Random Walking with a Purpose

Trenton W. Ford
CSE 60742

Graph Kernel: Random Walk Pseudocode

A Random Walk:

Given a graph $G(V, E)$ where V is a set of vertices, and E is a set of edges.

1. Select an arbitrary starting node u in G.
2. Randomly select a neighbor of u, say v.
3. Move across the edge (u, v) , let $u < - v$ and repeat steps 2 and 3... or stop.

Complexity: It depends on the purpose(stopping criteria) of the walk.

- For practically all scenarios: Ω(RW) = Ω(|V|+|E|) for basic traversal

Random Walk Applications

- Modeling diffusion (Brownian, epidemics, mosquitos, etc.)
- Sampling from large networks (embeddings, grammars, etc.)
- Fun games (Conway's Game of Life… others?)

Data Set Considerations

Changes in epidemic transmission behaviors rely on many network properties:

Recovery Time, Epidemic Infectiousness, Average Node Degree, Clustering Propensity, and many more.

Data Sets: Real-world Networks

Advantages:

Results are more meaningful

Potential Datasets

Dataset	Approximate Order	Approximate Size
DOT Railway Data	196K	250K
SNAP Airport Data	456	71K
KNB Shipping Data	3700	15K

Disadvantages:

Scaling via subsetting real-world datasets is hard

Data Sets: Synthetic Networks

Advantages:

Scaling is easy and repeatable.

Disadvantages:

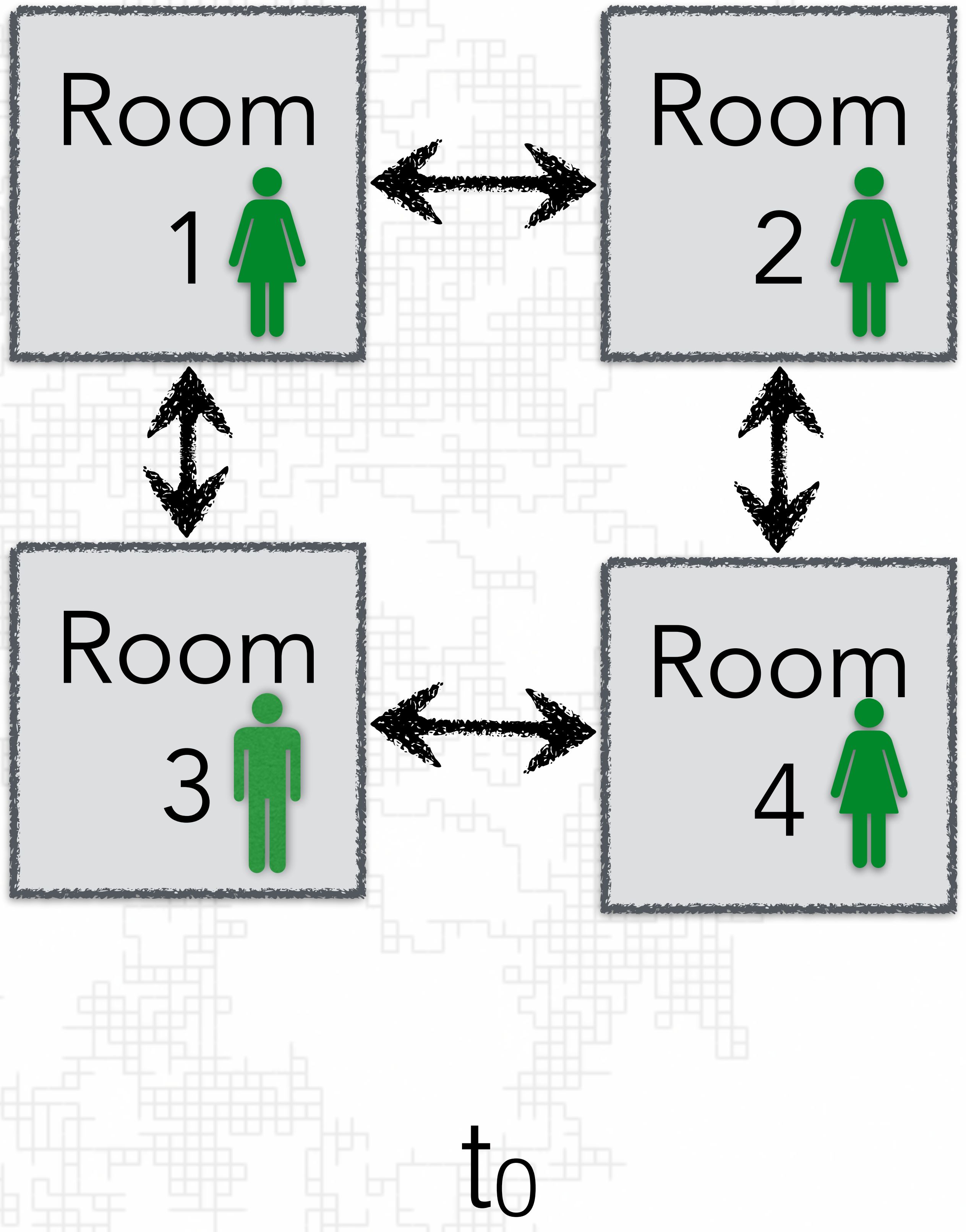
The networks may not accurately represent a real-world scenario

Source:

Temporal Graph Generation Based on a Distribution of Temporal Motifs

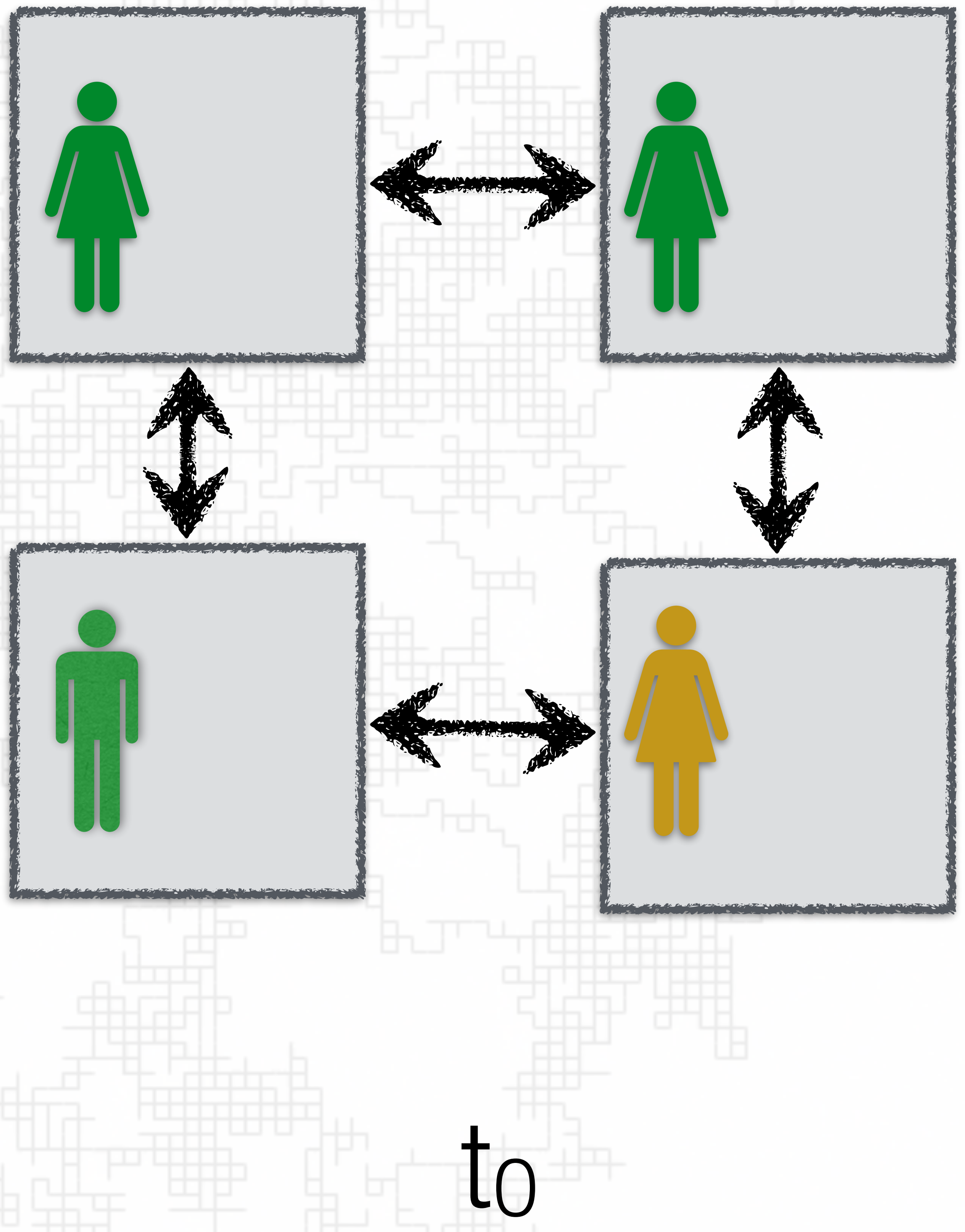
Random Walk Application: Simple Epidemic Transmission using Grids

1. At t_0 each room contains a person, and one person is infectious and if in contact with a susceptible person will infect them with probability p.
2. Between each time step, each person has a probability λ to move from one room to another, with a uniform probability between the rooms they have access to.
3. Each infectious person has a μ probability of recovering at each time step after their initial infection.



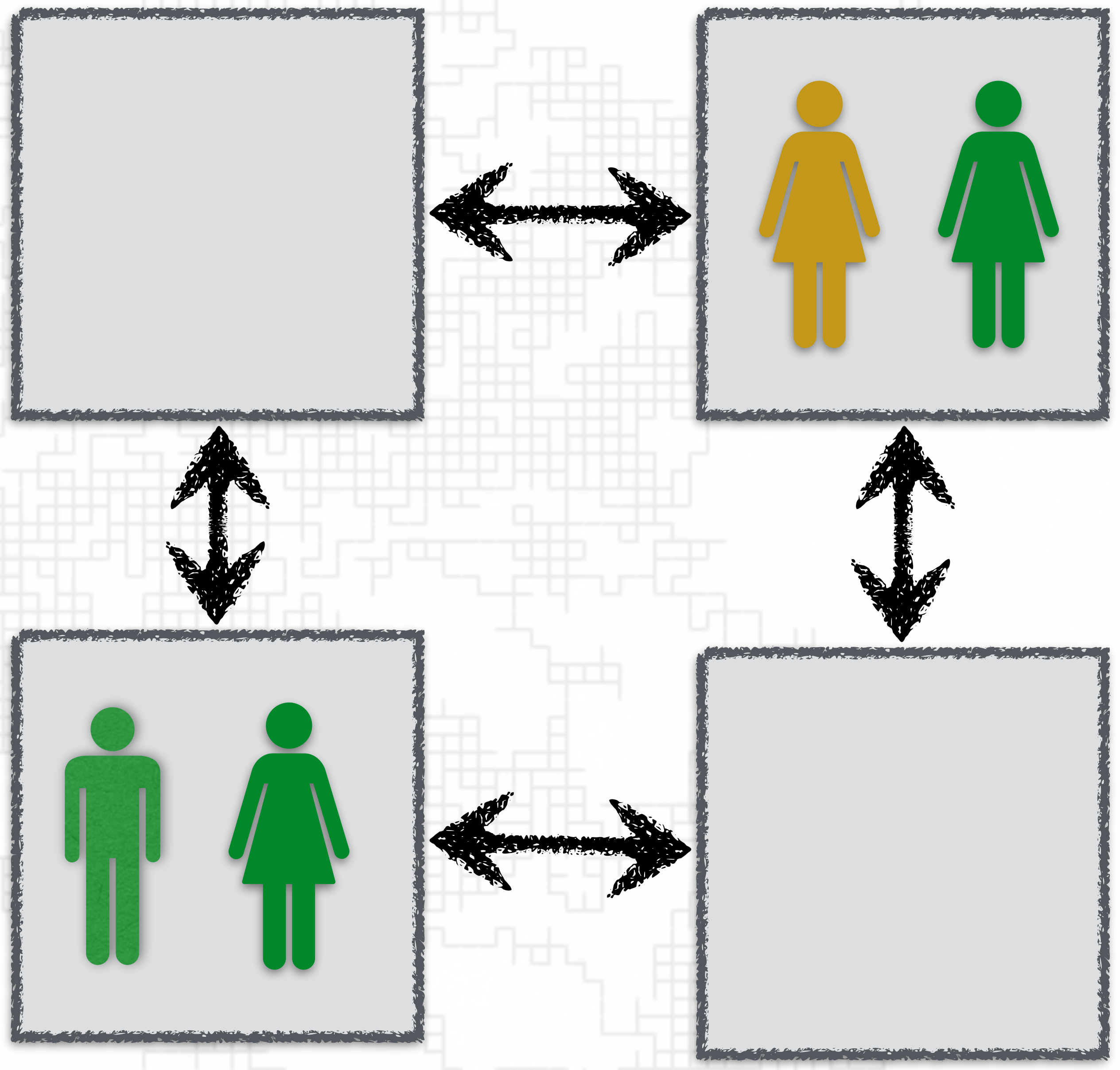
Random Walk Application: Simple Epidemic Transmission using Grids Example

