BrandenburgTechnical
University at Cottbus,
Computer Science Institute

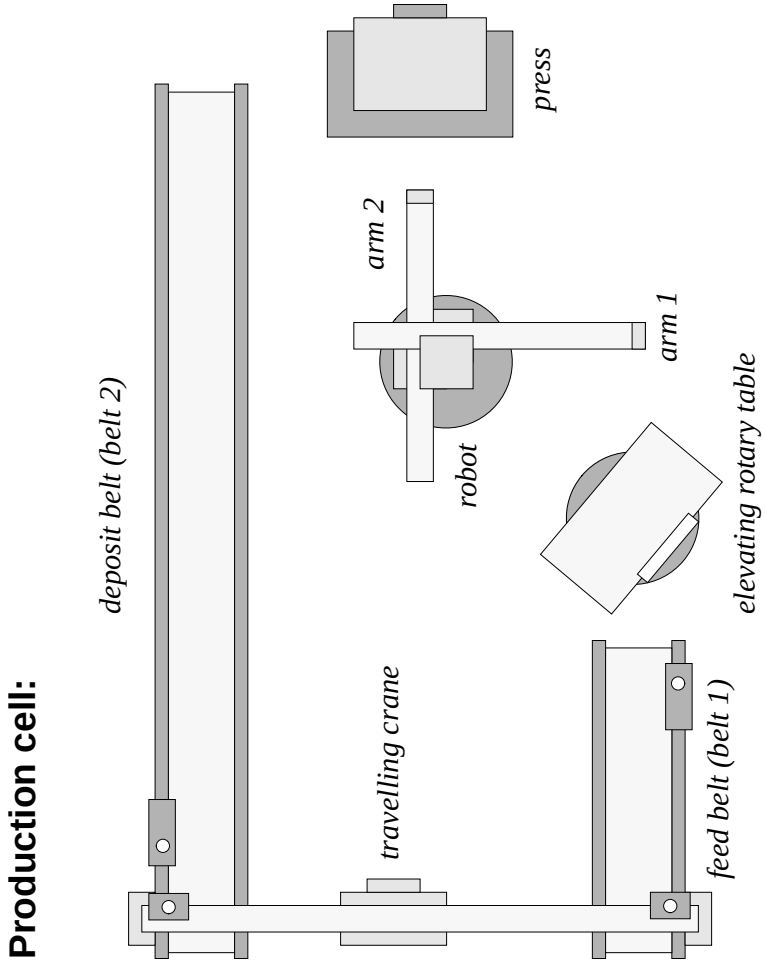
# PETRI NET BASED DESIGN AND ANALYSIS OF REACTIVE SYSTEMS

MONIKA HEINER
PETER DEUSSEN
JOCHEN SPRANGER

{mh, pd, jsp}@informatik.tu-cottbus.de
http://www.informatik.tu-cottbus.de







## Informal safety requirements:

* The press must not be moved downward,
  if sensor 1 is true, and
  t must not be moved upward, if sensor 3 is true.
  -> *Restrictions of machine mobility.*

* The press may only be closed,
  when no robot arm is positioned inside it.
  -> *Avoidance of machine collisions.*

* The feed belt may only convey a blank through its light barrier, if the table is in loading position.
  -> *Blanks are not dropped outside safe areas.*

* Blanks may not be put into the press, if it is already loaded.
  -> *Insurance of a sufficient distance between consecutively processed blanks.*

## additional requirements related to design consistency:

* The robot swivel is either stopped or
  moves in exactly one direction.

* ...





## Objectives:

### Modelling:

* How many basic patterns (Petri net components) are necessary ?
  -> *Small set of flexible, reusable components ?*

* Is it possible to find an adequate environment model ?
  -> *Representation of actuator & (continuous) sensor states ?*

* Suitability of the Petri net editor in use ?
  -> *Additional features ?*

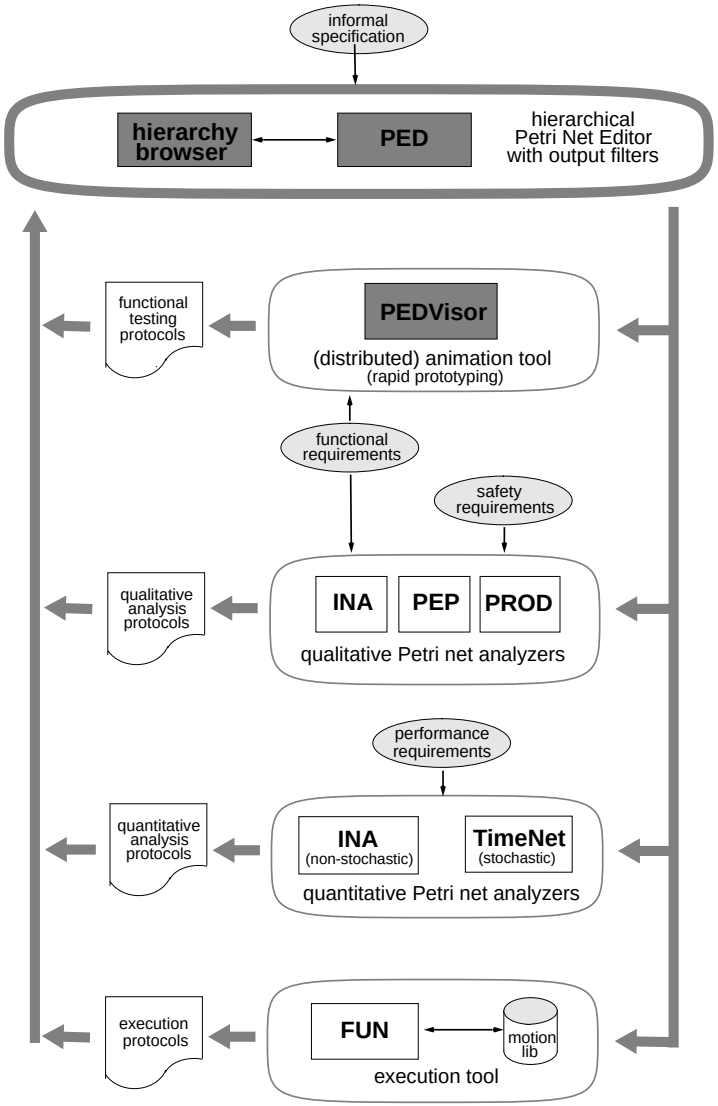
### Analysis:

