Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Chair for
INFORMATICS I
Efficient Algorithms and
Knowledge-Based Systems

Institute for Informatics

Faster Force-Directed Graph Drawing with the Well-Separated Pair Decomposition

Fabian Lipp Alexander Wolff Johannes Zink

Julius-Maximilians-Universität Würzburg

September 24, Graph Drawing 2015

Introduction

Classical force-directed algorithm

- [Eades, 1984, Fruchterman & Reingold, 1991]
- Attracting forces: $O(m)$ time with $m = \#$edges
- Repulsive forces: $\Theta(n^2)$ time with $n = \#$vertices

Introduction

Classical force-directed algorithm

- [Eades, 1984, Fruchterman & Reingold, 1991]
- Attracting forces: $O(m)$ time with $m = \#\text{edges}$
- Repulsive forces: $\Theta(n^2)$ time with $n = \#\text{vertices}$

Many speed-up techniques for force-directed algorithms known:

- Hierarchical $O(n \log n)$ force-calculation [Barnes & Hut, 1986]
- Potential-field-based multilevel algorithm [Hachul & Jünger, 2005]
- Evaluation of multilevel layout methods [Bartel et al., 2011]

Introduction

Classical force-directed algorithm

- [Eades, 1984, Fruchterman & Reingold, 1991]
- Attracting forces: $O(m)$ time with $m = \#$edges
- Repulsive forces: $\Theta(n^2)$ time with $n = \#$vertices

Many speed-up techniques for force-directed algorithms known:

- Hierarchical $O(n \log n)$ force-calculation [Barnes & Hut, 1986]
- Potential-field-based multilevel algorithm [Hachul & Jünger, 2005]
- Evaluation of multilevel layout methods [Bartel et al., 2011]

Our approach

- Using well-separated pair decomposition to speed up repulsive force calculation to $O(n \log n)$ time.
- Can be combined with other speed-up techniques.

Well-Separated Pairs

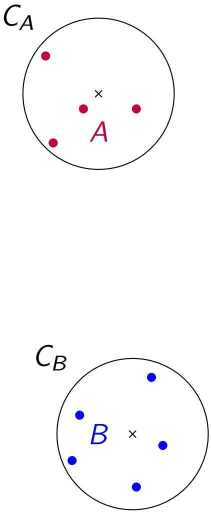
Definition

Let $s > 0$. A and B in $\mathbb{R}^2$ are s-well-separated iff there are two balls C_A and C_B with

- $A \subseteq C_A$ and $B \subseteq C_B$,
- radius $r := r(C_A) = r(C_B)$, and
- distance $\geq s \cdot r$.

[Callahan & Kosaraju, 1995]

Well-Separated Pairs



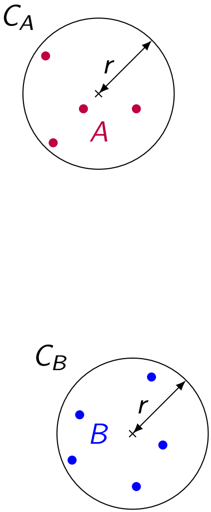
Definition

Let $s > 0$. A and B in $\mathbb{R}^2$ are s-well-separated iff there are two balls C_A and C_B with

- $A \subseteq C_A$ and $B \subseteq C_B$,
- radius $r := r(C_A) = r(C_B)$, and
- distance $\geq s \cdot r$.

[Callahan & Kosaraju, 1995]

Well-Separated Pairs



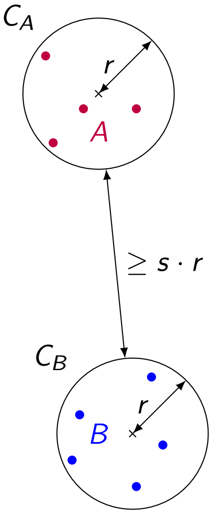
Definition

Let $s > 0$. A and B in $\mathbb{R}^2$ are s-well-separated iff there are two balls C_A and C_B with

- $A \subseteq C_A$ and $B \subseteq C_B$,
- radius $r := r(C_A) = r(C_B)$, and
- distance $\geq s \cdot r$.

[Callahan & Kosaraju, 1995]

Well-Separated Pairs



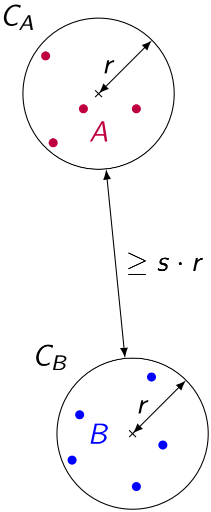
Definition

Let $s > 0$. A and B in $\mathbb{R}^2$ are s-well-separated iff there are two balls C_A and C_B with

- $A \subseteq C_A$ and $B \subseteq C_B$,
- radius $r := r(C_A) = r(C_B)$, and
- distance $\geq s \cdot r$.

[Callahan & Kosaraju, 1995]

Well-Separated Pairs



Definition

Let $s > 0$. A and B in $\mathbb{R}^2$ are s-well-separated iff there are two balls C_A and C_B with

- $A \subseteq C_A$ and $B \subseteq C_B$,
- radius $r := r(C_A) = r(C_B)$, and
- distance $\geq s \cdot r$.

[Callahan & Kosaraju, 1995]

Well-Separated Pair Decomposition (WSPD)

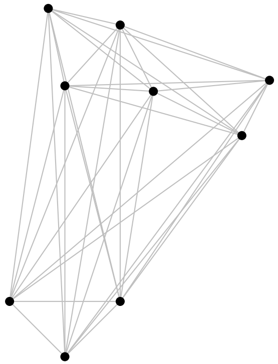
Definition

Let S be a finite set in $\mathbb{R}^2$. A WSPD for S with respect to $s > 0$ is a sequence

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_k, B_k\}$$

of s-well-separated pairs A_i, B_i such that

- $A_i, B_i \subseteq S$, and
- $\forall a \neq b \in S$ there is exactly one i with $a \in A_i$ and $b \in B_i$ or vice versa.



