Dynamic Networks

Wednesday, July 30, 2014
11:23 PM

modern dis. systems are often dynamic
$\rightarrow$ structure changes over time

e.g., wireless networks, p2p networks

goal: study effect on distributed algorithms

Model:

- synchronous communication (rounds)
- set of nodes V is fixed (n nodes)
- set of edges is dynamic

  each round, possibly a new set of edges

  round r : graph $G(r) = (V, E(r))$

- $E(r)$ determined by an adversary
  (worst-case dynamic networks)

  ~~what about $E(r) = \emptyset$ $(\forall r)$~~

- assumption:

  $\forall r \geq 1$: $G(r)$ is connected

  (only assumption)

- assume unique IDs
- communication by local broadcast

  in each round r, each node v
  sends message $M(v,r)$ to all its neighbors

Variants:

nodes know n / nodes don't know n

synchronous start / asychronous start

all nodes start alg. at time 0

nodes can start at any time (spontaneously, when rcv. first message)

Broadcasting a single piece of information

token

single source v, message M

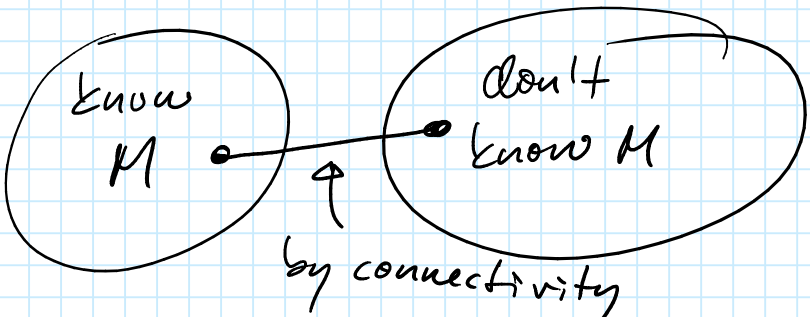
token

send M to everyone by flooding

→ flooding: all nodes that know M, broadcast M

How long until all nodes know M?

in each round



all nodes know M after ≤ $n-1$ rounds

Let's assume that nodes don't know n ...

How to stop broadcasting?

Problems:

1) Counting: compute n

2) Token Dissemination:

k tokens (pieces of inf.)

$\Rightarrow$ goal: send all k tokens to everyone

Counting:

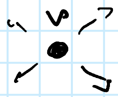
asynchronous start:

assumption: algorithms need to terminate after $T(n)$ rounds

v starts the computation, $ID(v) = 1$

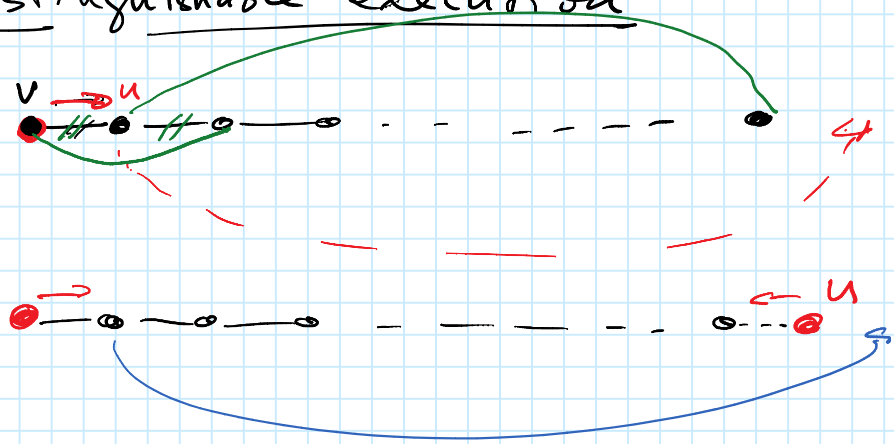
Counting is not possible

- Consider solo execution of v



$\Rightarrow$ after $T(1)$ rounds, v terminates and outputs "$n = 1$"

- Indistinguishable execution



choose $n > 2T(1) + O(1)$

after $T(1)$ rounds, v terminates outputting "$n=1$"

2 solutions

→ stronger connectivity requirement
2-interval connectivity
$\forall r: \quad G'(r) = (V, E(r) \cap E(r+1))$
is connected
→ can solve counting

→ **synchronous start**

Counting

(no restriction on message size)

"in every round all nodes broadcast the set of known IDs"

