Project Group "DynaSearch"
Intermediate Report: Results and Plans

Table of Contents

Introduction

Project Group

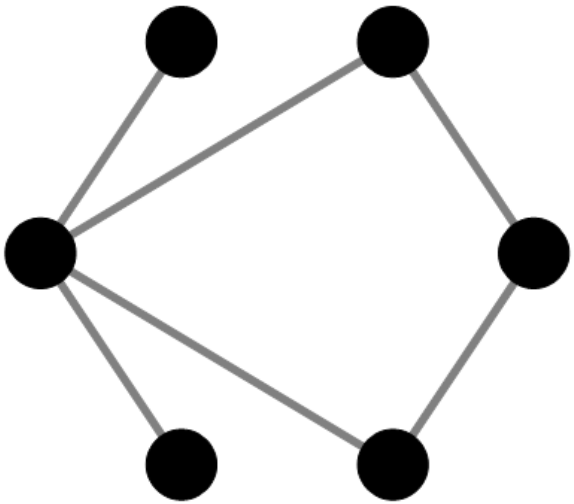
Our Work in the CRC 901

1. Big software from small pieces – Search for pieces that
   - Maximize some objective function or
   - Fulfill certain properties
2. Communicating entities with varying interests
   - Adapt network to these interests

Network Creation Games

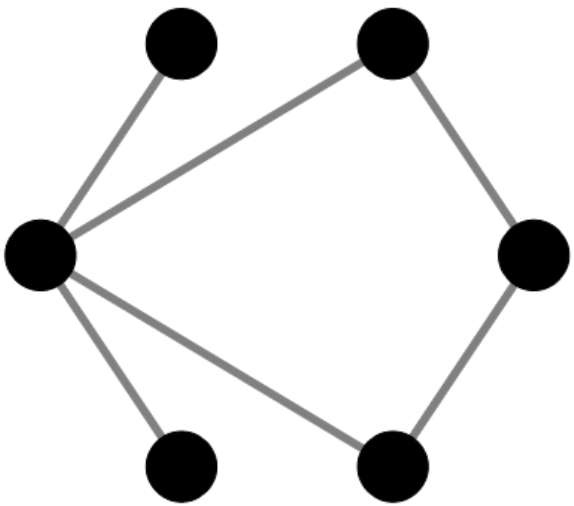
Scenario

Scenario



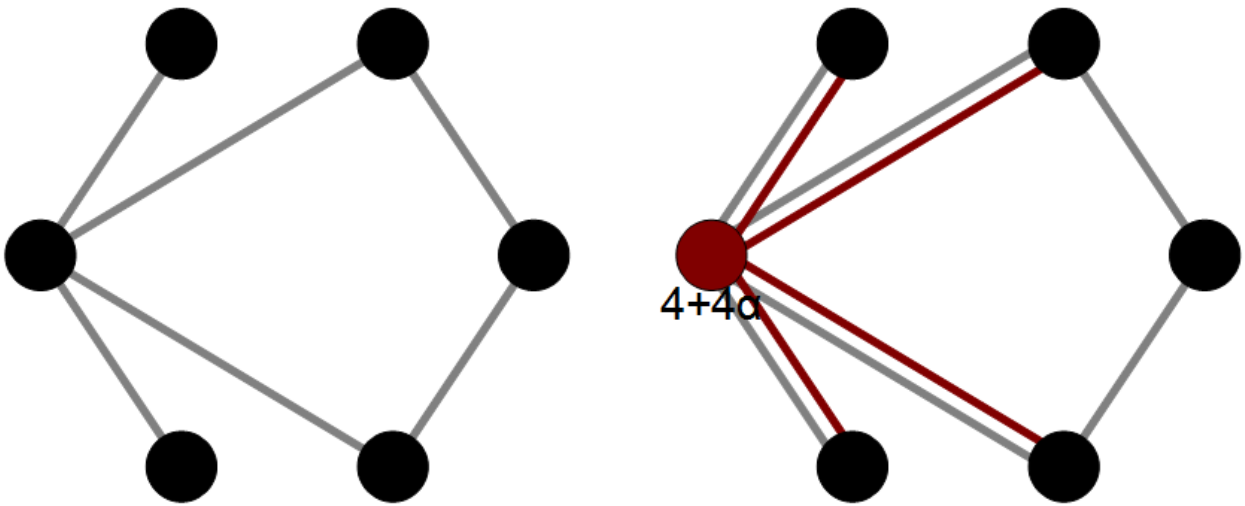
- Node $v \in V$ has friends $\mathcal{F}(v) \subseteq V$

Scenario



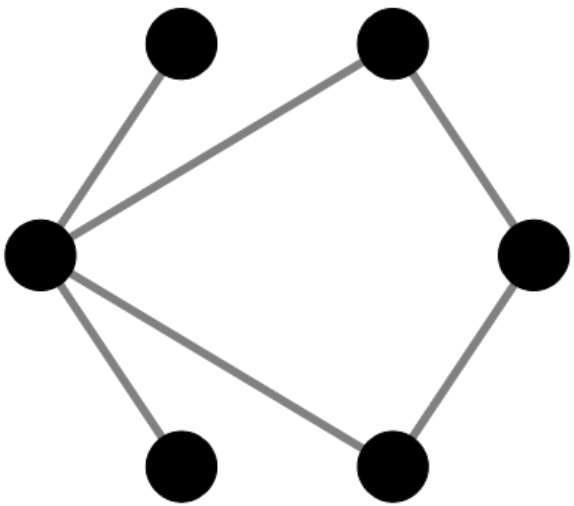
- Node $v \in V$ has friends $\mathcal{F}(v) \subseteq V$
- Node $v \in V$ maintains edges to selfishly chosen neighbors $N(v) \subseteq V$
- Costs per maintained edge: $\alpha \in \mathbb{R}$
- $v \in V$ minimizes costs: $\sum_{u \in \mathcal{F}(v)} d(u, v) + \alpha |N(v)|$

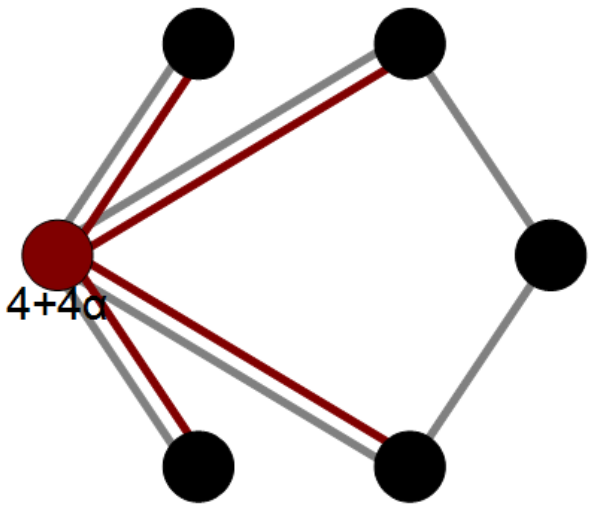
Scenario

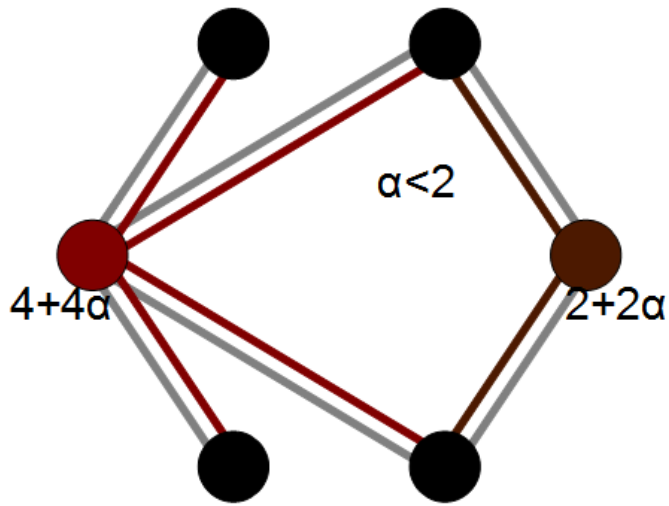


- Node $v \in V$ has friends $\mathcal{F}(v) \subseteq V$
- Node $v \in V$ maintains edges to selfishly chosen neighbors $N(v) \subseteq V$
- Costs per maintained edge: $\alpha \in \mathbb{R}$
- $v \in V$ minimizes costs: $\sum_{u \in \mathcal{F}(v)} d(u, v) + \alpha |N(v)|$

Scenario

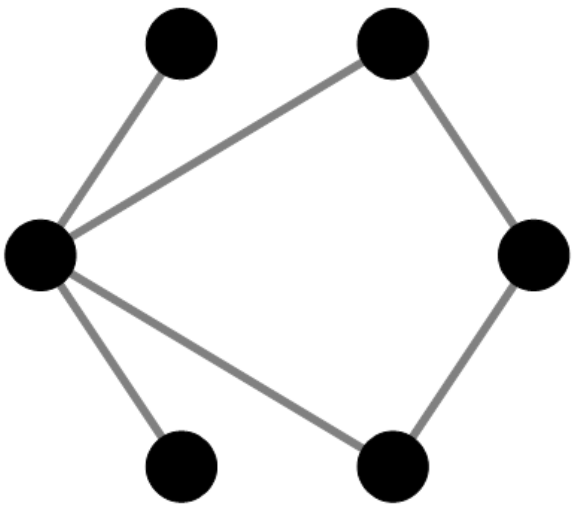


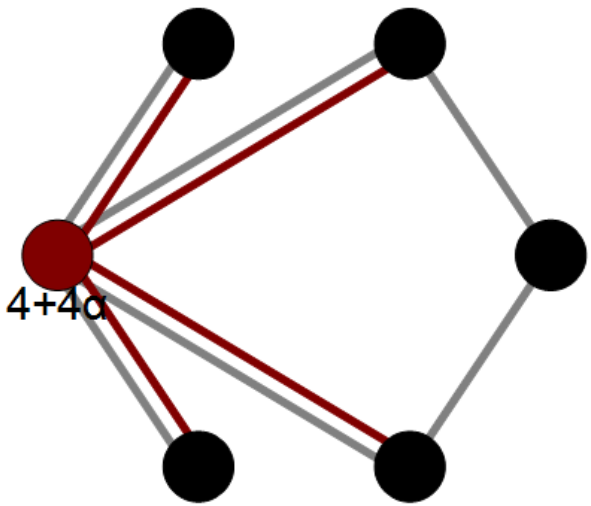


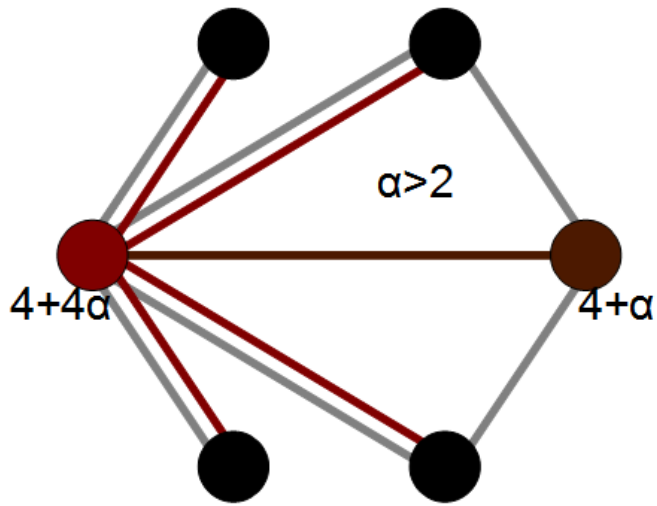


- Node $v \in V$ has friends $\mathcal{F}(v) \subseteq V$
- Node $v \in V$ maintains edges to selfishly chosen neighbors $N(v) \subseteq V$
- Costs per maintained edge: $\alpha \in \mathbb{R}$
- $v \in V$ minimizes costs: $\sum_{u \in \mathcal{F}(v)} d(u, v) + \alpha |N(v)|$

Scenario







- Node $v \in V$ has friends $\mathcal{F}(v) \subseteq V$
- Node $v \in V$ maintains edges to selfishly chosen neighbors $N(v) \subseteq V$
- Costs per maintained edge: $\alpha \in \mathbb{R}$
- $v \in V$ minimizes costs: $\sum_{u \in \mathcal{F}(v)} d(u, v) + \alpha |N(v)|$

Scenario
Research Focus

Definition (Nash equlibrium)

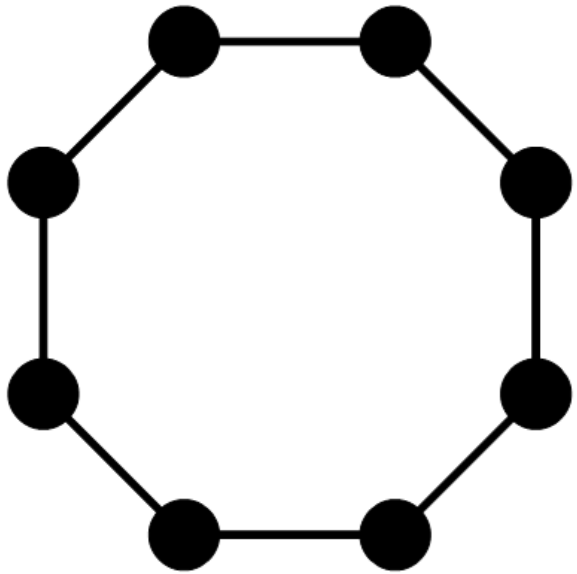
A state in which no node can improve its costs by choosing other neighbors.

- Focus of past work:
  - What do Nash equilibria look like?
  - How good are Nash equilibria compared to a global optimum?
- Our main focus: Under which circumstances are Nash equilibria reached?

Current Progress

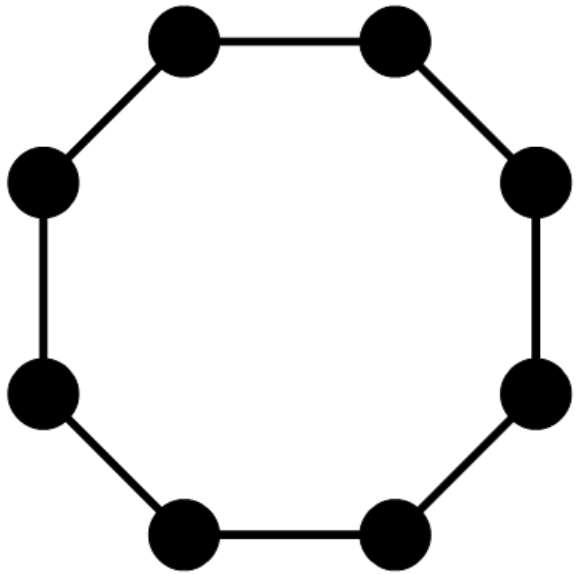
Start Problem

Current Progress
Start Problem



- Nodes on static ring topology

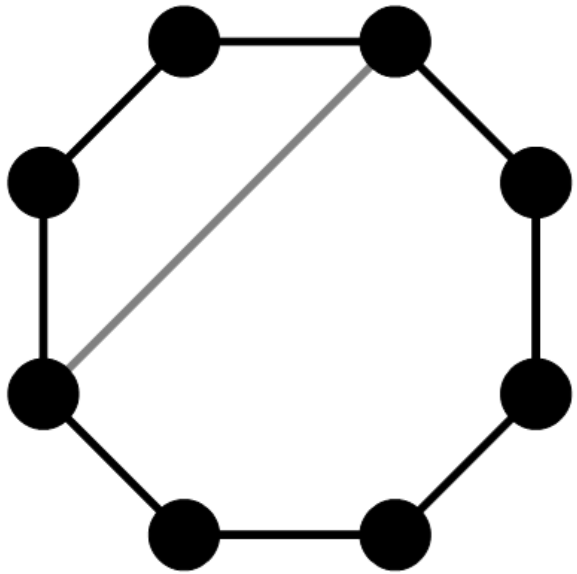
Current Progress
Start Problem



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached

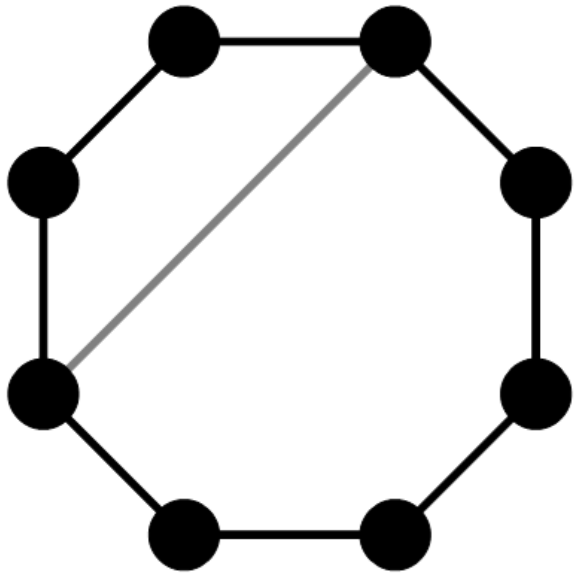
Current Progress

Start Problem



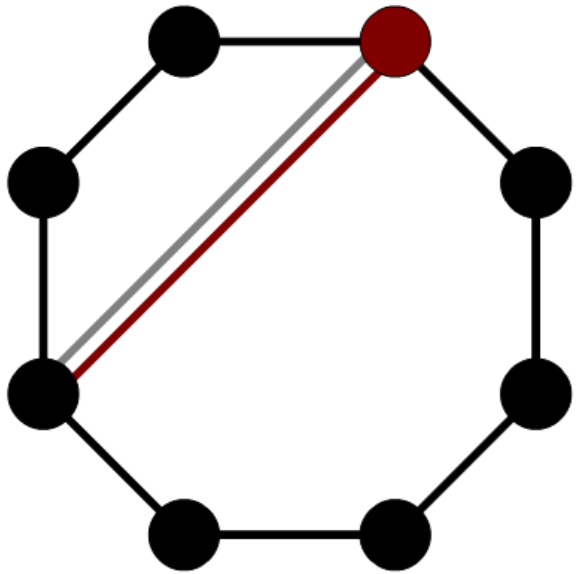
- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached

Current Progress
Start Problem



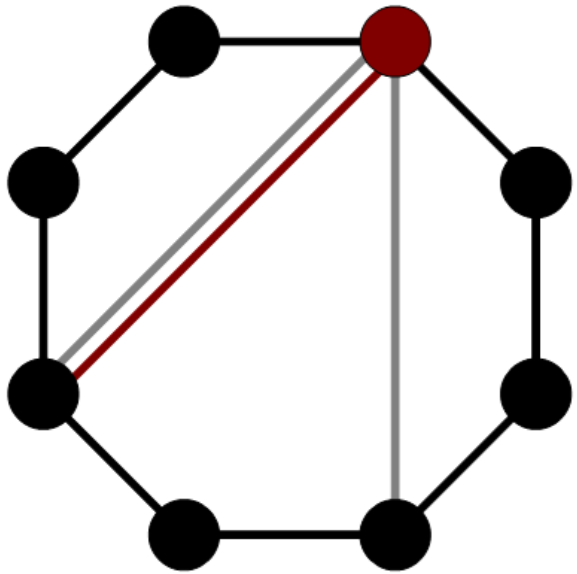
- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- Nodes can maintain up to 1 additional edge (without costs)
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress
Start Problem



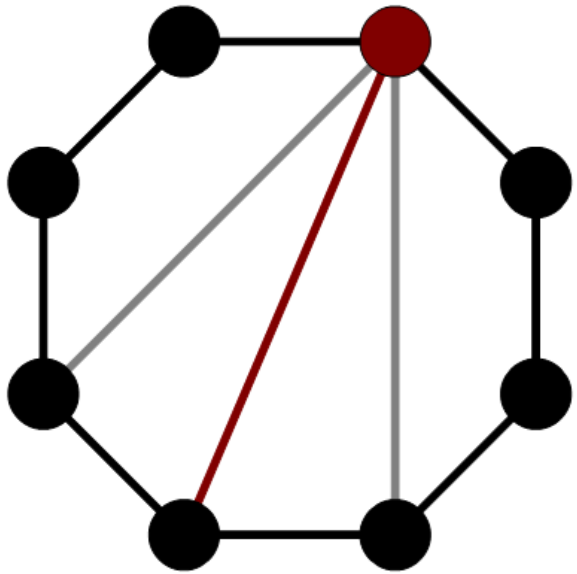
- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- Nodes can maintain up to 1 additional edge (without costs)
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress
Start Problem



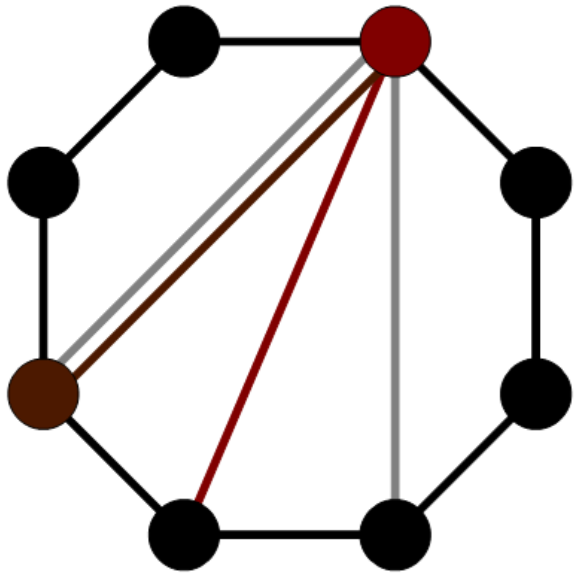
- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- Nodes can maintain up to 1 additional edge (without costs)
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress
Start Problem