Well-Separated Pair Decomposition (WSPD)

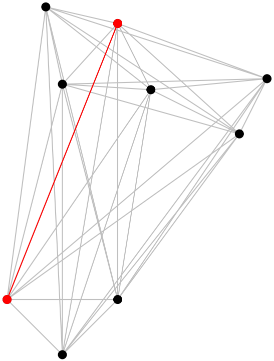
Definition

Let S be a finite set in $\mathbb{R}^2$. A WSPD for S with respect to $s > 0$ is a sequence

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_k, B_k\}$$

of s-well-separated pairs A_i, B_i such that

- $A_i, B_i \subseteq S$, and
- $\forall a \neq b \in S$ there is exactly one i with $a \in A_i$ and $b \in B_i$ or vice versa.



Well-Separated Pair Decomposition (WSPD)

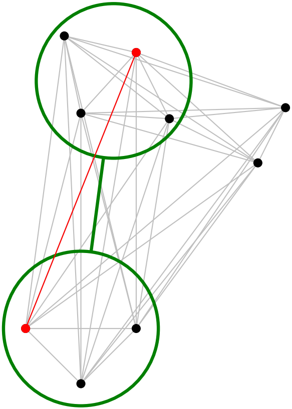
Definition

Let S be a finite set in $\mathbb{R}^2$. A WSPD for S with respect to $s > 0$ is a sequence

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_k, B_k\}$$

of s-well-separated pairs A_i, B_i such that

- $A_i, B_i \subseteq S$, and
- $\forall a \neq b \in S$ there is exactly one i with $a \in A_i$ and $b \in B_i$ or vice versa.



Well-Separated Pair Decomposition (WSPD)

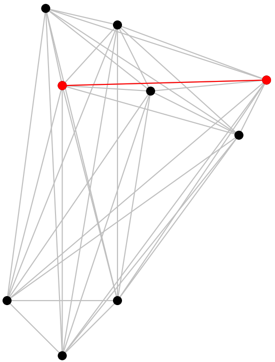
Definition

Let S be a finite set in $\mathbb{R}^2$. A WSPD for S with respect to $s > 0$ is a sequence

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_k, B_k\}$$

of s-well-separated pairs A_i, B_i such that

- $A_i, B_i \subseteq S$, and
- $\forall a \neq b \in S$ there is exactly one i with $a \in A_i$ and $b \in B_i$ or vice versa.



Well-Separated Pair Decomposition (WSPD)

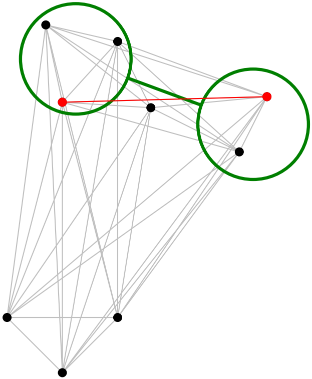
Definition

Let S be a finite set in $\mathbb{R}^2$. A WSPD for S with respect to $s > 0$ is a sequence

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_k, B_k\}$$

of s-well-separated pairs A_i, B_i such that

- $A_i, B_i \subseteq S$, and
- $\forall a \neq b \in S$ there is exactly one i with $a \in A_i$ and $b \in B_i$ or vice versa.



Well-Separated Pair Decomposition (WSPD)

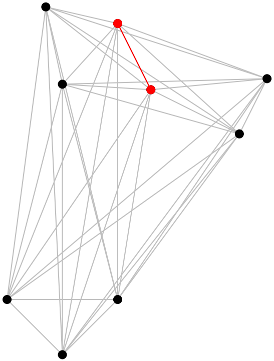
Definition

Let S be a finite set in $\mathbb{R}^2$. A WSPD for S with respect to $s > 0$ is a sequence

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_k, B_k\}$$

of s-well-separated pairs A_i, B_i such that

- $A_i, B_i \subseteq S$, and
- $\forall a \neq b \in S$ there is exactly one i with $a \in A_i$ and $b \in B_i$ or vice versa.



Well-Separated Pair Decomposition (WSPD)

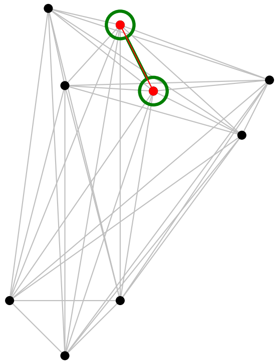
Definition

Let S be a finite set in $\mathbb{R}^2$. A WSPD for S with respect to $s > 0$ is a sequence

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_k, B_k\}$$

of s-well-separated pairs A_i, B_i such that

- $A_i, B_i \subseteq S$, and
- $\forall a \neq b \in S$ there is exactly one i with $a \in A_i$ and $b \in B_i$ or vice versa.



Computation of WSPD

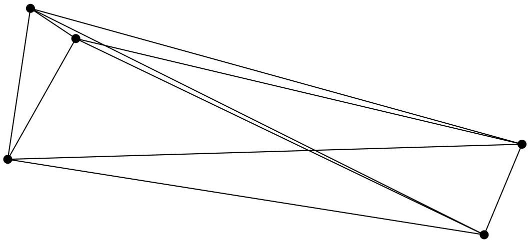
Theorem

Let S be a set of n points in $\mathbb{R}^2$. For any $s > 0$, a WSPD consisting of $O(n)$ pairs can be computed in $O(n \log n)$ time.