After round r, node u knows $A_u(r)$

$A_u(0) = \{ ID(u) \}$

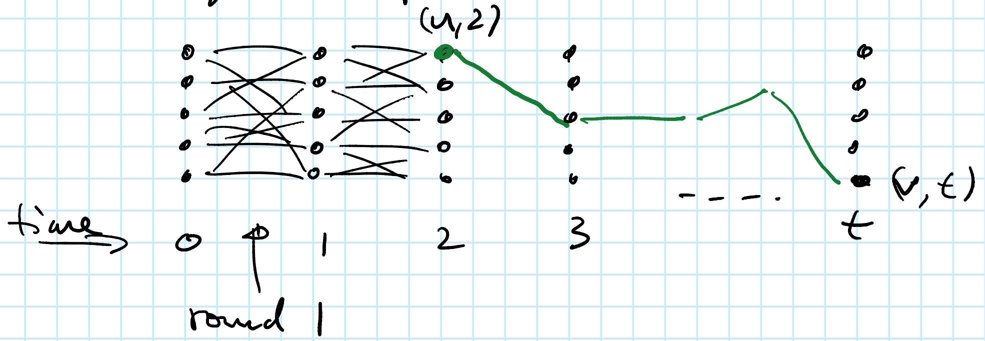
After $n-1$ rounds: $\forall v \in V: \quad A_v(n-1) = V$

Can we say something about $|A_v(r)|$?

<u>Claim:</u> $\forall v, r: \quad |A_v(r)| \geq \min\{r+1, n\}$

<u>Def:</u> (u,t) und (v,t') $\quad (t' \geq t)$

"dynamic path from (u,t) to (v,t')"

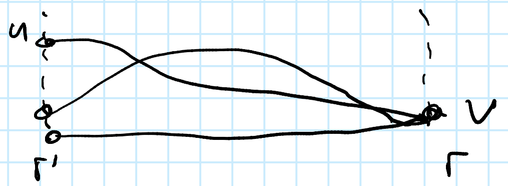


Claim:

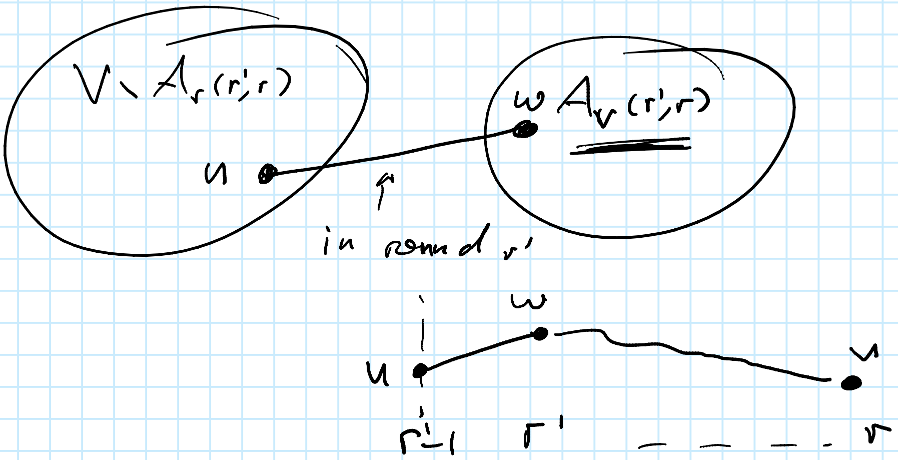
$u \in A_v(r)$ iff $(u,0) \rightsquigarrow (v,r)$

$A_v(r',r)$: set of nodes s.t.

$(u,r') \rightsquigarrow (v,r)$



$$|A_v(r'-1,r)| \geq |A_v(r',r)| + 1$$



$\Longrightarrow$ $|A_v(r)| \geq \min\{r+1, n\}$

$\Longrightarrow$ interesting question

what happens if msg. size restricted

$\Longrightarrow$ $O(\log n)$ bits per message

$\Longrightarrow$ best alg. we know needs $\Theta(n^2)$ rounds

$\longrightarrow$ T-interval connectivity : $O(n + \frac{n^2}{T})$ rounds

Algorithm uses token dissemination

→ all nodes learn all IDs

→ exchanging all IDs:

broadcast IDs one after the other

exchanging k tokens

assumption 1 token per message

know n, bruteforce: $O(n \cdot k)$

Brute-force is almost optimal

→ deterministic, $\underbrace{\text{store-and-forward}}_{\text{forward tokens as they are}}$

Tokens known by node v at time t: $A_v(t)$

det. alg:

round $r+1$:

each node v picks $x \in A_v(r)$ ($x_v(r+1)$)

v broadcasts x to neighbors

our job: determine graph

free edge: (round $r+1$)

nodes u & v:

