MATHEMATICAL PROBLEMS IN COMPUTATION THEORY
BANACH CENTER PUBLICATIONS, VOLUME 21
PWN – POLISH SCIENTIFIC PUBLISHERS
WARSAW 1988

FAIRNESS AND CONTROL

HANS-DIETER BURKHARD

Sektion Mathematik, Humboldt-Universität, Berlin, G.D.R.

The paper is based on the consideration of controls by restrictions of behaviour as introduced in [3]. Different notions of fairness and fair controls are derived from the fairness notions given in [6]: fairness is considered with or without reference to the behaviour of the original uncontrolled system, in connection with deadlock avoidance (infinite fair controls) and by omitting termination caused by control (nonblocking). In general, not all fair execution sequences can be obtained by a single control. In those cases, each control can be replaced by a less restrictive control. We give necessary and sufficient conditions for this fact. Different possibilities of realizing controls by finite state mechanisms or by predefined delay functions are investigated. The control automata controlling arbitrary systems in a fair way are characterized. It turns out that the approach by control automata is preferable to the use of delay functions.

1. Fairness and fair controls

A useful framework to study controls for concurrent systems was introduced in [3] using abstract languages. We will use these concepts in the investigation of fair controls.

Controls can be viewed under two aspects: Application of certain control rules (queues etc.), and controls which are defined as restrictions of behaviour in order to enforce properties like fairness, deadlock avoidance, termination etc. Both aspects should be considered on a common base since realizations of properties by control rules (e.g. fairness by queues) are an important subject.

Two observations are essential to come to that general calculus:

(1) Control is considered as a restriction of the behaviour of the system to be controlled.

(2) Each restriction of the behaviour of the uncontrolled system determines a control since all decisions of the control are well defined.

This correspondence between control and behaviour is employed for our purposes: A system is represented by its behaviour, controlled systems (and controls) are represented by restrictions of this behaviour. Then it depends on the description of the behaviour which control problems can be examined. As shown in [3] abstract languages are a convenient tool for many such problems. Special systems are examined by the corresponding families of languages, various types of controls are specified by the notion of control principles (see Definition (2) below).

The following notation is used:

N denotes the nonnegative integers.

T^* (T^ω) is the set of all finite (infinite) sequences over the alphabet T, e denotes the empty word.

A sequence $u \in T^*$ is a *prefix* of $v \in T^* \cup T^\omega$ ($u \sqsubseteq v$) if there exists a sequence v' with $v = uv'$.

The length of a sequence $u \in T^*$ is denoted by $|u|$.

The closure of a language $L \subseteq T^*$ with respect to initial segmentation (prefixes) is denoted by

$$\bar{L} := \{u / \exists v \in L\colon\ u \sqsubseteq v\}.$$

The adherence of a language L is defined by

$$\mathrm{Adh}(L) := \{w / w \in T^\omega \wedge \tilde{w} \subseteq L\},$$

where $\tilde{w} := \{u / u \in T^* \wedge u \sqsubseteq w\}$.

The power set of a set A is denoted by $P(A)$.

$\exists^\infty$ denotes *infinitely many*, $\forall^\infty$ denotes *almost all.*

By $\pi_v \in (N \cup \{\omega\})^T$ we denote the Parikhvector of a sequence $v \in T^* \cup T^\omega$, i.e., $\pi_v(t)$ denotes the number of occurrences of t in v.

By $\boldsymbol{a}$ we denote the vector $(a, \ldots, a)$ (of given dimension; $a \in N \cup \{\omega\}$).

Operations and relations for vectors are understood componentwise.

While many basic properties are valid for arbitrary systems, we are forced to consider special systems e.g. in decidability questions. In such cases we refer to finite transition systems and to Petri nets.

PREG denotes the family of prefix-closed regular languages (which can be understood as the behaviour of finite transition systems), and FNL denotes the family of the firing languages of finite Petri nets (the firing language of a Petri net is given by the set of all transition sequences which are firable starting in the initial marking).

Since the systems are represented by their languages, the results for languages from PREG (FNL) are understood as results for finite transition systems (Petri nets).

We consider (controlled or uncontrolled) systems by means of their behaviour, given by languages L over a finite fixed alphabet T with at least two elements. We suppose these languages to be nonempty and closed with respect to prefixes since control may influence the behaviour at any time. A control of a system is regarded as a restriction of its possibilities, thus the language L' of a controlled system is a subset of the language L of the original (uncontrolled) system.

(1) DEFINITION. (1) $\text{CONT} := \{L / L \subseteq T^* \wedge \emptyset \neq L = \bar{L}\}$ is the family of all control languages over T.

(2) $\text{cont}(L) := P(L) \cap \text{CONT}$ is the family of all control languages for $L \in \text{CONT}$.

Since the behaviour of a control (the decisions to be made with respect to the system L) is defined by a language $L' \in \text{cont}(L)$, the family $\text{cont}(L)$ describes all possible controls of the system L. Having a special way to perform controls (like scheduling disciplines) we obtain a special subset of $\text{cont}(L)$. Having also in mind special conditions to be satisfied (like fairness) we will study subsets of $\text{cont}(L)$:

(2) DEFINITION. (1) A *control principle* is a mapping $c: \text{CONT} \to P(\text{CONT})$ with $c(L) \subseteq \text{cont}(L)$ for all $L \in \text{CONT}$.