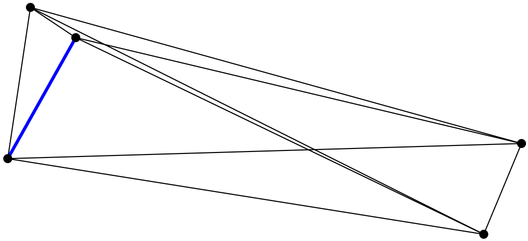
[Callahan & Kosaraju, 1995]

Repulsive Force Calculation

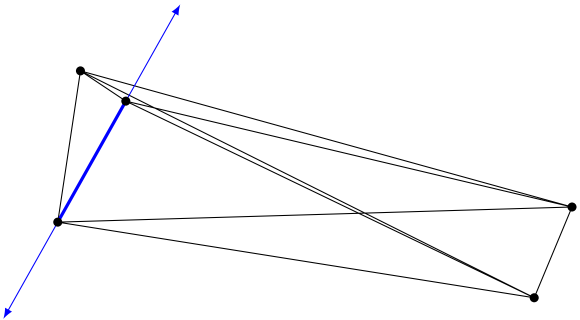
Repulsive Force Calculation



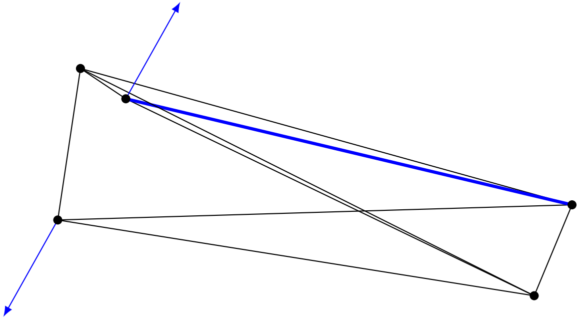
Repulsive Force Calculation



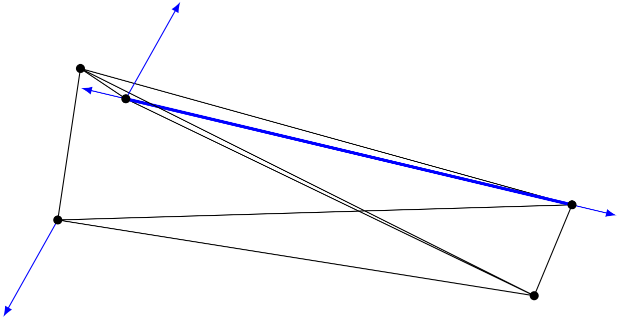
Repulsive Force Calculation



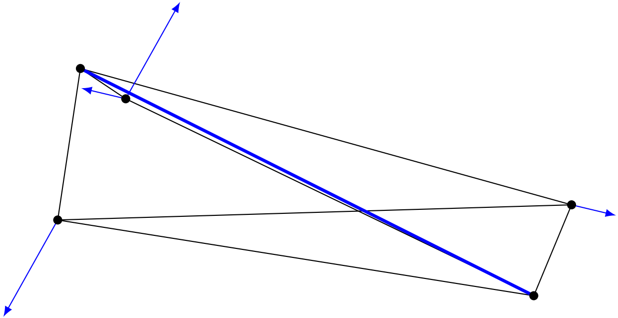
Repulsive Force Calculation



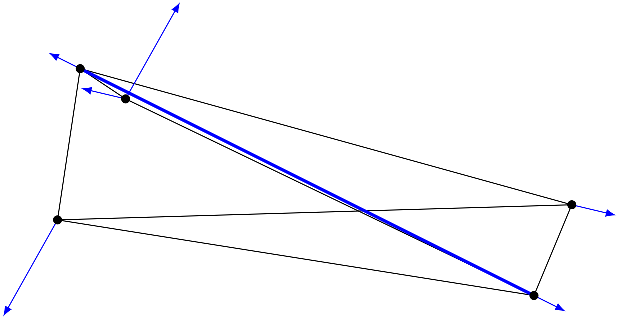
Repulsive Force Calculation



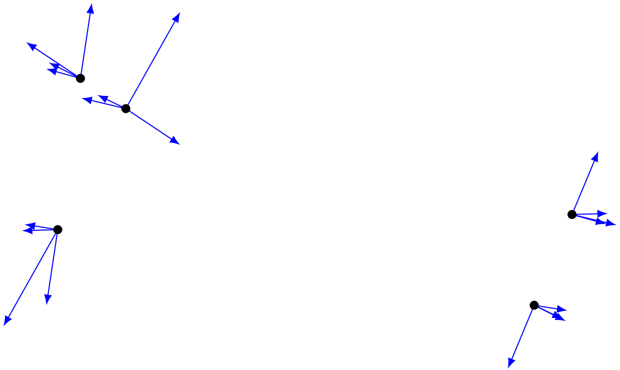
Repulsive Force Calculation



Repulsive Force Calculation

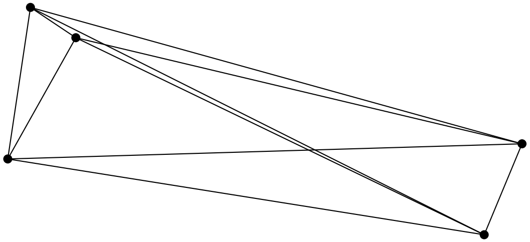


