Learning a 2-Manifold with a Boundary in $\mathbb{R}^3$

Christian Scheffer & Jan Vahrenhold

Technische Universität Dortmund

July 11, 2011

Reconstructing surfaces with a boundary is not trivial but efficiently solvable

- How to reconstruct an unknown surface (without a boundary) from a point set?
- Why is handling a boundary not trivial?
- How can the problems arising from the presence of a boundary can be solved efficiently?

Reconstructing surfaces with a boundary is not trivial but efficiently solvable

- How to reconstruct an unknown surface (without a boundary) from a point set?
- Why is handling a boundary not trivial?
- How can the problems arising from the presence of a boundary can be solved efficiently?

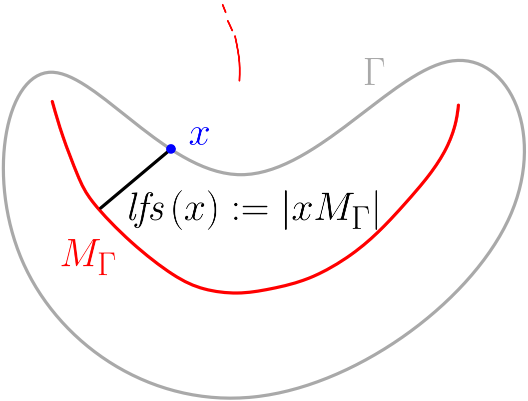
Reconstructing surfaces with a boundary is *not trivial* but efficiently solvable

- How to reconstruct an unknown surface (without a boundary) from a point set?
- Why is handling a boundary not trivial?
- How can the problems arising from the presence of a boundary can be solved efficiently?

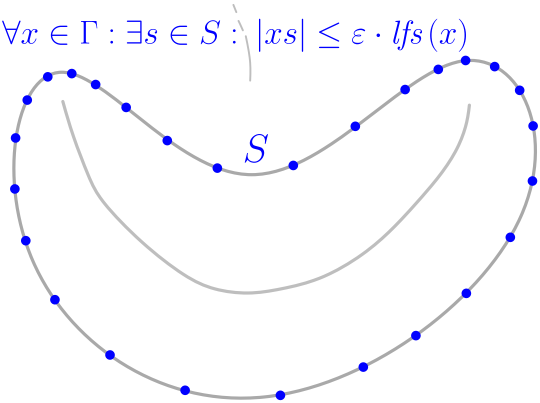
Reconstructing surfaces with a boundary is not trivial but *efficiently solvable*

- How to reconstruct an unknown surface (without a boundary) from a point set?
- Why is handling a boundary not trivial?
- How can the problems arising from the presence of a boundary can be solved efficiently?

