Fast Self-Healing Gradients

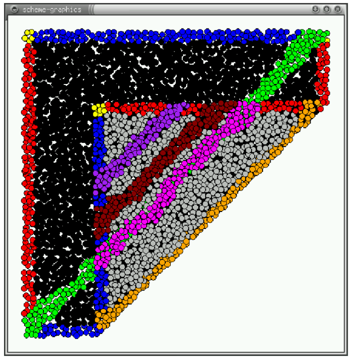
Jacob Beal, Jonathan Bachrach, Dan Vickery, Mark Tobenkin
MIT CSAIL

"Gradient": Local Calculation of Shortest-Distance Estimates

Common SA/SO building block

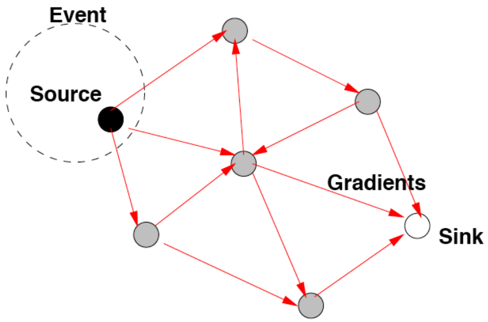
- Pattern Formation
  - Nagpal, Coore, Butera
- Distributed Robotics
  - Stoy, Werfel, McLurkin
- Networking
  - DV routing, Directed Diffusion

Need to adapt to changes



Nagpal, 2001

McLurkin, 2004

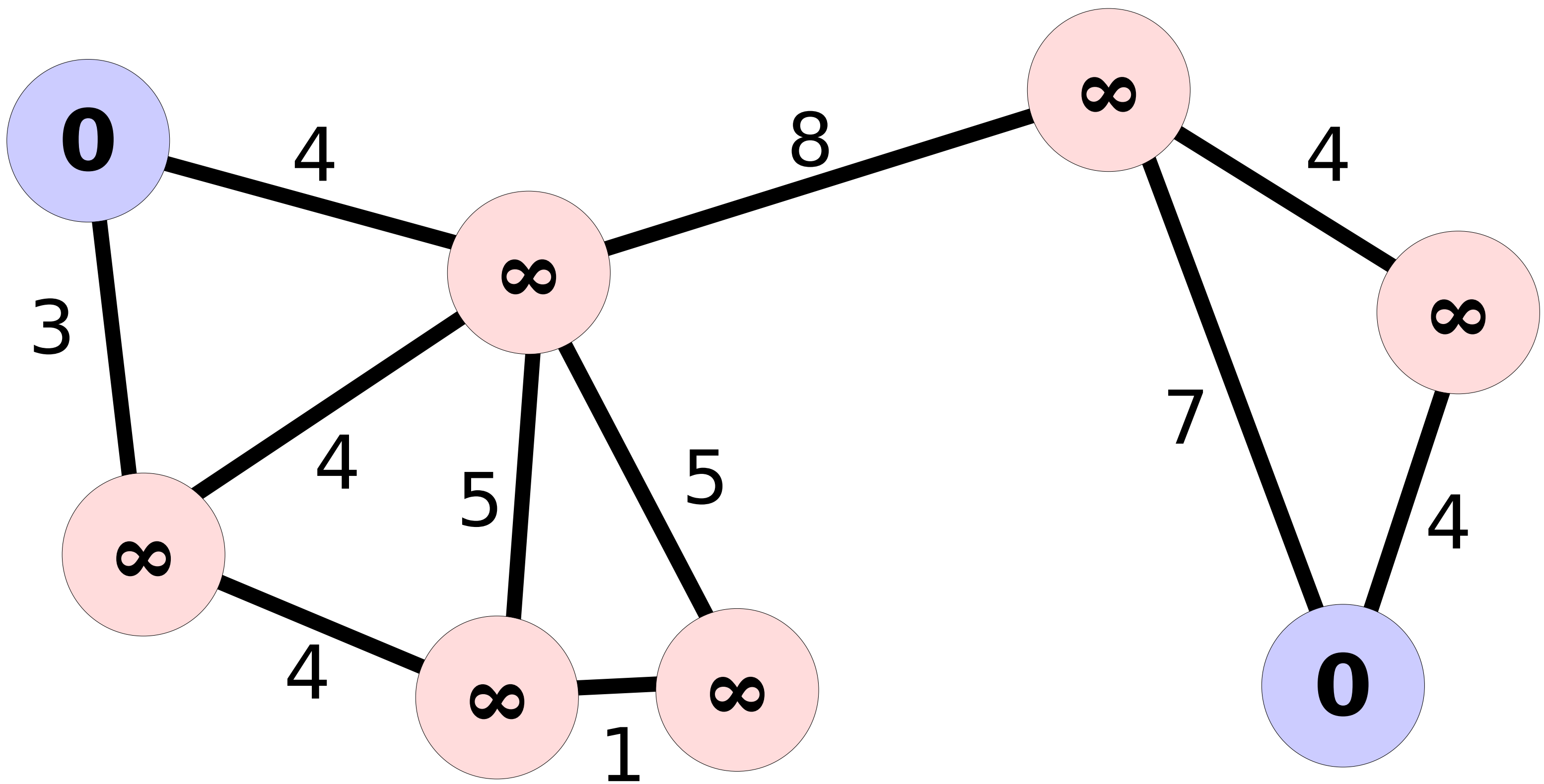


Intanagonwiwat, et al. 2002

Outline

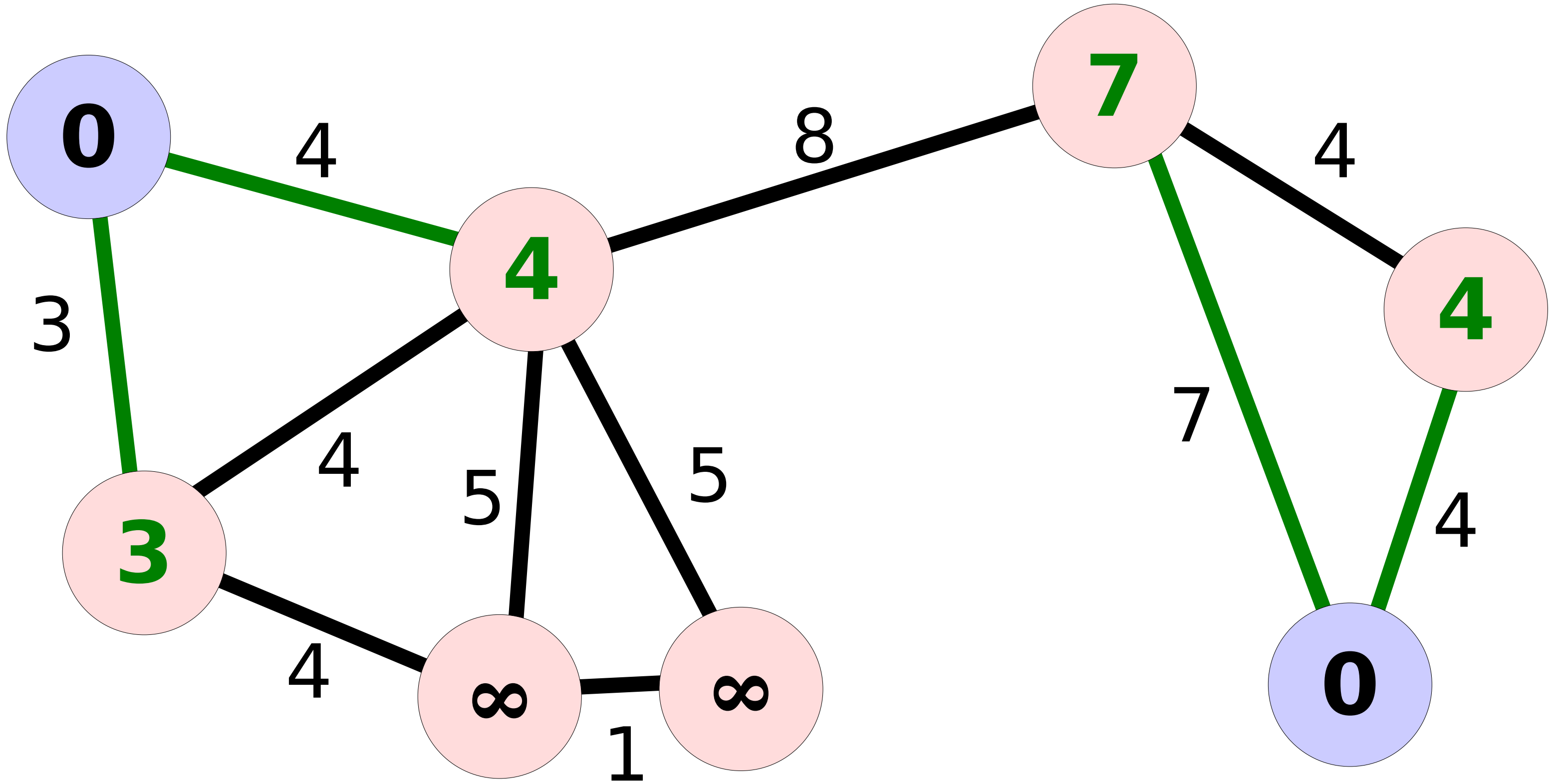
- Rising Value Problem
- CRF-Gradient: self-stabilize in *O(diameter)*
- Verified in simulation and on Mica2 Motes

Calculation By Relaxation



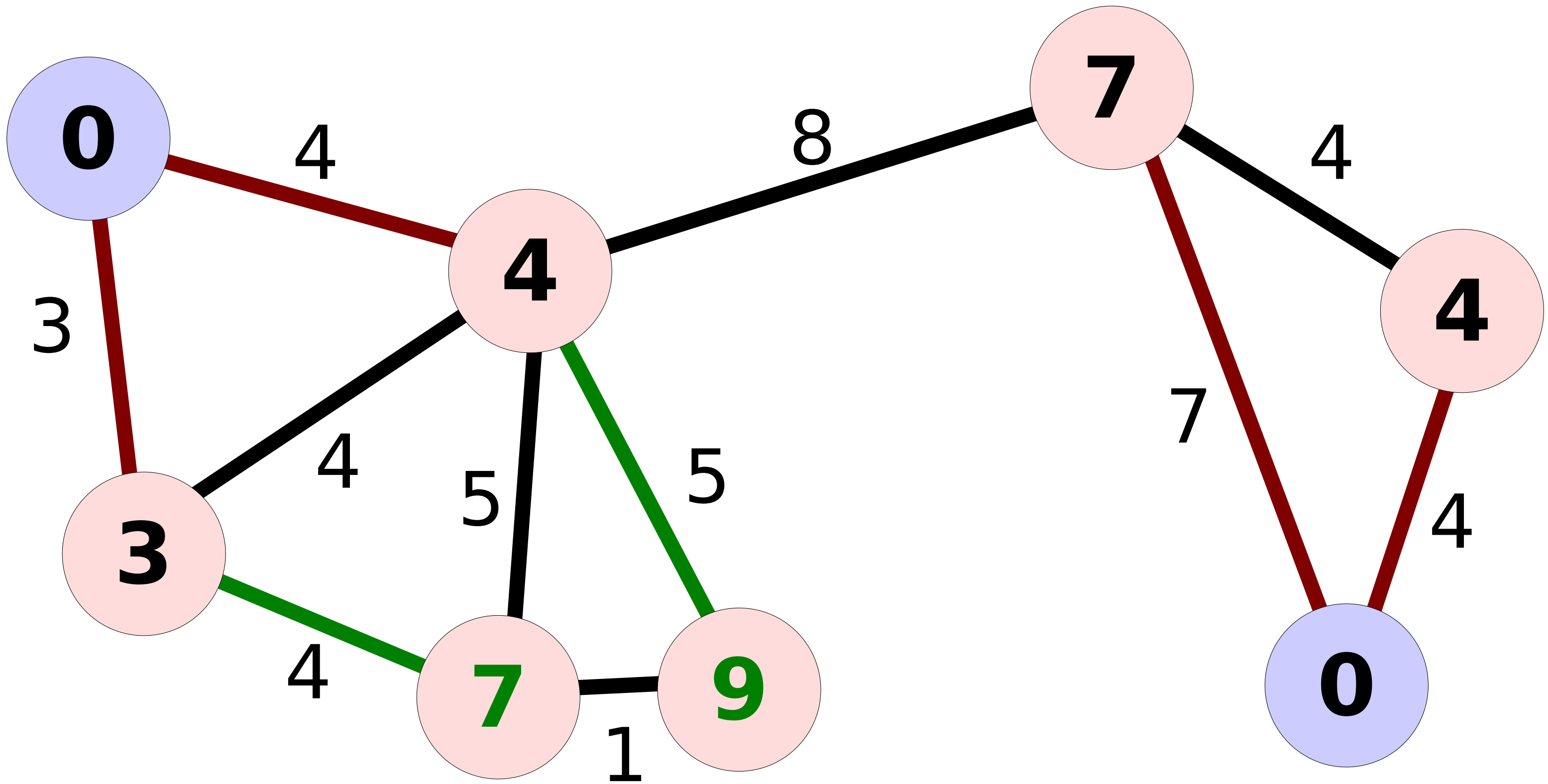
$$g_x = \begin{cases} 0 & if\ x \in S \\ min\{g_y + d(x,y) \mid y \in N_x\} & if\ x \notin S \end{cases}$$

