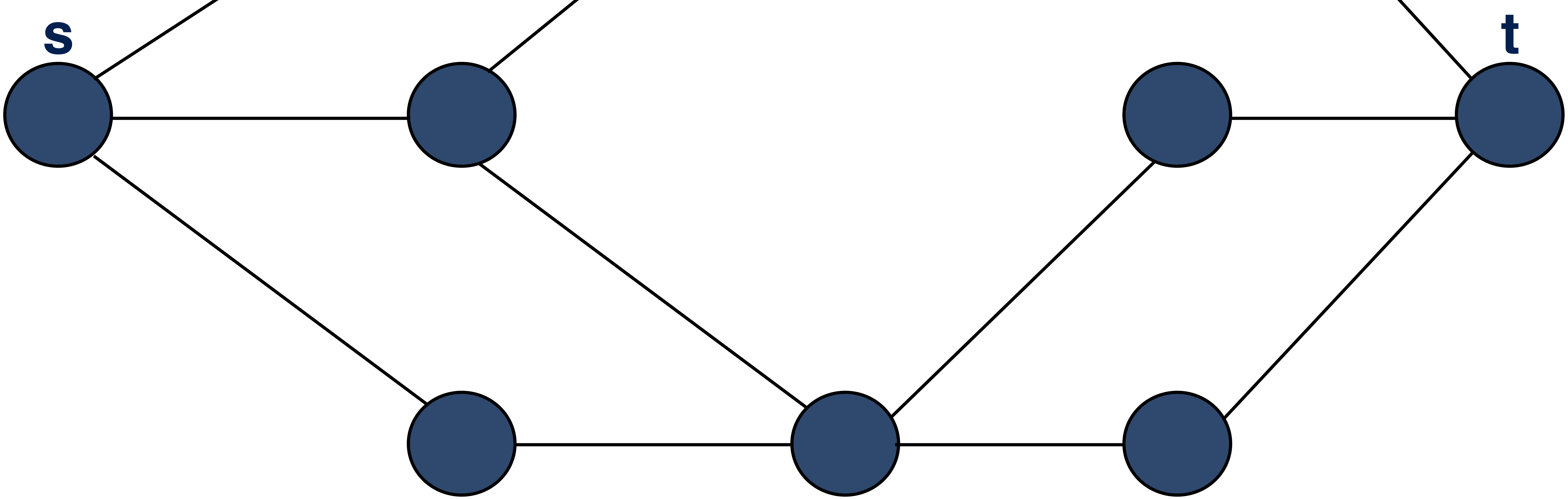
Saving Critical Nodes with Firefighters is FPT

Jayesh Choudhari*, Anirban Dasgupta*,
Neeldhara Misra*, M. S. Ramanujan†

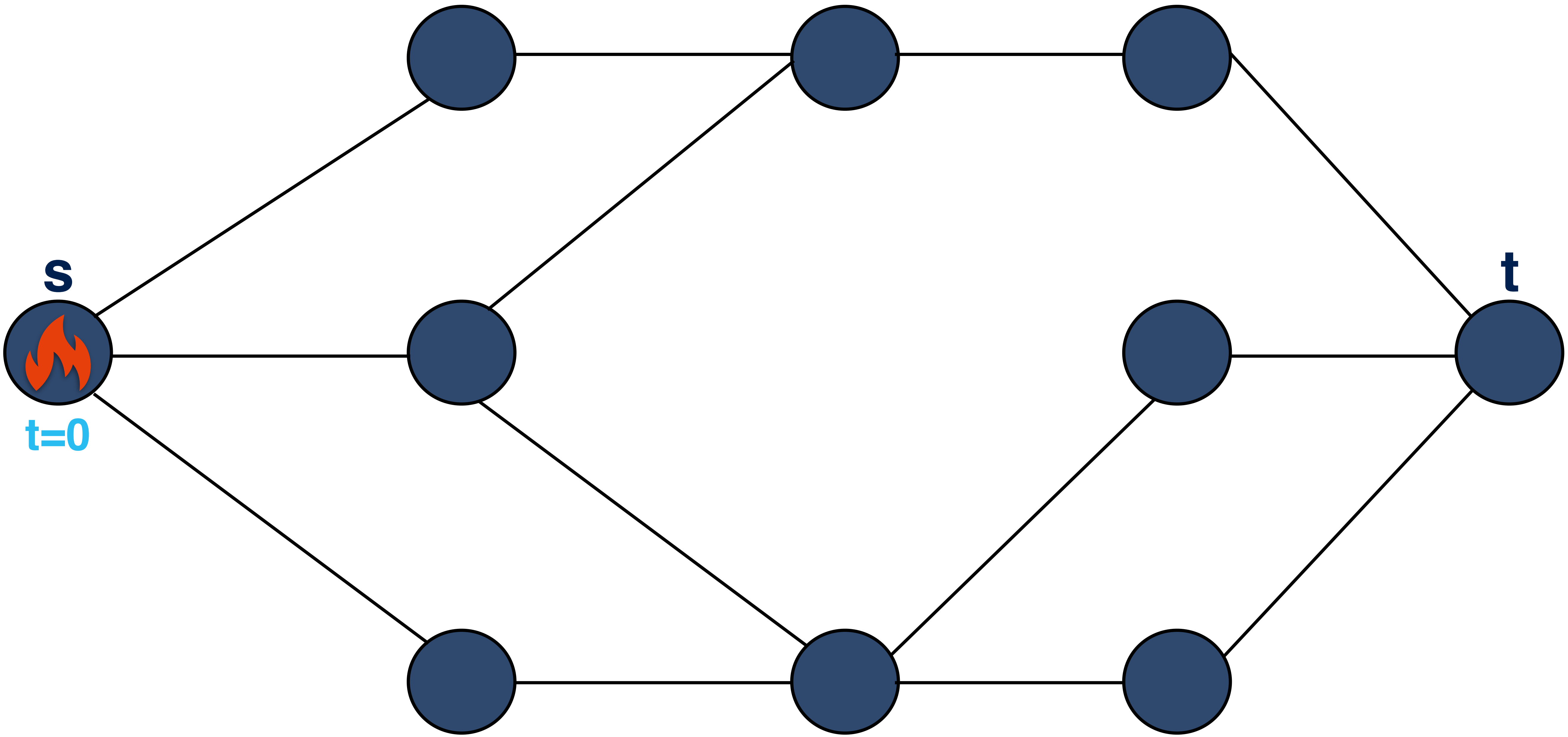
* - IIT Gandhinagar, † - University of Vienna

