Fairness, Computable Fairness, and Randomness

Manfred Jaeger
Max-Planck-Institut für Informatik
Im Stadtwald, 66123 Saarbrücken, Germany
jaeger@mpi-sb.mpg.de

Abstract

Motivated by the observation that executions of a probabilistic system almost surely are fair, we interpret concepts of fairness for nondeterministic processes as partial descriptions of probabilistic behavior. We propose computable fairness as a very strong concept of fairness, attempting to capture all the qualitative properties of probabilistic behavior that we might reasonably expect to see in the behavior of a nondeterministic system. It is shown that computable fairness does describe probabilistic behavior by proving that runs of a probabilistic system almost surely are computable fair. We then turn to the question of how sharp an approximation of randomness is obtained by computable fairness by discussing completeness of computable fairness for certain classes of path properties.

1 Introduction

The concept of fairness was introduced to formulate certain conditions on the behavior of nondeterministic systems. These conditions amount to more or less strong assumptions on the actions of the environment, a scheduler, an adversary, or whatever unknown (or unspecified) component is responsible for the nondeterminism in a system. Depending on what kind of unspecified component is modeled by nondeterminism, a specific fairness assumption will be more or less reasonable to make. If, for instance, the nondeterministic transitions in a system model the moves of a malicious, intelligent adversary, whose aim it is to generate a run of the system with certain properties, then no fairness assumption will usually be admissible. If, on the other hand, nondeterminism represents the actions of a scheduler in a model for concurrent computation, then standard fairness notions (which, of course, originally were just proposed in this context) amount to legitimate assumptions on the policy of the scheduler. Finally, take the case where nondeterminism models the interaction of a system with a chaotic environment (e.g. the interaction of a cafeteria coffee machine with its customers). Here we often can view nondeterminism as a qualitative approximation to randomness, i.e. the nondeterministic system really represents a probabilistic system without any concrete probability values filled in. In this last case about any kind of fairness assumption is justified, because for a concept of x-fairness, say, and a probabilistic system obtained from a nondeterministic one by filling in probability values, we will obtain a theorem of the form

The set of all x-fair executions has probability 1. (1)

Pnueli and Zuck (1993), for instance, have shown such a theorem for their concept of α-fairness; Baier and Kwiatkowska (1998) for a whole class of fairness notions. Thus seeing that fairness assumptions are generally adequate for probabilistic systems, we can interpret fairness itself as a requirement that runs of a nondeterministic system have some of the properties that runs of a probabilistic system almost surely have, or, loosely speaking, that the adversary or scheduler in the nondeterministic system be not too malicious, and behave a little bit like a fair coin tosser. This point of view was already taken by Pnueli and Zuck, whose notions of extreme fairness (Pnueli 1983) and α-fairness (Pnueli & Zuck 1993) were proposed explicitly as an approximation to probabilistic behavior.

When we have shown a theorem (1) for two different notions of fairness, say x-fairness and y-fairness, then the theorem also holds for the new concept xy-fairness, obtained by requiring a run to be both x-fair and y-fair. Similarly for the conjunction of a countably infinite set of different fairness notions. This suggests the question: is there a canonical, strongest concept of fairness, one that subsumes most or all previously proposed notions of fairness, one that still allows us to derive for it theorem (1), and, finally, that still can be seen as a natural requirement for a nondeterministic system? In section 3 we are going to propose *computable fairness* as an answer to this question. Computable fairness is based on the concept of unpredictability, i.e. we are going to call a behavior of a nondeterministic system computable fair, if it is sufficiently unpredictable.

In some cases we also get as a weak converse of (1) a theorem that for some class $\mathscr{C}$ of properties of runs says

For all $C \in \mathscr{C}$: if C holds with probability 1 in a probabilistic system, then C holds for all x-fair runs. (2)

Instances of (2) are much scarcer than instances of (1), being essentially exhausted by the result of Pnueli and Zuck (1993) that (2) holds for the class of properties expressible in PLTL, and for α-fairness. Clearly, (2) cannot hold without the relativization to a suitable class $\mathscr{C}$, since otherwise we would just let C be the set of all runs except one specific run that is x-fair. Then C will have probability one, and yet the conclusion of (2) will not hold.

In section 4 we will show that (2) holds for computable fairness and $\mathscr{C}$ the class of properties definable by deterministic Büchi automata. This is not yet a very strong result, and we expect to extend it to more general classes $\mathscr{C}$ in future work.