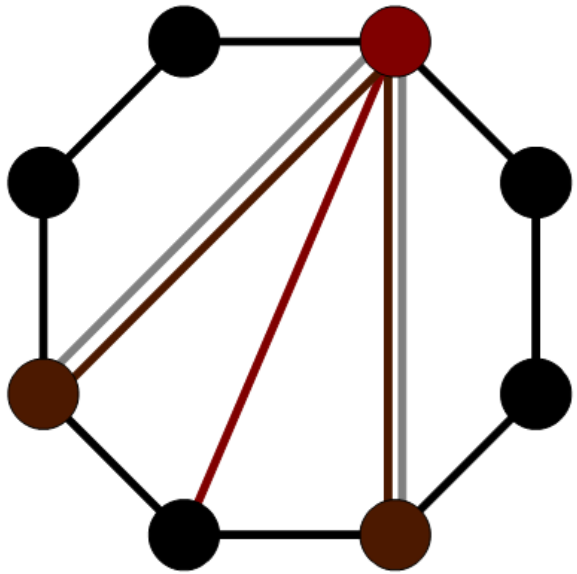
- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- Nodes can maintain up to 1 additional edge (without costs)
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress
Start Problem



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- Nodes can maintain up to 1 additional edge (without costs)
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress
Start Problem



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- Nodes can maintain up to 1 additional edge (without costs)
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress
Simulator

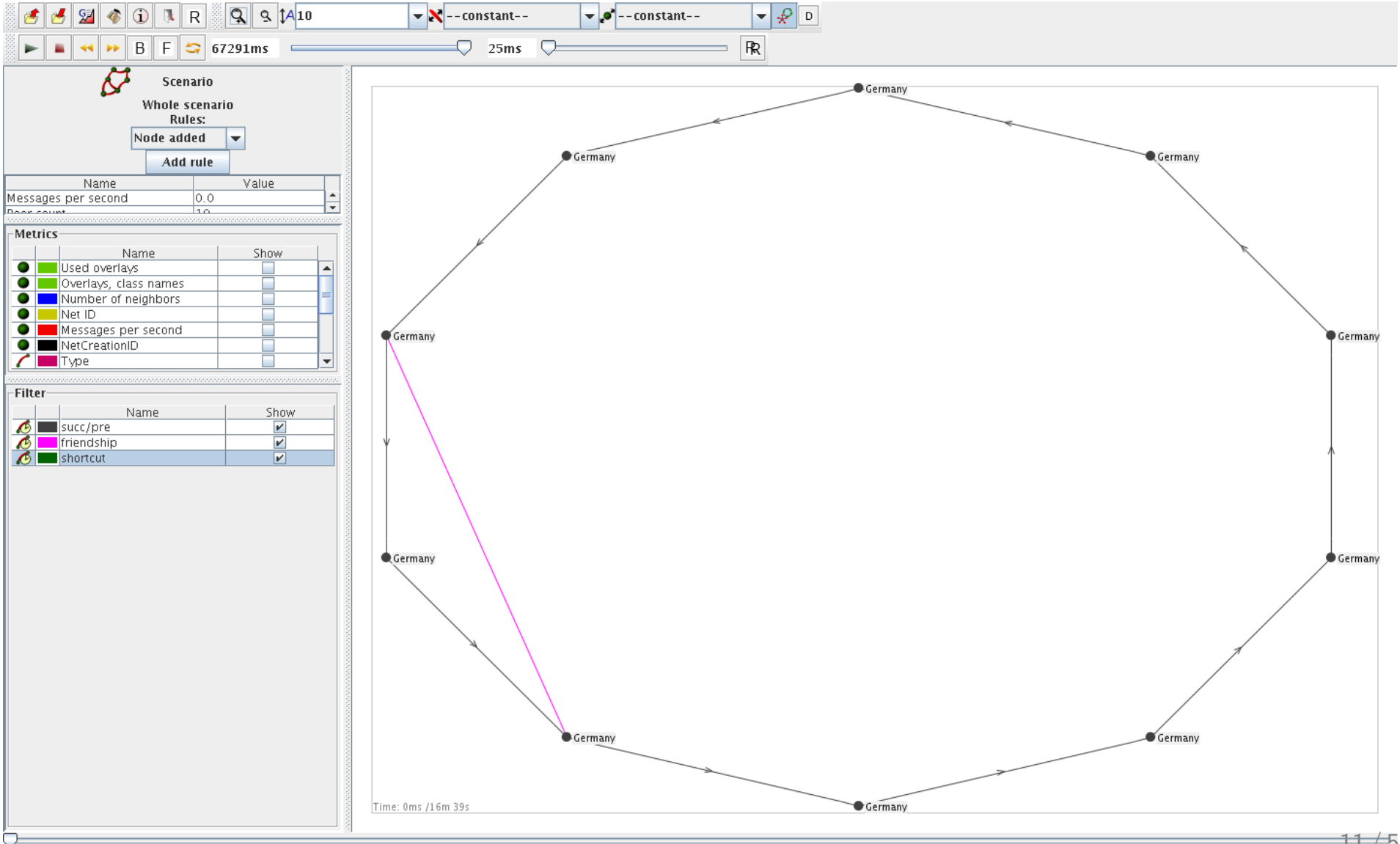
PeerfactSim.KOM

A simulation framework for large scale peer-to-peer systems.

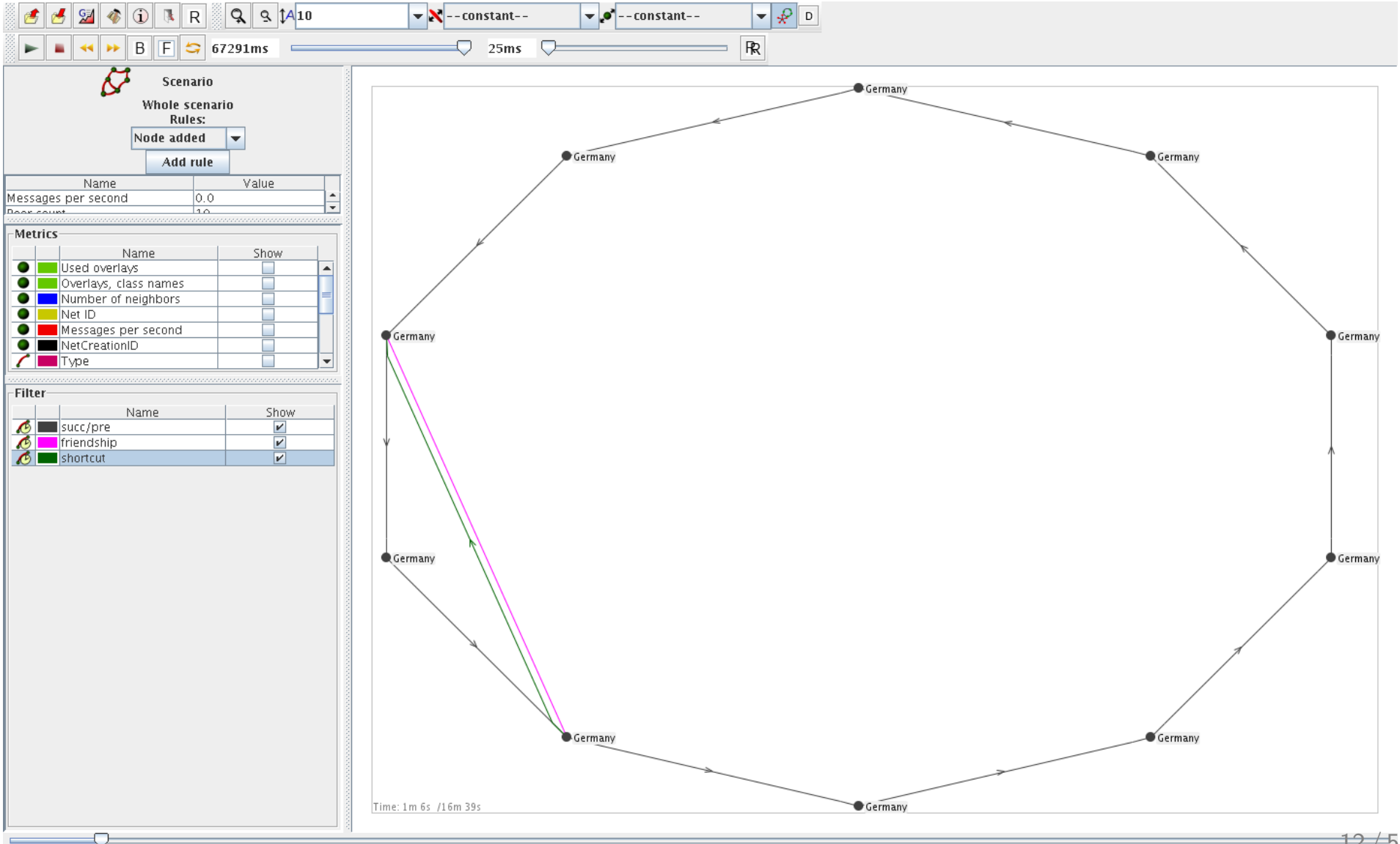
Extended by:

- Network creation games
- Content adressable network (CAN)