Firefighting

The Firefighting Problem

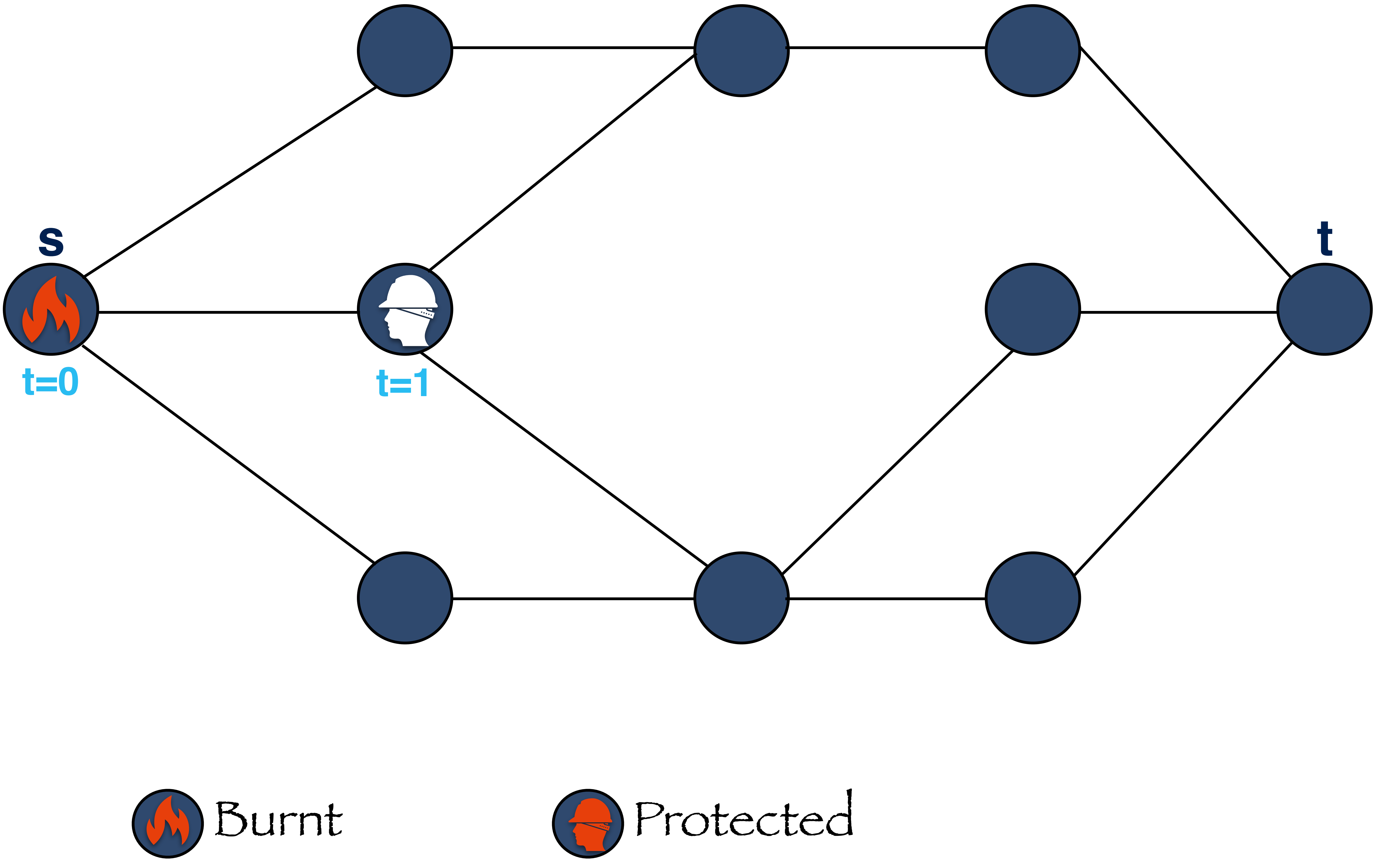


The Firefighting Problem

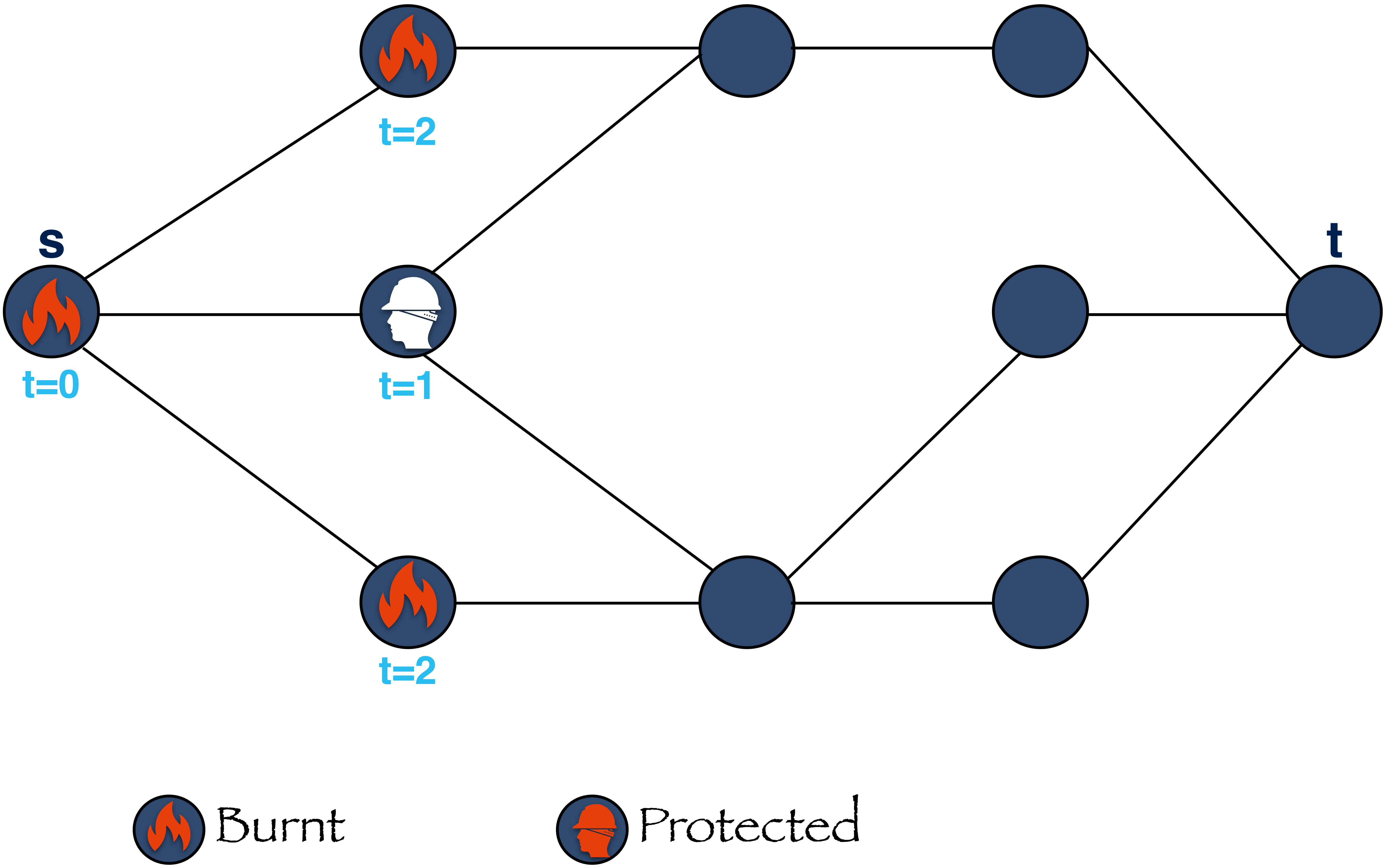


Burnt

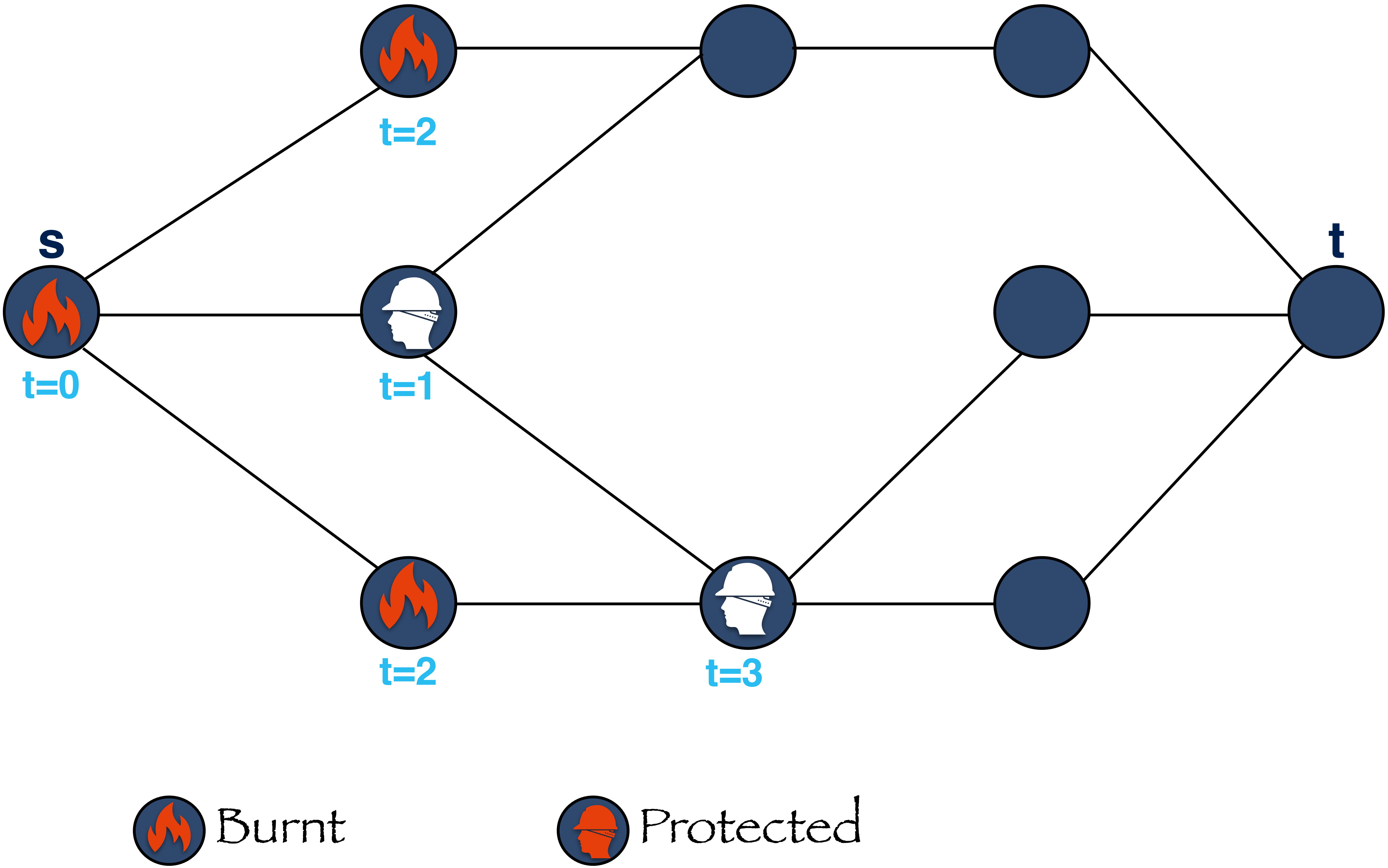
The Firefighting Problem



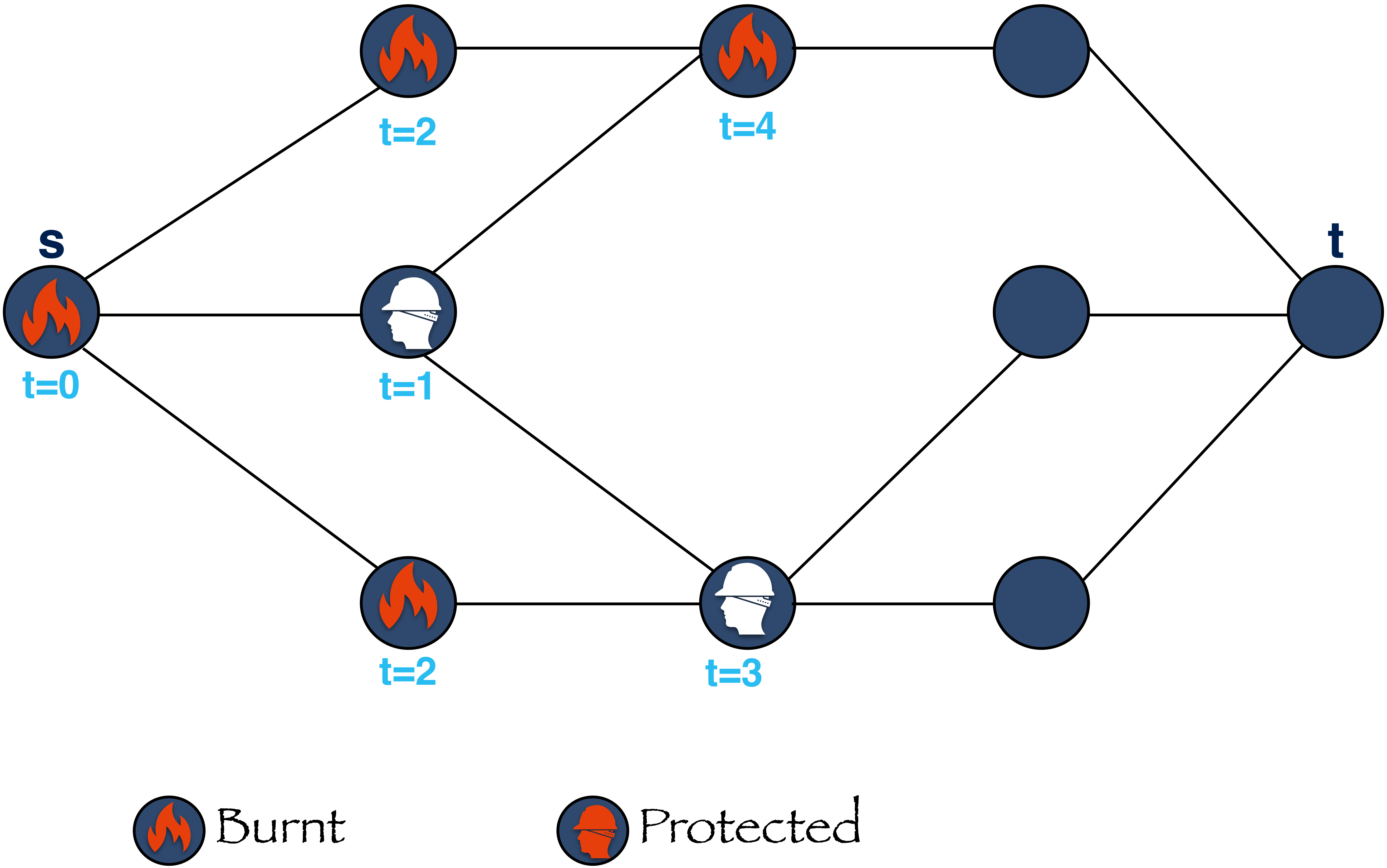
The Firefighting Problem



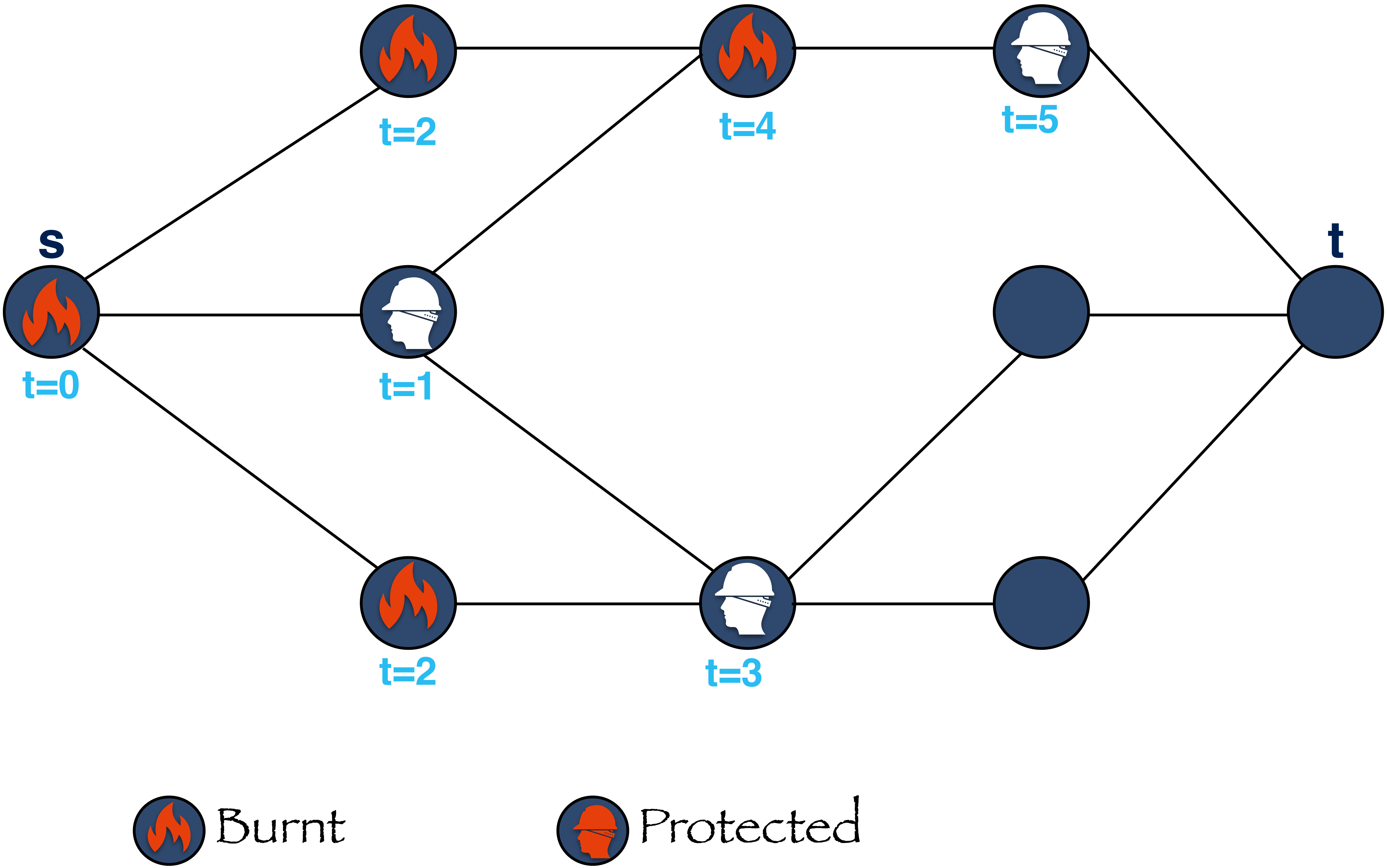
The Firefighting Problem



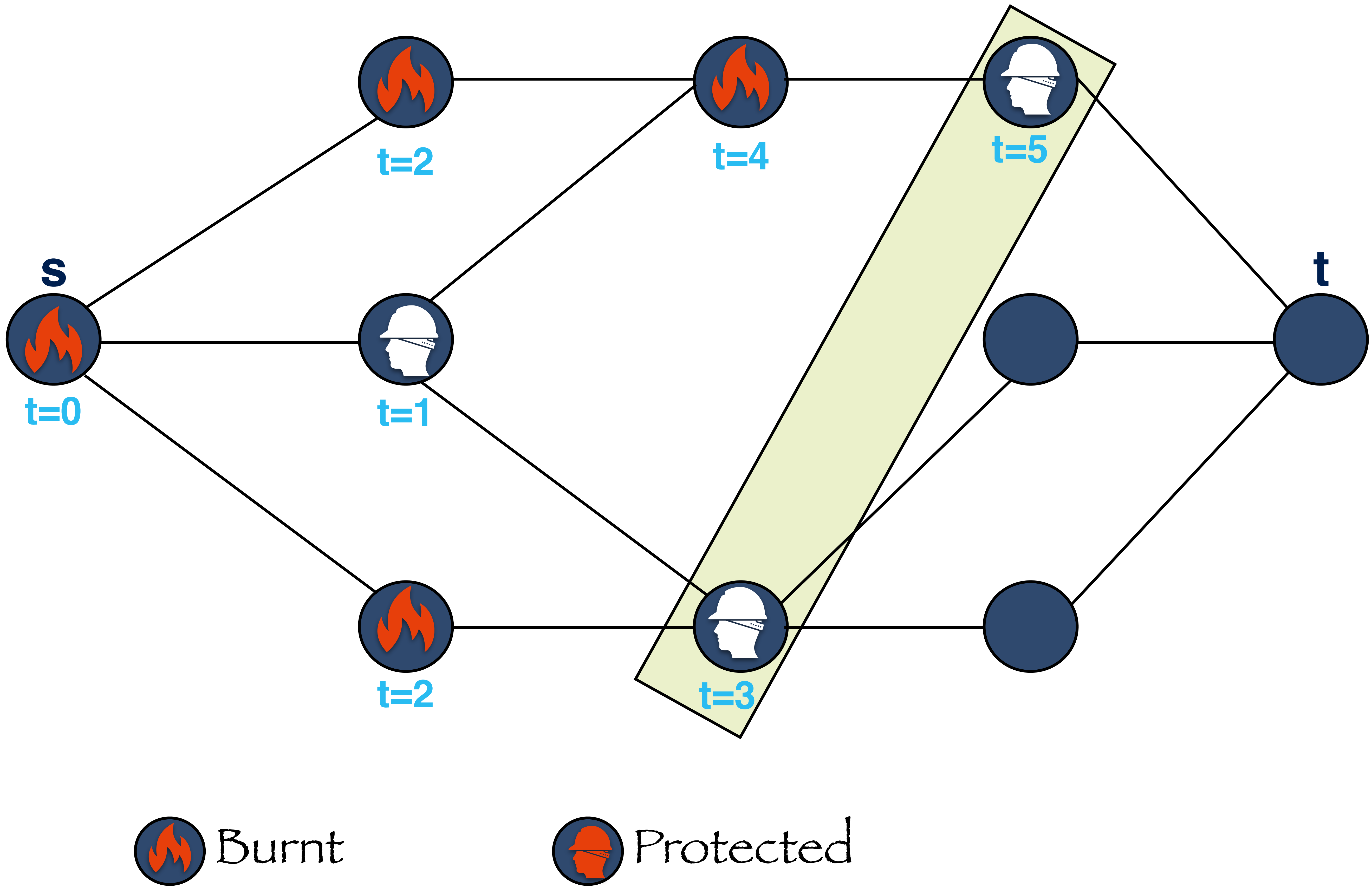
The Firefighting Problem



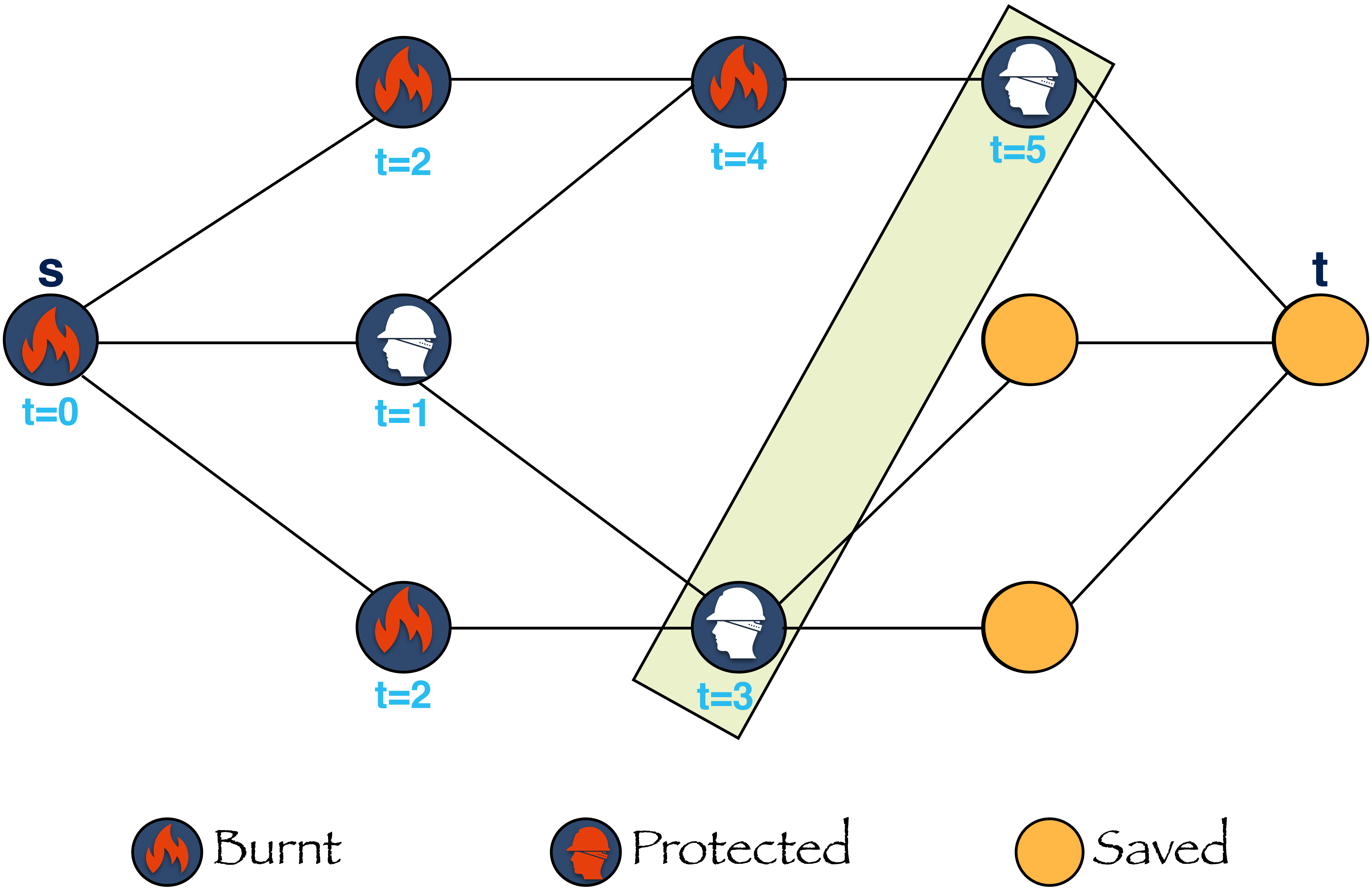
The Firefighting Problem



The Firefighting Problem



The Firefighting Problem



Objectives of the Firefighting Problem

- Maximising the number of saved vertices [Cai, Verbin, and Yang, 08]
- Minimising the number of burned vertices [Cai, Verbin, and Yang, 08, Finbow, Hartnell, et. al., 09]
- Minimising the number of rounds, minimising the number of firefighters per round [Anshelevich, Chakrabarty, et. al., 09]
- ***Saving a specific set of vertices [King, MacGillivray, 09]***

Saving a Critical Set (SACS)

SACS:
Input: An undirected n-vertex graph G, a vertex s, a subset $C \subseteq V(G) \setminus \{s\}$, and an integer k.
Question: Is there a valid k-step strategy that saves C when a fire breaks out at s?

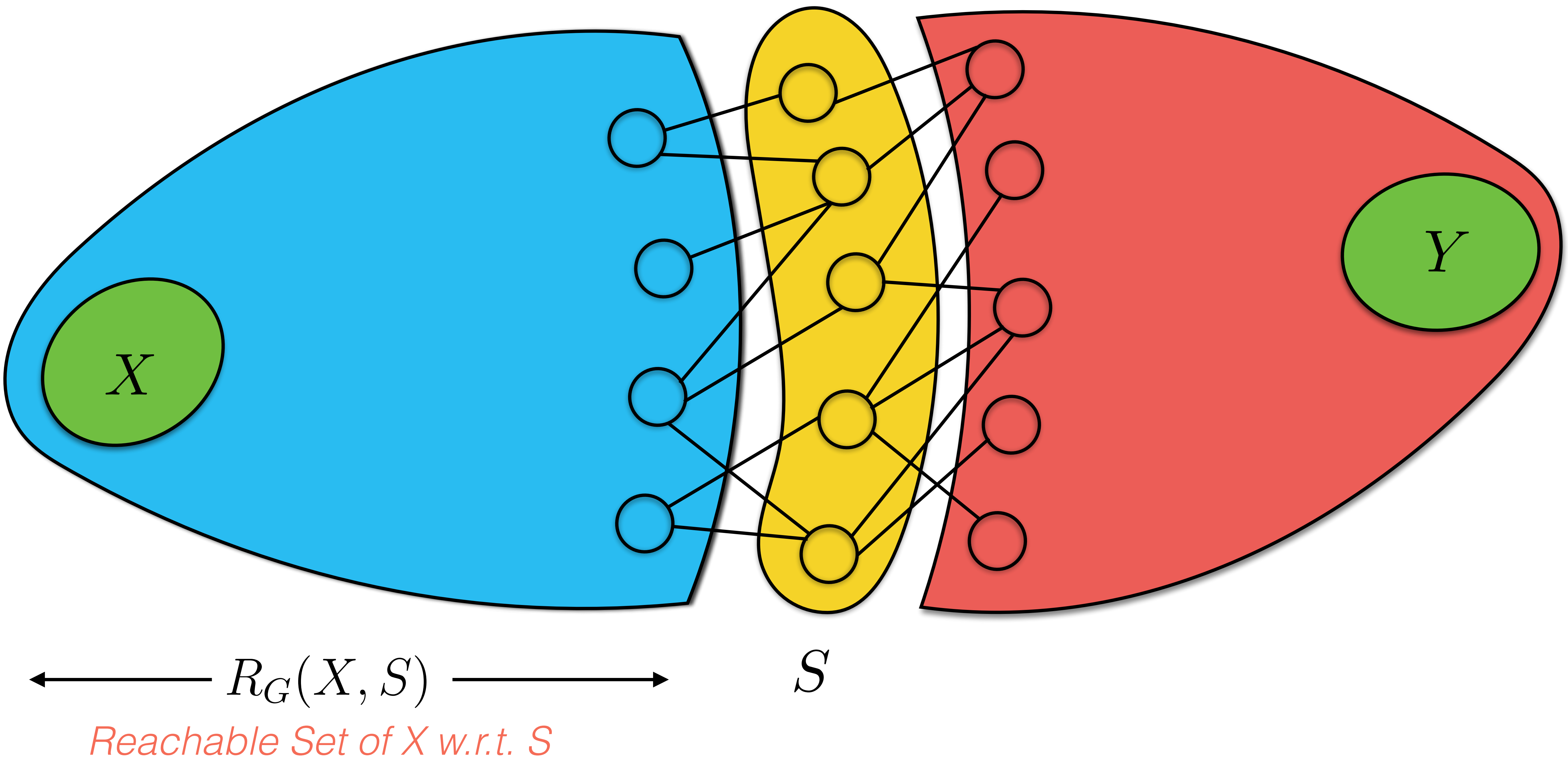
Basic Definitions

Fixed-Parameter Tractability

Definition: *A parameterization of a decision problem is a function that assigns an integer parameter k to each input instance I.*

