# Maximizing the Probability of Fixation in the Positional Voter Model 

Petros Petsinis, Andreas Pavlogiannis, Panagiotis Karras<br>Department of Computer Science<br>Aarhus University

February 11, 2023


## Diffusion processes

Natural spread through networks

- propagation of information in social networks
- spread of virus in human population
- spread of mutation in biological networks


## Diffusion processes

Natural spread through networks

- propagation of information in social networks
- spread of virus in human population
- spread of mutation in biological networks

Type of Diffusion process

## Diffusion processes

Natural spread through networks

- propagation of information in social networks
- spread of virus in human population
- spread of mutation in biological networks

Type of Diffusion process
(1) Progressive: independent cascade, linear threshold, triggering,...


## Diffusion processes

Natural spread through networks

- propagation of information in social networks
- spread of virus in human population
- spread of mutation in biological networks

Type of Diffusion process
(1) Progressive: independent cascade, linear threshold, triggering,...
(2) Non-Progressive: moran, voter, SIR, SIS,...


## Diffusion processes

Natural spread through networks

- propagation of information in social networks
- spread of virus in human population
- spread of mutation in biological networks

Type of Diffusion process
(1) Progressive: independent cascade, linear threshold, triggering,...
(2) Non-Progressive: moran, voter, SIR, SIS,...

This paper: Non-Progressive model that describes the spread of mutation/novel-trait.

## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.

## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.
Agents:


Nodes:


## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.
Agents:


Nodes:


## Process:

## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.

## Agents:



Nodes:


## Process:

$\Rightarrow$ Initially residents occupy all nodes.

## Example



## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.

Agents:


Nodes:


## Process:

$\Rightarrow$ Initially residents occupy all nodes.
$\Rightarrow t=0$ : random mutation in one agent.


## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.

Agents:


Nodes:


## Process:

$\Rightarrow$ Initially residents occupy all nodes.
$\Rightarrow t=0$ : random mutation in one agent.
$\Rightarrow t>0$ : repeat death-Birth steps:


## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.

Agents:


Nodes:


## Process:

$\Rightarrow$ Initially residents occupy all nodes.
$\Rightarrow t=0$ : random mutation in one agent.
$\Rightarrow t>0$ : repeat death-Birth steps:
(1) death: Pick random node $\boldsymbol{v}$ to update.


## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.
Agents:


Nodes:


Fitness:


## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.

Agents:


Nodes:


Fitness:

| death - Birth | f(u\|v) |  |
| :---: | ---: | :---: |
| v) | u | $1+\delta$ |
| v | u | 1 |
| v | u | 1 |
| v | u | 1 |

## Process:

$\Rightarrow$ Initially residents occupy all nodes.
$\Rightarrow t=0$ : random mutation in one agent.
$\Rightarrow t>0$ : repeat death-Birth steps:
(1) death: Pick random node $\boldsymbol{v}$ to update.
(2) Birth: Pick an in-neighbor node $\boldsymbol{u}$ of $\boldsymbol{v}$ proportionally to its fitness $f(u \mid v)$ and edge-weight $w(u, v)\left(\frac{f(u \mid v) w(u, v)}{\sum_{k} f(k \mid v) w(k, v)}\right)$ to transfer its trait on $\boldsymbol{v}$.


## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.

Agents:


Nodes:


Fitness:

| death - Birth | f(u\|v) |  |
| :---: | ---: | :---: |
| v) | u | $1+\delta$ |
| v | u | 1 |
| v | u | 1 |
| v | u | 1 |

## Process:

$\Rightarrow$ Initially residents occupy all nodes.
$\Rightarrow t=0$ : random mutation in one agent.
$\Rightarrow t>0$ : repeat death-Birth steps:
(1) death: Pick random node $\boldsymbol{v}$ to update.
(2) Birth: Pick an in-neighbor node $\boldsymbol{u}$ of $\boldsymbol{v}$ proportionally to its fitness $f(u \mid v)$ and edge-weight $w(u, v)\left(\frac{f(u \mid v) w(u, v)}{\sum_{k} f(k \mid v) w(k, v)}\right)$ to transfer its trait on $\boldsymbol{v}$.


## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.