Current Progress
Simulator

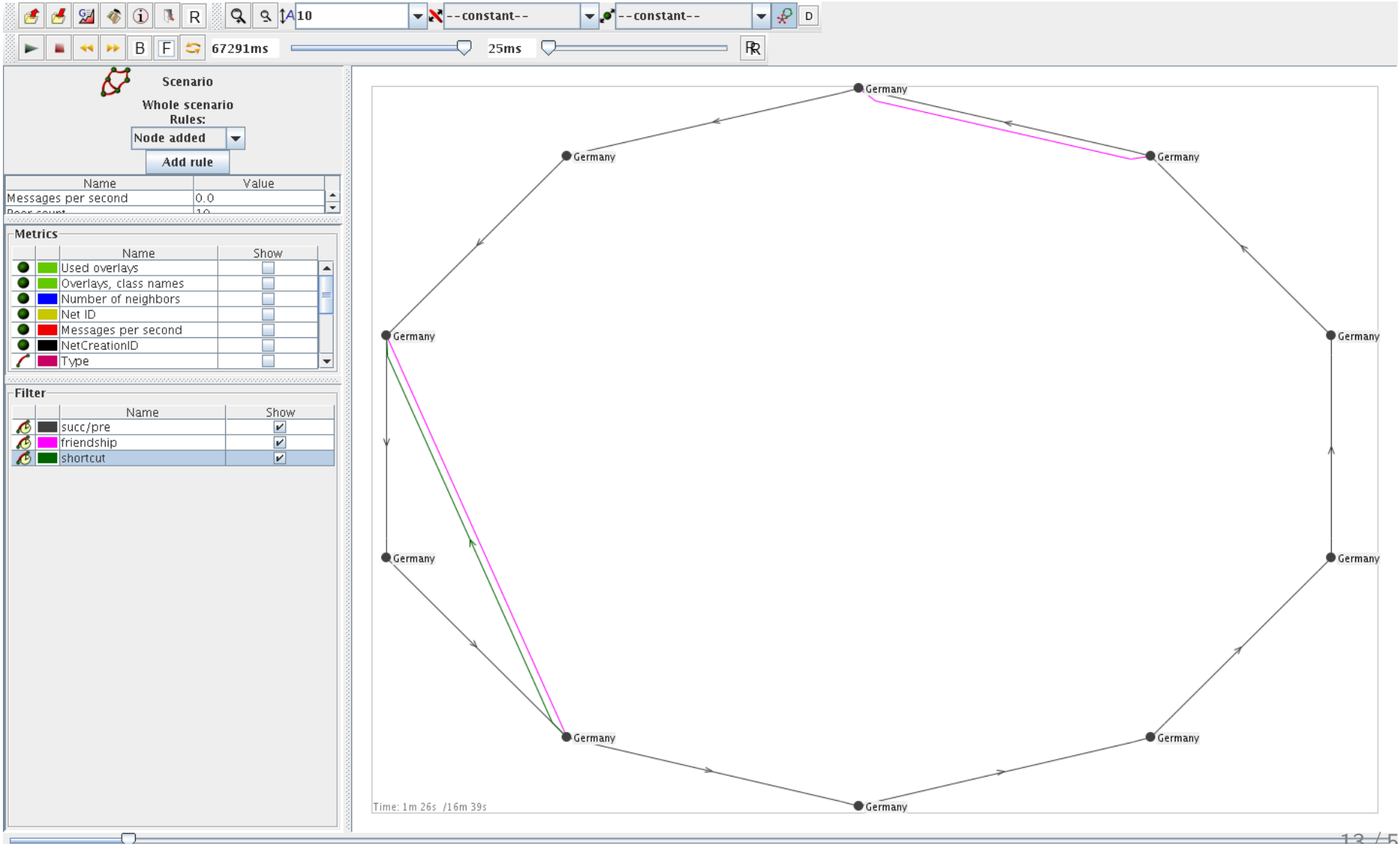


Current Progress
Simulator

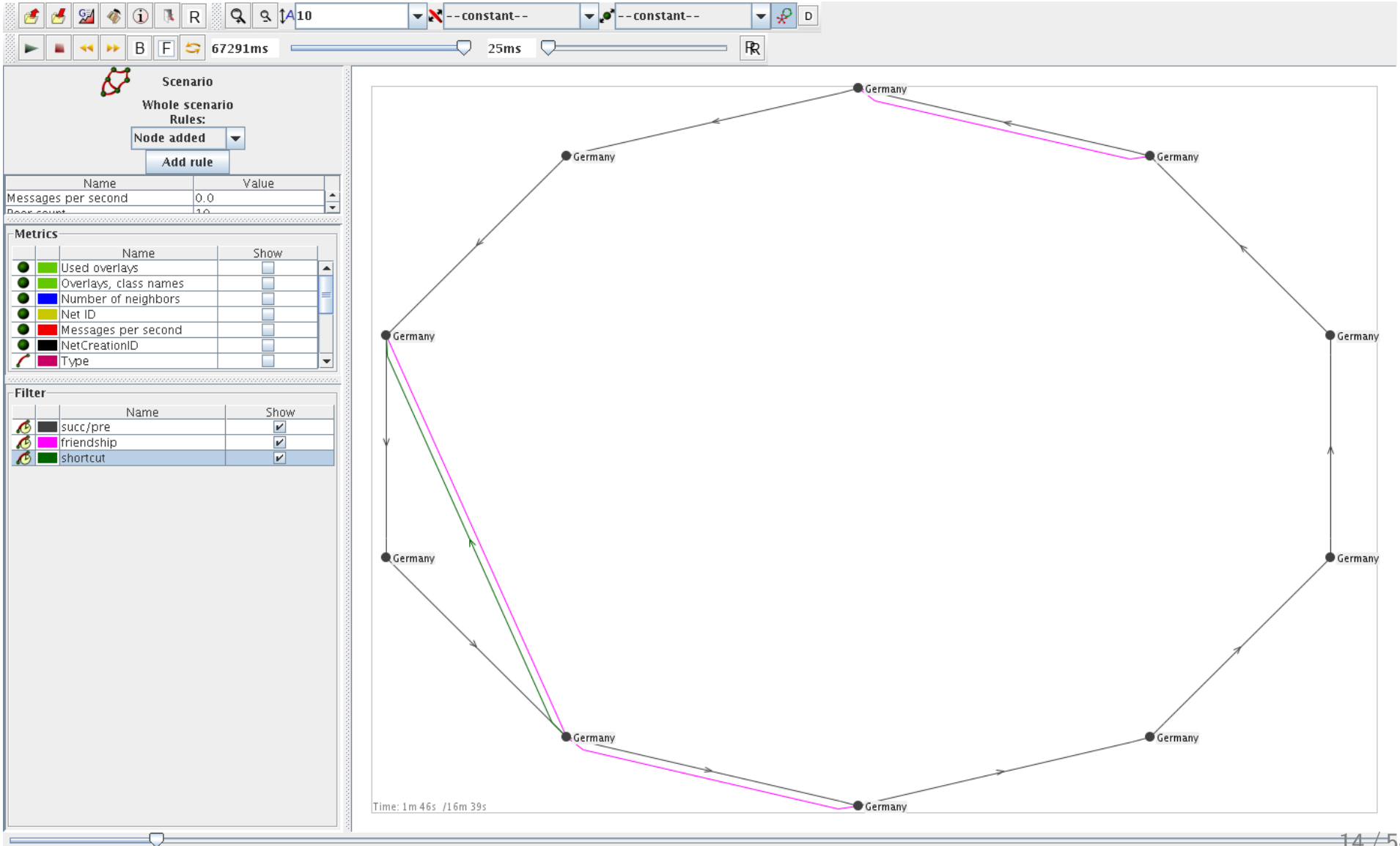


Current Progress

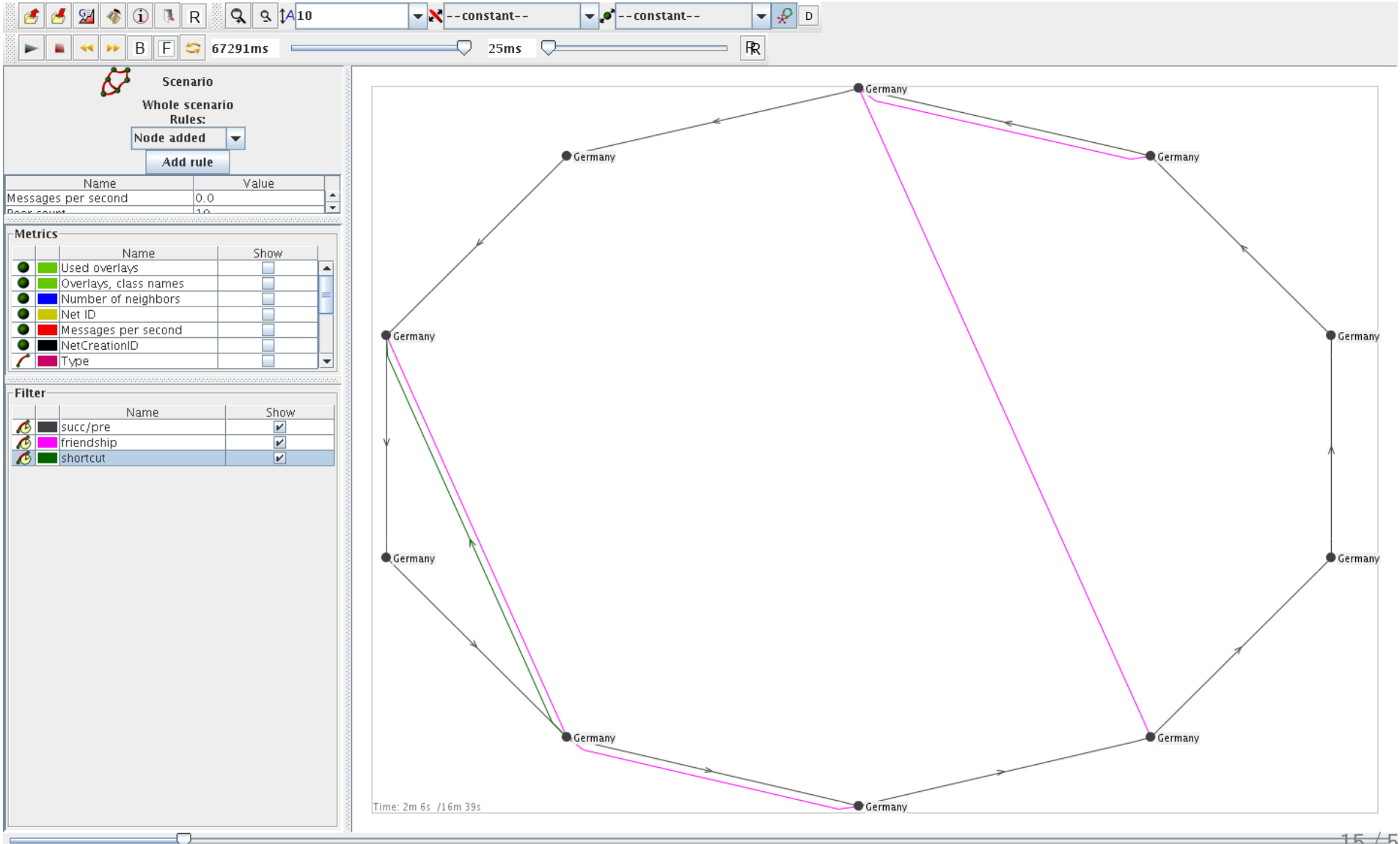
Simulator



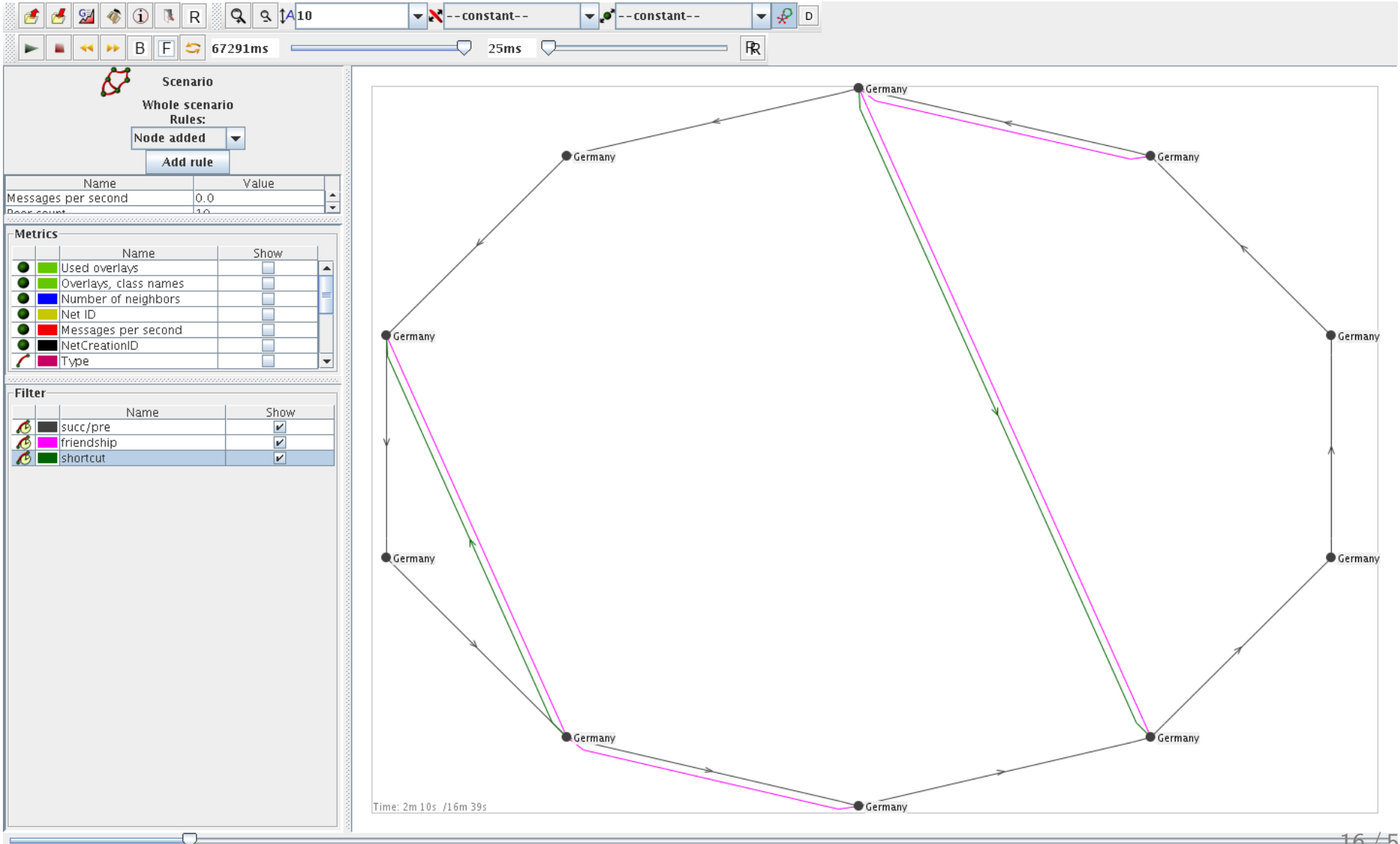
Current Progress
Simulator



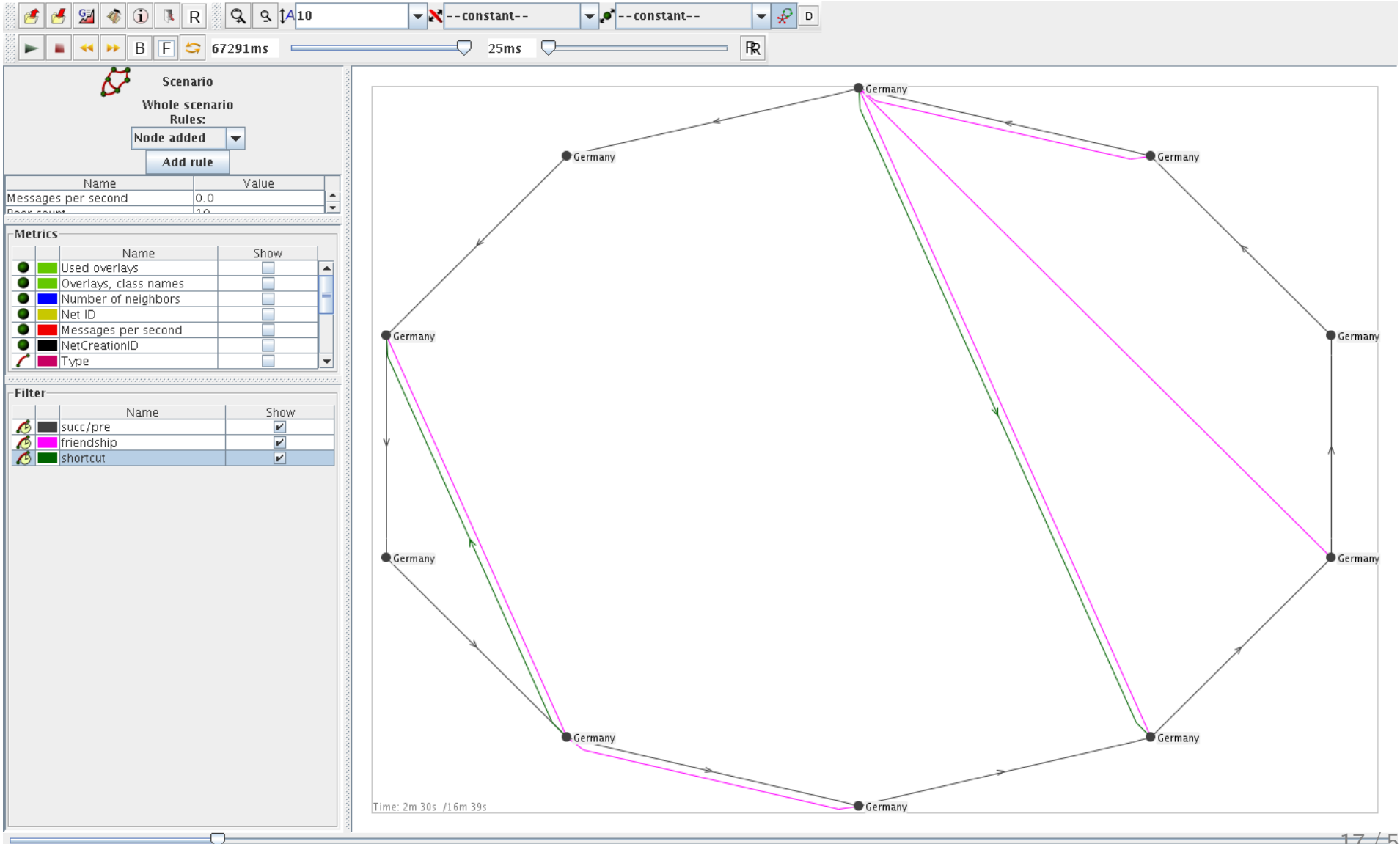
Current Progress
Simulator



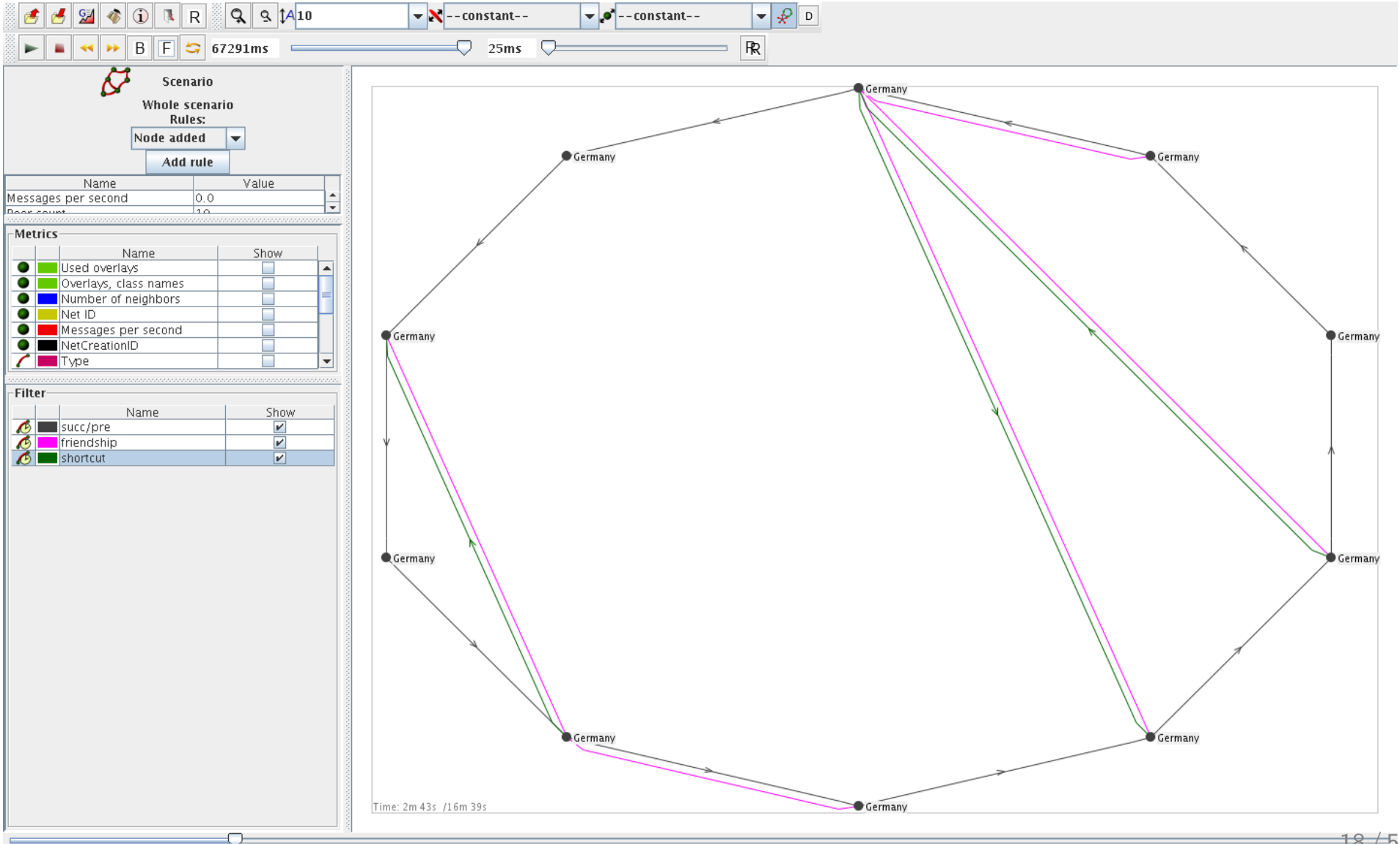
Current Progress
Simulator



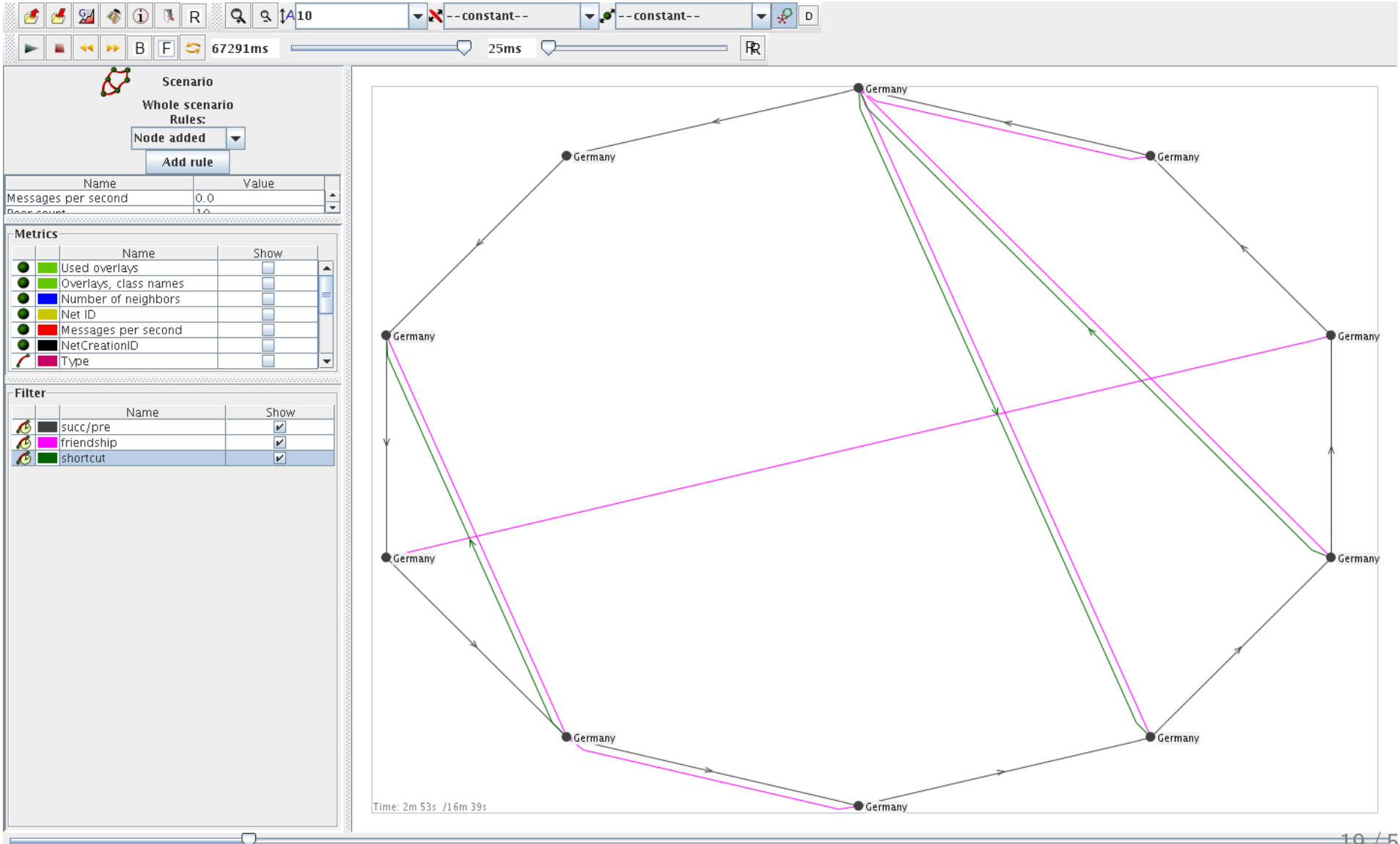
Current Progress
Simulator



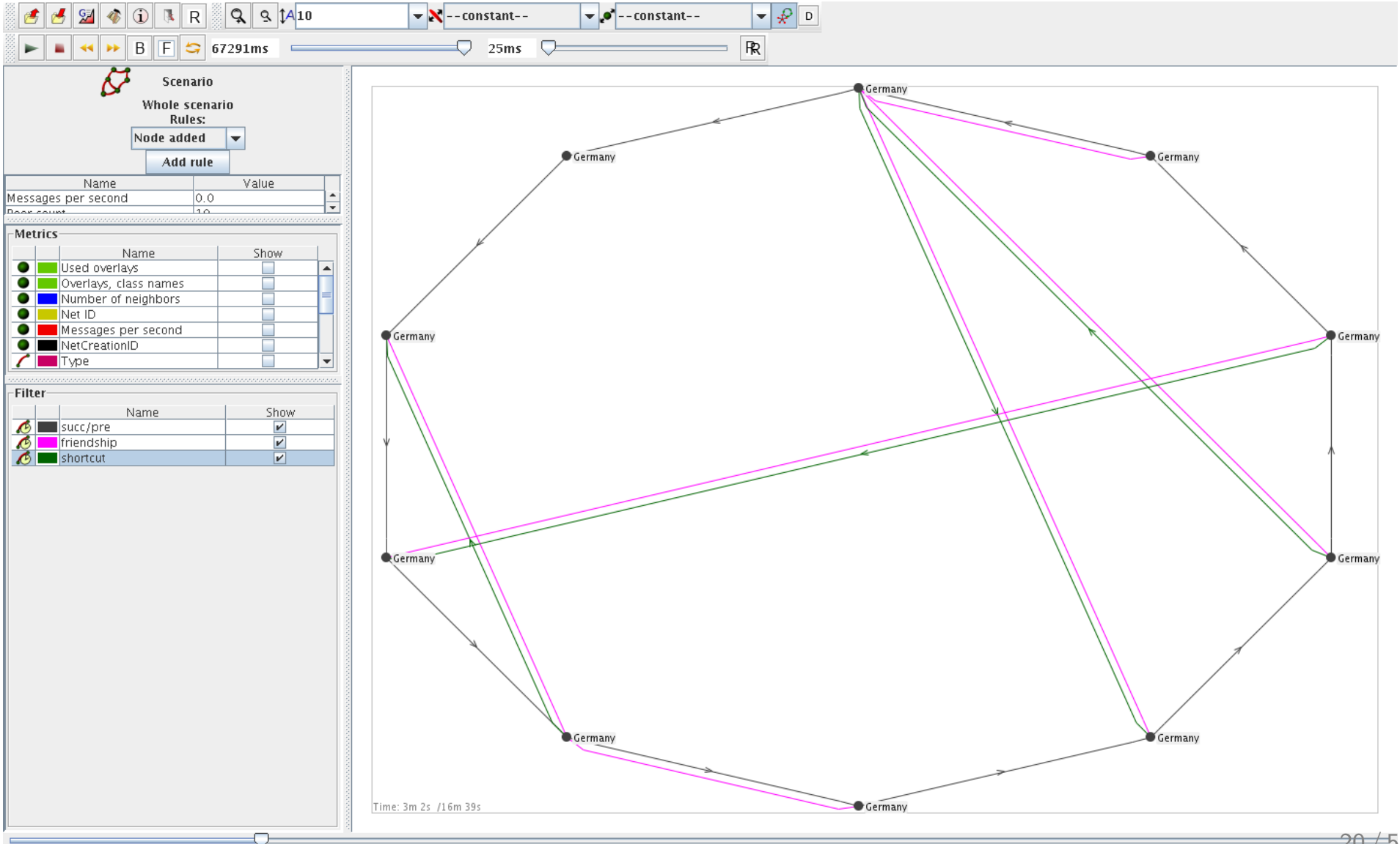