Calculation By Relaxation



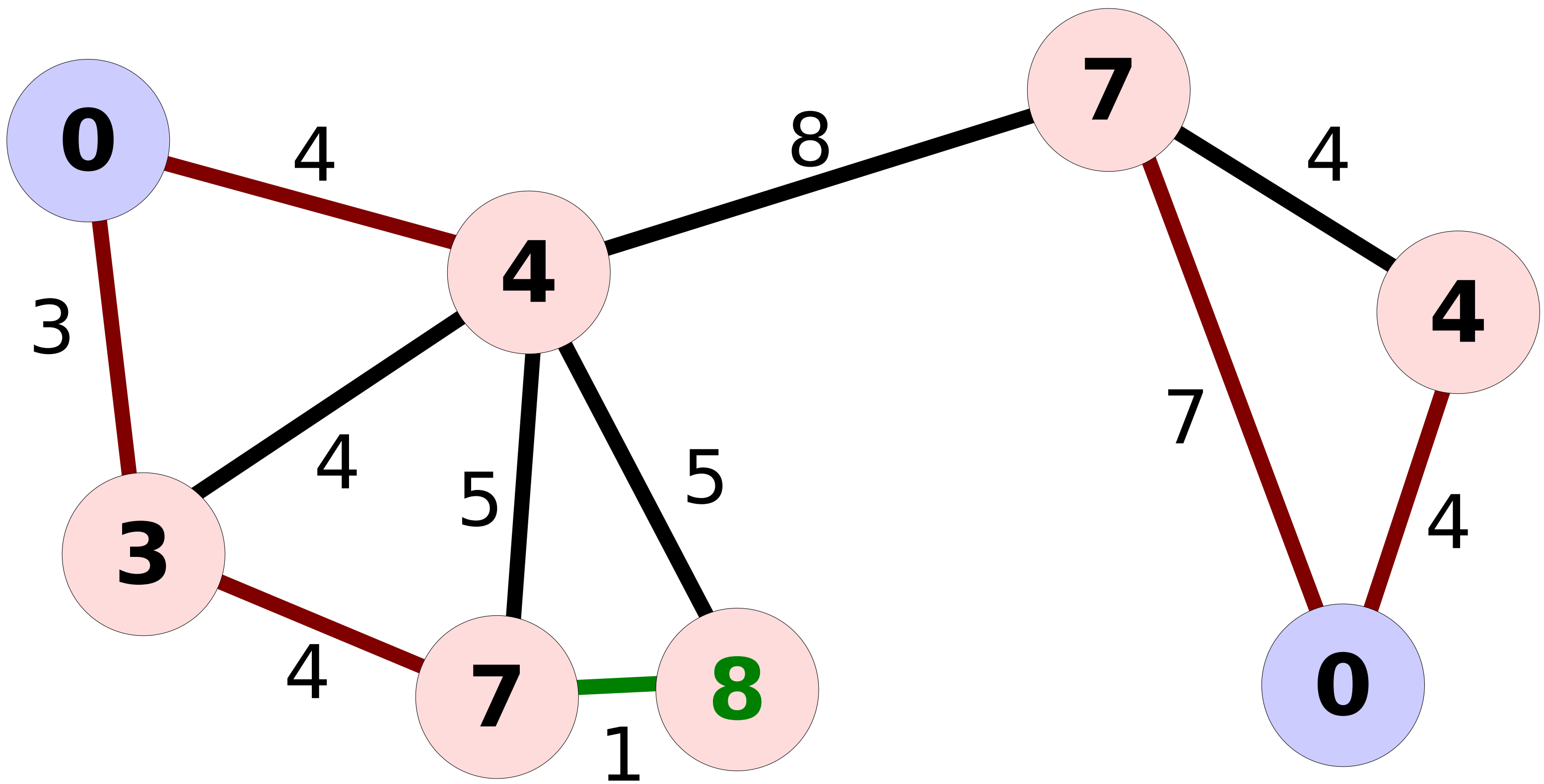
$$g_x = \begin{cases} 0 & if\ x \in S \\ min\{g_y + d(x,y) \mid y \in N_x\} & if\ x \notin S \end{cases}$$

Calculation By Relaxation



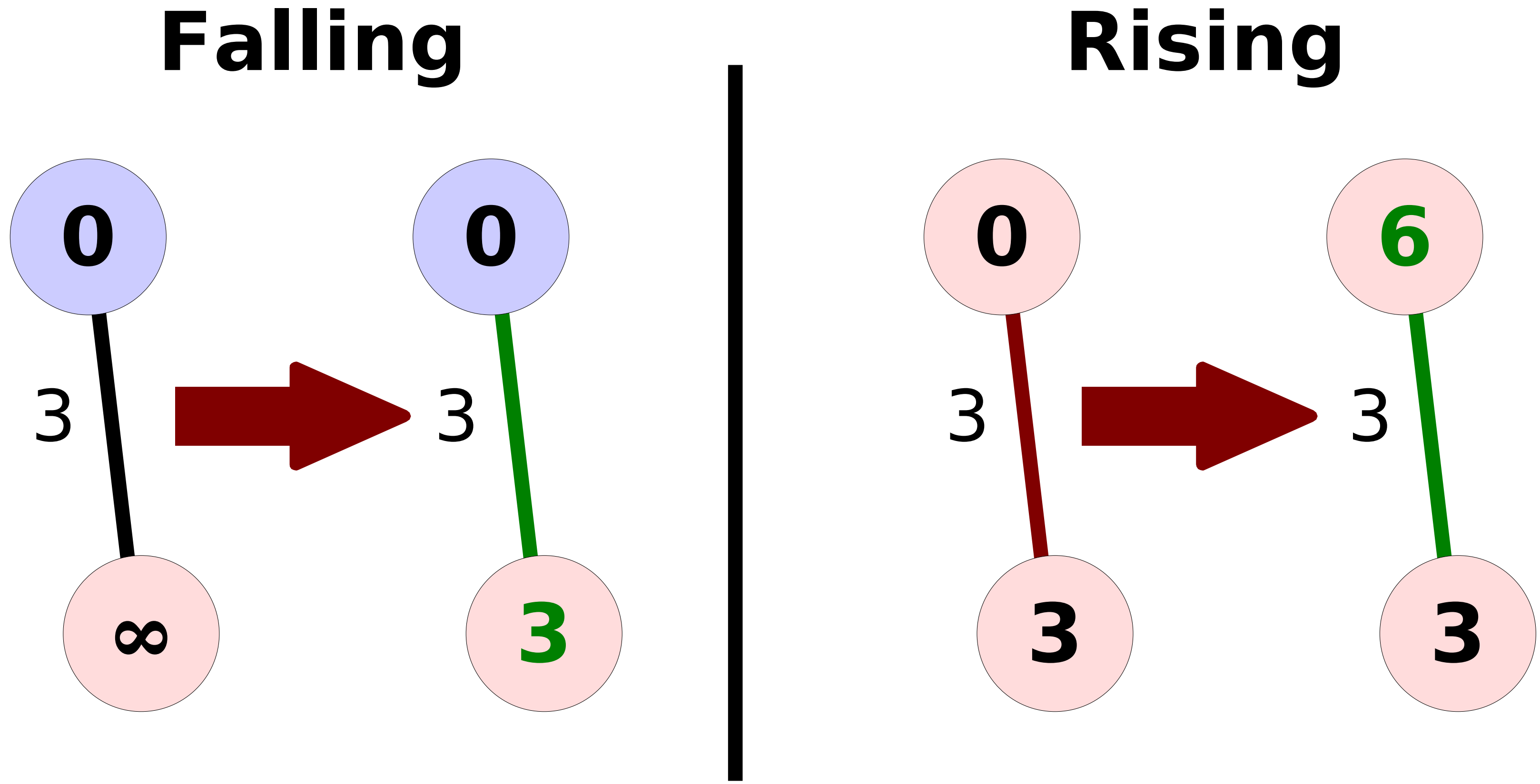
$$g_x = \begin{cases} 0 & if\ x \in S \\ min\{g_y + d(x,y) \mid y \in N_x\} & if\ x \notin S \end{cases}$$

Calculation By Relaxation



$$g_x = \begin{cases} 0 & \text{if } x \in S \\ \min\{g_y + d(x,y) \mid y \in N_x\} & \text{if } x \notin S \end{cases}$$

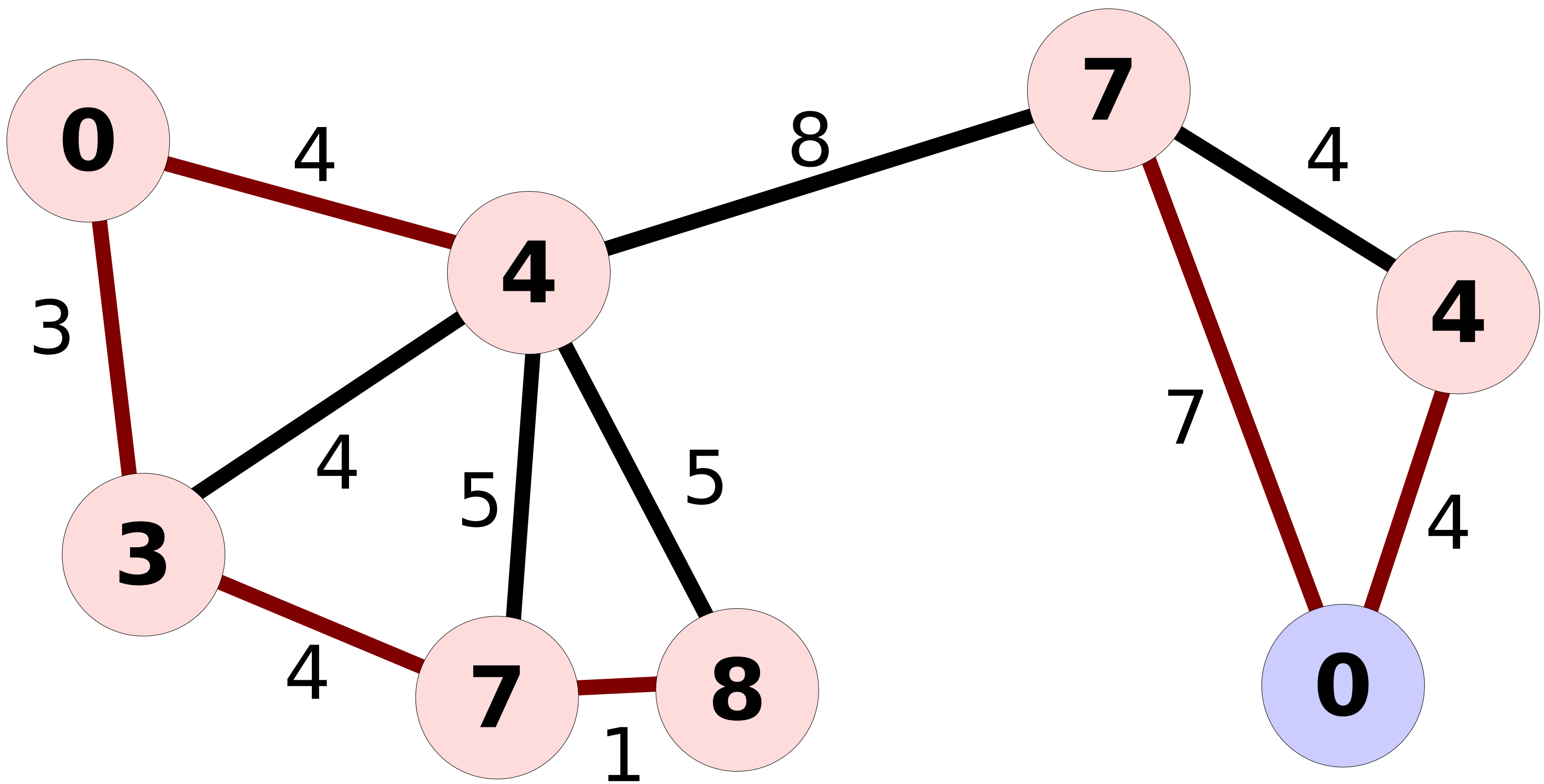
Gradient + Communication Lag



$$c_x(y,t) = g_x(t - \lambda_x(y,t)) + d(x,y)$$

$$g_x(t) = \begin{cases} 0 & \text{if } x \in S(t) \\ \min\{c_x(y,t) \mid y \in N_x(t)\} & \text{if } x \notin S(t) \end{cases}$$