* Which kinds of function & safety requirements are used ?
  -> *Which temporal operators are really necessary ?*

* Which kinds of analysis techniques are helpful ?
  -> *Recommendable order ?*

* What about the chance to avoid 'state explosion' ?
  -> *How strong do 'alternative' analysis techniques work ?*

# Tool overview:








# Qualitative analysis methods:

- NET REDUCTION
- STRUCTURE PROPERTIES
  - simple properties
  - deadlock trap property
  - state machine allocatable nets
  - rank theorem
- INVARIANTS
  - place invariants
  - transition invariants

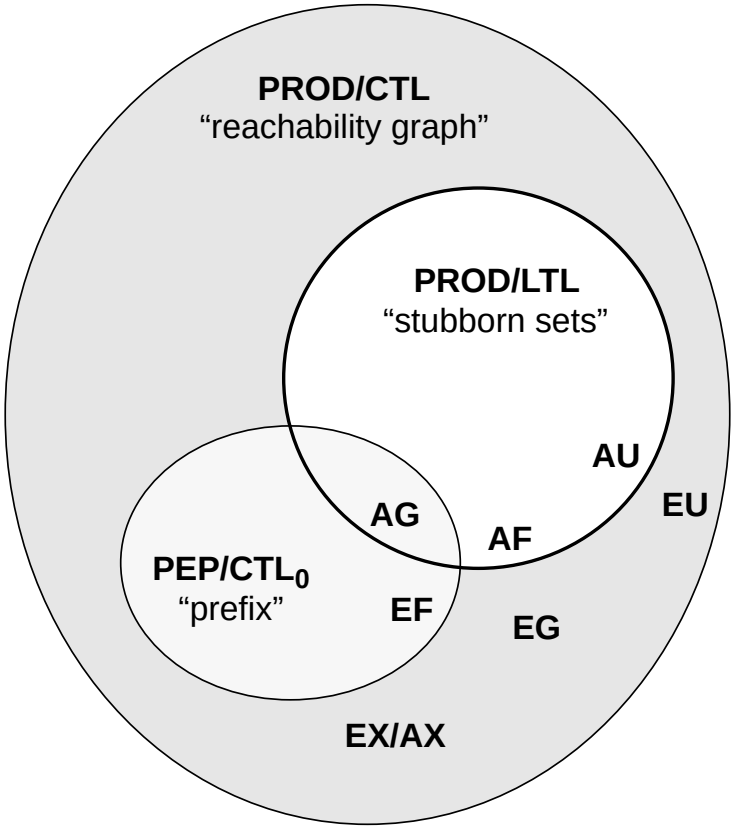
static analysis

- REACHABILITY ANALYSIS
  - (complete) reachability graph
  - reduced state set
    - coverability graph
    - symmetry
    - stubborn set

dynamic analysis

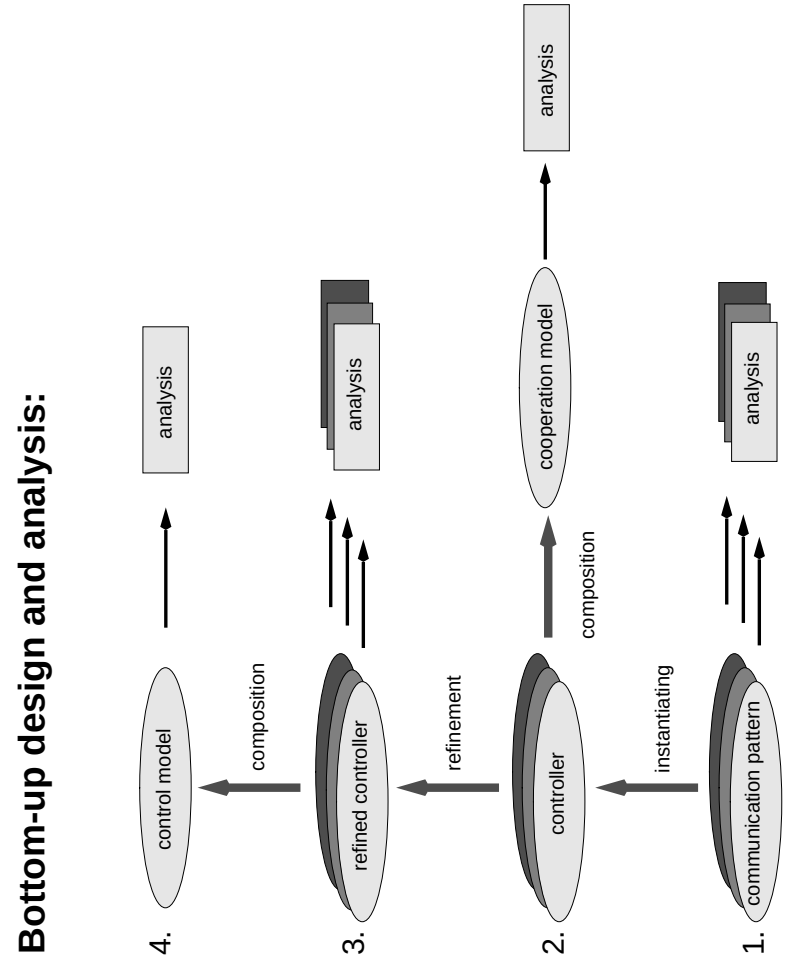
- MODEL CHECKING

# Temporal Logics:





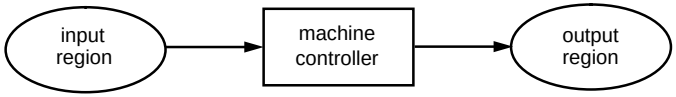


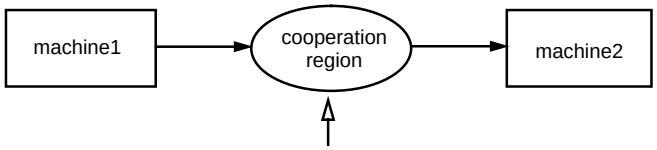
# Basic design principles:

