Automata, Logic and Games: Theory and Application

1. Büchi Automata and S1S

Luke Ong

University of Oxford

TACL Summer School
University of Salerno, 14-19 June 2015

Motivation

Model Checking—an approach to verification that promises accurate analysis with push-button automation—has been a truly successful application of logic to computer science.

2007 ACM Turing Award (Clarke, Emerson and Sifakis) “for their rôle in developing model checking into a highly effective verification technology, widely adopted in hardware and software industries”.

What is Model Checking?

Problem: Given a system *Sys* (e.g. an operating system) and a correctness property *Spec* (e.g. deadlock freedom), does *Sys* satisfy *Spec*?

The model checking approach:

1. Find an abstract model M of the system *Sys*.
2. Describe the property *Spec* as a formula φ of a (decidable) logic.
3. Exhaustively check if φ is violated by M.

Mathematical and scientific background

Logic and Automata

- An old tradition in analysis of digital circuits (Church IMU 1962).
- Infinite-state breakthrough by Rabin (1969): effective equi-expressivity between MSOL and tree automata, and hence decidability of MSOL.
- Automata-theoretic approach to model checking (Vardi & Wolper 1984, etc.).

Games

- Ideas from logic (descriptive set theory & proof theory) & combinatorics.
- Connexions with algorithmics and semantics:
  - "Verification games": reduction of model checking to game solving (Gurevich & Harrington 1982; Stirling 1995, etc.)
  - "Semantic games": game semantics and the full abstraction problem (Abramsky et al.; Hyland & O. 1994–2000, etc.)

Aims

To introduce the mathematical theory underpinning the computer-aided verification of computing systems.

- Automata (on infinite words, trees and graphs) as a model of computation of state-based systems.
- Logical systems (such as temporal and modal logics) for specifying correctness properties.
- Two-person games as a mathematical model of the interactions between a system and its environment.

Lecture Course Outline

Part 1: Foundations. Ideas and some technical details.

1. Büchi Automata and S1S
2. Parity Games, Tree Automata, Rabin's Theorems and S2S

Part 2: Active research topic. Mainly ideas.

Higher-Order Model Checking

Contents of Lecture 1

Aim: Prove Büchi's Theorem (S1S is decidable) via Büchi automata.

A Büchi automaton is a method of defining a set of ω-words over a finite alphabet Σ.

A (nondeterministic) ***Büchi automaton*** is a 5-tuple $A = (Q, \Sigma, q_0, \Delta, F)$ where

- Q is a finite set of states
- Σ is a finite alphabet of letters
- $q_0 \in Q$ is the initial state
- $\Delta \subseteq Q \times \Sigma \times Q$ is a transition relation
- $F \subseteq Q$ is the set of final (or accepting) states.

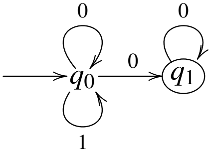
If Δ is a function $Q \times \Sigma \longrightarrow Q$, we say A is deterministic.

Think of an automaton as a finite digraph, whose nodes are states, and whose edges are labelled by letters from Σ.

A ω-word is accepted by a Büchi automaton if it is spelt out by an infinite path which, starting from the initial node, visits some final state infinitely often.

Convention. *When drawing automata as graphs, we circle the final states, and indicate the initial state by an arrow.*

Example: Two-state Büchi automaton, over $\Sigma = \{ 0, 1 \}$.



This Büchi automaton accepts all binary ω-words that contain only finitely many occurrences of 1.

Language recognised by a Büchi automaton A

A run ρ on $\alpha \in \Sigma^\omega$ is an infinite path in the digraph underlying A (so $\rho \in Q^\omega$), starting from the initial node, whose labels on the edges spell out α.

Büchi acceptance condition: A run $\rho \in Q^\omega$ on α is ***accepting*** just if there is a final state that occurs infinitely often in ρ; or equivalently (because F is finite) $\mathsf{inf}(\rho) \cap F \neq \varnothing$, writing $\mathsf{inf}(\rho)$ for the set of states that occur infinitely often in ρ.

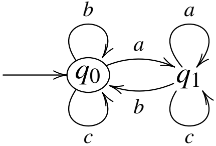
An ω-word α is ***accepted*** by an automaton A if there is an accepting run of A on α.

The ***language recognised*** by A, written $L(A)$, is the set of ω-words accepted by A.

