Supervisory Control of Manufacturing Systems with Time Specifications

Alexander Schaub

Institute of Automatic Control Engineering
Technische Universität München

Bibliography

1. P.C.Y. Chen, W.M. Wonham. Stable supervisory control of flexible manufacturing systems with fixed supply and demand rates. Int. J. Prod. Res., 2001, Vol. 39, No.2, 347-368.
2. B.A. Brandin and W.M. Wonham. Supervisory Control of Timed Discrete-Event Systems. IEEE Transactions on Automatic Control, Vol. 39, No. 2, February 1994.
3. P.J.G. Ramagde, W.M. Wonham. The control of Discrete Event Systems. Proceedings of the IEEE, Vol 77, No.1, January 1989.

Overview

- **Introduction**
- **Methods:** DES, TDES, Supervisory Control, Synchronization Operators, Procedure Scheme.
- **CNC Machine:** Model, Specifications, Resulting Supervisor, Activities.
- **Conclusions**

Introduction

Manufacturing system:

- Flexible: different tasks on different types of parts.
- Reconfigurable processors: it takes time to initialize another task.
- Limited buffers.
- Fixed supply and demand rate.
- Hard-real-time: completion of a task within a given deadline is guaranteed.

Supervisory controller manages the system.

Both modeled as timed discrete-event systems. Formal constructive method to:

1. decide whether a stable supervisory control exists.
2. compute the stable supervisory control, if it exists.

Discrete Event Systems

$G_{act} = (\Sigma_{act}, A, \delta_{act}, a_0, A_m)$

- Σ_{act}: finite alphabet of event labels (events)
- A: Activity set containing activities a (states).
- δ_{act}: Activity transition function.

$$\delta_{act} : \Sigma_{act} \times A \rightarrow A$$

- Activity transition $\sigma : a' = \delta_{act}(\sigma, a)$.
- a_0: initial activity.
- $A_m \subseteq A$: subset of marker activities.

Timed Discrete-Event Systems

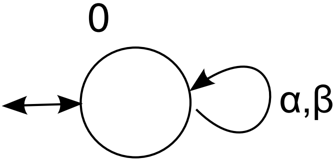
$G = (\Sigma, Q, \delta, q_0, Q_m)$

- $q_0 \in Q$, $Q_m \subseteq Q$
- Discrete time event *tick*: $\Sigma := \Sigma_{act} \cup \{tick\}$
- Lower l_σ and upper u_σ time bounds for each transition σ.
- Two possible types:
  1. prospective events σ_{spe} with $0 \leq l_\sigma \leq u_\sigma < \infty$
  2. remote events σ_{rem} with $0 \leq l_\sigma < u_\sigma = \infty$
- Timed event triples $\Sigma_{tim} := \{(\sigma, l_\sigma, u_\sigma) | \sigma \in \Sigma_{act}\}$
- Every state q is related to an activity and a timer: $q = (a, \{t_\sigma | \sigma \in \Sigma_{act}\})$

Example of Timed Discrete-Event Systems

$G_{act} = (\Sigma_{act}, A, \delta_{act}, a_0, A_m)$

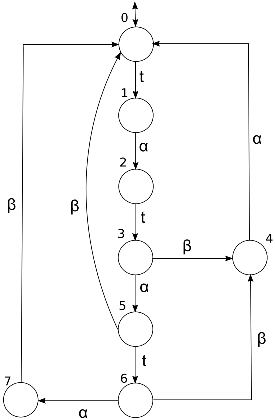
- $\Sigma_{act} = \{\alpha, \beta\}$
- $a_0 = 0$
- $\delta_{act}(\alpha, 0) = \delta_{act}(\beta, 0) = 0$
- $A = A_m = \{0\}$



Example of Timed Discrete-Event Systems

$G = (\Sigma, Q, \delta, q_0, Q_m)$

- $\Sigma = \{\alpha, \beta, \mathit{tick}\}$
- $q_0 = a_0 = 0$
- $\delta_{act}(\alpha, 0) = \delta_{act}(\beta, 0) = 0$
- $Q_m = \{0\}$
- Timed events: $(\alpha, 1, 1), (\beta, 2, 3)$.
- $Q = \{0\} \times \{0, 1\} \times \{0, 1, 2, 3\}$, $|Q| = 8$.



