



## ARGESIM BENCHMARKS

### A Petri Net approach to ARGESIM Benchmark C2 'Flexible Assembly System' using the MATLAB PetriSimM toolbox

Thomas Löscher, Christoph Habersohn, Gregor Hinker, TU Wien, Austria

*thomas@loescher.at, christoph.habersohn@gmail.com, gregor.hinker@esperanto.at*

SNE 17/3-4, December 2007

59

**Simulator:** The sophisticated MATLAB PetriSimM toolbox offers the capabilities of analysis, modelling and simulation of Petri Nets. Furthermore it is possible to optimise scheduling problems based on Timed, Coloured, and Stochastic Petri Nets. The open source MATLAB PetriSimM toolbox can be used for education in a graduate level and for modelling and simulating real life processes of discrete event systems in equal measure. The toolbox is embedded in the MATLAB environment and its usage requires version 7.0 or higher. In this toolbox a sophisticated way of defining firing sequences and priorities is implemented. Furthermore, results can be shown with Gantt charts and marking plots or the data can be exported for external post-processing.

**Model:** An assembly system described in Comparison 2 is an atypical application for Petri Nets. The aim of this simulation was to prove that even such a process can be simulated by a discrete system based on Petri Nets, although the programming and processing efforts are considerably high. The C2 comparison "Flexible Assembly System" consists of two times two conveyors, eight assembly stations and two shifting parts.

The priorities at the switches in front of each assembly station have been assigned as follows:

1. Leaving the assembly station
2. Entering the assembly station
3. Forwarding along the conveyor

At the end of the conveyor, the shifting part obeys similar rules:

1. Shift directly from station to station
2. Shift from station to conveyor
3. Shift from conveyor to station
4. Shift from conveyor to conveyor

All these prioritizations can be represented as conflicts of the Petri Net. Therefore, priorities are as-

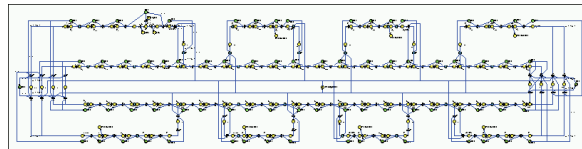


Figure 2. Whole Model

signed to selected transitions to solve these control strategies by the use of the basic properties of Petri Nets.

In contrast to most usual simulation strategies the palets are not modelled as places, but as combination of multiple coloured tokens, the colours making statements about the state of procession of the palet. Therefore, following one single defined palet through the itinerary is not evidently possible and post-processing of the node data becomes necessary.

Figure 1 shows a screenshot of the whole model. Each procession unit and the discretized conveyor elements are designed using three-places-units (see figure 2). A "redlight" place ("frei") signalizes whether entrance into the unit is possible or not. During the given processing time the tokens pass from the entranceplace to the exitplace. For a more concise data recording, the processing unit additionally disposes of a place ("Bearbeitet") indicating when the palet is assembled.

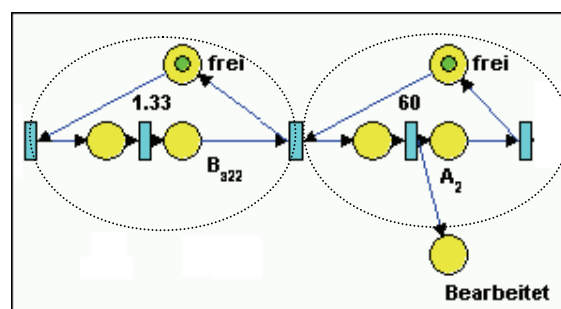
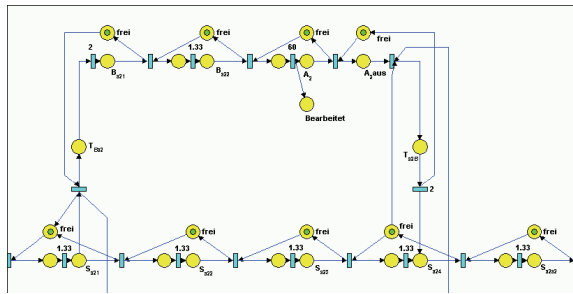


Figure 1. Three-places-units: conveyor unit and processing unit

**Figure 3.** An assembly station

An assembly station (Figure 3) consists of successive conveyor units and one procession unit, both mentioned above. In order to enable simulations with a number of palets exceeding the band capacity (40), an additional tool had to be introduced: this control unit permits loading of new palets from the assembly stations to the main conveyor only in case of free capacity and thus avoids system deadlocks.