* production cell = pipeline of machines

* each machine
  takes plates from some input places;
  processes them;
  puts plates on some output places;




* cooperation region between two consecutive machines

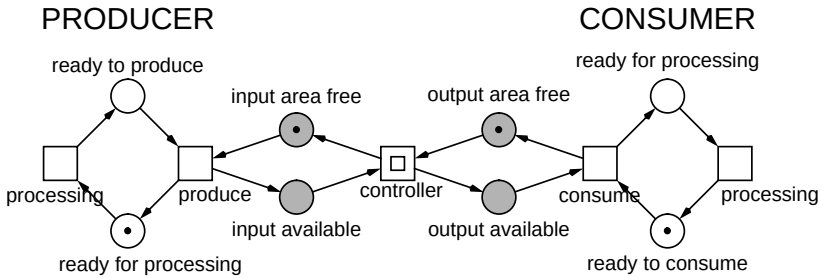


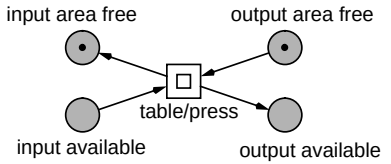

* mutual exclusive shared resources
  robot swivel
  (to rotate both arms)
  physical regions
  (intersection of trajectories
  of different machines)

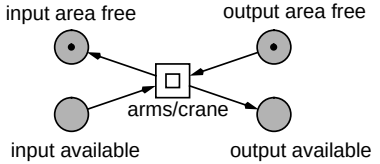


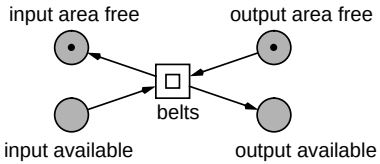


# Producer / consumer relation:



# Three types of cooperation pattern:

**(A) Independent input/output**
arms/crane:

step-wise synchronization with only one of its adjacent controllers,
e.g. crane:

$$\mathbf{G_A}(\neg(\text{ch\_DC\_free} \wedge \text{ch\_DC\_full}) \vee \neg(\text{ch\_CF\_free} \wedge \text{ch\_CF\_full}))$$

**(B) Dependent input/output**
belts:

simultaneous control of input and output region necessary,
e.g. feed belt:

$$\mathbf{G_A}(\text{feed\_belt\_transporting} \rightarrow \neg(\text{ch\_CF\_free} \vee \text{ch\_CF\_full} \vee \text{ch\_FT\_free} \vee \text{ch\_FT\_full}))$$

**(C) Mutually exclusive input/output**
table/press:

the controller must always hold a lock on one of its cooperation regions,
e.g. table:

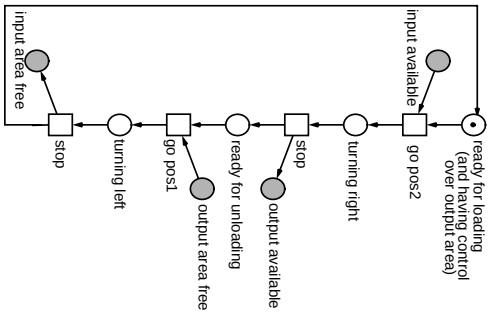
$$\mathbf{G_A}(\neg(\text{ch\_FT\_full} \vee \text{ch\_FT\_free}) \wedge \neg(\text{ch\_TA1\_full} \vee \text{ch\_TA1\_free})$$



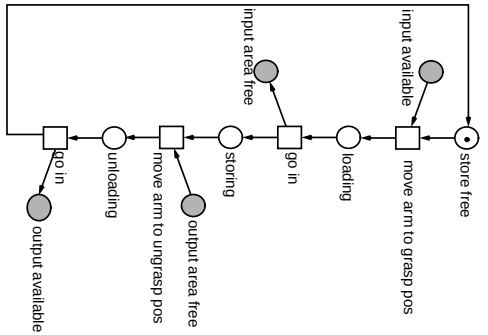


# Three types of cooperation pattern (cont.):

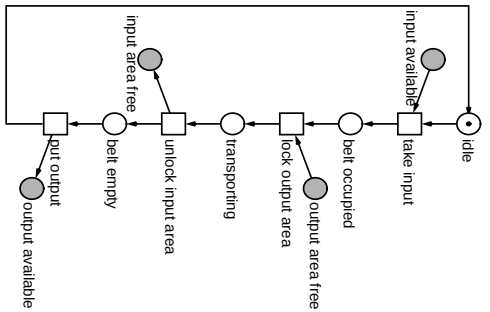
**table / press**
(mutually exclusive input / output)