Repulsive Force Calculation

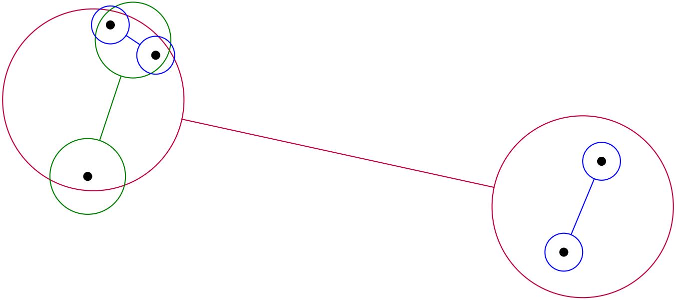


Compute forces for $\Theta(n^2)$ pairs of vertices

Repulsive Force Calculation

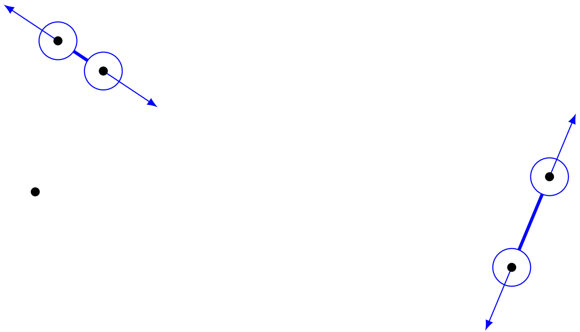


Repulsive Force Calculation

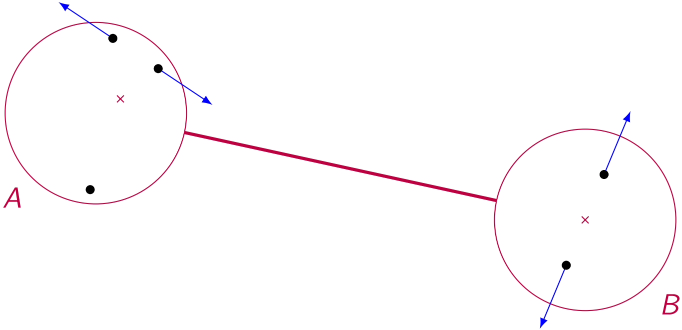


Repulsive Force Calculation

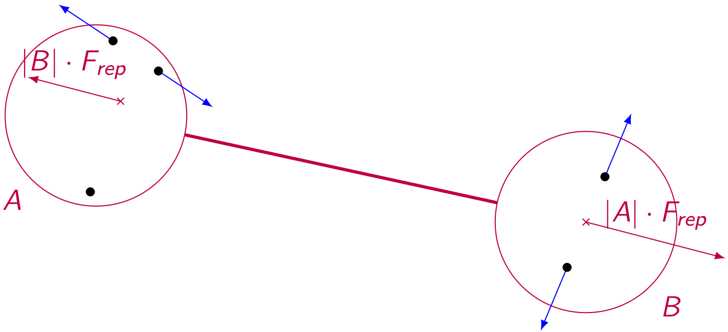
Repulsive Force Calculation



Repulsive Force Calculation



Repulsive Force Calculation



Repulsive Force Calculation

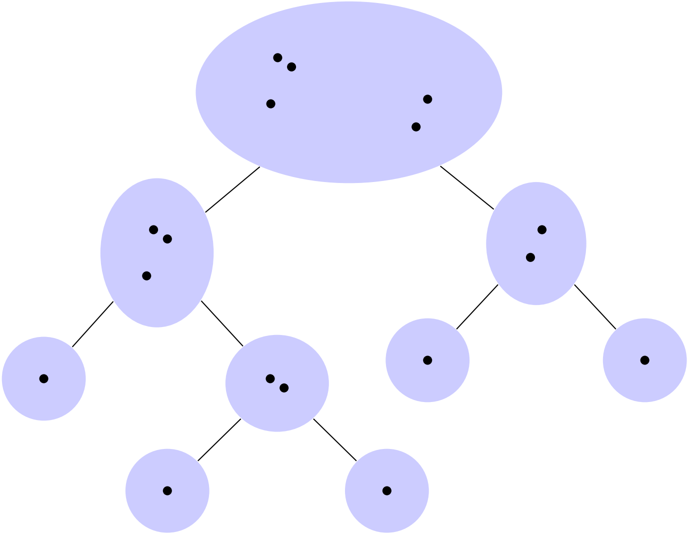
Repulsive Force Calculation

Repulsive Force Calculation