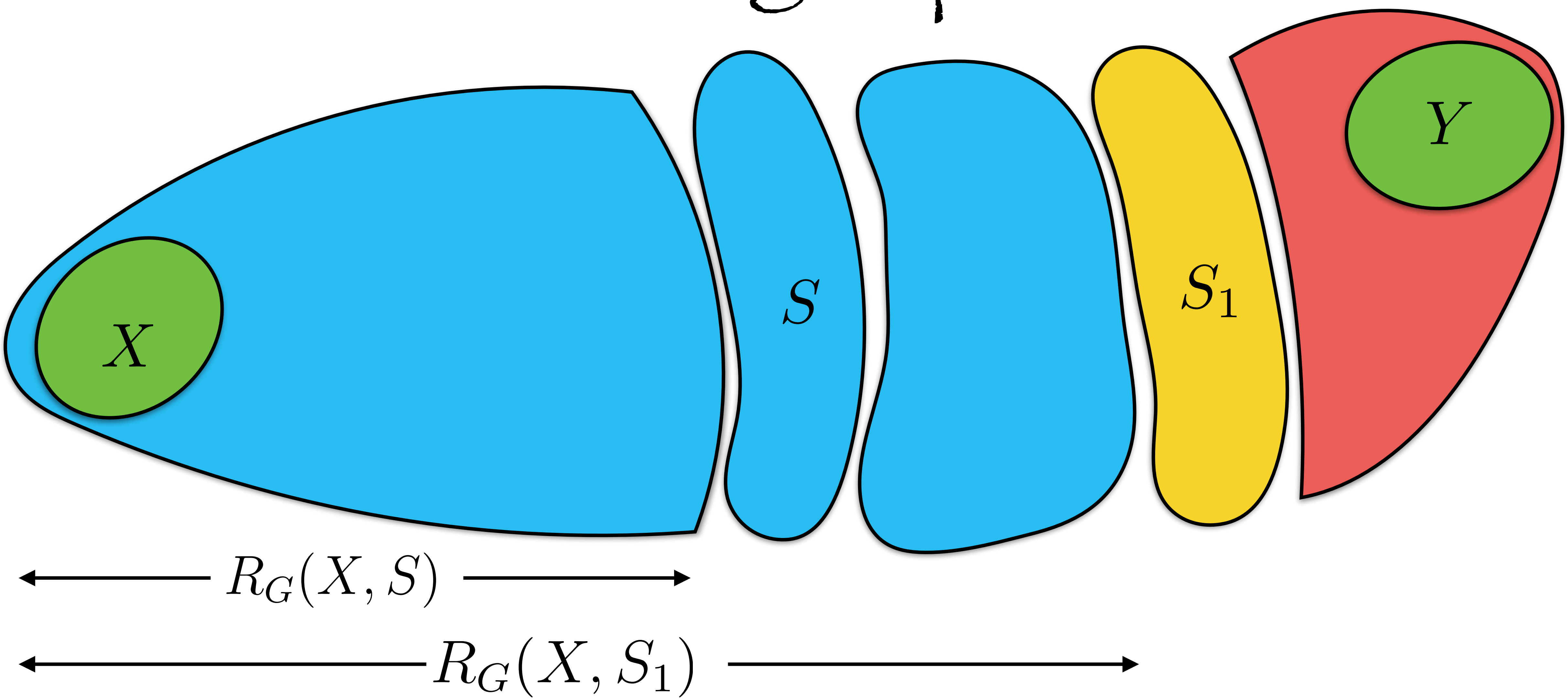
Definition: *A parameterized problem is fixed-parameter tractable (FPT) if there is an $f(k)n^c$ time algorithm for some constant c.*

Separators



A subset $S \subseteq V(G) \setminus (X \cup Y)$ is said to be a separator if $R_G(X, S) \cap Y = \phi$ or in other words there is no path from X to Y in $G \setminus S$

Dominating Separators



A separator S_1 is said to be dominating w.r.t separator S

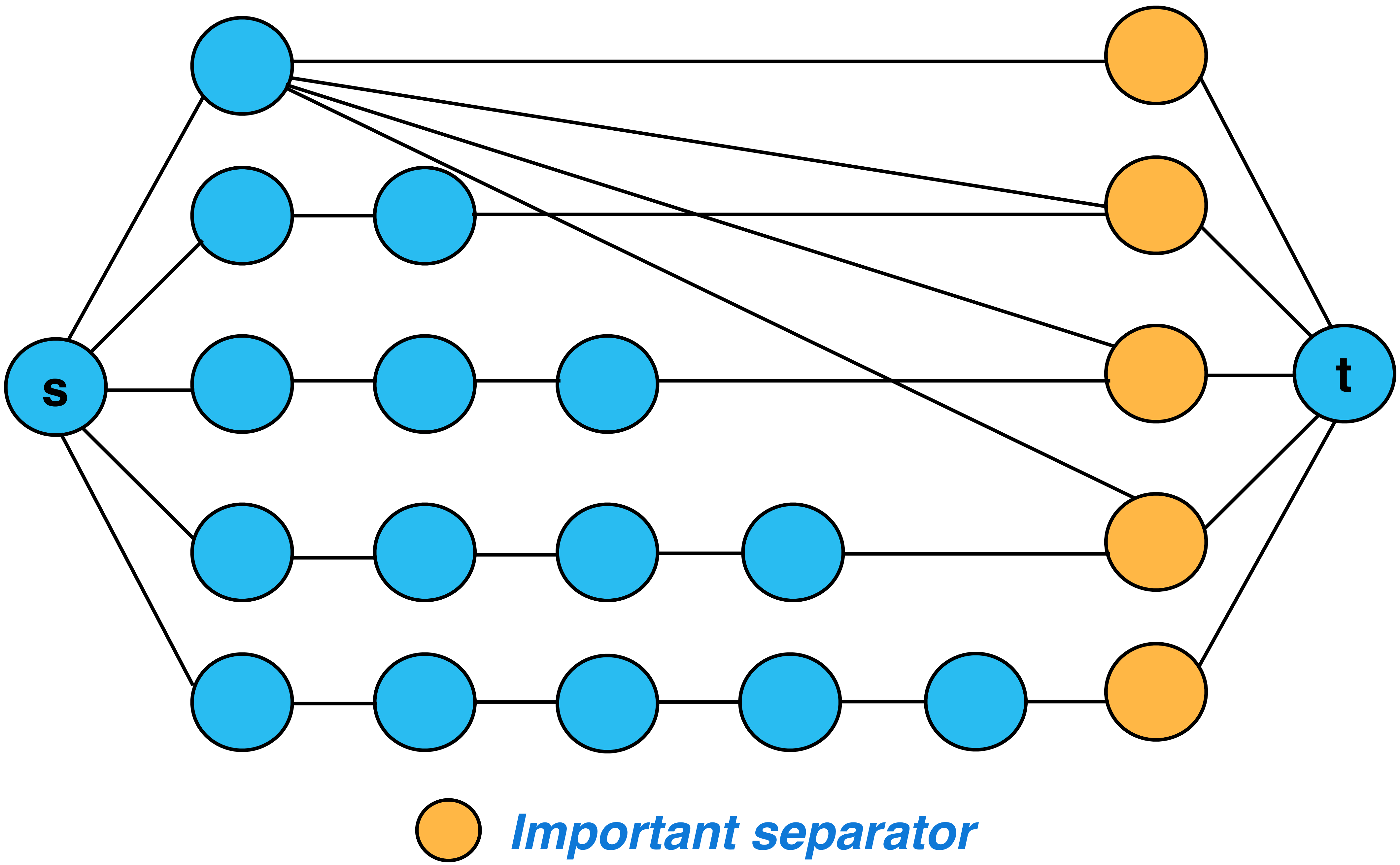
- $|S_1| \leq |S|$
- $R_G(X, S) \subseteq R_G(X, S_1)$