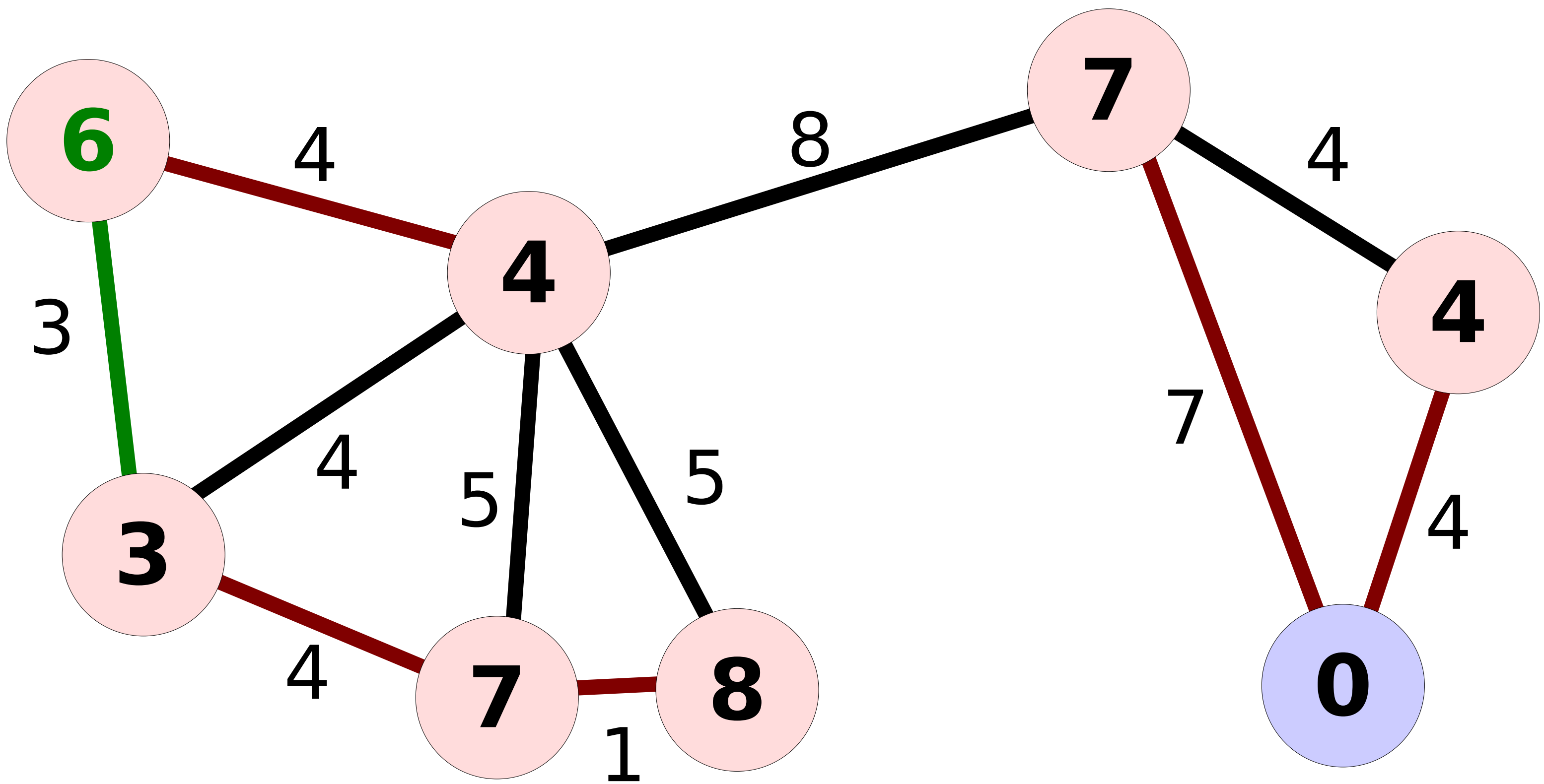
Rising Value Problem



$$c_x(y,t) = g_x(t - \lambda_x(y,t)) + d(x,y)$$

$$g_x(t) = \begin{cases} 0 & \text{if } x \in S(t) \\ \min\{c_x(y,t) \mid y \in N_x(t)\} & \text{if } x \notin S(t) \end{cases}$$

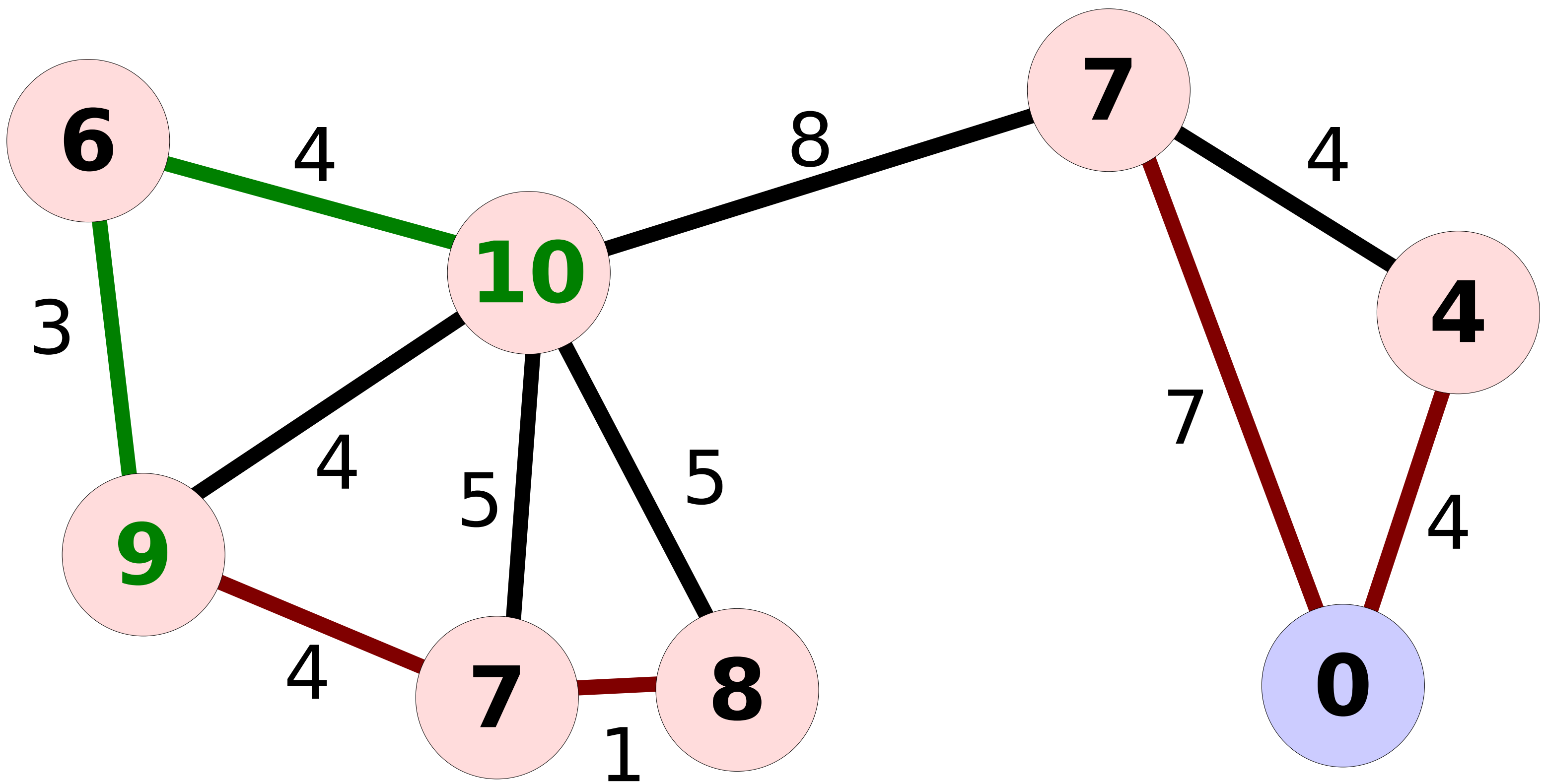
Rising Value Problem



$$c_x(y,t) = g_x(t - \lambda_x(y,t)) + d(x,y)$$

$$g_x(t) = \begin{cases} 0 & \text{if } x \in S(t) \\ \min\{c_x(y,t) \mid y \in N_x(t)\} & \text{if } x \notin S(t) \end{cases}$$

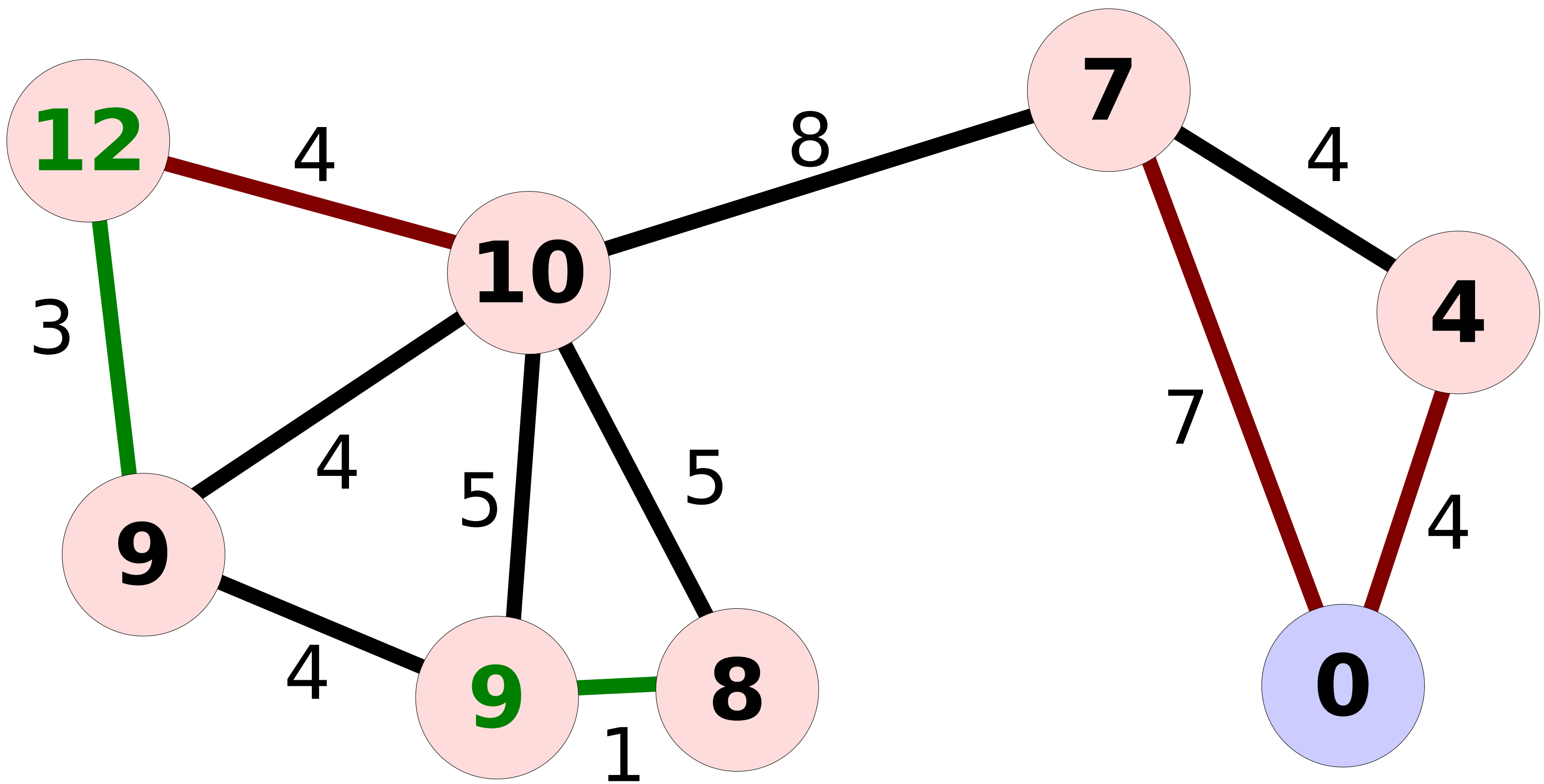
Rising Value Problem



$$c_x(y,t) = g_x(t - \lambda_x(y,t)) + d(x,y)$$

$$g_x(t) = \begin{cases} 0 & \text{if } x \in S(t) \\ \min\{c_x(y,t) \mid y \in N_x(t)\} & \text{if } x \notin S(t) \end{cases}$$