2 Preliminaries

In this section we introduce the basic terminology we shall need. In most works related to the issues we are here dealing with, models for systems are introduced that combine nondeterminism with randomness (e.g. concurrent Markov chains (Vardi 1985), or similar models (Pnueli & Zuck 1993, Courcoubetis & Yannakakis 1995, Baier & Kwiatkowska 1998)). For the study of our question how concepts of fairness relate to probabilistic behavior, however, we only need the following two very basic models of nondeterministic, respectively probabilistic, transition systems.

Here and in the sequel we use $\mathscr{P}(S)$ to denote the powerset of S, and $\mathscr{F}\mathscr{P}(S)$ to denote the set of all finite subsets of S.

Definition 2.1 A *nondeterministic transition system* is a structure of the form

$$G = (S, s_0, t),$$

where S is a countable set of states, $s_0 \in S$ is the initial state, and $t: S \to \mathscr{P}(S)$ is a transition relation.

In the following definition we use ΔS to denote the set of probability distributions on the set S.

Definition 2.2 A *probabilistic transition system* (a.k.a. Markov chain) is a structure of the form

$$\Gamma = (S, s_0, \tau),$$

where S, s_0, are as in definition 2.1, and $\tau: S \to \Delta S$. Γ is called *bounded* if there exists an $\epsilon > 0$ with $\tau(s)(s') = 0$ or $\tau(s)(s') \geq \epsilon$ for all $s, s' \in S$.

In particular, we always have that in a bounded probabilistic transition system the set of possible successor states of some state is finite.

From a probabilistic transition system we can obtain a nondeterministic transition system by ignoring the probability values. Similarly, a nondeterministic system can be "refined" to a probabilistic one by adding probabilities for the transitions.

Definition 2.3 We say that a bounded probabilistic transition system (S, s_0, τ) *corresponds* to a nondeterministic transition system (S', s_0', t) if $S = S', s_0' = s_0$, and $t(s) = \{s' \mid \tau(s)(s') > 0\}$ for all $s \in S$.

If S is a set of states then S^ω denotes the set of infinite sequences, or paths, of states. For $\sigma \in S^\omega$ we denote by $\sigma[i]$ the ith element of σ, and by σ_i its prefix of length i.

A nondeterministic transition system G defines a subset $R(G) \subseteq S^\omega$, the set of possible runs of G.

A probabilistic transition system Γ induces a probability measure $P(\Gamma)$ on S^ω in the usual manner: for each $i \in \omega$ we denote by $\mathfrak{A}_i$ the (finite) σ-algebra on S^ω generated by all sets of the form $\boldsymbol{s}S^\omega$ with $\boldsymbol{s} \in S^i$. A probability measure $P_i(\Gamma)$ is defined on $\mathfrak{A}_i$ in the obvious manner. Taking $\mathfrak{A}_\omega$ the σ-algebra generated by the $\mathfrak{A}_i$, we obtain by standard results in probability theory a unique measure $P(\Gamma)$ on $\mathfrak{A}_\omega$ that extends all the $P_i(\Gamma)$.

3 Computable Fairness

Our concept of computable fairness is based on the observation that violations of standard concepts of fairness (e.g. extreme fairness (Pnueli 1983)) by the run of a system lead to a partial predictability of that run. As an example consider the nondeterministic transition system shown in figure 1. Extreme fairness (and about every other imaginable notion of fairness) would require that in a run of the system both the states `-1` and `1` occur infinitely often. Now consider the run

$$\sigma = \texttt{1 -1 -1 1 -1 1 1 1} \ldots = \texttt{1 -1 -1 1 -1 1}^{\omega}$$

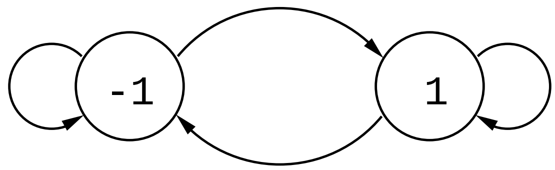

Figure 1: A simple transition system

that is not fair in this sense, because the state `-1` never occurs after step 5. An observer of the system's behavior who from the 5th transition onwards predicts the next state to be `1` then will always make correct predictions, and, moreover, his predictions will infinitely often be nontrivial in that the predicted state `1` is only one of several possible successor states.