Important Separators

Important separators are those which are not dominated by any other separator

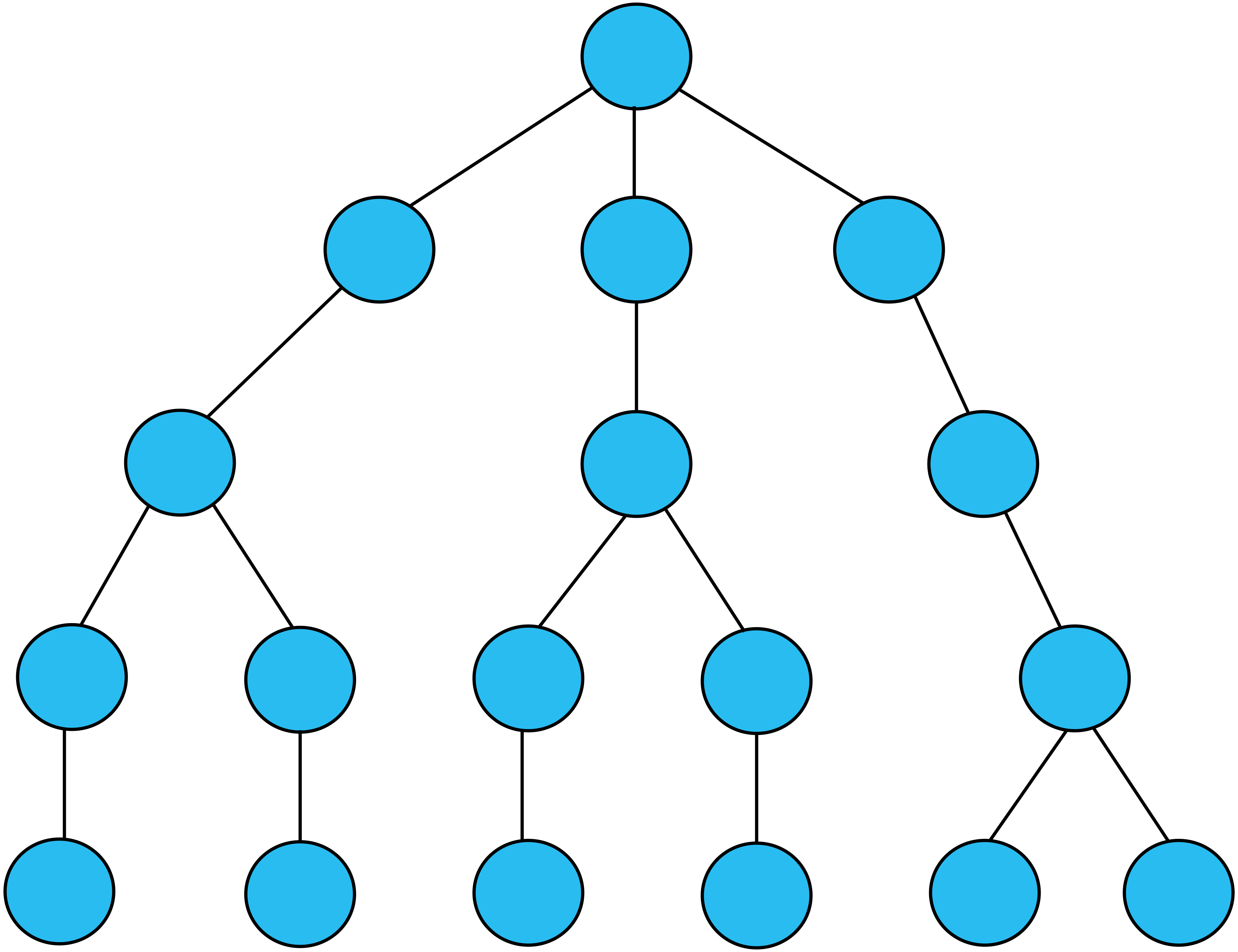
Important Separators

Important separators are those which are not dominated by any other separator

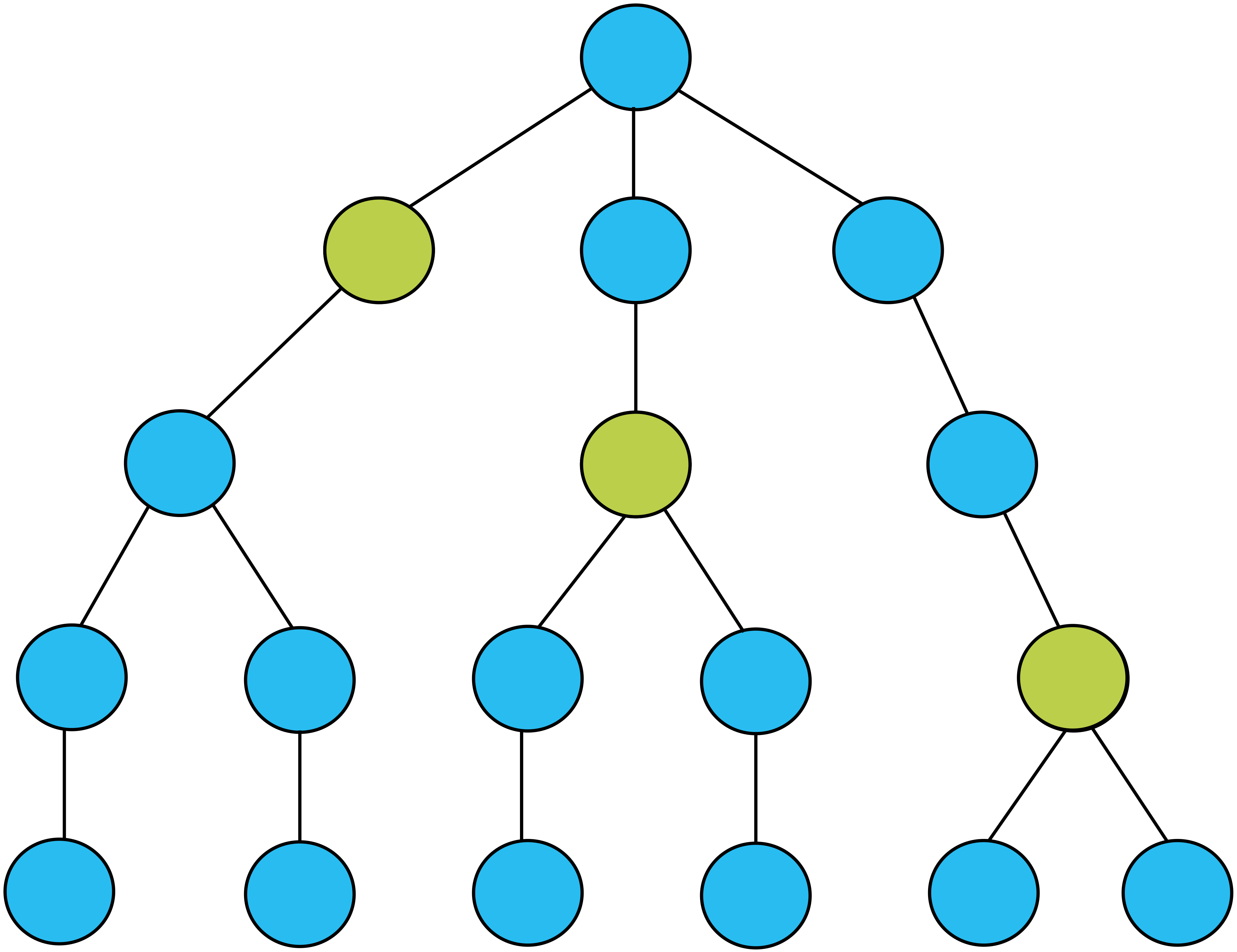


Firefighting on Trees

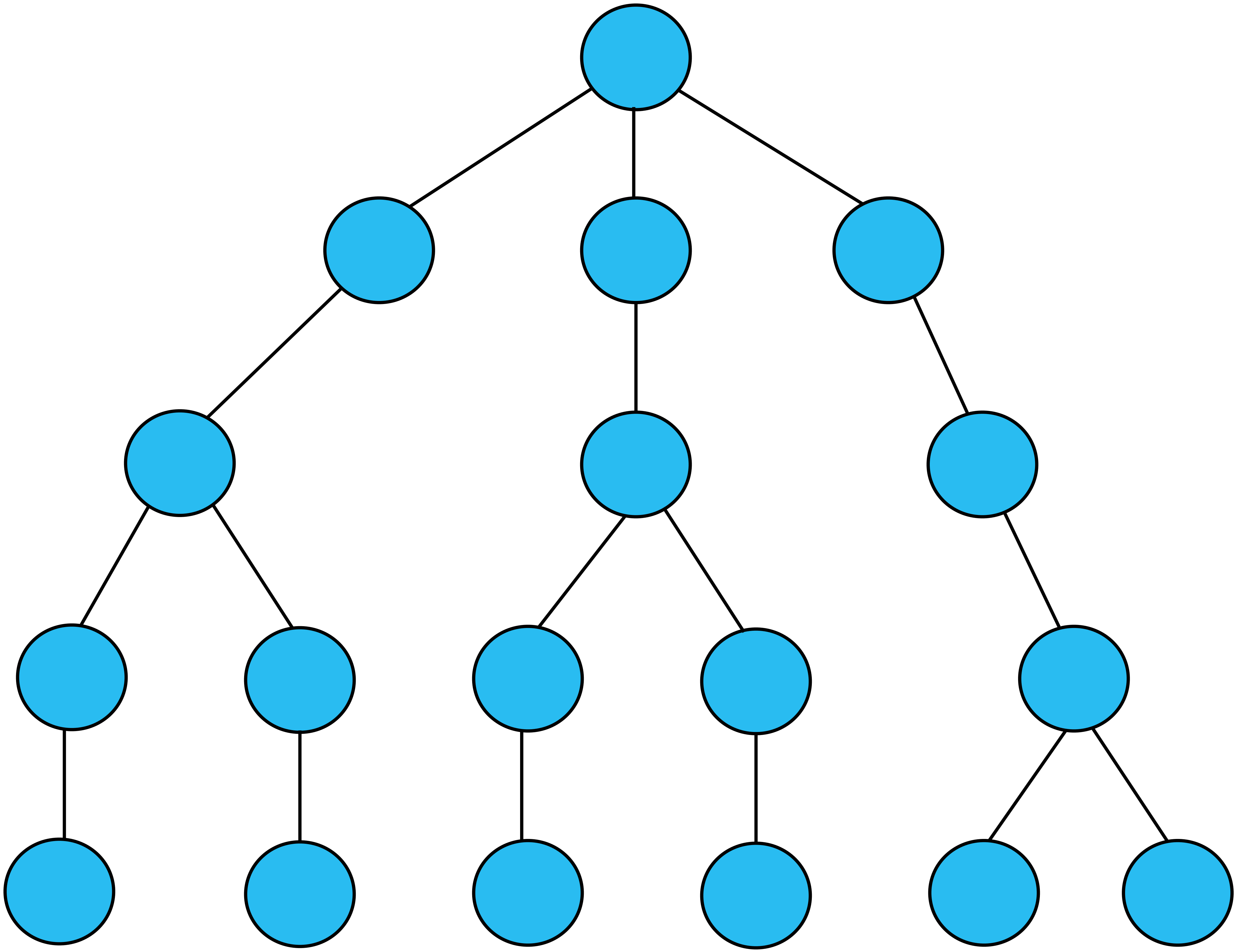
Firefighting on Trees



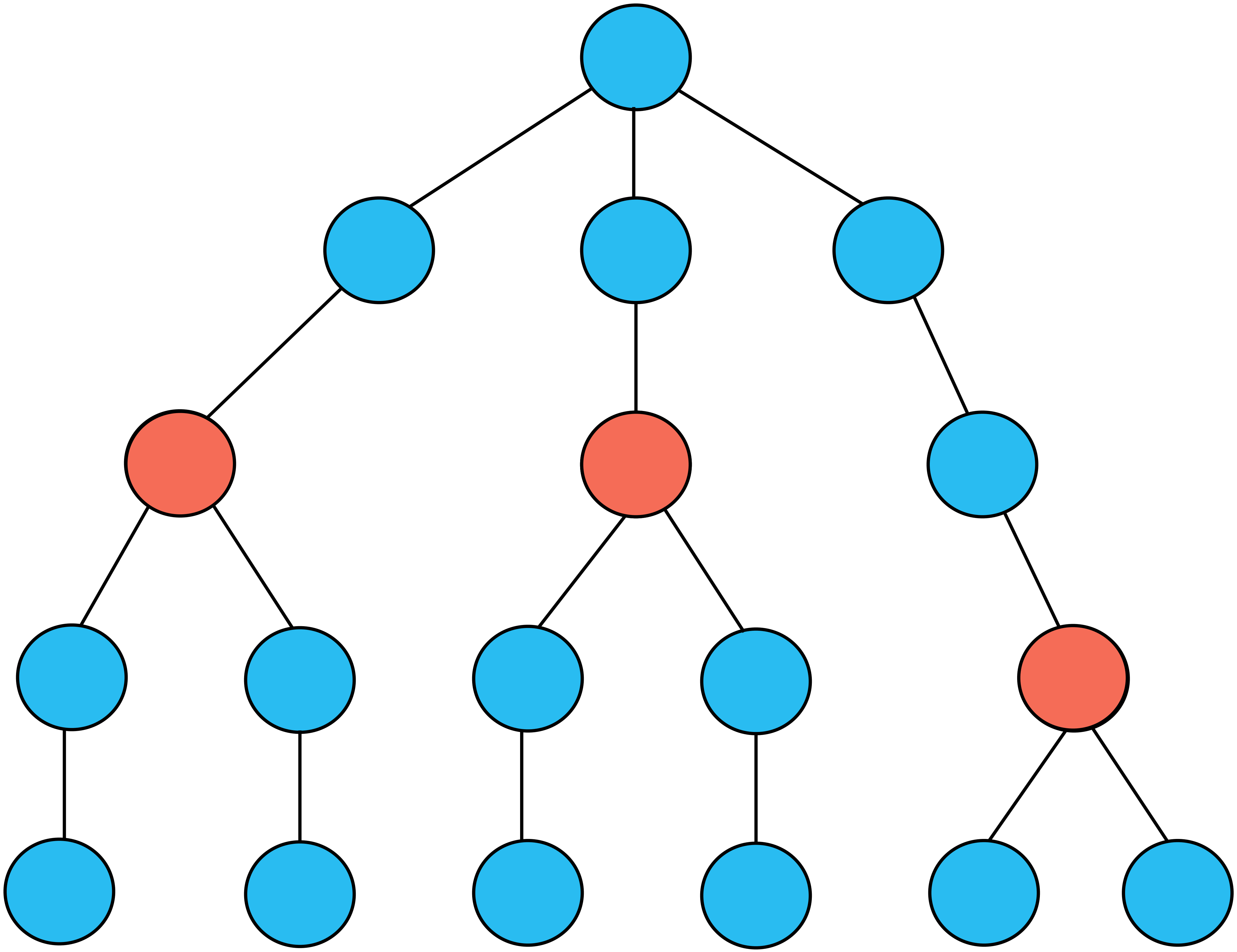
Firefighting on Trees



Firefighting with Important Separators



Firefighting with Important Separators



Firefighting with Important Separators

Theorem: (Marx, 2011)

For trees, there are at most 4^k important separators of size at most k.

Firefighting with Important Separators

Theorem: (Marx, 2011)

For trees, there are at most 4^k important separators of size at most k.

$SACS$ on trees takes time $O^*(4^k)$

Firefighting on Graphs

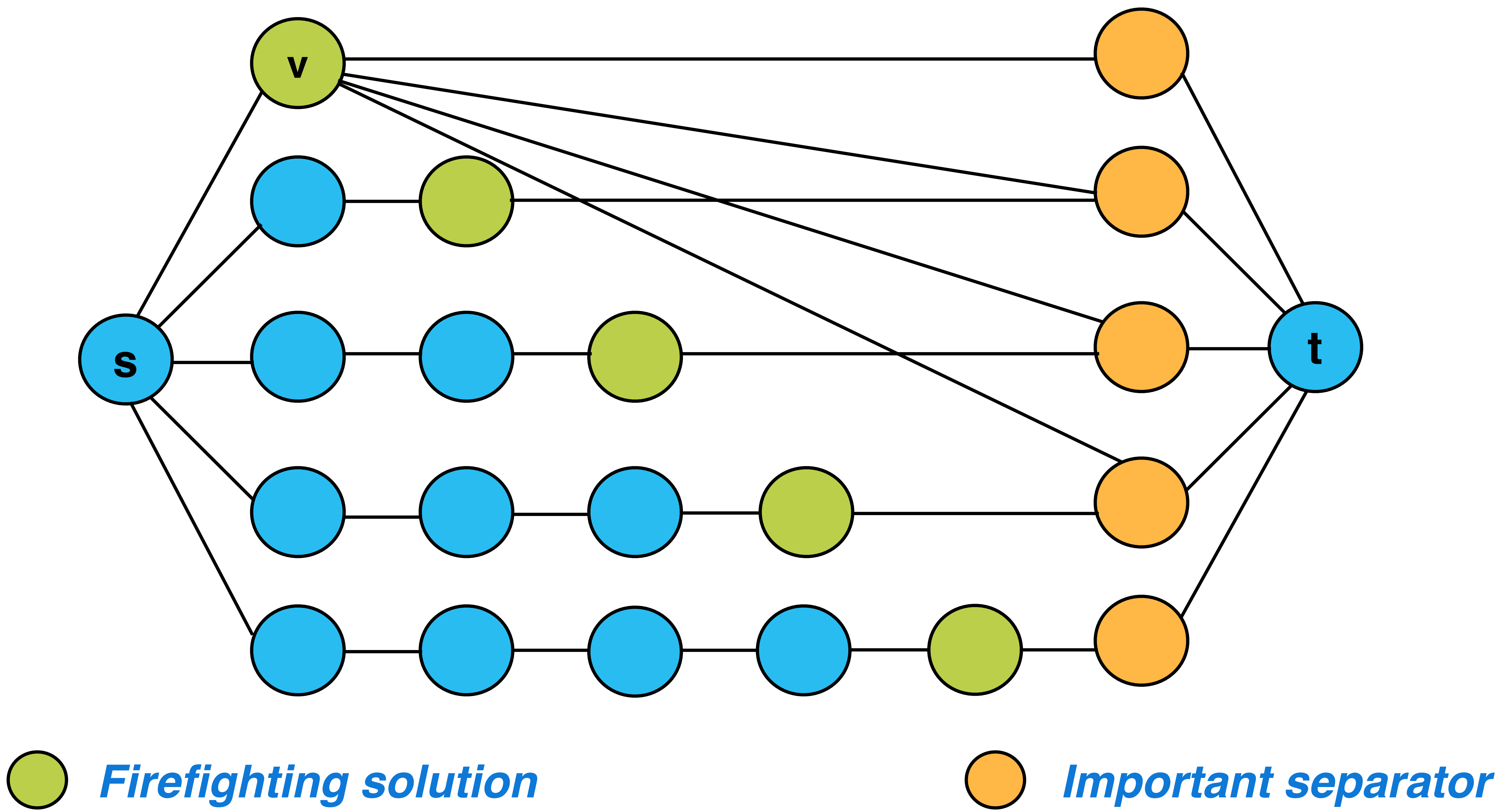
Important Separators

Important Separators

Important separators do not suffice !!!

Important Separators

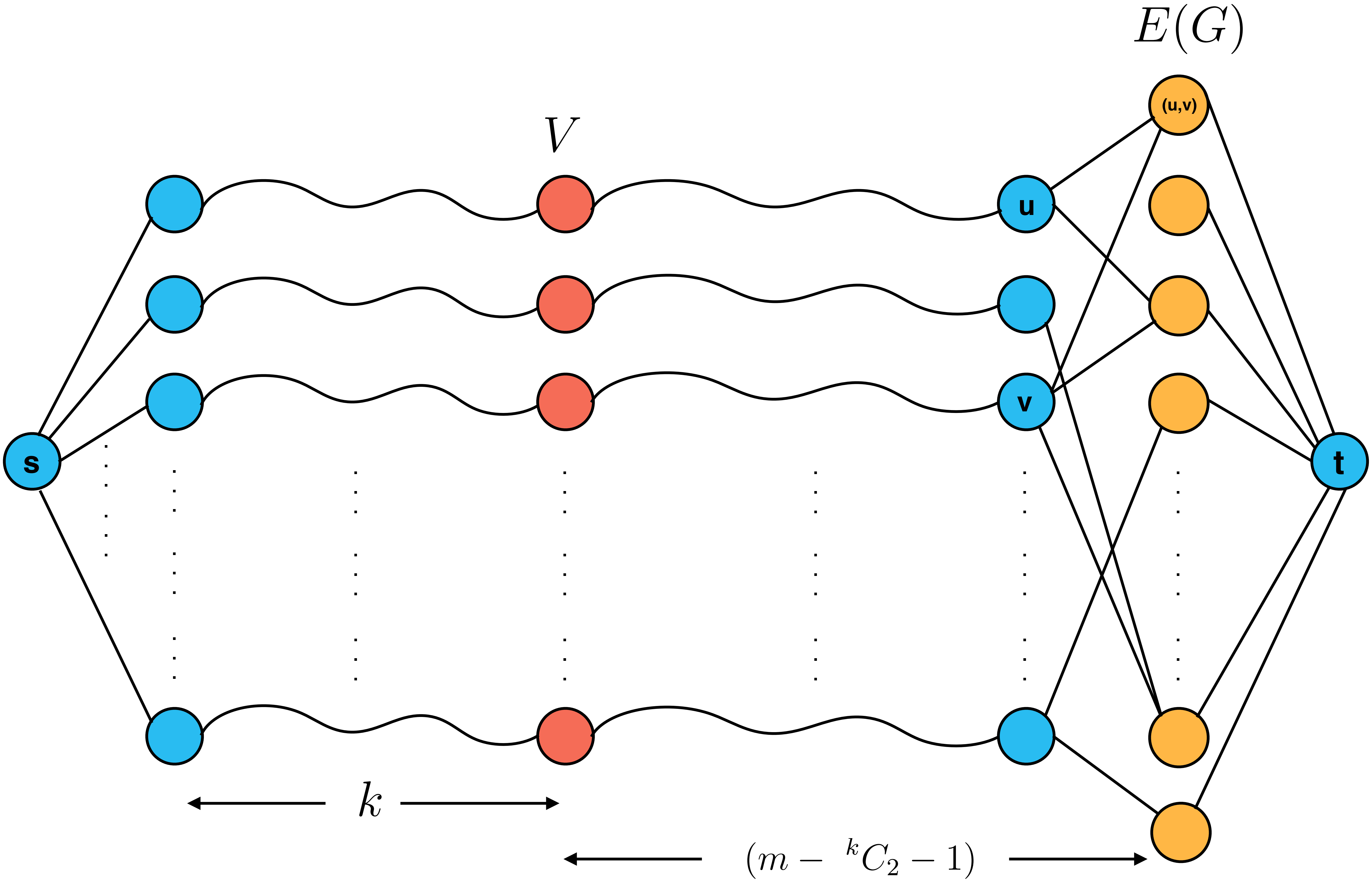
Important separators do not suffice !!!



Saving a Critical Set - NPC

Saving A Critical Set ***(SACS)*** with **critical set of size 1** is a **YES-instance**
if and only if
k-CLIQUE is an **YES-instance**

Saving a Critical Set - NPC



Saving a Critical Set - NPC

SACS with size 1 has a successful strategy with $(k + m - \ {}^{k}C_2)$ firefighters in this new graph G' if and only if G has a clique of size k.

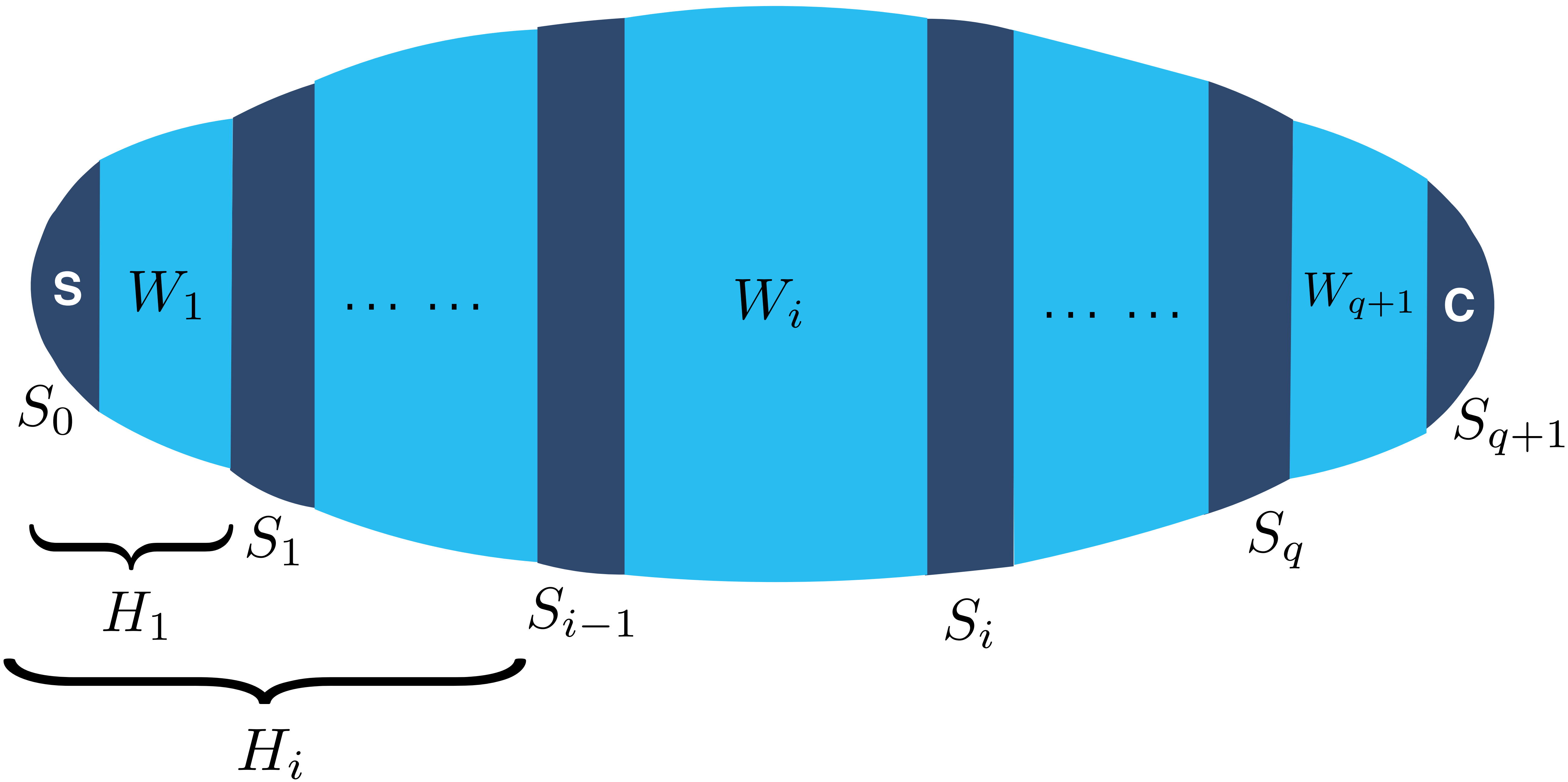
Tight Separator Sequence

Let X, Y be two subset of vertices in graph G. Then, a tight (X, Y)-reachability sequence of order k is an ordered collection $H = \{H_1, H_2, ..., H_q\}$ of sets in $V(G)$ satisfying the following properties:

1. $H_1 \subset H_2 \subset \cdots \subset H_q$,
2. $|N(H_i)| \leq k, \forall i, 1 \leq i \leq q$,
3. $S_i = N(H_i), \forall 1 \leq i \leq q$ is a minimal (X, Y)-separator in G

[M. S. Ramanujan, 13]

Tight Separator Sequence



There is an algorithm that runs in time $O(kmn^2)$ that either correctly concludes that there is no $X - Y$ separator of size at most k or outputs the required sequence.

Case-1: q > k

Let $\mathcal{S} = \cup_{i=1}^{q} S_i \qquad \mathcal{W} = \cup_{i=1}^{q+1} W_i$

Case-1: q > k

Let $\mathcal{S} = \cup_{i=1}^{q} S_i$ $\qquad$ $\mathcal{W} = \cup_{i=1}^{q+1} W_i$

Claim: If G admits a tight (s, C)-separator sequence of order q in $G \setminus Y$ where $q > k$, then there exists a k-step firefighting strategy.

Case-1: q > k

Let $\mathcal{S} = \cup_{i=1}^{q} S_i$ $\quad$ $\mathcal{W} = \cup_{i=1}^{q+1} W_i$

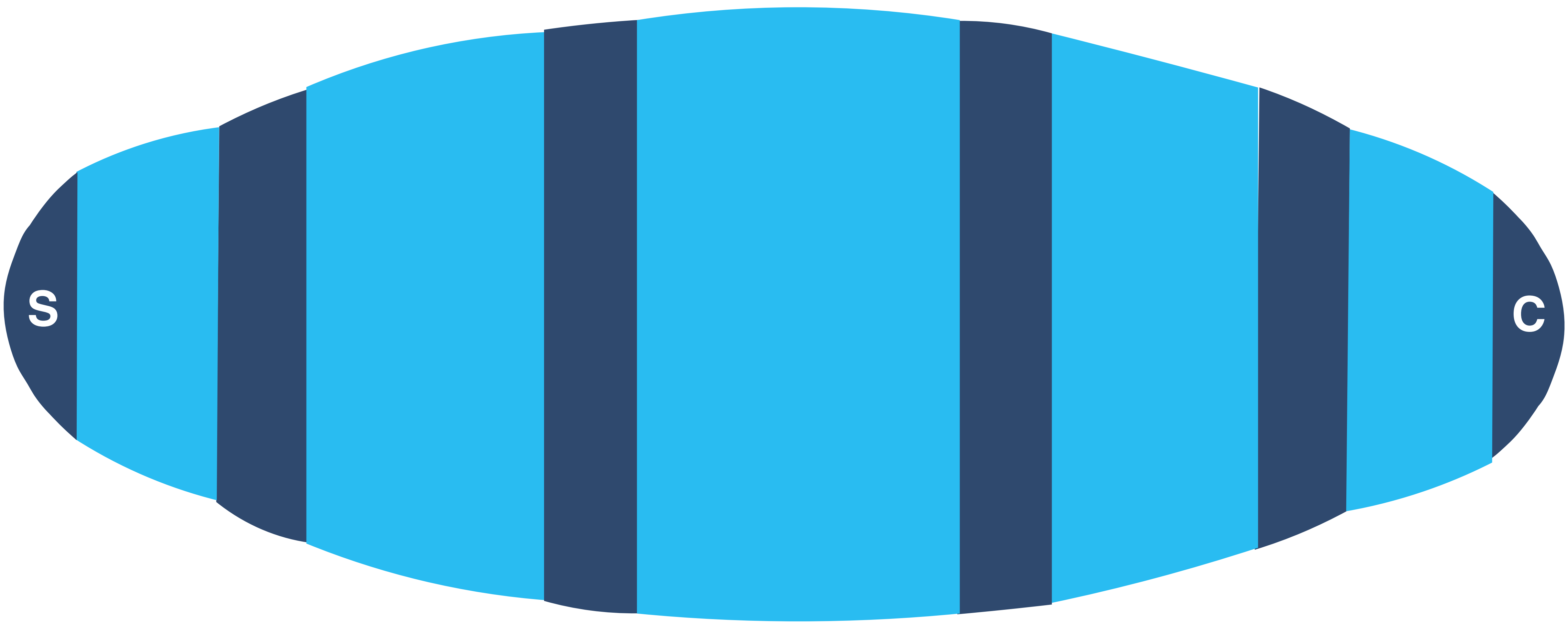
Claim: If G admits a tight (s, C)-separator sequence of order q in $G \setminus Y$ where $q > k$, then there exists a k-step firefighting strategy.

