Equivalences for Fair Kripke Structures

Adnan Aziz[*] Felice Balarin[†] Vigyan Singhal[†]
Robert Brayton[‡] Alberto Sangiovanni-Vincentelli[‡]

[*]Department of ECE, University of Texas, Austin, TX 78712
[†]Cadence Berkeley Labs, Berkeley CA 94704
[‡]Department of EECS, University of California, Berkeley, CA 94720

Abstract

We extend the notion of bisimulation to Kripke structures with fairness. We define equivalences that preserve fairness and are akin to bisimulation. Specifically we define an equivalence and show that it is complete in the sense that it is the coarsest equivalence that preserves the logic CTL* interpreted with respect to the fair paths. The addition of fairness might cause two Kripke structures, which can be distinguished by a CTL* formula, to become indistinguishable by any CTL formula. We define a weaker equivalence that is the weakest equivalence preserving CTL interpreted on the fair paths. As a consequence of our proofs, we also obtain characterizations of states in the fair structure in terms of CTL* and CTL formulae.

1 Introduction

Branching time propositional temporal logic has been found very useful in the automatic verification of concurrent finite-state systems [4]. Systems are modeled using labeled state transition structures called Kripke or temporal structures [8]. Properties are expressed as formulae from a branching-time temporal logic.

One of the simplest such logics is CTL (Computational Tree Logic) described in [6]. While the problem of model-checking CTL formulae of a Kripke structure is of polynomial complexity [8], CTL suffers in expressiveness. The richer logic CTL*, described in [10], adds the power of linear-time propositional logic to CTL, and subsumes both CTL and PLTL (Propositional Linear Time Logic). However, the problem of model-checking becomes PSPACE-complete [6].

At the initial stages of top-down verification driven design methodologies [15], nondeterminism provides a powerful mechanism for modeling abstract designs. However, nondeterminism can introduce too much behavior. Fairness is used to restrict analysis to those infinite paths in the Kripke structure which satisfy some specification, which is evaluated over the infinite path. A major limitation of CTL is that it cannot express correctness under fairness constraints [8].

The logic FairCTL allows the specification of a CTL formula p along with a path formula ϕ. The fairness constraint ϕ is a pure path formula, specifically a Boolean combination of the set of infinitary linear-time operators applied to propositional arguments [8, page 1063]. The path quantifiers in the syntax of the formula now range over only those infinite paths which meet the fairness constraint ϕ. In [6], a more general specification CTL$^{\mathrm{F}}$ is allowed, where the fairness constraints can refer to individual states. Such fine granularity allows us to distinguish between two different states which cannot be distinguished by any propositional temporal logic formulae. The notion of fairness used in our paper is the extension of [8], where we allow the infinitary linear-time operators in ϕ to refer to state labels also. The model-checking problem for FairCTL (and CTL$^{\mathrm{F}}$) can be solved in polynomial time.

Often, it is more natural to think of the fairness constraints as part of the system specification, instead of part of the property being verified. We will refer to

Kripke structures with fairness constraints as fair-Kripke structures, and the problem of checking a CTL (or CTL*) formula on a fair-Kripke structure as the fair–CTL (or fair–CTL*) model-checking. Since we allow the fairness constraints to be Boolean combinations of infinitary linear-time operators applied to *state labels*, the fair-Kripke structure specification is as powerful as any kind of ω-automaton [17], and in fact is essentially a syntactic variant on the theme of ω-automata [20].

Browne, Clarke and Grumberg [3] proved that bisimulation equivalence [16] on Kripke structures is exactly the weakest equivalence that preserves all CTL and CTL* formulae. In doing so, they constructed CTL formulae which characterized Kripke structures up to bisimulation equivalence. As a corollary, they showed that any two Kripke structures which can be distinguished by a CTL* formula can also be distinguished by a CTL formula. In this paper, we solve an open problem proposed in [3], namely that of characterizing fair-Kripke structures up to equivalence over branching-time temporal logics.

We show that, unlike ordinary Kripke structures, there exists a pair of fair-Kripke structures which can be distinguished by a CTL* formula, whereas no CTL formula can distinguish these two. In fact, these two structures are not even trace equivalent, which is surprising because in the case of ordinary Kripke structures bisimulation equivalence is the finest equivalence and trace equivalence is the coarsest equivalence in the linear-time — branching-time spectrum [21].

Since, for fair-Kripke structures, the notion of equivalence is different for CTL and CTL* formulae, we provide two different extensions of bisimulation equivalence which deal with fairness constraints. $\mathcal{E}^{fair}$-bisimulation characterizes states in fair-Kripke structures with respect to equivalence over CTL formulae, and $\mathcal{E}^{fair*}$-bisimulation characterizes states in fair-Kripke structures with respect to equivalence over CTL* formulae.

The problem of fair–CTL* model-checking can be solved using the algorithm for CTL* model-checking by introducing a new atomic proposition for each state in the Kripke structure. and then transforming the formula using these additional atomic propositions [6]. However, this means that the characterization of states in Kripke structures for CTL* equivalence as shown in [3] does not solve the problem of characterizing states in *fair* Kripke structures, since the additional atomic propositions result in the ability of CTL to differentiate *all* states. Again, this is especially important when one considers fairness constraints as part of the *system* specification.

The remainder of this paper is organized as follows: In Section 2 we give the definitions of fair-Kripke structures and CTL/CTL* syntax and semantics on such structures. In Section 3, the relationship between bisimulation and CTL/CTL* model-checking is reviewed. In Section 4 we present the definitions of $\mathcal{E}^{fair*}$–bisimulation and $\mathcal{E}^{fair}$–bisimulation, and prove completeness results for both equivalences. We conclude in Section 5 with plans for future research and applications of our results.

2 Definitions

2.1 Sequences

A *finite sequence* on a set A is a function whose range is A and domain is a prefix of the set of natural numbers, $\omega = \{0, 1, 2, \ldots\}$. An *infinite sequence* on A is a function mapping ω to A. We will denote the finite sequence α by $\langle a_0, a_1, \cdots, a_{n-1}\rangle$; an infinite sequence β will be written as $\langle b_0, b_1, b_2, \cdots\rangle$. Given a sequence α (finite or infinite), we will denote by $[\alpha]_k$ the k-th element in the sequence, i.e., $\alpha(k)$. The elements of the range that occur infinitely often in an infinite sequence α will be denoted by $\mathtt{inf}(\alpha)$.

The *length* of a finite sequence is the cardinality of its domain. Given finite sequences α_1 and α_2, of length l_{a_1} and l_{a_2} respectively, we define the *concatenation* of α_1 and α_2 to be the sequence α of length $l_{a_1} + l_{a_2}$ taking value $\alpha_1(k)$ when $k < l_{a_1}$, and $\alpha_2(k - l_{a_1})$, when $k \geq l_{a_1}$; this will be denoted by $\alpha_1 \cdot \alpha_2$. The notion of concatenation readily extends to the case where α_2 is an infinite sequence.