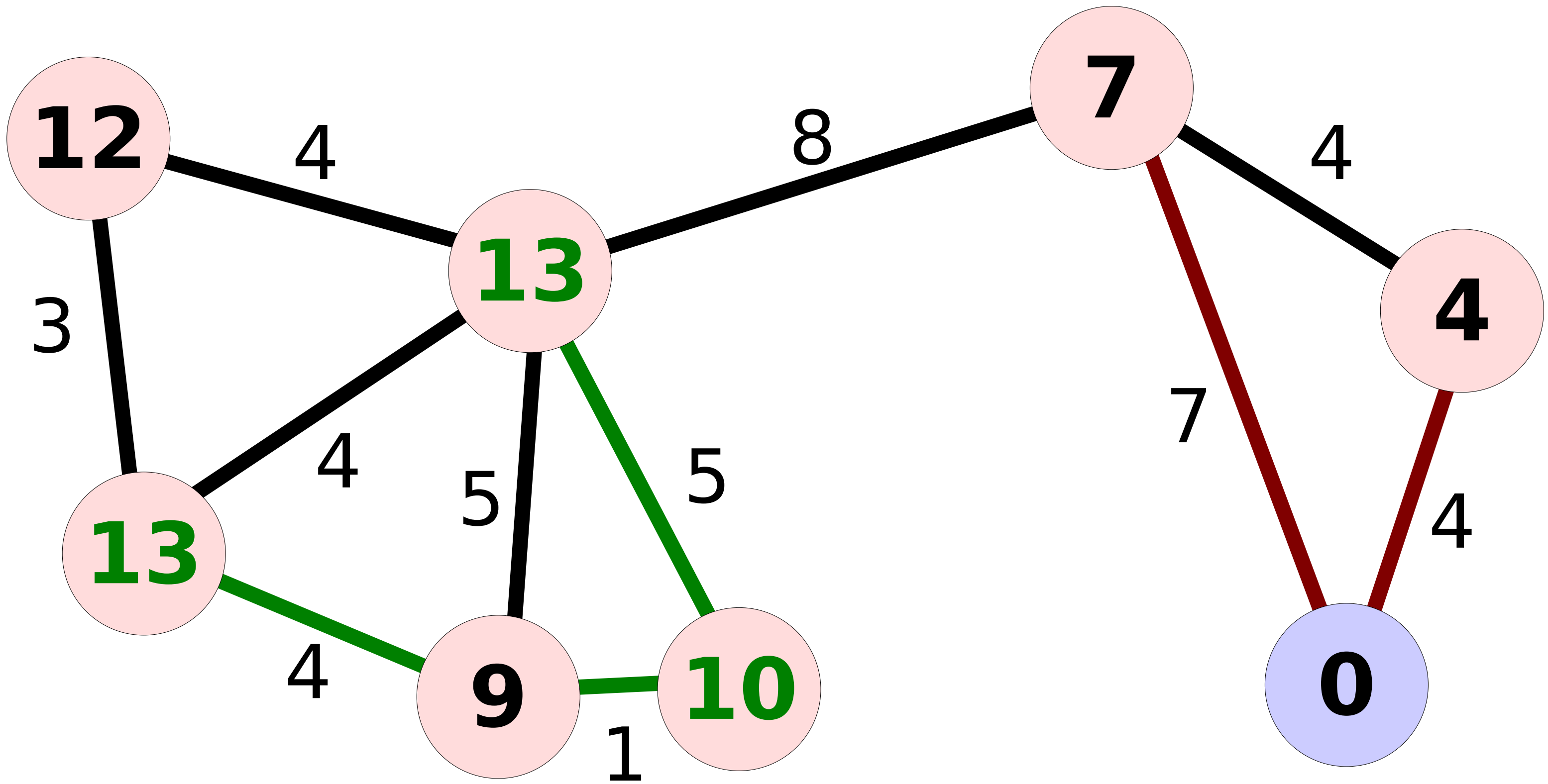
Rising Value Problem



$$c_x(y,t) = g_x(t - \lambda_x(y,t)) + d(x,y)$$

$$g_x(t) = \begin{cases} 0 & \text{if } x \in S(t) \\ \min\{c_x(y,t) \mid y \in N_x(t)\} & \text{if } x \notin S(t) \end{cases}$$

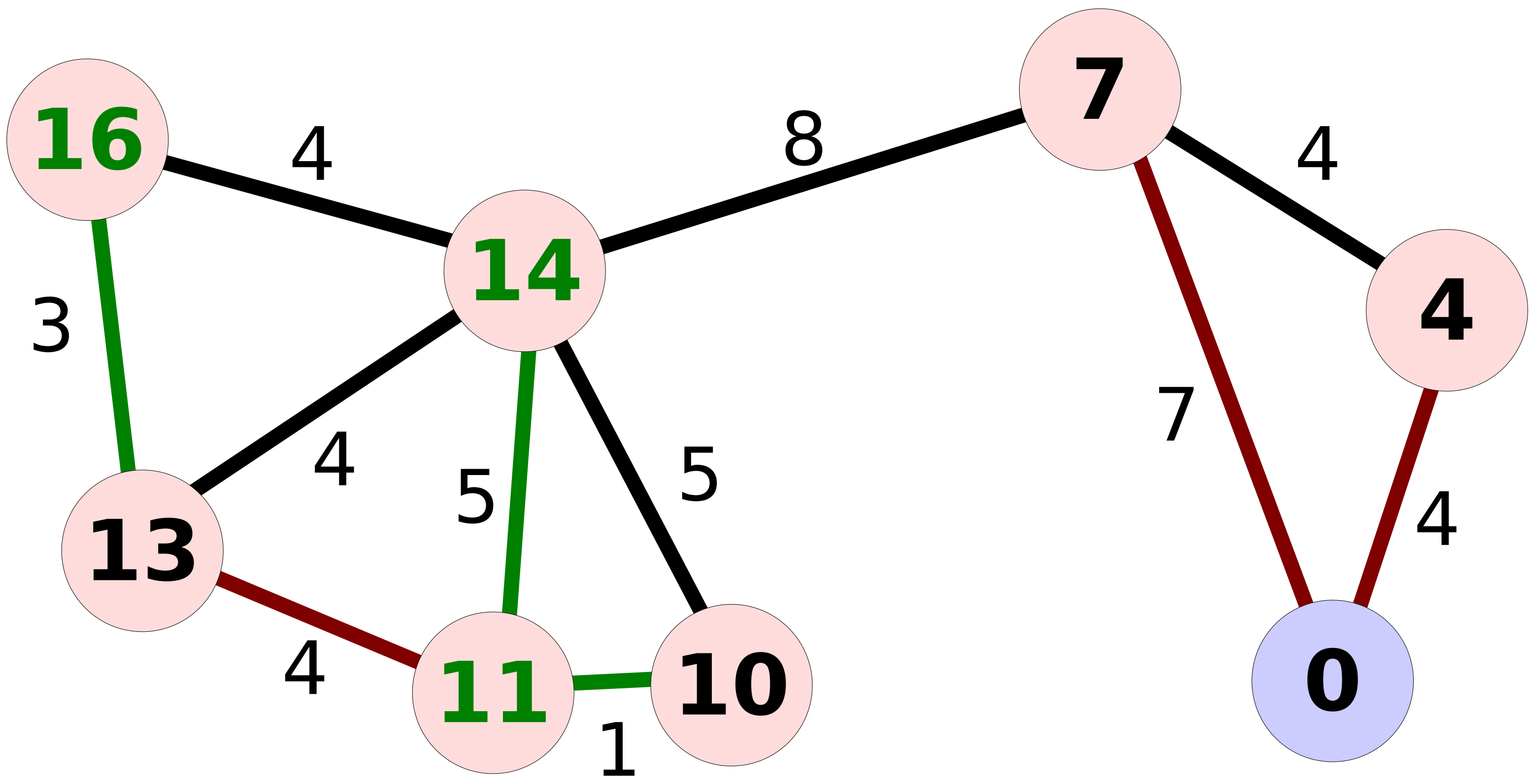
Rising Value Problem



$$c_x(y,t)=g_x(t-\lambda_x(y,t))+d(x,y)$$

$$g_x(t)=\begin{cases} 0 & if\ x\in S(t) \\ min\{c_x(y,t)\,|\,y\in N_x(t)\} & if\ x\notin S(t) \end{cases}$$

Rising Value Problem



$$c_x(y,t) = g_x(t - \lambda_x(y,t)) + d(x,y)$$

$$g_x(t) = \begin{cases} 0 & \text{if } x \in S(t) \\ \min\{c_x(y,t) \mid y \in N_x(t)\} & \text{if } x \notin S(t) \end{cases}$$

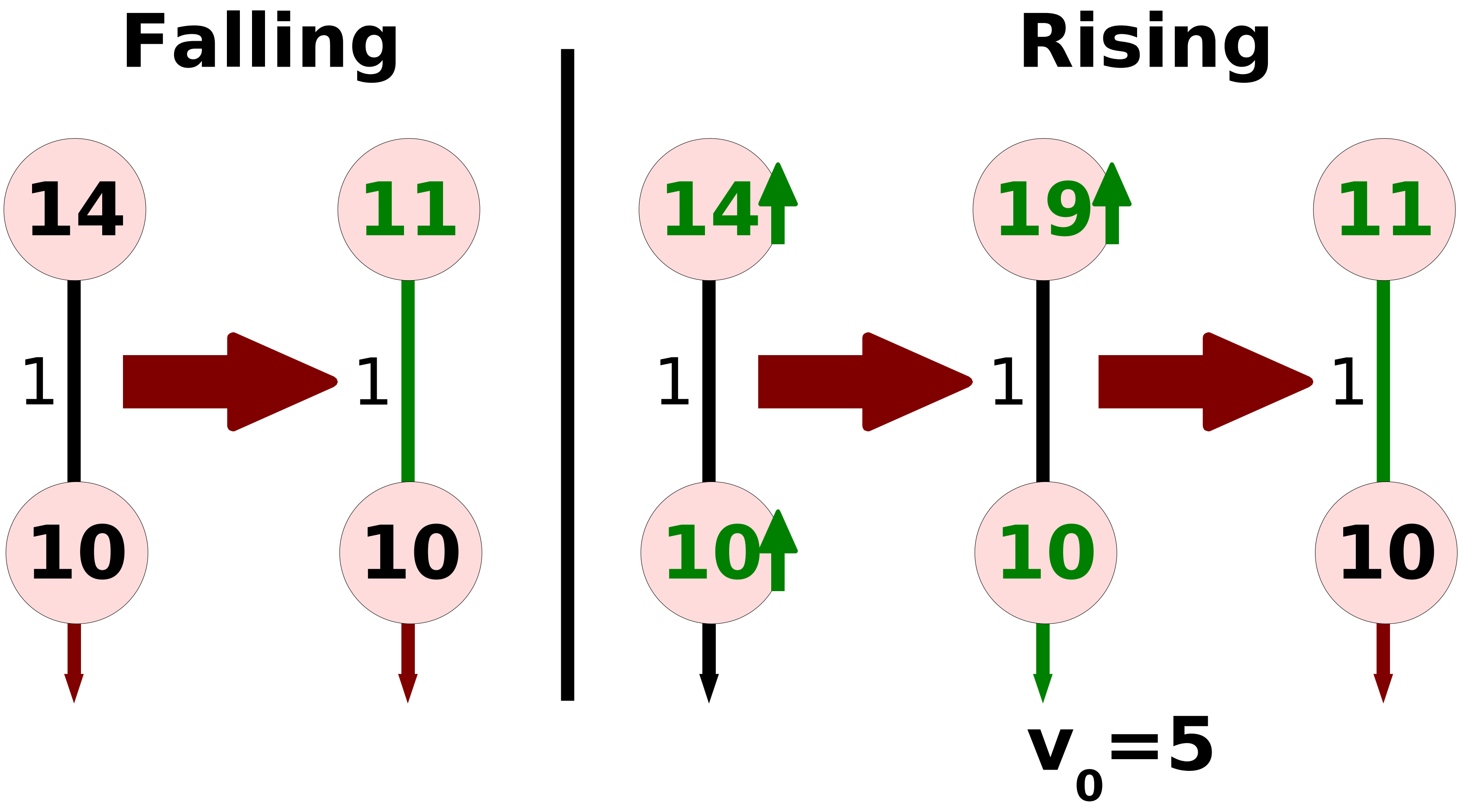
Previous Algorithms

- “Invalidate and Rebuild”
  - GRAB: single source, rebuild on high error
  - TTDD: static subgraph, rebuild on lost msg.
- “Incremental Repair”
  - Hopcount: Clement & Nagpal, Butera
  - Distorted Measure: Beal & Bachrach