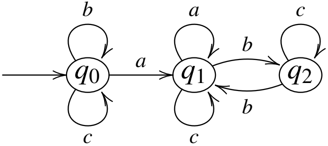
An ω-language is ***Büchi recognisable*** if it is recognised by some Büchi automaton.

Set $\Sigma = \{ a, b, c \}$.

(i) $L_1 \subseteq \Sigma^\omega$ consists of ω-words in which after every occurrence of a there is some occurrence of b.



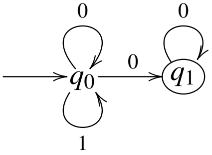
(ii) L_2 consists of ω-words in which between every two occurrences of a, there is an even number of b.



Question. Which is recognised by a deterministic automaton? Ans: Both.

Büchi automata are not determinisable

The Büchi-recognisable language L_3, which consists of binary ω-words that have only finitely many occurrences of 1, is not recognised by any deterministic Büchi automaton.



Thus, unlike automata over finite words, deterministic Büchi automata are less expressive than nondeterministic Büchi automata.

Are Büchi automata closed under complementation?

Closure properties

Important for proving decidability of S1S.

Theorem (Büchi)

Büchi recognisable languages are closed under Boolean operations: union, intersection, and complementation.

- Closure under union: easy, like automata over finite words.
- Closure under intersection: tricky, need to "synchronise" visits of final states of respective automata
- Closure under complementation: trickier!
  - Büchi's proof uses Ramsey's Theorem.

Closure of Büchi automata under complementation: alternative proof

Recall: let $\rho \in Q^\omega$, $\mathsf{inf}(\rho)$ is the set of states that occur infinitely often in the run ρ.

Muller acceptance condition takes the form $\mathcal{F} = \{ F_1, \cdots, F_k \}$ where each $F_i \subseteq Q$; and a run $\rho \in Q^\omega$ is Muller accepting just if $\mathsf{inf}(\rho) \in \mathcal{F}$.

Deterministic Muller automata are complementable: If $L \subseteq \Sigma^\omega$ is recognised by a deterministic automaton with Muller condition $\mathcal{F}$, then $\Sigma^\omega \setminus L$ is recognised by the same automaton with Muller condition $2^Q \setminus \mathcal{F}$.

Theorem (McNaughton 1966)

Deterministic Muller automata and nondeterministic Büchi automata are equivalent.

Non-emptiness problem: Given a Büchi automaton A, is $L(A) \neq \varnothing$?

Theorem

The non-emptiness problem for Büchi automata $A = (Q, \Sigma, \Delta, q_0, F)$ is decidable in time $O(|Q| + |\Delta|)$. In fact, the problem is NL-complete.

We have:

$L(A) \neq \varnothing$

iff there is a "lasso" i.e. a path from q_0 to some $q \in F$, and a path from q back to itself

iff automaton A (viewed as a directed graph) has a *non-trivial* SCC which is reachable from q_0 and contains an accepting state q

Recall: A strongly connected component (SCC) of a directed graph is a maximal subgraph such that for every pair of vertices in the subgraph, there is a path from one vertex to the other.

Monadic second-order logic of one successor (S1S)

Aim. Introduce ***S1S*** and prove that it is equivalent to Büchi automata, and hence decidable.

- Second-order means that we allow quantification over *relations*.
- Monadic means that quantification is restricted to *monadic* relations, namely, sets.

The vocabulary consists of a unary function symbol **s** and a binary predicate symbol $\in$.

Fix a logical structure $(\omega, \mathbf{s}, \in)$

- **s** is the successor function $x \mapsto x + 1$
- $\in \; \subseteq \; \omega \times 2^\omega$ is the standard membership relation between elements and sets.

Syntax of S1S (parsimonious by design!)

Variables

- First-order variables (x, y, z, etc.) range over natural numbers (regarded as positions in ω-words)
- Second-order variables (X, Y, Z, etc.) range over sets of natural numbers.

Terms

- First-order variables are terms.
- If t is a term, so is $\mathbf{s}\,t$.

Formulas

- Atomic formulas are of the shape "$t \in X$" where t is a term and X is a 2nd-order variable.
- S1S formulas are built up from atomic formulas using standard Boolean connectives, with $\forall$- and $\exists$-quantifications over 1st and 2nd-order variables.

Constructs definable in S1S

Note that 0, and the atomic formulas $s = t$ and $s < t$ are definable in S1S.