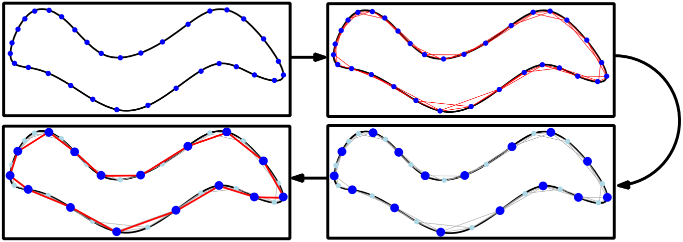
Local feature size

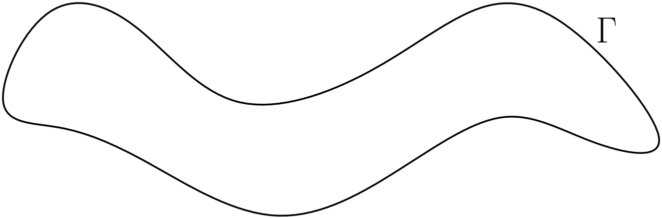


ε-sample

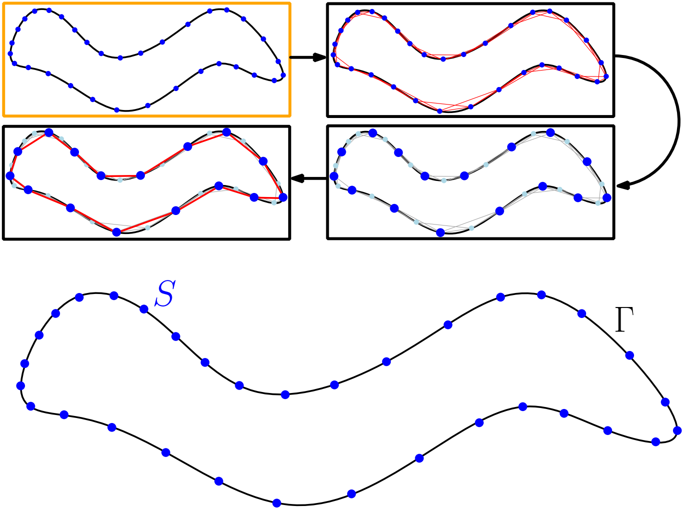


Comb. Reconstructionalgorithm from Dumitriu *et al.*

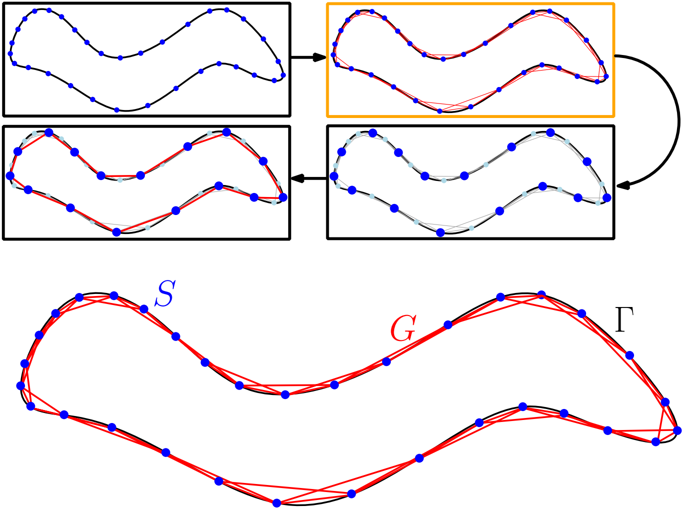




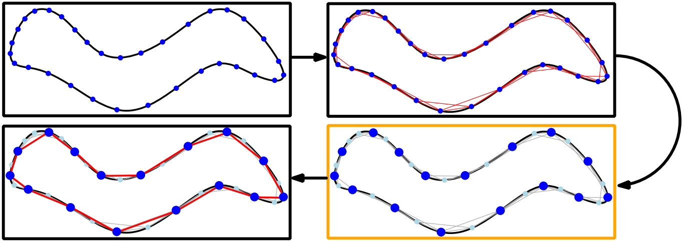
Comb. Reconstructionalgorithm from Dumitriu *et al.*

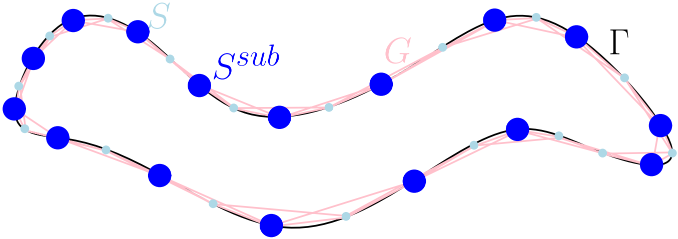


Comb. Reconstructionalgorithm from Dumitriu *et al.*

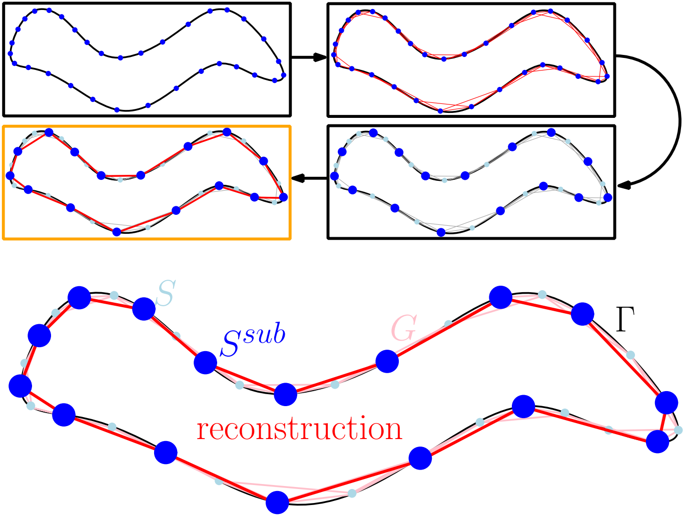


Comb. Reconstructionalgorithm from Dumitriu *et al.*

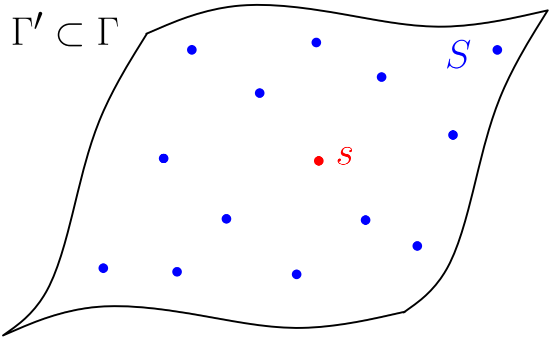




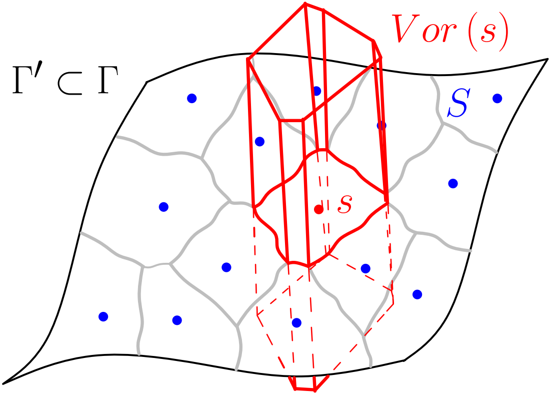
Comb. Reconstructionalgorithm from Dumitriu *et al.*