(2) The control principles imp, fair, rfair, just, rjust, dfr, nbl, pfin are defined as follows (we suppose $L' \in \text{cont}(L)$ in all cases):

$$L' \in \text{imp}(L) \quad \text{iff} \quad \forall w \in \text{Adh}(L'): \pi_w = \omega;$$

$$L' \in \text{fair}(L) \quad \text{iff} \quad \forall w \in \text{Adh}(L') \forall t \in T: (\exists^\infty u \sqsubseteq w: ut \in L') \to \pi_w(t) = \omega;$$

$$L' \in \text{rfair}(L) \quad \text{iff} \quad \forall w \in \text{Adh}(L') \forall t \in T: (\exists^\infty u \sqsubseteq w: ut \in L) \to \pi_w(t) \in \omega;$$

$$L' \in \text{just}(L) \quad \text{iff} \quad \forall w \in \text{Adh}(L') \forall t \in T: (\forall^\infty u \sqsubseteq w: ut \in L') \to \pi_w(t) = \omega;$$

$$L' \in \text{rjust}(L) \quad \text{iff} \quad \forall w \in \text{Adh}(L') \forall t \in T: (\forall^\infty u \sqsubseteq w: ut \in L) \to \pi_w(t) = \omega;$$

$$L' \in \text{dfr}(L) \quad \text{iff} \quad \forall u \in L' \exists t \in T: ut \in L';$$

$$L' \in \text{nbl}(L) \quad \text{iff} \quad \forall u \in L': (\exists t \in T: ut \in L) \to (\exists t' \in T: ut' \in L');$$

$$L' \in \text{pfin}(L) \quad \text{iff} \quad L' \text{ is finite.}$$

(3) The *conjunction* $c\&c'$ of two control principles c, c' is defined by

$$c\&c'(L) := c(L) \cap c'(L).$$

(4) A control principle c is *covered* by a control principle c' ($c \leqslant c'$) iff $c(L) \subseteq c'(L)$ for all $L \in \text{CONT}$.

(5) A control principle c is called *unitary* iff $\bigcup c(L) \in c(L)$ for all $L \in \text{CONT}$ with $c(L) \neq \emptyset$. (Iff a maximum element exists among the nonempty sets $c(L)$.)

By imp (*impartiality*), fair (*fairness*) and just (*justice*) we have specified the controlled systems satisfying the fairness notions of [6]. Thus, no reference is made to the original uncontrolled system, and we have e.g. $\{a\}^* \in \mathrm{fair}(\{a, b\}^*)$, where the action b of the uncontrolled system is star ved. This can be corrected by using the control principles rfair (*relatively fair* with respect to the uncontrolled system) and rjust (*relatively just*). As usual in literature we shall use *fairness* also as a general notion for impartiality (relative), fairness and (relative) justice. Further definitions of fairness (like in [2], for example) could be studied in our calculus as well.

By the control principle dfr we can consider deadlock-free controls, while the control principle nbl gives us controls, where the work of a system cannot be blocked by a control. These control principles are considered here in connection with fairness control principles using the conjunction: We can consider e.g. fair&dfr describing controls resulting in deadlock-free (infinite) fair behaviour (as studied for Petri nets in [4]) and fair&nbl describing the nonblocking fair controls.

The consideration of nonblocking fair controls is important since the definitions of fairness give only restrictions concerning the infinite behaviour such that fairness could always be obtained by restrictions to finite behaviour (finite languages). The controls resulting in finite languages are given by $\mathrm{pfin}(L)$.

Program termination under fairness assumptions (cf. [6], [1]) can be formulated by

$$c\&\mathrm{nbl}(L) \subseteq \mathrm{pfin}(L),$$

where c denotes a fairness control principle and L is the set of all computations of the examined program.

If c is covered by c', then each c-control is a c'-control, too. The relations between fairness control principles are given by Proposition (3) below.

If a maximum element exists in a family $c(L)$, then it represents the least restrictive c-control for L. Since all other controls can be understood as partial controls with respect to that maximum element, the maximum element is in some sense the canonical c-control for L. If no maximum element exists, then it makes no sense to speak about "the" c-control for L.

(3) PROPOSITION. (1) $\mathrm{pfin} \leqslant \mathrm{imp} \leqslant \mathrm{rfair} \leqslant \mathrm{fair} \leqslant \mathrm{just}$, $\mathrm{rfair} \leqslant \mathrm{rjust} \leqslant \mathrm{just}$, fair *and* rjust *are incomparable.*

(2) *The same relations as in* (1) *are valid if we substitute each c in* (1) *by* c&nbl *or each by* c&dfr.

(3) $c\&\mathrm{dfr} \leqslant c\&\mathrm{nbl} \leqslant c$ *for arbitrary control principles c.*

The relations above are immediate consequences of the definitions, to see the incomparability of fair and rjust we may consider

$$L := \{a, b\}^*, \quad \{a\}^* \in \mathrm{fair}(L) \setminus \mathrm{rjust}(L), \quad \overleftarrow{\{aa\}^* \{b\}^*} \in \mathrm{rjust}(L) \setminus \mathrm{fair}(L).$$

Furthermore, the following proposition can be proved:

(4) PROPOSITION. *Suppose* $c \in \{\mathrm{pfin}, \mathrm{imp}, \mathrm{fair}, \mathrm{rfair}, \mathrm{just}, \mathrm{rjust}\}$, $L \in \mathrm{CONT}$.

(1) $L' \in c(L) \to \mathrm{cont}(L') \subseteq c(L)$.

