# Constructive Analysis of S1S and Büchi Automata

Moritz Lichter and Gert Smolka
Saarland University

April 13, 2018

We study S1S and Büchi automata in the constructive type theory of the Coq proof assistant. For UP semantics (ultimately periodic sequences), we verify Büchi's translation of formulas to automata and thereby establish decidability of S1S constructively. For AS semantics (all sequences), we verify Büchi's translation assuming that sequences over finite semigroups have Ramseyan factorisations (RF). Assuming RF, UP semantics and AS semantics agree. RF is a consequence of Ramsey's theorem and implies the infinite pigeonhole principle, which is known to be unprovable constructively. We show that each of the following properties holds for UP semantics but is equivalent to RF for AS semantics: excluded middle of formula satisfaction, excluded middle of automaton acceptance, and existence of complement automata.

## 1 Introduction

S1S is the monadic second-order logic of order with first-order variables ranging over natural numbers and second-order variables ranging over possibly infinite sets of numbers. Following Büchi [4], decidability of S1S can be shown with a compositional translation of formulas to automata realizing constructions of formulas with operations on automata. The automata employed by the translation are NFAs accepting infinite sequences. One speaks of Büchi automata to indicate a particular acceptance condition for infinite sequences formulated by Büchi. The reduction of formulas to automata works well for S1S and various other logics, including temporal logics [9, 13, 18].

We study S1S and Büchi's translation to automata in the constructive type theory of the Coq proof assistant [1]. Coq's type theory extends Martin-Löf type theory with an impredicative universe of propositions such that excluded middle can be assumed consistently for all propositions. This matters for our purposes since several aspects of S1S and Büchi's translation cannot be verified constructively.

We represent sequences over a type $A$ as functions $\mathsf{N} \to A$ from natural numbers to $A$ and sets of numbers as boolean sequences (functions $\mathsf{N} \to \mathsf{2}$). An automaton accepts a sequence if there is a run on the sequence that passes through accepting states infinitely often.

When we verify the operations on automata needed for the translation of formulas, all operations but Büchi's complement operation can be verified constructively. The verification of Büchi's complement operation requires a restricted form of Ramsey's theorem known as additive Ramsey theorem [11, 15]. We refer to the property asserted by the theorem as RA. If we assume RA, we can verify the translation of formulas into automata and thereby show that S1S is decidable.

We will mostly work with a property RF that is constructively equivalent to RA. The verification of Büchi's complement operation is constructive except a single spot where an instance of RF is needed. RF says that every sequence over a finite semigroup of colors has a factorisation $u_0, u_1, u_2, \dots$ such that all strings $u_1, u_2, u_3 \dots$ have the same color. Following Blumensath [2] we call factorisations with this property Ramseyan factorisations. Variants of Ramseyan factorisations appear in the literature [13, 16].

We show that RF implies the infinite pigeonhole principle (for every sequence over a finite type there is an element occurring infinitely often), which is known to be unprovable constructively [20]. It follows that RF and RA are unprovable constructively, too.

Let FX be the property that the satisfaction relation $I \models \varphi$ between interpretations and formulas of S1S satisfies XM (excluded middle): $\forall I\varphi.\ I \models \varphi \vee I \not\models \varphi$. Note that FX is a special instance of general excluded middle. A main result of this paper is a proof that FX and RA are equivalent constructively. As a consequence, we know that RA is necessary and sufficient to correctly formalise S1S in constructive type theory.

We provide two further characterisations of RF by showing that RF is constructively equivalent to AC (complement automata exist) and AX (acceptance by automata satisfies XM). AC is interesting since it implies that no complement operation can be verified constructively.

We refer to the standard semantics of automata and formulas introduced so far as *AS semantics* (for all sequences) to distinguish it from an alternative semantics we call *UP semantics*. UP semantics [3, 6, 7] is based on ultimately periodic sequences $xy^\omega$, finitely specified with two strings $x$ and $y$. Since UP sequences over finite semigroups obviously have Ramseyan factorisations, correctness of Büchi's complement operation for UP sequences can be verified constructively.

We show that in constructive type theory S1S with UP semantics is decidable and classical (i.e., UP satisfaction of formulas satisfies XM). This shows that UP semantics provides a purely constructive formalisation of S1S. This is in contrast to AS semantics, which requires RF to adequately formalise S1S. We show that, given RF, UP semantics agrees with AS semantics as it comes to satisfiability of formulas.

We provide one further constructive characterisation of RF we call AU. AU says that two automata accept the same sequences if they accept the same UP sequences. AU is known to hold classically [6, 7].

There is a remarkable coincidence between our work and the work of Kołodziejczyk et al. [11] who study the translation of S1S formulas to automata in $\mathrm{RCA}_0$, a system of weak second-order arithmetic (a classical logic satisfying XM). They show that the following properties are pairwise equivalent in $\mathrm{RCA}_0$: correctness of Büchi complementation, decidability of S1S, and the additive Ramsey theorem.

We spend considerable effort on proving that both FX and AX imply RF. For this we establish a further constructive characterisation of RF we call RP for Ramseyan pigeonhole principle. RP has a straightforward classical proof and can be related to satisfaction of S1S formulas and Büchi acceptance of automata. RP is based on a relation for sequences over finite semigroups appearing as merging relation in the literature [5, 12, 13, 15, 16]. The relation is used in the literature to prove Ramseyan properties similar to RA and RF using excluded middle.

**Organisation of the paper** The paper is written at a level of abstraction that does not require detailed knowledge of constructive type theory. We start with preliminaries concerning type theory, sequences, and Ramseyan factorisations and show that RF and RA are equivalent. We postpone the definition of full S1S and start with minimal S1S providing the basis for the translation to automata. We review Büchi automata and show that all operations but complement can be verified constructively. In particular, we verify the correctness of the operation for existential quantification for UP semantics. We then show correctness of Büchi's complement operation, both for UP sequences (no assumption needed) and all sequences (RF needed). We now show that RF, AC, and AU are constructively equivalent and obtain the decidability results for minimal S1S for both AS semantics and UP semantics.

We then define full S1S and reduce it to minimal S1S. What is missing at this point are proofs that FX and AX imply RF. For this purpose we introduce RP and show that it is equivalent to RF. We then show that both FX and AX both entail RP.

**Coq development** There is a Coq development proving all results of the paper. Instead of defining AS and UP semantics of S1S separately, we work with a generalisation, which can be instantiated for AS and UP semantics. The Coq development is available at `http://www.ps.uni-saarland.de/extras/S1S`. The definitions and statements in this paper are hyperlinked with our Coq development available for browsing on our project web page.

## 2 Preliminaries

In constructive type theory, the law of excluded middle (XM) is not built-in and thus propositions like $P \vee \neg P$ and $\neg\neg P \to P$ are not trivially provable. Moreover, there are no native sets, and functions must be total and can only be defined computationally.

We write $\mathsf{1}$ and $\mathsf{2}$ for the inductive types providing the single value $\mathsf{1}$ and the boolean values $\mathsf{true}$ and $\mathsf{false}$, respectively. We write $\mathsf{N}$ for the inductive type providing the numbers $0$, $S0$, $S(S0)$, … . Note that $Sn = n + 1$. The letters $i$, $j$, $k$, $l$, $m$, and $n$ will range over numbers.

We write $\exists^{\omega} n.\, pn$ for $\forall k \exists n.\, n \geqslant k \wedge pn$ and say that $p$ **holds infinitely often**. Moreover, we write $\exists n \geqslant k.\, pn$ for $\exists n.\, n \geqslant k \wedge pn$, and $\forall n \geqslant k.\, pn$ for $\forall n.\, n \geqslant k \to pn$.

Let $p$ be a unary predicate on a type $A$. We say that $p$ **satisfies XM** if $\forall a.\, pa \vee \neg pa$, and that $p$ **is decidable** if we have a function $f : A \to \mathsf{2}$ such that $\forall a.\, pa \leftrightarrow fa = \mathsf{true}$. For propositions and predicates with $n \geqslant 2$ arguments satisfaction of XM and decidability are defined analogously. Note that decidable propositions and predicates satisfy XM. Since functions definable in constructive type theory are computable, decidable predicates are computationally decidable.

We will make use of the fact that the predicates $\lambda ij.i < j$ and $\lambda ij.i \leqslant j$ are decidable $(i, j : \mathsf{N})$.

We will occasionally use the proposition

$$\mathsf{XM} := \forall P.\, P \vee \neg P$$

which states that every proposition satisfies XM. Note that every predicate satisfies XM if we assume $\mathsf{XM}$. Assuming $\mathsf{XM}$ in Coq is consistent and does not change our notion of decidability, as functions on non propositional types stay computable. Given a proposition $P$, we write $\mathsf{xm}(P) := P \vee \neg P$.

**Fact 2.1** For all propositions $P$, $Q$ and all predicates $p$:

1. $\neg(P \wedge Q) \leftrightarrow (P \to \neg Q)$.
2. $\neg(\exists x.px) \leftrightarrow \forall x.\neg px$.
3. $\mathsf{xm}(P) \to (P \leftrightarrow \neg\neg P)$.
4. $\mathsf{xm}(\exists x.px) \to ((\exists x.px) \leftrightarrow \neg\forall x.\neg px)$.

A **sequence** over a type $A$ is a function $\sigma : \mathsf{N} \to A$. If $\sigma n = a$, we say that **$\sigma$ is $a$ at position $n$**. We will use the notation $A^\omega := \mathsf{N} \to A$ for the type of sequences over $A$.

Two sequences $\sigma$ and $\tau$ over $A$ **agree** if $\sigma n = \tau n$ for all $n$. We write $\sigma \equiv \tau$ to say that $\sigma$ and $\tau$ agree. We also say that two sequences are **equivalent** if they agree. The notion of agreement is needed since we work in a non-extensional type theory.

A **boolean sequence** is a sequence over $\mathsf{2}$. The letter $\beta$ will range over boolean sequences. A boolean sequence may be seen as a decidable set of numbers. Following this interpretation, we write $n \in \beta$ for $\beta n = \mathsf{true}$. A boolean sequence is called **infinite** if it is infinitely often $\mathsf{true}$ (i.e., $\exists^\omega n.\, n \in \beta$), and **nonempty** if $\exists n.\, n \in \beta$.