Next consider the run

$$\sigma' = \texttt{-1}\ \texttt{-1}\ \texttt{1}\ \texttt{1}\ \texttt{-1}\ \texttt{-1}\ \ldots = (\texttt{-1}\ \texttt{-1}\ \texttt{1}\ \texttt{1})^\omega.$$

Most notions of fairness are satisfied by this run, because every possible transition out of every state is taken infinitely often. However, σ' is not α-fair. To see why, briefly recall the definition of α-fairness (here somewhat adjusting Pnueli and Zuck's (1993) original definition to our simpler system model): For a run σ, a past formula χ in linear time temporal logic (using S as the set of propositional variables), and a state $s \in S$, call $i \in \omega$ a (χ, s)-position in σ, if $\sigma[i] = s$ and $\sigma_i \models \chi$. A run σ then is called α-fair, if for every past formula χ, every $s \in S$, and every $s' \in t(s)$: if σ contains infinitely many (χ, s)-positions, then $\sigma[i+1] = s'$ for infinitely many (χ, s)-positions i.

Returning to the run σ', we see that it is not α-fair, because letting $\chi \equiv \ominus\texttt{1}$ ("the previous state was `1`"), and considering $s = \texttt{1}$, we find that there are infinitely many (χ, s)-positions in σ', but from each of these (χ, s)-positions the successor state is `-1`. This property of σ', again, would enable an observer to make predictions on the next transition of the system that are always correct and infinitely often nontrivial. The following two definitions make precise the concept of predicting the behavior of a run, and of a run being unpredictable, i.e. computable fair.

Definition 3.1 Let S be a countable set. A prediction algorithm for S is a computable function

$$\pi : S^* \to \mathscr{F}\mathscr{P}(S).$$

Definition 3.2 Let $G = (S, s_0, t)$ be a nondeterministic transition system, $\sigma \in R(G)$. We call σ *computable fair* for G iff there does not exist a prediction algorithm π for S, with

(i) (Correctness) $\forall i \in \omega: \ \sigma[i+1] \in \pi(\sigma_i)$.

(ii) (Non-triviality) For infinitely many i: $\pi(\sigma_i) \subsetneq t(\sigma[i])$.

We denote by $CF(G)$ the set of computable fair runs of G.

To understand the intuition behind this notion of fairness it should be borne in mind that it is intended not so much to be applied to schedulers in concurrent systems in particular, but to capture certain aspects of nondeterministic behavior in general. In fact, one might imagine that the most perfectly fair scheduler ("fair" in the sense of "impartial") in a concurrent system follows a computable policy, and hence would not be computable fair in the sense of definition 3.2. However, such a scheduler would also, in effect, take the nondeterminism out of the concurrent system model. The intention of computable fairness is to formalize, in part, what it means for a system to be truly nondeterministic by taking to a logical conclusion previously proposed concepts of fairness. That computable fairness indeed is a strengthening of previously considered fairness notions is demonstrated by the following proposition.

Proposition 3.3 Every computable fair run is α-fair.

The proof of this proposition is a straightforward generalization of the arguments given above for the sequences σ and σ', observing that it is decidable whether $\sigma_j \in S^j$ ends with a (χ, s)-position, and that thus there is a prediction algorithm π that for some $s' \in t(s)$ and every (χ, s)-position occurring after some step i predicts the next state to belong to $t(s) \setminus s'$.

We now show that we get the theorem of type (1) for computable fairness.

Theorem 3.4 Let G be a nondeterministic transition system, and Γ a corresponding bounded probabilistic transition system. Then

$$P(\Gamma)(CF(G)) = 1.$$

Intuitively, the statement is fairly obvious, because, loosely speaking, a prediction algorithm π can be interpreted as a gambling system, and a path σ for which π is correct and nontrivial as a sequence of outcomes of gambles for which π amounts to a winning strategy. This, of course, will happen with probability 0, and since there are only countably many prediction algorithms, we also get probability 0 for any of them being correct and nontrivial. A rigorous proof, nevertheless, requires a little care. We here give a proof that seems to be the most straightforward one, though it does use some non-elementary concepts from probability theory. It should be noted that it would also be possible to reduce the proof of theorem 3.4 to an application of theorem 1 in (Baier & Kwiatkowska 1998) by identifying computable fairness for G with fairness in the sense of Baier and Kwiatkowska for a suitably defined system G'. We here give our proof only in outline. For the probability theoretic background we refer the reader e.g. to (Loève 1963, §38).