(2) *In general,* $L' \in c\&\mathrm{dfr}(L)$ *does not imply* $\mathrm{cont}(L') \subseteq c\&\mathrm{dfr}(L)$.
In general, $L' \in c\&\mathrm{nbl}(L)$ *does not imply* $\mathrm{cont}(L') \subseteq c\&\mathrm{nbl}(L)$.
But we have

$$L' \in c(L) \to \mathrm{dfr}(L') \subseteq c\&\mathrm{dfr}(L),$$

$$L' \in c\&\mathrm{nbl}(L) \to \mathrm{nbl}(L') \subseteq c\&\mathrm{nbl}(L).$$

Assertion (1) of the proposition above shows that fairness properties of a system cannot be destroyed by further controls (while deadlock freedom and nonblocking are not, in general, preserved under further controls). Assertion (2) is related to stepwise refinements of controls (in order to satisfy different properties). It turns out that realization of fairness can be performed in a first step while other properties like deadlock avoidance should be realized later on (cf. [3]).

The language $\bigcup c(L)$ for a given $L \in \mathrm{CONT}$ contains exactly those sequences $u \in L$ which can be performed under at least one c-control for L. Thus the set $L \setminus \bigcup c(L)$ contains those sequences from L which must be avoided by each c-control for L. It turns out that no general restrictions arise from fairness assumptions, but they may arise from deadlock avoidance and by nonblocking, as the following proposition shows:

(5) PROPOSITION. (1) $\bigcup c(L) = L$ *for* $c \in \{\mathrm{pfin}, \mathrm{imp}, \mathrm{fair}, \mathrm{rfair}, \mathrm{just}, \mathrm{rjust}, \mathrm{fair}\&\mathrm{nbl}, \mathrm{rfair}\&\mathrm{nbl}, \mathrm{just}\&\mathrm{nbl}, \mathrm{rjust}\&\mathrm{nbl}\}$ *and each* $L \in \mathrm{CONT}$.

(2) $\bigcup c(L) = \bigcup \mathrm{dfr}(L)$ *for* $c \in \{\mathrm{fair}\&\mathrm{dfr}, \mathrm{just}\&\mathrm{dfr}\}$, $L \in \mathrm{CONT}$.

(3) *There exist languages* $L \in \mathrm{CONT}$ *for*

$$c \in \{\mathrm{pfin}\&\mathrm{nbl}, \mathrm{imp}\&\mathrm{nbl}, \mathrm{imp}\&\mathrm{dfr}, \mathrm{rfair}\&\mathrm{nbl}, \mathrm{rjust}\&\mathrm{nbl}\}$$

with $\bigcup c(L) = \emptyset$ *(as a consequence of* $c(L) = \emptyset$*), while* $\mathrm{nbl}(L) \neq \emptyset$ *and* $\mathrm{dfr}(L) \neq \emptyset$.

Proof. (1) $\bigcup c(L) \subseteq L$ is trivial. To show $L \subseteq \bigcup c(L)$ we have $\tilde{u} \in c(L)$ and hence $u \in \bigcup c(L)$ for each $u \in L$ in the cases $c = \mathrm{pfin}$, imp, fair, rfair, just, rjust.

Concerning $c = \mathrm{fair}\&\mathrm{nbl}$, $\mathrm{rfair}\&\mathrm{nbl}$, $\mathrm{just}\&\mathrm{nbl}$, $\mathrm{rjust}\&\mathrm{nbl}$ we consider controls by fifo-queues which can be organized as follows:

Actions which are possible enter at the end of the queue if they have not actually been in the queue. The first possible action from the queue is the next action to be performed, it is deleted from the queue at that time.

The queues can be regarded as sequences from T^*. If we start with some $u \in L$ as initial queue, then the controlled system performs u first.

Obviously, each such queue regime works nonblocking and relatively fair. Hence it also works fair and (relatively) just by Proposition (3).

Again we have $u \in \bigcup c(L)$ for each $u \in L$ and therefore $L \subseteq \bigcup c(L)$.

We remark that finite control automata from Section 2 below are able to realize such queue regimes if the reorganization steps are performed in some regular way (this concerns the way of ordering if several new actions enter at the end of a queue in one step, cf. Theorem (13) below).

(2) $\bigcup c(L) \subseteq \bigcup \mathrm{dfr}(L)$ is trival. On the other hand, for each $u \in \bigcup \mathrm{dfr}(L)$, there exists some $w \in \mathrm{Adh}(L)$ with $u \sqsubseteq w$. Since $\tilde{w} \in c(L)$ for each $w \in \mathrm{Adh}(L)$, we have $u \in \bigcup c(L)$ and hence $\bigcup \mathrm{dfr}(L) \subseteq \bigcup c(L)$.

(3) As examples we may consider $T = \{a, b\}$, $L = \{a\}^*$ for $c =$ pfin&nbl, imp&nbl, imp&dfr, and $L = \overleftarrow{\{a\}^* \{b\}}$ for $c =$ rfair&dfr, rjust&dfr.

Note that $\mathrm{nbl}(L) \neq \emptyset$ and $\mathrm{dfr}(L) \neq \emptyset$ in both cases. ∎

(6) THEOREM (EXISTENCE OF CONTROLS). (1) *$c(L) \neq \emptyset$ for*

$c \in \{$pfin, imp, fair, rfair, just, rjust, fair&nbl, rfair&nbl, just&nbl, rjust&nbl$\}$ *and each $L \in$ CONT.*

(2) *The problems "$c(L) \neq \emptyset$?" are decidable for* $c \in \{$pfin&nbl, imp&nbl, imp&dfr, fair&dfr, just&dfr$\}$ *and* $L \in \mathrm{PREG} \cup \mathrm{FNL}$.