In Coq's constructive type theory one can (computationally) obtain an element for a nonempty sequence. This fact is known as **constructive choice**. We will repeatedly make use of constructive choice when we construct functions. Here is a more precise formulation of this fact.

**Fact 2.2 (Constructive Choice)** One can define a function that given a boolean sequence $\beta$ and a proof of $\exists n.\, n \in \beta$ yields an $n \in \beta$.

A function $f : \mathsf{N} \to \mathsf{N}$ **enumerates** a boolean sequence $\beta$ if $\forall n.\, n \in \beta \leftrightarrow \exists k.\, fk = n$.

**Fact 2.3** For every infinite boolean sequence $\beta$ one can obtain a strictly monotone function enumerating $\beta$. Moreover, for every strictly monotone function $f : \mathsf{N} \to \mathsf{N}$ one can obtain an infinite boolean sequence $\beta$ such that $f$ enumerates $\beta$.

**Proof** The first claim follows with constructive choice. The second claim follows since $\exists k.\, fk = n$ is equivalent to $\exists k \leqslant n.\, fk = n$ for strictly monotone $f$. ■

**Strings** over a type $A$ are provided with an inductive type $A^+$ defined with two constructors:

$$x : A^+ \ ::= \ a \mid ax \qquad (a : A)$$

Our definition does not provide an empty string, since this is advantageous for the purposes of this paper. When we say *string over* $A$ we will always mean an element of $A^+$. We write $xy$ for the **concatenation** of two strings $x$ and $y$.

Given two strings $x$ and $y$ over $A$, we write $xy^\omega$ for a sequence agreeing with the infinite concatenation $xyyy\cdots$. Formally, we define the sequence $xy^\omega : A^\omega$ by recursion on numbers:

$$\begin{aligned} ay^\omega(0) &= a & \qquad ay^\omega(Sn) &= yy^\omega(n) \\ (ax)y^\omega(0) &= a & \qquad (ax)y^\omega(Sn) &= xy^\omega(n) \end{aligned}$$

We call a pair $(x, y)$ of two strings over $A$ a **UP sequence** over $A$. Notationally, we will identify the pair $(x, y)$ with the sequence $xy^\omega$.

A **discrete type** is a type $X$ together with a boolean function deciding equality on $X$. A **finite type** is a discrete type $X$ together with a duplicate-free list containing all elements of $X$. The letters $\Sigma$ and $Q$ will range over finite types. The type $1$ and $2$ can be accommodated as finite types and $\mathsf{N}$ can be accommodated as discrete type. Moreover, finite types are closed under taking **products** $\Sigma_1 \times \Sigma_2$ and **sums** $\Sigma_1 + \Sigma_2$. Given a finite type $\Sigma$, there is a finite type $2^{\Sigma}$ containing exactly the sets over $\Sigma$. The sets in $2^{\Sigma}$ have decidable membership.

## 3 Ramseyan Factorisations

A **factorisation** of a sequence $\sigma$ over $A$ is a sequence $\tau$ over $A^+$ such that $\sigma$ agrees with the infinite concatenation $(\tau 0)(\tau 1)(\tau 2)\cdots$.

A **finite semigroup** is a finite type $\Gamma$ together with an associative operation $+$. We will call the elements of finite semigroups **colors** and define the **color of a string** over $\Gamma$ as follows:

$$\mathsf{C}(a_0 \cdots a_n) \ := \ a_0 + \cdots + a_n$$

We have $\mathsf{C}(xy) = \mathsf{C}x + \mathsf{C}y$ for all strings $x$ and $y$ over $\Gamma$. Note that string concatenation is a semigroup operation for $\Gamma^+$ and that $\mathsf{C}$ is a semigroup morphism $\Gamma^+ \to \Gamma$. The letter $\Gamma$ will range over finite semigroups.

A **Ramseyan factorisation** of a sequence $\sigma$ over $\Gamma$ is a factorisation $\tau$ of $\sigma$ such that all strings $\tau 1, \tau 2, \tau 3, \ldots$ have the same color. We call $\mathsf{C}(\tau 1)$ the **color of the factorisation**. Note that $\tau 0$ and $\tau 1$ may have different colors. We define the proposition $\mathsf{RF}$ as follows:

$\mathsf{RF} :=$ Every sequence over a finite semigroup has a Ramseyan factorisation.

**Fact 3.1** Every UP sequence over a finite semigroup has a Ramseyan factorisation.

**Proof** The sequence $x, y, y, y, \ldots$ is a Ramseyan factorisation of $xy^{\omega}$. ■

We will now show that $\mathsf{RF}$ implies the infinite pigeonhole principle and Markov's principle. The infinite pigeonhole principle is unprovable constructively [20] and Markov's priniciple is unprovable in CIC [8, 14], a type theory similar to the one of Coq. Hence $\mathsf{RF}$ is unprovable, too.

We define two propositions expressing the *infinite pigeonhole principle* and *Markov's principle*:

$\mathsf{IP} :=$ For every finite type $\Sigma$ and every sequence $\sigma$ over $\Sigma$ there exists a value $a : \Sigma$ such that $\exists^{\omega} n.\ \sigma n = a$.

$\mathsf{MP} :=$ If a boolean sequence is not constantly false, then there exists a position where it is true.

**Fact 3.2** RF implies IP and IP implies MP.

**Proof** RF → IP. Assume RF and let $\sigma$ be a sequence over a finite type $\Sigma$. We fix $\lambda ab.a$ as semigroup operation on $\Sigma$. By RF we have a Ramseyan factorisation $\tau$ of $\sigma$. Thus there is a color $a : \Sigma$ such that the first symbol of $\tau n$ is $a$ for all $n \geqslant 1$. Hence $\exists^{\omega} n.\ \sigma n = a$.

IP → MP. Assume IP and let $\beta$ be a boolean sequence such that $\neg\forall n.\ \beta n = \mathsf{false}$. Let $\beta'$ be the boolean sequence such that $\beta' n$ is the boolean disjunction of $\beta 0, \dots, \beta n$. By IP we have a boolean value $b$ such that $\exists^{\omega} n.\ \beta' n = b$. If $b = \mathsf{true}$, we have a position where $\beta$ is true. If $b = \mathsf{false}$, we can show that $\beta$ is constantly false, which contradicts the assumption. ■

Next we show that RF is equivalent to a proposition RA expressing a weakening of Ramsey's theorem called *additive Ramsey theorem* in Kołodziejczyk et al. [11] and *Ramsey's theorem with additive coloring* in Riba [15]. The definition of RA requires some preparation.

Let $\Gamma$ be a finite semigroup and $\gamma$ be a function $\mathsf{N} \to \mathsf{N} \to \Gamma$. We call $\gamma$ an **additive coloring into $\Gamma$** if $\gamma ij + \gamma jk = \gamma ik$ for all $i < j < k$. A boolean sequence $\beta$ is called **homogeneous for $\gamma$** if there exists a color $c$ such that $\gamma ij = c$ for all $i, j \in \beta$ such that $i < j$. We now define the proposition RA as follows:

RA := For every additive coloring into a finite semigroup there exists an infinite and homogeneous boolean sequence.

Given a sequence $\sigma$ and numbers $i < j$, we write $\sigma_i^j$ for the **substring of $\sigma$** that starts at position $i$ and ends at position $j - 1$ (i.e., position $i$ is inclusive and position $j$ is exclusive). We realise the notation with a polymorphic function $\forall A.\ A^{\omega} \to \mathsf{N} \to \mathsf{N} \to A^{+}$ that yields $\sigma_i^j$ for $i < j$.

Let $\Gamma$ be a finite semigroup. A sequence $\sigma$ over $\Gamma$ may be represented as the additive coloring $\lambda ij.\ \mathsf{C}(\sigma_i^j)$, and the relevant part (i.e., $i < j$) of an additive coloring $\gamma$ into $\Gamma$ may be represented as the sequence $\lambda n.\gamma n(Sn)$.

A factorisation $\tau$ of a sequence $\sigma$ may be represented as the infinite set of the starting positions of the factors $\tau 1, \tau 2, \dots$ in $\sigma$. We represent this set as an infinite boolean sequence. By Fact 2.3 we can obtain for a factorisation a corresponding infinite boolean sequence, and for an infinite boolean sequence a corresponding factorisation.

An element $a$ of a semigroup is **idempotent** if $a + a = a$. It is well-known [13] that for every element $a$ of a finite semigroup there exists a number $n$ such that $n \cdot a$ is idempotent ($n \cdot a$ is notation for the sum $a + \cdots + a$ with $n$ summands).

**Fact 3.3** Let $\sigma$ be a sequence over a finite semigroup $\Gamma$ that has a Ramseyan factorisation. Then $\sigma$ has a Ramseyan factorisation with an idempotent color.

**Proof** Let $\tau$ be a Ramseyan factorisation of $\sigma$. Since $\Gamma$ is finite, there exists some number $n$ such that $n \cdot \mathsf{C}(\tau 1)$ is idempotent. Since all factors $\tau 1, \tau 2, \tau 3, \ldots$ have identical color, we obtain an idempotent Ramseyan factorisation of $\sigma$ by successively merging $n$ adjacent factors of $\tau$ into a single factor. ■

**Fact 3.4** Let $\sigma$ be a sequence over a finite semigroup $\Gamma$. Then $\sigma$ has a Ramseyan factorisation if and only if there exists an infinite boolean sequence that is homogeneous for $\lambda ij.\, \mathsf{C}(\sigma_i^j)$.

**Proof** Let $\sigma$ have an Ramseyan factorisation. By Fact 3.3 we have a Ramseyan factorisation $\tau$ for $\sigma$ that has an idempotent color $c$. Now the infinite boolean sequence representing $\tau$ satisfies the claim.