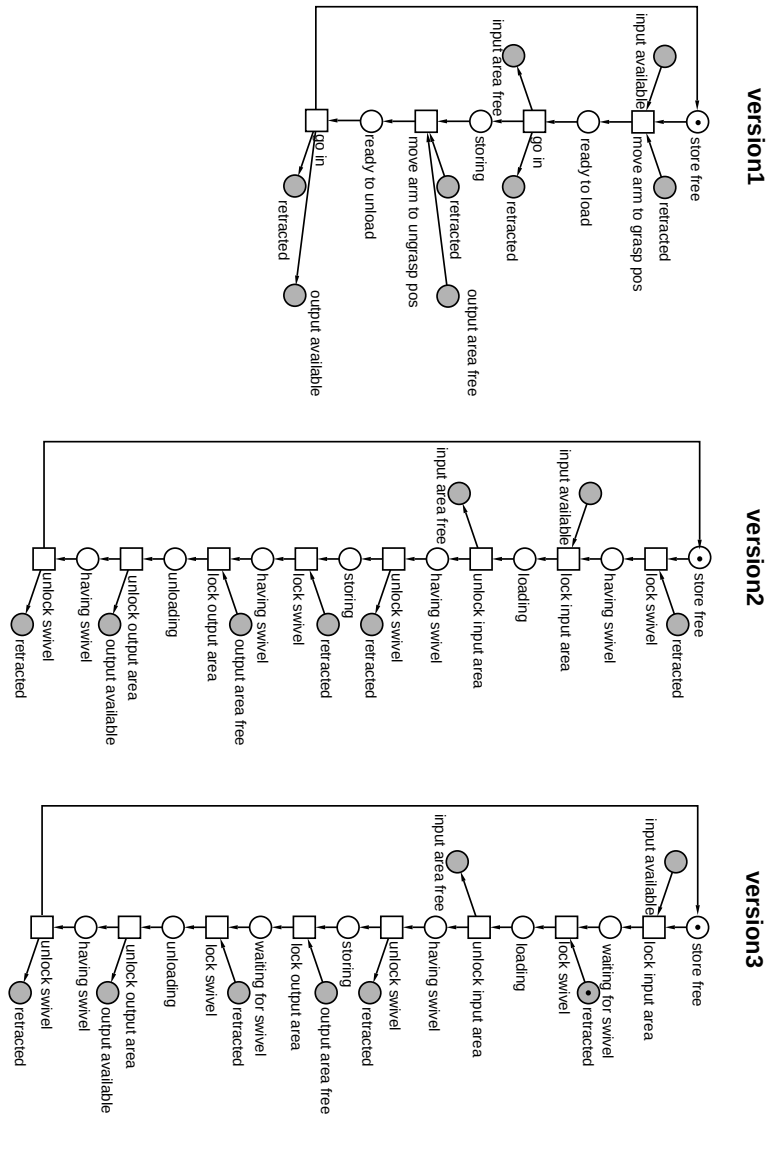

**arms / crane**
(independent input / output)




**feed / deposit belt**
(dependent input / output)



# Three arm versions:








# Source text examples [Casais 94a,b]:

**arm:** procedure to take a plate

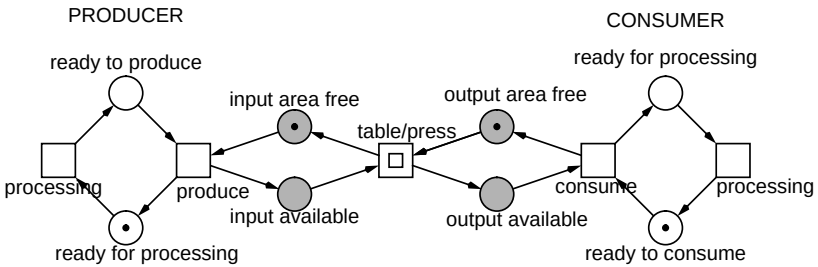
```
Take /* version2 */
    acquire locks on shared resources (swivel)
    acquire lock on input area
        move_arm_to_grasppos
        do_grasp
        go_in
    release lock on input area
    release locks on shared resources (swivel)

Take /* version3 */
    acquire lock on input area
    acquire locks on shared resources (swivel)
        move_arm_to_grasppos
        do_grasp
        go_in
    release lock on input area
    release locks on shared resources (swivel)
```

**belt:** procedure to transport a plate

```
Transport
    acquire lock on input area
    acquire lock on output area
        transport
    release lock on input area
    release lock on output area
```

# table / press:
# (mutually exclusive input / output)
# (with init part)

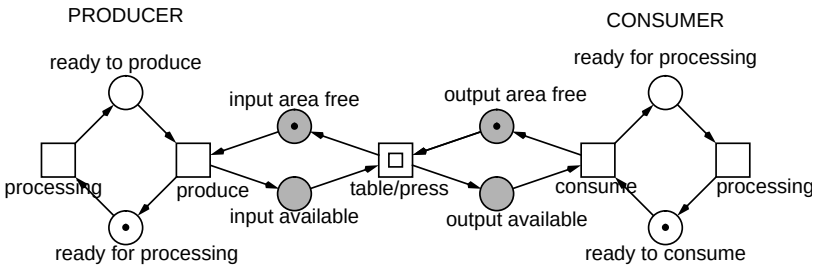



| ORD | HOM | NBM | PUR | CSV | SCF | CON | SC | Ft0 | tF0 | Fp0 | pF0 | MG | SM | FC | EFC | ES |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Y | Y | N | Y | N | N | Y | N | N | N | Y | N | N | N | N | N | Y |
| DTP | SMC | SMD | SMA | CPI | CTI | B | SB | REV | DSt | BSt | DTr | DCF | L | LV | L&S | |
| N | N | N | N | ? | N | Y | ? | N | N | N | N | Y | N | N | N | |



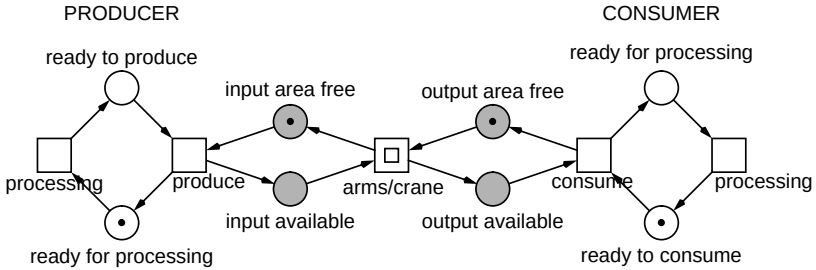


# table / press:
# (mutually exclusive input / output)
# (without init part)




| ORD | HOM | NBM | PUR | CSV | SCF | CON | SC | Ft0 | tF0 | Fp0 | pF0 | MG | SM | FC | EFC | ES |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Y | Y | Y | Y | N | Y | Y | Y | N | N | N | N | Y | N | Y | Y | Y |
| DTP | SMC | SMD | SMA | CPI | CTI | B | SB | REV | DSt | BSt | DTr | DCF | L | LV | L&S | |
| Y | Y | Y | Y | Y | Y | Y | Y | Y | N | N | N | Y | Y | Y | Y | |

