Focus Games for Satisfiability and Completeness of Temporal Logic

Martin Lange   Colin Stirling
LFCS, Division of Informatics, University of Edinburgh,
JCMB, King's Buildings, Edinburgh, EH9 3JZ
{martin,cps}@dcs.ed.ac.uk

Abstract

We introduce a simple game theoretic approach to satisfiability checking of temporal logic, for LTL and CTL, which has the same complexity as using automata. The mechanisms involved are both explicit and transparent, and underpin a novel approach to developing complete axiom systems for temporal logic. The axiom systems are naturally factored into what happens locally and what happens in the limit. The completeness proofs utilise the game theoretic construction for satisfiability: if a finite set of formulas is consistent then there is a winning strategy (and therefore construction of an explicit model is avoided).

1 Introduction

The automata theoretic approach to satisfiability checking for temporal logic is very popular and successful [6, 17]. However there is a cost with the involvement of automata mechanisms and in particular the book keeping implicit in the product construction, when a local automaton is paired with an eventuality automaton. While this is not an impediment for checking satisfiability it appears to be for other formal tasks such as showing that an axiomatisation of a temporal logic is complete. When proving completeness, one needs to establish that a finite consistent set of formulas is satisfiable. It is not known, in general, how to plug into such a proof automata theoretic constructions (such as product and determinisation) for satisfiability. Instead standard completeness proofs either appeal to "canonical" structures built from maximal consistent sets [15, 8] or tableaux which explicitly build models from consistent sets, as illustrated by the delicate proofs of completeness for CTL* [14] and modal μ-calculus [18], and even the proofs of completeness for LTL [7, 13] (future linear time logic) and CTL [5] (computation tree logic).

In this paper we introduce a simple game theoretic approach to satisfiability checking of temporal logic, for LTL and CTL, which has the same complexity as using automata. The mechanism involved, the use of a "focus", is both explicit and transparent, and underpins a novel approach to developing complete axiom systems for temporal logic. The axiom systems are naturally factored into what happens locally and what happens in the limit. The completeness proofs use the game theoretic construction for satisfiability: if a finite set of formulas is consistent then there is a winning strategy (and therefore construction of an explicit model is avoided).

Although the origin of these games is model checking CTL* [12], it remains to be seen if the game technique extends to satisfiability checking of CTL* and modal μ-calculus. Moreover, it remains to be seen if the technique is practically viable for testing satisfiability of LTL and CTL.

2 LTL

We present LTL [7] in positive form, where only atomic formulas are negated. Let Prop be a family of atomic propositions closed under negation, where $\neg\neg q = q$, and containing the constants `tt` (true) and `ff` (false). Formulas of LTL are built from Prop using boolean connectives $\vee$ and $\wedge$, the unary temporal operator X (next) and the binary temporal connectives U (until) and its dual R (release).

We assume a usual ω-model for formulas, consisting of an infinite sequence of states which are maximal consistent sets of atomic formulas. A state s therefore obeys the condition that for any $q \in \mathsf{Prop}$, $q \in s$ iff $\neg q \notin s$, and `tt` $\in s$ and `ff` $\notin s$. The semantics inductively defines when an ω-sequence of states σ satisfies a formula Φ, written $\sigma \models \Phi$. In the case of $q \in \mathsf{Prop}$, $\sigma \models q$ iff q is in the initial state of σ. The clauses for the boolean connectives are as usual. If $\sigma = s_0 s_1 \ldots$ and $i \geq 0$ then $\sigma^i = s_i s_{i+1} \ldots$ is the ith suffix of σ. The remaining clauses are as follows.

$$\begin{array}{lll} \sigma \models X\Phi & \text{iff} & \sigma^1 \models \Phi \\ \sigma \models \Phi U \Psi & \text{iff} & \exists i \geq 0.\ \sigma^i \models \Psi \text{ and} \\ & & \quad \forall j : 0 \leq j < i.\ \sigma^j \models \Phi \\ \sigma \models \Phi R \Psi & \text{iff} & \forall i \geq 0.\ \sigma^i \models \Psi \text{ or} \\ & & \quad \exists j : 0 \leq j < i.\ \sigma^j \models \Phi \end{array}$$

We assume that $F\Psi$ (eventually Ψ) abbreviates $\mathtt{tt}U\Psi$ and its dual $G\Psi$ (always Ψ) abbreviates $\mathtt{ff}R\Psi$. The meanings of U and R are determined by their fixed point definitions, $\Phi U\Psi$ is the least solution to $\alpha = \Psi \vee (\Phi \wedge X\alpha)$ whereas $\Phi R\Psi$ is the largest solution of $\alpha = \Psi \wedge (\Phi \vee X\alpha)$.

A formula Φ is satisfiable if there is a model σ such that $\sigma \models \Phi$. In the naive tableau approach to deciding satisfiability, one constructs an "or" decision tree. The root is a finite set of initial formulas, and the decision question is whether their conjunction is satisfiable. Child nodes are produced by local rules on formulas. A node $\Gamma \cup \{\Phi \wedge \Psi\}$ has child $\Gamma \cup \{\Phi, \Psi\}$. A node $\Gamma \cup \{\Phi \vee \Psi\}$ has two children $\Gamma \cup \{\Phi\}$ and $\Gamma \cup \{\Psi\}$. Formulas $\Phi U\Psi$ and $\Phi R\Psi$ are replaced by their fixed point unfolding, $\Psi \vee (\Phi \wedge X(\Phi U\Psi))$ and $\Psi \wedge (\Phi \vee X(\Phi R\Psi))$. After repeated applications of these rules, a node without children has the form $\{q_1, \ldots, q_n, X\Phi_1, \ldots, X\Phi_m\}$, where each $q_i \in \mathsf{Prop}$. If the set $P = \{q_1, \ldots, q_n\}$ is unsatisfiable then the node is an unsuccessful leaf. If P is satisfiable and $m = 0$ then the node is a successful leaf. Otherwise a new child $\{\Phi_1, \ldots, \Phi_m\}$ is produced, which amounts to moving to a new state.

Nodes with until or release formulas may continually produce children, and therefore one also needs another criterion for when a node counts as a leaf. An obvious candidate is when a node is a repetition, contains the same formulas as an earlier node (and in between there is at least one application of the new state rule). Whether or not such a leaf is successful will depend on whether formulas are the result of the fixed point unfolding of a release or an until formula. A repeat of $\Phi R\Psi$ should be successful whereas a repeat of $\Phi U\Psi$ is unsuccessful.