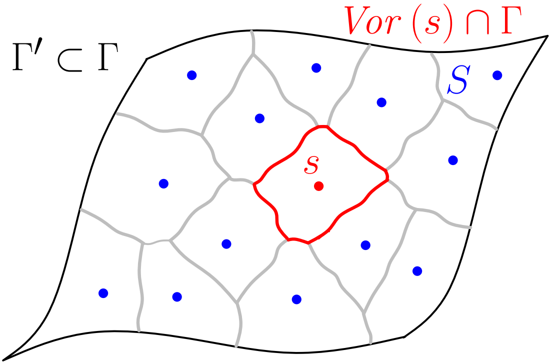
Computing the neighbourhoodgraph G - w/o boundary



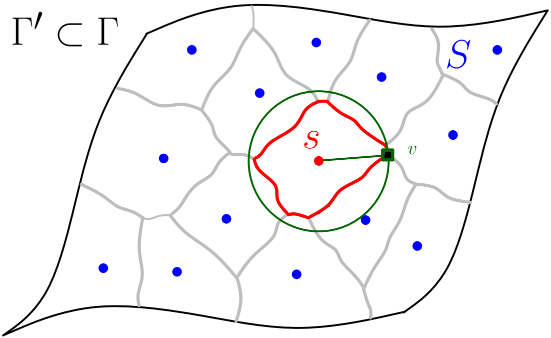
Computing the neighbourhoodgraph G - w/o boundary



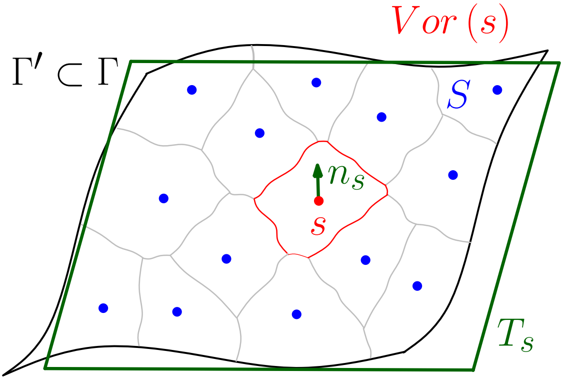
Computing the neighbourhoodgraph G - w/o boundary



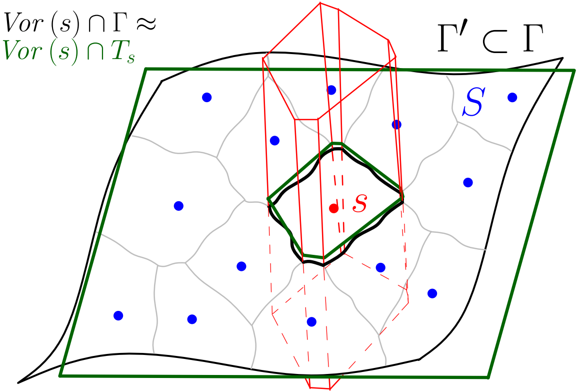
Computing the neighbourhoodgraph G - w/o boundary



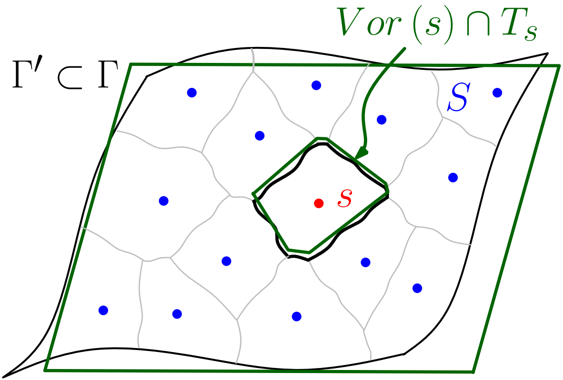
Computing the neighbourhoodgraph G - w/o boundary



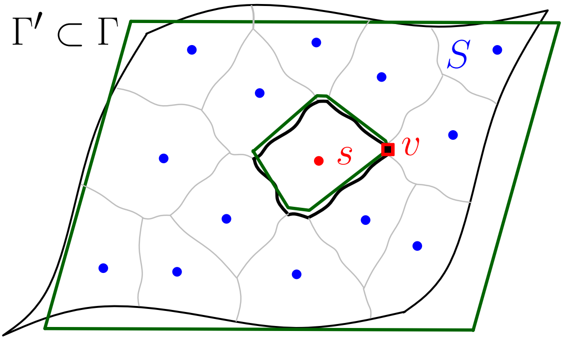
Computing the neighbourhoodgraph G - w/o boundary