$x_v(r+1) \in A_u(r) \wedge x_u(r+1) \in A_v(r)$

⟹ can add edge $\{u,v\}$

⟹ no progress on $\{u,v\}$ in rd. $r+1$

strategy: 1. add all free edges

2. add additional edges to connect graph

define second set $A'_v(t)$
(for node v, time t)

$$A''_v(t) \subseteq A'_v(t+1)$$

"pretend" that at time t, v knows $A_v(t) \cup A'_v(t)$

$$\phi(t) = \sum_{v \in V} |A_v(t) \cup A'_v(t)|$$

solve token diss. $\Longrightarrow \phi(t) \geq n \cdot k$

How to choose A'_v?

For all nodes v: $A'_v(0)$ contains every token independently with probability $1/2$.

initial potential: $\phi(0) \leq \frac{2}{3} \cdot n \cdot k$ w.h.p.

Edge free: $(\{u,v\})$ (round r)

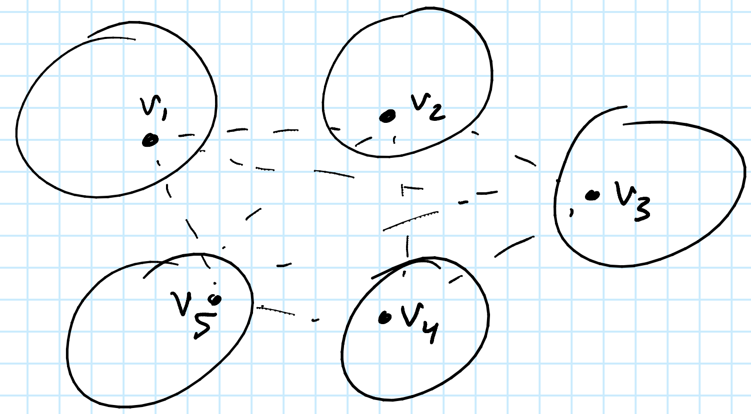
$x_u(r) \in A'_v(0)$, $x_v(r) \in A'_u(0)$ or if $x_u(r) = x_v(r)$

"new free" $\rightarrow$ "old free"

Observation:
We can add all free edges, alg. does not make progress w.r.t. $\phi(t)$

Goal: after adding all free edges, at most $O(\log n)$ components

Assume: after adding free edges, we have ℓ components



round r

ℓ components $\longrightarrow$ independent set of size ℓ

non-free edges induce a clique of size ℓ

and $\quad X_{v_i}(r) \neq X_{v_j}(r)$

$$X_{v_i}(r) \notin A'_{v_j}(0) \text{ or } \overbrace{X_{v_j}(r) \notin A'_{v_i}(0)}^{\mathcal{E}_{j,i}}$$

if $\{v_1, \dots, v_\ell\}$ and $\{X_{v_1}(r), \dots, X_{v_\ell}(r)\}$ fixed

$\Pr(\mathcal{E}_{ij}) = 1/2$

$\mathcal{E}_{ij}$ are independent

$\mathcal{E}_{ij}$ and $\mathcal{E}_{i'j'}$ because $X_{v_i}(r) \neq X_{v_{i'}}(r)$

$$\Pr\Big(\bigcap_{i<j} \underbrace{(\mathcal{E}_{i,j} \cup \mathcal{E}_{j,i})}_{3/4}\Big) = \left(3/4\right)^{\binom{\ell}{2}}$$

sets $\{v_1, \dots, v_\ell\} = \binom{n}{\ell} \leq n^\ell$

sets $\{X_{v_1}(r), \dots, X_{v_\ell}(r)\} = k^\ell$

Probability that #comp. $\geq \ell$

$$\leq n^{\ell} \cdot k^{\ell} \cdot \left(\frac{3}{4}\right)^{\binom{\ell}{2}}$$

$$= \exp\left(\ell \cdot \log(nk) - \binom{\ell}{2} \underbrace{\log\left(\tfrac{4}{3}\right)}_{2c}\right)$$

$$= \exp\left(\ell \cdot \left(\log(nk) - (\ell-1)\cdot c\right)\right)$$

choose ℓ s.t. $c(\ell-1) = 2\log(nk)$

$$\leadsto = e^{-\ell \log(nk)} = e^{-\Theta(\log^2 n)}$$

$\Longrightarrow$ progress in each round at most $O(\log(n))$

on $\phi(t)$

$\Longrightarrow$ Thm: need $\Omega\left(\frac{nk}{\log(n)}\right)$ rounds to solve k token diss.

(det. alg., store-and-forward)