Consider the following example decision tree, where set braces are dropped (and $\mathtt{tt}$ and $\mathtt{ff}$ are dispensed with and so the unfolding of $F\Psi$ is $\Psi \vee XF\Psi$ and the unfolding of $G\Psi$ is $\Psi \wedge XG\Psi$).

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{q, XGFq}{GFq}\ \text{Next}}{Fq \wedge XGFq}}{Fq, XGFq} \qquad \dfrac{\dfrac{\dfrac{XFq, XGFq}{Fq, GFq}\ \text{Next}}{Fq, Fq \wedge XGFq}}{Fq, XGFq}}{q \vee XFq, XGFq}}{Fq, XGFq}$$

Next labels a transition to a new state. Both leaves in this tree are repetitions of the root. However the left leaf should count as successful because the formula Fq at the initial node is "fulfilled" in the left branch, giving the model s_0^ω where $q \in s_0$. In contrast Fq is not fulfilled in the right branch and is thereby "regenerated", and therefore the right leaf should count as unsuccesful.

The problem of which fixed points are regenerated disappears in the automata theoretic approach to satisfiability [17]. Roughly speaking, the decision tree is then only part of the story. It is captured by the "local" automaton and one also needs to factor in the "eventuality" automaton which automatically deals with regeneration of fixed points, and therefore the problem does not arise. However the cost is the use of the product construction between the two automata. While this is not an impediment for checking satisfiability it appears to be for other formal tasks such as showing that an axiomatisation of a temporal logic is complete.

We now show that a simple game theoretic approach to satisfiability checking, where the mechanisms are both explicit and transparent, has the virtue that it also leads to very simple proofs of completeness for both LTL and CTL.

3 Games for LTL

In the naive tableau approach to satisfiability there are "or" choices but there are no "and" choices. Recasting as a game, "or" choices are $\exists$-choices for the player $\exists$ and "and" choices are $\forall$-choices for the player $\forall$. The role of player $\exists$ is that of verifier, "I want to show that the initial set of formulas is satisfiable" whereas the role of $\forall$ is that of refuter, "I want to show that the initial set of formulas is unsatisfiable". In a position $\Gamma, \Phi_1 \vee \Phi_2$ player $\exists$ chooses the disjunct Φ_i, and play continues from the position Γ, Φ_i. The idea is that $\exists$ ($\forall$) has a winning strategy iff the initial set of formulas is satisfiable (unsatisfiable).

We need to force player $\forall$ to make choices. A new component, the "focus", is introduced into a set of formulas for this purpose. One of the formulas in a position is in focus. We write $[\Phi], \Gamma$ to represent the position $\Gamma \cup \{\Phi\}$ when Φ is in focus. Player $\forall$ chooses which formula is in focus. If it is an "and" formula then $\forall$ chooses which subformula to keep in focus. During a play $\forall$ may also change mind, and move the focus to a different formula.

Given a starting formula Φ_0 (the conjunction of the initial formulas) we will define its focus game $\mathsf{G}(\Phi_0)$. The set of subformulas of Φ_0, Sub(Φ_0), is defined as expected but with the requirement that the unfolding of an until $\Psi \vee (\Phi \wedge X(\Phi U\Psi))$ is a subformula of $\Phi U\Psi$ and the unfolding of a release $\Psi \wedge (\Phi \vee X(\Phi R\Psi))$ is a subformula of $\Phi R\Psi$. A position in a play of $\mathsf{G}(\Phi_0)$ is an element $[\Phi], \Gamma$ where $\Phi \in \mathrm{Sub}(\Phi_0)$ and $\Gamma \subseteq \mathrm{Sub}(\Phi_0) - \{\Phi\}$. A play of the game $\mathsf{G}(\Phi_0)$ is a sequence of positions $P_0 P_1 \ldots P_n$ where P_0 is the initial position $[\Phi_0]$, and the change in position P_i to P_{i+1} is determined by one of the moves of Figure 1. They are divided into three groups. First are rules for $\exists$ who chooses disjuncts in and out of focus. Second are the moves for player $\forall$ who chooses which conjunct remains in focus and who also can change focus with the rule change. Finally, there are the remaining moves which do not involve

Player $\exists$

$$\frac{[\Phi_0 \vee \Phi_1], \Gamma}{[\Phi_i], \Gamma} \qquad \frac{[\Phi], \Phi_0 \vee \Phi_1, \Gamma}{[\Phi], \Phi_i, \Gamma}$$

Player $\forall$

$$\frac{[\Phi_0 \wedge \Phi_1], \Gamma}{[\Phi_i], \Phi_{1-i}, \Gamma} \qquad \frac{[\Phi], \Psi, \Gamma}{[\Psi], \Phi, \Gamma}\ \text{change}$$

Other moves

$$\frac{[\Phi U \Psi], \Gamma}{[\Psi \vee (\Phi \wedge X(\Phi U \Psi))], \Gamma} \qquad \frac{[\Phi'], \Phi U \Psi, \Gamma}{[\Phi'], \Psi \vee (\Phi \wedge X(\Phi U \Psi)), \Gamma}$$

$$\frac{[\Phi R \Psi], \Gamma}{[\Psi \wedge (\Phi \vee X(\Phi R \Psi))], \Gamma} \qquad \frac{[\Phi'], \Phi R \Psi, \Gamma}{[\Phi'], \Psi \wedge (\Phi \vee X(\Phi R \Psi)), \Gamma}$$

$$\frac{[\Phi], \Phi_0 \wedge \Phi_1, \Gamma}{[\Phi], \Phi_0, \Phi_1, \Gamma} \qquad \frac{[X\Phi_1], \ldots, X\Phi_m, q_1, \ldots, q_n}{[\Phi_1], \ldots, \Phi_m}\ \text{next}$$

Figure 1. Game moves

any choices, and so neither player is responsible for them. These include the fixed point unfolding of until and release in and out of focus, the removal of $\wedge$ out of focus and the next state rule, next, where the focus remains with the subformula of the next formula in focus. It is therefore incumbent on $\forall$ to make sure that an X formula is in focus when next is applied.

The next ingredient in the definition of the game is the winning conditions for a player, when a play counts as a win.

Definition 1 Player $\forall$ wins the play $P_0, \ldots, P_n$ if

1. P_n is $[q], \Gamma$ and (q is `ff` or $\neg q \in \Gamma$) or
2. P_n is $[\Phi U \Psi], \Gamma$ and for some $i < n$ the position P_i is $[\Phi U \Psi], \Gamma$ and between $P_i \ldots P_n$ player $\forall$ has not applied the rule change.

Therefore $\forall$ wins if there is a simple contradiction or a repeat position with the same until formula in focus and no application of change between the repeats.

