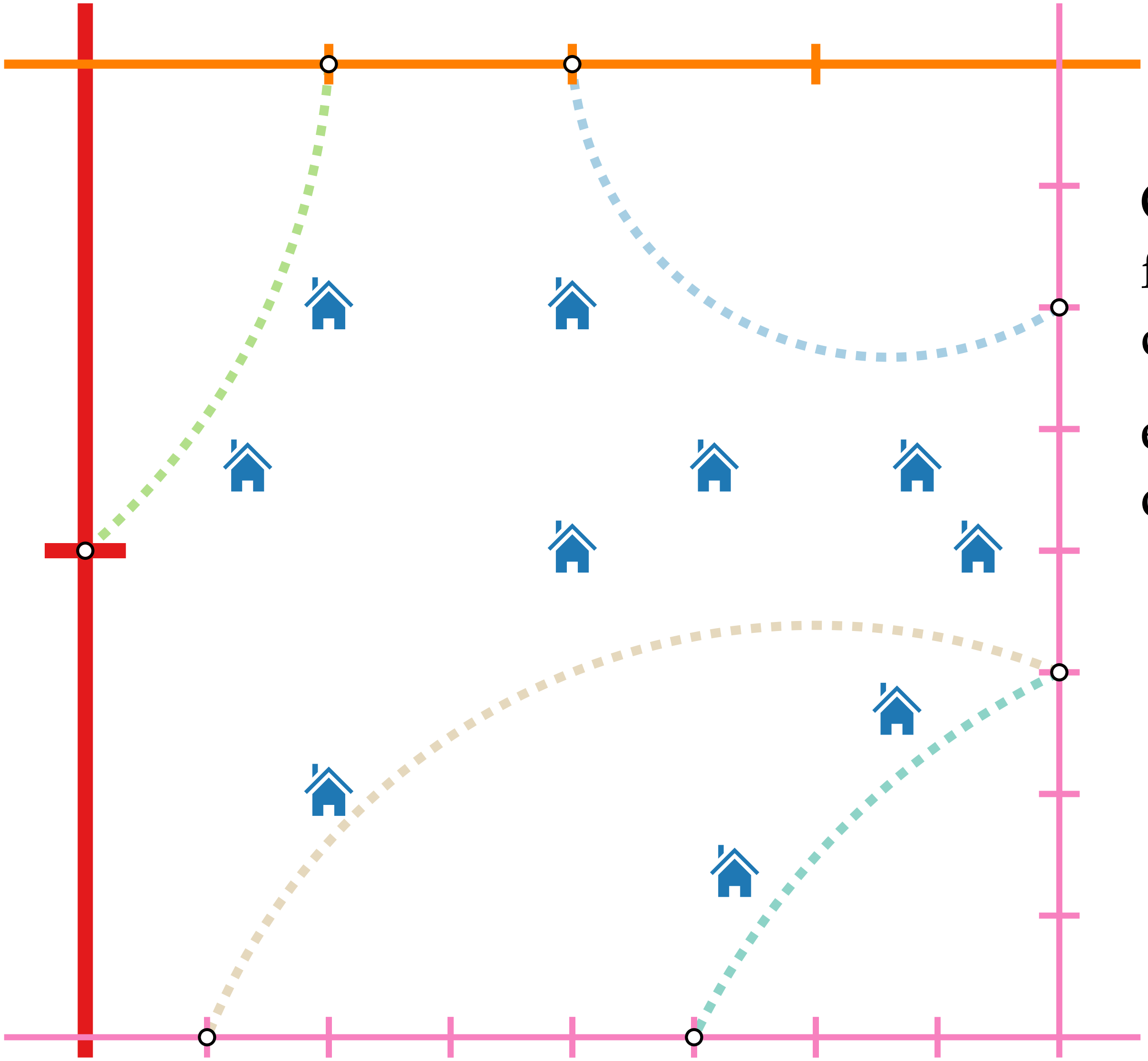
Compute:

Dynamic Program (II)



Compute:
for each square Q in the dissection and

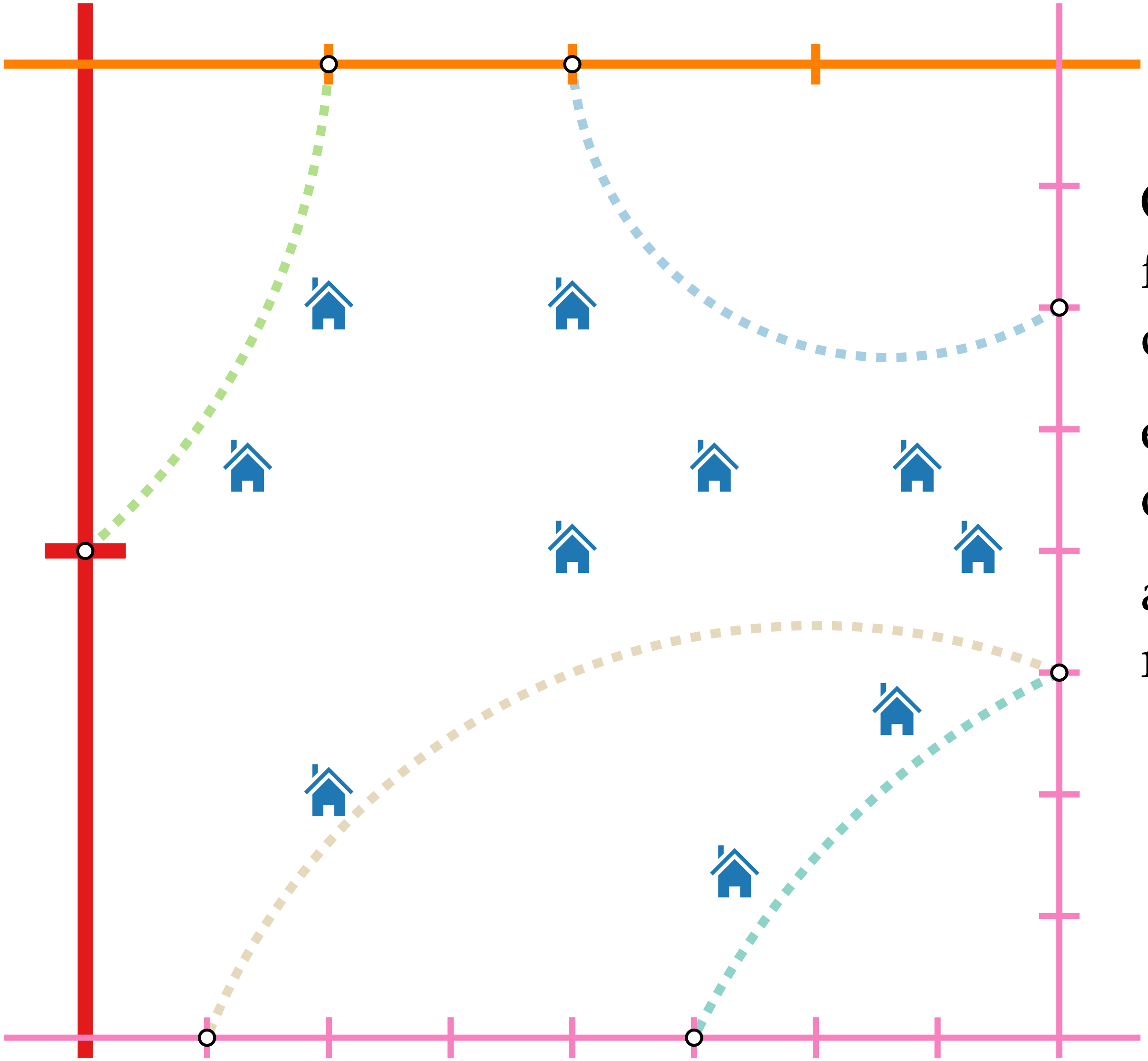
Dynamic Program (II)



Compute:
for each square Q in the dissection and

each crossing-free pairing P of Q

Dynamic Program (II)

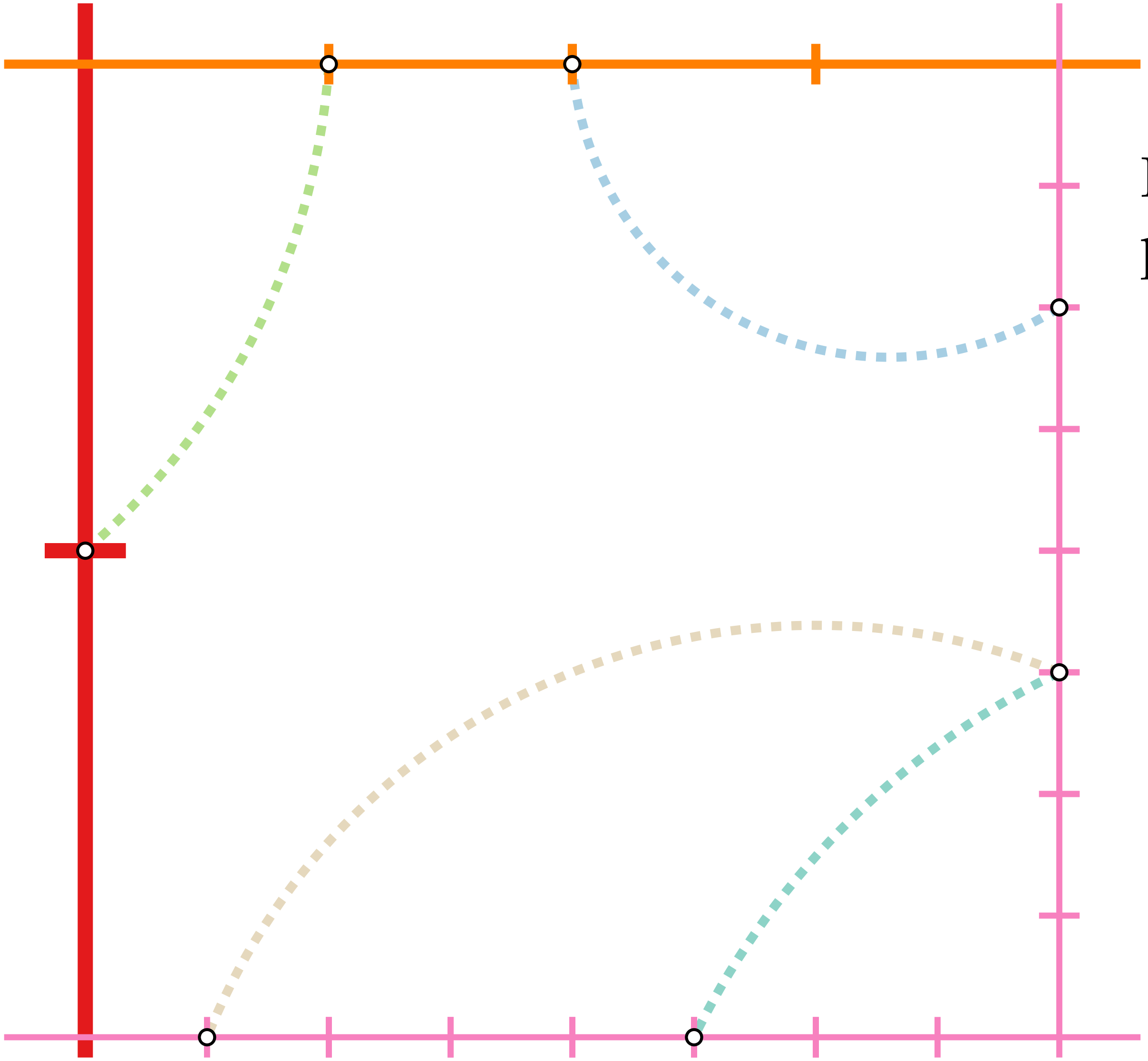


Compute:
for each square Q in the dissection and

each crossing-free pairing P of Q

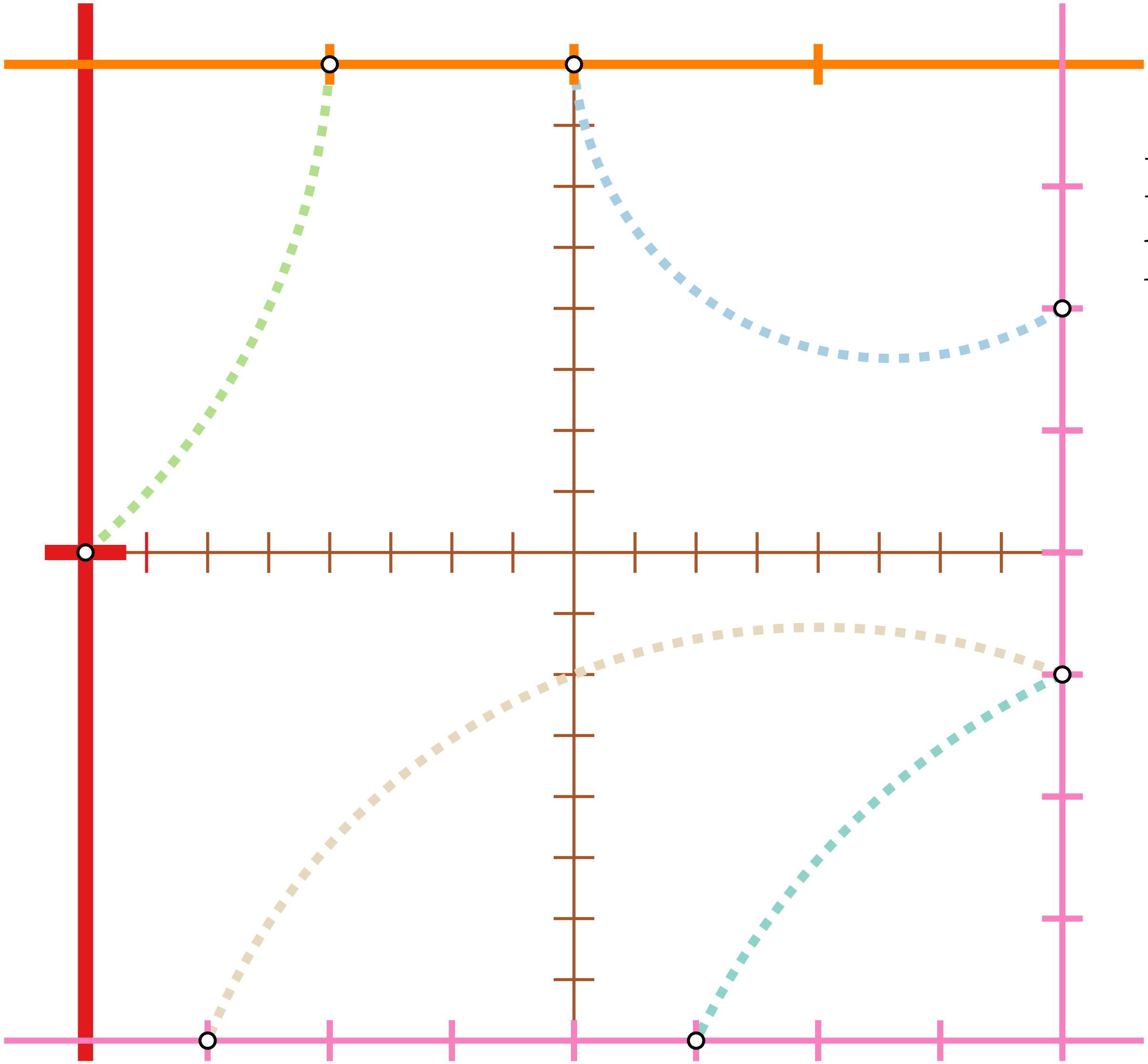
an optimal path cover that respects P.