# arms / crane:
# (independent input / output)
# (version 2 / 3)

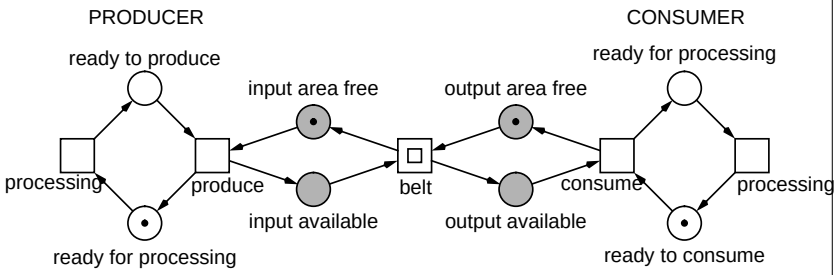



| ORD | HOM | NBM | PUR | CSV | SCF | CON | SC | Ft0 | tF0 | Fp0 | pF0 | MG | SM | FC | EFC | ES |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Y | Y | Y | Y | N | N | Y | Y | N | N | N | N | N | N | N | N | Y |
| DTP | SMC | SMD | SMA | CPI | CTI | B | SB | REV | DSt | BSt | DTr | DCF | L | LV | L&S | |
| Y | Y | Y | N | Y | Y | Y | Y | Y | N | N | N | Y | Y | Y | Y | |





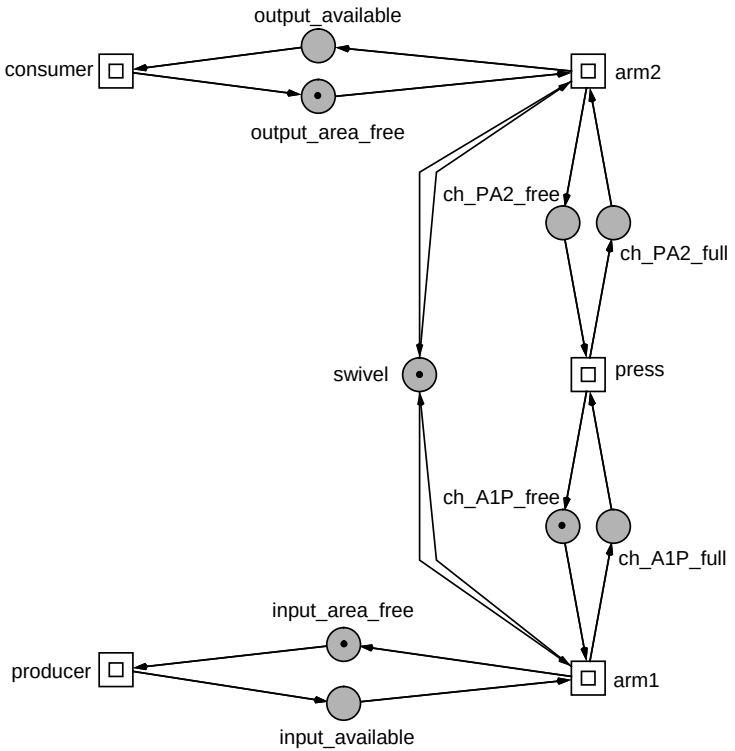
# feed / deposit belt:
# (dependent input / output):




| ORD | HOM | NBM | PUR | CSV | SCF | CON | SC | Ft0 | tF0 | Fp0 | pF0 | MG | SM | FC | EFC | ES |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Y | Y | Y | Y | N | Y | Y | Y | N | N | N | N | Y | N | Y | Y | Y |
| DTP | SMC | SMD | SMA | CPI | CTI | B | SB | REV | DSt | BSt | DTr | DCF | L | LV | L&S | |
| Y | Y | Y | Y | Y | Y | Y | Y | Y | N | N | N | Y | Y | Y | Y | |