Proof of theorem 3.4: In the following we write P_i and P instead of $P_i(\Gamma)$ and $P(\Gamma)$. We can view a path σ as the path of an S-valued stochastic process $\boldsymbol{X} = (X[i])_{i\in\omega}$ with distribution P. Let π be some prediction algorithm. We define stopping times $(T_j)_{j\in\omega}$ for $\boldsymbol{X}$ as follows (writing $\boldsymbol{X}_i$ for $(X[1], \ldots, X[i])$):

$$T_1 \;:=\; min\{i \mid \forall k < i : X[k+1] \in \pi(\boldsymbol{X}_k);\; \pi(\boldsymbol{X}_i) \subsetneq t(X[i])\} \tag{3}$$

$$T_{j+1} \;:=\; min\{i > T_j \mid \forall k < i : X[k+1] \in \pi(\boldsymbol{X}_k);\; \pi(\boldsymbol{X}_i) \subsetneq t(X[i])\} \tag{4}$$

We use the convention $min\emptyset = \infty$. Thus, the random variable T_j denotes the time of the jth instance that π makes a nontrivial prediction, provided that π has been correct up to that time. Once an incorrect prediction occurs, all further stopping times T_j are set to ∞.

Formally, all the random variables $X[i], T_j$ are defined on some common probability space Ω. Following common practice, we tend to suppress the arguments $\omega \in \Omega$, but note that in more comprehensive notation we would write, for example, $X[k+1](\omega) \in \pi(\boldsymbol{X}_k(\omega))$, rather than $X[k+1] \in \pi(\boldsymbol{X}_k)$.

It is easy to see that for all j, i the event $\{T_j = i\}$ belongs to $\mathfrak{A}_i$, so that the T_j really are stopping times for $\boldsymbol{X}$.

The event $\cap_{j\geq 1}\{T_j < \infty\}$ denotes the set of all paths on which π is correct and nontrivial. We have to show that $P(\cap_{j\geq 1}\{T_j < \infty\}) = 0$. For this it is sufficient to show that

$$P(T_{j+1} < \infty) \leq (1-\epsilon)P(T_j < \infty) \tag{5}$$

for all $j \geq 1$, where ϵ is the probability bound of Γ. If $P(T_j < \infty) = 0$ we are done. Hence, we assume in the sequel that $P(T_j < \infty) > 0$. In that case, (5) is equivalent to

$$P(T_{j+1} < \infty \mid T_j < \infty) \leq 1 - \epsilon. \tag{6}$$

Using that $\boldsymbol{X}$ has the strong Markov property, i.e.

$$P(X[T+1] = s \mid \mathfrak{A}_T) = P(X[T+1] = s \mid X[T]) = \tau(X[T])(s)$$

for every stopping time T for $\boldsymbol{X}$, we obtain (6) by direct computations:

$$\begin{aligned}
&P(T_{j+1} < \infty \mid T_j < \infty) \\
&\quad\leq\ P(X[T_j+1] \in \pi(\boldsymbol{X}_{T_j}) \mid T_j < \infty) \\
&\quad=\ \sum_{s\in S}\sum_{A\in\mathscr{F}\mathscr{P}(S)} P(X[T_j+1] \in A, \pi(\boldsymbol{X}_{T_j}) = A, X[T_j] = s \mid T_j < \infty) \\
&\quad=\ \sum_{s\in S}\sum_{A\in\mathscr{F}\mathscr{P}(S)} P(X[T_j+1] \in A \mid \pi(\boldsymbol{X}_{T_j}) = A, X[T_j] = s, T_j < \infty) \\
&\qquad\qquad\qquad P(\pi(\boldsymbol{X}_{T_j}) = A, X[T_j] = s \mid T_j < \infty) \\
&\quad=\ \sum_{s\in S}\sum_{A\in\mathscr{F}\mathscr{P}(S)} P(X[T_j+1] \in A \mid X[T_j] = s) \\
&\qquad\qquad\qquad P(\pi(\boldsymbol{X}_{T_j}) = A, X[T_j] = s \mid T_j < \infty) \\
&\quad=\ \sum_{s\in S}\sum_{A\in\mathscr{F}\mathscr{P}(S)} \tau(s)(A) P(\pi(\boldsymbol{X}_{T_j}) = A, X[T_j] = s \mid T_j < \infty) \\
&\quad\leq\ (1-\epsilon)\sum_{s\in S}\sum_{A\in\mathscr{F}\mathscr{P}(S)} P(\pi(\boldsymbol{X}_{T_j}) = A, X[T_j] = s \mid T_j < \infty) \\
&\quad=\ 1 - \epsilon.
\end{aligned}$$

Thus, we have shown that the set of paths for which π is a correct and nontrivial prediction algorithm has probability 0. Since by the condition of computability there only are countably many prediction algorithms, the result follows.

