# A Tighter Insertion-based Approximation of the Graph Crossing Number

**Petr Hliněný**

Faculty of Informatics, Masaryk University
Botanická 68a, 602 00 Brno, Czech Rep.

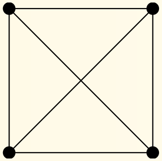
joint work with **Markus Chimani**

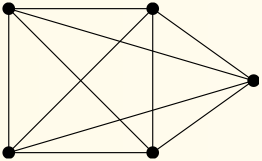
Friedrich-Schiller-University Jena, Germany

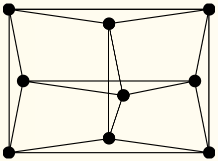
# 1 Graph Crossing Number

**Definition**. *Drawing of a graph* $G$:

- The vertices of $G$ are distinct points, and every edge $e = uv \in E(G)$ is a simple curve joining $u$ to $v$.
- No edge passes through another vertex, and no three edges intersect in a common point.



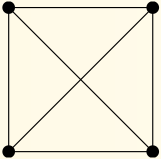


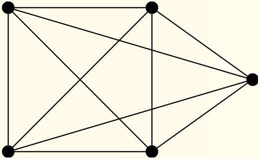


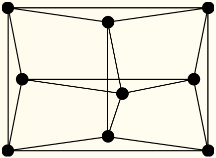
# 1 Graph Crossing Number

**Definition**. *Drawing of a graph $G$*:

- The vertices of $G$ are distinct points,
  and every edge $e = uv \in E(G)$ is a simple curve joining $u$ to $v$.
- No edge passes through another vertex,
  and no three edges intersect in a common point.







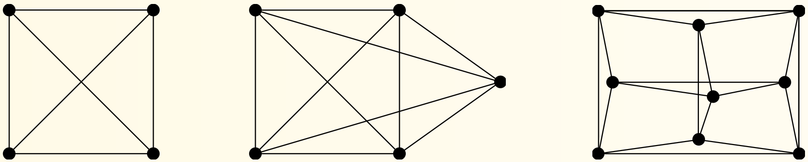
**Definition**. **Crossing number** *cr*$(G)$
is the smallest number of edge crossings in a drawing of $G$.

# 1 Graph Crossing Number

**Definition**. *Drawing of a graph $G$*:

– The vertices of $G$ are distinct points,
and every edge $e = uv \in E(G)$ is a simple curve joining $u$ to $v$.

– No edge passes through another vertex,
and no three edges intersect in a common point.



**Definition**. **Crossing number** $cr(G)$
is the smallest number of edge crossings in a drawing of $G$.

**Warning.** There are slight variations of the definition of crossing number, some giving different numbers! Such as counting *odd-crossing pairs* of edges. [Pelsmajer, Schaeffer, Štefankovič, 2005]...

# Computing the Crossing Number

**Importance**, e.g.

# Computing the Crossing Number

**Importance**, e.g.

- VLSI design, cf. Leighton
- Graph visualization

**What is hard?** i.e., NP-hard

## Computing the Crossing Number

**Importance**, e.g.

- VLSI design, cf. Leighton
- Graph visualization

**What is hard?** i.e., NP-hard

- The general case (of course. . . ); [**Garey and Johnson**, 1983]

## Computing the Crossing Number

**Importance**, e.g.

- VLSI design, cf. Leighton
- Graph visualization

**What is hard?** i.e., NP-hard

- The general case (of course. . . ); [**Garey and Johnson**, 1983]
- The degree-$3$ and *minor-monotone* cases; [PH, 2004]

# Computing the Crossing Number

**Importance**, e.g.

- VLSI design, cf. Leighton
- Graph visualization

**What is hard?** i.e., NP-hard

- The general case (of course...); [**Garey and Johnson**, 1983]
- The degree-$3$ and *minor-monotone* cases; [PH, 2004]
- Even fixed *rotation scheme*; [**Pelsmajer, Schaeffer, Štefankovič**, 2007]

## Computing the Crossing Number

**Importance**, e.g.

- VLSI design, cf. Leighton
- Graph visualization

**What is hard?** i.e., NP-hard

- The general case (of course…); [**Garey and Johnson**, 1983]
- The degree-$3$ and *minor-monotone* cases; [PH, 2004]
- Even fixed *rotation scheme*; [**Pelsmajer, Schaeffer, Štefankovič**, 2007]
- Much worse – hard already for planar graphs plus one edge !
  [**Cabello and Mohar**, 2010]

Can anything be computed efficiently?