# Step-wise composition:

## e.g. subsystem: arm1 - press - arm2 (arms: version2)



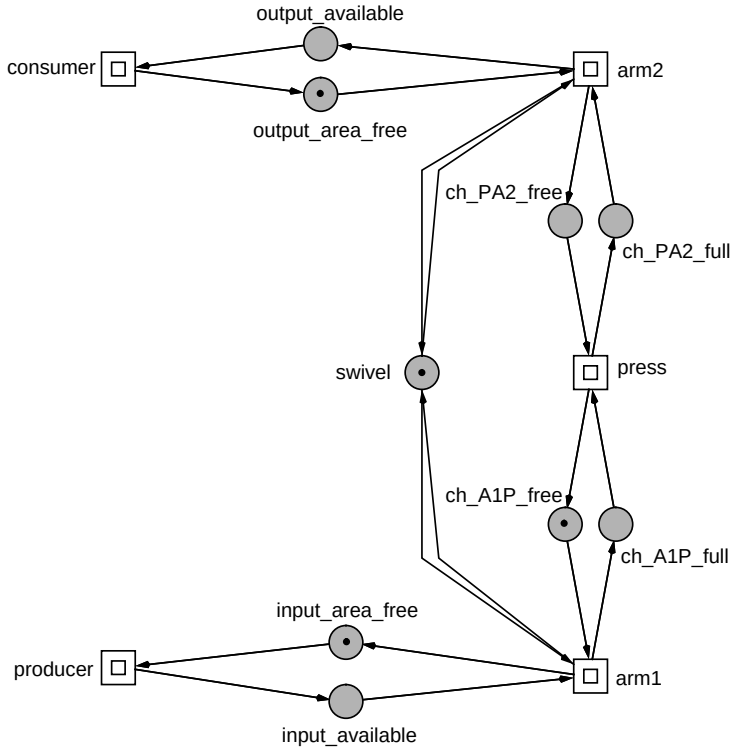

| ORDY | HOMY | NBMY | PURY | CSVN | SCFN | CONY | SCY | Ft0N | tF0N | Fp0N | pF0N | MGN | SMN | FCN | EFCN | ESY |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| DTPY | SMCY | SMDY | SMAN | CPIY | CTIY | BY | SBY | REVN | DStY | BStN | DTrN | DCF? | LN | LVN | L&SN | |





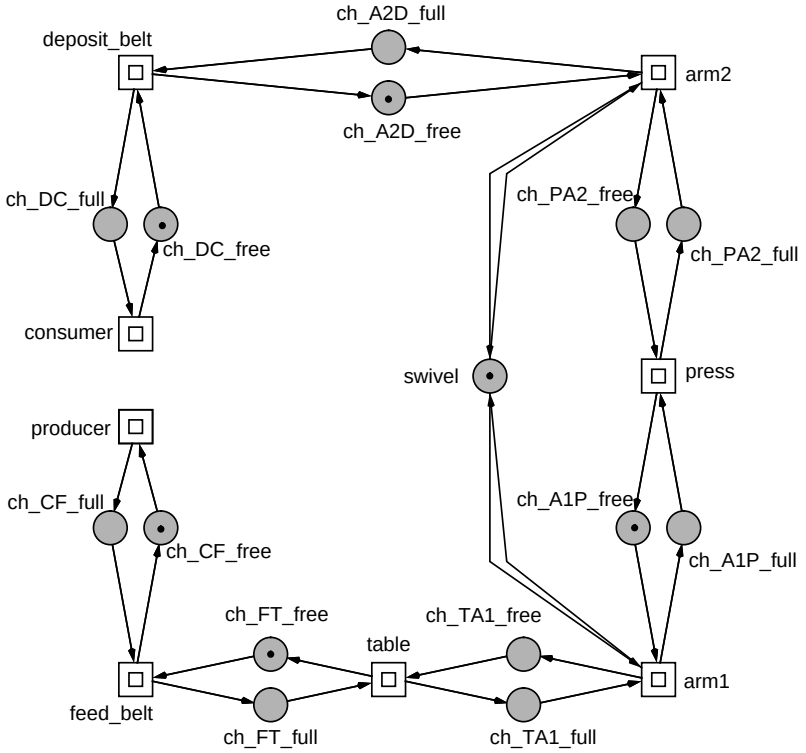
# Step-wise composition:

## e.g. subsystem: arm1 - press - arm2 (arms: version3):




| ORDY | HOMY | NBMY | PURY | CSVN | SCFN | CONY | SCY | Ft0N | tF0N | Fp0N | pF0N | MGN | SMN | FCN | EFCN | ESY |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| DTPY | SMCY | SMDY | SMAN | CPIY | CTIY | BY | SBY | REVY | DStN | BStN | DTrN | DCFN | LY | LVY | L&SY | |

# Coarse structure of the open system:

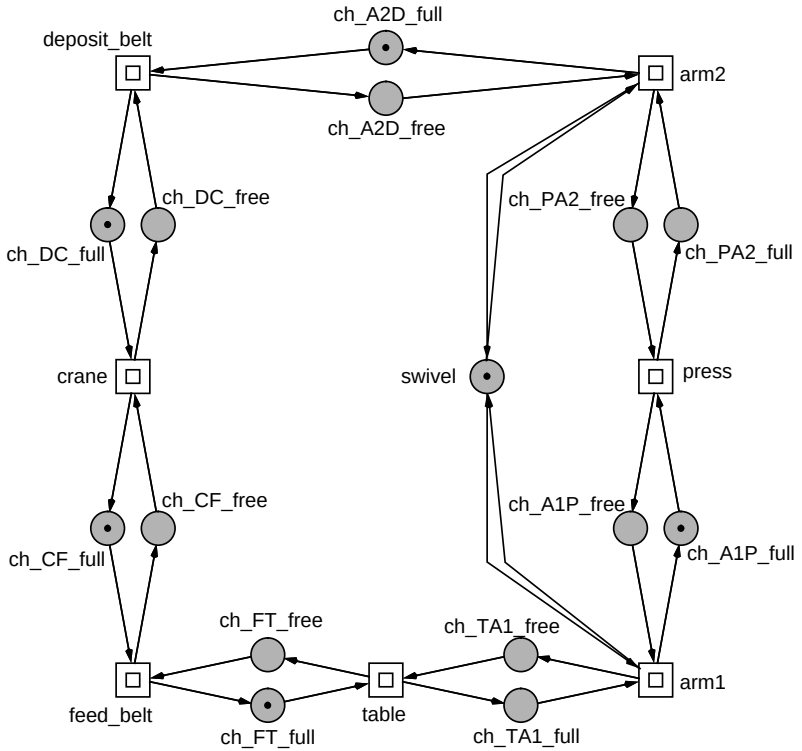



| ORD | HOM | NBM | PUR | CSV | SCF | CON | SC | Ft0 | tF0 | Fp0 | pF0 | MG | SM | FC | EFC | ES |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Y | Y | Y | Y | N | N | Y | Y | N | N | N | N | N | N | N | N | Y |
| DTP | SMC | SMD | SMA | CPI | CTI | B | | SB | REV | DSt | BSt | DTr | DCF | L | LV | L&S | |
| Y | Y | Y | N | Y | Y | Y | | Y | Y | N | N | N | N | Y | Y | Y | |