Let $p = 0.5$, $\lambda = 0.5$, and $\mu = 0.25$



Random Walk Application: Simple Epidemic Transmission using Grids Example

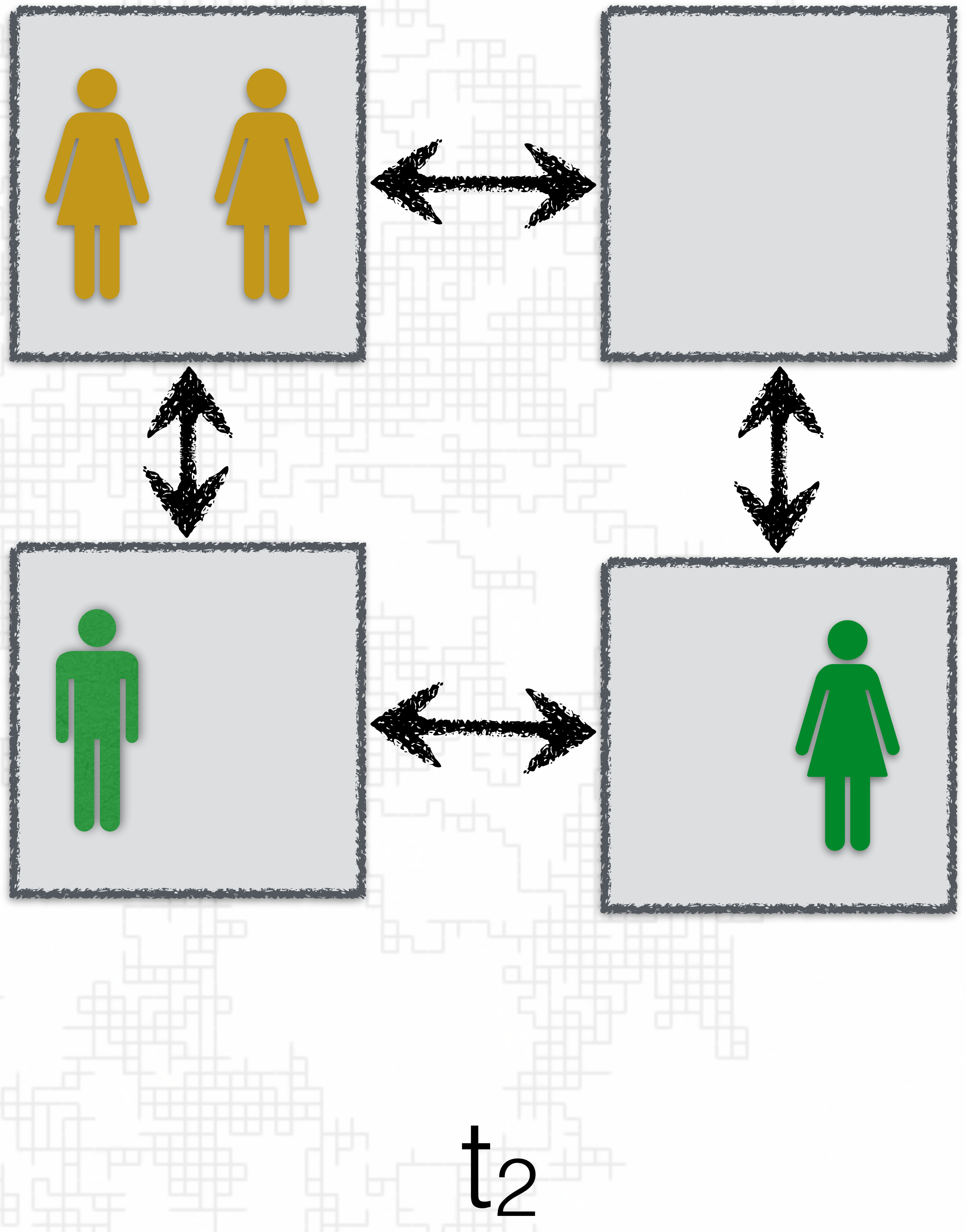
Let $p = 0.5$, $\lambda = 0.5$, and $\mu = 0.25$



t_1

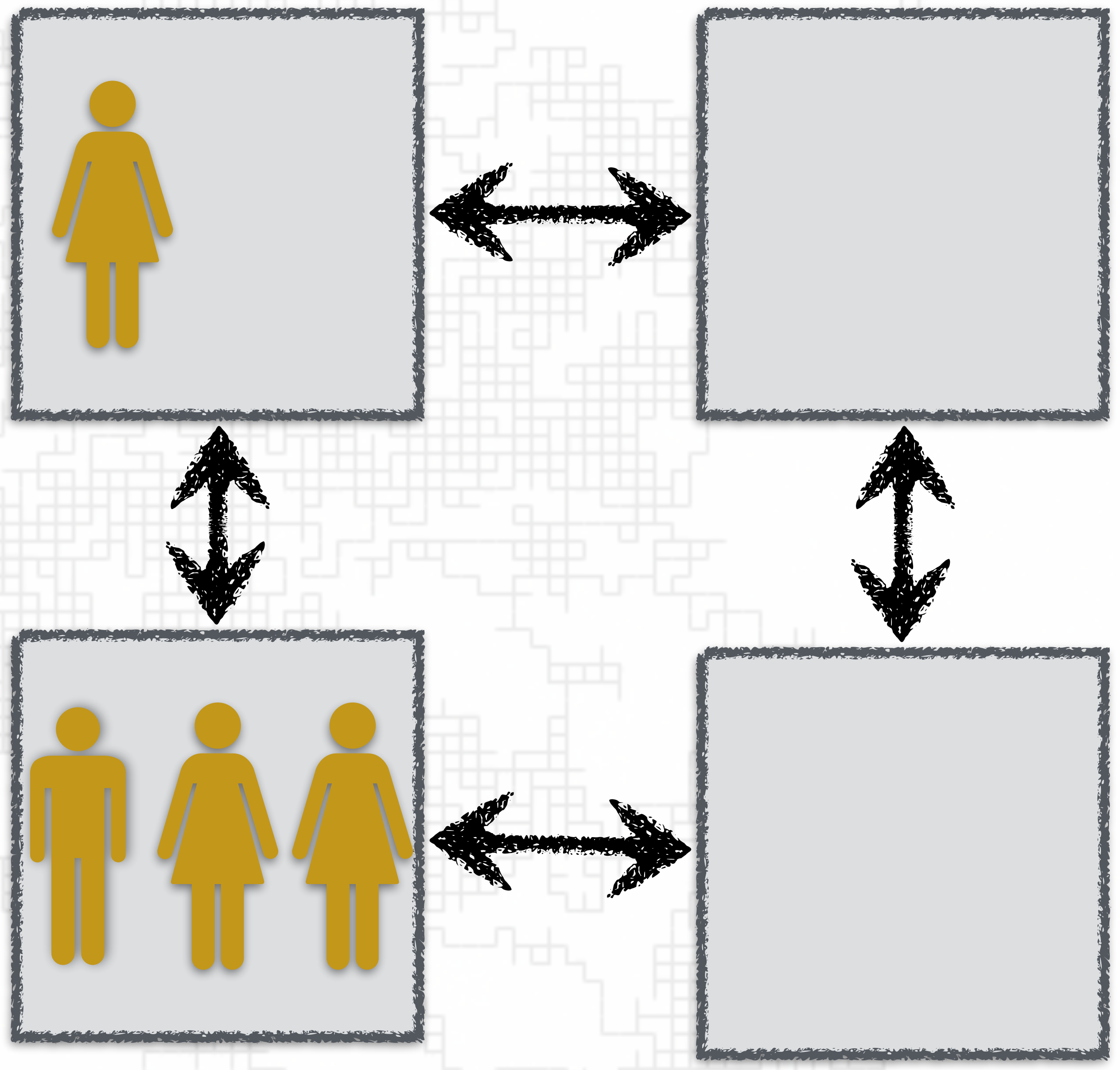
Random Walk Application: Simple Epidemic Transmission using Grids Example