For a finite nonempty sequence α of length l_α, we use α^ω to denote the infinite sequence which for any natural number k evaluates to $\alpha_{k \bmod l_\alpha}$. Given any infinite sequence α and natural number k, the k-th prefix of α is the finite sequence $\langle\alpha(0), \alpha(1), \cdots, \alpha(k-1)\rangle$, and the k-th suffix of α is the sequence $\langle\alpha(k), \alpha(k+1), \alpha(k+2), \cdots\rangle$. Observe that α is the concatenation of its k-th prefix with its k-th suffix.

2.2 Fair Kripke Structures

A Kripke structure K is a triple $(S, T, \mathcal{L})$, where

1. S is a finite set of *states*.
2. $T \subset S \times S$ is the *transition relation*.
3. $\mathcal{L} : S \rightarrow 2^{AP}$ is the *labeling function* and AP is the underlying set of *atomic propositions*.

The infinite sequence of states $\sigma = \langle s_0, s_1, s_2, s_3, \cdots \rangle$ is said to be a *path* starting at state s_0 if for every k we have $T(s_k, s_{k+1})$.

Intuitively, fairness conditions express restrictions on the infinitary behavior of the system; a path is said to be *fair* if is satisfies the condition. Various formalisms exist for defining fair paths, e.g., Büchi (variously *weak* or *unconditional*), Streett (variously *strong* or *conditional*), Rabin, and Muller conditions [17, 20]. An important observation is that in all these formalisms, fairness of the path is a function of the set of states occurring infinitely often on the path.

Definition 1 A **Muller Fairness condition** (MFC) f on Kripke structure K is characterized by a class $\mathcal{C} = \{M_1, M_2, \ldots, M_n\}$ of subsets of S; the path σ is fair, if and only if the set of states occurring infinitely often in σ, (denoted by $\mathtt{inf}(\sigma)$) is an element of $\mathcal{C}$. The sets M_i will be referred to as the **Muller fair** subsets. We will use $\mathcal{F}^f_{s_0}$ to denote the set of all paths starting at s_0 that satisfy the MFC f.

Since in all formalisms of fairness it is the case that fairness of a path is a function of the set of states occurring infinitely often on the path, Muller conditions can express arbitrary constraints on the set of infinitary states. Thus without loss of generality, we will restrict our analysis to Kripke structures with Muller fairness conditions, which will be referred to as **fair-Kripke structures**, denoted by the 4-tuple $(S, T, \mathcal{L}, f)$. Also, to simplify analysis we will always assume that every state lies on a fair path. This is not a serious restriction; note that under Muller fairness conditions, one can compute the set of states lying on a fair path by simple strongly connected components calculation [7].

2.3 CTL/CTL* Model Checking on fair-Kripke structures

There are two types of formulae in CTL and CTL*: state formulae (which are true or false in a specific state), and path formulae (which are true or false along a specific path). Let AP be the set of atomic propositions. A state formula is given by the following syntax:

1. $\underline{\text{a}}$ if a $\in$ AP.

2. If f_1 and f_2 are state formula, then so are $\neg f_1$ and $f_1 \vee f_2$.

3. If g is a path formula, then $\exists g$ and $\forall g$ are state formulae.

A path formula is given by the following syntax:

1. A state formula

2. If g_1 and g_2 are path formulae, then so are $\neg g_1$ and $g_1 \vee g_2$.

3. If g_1 and g_2 are path formulae, then so are Xg_1 and $g_1 U g_2$.

CTL* is the set of state formulae that are generated by the above rules; CTL is a subset of CTL* in which the path formulae are restricted to be:

1. If f_1 and f_2 are state formulae then Xf_1 and $f_1 U f_2$ are path formulae.

This clause replaces clauses 1–3 from the previous definition of path formula.

Given a fair-Kripke structure $K = (S, R, \mathcal{L}, f)$, state and path formulae are interpreted over states and fair paths as defined below. The formulae f_1 and f_2 are state formulae, and g_1 and g_2 are path formulae.

1. $s_0 \models \underline{\text{a}}$ if and only if $a \in \mathcal{L}(s_0)$

2. $s_0 \models \neg f_1$ if and only if $s_0 \not\models f_1$, and $s_0 \models f_1 \vee f_2$ if and only if $s_0 \models f_1$ or $s_0 \models f_2$

3. (a) $s_0 \models \exists g_1$ if and only if there exists a **fair** path π starting at s_0 such that $\pi \models g_1$, and

   (b) $s_0 \models \forall g_1$ if and only if for all **fair** paths π starting at s_0, $\pi \models g_1$

4. $\pi \models f_1$ if and only if $[\pi]_0 \models f_1$
5. $\pi \models \neg g_1$ if and only if $\pi \not\models g_1$, and $\pi \models g_1 \vee g_2$ if and only if $\pi \models g_1$ or $\pi \models g_2$
6. (a) $\pi \models X g_1$ if and only if $\pi^1 \models g_1$
   (b) $\pi \models g_1 U g_2$ if and only if there exists a $k \geq 0$ such that $\pi^k \models g_2$ and for all $0 \leq j < k$, $\pi^j \models g_1$

It will be convenient to denote k concatenations of the next time operator X by X^k. Given a path formula g_1, we will use the abbreviation Gg_1 to denote the CTL* path formula $\neg(\texttt{TRUE}\, U \neg g_1)$, where TRUE is a logical tautology. It can be shown that the formulae of CTL presented above can be transformed into logically equivalent formula which use only $\exists X$, $\exists U$, and $\exists G$ as temporal connectives [8].

3 Equivalences preserving CTL/CTL*

Let $K = (S, T, \mathcal{L})$ be a Kripke structure. An equivalence relation $\mathcal{E} \subset S \times S$ is said to be a bisimulation on K if the following holds:

$$\begin{aligned} \mathcal{E}(s,t) \quad \Rightarrow \quad & \big(\mathcal{L}(s) = \mathcal{L}(t)\big) \wedge \\ & \forall s' \Big(T(s,s') \rightarrow \exists t' \big(T(t,t') \wedge \mathcal{E}(s',t')\big)\Big) \wedge \\ & \forall t' \Big(T(t,t') \rightarrow \exists s' \big(T(s,s') \wedge \mathcal{E}(s',t')\big)\Big) \end{aligned} \tag{1}$$

Observe:

1. the identity relation satisfies the above, and
2. given any two distinct bisimulations $\mathcal{E}_1$ and $\mathcal{E}_2$, their union, i.e., $\mathcal{E}_1 \cup \mathcal{E}_2$, is also a bisimulation.