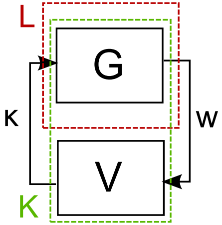
$State$	0	1	2	3	4	5	6	7
$[t_\alpha, t_\beta]$	[1,3]	[0,2]	[1,2]	[0,1]	[0,3]	[1,1]	[0,0]	[1,0]

Supervisory Control of DES

- Inclusion of all possible transition sequences of (T)DES G in its language $L(G)$.
- Representation of the supervisor by an automaton V monitoring G.
- Disablement of certain events in transition structure of G to meet certain specifications.
- Differentiation between *controllable* and *uncontrollable* events: $\Sigma_{act} = \Sigma_c \cup \Sigma_u$
- Possibility to *force* some events Σ_{for}.

Supervisory Control of DES

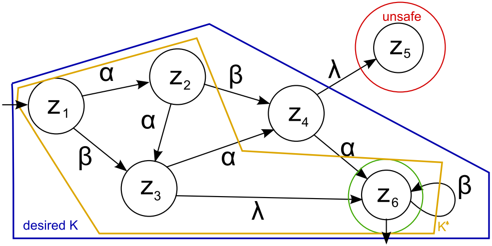
- Specification of the control input for every possible string w of G by a supervisor map s: $\kappa = s(w)$
- Closed loop behavior of the system $L(V|G) =: K$
  1. $\epsilon \in K$
  2. $w\sigma \in K$ iff $w \in K, \sigma \in V(w), w\sigma \in L$



Example for Supervisory Control of DES

- Supremal controllable language $K^{\uparrow}$: largest controllable language $K^{\uparrow} \subseteq K$.
- $\Sigma_c = \{\alpha, \beta\}$, $\Sigma_u = \{\lambda\}$.
- $L = (\alpha(\alpha\alpha + \beta)(\lambda + \alpha) + \beta(\alpha\lambda + \alpha\alpha + \lambda))\beta^*$
- $L_m = (\alpha(\alpha\alpha + \beta)\alpha + \beta(\alpha\alpha + \lambda))\beta^*$
- $K^{\uparrow} = (\alpha\alpha + \beta)\lambda\beta^*$

- K is called controllable if: $\overline{K}\Sigma_u \cap L \subseteq \overline{K}$



Supervisory Control of TDES

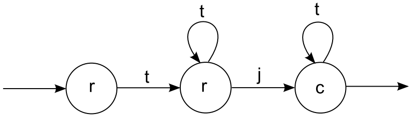
- Considering time bounds (l_σ, u_σ) as specifications.
- Minimal restrictive supervisor: disabling certain events only if necessary $\rightarrow$ creation of largest possible subset of legal sequences.
- Software TTCT available to create, combine TDESs and to compute the supremal controllable sublanguage of a given language.
- Creation of a supervisory TDES by three main steps: *sync*,*meet* and *supcon*

Parallelisation of Generators (sync)

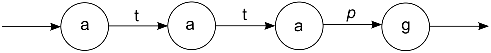
- Synchronization of two TDESs: $G_3 = G_1 \| G_2$
- For all $\sigma \in \Sigma_{3,act} : \sigma \in (\Sigma_{1,act} - \Sigma_{2,act}) \cup (\Sigma_{2,act} - \Sigma_{1,act})$
- Timed events must be synchronisable:
  1. $\sigma \in \Sigma_{1,act} \cap \Sigma_{2,act}$
  2. $(l_\sigma, u_\sigma) = (\max(l_{1,\sigma}, l_{2,\sigma}), \min(u_{1,\sigma}, u_{2,\sigma}))$

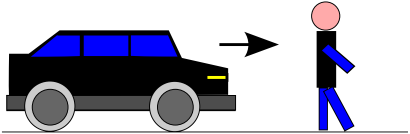
Example: the Endangered Pedestrian

- $G = (\Sigma, Q, \delta, q_0, Q_m)$
- PED $= (\{j\}, \{r, c\}, \{[r, j, c]\}, r, \{c\}); \ \Sigma_{tim} = (j, 1, \infty)$:
- $j =$'jump',$r =$'road', $c =$'curb'.