## So, what is efficiently computable?

- The case of *cubic* planar graphs plus one edge; [**Riskin**, 1996]

## So, what is efficiently computable?

- The case of *cubic* planar graphs plus one edge; [**Riskin**, 1996]
- *FPT* when parameterized by itself;
  [**Grohe**, 2001], [**Kawarabayashi and Reed**, 2007]

## So, what is efficiently computable?

- The case of *cubic* planar graphs plus one edge; [**Riskin**, 1996]
- *FPT* when parameterized by itself;
  [**Grohe**, 2001], [**Kawarabayashi and Reed**, 2007]
- An exact *branch & bound* approach for “real-world” graphs on up to $\sim 100$ vertices; [**Chimani, Mutzel, and Bomze**, 2008]

## So, what is efficiently computable?

- The case of *cubic* planar graphs plus one edge; [**Riskin**, 1996]
- *FPT* when parameterized by itself;
  [**Grohe**, 2001], [**Kawarabayashi and Reed**, 2007]
- An exact *branch & bound* approach for "real-world" graphs on up to $\sim 100$ vertices; [**Chimani, Mutzel, and Bomze**, 2008]
- NO rich *natural graph class* with nontrivial and yet efficiently computable crossing number problem is known...

## So, what is efficiently computable?

- The case of *cubic* planar graphs plus one edge; [**Riskin**, 1996]
- *FPT* when parameterized by itself; [**Grohe**, 2001], [**Kawarabayashi and Reed**, 2007]
- An exact *branch & bound* approach for "real-world" graphs on up to $\sim 100$ vertices; [**Chimani, Mutzel, and Bomze**, 2008]
- NO rich *natural graph class* with nontrivial and yet efficiently computable crossing number problem is known...

## Approximations, at least?

- Up to factor $\log^3|V(G)|$ $(\log^2 \cdot)$ for $cr(G)+|V(G)|$ with bounded degrees; [**Even, Guha and Schieber**, 2002]

## So, what is efficiently computable?

- The case of *cubic* planar graphs plus one edge; [**Riskin**, 1996]
- *FPT* when parameterized by itself;
  [**Grohe**, 2001], [**Kawarabayashi and Reed**, 2007]
- An exact *branch & bound* approach for "real-world" graphs on up to $\sim 100$ vertices; [**Chimani, Mutzel, and Bomze**, 2008]
- NO rich *natural graph class* with nontrivial and yet efficiently computable crossing number problem is known...

## Approximations, at least?

- Up to factor $\log^3 |V(G)|$ $(\log^2 \cdot)$ for $cr(G)+|V(G)|$ with bounded degrees;
  [**Even, Guha and Schieber**, 2002]
- Constant factors for surface-embedded bounded-degree graphs;
  [**Gitler et al**, 2007], [**PH and Salazar**, 2007], [**PH and Chimani**, 2010]

# 2 Planar Insertion Problems

**Definition**. Given a planar graph $G$ and a set $F$ of additional edges (vert.?).
Find a *drawing of* $G+F$ minimizing the edge crossings *ins*$(\boldsymbol{G}, \boldsymbol{E})$
such that the subdrawing of $G$ is plane.

# 2 Planar Insertion Problems

**Definition**. Given a planar graph $G$ and a set $F$ of additional edges (vert.?). Find a *drawing of $G + F$* minimizing the edge crossings *ins*$(\boldsymbol{G}, \boldsymbol{E})$ such that the subdrawing of $G$ is plane.