Proof. Assertion (1) holds by Proposition (5.1). Assertion (2) was proved in [3]:

We have $\text{pfin\&nbl}(L) \neq \emptyset$ iff there exists $u \in L$ with $ut \notin L$ for all $t \in T$ (i.e., $\tilde{u} \in \mathrm{nbl}(L)$; decidable for $L \in \mathrm{PREG} \cup \mathrm{FNL}$).

We have $\text{imp\&dfr}(L) \neq \emptyset$ iff there exist $u, v \in T^*$ with $\pi_v \geq 1$ and $uv^\omega \in \mathrm{Adh}(L)$ in the case $L \in \mathrm{PREG} \cup \mathrm{FNL}$.

We have $\text{imp\&nbl}(L) \neq \emptyset$ iff $\text{pfin\&nbl}(L) \neq \emptyset$ or $\text{imp\&dfr}(L) \neq \emptyset$.

We have $\text{fair\&dfr}(L) \neq \emptyset$ ($\text{just\&dfr}(L) \neq \emptyset$) iff $\mathrm{dfr}(L) \neq \emptyset$ by Proposition (5.2); and for $L \in \mathrm{PREG} \cup \mathrm{FNL}$ we have $\mathrm{dfr}(L) \neq \emptyset$ iff there exist some $u, v \in T^*$ with $uv^\omega \in \mathrm{Adh}(L)$.

Thus the problems are reducable to the problems of the existence of the corresponding sequences u, v. These problems are decidable for $L \in \mathrm{PREG} \cup \mathrm{FNL}$, and if such sequences exist, then they can be constructed effectively (cf. [3]). ∎

We remark that the problems "$c\&\mathrm{dfr}(L) \neq \emptyset$?" for $c =$ rfair, rjust are decidable for $L \in$ PREG, too, while they are open for $L \in$ FNL. In particular $\mathrm{dfr}(L) \neq \emptyset$ does not imply $c\&\mathrm{dfr}(L) \neq \emptyset$ (e.g. for $L = \overleftarrow{\{a\}^* \{b\}}$).

Fairness control principles are not unitary, hence in general, it is not possible to obtain all fair system runs by a special fair control. Moreover, in general, each fair control of a system can be "improved" by making it less restrictive. We have

(7) THEOREM. *The control principles* $c \in \{$pfin, imp, fair, rfair, just, rjust, pfin&nbl, imp&nbl, fair&nbl, rfair&nbl, just&nbl, rjust&nbl, imp&dfr, fair&dfr, rfair&dfr, just&dfr, rjust&dfr$\}$ *are not unitary, and for each* $L \in \mathrm{CONT}$

$$\forall L' \in c(L) \exists L'' \in c(L)\colon\ L' \subsetneqq L'' \quad \textit{iff} \quad \bigcup c(L) \notin c(L).$$

To prove the theorem we show the following lemma:

(8) LEMMA. *Let c be a control principle as in Theorem (7). Then the families* $c(L)$ *for* $L \in \mathrm{CONT}$ *are closed with respect to finite unions but, in general they are not closed with respect to arbitrary unions.*

Proof. The closure with respect to finite unions is an immediate consequence of the definitions and of the fact that $\mathrm{Adh}(L \cup L') = \mathrm{Adh}(L) \cup \mathrm{Adh}(L')$ for prefix-closed languages L, L'.

To show that in general the languages are not closed with respect to arbitrary unions, we consider $L = \{a, b\}^*$, $T = \{a, b\}$, where, except for c = pfin&nbl, $\bigcup c(L) = L$ (since $\tilde{u} \in c(L)$ or $\overleftarrow{u(ab)^\omega} \in c(L)$ for each $u \in L$). But, since $L \notin c(L)$, $c(L)$ is not closed under arbitrary unions.

For c = pfin&nbl we may consider $L = \overleftarrow{\{a\}^* \{b\}}$ with $\bigcup c(L) = L \notin c(L)$, which completes the proof of the lemma.

Now, the control principles c of Theorem (7) are not unitary by the examples given in the proof of Lemma (8). Furthermore, if $\bigcup c(L) \notin c(L)$, then $L' \neq \bigcup c(L)$ for each $L' \in c(L)$, and hence there exists $L_1 \in c(L)$ with $L_1 \not\subseteq L'$. Thus we have $L' \subsetneqq L'' := L' \cup L_1$ and $L'' \in c(L)$ since $c(L)$ is closed under finite unions. On the other hand, if $\bigcup c(L) \in c(L)$, then $L'' := \bigcup c(L)$ is the maximum element in $c(L)$. ∎

From Theorem (7) and Proposition (5) we obtain

(9) COROLLARY. *Suppose* $L \in \mathrm{CONT}$. *The condition*

$$\forall L' \in c(L) \exists L'' \in c(L)\colon\ L' \subsetneqq L''$$

holds for $c \in \{$pfin, imp, fair, rfair, just, rjust, fair&nbl, rfair&nbl, just&nbl, rjust&nbl$\}$ *iff* $L \notin c(L)$, *and for* $c \in \{$fair&dfr, just&dfr$\}$ *iff* $\bigcup \mathrm{dfr}(L) \notin c(L)$.