Let p = 0.5, λ = 0.5, and μ = 0.25



Random Walk Application: Simple Epidemic Transmission using Grids Example

Let $p = 0.5$, $\lambda = 0.5$, and $\mu = 0.25$



t_3

Random Walk Application: Simple Epidemic Transmission using Grids Example

1. Transportation networks can be converted to planar networks by duplicating confounding nodes and edges.

2. **Fáry's theorem** says that any planar network can be represented via a grid layout

3. Simple Grid Model is extensible, but becomes a less intuitive abstraction.

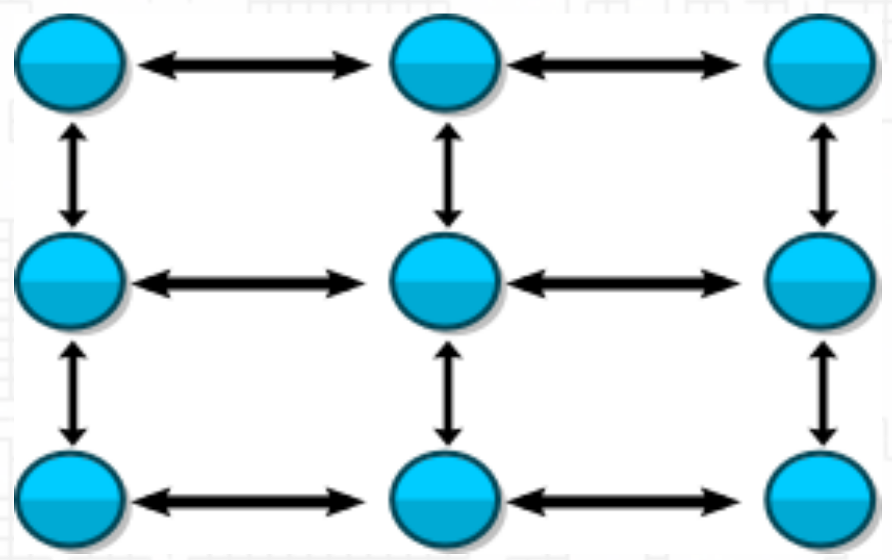
Simplified Grid Random Walk

Language: Boost Graph Library

Basic Structures:

```
struct VertexProperties
{
    struct step_data{};

    step_data current_step;
    step_data future_step;

    uint row = 0;
    uint col = 0;

    static boost::random::mt19937 rng;
    static boost::random::uniform_real_distribution<double> gen;

    static constexpr float p = 0.5; // Infection Probability
    static constexpr float lamb = 0.5; // Movement Probability
    static constexpr float mu = 0.25; // Recovery Probability

    VertexProperties(){}

    VertexProperties(uint r, uint c, uint pop){}

    void infect(){}
    void recover(){}
    void update(){}
    void set_current_values(pop , ipop , spop){}
    void adjust_future_values(pop = 0,ipop = 0,spop = 0){}
    void advance_timestep(){}
};
```

```
#include <boost/graph/grid_graph.hpp>
typedef grid_graph<2> GraphType;
```



```
for (uint i = 0; i < DIMENSIONS; ++i)
    for (uint j = 0; j < DIMENSIONS; ++j)
        Vertex v = Traits::vertex_descriptor {{i, j}}
        vertex_data = get(dataMap, v);
        for (out_verts in boost::adj(vertex_data))
            do a thing using vertex_data
        put(dataMap, v, vertex_data)
```

Simplified Grid Random Walk

Complexity:

$$O\left(|V| * \frac{1}{convergenceRate} \right)$$

convergenceRate is based on the graph structure distribution of *p,* λ, and μ, and has yet to be thoroughly analyzed, but appears that

