Probabilistic Relational Agent-Based Models

1. Motivation

- What
  - Accelerate modeling of all kinds
- Target language for semi-automated construction of probabilistic relational agent-based models (PRAMs)
- How
  - By combining agent-based models with probability theory
  - Effectively arriving at a mass redistribution system

2. Elements of a PRAM

- Entities
  - Groups
    - Mass (e.g., 500 agents)
    - Attributes (e.g., age, sex, flu)
    - Relations (e.g., school, hospital)
- Sites
- Locations agents can be at

3. Modeling Levels

- Domain
  - Invoke domain-specific models (e.g., SIRS)
- Class
  - Invoke a class of processes or models (e.g., MC)
- Rule
  - Write rules directly

  Example: The SIRS model

  - $\beta$ - transmission rate
  - $\gamma$ - recovery rate
  - $\alpha$ - immunity loss rate ($\alpha = 0$ implies life-long immunity)

3.1. Domain Level (example in Python)

```python
SIRModel('flu', $\beta=0.85$, $\gamma=0.50$, $\alpha=0.10$)
```

3.2. Class Level (example in Python)

```python
$\beta$, $\gamma$, $\alpha = 0.85$, 0.50, 0.10
transition_matrix = {
    'S': [1 - $\beta$, $\beta$, 0.00],
    'I': [0.80, 1 - $\gamma$, $\gamma$],
    'R': [0.10, 0.00, 1 - $\alpha$],
}
```

3.3. Rule Level (more elaborate example in pseudo-code)

```python
rule_flu_progression():
    if group.feature.flu == 'S':
        p_inf = n@ - count at the group's location
dx = p_inf -> F:flu = 'I', F:emo = 'annoyed'
c = n_inf
if group.feature.flu == 'I':
    dx 0.1 -> F:flu = 'S', F:emo = 'happy'
dx 0.2 -> F:flu = 'R', F:emo = 'bored'
dx 0.3 -> F:flu = 'I', F:emo = 'annoyed'
if group.feature.flu == 'R':
    dx 0.1 -> F:flu = 'S'
```

3.4. Rule Level (ODEs support in development; Python)

```python
x,y = state
return [x * (a - by), y * (d - rx)]
r = ODESystem(f_lotka_volterra, ['x', 'y'], dt=0.01)
Simulation().add(r, Group(s=2, attr=[['x':10, 'y':10]]), run(1000)) # Initial (10,10)
```

4. Semi-Automated Construction

- Static and dynamic rule analysis
- Identify essential group attributes and relations
- Automatic population generation from rel. databases

5. Examples of Supported Rules/Models

- Fundamental stochastic processes
- Poisson point process
- Probabilistic
- Finite state-space time-(in)variant Markov chain
- Epidemiological
  - SIS, SIR, SIRS
- Segregation model

6. On-Going Efforts

- Theoretical work on the equivalence between PRAMs and other model types
- Investigating the relationship between PRAMs and dynamical systems specified via ordinary differential equations
- Accounting for continuous group features and continuous-time simulations
- Extending the definition of population
- Allowing changes to the total population mass
- Ensuring proper amalgamation of different models within the same simulation
- Investigating the pedagogical value of PRAMs