Definition 2 Player $\exists$ wins the play $P_0, \ldots, P_n$ if

1. P_n is $[q_1], \ldots, q_n$ and $\{q_1, \ldots, q_n\}$ is satisfiable or
2. P_n is $[\Phi R \Psi], \Gamma$ and for some $i < n$ the position P_i is $[\Phi R \Psi], \Gamma$ or
3. P_n is $[\Phi], \Gamma$ and for some $i < n$ the position P_i is $[\Phi], \Gamma$ and between $P_i \ldots P_n$ player $\forall$ has applied the rule change.

So $\exists$ wins if player $\forall$ is unable to focus on a X formula so that next can be applied when the atomic formulas are satisfiable. The other two conditions cover repeat positions. First is the case if the repeat position has the same release formula in focus, and second is the case of a repeat when the same formula is in focus and change has been applied between the repeat positions. The following upper bound on the length of a play is obvious.

Fact 1 *Every play of* $\mathsf{G}(\Phi_0)$ *has finite length less than* $|\mathrm{Sub}(\Phi_0)| \times 2^{|\mathrm{Sub}(\Phi_0)|}$.

A player wins the game $\mathsf{G}(\Phi_0)$ if the player is able to win every play of the game, that is has a winning strategy[1]. The following is a simple consequence of Fact 1 and the fact that the winning conditions are mutually exclusive.

Fact 2 *Every game* $\mathsf{G}(\Phi_0)$ *has a unique winner*.

Next we come to the game characterisation of satisfiability, which we split into two halves.

Proposition 1 *If* $\exists$ *wins the game* $\mathsf{G}(\Phi_0)$ *then* Φ_0 *is satisfiable*.

Proof: Assume $\exists$ wins the game $\mathsf{G}(\Phi_0)$. Consider the play where $\forall$ uses the following optimal strategy. Let $\Phi_1 U \Psi_1 \ldots, \Phi_n U \Psi_n$ be a priority list of all until subformulas of Φ_0, in decreasing order of size. We say that $\Phi U \Psi$ is present in a position P if either $\Phi U \Psi \in P$ or $\Psi \vee (\Phi \wedge X(\Phi U \Psi)) \in P$ or $X(\Phi U \Psi) \in P$. Player $\forall$ starts with the focus on Φ_0. If the formula in focus is a release formula $\Phi R \Psi$ and Ψ contains an until subformula then $\forall$ chooses Ψ when the release formula is unfolded. If the formula is a conjunction then $\forall$ chooses a conjunct with an until subformula. If the focus remains on a release formula or ends up on a member of Prop then $\forall$ changes focus, if this is possible, to the until formula which is present in the position and which is earliest in the priority list. If the focus is on an until formula $\Phi_i U \Psi_i$ then $\forall$ keeps the focus on it until it is "fulfilled", that is until player $\exists$ chooses Ψ_i when it is unfolded. This until formula is then moved to the end of the priority list. Player $\forall$ then changes focus to the earliest until formula in the priority list which is present in the position, if this is possible. This argument is then repeated. By assumption player $\exists$ wins against this strategy, and the play has finite length. It is now straightforward to extract an eventually cyclic model from the play, where every until formula present in some position will be fulfilled. $\square$

Next we prove the converse of Proposition 1. One proof is to show how a winning strategy for $\exists$ can be extracted

[1] Formally a winning strategy, see for example [9], for player $\exists$ is a set of rules π of the form, if the play so far is $P_0 \ldots P_n$ and P_n is $[\Phi_0 \vee \Phi_1], \Gamma$ ($[\Phi], \Phi_0 \vee \Phi_1, \Gamma$) then choose $[\Phi_i], \Gamma$ ($[\Phi], \Phi_i, \Gamma$). Similarly for player $\forall$. A play obeys π if all the moves played by the player obey the rules in π. A strategy π is winning for a player if she wins every play in which she uses π.

from a model of Φ_0. However we provide an alternative proof which is the key to obtaining a complete axiom system. We utilise an observation from fixed point logics about least fixed points. Given Park's fixed point induction principle (1) below and that a fixed point is equivalent to its unfolding (2), Lemma 1 below holds (as observed by a number of researchers, for instance [10, 15, 19]). Standard substitution is assumed, $\Psi\{\Phi/Y\}$ is the replacement of all free occurrences of Y in Ψ with Φ. Moreover we write $\models \Phi$ to mean Φ is valid (true everywhere in all models).

$$(1) \qquad \text{if } \models \Psi\{\Phi/Y\} \to \Phi \text{ then } \models \mu Y.\, \Psi \to \Phi$$
$$(2) \qquad \models \mu Y.\, \Psi \leftrightarrow \Psi\{\mu Y.\, \Psi/Y\}$$

Lemma 1 *If Y is not free in Φ and $\Phi \wedge \mu Y.\, \Psi$ is satisfiable then the formula $\Phi \wedge \Psi\{(\mu Y.\, \neg\Phi \wedge \Psi)/Y\}$ is satisfiable.*

Proof: Suppose $\Phi \wedge \mu Y.\, \Psi$ is satisfiable, but $\models \Psi\{(\mu Y.\, \neg\Phi \wedge \Psi)/Y\} \to \neg\Phi$. Therefore $\models \Psi\{(\mu Y.\, \neg\Phi \wedge \Psi)/Y\} \to \neg\Phi \wedge \Psi\{(\mu Y.\, \neg\Phi \wedge \Psi)/Y\}$. Hence by (2) $\models \Psi\{(\mu Y.\, \neg\Phi \wedge \Psi)/Y\} \to \mu Y.\, \neg\Phi \wedge \Psi$ and so by (1) $\models \mu Y.\, \Psi \to \neg\Phi$ which contradicts that $\Phi \wedge \mu Y.\, \Psi$ is satisfiable. □

Lemma 1 sanctions the following property of until unfolding.