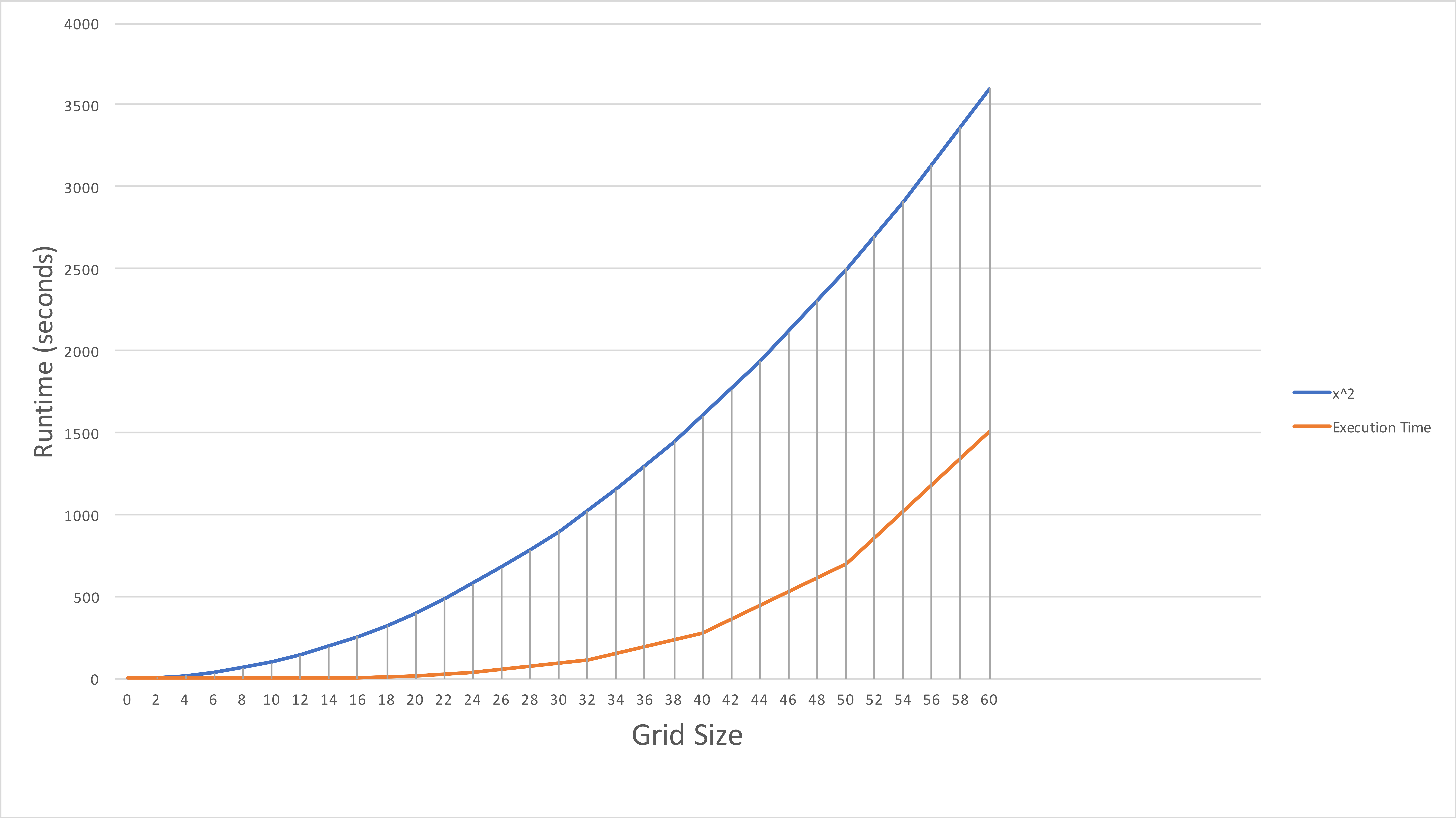
$$O(convergenceRate^{-1}) < O(n^2)$$

Runtime Analysis:

System already distributing simple operations.

Process Name	% CPU	CPU Time	Threads	Idle Wake Ups	PID	User
Terminal	135.6	11:05.06	7	0	937	trentonford

Simplified Grid Random Walk



Future Steps

Random Walking + Dynamic Weights: Parallel Implementation

Refactor for parallelism using Parallel Boost

Change graph type from grid for greater flexibility

Work on graph partitioning strategy for different distributions

Analyze hyper-parameter sensitivity (p,λ,μ)

Questions?

Random Walk Application: Epidemic Transmission

Let I_i be the set of infective vertices at time t_i.
Let $New\_I_i = I_i - I_{i-1}$

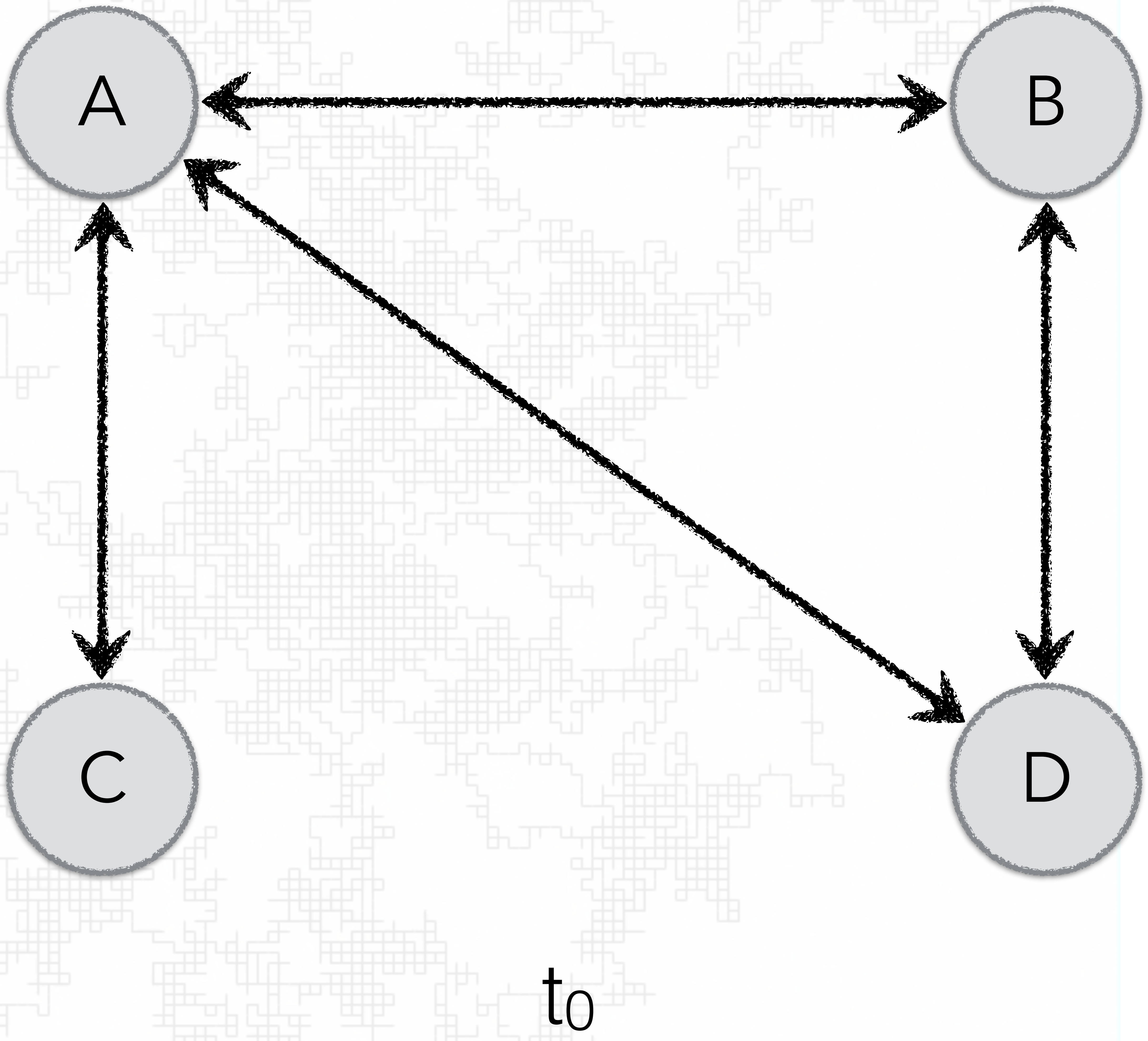
Define: $IR_i(u) = \dfrac{ipop_i(u)}{pop_i(u)}$

$Transfer_i(u, v) =$ contact between u and v beteeen time t_{i-1} and t_i.

Let $T_i(u, v) =$ transition count from u to v at time t_i.