2. Fair controls by automata

If a system L does not work in a (relatively) fair or just way, then it has infinitely many different controls which realize the corresponding fairness properties (Corollary (9)). Of special interest are the controls which are nonblocking and, among them, those which can be realized by finite state mechanisms. We consider finite nondeterministic *control automata*

$$A = (P(T), T, Z, h, z_0),$$

where $P(T)$, T, Z are finite nonempty sets of inputs, outputs and states, respectively, $z_0 \in Z$ is the initial state, and h: $Z \times P(T) \to P(T \times Z)$ is the output/next-state function with

$$h(z, \emptyset) = \emptyset \quad \text{and} \quad \emptyset \neq \{t/\exists z'\colon (t, z') \in h(z, U)\} \subseteq U$$

for all $z \in Z$, $U \in P(T) \setminus \{\emptyset\}$. (This condition ensures that the control by control automata works nonblocking.)

Without loss of generality we assume the control automata to be initially connected.

A control automaton A and the system controlled by A form an interactive system: The automaton receives as input the set U of all actions from T which could be performed in the next step by the system and decides by its output $t \in U$ which action can be performed. We define in this sense:

(10) Definition. Let $A = (P(T), T, Z, h, z_0)$ be a control automaton. The *result of control* of $L \in$CONT by A is the language L/A with

$$e \in L/A, \quad t_1 \ldots t_n \in L/A \quad \text{iff}$$

$$\exists z_1, \ldots, z_n \in Z \,\forall i = 0, \ldots, n-1\colon (t_{i+1}, z_{i+1}) \in h(z_i, \{t/t_1 \ldots t_i t \in L\}).$$

By the definition of control automata we have:

(11) Corollary. *$L/A \in$nbl(L) for all $L \in$CONT and all control automata A.*

As mentioned in the proof of Proposition (5), the queue regimes introduced there can be realized by finite control automata. The queues may serve as states. Automata like these have the property of controlling arbitrary systems in the desired fair way. We now characterize the control automata having such properties.

(12) Definition. A control automaton A is a universal *c-control automaton* for the control principle c if $L/A \in c(L)$ for all $L \in$CONT.

As we shall see, universal control automata exist for $c \in$ {fair, rfair, just, rjust} (and therefore for c&nbl by Corollary (11)), but not for the other control principles considered in this paper. We use the following notation for control automata $A = (P(T), T, Z, h, z_0)$, $z, z' \in Z$, $t \in T$, $u = t_1 \ldots t_n \in T^*$, $\mathrm{u} = U_1 \ldots U_n \in (P(T))^*$:

$$z \xrightarrow{u,\mathrm{u}} z' \quad \text{iff} \quad \exists z_1, \ldots, z_{n+1} \in Z\colon z_1 = z \wedge z_{n+1} = z' \wedge \forall i = 1, \ldots, n\colon (t_i, z_{i+1}) \in h(z_i, U_i),$$

t in u iff $\exists i \in \{1, \ldots, n\}\colon t = t_i$,

t in u iff $\exists i \in \{1, \ldots, n\}\colon t \in U_i$,

t allin u iff $\forall i \in \{1, \ldots, n\}\colon z \in U_i$,

t enabled by $z \xrightarrow{u,\mathrm{u}} z'$ iff

$$z = z_1 \xrightarrow{U_1, t_1} z_2 \xrightarrow{U_2, t_2} \dots \xrightarrow{U_n, t_n} z_{n+1} = z' \wedge \exists i \in \{1, \dots, n\} \exists z'' \in Z:$$
$$(t, z'') \in h(h_i, U_i),$$

t allenabled by $z \xrightarrow{\mathfrak{u}, u} z'$ iff

$$z = z_1 \xrightarrow{U_1, t_1} z_2 \xrightarrow{U_2, t_2} \dots \xrightarrow{U_n, t_n} z_{n+1} = z' \wedge \forall i \in \{1, \dots, n\} \exists z'' \in Z:$$
$$(t, z'') \in h(z_i, U_i).$$

(13) THEOREM. *Let* $A = (P(T), T, Z, h, z_0)$ *be a control automaton.* A *is a universal*

(a) rfair-*control automaton,*
(b) rjust-*control automaton,*
(c) fair-*control automaton,*
(d) just-*control automaton,*

iff $\forall z \in Z\, \forall t \in T\, \forall \mathfrak{u} \in (P(T))^*\, \forall u \in T^*$: *if* $z \xrightarrow{\mathfrak{u}, u} z$ *and*

(a) t *in* $\mathfrak{u}$,
(b) t *allin* $\mathfrak{u}$,
(c) t *enabled by* $z \xrightarrow{\mathfrak{u}, u} z$,
(d) t *allenabled by* $z \xrightarrow{\mathfrak{u}, u} z$,

then t *in* u.

Proof. We first show that the given conditions are necessary: Let A be a control automaton such that there exist t, z, $\mathfrak{u} = U_1 \dots U_n$, u with $z \xrightarrow{\mathfrak{u}, u} z$ and not t in u. Since A is initially connected, there exist $\mathfrak{u}' = U'_1 \dots U'_m \in P(T)^*$, $u' \in T^*$ such that

$$z_0 \xrightarrow{\mathfrak{u}', u'} z \xrightarrow{\mathfrak{u}, u} z \xrightarrow{\mathfrak{u}, u} z \dots$$

Then we consider the language $L := \overleftarrow{U'_1 \dots U'_m (U_1 \dots U_n)^*}$ with $w := u' u^\omega \in \mathrm{Adh}(L/A)$ and $\pi_t(w) < \omega$. If we would have t in $\mathfrak{u}$, then we would obtain $\exists^\infty v \sqsubseteq w$: $vt \in L$ and hence $L/A \notin \mathrm{rfair}(L)$.