Place the firefighters on the separator S_q

Case-2: $q < k$

Guess the partition of the timestamps P for a firefighting strategy

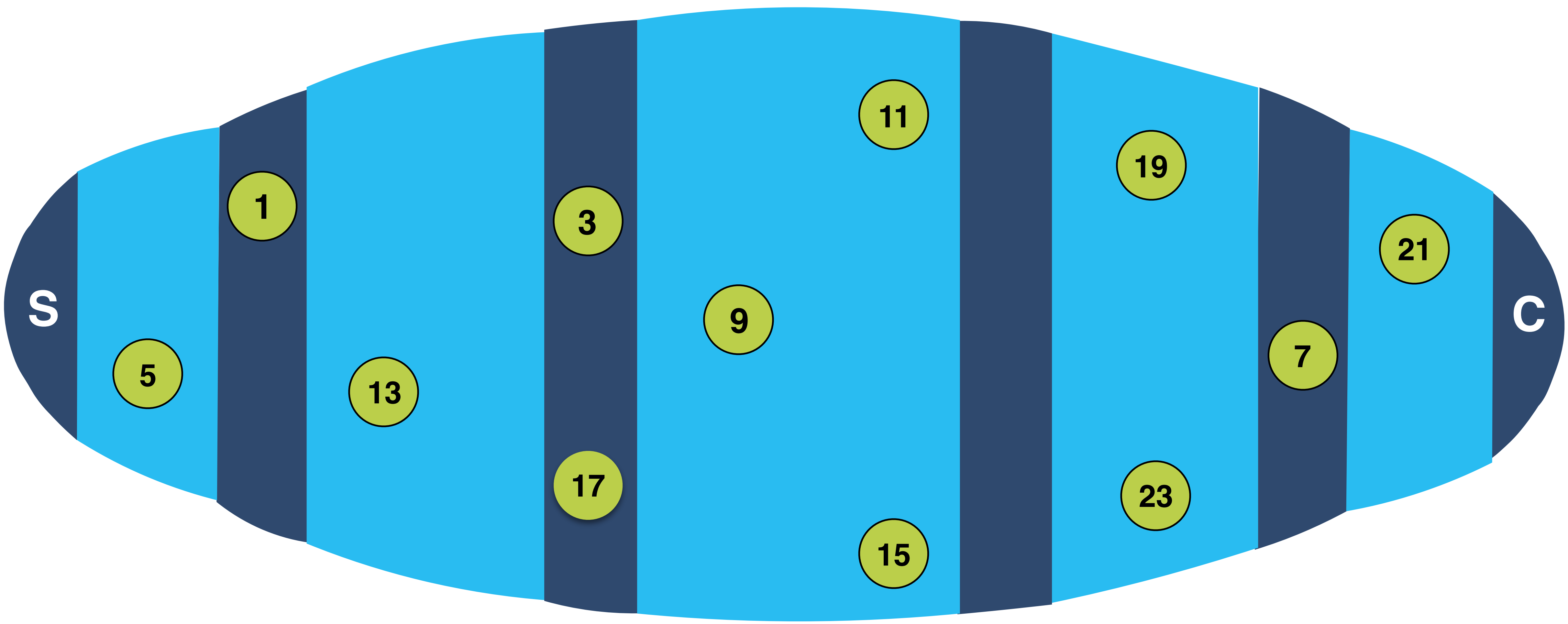
For e.g., $P = \{1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23\}$



Case-2: $q < k$

Guess the partition of the timestamps P for a firefighting strategy

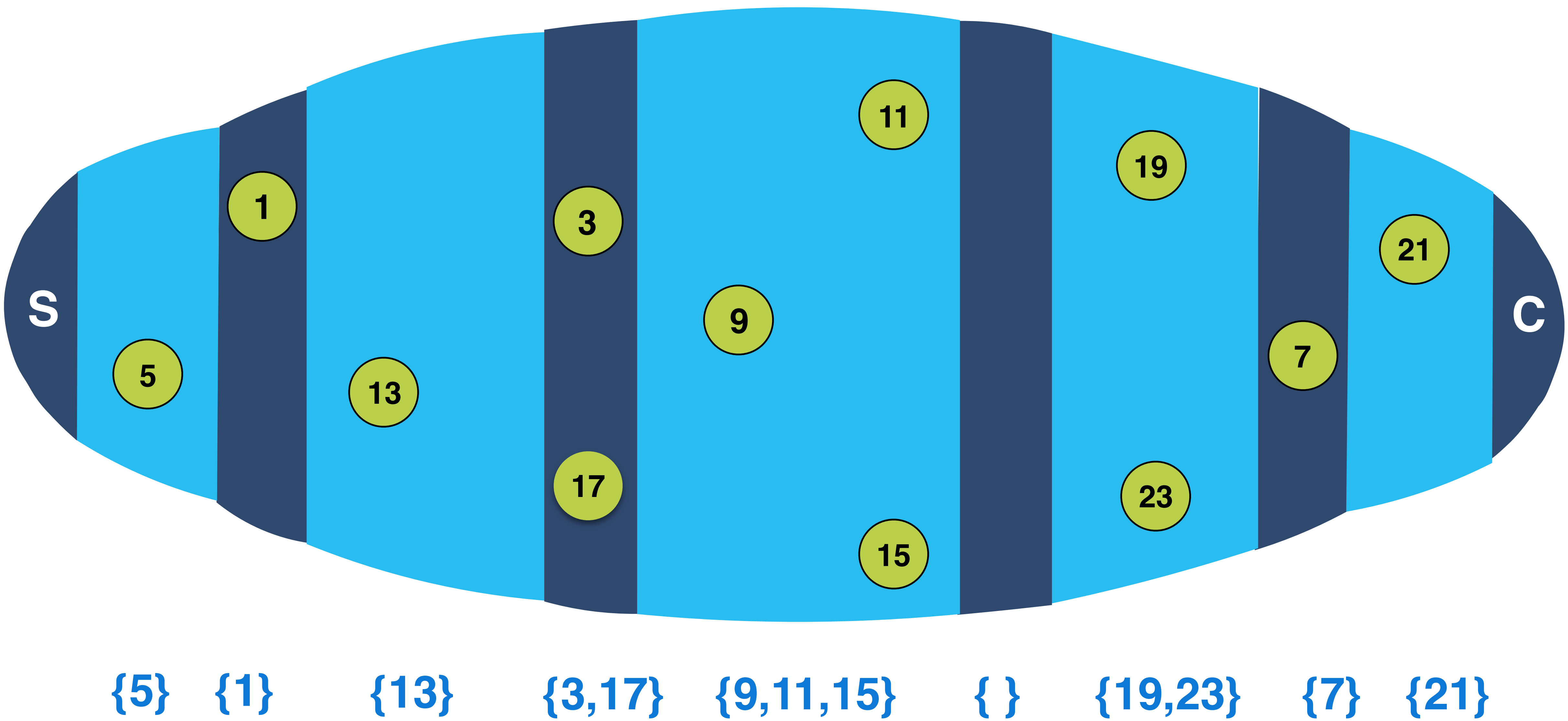
For e.g., $P = \{1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23\}$



Case-2: $q < k$

Guess the partition of the timestamps P for a firefighting strategy

For e.g., $P = \{1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23\}$



Partitioned Timestamps

Let

- $A_1, A_2, \ldots, A_q$ denote the timestamps for the nodes inside $\mathcal{S}$ and
- $B_1, B_2, \ldots, B_{q+1}$ denote the timestamps for the nodes inside $\mathcal{W}$.

$$P = \cup_{i=1}^{q} A_i \cup \cup_{I=1}^{q+1} B_i$$

$$|P| = p$$

Partitioned Timestamps

Let

- $A_1, A_2, \ldots, A_q$ denote the timestamps for the nodes inside $\mathcal{S}$ and
- $B_1, B_2, \ldots, B_{q+1}$ denote the timestamps for the nodes inside $\mathcal{W}$.

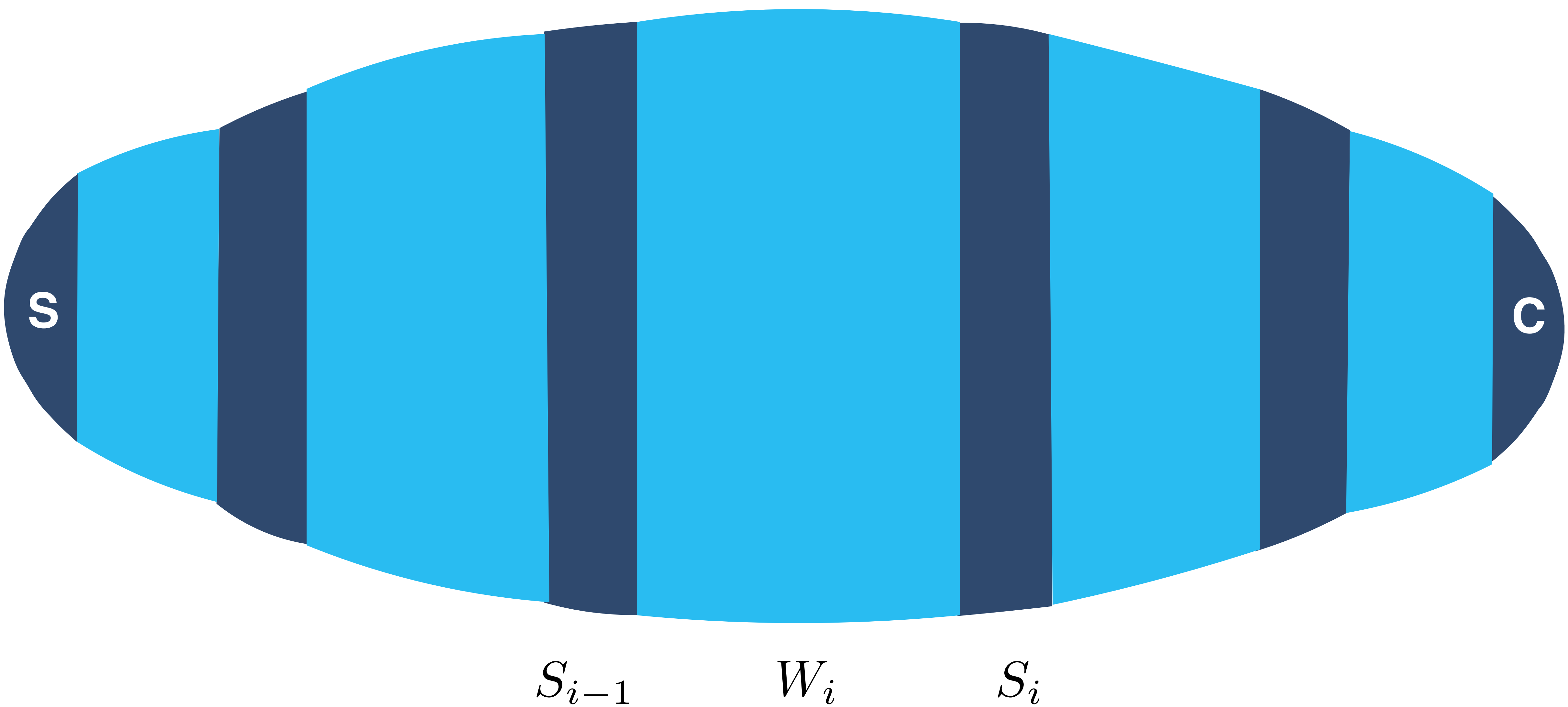
$$P = \cup_{i=1}^{q} A_i \cup \cup_{I=1}^{q+1} B_i$$

$$|P| = p$$

The number of possible partitions $= (2q+1)^p \leq (2k+1)^k$

Possible Labelings

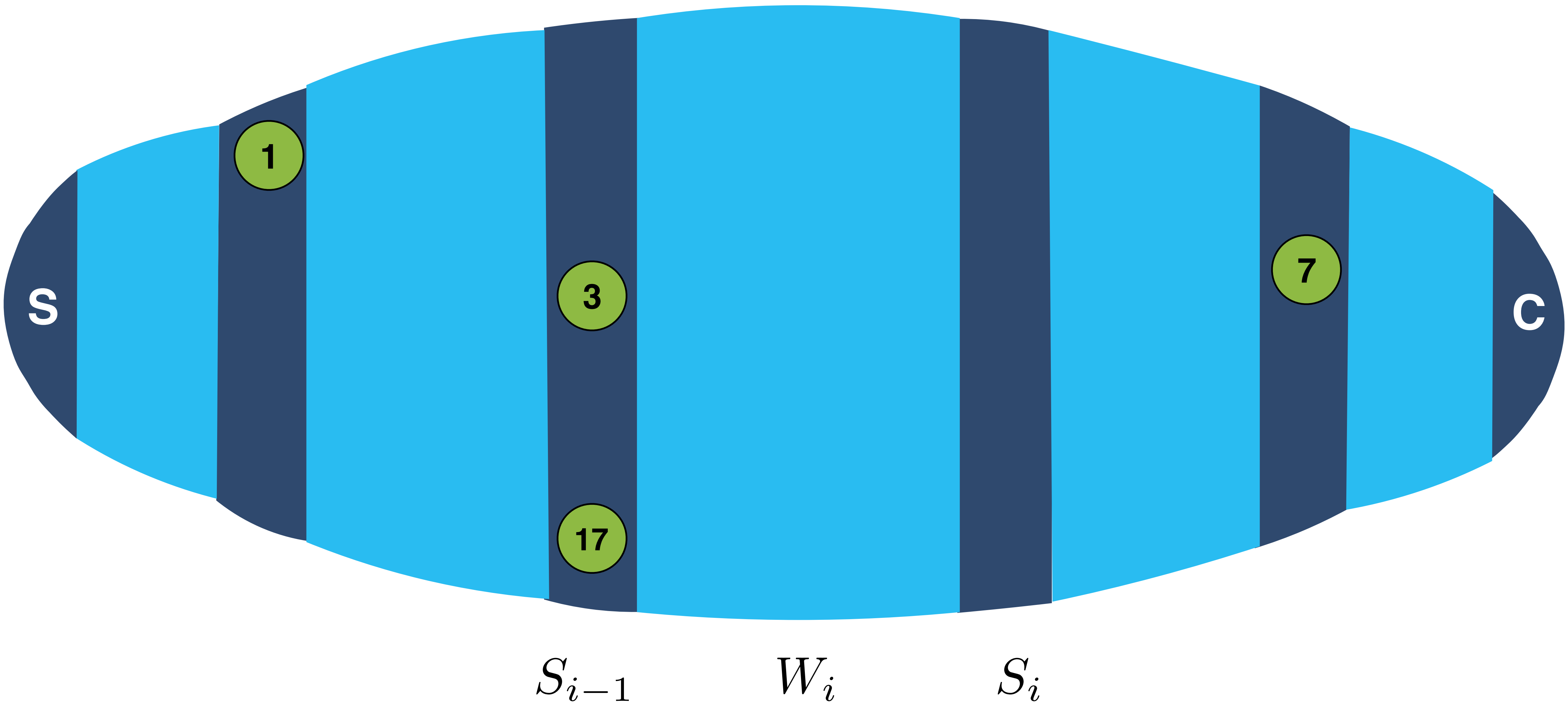
Guess the behaviour of the strategy restricted to $\mathcal{S} = \cup_{i=1}^{q} S_i$



$$G_i = G[S_{i-1} \cup W_i \cup S_i]$$

Possible Labelings

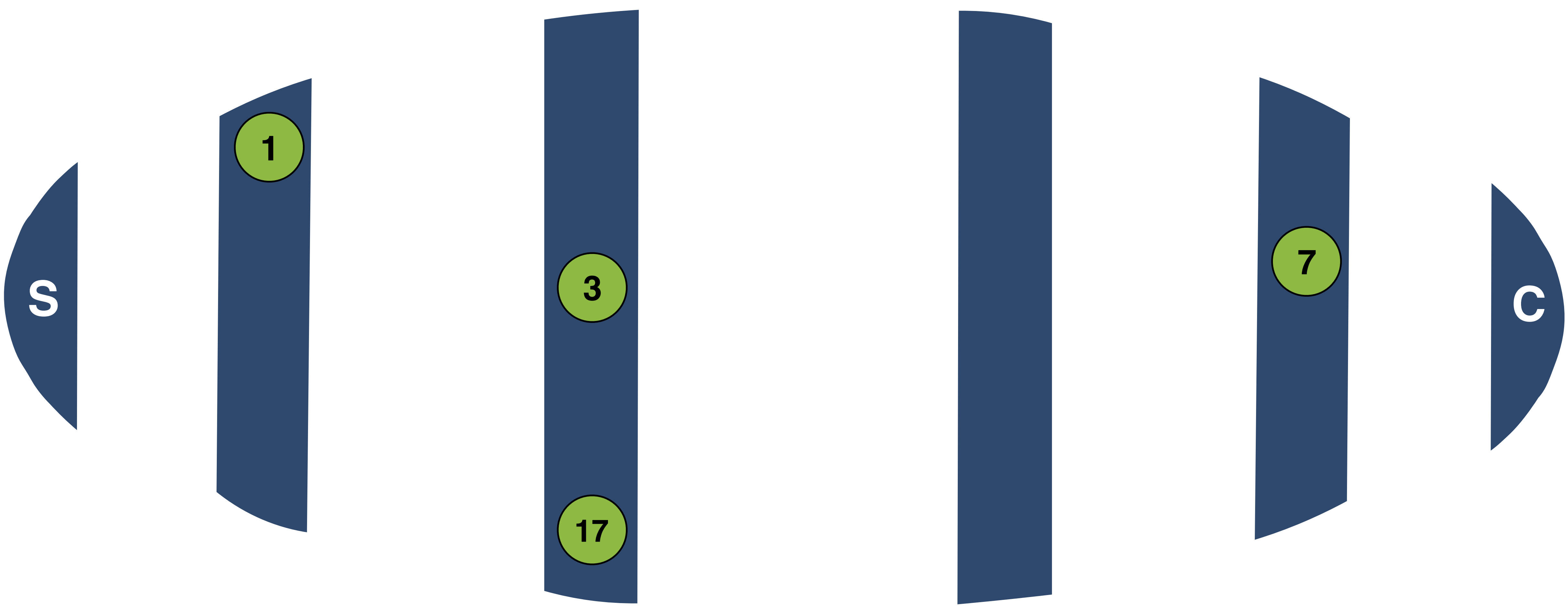
Guess the behaviour of the strategy restricted to $\mathcal{S} = \cup_{i=1}^{q} S_i$



$$G_i = G[S_{i-1} \cup W_i \cup S_i]$$

Possible Labelings

Guess the behaviour of the strategy restricted to $\mathcal{S} = \cup_{i=1}^{q} S_i$

