REDUCED STATE SPACE CONSTRUCTION
- COVERABILITY GRAPH

QUALITATIVE ANALYSIS
METHODS, OVERVIEW

- NET REDUCTION
- STRUCTURAL PROPERTIES
- LINEAR PROGRAMMING
  - place / transition invariants
  - state equation
  - trap equation
- REACHABILITY ANALYSIS
  - (complete) reachability graph
  - compressed state spaces
    - BDDs, NDDs, ..., XDDs
    - Kronecker products
  - reduced state spaces
    - coverability graph
    - symmetry
    - stubborn sets
  - branching process

static analysis
(dynamic analysis
(model checking)
**REACHABILITY GRAPH, CONSTRUCTION ALGORITHM**

**PROCEDURE** `rg` (**IN** Net `pn`, **IN** Marking `m0`, **OUT** MSet `nodes`, **OUT** ArcSet `arcs`);

MSet `U` = `{m0}`, // unprocessed markings
`N` = `∅`; // rg nodes
ArcSet `E` = `∅`; // rg arcs (pre, post, t)
Marking `m'`; // successor marking
Transition `t`;

WHILE `U` ≠ `∅` DO
  choose one `m` ∈ `U`;
  `U` = `U` - `{m}`; `N` = `N` ∪ `{m}`;

  FOR ALL `t` enabled at `m` DO
    `m'` = `m` + `Δt`;
    IF `m'` ∉ `N` ∪ `U` // new marking
      THEN `U` = `U` ∪ `{m'}`
      ENDIF;
    `E` = `E` ∪ `{(m, m', t)}`
  ENDFOR

ENDWHILE;

`nodes` = `N`; `arcs` = `E`;

ENDPROC `rg`.

---

**COVERABILITY GRAPH, CONSTRUCTION ALGORITHM**

**-> TWO CHANGES IN PROCEDURE RG**

- **PROCEDURE** `cg` (**IN** Net `pn`, **IN** Marking `m0`, **OUT** OmegaMSet `nodes`, **OUT** ArcSet `arcs`);

  OmegaMSet // omega for infinite
  `U` = `{m0}`, // unprocessed markings
  `N` = `∅`; // rg nodes

- **FOR ALL** `t` enabled at `m` DO
  `m'` = `m` + `Δt`;

  IF `m'` covers some `mOld` ∈ `N` ∪ `U` with path `(mOld, m')` in `E`
    THEN
      FOR ALL `p` ∈ `P` DO
        IF `mOld(p) < m'(p)`
          THEN `m'(p)` = `ω`
          ENDIF
      ENDFOR
    ENDIF

ENDFOR
COVERABILITY GRAPH

- finite also for unbounded nets

- omega-marking
  - generalization of marking
  - omega stands for infinite token numbers

- for bounded nets pn:
  \( rg(pn) = cg(pn) \)

- decidable properties
  - place unboundedness
  - simultaneous unboundedness of places
  - \( m_0 \)-dead transitions

- semi-decidable property
  - non-reachability of states

- non-decidable properties
  - deadlock freedom,
  - liveness
  - reversibility

the result of this Karp-Miller algorithm
- depends on the order markings are considered
- is, generally, not minimal

Finkel algorithm constructs always the minimal \( cg \)
- but much more expensive

BUT

- All basic Petri net properties are known to be generally decidable!
- What we do not know is whether there is a primitive recursive algorithm to decide it practically.

Examples:
\( cg1 .. cg8.spped \)