Analogously,

t allin $\mathfrak{u}$ would imply that $\forall^\infty v \sqsubseteq w$: $vt \in L$ and hence $L/A \notin \mathrm{rjust}(L)$;

t enabled by $z \xrightarrow{\mathfrak{u}, u} z$ would imply that $\exists^\infty v \sqsubseteq w$: $vt \in L/A$ and hence $L/A \notin \mathrm{fair}(L)$;

t allenabled by $z \xrightarrow{\mathfrak{u}, u} z$ would imply that $\forall^\infty v \sqsubseteq w$: $vt \in L/A$ and hence $L/A \notin \mathrm{just}(L)$.

Now we show that the conditions are sufficient: Suppose $w = t_0 t_1 \dots \in \mathrm{Adh}(L/A)$ and let $\mathfrak{z} = z_0 z_1 \dots \in Z^\omega$, $\mathfrak{w} = U_0 U_1 \dots \in (P(T))^\omega$ be an infinite state sequence and an infinite input sequence, respectively, for A such that

$$z_0 \xrightarrow{U_0, t_0} z_1 \xrightarrow{U_1 t_1} z_2 \xrightarrow{U_2, t_2} \dots$$

Then there exists some $z \in Z$ which occurs infinitely often in $\mathfrak{z}$, i.e., we have

$\mathfrak{w} = \mathfrak{u}_0 \mathfrak{u}_1 \ldots$, $w = u_0 u_1 \ldots$ with

$$z_0 \xrightarrow{\mathfrak{u}_0, u_0} z \xrightarrow{\mathfrak{u}_1, u_1} z \xrightarrow{\mathfrak{u}_2, u_2} z \ldots$$

Now, if $\exists^\infty v \sqsubseteq w$: $vt \in L$, then there are infinitely many $\mathfrak{u}_i$ with t in $\mathfrak{u}_i$, and the condition corresponding to assertion (a) implies the existence of infinitely many u_i with t in u_i such that $\pi_w(t) = \omega$. Hence $L/A \in \text{rfair}(L)$.

The remaining cases are proved analogously. ∎

3. Fairness by delay function and realization of controls

The fact that fairness control principles are not unitary (Theorem (7)) can be seen as a consequence of the freedom to have arbitrarily long (but finite) delays for the actions which must be performed according to fairness conditions. Following some ideas in [5] we can consider delay functions d: $T^* \times T \to N$ and define fairness with respect to a given delay function:

(14) DEFINITION. Let d: $T^* \times T \to N$ be a delay function and let $L \in \text{CONT}$, $L' \in \text{cont}(L)$.

(1) $L' \in d\text{-imp}(L)$ iff $\forall t \in T \forall uv \in L'$: $|v| > d(u, t) \to \pi_v(t) > 0$.

(2) $L' \in d\text{-fair}(L)$ iff $\forall t \in T \forall uv \in L'$: $\text{card}(\{v'/v' \sqsubsetneqq v \wedge uv't \in L'\}) > d(u, t) \to \pi_v(t) > 0$.

(3) $L' \in d\text{-rfair}(L)$ iff $\forall t \in T \forall uv \in L'$: $\text{card}(\{v'/v' \sqsubsetneqq v \wedge uv't \in L\}) > d(u, t) \to \pi_v(t) > 0$.

(4) $L' \in d\text{-just}(L)$ iff $\forall t \in T \forall uv \in L'$: $(|v| > d(u, t) \wedge \forall v' \sqsubsetneqq v$: $uv't \in L') \to \pi_v(t) > 0$.

(5) $L' \in d\text{-rjust}(L)$ iff $\forall t \in T \forall uv \in L'$: $(|v| > d(u, t) \wedge \forall v' \sqsubsetneqq v$: $uv't \in L) \to \pi_v(t) > 0$.

(15) PROPOSITION. (1) d-imp, d-fair, d-rfair, d-just, d-rjust *are unitary.*

(2) *For* $L \in \text{CONT}$ *and for* $c \in \{\text{imp, fair, rfair, just, rjust}\}$: we have $\emptyset \neq d\text{-}c(L) \subseteq c(L)$.

(3) d-imp $\leqslant$ d-rfair $\leqslant$ d-fair $\leqslant$ d-just, d-rfair $\leqslant$ d-rjust $\leqslant$ d-just. *In general,* d-fair *and* d-rjust *are incomparable.*

Proof. (1) The families $c(L)$, $c = d$-imp, ..., d-rjust are obviously closed under arbitrary unions, and hence $\bigcup c(L) \in c(L)$.

(2) The families $d\text{-}c(L)$ are not empty since we always have $\{e\} \in d\text{-}c(L)$. To show the inclusions $d\text{-}c(L) \subseteq c(L)$ we consider the case $c = \text{rfair}$, the remaining proofs are similar:

If $L' \notin \text{rfair}(L)$, then there exist $t \in T$ and $w \in \text{Adh}(L')$ with $ut \in L$ for infinitely many $u \sqsubseteq w$ and with $\pi_w(t) < \omega$. Hence there exists a sequence $uv \in L'$ with $\text{card}(\{v'/v' \sqsubsetneqq v \wedge uv't \in L\}) > d(u, t)$ and $\pi_v(t) = 0$. Thus $L' \notin d\text{-rfair}(L)$.

(3) This assertion follows from the definitions and the example given for Proposition (3).

The next theorem shows that impartiality and (relative) justice are completely expressible by means of delay functions. This is not the case for (relative) fairness.