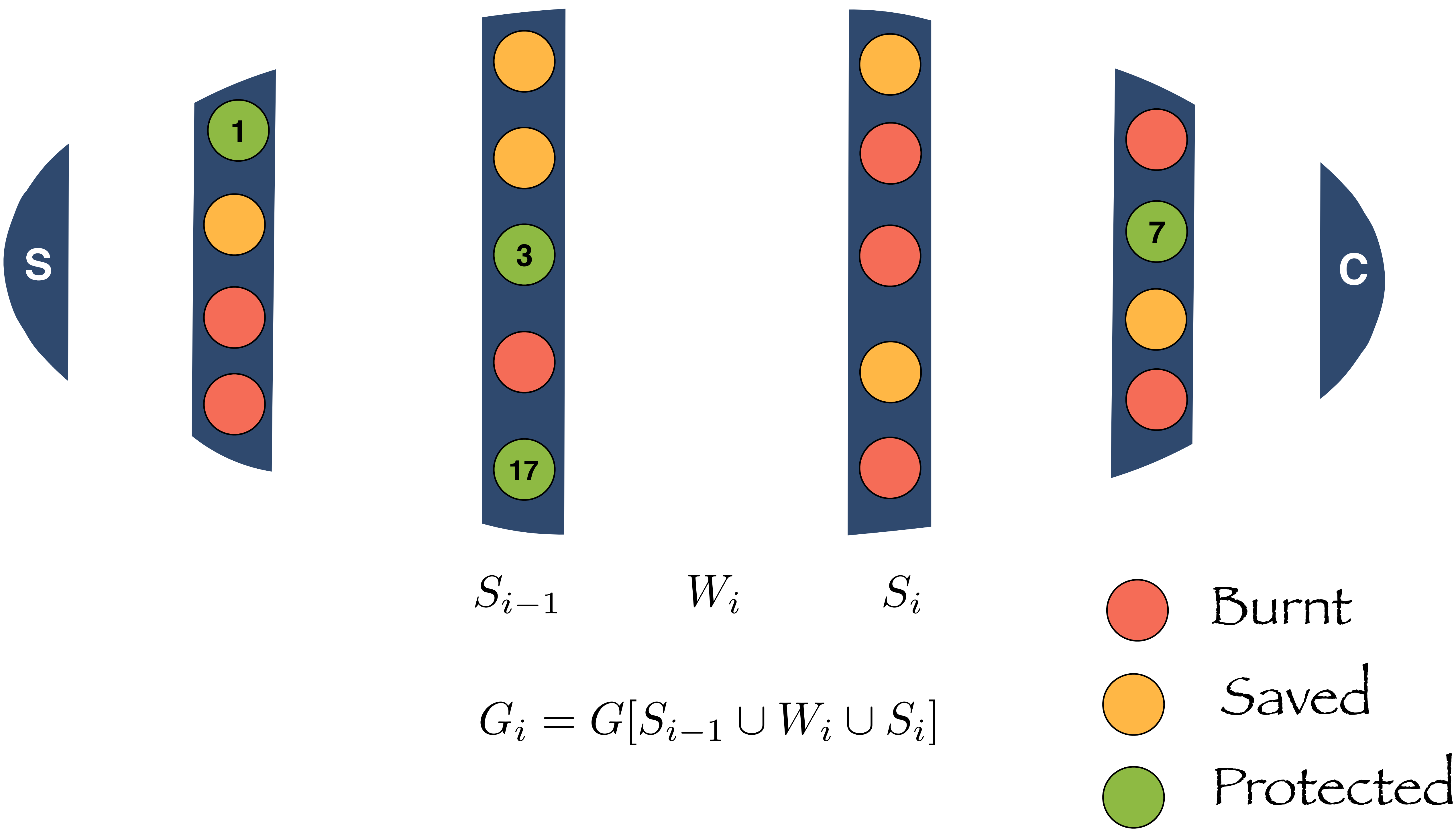


S_{i-1} W_i S_i

$$G_i = G[S_{i-1} \cup W_i \cup S_i]$$

Possible Labelings

Guess the behaviour of the strategy restricted to $\mathcal{S} = \cup_{i=1}^{q} S_i$



$G_i = G[S_{i-1} \cup W_i \cup S_i]$

Possible Labelings

$$\mathfrak{L} = (\{\mathfrak{f}\} \times X) \cup (\{\mathfrak{b}\} \times [2k]_E) \cup \{\mathfrak{p}\}$$

Possible Labelings

$$\mathfrak{L} = (\{\mathfrak{f}\} \times X) \cup (\{\mathfrak{b}\} \times [2k]_E) \cup \{\mathfrak{p}\}$$

$$\mathfrak{L}_{\mathfrak{h}}(v) = \begin{cases} (\mathfrak{f}, t) & \text{if } \mathfrak{h}(t) = v, \\ (\mathfrak{b}, t) & \text{if } t \text{ is the earliest timestep at which } v \text{ burns,} \\ \mathfrak{p} & \text{if } v \text{ is not reachable from } s \text{ in } G \setminus (\{\mathfrak{h}(i) \mid i \in [2k]_O\}) \end{cases}$$

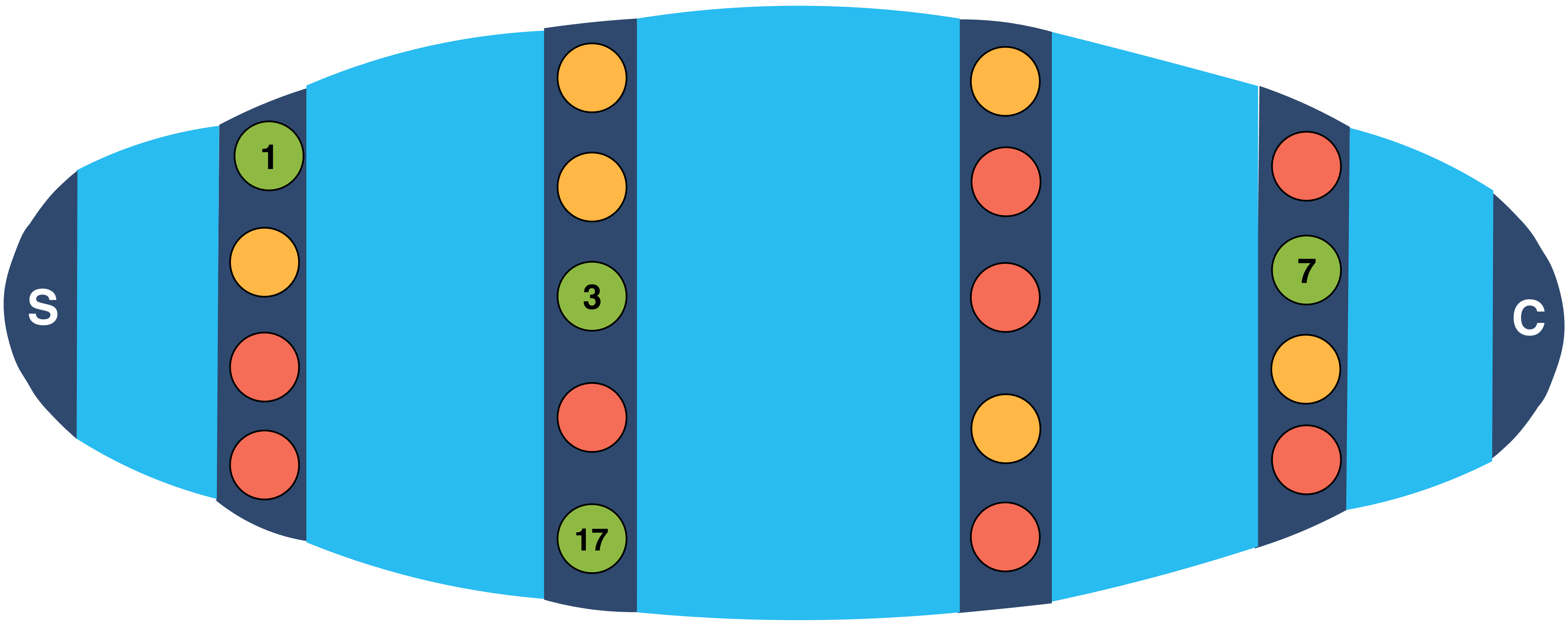
Possible Labelings

$$\mathfrak{L} = (\{\mathfrak{f}\} \times X) \cup (\{\mathfrak{b}\} \times [2k]_E) \cup \{\mathfrak{p}\}$$

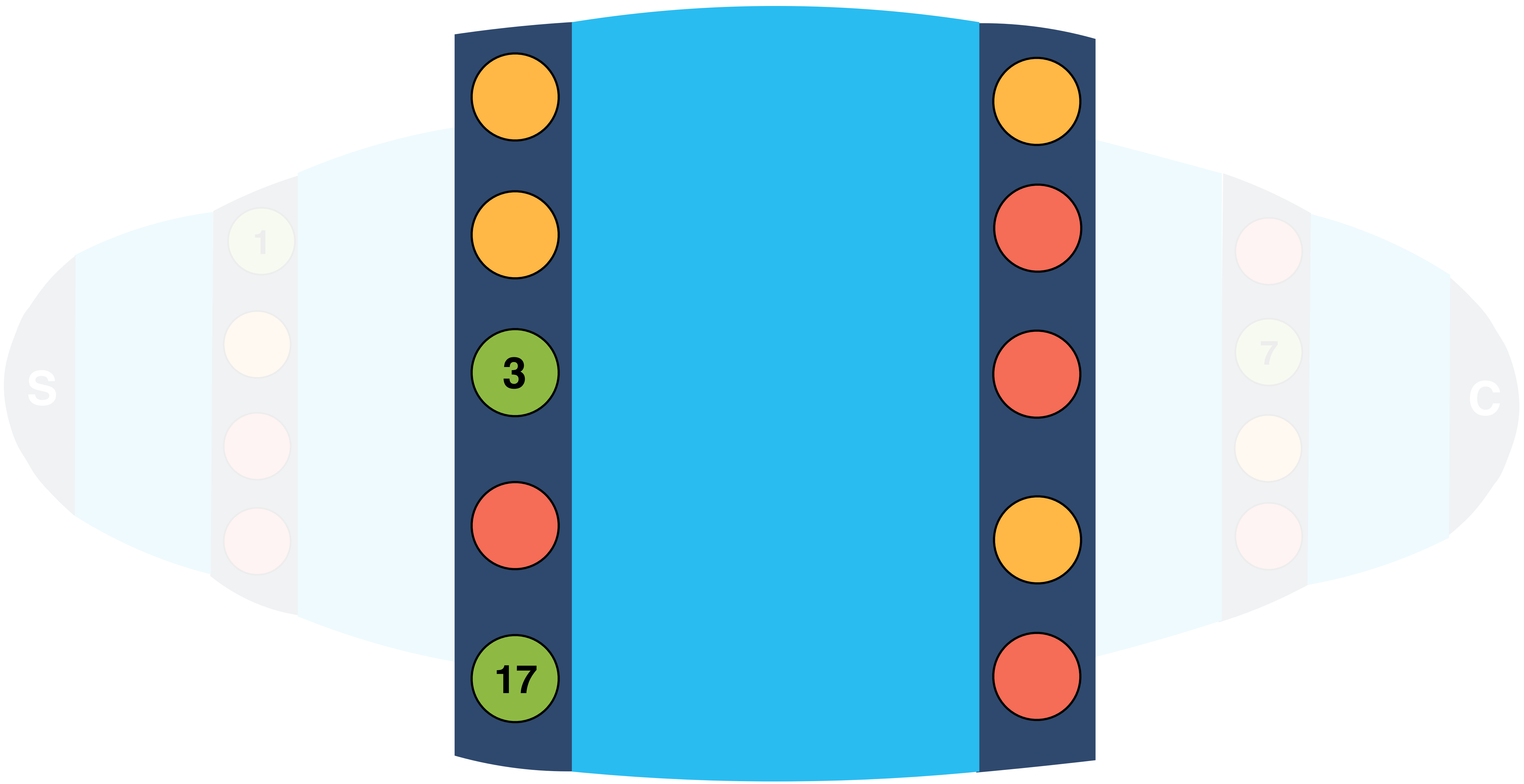
$$\mathfrak{L}_{\mathfrak{h}}(v) = \begin{cases} (\mathfrak{f}, t) & \text{if } \mathfrak{h}(t) = v, \\ (\mathfrak{b}, t) & \text{if } t \text{ is the earliest timestep at which } v \text{ burns}, \\ \mathfrak{p} & \text{if } v \text{ is not reachable from } s \text{ in } G \setminus (\{\mathfrak{h}(i) \mid i \in [2k]_O\}) \end{cases}$$

The number of possible labelings $= (p + k + 1)^{pk} \leq (3k)^{k^2} \leq k^{(O(k^2))}$