Lemma 2 *If $\Phi' \wedge (\Phi U \Psi)$ is satisfiable then $\Phi' \wedge (\Psi \vee (\Phi \wedge X((\Phi \wedge \neg\Phi')U(\Psi \wedge \neg\Phi'))))$ is satisfiable.*

Proof: Assume $\Phi' \wedge (\Phi U \Psi)$ is satisfiable. So there is a model σ such that $\sigma \models \Phi'$ and $\sigma \models \Phi U \Psi$, and therefore $\sigma^i \models \Psi$ and $\sigma^j \models \Phi$ for $j : 0 \leq j < i$, for some $i \geq 0$. Also assume $\Phi' \wedge (\Psi \vee (\Phi \wedge X((\Phi \wedge \neg\Phi')U(\Psi \wedge \neg\Phi'))))$ is not satisfiable, and so the following validity holds $\models \Phi' \to (\neg\Psi \wedge (\neg\Phi \vee X((\neg\Phi \vee \Phi')R(\neg\Psi \vee \Phi'))))$. Because $\sigma \models \Phi'$ therefore $\sigma \models \neg\Psi \wedge (\neg\Phi \vee X((\neg\Phi \vee \Phi')R(\neg\Psi \vee \Phi')))$. So $\sigma \models \neg\Psi$ and because $\sigma \models \Phi U \Psi$ it follows that $\sigma \models \Phi$. And so $\sigma \models X((\neg\Phi \vee \Phi')R(\neg\Psi \vee \Phi'))$, and therefore $\sigma^1 \models (\neg\Phi \vee \Phi')R(\neg\Psi \vee \Phi')$. And so $\sigma^1 \models \neg\Psi \vee \Phi'$ and $\sigma^1 \models \neg\Phi \vee \Phi' \vee X((\neg\Phi \vee \Phi')R(\neg\Psi \vee \Phi'))$. If $\sigma^1 \models \Phi'$ then $\sigma^1 \models \neg\Psi$ by the valid formula above, and so $\sigma^1 \models \neg\Psi$ and because $\sigma^1 \models \Phi U \Psi$ it follows that $\sigma^1 \models \Phi$, and so $\sigma^1 \models X((\neg\Phi \vee \Phi')R(\neg\Psi \vee \Phi'))$. The argument is now repeated for subsequent σ^j, $j \geq 0$, which contradicts that $\sigma \models \Phi U \Psi$. □

Proposition 2 *If Φ_0 is satisfiable then player $\exists$ wins the game $\mathsf{G}(\Phi_0)$.*

Proof: Assume that Φ_0 is satisfiable. We show that player $\exists$ wins the game $\mathsf{G}(\Phi_0)$. The idea is that $\exists$ always chooses a move which preserves satisfiability (and $\forall$ has to choose moves which preserve satisfiability). If $\Gamma \wedge (\Phi_0 \vee \Phi_1)$ is satisfiable then $\Gamma \wedge \Phi_i$ is satisfiable for at least one $i \in \{0, 1\}$, and so player $\exists$ chooses such an i. If the position is $[\Phi U \Psi], \Gamma$ where the until formula is in focus then player $\exists$ adorns the interpretation of it when it is unfolded, $[\Psi \vee (\Phi \wedge X(\Phi_{\neg\Gamma} U \Psi_{\neg\Gamma}))], \Gamma$ where $\Phi_{\neg\Gamma}$ and $\Psi_{\neg\Gamma}$ are to be understood as $\Phi \wedge \neg \bigwedge \Gamma$ and $\Psi \wedge \neg \bigwedge \Gamma$. This adornment, which is justified by Lemma 2, is repeated as long as the until formula is in focus. Whenever $\forall$ changes mind, an adorned until subformula $\Phi_{\neg\Gamma_1 \wedge \ldots \wedge \neg\Gamma_n} U \Psi_{\neg\Gamma_1 \wedge \ldots \wedge \neg\Gamma_n}$ loses its adornment and is returned to its intended interpretation $\Phi U \Psi$. Now it is easy to see that $\forall$ can never win. Condition 1 of the winning condition for $\forall$ can not be reached because $\exists$ preserves satisfiability. And condition 2, the repeat position, cannot occur because $\models \Phi_{\neg\Gamma_1 \wedge \ldots \wedge \neg\Gamma_n} U \Psi_{\neg\Gamma_1 \wedge \ldots \wedge \neg\Gamma_n} \to \neg \bigwedge \Gamma_i$. □

Proposition 3 *The complexity of deciding the winner of $\mathsf{G}(\Phi_0)$ is in PSPACE.*

Proof: Consider the tree of all plays in $\mathsf{G}(\Phi_0)$ where the position of the focus is completely determined by the strategy described in the proof of Proposition 1, above. Player $\exists$ wins $\mathsf{G}(\Phi_0)$ iff there exists a path in this tree such that $\exists$ wins the play of this path. An algorithm P can nondeterministically choose this path. The required space is polynomial in the size of the input. P only has to store a counter and two configurations: the actual one which gets overwritten every time a new game rule is applied, and the one which is repeated in case $\exists$ wins the play with her winning condition 2 or 3. The latter can be chosen nondeterministically, too, and gets deleted every time the rule change is applied. The counter is needed to terminate the algorithm if it did not find a repeat after $|\mathrm{Sub}(\Phi_0)| * 2^{|\mathrm{Sub}(\Phi_0)|}$ configurations. Notice that the size of the counter also is polynomial in the length of the input $|\Phi_0|$. Hence by Savitch's Theorem the problem can be solved in PSPACE. □

4 A complete axiomatisation for LTL

The game theoretic characterisation of satisfiability offers a simple basis for extracting a complete axiom system for LTL. Given an axiom system A a formula Φ is A-consistent if $\mathsf{A} \not\vdash \neg\Phi$. The axiom system A is complete provided that for any Φ if Φ is A-consistent then Φ has a model. In this framework this becomes

(*) if Φ is A-consistent then $\exists$ wins the game $\mathsf{G}(\Phi)$.

The axiom system A for LTL is presented in Figure 2. The axioms and rules were developed with the proof of $(*)$ in mind. Axioms 1-6 and the rules MP and XGen provide "local" justifications for the rules of the focus game for LTL, and axiom 7 and the rule Rel capture $\exists$'s winning strategy.

Theorem 1 *The axiom system A is sound and complete for LTL.*

Axioms

1. any tautology instance
2. $\Phi U \Psi \to \Psi \vee (\Phi \wedge X(\Phi U \Psi))$
3. $\Phi R \Psi \to \Psi \wedge (\Phi \vee X(\Phi R \Psi))$
4. $X \neg \Phi \leftrightarrow \neg X \Phi$
5. $X\Phi \wedge X\Psi \to X(\Phi \wedge \Psi)$
6. $X(\Phi \to \Psi) \to X\Phi \to X\Psi$
7. $\neg(\Phi R \Psi) \leftrightarrow \neg\Phi U \neg\Psi$

Rules

MP if $\vdash \Phi$ and $\vdash \Phi \to \Psi$ then $\vdash \Psi$

XGen if $\vdash \Phi$ then $\vdash X\Phi$

Rel if $\vdash \Phi' \to (\Psi \wedge (\Phi \vee X((\Phi \vee \Phi')R(\Psi \vee \Phi'))))$
then $\vdash \Phi' \to (\Phi R \Psi)$