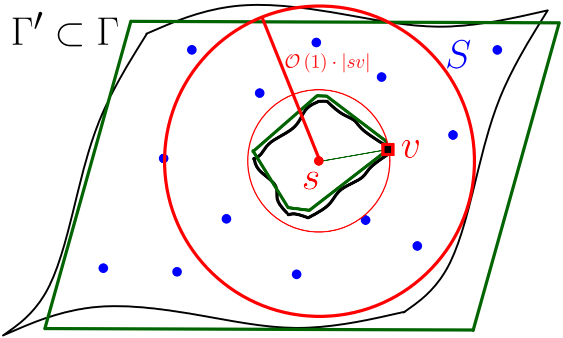
Computing the neighbourhoodgraph G - w/o boundary



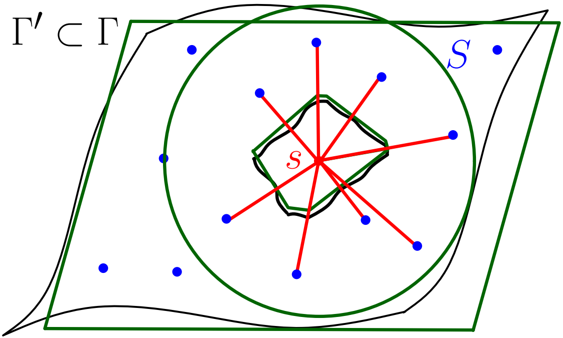
Computing the neighbourhoodgraph G - w/o boundary



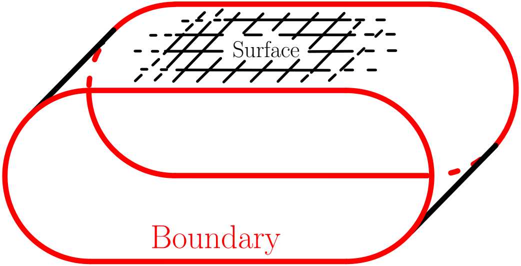
Computing the neighbourhoodgraph G - w/o boundary



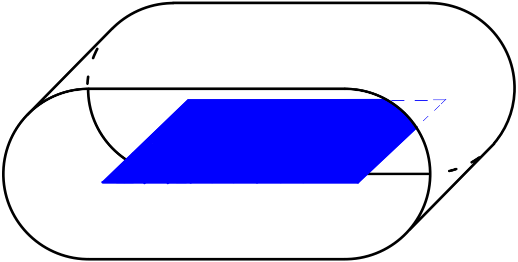
Computing the neighbourhoodgraph G - w/o boundary



Extending the definition of the local feature size

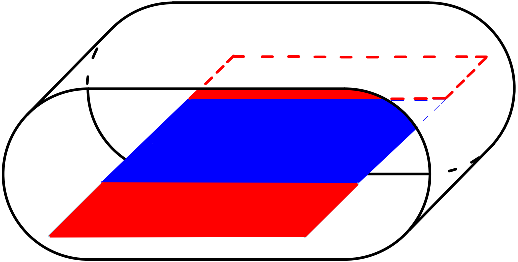


Extending the definition of the local feature size



$$lfs\,(x) = dist\,(x, M_\Gamma)$$

Extending the definition of the local feature size

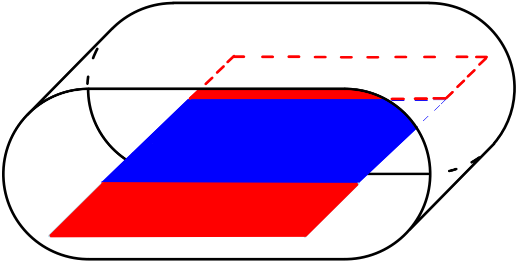


$$lfs(x) = \min \{dist(x, M_\Gamma), lfs_{\partial\Gamma}(x)\},$$

with

$$lfs_{\partial\Gamma} = \begin{cases} \min_{y \in \partial\Gamma} \{dist(y, M_{\partial\Gamma}) + |xy|\} & x \in \Gamma^\circ \\ dist(x, M_{\partial\Gamma}) & x \in \partial\Gamma \end{cases}$$