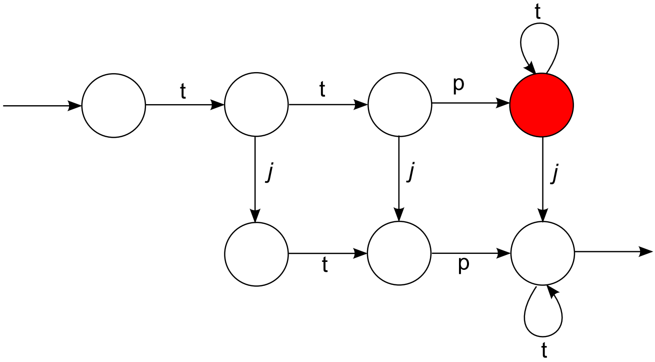
- CAR $= (\{p\}, \{a, g\}, \{[a, p, g]\}, a, \{g\}; \ \Sigma_{tim} = (p, 2, 2)$:
- $p =$'pass', $a =$'approaching', $g =$'gone by'.





Example: the Endangered Pedestrian

- $j =$'jump', $p =$'pass'.
- $CP = sync(CAR, PED)$, $\Sigma_{for} = \{j\}$.
- TDES of CP:

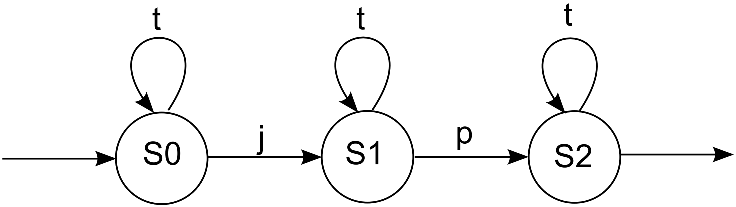


Restriction of Synchronization on Common Symbols (meet)

- $G_3 = G_1 \sqcap G_2$
- Finding a TDES fulfilling all conditions of different TDESs simultaneously.
- Special case of *sync* with $\Sigma_1 = \Sigma_2$.

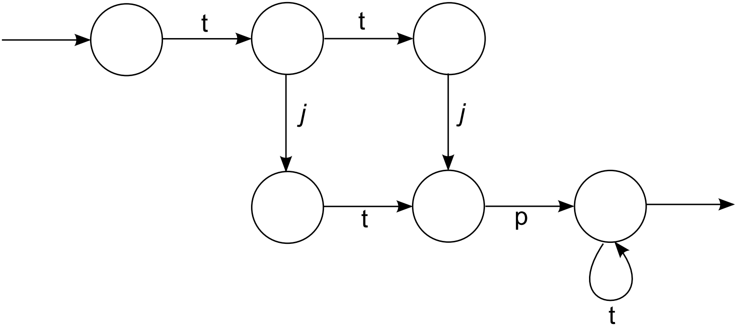
Example: Saving the Pedestrian

- For safety: *Jump* before the car *passes*.
- SAVE $= (\{j, p\}, \{s0, s1, s2\}, \{[s0, j, s1], [s1, p, s2]\}, s0, \{s2\})$, $\Sigma_{tim} = \{(j, 0, \infty), (p, 0, \infty)\}$:
- $j =$'jump', $p =$'pass'.



Example: Saving the Pedestrian

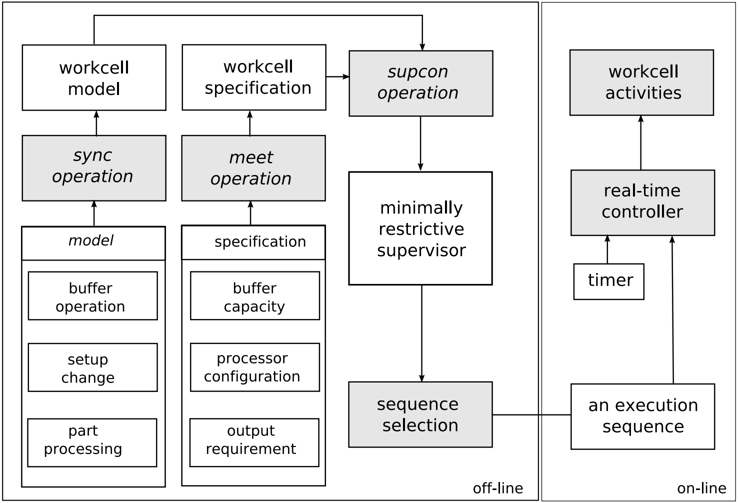
- Adding the safety specification to the endangered pedestrian example.
- $CPSAVE = meet(CP, SAVE)$:



Computation of $K^{\uparrow}$ (supcon)

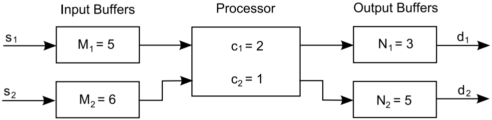
- Finding the supremal controllable sublanguage for a certain model TDES G and its specification TDES S.
- *supcon*: $V = \Phi(G, S)$
- Every contained sequence observes the specifications.
- Erasure of all undesired transitions paths.
- Possibility of an empty supervisor $\longrightarrow$ specifications too hard.