**Particular variants**

- *Single edge insertion*: solvable in linear time using SPQR trees (easily implementable!); [**Gutwenger, Mutzel, and Weiskircher**, 2005]

# 2 Planar Insertion Problems

**Definition**. Given a planar graph $G$ and a set $F$ of additional edges (vert.?). Find a *drawing of $G+F$* minimizing the edge crossings *ins*$(\boldsymbol{G},\boldsymbol{E})$ such that the subdrawing of $G$ is plane.

**Particular variants**

- *Single edge insertion*: solvable in linear time using SPQR trees (easily implementable!); [**Gutwenger, Mutzel, and Weiskircher**, 2005]
- *Single vertex insertion*: solvable in polynomial time; [**Chimani, Gutwenger, Mutzel, and Wolf**, 2009]

# 2 Planar Insertion Problems

**Definition**. Given a planar graph $G$ and a set $F$ of additional edges (vert.?). Find a *drawing of $G + F$* minimizing the edge crossings *ins*$(\boldsymbol{G}, \boldsymbol{E})$ such that the subdrawing of $G$ is plane.

**Particular variants**

- *Single edge insertion*: solvable in linear time using SPQR trees (easily implementable!); [**Gutwenger, Mutzel, and Weiskircher**, 2005]
- *Single vertex insertion*: solvable in polynomial time; [**Chimani, Gutwenger, Mutzel, and Wolf**, 2009]
- *Multiple edge insertion (MEI)*: for general edge set $F$ is NP-complete; [**Ziegler**, 2001]

# 2 Planar Insertion Problems

**Definition**. Given a planar graph $G$ and a set $F$ of additional edges (vert.?). Find a *drawing of $G + F$* minimizing the edge crossings *ins*$(\boldsymbol{G}, \boldsymbol{E})$ such that the subdrawing of $G$ is plane.

**Particular variants**

- *Single edge insertion*: solvable in linear time using SPQR trees (easily implementable!); [**Gutwenger, Mutzel, and Weiskircher**, 2005]
- *Single vertex insertion*: solvable in polynomial time; [**Chimani, Gutwenger, Mutzel, and Wolf**, 2009]
- *Multiple edge insertion (MEI)*: for general edge set $F$ is NP-complete; [**Ziegler**, 2001]

**Remark.** Difficulty of insertion problems comes from possible inequivalent embeddings of $G$.

# Connections between Insertion and Crossing number

- Single edge insertion $\leftrightarrow$ *almost-planar* graph (near-planar) $G + e$

# Connections between Insertion and Crossing number

- Single edge insertion $\leftrightarrow$ *almost-planar* graph (near-planar) $G + e$
  - *cr*$(G + e)$ approximated by *ins*$(G, e)$ up to factor $\Delta(G)$; [**PH and Salazar**, 2006]
  - factor $\lfloor \Delta(G)/2 \rfloor$, tight; [**Cabello and Mohar**, 2008]

## Connections between Insertion and Crossing number

- Single edge insertion $\leftrightarrow$ *almost-planar* graph (near-planar) $G + e$
  - $cr(G + e)$ approximated by $ins(G, e)$ up to factor $\Delta(G)$; [**PH and Salazar**, 2006]
  - factor $\lfloor \Delta(G)/2 \rfloor$, tight; [**Cabello and Mohar**, 2008]
- Single vertex insertion $\leftrightarrow$ *apex* graph $G + x$ (specif. neighbourhood)

## Connections between Insertion and Crossing number

- Single edge insertion $\leftrightarrow$ *almost-planar* graph (near-planar) $G+e$
  - *cr*$(G+e)$ approximated by *ins*$(G,e)$ up to factor $\Delta(G)$;
    [**PH and Salazar**, 2006]
  - factor $\lfloor\Delta(G)/2\rfloor$, tight; [**Cabello and Mohar**, 2008]

- Single vertex insertion $\leftrightarrow$ *apex* graph $G+x$ (specif. neighbourhood)
  - *cr*$(G+x)$ approximated by *ins*$(G,x)$ up to factor $d(x)\cdot\lfloor\Delta(G)/2\rfloor$;
    [**Chimani, PH, and Mutzel**, 2008]
  - tight factor – half of that? waiting for Cabello–Mohar's turn. . .