    (naïve generalization of hopcount to continuous)

Can't exploit distance info in large nets

CRF-Gradient: Local Deconstraint



- Self-stabilization in *O(diameter)*

CRF-Gradient: Local Deconstraint

$$c_x(y,t) = g_x(t - \lambda_x(y,t)) + d(x,y)$$

$$c'_x(y,t) = c_x(y,t) + (\lambda_x(y,t) + \Delta_t) \cdot v_x(t)$$

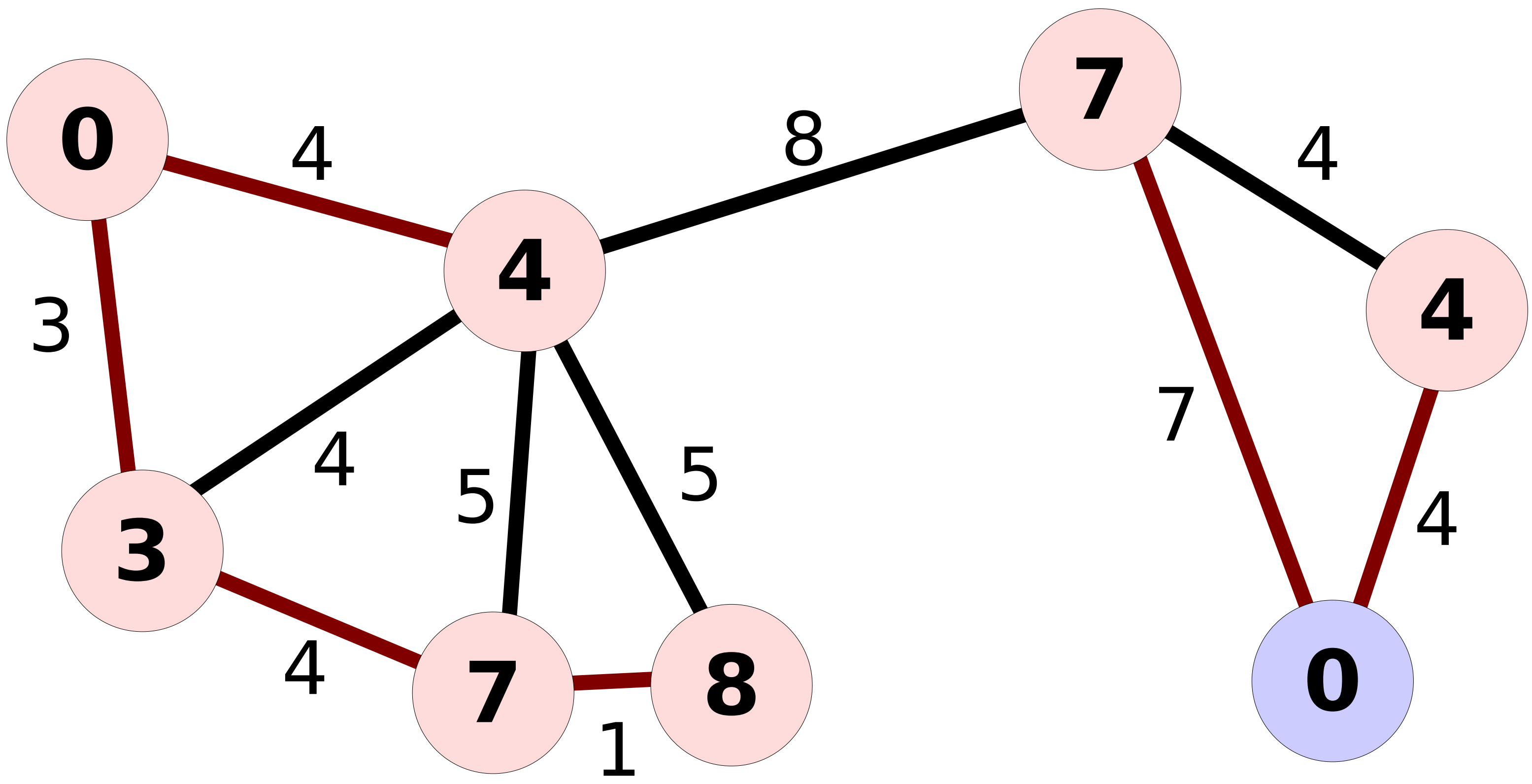
$$N'_x(t) = \{ y \in N_x(t) \mid c'_x(y,t) \le g_x(t - \Delta_t) \}$$

$$g_x(t) = \begin{Bmatrix} 0 & \text{if } x \in S(t) \\ \min\{ c_x(y,t) \mid y \in N'_x(t) \} & \text{if } x \notin S(t), N'_x(t) \neq \emptyset \\ g_x(t) + v_0 \cdot \Delta_t & \text{if } x \notin S(t), N'_x(t) = \emptyset \end{Bmatrix}$$

$$v_x(t) = \begin{Bmatrix} 0 & \text{if } x \in S(t) \\ 0 & \text{if } x \notin S(t), N'_x(t) \neq \emptyset \\ v_0 & \text{if } x \notin S(t), N'_x(t) = \emptyset \end{Bmatrix}$$

- Self-stabilization in *O(diameter)*

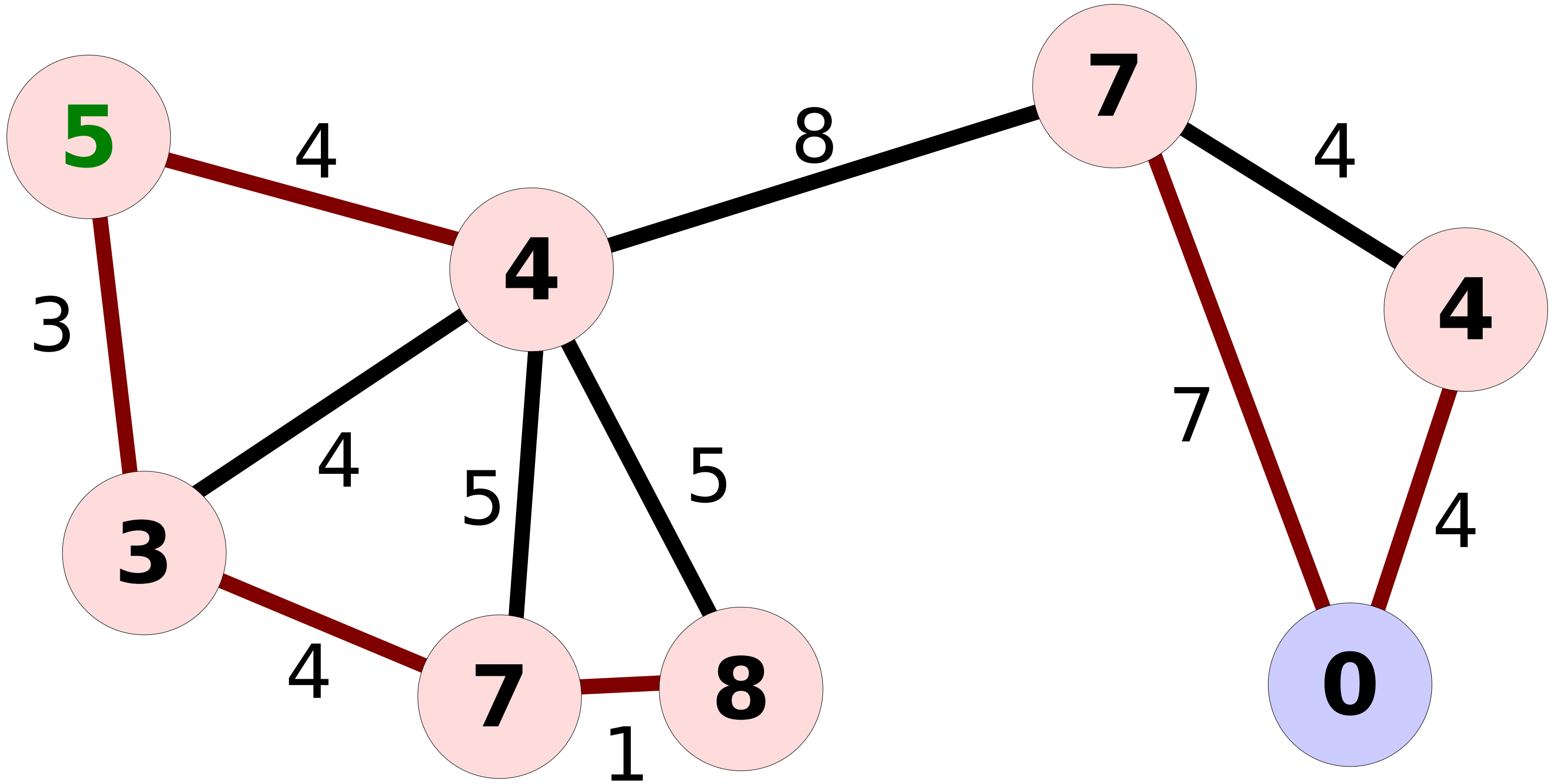
CRF Rising Values



- *zero at source*
- *rise at v_0 with relaxed constraint*
- *otherwise snap to constraint*

$\mathbf{v_0=5}$

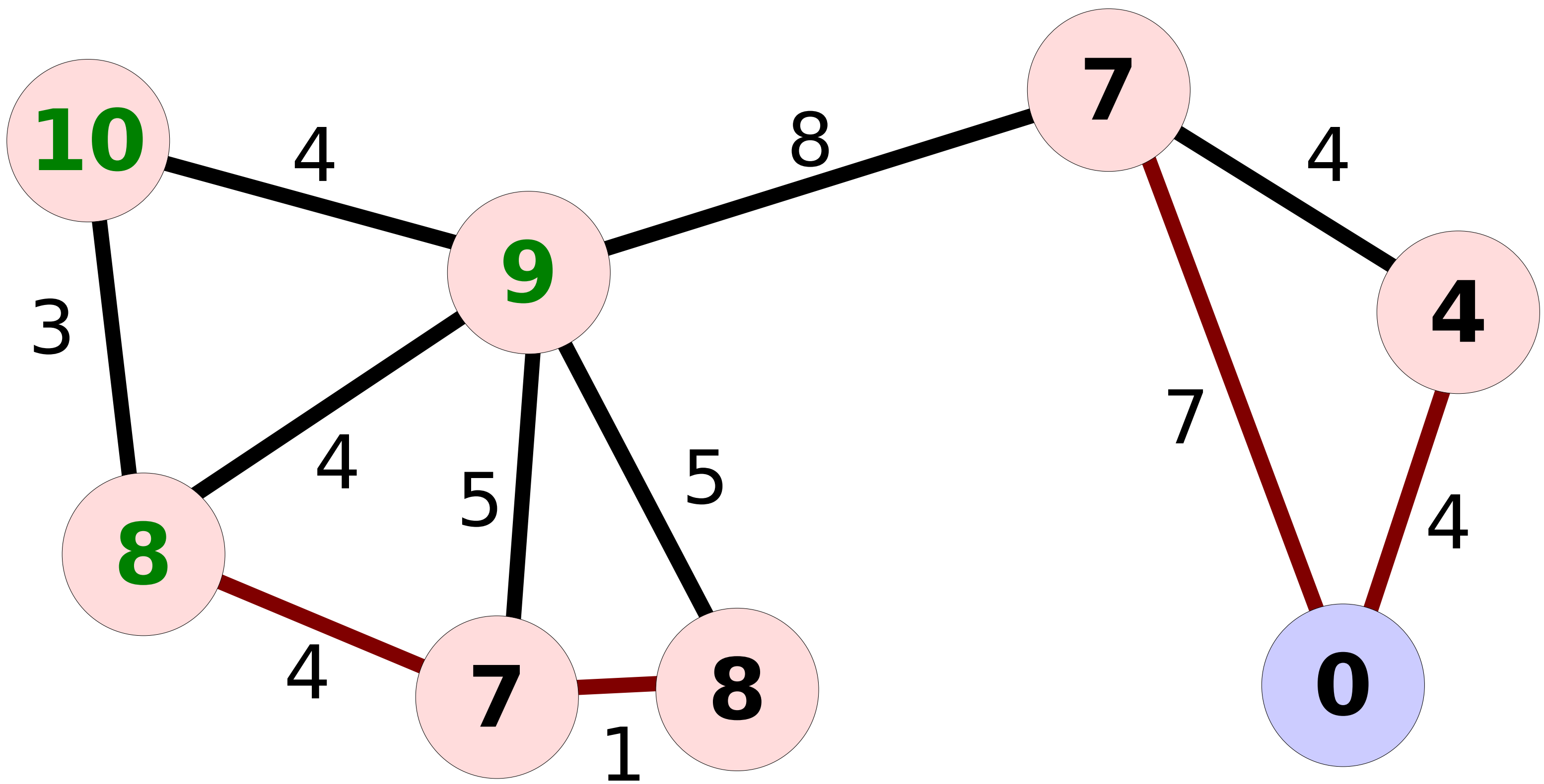
CRF Rising Values



- *zero at source*
- *rise at v_0 with relaxed constraint*
- *otherwise snap to constraint*