Let $\beta$ be an infinite boolean sequence that is homogeneous for $\lambda ij.\, \mathsf{C}(\sigma_i^j)$. Then the factorisation represented by $\beta$ is a Ramseyan factorisation of $\sigma$. ■

**Fact 3.5** RA implies RF.

**Proof** Assume RA and let $\sigma$ be a sequence over a finite semigroup $\Gamma$. By Fact 3.4 it suffices to show that there is an infinite boolean sequence that is homogeneous for $\lambda ij.\, \mathsf{C}(\sigma_i^j)$. This follows with RA since $\lambda ij.\, \mathsf{C}(\sigma_i^j)$ is an additive coloring into $\Gamma$. ■

**Fact 3.6** RF implies RA.

**Proof** Assume RF and let $\gamma$ be an additive coloring into a finite semigroup $\Gamma$. We show that there exists an infinite boolean sequence $\beta$ that is homogeneous for $\gamma$. We consider the sequence $\sigma n := \gamma n(Sn)$. By RF and Fact 3.4 there exist an infinite $\beta$ that is homogeneous for $\lambda ij.\mathsf{C}(\sigma_i^j)$. The claim follows since $\gamma ij = \mathsf{C}(\sigma_i^j)$ for all $i < j$. ■

# 4 Minimal S1S

We consider a minimal variant of S1S that has no first-order variables. Full S1S with both kinds of variables reduces to the minimal variant of S1S we consider. We shall use the shorthand **S1S$_0$** for minimal S1S. Other variants of S1S not using first-order variables appear in [2, 18].

We start with a finite type $\mathsf{V}$ of **variables** and formalise the syntax of minimal S1S with an inductive type of **formulas**:

$$\varphi, \psi \ ::= \ X \lhd Y \mid X \subseteq Y \mid \varphi \wedge \psi \mid \neg \varphi \mid \exists X. \varphi \qquad (X, Y : \mathsf{V})$$

Informally speaking, variables range over sets represented as boolean sequences. A formula $X \lhd Y$ says that there are numbers $m < n$ such that $m \in X$ and $n \in Y$. Moreover, a formula $X \subseteq Y$ says that $X$ is a subset of $Y$.

We now formally define the **AS semantics** of $\mathrm{S1S}_0$. An **interpretation** is a sequence over $2^V$ (the finite type containing all sets of variables). We write $\sigma_X$ for the boolean sequence such that $\forall n.\ \sigma_X n = \mathsf{true} \leftrightarrow X \in \sigma n$. Note that $\sigma_X$ represents the set for the variable X. We define interpretations as sequences so that we can translate a formula into an automaton that accepts exactly the sequences satisfying the formula. The letter $\sigma$ will range over interpretations in the following.

We define the satisfaction relation $\sigma \models \varphi$ between interpretations and formulas by recursion on formulas such that the following equivalences trivially hold:

$$\begin{aligned}
\sigma \models X{\lhd}Y &\leftrightarrow \exists m n.\ m < n \land m \in \sigma_X \land n \in \sigma_Y \\
\sigma \models X{\subseteq}Y &\leftrightarrow \forall n.\ n \in \sigma_X \to n \in \sigma_Y \\
\sigma \models \varphi{\land}\psi &\leftrightarrow \sigma \models \varphi \land \sigma \models \psi \\
\sigma \models \neg\varphi &\leftrightarrow \neg(\sigma \models \varphi) \\
\sigma \models \exists X.\varphi &\leftrightarrow \exists \tau.\ \tau \models \varphi \land \sigma \approx_X \tau
\end{aligned}$$

The notation $\sigma \approx_X \tau$ stands for $\forall Z.\ Z = X \lor \sigma_Z \equiv \tau_Z$ and says that $\sigma$ and $\tau$ agree for all variables but possibly X.

We define the **UP semantics** for $\mathrm{S1S}_0$. Everything stays as it is except that all sequences over $2^V$ are replaced with UP sequences over $2^V$. This is also the case for the existentially quantified sequence $\tau$. Recall that a UP sequence is a *pair* of two strings $x$ and $y$ that is interpreted as the sequence $xy^\omega$. We will write $xy^\omega \models_{\mathsf{UP}} \varphi$ for the satisfaction relation of $\mathrm{S1S}_0$ with UP semantics.

## 5 Translation to Abstract Automata

Many aspects of the translation of formulas to automata can be explained without knowing the details of automata. We will therefore work with an abstract type of **automata** in this section. The letters A and B will range over automata of this type, and the letters $\sigma$ and $\tau$ will range over sequences over $2^V$. We assume an **acceptance relation** $\sigma \models A$ between sequences and automata. We read $\sigma \models A$ as $\sigma$ **satisfies** $A$ or as $A$ **accepts** $\sigma$. We also assume functions $\mathsf{A}^{\subseteq}_{XY}$, $\mathsf{A}^{\lhd}_{XY}$, $A \cap B$, $\overline{A}$, and $\exists_X A$ on automata mimicking the constructors for formulas. We refer to these functions as *operations* and assume they have the following properties.

1. $\sigma \models \mathsf{A}^{\lhd}_{XY} \leftrightarrow \exists m n.\ m < n \land m \in \sigma_X \land n \in \sigma_Y$
2. $\sigma \models \mathsf{A}^{\subseteq}_{XY} \leftrightarrow \forall n.\ n \in \sigma_X \to n \in \sigma_Y$
3. $\sigma \models A \cap B \leftrightarrow \sigma \models A \land \sigma \models B$
4. $\sigma \models A \to \sigma \models \overline{A} \to \bot$
5. $\sigma \models A \lor \sigma \models \overline{A}$
6. $\sigma \models \exists_X A \leftrightarrow \exists \tau.\ \tau \models A \land \sigma \approx_X \tau$

Note that the specification of complement automata deviates from the specification of the other operations. It consists of two assumptions (4) and (5) we call **disjointness** and **exhaustiveness**. The specification of complement automata with disjointness and exhaustiveness rather than a single equivalence is necessitated by the constructive analysis.

**Fact 5.1** $\sigma \models \overline{A} \leftrightarrow \sigma \not\models A$ and $\sigma \models A \vee \sigma \not\models A$.

Together, the two statements of Fact 5.1 are equivalent to disjointness and exhaustiveness of complement automata. However, the first statement of Fact 5.1 does not suffice for exhaustiveness constructively.

We now define a function $\alpha$ translating formulas into automata:

$$\begin{aligned} \alpha(X \lhd Y) &= A^{\lhd}_{XY} & \qquad \alpha(\varphi \wedge \psi) &= \alpha(\varphi) \cap \alpha(\psi) \\ \alpha(X \subseteq Y) &= A^{\subseteq}_{XY} & \alpha(\neg\varphi) &= \overline{\alpha(\varphi)} \\ & & \alpha(\exists X.\varphi) &= \exists_X(\alpha(\varphi)) \end{aligned}$$

**Fact 5.2** Let automata and operations satisfying the assumptions stated above be given. Then $\sigma \models \varphi \leftrightarrow \sigma \models \alpha(\varphi)$. Thus formala satisfaction satisfies XM and satisfiability of formulas is decidable if satisfiability of automata is decidable.

# 6 Büchi Automata

We now consider NFAs with Büchi acceptance. It turns out that all operations but complement can be defined and verified constructively following familiar ideas. Care must be taken with the formulation of Büchi acceptance. We require that accepting states are visited infinitely often. Constructively, this is weaker than requiring that a single accepting state is visited infinitely often, as it is sometimes done in the literature [18]. Choosing the weak version is of particular importance for the complement operation, which we will consider in a later section.

The usual method for deciding satisfiability of Büchi acceptance (non-emptiness problem) can be verified constructively. In fact, the method yields a satisfying UP sequence in case the NFA is satisfiable.

As it comes to UP sequences and UP semantics, only the operation for existential quantification needs special attention. We introduce the notion of a match to deal with this issue.

We formalise **NFAs** (nondeterministic finite automata) over a finite type $\Sigma$ called **input alphabet** as tuples consisting of a finite type of **states**, a decidable **transition relation**, and decidable predicates identifying **initial** and **accepting** states.

Let $A$ be an NFA over $\Sigma$ with state type $Q$. A sequence $\rho$ over $Q$ **admits** a sequence $\sigma$ over $\Sigma$ if $(\rho n, \sigma n, \rho(Sn))$ is a transition of $A$ for all $n$. A **run on $\sigma$** is a sequence $\rho$ over $Q$ that starts with an initial state and admits $\sigma$. A run is **accepting** if it passes infinitely often through accepting states.

An NFA $A$ over $\Sigma$ **accepts** a sequence $\sigma$ over $\Sigma$ if $A$ has an accepting run on $\sigma$. We write $\sigma \models A$ if $A$ accepts $\sigma$. We also say that $\sigma$ satisfies $A$ if $A$ accepts $\sigma$, and that $A$ is **satisfiable** if $A$ accepts some sequence.

Recall that the type theory we are working in does not provide functional extensionality. This does not hurt since automata access sequences only pointwise.

**Fact 6.1** One can define a function that given two variables $X$ and $Y$ yields an NFA $\mathsf{A}^{\subseteq}_{XY}$ over $2^V$ such that $\sigma \models \mathsf{A}^{\subseteq}_{XY}$ if and only if $\forall n.\ n \in \sigma_X \to n \in \sigma_Y$.

**Fact 6.2** One can define a function that given two variables $X$ and $Y$ yields an NFA $\mathsf{A}^{\triangleleft}_{XY}$ over $2^V$ such that $\sigma \models \mathsf{A}^{\triangleleft}_{XY}$ if and only if $\exists mn.\ m < n \wedge m \in \sigma_X \wedge n \in \sigma_Y$.

**Fact 6.3** One can define a function that given two strings $x$ and $y$ over $\Sigma$ yields an NFA $\mathsf{A}_{xy^\omega}$ over $\Sigma$ accepting exactly the sequences equivalent to $xy^\omega$.

Let $A$ be an NFA over $\Sigma$ with state type $Q$. Given a string $u$ over $Q$ and a string $x$ over $\Sigma$, we say that $u$ is a **path on** $x$ if $x$ provides symbols yielding transitions between adjacent states of $u$. We say that $A$ **accepts** $x$ if $A$ has a path on $x$ starting with an initial state and ending with an accepting state. We define two decidable predicates:

$$\begin{aligned} p \Rightarrow^x_A q &:= A \text{ has a path on } x \text{ from } p \text{ to } q. \\ p \Rrightarrow^x_A q &:= A \text{ has a path on } x \text{ from } p \text{ to } q \text{ passing through} \\ &\phantom{:=}\ \text{an accepting state before the last position.} \end{aligned}$$

A pair $(x, y)$ is a **match** of $A$ if there exist states $p$ and $q$ such that $p$ is initial, $p \Rightarrow^x_A q$, and $q \Rrightarrow^y_A q$.