From the above two observations, it follows that there exists a unique maximal bisimulation, which will be denoted by $\mathcal{E}^{bis}$.

In the absence of fairness conditions on the paths through the Kripke structure, Browne, Clarke, and Grumberg prove the following soundness and completeness result [3]:

Theorem 3.1 Let K be a Kripke structure, and and s and t be states from K. Then $\mathcal{E}^{bis}(s,t)$ if and only if there is no CTL* formula ϕ such that $s \models \phi$ and $t \not\models \phi$.

They construct CTL formulae that characterize states and structures up to bisimulation equivalence. As a corollary, they note that states in a Kripke structure that can be differentiated by a formula of CTL* can also be differentiated by a formula of CTL.

4 Equivalences on fair-Kripke structures

In the presence of fairness, states that have the same branching structure may have different infinitary behavior. In the fair-Kripke structure defined in Figure 1, the Muller fairness condition is $\{U_1, V_1\}$ (shown in the dotted boxes), and the set of APs is $\{a, b\}$. States s_0 and t_0 have identical finite branching structure, but state t_0 satisfies the CTL formula $\exists Ga$ (there exists a path on which a is always true), while $s_0 \not\models \exists Ga$.

In this section we define two equivalences on the states of fair-Kripke structures. We prove completeness results with respect to CTL* and CTL, using arguments analogous to those in [3]. Essentially, our equivalences incorporate fairness constraints by requiring that states be equivalent on all fair paths. We show that it suffices to examine a restricted class of paths, namely the rational paths defined below.

Definition 2 A path σ is a *rational path* if there exist natural numbers N and M such that for every natural number i greater than N it is the case that $[\sigma]_{N+i} = [\sigma]_{N+(i \bmod M)}$. Thus rational paths are those which end in a cycle.

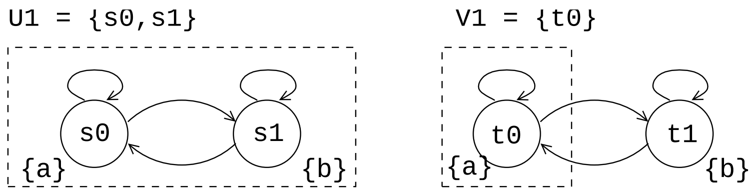

Figure 1: States that agree on all CTL formulae in the absence of fairness

Given an equivalence relation $\mathcal{E}$ on the state space of a Kripke structure, we extend it to an equivalence $\mathcal{E}^\omega$ on paths through the Kripke structure as follows: $\mathcal{E}^\omega(\sigma, \tau)$ holds exactly when for every natural number i, we have $\mathcal{E}([\sigma]_i, [\tau]_i)$. In the sequel, we will overload $\mathcal{E}$ and use it when we mean $\mathcal{E}^\omega$; we will refer to σ and τ as being $\mathcal{E}$-equivalent.

4.1 Equivalences preserving CTL* on fair-Kripke structures

Definition 3 Let K be a fair-Kripke structure $(S, T, \mathcal{L}, f)$. An equivalence relation $\mathcal{E} \subset S \times S$ is said to be a $fair*$ bisimulation if for any $(s, t) \in \mathcal{E}$ the following are true:

1. $\big(\mathcal{L}(s) = \mathcal{L}(t)\big)$, and
2. for every fair rational path σ starting at s there exists a fair rational path τ starting at t such that $\mathcal{E}(\sigma, \tau)$, and
3. for every fair rational path τ starting at t there exists a fair rational path σ starting at s such that $\mathcal{E}(\tau, \sigma)$.

The relation $\mathcal{E}^{fair*}$ is defined to be the coarsest relation that is a $fair*$ bisimulation.

The soundness of this definition follows in a manner analogous to that of the definition of $\mathcal{E}^{bis}$, as given in Section 4.

At first glance, the definition of $\mathcal{E}^{fair*}$ appears unnatural in that it restricts the paths σ and τ to be rational. However, we will later prove (cf. Lemma 4.3) that if we drop the requirement of rationality, the resulting equivalence relation is equal to $\mathcal{E}^{fair*}$ given in Definition 3. Note that under the assumption that every state lies on a fair path, the $\mathcal{E}^{fair*}$ relation is finer than bisimulation.

The $\mathcal{E}^{fair*}$-bisimilarity relation is the natural extension to bisimulation in the presence of fairness in the sense that the following soundness and completeness result holds:

Theorem 4.1 Let s and t be states in a fair-Kripke structure $K = (S, T, \mathcal{L}, f)$. Then $\mathcal{E}^{fair*}(s, t)$ if and only if there is no CTL* formula ϕ such that $s \models \phi$ and $t \not\models \phi$.

Proof: Proving soundness, i.e., that $\mathcal{E}^{fair*}(s,t) \Rightarrow$ for every CTL* formula ϕ we have $s \models \phi$ if and only if $t \models \phi$ is straightforward; simply use induction on the length of the CTL* formulae.

To show completeness, i.e., that $\neg\mathcal{E}^{fair*}(s,t) \Rightarrow$ there exists a formula ϕ of CTL* such that $s \models \phi \ \wedge \ t \not\models \phi$, we first define a series of equivalence relations $E_0, E_1, \ldots$ as follows:

1. $E_0(s,t)$ if and only if $\mathcal{L}(s) = \mathcal{L}(t)$.
2. $E_{k+1}(s,t)$ if and only if
   (a) for every fair rational path σ starting at s there is a fair rational path τ starting at t which is E_k-equivalent to σ, and
   (b) for every fair rational path τ starting at t there is a fair rational path σ starting at s which is E_k-equivalent to τ.

Observe that $E_{l+1}(s,t) \subseteq E_l(s,t)$. Also every equivalence in the sequence contains the equality relation. Since the state space is finite, the sequence converges to a fixed point in some finite number of steps, i.e., there is some k such that $E_{k+1} = E_k$, which we will refer to as E_∞.

Note that E_∞ satisfies Conditions 1–3 of Definition 3, and hence E_∞ must lie in $\mathcal{E}^{fair*}$, since by definition, $\mathcal{E}^{fair*}$ is the coarsest relation satisfying these conditions. Thus, if we show that states which are not E_∞-equivalent can be differentiated by CTL* formulae, we will have shown that states which are not $\mathcal{E}^{fair*}$-equivalent can be differentiated by CTL* formulae.

We now characterize states up to E_l-equivalence by CTL*-equivalence. This is done by induction on l. Specifically we will prove that

1. If $\neg\big(E_l(s,t)\big)$ then there is a CTL* formula $d_l(s,t)$ such that for each state v which is E_l-equivalent to s we have $v \models d_l(s,t)$, and $t \not\models d_l(s,t)$.
2. For every state s, there is a CTL* formula $C_l(s)$ such that for every state t we have $t \models C_l(s)$ if and only if $E_l(s,t)$.