Current Progress
Simulator



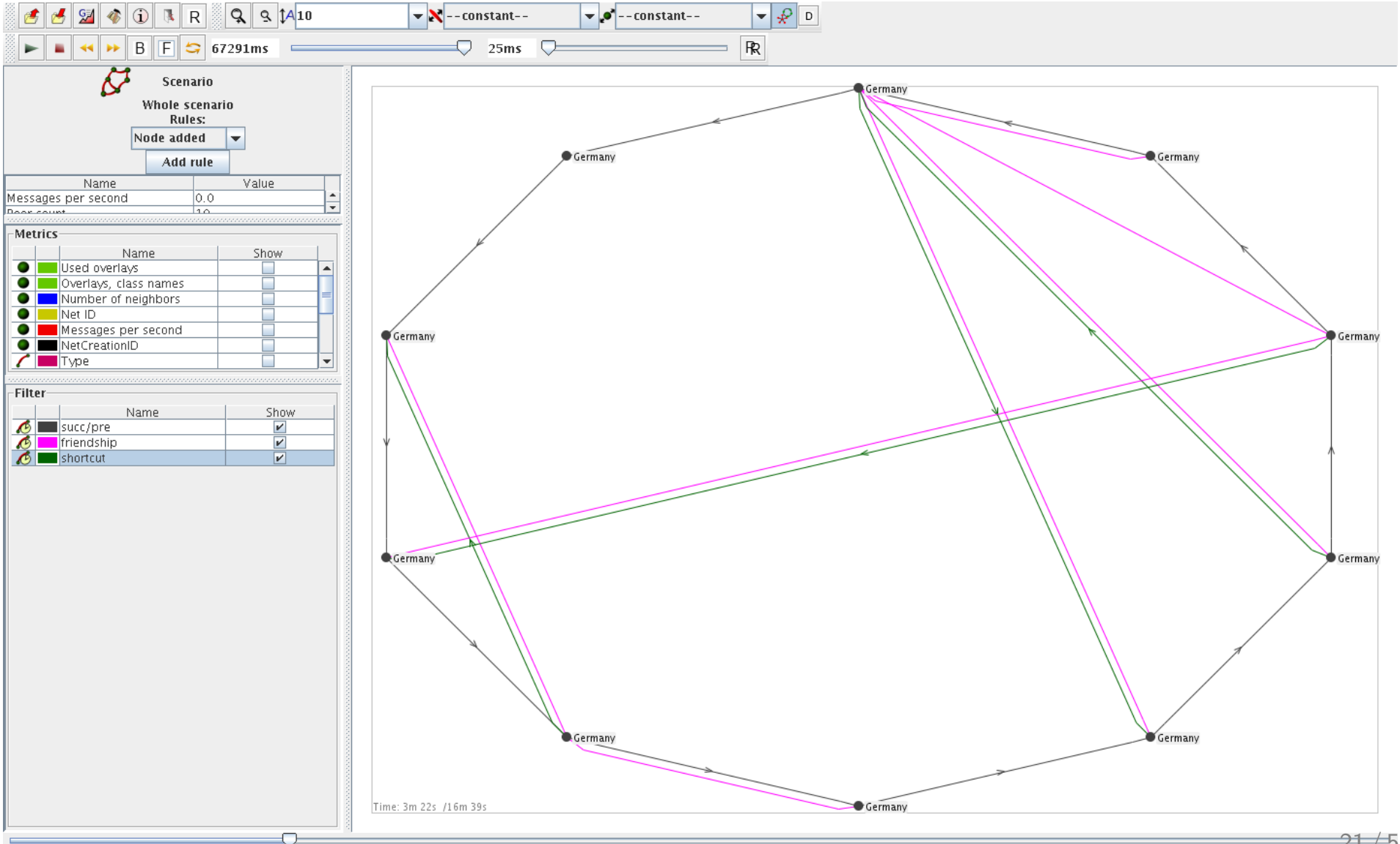
Current Progress
Simulator



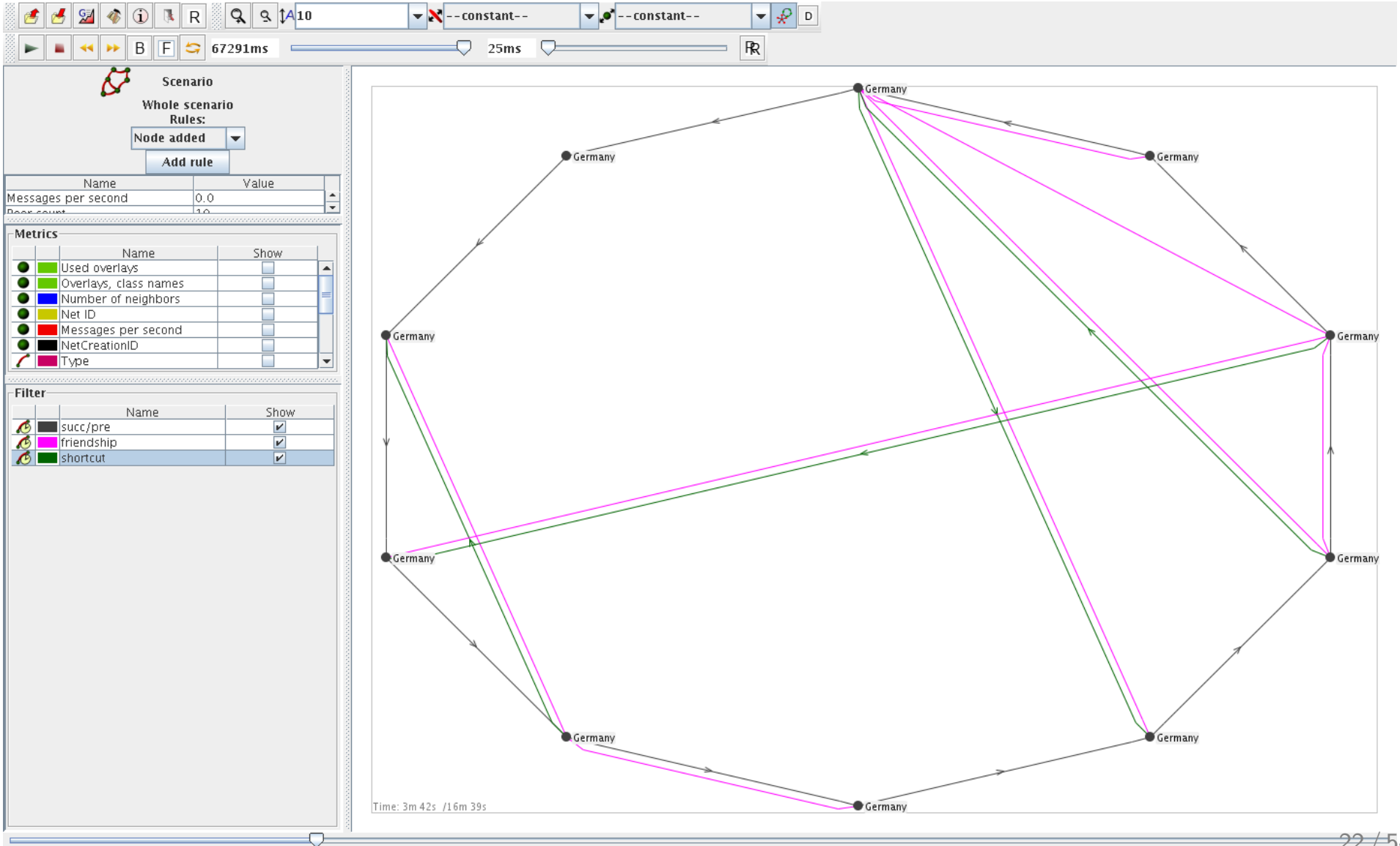
Current Progress
Simulator



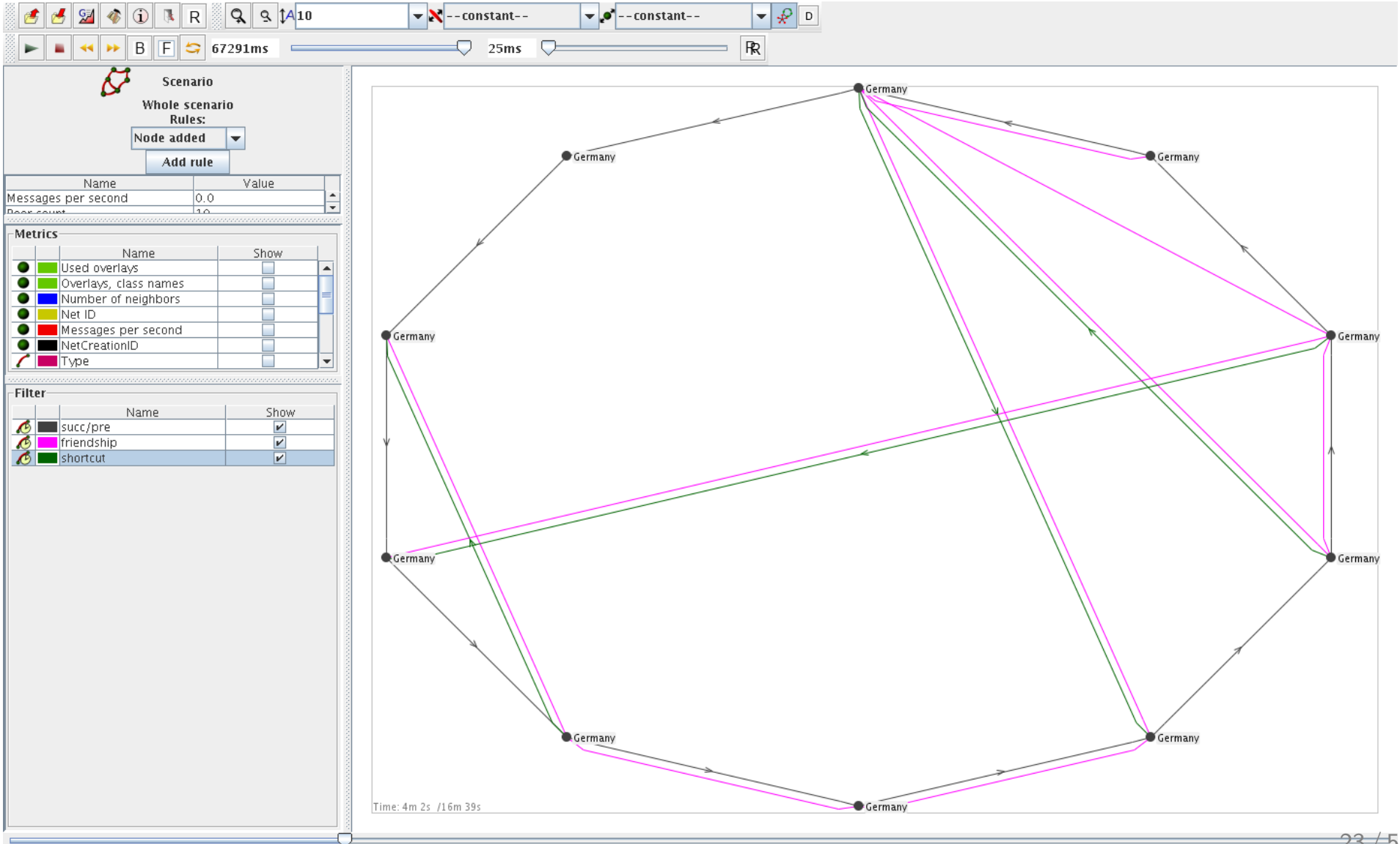
Current Progress
Simulator



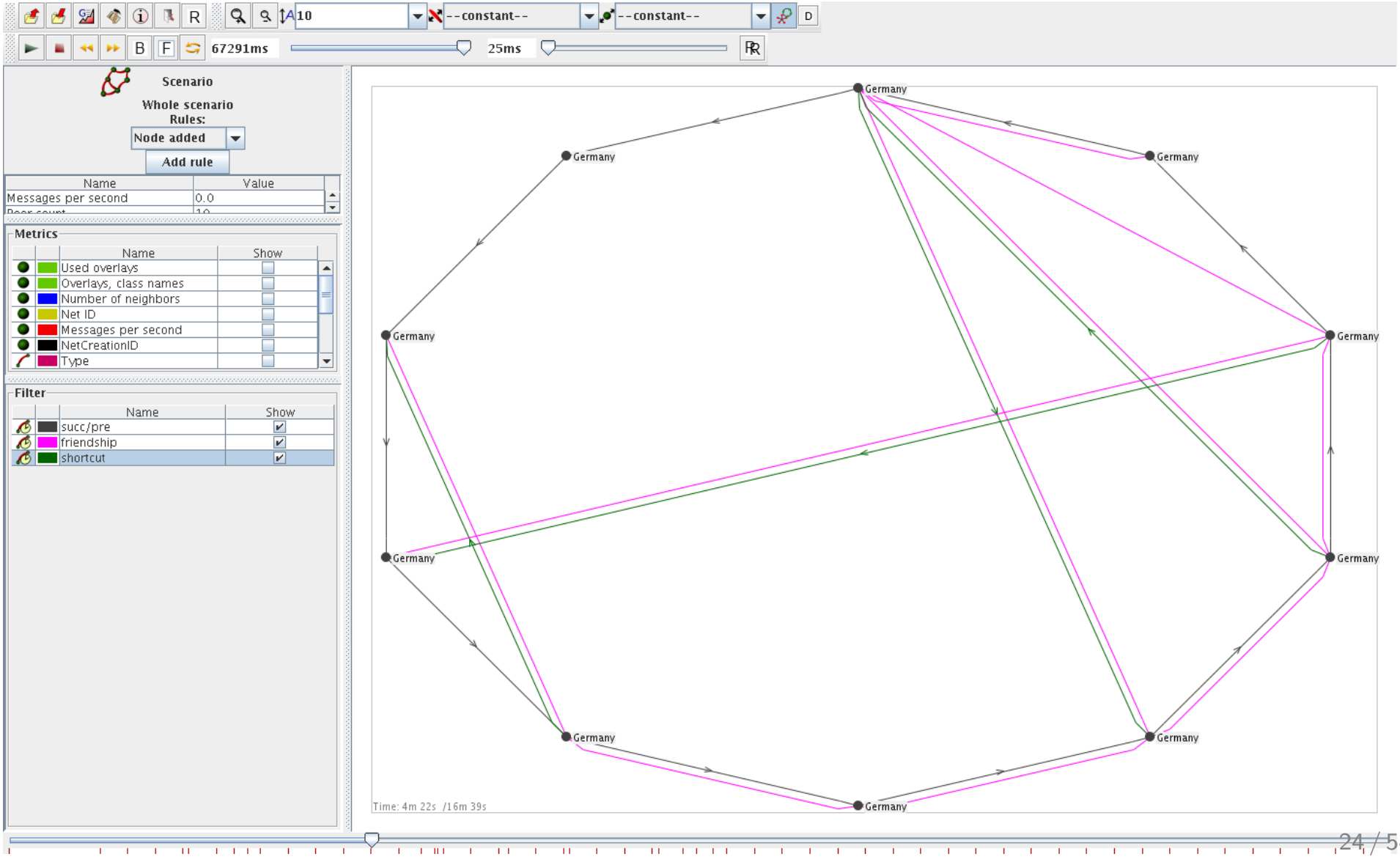
Current Progress
Simulator



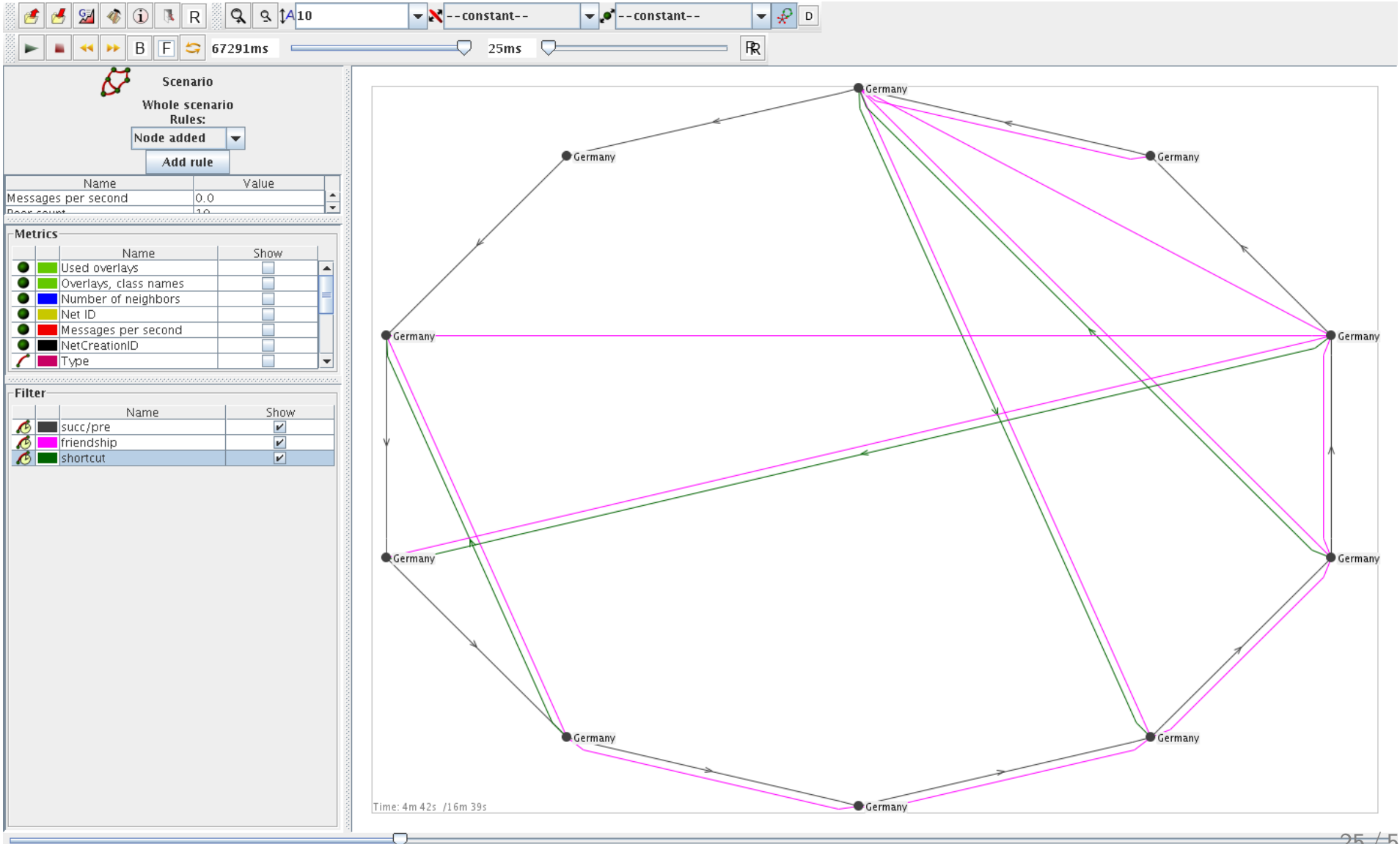
Current Progress
Simulator



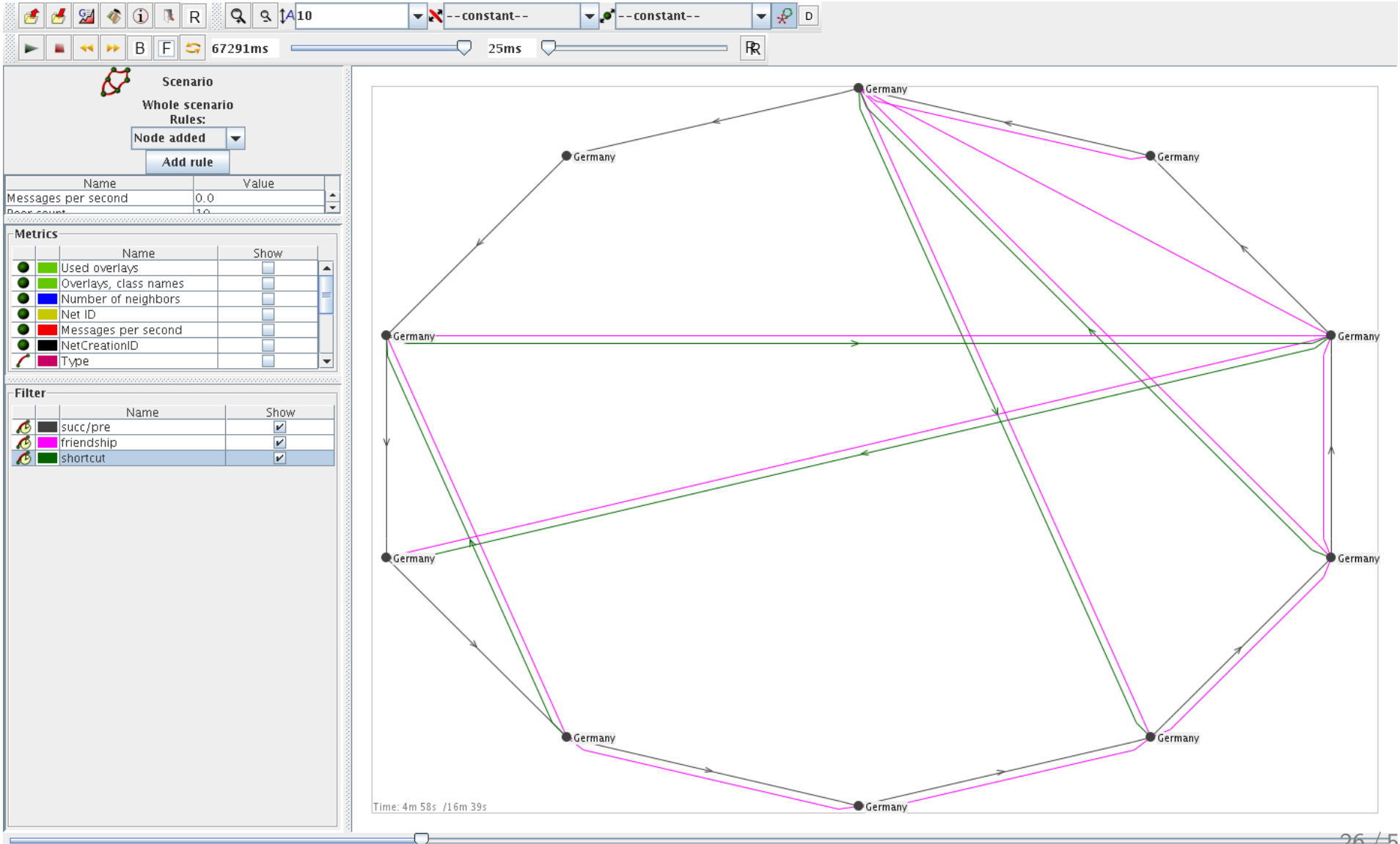
Current Progress
Simulator



Current Progress
Simulator

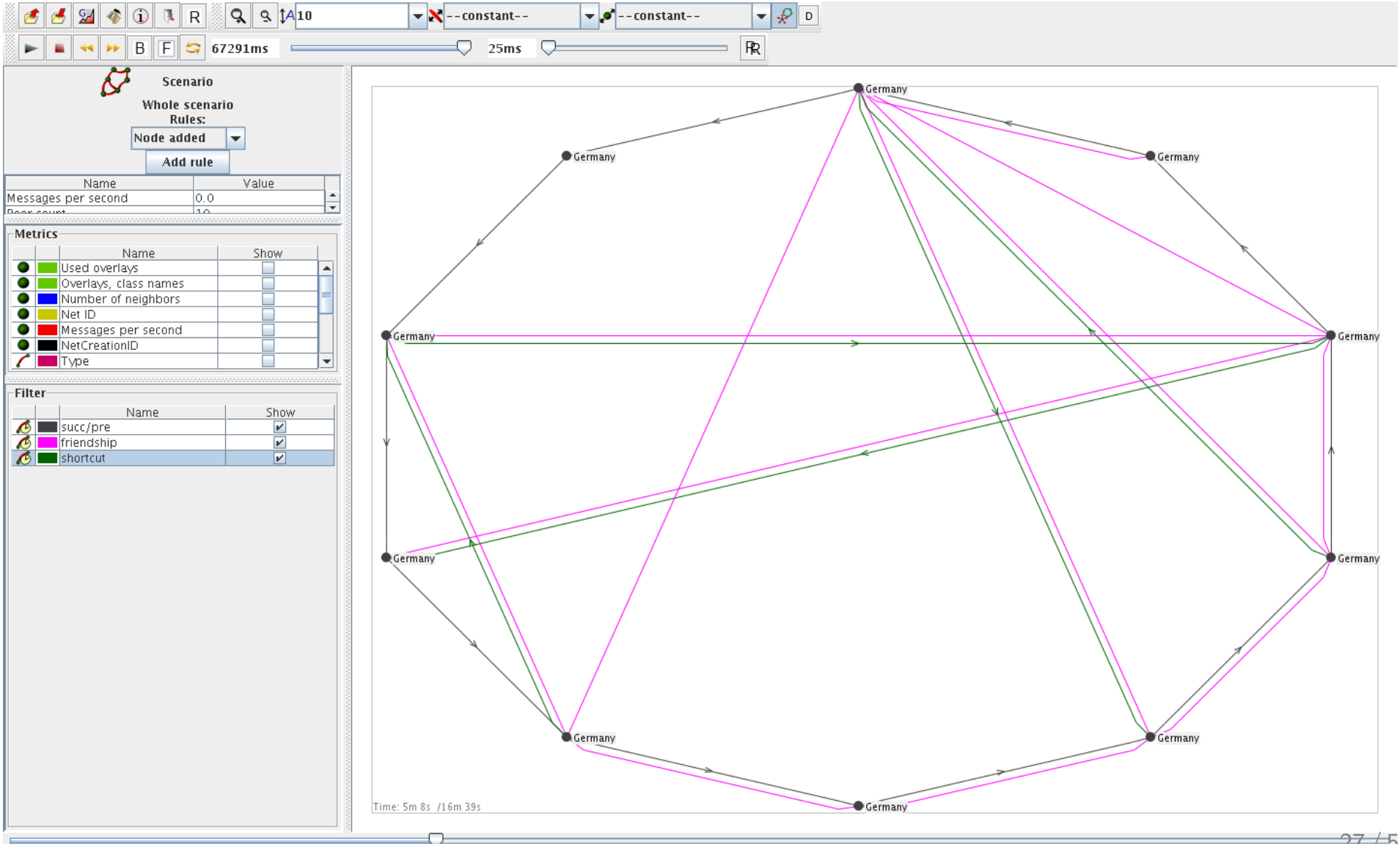


Current Progress
Simulator