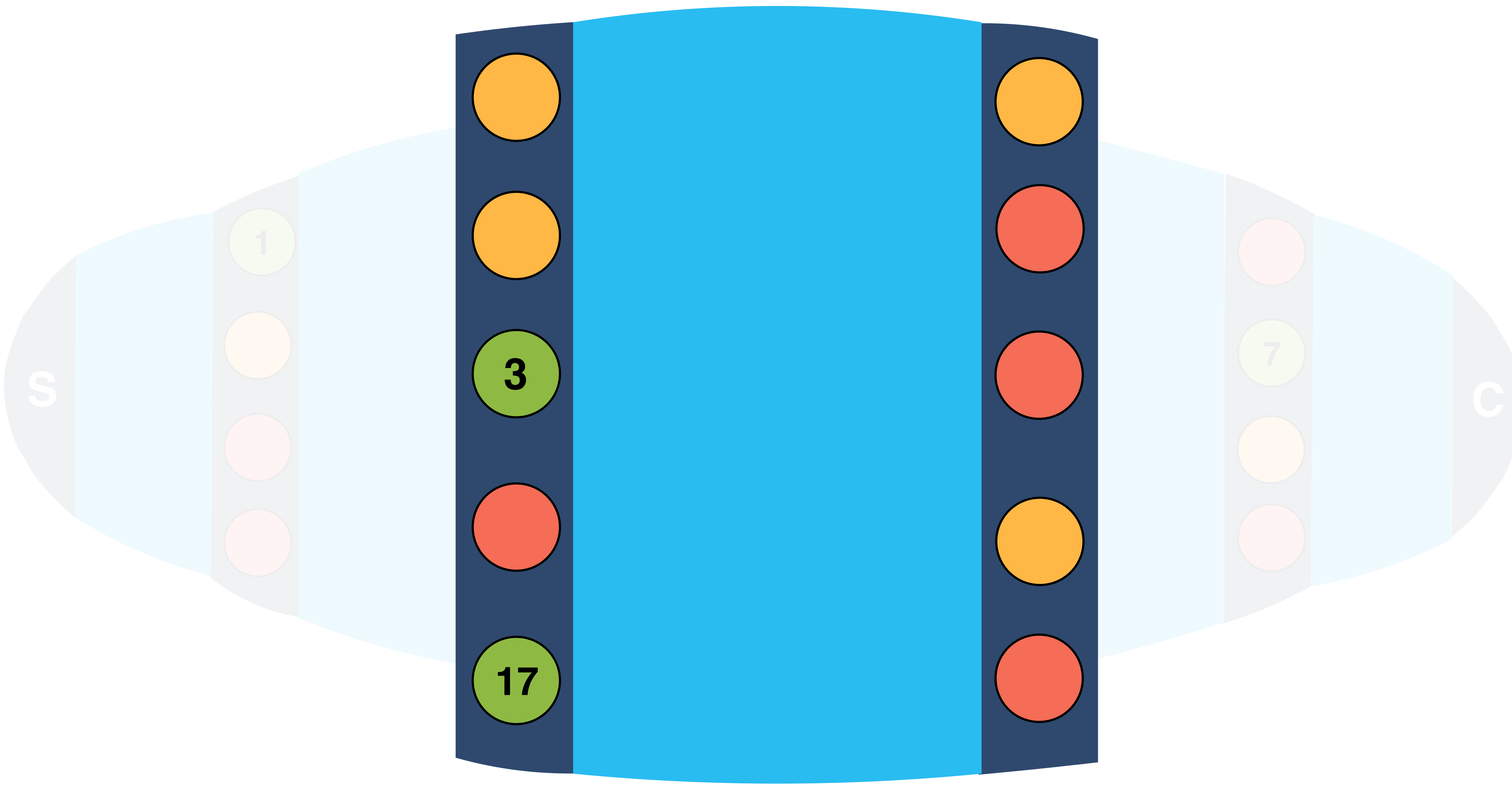
Solving the Border Problem Recursively



Solving the Border Problem Recursively



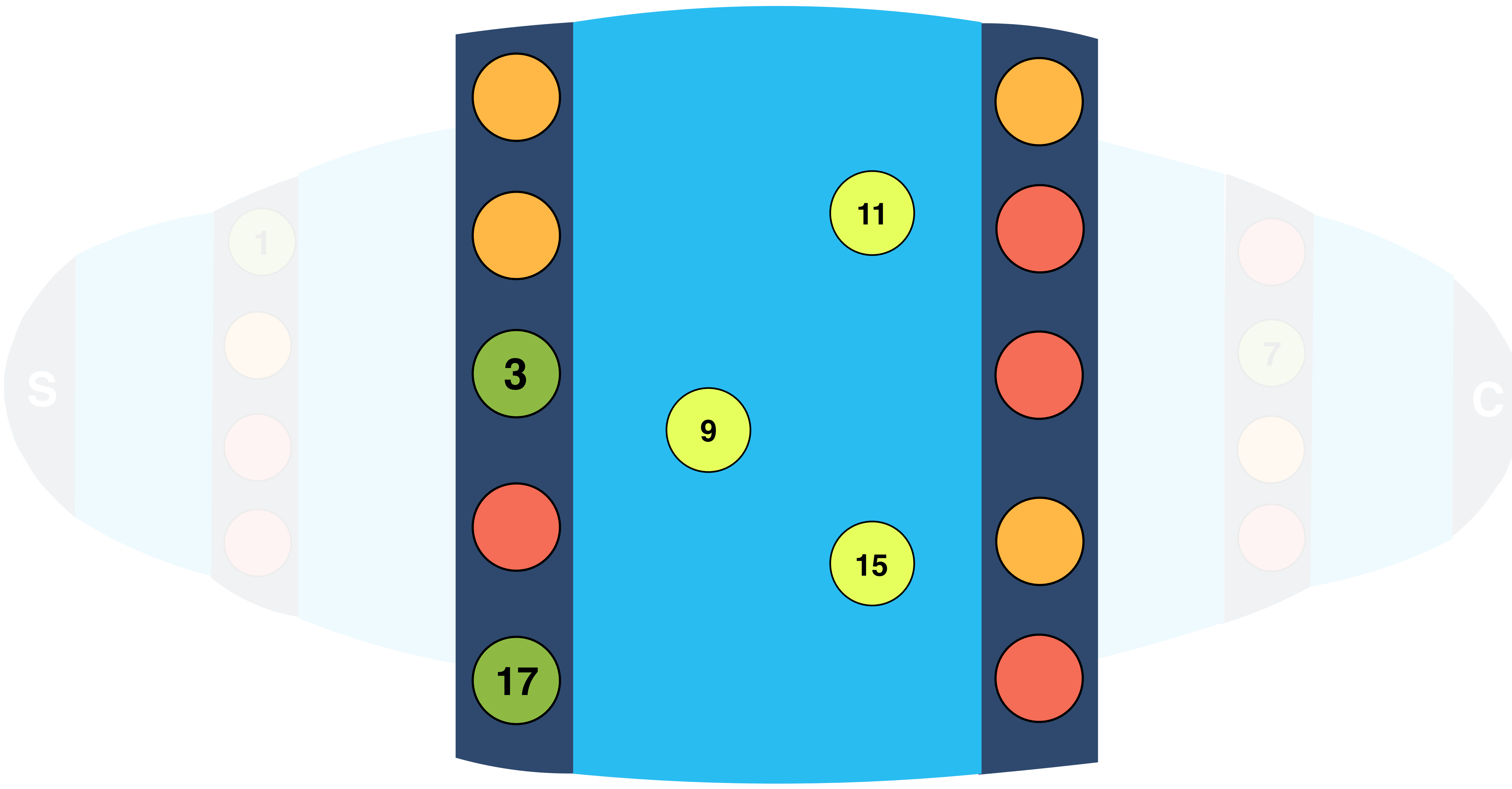
Solving the Border Problem Recursively



S_{i-1} W_i S_i

$$G_i = G[S_{i-1} \cup W_i \cup S_i]$$

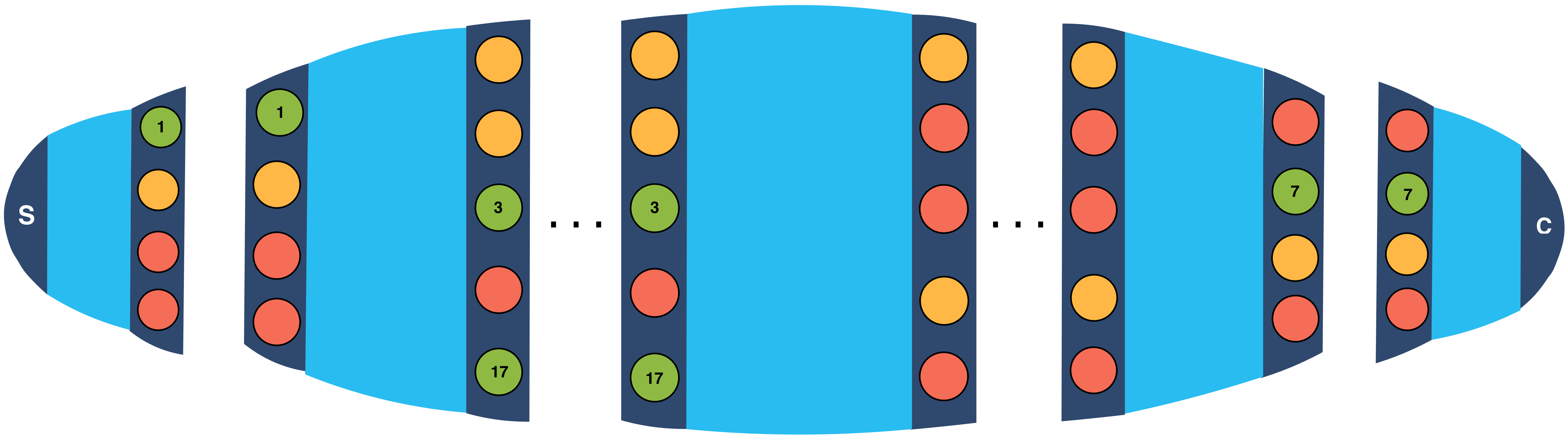
Solving the Border Problem Recursively



S_{i-1} W_i S_i

$$G_i = G[S_{i-1} \cup W_i \cup S_i]$$

Solving Recursively



$G[S_1 \cup W_2 \cup S_2]$

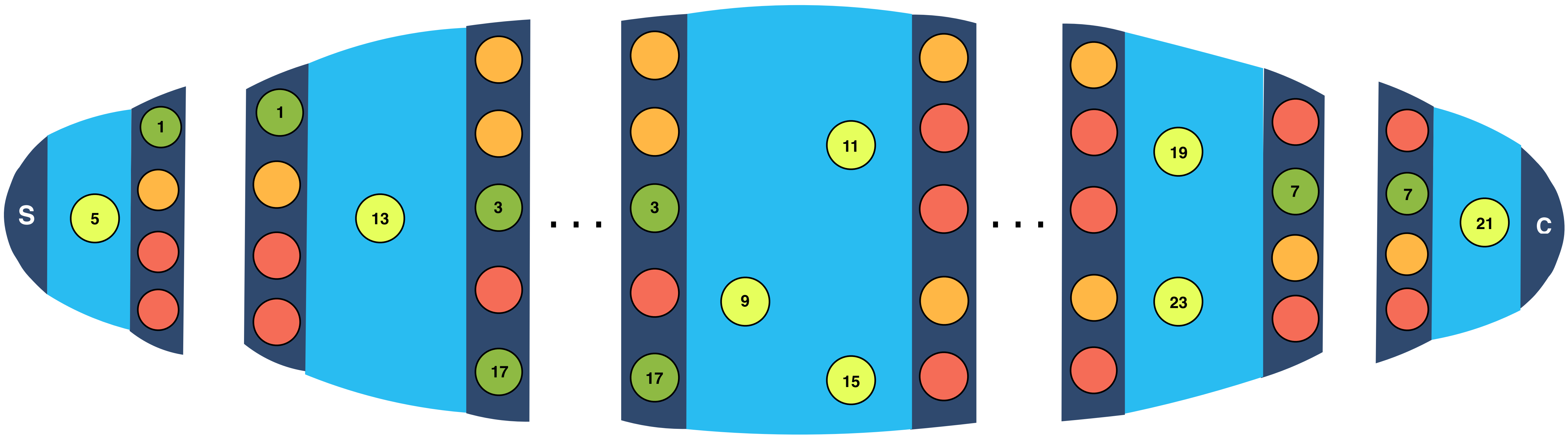
$G[S_{i-1} \cup W_i \cup S_i]$

$G[S_{q-1} \cup W_q \cup S_q]$

$G[S_0 \cup W_1 \cup S_1]$

$G[S_q \cup W_{q+1} \cup S_{q+1}]$

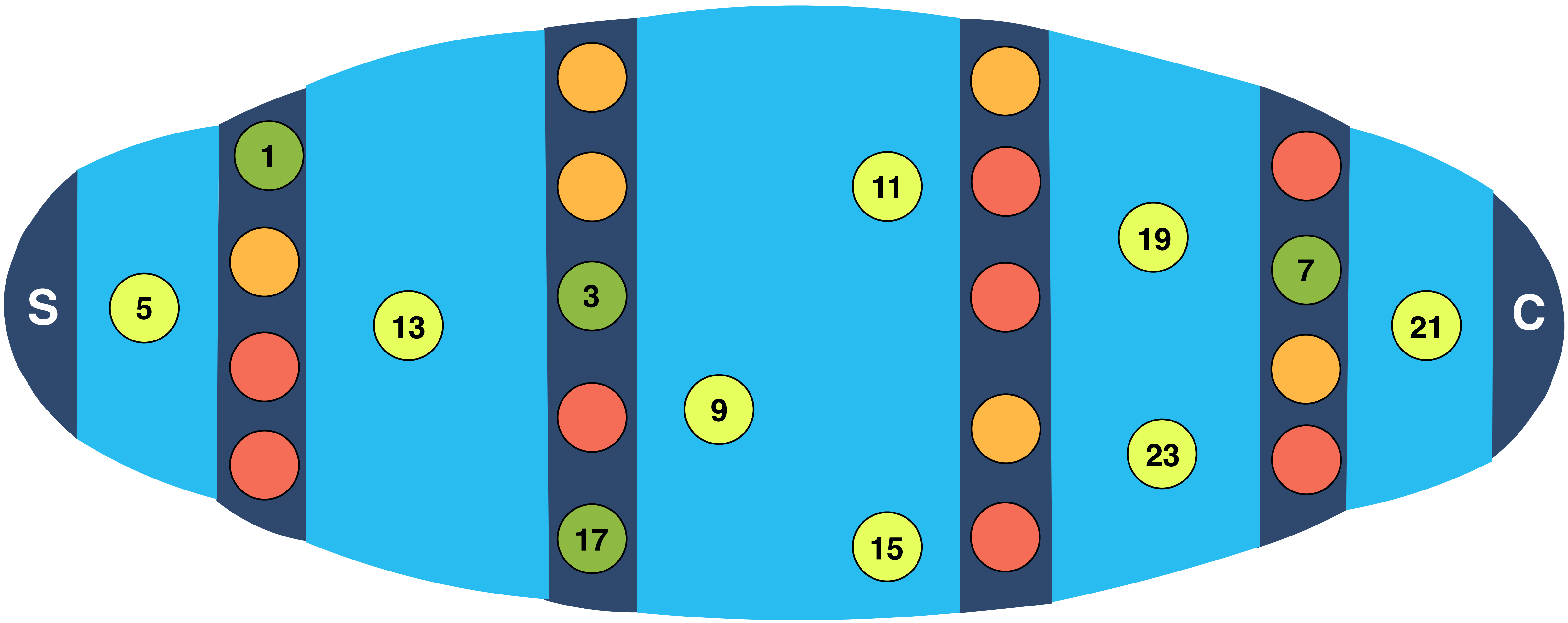
Solving Recursively



$G[S_1 \cup W_2 \cup S_2]$ $G[S_{i-1} \cup W_i \cup S_i]$ $G[S_{q-1} \cup W_q \cup S_q]$

$G[S_0 \cup W_1 \cup S_1]$ $G[S_q \cup W_{q+1} \cup S_{q+1}]$

Combining the solutions



Algorithm

Algorithm 1: Solve-SACS-R($\mathcal{I}$)

Input: An instance $(G, s, C, k, g, P, Q, Y, \gamma)$, $p := |P|$

Result: YES if $\mathcal{I}$ is a YES-instance of SACS-R, and NO otherwise.

if $p = 0$ *and* s *and* C *are in different components of* $G \setminus Y$ **then return** YES;

else return NO;

if $p > 0$ *and* s *and* C *are in different components of* $G \setminus Y$ **then return** YES;

if *there is no* $s - C$ *separator of size at most* p **then return** NO;

Compute a tight $s - C$ separator sequence $\mathcal{S}$ of order p.

if *the number of separators in* $\mathcal{S}$ *is greater than* k **then return** YES;

else

  for *a non-trivial partition* $\mathcal{T}_1(P), \mathcal{T}_2(P)$ *of* P *into* $2q + 1$ *parts* **do**

    for *a labeling* $\mathfrak{T}$ *compatible with* $\mathcal{T}_1(P)$ **do**

      if $\bigwedge_{i=1}^{q+1}$ *(Solve-*SACS-R*(*$\mathcal{I}\langle i, \mathcal{T}_1(P), \mathcal{T}_2(P), \mathfrak{T}_i \rangle$*))* **then return** YES;

  return NO

The Measure drops

Proposed measure is $\mu(G) = p$, which is the number of timestamps.

Claim: The quantity p always decrease when we recurse

Running Time

$$T(n,m,k,p) \leq O(n^2mp) + (p+k+1)^{kp} \sum_{I=1}^{q+1} T(n_i, m_i, k, p_i)$$

Running Time

$$T(n, m, k, p) \leq O(n^2 mp) + (p + k + 1)^{kp} \sum_{I=1}^{q+1} T(n_i, m_i, k, p_i)$$

Recall that:

- each $p_i \leq k$,
- the depth of the recursion is bounded by p, and
- at the each level the work done is proportional to $k^{O(kp)} n^2 m$

Running Time

$$T(n,m,k,p) \leq O(n^2mp) + (p+k+1)^{kp} \sum_{I=1}^{q+1} T(n_i,m_i,k,p_i)$$

Recall that:

- each $p_i \leq k$,
- the depth of the recursion is bounded by p, and
- at the each level the work done is proportional to $k^{O(kp)}n^2m$

SACS is FPT and has an algorithm with running time $f(k)O(n^2m)$ where, $f(k) = k^{O(k^3)}$

Kernels on Trees

Kernelization

A kernelization algorithm, or simply a kernel, for a ***parameterized problem Q*** is an ***algorithm A*** that, given an instance ***(I,k) of Q***, works in polynomial time and returns an ***equivalent instance (I’,k’) of Q***. Moreover, we require that ***k’ ≤ k***.

No Poly Kernels on Trees

The unparameterized version of $SACS$ restricted to trees cross composes to $SACS$ restricted to trees when parameterized by the number of firefighters

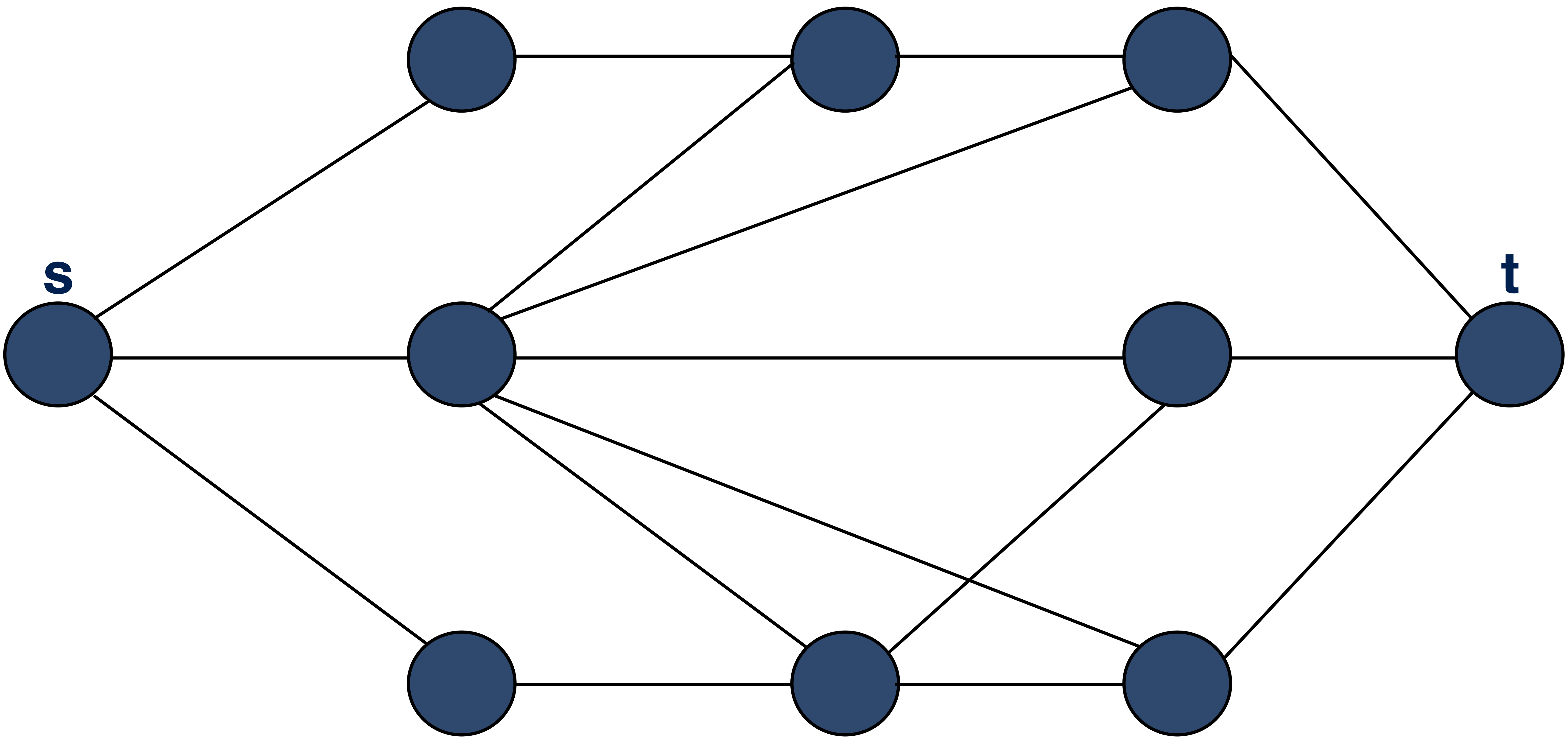
No Poly Kernels on Trees

The unparameterized version of $SACS$ restricted to trees cross composes to $SACS$ restricted to trees when parameterized by the number of firefighters