The formula $d_l(s,t)$ distinguishes between t and states E_l-equivalent to s; the formula $C_l(s)$ characterizes E_l-equivalence to state s within the fair-Kripke structure.

If we let $C_l(s)$ be the conjunction of $C_{l-1}(s)$ and $d_l(s,t)$ for every t which is not E_l-related to s, the second assertion follows immediately. Now it is necessary to show how to construct $d_l(s,t)$ by induction on l.

Base Case:$(l=0)$

Let $\{p_i\}$ be the set of atomic propositions in $\mathcal{L}(s)$ and $\{q_j\}$ be the set of atomic propositions in AP $-\ \mathcal{L}(s)$. Now let $d_0(s,t) = \bigwedge_i \underline{p_i} \wedge \bigwedge_j \neg \underline{q_j}$. It is clear that this formula is only true in states with exactly the same labeling as s. Thus the base case is established.

Induction Hypothesis (IH):

Assume the result is true for l. We will show it for $l+1$.

Let s and t be any states in the structure such that $\neg\big(E_{l+1}(s,t)\big)$. This can only happen if there is a fair rational path from s for which there is no E_l-corresponding fair rational path out of t, or there is a fair rational path from t for which there is no E_l-corresponding fair rational path out of s. In the latter case, we will use the argument below to find a $d_{l+1}(t,s)$ such that $t \models d_{l+1}(t,s)$ and $s \not\models d_{l+1}(t,s)$. We can negate this formula to obtain the desired $d_{l+1}(s,t)$.

Let $\sigma = \langle s_0, s_1, s_2, \cdots, s_{N-1}\rangle \cdot \langle s_N, \cdots, s_{N+k-1}\rangle^\omega$ be a fair rational path from s with no corresponding path from t, where we take $s_0 = s$ for notational convenience.

First define the CTL* path formulae $\mathtt{stem}_{l+1}(s,t)$ to be

$$\mathtt{stem}_{l+1}(s,t) \quad = \quad C_l(s_0) \wedge XC_l(s_1) \wedge X^2C_l(s_2) \wedge \cdots \wedge X^{N-1}C_l(s_{N-1})$$

Now define the path formula $\mathtt{cycle}_{l+1}(s,t)$ to be

$$\mathtt{cycle}_{l+1}(s,t) \quad = \quad C_l(s_N) \wedge XC_l(s_{N+1}) \wedge X^2C_l(s_{N+2}) \wedge \cdots \wedge X^{k-1}C_l(s_{N+k-1})$$

Then define the CTL* path formulae $\mathtt{repeat}^i_{l+1}(s,t)$ for $i \in \{0,1,\ldots,k-1\}$ to be

$$\mathtt{repeat}^i_{l+1}(s,t) \quad = \quad \big(C_l(s_{N+i}) \rightarrow X^kC_l(s_{N+i})\big)$$

Finally, define the path formula $\texttt{path}_{l+1}(s,t)$ to be

$$\begin{aligned}\texttt{path}_{l+1}(s,t) \quad = \quad & \texttt{stem}_{l+1}(s,t) \quad \wedge \\ & X^N\big(\texttt{cycle}_{l+1}(s,t)\big) \quad \wedge \\ & X^N\left(G\Big(\bigwedge_{i=0}^{k-1} \texttt{repeat}^i_{l+1}(s,t)\Big)\right)\end{aligned}$$

Proposition 4.2 A path υ satisfies the path formula $\texttt{path}_{l+1}(s,t)$ if and only if it is E_l-equivalent to the path σ.

Proof: The "if" part is straightforward; the E_l-equivalent fair path will satisfy $\texttt{path}_{l+1}(s,t)$.

We now prove the "only if" part. Observe that states $[\upsilon]_0, [\upsilon]_1, \dots, [\upsilon]_{N-1}$ must satisfy $C_l(s_0), C_l(s_1), \dots, C_l(s_{N-1})$ respectively. By the IH, any state satisfying $C_l(u)$ must be E_l-equivalent to u. Hence, the N-th prefix of υ is E_l-equivalent to the N-th prefix of σ.

Now consider the N-th suffix of υ. Again, by construction of $\texttt{cycle}_{l+1}(s,t)$, the states $[\upsilon]_N, [\upsilon]_{N+1}, \dots, [\upsilon]_{N+k-1}$ must satisfy $C_l(s_N), C_l(s_{N+1}), \dots, C_l(s_{N+k-1})$ respectively. Again invoking the IH, we see that states $[\upsilon]_N, [\upsilon]_{N+1}, \dots, [\upsilon]_{N+k-1}$ are E_l-equivalent to $[\sigma]_N, [\sigma]_{N+1}, \dots, [\sigma]_{N+k-1}$ respectively.

Since υ^N satisfies $G\Big(\bigwedge_{i=0}^{k-1} \texttt{repeat}^i_{l+1}(s,t)\Big)$, we see that for each $i \in \{0, 1, \dots, k-1\}$, we have $[\upsilon]_{N+i}$ satisfies $\texttt{repeat}^i_{l+1}(s,t)$. Hence by the IH, all states of the form $[\upsilon]_{N+i+\lambda\cdot k}$, where λ is an arbitrary natural number, are E_l-equivalent. Thus N-th suffix of υ is E_l-equivalent to the N-th suffix of of σ. This proves the proposition. ■

Now define $d_{l+1}(s,t)$ to be the state formula $\exists\,\texttt{path}_{l+1}(s,t)$. Clearly, the state s satisfies $d_{l+1}(s,t)$ (take σ to be the witness), as does any state E_l-equivalent to s (since E_l-equivalence implies the existence of a path E_l-equivalent to σ). Furthermore, since there does not exist a path from t which is E_l-equivalent to σ, t does not satisfy $d_{l+1}(s,t)$. ■

In Definition 3, states were taken to be equivalent over rational fair paths. The following lemma demonstrates that equivalence over rational fair paths implies equivalence over *all* fair paths, establishing a more intuitive characterization of $\mathcal{E}^{fair*}$-bisimulation.

