

Exercises in Graph Rewriting

MGS 2024, Leicester

1. Petri Nets

- Grammar to generate all marked graphs:
 - All transitions have one pre and one post place
 - All places are marked by 0
- Firing rule for marked graphs:
 - Transition t is enabled if $\text{mark} > 0$ for place $\text{pre}(t)$
 - Firing t , remove a mark from $\text{pre}(t)$ and add one to $\text{post}(t)$
- Net optimisation
 - Replace two-step sequence by transitive step if intermediate place is not connected
- Dynamic allocation
 - Replicate transition if $\text{pre}(t)$ has $\text{mark} > 3$
 - Delete parallel transition if $\text{pre}(t)$ has $\text{mark} < 2$

2. Voting System

Requirements

- Voters and Motions can be created
- Voters are invited to vote for Motions, counting the number Invites issued
- Once voting is open, Voters can vote to in favour of Motion
- Votes in favour of Motion are counted when they are made
- Abstention counts as rejection
- A motion is carried if more than half of eligible voters in favour

Additional (time permitting)

- Voters can delegate their vote to others in their network

2. Polyonymoes

Graph model

- Integer square grid graph
 - nodes (x, y) for integers x, y
 - edges linking nodes at distance 1
- Abstract square grid graph
 - no integer coordinates
- Polyonymoes as connected complete subgraphs
 - Concrete
 - Free

Rules

- Create I and L pentomino
- Check applicability

2. Polyonymoes (cont.)

Puzzles: use model checking to solve

- Space filling problem
- Duplication problem

Model Bokus Game

- Rules
- Strategies

Polymonds

- As above for triangular grids: Blockus trigon