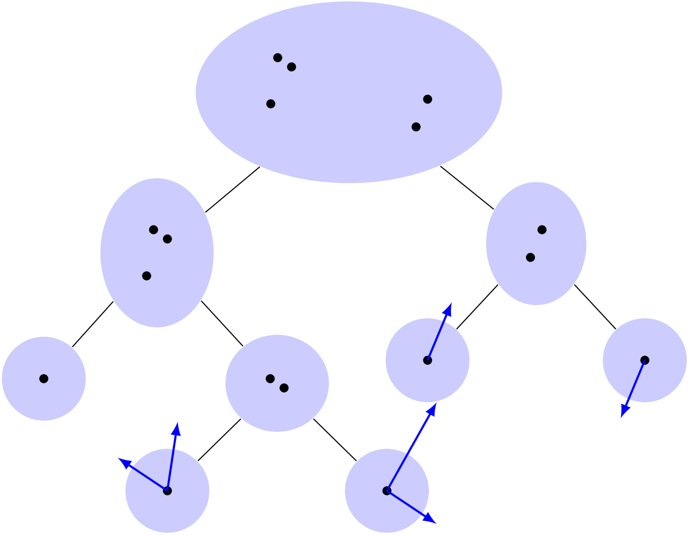
Repulsive Force Calculation

Compute forces for $k \in O(n)$ well-separated pairs

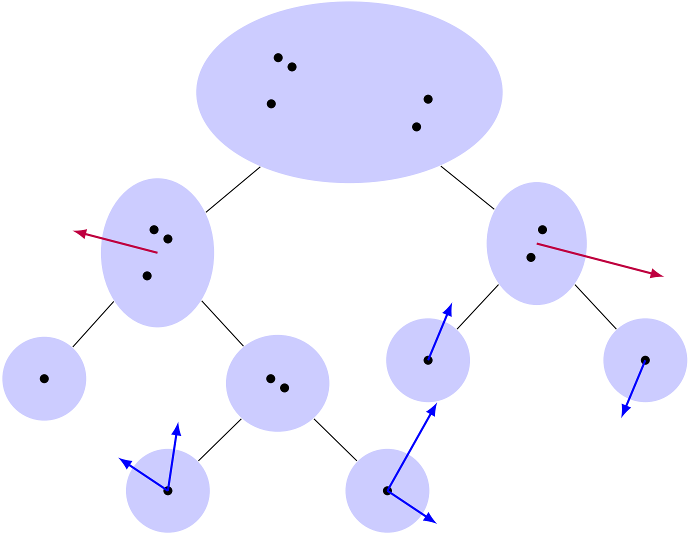
Using the Split Tree



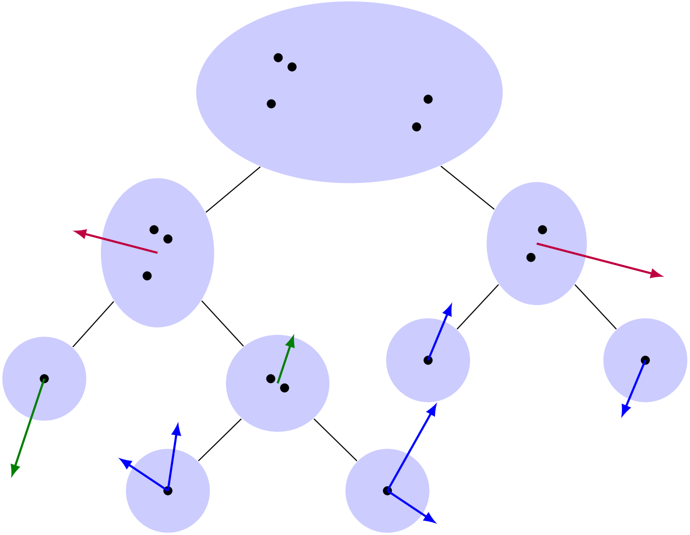
Using the Split Tree



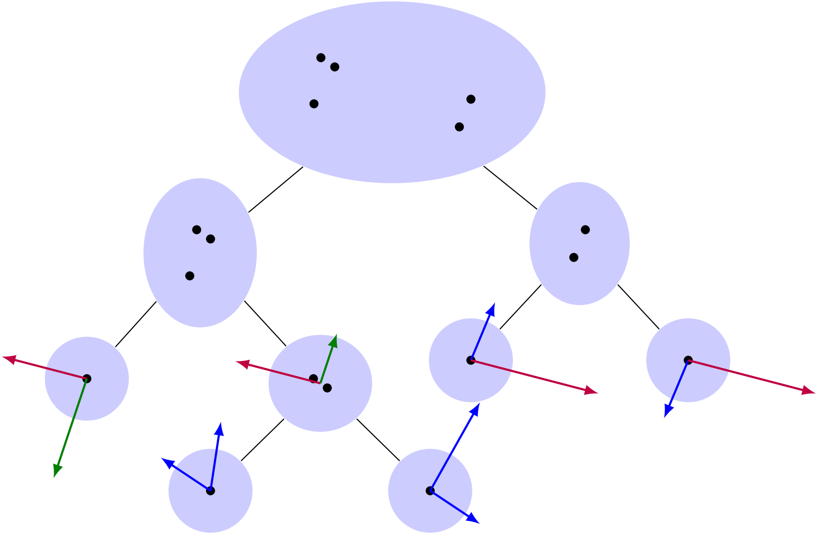
Using the Split Tree



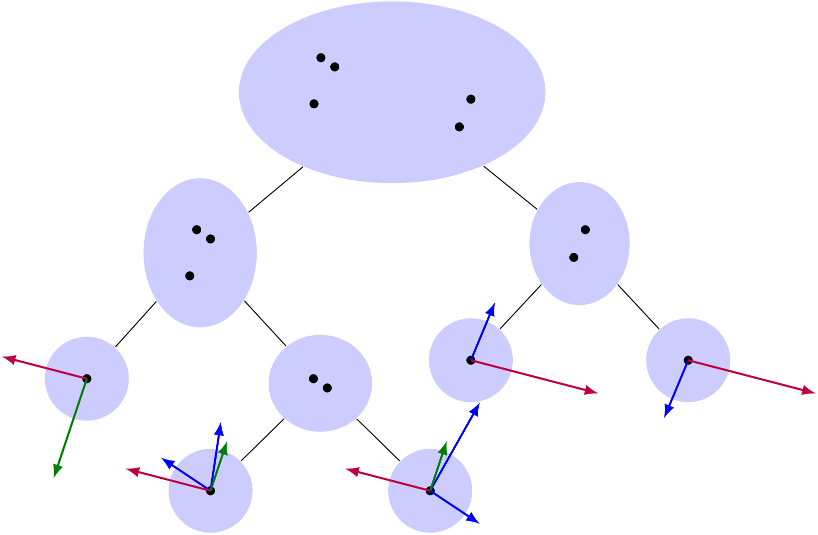
Using the Split Tree



Using the Split Tree



Using the Split Tree



Running time

Compute the WSPD using a split tree	$O(n \log n)$
Compute forces for every pair	$O(n)$
Propagate forces in the split tree	$O(n)$
Total	$O(n \log n)$