**Fact 6.4**

1. If $(x, y)$ is a match of an NFA $A$, then $xy^\omega$ satisfies $A$.
2. An NFA is satisfiable if and only if it has a match.
3. It is decidable whether an NFA has a match.
4. If an NFA has a match, one can obtain a match.
5. It is decidable whether an NFA is satisfiable.
6. In case an NFA $A$ is satisfiable, one can obtain a match of $A$.

**Fact 6.5** One can define a function that given two NFAs $A$ and $B$ over $\Sigma$ yields an NFA $A \cup B$ over $\Sigma$ with $\sigma \in A \cup B \leftrightarrow \sigma \models A \vee \sigma \models B$.

**Proof** Define $A \cup B$ as the disjoint union of $A$ and $B$. ■

**Fact 6.6** One can define a function that given two NFAs $A$ and $B$ over $\Sigma$ yields an NFA $A \cap B$ over $\Sigma$ such that:

1. $\sigma \in A \cap B \leftrightarrow \sigma \models A \wedge \sigma \models B$.
2. Every match of $A \cap B$ is a match of $A$ and $B$.

**Proof** Let $A$ and $B$ be two NFAs with state types $Q_A$ and $Q_B$. For $A \cap B$ we use the product $\mathsf{2} \times Q_A \times Q_B$ as state type. A state $(b, p, q)$ is initial if $b = \mathsf{false}$, $p$ is an initial state of $A$, and $q$ is an initial state of $B$. A state $(b, p, q)$ is accepting if $b = \mathsf{true}$ and $p$ is an accepting state of $A$. The trick is to switch to $\mathsf{false}$ when leaving an accepting state of $A$, and to $\mathsf{true}$ when leaving an accepting state of $B$. ■

**Fact 6.7** $\lambda xyA.\ xy^\omega \models A$ is decidable.

**Proof (from [3])** Let $x$, $y$ and $A$ be given. Fact 6.3 provides an NFA $\mathsf{A}_{xy^\omega}$ that accepts exactly the sequences equivalent to $xy^\omega$. Now $xy^\omega \models A$ iff $A \cap \mathsf{A}_{xy^\omega}$ is satisfiable. The claim follows with Fact 6.4. ■

**Fact 6.8** If $xy^\omega \models A$, then $A$ has a match $(u, v)$ such that $xy^\omega \equiv uv^\omega$.

**Proof** Let $xy^\omega \models A$. By Fact 6.3 there is an automaton $\mathsf{A}_{xy^\omega}$ that accepts exactly the sequences equivalent to $xy^\omega$. By Fact 6.4 we obtain a match of $A \cap \mathsf{A}_{xy^\omega}$. The claim follows with Fact 6.6. ■

**Fact 6.9** One can define a function that given a variable $X$ and an NFA $A$ over $2^V$ yields an NFA $\exists_X A$ over $2^V$ satisfying the following conditions:

1. If $\sigma \models A$, then $\sigma \models \exists_X A$.
2. If $\sigma \models \exists_X A$, then there exists $\tau \models A$ such that $\tau \approx_X \sigma$.
3. If $xy^\omega \models \exists_X A$, then there exist $u$ and $v$ such that $uv^\omega \models A$ and $uv^\omega \approx_X xy^\omega$.

**Proof** Let $X$ and $A$ be given. We obtain $\exists_X A$ from $A$ by adding transitions: for every transition $(p, a, q)$ of $A$ and every set $b : 2^V$ such that $a \cup \{X\} = b \cup \{X\}$, $\exists_X A$ contains the transition $(p, b, q)$. The first two claims are easily verified.

Let $xy^\omega \models \exists_X A$. Fact 6.8 yields a match $(u, v)$ of $\exists_X A$ such that $xy^\omega \equiv uv^\omega$. We can now obtain a match $(w, z)$ of $A$ such that $uv^\omega \approx_X wz^\omega$. ■

# 7 NFAs for Ramseyan Factorisations

Given a finite semigroup $\Gamma$, we can construct an NFA accepting exactly the sequences over $\Gamma$ that have a Ramseyan factorisation. The construction exemplifies ideas that will also appear in the construction of complement automata. The result itself will be used for the result about the existence of complement automata.

For an NFA we write $q \xrightarrow{a} p$ to say that $(q, a, p)$ is a transition.

**Fact 7.1** For every color $c$ of a finite semigroup $\Gamma$ one can construct an NFA accepting exactly the strings over $\Gamma$ that have color $c$. The NFA can be constructed such that it has a single intial state with no incoming transitions.

**Proof** Let $\Gamma$ be a finite semigroup and $c : \Gamma$. We construct an NFA with $Q := 1 + \Gamma$ where 1 serves as initial state and $c$ serves as accepting state. There are transitions $1 \xrightarrow{a} a$ and $b \xrightarrow{a} b + a$ for every $a, b : \Gamma$. ■

**Fact 7.2** Given two colors $c$ and $d$ of a finite semigroup $\Gamma$, one can construct an NFA accepting all sequences over $\Gamma$ that have a factorisation $\tau$ such that $C(\tau 0) = c$ and $C(\tau n) = d$ for all $n \geqslant 1$.

**Proof** Let $\Gamma$ be a finite semigroup and $c, d : \Gamma$. Let $A$ and $B$ be the NFAs for $c$ and $d$ we obtain with Fact 7.1. We start with the disjoint union of $A$ and $B$ as it comes to states and transitions. The new initial state $q_1$ is the initial state of $A$, and the new accepting state $q_2$ is the initial state of $B$. For every transition $q \xrightarrow{a} p$ to an accepting state of $A$ we add the transition $q \xrightarrow{a} q_2$, and for every transition $q \xrightarrow{a} p$ to an accepting state of $B$ we add the transition $q \xrightarrow{a} q_2$. ■

**Fact 7.3** For every finite semigroup $\Gamma$ one can construct an NFA accepting exactly the sequences over $\Gamma$ that have a Ramseyan factorisation.

**Proof** For every pair $(a, b)$ of colors we obtain an automaton as specified by Fact 7.2. The union of these automata (Fact 6.5) satisfies the claimed property. ■

## 8 Complement Operation

We fix an NFA $A$ over $\Sigma$ with state type $Q$. We will construct a complement NFA following Büchi's construction [4, 19]. We carefully arrange the technical details of the construction such that the required properties follow constructively and the connection with Ramseyan factorisations becomes clear.

The letters $x$ and $y$ will range over strings in $\Sigma^+$, and the letters $p$ and $q$ will range over states in $Q$. Moreover, the letters $\sigma$, $\tau$, and $\rho$ will range over sequences over $\Sigma$, $\Sigma^+$, and $Q$, respectively.

We say that **$\rho$ is an accepting quasi-run on $\tau$** if $\rho$ starts with an initial state, satisfies $\rho n \Rightarrow_A^{\tau n} \rho(Sn)$ for all $n$, and satisfies $\rho n \Rrightarrow_A^{\tau n} \rho(Sn)$ for infinitely many $n$.

**Fact 8.1** Let $\tau$ be a factorisation of $\sigma$. Then $A$ accepts $\sigma$ if and only if there exists an accepting quasi-run on $\tau$.

We define a finite type $\Gamma$ and a function $\gamma : \Sigma^+ \to \Gamma$:

$$\begin{aligned}\Gamma &:= 2^{Q\times Q} \times 2^{Q\times Q} \\ \gamma x &:= (\{\,(p,q) \mid p \Rightarrow_A^x q\,\}, \{\,(p,q) \mid p \Rrightarrow_A^x q\,\})\end{aligned}$$

Note that $\gamma$ can be defined constructively since $p \Rightarrow_A^x q$ and $p \Rrightarrow_A^x q$ are decidable predicates and $Q$ is a finite type. We call the elements of $\Gamma$ **colors** and $\gamma x$ the **color of** $x$. The letters $V$ and $W$ will range over colors.

**Fact 8.2** Let $\tau$ and $\tau'$ be sequences over $\Sigma^+$ such that $\gamma(\tau n) = \gamma(\tau' n)$ for all $n$. Then $\tau$ and $\tau'$ admit the same accepting quasi-runs.

We now define an operation on colors turning $\Gamma$ into a finite semigroup and $\gamma$ into a semigroup morphism.

$$\begin{aligned}V + W := (&\{\,(p,q) \mid \exists r.\ (p,r) \in \pi_1 V \wedge (r,q) \in \pi_1 W\,\}, \\ &\{\,(p,q) \mid \exists r.\ (p,r) \in \pi_2 V \wedge (r,q) \in \pi_1 W \vee \\ &\qquad\qquad (p,r) \in \pi_1 V \wedge (r,q) \in \pi_2 W\,\})\end{aligned}$$

**Fact 8.3** $V_1 + (V_2 + V_3) = (V_1 + V_2) + V_3$ and $\gamma(xy) = \gamma x + \gamma y$.

A **kind** $V/W$ is a pair of two colors $V$ and $W$. We say that a sequence $\sigma$ **has kind** $V/W$ if $\sigma$ has a factorisation $\tau$ that has color $V$ at position 0 and color $W$ at all positions $n \geqslant 1$. We say that a kind $V/W$ **is compatible with** $A$ if $A$ accepts some sequence of kind $V/W$. Note that there may be sequences having more than one kind.

**Fact 8.4** If $V/W$ is compatible with $A$, then $A$ accepts all sequences of kind $V/W$.

**Proof** Let $\sigma$ and $\sigma'$ be sequences of kind $V/W$ and let $\sigma \models A$. We show $\sigma' \models A$. Let $\tau$ and $\tau'$ be $V/W$-factorisations of $\sigma$ and $\sigma'$ respectively. By Facts 8.1 and 8.2 we have an accepting quasi-run $\rho$ on $\tau$ and $\tau'$. Now $\sigma' \models A$ by Fact 8.1. ■

**Fact 8.5** A UP sequence $xy^\omega$ has kind $\gamma x/\gamma y$.

**Fact 8.6** Assuming $\mathsf{RF}$, every sequence over $\Sigma$ has a kind.

**Proof** Let $\sigma$ be a sequence over $\Sigma$. By $\mathsf{RF}$ and Fact 8.3 we obtain a Ramseyan factorisation $\mu : (\Gamma^+)^\omega$ of $\lambda n.\gamma(\sigma n)$. From $\mu$ we obtain a factorisation $\tau$ of $\sigma$ such that $\gamma(\tau n) = \mathsf{C}(\mu n)$ for all $n$. Thus $\sigma$ has kind $\gamma(\tau 0)/\gamma(\tau 1)$. ■

**Fact 8.7** One can define a function that given a kind $V/W$ yields an NFA $VW^\omega$ accepting exactly the sequences of kind $V/W$.