Agents:


Nodes:


Fitness:

| death - Birth | $f(u \mid v)$ |  |
| :---: | ---: | :---: |
| $v$ | $u$ | $1+\delta$ |
| $v$ | $u$ | 1 |
| $v$ | $u$ | 1 |
| $v$ | $u$ | 1 |

## Process:

$\Rightarrow$ Initially residents occupy all nodes.
$\Rightarrow t=0$ : random mutation in one agent.
$\Rightarrow t>0$ : repeat death-Birth steps:
(1) death: Pick random node $\boldsymbol{v}$ to update.
(2) Birth: Pick an in-neighbor node $\boldsymbol{u}$ of $\boldsymbol{v}$ proportionally to its fitness $f(u \mid v)$ and edge-weight $w(u, v)\left(\frac{f(u \mid v) w(u, v)}{\sum_{k} f(k \mid v) w(k, v)}\right)$ to transfer its trait on $\boldsymbol{v}$.


## Positional Voter Model - This Work

Graph: Population of $n$ agents spread over nodes of graph $G=(V, E, w)$.

Agents:


Nodes:


Fitness:

| death - Birth | $f(u \mid v)$ |  |
| :---: | ---: | :---: |
| $v$ | $u$ | $1+\delta$ |
| $v$ | $u$ | 1 |
| $v$ | $u$ | 1 |
| $v$ | $u$ | 1 |

## Process:

$\Rightarrow$ Initially residents occupy all nodes.
$\Rightarrow t=0$ : random mutation in one agent.
$\Rightarrow t>0$ : repeat death-Birth steps:
(1) death: Pick random node $\boldsymbol{v}$ to update.
(2) Birth: Pick an in-neighbor node $\boldsymbol{u}$ of $\boldsymbol{v}$ proportionally to its fitness $f(u \mid v)$ and edge-weight $w(u, v)\left(\frac{f(u \mid v) w(u, v)}{\sum_{k} f(k \mid v) w(k, v)}\right)$ to transfer its trait on $\boldsymbol{v}$.


Setting Parameters: graph $G$, set of biased nodes $S$, bias $\delta$.
Fixation Probability: The probability $\operatorname{fp}\left(G^{S}, \delta\right)$ that a random mutation leads to fixation.


$$
S=V \Longrightarrow \text { Positional = Standard }
$$



Positional


Standard

## Fixation Maximization

Optimization Problem: Given a graph $G$ and a budget $k$, which $k$ nodes should we bias with $\delta$ to maximize the fixation probability?

$$
S^{*}=\arg \max _{S,|S|=k} \operatorname{fp}\left(G^{S}, \delta\right)
$$



## Fixation Maximization

Optimization Problem: Given a graph $G$ and a budget $k$, which $k$ nodes should we bias with $\delta$ to maximize the fixation probability?

$$
S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}\left(G^{S}, \delta\right)
$$

## Results Overview:



## Fixation Maximization

Optimization Problem: Given a graph $G$ and a budget $k$, which $k$ nodes should we bias with $\delta$ to maximize the fixation probability?

$$
S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}\left(G^{S}, \delta\right)
$$

## Results Overview:

(1) FPRAS: for $\operatorname{fp}\left(G^{S}, \delta\right)$ in undirected graphs.

## Fixation Maximization

Optimization Problem: Given a graph $G$ and a budget $k$, which $k$ nodes should we bias with $\delta$ to maximize the fixation probability?

$$
S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}\left(G^{S}, \delta\right)
$$

## Results Overview:

(1) FPRAS: for $\operatorname{fp}\left(G^{S}, \delta\right)$ in undirected graphs.
(2) Monotone and non-Submodular in general.


## Fixation Maximization

Optimization Problem: Given a graph $G$ and a budget $k$, which $k$ nodes should we bias with $\delta$ to maximize the fixation probability?

$$
S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}\left(G^{S}, \delta\right)
$$

## Results Overview:

(1) FPRAS: for $\operatorname{fp}\left(G^{S}, \delta\right)$ in undirected graphs.
(2) Monotone and non-Submodular in general.
(3) NP-hardness of
$S^{*}=\arg \max _{S,|S|=k} \operatorname{fp}\left(G^{S}, \delta\right)$.


Optimization Problem: Given a graph $G$ and a budget $k$, which $k$ nodes should we bias with $\delta$ to maximize the fixation probability?

$$
S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}\left(G^{S}, \delta\right)
$$

## Results Overview:

(1) FPRAS: for $\operatorname{fp}\left(G^{S}, \delta\right)$ in undirected graphs.
(2) Monotone and non-Submodular in general.
(3) NP-hardness of
$S^{*}=\arg \max _{S,|S|=k} \operatorname{fp}\left(G^{S}, \delta\right)$.
(4) Approximations for undirected graphs with self-loops and $\delta \rightarrow \infty$.