## Connections between Insertion and Crossing number

- Single edge insertion $\leftrightarrow$ *almost-planar* graph (near-planar) $G + e$
  - *cr*$(G + e)$ approximated by *ins*$(G, e)$ up to factor $\Delta(G)$; [**PH and Salazar**, 2006]
  - factor $\lfloor \Delta(G)/2 \rfloor$, tight; [**Cabello and Mohar**, 2008]

- Single vertex insertion $\leftrightarrow$ *apex* graph $G + x$ (specif. neighbourhood)
  - *cr*$(G + x)$ approximated by *ins*$(G, x)$ up to factor $d(x) \cdot \lfloor \Delta(G)/2 \rfloor$; [**Chimani, PH, and Mutzel**, 2008]
  - tight factor – half of that? waiting for Cabello–Mohar's turn...

- Multiple edge insertion $\leftrightarrow$ graph $G + F$ (a very general case)

## Connections between Insertion and Crossing number

- Single edge insertion $\leftrightarrow$ *almost-planar* graph (near-planar) $G+e$
  - *cr*$(G+e)$ approximated by *ins*$(G,e)$ up to factor $\Delta(G)$;
    [**PH and Salazar**, 2006]
  - factor $\lfloor\Delta(G)/2\rfloor$, tight; [**Cabello and Mohar**, 2008]

- Single vertex insertion $\leftrightarrow$ *apex* graph $G+x$ (specif. neighbourhood)
  - *cr*$(G+x)$ approximated by *ins*$(G,x)$ up to factor $d(x)\cdot\lfloor\Delta(G)/2\rfloor$;
    [**Chimani, PH, and Mutzel**, 2008]
  - tight factor – half of that? waiting for Cabello–Mohar's turn. . .

- Multiple edge insertion $\leftrightarrow$ graph $G+F$ (a very general case)
  - *cr*$(G+F)$ approximated by *ins*$(G,F)$;
    [**Chimani, PH, and Mutzel**, 2008]
  - however, *ins*$(G,F)$ is NP-complete! (as well as finding $F$)

# 3 Approximating MEI up to Additive Factor

- [**Chuzhoy, Makarychev, and Sidiropoulos**, 2011 SODA]
  Using MEI, a solution to $cr(G+F)$ for given planar $G$ and $F$, with

  $$\leq O(\Delta(G)^3 \cdot |F| \cdot cr(G+F) + \Delta(G)^3 \cdot |F|^2) \text{ crossings.}$$

# 3 Approximating MEI up to Additive Factor

- [**Chuzhoy, Makarychev, and Sidiropoulos**, 2011 SODA]
  Using MEI, a solution to $\mathit{cr}(G+F)$ for given planar $G$ and $F$, with
  $$\leq\ O(\Delta(G)^3 \cdot |F| \cdot \mathit{cr}(G+F) + \Delta(G)^3 \cdot |F|^2) \text{ crossings.}$$
- Our alternative approach directly focuses on approximating MEI:

# 3 Approximating MEI up to Additive Factor

- [**Chuzhoy, Makarychev, and Sidiropoulos**, 2011 SODA]
  Using MEI, a solution to $\mathit{cr}(G+F)$ for given planar $G$ and $F$, with

  $$\leq O(\Delta(G)^3 \cdot |F| \cdot \mathit{cr}(G+F) + \Delta(G)^3 \cdot |F|^2) \text{ crossings.}$$

- Our alternative approach directly focuses on approximating MEI:
  - only additive approximation factor for MEI $\mathit{ins}(G,F)$,
  - consequently improved multiplicative factor for $\mathit{cr}(G+F)$,
  - and practically implementable using SPQR trees.