$\mathbf{v_0=5}$

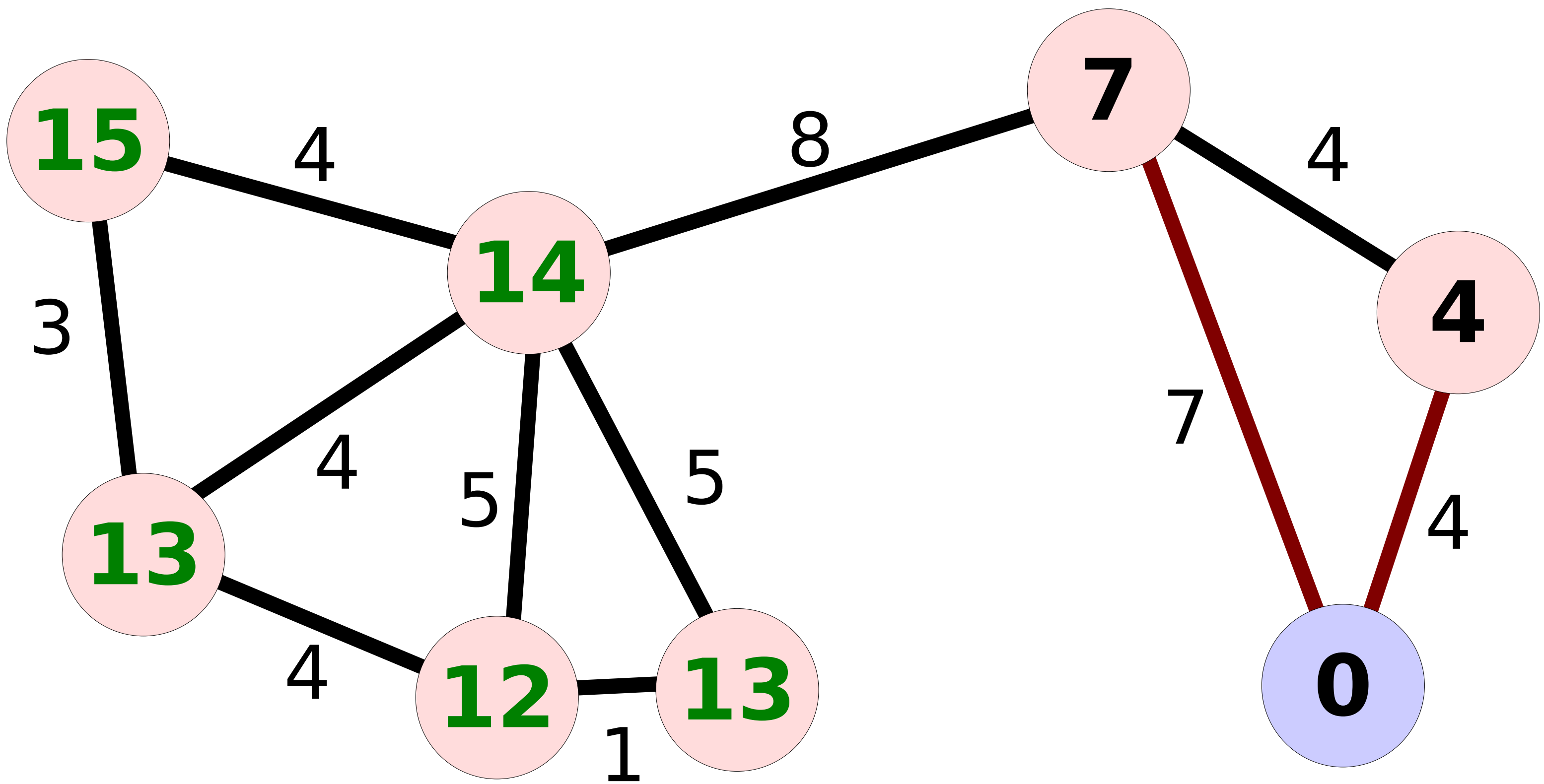
CRF Rising Values



- *zero at source*
- *rise at v_0 with relaxed constraint*
- *otherwise snap to constraint*

$\mathbf{v_0=5}$

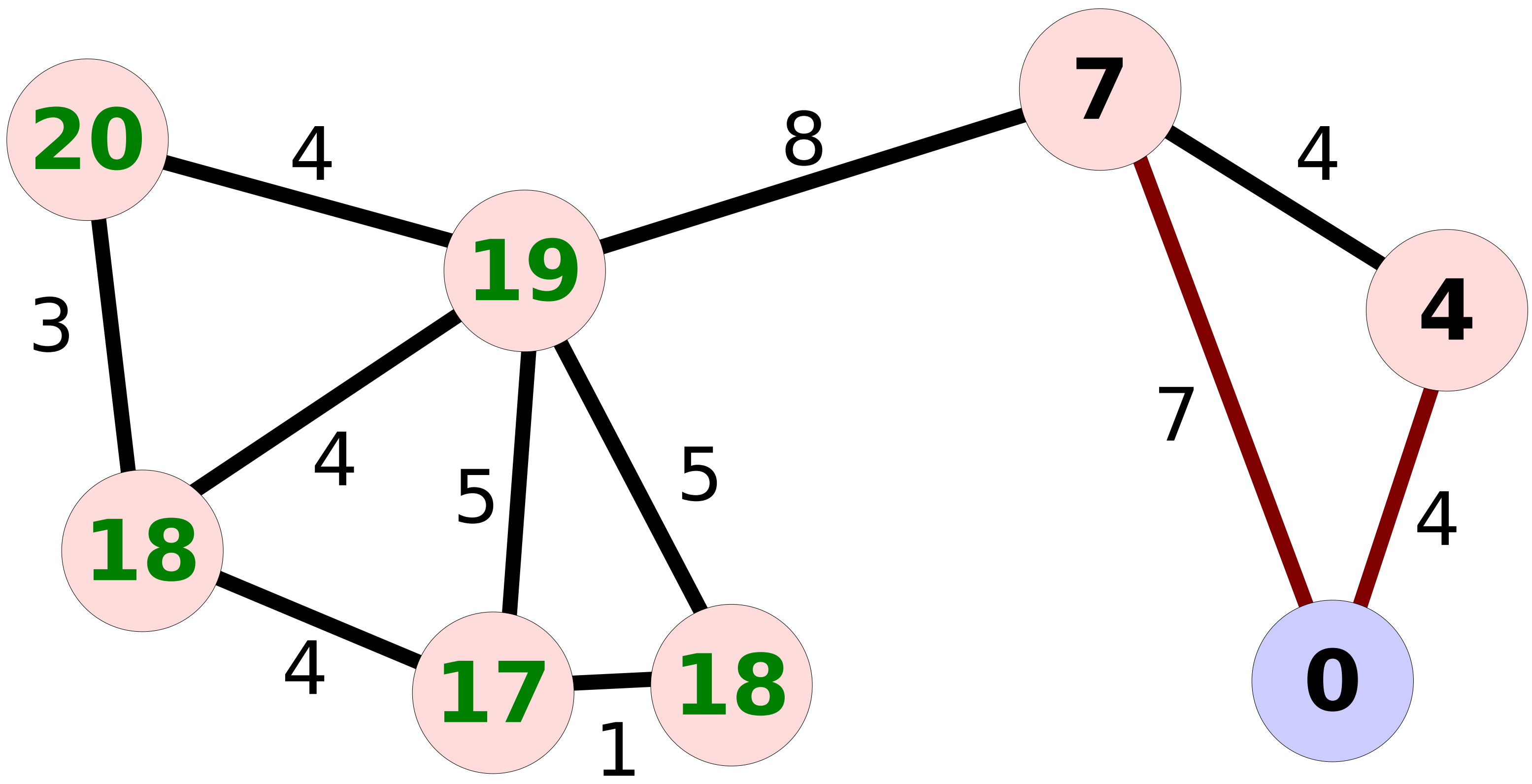
CRF Rising Values



- *zero at source*
- *rise at v_0 with relaxed constraint*
- *otherwise snap to constraint*

$\mathbf{v_0=5}$

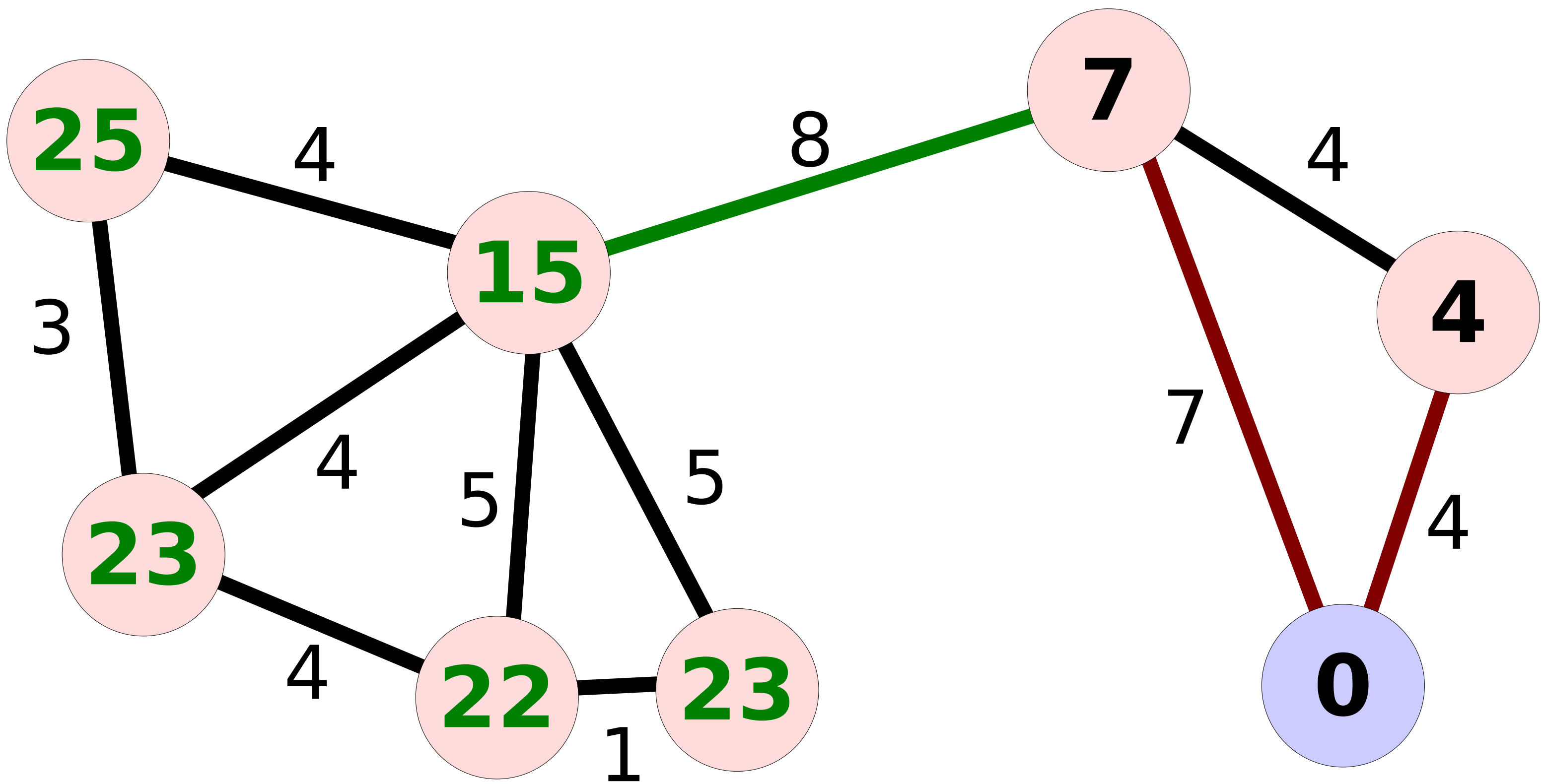
CRF Rising Values



- *zero at source*
- *rise at v_0 with relaxed constraint*
- *otherwise snap to constraint*