Dynamic Program (III)



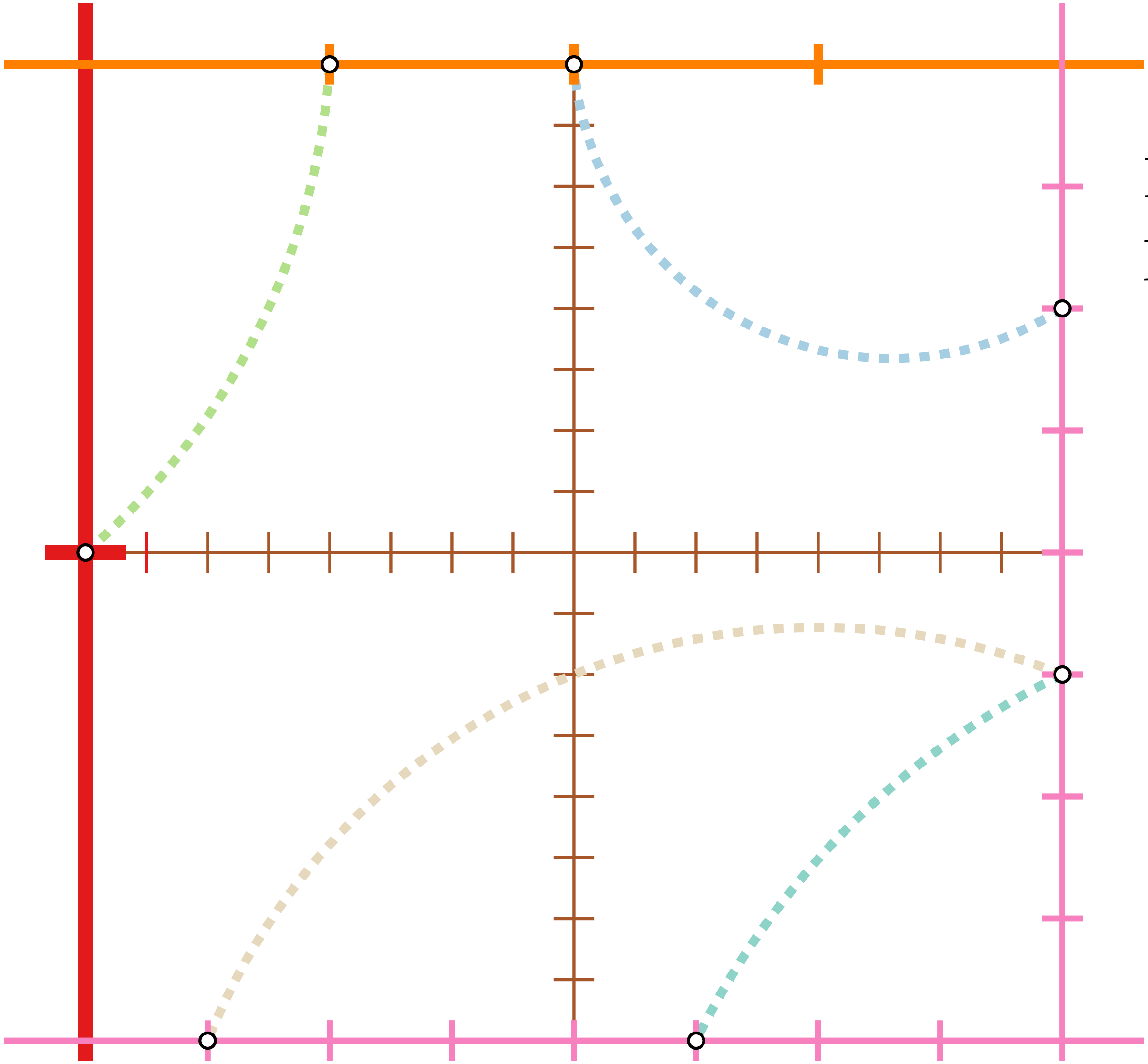
For a given square Q and pairing P:

Dynamic Program (III)



For a given square Q and pairing P:

Dynamic Program (III)

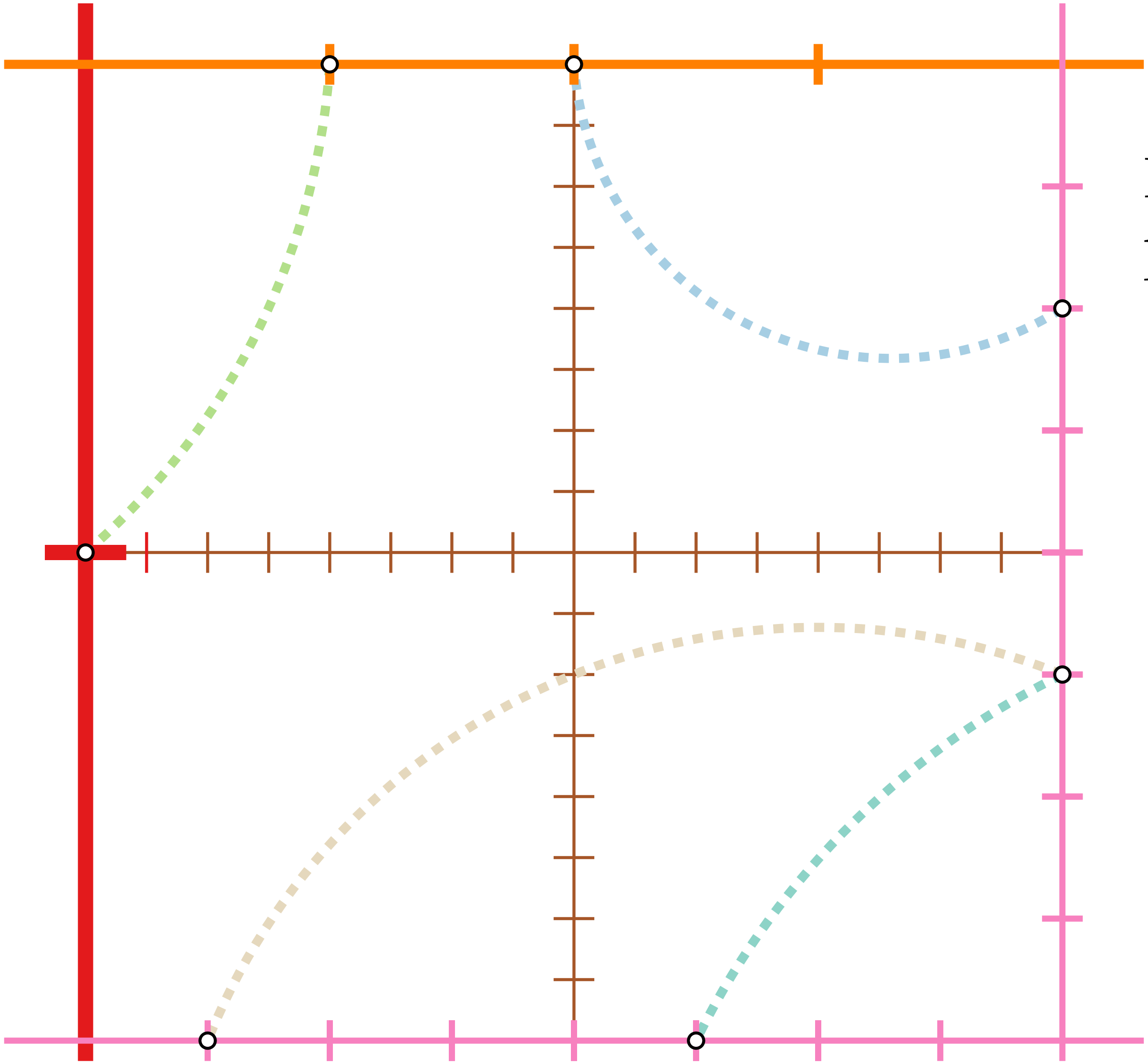


For a given square Q and pairing P:

- Iterate over all

  crossing-free pairings of the child-squares

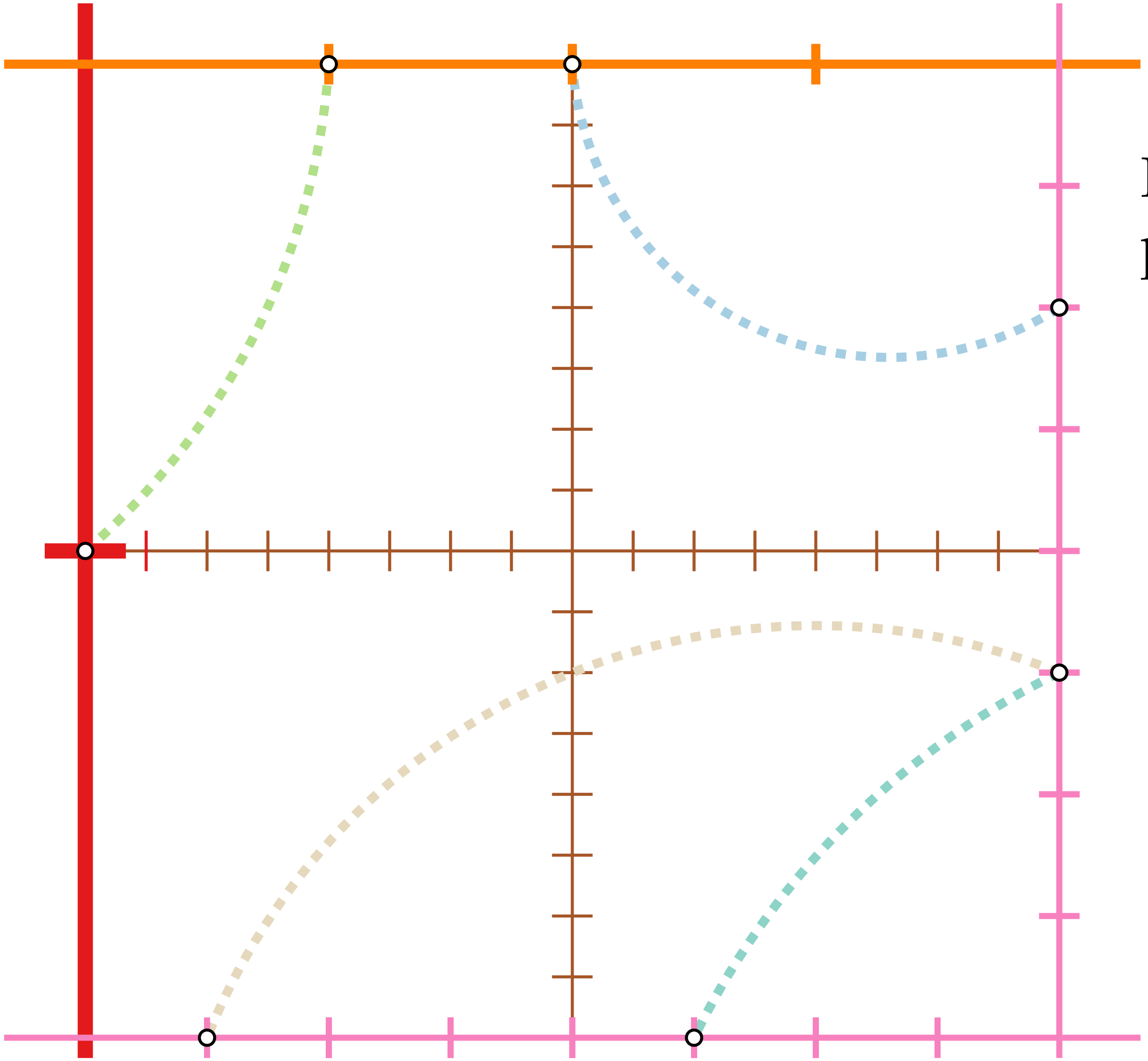
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 =$ crossing-free pairings of the child-squares

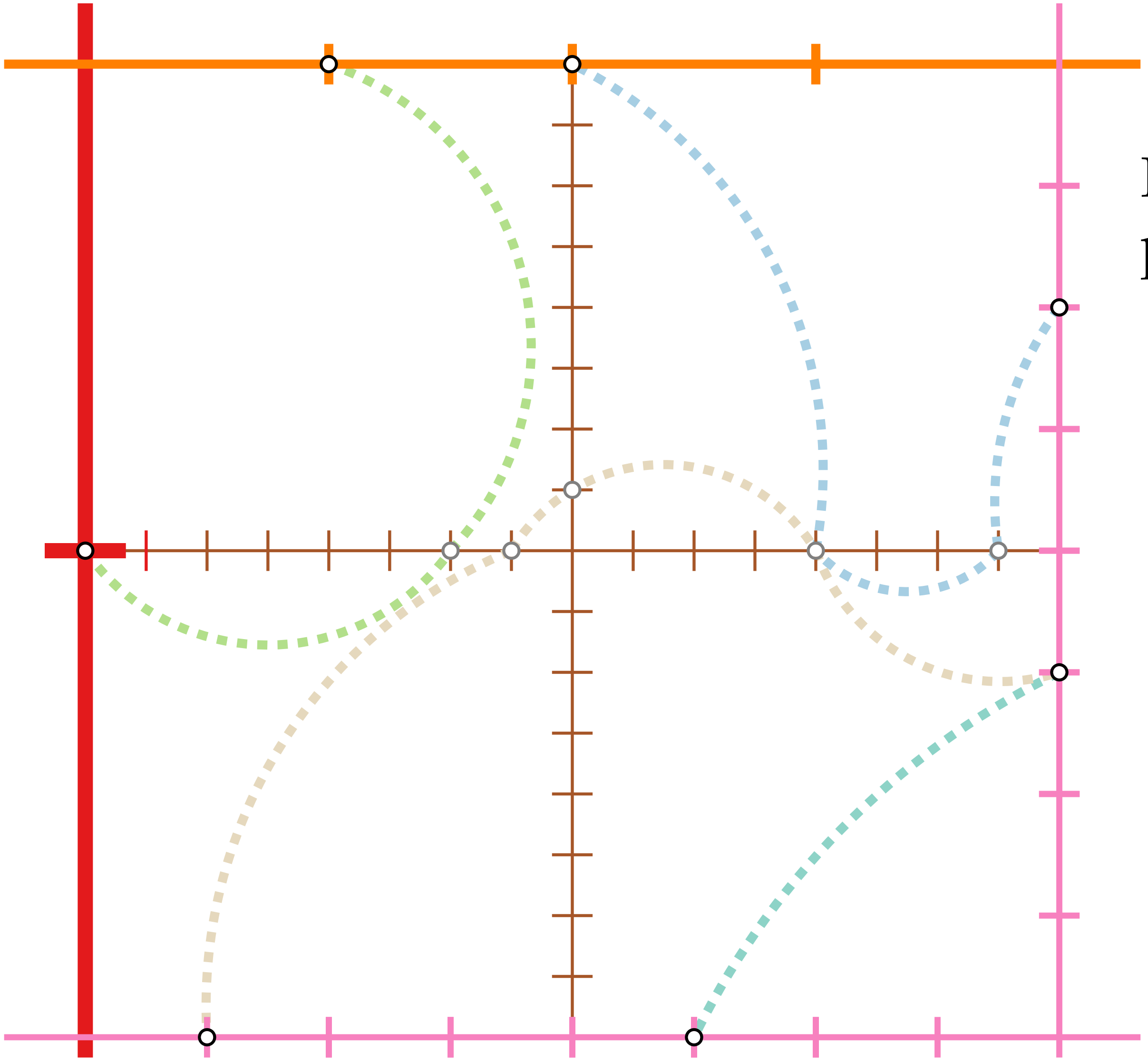
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $\left(n^{O(1/\varepsilon)}\right)^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares

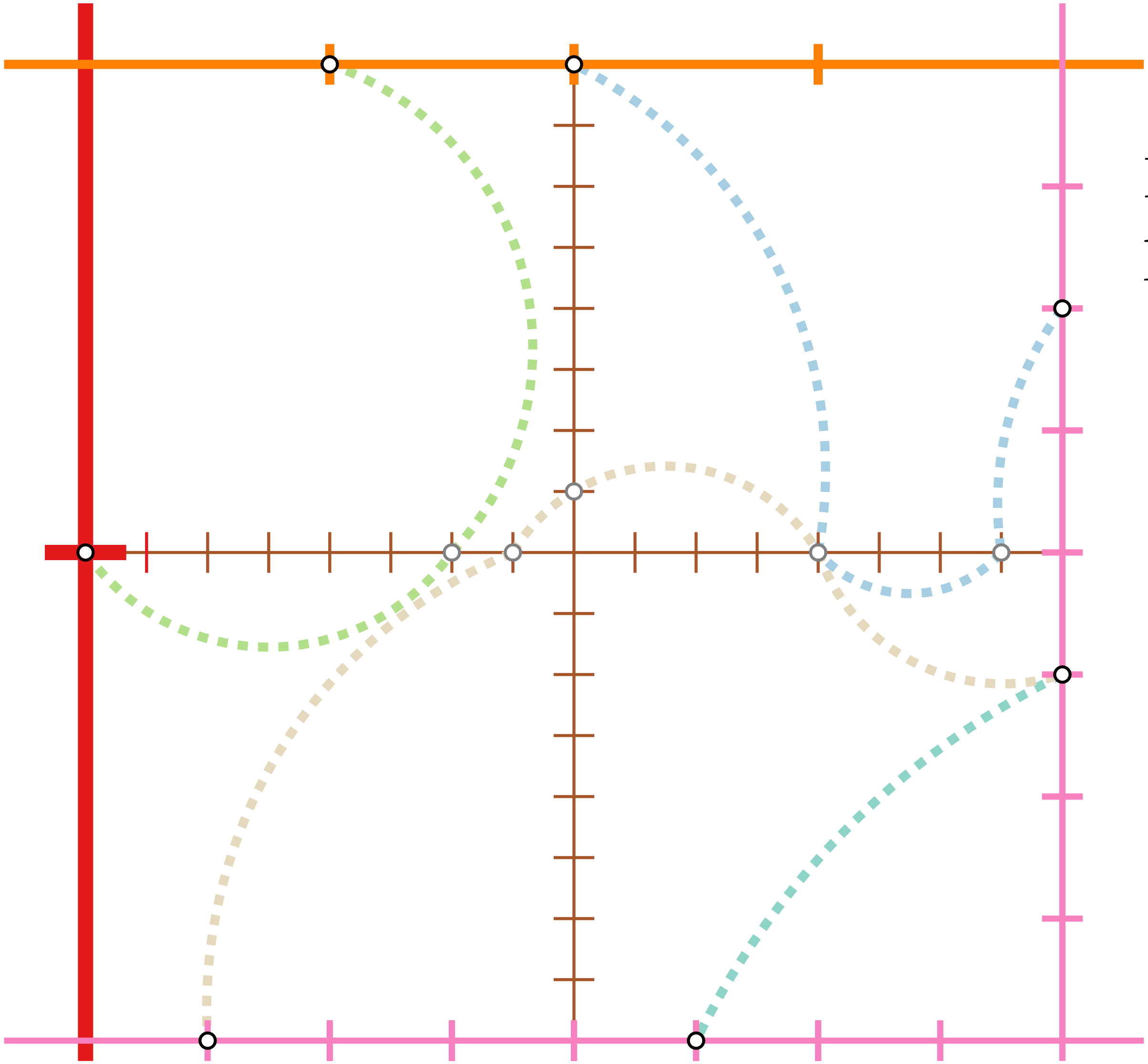
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares

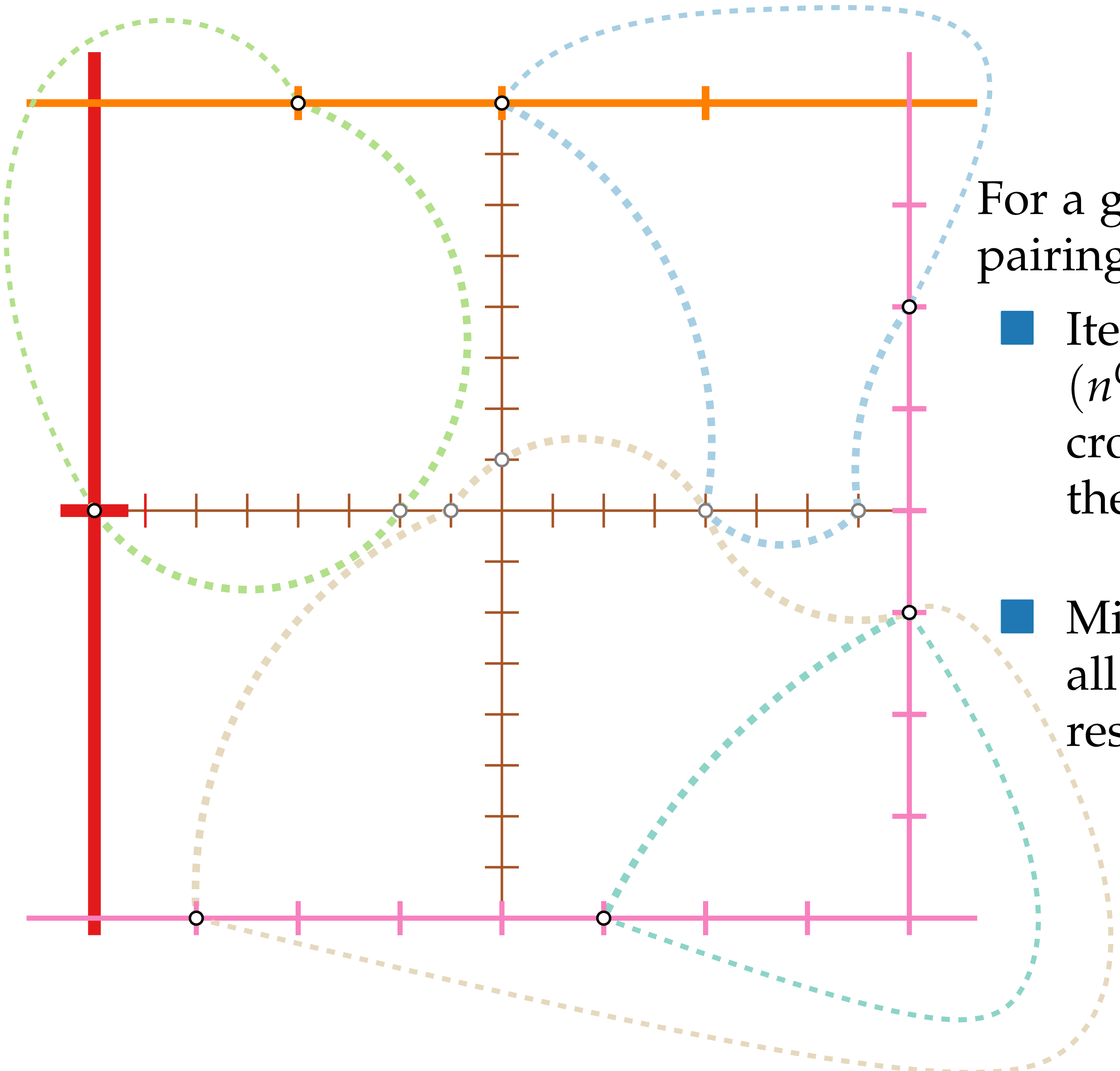
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares
- Minimize the cost over all such pairings that respect P

Dynamic Program (III)

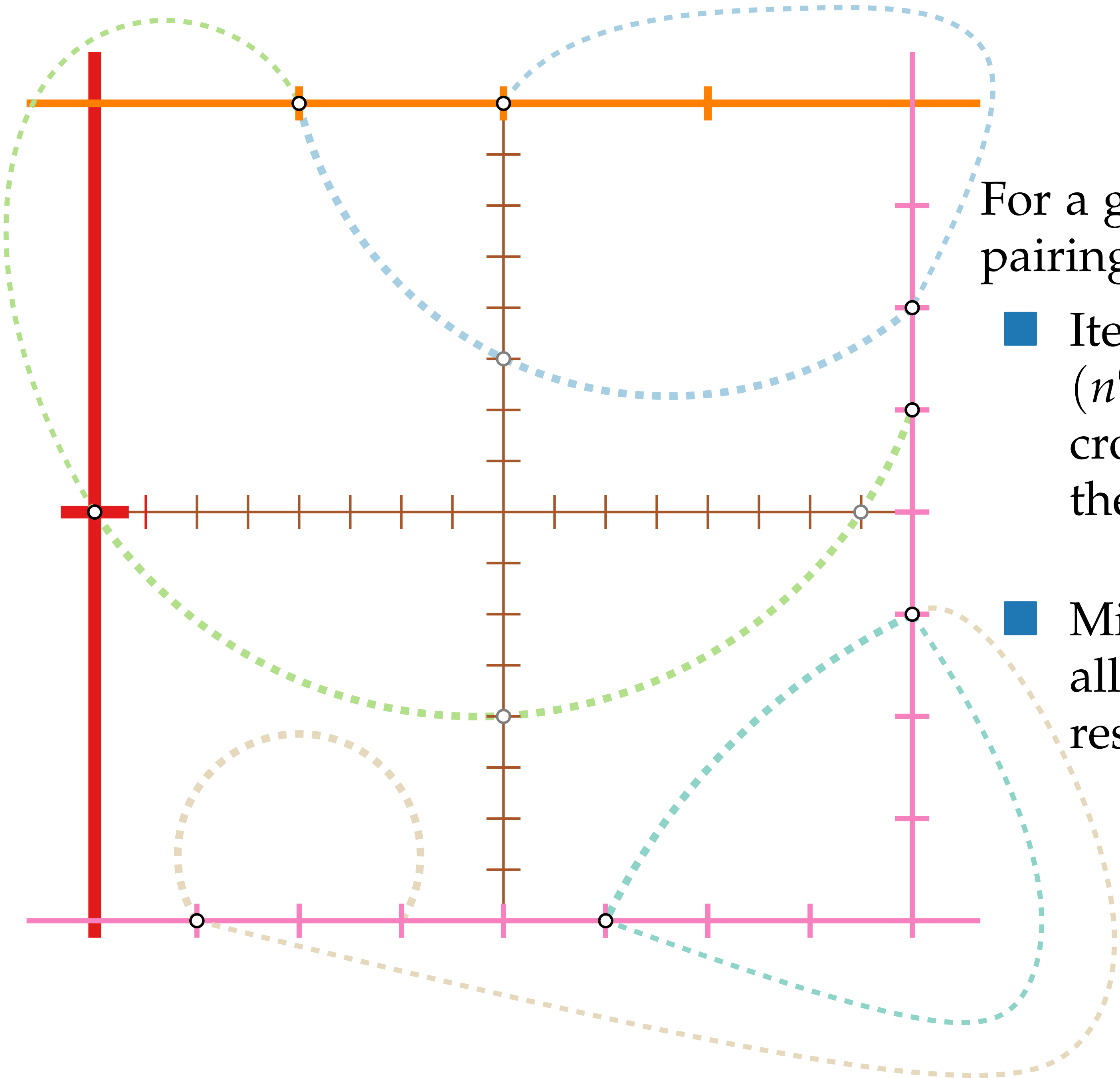


For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares

- Minimize the cost over all such pairings that respect P

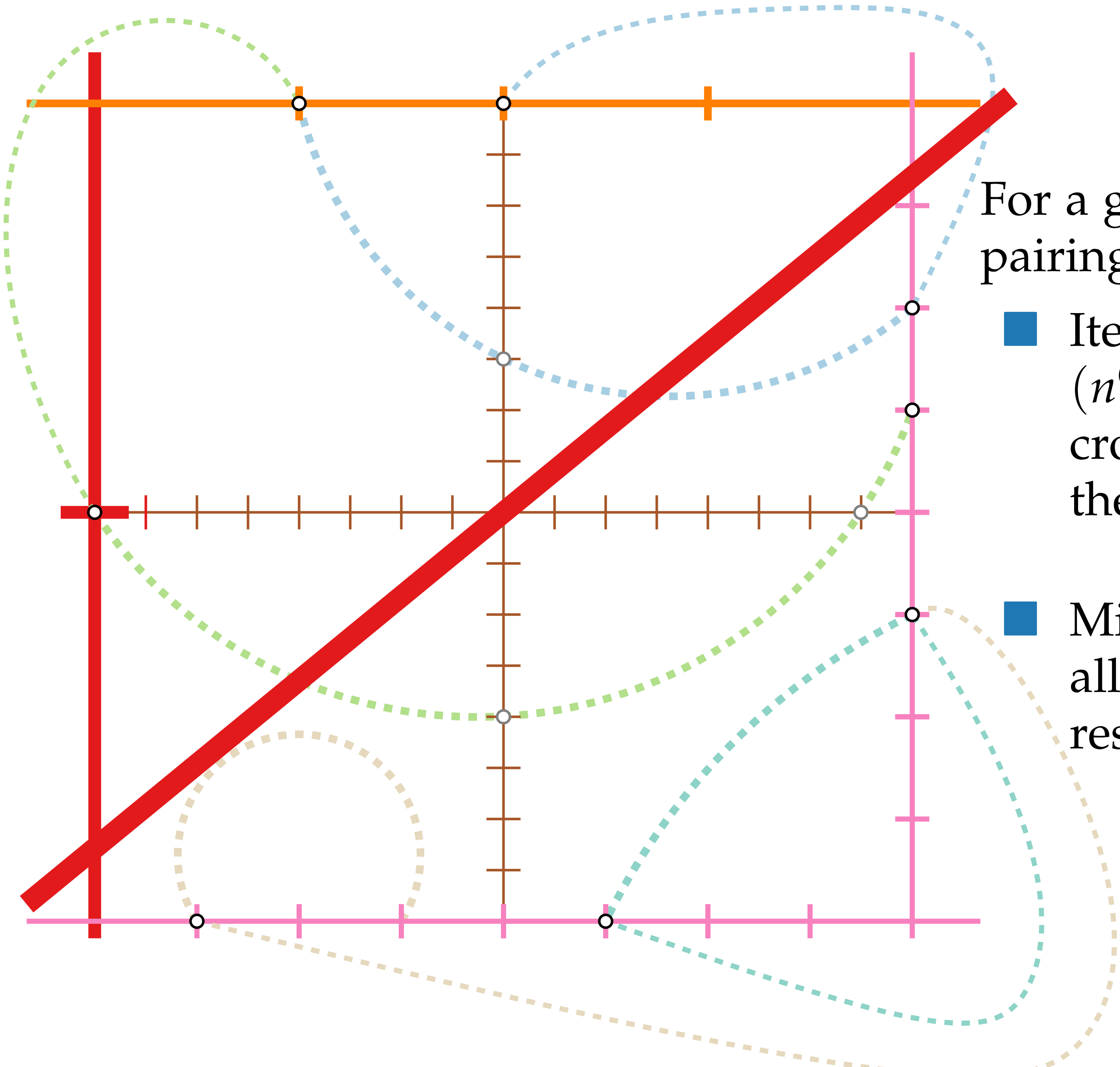
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares
- Minimize the cost over all such pairings that respect P

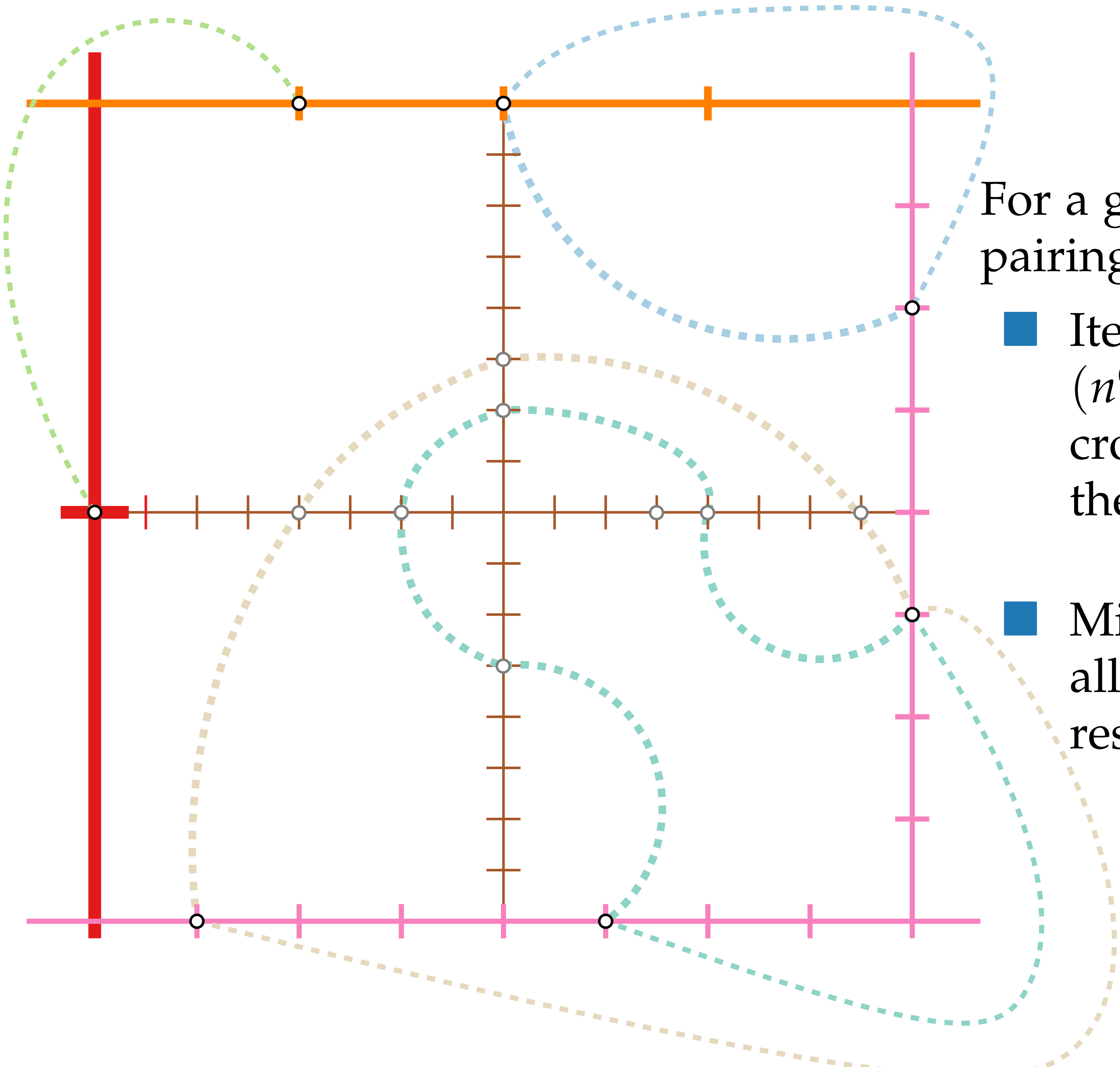
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares
- Minimize the cost over all such pairings that respect P

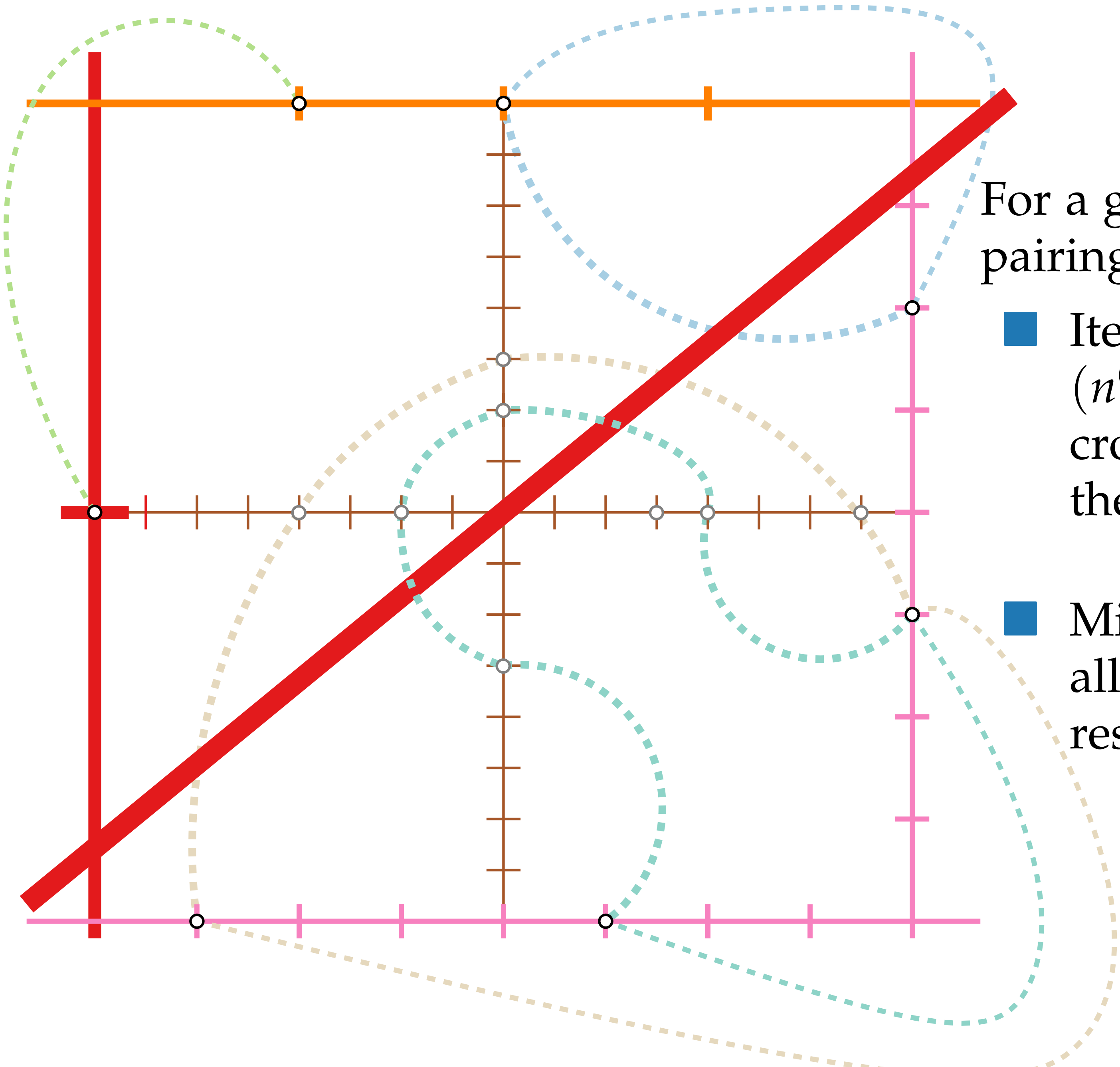
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares
- Minimize the cost over all such pairings that respect P