# Coarse structure of the closed system:




| ORD | HOM | NBM | PUR | CSV | SCF | CON | SC | Ft0 | tF0 | Fp0 | pF0 | MG | SM | FC | EFC | ES |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Y | Y | Y | Y | N | N | Y | Y | N | N | N | N | N | N | N | N | Y |
| DTP | SMC | SMD | SMA | CPI | CTI | B | | SB | REV | DSt | BSt | DTr | DCF | L | LV | L&S | |
| Y | Y | Y | N | Y | Y | Y | | Y | Y | N | N | N | N | Y | Y | Y | |

# Analysis efforts (cooperation model):

| | places/ transitions | DTP[a] | $R_{stub}$[b] | R[b] |
|---|---|---|---|---|
| table / press | | | | |
| with init part | 13 / 9 | (N) | 12 | 28 |
| without init part | 12 / 8 | 28 | 8 | 24 |
| crane | 12 / 8 | 31 | 11 | 48 |
| arms | | | | |
| version 1 | 13 / 8 | 38 | 11 | 48 |
| version 2 | 17 / 12 | 109 | 15 | 112 |
| version 3 | 17 / 12 | 88 | 15 | 96 |
| belts | 12 / 8 | 26 | 8 | 36 |
| subsystem with | | | | |
| arm version 1 | 25 / 16 | 175 | 47 | 640 |
| arm version 2 | 33 / 24 | 3.851 (N) | 75 | 1.984 |
| arm version 3 | 33 / 24 | 725 | 140 | 1.800 |
| open system | 51 / 36 | 1.145 | 299 | 77.760[c] |
| closed system | 51 / 36 | 1.140 | | |
| with 1 plate | | | 36 | 864 |
| with 2 plates | | | 7 2 | 4.776 |
| with 3 plates | | | 94 | 12.102 |
| with 4 plates | | | 98 | 16.362[d] |
| with 5 plates | | | 121 | 12.144[e] |

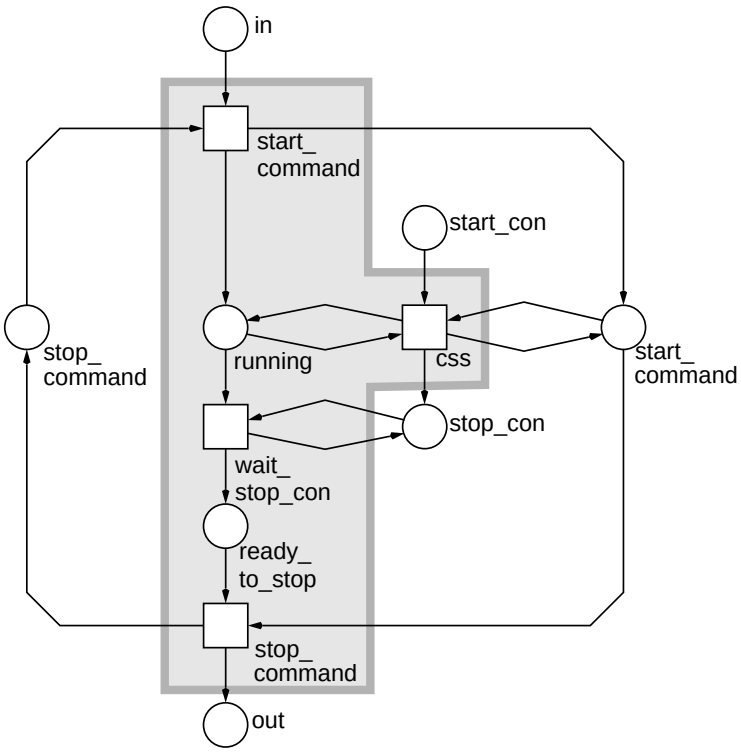
a) processed candidates to check the Deadlock Trap Property
b) number of states generated
c) after about 8.5 h on PC 80486, 75 MHz
d) after about 45' computing time
e) after about 20' computing time

////// just for fun





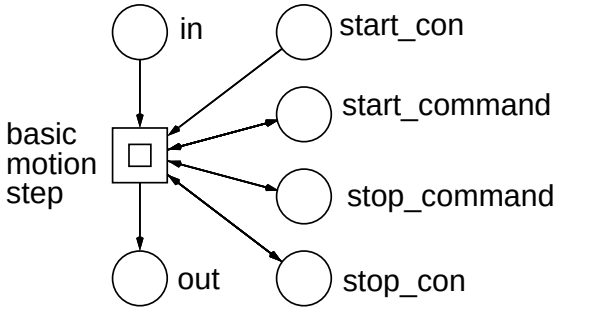
# Basic motion step + environment:




fusion nodes:
- interface
- actuator states
- sensor states

css - change sensor state

# Macro component of basic motion step:




**formal parameters**

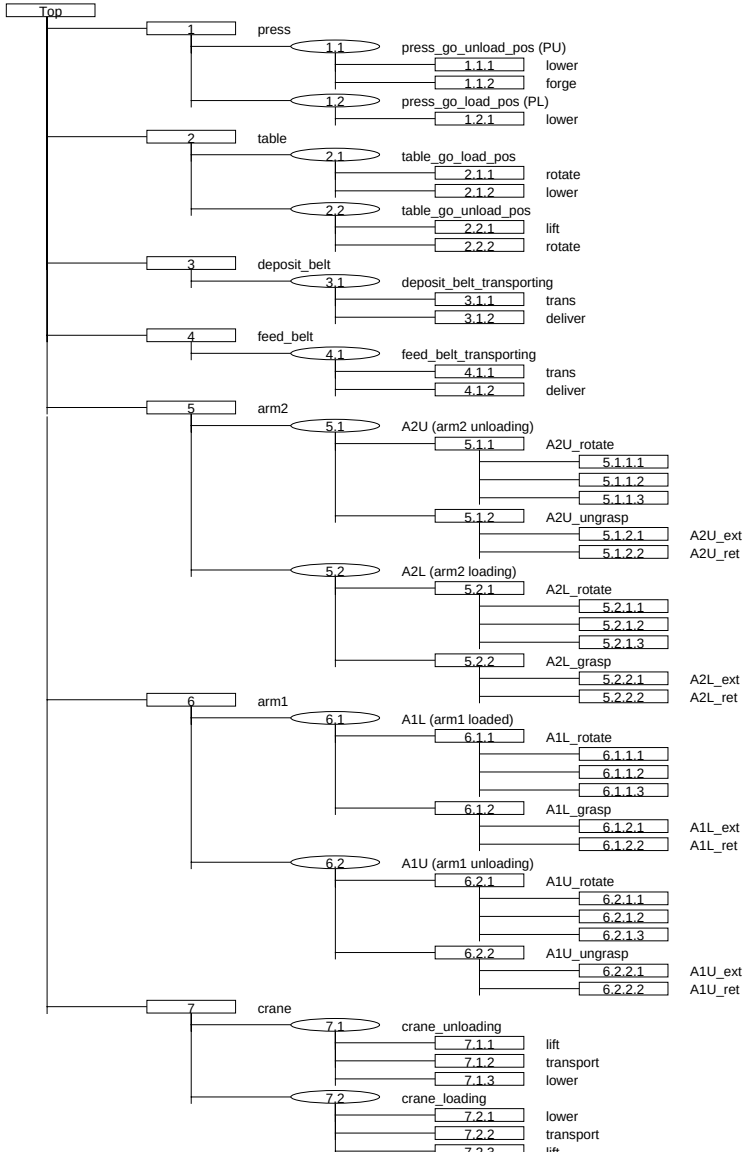
## actual parameters, e.g.:

| press_forge | press_lift |
|---|---|
| press_at_middle_pos | press_at_lower_pos |
| press_upward | press_up |
| press_stop | press_stop |
| press_at_upper_pos | press_at_middle_pos |





# Net hierarchy:



# Temporal logics, samples of analyzed properties:

## General analysis

* liveness
  **AG**(**EF** en( *t* )) for each transition *t* (PEP).

## Special analysis

* design demanded properties, e.g. (PROD/LTL)
  $\mathbf{G}\left(\text{robot\_stop} \dot{\vee} \text{robot\_left} \dot{\vee} \text{robot\_right}\right)$

* functional properties, e.g.
  $\mathbf{EF}\,(\text{arm1\_mag\_on} \wedge \text{arm2\_mag\_on})$ (PEP)

* safety properties, e.g.
  *If a robot arm is loaded, its magnet is not deactivated until the robot is in its unloading position.* (PROD/LTL)
  $\mathbf{G}\,(\varphi \rightarrow \neg\chi\mathbf{U}\psi)$ , where

  $\varphi$ = arm1_mag_on
  $\wedge$ arm1_pickup_angle
  $\wedge$ arm1_pickup_ext

  $\chi$ = arm1_mag_off

  $\psi$ = arm1_release_angle $\wedge$ arm1_release_ext





## Analysis efforts (control model):

| | P/T | PEP | | PROD | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | finite prefix | | full RG | | stubborn (deletion algorithm) | | stubborn (incremental algorithm) | |
| | | C/E | time | size | time | size | time | size | time |
| controllers, e.g. | | | | | | | | | |
| crane | 45/34 | 154/71 | 0.02” | 256 | 0.78” | 51 | 0.16” | 38 | 0.08” |
| composed systems, e.g. | | | | | | | | | |
| open system | 198/176 | 2773/1348 | 5.15” | ? | ? | 798 | 5.90” | 507 | 0.62” |
| closed system | 231/202 | | | | | | | | |
| with 3 plates | | 2009/960 | 3.02” | > 1.7 Mio | > 20 h | 523 | 4.51” | 635 | 0.95” |
| with 4 plates | | 2164/1035 | 3.38” | > 3.1 Mio | > 42 h | 471 | 4.02” | 678 | 1.06” |
| with 5 plates | | 1619/768 | 1.68” | 1,657,242 | ca. 14 h | 585 | 5.05” | 608 | 0.98” |

## Main analysis results:

| | cooperation model | control model |
|---|---|---|
| size<br># pages | 51 P, 36 T<br>8 pages | 231 P, 202 T<br>65 pages |
| general analysis | covered by P-Invariats → BOUNDED | |
| | DTP & ES → LIVE<br>size $(RG_{stub})_5$: 121 → Deadlock-free<br>size $(prefix)_5$: 252 B, 159 E<br>size $(RG)_5$: 12.144 | not ES<br>size $(RG_{stub})_5$: 585 → Deadlock-free<br>size $(prefix)_5$: 1619 B, 768 E → LIVE<br>size $(RG)_5$: 1.657.242 |
| special analysis | PROD/CTL: rich, but too slowly<br>AG ($\neg\varphi$): acceptable<br>AG ($\varphi \rightarrow AF\chi$): slowly | PROD/LTL<br>PEP/$CTL_0$<br>* lack of quantification on pathes<br>* lack of AF, AU |





## Main lessons learnt:

### Modelling:

* management of medium-sized Petri nets
  *-> hierarchical structure + fusion nodes;*

* new editor feature: parameter substitution
  *-> library of reusable Petri net components;*

* environment model unavoidable
  *-> appropriate environment library;*

### Analysis:

* combination of different analysis tools
  *-> general Petri net framework;*

* user guidelines:
  analysis question -> analysis technique(s)
  *-> dedicated Petri net tool kits;*

* separate formal specification of
  function & safety requirements
  *-> dedicated technical language;*

* batch processing of requirement specifications
*-> distributed over different tools & processors;*

# Outlook:

* quantitative analysis
  by different types of time-dependent Petri nets
  Duration Interval Nets,
  Stochastic Nets)

* fault tolerance