**Proof** The construction is similar to the construction given for Fact 7.2. ■

**Fact 8.8** It is decidable whether a kind is compatible with $A$.

**Proof** $V/W$ is compatible with $A$ if and only if the NFA $VW^\omega \cap A$ is satisfiable. Thus the claim follows with Fact 6.4. ■

**Fact 8.9** One can construct an NFA $\overline{A}$ accepting exactly the sequences that have a kind incompatible with $A$.

**Proof** We construct $\overline{A}$ as the union of the NFAs for the kinds incompatible with $A$. The construction is possible due to Facts 6.5, 8.8, and 8.7. ■

**Theorem 8.10 (Complement)**

1. No sequence is accepted by both $A$ and $\overline{A}$.
2. Every sequence that has a kind is accepted by either $A$ or $\overline{A}$.
3. Every UP sequence is accepted by either $A$ or $\overline{A}$.
4. Assuming $\mathsf{RF}$, every sequence is accepted by either $A$ or $\overline{A}$.

**Proof** Follows with Facts 8.9, 8.8, 8.4, 8.5, and 8.6. ■

# 9 Main Results so Far

We now have a translation of $S1S_0$ formulas to NFAs for which we have shown many properties. We combine the results obtained so far into main results for $S1S_0$ distinguishing between UP semantics and AS semantics.

**Theorem 9.1 ($S1S_0$, AS semantics)** Assuming $\mathsf{RF}$, we have the following:

1. The translation of formulas to automata is correct for all sequences.
2. $\lambda\sigma\varphi.\ \sigma \models \varphi$ satisfies XM.
3. $\lambda\varphi.\ \exists\sigma.\ \sigma \models \varphi$ is decidable.

**Proof** Follows with Facts 6.1, 6.2, 6.5, 6.6, and 6.9, Theorem 8.10, and Fact 6.4. ■

**Theorem 9.2 ($S1S_0$, UP semantics)**

1. The translation of formulas to NFAs is correct for UP sequences.
2. $\lambda xy\varphi.\ xy^\omega \models_{\mathsf{UP}} \varphi$ is decidable.
3. $\lambda\varphi.\ \exists xy.\ xy^\omega \models_{\mathsf{UP}} \varphi$ is decidable.
4. If a formula is UP satisfiable, one can obtain a satisfying UP sequence.
5. Assuming $\mathsf{RF}$, $\exists\sigma.\ \sigma \models \varphi$ if and only if $\exists xy.\ xy^\omega \models_{\mathsf{UP}} \varphi$.

**Proof** Follows with Facts 6.1, 6.2, 6.5, 6.6, and 6.9, Theorem 8.10, and Facts 6.4 and 6.7. ■

For the rest of the paper we will be concerned with results showing that RF is a necessary condition for AS semantics. We now show that RF is equivalent to the existence of complement automata. We also show that RF is equivalent to the agreement of UP equivalence with AS equivalence of automata.

Given an NFA $A$ over $\Sigma$, we call an NFA $A'$ over $\Sigma$ a **complement of $A$** if every sequence $\sigma$ over $\Sigma$ satisfies (1) $\sigma \models A \to \sigma \models A' \to \perp$ (*disjointness*) and (2) $\sigma \models A \vee \sigma \models A'$ (*exhaustiveness*).

Given two NFAs over $\Sigma$, we write $A \equiv B$ if $A$ and $B$ accept the same sequences, and $A \equiv_{\mathsf{UP}} B$ if $A$ and $B$ accept the same UP sequences. We define the following propositions:

AC := For every $\Sigma$, every NFA over $\Sigma$ has a complement.

AU := For every $\Sigma$ and all $A$ and $B$ over $\Sigma$, $A \equiv_{\mathsf{UP}} B$ implies $A \equiv B$.

AX := For every $\Sigma$ and all $\sigma$ and $A$ over $\Sigma$, either $\sigma \models A$ or $\sigma \not\models A$.

**Fact 9.3** AC implies AX.

**Theorem 9.4** RF and AC are equivalent.

**Proof** RF $\to$ AC follows with Theorem 8.10. For the other direction, assume AC and let $\Gamma$ be a semigroup. By Fact 7.3 we have an NFA $A$ accepting exactly the sequences over $\Gamma$ that have a Ramseyan factorisation. Let $A'$ be a complement of $A$ and let $\sigma$ be a sequence over $\Gamma$. We show that $\sigma$ has a Ramseyan factorisation. Case analysis over $\sigma \models A \vee \sigma \models A'$. If $\sigma \models A$, then $\sigma$ has a Ramseyan factorisation by the construction of $A$. If $\sigma \models A'$, Fact 6.4 gives us a UP sequence $xy^\omega$ accepted by $A'$. By disjointness of $A'$ we have $xy^\omega \not\models A$. Contradiction since every UP sequence has a Ramseyan factorisation (Fact 3.1). ■

**Theorem 9.5** AC and AU are equivalent.

**Proof** AC $\to$ AU from [6]. Assume AC and let $A \equiv_{\mathsf{UP}} B$. Let $\sigma \models A$. By symmetry it suffices to show $\sigma \models B$. Let $B'$ be a complement of $B$. Case analysis using Fact 6.4. If $A \cap B'$ is unsatisfiable, we have $\sigma \models B$ by exhaustiveness of $B$ and $B'$. Otherwise, we have a UP sequence $xy^\omega$ accepted by $A$ and $B'$. Thus $xy^\omega \models B$ since we assumed $A \equiv_{\mathsf{UP}} B$. Contradiction with the disjointness of $B$ and $B'$.

AU $\to$ AC. Assume AU and let $A$ be an NFA. We show that $\overline{A}$ is a complement for $A$. By Theorem 8.10 we know that $\overline{A}$ is disjoint with $A$. Let $A_{\Sigma^\omega}$ be an NFA accepting all sequences. By Theorem 8.10 we have $A \cup \overline{A} \equiv_{\mathsf{UP}} A_{\Sigma^\omega}$. By the assumption we have $A \cup \overline{A} \equiv A_{\Sigma^\omega}$. Thus $\overline{A}$ is exhaustive for $A$. ■

# 10 Full S1S

We now define **S1S** (full S1S) and give a reduction to $\mathrm{S1S}_0$ (minimal S1S). With the reduction, the results shown for $\mathrm{S1S}_0$ carry over to S1S. One reason for considering S1S in addition to $\mathrm{S1S}_0$ in this paper is that it better supports the codings needed for the proof that FX implies RF.

Given two finite types $V_1$ and $V_2$ of **variables**, we obtain the **formulas** of S1S with an inductive type:

$$\varphi, \psi \ ::= \ x < y \mid x \in X \mid \varphi \wedge \psi \mid \neg\varphi \mid \exists x.\varphi \mid \exists X.\varphi \qquad (x, y : V_1)\ (X, Y : V_2)$$

The variables $x$ from $V_1$ range over numbers and are called **first-order variables**. The variables $X$ from $V_2$ range over sets of numbers represented as boolean sequences and are called **second-order variables**.

Formally, we define the **AS semantics** of S1S with **interpretations** $I$ consisting of two functions $V_1 \to \mathsf{N}$ and $V_2 \to 2^\omega$. The **satisfaction relation** $I \models \varphi$ is defined as one would expect.

The reduction of S1S to $\mathrm{S1S}_0$ represents first-order variables as second-order variables that are constrained to singleton sets. $\mathrm{S1S}_0$ can express a singleton constraint as follows:

$$\mathsf{sing}\, X \ := \ \neg(X \lhd X) \wedge \exists Y.\, X \lhd Y$$

Note that $\neg(X \lhd X)$ is satisfied if $X$ has at most one element, and that $\exists Y.\, X \lhd Y$ is satisfied if $X$ has at least one element. If $X$ and $Y$ are the singleton variables for two first-order variables $x$ and $y$, then $x < y$ can be expressed as $X \lhd Y$ and $x \in Z$ can be expressed as $X \subseteq Z$.

**Fact 10.1** Consider S1S with variable types $V_1$ and $V_2$ and $\mathrm{S1S}_0$ with variable type $V_1 + V_2 + 1$. Then one can obtain for every interpretation $I$ and every formula $\varphi$ of S1S an interpretation $\hat{I}$ and a formula $\hat{\varphi}$ of $\mathrm{S1S}_0$ such that $I \models \varphi \leftrightarrow \hat{I} \models \hat{\varphi}$. Moreover, for every interpretation $\sigma$ of $\mathrm{S1S}_0$ that interprets variables from $V_1$ as singletons, one can obtain an interpretation $\tilde{\sigma}$ of S1S such that $\sigma \models \hat{\varphi} \leftrightarrow \tilde{\sigma} \models \varphi$.

**Proof** We obtain $\hat{\varphi}$ from $\varphi$ by constraining every variable from $V_1$ to a singleton set. The *extra variable* provided by $V_1 + V_2 + 1$ provides for the variable $Y$ in the quantification $\exists Y.\, x \lhd Y$ employed by the singleton constraint.

We obtain $\hat{I}$ from $I$ by representing numbers as singleton sets. The value for the extra variable does not matter since the extra variable does not occur free in $\hat{\varphi}$.

Finally, for every interpretation $\sigma$ of $\mathrm{S1S}_0$ that interprets variables from $V_1$ as singletons, one can obtain an interpretation $\tilde{\sigma}$ of S1S by assigning to the variables from $V_1$ the numbers provided by the singleton sets. Obtaining the unique element of a singleton set represented as a boolean sequence requires constructive choice. ■

**Theorem 10.2 (S1S, AS semantics)** Assuming $\mathsf{RF}$, we have the following:

1. $\lambda I\varphi.\, I \models \varphi$ satisfies XM.
2. $\lambda\varphi.\, \exists I.\, I \models \varphi$ is decidable.