$T_i(u, v) = Transfer_i(u, v) \times IR_{i-1}(u)$

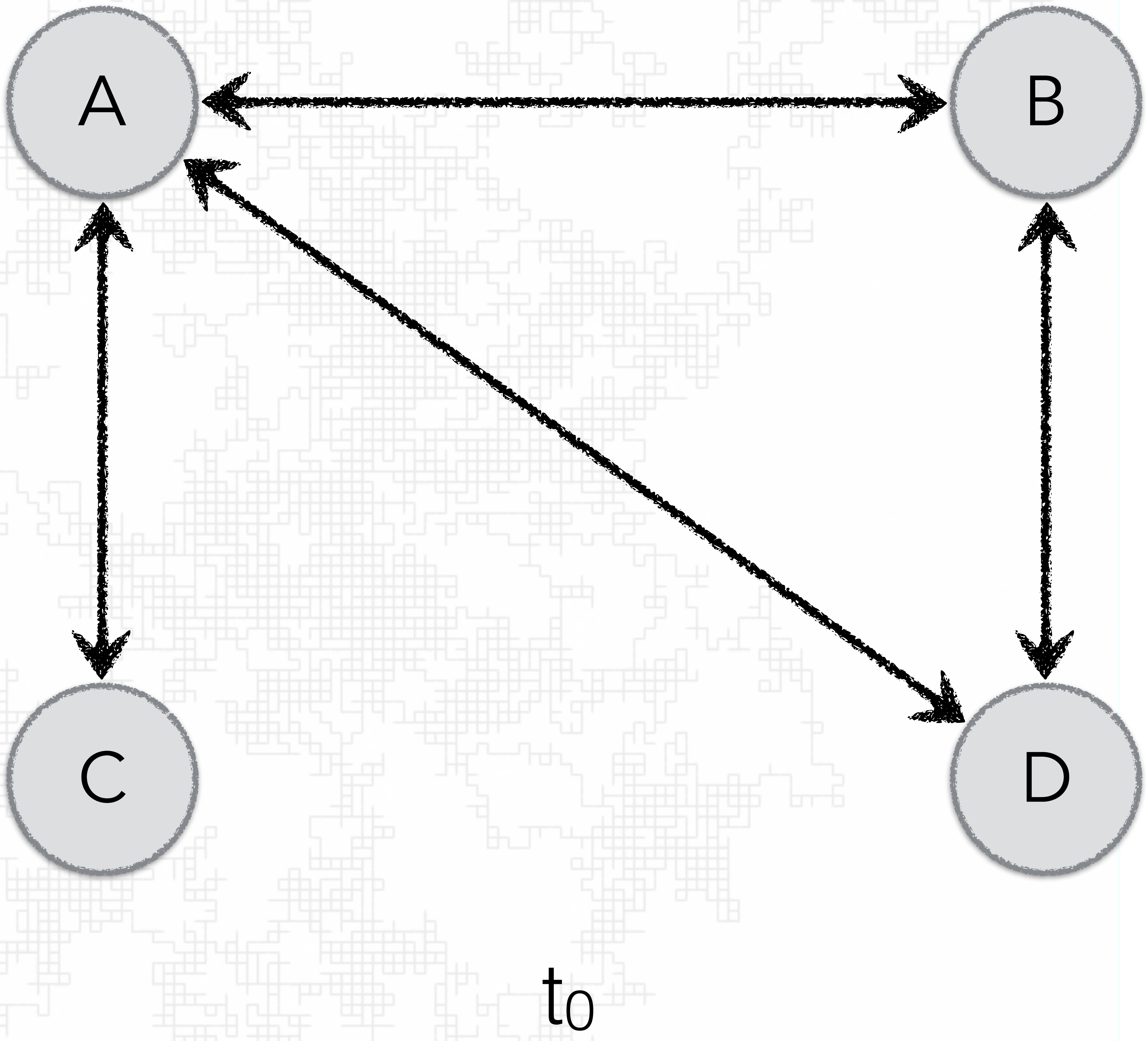
This can be defined in many ways.



Random Walk Application: Epidemic Transmission

Factors not being considered:

1. Random Spread
2. Disease Carriers
3. Non-Random Distribution of Carriers
4. Population Shift
5. Inanimate Disease Vectors

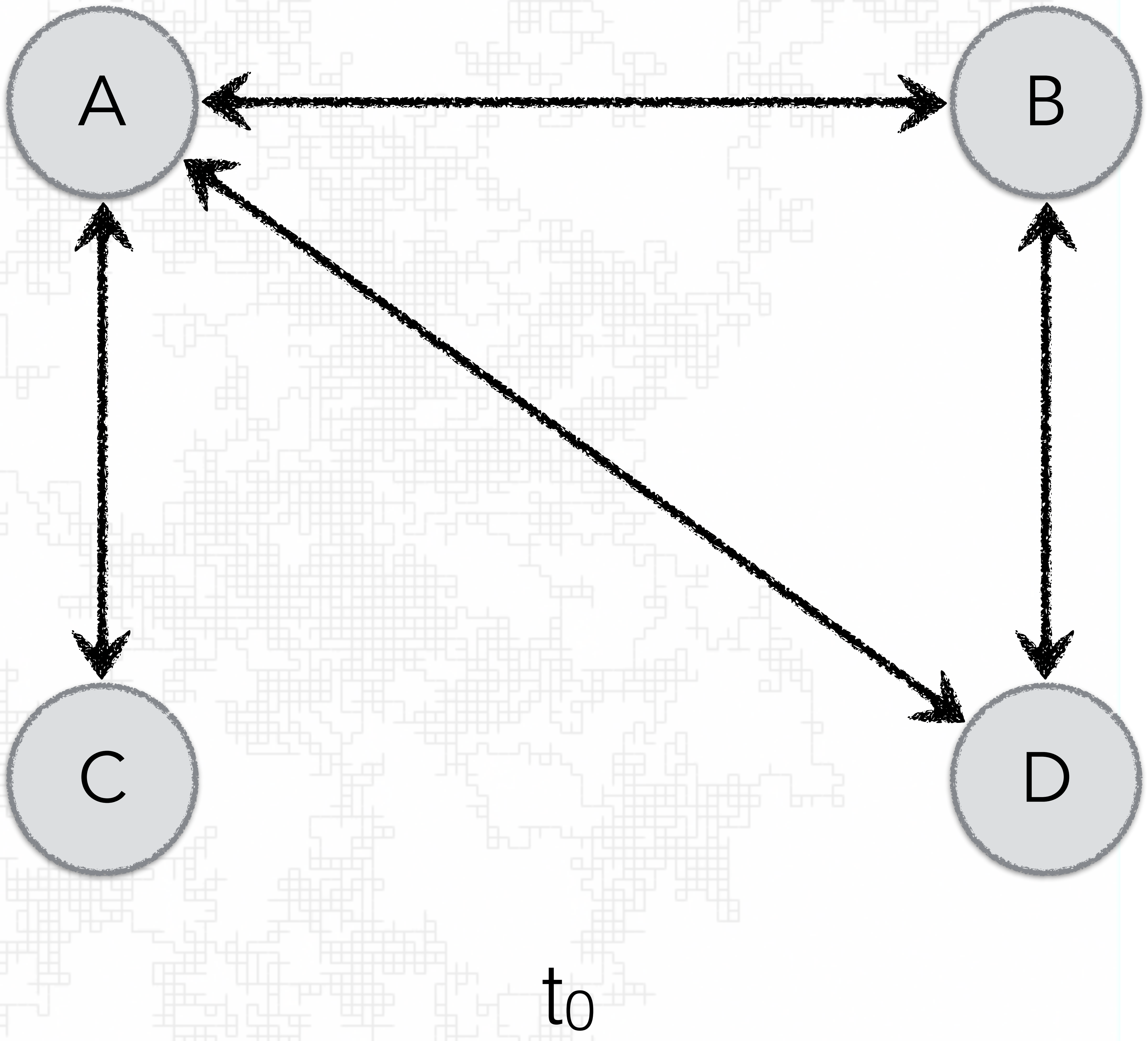


Random Walk Application: Epidemic Transmission

$$I_0 = \{\}$$

$$IR_0(v \in V) = \frac{ipop_i(v)}{pop_i(v)} = \frac{0}{pop(v)}$$

At time t_0 there are no infective vertices. But they are all susceptible.