Lemma 4.3 Let K be a given fair-Kripke structure. Let $\mathcal{E}$ be an equivalence relation on states satisfying the following:

$$\begin{aligned}
\mathcal{E}(s,t) \quad \Leftrightarrow \quad & \big(\mathcal{L}(s) = \mathcal{L}(t)\big) \wedge \\
& \big(\text{for every fair rational path}\, \sigma \,\text{starting at } \; s \\
& \quad \text{there exists a fair rational path}\, \tau \,\text{starting at}\; t \,\text{such that } \; \mathcal{E}(\sigma,\tau)\big) \wedge \\
& \big(\text{for every fair rational path}\, \tau \,\text{starting at } \; t \\
& \quad \text{there exists fair rational path}\, \sigma \,\text{starting at } \; s \,\text{such that } \; \mathcal{E}(\sigma,\tau)\big)
\end{aligned}$$

Then $\mathcal{E}$ preserves equivalence across all fair paths, i.e., for every fair path σ starting at s there exists a fair path τ starting at t such that $\mathcal{E}(\sigma,\tau)$, and for every fair path τ starting at t there exists a fair path σ starting at s such that $\mathcal{E}(\tau,\sigma)$.

Proof: Let $\{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n\}$ be the equivalence classes of $\mathcal{E}$. Define an alphabet $\Sigma = \{c_1, c_2, \ldots, c_n\}$ corresponding to the equivalence classes. Define the ω-language L_s over Σ as $L_s = \{x \in \Sigma^\omega \mid \exists \sigma \in \mathcal{F}_s^f \text{ such that } (\forall i)\, [\sigma]_i \in \mathcal{C}([x]_i)\}$, where $\mathcal{C} : \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n\} \rightarrow \Sigma$ maps alphabets to their corresponding equivalence classes. Observe that L_s is ω-regular [20], since the fair-Kripke structure can be viewed as a Muller automaton, with the output at a state being the symbol of Σ corresponding to its equivalence class. Similarly, define the ω-regular language $L_t = \{y \in \Sigma^\omega \mid \exists \tau \in \mathcal{F}_t^f \text{ such that } \forall i\, [\tau]_i \in \mathcal{C}([y]_i)\}$.

It is clear that given any fair path σ starting at s, there is a fair path τ starting at t which is $\mathcal{E}$-equivalent to σ, and vice versa, if and only if $L_s = L_t$.

Note that if W_1 and W_2 are two ω-regular languages over the same alphabet, and they agree on all rational words (i.e., words that are ultimately periodic), then they are in fact equal; this follows from the following observation: Let W be the symmetric difference of W_1 and W_2, i.e., $W = (W_1 \cap \bar{W_2}) \cup (\bar{W_1} \cap W_2)$. Then W is ω-regular, and so is non-empty if and only if it contains a rational word [20]. But W_1 and W_2 contain exactly the same set of rational words; consequently W is empty, and so $W_1 = W_2$.

Since L_1 and L_2 agree on all rational words (because s and t agree on all rational paths), it follows that $L_1 = L_2$, and so s and t must agree on all paths, proving the lemma. ■

4.1.1 Upper bounds on computing $\mathcal{E}^{fair*}$

The previous lemma can be used to derive a PSPACE upper bound on the complexity of deciding $\mathcal{E}^{fair*}$ for Kripke structures with practical forms of expressing fairness. Specifically, we consider the case of Büchi fairness constraints, which due to their succinctness relative to Muller fairness constraints are more meaningful in practice.

A Büchi fairness constraint is a subset B of the state space of the Kripke structure; a path σ is fair if $\mathtt{inf}(\sigma) \cap B \neq \emptyset$.

Proposition 4.4 The $\mathcal{E}^{fair*}$ equivalence relation can be computed in PSPACE for Kripke structures with Büchi fairness constraints.

Proof: Consider the equivalences $E_0, E_1, \ldots$ as defined in the proof of Theorem 4.1. Observe that E_{k+1} can be derived from E_k by viewing the Kripke structure as a Büchi automaton where the state label is the E_k-equivalence class to which it belongs, and the acceptance is the corresponding Büchi fairness constraint. The relation E_{k+1} is simply the language equivalence relation on the states of this automaton [20].

Since $\mathcal{E}^{fair*} = E_\infty$ is reached from E_0 in at most n steps, where n is the number of states, it is clear that the fair bisimulation relation can be computed using a polynomial number of calls to trace equivalence on Büchi automata which are no larger than the original Kripke structure. Trace equivalence for Büchi automata is known to be complete for PSPACE [18]. Consequently, $\mathcal{E}^{fair*}$ can be decided in PSPACE for Kripke structures with such fairness constraints. ■

Given the close relationship between trace equivalence and fair bisimilarity shown above, it is reasonable to conjecture that deciding fair bisimilarity is as hard as deciding trace equivalence which is PSPACE-complete; this was proved by Kupferman and Vardi [14], who used a reduction from the universality problem for regular expressions, similar to that used to show trace universality is PSPACE-hard [13, 19].

4.2 Equivalences preserving CTL on fair-Kripke structures

The logic CTL is a subset of the logic CTL* where nesting of path operators is not allowed, i.e., every path operator must be immediately preceded by a path quantifier. Since it is a subset of CTL*, it follows from Theorem 4.1 that states that are $\mathcal{E}^{fair*}$-equivalent must agree on all CTL formulae. However the converse is not true, as was illustrated by Clarke and Draghicescu [5] as part of their proof of the fact that the CTL$*$ property $A(FGp)$ is not expressible in CTL. We reproduce their argument below.

Consider the states s_1 and t_1 in Figure 2.

1. The set of atomic propositions is $\{a, b\}$, the set of APs true at s_1 and t_1 is $\{a\}$, the set of APs true at s_2 and t_2 is $\{b\}$.

2. The fairness conditions are of Muller type. The sets $U_1 = \{s_1\}$, $U_2 = \{s_1, s_2\}$, and $V_1 = \{t_1\}$ are the fair Muller sets, i.e., fair paths are those in which the infinitary set of states is exactly one of U_1, U_2, V_1.

State s_1 can not be differentiated from t_1 by any CTL formula, since the only difference is the fact that there are paths from s_1 on which b occurs infinitely often, and CTL can not express $\exists GF\phi$, i.e., there exists a path such that infinitely often ϕ is true. This fact was first proved by Emerson and Clarke [9]; a considerably shorter proof was presented by Clarke and Draghicescu [5]. See [8, page 1029] for a proof sketch.

More formally, the equivalence of s_1 and t_1 with respect to all CTL formula can be proved by using induction on the length of the formula.