# 3 Approximating MEI up to Additive Factor

- [**Chuzhoy, Makarychev, and Sidiropoulos**, 2011 SODA]
  Using MEI, a solution to $cr(G+F)$ for given planar $G$ and $F$, with
  $$\leq O(\Delta(G)^3 \cdot |F| \cdot cr(G+F) + \Delta(G)^3 \cdot |F|^2) \text{ crossings.}$$
- Our alternative approach directly focuses on approximating MEI:
  - only additive approximation factor for MEI $ins(G,F)$,
  - consequently improved multiplicative factor for $cr(G+F)$,
  - and practically implementable using SPQR trees.

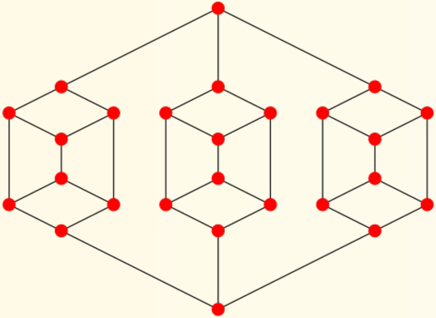
**Theorem 1.** *Given a conn. planar graph $G$ and an edge set $F$, $F \cap E(G) = \emptyset$, Algorithm 2 described below finds, in*

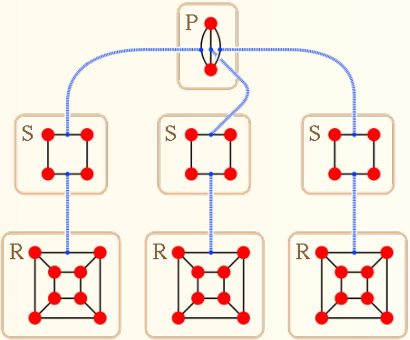
$$O(|F| \cdot |V(G)| + |F|^2) \quad \textit{time,}$$

*an approximate solution to the MEI problem for $G$ and $F$ with*

$$\leq \; ins(G,F) + \left(\lfloor \tfrac{1}{2}\Delta(G) \rfloor + \tfrac{1}{2}\right) \cdot (|F|^2 - |F|) \quad \textit{crossings.}$$

# Gentle introduction to SPQR trees






- Graph broken into the *blocks* first.
- Then, for pairwise gluing on *virtual skeleton edges*, we have got
  - *S-nodes* for serial skeletons,
  - *P-nodes* for parallel skeletons,
  - *R-nodes* for $3$-connected components.

## The algorithm for MEI

- *Con-tree* = a combination of a block-cut tree with SPQR trees.

## The algorithm for MEI

- *Con-tree* = a combination of a block-cut tree with SPQR trees.

  *Con-chain* = a path traversing the con-tree nodes relevant for inserting a specific edge; only the *C-, P-, and R-nodes* on it do matter.

## The algorithm for MEI

- *Con-tree* $=$ a combination of a block-cut tree with SPQR trees.

  *Con-chain* $=$ a path traversing the con-tree nodes relevant for inserting a specific edge; only the *C-, P-, and R-nodes* on it do matter.

**Algorithm 2.** Computing an approximate solution to the **multiple edge insertion problem** for a connected planar graph $G$ and new edges $F$.

1. Build the con-tree $\mathcal{C}$ of $G$.

## The algorithm for MEI

- *Con-tree* $=$ a combination of a block-cut tree with SPQR trees.

  *Con-chain* $=$ a path traversing the con-tree nodes relevant for inserting a specific edge; only the *C-, P-, and R-nodes* on it do matter.

**Algorithm 2.** Computing an approximate solution to the **multiple edge insertion problem** for a connected planar graph $G$ and new edges $F$.

1. Build the con-tree $\mathcal{C}$ of $G$.
2. Using $\mathcal{C}$, compute *single-edge insertions* (the con-chains) for each edge $e \in F$ independently,
   and centrally store their *embedding preferences*.

## The algorithm for MEI

- *Con-tree* = a combination of a block-cut tree with SPQR trees.