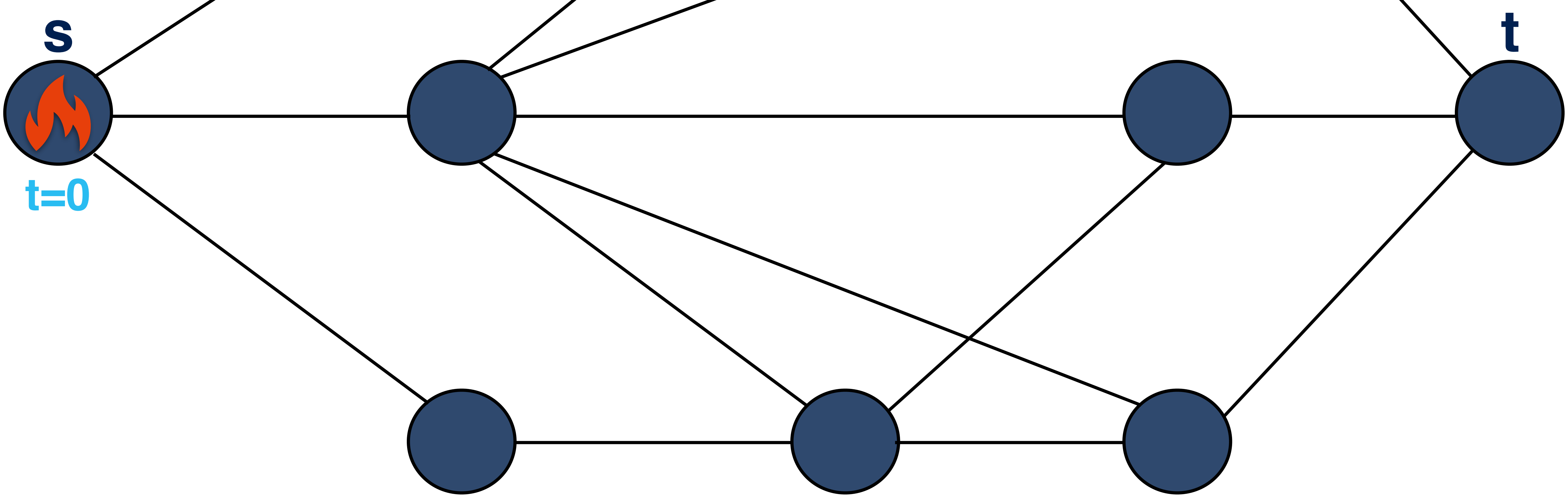
$SACS$ when restricted to trees does not admit a polynomial kernel, unless $NP \subseteq coNP/poly$

The Spreading Model

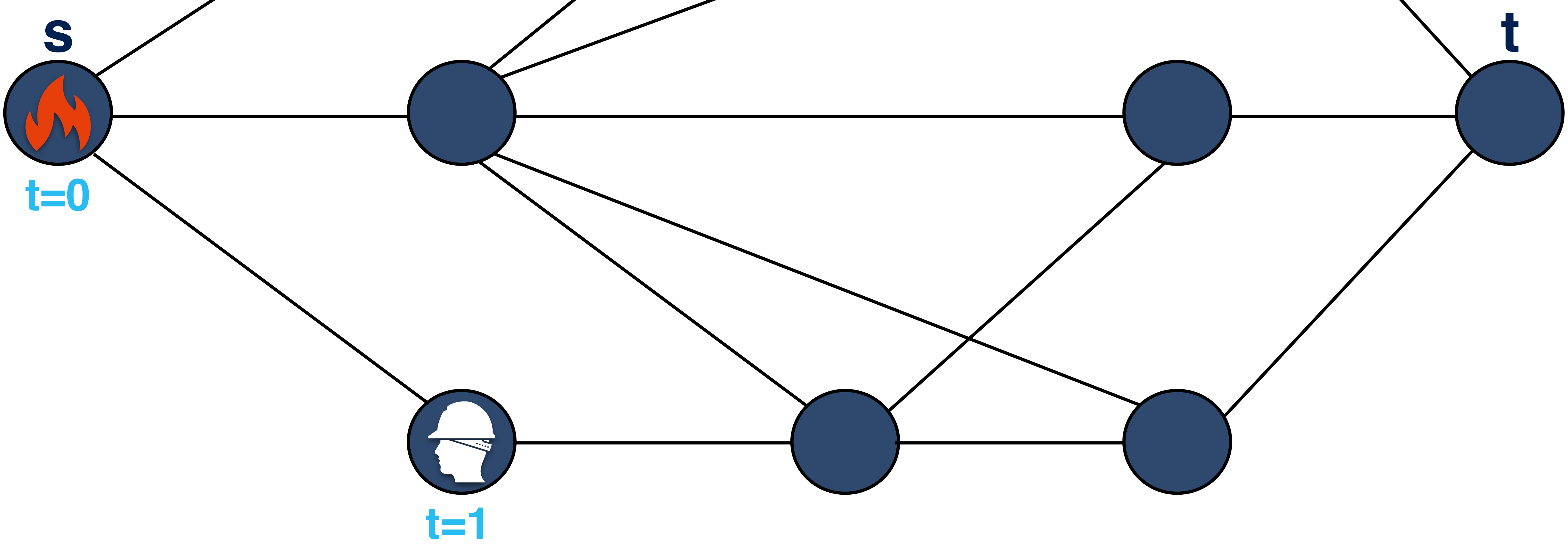
Spreading Vaccination Model



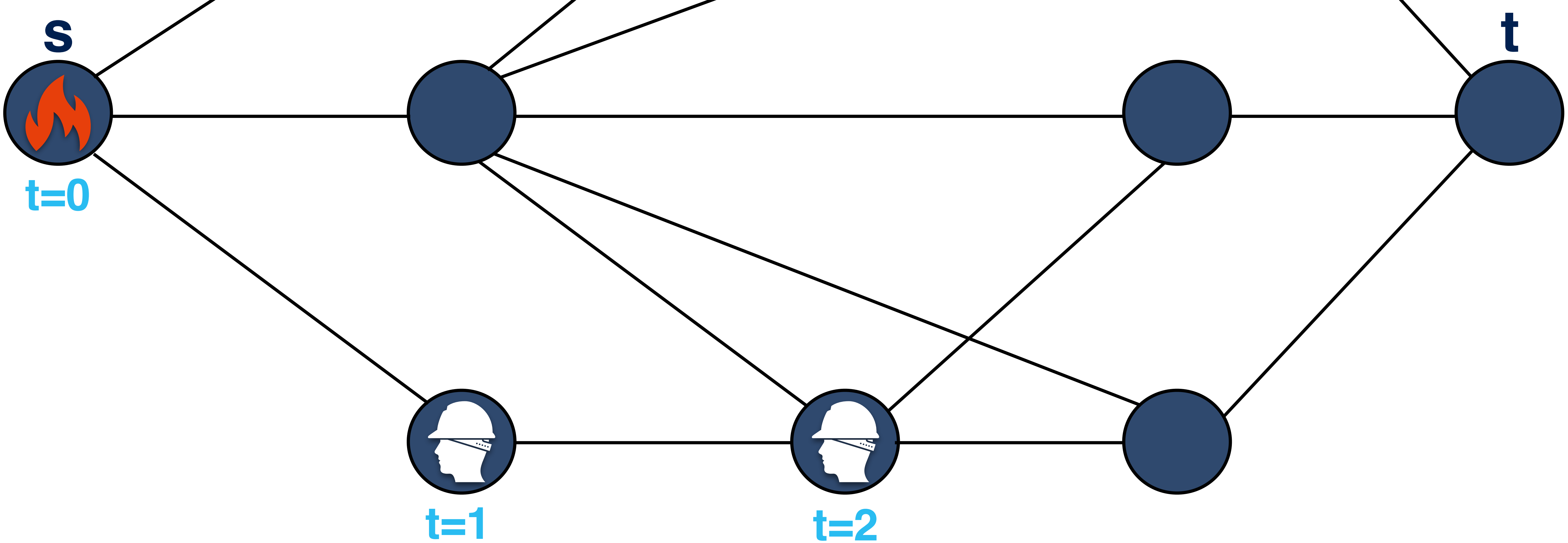
Spreading Vaccination Model



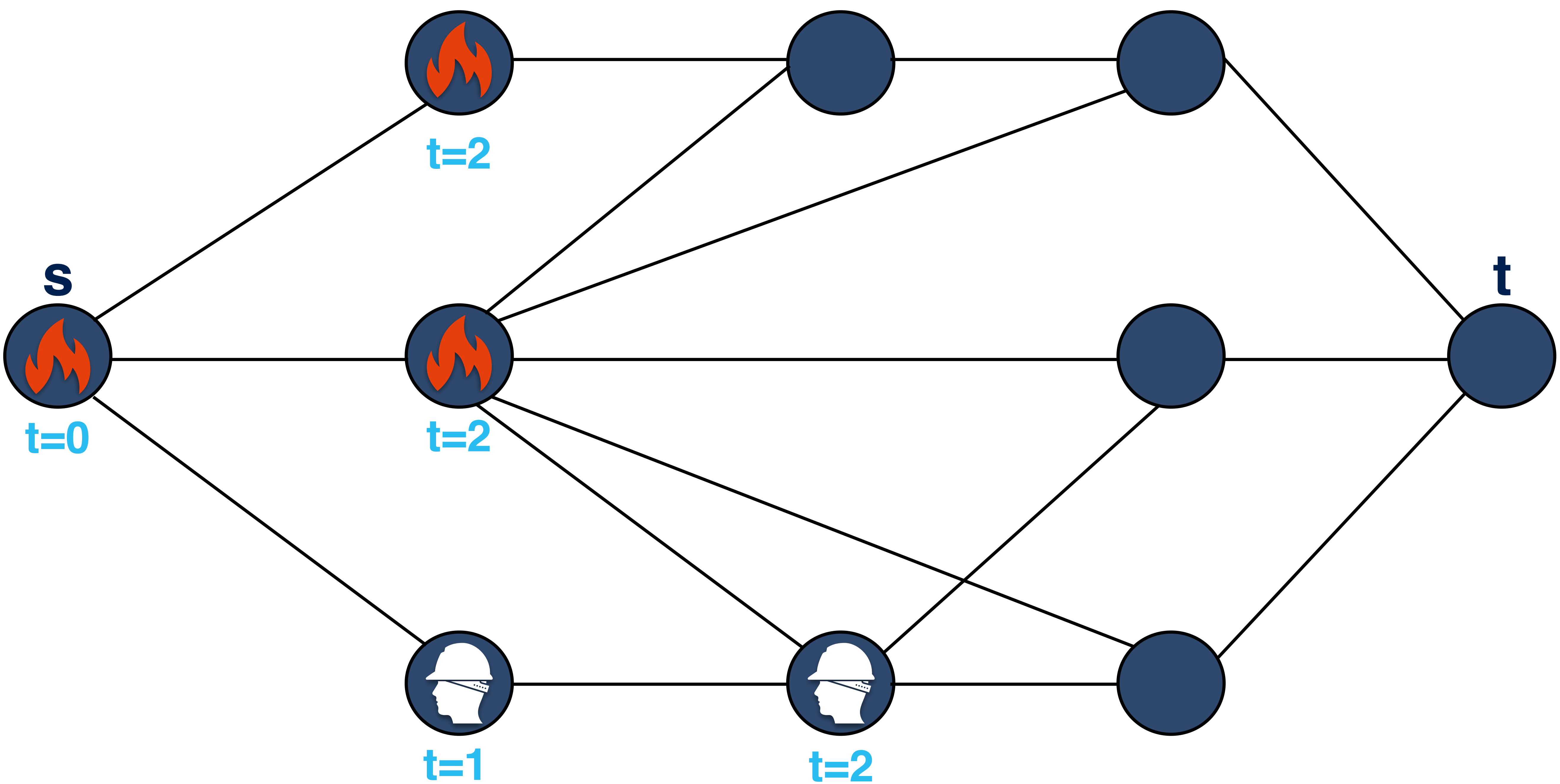
Spreading Vaccination Model



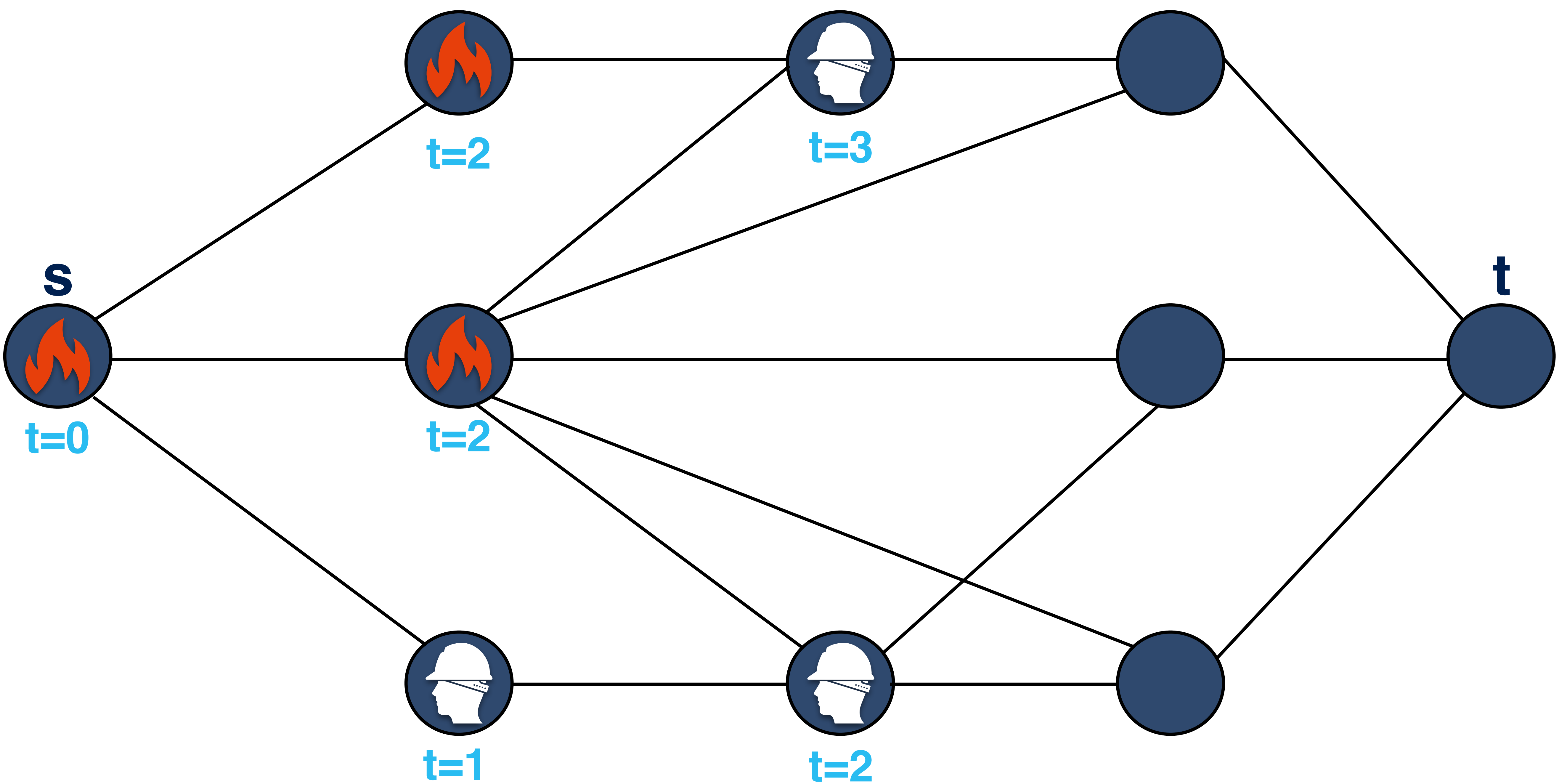
Spreading Vaccination Model



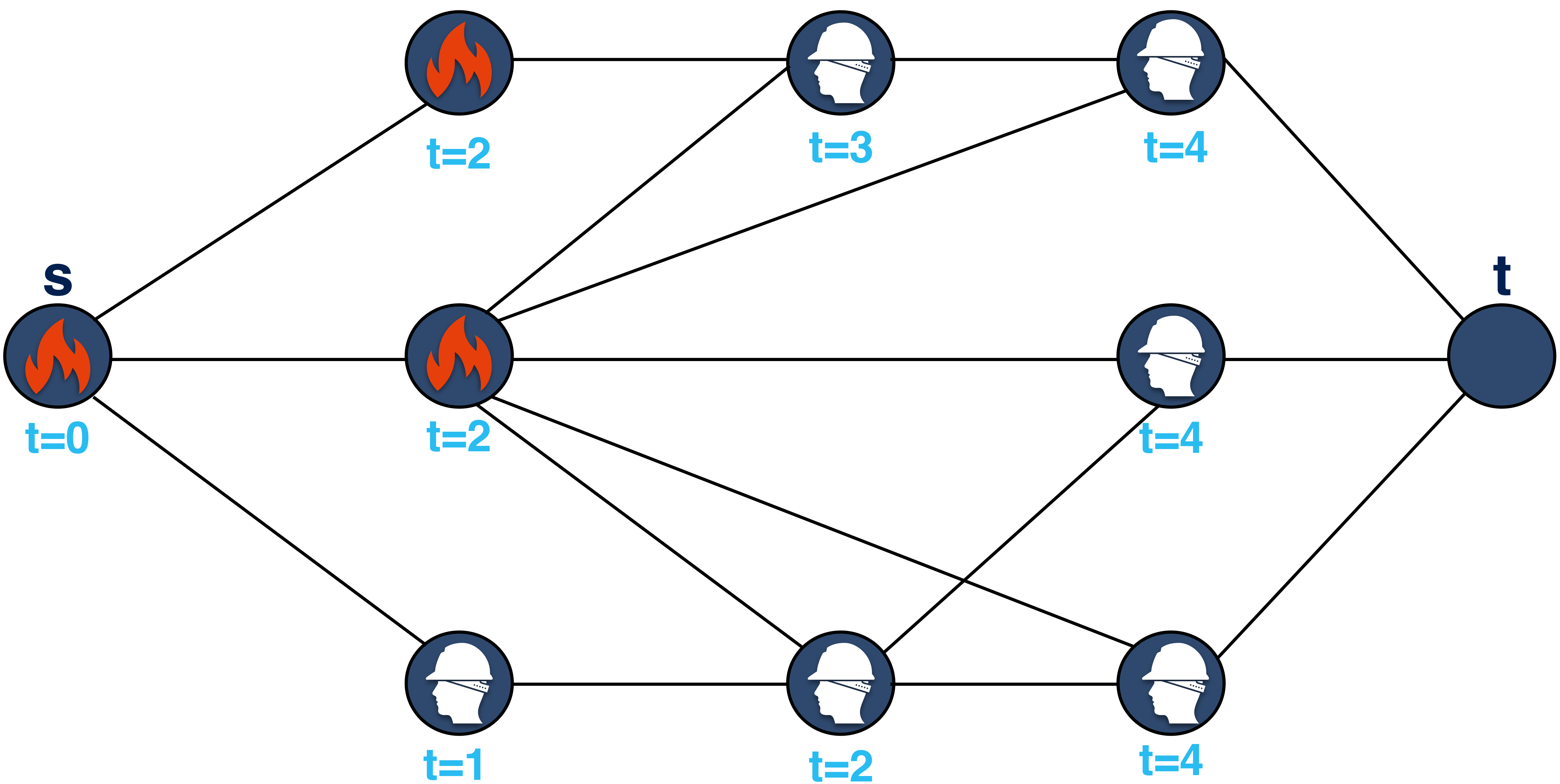
Spreading Vaccination Model



Spreading Vaccination Model



Spreading Vaccination Model



Spreading Vaccination Model

Theorem:

In the spreading model, SACS is as hard as k-DOMINATING SET

Conclusion

Conclusion and Future Work

1. Saving a Critical Set when parameterized by number of firefighters is FPT

2. There are no polynomial kernels for trees

3. In contrast to the general firefighting model, the spreading model is W[2]-Hard

4. ***Future Work:***
    - Kernels for graphs
    - Smarter FPT algorithm
    - Firefighting on graphs with bounded clique width, bounded clique-cover number, interval graphs, split graphs, permutation graphs, etc.

Questions?

Thank You