Extending the definition of the local feature size



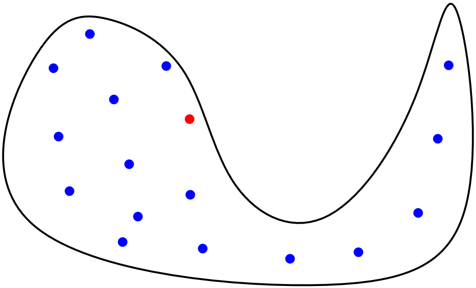
$$lfs(x) = \min\{dist(x, M_\Gamma), lfs_{\partial\Gamma}(x)\},$$

with

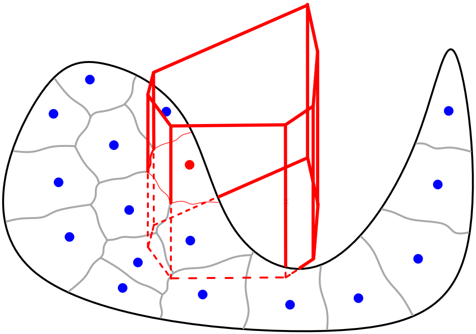
$$lfs_{\partial\Gamma} = \begin{cases} \min_{y\in\partial\Gamma}\{dist(y, M_{\partial\Gamma}) + |xy|\} & x \in \Gamma^\circ \\ dist(x, M_{\partial\Gamma}) & x \in \partial\Gamma \end{cases}$$

"good" properties are maintained

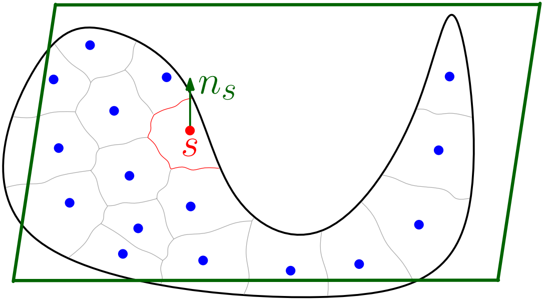
Computing the neighbourhoodgraph G - w/ boundary



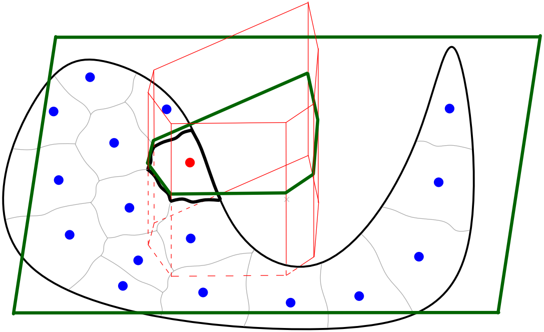
Computing the neighbourhoodgraph G - w/ boundary



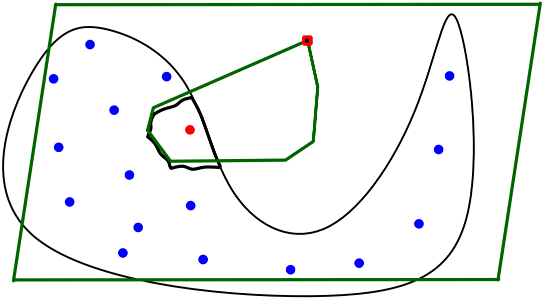
Computing the neighbourhoodgraph G - w/ boundary



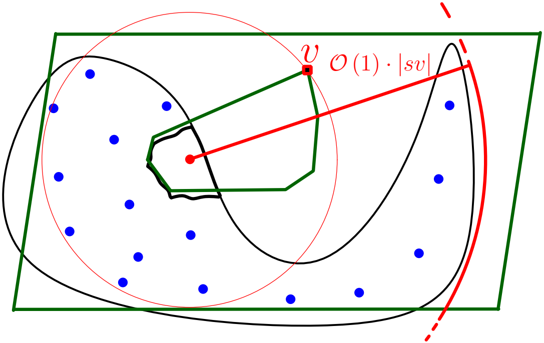
Computing the neighbourhoodgraph G - w/ boundary



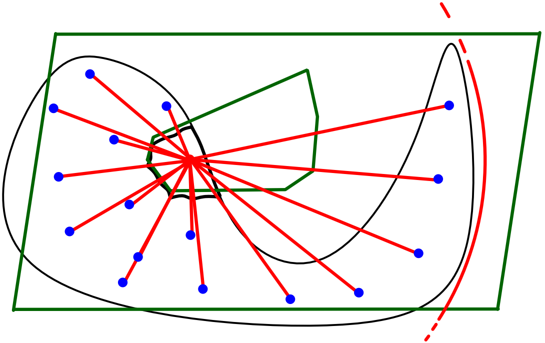
Computing the neighbourhoodgraph G - w/ boundary



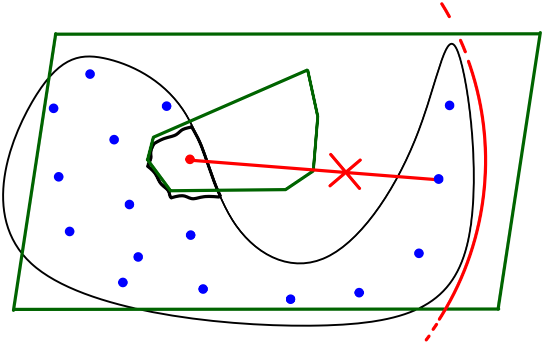
Computing the neighbourhoodgraph G - w/ boundary