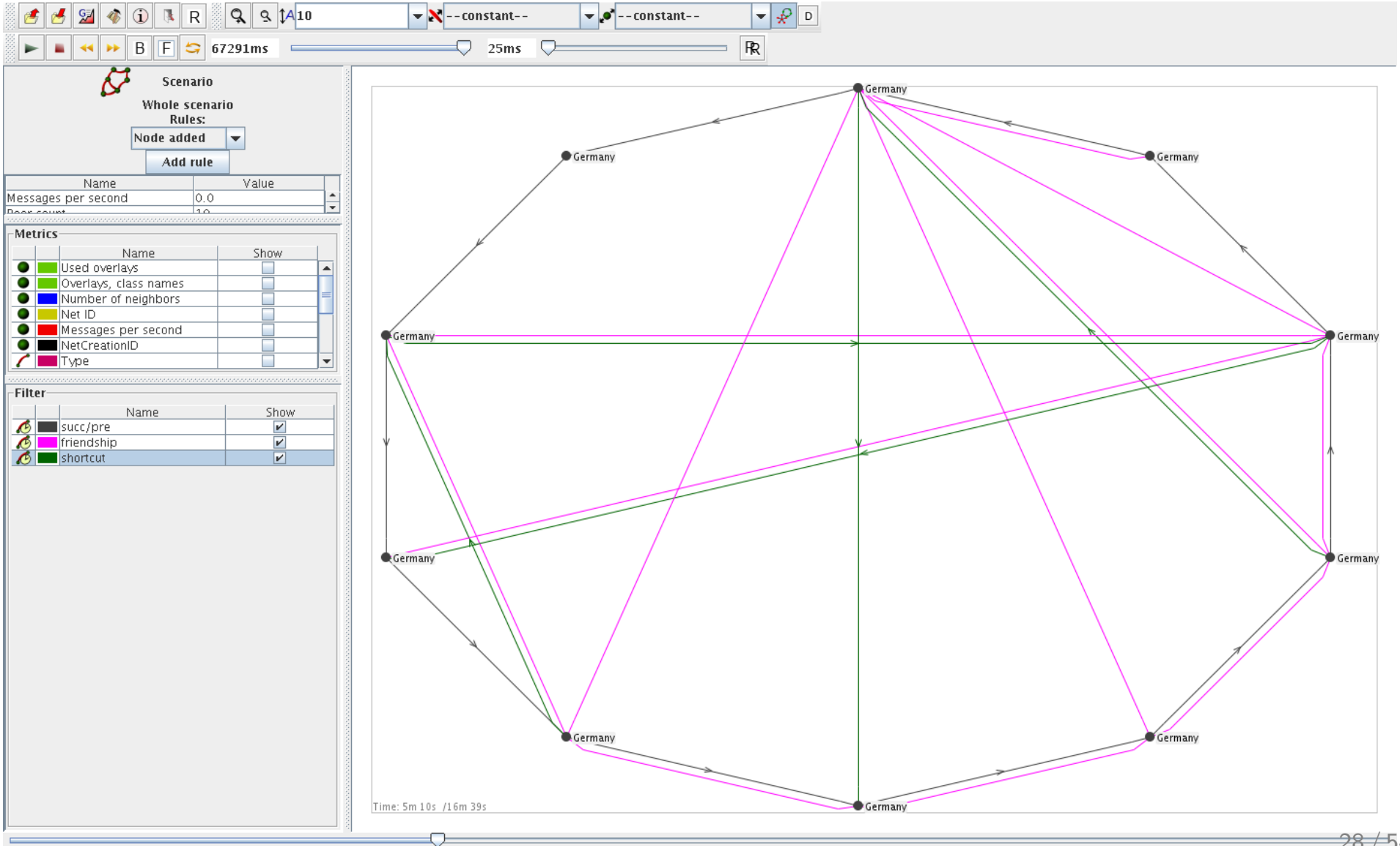


Current Progress

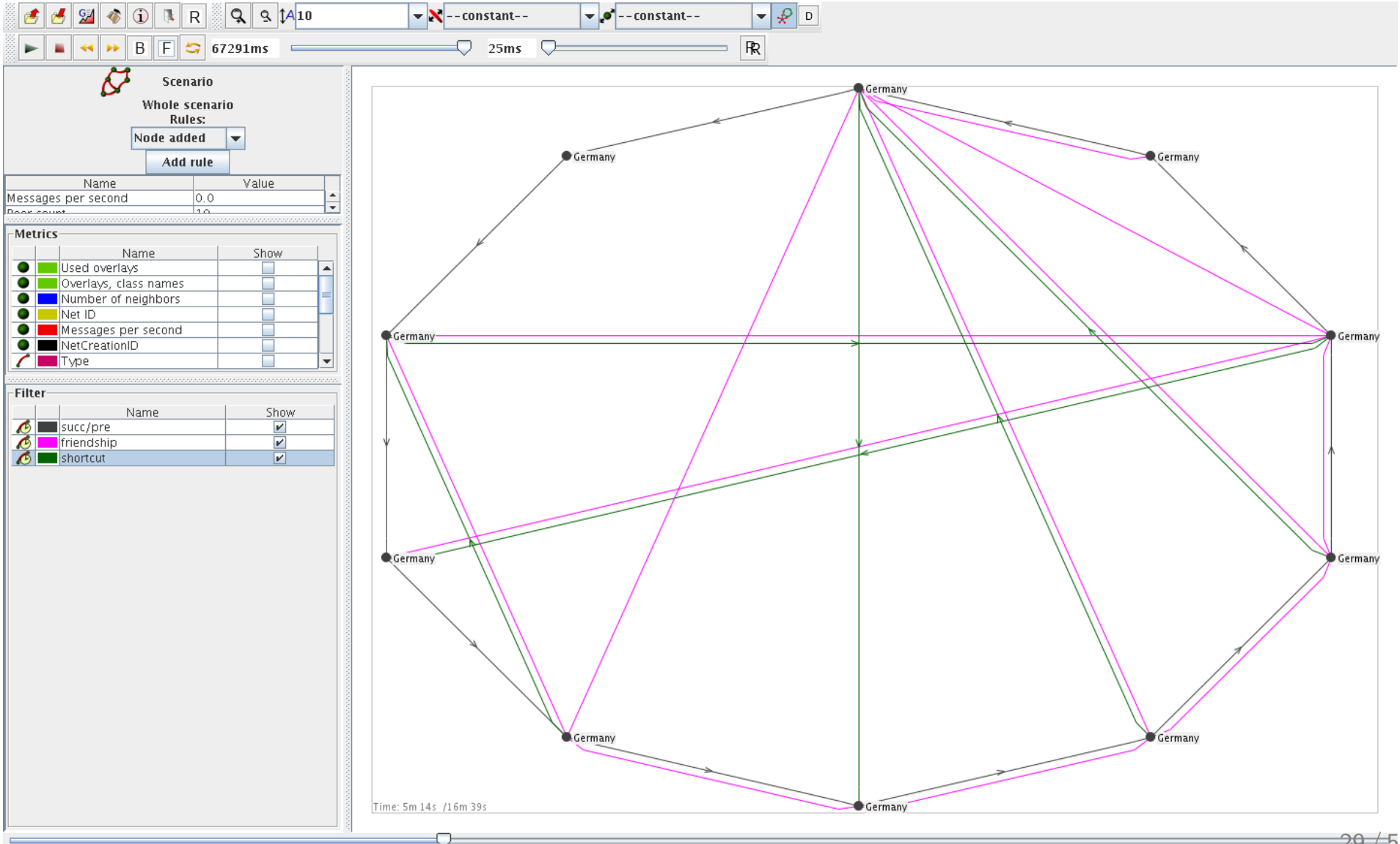
Simulator



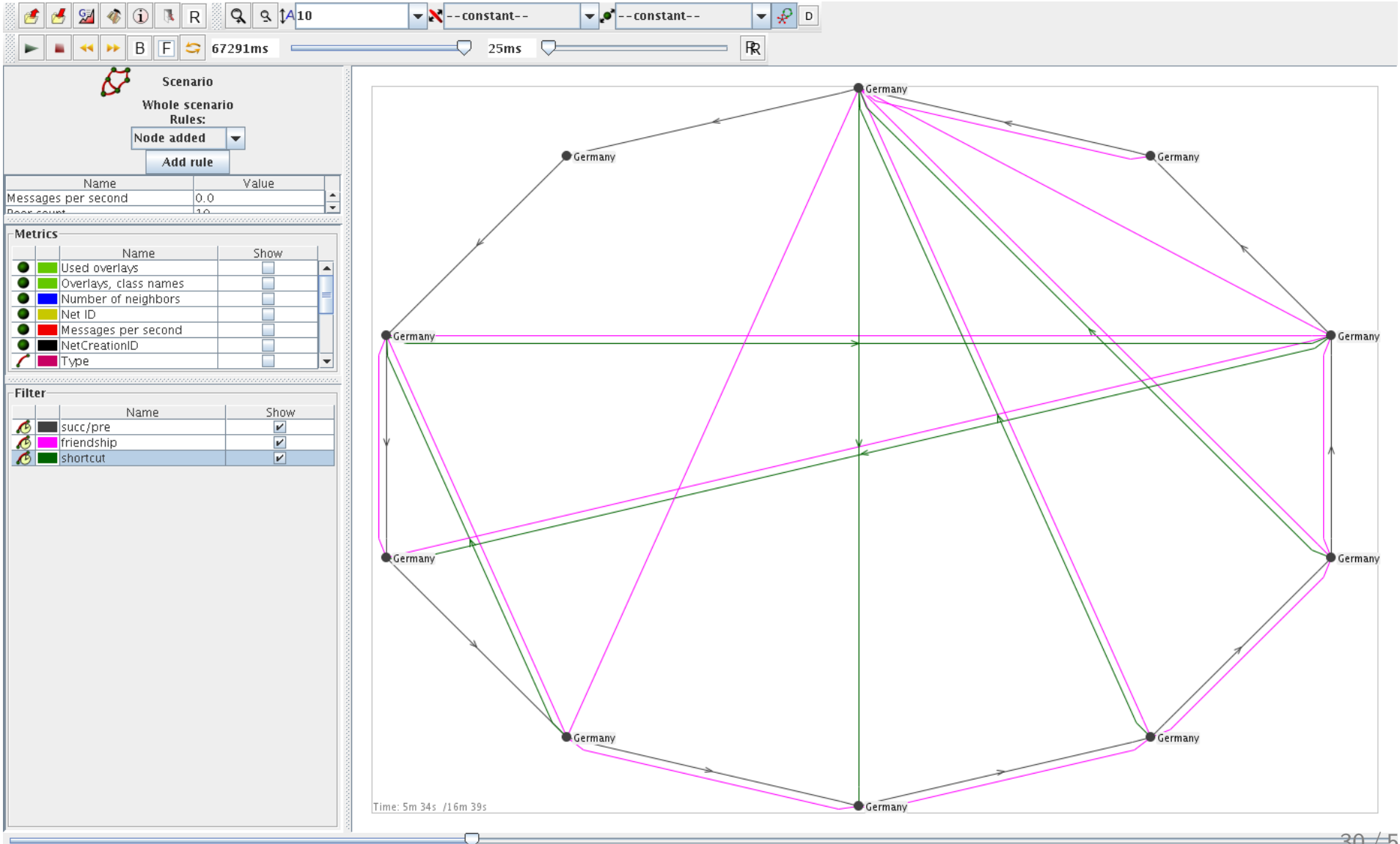
Current Progress
Simulator



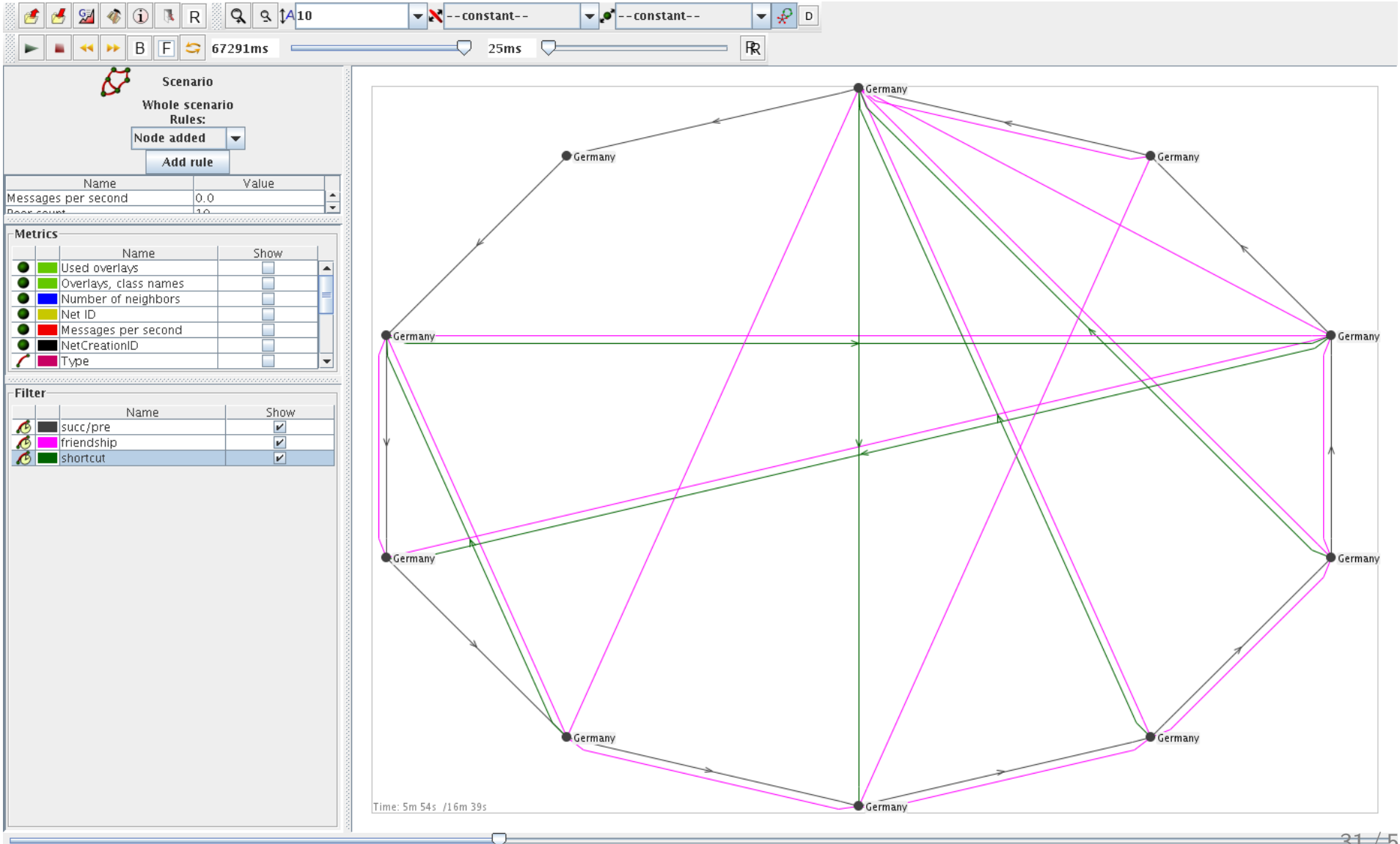
Current Progress
Simulator



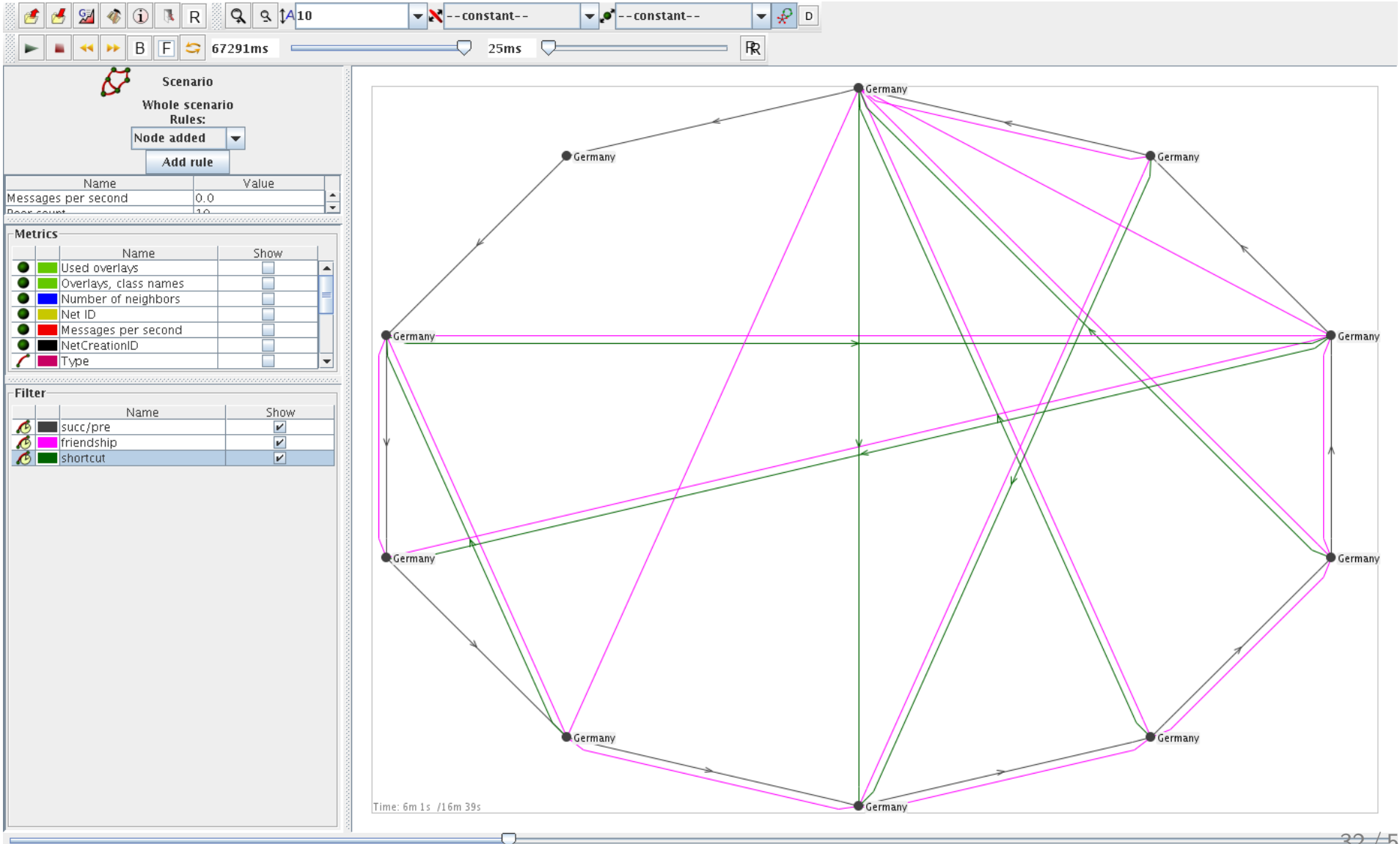
Current Progress
Simulator



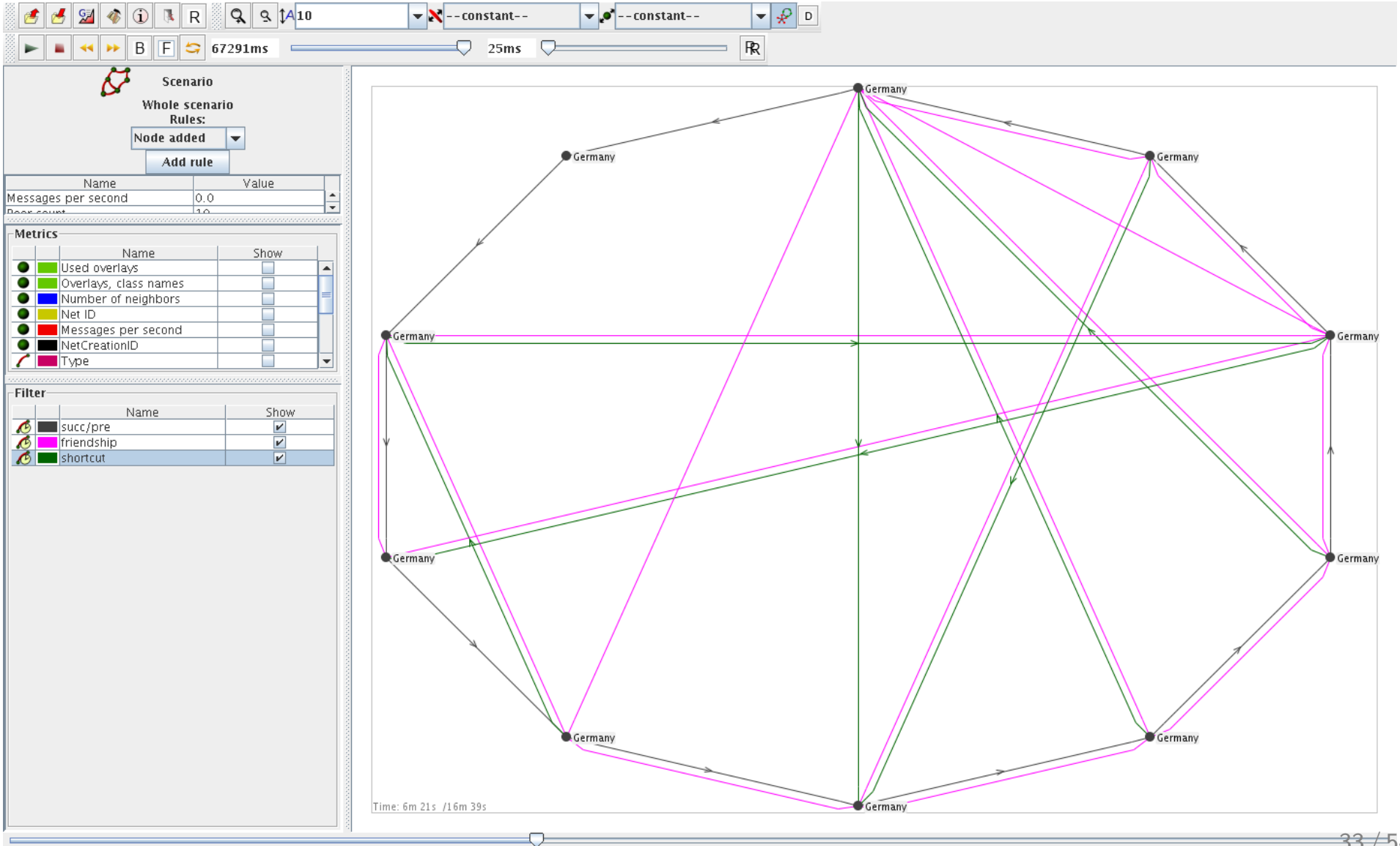
Current Progress
Simulator



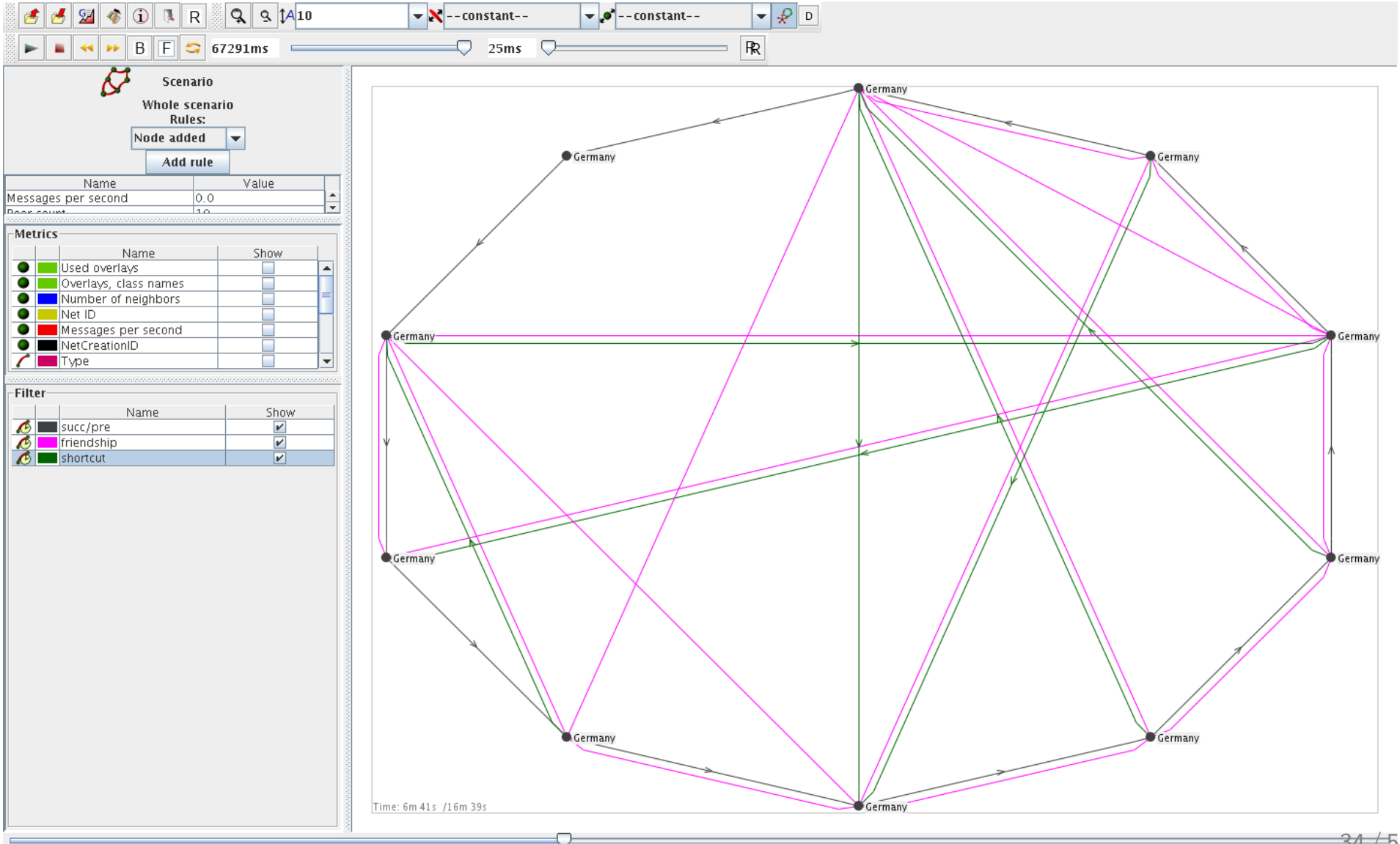
Current Progress
Simulator



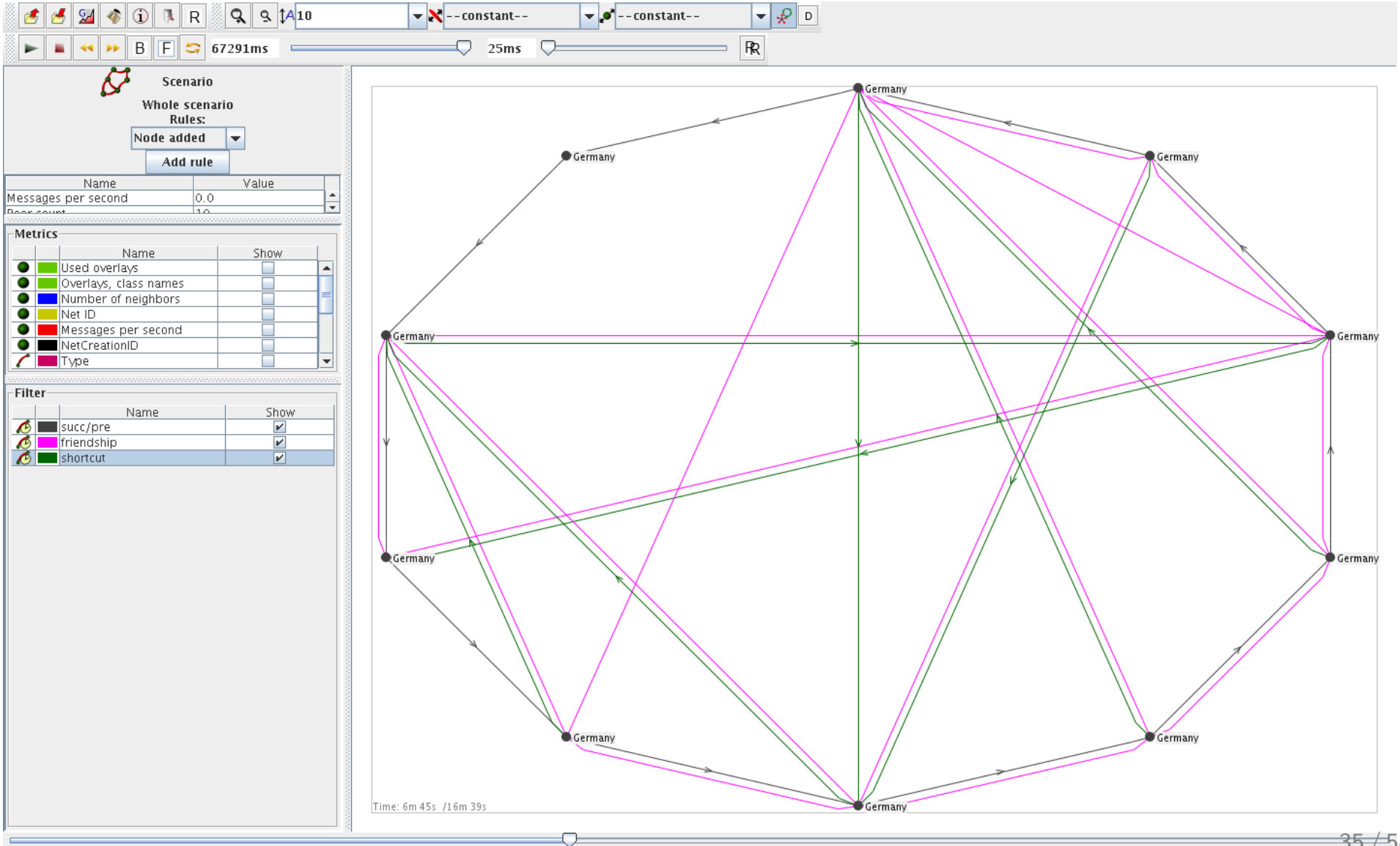
Current Progress
Simulator