The complexity of the model (243 places, 8 colors) results in largescaled matrices (from 1145 x 25.670 to 1145 x 612.434) that increase both processing time and quantity of data: each single simulation took between one and three days on a home computer; furthermore, up to 29 GB of data were produced during one simulation session, in total 110 GB. Processing was therefore delegated to the phoenix-cluster of the TU Wien.

**A-Task:** Due to systematic restrictions of the method and the model it is not possible to follow one single palet throughout the process; hence, the statistical evaluation was done by external post-processing of the logfiles of every place node based on small C programs. Following the recommendation, the data used was from the 120th to the 600th minute of simulation.

**B-Task:** The simulation time was eight hours.

Number of palets	Total throughput	Average throughput time [s]
5	480	292.50
10	960	292.50
13	1333	273.29
<b>14</b>	<b>1440</b>	<b>272.50</b>
15	1440	292.50
20	1444	391.92
40	1441	792.04
60	1433	1198.66

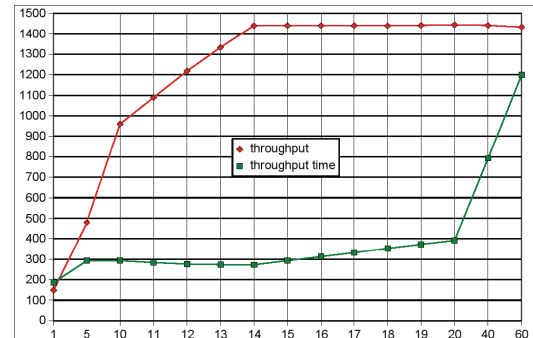
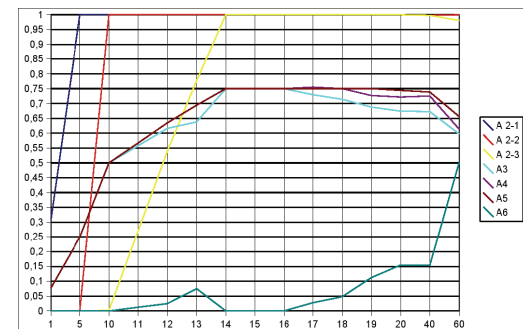
**Table 1.** System behaviour: through put time (extract)**Figure 4.** System behaviour: throughput time/throughput**Figure 5.** System behaviour: assembly stations' utilization

Table 1 show an extract of the throughput times and the graph of Figure 4 aims to provide an overview of the system's behaviour:

Larger calculations than with 40 palets are purely theoretic, since they exceed the system's capacity. This was made possible by the deadlock control unit.

**C-Task:** As shown in the following graph (figure 5), the utilization of stations A2–A6 reaches its possible maximum (100% utilization of “bottleneck” stations A2-1–A2-3) starting at 14 palets. In this configuration, the highest possible throughput (1440 palets; at some points slightly higher numbers may have encountered due to the delayed measurement) can be obtained at the lowest throughput time (272.50 seconds).

**Résumé:** This work shows an alternative modelling approach using Petri nets and the sophisticated PetriSimM toolbox.

**Corresponding author:** Thomas Löscher,  
Department of Analysis and Scientific Computing  
Vienna University of Technology,  
Wiedner Hauptstraße 8-10, 1040 Vienna, Austria  
[thomas@loescher.at](mailto:thomas@loescher.at)

Received: June 12, 2007

Revised: October 10, 2007

Accepted: October 20, 2007