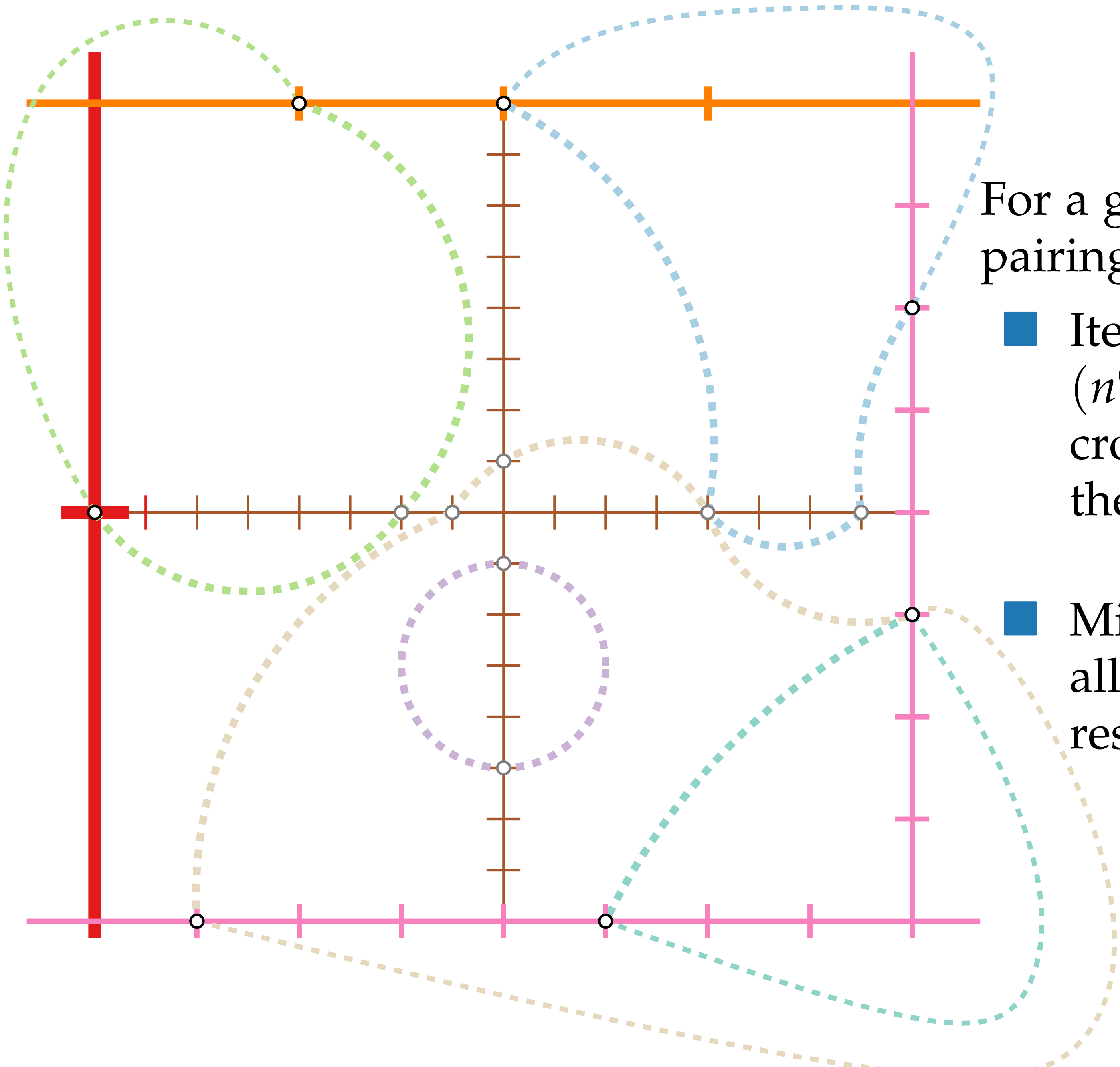
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares
- Minimize the cost over all such pairings that respect P

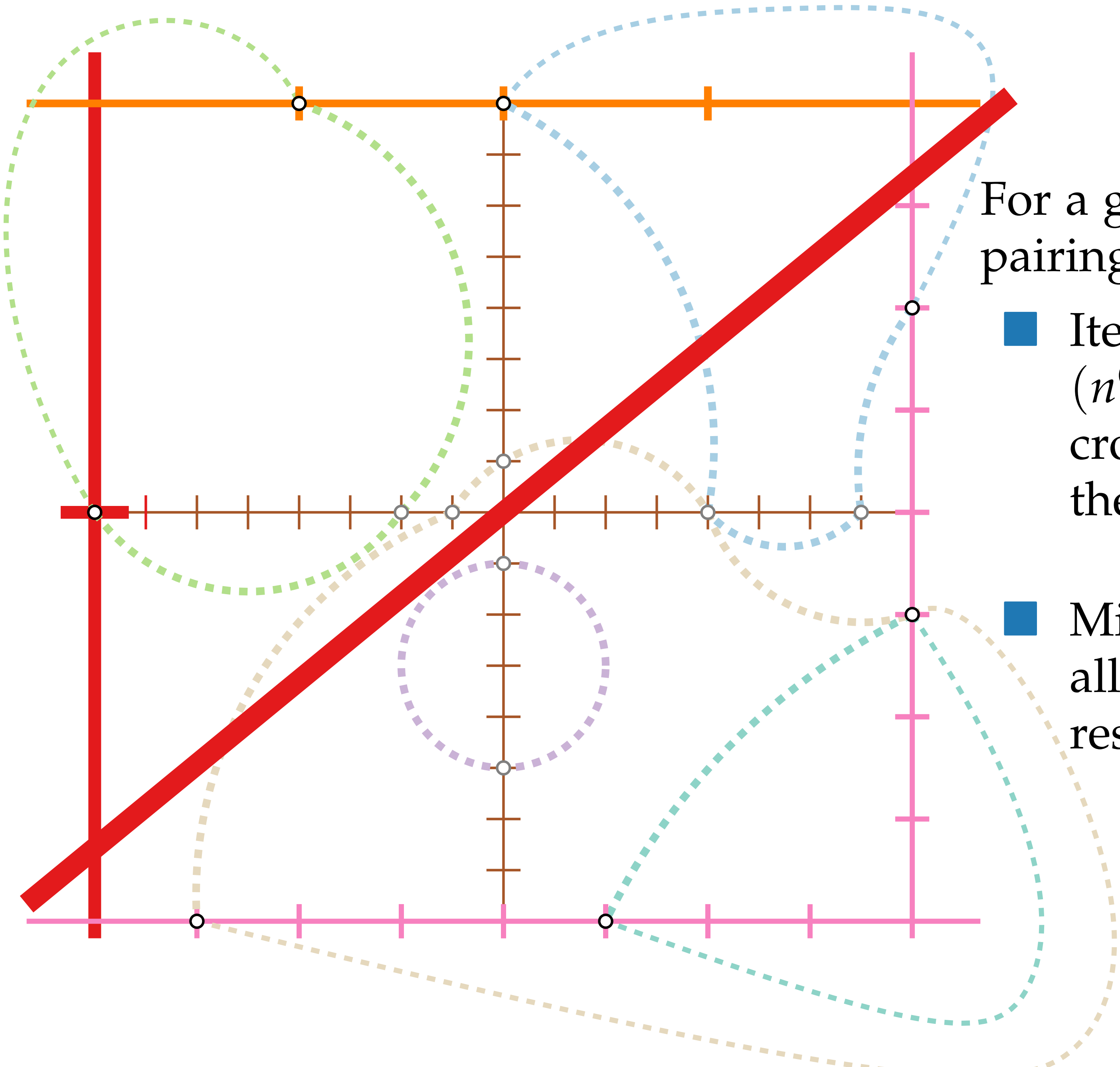
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares
- Minimize the cost over all such pairings that respect P

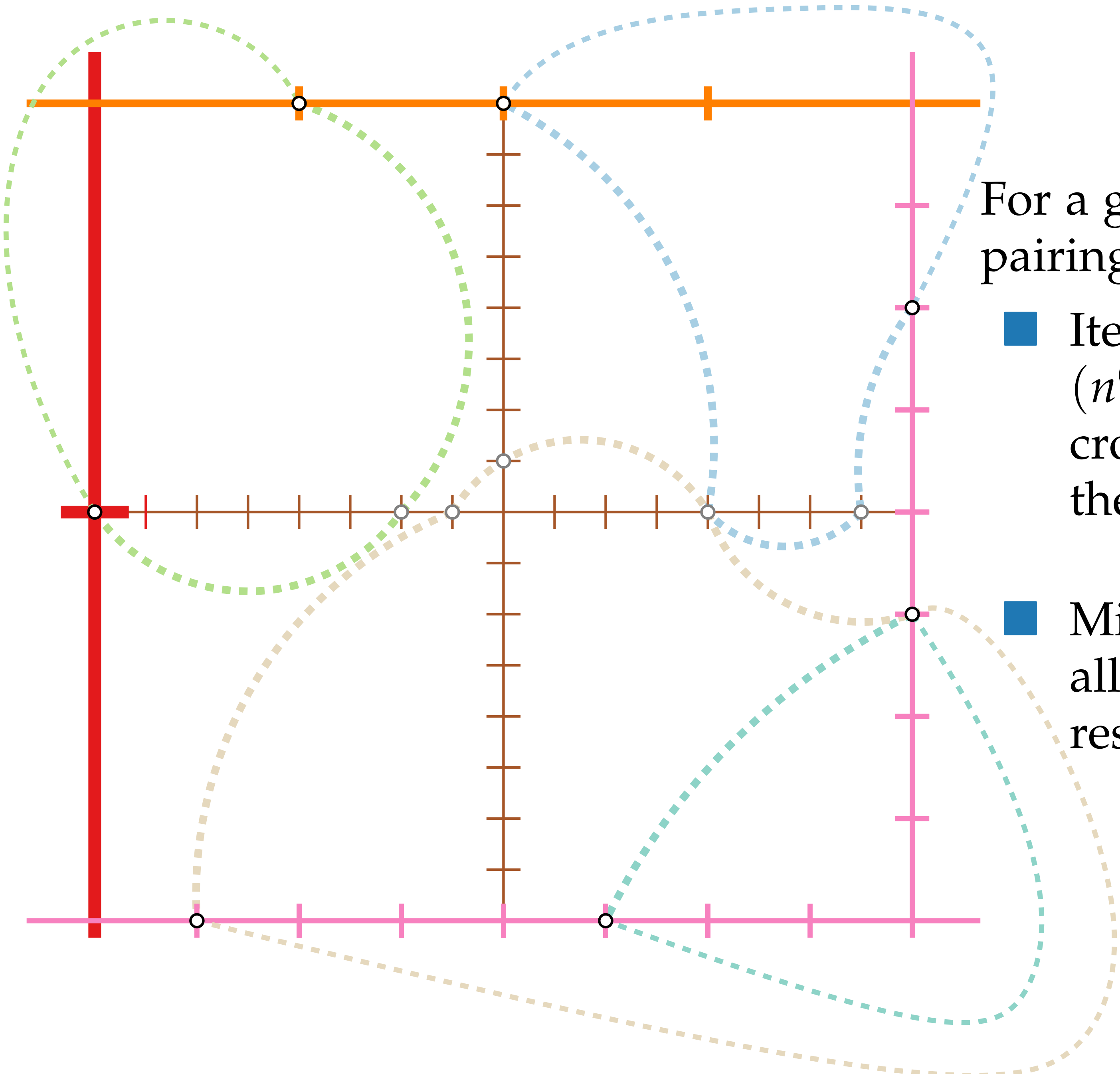
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares
- Minimize the cost over all such pairings that respect P

Dynamic Program (III)

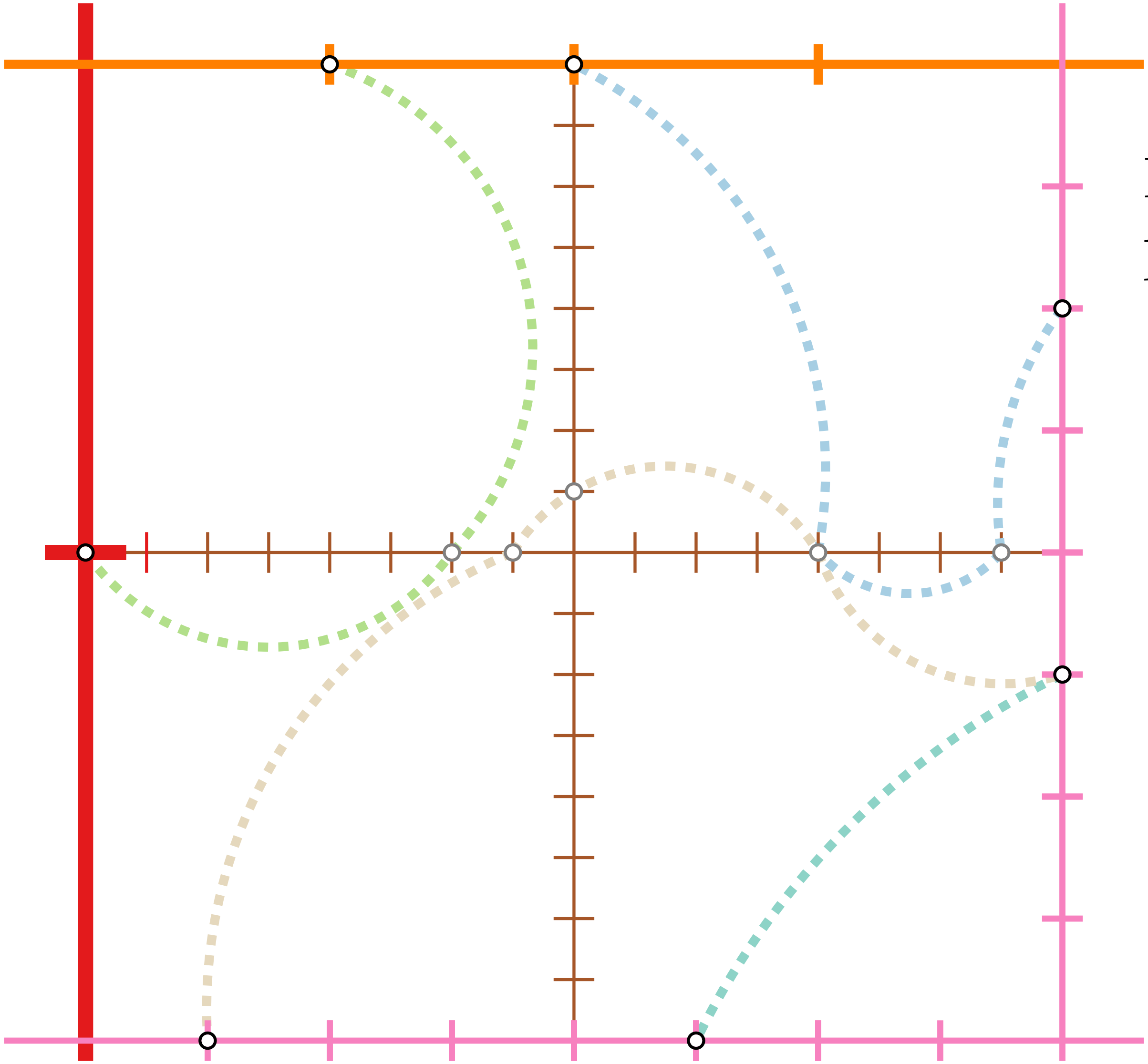


For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares

- Minimize the cost over all such pairings that respect P

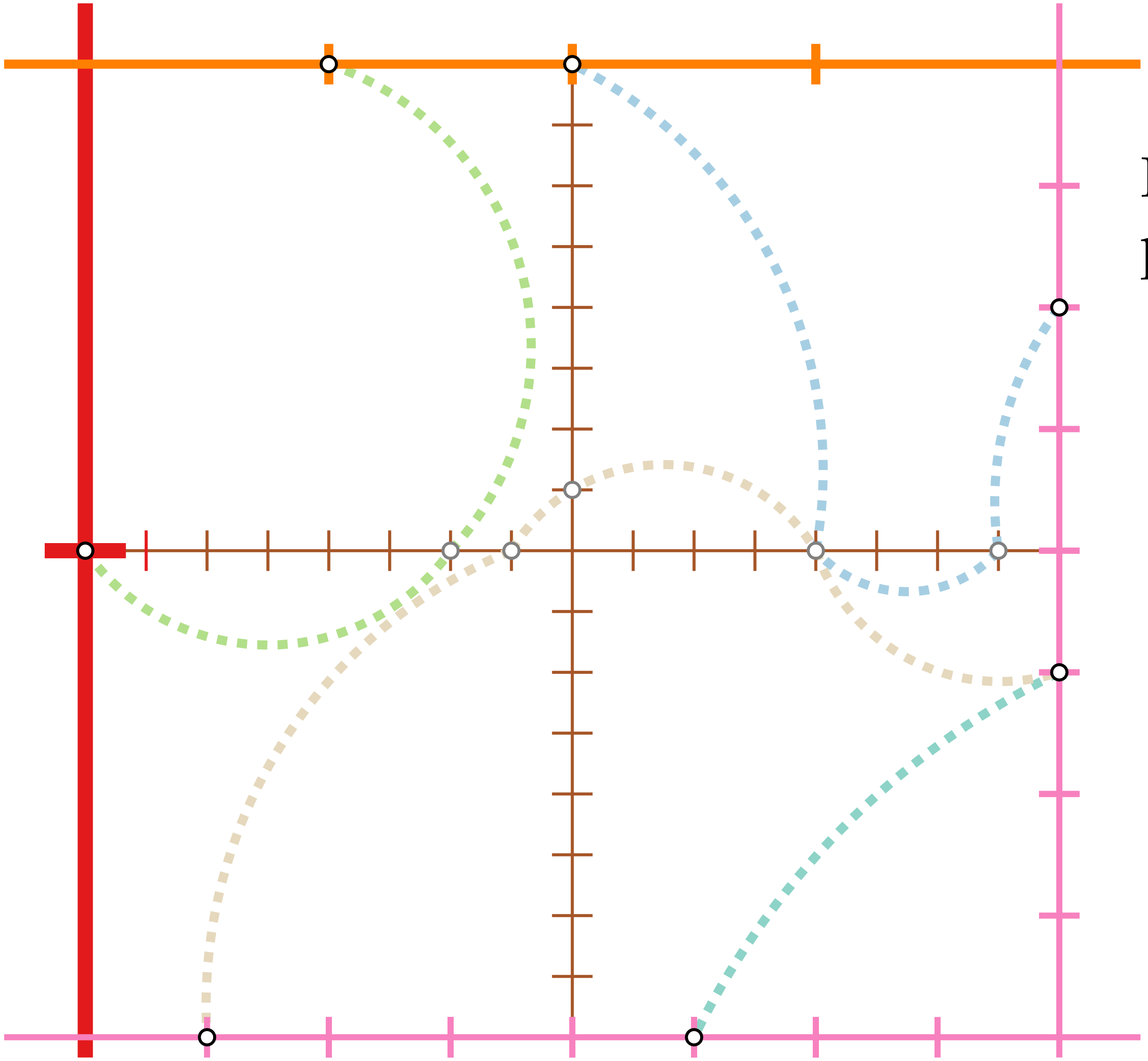
Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares
- Minimize the cost over all such pairings that respect P
- Correctness by induction

Dynamic Program (III)



For a given square Q and pairing P:

- Iterate over all $(n^{O(1/\varepsilon)})^4 = n^{O(1/\varepsilon)}$ crossing-free pairings of the child-squares
- Minimize the cost over all such pairings that respect P
- Correctness by induction

Lemma. An optimal well behaved tour can be computed in $2^{O(m)} = n^{O(1/\varepsilon)}$ time.