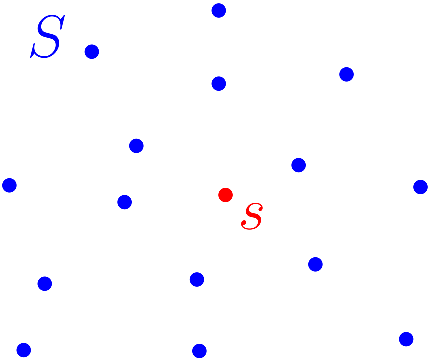
Computing the neighbourhoodgraph G - w/ boundary



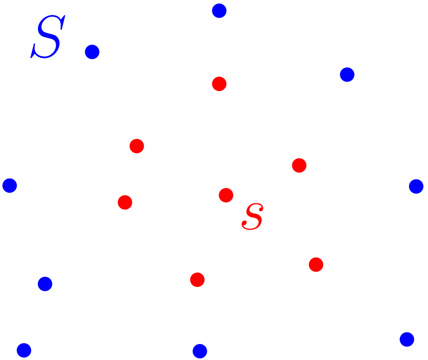
Computing the neighbourhoodgraph G - w/ boundary



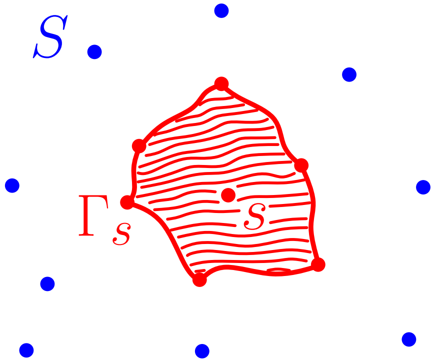
Approach - high level view



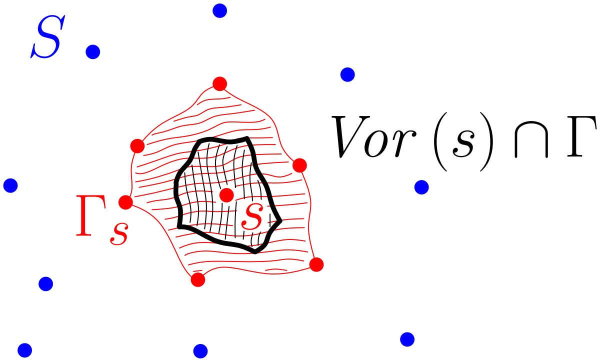
Approach - high level view



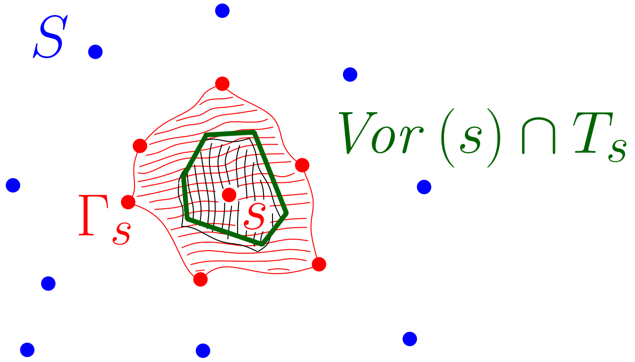
Approach - high level view



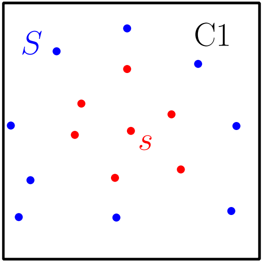
Approach - high level view

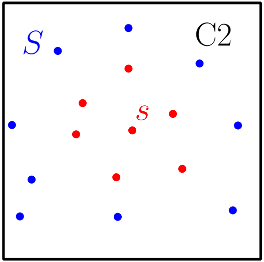


Approach - high level view

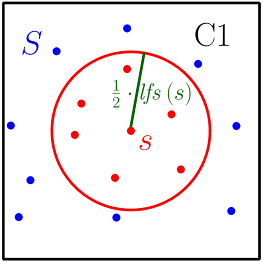


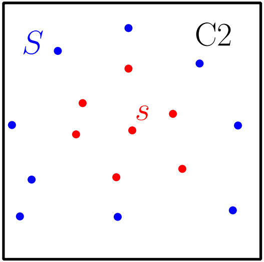
Approach - low level view



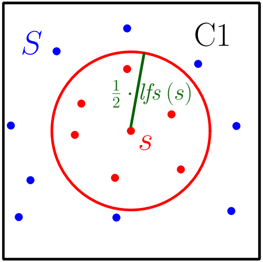


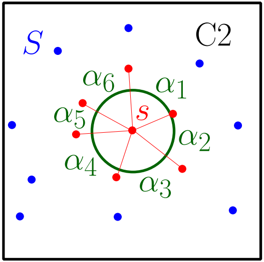
Approach - low level view



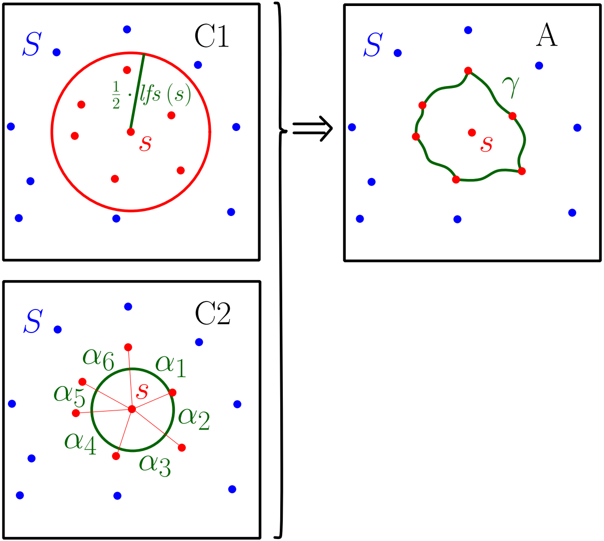


Approach - low level view

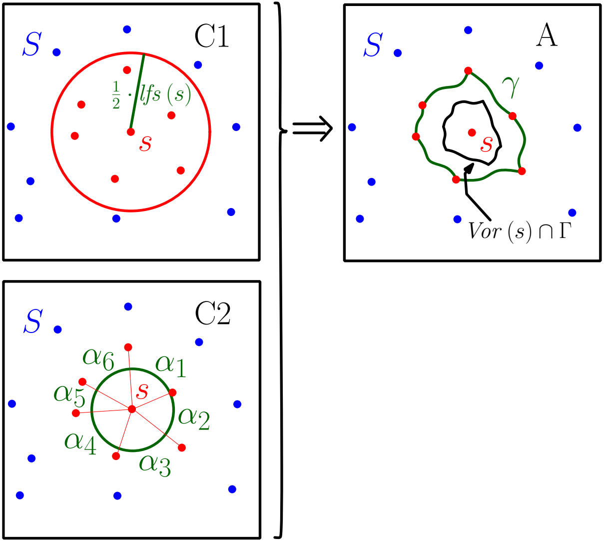




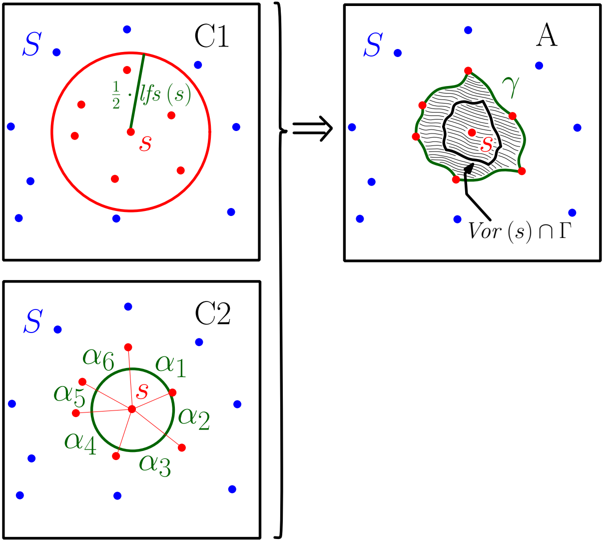
Approach - low level view



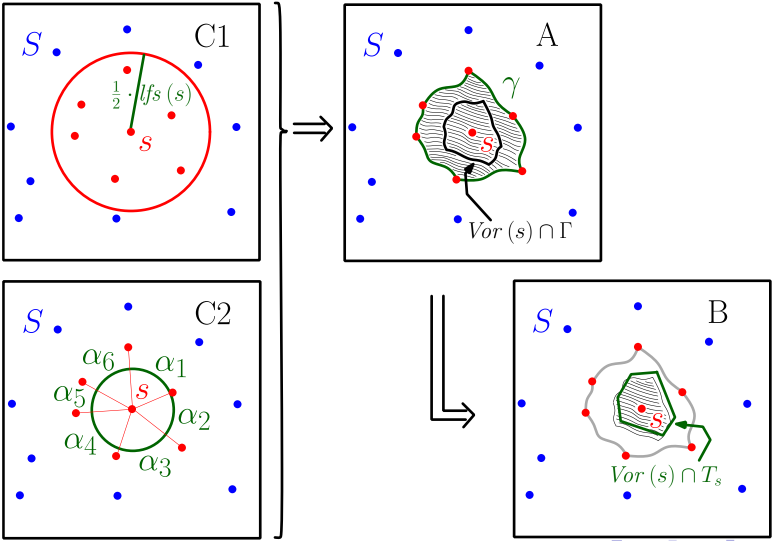
Approach - low level view



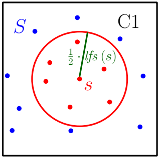
Approach - low level view

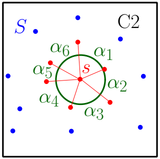


Approach - low level view

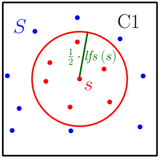