Procedure for computing a supervisor



Task

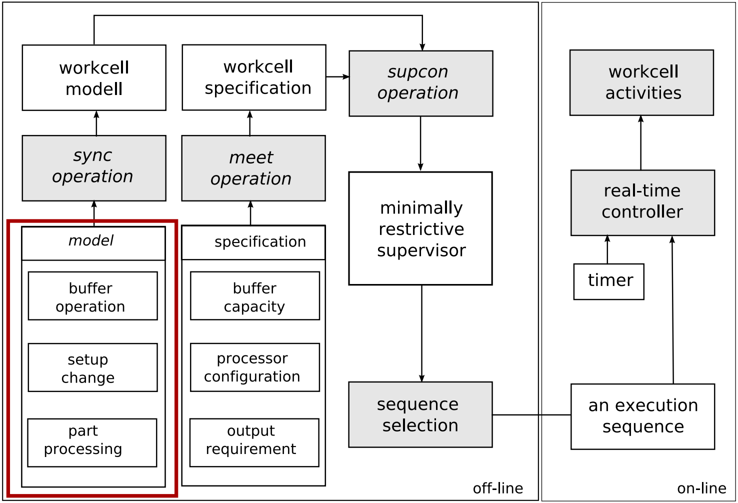
- Processor P, input buffers F_1, F_2 and output buffers H_1, H_2.
- Input rates: $s_1 = 0.5$ parts/min, $s_2 = 1/3$ parts/min.
- Output rates: $d_1 = 1/3$ parts/min, $d_2 = 0.25$ parts/min.



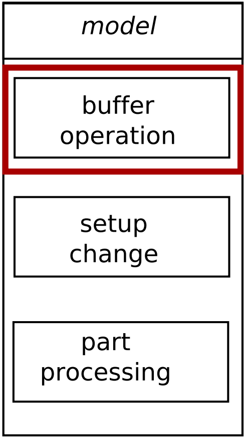
Setup time

from \ to	a_0	a_1	a_2
a_0	0	0	1
a_1	0	0	—
a_2	1	—	0

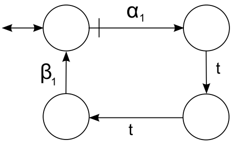
Model


workcell
modell
workcell
specification
supcon
operation
sync
operation
meet
operation
model
buffer
operation
setup
change
part
processing
specification
buffer
capacity
processor
configuration
output
requirement
minimally
restrictive
supervisor
sequence
selection
off-line
workcell
activities
real-time
controller
timer
an execution
sequence
on-line

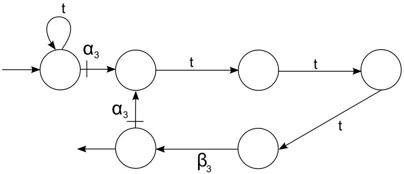
Model



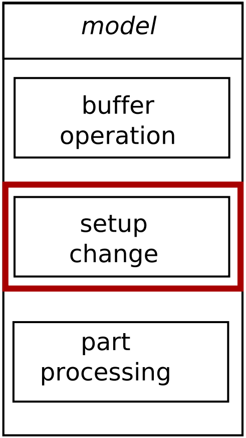
- Input buffer G_{F_1} ($\alpha_1 =$'request', $\beta_1 =$'enter'):



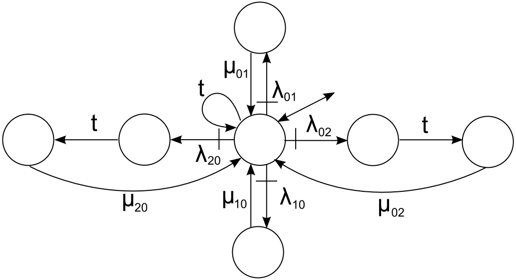
- Output buffer G_{H_1} ($\alpha_3 =$'leave', $\beta_3 =$'fetched'):