Approximation Algorithms

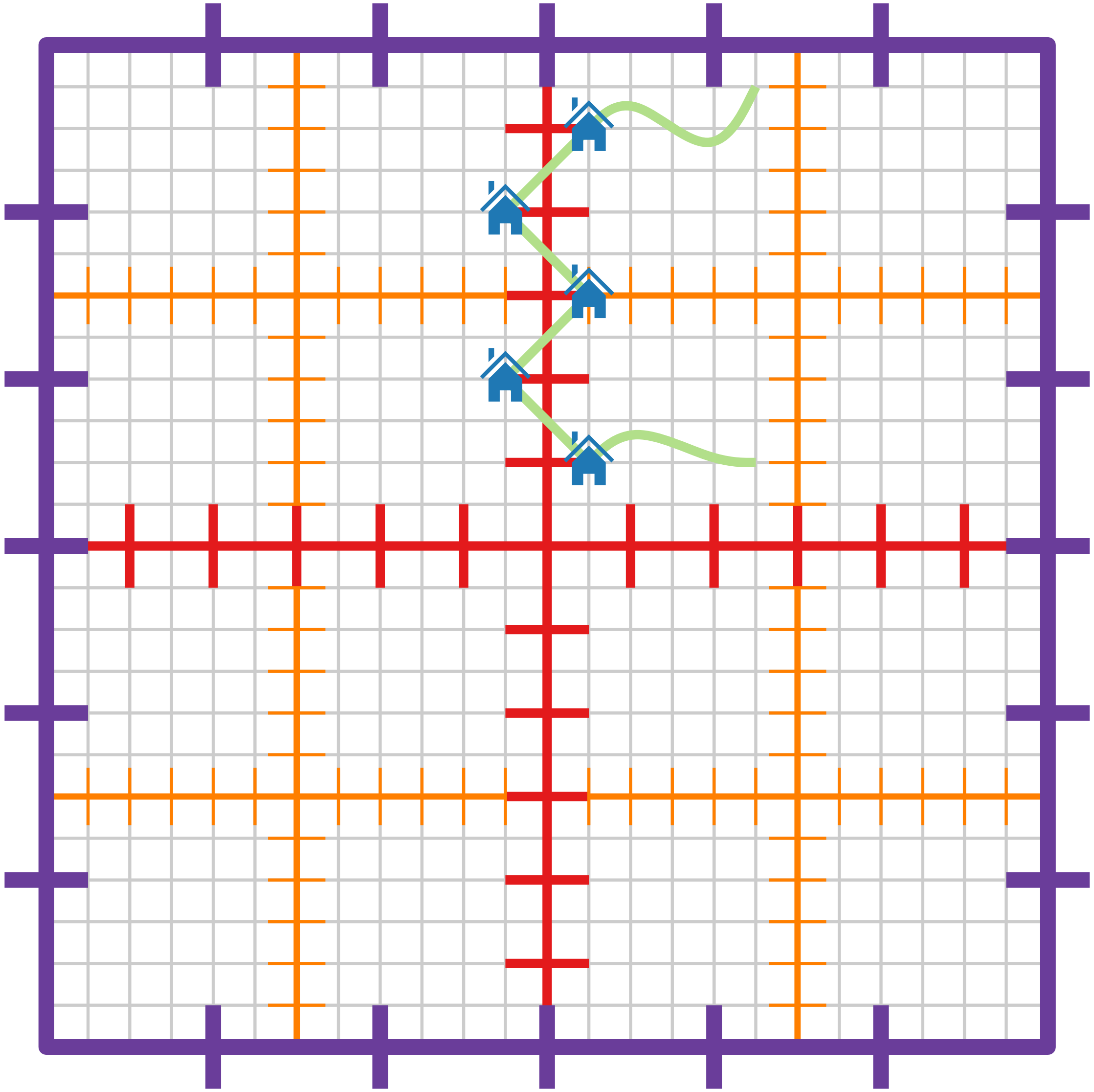
Lecture 9:
PTAS for EuclideanTSP

Part V:
Shifted Dissections

Joachim Spoerhase

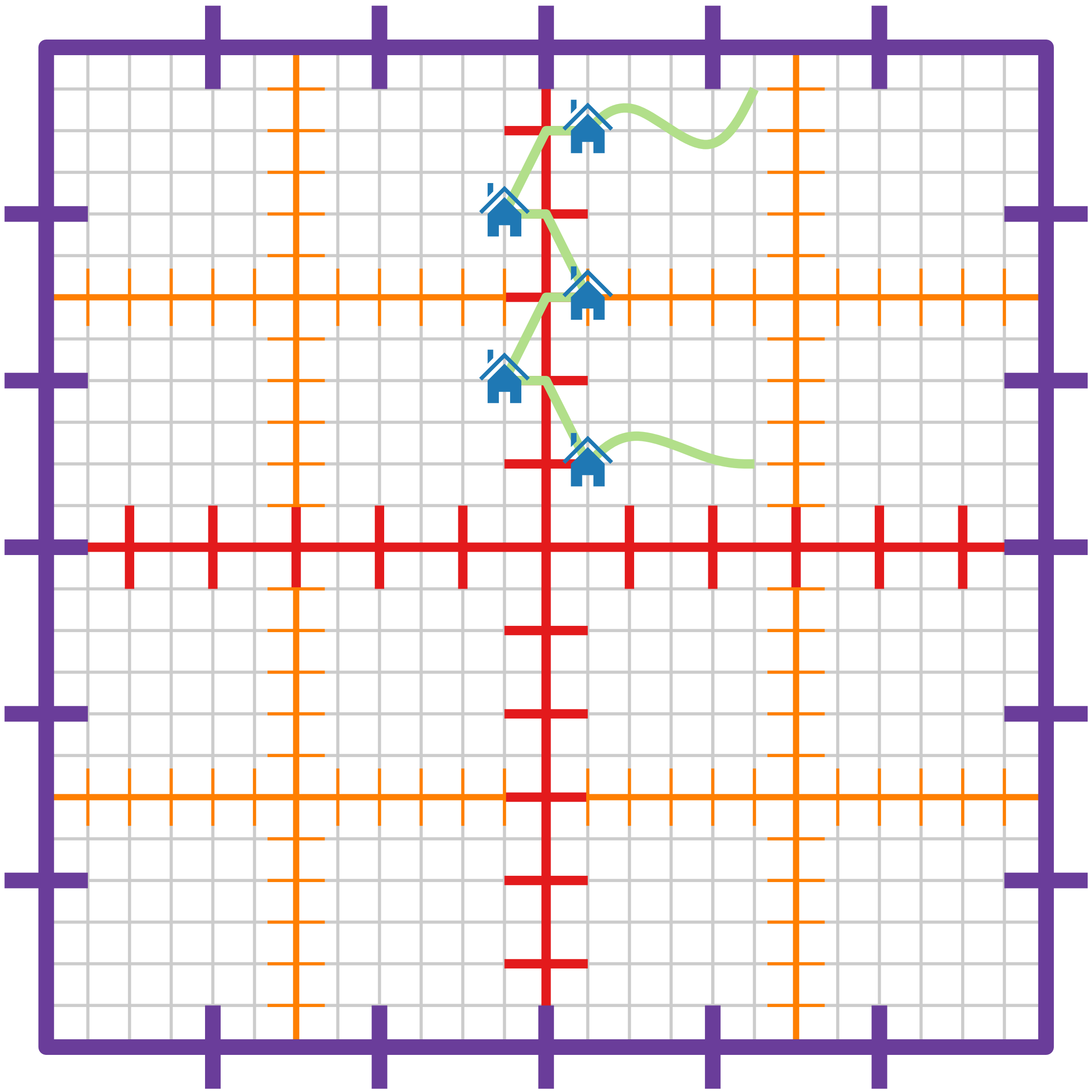
Winter 2021/22

Shifted Dissections



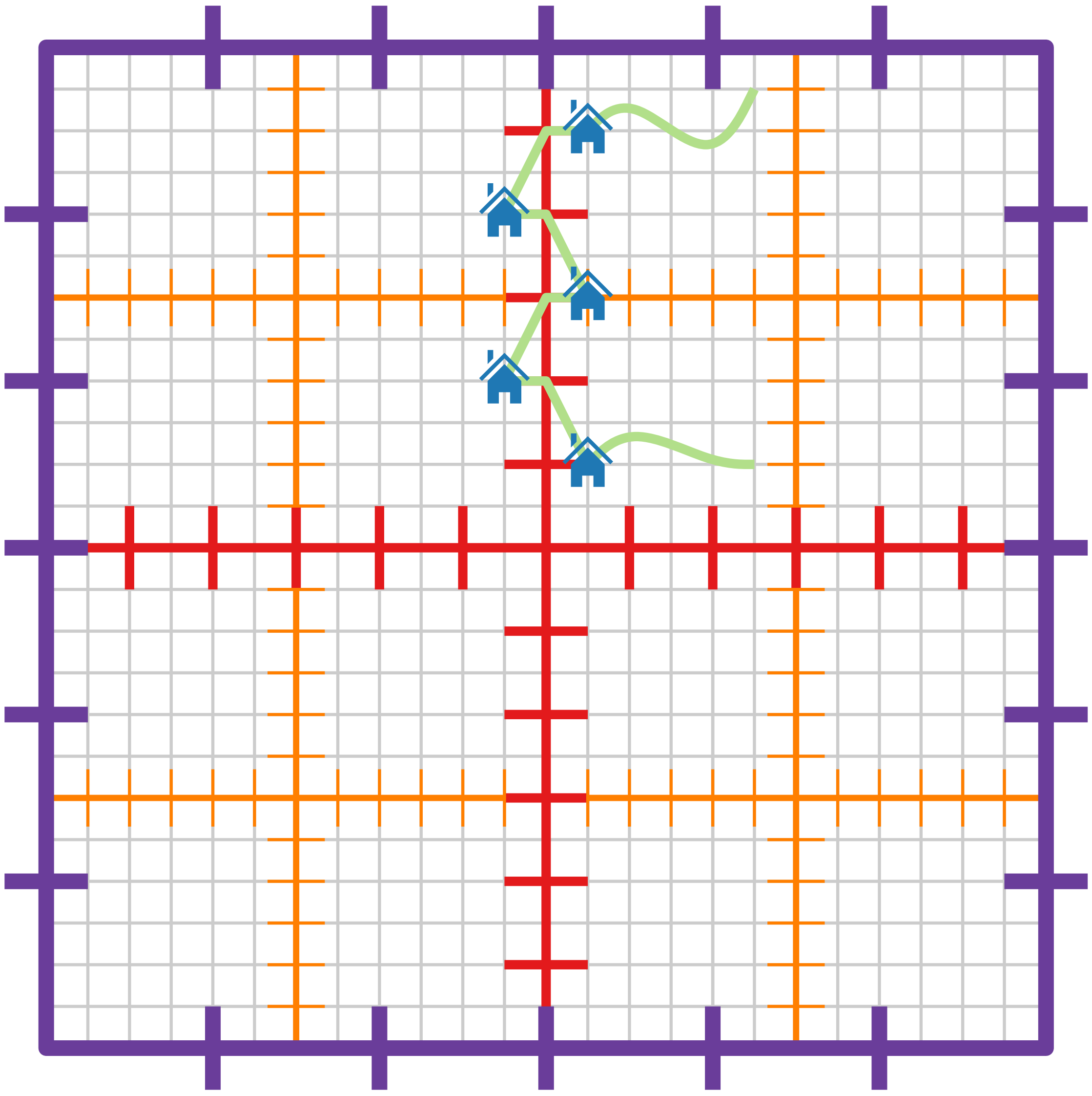
- The best well behaved tour can be a bad approximation.

Shifted Dissections



- The best well behaved tour can be a bad approximation.

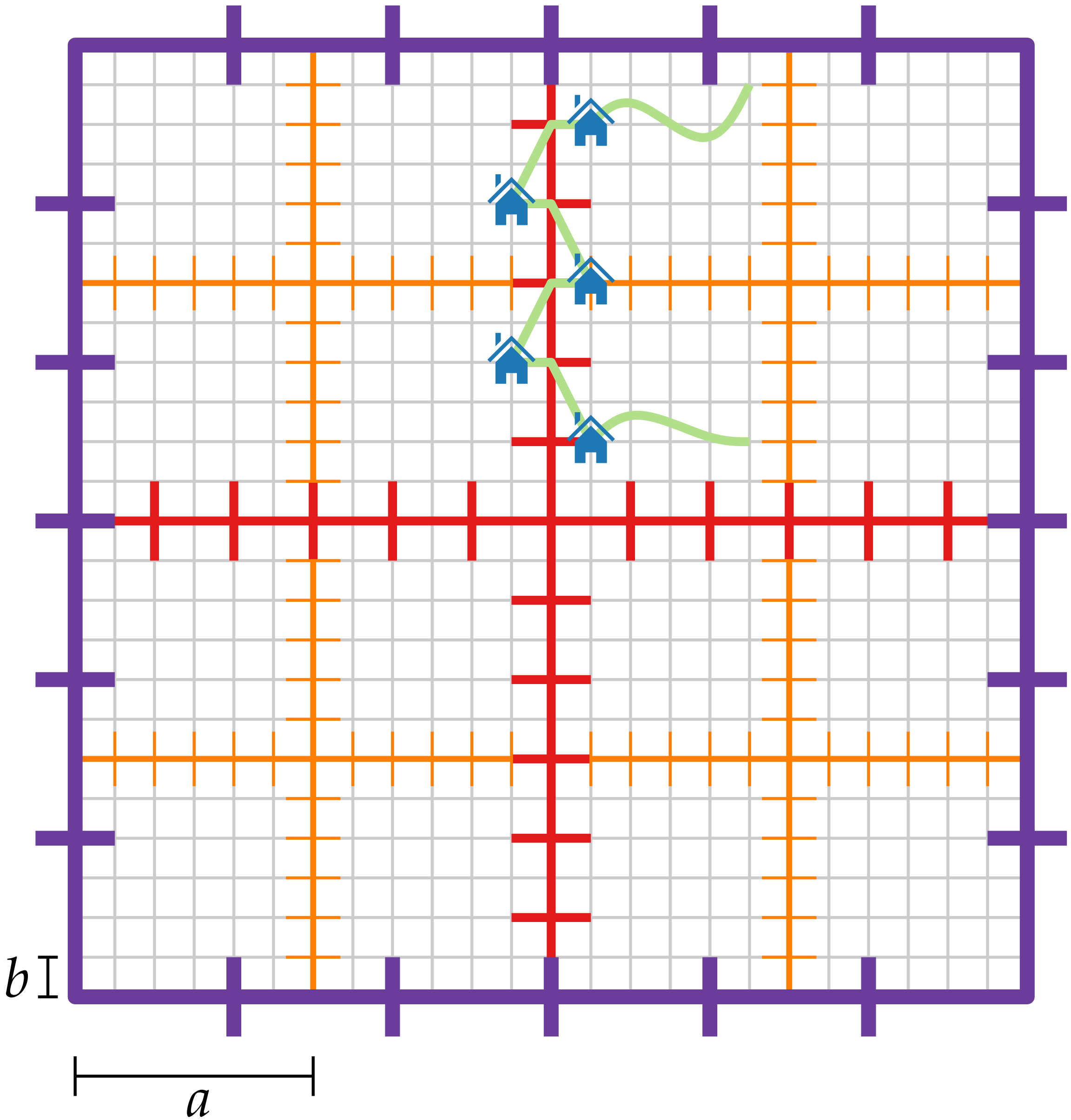
Shifted Dissections



- The best well behaved tour can be a bad approximation.
- Consider an (a, b)-shifted dissection:

$$x \mapsto (x + a) \bmod L$$
$$y \mapsto (y + b) \bmod L$$

Shifted Dissections

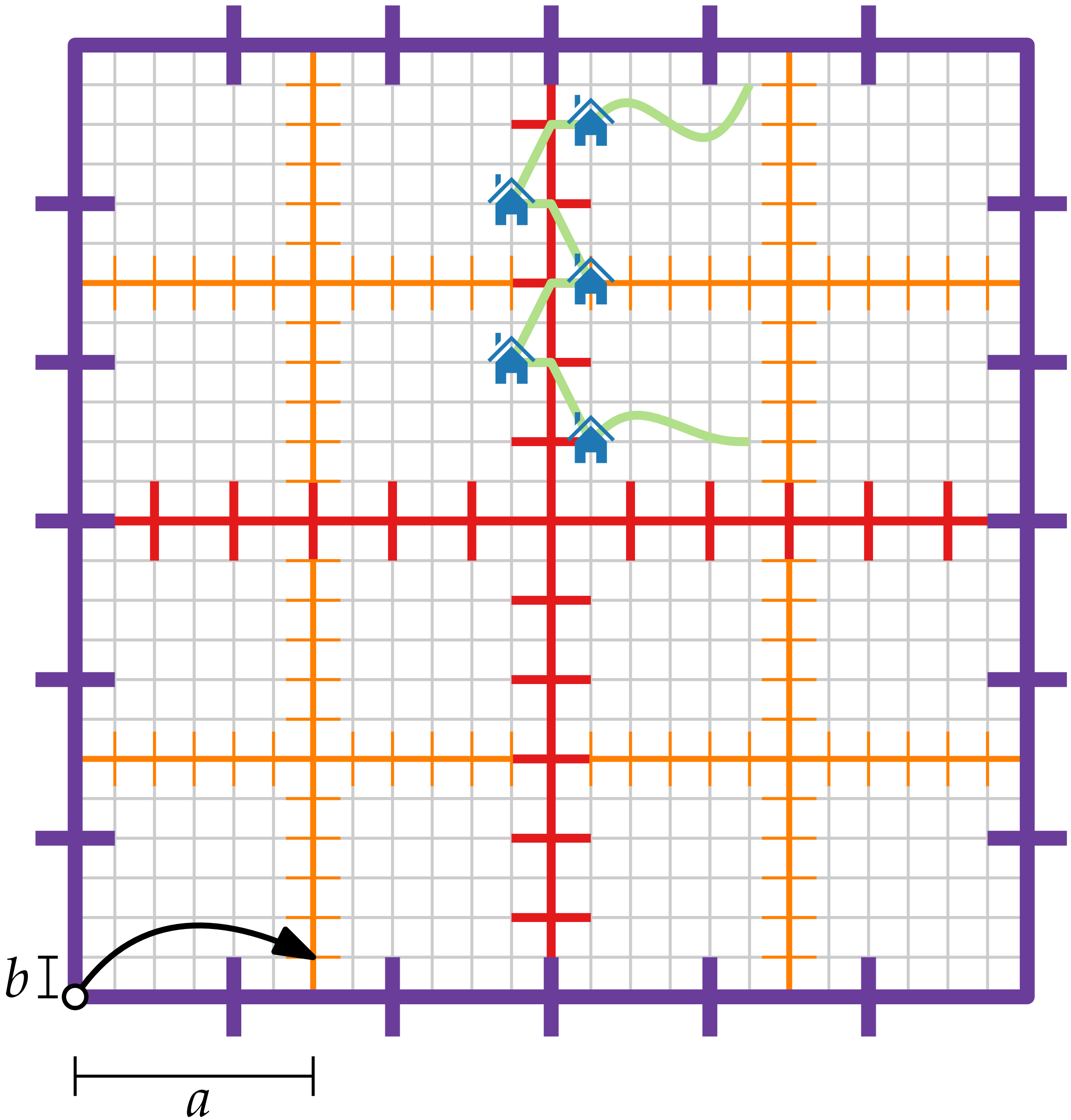


- The best well behaved tour can be a bad approximation.

- Consider an (a, b)-shifted dissection:

$$x \mapsto (x + a) \bmod L$$
$$y \mapsto (y + b) \bmod L$$

Shifted Dissections

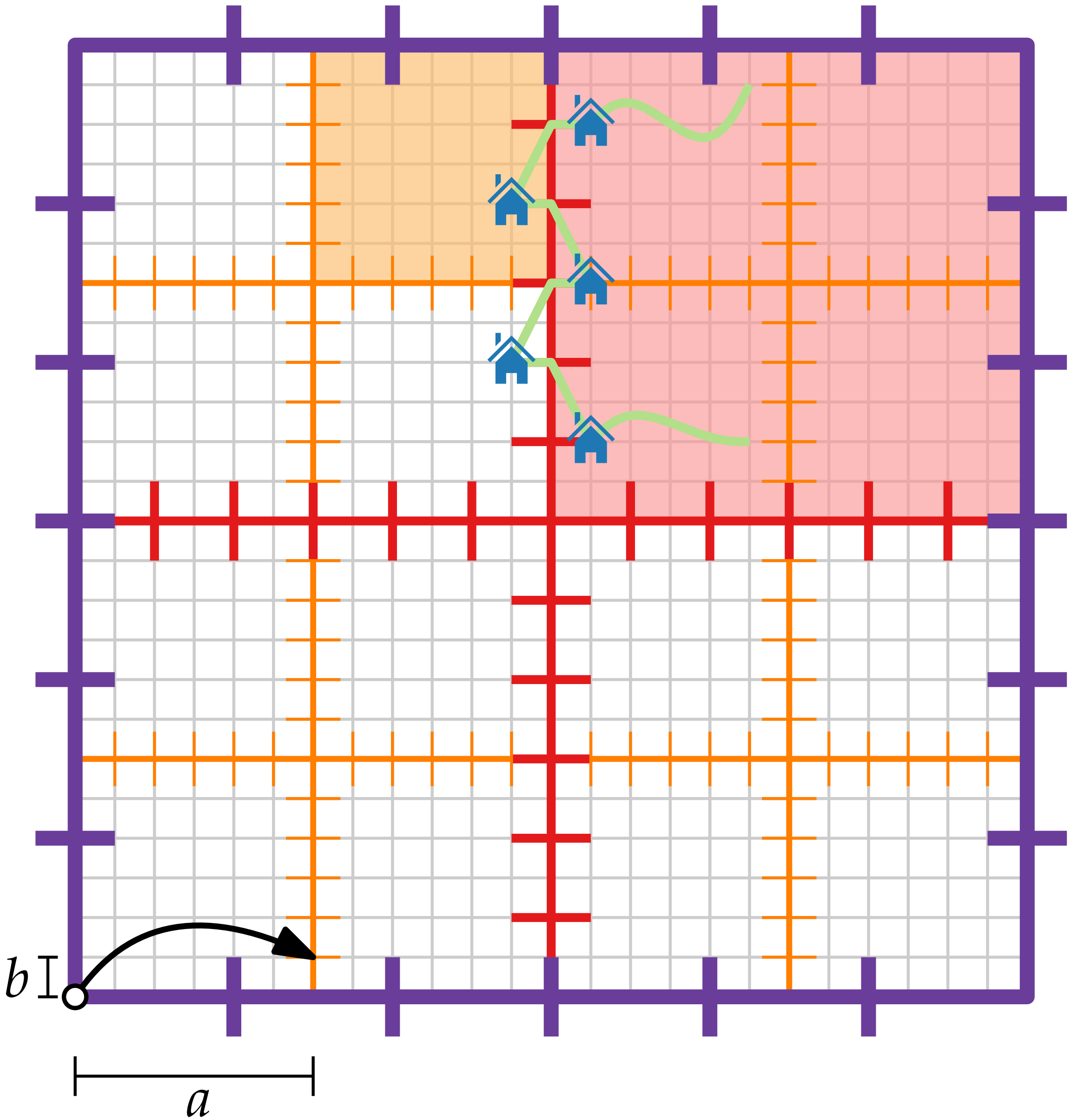


- The best well behaved tour can be a bad approximation.

- Consider an (a, b)-shifted dissection:

$$x \mapsto (x + a) \bmod L$$
$$y \mapsto (y + b) \bmod L$$

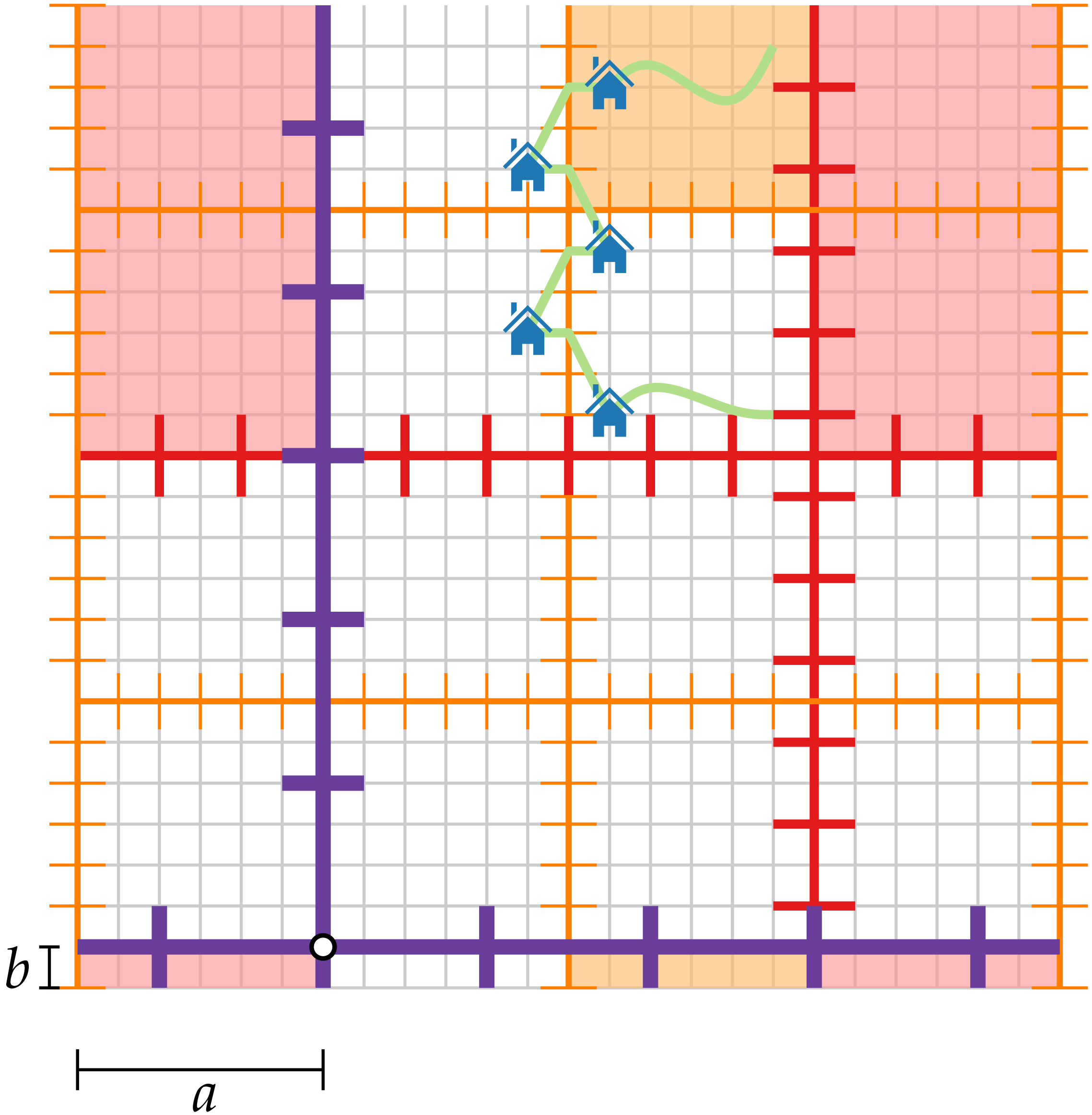
Shifted Dissections



- The best well behaved tour can be a bad approximation.
- Consider an (a, b)-shifted dissection:

$$x \mapsto (x + a) \bmod L$$
$$y \mapsto (y + b) \bmod L$$

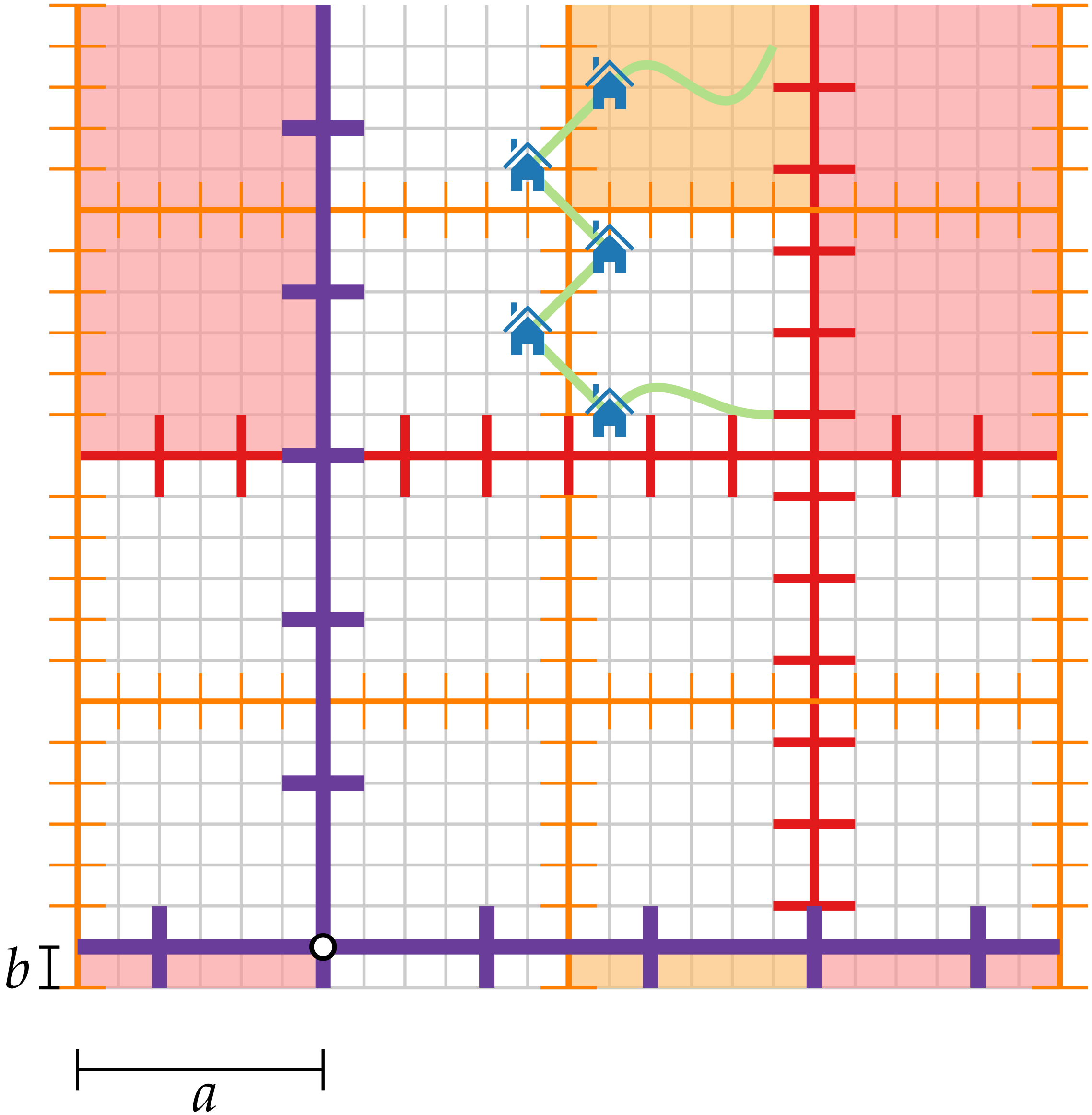
Shifted Dissections



- The best well behaved tour can be a bad approximation.
- Consider an (a, b)-shifted dissection:

$$x \mapsto (x + a) \bmod L$$
$$y \mapsto (y + b) \bmod L$$

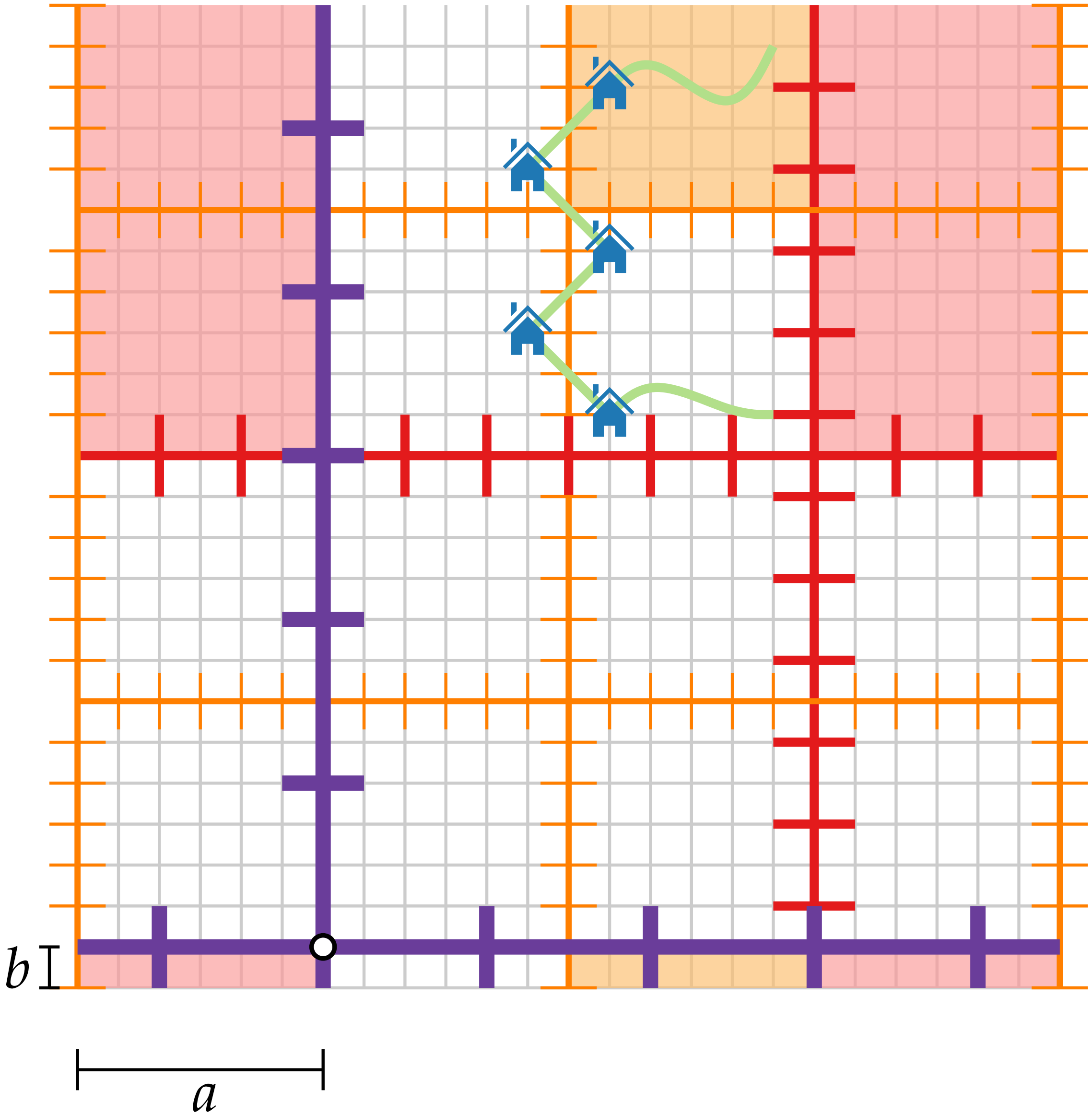
Shifted Dissections



- The best well behaved tour can be a bad approximation.
- Consider an (a, b)-shifted dissection:

$$x \mapsto (x + a) \bmod L$$
$$y \mapsto (y + b) \bmod L$$

Shifted Dissections

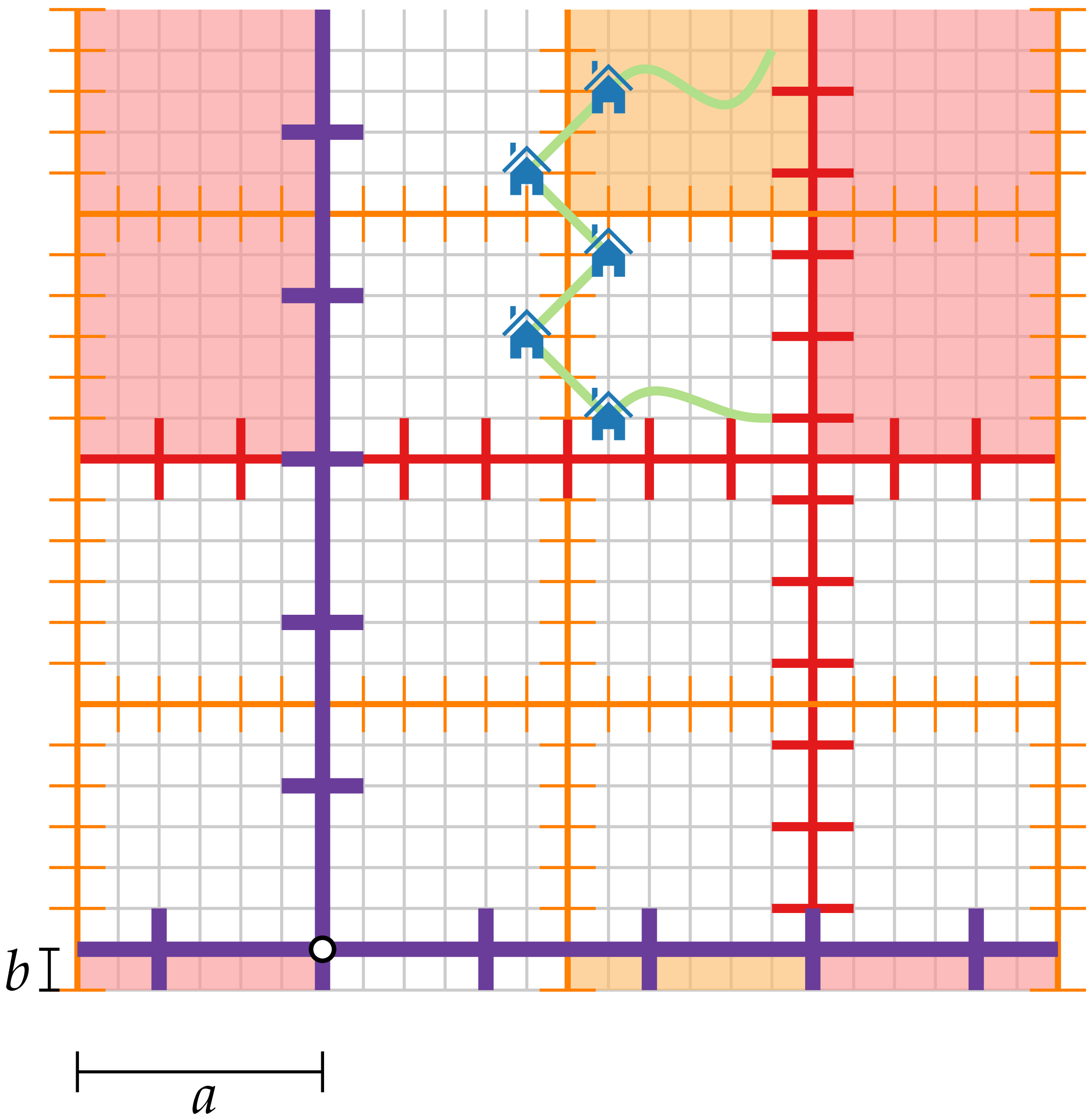


- The best well behaved tour can be a bad approximation.
- Consider an (a, b)-shifted dissection:

$$x \mapsto (x + a) \bmod L$$
$$y \mapsto (y + b) \bmod L$$

- Squares in the dissection tree are "wrapped around".

Shifted Dissections



- The best well behaved tour can be a bad approximation.
- Consider an (a, b)-shifted dissection:

$$x \mapsto (x + a) \bmod L$$
$$y \mapsto (y + b) \bmod L$$

- Squares in the dissection tree are "wrapped around".
- Dynamic program must be modified accordingly.

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid.

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

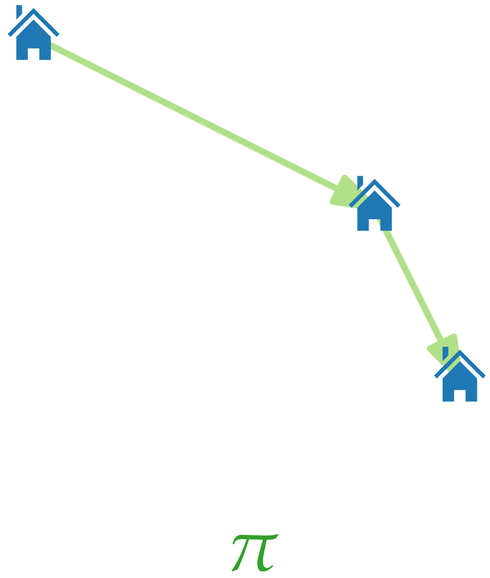
Proof.

- Consider a tour as an ordered cyclic sequence.

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.
- Consider a tour as an ordered cyclic sequence.

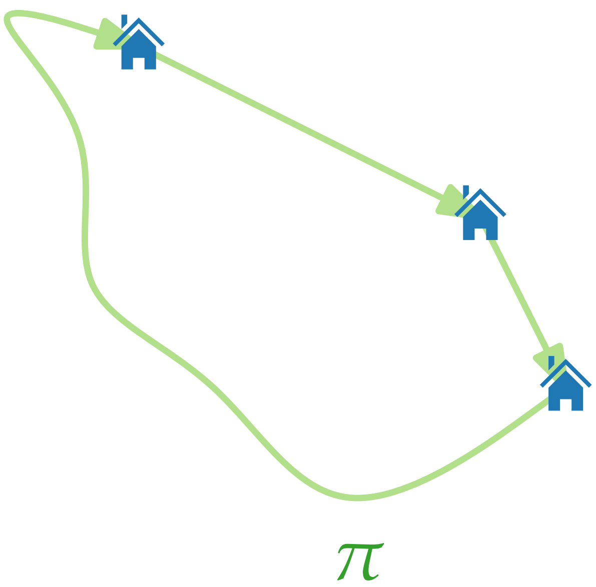


Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.

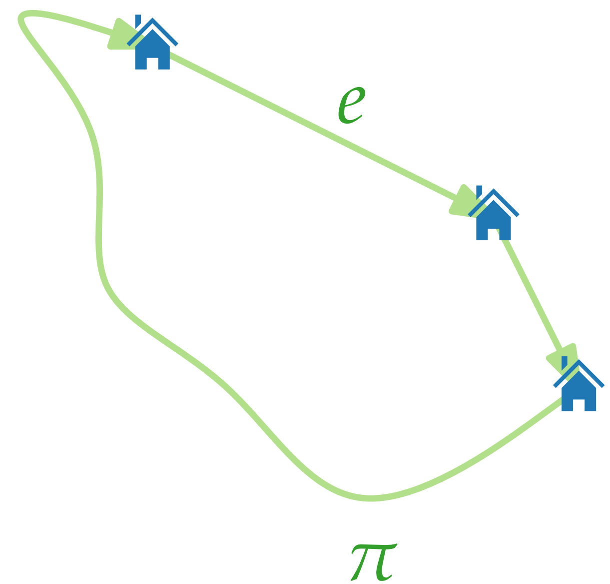


Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \text{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.

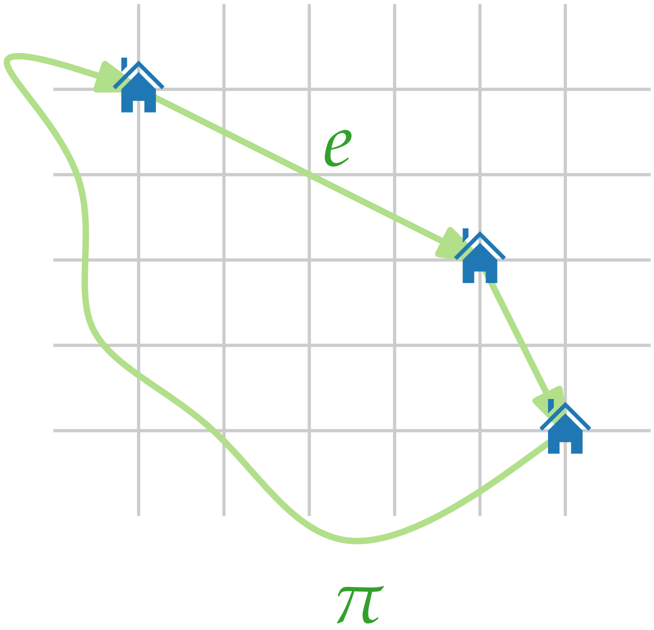


Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.

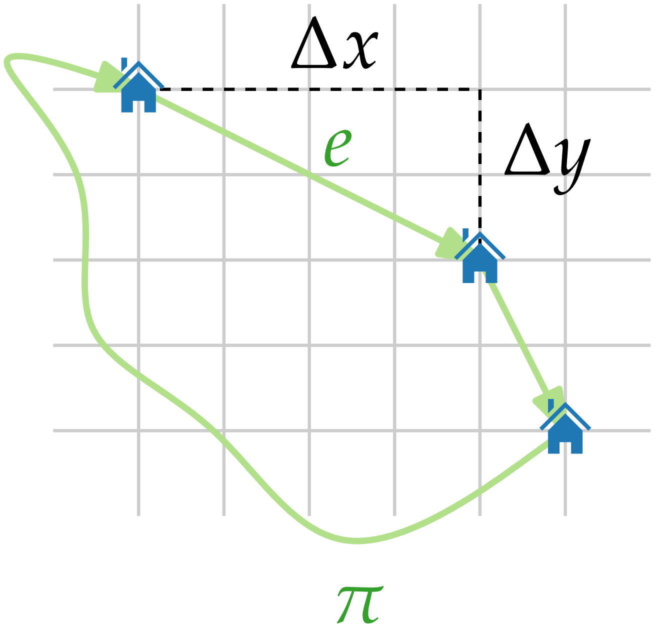


Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.

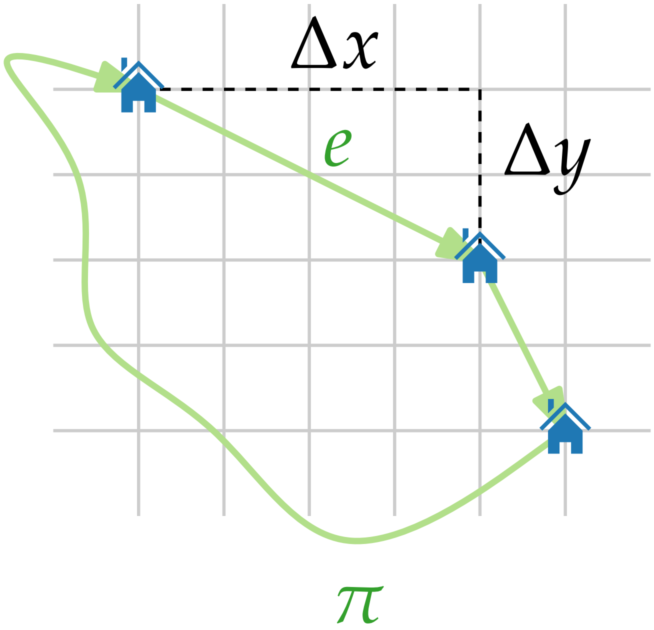


Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.

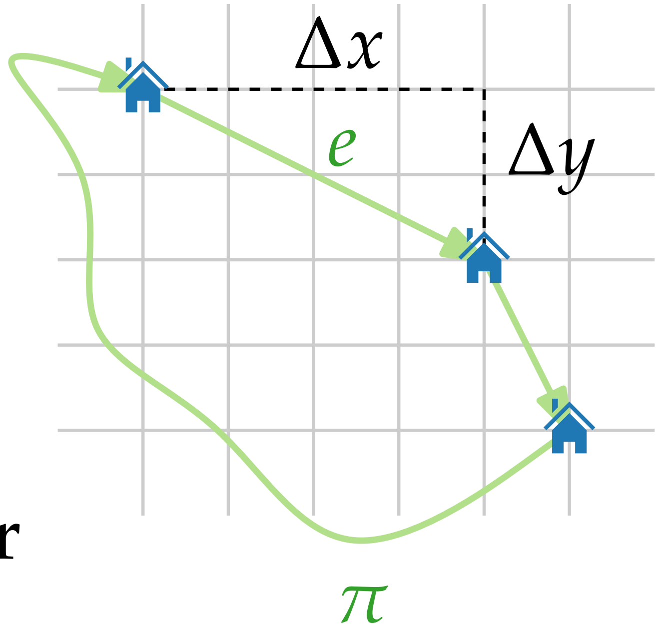


Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \text{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.
- Crossings at the endpoint of an edge are counted for the next edge.

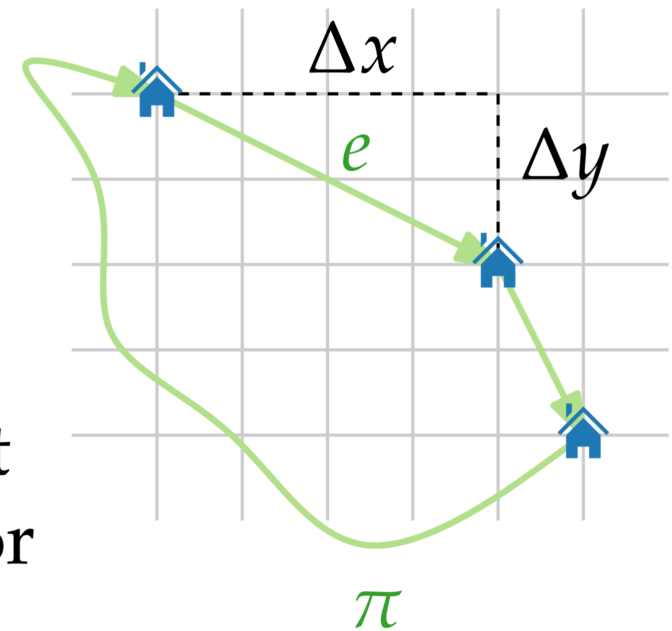


Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.
- Crossings at the endpoint of an edge are counted for the next edge.



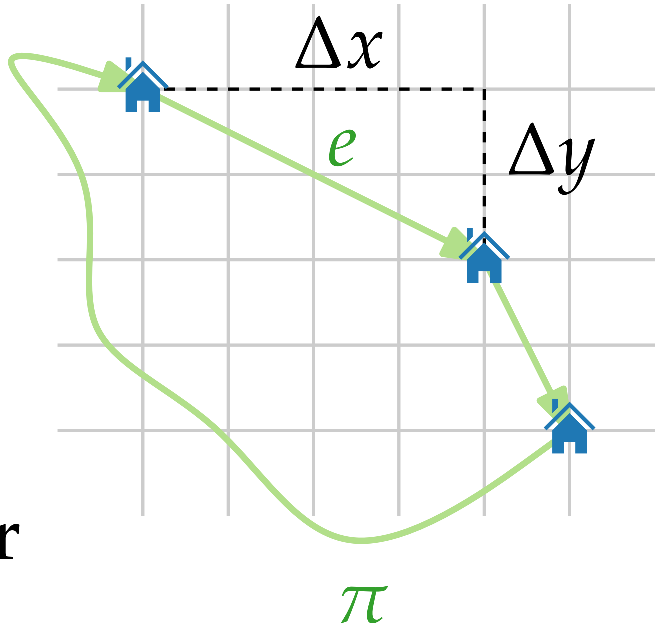
- ${N_e}^2 \leq$

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.
- Crossings at the endpoint of an edge are counted for the next edge.