- $x = y \ := \ \forall X. x \in X \leftrightarrow y \in X$
- $X \subseteq Y \ := \ \forall x. x \in X \to x \in Y$
- $x = 0 \ := \ \forall y. \neg(x = \mathbf{s}\, y)$ "x has no predecessor"
- $x \leq y \ := \ \forall X. (x \in X \wedge (\forall z. z \in X \to \mathbf{s}\, z \in X)) \ \to \ y \in X$
  "Every set X that contains x and is closed under successor also contains y."
- "X is finite" $:= \ \exists x. \forall y. (y \in X \to y \leq x)$
  "X has an upper bound"

Semantics of S1S is standard

Write $\varphi(x_1, \cdots, x_m, X_1, \cdots, X_n)$ to mean: φ has free 1st-order variables from $x_1, \cdots, x_m$ and free 2nd-order variables from $X_1, \cdots, X_n$.

Let $a_i \in \omega$ and $P_j \subseteq \omega$. For $\overline{a} = a_1, \cdots, a_m$ and $\overline{P} = P_1, \cdots, P_n$, write

$$\overline{a}; \overline{P} \models \varphi(x_1, \cdots, x_m, X_1, \cdots, X_n)$$

to mean " the structure $(\omega, \mathbf{s}, \in)$ with the valuation $\{ \bar{x} \mapsto \overline{a}; \overline{X} \mapsto \overline{P} \}$ satisfies φ".

Think of $(\overline{a}, \overline{P})$ as a model of $\varphi(\bar{x}, \overline{X})$

Representing a set of natural numbers as an infinite word

We represent any $P \subseteq \omega$ by its characteristic word, written $\ulcorner P \urcorner \in \mathbb{B}^\omega$, defined by

$$\ulcorner P \urcorner(i) = 1 \quad \leftrightarrow \quad i \in P.$$

Example

subsets of ω	characteristic words
multiples of 3	$100100100100100100100 \cdots$
prime numbers	$001101010001010001010001 \cdots$

We represent $a \in \omega$ by the characteristic word of the singleton set $\{ a \}$.

More generally the characteristic word of a tuple

$$(a_1, \cdots, a_m, P_1, \cdots, P_n) \in \omega^m \times (2^\omega)^n$$

written $\ulcorner a_1, \cdots, a_m, P_1, \cdots, P_n \urcorner$, is an infinite word over the alphabet $\mathbb{B}^{m+n}$ such that each of the $m + n$ tracks (or rows) is the characteristic word of the corresponding component of the tuple $(\overline{a}, \overline{P})$.

Defining ω-languages by S1S formulas

$L \subseteq \mathbb{B}^\omega$ is S1S-definable by $\varphi(X)$ just if $L = \{\, \ulcorner P \urcorner \in \mathbb{B}^\omega : P \vDash \varphi(X) \,\}$.
I.e. Each P that satisfies $\varphi(X)$ consists of the positions of '1' in an ω-word in $L \subseteq \mathbb{B}^\omega$.

Examples

1. The set $L_1 = \{\, \alpha \in \mathbb{B}^\omega : \alpha \text{ has infinitely many 1s} \,\}$ is first-order definable by
$$\varphi_1(X) \;=\; \forall x.\exists y.x < y \,\wedge\, y \in X$$

2. $(00)^* 1^\omega$ is definable by
$$\varphi_2(X) \;=\; \exists Y.\exists x. \left(\begin{array}{ll} & 0 \in Y \\ \wedge & \forall y.y \in Y \leftrightarrow \mathbf{s}\, y \notin Y \\ \wedge & x \in Y \\ \wedge & \forall z.z < x \to z \notin X \\ \wedge & \forall z.z \geq x \to z \in X \end{array} \right)$$

Büchi-recognisable ω-languages are S1S-definable

Recall: An ω-language $L \subseteq (\mathbb{B}^n)^\omega$ is ***S1S definable*** just if there is an S1S-formula $\varphi(X_1, \cdots, X_n)$ such that

$$L = \{ \ulcorner P_1, \cdots, P_n \urcorner \in (\mathbb{B}^n)^\omega : \overline{P} \vDash \varphi(\overline{X}) \}.$$

Theorem (Büchi 1)

For every Büchi automaton A over the alphabet $\mathbb{B}^n$, there is an S1S formula $\varphi_A(X_1, \cdots, X_n)$ such that

$$\forall (P_1, \cdots, P_n) \in (2^\omega)^n \; : \; \overline{P} \vDash \varphi_A(\overline{X}) \; \leftrightarrow \; \ulcorner P_1, \cdots, P_n \urcorner \in L(A).$$

Proof idea. Assume $n = 1$.