**Proof** Follows with Theorem 9.1 and Fact 10.1. ■

We now formally define the proposition $\mathsf{FX}$:

$$\mathsf{FX} := \forall \mathsf{V}_1 \mathsf{V}_2\, I\, \varphi.\; I \models \varphi \vee I \not\models \varphi$$

It is understood that in $\mathsf{FX}$ the interpretation I and the formula $\varphi$ are taken over the types $\mathsf{V}_1$ and $\mathsf{V}_2$, which are (notationally implicit) parameters of S1S.

**Corollary 10.3** $\mathsf{RF}$ implies $\mathsf{FX}$.

The reduction of S1S to $\mathrm{S1S}_0$ also works for UP semantics. A **UP interpretation** of S1S consists of two functions $\mathsf{V}_1 \to \mathsf{N}$ and $\mathsf{V}_2 \to 2^+ \times 2^+$. We write $I \models_{\mathsf{UP}} \varphi$ for the satisfaction relation for UP semantics.

**Theorem 10.4 (S1S, UP semantics)**

1. $\lambda I\varphi.\, I \models_{\mathsf{UP}} \varphi$ is decidable.
2. $\lambda\varphi.\, \exists I.\, I \models_{\mathsf{UP}} \varphi$ is decidable.
3. If a formula is UP satisfiable, one can obtain a satisfying UP interpretation.
4. Given $\mathsf{RF}$, satisfiable formulas are UP satisfiable.

**Proof** Fact 10.1 can be adapted so that $\hat{I}$ is a UP interpretation if I is a UP interpretation, and $\tilde{\sigma}$ is a UP interpretation if $\sigma$ is a UP interpretation. Now Theorems 9.2 and 9.1 yield the claims. ■

## 11 Ramseyan Pigeonhole Principle

We now prepare the proofs of the implications $\mathsf{FX} \to \mathsf{RF}$ and $\mathsf{AX} \to \mathsf{RF}$. For this, we will define a proposition $\mathsf{RP}$ we call **Ramseyan pigeonhole principle**. We will show that $\mathsf{RP}$ is equivalent to $\mathsf{RF}$.

We will also consider a variant $\mathsf{RP}^\mathsf{c}$ of $\mathsf{RP}$ that can be obtained from $\mathsf{RP}$ by applying double negation and de Morgan rules. We will show that $\mathsf{RP}^\mathsf{c}$ holds constructively. Now the trick consists in using $\mathsf{FX}$ and $\mathsf{AX}$ to prove the equivalence $\mathsf{RP} \leftrightarrow \mathsf{RP}^\mathsf{c}$. Since $\mathsf{RP}^\mathsf{c}$ is obtained from $\mathsf{RP}$ by double negation and de Morgan rules, the special instances of excluded middle present in $\mathsf{FX}$ and $\mathsf{AX}$ will suffice to show the equivalence $\mathsf{RP} \leftrightarrow \mathsf{RP}^\mathsf{c}$. For this it is important that $\mathsf{RP}$ and $\mathsf{RP}^\mathsf{c}$ can be expressed with the satisfaction relations for formulas and automata.

RP is based on a relation for sequences over finite semigroups appearing as merging relation in the literature [5, 12, 13, 15, 16]. The relation is used in the literature to prove Ramseyan properties similar to RA and RF using excluded middle.

Given a sequence $\sigma$ over a finite semigroup, we define the **merging relation for $\sigma$** as follows:

$$i \sim_\sigma j \;:=\; \exists k.\, i < k \land j < k \land \mathsf{C}(\sigma_i^k) = \mathsf{C}(\sigma_j^k)$$

The numbers $i$ and $j$ act as positions of $\sigma$. We say that **$i$ merges with $j$ in $\sigma$** if $i \sim_\sigma j$, and that **$i$ merges with $j$ at $k$ in $\sigma$** if $i, j < k$ and $\mathsf{C}(\sigma_i^k) = \mathsf{C}(\sigma_j^k)$.

One easily checks that merging is an equivalence relation using the following fact.

**Fact 11.1** Let $\sigma$ be a sequence over a finite semigroup. If $i$ merges with $j$ at $k$ in $\sigma$, then $i$ merges with $j$ at $n$ in $\sigma$ for all $n \geqslant k$.

The following fact says that $i \sim_\sigma j$ has only finitely many equivalence classes.

**Fact 11.2** Let $\sigma$ be a sequence over a finite semigroup $\Gamma$, $k \geqslant |\Gamma|$, and $n_0 < \cdots < n_k$. Then there exist numbers $i < j \leqslant k$ with $n_i \sim_\sigma n_j$.

**Proof** Since there are $k+1$ many strings $\sigma_{n_0}^{n_k+1}, \dots, \sigma_{n_k}^{n_k+1}$ and at most $k$ many colors, there are two numbers $i < j \leqslant k$ such that $\mathsf{C}(\sigma_{n_i}^{n_k+1}) = \mathsf{C}(\sigma_{n_j}^{n_k+1})$. ■

We now define the propositions RP and $\mathsf{RP}^c$:

$$\begin{aligned} \mathsf{RP}_\sigma &:= \exists i\, \exists^\omega j.\, i \sim_\sigma j & \qquad \mathsf{RP} &:= \forall \Gamma\, \forall \sigma{:}\, \Gamma^\omega.\, \mathsf{RP}_\sigma \\ \mathsf{RP}_\sigma^c &:= \neg \forall i\, \exists k\, \forall j \geqslant k.\, i \not\sim_\sigma j & \qquad \mathsf{RP}^c &:= \forall \Gamma\, \forall \sigma{:}\, \Gamma^\omega.\, \mathsf{RP}_\sigma^c \end{aligned}$$

Note that RP states that every sequence over a finite semigroup has an infinite merging class.

**Fact 11.3** Assuming XM, RP is equivalent to $\mathsf{RP}^c$.

**Proof** By application of de Morgan laws and double negation. ■

**Fact 11.4** $\mathsf{RP}^c$ holds.

**Proof** Let $\sigma$ be a sequence over a finite semigroup $\Gamma$. We assume $\forall i\, \exists k\, \forall j \geqslant k.\, i \not\sim_\sigma j$ and derive a contradiction. We show by induction on $k$ that for every $k$ there are pairwise non-merging numbers $n_0 < \cdots < n_k$ such that $\forall i{<}k\, \forall j{\geqslant}n_k.\, n_i \not\sim_\sigma j$. Contradiction with Fact 11.2. ■

**Fact 11.5** RF implies RP.

**Proof** Assume RF and let $\sigma$ be a sequence over a finite semigroup. We show $\mathsf{RP}_\sigma$. By Fact 3.4 there is an infinite boolean sequence $\beta$ such that $\mathsf{C}(\sigma_i^j)$ is constant for all $i, j \in \beta$ with $i < j$. We fix some $i \in \beta$ and some number $k$. Then there are $j, l \in \beta$ such that $k \leqslant j$ and $i, j < l$. We now have $i \sim_\sigma j$ since $\mathsf{C}(\sigma_i^l) = \mathsf{C}(\sigma_j^l)$. ■

**Fact 11.6** RP implies IP.

**Proof** Assume RP and let $\sigma$ be a sequence over a finite type $\Sigma$. We use $\lambda ab.a$ as semigroup operation for $\Sigma$. By $\mathsf{RP}_\sigma$ there is a number $i$ and infinitely many numbers $j$ merging with $i$. Let $i$ merge with $j$ at $k$. Then $\sigma j = \mathsf{C}(\sigma_j^k) = \mathsf{C}(\sigma_i^k) = \sigma i$. Hence $\sigma i$ occurs infinitely often in $\sigma$. ■

**Fact 11.7** RP implies RF.

**Proof** Assume RP and let $\sigma$ be a sequence over a finite semigroup. We show that $\sigma$ has a Ramseyan factorisation using Fact 3.4.

By $\mathsf{RP}_\sigma$ we have a number $m$ such that $\exists^\omega i.\, m \sim_\sigma i$. Using constructive choice, we obtain a function $f : \mathsf{N} \to \mathsf{N} \times \mathsf{N}$ such that $fn$ yields two numbers $n < i < k$ such that $m$ merges with $i$ at $k$ in $\sigma$ (possible since $\lambda k.\, \exists i < k.\, n < i \wedge \mathsf{C}(\sigma_m^k) = \mathsf{C}(\sigma_i^k)$ is decidable).

We now obtain a strictly monotone function $g : \mathsf{N} \to \mathsf{N}$ such that $m$ merges with $gi$ at $gj$ in $\sigma$ for all $i < j$ (using Fact 11.1).

We consider the sequence $\lambda n.\mathsf{C}(\sigma_m^{gn})$. By IP (Fact 11.6) there exist a color $a$ such that $\exists^\omega n.\, \mathsf{C}(\sigma_m^{gn}) = a$. We now obtain (using Fact 2.3) an infinite boolean sequence $\beta$ such that for all $n$

$$n \in \beta \;\leftrightarrow\; \exists k.\, n = gk \wedge \mathsf{C}(\sigma_m^{gk}) = a$$

Let $i, j \in \beta$ such that $i < j$. By Fact 3.4 it suffices to show $\mathsf{C}(\sigma_i^j) = a$. By the definition of $\beta$ we have $k_i < k_j$ such that $i = g(k_i)$, $j = g(k_j)$, and $\mathsf{C}(\sigma_m^{g(k_j)}) = a$. Now $\mathsf{C}(\sigma_i^j) = \mathsf{C}(\sigma_{g(k_i)}^{g(k_j)}) = \mathsf{C}(\sigma_m^{g(k_j)}) = a$. ■

The proof of Fact 11.7 is complicated by the fact that $\beta$ must be obtained as a computational function in constructive type theory. Similar constructions carried out in classical set theory appear in the literature [5, 13, 15, 16] as part of proofs of properties similar to RF. The properties RP and $\mathsf{RP}^c$ are not made explicit in the literature. Recall that we have given constructive proofs of $\mathsf{RP}^c$ and RP → RF, and that there is a trivial classical proof of RP ↔ $\mathsf{RP}^c$.

**Corollary 11.8** XM implies RF.

Note that RA and RF existentially quantify over functions and that this is not the case for RP. The proof of RP → RF reveals how in a constructive setting one can construct a nontrivial function from existential assumptions for numbers.