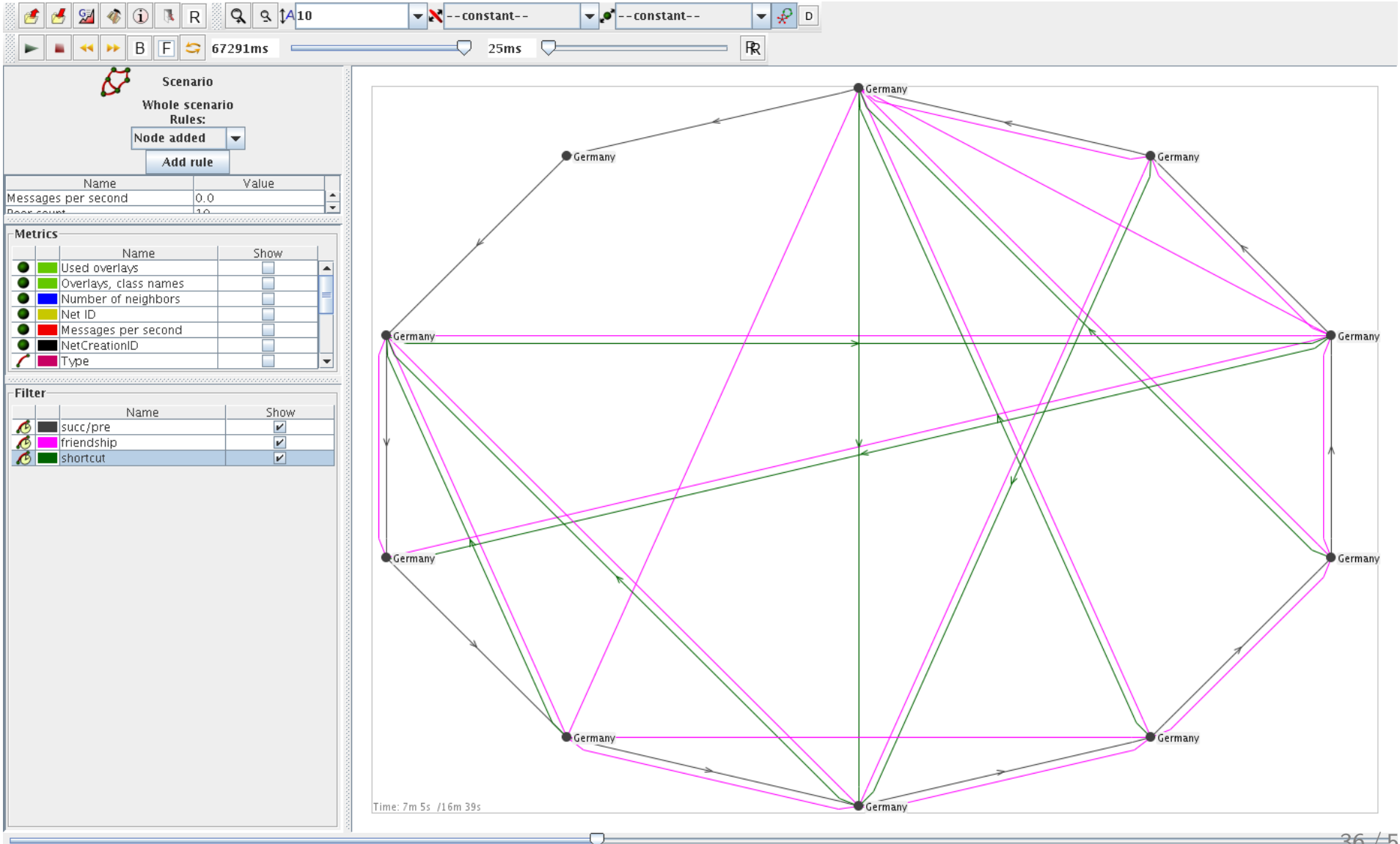
Current Progress
Simulator



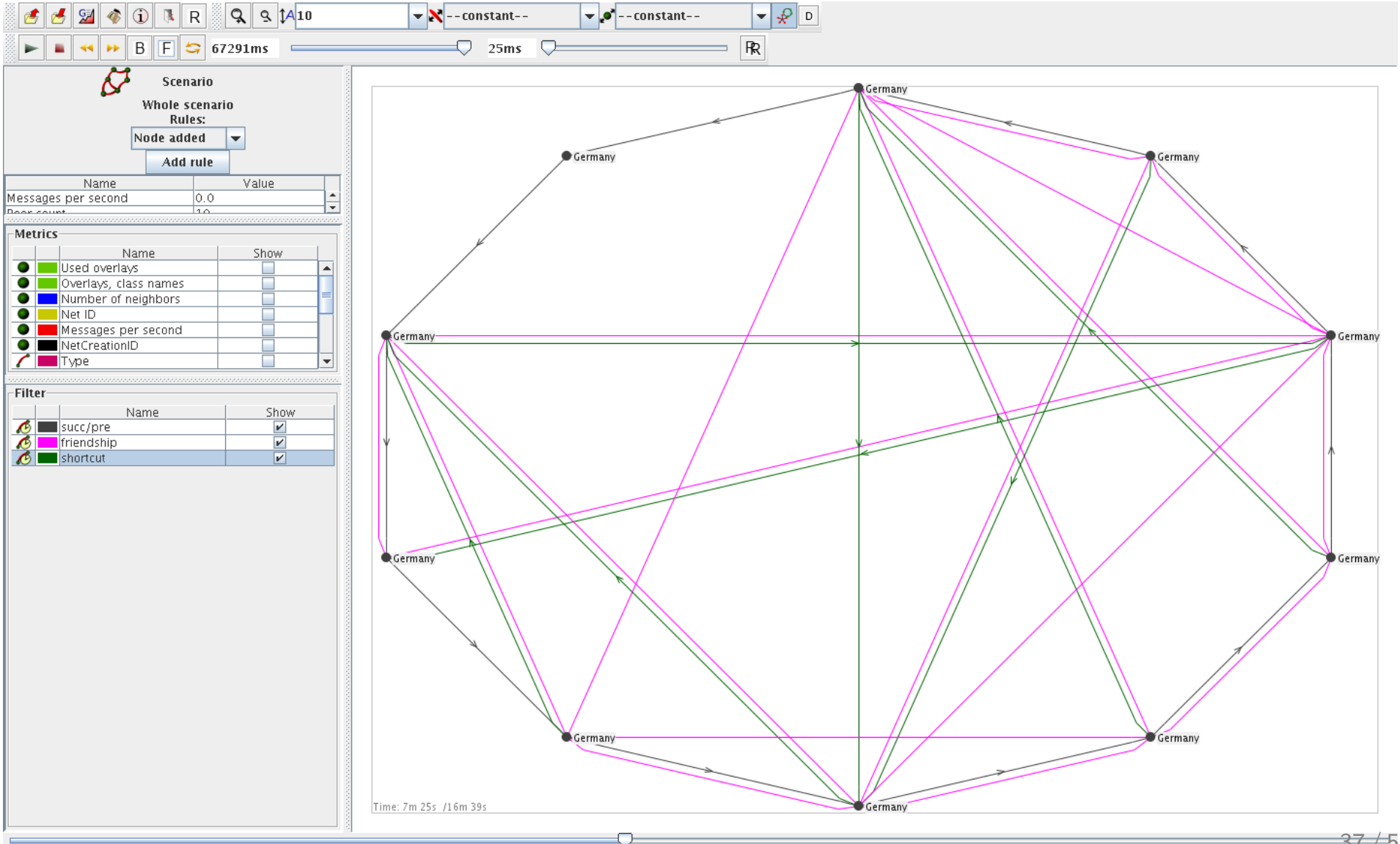
Current Progress
Simulator



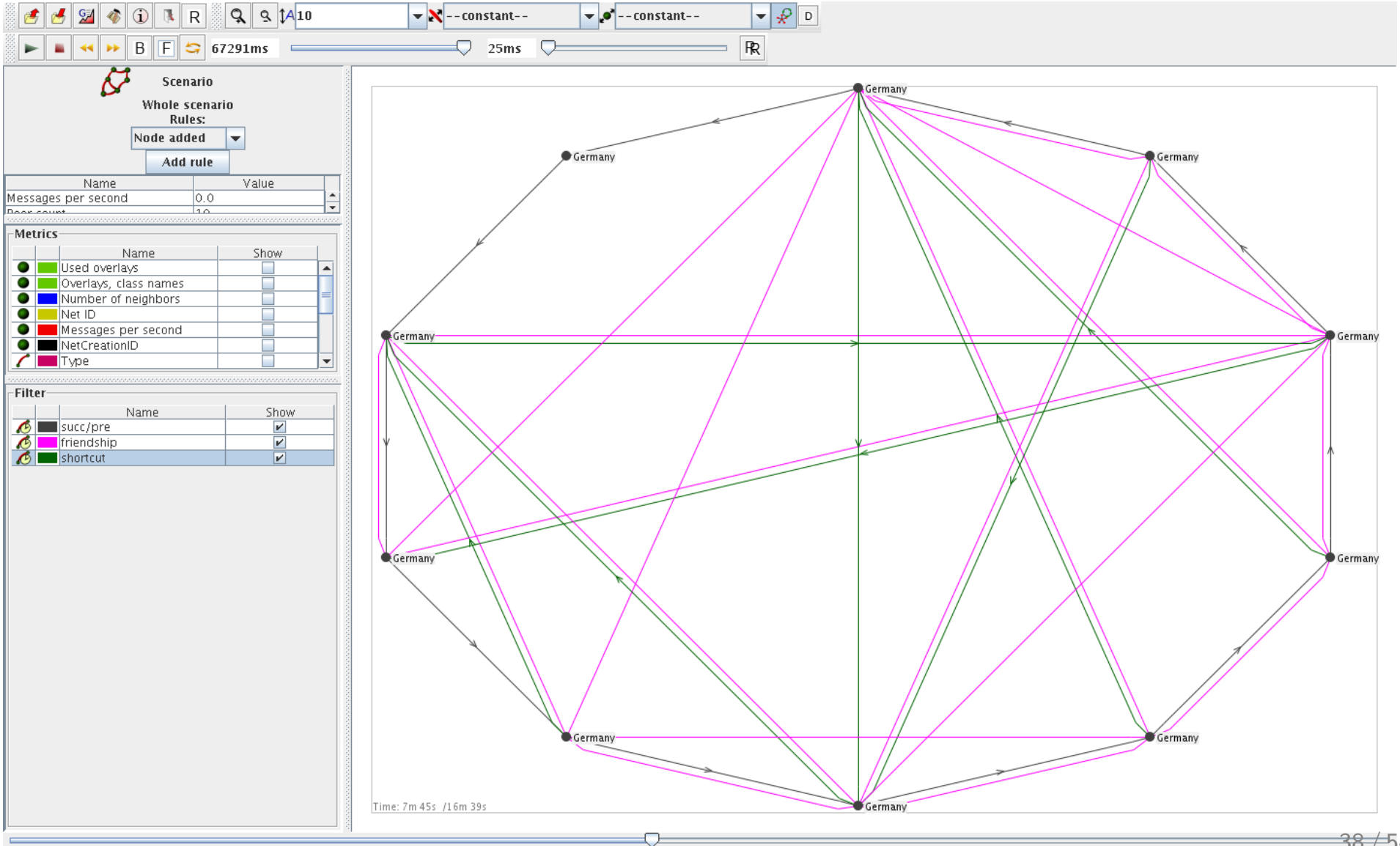
Current Progress
Simulator



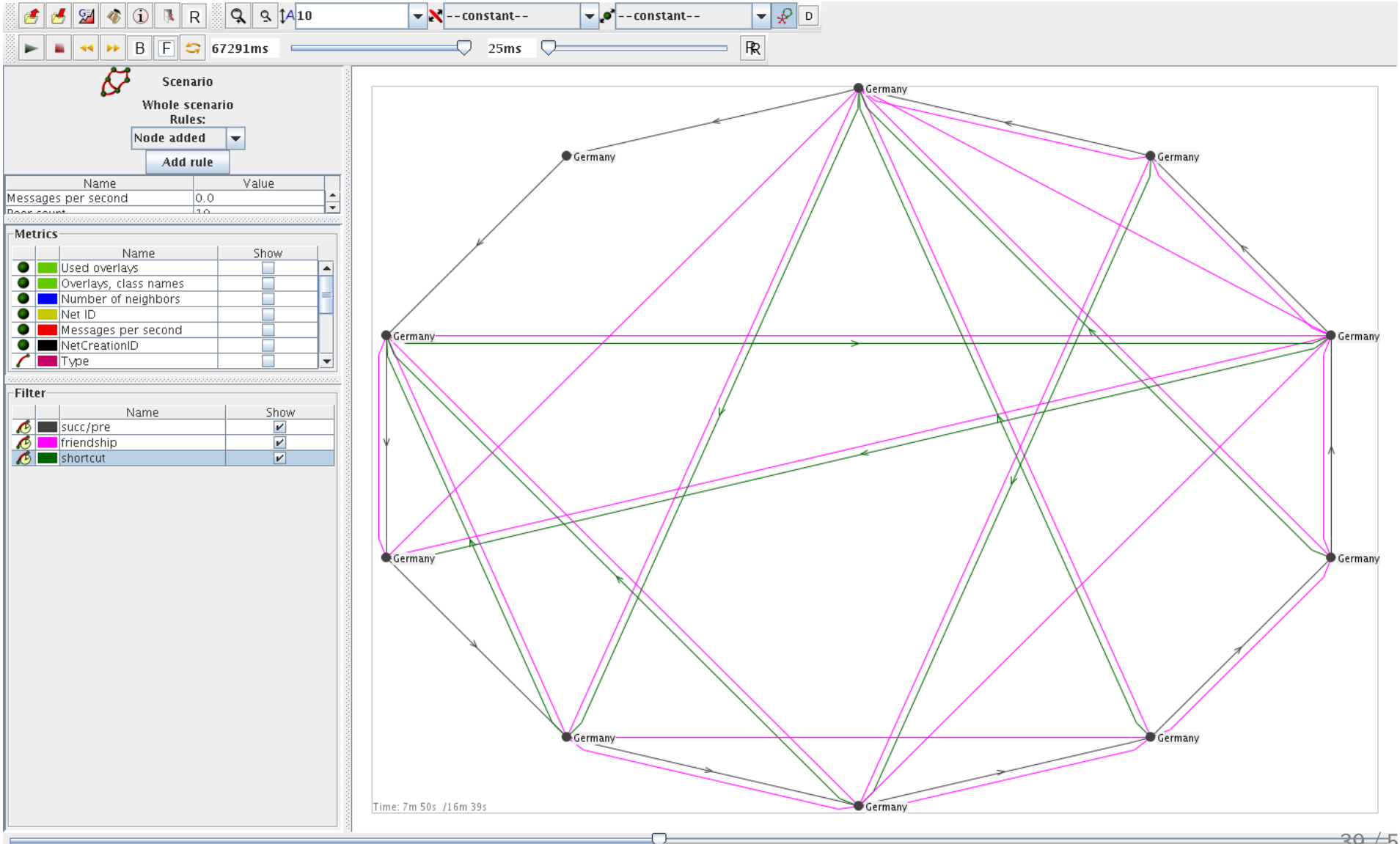
Current Progress
Simulator



Current Progress
Simulator



Current Progress
Simulator

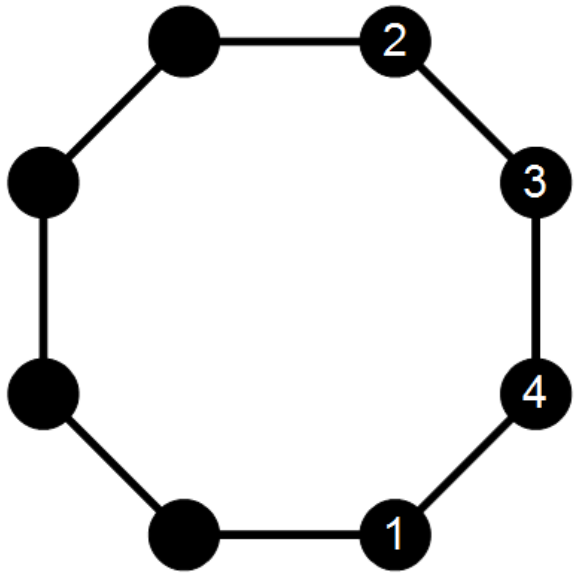


Current Progress

Looking for easier problems

Current Progress

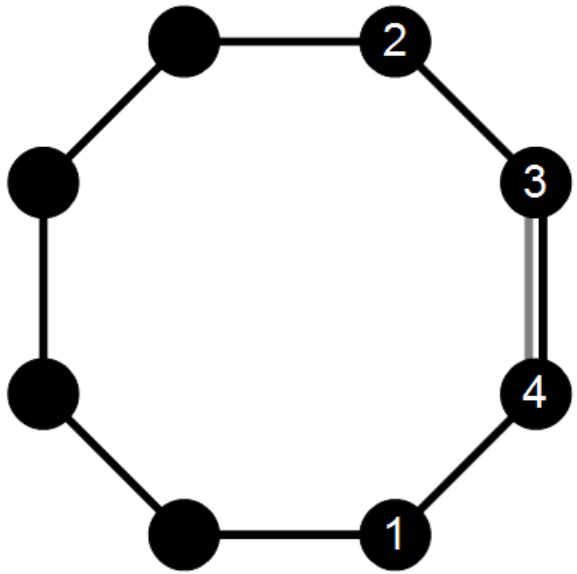
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