□

4 Completeness

We now turn our attention to possible type (2) theorems that we can obtain for computable fairness. As mentioned in the introduction, (2) can be seen as a relativized converse of

(1). Putting (1) and (2) together, we obtain

$$\text{\textit{For all $C \in \mathscr{C}$: C holds with probability 1 in a probabilistic transition system iff C holds for all x-fair runs in the corresponding nondeterministic system.}} \tag{7}$$

Thus, theorem (7) says that the two verification problems – testing whether C has probability 1, and testing whether C holds under the assumption of x-fairness – are equivalent. In particular, (2) is the statement that verifying C under the x-fairness assumption is a *complete* method for probabilistic verification. This is why, following Pnueli and Zuck (1993), we call theorems (2) completeness results (and (1) a *correctness* result). It should be noted, however, that this terminology derives from an original intention of reducing probabilistic verification problems to nondeterministic verification problems. Nothing prevents us to use theorems (1) and (2) also to try to solve verification problems for nondeterministic systems (under the stated fairness assumption) using probabilistic methods – in which case (2) would express correctness, and (1) completeness.

We formalize in a definition:

Definition 4.1 Let S be a set of states, $\mathscr{C}$ a class of properties of S-paths, i.e. $\mathscr{C} \in \mathscr{P}(\mathscr{P}(S^\omega))$. We say that computable fairness is *complete* for $\mathscr{C}$ if the following holds: for every pair G, Γ of corresponding nondeterministic and probabilistic transition systems, and for every $C \in \mathscr{C}$: if $P(\Gamma)(C) = 1$ then $CF(G) \subseteq C \cap R(G)$.

Natural classes $\mathscr{C}$ that we may consider are those that are defined by automata or logics over the alphabet S, e.g. $C \in \mathscr{C}$ iff C is accepted by some Büchi automaton. (The reader is referred to (Thomas 1990) for the background in the theory of ω-languages that we need in the sequel.) To investigate such natural classes, for the remainder of this paper we limit ourselves to finite state spaces S.

We show that for finite S computable fairness is complete for the class of properties definable by deterministic Büchi automata. Compared to Pnueli and Zuck's (1993) result that α-fairness is complete for properties expressible in propositional linear time logic this is not a very strong result, as we seem to be using a much stronger fairness notion, and yet prove completeness only for a class $\mathscr{C}$ that is not strictly more expressive than propositional linear time logic. However, we expect the following result only to be a first step that can be extended to more general classes, in particular the class of ω-regular languages.

For a finite set S we denote by $\mathscr{C}_0$ the class of subsets of S^ω that are definable by deterministic Büchi automata over S, i.e. $C \in \mathscr{C}_0$ iff there exists a deterministic Büchi automaton B for the alphabet S, such that C is the language accepted by B.

Theorem 4.2 Computable fairness is complete for $\mathscr{C}_0$.

Proof: Let B be a deterministic Büchi automaton that accepts C. Let Γ be a probabilistic transition system with $P(\Gamma)(C) = 1$, and G the corresponding nondeterministic transition system. Let $\sigma \in R(G) \setminus C$. We have to show that $\sigma \notin CF(G)$.

The basic idea is very simple: we define a prediction algorithm that tracks the moves defined by σ in B, and eventually predicts moves of G that will keep the path in B away from accepting states. This algorithm will be correct and nontrivial for σ. The details are as follows.

Let $\{q_1, \ldots, q_n\}$ be the set of states of B, and $S = \{s_1, \ldots, s_m\}$ the state set of G. We construct a deterministic Büchi automaton $B \times G$ for the alphabet S as usual: the states of $B \times G$ are the pairs (q_i, s_j) $(i = 1, \ldots, n;\ j = 1, \ldots, m)$. The transition labeled with $s \in S$ leads from a state (q', s') to the state (q'', s), if $s \in t(s')$ and q'' is the s-successor of q' in B. If $s \notin t(s')$ then the s-successor of (q', s') is undefined in $B \times G$. The accepting states of $B \times G$ are all pairs (q, s) for which q is an accepting state of B. The language accepted by $B \times G$ then is just $C \cap R(G)$.