Figure 2. The axiom system A

Proof: Soundness of A is straightforward. Each axiom is valid and each rule preserves validity. The interesting case is the rule Rel, whose soundness was proved in lemma 2 of the previous section. For completeness of A we establish $(*)$, if Φ_0 is A-consistent then $\exists$ wins the game $\mathsf{G}(\Phi_0)$. The proof is similar to Proposition 2 of the previous section. Given a finite A-consistent set of LTL formulas we show that any player $\forall$ move or other move in Figure 1 preserves A-consistency, and that player $\exists$ can preserve A-consistency when she moves. If $\Gamma, \Phi_1 \vee \Phi_2$ is A-consistent then Γ, Φ_i is A-consistent for some i by axiom 1, and the rule MP. Axioms 2 and 3 are needed for the fixed point unfolding moves. Axioms 4-6 and rule XGen are required for the next move. If $\Phi_1, \ldots, \Phi_m$ is not A-consistent then A $\vdash \Phi_1 \wedge \ldots \wedge \Phi_{m-1} \to \neg\Phi_m$ and so A $\vdash X\Phi_1 \wedge \ldots \wedge X\Phi_{m-1} \to \neg X\Phi_m$ using XGen and axioms 6, 5 and one half of 4. Finally rule Rel is used to capture $\exists$'s winning strategy. If the position is $[\Phi U \Psi], \Gamma$ and $\Gamma, \Phi U \Psi$ is A-consistent then by rule Rel, the other half of axiom 4 and axiom 7 $\Gamma, \Psi \vee (\Phi \wedge X(\Phi_{\neg\Gamma} U \Psi_{\neg\Gamma}))$ is A-consistent. □

In [7] soundness and completeness of the following axiom system DUX for LTL is proved using maximal consistent sets of formulas[2].

[2]A4, A5 and U2 as presented here differ slightly from their original form which is due to the different semantics of the G and U operator used there.

A1. $\mathtt{ff}R(\Phi \to \Psi) \to (\mathtt{ff}R\Phi \to \mathtt{ff}R\Psi)$

A2. $X(\neg\Phi) \leftrightarrow \neg X\Phi$

A3. $X(\Phi \to \Psi) \to (X\Phi \to X\Psi)$

A4. $\mathtt{ff}R\Phi \to \Phi \wedge X(\mathtt{ff}R\Phi)$

A5. $\mathtt{ff}R(\Phi \wedge X\Phi) \to (\Phi \to \mathtt{ff}R\Phi)$

U1. $\Phi U \Psi \to F\Psi$

U2. $\Phi U \Psi \leftrightarrow \Psi \vee (\Phi \wedge X(\Phi U \Psi))$

R1. any tautology instance

R2. if $\vdash \Phi$ and $\vdash \Phi \to \Psi$ then $\vdash \Psi$

R3. if $\vdash \Psi$ then $\vdash \mathtt{ff}R\Psi$

Soundness of DUX and completeness of A ensure that, if DUX $\vdash \Phi$ then A $\vdash \Phi$. However, it is also interesting to compare the two axiomatisations in details.

Axioms and rules A2, A3, U2, R1 and R2 are present in A. A4 is an instance of axioms 3 and U1 simply reflects an abbreviation. R3 can be simulated in A as follows. Suppose there is a proof using R3. Then there is a shorter proof of Ψ in DUX for which by hypothesis there is an A-proof, too. Instantiate Rel with $\Phi' = \mathtt{tt}$ and $\Phi = \mathtt{ff}$. This proves $\vdash \mathtt{ff}R\Psi$ if $\vdash \Psi \wedge X\mathtt{tt}$ is provable. But this can be done using the hypothesis, axiom 1 and rule XGen.

The remaining axioms A1 and A5 are more complicated to prove in A. A simple way is to show that $\forall$ wins the focus game on the negations of these axioms. The game rules and winning conditions resemble the axioms and rules of A which are needed for the proof. We show this for A5. The negation of this axiom is $\Phi \wedge (\mathtt{ff}R(\Phi \wedge X\Phi)) \wedge (\mathtt{tt}U\neg\Phi)$. Let $\Phi' = \Phi \wedge (\mathtt{ff}R(\Phi \wedge X\Phi))$.

$$\begin{array}{c}
\Phi, \mathtt{ff}R(\Phi \wedge X\Phi), [\mathtt{tt}U\neg\Phi] \\
\hline
\Phi, X\Phi, X(\mathtt{ff}R(\Phi \wedge X\Phi)), [\neg\Phi \vee X(\mathtt{tt}_{\neg\Phi'}U\neg\Phi_{\neg\Phi'}] \\
\hline
\Phi, X\Phi, X(\mathtt{ff}R(\Phi \wedge X\Phi)), [X(\mathtt{tt}_{\neg\Phi'}U\neg\Phi_{\neg\Phi'}] \\
\hline
\Phi, \mathtt{ff}R(\Phi \wedge X\Phi), [\mathtt{tt}_{\neg\Phi'}U\neg\Phi_{\neg\Phi'}]
\end{array}$$

The game rules used are the unfolding of R, the adorned unfolding of U, the disjunctive choice and the next rule. Player $\forall$ wins with winning condition 2. Therefore the axioms and rules needed to prove A5 are 1 and MP (for $\vee$), 2 and 3 (for the unfoldings), 4 – 6, XGen (for next), 7 (to reason about the negation of A5), and Rel to describe the winning condition.

5 CTL

In this section we define focus games for CTL. Again we present CTL in positive form. Formulas of CTL are built from Prop, the boolean connectives $\vee$ and $\wedge$, the two unary temporal operators QX and the four binary temporal operators $Q(\ldots U \ldots)$, $Q(\ldots R \ldots)$ where $Q \in \{E, A\}$. E is the "some paths" quantifier and A is the "for all paths" quantifier.

A Kripke model for CTL formulas consists of a set of states S, a binary transition relation R which is total (for all $s \in \mathsf{S}$ there is a $t \in \mathsf{S}$ such that $s\mathsf{R}t$) and a valuation which assigns to each state $s \in \mathsf{S}$ a maximal consistent set of atomic formulas in Prop. The semantics defines when a state s satisfies a formula Φ, $s \models \Phi$, and it appeals to full paths from a state s_0 which is an ω-sequence of states $s_0 s_1 \ldots$ such that $s_i \mathsf{R} s_{i+1}$ for each $i \geq 0$. In the case of $q \in$ Prop, $s \models q$ iff q belongs to the valuation of s. The clauses for the boolean connectives are as usual. The remaining clauses are as follows.