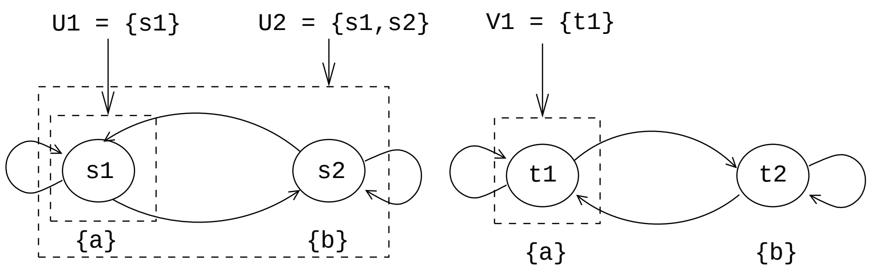

Figure 2: States that agree on all CTL formulae but can be differentiated by CTL*

It is surprising that the set of output traces from s_1 is not equal to the set of output traces from t_1; consider for example the trace $\langle a, b\rangle^\omega$. This in contrast to the fact that in the absence of fairness constraints states that are bisimilar equivalent must have the same set of traces.

We can still characterize states that agree on all CTL formula as shown in the following theorem.

Definition 4 Let K be a fair-Kripke structure $(S, T, \mathcal{L}, f)$. An equivalence relation $\mathcal{E} \subset S \times S$ is said to be a $fair$ bisimulation if for any $(s,t) \in \mathcal{E}$ the following are true:

1. $\mathcal{L}(s) = \mathcal{L}(t)$, and

2. for every fair rational path σ starting at s, and every N and k such that $\sigma = \langle s, s_1, s_2, \cdots, s_{N-1}\rangle \cdot \langle s_N, s_{N+1}, \cdots, s_{N+k-1}\rangle^\omega$ there exists a fair rational path τ starting at t such that for every $i \in \{0, 1, \ldots, N-1\}$, it is true that $\mathcal{E}([\sigma]_i, [\tau]_i)$ and for every $i \geq N$ it is true that $[\tau]_i$ is $\mathcal{E}$-equivalent to some state in $\texttt{inf}(\sigma)$, and

3. for every fair rational path τ starting at t, and every N and k such that $\tau = \langle t, t_1, \cdots, t_{N-1}\rangle \cdot \langle t_N, t_{N+1}, \cdots, t_{N+k-1}\rangle^\omega$ there exists a fair rational path σ starting at s such that for every $i \in \{0, 1, \ldots, N-1\}$, it is true that $\mathcal{E}([\tau]_i, [\sigma]_i)$ and for every $i \geq N$ it is true that $[\sigma]_i$ is $\mathcal{E}$-equivalent to some state in $\texttt{inf}(\tau)$.

The relation $\mathcal{E}^{fair}$ is defined to be the coarsest relation that is a $fair$ bisimulation.

The soundness of this definition follows in a manner analogous to that of the definition of $\mathcal{E}^{bis}$, as given in Section 4.

In the sequel, given an equivalence relation $\mathcal{E}$ on states and a fair path τ, we will say that τ $\mathcal{E}$-corresponds to a fair rational path σ if for every N and k such that $\sigma = \langle s_0, s_1, s_2, \cdots, s_{N-1}\rangle \cdot \langle s_N, s_{N+1}, \cdots, s_{N+k-1}\rangle^\omega$, for every $i \in \{0, 1, \ldots, N-1\}$, it is true that $\mathcal{E}([\sigma]_i, [\tau]_i)$ and for every $i \geq N$, we have $[\tau]_i$ is $\mathcal{E}$-equivalent to some state in $\texttt{inf}(\sigma)$

The $\mathcal{E}^{fair}$-bisimilarity relation is the natural extension to bisimulation when model checking CTL on fair-Kripke structures in the sense that

Theorem 4.5 Let s and t be states in a fair-Kripke structure $K = (S, T, \mathcal{L}, f)$. Then $\mathcal{E}^{fair}(s,t)$ if and only if there is no CTL formula ϕ such that $s \models \phi$ and $t \not\models \phi$.

Proof:

First we consider soundness, i.e., that $\mathcal{E}^{fair}(s,t) \Rightarrow$ for any CTL formula ϕ, state $s \models \phi$ if and only if state $t \models \phi$. Unlike in Theorem 4.1, this is not trivial to show, and we present the proof.

We proceed by induction on the length of the formula.

Base Case: $\phi = \underline{\text{a}}$ where a $\in$ AP. It follows from the definition of $\mathcal{E}^{fair}$ that $\mathcal{L}(s) = \mathcal{L}(t)$, and so $s \models \underline{\text{a}}$ if and only if $t \models \underline{\text{a}}$. Hence the base case is proved.

Inductive Step:

1. $\phi = \neg\phi_1$ — Follows from elementary propositional logic.

2. $\phi = (\phi_1 \vee \phi_2)$ — Follows from elementary propositional logic.

3. $\phi = \exists X(\phi_1)$ — Observe that $\mathcal{E}^{fair}(s,t)$ implies that we must have $\forall s' \Big(T(s,s') \;\Rightarrow\; \exists t'\big(T(t,t') \wedge \mathcal{E}^{fair}(s',t')\big)\Big)$. This follows from the fact that s' can be continued to a fair rational path, and so there must be a corresponding fair rational path from t. The state t' can be taken to be the state following t in this path. From this observation and the IH, it follows that $\exists s'\big(T(s,s') \wedge s' \models \phi_1\big)$ if and only if $\exists t'\big(T(t,t') \wedge t' \models \phi_1\big)$. The converse, namely to show $t \models \phi \;\Rightarrow s \models \phi$, follows by symmetry.

4. $\phi = \exists(\phi_1 U \phi_2)$ — Suppose $s \models \phi$. Then from the semantics of CTL, it follows that there exists a fair path $\sigma = \langle s, s_1, s_2, \cdots \rangle$ such that there is a natural number N for which for all natural numbers i such that $i < N$ we have $[\sigma]_i \models \phi_1$ and $[\sigma]_N \models \phi_2$. Reasoning as above, there must exist a finite sequence $\langle t, t_1, t_2, \cdots t_N \rangle$ such that for every $i \leq N$ we have $\mathcal{E}^{fair}(s_i, t_i)$. Hence by the IH, for every $i < N$ it must be the case that $t_i \models \phi_1$, and $t_N \models \phi_2$. This finite path can be extended to an infinite path, since every state is assumed to lie on a fair path, and this infinite path satisfies $\phi_1 U \phi_2$. Thus $t \models \phi$. The converse, namely to show $t \models \phi \;\Rightarrow s \models \phi$, follows by symmetry.

5. $\phi = \exists G(\phi_1)$ — Suppose $s \models \phi$. Then must be a fair rational path σ such that for each i, we have $[\sigma]_i \models \phi$. Since σ is rational, it can be expressed as a sequence $\langle s, s_1, s_2, \cdots, s_{N-1}\rangle \cdot \langle s_N, s_{N+1}, \cdots, s_{N+k-1}\rangle^\omega$. Because s is $\mathcal{E}^{fair}$-equivalent to t, it follows that there is a fair rational path τ $\mathcal{E}^{fair}$-corresponding to σ. Since $\sigma \models G(\phi_1)$, it follows that every state on σ satisfies ϕ_1. Since every state on the path τ is $\mathcal{E}^{fair}$-equivalent to some state on σ, and the induction hypothesis requires that states that are $\mathcal{E}^{fair}$-equivalent agree on all CTL formula of length less than the length of ϕ, it follows that all states on τ satisfy ϕ_1; thus $t \models \phi$. The converse, namely to show $t \models \phi \;\Rightarrow s \models \phi$, follows by symmetry.