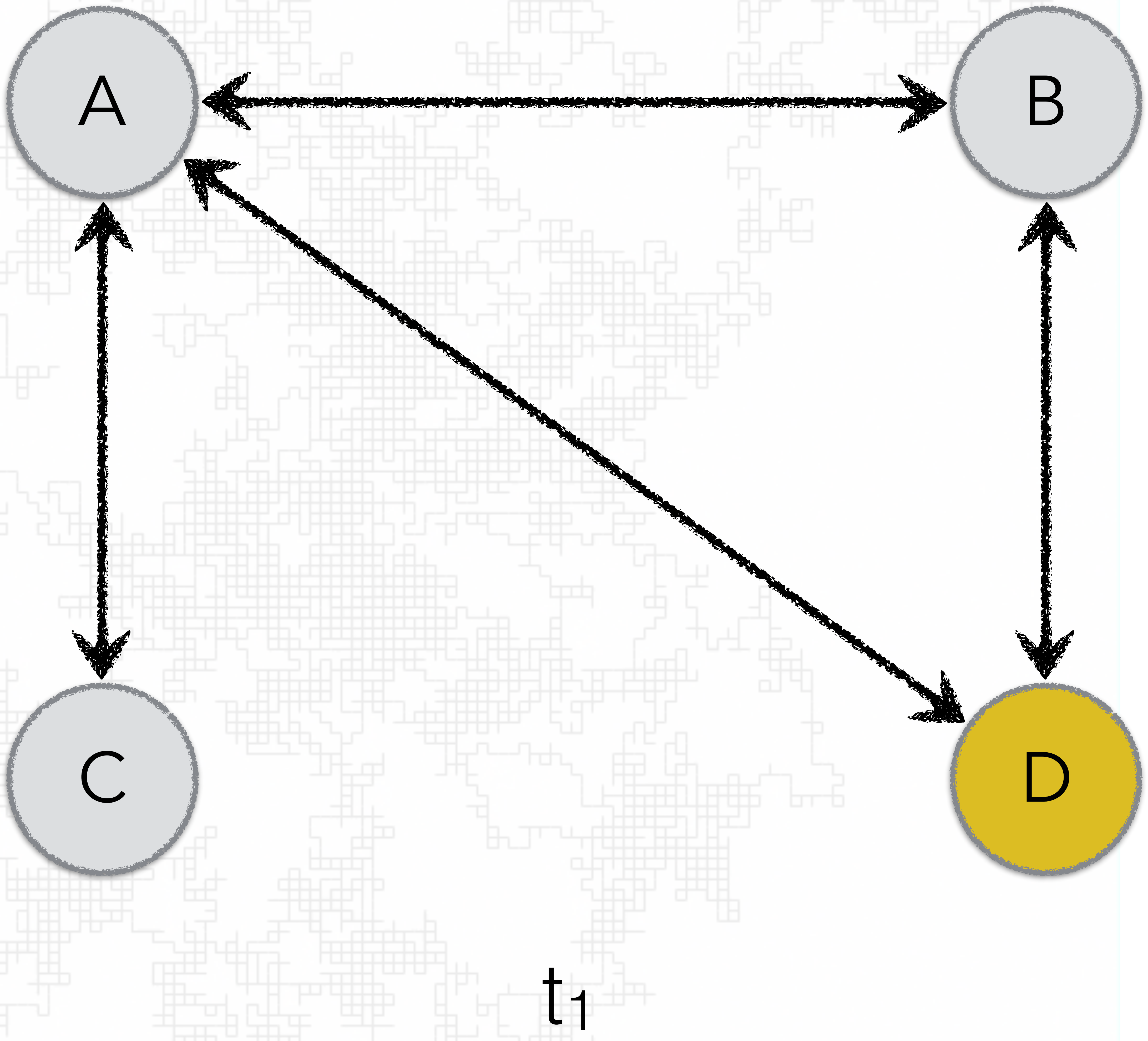


Random Walk Application: Epidemic Transmission

$$I_1 = \{D\}$$
$$New\_I_1 = I_1 - I_0 = \{D\}$$
$$IR_1(D) = 1/100 = .01$$

$$T_2(D, A) = Transfer_2(D, A) \times IR_1(D) = 1$$
$$T_2(D, B) = Transfer_2(D, B) \times IR_1(D) = .2$$