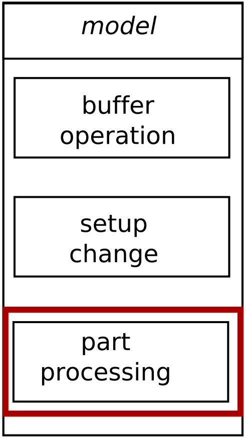
Model



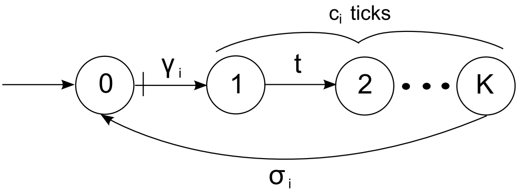
- Processor reconfiguration G_r ($\lambda =$'reconfig', $\mu =$'finished reconfig'):



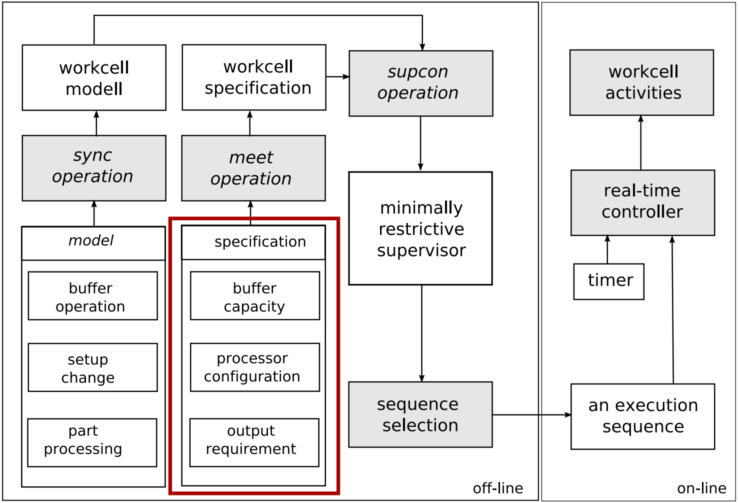
Model



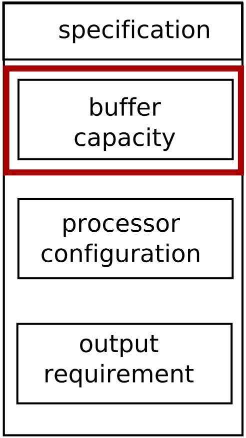
- Part processing G_p ($\gamma =$'produce', $\sigma =$'finished producing'):



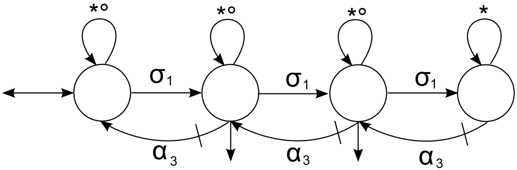
Specification



Specification



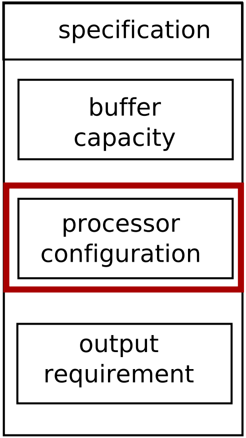
- Output buffer specification S_{H_1} ($\alpha_3 =$'leave', $\sigma_1 =$'finished producing'):



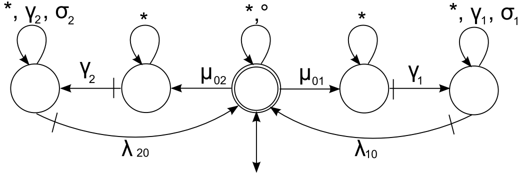
*: t, α_1, β_1, α_2, β_2, α_4, β_4, γ_2, σ_2, λ_{01}, μ_{01}, λ_{10}, μ_{10}, λ_{02}, μ_{02}, λ_{20}, μ_{20}

°: β_3, γ_1

Specification



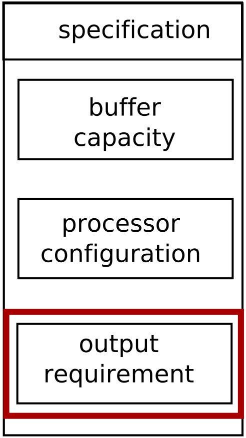
- Proper configuration of processor S_P ($\gamma =$'produce', $\lambda =$'reconfig', $\mu =$'finished reconfig'):