Hence by induction, states that are E_∞-equivalent satisfy exactly the same set of CTL formula. This completes the proof of soundness.

We now show completeness, i.e., that states which are not $\mathcal{E}^{fair}$-equivalent can be differentiated by CTL. We proceed in a manner similar to that in the proof of Theorem 4.1.

First we define the series of equivalences $E_0, E_1, \ldots$ as follows:

1. $E_0(s,t)$ if and only if $\mathcal{L}(s) = \mathcal{L}(t)$.
2. $E_{l+1}(s,t)$ if and only if
   (a) for every fair rational path σ from s, there is an E_l-corresponding fair rational path τ from t, and
   (b) for every fair rational path τ from t, there is a an E_l-corresponding fair rational path σ from t.

Observe that $E_{l+1}(s,t) \subseteq E_l(s,t)$. Also every equivalence in the sequence contains the equality relation (the binary relation where an element is related only to itself). Since the state space is finite the sequence converges to a fixed point in some finite number of steps, i.e., there is some k such that $E_{k+1} = E_k$, which we will refer to as E_∞.

Notice that E_∞ satisfies Conditions 1–3 in Definition 4, and hence E_∞ must lie in $\mathcal{E}^{fair}$, since by definition, $\mathcal{E}^{fair}$ is the coarsest relation satisfying Conditions 1–3 in Definition 4. Thus if we show that states which are not E_∞-equivalent can be differentiated by CTL formulae, we will have shown that states which are not $\mathcal{E}^{fair}$-equivalent can be differentiated by CTL formulae.

We now characterize states up to E_l-equivalence by CTL formulae. This is done by induction on l. Specifically we will demonstrate:

1. If $\neg\big(E_l(s,t)\big)$ then there is a CTL formula $d_l(s,t)$ such that every state v which is E_l to s satisfies $v \models d_l(s,v)$, and $t \not\models d_l(s,t)$, and,

2. for every state $s \in S$, there is a formula of CTL $C_l(s)$ such that for every state t it is true that $t \models C_l(s)$ if and only if $E_l(s,t)$.

The formula $d_l(s,t)$ distinguishes between t and states E_l-equivalent to s and $C_l(s)$ is a formula that characterizes E_l-equivalence to state s within the fair-Kripke structure.

If we let $C_l(s)$ be a conjunction of $C_{l-1}(s)$ and $d_l(s,t)$ for every t which is not E_l-related to s, the second assertion follows immediately. Now it is necessary to show how to construct $d_l(s,t)$ by induction on l.

Base Case:(l=0)

Let $\{p_i\}$ be the set of atomic propositions in $\mathcal{L}(s)$ and $\{q_j\}$ be the set of atomic propositions in AP $-$ $\mathcal{L}(s)$. Now let $d_0(s,t) = \bigwedge_i \underline{p_i} \wedge \bigwedge_j \neg \underline{q_j}$. It is clear that this formula is only true in states with exactly the same labeling as s. Thus the base case is established.

Induction:

Assume the result is true for l. We will show it for $l+1$.

Let s and t be any states in the structure such that $\neg\big(E_{l+1}(s,t)\big)$. This can only happen if there is a fair rational path from s with no E_l-corresponding (in the sense used in Definition 4) fair rational path from t, or there is a fair rational path from t for which there is no E_l-corresponding fair rational path out of s. In the latter case, we will use the argument below to find a $d_{l+1}(t,s)$ such that $t \models d_{l+1}(t,s)$ and $s \not\models d_{l+1}(t,s)$. We can negate this formula to obtain the desired $d_{l+1}(s,t)$.

Let $\sigma = \langle s, s_1, \cdots, s_{N-1}\rangle \cdot \langle s_{N+1}, \cdots, s_{N+k}\rangle^\omega$ be a fair rational path from s with no E_l-corresponding fair rational path from t.

First define the CTL formula $\texttt{infinite}_{l+1}(s,t)$ as below:

$$\texttt{infinite}_{l+1}(s,t) \quad = \quad \exists G\Big(C_l(s_N) \vee C_l(s_{N+1}) \vee \cdots \vee C_l(s_{N+k-1})\Big)$$

Now define $d_{l+1}(s,t)$:

$$\begin{aligned} d_{l+1}(s,t) \quad = \quad & C_l(s) \wedge \exists X\Big(C_l(s_1) \wedge \exists X\Big(C_l(s_2) \wedge \cdots \wedge \exists X\Big(C_l(s_{N-1}) \wedge \\ & \exists X\big(\texttt{infinite}_{l+1}(s,t)\big)\Big) \cdots \Big) \end{aligned}$$

Proposition 4.6 A state u satisfies $d_{l+1}(s,t)$ if and only if lies at the beginning of a fair path which E_l-corresponds to σ.

Proof: The "if" part is straightforward; the E_l-corresponding fair path will provide the existential witnesses.

We now prove the "only if" part. Since u satisfies $d_{l+1}(s,t)$, the formula $C_l(s)$ must be true at u. Furthermore, there must be a successor state for which $C_l(s_1)$ is true, and $\exists X\big(C_l(s_2) \wedge \cdots \wedge \exists X\big(C_l(s_{N-1}) \wedge \exists X\big(\texttt{infinite}_{l+1}(s,t)\big) \cdots\big)$ is also true; call this state u_1. Continuing in this manner, we see that there must be a finite sequence of states $\langle u, u_1, u_2, \cdots, u_{N-1}\rangle$ which satisfy $C_l(s), C_l(s_1), \cdots, C_l(s_{N-1})$ respectively. Furthermore, u_{N-1} satisfies $\exists X\big(\texttt{infinite}_{l+1}(s,t)\big)$; hence u_{N-1} has a successor state from which there is a fair path on which every state satisfies some $C_l(s_i)$, where $i \in \{N, N+1, \ldots, N+k-1\}$; call this path υ. Consider the path $\langle u, u_1, \cdots, u_{N-1}\rangle \cdot \upsilon$. By the IH, states which satisfy $C_l(r)$ must be E_l-equivalent to r, and so the path $\langle u, u_1 \cdots u_{N-1}\rangle \cdot \upsilon$ must E_l-correspond to σ, proving the proposition. ■

Since the state t does not lie at the beginning of a fair path which E_l-corresponds to σ, it does not satisfy the formula $d_{l+1}(s,t)$. ■

4.3 $\forall$CTL*, similarity and fairness

The logic $\forall$CTL* is the subset of CTL* state formulae where negations occur only on the atomic propositions, and all path quantifiers are universal. It has been argued that this logic suffices to express most commonly encountered specifications, and that it is well suited to reasoning about compositional designs [12].