$$T_i(u, v) = Transfer_i(u, v) \times IR_{i-1}(u)$$



Random Walk Application: Epidemic Transmission

$$I_2 = \{D, A\}$$

$$New\_I_2 = I_2 - I_1 = \{A\}$$

What is the probability that vertex *A* was infected via vertex *D*

$$Transfer_2(B, A) = 20\ people$$

$$Transfer_2(C, A) = 50\ people$$

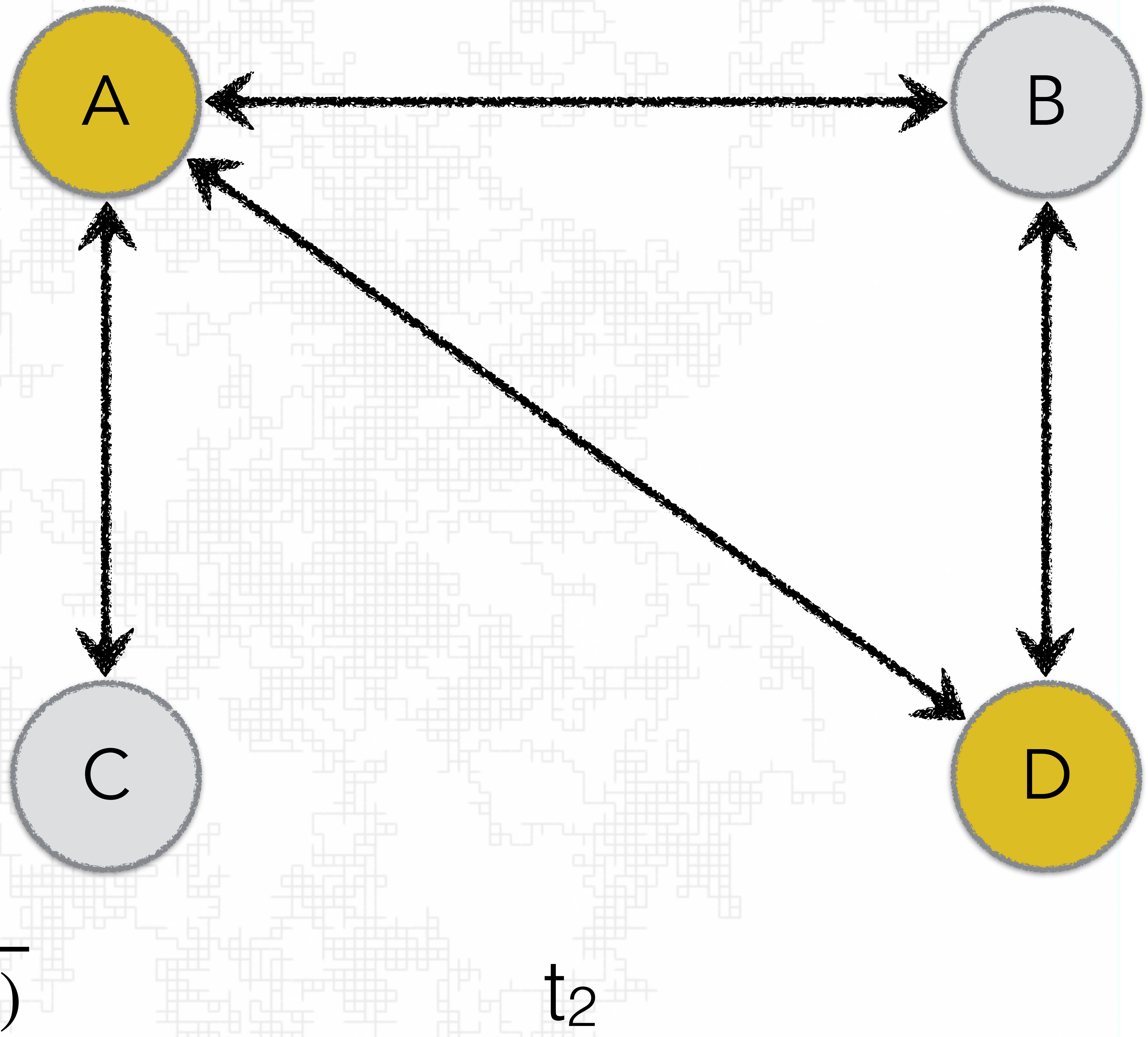
$$Transfer_2(D, A) = 100\ people$$

$$T_2(B, A) = Transfer_2(B, A) \times IR_1(B) = 0$$

$$T_2(C, A) = Transfer_2(C, A) \times IR_1(C) = 0$$

$$T_2(D, A) = Transfer_2(D, A) \times IR_1(D) = 1$$

$$likelihood(A < -D \,|\, A \in New\_I_2) \propto \frac{T_2(D, A)}{\sum_{v \in Adj(A)} T_2(v, A)}$$



Random Walk Application: Epidemic Transmission

$I_3 = \{D, A, B\}$

$New\_I_3 = I_3 - I_2 = \{B\}$

Through which vertex is B more likely to be infected?

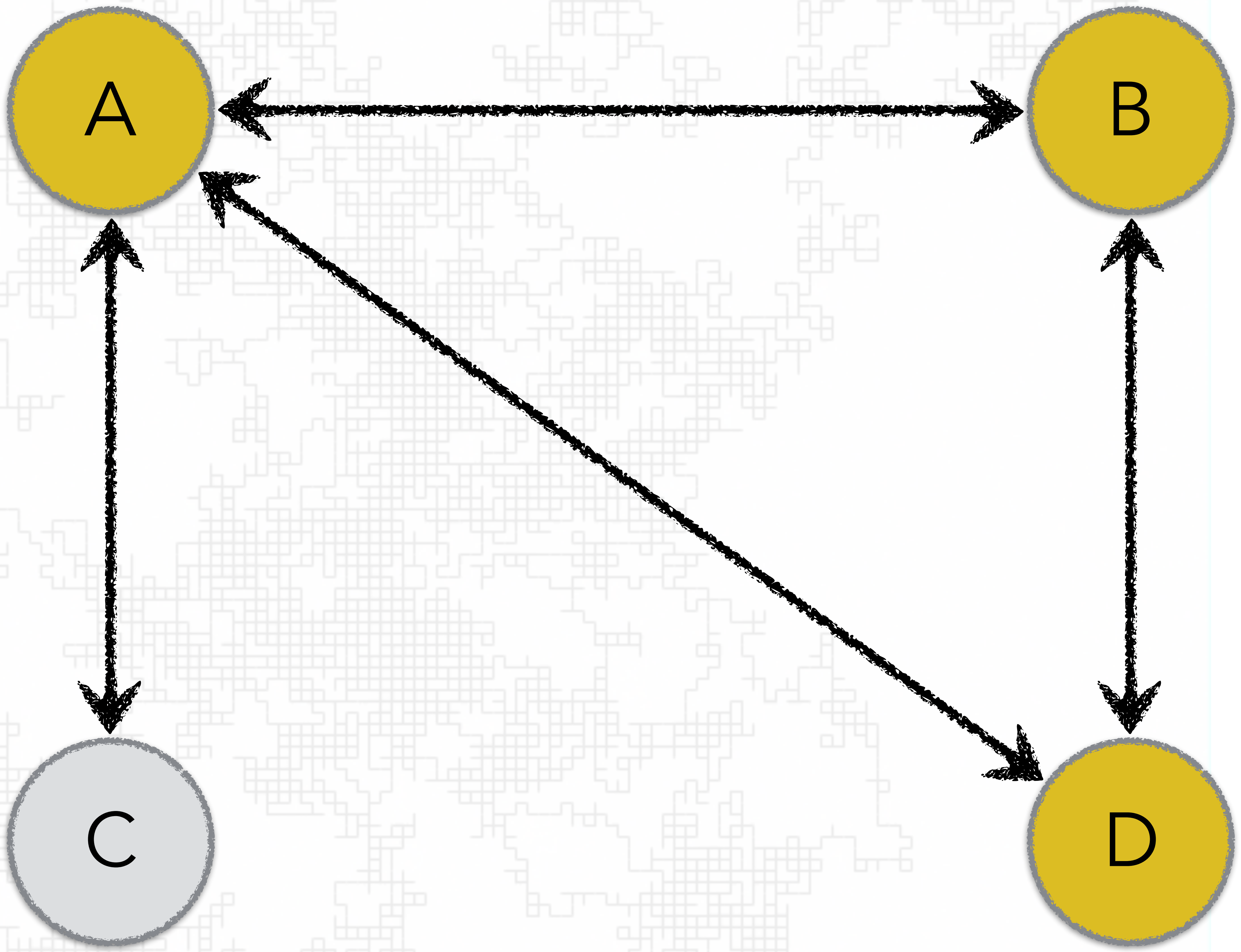
$Transfer_3(A, B) = 100\ people$

$Transfer_3(D, B) = 100\ people$

$T_3(A, B) = Transfer_3(A, B) \times IR_2(A) = 0$

$T_3(D, B) = Transfer_3(D, B) \times IR_2(D) = 0$

$$likelihood(A < - D \mid A \in New\_I_2) \propto \frac{T_2(A, D)}{\sum_{v \in Adj(A)} T_2(v, A)}$$



t_3