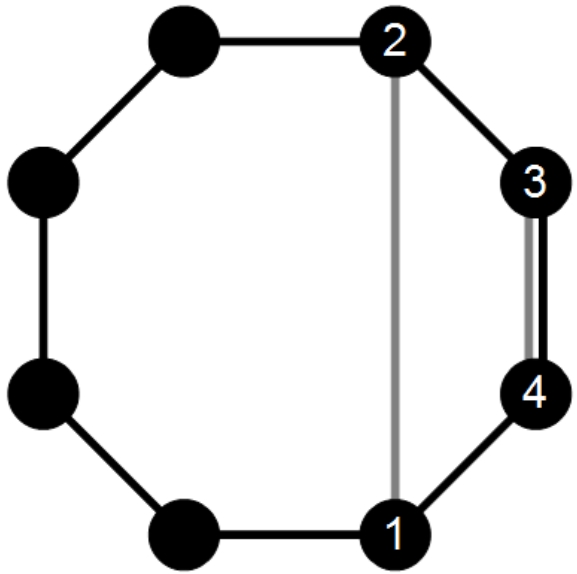
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

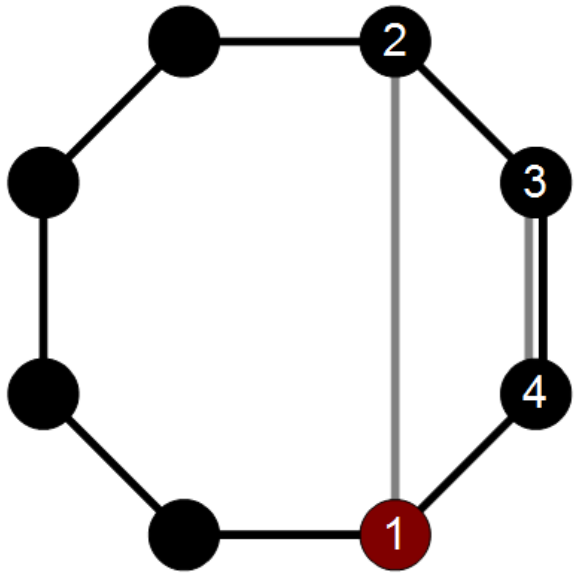
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

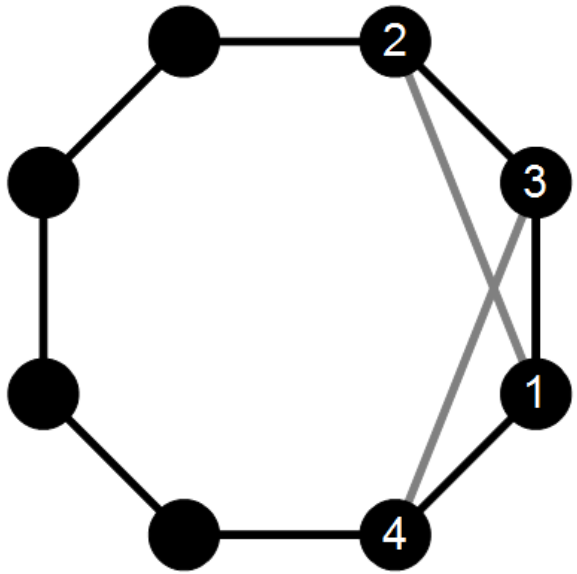
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

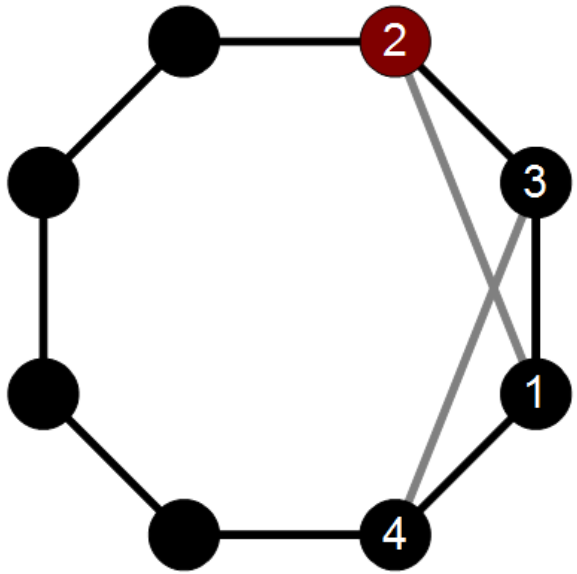
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

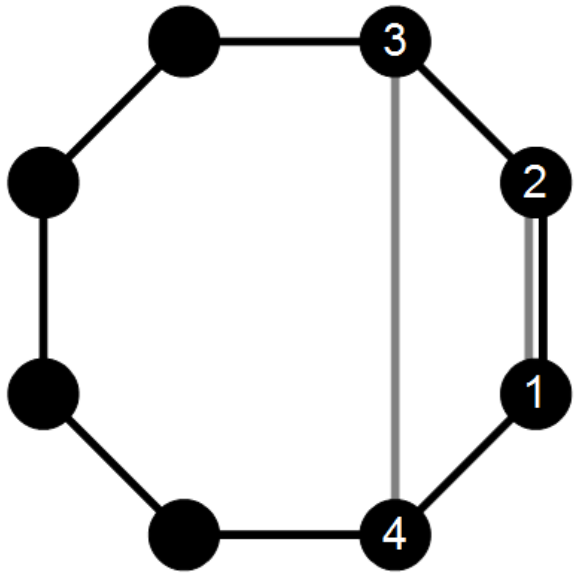
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

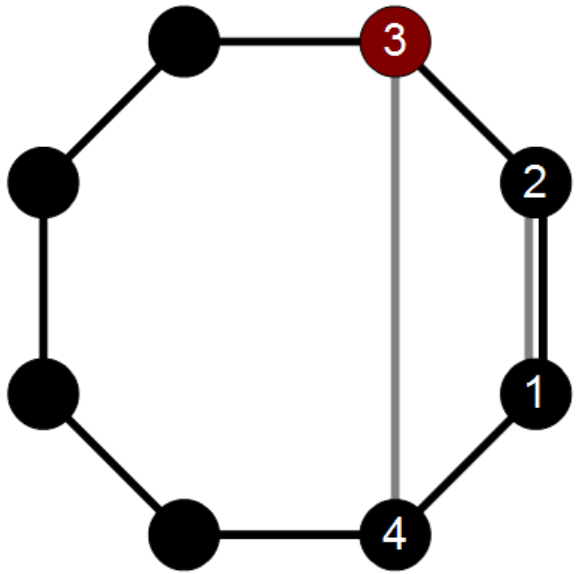
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

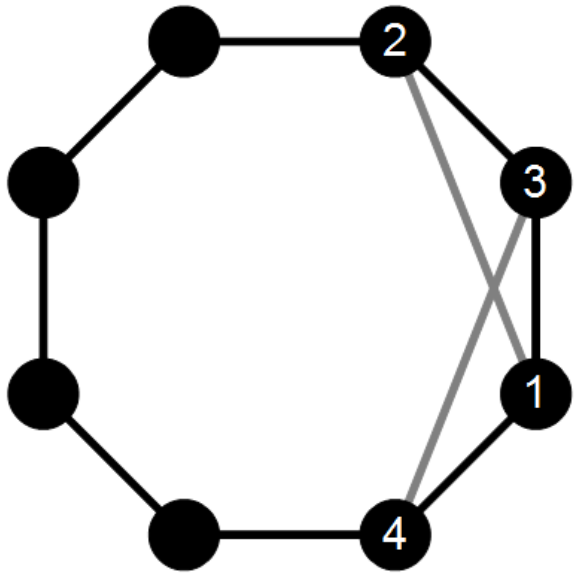
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

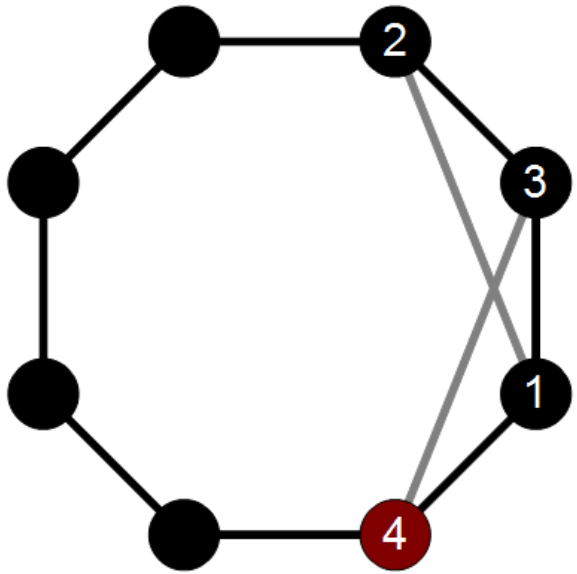
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

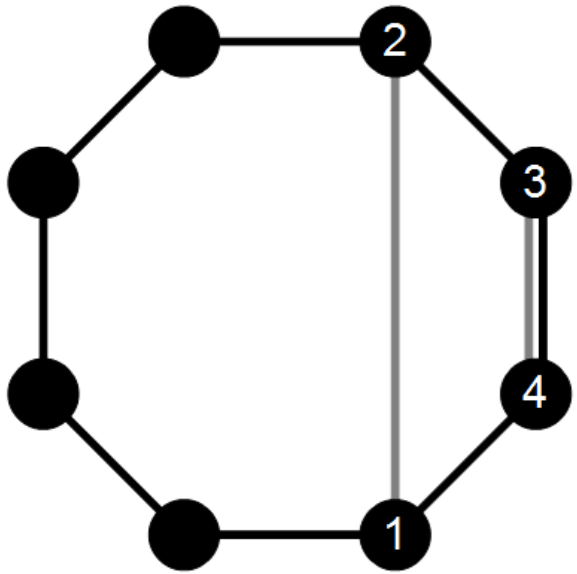
Looking for easier problems



- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Current Progress

Looking for easier problems



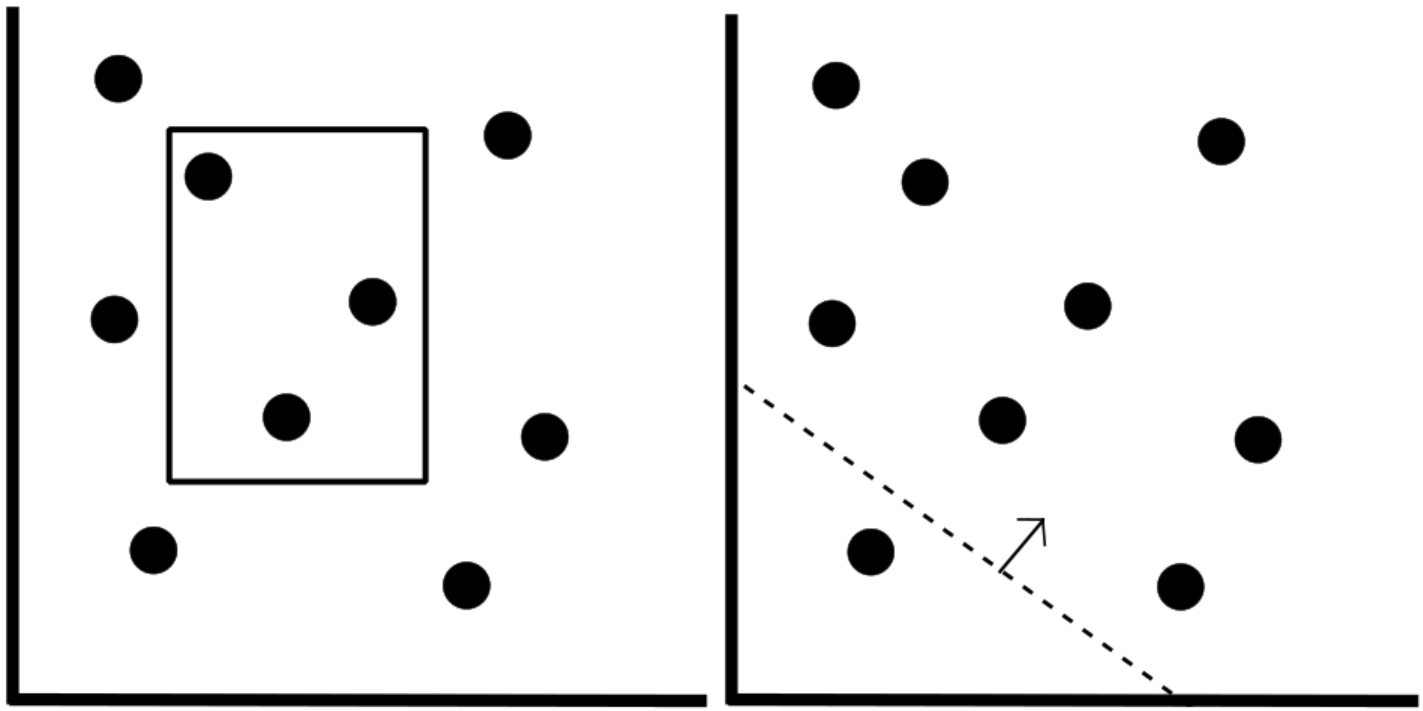
- Nodes on static ring topology
- No initial friendships, friends get externally added whenever an equilibrium is reached
- **Nodes can swap position with a neighbor**
- $v \in V$ minimizes $\max_{u \in \mathcal{F}(v)} d(u, v)$

Future Plans

- Approach the start problem
  - Find more easy models and results
  - Generalize techniques
- Identify further reasonable models
  - Consider bandwidths / costs
  - Local view of nodes
- Compare adapting to non-adapting overlays

Range and Search Queries

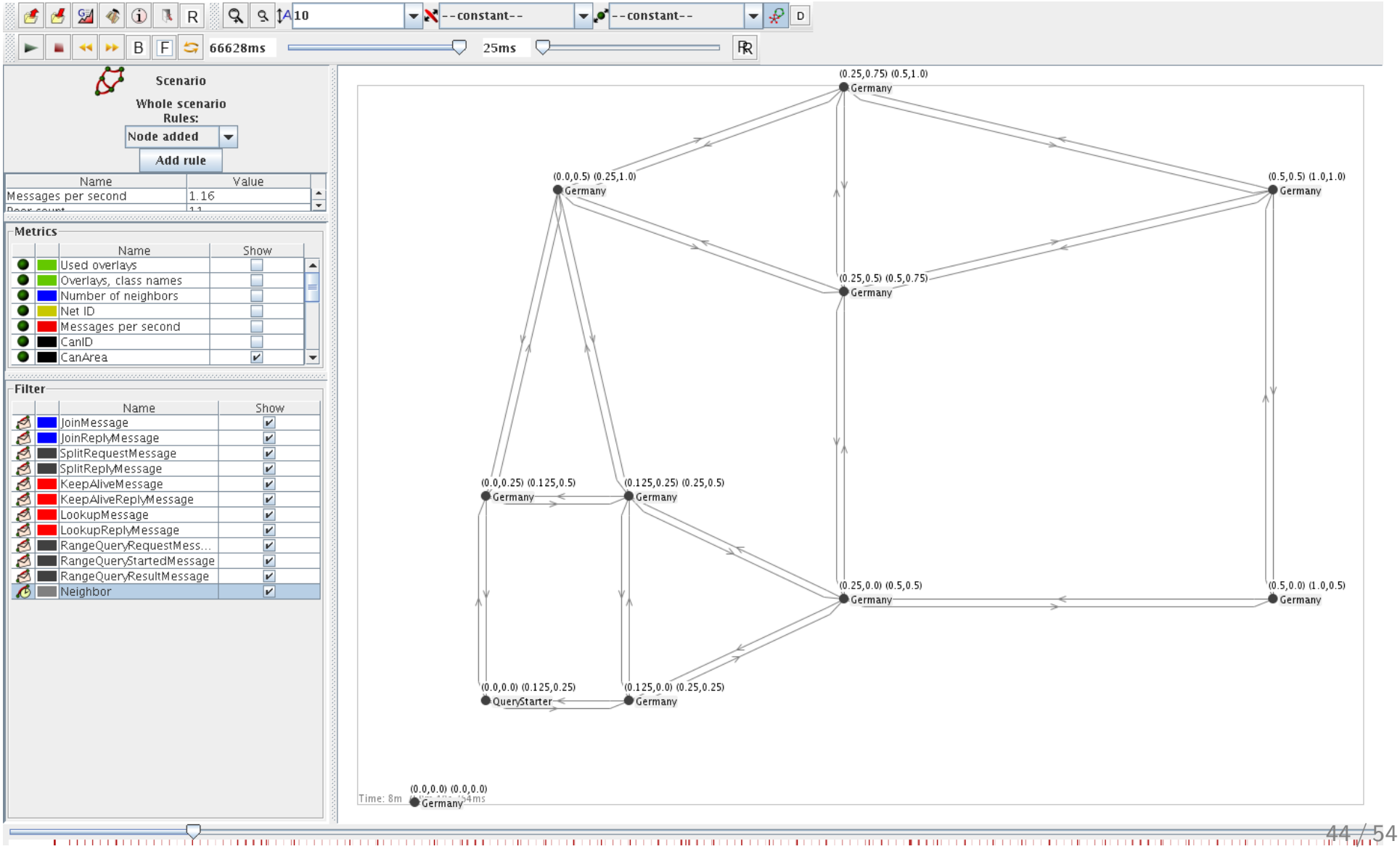
Scenario



- Data items are associated with coordinates in $[0,1]^d$. Tasks:
  - Find all data items in the given range $\mathcal{R} \subseteq [0,1]^d$ or
  - Find a data item $x^* \in [0,1]^d$ maximizing $f : [0,1]^d \to \mathbb{R}$.

