Competitive Self-Stabilizing $k$-Clustering

Yvan Rivierre

Joint work with
Ajoy K. Datta, Stéphane Devismes,
Karel Heurtefeux, and Lawrence L. Larmore
Competitive Self-Stabilizing $k$-Clustering

Yvan Rivierre

Joint work with
Ajoy K. Datta, Stéphane Devismes,
Karel Heurtefeux, and Lawrence L. Larmore
Introduction to $k$-Clustering

Hierarchical organization of processes
Introduction to $k$-Clustering

Hierarchical organization of processes
Introduction to \( k \)-Clustering

Hierarchical organization of processes
Introduction to $k$-Clustering

Hierarchical organization of processes
Self-Stabilization

transient faults

transient faults

System States

Time
Competitiveness

Finding a \textit{minimum} $k$-clustering set is \textit{NP}-hard.

An algorithm is \textit{X-competitive} if it builds a $k$-clustering of size at most $X$ times the smallest possible number of $k$-clusters.
Unit Disk Graphs
Quasi Unit Disk Graphs

\[ \lambda \geq 1 \]
Contribution

A $k$-clustering algorithm that is

- self-stabilizing,
- for identified networks,
- in the shared memory model,
- building at most $O(n/k)$ $k$-clusters,
- $7.2552k+O(1)$-competitive in UDGs,
- and $7.2552\lambda^2 k+O(\lambda)$-competitive in QUDGs.
Related Work

- Self-stabilizing $k$-Clustering algorithms
  \cite{(Datta & al., 2009), (Caron & al., 2010)}

- $8k+O(1)$-competitive $k$-Clustering algorithm
  \cite{(Fernandess & Malkhi, 2002)}
Algorithm: General Overview

Combination of self-stabilizing algorithms:
- Spanning Tree Construction (Huang & Chen, 1992)
- MIS Tree Build (imp. Fernandess & Malkhi, 2002) \(\rightarrow\) for the sole purpose of competitiveness
- \(k\)-Clusterheads Selection
- \(k\)-Clustering Construction

Parallel execution of both algorithms
$k$-Clusterheads Selection: $\alpha$
$k$-Clusterheads Selection: $\alpha$
$k$-Clusterheads Selection: $\alpha$

2k hops
$k$-Clusterheads Selection: $\alpha$
$k$-Clusterheads Selection: $\alpha$
$k$-Clusterheads Selection: $\alpha$

2k hops

$k$
$k$-Clusterheads Selection: $\alpha$

$\alpha$ hops
Inductive computation of $\alpha$
Inductive computation of $\alpha$

\[
\begin{align*}
\alpha &= A \\
\alpha &= B \\
\alpha &= 0 \\
\alpha &= k \\
\alpha &= 0
\end{align*}
\]

\[(0 \leq A < k \leq B \leq 2k)\]

\[A = \text{MaxChild} \{ \alpha < k \}\]

\[B = \text{MinChild} \{ \alpha \geq k \}\]
Inductive computation of $\alpha$

$A = \text{MaxChild} \{ \alpha < k \}$

$B = \text{MinChild} \{ \alpha \geq k \}$

$(0 \leq A < k \leq B \leq 2k)$
Inductive computation of $\alpha$

$\alpha = A$

$\alpha = B + 1$

$\alpha = B$

$\alpha = 0$

$\alpha = k$

$A = \text{MaxChild}\{\alpha < k\}$

$B = \text{MinChild}\{\alpha \geq k\}$

$A + B + 2 \leq 2k$

$(0 \leq A < k \leq B \leq 2k)$
Inductive computation of $\alpha$

$A = \text{MaxChild} \{ \alpha < k \}$

$B = \text{MinChild} \{ \alpha \geq k \}$

$
\begin{align*}
A &= \text{MaxChild} \{ \alpha < k \} \\
B &= \text{MinChild} \{ \alpha \geq k \}
\end{align*}
$

$(0 \leq A < k \leq B \leq 2k)$

$A + B + 2 > 2k$
7.2552k+0(1)-competitive in UDGs
7.2552k + O(1)-competitive in UDGs
7.2552k+0(1)-competitive in UDGs

$k + 1$ processes
7.2552\(k+0(1)\)-competitive in UDGs
7.2552k + O(1)-competitive in UDGs

(MIS = Maximal Independent Set)
7.2552k+0(1)-competitive in UDGs

(MIS = Maximal Independent Set)

[k/2] proc. ∈ MIS
7.2552k+0(1)-competitive in UDGs

(MIS = Maximal Independent Set)

(|C| - 1) \left\lceil \frac{k}{2} \right\rceil \leq |MIS| - 1
7.2552k+0(1)-competitive in UDGs

\( |\text{Clr}| - 1 \) \( \lfloor k/2 \rfloor \leq |\text{MIS}| - 1 \) \( |S| \leq (2\pi/\sqrt{3})k^2 + \pi k + 1 \) \( |\text{Opt}| \)

(MIS = Maximal Independent Set)
7.2552k + O(1)-competitive in UDGs

(MIS = Maximal Independent Set)

\[ |\text{Clr}| \leq 1 + \left(\frac{4\pi}{\sqrt{3}}k + 2\pi\right) |\text{Opt}| \approx 7.2552k + O(1) |\text{Opt}| \]
Conclusion

Self-stabilizing $k$-clustering algorithm which

- stabilizes in $O(n)$ rounds,
- uses $O(\log n)$ space per process,
- builds at most $O(n/k)$ $k$-clusters,
- is competitive in UDGs and QUDGs.
Thanks for your attention.