$\mathbf{v_0=5}$

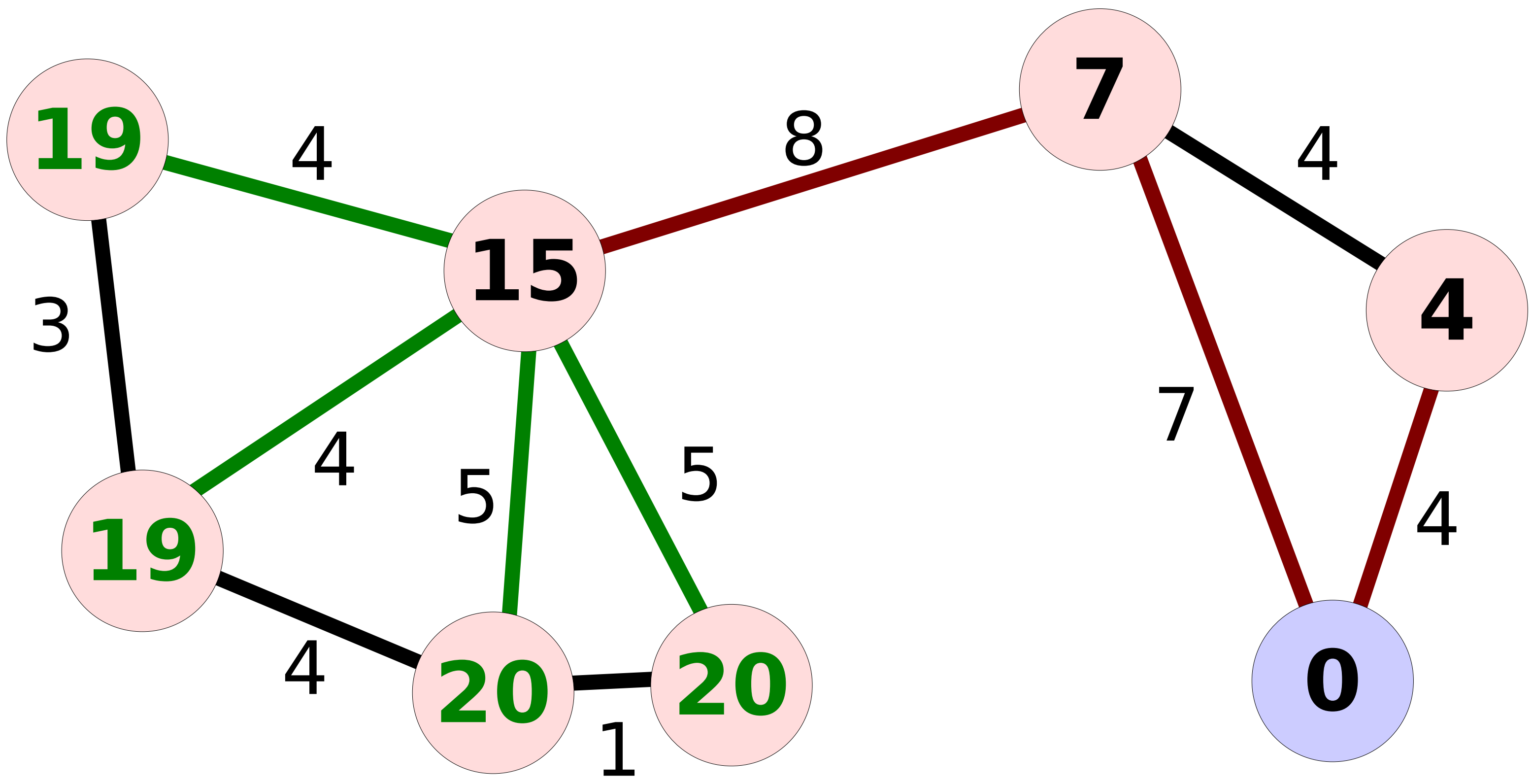
CRF Rising Values



- *zero at source*
- *rise at v_0 with relaxed constraint*
- *otherwise snap to constraint*

$\mathbf{v_0=5}$

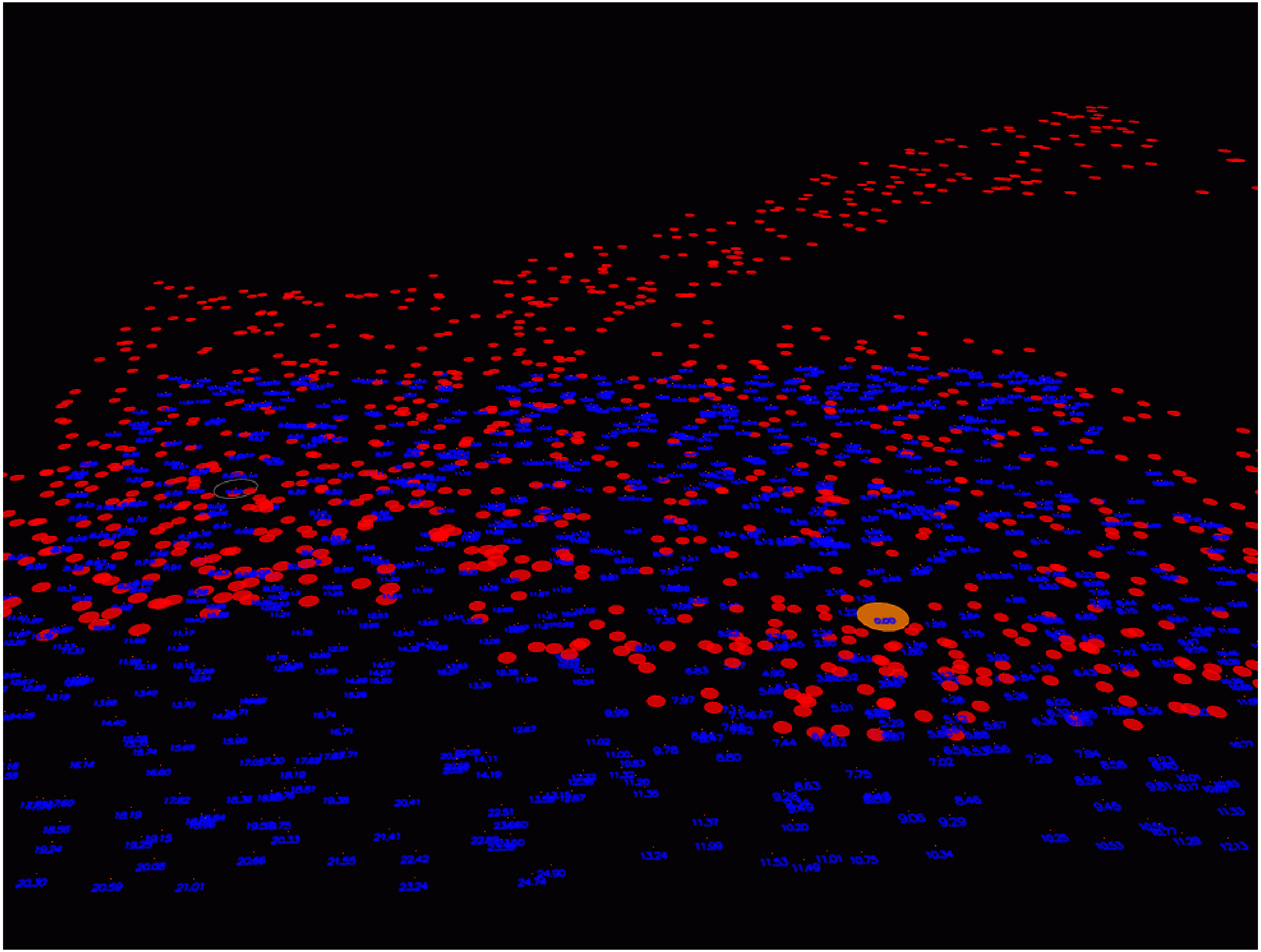
CRF Rising Values



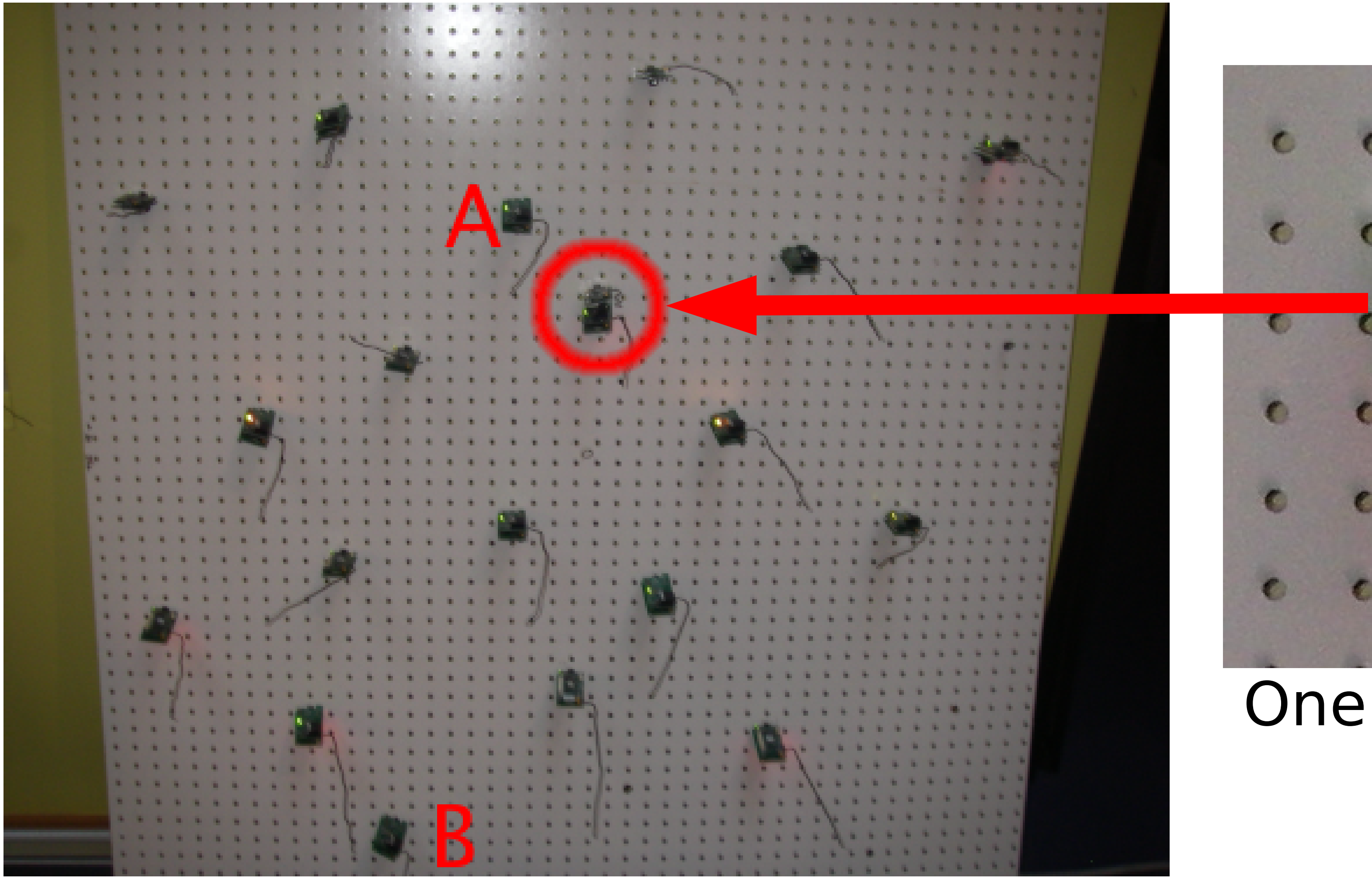
- *zero at source*
- *rise at v_0 with relaxed constraint*
- *otherwise snap to constraint*

