Reduction and Design of
Well-behaved Concurrent Systems\textsuperscript{1}

Jörg Desel
Institut für Informatik der TU München
Arcisstr. 21, D-8000 München 2

Abstract
It is shown that each live and safe free-choice system without frozen token can be reduced either
to a live and safe marked T-graph (marked graph) or to a live and safe marked P-graph (state
machine). The four proposed reduction rules are purely local and preserve the behavioural
properties in both directions. Hence the method can be used for both, effective analysis and
correct design.
The class of systems which can be reduced to marked P-graphs (T-graphs, respectively) can
be characterized without using the reduction rules by their P- and T-components. The two
classes are not disjoint; systems in the intersection of the classes can be reduced to a unique
systems with only two elements.

1 Introduction
Studying transformations and reductions of models for concurrent systems we were led to
distinguish distributed systems and parallel devices.
Often a distributed system is a collection of subsystems which are connected by means of
communication links. Concurrency is an inherent phenomenon of distributed systems since
the subsystems work mutually independent as long as they do not wait for a communication.
The design of well-behaved distributed systems is a severe problem. The analysis with respect
to liveness, fairness or safety properties is known to be a highly complex task.
Concurrency plays a completely different role in parallel devices. The core idea of parallel
devices is the parallelization of independent actions to gain a speed-up of computations. In
contrast to distributed systems the presence of concurrency in parallel devices does not cause
synchronization problems such as danger of deadlocks.
Both aspects of concurrency can appear together when parts of distributed systems behave
like parallel devices.
Distributed systems should have the capability to react differently on different inputs. Hence
usually we find the notion of alternatives or choice. The selection of one alternative (i.e.,
a concrete input) may have global consequences. As with concurrency, a specific kind of
alternatives can be distinguished where each choice has only local effects.
We shall concentrate on concurrent systems which are well-behaved in a sense to be defined.
The behaviour restriction includes aspects of liveness, safety and fairness. It turns out that
local aspects of concurrency and alternatives of well-behaved systems can be identified locally.

\textsuperscript{1}Work supported partly by the Esprit Basic Research Action No. 3148: DEMON.
We propose reduction rules which can be seen as a grammar for well-behaving local parts of distributed systems. These rules include concepts such as:

- `action → action; action`
- `action → PARBEGIN action PAREND`
- `action → CHOICEBEGIN action CHOICEEND`

The exhaustive application of the rules leads to the abstraction of the internal behaviour of local devices. Moreover, subparts of the systems which might be distributed but behave like local devices can be reduced without affecting the well-behavedness. The (four) rules proposed in this paper are purely local: Substructures which allow the application of a rule are limited in size and can easily be identified. The respective reductions are very simple. The main result is that maximally reduced well-behaved systems either exhibit no concurrency or have no alternatives. In other words:

*In well-behaved systems either concurrency or choice is purely local.*

The result has two consequences: the analysis of concurrent systems (w.r.t. well-behavedness) can be reduced to the analysis of systems without concurrency or without choice which is much simpler as the general case. The design of well-behaved concurrent systems can always be done in two steps: First the global topology of concurrency or of choice has to be designed. Then the grammar rules can be applied to get an arbitrary well-behaved system.

Formally, we use Petri-Nets for modelling concurrent systems and we consider the class of *live and safe free choice systems without frozen token* to be the class of *well-behaved systems*. With these notions the results read as follows:

Each well behaved system can be either reduced to a well-behaved S-graph (which exhibits choice but no concurrency) or to a well-behaved T-graph (which exhibits concurrency but no choice). We shall give a characterization of well-behaved systems which can be reduced to S-graphs (T-graphs, respectively) without using reductions. These characterization results give new insight to the general structure of well-behaved systems. The intersection of these two classes is not empty; well-formed nets without global aspects of concurrency or choice can be reduced to a net with only two elements.

The paper is organized as follows: In section 2 we introduce the used model for well-behaved systems and the reduction rules. We define live and safe free choice systems without frozen tokens and well-formed nets as their structural counterparts. The reduction rules and some of their properties are presented as well. For proving the main results in section 4 we need some general properties of the considered class of systems which are given in section 3. Section 5 concludes the paper with consequences and related work.

![Figure 1](image-url)