$s \models EX\Phi$	iff	$\exists t.\ s\mathsf{R}t$ and $t \models \Phi$
$s \models AX\Phi$	iff	$\forall t.$ if $s\mathsf{R}t$ then $t \models \Phi$
$s_0 \models E(\Phi U \Psi)$	iff	$\exists$ full path $s_0 s_1 \ldots\ \exists i \geq 0.\ s_i \models \Psi$ and $\forall j : 0 \leq j < i.\ s_j \models \Phi$
$s_0 \models A(\Phi U \Psi)$	iff	$\forall$ full paths $s_0 s_1 \ldots\ \exists i \geq 0.\ s_i \models \Psi$ and $\forall j : 0 \leq j < i.\ s_j \models \Phi$
$s_0 \models E(\Phi R \Psi)$	iff	$\exists$ full path $s_0 s_1 \ldots\ \forall i \geq 0.\ s_i \models \Psi$ or $\exists j : 0 \leq j < i.\ s_j \models \Phi$
$s_0 \models A(\Phi U \Psi)$	iff	$\forall$ full paths $s_0 s_1 \ldots\ \forall i \geq 0.\ s_i \models \Psi$ or $\exists j : 0 \leq j < i.\ s_j \models \Phi$

The semantics of until and release formulas are determined by their fixed point definitions. $Q(\Phi U \Psi)$ is the least solution to $\alpha = \Psi \vee (\Phi \wedge QX\alpha)$ and $Q(\Phi R \Psi)$ is the largest solution to $\alpha = \Psi \wedge (\Phi \vee QX\alpha)$.

We now define the focus game $\mathsf{G}'(\Phi_0)$ for a CTL formula Φ_0. As with the LTL game, a position in a play of $\mathsf{G}'(\Phi_0)$ is an element $[\Phi], \Gamma$ where $\Phi \in \mathrm{Sub}(\Phi_0)$ and $\Gamma \subseteq \mathrm{Sub}(\Phi_0) - \{\Phi\}$, and a play is a sequence of positions $P_0 P_1 \ldots P_n$ where P_0 is the initial position $[\Phi_0]$. The change in position P_i to P_{i+1} is determined by one of the moves of Figure 3. Again they are divided into three groups. First are rules for $\exists$ who chooses disjuncts in and out of focus. Second are the moves for player $\forall$ who chooses which conjunct remains in focus and who also can change focus with the rule change. Player $\forall$ also chooses the next state when an AX formula is in focus, by choosing a single $EX\Psi_j$, if there is one: we include here the case where $l = 0$ and $\forall$ does not have any choice. Finally, there are the remaining moves which do not involve any choices, and so neither player is responsible for them. These include the fixed point unfolding of until and release in and out of focus, the removal of $\wedge$ out of focus and the next state rule

Player $\exists$

$$\frac{[\Phi_0 \vee \Phi_1], \Gamma}{[\Phi_i], \Gamma} \qquad \frac{[\Phi], \Phi_0 \vee \Phi_1, \Gamma}{[\Phi], \Phi_i, \Gamma}$$

Player $\forall$

$$\frac{[\Phi_0 \wedge \Phi_1], \Gamma}{[\Phi_i], \Phi_{1-i}, \Gamma} \qquad \frac{[\Phi], \Psi, \Gamma}{[\Psi], \Phi, \Gamma}\ \mathrm{change}$$

$$\frac{[AX\Phi_1], \ldots, AX\Phi_n, EX\Psi_1, \ldots EX\Psi_l, q_1, \ldots, q_m}{[\Phi_1], \ldots, \Phi_n, \Psi_j}\ \mathrm{next}$$

Other moves

$$\frac{[Q(\Phi U \Psi)], \Gamma}{[\Psi \vee (\Phi \wedge QXQ(\Phi U \Psi))], \Gamma}$$

$$\frac{[\Phi'], Q(\Phi U \Psi), \Gamma}{[\Phi'], \Psi \vee (\Phi \wedge QXQ(\Phi U \Psi)), \Gamma}$$

$$\frac{[Q(\Phi R \Psi)], \Gamma}{[\Psi \wedge (\Phi \vee QXQ(\Phi R \Psi))], \Gamma}$$

$$\frac{[\Phi'], Q(\Phi R \Psi), \Gamma}{[\Phi'], \Psi \wedge (\Phi \vee QXQ(\Phi R \Psi)), \Gamma}$$

$$\frac{[\Phi], \Phi_0 \wedge \Phi_1, \Gamma}{[\Phi], \Phi_0, \Phi_1, \Gamma}$$

$$\frac{[EX\Psi_1], \ldots, EX\Psi_l, AX\Phi_1, \ldots, AX\Phi_n, q_1, \ldots, q_m}{[\Psi_1], \Phi_1 \ldots, \Phi_n}\ \mathrm{next}$$

Figure 3. CTL Game moves

when an EX formula is in focus. The winning conditions for a player are almost identical to the LTL game.

Definition 1 Player $\forall$ wins the play $P_0, \ldots, P_n$ if

1. P_n is $[q], \Gamma$ and (q is $\mathtt{ff}$ or $\neg q \in \Gamma$) or
2. P_n is $[Q(\Phi U \Psi)], \Gamma$ and for some $i < n$ the position P_i is $[Q(\Phi U \Psi)], \Gamma$ and between $P_i \ldots P_n$ player $\forall$ has not applied the rule change.

Definition 2 Player $\exists$ wins the play $P_0, \ldots, P_n$ if

1. P_n is $[q_1], \ldots, q_n$ and $\{q_1, \ldots, q_n\}$ is satisfiable or
2. P_n is $[Q(\Phi R \Psi)], \Gamma$ and for some $i < n$ the position P_i is $[Q(\Phi R \Psi)], \Gamma$ or
3. P_n is $[\Phi], \Gamma$ and for some $i < n$ the position P_i is $[\Phi], \Gamma$ and between $P_i \ldots P_n$ player $\forall$ has applied the rule change.

Facts 1 and 2 of Section 3 also hold for CTL games. A main result is again the game characterisation of satisfiability.