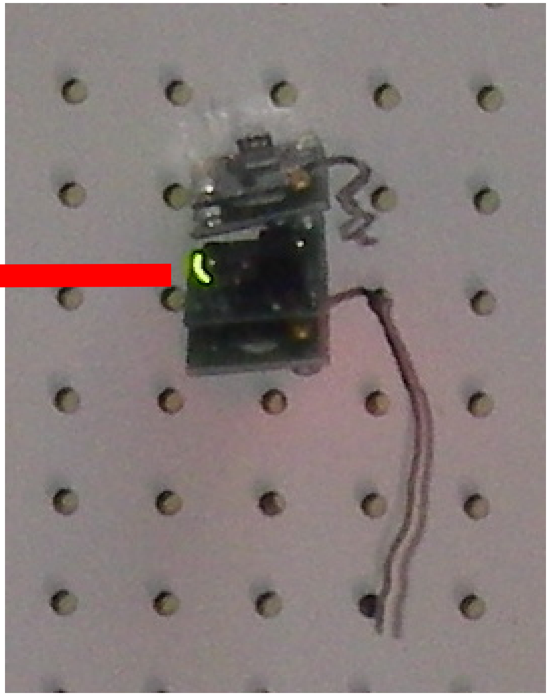
$\mathbf{v_0=5}$

Simulated CRF-Gradient



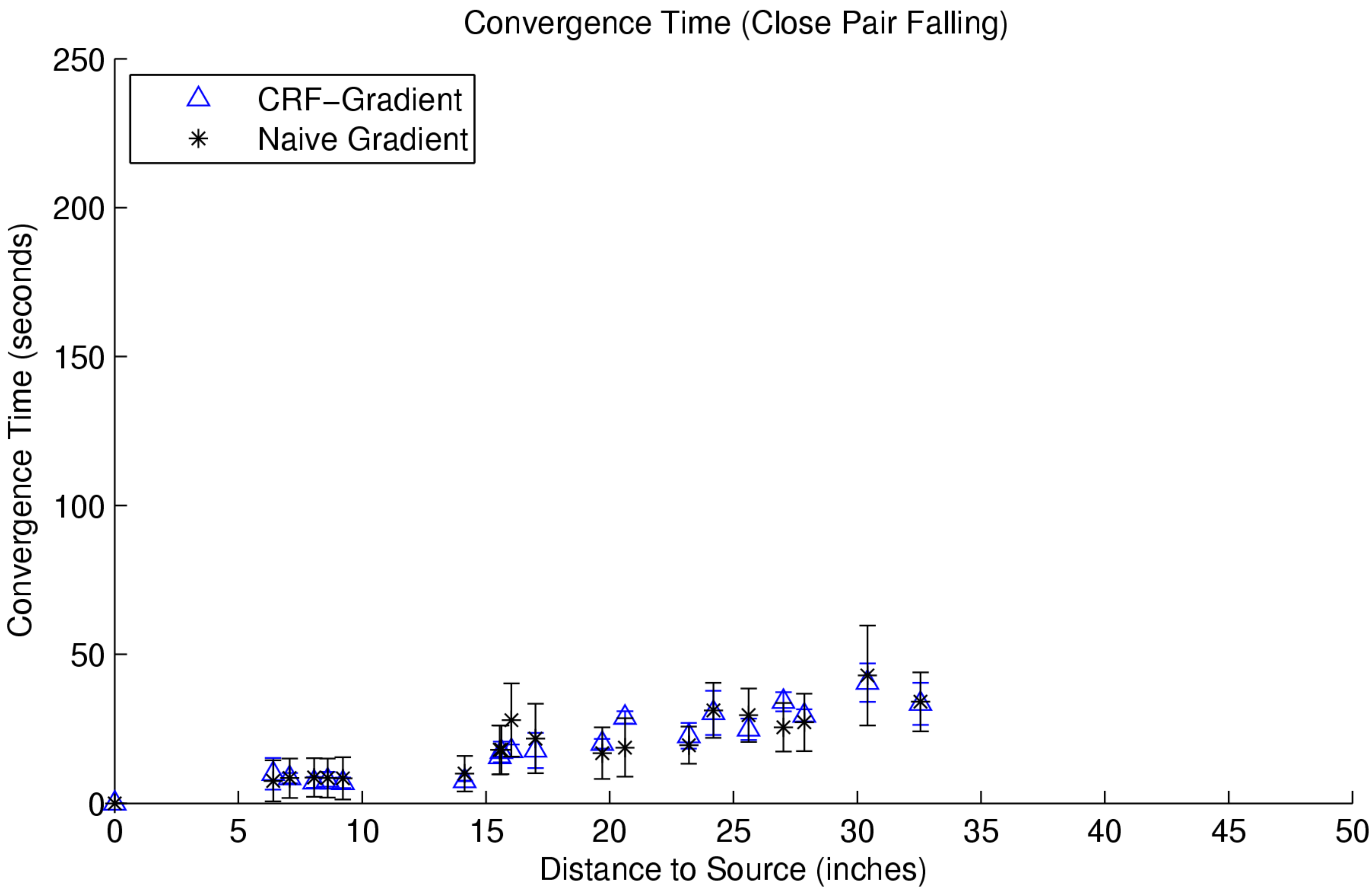
Experimental Setup



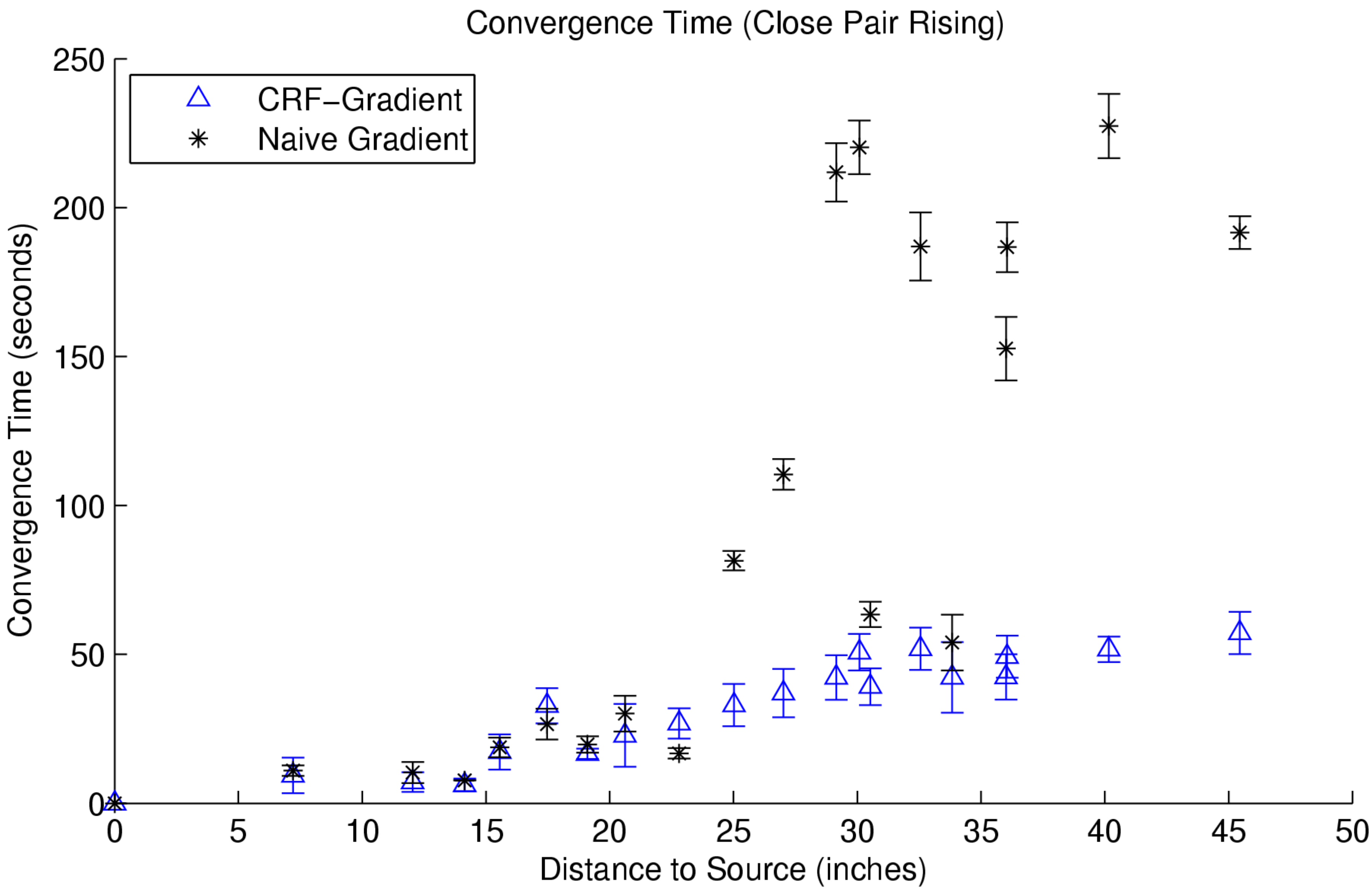


One close pair

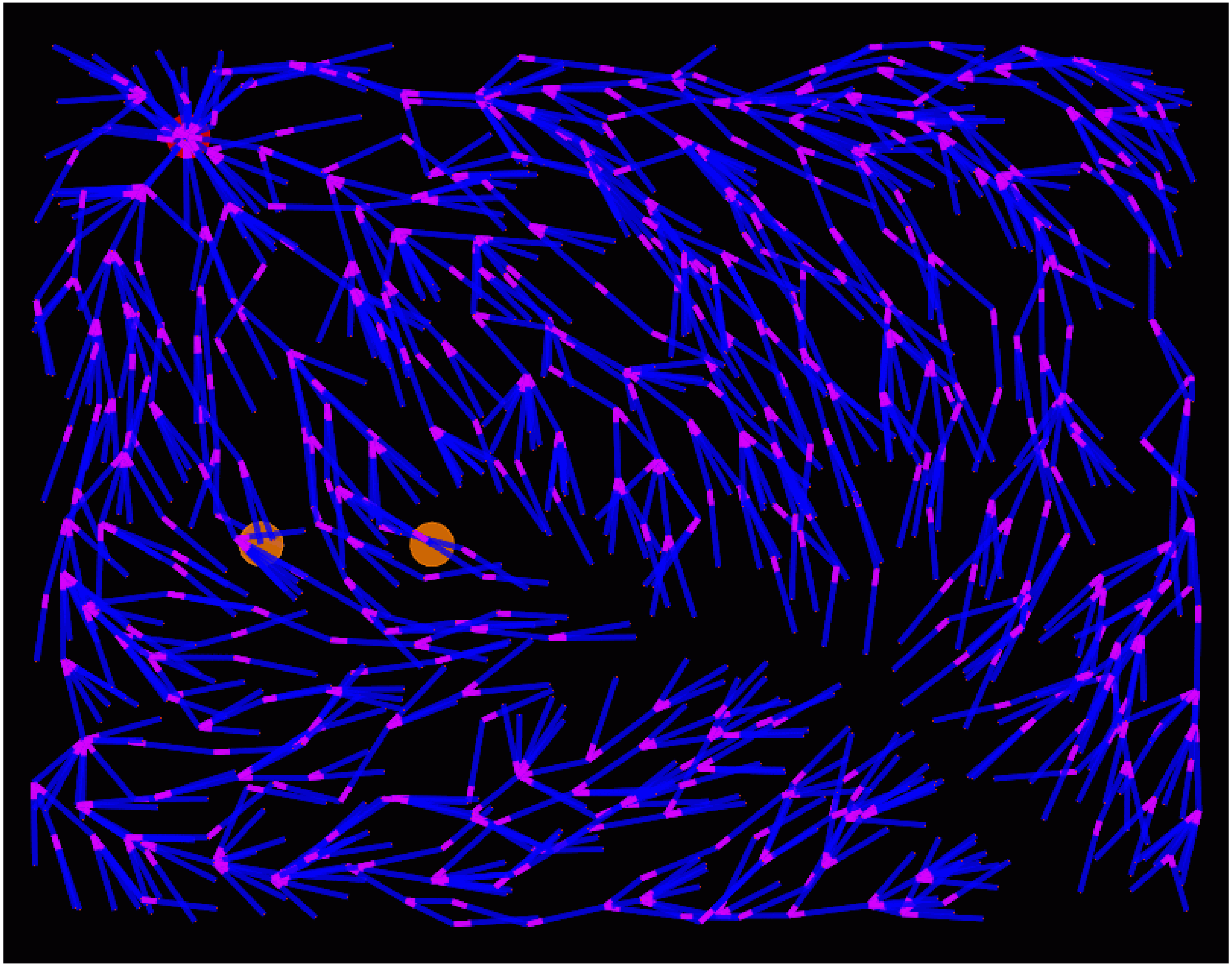
Experimental Results: Falling



Experimental Results: Rising



Generalized CRF



Contributions

- Rising Value Problem
- CRF-Gradient: self-stabilize in *O(diameter)*
- Verified in simulation and on Mica2 Motes