Optimization Problem: Given a graph $G$ and a budget $k$, which $k$ nodes should we bias with $\delta$ to maximize the fixation probability?

$$
S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}\left(G^{S}, \delta\right)
$$

## Results Overview:

(1) FPRAS: for $\operatorname{fp}\left(G^{S}, \delta\right)$ in undirected graphs.
(2) Monotone and non-Submodular in general.
(3) NP-hardness of
$S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}\left(G^{S}, \delta\right)$.
(4) Approximations for undirected graphs with self-loops and $\delta \rightarrow \infty$.
(5) Optimal Solution in polynomial time for symmetric graphs (i.e. $w(u, v)=w(v, u))$ as $\delta \rightarrow 0$.

The complexity of computing $\operatorname{fp}\left(G^{S}, \delta\right)$ is OPEN even when $S=V$.

## Lemma 1 - Expected Time

For undirected graphs, the expected time to a homogeneous state (all nodes are either mutants or residents) is $\mathcal{O}\left(n^{5}\right)$.

Approximations of $\operatorname{fp}\left(G^{S}, \delta\right)$ via monte-carlo simulations in P-time.

## Monotonicity \& Non-Submodularity

## Lemma 2 - Monotonicity

Given biased sets $S_{1}, S_{2}$ with $S_{1} \subseteq S_{2}$ and $\delta_{1}, \delta_{2} \geq 0$ with $\delta_{1} \leq \delta_{2}$, we have:

$$
\operatorname{fp}\left(G^{S_{1}}, \delta_{1}\right) \leq \operatorname{fp}\left(G^{S_{2}}, \delta_{2}\right)
$$

| death - Birth | $f(u \mid v)$ |
| :---: | :---: |
| (u) | u |
| $1+\delta$ |  |
| (u) | 1 |
| (u) | 1 |
| (u) u | 1 |

## Lemma 3 - Non-Submodularity

- $\operatorname{fp}\left(G^{S}, \delta\right)$ is not submodular.
- $\mathrm{fp}^{\infty}\left(G^{S}\right)$ is not submodular in general.

Submodular function: $\forall S_{1}, S_{2} \subseteq V \Rightarrow f\left(S_{1}\right)+f\left(S_{2}\right) \geq f\left(S_{1} \cup S_{2}\right)+f\left(S_{1} \cap S_{2}\right)$

## Key Lemma

## Lemma 4 - Self-looped Graphs

In undirected graphs with self-loops, if
$\delta \rightarrow \infty$, mutant agents in biased nodes are deathless; reproduce to themselves with probability 1.


| death - Birth | $\mathrm{f}(\mathrm{u} \mid \mathrm{v})$ |  |
| :---: | :---: | :---: |
| V | u | $1+\delta$ |
| V | u | 1 |
| v | u | 1 |
| v | u | 1 |

If trajectory $X_{t}=\left(X_{0}, X_{1}, \ldots, X_{t}\right)$ hits $S \Longrightarrow$ mutants fixate


## NP-hardness

## Theorem 5 - NP-hard <br> Maximizing $\operatorname{fp}\left(G^{S}, \delta\right)$ with $|S|=k$, is NP-hard.

## Theorem 5 - NP-hard

Maximizing $\operatorname{fp}\left(G^{S}, \delta\right)$ with $|S|=k$, is NP-hard.

## Proof.

Reduction from Vertex Cover in regular graphs, which is NP-hard. On undirected $d$-regular graphs with self-loops:

$$
\mathrm{fp}^{\infty}\left(G^{S}\right)=\frac{\frac{|S|}{n}+d}{1+d} \Leftrightarrow S \text { is a vertex-cover. }
$$



$$
\mathrm{fp}^{\infty}\left(G^{S}\right)=\frac{\frac{2}{4}+3}{1+3}=\frac{3.5}{4}
$$



$$
\mathrm{fp}^{\infty}\left(G^{S}\right)<\frac{3.5}{4}
$$

## Approximations

Lemma 6 - Submodularity
For undirected graphs with self-loops $\mathrm{fp}^{\infty}\left(G^{S}\right)$ is submodular;

## Approximations

Lemma 6 - Submodularity
For undirected graphs with self-loops $\mathrm{fp}^{\infty}\left(G^{S}\right)$ is submodular;

## Proof.

Submodular function: $\forall S_{1}, S_{2} \subseteq V \Rightarrow f\left(S_{1}\right)+f\left(S_{2}\right) \geq f\left(S_{1} \cup S_{2}\right)+f\left(S_{1} \cap S_{2}\right)$


## $X_{t}$ hits $S \Longrightarrow$ mutants fixate

$$
\begin{gathered}
\underline{\mathrm{fp}^{\infty}\left(G^{S_{1}}\right)}+\mathrm{fp}^{\infty}\left(G^{S_{2}}\right) \\
\geq \\
\mathrm{fp}^{\infty}\left(G^{S_{1} \cup S_{2}}\right)+\mathrm{fp}^{\infty}\left(G^{S_{1} \cap S_{2}}\right)
\end{gathered}
$$