Experiments

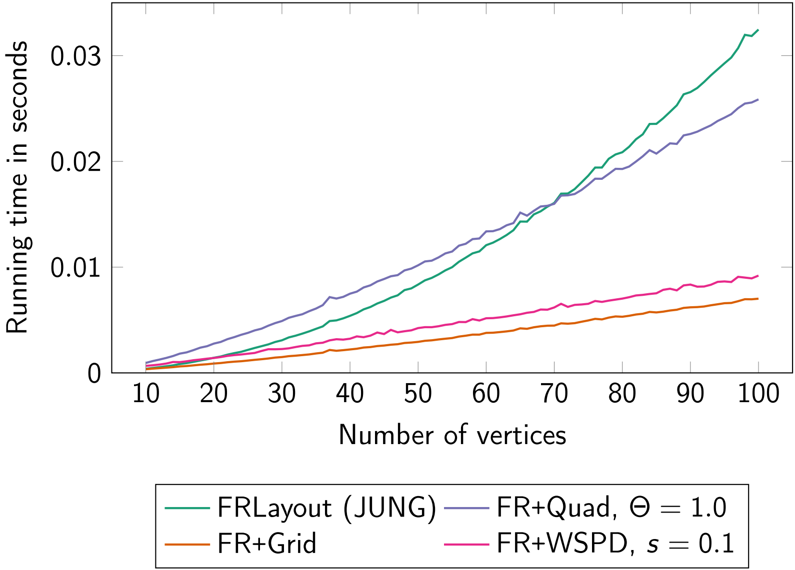
Implementation

- in Java
- based on FRLayout in Jung (Java Universal Network/Graph Framework)

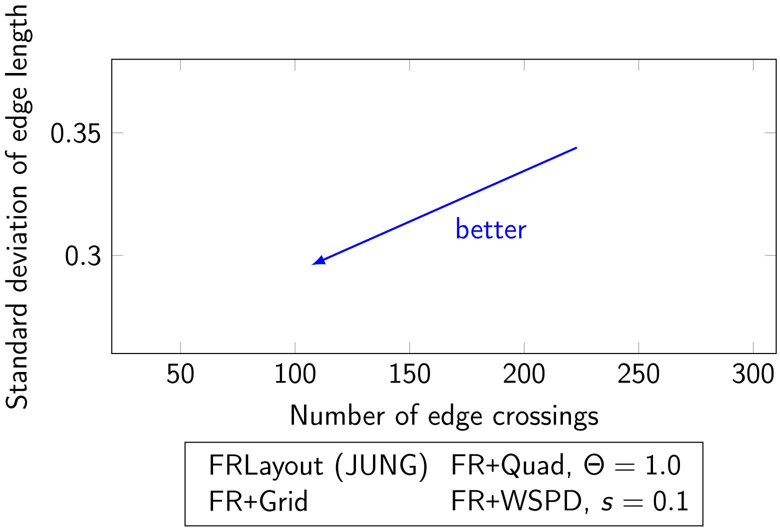
Comparison to similar speed-up techniques:

- Quadtree based [Barnes & Hut, 1986]
- Grid variant [Fruchterman & Reingold, 1991]