Recognition of C1 and C2



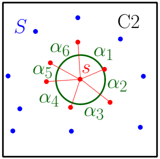


Recognition of C1 and C2

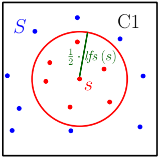


Thereom

Can compute $k_\varepsilon \in \Omega\left(\frac{1}{\varepsilon^2}\right)$, s.t. k_ε nearest neighbours of $s \in S$ are inside $B\left(x, \frac{1}{2} \cdot lfs(s)\right)$



Recognition of C1 and C2



Thereom

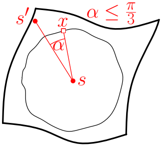
Can compute $k_\varepsilon \in \Omega\left(\frac{1}{\varepsilon^2}\right)$, s.t. k_ε nearest neighbours of $s \in S$ are inside $B\left(x, \frac{1}{2} \cdot lfs(s)\right)$

Require new sampling condition for $k \leq k_\varepsilon$
$\longrightarrow (\varepsilon, k)$-sample

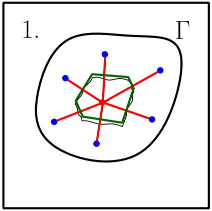
Definition

(ε, k)-sample is an ε-sample S with $\forall s \in S$:

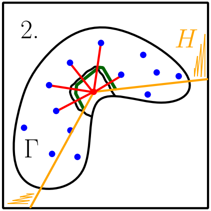
$$\forall x \in \Gamma : |xs| = \frac{\varepsilon}{1-\varepsilon} \cdot lfs(s) :$$
$$\exists s' \in \{NN_1, ..., NN_k\} : \angle(xss') \leq \frac{\pi}{3}$$



Implications of working with an (ε, k)-sample

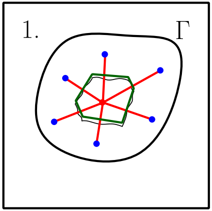


$|s\partial\Gamma| \geq 2 \cdot \varepsilon \cdot lfs\,(s)$:

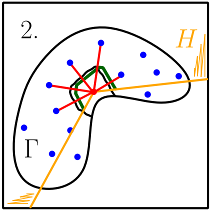


$|s\partial\Gamma| < 2 \cdot \varepsilon \cdot lfs\,(s)$:

Implications of working with an (ε, k)-sample

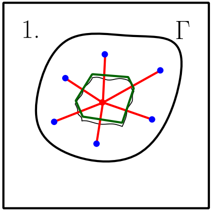


$|s\partial\Gamma| \geq 2 \cdot \varepsilon \cdot lfs\,(s)$:
Can prove: edges are superset of Delaunay-edges restricted to Γ

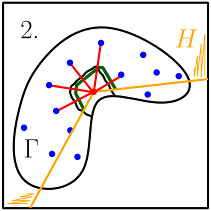


$|s\partial\Gamma| < 2 \cdot \varepsilon \cdot lfs\,(s)$:

Implications of working with an (ε, k)-sample



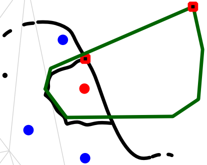
$|s\partial\Gamma| \geq 2 \cdot \varepsilon \cdot lfs(s)$:
Can prove: edges are superset of Delaunay-edges restricted to Γ

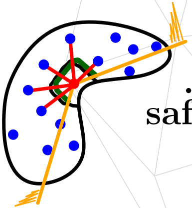


$|s\partial\Gamma| < 2 \cdot \varepsilon \cdot lfs(s)$:
Can compute "wedge" H, s.t.
$\partial\Gamma$ avoids $Vor(s) \cap \Gamma \cap H$
$\Longrightarrow Vor(s) \cap T_s \cap H \approx Vor(s) \cap \Gamma \cap H$

SURFACE RECONSTRUCTION

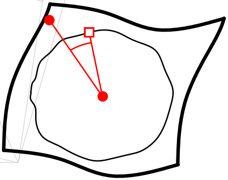
Boundary leads to problems...





... which can be solved by safegurading from the boundary...

... guaranteed by assuming an (ε, k)-sample.



In the full paper:

- **Analysis of the runtime:** $\mathcal{O}(n \cdot \log(n))$
- **Analysis of the approximation quality:** $|e| \leq 1.135 \cdot c \cdot \varepsilon \cdot \boldsymbol{lfs}$