(16) THEOREM. *For arbitrary* $L \in \mathrm{CONT}$ *we have*

(1) $\bigcup_d d\text{-}c(L) = c(L)$ *for* $c = $ imp, just, rjust.

(2) $\bigcup_d d\text{-}c(L) \subseteq c(L)$ *for* $c = $ fair, rfair, *and there exist languages* $L \in \mathrm{CONT}$ *with* $\bigcup_d d\text{-}c(L) \neq c(L)$.

Proof. The left-to-right inclusions follow by Proposition (15.2). The right-to-left inclusions in (1) are proved using Koenig's Lemma (as in [5] for impartiality).

If $L' \notin \bigcup_d d\text{-imp}(L)$ for some $L' \in \mathrm{cont}(L)$, then we have

$$\neg(\exists d(d\colon T^* \times T \to N)\,\forall t \in T\,\forall uv \in L'\colon |v| > d(u, t) \to \pi_v(t) > 0),$$

i.e.,

$$\exists t \in T\,\exists u \in L'\,\forall n \in N\,\exists v \in T^*\colon uv \in L' \wedge |v| > n \wedge \pi_v(t) = 0.$$

Forming a tree out of the sequences v, we can apply Koenig's Lemma and obtain an infinite path w such that $uw \in \mathrm{Adh}(L')$ and $\pi_{uw}(t) < \omega$. Hence we have $L' \notin \mathrm{imp}(L)$.

If $L' \notin \bigcup_d d\text{-rjust}(L)$ (and analogously for $d\text{-just}(L)$) for some $L' \in \mathrm{cont}(L)$, then we have

$$\neg(\exists d(d\colon T^* \times T \to N)\,\forall t \in T\,\forall uv \in L'\colon (|v| > d(u, t) \wedge \forall v' \subsetneqq v\colon uv't \in L) \to \pi_v(t) > 0),$$

i.e.,

$$\exists t \in T\,\exists u \in L'\,\forall n \in N\,\exists v \in T^*\colon uv \in L' \wedge |v| > n \wedge \forall v' \subsetneqq v\colon uv't \in L \wedge \pi_v(t) = 0.$$

By Koenig's Lemma we obtain $uw \in \mathrm{Adh}(L')$ with $uvt \in L$ for all $v \subsetneqq w$ and $\pi_w(t) = 0$ so that $L' \notin \mathrm{rjust}(L)$. To show that the inclusions in assertion (2) may be proper we consider

$$L := \overleftarrow{\{(ac)^n c^{m+1} b / 0 \leqslant m \leqslant n\}} \in \mathrm{fair}(L).$$

L is not contained in $\bigcup_d d\text{-fair}(L)$. Otherwise, if $L \in d\text{-fair}(L)$ for some delay function d, then we consider

$$v = (ac)^{k+1} c^{k+2} \quad \text{for} \quad k = d(e, b).$$

Then we have

$$\mathrm{card}(\{v'/v' \subsetneqq v \wedge v'b \in L\}) = k + 1 > d(e, b),$$

but $\pi_v(b) = 0$ so that $L \notin d\text{-fair}(L)$ – contradiction. We also have $L \in \text{rfair}(L) \setminus \bigcup_d d\text{-rfair}(L)$. ∎

It is an open problem whether fairness and relative fairness may be expressed by other appropriate delay functions or at least without reference to the infinite behaviour (L. Czaja's question in [5]).

The following proposition shows that the use of fairness with respect to delay functions may give rise to problems, especially the maximum elements $\bigcup c(L)$ may lead to blockings by control.

(17) PROPOSITION. *Let d: $T^* \times T \to N$ be a delay function and let L_c $(:= \bigcup c(L))$ denote the maximum element in $c(L)$ for $c = d$-imp, d-fair, d-rfair, d-just, d-rjust (the maximum element exists by Proposition (15.1)).*

(1) *There exists $L \in \text{PREG}$ with $L_{d\text{-imp}} \notin \text{nbl}(L)$.*

(2) *If $\text{card}(T) \geqslant 3$ and if d is a constant function, then there exists $L \in \text{PREG}$ with $L_c \notin \text{nbl}(L)$ for $c = d$-fair, ..., d-rjust.*

To prove (1) it suffices to consider $T = \{a, b\}$, $L = \{a\}^*$.

To prove (2) we fix some $k \in N$ and suppose $d(u, t) = k$ for all $u \in T^*$ with $|u| \leqslant k$. Then we consider $L = \{a, b, c\}^*$ with $L_{d\text{-fair}} \supseteq \{u/u \in L \wedge |u| \leqslant k\}$, where the further work is blocked for $u = a^k$ under d-fairness. The same arguments apply to relative fairness and (relative) justice. ∎

The maximum elements L_c, $c = d\text{-imp}, \ldots, d\text{-rjust}$, represent those controls which allow all possibilities according to the underlying delay function d. As a consequence of Proposition (17), these controls cannot in general be realized by control automata, since control automata work nonblocking (Corollary (11)).

The universal control automata of Theorem (13) realize controls corresponding to constant delay functions:

(18) PROPOSITION. *Suppose $c \in \{\text{fair, rfair, just, rjust}\}$ and let A be a universal c-control automaton having n states. Furthermore let d be the constant delay function with $d \equiv n$. Then we have*

$$L/A \in d\text{-}c(L) \quad \textit{for all } L \in \text{CONT}.$$

Proof. Let $A = (P(T), T, Z, h, z_0)$ be a control automaton having n states and suppose $t \in T$, $uv \in L/A$.