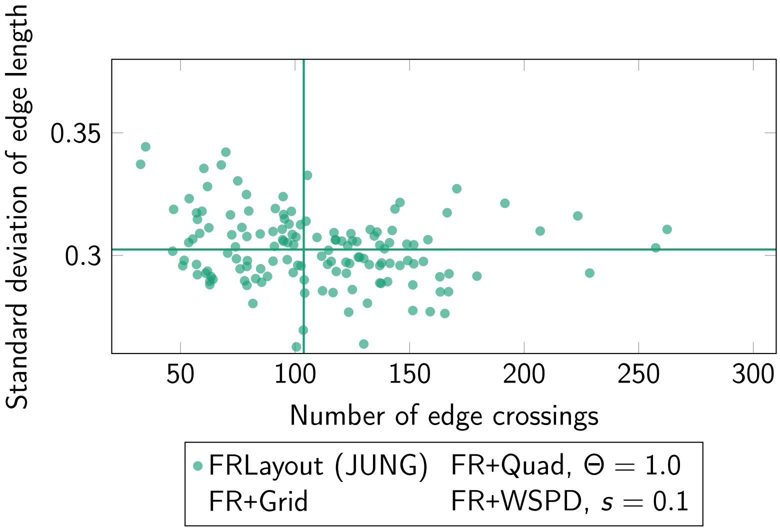
Experiments: Running time



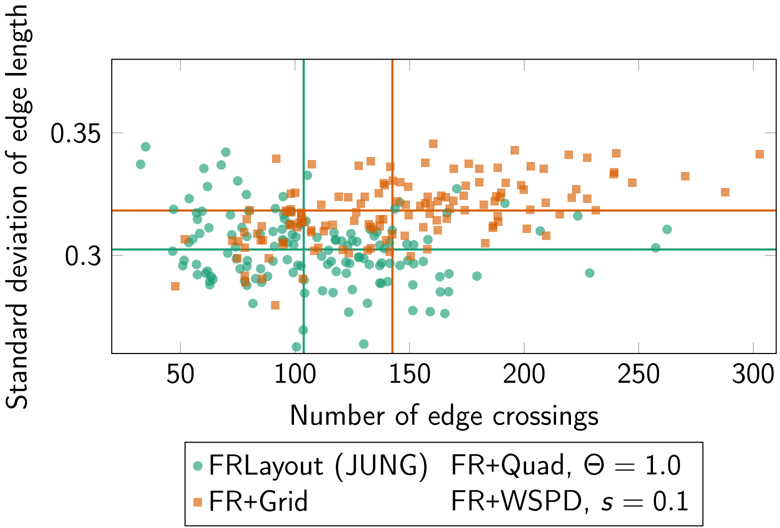
Experiments: Quality



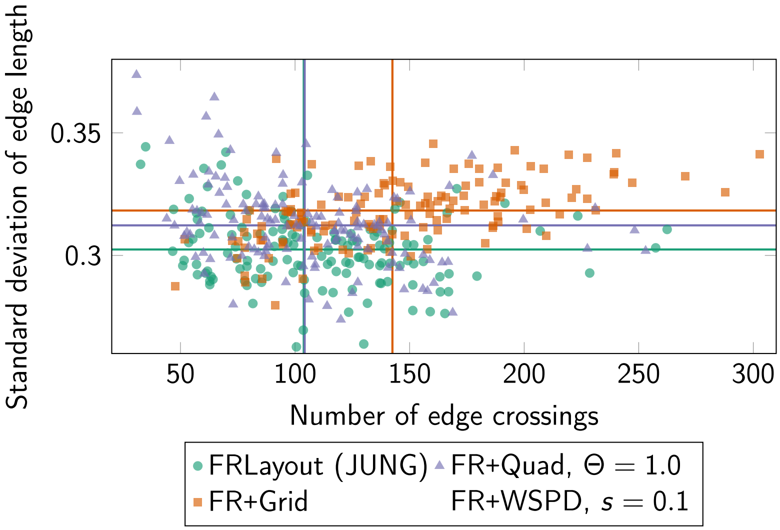
Experiments: Quality



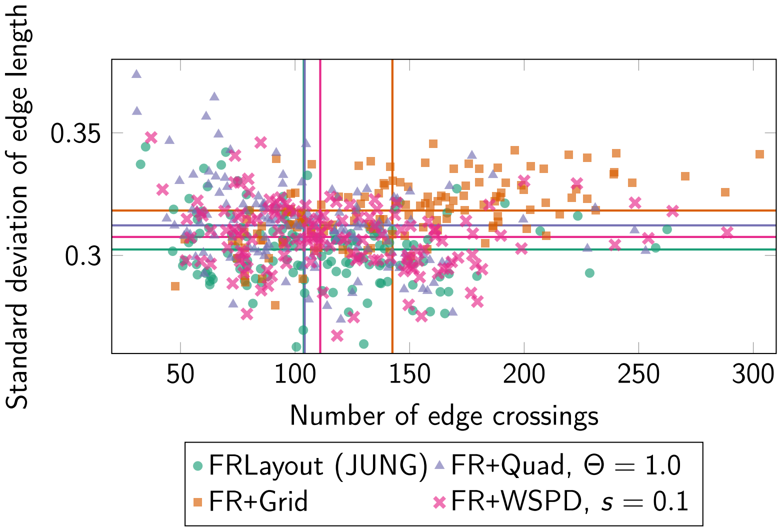
Experiments: Quality



Experiments: Quality



Experiments: Quality



Summary

- Asymptotic running time of $O(n \log n)$ for each iteration
- Notably faster than unmodified algorithm
- Similar to unmodfied algorithm in terms of quality
- Better quality than Grid variant of Fruchterman & Reingold

Summary

- Asymptotic running time of $O(n \log n)$ for each iteration
- Notably faster than unmodified algorithm
- Similar to unmodfied algorithm in terms of quality
- Better quality than Grid variant of Fruchterman & Reingold

Demonstration during the poster session this afternoon

Summary

- Asymptotic running time of $O(n \log n)$ for each iteration
- Notably faster than unmodified algorithm
- Similar to unmodfied algorithm in terms of quality
- Better quality than Grid variant of Fruchterman & Reingold

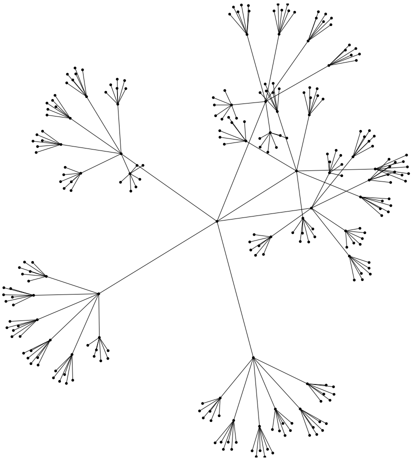
Demonstration during the poster session this afternoon

Thank you!

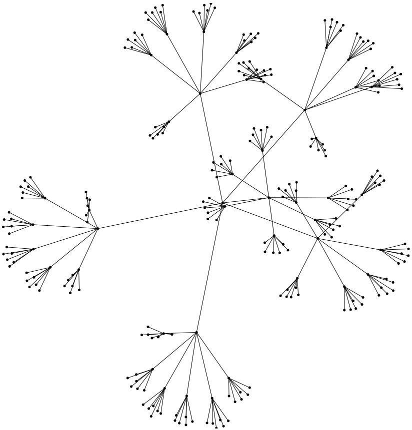
APPENDIX

Example drawings

FRLayout (JUNG)