Let $K = (S, T, \mathcal{L})$ be a Kripke structure. The relation $\mathcal{F}^{sim} \subset S \times S$ is said to be a *simulation relation* if for all $(s, t) \in S \times S$ it satisfies the following:

$$\begin{aligned} \mathcal{F}^{sim}(s,t) \quad \Rightarrow \quad & \big(\mathcal{L}(s) = \mathcal{L}(t)\big) \wedge \\ & \forall t' \Big(T(t,t') \rightarrow \exists s' \big(T(s,s') \wedge \mathcal{F}^{sim}(s',t') \big) \Big) \end{aligned} \tag{2}$$

State s is said to *simulate* t when there exists a simulation relation $\mathcal{F}^{sim}$ such that $\mathcal{F}^{sim}(s, t)$; if s simulates t and t simulates s then s and t are said to be *similar*. It is straightforward to show the existence and uniqueness of a maximal equivalence relation on S where equivalent states are similar; this relation is referred to as *similarity*.

The relationship between $\forall$CTL* and similarity is completely analogous to the relationship between CTL* and bisimulation. Grumberg and Long [11] introduced the notion of a fair simulation relation for fair-Kripke structures. They proved fair simulation is sound with respect to $\forall$CTL*, i.e., if state s fair simulates state t, then all formulae in the logic $\forall$CTL* which hold at s also hold at t. Their definition of fair simulation is completely analogous to the definition of fair bisimulation given in this paper. However, they did not prove any completeness results, which are substantially more difficult to show than soundness results. The techniques in our paper can be trivially extended to demonstrate that the fair similarity relation is the weakest equivalence preserving $\forall$CTL* over fair-Kripke structures.

5 Conclusion and Future Work

We have defined state equivalences on Kripke structures that incorporate fairness. These equivalences were shown to be complete in the sense that they are the weakest equivalences preserving branching time logics interpreted on the structures. Furthermore we characterized the equivalence classes by formulae from the logic.

We have developed approximations to the complete equivalence that can be efficiently computed for Büchi, Rabin, and Streett fairness conditions. These are used in a hierarchical procedure for minimizing systems of interacting state machines [2]. We plan to continue developing generalized notions of equivalence that are property specific, and can be used to reduce or abstract components in large designs [1]. We are also developing adequate notions of bisimulation for more complex designs, such as those which include uninterpreted functions, or timing/statistical functionality.

References

[1] A. Aziz, T. Shiple, V. Singhal, and A. Sangiovanni-Vincentelli. Formula-Dependent Equivalence for Compositional CTL Model Checking. In *Computer Aided Verification*, pages 324–337, July 1994.

[2] A. Aziz, V. Singhal, G. Swamy, and R. Brayton. Minimizing Interacting Finite State Machines: A Compositional Approach to Language Containment. In *International Conference on Computer Design*, pages 255–261, October 1994.

[3] M. C. Browne, E. M. Clarke, and O. Grumberg. Characterizing Finite Kripke Structures in Propositional Temporal Logic. *Theoretical Computer Science*, 59:115–131, 1988.

[4] E. M. Clarke, J. R. Burch, O. Grumberg, D. E. Long, and K. L. McMillan. Automatic Verification of Sequential Circuit Designs. *Phil. Trans. of the Royal Society of London*, 339:105–120, 1992.

[5] E. M. Clarke and I. A. Draghicescu. Expressibility results for linear-time and branching-time logics. In *Workshop on Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency*, Noordwijkerhout, Norway, 1988.

[6] E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic Verification of Finite-State Concurrent Systems Using Temporal Logic Specifications. *ACM Transactions on Programming Languages and Systems*, 8(2):244–263, 1986.

[7] T. H. Cormen, C. E. Leiserson, and R. H. Rivest. *Introduction to Algorithms.* MIT Press, 1989.

[8] E. A. Emerson. Temporal and Modal Logic. In J. van Leeuwen, editor, *Formal Models and Semantics*, volume B of *Handbook of Theoretical Computer Science*, pages 996–1072. Elsevier Science, 1990.

[9] E. A. Emerson and E. M. Clarke. Characterizing Correctness Properties of Programs as Fixpoints. In *Proceedings of the Colloquium on Automata, Languages, and Programming*, 1981.

[10] E. A. Emerson and J. Y. Halpern. "Sometimes" and "Not Never" Revisited: on Branching versus Linear Time Temporal Logic. *Journal of the ACM*, 33(1):151–178, 1986.

[11] O. Grumberg and D. Long. Model checking and modular verification. *ACM Transactions on Programming Languages and Systems*, 16(3):843–871, 1994.

[12] O. Grumberg and D. E. Long. Model Checking and Modular Verification. In J. C. M. Baeten and J. F. Groote, editors, *Proc. of CONCUR '91: 2nd Inter. Conf. on Concurrency Theory*, volume 527 of *Lecture Notes in Computer Science*. Springer-Verlag, August 1991.

[13] J. Hopcroft and J. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.

[14] O. Kupferman and M. Vardi. Verification of Fair Transition Systems. In *Computer Aided Verification*, July 1996.

[15] R. P. Kurshan. *Automata-Theoretic Verification of Coordinating Processes*. Princeton University Press, 1993.

[16] R. Milner. *Communication and Concurrency*. Prentice Hall, New York, 1989.

[17] S. Safra. *Complexity of Automata on Infinite Objects*. PhD thesis, The Weizmann Institute of Science, Rehovot, Israel, March 1989.

[18] A. P. Sistla, M. Y. Vardi, and P. L. Wolper. The Complementation Problem for Büchi Automata, with Applications to Temporal Logic. *Theoretical Computer Science*, 49:217–237, 1987.

[19] L. J. Stockmeyer. The Complexity of Decision Problems in Automata Theory and Logic. Technical Report MAC TR-133, MIT, Cambridge MA, Project MAC, 1974.

[20] W. Thomas. Automata on Infinite Objects. In J. van Leeuwen, editor, *Formal Models and Semantics*, volume B of *Handbook of Theoretical Computer Science*, pages 133–191. Elsevier Science, 1990.

[21] R. J. van Glabbeek. *Comparative Concurrency Semantics and Refinement of Actions*. PhD thesis, Centrum voor Wiskunde en Informatica, Vrije Universiteit te Amsterdam, Amsterdam, May 1990.