For each state (q, s) let $d(q, s)$ denote the length of the shortest S-path that leads from (q, s) to an accepting state ($d(q, s) = 0$ if (q, s) is an accepting state; $d(q, s) = \infty$ if from (q, s) no accepting state is reachable). For the given path σ define $D(\sigma)$ to be the smallest number i such that the path σ through $B \times G$ passes infinitely often through a state (q, s) with $d(q, s) = i$.

Since $\sigma \notin C$ clearly $D(\sigma) \geq 1$. We also have $D(\sigma) < \infty$: otherwise there would exist a finite prefix σ_k such that no extension of σ_k belongs to $C \cap R(G)$, meaning that $P(C) = P(C \cap R(G)) \leq 1 - P_k(\sigma_k)$. Since $\sigma \in R(G)$, however, we have that $P_k(\sigma_k) > 0$, yielding a contradiction to $P(C) = 1$.

Now let $k \in \omega$ be such that $d(q, s) \geq D(\sigma)$ for all states (q, s) reached by prefixes σ_i with $i \geq k$. We define a prediction algorithm π on S^* by letting for $\boldsymbol{s} \in S^*$ of length l, and with last element s:

$$\pi(\boldsymbol{s}) = \begin{cases} t(s) & \text{if } l < k \\ t(s) \setminus \{s' \mid d(q(\boldsymbol{s}s'), s') < D(\sigma)\} & \text{if } l \geq k \end{cases}$$

where $q(\boldsymbol{s}s')$ is the state of B reached by the sequence $\boldsymbol{s}s'$. Since $\sigma \in R(G)$ we have that π is correct for σ, and since $d(q(\sigma_i), \sigma[i]) = D(\sigma) < \infty$ infinitely often, we have that $t(s) \setminus \{s' \mid d(q(\boldsymbol{s}s'), s') < D(\sigma)\} \neq t(s)$ infinitely often, hence π is nontrivial. $\square$

Theorem 4.2 gives one example of a class $\mathscr{C}$ of path properties for which computable fairness is an adequate approximation of randomness, in the sense that (7) holds for $\mathscr{C}$. Clearly, the class $\mathscr{C}_0$ is not maximal with this property, i.e. there exist $C \in \mathscr{P}(S^\omega) \setminus \mathscr{C}_0$ such that $P(\Gamma)(C) = 1$ implies $CF(G) \subseteq C \cap R(G)$ for all corresponding pairs Γ, G of systems. What, then, can we say about the maximal set

$$\mathscr{C}^* := \{C \in \mathscr{P}(S^\omega) \mid \forall \,\text{corresponding}\, \Gamma, G:\ P(\Gamma)(C) = 1 \Rightarrow CF(G) \subseteq C\}$$

for which computable fairness is complete? Finding meaningful characterizations or approximations of $\mathscr{C}^*$ is an interesting topic for future work. We close this section by discussing two examples of properties that do not belong to $\mathscr{C}^*$.

Taking $S = \{\,-1, 1\}$ we define

$$C_0 := \{\sigma \in \{-1, 1\}^\omega \mid \text{for infinitely many}\, k:\ \sum_{i=1}^{k} \sigma[i] = 0\}.$$

To see that $C_0 \notin \mathscr{C}^*$ consider two different probabilistic transition systems Γ_1 and Γ_2, obtained by labeling the transitions of the system G in figure 1 with probabilities as follows: for Γ_1 every transition is assigned probability 0.5; for Γ_2 transitions leading to -1 are assigned probability 0.3, and transitions leading to 1 probability 0.7. Then, by well known results on the one-dimensional random walk, $P(\Gamma_1)(C_0) = 1$, but

$P(\Gamma_2)(C_0) = 0$. From $C_0 \in \mathscr{C}^*$ we could infer $CF(G) \subseteq C_0$, which with theorem 3.4 would yield $P(\Gamma_2)(C_0) = 1$, a contradiction. More generally, we can say that $C \notin \mathscr{C}^*$ whenever $P(\Gamma_1)(C) = 1 \neq P(\Gamma_2)(C)$ for two probabilistic systems Γ_1, Γ_2 with the same corresponding nondeterministic system G.