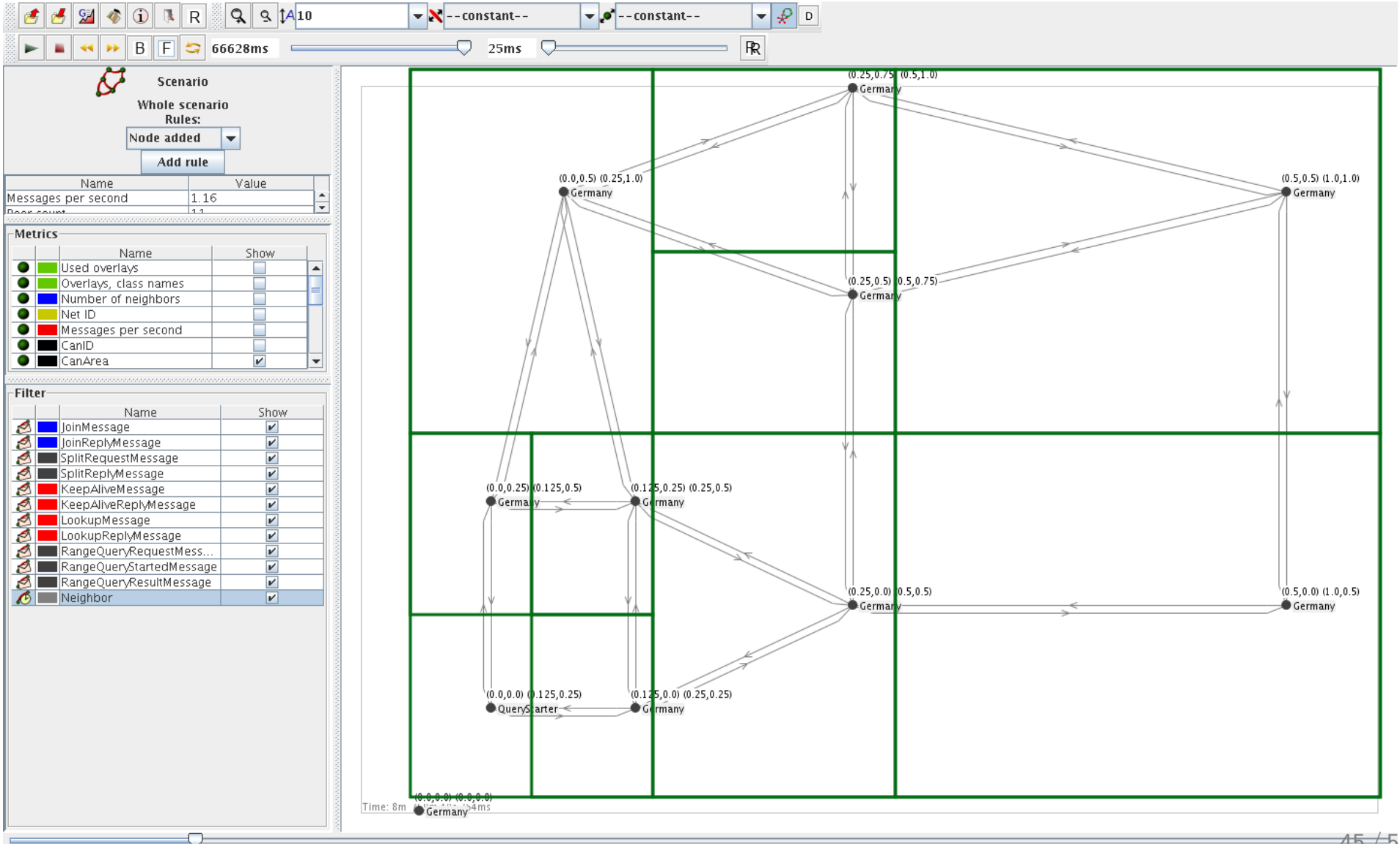
Current Progress

Simulator



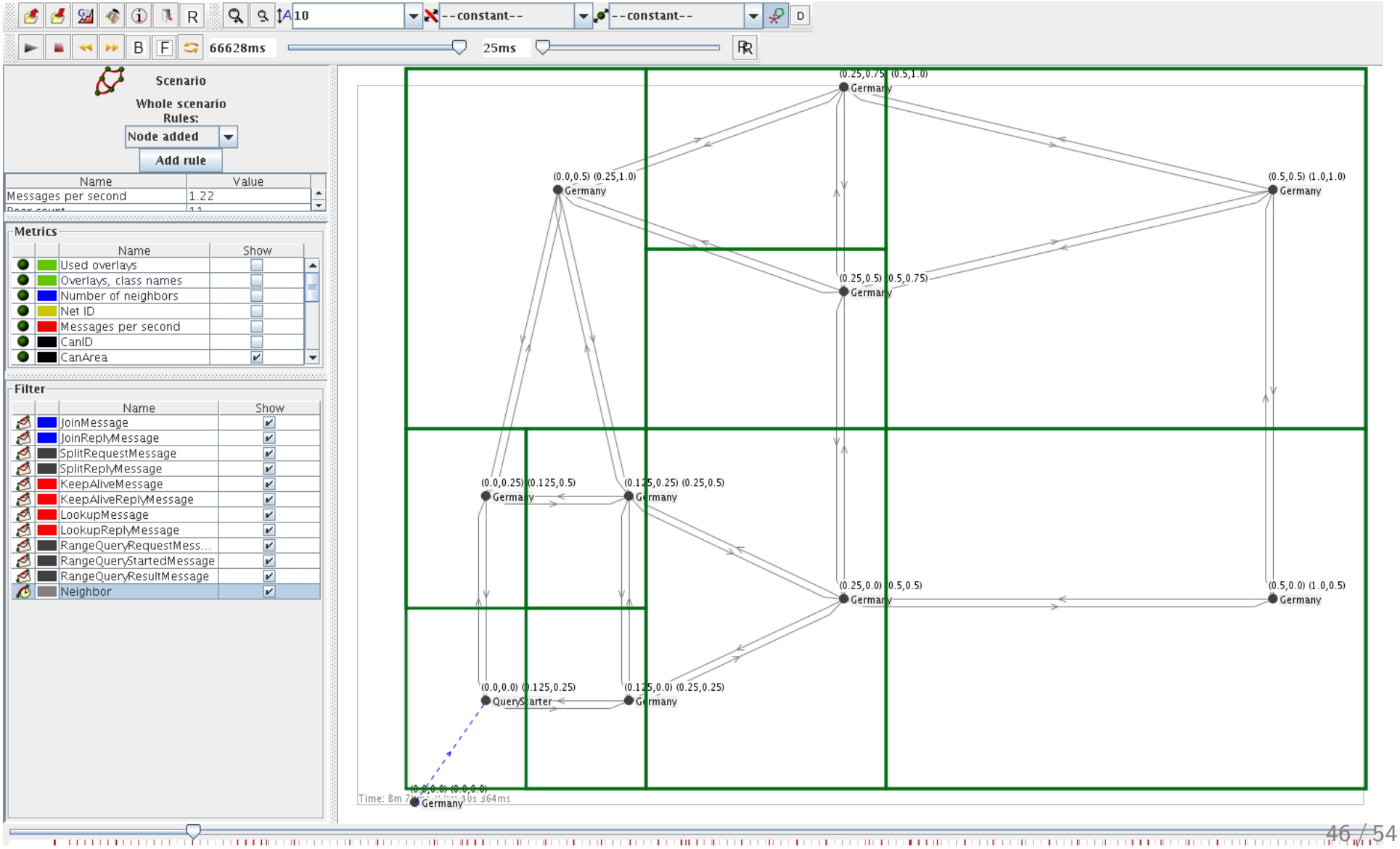
Current Progress

Simulator



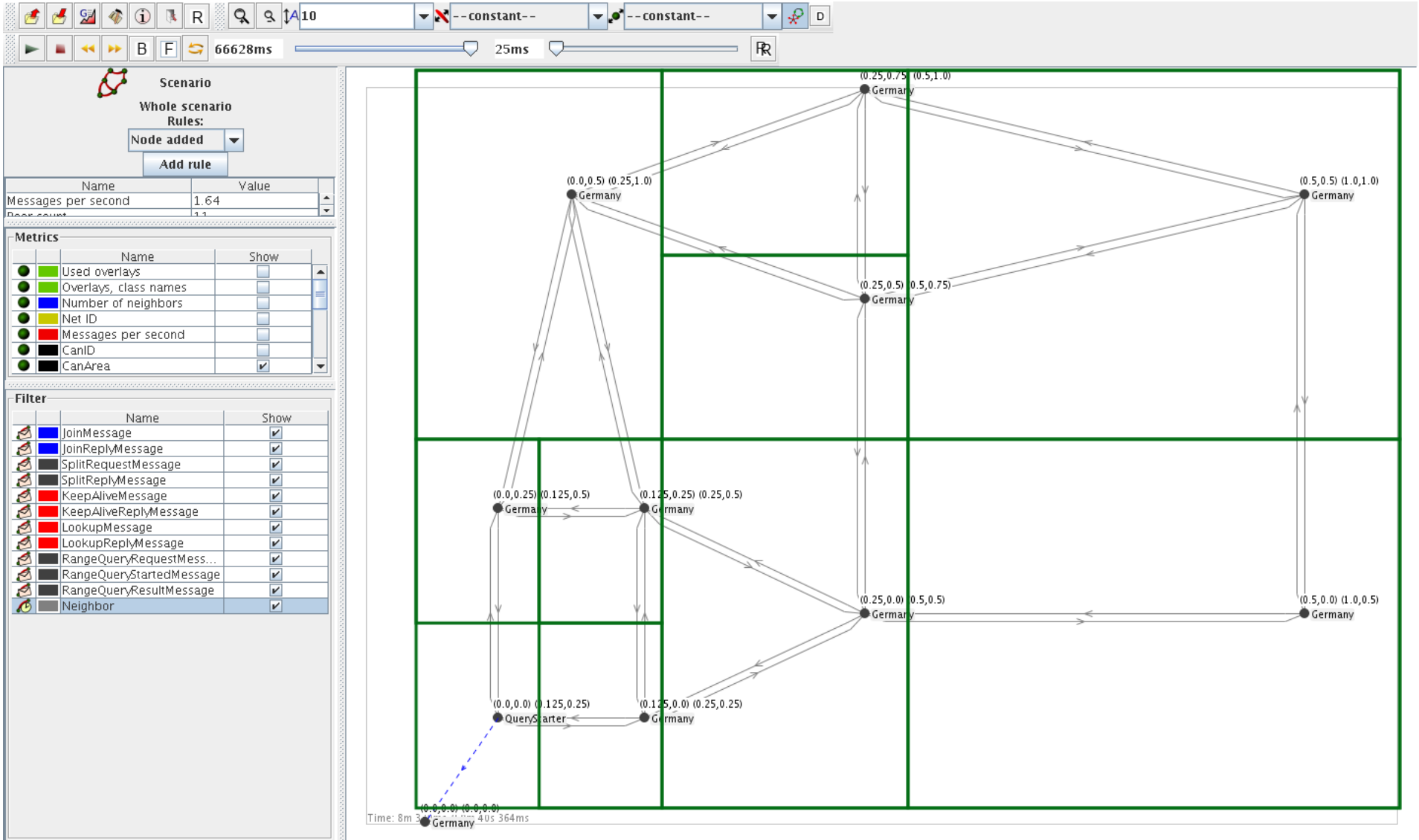
Current Progress

Simulator



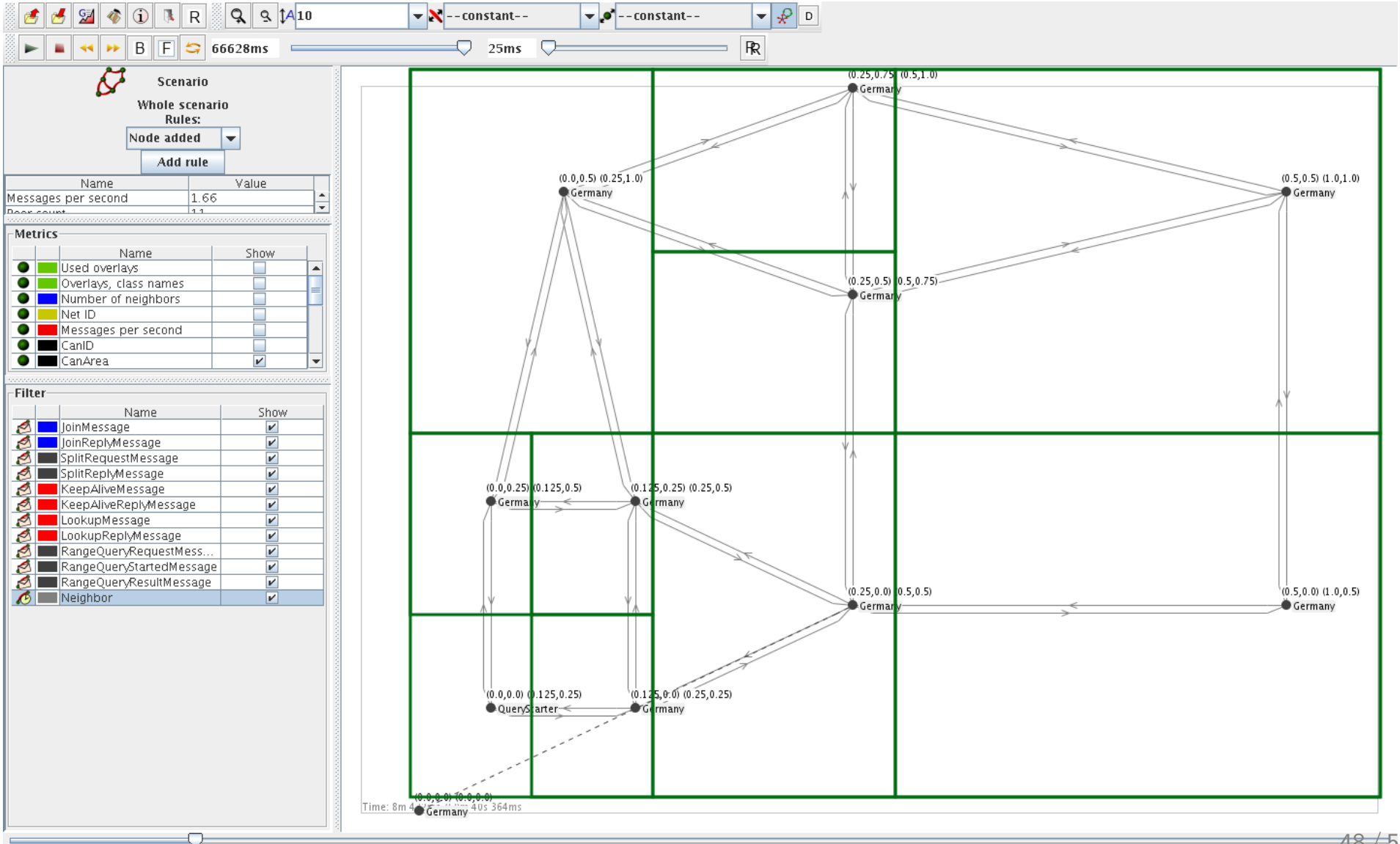
Current Progress

Simulator



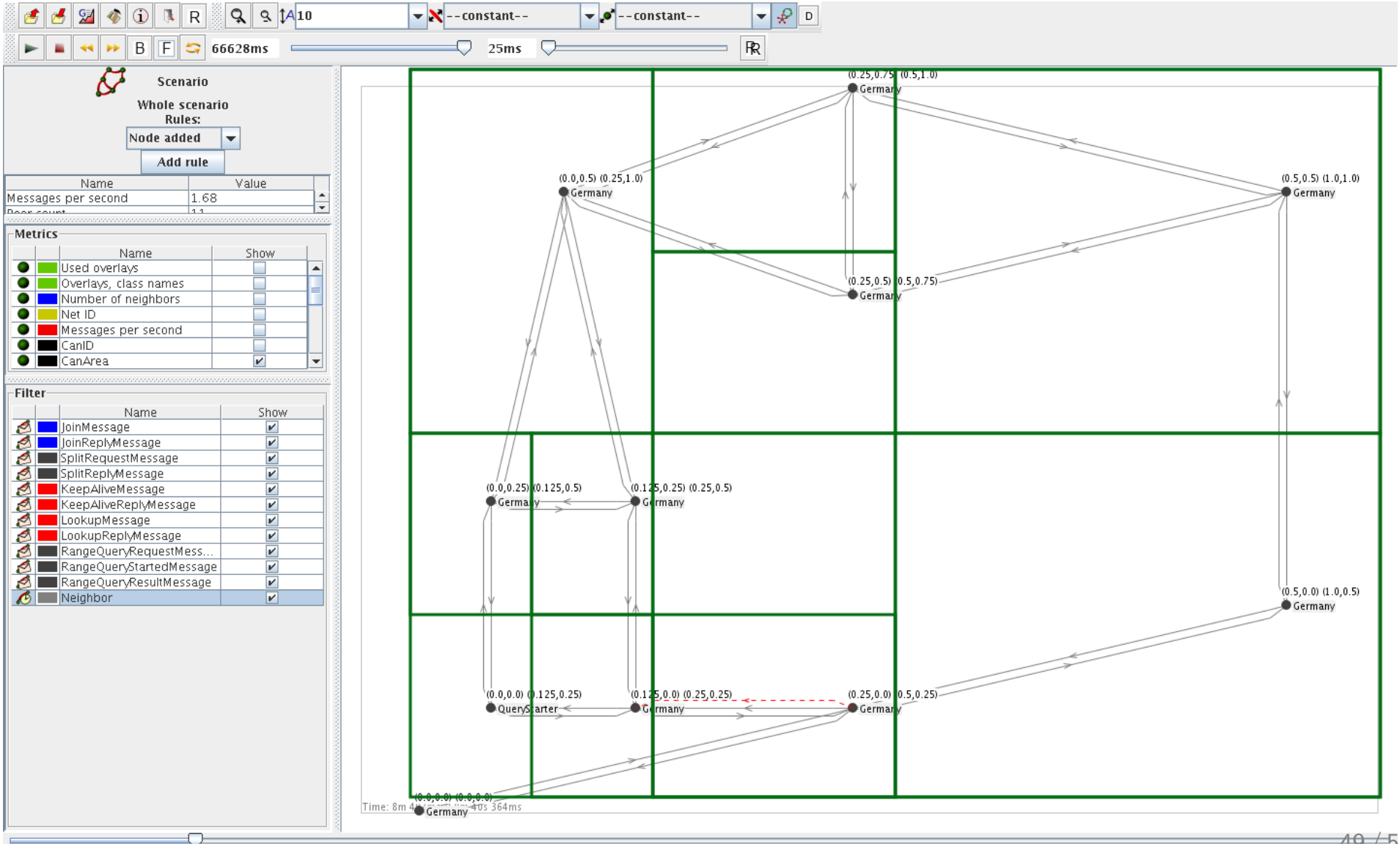
Current Progress

Simulator



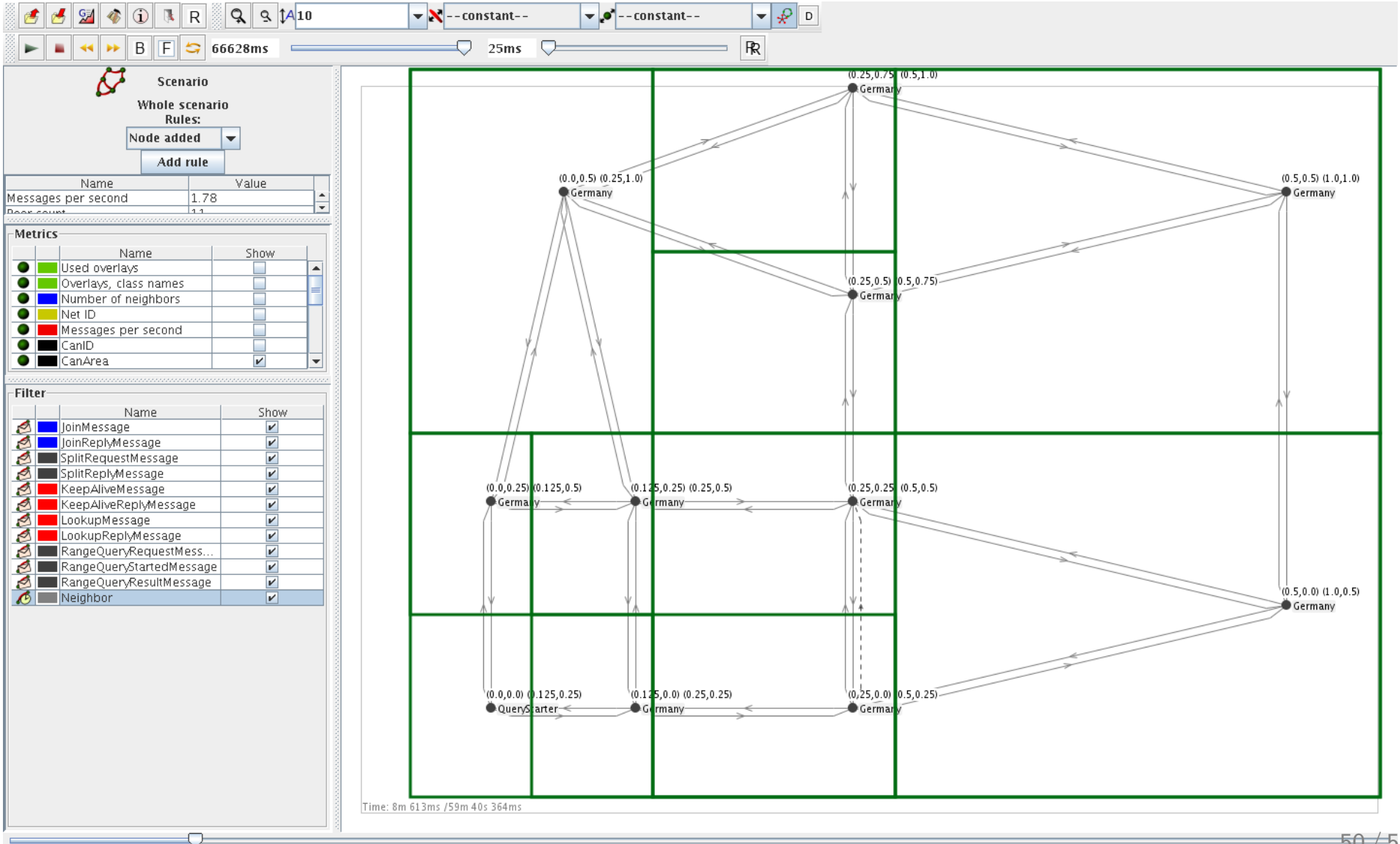
Current Progress

Simulator



Current Progress

Simulator



Current Progress

- Algorithm for objective function scenario
- Assumptions:
  - $n = 2^{2k}$ nodes, $k \in \mathbb{N}$
  - Nodes uniformly distributed
  - Areas of equal size
  - Linear function $f : \mathbb{R}^2 \to \mathbb{R}$, non-negative coefficients
- Result: Message complexity $\mathcal{O}(\sqrt{n})$ with overhead $\mathcal{O}(1)$ per node

Next Steps

- Extensions to non-uniform areas, negative coefficients
- Analyze and reduce message size
- Improve routing time
- Looking for more powerful metastructures

Future Plans

- Objective function search extensions
  - More dimensions
  - Broader classes of functions (e. g. convex functions)
- Range queries
  - Start our research

Thank you for your attention.