Then there are $\mathfrak{u}, \mathfrak{v} \in P(T)^*$, $z, z' \in Z$ such that

$$z_0 \xrightarrow{\mathfrak{u}, u} z \xrightarrow{\mathfrak{v}, v} z'.$$

Concerning fairness we consider the case

$$\text{card}(\{v'/v' \subsetneqq v \wedge uv't \in L/A\}) > d(u, t) = n.$$

Then we can decompose $v = v_1 \ldots v_{n+2}$,

$$\mathfrak{v} = \mathfrak{v}_1 \ldots \mathfrak{v}_{n+2}$$

and find $z_1, \ldots, z_{n+1} \in Z$ such that $uv_1 \ldots v_i t \in L/A$ for $i = 1, \ldots, n+1$ and

$$z_0 \xrightarrow{u,\mathfrak{u}} z \xrightarrow{\mathfrak{v}_1, v_1} z_1 \xrightarrow{\mathfrak{v}_2, v_2} \ldots z_{n+1} \xrightarrow{\mathfrak{v}_{n+2}, v_{n+2}} z'.$$

By $\mathrm{card}(Z) = n$, there must exist $i, j \in N$ with $1 \leqslant i < j \leqslant n+1$ and $z_i = z_j$, and hence t is enabled by

$$z_i \xrightarrow{\mathfrak{v}_{i+1} \ldots \mathfrak{v}_j, v_{i+1} \ldots v_j} z_i = z_j.$$

Now, if A is a universal fair-control automaton, then we have t in $v_{i+1} \ldots v_j$ by Theorem (13). Thus we have $\pi_v(t) > 0$ and $L/A \in d\text{-fair}(L)$.

The proof for relative fairness is similar.

Concerning justice we consider the case

$$|v| > d(u, t) = n \quad \text{and} \quad \forall v' \sqsubsetneq v\colon uv't \in L/A.$$

Then we find $v = t_1 \ldots t_{n+1}$,

$$\mathfrak{v} = U_1 \ldots U_{n+1}, \quad z_1, \ldots, z_{n+1} \in Z$$

with $ut \in L/A$, $ut_1 \ldots t_i t \in L/A$ for $i = 1, \ldots, n$,

$$z_0 \xrightarrow{u,\mathfrak{u}} z = z_1 \xrightarrow{U_1 t_1} \ldots z_{n+1} \xrightarrow{U_{n+1}, t_{n+1}} z'.$$

Again, there must exist $z_i = z_j$ with $1 \leqslant i < j \leqslant n+1$, and hence t is allenabled by

$$z_i \xrightarrow{U_i \ldots U_{j-1}, t_i \ldots t_{j-1}} z_i = z_j.$$

Now, if A is a universal just-control automaton, then we have t in $t_i \ldots t_{j-1}$ by Theorem (13), i.e., $\pi_v(t) > 0$ and hence $L/A \in d\text{-just}(L)$.

The proof for relative justice is similar. ∎

(19) Theorem (Realization of controls). (1) *Suppose* $c \in$ {fair, rfair, just, rjust, fair&nbl, rfair&nbl, just&nbl, rjust&nbl}.

Then a control automaton A can be constructed such that

$$L/A \in c(L) \quad \textit{for all } L \in \mathrm{CONT}.$$

(2) *Suppose* $c \in$ {pfin, pfin&nbl, imp&dfr, fair&dfr, just&dfr} *and* $L \in \mathrm{PREG} \cup \mathrm{FNL}$, $c(L) \neq \emptyset$.

Then a control automaton A can be constructed with $L/A \in c(L)$.

Proof. Assertion (1) is a consequence of Theorem (13).

The proof of assertion (2) refers to the proof of Theorem (6): The sequences u, v mentioned there can be constructed if they exist. The construction of a control automaton A with $L/A = \overleftarrow{uv^{\omega}} \in c(L)$ (or $= \tilde{u} \in c(L)$

in the corresponding cases) poses no problem. But note that these automata have to be constructed individually for each system, while the automata in assertion (1) can be chosen as universal control automata. ∎

We remark that Theorem (19) also holds for $c \in \{\text{rfair\&dfr, rjust\&dfr}\}$ in the case $L \in \text{PREG}$, but not in the case $L \in \text{FNL}$.

References

[1] K. R. Apt, A. Pnueli and J. Stavi, *Fair termination revisited – with delay*, Publ. du L.I.T.P., Univ. Paris VII (1982), 82–51.

[2] E. Best, *Fairness and conspiracies*, Inform. Process. Lett. 18 (1984), 215–220.

[3] H. D. Burkhard, *An investigation of controls for concurrent systems based on abstract control languages*, Theoret. Comput. Sci. 38 (1985), 193–222.

[4] H. Carstensen and R. Valk, *Infinite behaviour and fairness in Petri nets*, in: Proc. 4th European Workshop on Application and Theory of Petri nets, Toulouse 1983, 104–123.

[5] L. Czaja, *Are infinite behaviours of parallel schemata necessary*? In: A. Salwicki, Ed., *Logics of Programs and Their Application*, Lecture Notes in Comput. Sci. 148 (1983), 108–117.

[6] D. Lehmann, A. Pnueli and J. Stavi, *Impartiality, justice, fairness: The ethics of concurrent termination*, In: S. Even and O. Kariv, Ed., *Automata, Languages and Programming*, ibidem 115 (1981), 264–277.

Presented to the semester
Mathematical Problems in Computation Theory
September 16 – December 14, 1985