Proposition 1 *$\exists$ wins the game $\mathsf{G}'(\Phi_0)$ iff Φ_0 is satisfiable.*

Proof: Assume $\exists$ wins the game $\mathsf{G}'(\Phi_0)$. The proof is similar to that of Proposition 1 of Section 3, except that all "next" state choices are examined, and so we have a tree of plays instead of a single play. Let $Q_1(\Phi'_1 U \Psi'_1), \ldots, Q_n(\Phi'_n U \Psi'_n)$ be an initial priority list of all until subformulas of Φ_0 in order of decreasing size. Each play in the tree of plays has its own associated current priority list. Player $\forall$ starts with the focus on Φ_0. Once the focus is on an until formula, $Q_i(\Phi'_i U \Psi'_i)$, player $\forall$ keeps the focus on it until it is fulfilled (player $\exists$ chooses Ψ'_i) or there is branching. At an application of next a play splits into all choices, each with its own priority list. If the focus is on a formula $AX\Phi_1$ then it will be on Φ_1 in all these plays and they each have the same priority list. If the position is $[EX\Psi_1], \ldots, EX\Psi_l, AX\Phi_1, \ldots, AX\Phi_n, q_1, \ldots, q_m$ and l is the current priority list then the focus remains on Ψ_1 in the play with this subformula with list l. Otherwise for each $i > 1$ there is the play where $\forall$ changes focus for the position $\Psi_i, \Phi_1, \ldots, \Phi_n$. If Ψ_1 is $E(\Phi'_j U \Psi'_j)$ then this formula is moved to the end of the priority list l_i and $\forall$ chooses as focus the earliest until formula in l_i present in the position $EX\Psi_i, AX\Phi_1, \ldots, AX\Phi_n$, if this is possible. This argument is repeated. By assumption player $\exists$ wins the finite tree of plays. It is now straightforward to read off a Kripke model from this tree of plays where Φ_0 is true at the initial state.

For the converse assume that Φ_0 is satisfiable. We show that $\exists$ has a winning strategy for the game $\mathsf{G}'(\Phi_0)$. We use the fact that for each $Q \in \{A, E\}$ if $\Phi' \wedge Q(\Phi U \Psi)$ is satisfiable then $\Phi' \wedge (\Psi \vee (\Phi \wedge QXQ(\Phi \wedge \neg\Phi' U \Psi \wedge \neg\Phi')))$ is satisfiable. So the interpretation of $Q(\Phi U \Psi)$ can be adorned whenever it is unfolded in focus as with Proposition 2 of Section 3. □

One important difference with LTL is the complexity of checking the winner of a game $\mathsf{G}'(\Phi_0)$, because of branching choices for $\forall$.

Proposition 2 *The complexity of deciding the winner of $\mathsf{G}'(\Phi_0)$ is in EXPTIME.*

Proof: The proof is very similar to that of Proposition 3 of Section 3. However, the tree of all plays is now an and-or tree because of player $\forall$'s choices using rule next. Therefore the polynomial space algorithm deciding the winner of $\mathsf{G}'(\Phi_0)$ is alternating instead of nondeterministic. By [3] the problem is therefore in EXPTIME. □

6 A complete axiomatisation for CTL

The game theoretic characterisation of CTL satisfiability also allows one to extract a sound and complete axiom system for CTL, the system B in Figure 4.

Theorem 1 *The axiom system B is sound and complete for CTL.*

Proof: Soundness of B is straightforward. The most interesting cases are soundness of ARel and ERel rules, and in the case of ERel the rule captures "limit closure". For completeness of B, the proof is similar to Theorem 1 of Section 4. If Φ_0 is B-consistent then player $\exists$ wins the game $\mathsf{G}'(\Phi_0)$. Given a finite B-consistent set of formulas, any move by player $\forall$ or other move in Figure 1 preserves B-consistency. The important cases are the next state rules. Assume $\Phi_1, \ldots, \Phi_n, \Psi_j$ is not B-consistent, and so $\mathsf{B} \vdash \Phi_1 \wedge \ldots \wedge \Phi_n \rightarrow \neg\Psi_j$. So by AXGen and axioms 9,8 and 6 $\mathsf{B} \vdash AX\Phi_1 \wedge \ldots \wedge AX\Phi_n \rightarrow \neg EX\Psi_j$ (and using 7 instead of 6 one deals with the case when $l = 0$). Finally the ARel and ERel rules are used to capture $\exists$'s winning strategy. □

In [5] soundness and completeness of the following axiom system for CTL is proved using tableaux.

Ax1. any tautology instance

Ax2. $EF\Phi \leftrightarrow E(\mathtt{tt} U \Phi)$

Ax3. $AF\Phi \leftrightarrow A(\mathtt{tt} U \Phi)$

Ax4. $EX(\Phi \vee \Psi) \leftrightarrow EX\Phi \vee EX\Psi$

Ax5. $AX\Phi \leftrightarrow \neg EX \neg\Phi$

Ax6. $E(\Phi U \Psi) \leftrightarrow \Psi \vee (\Phi \wedge EXE(\Phi U \Psi))$

Axioms

1. any tautology instance
2. $E(\Phi U \Psi) \rightarrow \Psi \vee (\Phi \wedge EXE(\Phi U \Psi))$
3. $A(\Phi U \Psi) \rightarrow \Psi \vee (\Phi \wedge AXA(\Phi U \Psi))$
4. $E(\Phi R \Psi) \rightarrow \Psi \wedge (\Phi \vee EXE(\Phi R \Psi))$
5. $A(\Phi R \Psi) \rightarrow \Psi \wedge (\Phi \vee AXA(\Phi R \Psi))$
6. $AX\neg\Phi \leftrightarrow \neg EX\Phi$
7. $AX\neg\Phi \rightarrow \neg AX\Phi$
8. $AX\Phi \wedge AX\Psi \rightarrow AX(\Phi \wedge \Psi)$
9. $AX(\Phi \rightarrow \Psi) \rightarrow AX\Phi \rightarrow AX\Psi$
10. $\neg A(\Phi R \Psi) \leftrightarrow E(\neg\Phi U \neg\Psi)$
11. $\neg E(\Phi R \Psi) \leftrightarrow A(\neg\Phi U \neg\Psi)$

Rules

MP if $\vdash \Phi$ and $\vdash \Phi \rightarrow \Psi$ then $\vdash \Psi$

AXGen if $\vdash \Phi$ then $\vdash AX\Phi$

ERel if $\vdash \Phi' \rightarrow (\Psi \wedge (\Phi \vee EXE((\Phi \vee \Phi')R(\Psi \vee \Phi'))))$
then $\vdash \Phi' \rightarrow E(\Phi R \Psi)$

ARel if $\vdash \Phi' \rightarrow (\Psi \wedge (\Phi \vee AXA((\Phi \vee \Phi')R(\Psi \vee \Phi'))))$
then $\vdash \Phi' \rightarrow A(\Phi R \Psi)$