Take a Büchi automaton $A = (Q, \Sigma, q_1, \Delta, F)$ where $\Sigma = \mathbb{B}$, construct an S1S-formula $\varphi_A(X)$ that asserts

> *"there is an accepting run of A on an input ω-word given by the characteristic word of X".*

AIM: To code an accepting run $\rho \in Q^\omega$

Assume $Q = \{ q_1, \cdots, q_m \}$.

A run $\rho(0)\ \rho(1)\ \cdots \in Q^\omega$ is coded by m subsets of ω, namely $Y_1, \cdots, Y_m$, such that

$$i \in Y_k \quad \leftrightarrow \quad \rho(i) = q_k$$

Observe that $Y_1, \cdots, Y_m$ form a *partition* of ω.

Define predicate $partition(Y_1, \cdots, Y_m)$ to be

$$\forall x. \left(\bigvee_{i=1}^{m} x \in Y_i \right) \quad \wedge \quad \neg \left(\exists y. \bigvee_{i \neq j} (y \in Y_i \wedge y \in Y_j) \right)$$

Putting it all together

Given a Büchi automaton $A = (\{1, \cdots, m\}, \mathbb{B}, 1, \Delta, F)$, define $\varphi_A(X)$ to be

$$\exists Y_1 \cdots Y_m \,.\, \left(\begin{array}{ll} & \mathit{partition}(Y_1, \cdots, Y_m) \\ \wedge & 0 \in Y_1 \\ \wedge & \forall x. \bigvee_{(i,a,j) \in \Delta} (x \in Y_i \;\wedge\; [x \in X_a] \;\wedge\; \mathbf{s}\, x \in Y_j) \\ \wedge & \forall x. \exists y. (x < y \;\wedge\; \bigvee_{i \in F} y \in Y_i) \end{array} \right)$$

Thus for every $P \in 2^\omega$, A accepts $\ulcorner P \urcorner$ iff $P \models \varphi_A(X)$. $\square$

Theorem (Büchi 2)

For every S1S formula $\varphi(x_1, \cdots, x_m, X_1, \cdots, X_n)$, there is an equivalent non-determinstic Büchi automaton A_φ over alphabet $\mathbb{B}^{m+n}$, in the sense that

$$L(A_\varphi) = \{ \ulcorner a_1, \cdots, a_m, P_1, \cdots, P_n \urcorner \in (\mathbb{B}^{m+n})^\omega \mid \overline{a}, \overline{P} \vDash \varphi \}$$

Proof. By induction on the size of φ.

An **atomic formula** has the form $\underbrace{\mathbf{s}\,(\mathbf{s} \cdots (\mathbf{s}}_{k}\ x_i) \cdots) \in X_j$.

We build a Büchi automaton to read the tracks i and $m + j$ only (corresponding to x_i and X_j respectively), performing the following check: if the unique 1 of the i-track is at position l (say), then the $(m + j)$-track has a 1 at position $l + k$.

Proof of Büchi's Theorem 2 cont'd

Negation: Use closure of Büchi automata under complementation

Consider $\neg\varphi(\bar{x}, \overline{X})$.
By the IH, suppose A_φ is equivalent to φ. Set $A_{\neg\varphi}$ to be the automaton that recognises the complement of $L(A_\varphi)$.

Disjunction: Use closure of Büchi automata under union

2nd-order existential quantification: Use non-determinacy of Büchi automata

The theory S1S

The ***theory S1S*** is the set of S1S sentences that are satisfied in the structure $(\omega, \mathbf{s}, \in)$. For instance,

$$\forall X.\exists Y.\forall x.(x \in X \to x \in Y) \text{ is in the theory,}$$
$$\forall X.\exists y.\forall x.(x \in X \to x < y) \text{ is not in the theory.}$$

Corollary (Büchi)

The theory S1S is decidable: given an S1S sentence φ, it is decidable whether or not φ holds in $(\omega, \mathbf{s}, \in)$.
Procedure: Construct A_φ and test whether $L(A_\varphi)$ is non-empty.

Membership in the theory S1S is non-elementary.

$$exp_0(n) := n \qquad exp_{h+1}(n) := 2^{exp_h(n)}.$$