  *Con-chain* = a path traversing the con-tree nodes relevant for inserting a specific edge; only the *C-, P-, and R-nodes* on it do matter.

**Algorithm 2.** Computing an approximate solution to the **multiple edge insertion problem** for a connected planar graph $G$ and new edges $F$.

1. Build the con-tree $\mathcal{C}$ of $G$.

2. Using $\mathcal{C}$, compute *single-edge insertions* (the con-chains) for each edge $e \in F$ independently,
   and centrally store their *embedding preferences*.

3. Fix an *embedding $\Gamma$ of $G$* by suitably combining the embedding preferences from step 2 (at least "one happy con-chain per node").

## The algorithm for MEI

- *Con-tree* $=$ a combination of a block-cut tree with SPQR trees.

  *Con-chain* $=$ a path traversing the con-tree nodes relevant for inserting a specific edge; only the *C-, P-, and R-nodes* on it do matter.

**Algorithm 2.** Computing an approximate solution to the **multiple edge insertion problem** for a connected planar graph $G$ and new edges $F$.

1. Build the con-tree $\mathcal{C}$ of $G$.
2. Using $\mathcal{C}$, compute *single-edge insertions* (the con-chains) for each edge $e \in F$ independently,
   and centrally store their *embedding preferences*.
3. Fix an *embedding $\Gamma$ of $G$* by suitably combining the embedding preferences from step 2 (at least "one happy con-chain per node").
4. Independently compute the *insertion paths* for each edge $e \in F$ into the fixed embedding $\Gamma$, as shortest dual paths.

## Proof sketch

A very informal one, neglecting all technical obstacles. . .


- Identify *dirty passes* of con-chains – where the con-chain embedding preferences are not happy with the fixed embedding $\Gamma$.

# Proof sketch

A very informal one, neglecting all technical obstacles. . .


- Identify *dirty passes* of con-chains – where the con-chain embedding preferences are not happy with the fixed embedding $\Gamma$.
- Observe that con-chains rooted through the same neighbourhood are either both happy or both unhappy there.

# Proof sketch

A very informal one, neglecting all technical obstacles. . .


- Identify *dirty passes* of con-chains – where the con-chain embedding preferences are not happy with the fixed embedding $\Gamma$.
- Observe that con-chains rooted through the same neighbourhood are either both happy or both unhappy there.
- As every node has some happy con-chain, each dirty pass can be linked to a pair of con-chains that *split/merge* at that pass.

## Proof sketch

A very informal one, neglecting all technical obstacles. . .


- Identify *dirty passes* of con-chains – where the con-chain embedding preferences are not happy with the fixed embedding $\Gamma$.
- Observe that con-chains rooted through the same neighbourhood are either both happy or both unhappy there.
- As every node has some happy con-chain, each dirty pass can be linked to a pair of con-chains that *split/merge* at that pass.
- Two con-chains can split/merge twice, hence $\leq 2\binom{|F|}{2}$ dirty passes.

## Proof sketch

A very informal one, neglecting all technical obstacles. . .


- Identify *dirty passes* of con-chains – where the con-chain embedding preferences are not happy with the fixed embedding $\Gamma$.
- Observe that con-chains rooted through the same neighbourhood are either both happy or both unhappy there.
- As every node has some happy con-chain, each dirty pass can be linked to a pair of con-chains that *split/merge* at that pass.
- Two con-chains can split/merge twice, hence $\leq 2\binom{|F|}{2}$ dirty passes.
- Every dirty pass is associated with a $1$- *or* $2$-*cut*, and the inserted edge needs $\leq \lfloor \Delta(G)/2 \rfloor$ crossings to "pass by" it. Altogether

$$\leq \mathit{ins}(G,F) + \left(2\left\lfloor\frac{\Delta(G)}{2}\right\rfloor + 1\right)\cdot\binom{|F|}{2}. \qquad \square$$