## 12 FX implies RP

To show that FX implies RP, we encode RP and $\mathsf{RP}^{\mathsf{c}}$ into S1S and show that the encodings are equivalent. Recall from Section 10 that FX says that satisfaction in S1S satisfies XM. Assuming FX, double negation and de Morgan laws hold in S1S and we can use universal quantification and all boolean connectives.

We choose a finite type $\mathsf{V}_1$ providing at least three distinct first-order variables.

**Fact 12.1** Assume FX and let $\sigma$ be a sequence over a finite semigroup $\Gamma$ and $x, y : \mathsf{V}_1$ be two distinct first-order variables. There is an interpretation $I_\sigma$ and a formula $\varphi_{x\sim y}$ with variable types $\mathsf{V}_1$ and $\mathsf{V}_2 := \Gamma + \Gamma$ such that $I_\sigma[x := i, y := j] \models \varphi_{x\sim y}$ if and only if $i \sim_\sigma j$ for all $i$ and $j$.

**Proof** The formula is given in Figure 1. The sequence $\sigma$ is encoded as usual [13]: There are free second-order variables $X_a$ for all $a : \Gamma$ and the interpretation $I_\sigma$ is defined such that $I_\sigma X_a$ is the boolean sequence containing exactly the positions at which $\sigma$ is $a$.

We provide informal explanations for the formulas in Figure 1 (for readability, we use S1S variables in equations):

- $\varphi_{\mathsf{C}_x^y = c}$ says that the color of $\sigma_x^y$ is $c$. The variables $Y_a$ encode the colors of $\sigma_x^{Sx}, \dots, \sigma_x^y$.
- $\varphi_{\mathsf{unique}}$ says that $z$ can be in at most one $Y_a$.
- $\varphi_{\mathsf{first}}$ says that $\mathsf{C}(\sigma_x^{Sx}) = \sigma x$.
- $\varphi_{\mathsf{step}}$ says that $\mathsf{C}(\sigma_x^{Sz}) = \mathsf{C}(\sigma_x^z) + \sigma z$ for $x < z < y$.
- $\varphi_{\mathsf{last}}$ says that the color of $\sigma_x^y$ is $c$. ■

**Fact 12.2** FX implies RP.

**Proof** Assume FX and let $\sigma$ be a sequence over a finite semigroup. By Fact 11.4 it suffices to show that $\mathsf{RP}_\sigma^{\mathsf{c}}$ entails $\mathsf{RP}_\sigma$. We encode $\mathsf{RP}_\sigma^{\mathsf{c}}$ and $\mathsf{RP}_\sigma$ in S1S using Fact 12.1:

$$\mathsf{RP}_\sigma \leftrightarrow I_\sigma \models \exists x.\ \forall z.\ \exists y \geqslant z.\ \varphi_{x\sim y}$$
$$\mathsf{RP}_\sigma^{\mathsf{c}} \leftrightarrow I_\sigma \models \neg\forall x.\ \exists z.\ \forall y \geqslant z.\ \neg\varphi_{x\sim y}$$

Both encodings can be shown equivalent in S1S using de Morgan laws and double negation. Thus $\mathsf{RP}_\sigma^{\mathsf{c}}$ entails $\mathsf{RP}_\sigma$. ■

Siefkes [17] and Riba [15] show in a classical setting that S1S can encode propositions similar to RA and RP, respectively. In contrast to Riba [15], we use an explicit encoding of propositions $\mathsf{C}(\sigma_i^j) = a$.

$$\begin{aligned}
\varphi_{x\sim y} &:= \exists z.\, x < z \land y < z \land \bigvee_{c:\Gamma} \left(\varphi_{C^z_x=c} \land \varphi_{C^z_y=c}\right) \\
\varphi_{C^y_x=c} &:= \exists_{a:\Gamma} Y_a.\, \varphi_{\mathsf{unique}} \land \varphi_{\mathsf{first}} \land \varphi_{\mathsf{step}} \land \varphi_{\mathsf{last}} \\
\varphi_{\mathsf{unique}} &:= \forall z.\, \bigwedge_{a:\Gamma} \left(z \in Y_a \to \bigwedge_{b\neq a} z \notin Y_b\right) \\
\varphi_{\mathsf{first}} &:= \bigwedge_{a:\Gamma} (x \in X_a \to Sx \in Y_a) \\
\varphi_{\mathsf{step}} &:= \forall z.\, x < z < y \to \bigwedge_{a,b:\Gamma} (z \in Y_a \to z \in X_b \to Sz \in Y_{a+b}) \\
\varphi_{\mathsf{last}} &:= y \in Y_c
\end{aligned}$$

Figure 1: Encoding of $i \sim_\sigma j$ into S1S for Fact 12.1. We write $\varphi(Sx)$ for $\forall x'.\, x < x' \to (\neg\exists y.\, x < y < x') \to \varphi(x')$.

# 13 AX implies RP

We finally show that AX implies RP. Recall from Section 9 that AX says that acceptance by automata satisfies XM. Assuming AX, we will show that RP and $\mathsf{RP}^{\mathsf{c}}$ are equivalent. There is the difficulty that we cannot encode RP and $\mathsf{RP}^{\mathsf{c}}$ into automata since this requires complement automata, which we do not have since we do not have RF. We solve the problem with three predicates that can be encoded into automata without using complement and that suffice to justify the uses of double negation needed for the equivalence proof.

We define the helper predicates as follows, where $\Gamma$ ranges over finite semigroups and $\sigma$ over sequences over $\Gamma$:

$$\begin{aligned}
p_1 &:= \lambda\Gamma\sigma i k.\, \exists j \geqslant k.\, i \sim_\sigma j \\
p_2 &:= \lambda\Gamma\sigma.\, \exists i\, \exists^\omega j.\, i \sim_\sigma j \\
p_3 &:= \lambda\Gamma\sigma i.\, \exists k.\, \neg\exists j \geqslant k.\, i \sim_\sigma j
\end{aligned}$$

Note that $p_2$ is $\lambda\Gamma\sigma.\mathsf{RP}_\sigma$.

**Fact 13.1** Let $p_1$, $p_2$, and $p_3$ satisfy XM. Then RP holds.

**Proof** Let $\sigma$ be a sequence over a finite semigroup $\Gamma$. By Fact 11.4 it suffices to show that $\mathsf{RP}_\sigma$ and $\mathsf{RP}^c_\sigma$ are equivalent. This follows with the assumptions and Fact 2.1:

$$\begin{aligned} & \exists i\,\forall k\,\exists j.\ j \geqslant k \land i \sim_\sigma j \\ \leftrightarrow\ & \neg\forall i.\ \neg\forall k.\ \neg\neg\exists j.\ j \geqslant k \land i \sim_\sigma j && p_2,\ p_1 \\ \leftrightarrow\ & \neg\forall i\,\exists k.\ \neg\exists j.\ j \geqslant k \land i \sim_\sigma j && p_3 \\ \leftrightarrow\ & \neg\forall i\,\exists k\,\forall j.\ j \geqslant k \rightarrow i \not\sim_\sigma j \end{aligned}$$

■

We now encode the predicates $p_i$ into automata and show that they satisfy XM if $\mathsf{AX}$ is assumed.

**Fact 13.2** Assume $\mathsf{AX}$ and let $p$ be a decidable predicate on numbers. Then $\exists n.pn$ satisfies XM.

**Proof** Let $A$ be an automaton that accepts exactly all nonempty boolean sequences. We define $\beta$ to be a boolean sequence satisfying $n \in \beta \leftrightarrow pn$. Note that $\beta$ can be defined because $p$ is decidable. Then $\exists n.pn$ is equivalent to $\beta \models A$ and satisfies XM by $\mathsf{AX}$. ■

**Fact 13.3** Assume $\mathsf{AX}$. Then $p_1$ satisfies XM.

**Proof** First note that $p_1 \Gamma \sigma i k$ is equivalent to

$$\exists l.\ i < l \land \exists j.\ j < l \land j \geqslant k \land \mathsf{C}(\sigma_i^l) = \mathsf{C}(\sigma_j^l)$$

Now the claim follows with Fact 13.2 since the quantification over $j$ is bounded and thus decidable. ■

We denote with $\sigma_{i..}$ the sequence obtained from $\sigma$ by dropping the first $i$ positions.

**Fact 13.4** Let $\Gamma$ be a finite semigroup, $\sigma$ be a sequence over $\Gamma$, and $i \leqslant j$. Then $i \sim_\sigma j \ \leftrightarrow\ 0 \sim_{\sigma_{i..}} (j - i)$.

**Fact 13.5** Let $\Gamma$ be a finite semigroup. Then there is an NFA $A$ such that $\sigma \models A \leftrightarrow \exists^\omega j.\ 0 \sim_\sigma j$ for all sequences $\sigma$ over $\Gamma$.

**Proof** The NFA $A$ repeatedly guesses a position and verifies that it merges with 0. The state type of $A$ is $(1 + \Gamma) \times (1 + \Gamma)$. The NFA $A$ computes the color of the processed prefix $\sigma_0^n$ in the first component. After guessing a position $j$, $A$ computes the color of $\sigma_j^n$ in the second component. Once these two colors are equal, the guess was correct.

In the first component, $1$ is only used for the initial state. In the second, $1$ encodes the phase while $A$ is guessing the next position. Hence $(1,1)$ serves as initial state. The accepting states are $(a,a)$ for all $a : \Gamma$, meaning that $A$ verified a guess. For all $a$ and $b$ there are transitions $(1,1) \xrightarrow{a} (a,1)$, $(b,1) \xrightarrow{a} (b+a,1)$, and $(b,1) \xrightarrow{a} (b+a,a)$ to guess a position. To verify the guess there are for all $a$ and $b \neq c$ transitions $(b,c) \xrightarrow{a} (b+a,c+a)$ and $(b,b) \xrightarrow{a}, (b+a,1)$.

If a run of $A$ passes infinitely often through accepting states, $A$ guessed infinitely many positions merging with $0$. Conversely, if there are infinitely many positions merging with $0$, then $A$ accepts $\sigma$. If $0$ merges with $j$ at position $k$, there is always another $j' > k$ merging with $0$, which $A$ can guess. ■

**Fact 13.6** Let $\Gamma$ be a finite semigroup. Then there is an NFA $A$ such that $\sigma \models A \leftrightarrow \exists i\, \exists j^\omega.\, 0 \sim_{\sigma_{i..}} j$ for all sequences $\sigma$ over $\Gamma$.