## Approximations

## Lemma 6 - Submodularity

For undirected graphs with self-loops $\mathrm{fp}^{\infty}\left(G^{S}\right)$ is submodular;

## Proof.

Submodular function: $\forall S_{1}, S_{2} \subseteq V \Rightarrow f\left(S_{1}\right)+f\left(S_{2}\right) \geq f\left(S_{1} \cup S_{2}\right)+f\left(S_{1} \cap S_{2}\right)$


## $X_{t}$ hits $S \Longrightarrow$ mutants fixate

$$
\begin{gathered}
\operatorname{fp}^{\infty}\left(G^{S_{1}}\right)+\mathrm{fp}^{\infty}\left(G^{S_{2}}\right) \\
\geq \\
\underline{f p}^{\infty}\left(G^{S_{1} \cup S_{2}}\right) \\
\hline
\end{gathered}
$$

## Approximations

## Lemma 6 - Submodularity

For undirected graphs with self-loops $\mathrm{fp}^{\infty}\left(G^{S}\right)$ is submodular;

## Proof.

Submodular function: $\forall S_{1}, S_{2} \subseteq V \Rightarrow f\left(S_{1}\right)+f\left(S_{2}\right) \geq f\left(S_{1} \cup S_{2}\right)+f\left(S_{1} \cap S_{2}\right)$


## Approximations

Lemma 6 - Submodularity
For undirected graphs with self-loops $\mathrm{fp}^{\infty}\left(G^{S}\right)$ is submodular;

## Proof.

Submodular function: $\forall S_{1}, S_{2} \subseteq V \Rightarrow f\left(S_{1}\right)+f\left(S_{2}\right) \geq f\left(S_{1} \cup S_{2}\right)+f\left(S_{1} \cap S_{2}\right)$


## $X_{t}$ hits $S \Longrightarrow$ mutants fixate

$$
\begin{gathered}
\underline{\mathrm{fp}^{\infty}\left(G^{S_{1}}\right)}+\underset{\geq \mathrm{fp}^{\infty}\left(G^{S_{2}}\right)}{\geq \mathrm{fp}^{\infty}\left(G^{S_{1} \cup S_{2}}\right)}+\underline{f p}^{\infty}\left(G^{S_{1} \cap S_{2}}\right)
\end{gathered}
$$

## Corollary - Approximations

In undirected graphs with self-loops, $\mathrm{fp}^{\infty}\left(G^{S}\right)$ is:
Monotone + Submodular
(1-1/e) greedy approximation algorithm [Nemhauser1978]

## Optimal Solution

## Theorem 7 - Optimal Solution

For symmetric graphs $(w(u, v)=w(v, u))$, when $\delta \rightarrow 0$, finding $S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}\left(G^{S}, \delta\right)$ can be solved in P-time.

## Optimal Solution

## Theorem 7 - Optimal Solution

For symmetric graphs $(w(u, v)=w(v, u))$, when $\delta \rightarrow 0$, finding $S^{*}=\arg \max _{S,|S|=k} \operatorname{fp}\left(G^{S}, \delta\right)$ can be solved in P-time.

## Proof.

Using the Taylor expansion of $\operatorname{fp}\left(G^{S}, \delta\right)$ around $\delta=0$, that is:

$$
\begin{gathered}
\frac{\operatorname{fp}\left(G^{S}, 0\right)}{\Uparrow}+\delta \cdot \frac{\operatorname{fp}^{\prime}\left(G^{S}, 0\right)}{\Uparrow}+\frac{\mathcal{O}\left(\delta^{2}\right)}{\Uparrow} \\
1 / \mathrm{n}
\end{gathered} \text { Maximize this } 0
$$

 $S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}^{\prime}\left(G^{S}, 0\right)$.
(1) FPRAS: for $\mathrm{fp}\left(G^{S}, \delta\right)$ in undirected graphs.
(2) Monotone and not Submodular in general.
(3) NP-hardness of $S^{*}=\arg \max _{S,|S|=k} \mathrm{fp}\left(G^{S}, \delta\right)$.
(4) Approximation Algorithm for undirected graphs with self-loops and $\delta \rightarrow \infty$ : Monotone + Submodular $\rightarrow 1-1 / e$ greedy apx. algorithm.
(5) Optimal Solution in polynomial time for symmetric graphs (i.e. $w(u, v)=w(v, u))$ as $\delta \rightarrow 0$.