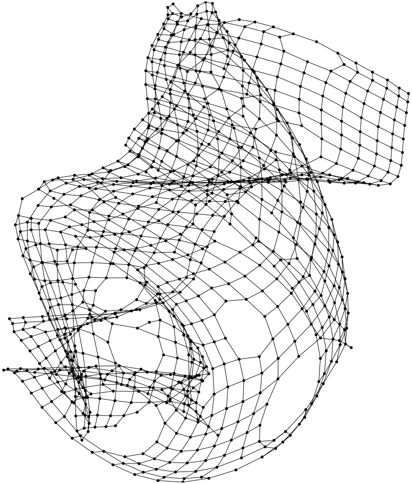


FR+WSPD, $s = 0.1$

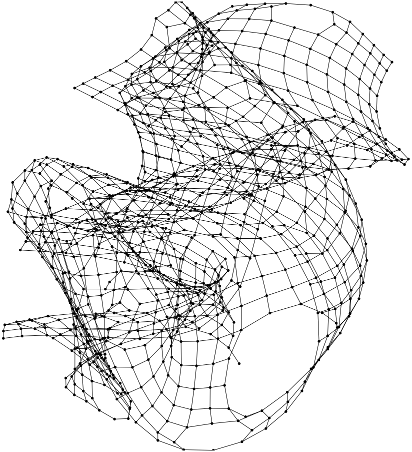


Example drawings

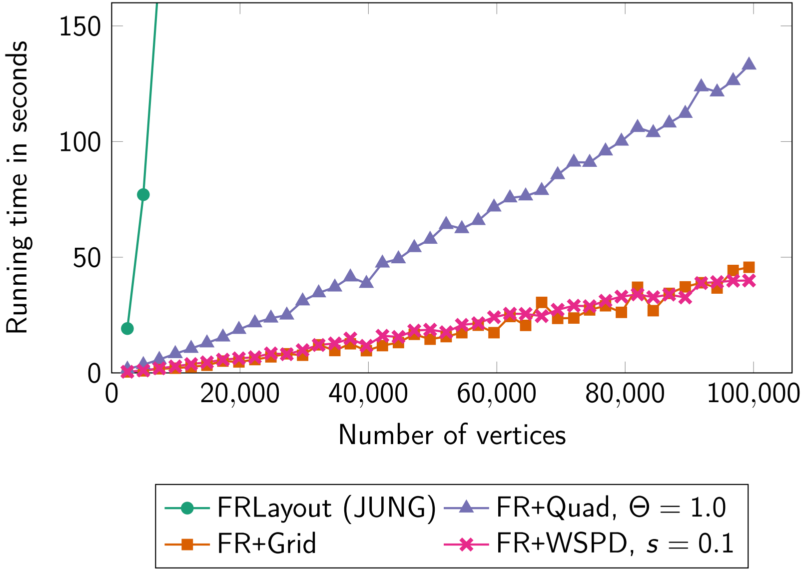
FRLayout (JUNG)



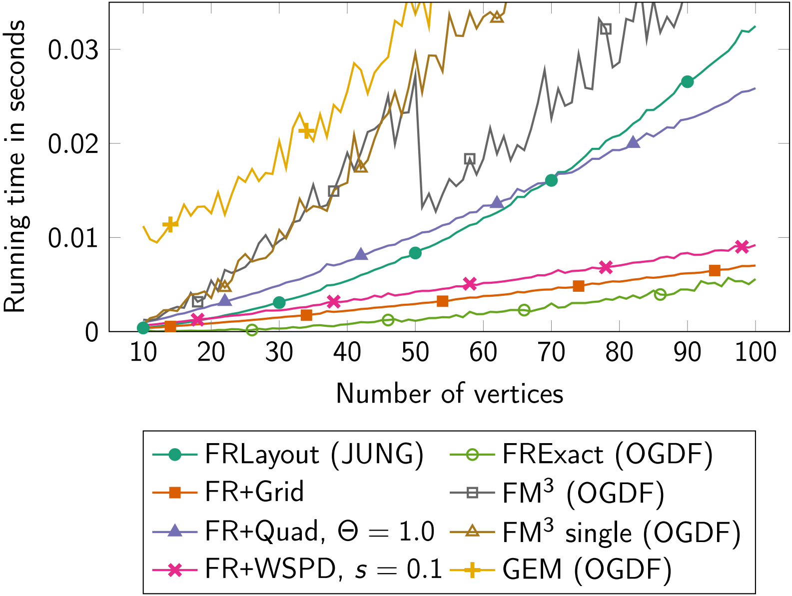
FR+WSPD, $s = 0.1$



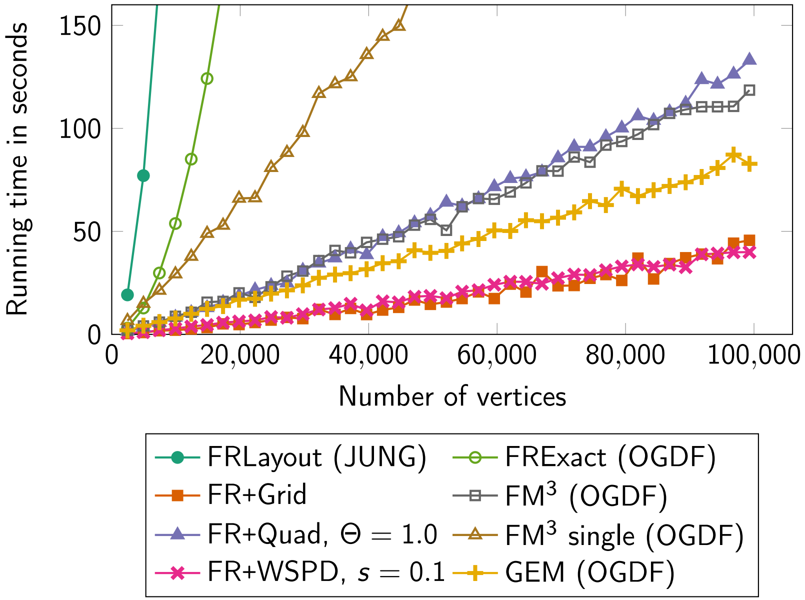
Experiments: Running time on large graphs



Experiments: Running times compared to OGDF



Experiments: Running times compared to OGDF



Experiments: Quality compared to OGDF