# 4 Consequences

**Theorem 3.** *Given a planar graph $G$ and an edge set $F$, $F \cap E(G) = \emptyset$, Algorithm 2 finds an approximate solution to* cr$(G+F)$ *with*

$$\leq \lfloor \tfrac{1}{2}\Delta(G) \rfloor \cdot 2|F| \cdot \mathit{cr}(G+F) + (\lfloor \tfrac{1}{2}\Delta(G) \rfloor + \tfrac{1}{2})(|F|^2 - |F|) \ \ \textit{crossings.}$$

# 4 Consequences

**Theorem 3.** *Given a planar graph $G$ and an edge set $F$, $F \cap E(G) = \emptyset$, Algorithm 2 finds an approximate solution to* $\mathit{cr}(G+F)$ *with*

$$\leq \lfloor \tfrac{1}{2}\Delta(G) \rfloor \cdot 2|F| \cdot \mathit{cr}(G+F) + (\lfloor \tfrac{1}{2}\Delta(G) \rfloor + \tfrac{1}{2})(|F|^2 - |F|) \ \textit{ crossings.}$$

- This improves over previous $O(\Delta(G)^3 \cdot |F| \cdot \mathit{cr}(G+F) + \Delta(G)^3 \cdot |F|^2)$

# 4 Consequences

**Theorem 3.** *Given a planar graph $G$ and an edge set $F$, $F \cap E(G) = \emptyset$, Algorithm 2 finds an approximate solution to cr$(G+F)$ with*

$$\leq \lfloor \tfrac{1}{2}\Delta(G) \rfloor \cdot 2|F| \cdot \mathit{cr}(G+F) + (\lfloor \tfrac{1}{2}\Delta(G) \rfloor + \tfrac{1}{2})(|F|^2 - |F|) \;\; \textit{crossings.}$$

- This improves over previous $O(\Delta(G)^3 \cdot |F| \cdot \mathit{cr}(G+F) + \Delta(G)^3 \cdot |F|^2)$
- ... with a simpler algorithm and a simpler proof.

# 4 Consequences

**Theorem 3.** *Given a planar graph $G$ and an edge set $F$, $F \cap E(G) = \emptyset$, Algorithm 2 finds an approximate solution to cr$(G+F)$ with*

$$\leq \lfloor \tfrac{1}{2}\Delta(G) \rfloor \cdot 2|F| \cdot \mathit{cr}(G+F) + (\lfloor \tfrac{1}{2}\Delta(G) \rfloor + \tfrac{1}{2})(|F|^2 - |F|) \quad \textit{crossings.}$$

- This improves over previous $O(\Delta(G)^3 \cdot |F| \cdot \mathit{cr}(G+F) + \Delta(G)^3 \cdot |F|^2)$
- ...with a simpler algorithm and a simpler proof.

# 5 Final Remark and Question

- In the MEI problem, the $O(\Delta(G) \cdot |F|^2)$ additive factor should be replaced with as. tight
$$O(\Delta(G) \cdot |F| \log |F| + |F|^2) \, .$$

# 4 Consequences

**Theorem 3.** *Given a planar graph $G$ and an edge set $F$, $F \cap E(G) = \emptyset$, Algorithm 2 finds an approximate solution to cr$(G+F)$ with*

$$\leq \lfloor \tfrac{1}{2}\Delta(G)\rfloor \cdot 2|F| \cdot \mathit{cr}(G+F) + (\lfloor \tfrac{1}{2}\Delta(G)\rfloor + \tfrac{1}{2})(|F|^2 - |F|) \ \ \textit{crossings.}$$

- This improves over previous $O(\Delta(G)^3 \cdot |F| \cdot \mathit{cr}(G+F) + \Delta(G)^3 \cdot |F|^2)$
- . . . with a simpler algorithm and a simpler proof.

# 5 Final Remark and Question

- In the MEI problem, the $O(\Delta(G) \cdot |F|^2)$ additive factor should be replaced with as. tight
$$O(\Delta(G) \cdot |F| \log |F| + |F|^2)\,.$$

- Can the MEI $(G, F)$ problem have, say, an FPT algorithm wrt. $|F|$?