This gives us a necessary, but not a sufficient condition for membership in $\mathscr{C}^*$: consider

$$C_1 := \{\sigma \in \{-1, 1\}^\omega \mid \frac{1}{k}\sum_{i=1}^{k} \sigma[i] \text{ converges for } k \to \infty\}.$$

Here we get $P(\Gamma)(C_1) = 1$ for all probabilistic transition systems Γ over $\{-1, 1\}$. However, there are paths $\sigma \notin C_1$ that are computable fair for the nondeterministic system of figure 1. To "construct" an example for such a σ, take two different probabilistic systems corresponding to G: the system Γ_2 defined above, and the system Γ_3 defined by assigning probability 0.3 to transitions leading to 1, and 0.7 to transitions leading to -1. Now generate σ as follows: use system Γ_2 to randomly generate $\sigma[1], \sigma[2], \ldots, \sigma[k_1]$ until $1/k_1 \sum_{i=1}^{k_1} \sigma[i] \geq 0.4$. Then use system Γ_3 to sample subsequent states $\sigma[k_1+1], \ldots, \sigma[k_2]$, until $1/k_2 \sum_{i=1}^{k_2} \sigma[i] \leq -0.4$, change again to system Γ_2, and continue in this manner. With probability 1 then we will switch infinitely often between using systems Γ_2 and Γ_3, i.e. a sequence $\sigma \notin C_1$ will be generated. By similar arguments as used in the proof of theorem 3.4, it furthermore follows that with probability 1 σ will be computable fair.

5 Conclusion

In this paper we have presented some initial steps towards a study of fairness as a characterization of true nondeterminism, which, in turn, can be seen as a qualitative approximation of randomness. This work is much indebted to two sources of inspiration: Pnueli and Zuck's (1993) treatment of α-fairness, and the classical characterizations of randomness by Martin-Löf(1966) and Schnorr (1971) (and others).

The equivalence proved by Pnueli and Zuck between probabilistic validity and validity for α-fair computations (in conjunction with similar results obtained for other fairness notions) suggests to interpret concepts of fairness as partial descriptions of probabilistic behavior. This point of view, then, views fairness in a similar light as concepts of invariance under selection rules, or of passing certain tests of randomness, which have been devised to define randomness, e.g. (Martin-Löf 1966, Schnorr 1971). Just as in these classic approaches to defining randomness one has to identify via conditions of computability an adequate subclass of selection rules, tests of randomness, etc., we here have aimed to define through the concept of computable fairness a class of fairness conditions that provides an adequate approximation of randomness for nondeterministic systems.

That said, it should be emphasized, however, that the basic aim of the classic characterizations of randomness is different from ours, in that they want to capture many properties of randomness (e.g. convergence to limiting frequencies) that we do not want to enforce by fairness constraints. Moreover, they treat randomness as an intrinsic property of a sequence σ, whereas we have to define fairness of σ always with respect to a given transition system G.

References

Baier, C. & Kwiatkowska, M. (1998), 'On the verification of qualitative properties of probabilistic processes under fairness constraints', *Information Processing Letters* **66**, 71–79.

Courcoubetis, C. & Yannakakis, M. (1995), 'The complexity of probabilistic verification', *Journal of the ACM* **42**(4), 857–907.

Loève, M. (1963), *Probability Theory*, 3rd edn, D. Van Nostrand Comp.

Martin-Löf, P. (1966), 'The definition of random sequences', *Information and Control* **6**, 602–619.

Pnueli, A. (1983), On the extremely fair treatment of probabilistic algorithms, *in* 'Proceedings of the 18th IEEE Symposium on Foundation of Computer Science'.

Pnueli, A. & Zuck, L. D. (1993), 'Probabilistic verification', *Information and Computation* **103**, 1–29.

Schnorr, C. P. (1971), 'A unified approach to the definition of random sequences', *Math. Systems Theory* **5**, 246–258.

Thomas, W. (1990), Automata on infinite objects, *in* J. van Leeuwen, ed., 'Handbook of Theoretical Computer Science', Vol. 2, Elsevier/MIT Press.

Vardi, M. Y. (1985), Automatic verification of probabilistic concurrent finte-state programs, *in* 'Proceedings of the 26th IEEE Symposium on Foundations of Computer Science', pp. 327–338.