Figure 4. The axiom system B

Ax7. $A(\Phi U \Psi) \leftrightarrow \Psi \vee (\Phi \wedge AXA(\Phi U \Psi))$

Ax8. $EX\mathtt{tt} \wedge AX\mathtt{tt}$

R1. if $\vdash \Phi \rightarrow \Psi$ then $\vdash EX\Phi \rightarrow EX\Psi$

R2. if $\vdash \Phi' \rightarrow \Psi \wedge EX\Phi'$ then $\vdash \Phi' \rightarrow E(\Phi R \Psi)$

R3. if $\vdash \Phi' \rightarrow \Psi \wedge AX(\Phi' \vee A(\Phi R \Psi))$
then $\vdash \Phi' \rightarrow A(\Phi R \Psi)$

R4. if $\vdash \Phi$ and $\vdash \Phi \rightarrow \Psi$ then $\vdash \Psi$

The same arguments for comparing the two LTL axiomatisations also hold for the two axiomatisations of CTL. Ax1, Ax5 – Ax7, and R4 are already present in B. Ax2 and Ax3 are covered by the abbreviation of F. Ax4 can be proved by a combination of 6 – 9, 1 and MP. 1, AXGen, 7, MP and 6 establish Ax8. Rule R1 is simulated using AXGen, 9, MP, 7 and the hypothesis of having a shorter proof of $\Phi \rightarrow \Psi$ in B. R2 is simulated in the following way. Suppose there is a B-proof of $\Phi' \rightarrow \Psi \wedge EX\Phi'$. Then, by 4, 1, and MP there is also a proof of $\Phi' \rightarrow \Psi \wedge (\Phi \vee EXE((\Phi \vee \Phi')R(\Psi \vee \Phi')))$ for any Φ. Using ERel yields a proof of $\Phi' \rightarrow E(\Phi R \Psi)$. Simulating R3 is similar.

7 Conclusion

We have introduced a game theoretic approach to satisfiability checking of LTL and CTL. It remains to be seen if focus games extend to richer logics such as CTL* and modal μ-calculus. In [12] it was shown that focus games can also be used to solve the model checking problem for CTL*. The game trees arising there are very similar to the tableau structures used in [2, 1]. However, in order to tackle the problem of deciding whether fixed point constructs are regenerated or reproduced these authors pursue a different strategy. Take the unfolding of $\Phi U \Psi$ for example. While the focus highlights the case that player $\exists$ always chooses the term in which $\Phi U \Psi$ occurs again, a path in the tableaux of [2] is successful if Ψ never occurs after $\Phi U \Psi$. The difference seems to be a point of view only. In the focus games it is checked whether a fixed point construct is regenerated, therefore it is never fulfilled. In the tableau approach it is checked whether it is never fulfilled, therefore it is regenerated.

In [1] the authors define Tableau Büchi Automata which are essentially the same as the tableaux of [2]. As with the focus games, this enables the authors to handle the regeneration problem of fixed points implicitly. Instead of explicitly requiring tableaux to be processed with a depth-first-search, the solution to the regeneration problem is encoded in an acceptance condition, which is in that case a generalised Büchi condition. However, this small difference is the key to the strengthening lemma (Lemma 1 of Section 3) which underpins the proofs of completeness of the axiomatisations.

A more recent automata theoretic approach to satisfiability and model checking employs alternating automata [16, 11]. Although these appear to be very game theoretic, they rely upon automata over trees which capture the “and” branching, both in the case of the boolean “and” and in the case for CTL of branching through next states. In both cases of LTL and CTL formulas are states of the automata, and transitions are determined by maximal consistent sets of atomic propositions. The acceptance conditions decide acceptable fixed point regeneration. It is not clear if this approach can underpin sound and complete axiomatisations.

References

[1] Bhat, G., and Cleaveland, R. (1996). Efficient model checking via the equational μ-calculus. *Proc. 11th Annual IEEE Symp. on Logic in Computer Science, LICS’96*, 304-312.

[2] Bhat, G., Cleaveland, R., and Grumberg, O. (1995). Efficient on-the-fly model checking for CTL*. *Proc. 10th Annual IEEE Symp. on Logic in Computer Science, LICS’95*, 388-397.

[3] Chandra, A., Kozen, D. and Stockmeyer, L. (1981). Alternation. *Journal of ACM*, **28**, 114-133.

[4] Clark, E., Emerson, E., and Sistla, P. (1986). Automatic verification of finite state concurrent systems using temporal logic. *ACM Trans. on Programming Languages and Systems*, **8**, 244-263.

[5] Emerson, E., and Halpern, J. (1985). Decision procedures and expressiveness in the temporal logic of branching time. *Journal of Comput. System Sci.*, **30**, 1-24.

[6] Emerson, E., and Jutla, C. (1988). The complexity of tree automata and logics of programs. *Procs. 29th IEEE Symp. on Foundations of Comput. Science*, 328-337.

[7] Gabbay, D., Pnueli, A., Shelah, S., and Stavi, J. (1980). The temporal analysis of fairness. *Procs. 7th ACM Symp. on Principles of Programming Languages*, 163-173.

[8] Goldblatt, R. (1992). *Logics of Time and Computation*, CSLI Lecture Notes No. 7.

[9] Hodges, W. (1993). *Model Theory*, Cambridge University Press.

[10] Kozen, D. (1983). Results on propositional mu-calculus. *Theoretical Computer Science*, **27**, 333-354.

[11] Kupferman, O., Vardi, M., and Wolper, P. (2000). An automata-theoretic approach to branching-time model checking. *Journal of ACM*, **47**, 312-360.

[12] Lange, M., and Stirling, C. (2000). Model checking games for CTL*. *Proc. Int. Conf. on Temporal Logic, ICTL’2000*.

[13] Lichtenstein, O., and Pnueli, A. (2000). Propositional temporal logics: decidability and completeness. *Logic Journal of the IGPL*, **8**, 55-85.

[14] Reynolds, M. (2000). An axiomatisation of full computation tree logic. To appear *Journal of Symbolic Logic*.

[15] Stirling, C. (1992). Modal and temporal logics. In *Handbook of Logic in Computer Science*, Vol 2, Oxford University Press, 477-563.

[16] Vardi, M. (1996). An automata-theoretic approach to linear temporal logic. *Lecture Notes in Computer Science*, **1043**, 238-266.

[17] Vardi, M., and Wolper P. (1994). Reasoning about infinite computations. *Information and Computation*, **115**, 1-37.

[18] Walukiewicz, I. (2000). Completeness of Kozen’s axiomatisation of the propositional mu-calculus. *Information and Computation*, **157**, 142-182.

[19] Winskel, G. (1991). A note on model checking the modal nu-calculus, *Theoretical Computer Science*, **83**, 157-167.