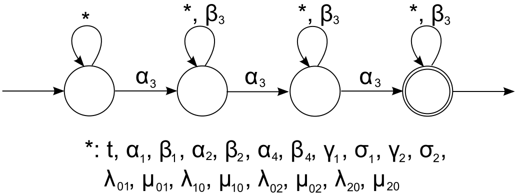
$*$: t, α_1, β_1, α_2, β_2, α_3, β_3, α_4, β_4

$\circ$: λ_{01}, μ_{10}, λ_{02}, μ_{20}

Specification



- Output specification for type-1 parts S_{O_1} ($\alpha_3 =$'leave', $\beta_3 =$'fetched'):

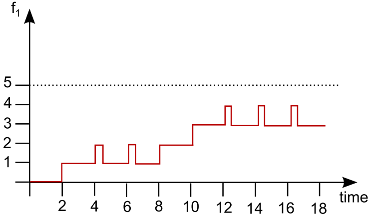


Resulting Supervisor

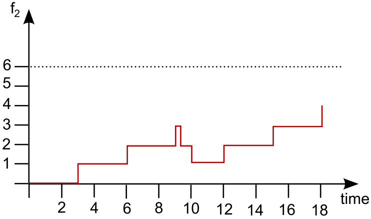
- *sync* all models: $G_w = G_{F_1} \| G_{F_2} \| G_{H_1} \| G_{H_2} \| G_p \| G_r$.
- Model G_w consists of 35280 states and 129792 transitions.
- Receiving V by repeated application of *supcon*:
  $V = \Phi(\Phi(\Phi(\Phi(G_w, S_p), S_{F_1} \sqcap S_{F_2}), S_{H_1} \sqcap S_{H_2}), S_{O_1} \sqcap S_{O_2})$.
- Supervisor V consists of 2538 states and 5945 transitions.
- One possible sequence: $\alpha_1\beta_2\alpha_2\sigma_1\gamma_1 tt\beta_1\alpha_1\sigma_1\lambda_{10}\mu_{10}\lambda_{02}t\ldots$

Example's Activities

- Level of type-1 part input buffer:

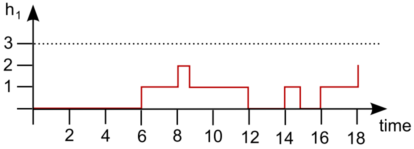


- Level of type-2 part input buffer:

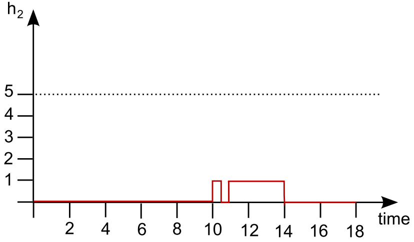


Example's Activities

- Level of type-1 part output buffer:

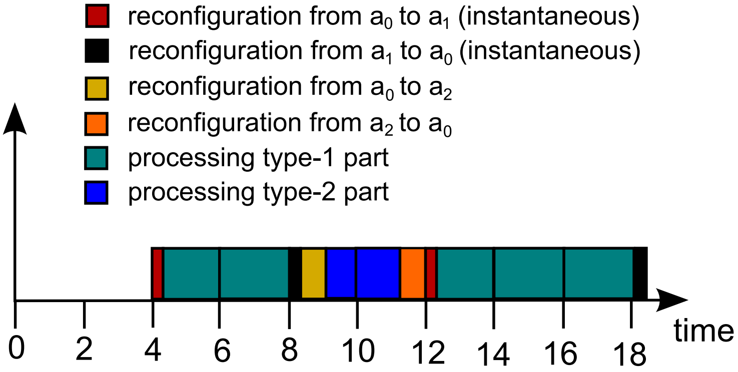


- Level of type-2 part output buffer:



Example's Activities

- Processor activity under supervisory control:



Conclusions

- Ability to find a supervisor containing all safe sequences.
- Minimally restrictive controller $\rightarrow$ optimization possibility.
- Computation of the supremal controllable sublanguage in polynomial time.
- Disadvantage: exponential increase of the number of states of a composite TDES.
- Suggested solution in the paper:
  - modular synthesis: set of concurrently operating modular supervisors.