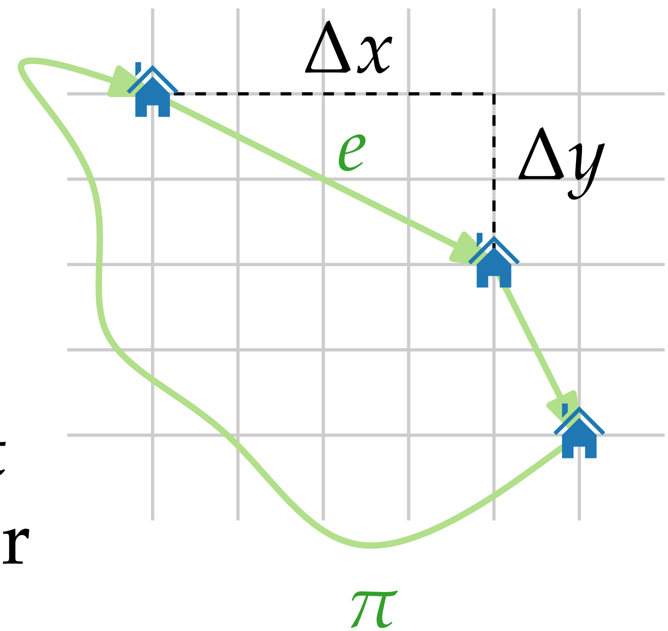
- ${N_e}^2 \leq (\Delta x + \Delta y)^2 \leq$

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \text{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.
- Crossings at the endpoint of an edge are counted for the next edge.



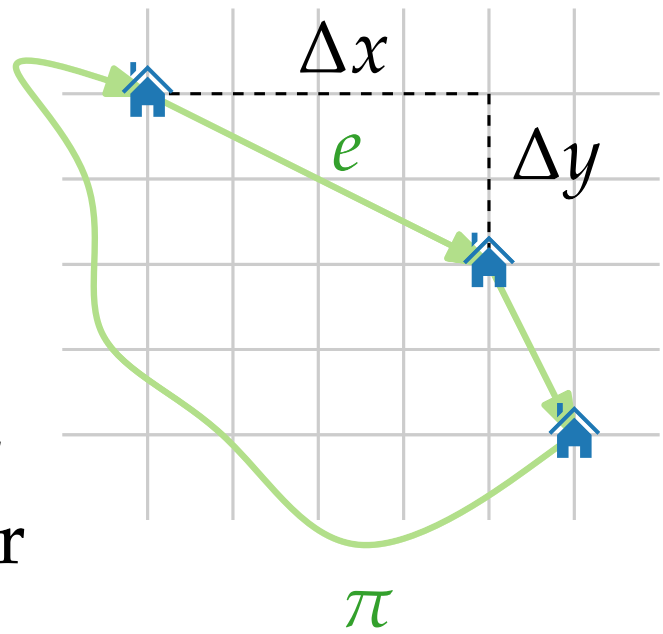
- ${N_e}^2 \leq (\Delta x + \Delta y)^2 \leq 2(\Delta x^2 + \Delta y^2) =$

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.
- Crossings at the endpoint of an edge are counted for the next edge.



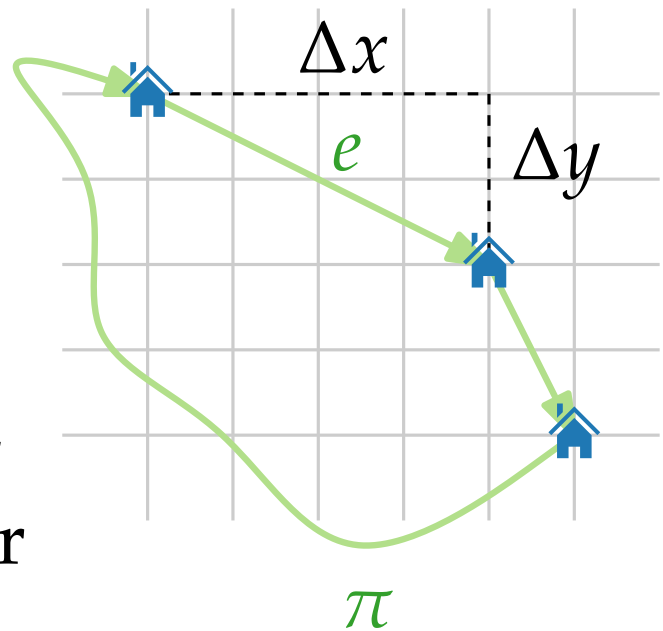
- $N_e{}^2 \leq (\Delta x + \Delta y)^2 \leq 2(\Delta x^2 + \Delta y^2) = 2|e|^2.$

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.
- Crossings at the endpoint of an edge are counted for the next edge.



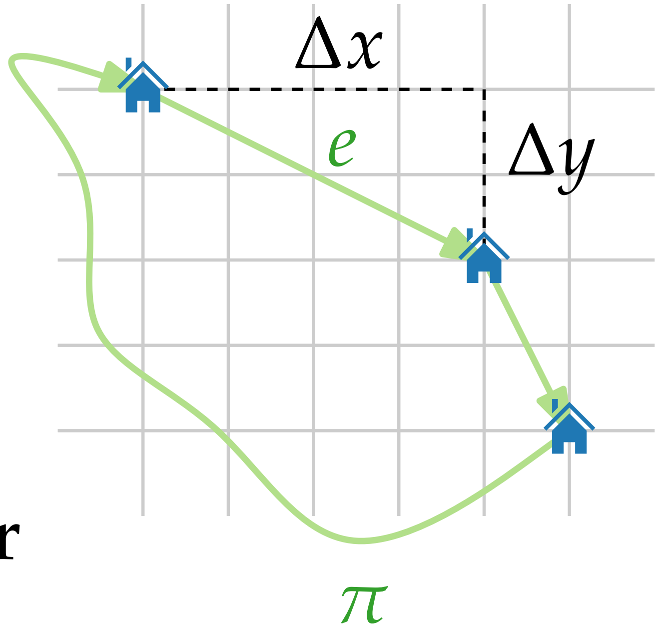
- ${N_e}^2 \leq (\Delta x + \Delta y)^2 \leq 2(\Delta x^2 + \Delta y^2) = 2|e|^2$.
- $N(\pi) =$

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.
- Crossings at the endpoint of an edge are counted for the next edge.



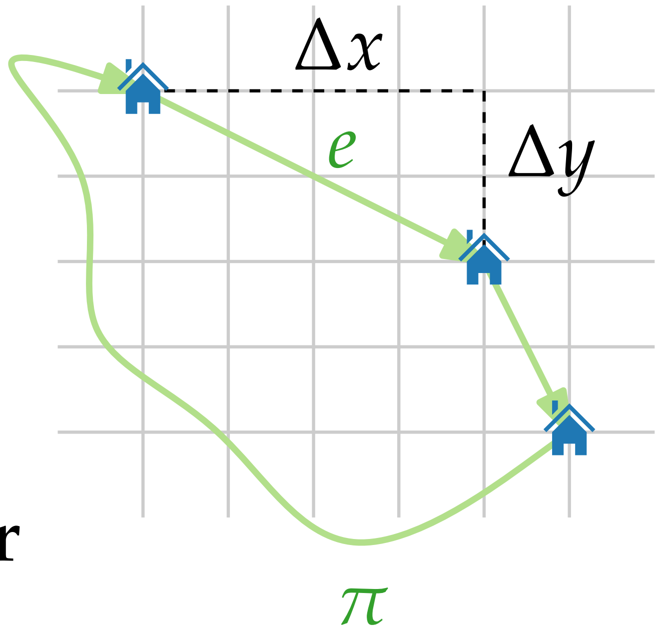
- ${N_e}^2 \leq (\Delta x + \Delta y)^2 \leq 2(\Delta x^2 + \Delta y^2) = 2|e|^2$.
- $N(\pi) = \sum_{e \in \pi} N_e \leq$

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.
- Crossings at the endpoint of an edge are counted for the next edge.



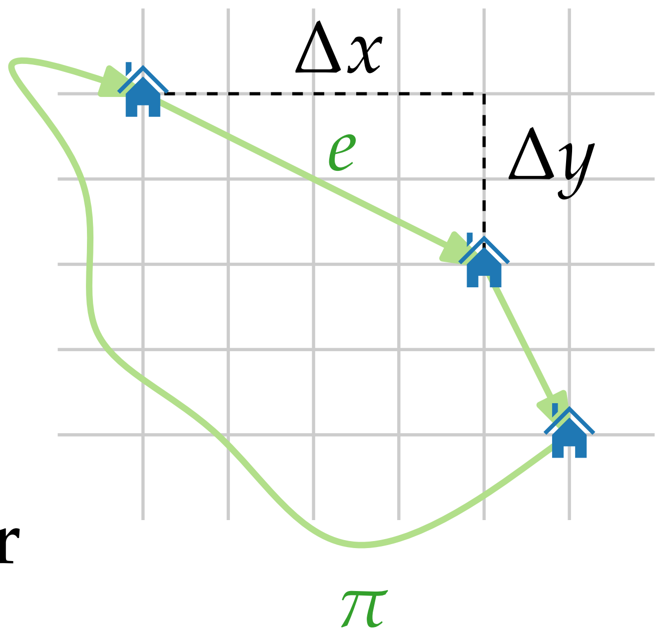
- $N_e{}^2 \leq (\Delta x + \Delta y)^2 \leq 2(\Delta x^2 + \Delta y^2) = 2|e|^2.$
- $N(\pi) = \sum_{e \in \pi} N_e \leq \sum_{e \in \pi} \sqrt{2|e|^2}$

Shifted Dissections (II)

Lemma. Let π be an optimal tour and $N(\pi)$ be the number of crossings of π with the lines of the $(L \times L)$-grid. Then we have $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$.

Proof.

- Consider a tour as an ordered cyclic sequence.
- Each edge e generates $N_e \leq \Delta x + \Delta y$ crossings.
- Crossings at the endpoint of an edge are counted for the next edge.



- ${N_e}^2 \leq (\Delta x + \Delta y)^2 \leq 2(\Delta x^2 + \Delta y^2) = 2|e|^2.$
- $N(\pi) = \sum_{e \in \pi} N_e \leq \sum_{e \in \pi} \sqrt{2|e|^2} = \sqrt{2} \cdot \mathrm{OPT}. \square$

Approximation Algorithms

Lecture 9:
PTAS for EuclideanTSP

Part VI:
Approximation Factor

Joachim Spoerhase

Winter 2021/22

Shifted Dissections (III)

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random.

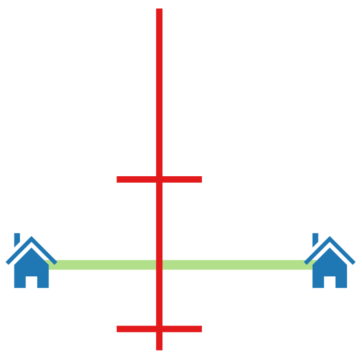
Shifted Dissections (III)

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\mathrm{OPT}$.

Shifted Dissections (III)

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\mathrm{OPT}$.

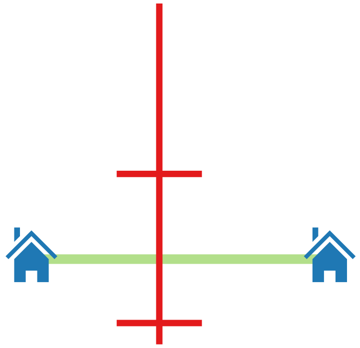
Proof. Consider optimal tour π.



Shifted Dissections (III)

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\text{OPT}$.

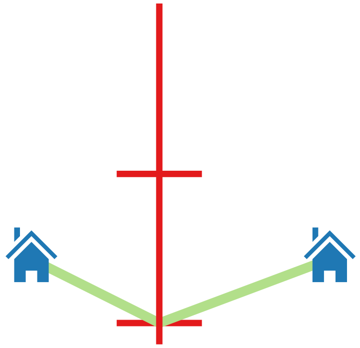
Proof. Consider optimal tour π. Make π well behaved by moving each intersection point with the $(L \times L)$-grid to the nearest portal.



Shifted Dissections (III)

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\text{OPT}$.

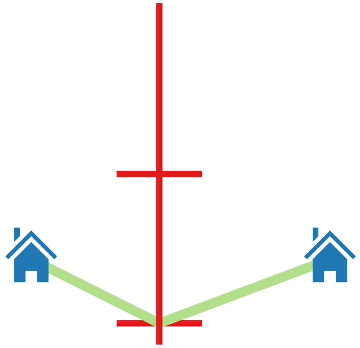
Proof. Consider optimal tour π. Make π well behaved by moving each intersection point with the $(L \times L)$-grid to the nearest portal.



Shifted Dissections (III)

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\mathrm{OPT}$.

Proof. Consider optimal tour π. Make π well behaved by moving each intersection point with the $(L \times L)$-grid to the nearest portal.

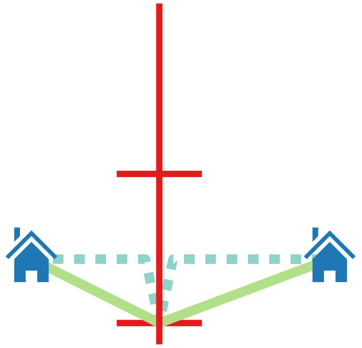


Detour per intersection $\leq$ inter-portal distance.

Shifted Dissections (III)

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\mathrm{OPT}$.

Proof. Consider optimal tour π. Make π well behaved by moving each intersection point with the $(L \times L)$-grid to the nearest portal.



Detour per intersection $\leq$ inter-portal distance.

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* , l is a level-i-line

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of (inter-portal distance).

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of $L/(2^i m)$ (inter-portal distance).

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of $L/(2^i m)$ (inter-portal distance).
- Thus, the expected increase in tour length due to this intersection is at most:

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of $L/(2^i m)$ (inter-portal distance).
- Thus, the expected increase in tour length due to this intersection is at most:

$$\sum_{i=0}^{k}$$

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of $L/(2^i m)$ (inter-portal distance).
- Thus, the expected increase in tour length due to this intersection is at most:

$$\sum_{i=0}^{k} \frac{2^i}{L} .$$

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of $L/(2^i m)$ (inter-portal distance).
- Thus, the expected increase in tour length due to this intersection is at most:

$$\sum_{i=0}^{k} \frac{2^i}{L} \cdot \frac{L}{2^i m} \leq$$

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of $L/(2^i m)$ (inter-portal distance).
- Thus, the expected increase in tour length due to this intersection is at most:

$$\sum_{i=0}^{k} \frac{2^i}{L} \cdot \frac{L}{2^i m} \leq \frac{k+1}{m} \leq$$

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of $L/(2^i m)$ (inter-portal distance).
- Thus, the expected increase in tour length due to this intersection is at most:

$m \in [k/\varepsilon, 2k/\varepsilon]$

$$\sum_{i=0}^{k} \frac{2^i}{L} \cdot \frac{L}{2^i m} \leq \frac{k+1}{m} \leq$$

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of $L/(2^i m)$ (inter-portal distance).
- Thus, the expected increase in tour length due to this intersection is at most:

$m \in [k/\varepsilon, 2k/\varepsilon]$

$$\sum_{i=0}^{k} \frac{2^i}{L} \cdot \frac{L}{2^i m} \leq \frac{k+1}{m} \leq 2\varepsilon.$$

Shifted Dissections (III)

- Consider an intersection point between π and a line l of the $(L \times L)$-grid.
- With probability *at most* $2^i/L$, l is a level-i-line $\rightsquigarrow$ an increase in tour length by a maximum of $L/(2^i m)$ (inter-portal distance).
- Thus, the expected increase in tour length due to this intersection is at most: $m \in [k/\varepsilon, 2k/\varepsilon]$

$$\sum_{i=0}^{k} \frac{2^i}{L} \cdot \frac{L}{2^i m} \leq \frac{k+1}{m} \leq 2\varepsilon.$$

- Summing over all $N(\pi) \leq \sqrt{2} \cdot \mathrm{OPT}$ intersection points, and applying linearity of expectation, provides the claim.

Approximation Scheme

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\mathrm{OPT}$.

Approximation Scheme

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\mathrm{OPT}$.

Theorem. There is a *deterministic* algorithm (PTAS) for EuclideanTSP that provides for every $\varepsilon > 0$ a $(1+\varepsilon)$-approximation in $n^{O(1/\varepsilon)}$ time.

Approximation Scheme

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\mathrm{OPT}$.

Theorem. There is a *deterministic* algorithm (PTAS) for EuclideanTSP that provides for every $\varepsilon > 0$ a $(1+\varepsilon)$-approximation in $n^{O(1/\varepsilon)}$ time.

Proof. Try all L^2 many (a, b)-shifted dissections.

Approximation Scheme

Theorem. Let $a, b \in [0, L-1]$ be chosen independently and uniformaly at random. Then the expected cost of an optimal well behaved tour with respect to the (a, b)-shifted dissection is $\leq (1 + 2\sqrt{2}\varepsilon)\mathrm{OPT}$.

Theorem. There is a *deterministic* algorithm (PTAS) for EuclideanTSP that provides for every $\varepsilon > 0$ a $(1 + \varepsilon)$-approximation in $n^{O(1/\varepsilon)}$ time.

Proof. Try all L^2 many (a, b)-shifted dissections. By the previous theorem, one of them is good enough. □