**Proof** Let $B$ be the NFA from Fact 13.5 for $\Gamma$. Then $A$ is obtained from $B$ as follows: $A$ guesses $i$ by reading the first $i$ positions of $\sigma$ and then transitions into the initial state of $B$. ■

**Fact 13.7** Assume $\mathsf{AX}$. Then $p_2$ satisfies XM.

**Proof** Let $\sigma$ be a sequence over a finite semigroup $\Gamma$. By Fact 13.6 it suffices to show that $p_2\Gamma\sigma$ is equivalent to $\exists i\, \exists j^\omega.\, 0 \sim_{\sigma_{i..}} j$:

$$\begin{aligned}
& \exists i\, \exists^\omega j.\, i \sim_\sigma j \\
\leftrightarrow\; & \exists i\, \exists^\omega j.\, j \geqslant i \wedge\, i \sim_\sigma j \\
\leftrightarrow\; & \exists i\, \exists j^\omega.\, j \geqslant i \wedge 0 \sim_{\sigma_{i..}} (j-i) && \text{Fact 13.4} \\
\leftrightarrow\; & \exists i\, \exists j^\omega.\, 0 \sim_{\sigma_{i..}} j
\end{aligned}$$
■

**Fact 13.8** Let $\Gamma$ be a finite semigroup. Then there is an NFA $A$ such that $\sigma \models A \leftrightarrow \exists k.\, \forall j \geqslant k.\, 0 \not\sim_\sigma j$ for all sequences $\sigma$ over $\Gamma$.

**Proof** The automaton $A$ first guesses $k$ and then asserts that all greater positions do not merge with $k$. The state type is $(1+\Gamma) \times (1+2^\Gamma)$. The first component is used to compute the color of $\sigma_0^n$ for $n > 0$ and the second component to compute the set $\{\, \mathsf{C}(\sigma_j^n) \mid k \leqslant j < n \,\}$ of colors of all suffixes of $\sigma_k^n$. The initial state is $(1,1)$.

To verify that the guessed position $k$ was correct, $A$ needs to ensure that the color of $\sigma_0^n$ is never equal to the color of a suffix $\sigma_j^n$ with $k \leqslant j < n$ (as then $j$ merges with $0$). If that is the case, $A$ gets stuck and cannot continue running. Hence all states $(a,s)$ are accepting for $a : \Gamma$ and $s : 2^\Gamma$.

There are transitions $(1,1) \xrightarrow{a} (a,1)$, $(b,1) \xrightarrow{a} (b+a,1)$, and $(b,1) \xrightarrow{a} (b+a,\{a\})$ for all $a : \Gamma$ to guess $k$. To verify the guess there are for all $a, b, c : \Gamma$ and $s : 2^\Gamma$ with $b \notin s$ transitions $(b,s) \xrightarrow{a} (b+a, \{a\} \cup \{\, c+a \mid c \in s \,\})$. ■

**Fact 13.9** Assume AX. Then $p_3$ satisfies XM.

**Proof** Let $\sigma$ be a sequence over a finite semigroup $\Gamma$ and $i$ be a number. By Fact 13.8 it suffices to show that $p_3\Gamma\sigma i$ is equivalent to $\exists k \forall j \geqslant k.\ 0 \not\sim_{\sigma_{i..}} j$:

$$
\begin{aligned}
& \exists k.\ \neg\exists j \geqslant k.\ i \sim_{\sigma} j & \\
\leftrightarrow\ & \exists k \forall j \geqslant k.\ i \not\sim_{\sigma} j & \text{Fact 2.1} \\
\leftrightarrow\ & \exists k \forall j \geqslant (k+i).\ i \not\sim_{\sigma} j & \\
\leftrightarrow\ & \exists k \forall j \geqslant (k+i).\ 0 \not\sim_{\sigma_{i..}} (j-i) & \text{Fact 13.4} \\
\leftrightarrow\ & \exists k \forall j \geqslant k.\ 0 \not\sim_{\sigma_{i..}} j & \blacksquare
\end{aligned}
$$

**Fact 13.10** AX implies RP.

**Proof** Follows with Facts 13.1, 13.3, 13.7, and 13.9. ■

**Theorem 13.11** FX, AX, AC, AU, RF, RA, and RP are pairwise equivalent.

**Proof** Follows with Facts 3.5, 3.6, and 9.3, Theorems 9.4 and 9.5, Corollary 10.3, and Facts 11.7, 12.2, and 13.10. ■

Note that each of the propositions in Theorem 13.11 follows from XM (since FX follows from XM) and is unprovable constructively (since RF implies IP and MP, Fact 3.2).

## 14 Final Remarks

In this paper we have studied the reduction of S1S to Büchi automata in Coq's constructive type theory. We have worked with two different semantics, AS semantics and UP semantics. For UP semantics, we showed without assumptions that Büchi's complement operation is correct and that S1S is decidable and classical. For AS semantics, we obtained these results assuming RF (a weak version of Ramsey's theorem following with excluded middle). We showed that the assumption RF is strictly necessary for AS semantics since (1) it is constructively unprovable and (2) it is entailed by each of the following properties: Complement automata exist, automaton acceptance satisfies XM, and formula satisfaction satisfies XM.

AS semantics is the canonical semantics for S1S and Büchi automata in the literature. Our results show that AS semantics does not work constructively. To make it work we need to assume RF. While RF is a consequence of excluded middle, it seems unlikely that RF entails excluded middle.

UP semantics admits only ultimately periodic sequences $xy^\omega$ specified by two strings $x$ and $y$. It is not surprising that UP semantics works constructively, and that UP semantics agrees with AS semantics if RF is assumed.

Doczkal and Smolka [10] give a purely constructive development of the temporal logic CTL. To make this possible, they admit only finite transition systems as models. Using tableau methods, they prove decidability and show soundness and completeness of a standard Hilbert proof system. This way they establish in a purely constructive way that the constructive semantics agrees with the standard semantics of CTL as given by the Hilbert system. Assuming XM and dependent choice, they also show that the standard path semantics of CTL agrees with the constructive semantics.

Sound and complete proof systems for S1S exist [15, 16, 17]. We expect that soundness and completeness for UP semantics can be shown constructively. For AS semantics, RF will be necessary for soundness.

In this paper, we have only considered Büchi's complement operation [4, 19]. We expect that other complement operations, in particular complementation by transformation to deterministic Muller automata [12, 19], can also be verified for AS semantics given RF. Recall that we have shown that RF is needed for the verification of every complement operation.

As it comes to future work, we plan to extend the constructive analysis of automata-based decision methods from S1S to further logics such as LTL, CTL, and S2S.

## References

[1] The Coq Proof Assistant, 2018.

[2] A. Blumensath. Monadic Second-Order Logic, 01 2015. Masaryk University, Lecture Notes.

[3] D. Bresolin, A. Montanari, and G. Puppis. A Theory of Ultimately Periodic Languages and Automata with an Application to Time Granularity. *Acta Inf.*, 46(5):331–360, 2009.

[4] J. R. Büchi. On a Decision Method in Restricted Second-Order Arithmetic. In *International Congress on Logic, Methodology, and Philosophy of Science*, pages 1–11. Stanford University Press, 1962.

[5] J. R. Büchi and D. Siefkes. *Decidable Theories: Vol. 2: The Monadic Second Order Theory of All Countable Ordinals*. Lecture Notes in Mathematics. Springer, 1973.

[6] H. Calbrix, M. Nivat, and A. Podelski. Ultimately Periodic Words of Rational **w**-Languages. In *MFPS 1993*, volume 802 of *LNCS*, pages 554–566, 1993.

[7] H. Calbrix, M. Nivat, and A. Podelski. Une méthode de décision de la logique mandique du second ordre d'une fonction successor. *Comptes rendus de l'académie des sciences, Serié I.*, 318:847–850, 1994.

[8] T. Coquand and B. Mannaa. The Independence of Markov's Principle in Type Theory. In *FSCD 2016*, volume 52 of *LIPIcs*, pages 17:1–17:18. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2016.

[9] S. Demri, V. Goranko, and M. Lange. *Temporal Logics in Computer Science: Finite-State Systems*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2016.

[10] C. Doczkal and G. Smolka. Completeness and Decidability Results for CTL in Constructive Type Theory. *Journal of Automated Reasoning*, 56(3):343–365, 2016.

[11] L. A. Kołodziejczyk, H. Michalewski, P. Pradic, and M. Skrzypczak. The Logical Strength of Büchi's Decidability Theorem. In *CSL 2016*, volume 62 of *LIPIcs*, pages 36:1–36:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016.

[12] R. McNaughton. Testing and Generating Infinite Sequences by a Finite Automaton. *Information and Control*, 9(5):521–530, 1966.

[13] D. Perrin and J.-E. Pin. *Infinite Words - Automata, Semigroups, Logic and Games*. Elsevier, 2004.

[14] P.-M. Pédrot and N. Tabareau. Failure is Not an Option: An Exceptional Type Theory. In *ESOP 2018*, LNCS, 2018. To appear.

[15] C. Riba. A Model Theoretic Proof of Completeness of an Axiomatization of Monadic Second-Order Logic on Infinite Words. In *IFIP TCS 2012*, volume 7604 of *LNCS*, pages 310–324, 2012.

[16] S. Shelah. The Monadic Theory of Order. *Annals of Mathematics*, 102(3):379–419, 1975.

[17] D. Siefkes. *Büchi's Monadic Second Order Successor Arithmetic*, volume 120 of *Lecture Notes in Mathematics*. Springer, 1970.

[18] W. Thomas. Languages, Automata, and Logic. In *Handbook of Formal Languages, Vol. 3*, chapter 7, pages 389–455. Springer, 1997.

[19] W. Thomas. Complementation of Büchi Automata Revisited. In *Jewels are Forever, Contributions on Theoretical Computer Science in Honor of Arto Salomaa*, pages 109–120. Springer, 1999.

[20] W. Veldman and M. Bezem. Ramsey's Theorem and the Pigeonhole Principle in Intuitionistic Mathematics. *Journal of the London Mathematical Society*, s2-47(2):193–211, 1993.