ELSEVIER Theoretical Computer Science 139 (1995) 1–25

Theoretical Computer Science

Fundamental Study

On Gabbay's temporal fixed point operator

Ian Hodkinson*

Department of Computing, Imperial College, 180 Queen's Gate, London SW7 2BZ, UK

Received March 1992; revised December 1993
Communicated by S. Abramsky

Abstract

We discuss the temporal logic "USF", involving *Until*, *Since* and the fixed point operator φ of Gabbay, with semantics over the natural numbers. We show that any formula not involving *Until* is equivalent to one without nested fixed point operators. We then prove that USF has expressive power matching that of the monadic second-order logic $S1S$. The proof shows that any USF-formula is equivalent to one with at most two nested fixed point operators – i.e., no branch of its formation tree has more than two φ's. We then axiomatise USF and prove that it is decidable, with PSPACE-complete satisfiability problem. Finally, we discuss an application of these results to the executable temporal logic system "MetateM".

1. Introduction

It is known that conventional temporal logic is insufficiently expressive to handle issues arising in areas such as concurrency. Several extended temporal logic systems, with second-order capability, now exist in the literature, including Wolper's ETL [20] and Banieqbal and Barringer's [2] calculus using minimal and maximal fixed points. Gabbay's [6] USF, of interest in the current paper, involves a fixed point operator with recursively defined semantics. All these systems are as expressive as the monadic second-order logic $S1S$ over the natural numbers, in which quantification over subsets as well as elements is allowed.

$S1S$ has been studied extensively; its decidability was proved by Büchi in 1962, using automata, and as USF is closely related to automata we can easily elicit the relationship between the two logics, and show that USF is also decidable and has the same expressive power as $S1S$ in a strong sense. Nonetheless, USF is itself surprisingly

*E-mail: imh@doc.ic.ac.uk.

0304-3975/95/$09.50
SSDI 0304-3975(94)00084-V

well-behaved, with elegant properties that can be studied without recourse to automata theory. In this vein we will prove that unbounded depth of nesting of the fixed point operator is not required for full expressive power. For the "past" fragment, this can indeed be done within *USF*, and in fact *no* nested fixed point operators are needed. Our current proof for full *USF* goes via the automata connection, and converts any formula effectively into one in which no fixed point operator is nested inside more than one other. We will give an example of this construction (Example 6.13 below). We would like to find a more direct proof that avoids the use of automata; such a proof might yield a more efficient conversion algorithm.

The recursive definition of the fixed point operator in *USF* is in the spirit of the executable temporal logic system *MetateM*, developed in London and Manchester and surveyed in [6, 3]. We will use the results mentioned above to prove that MetateM has the expressive power of *S1S*. We can also derive a simple axiomatisation of *USF*, and show that its satisfiability problem is PSPACE-complete.

We should mention that we will be using the logic *UYF* instead of *USF*. This has technical advantages and leads to no loss in expressive power (see Remark 2.9 for a discussion).

Notation

$\mathbb{N}$ will be the set $\{0, 1, 2, \dots\}$ of natural numbers. $\wp S$ will denote the set of all subsets of the set S. We often write $\bar{x}, \bar{a}, \dots$, for *tuples* – finite sequences of variables, atoms, elements of a structure, etc. Other notations will be defined when required.

2. Syntax and semantics of *UYF*

We start by developing the syntax and semantics of the fixed point operator. This is not entirely a trivial task. We will fix an infinite set of propositional atoms, with which our formulas will be written; we write $p, q, r, s, \dots$ for atoms.

Definition 2.1. (1) The set of formulas of *UYF* is the smallest class closed under the following.

(a) Any atom q is a formula of *UYF*, as is $\top$ (true).
(b) If A is a formula so is $\neg A$. (We let $\bot$ abbreviate $\neg\top$.)
(c) If A is a formula so is YA. We read Y as "yesterday".
(d) If A and B are formulas, so are $A \wedge B$ and $U(A, B)$. (The latter is read as "until". $A \vee B$ and $A \rightarrow B$ are regarded as abbreviations.)
(e) Suppose that A is a formula such that every occurrence of the atom q in A not within the scope of a φq is within the scope of a Y but not within the scope of a U. Then $\varphi q A$ is a formula. (The conditions ensure that $\varphi q A$ has fixed point semantics.)

(2) The *depth of nesting* of φ's in a formula A is defined by induction on its formation: formulas formed by clause (a) have depth 0, clause (e) adds 1 to the depth of nesting, clauses (b) and (c) leave it unchanged, and in clause (d), the depth of nesting of

$U(A,B)$ and $A \wedge B$ is the maximum of the depths of nesting of A and B. So, for example, $\neg\varphi r(\neg Yr \wedge \varphi qY(q \rightarrow r))$ has depth of nesting of 2.

(3) A UYF-formula is said to be a *YF-formula* if it does not involve U.

(4) Let A be a formula and q an atom. A *bound occurrence* of q in A is one in a subformula of A of the form φqB. All other occurrences of q in A are said to be *free*. An occurrence of q in A is said to be *pure past* in A if it is in a subformula of A of the form YB but not in a subformula of the form $U(B,C)$. So φqA is well-formed if and only if all free occurrences of q in A are pure past.

2.1. Semantics of UYF

An *assignment* is a map h providing a subset $h(q)$ of $\mathbb{N}$ for each atom q. If h, h' are assignments, and $\bar{q}$ a tuple of atoms, we write $h =_{\bar{q}} h'$ if $h(r) = h'(r)$ for all atoms r not occurring in $\bar{q}$. If $S \subseteq \mathbb{N}$ and q is an atom, we write $h_{q/S}$ for the unique assignment h' satisfying $h' =_q h, h'(q) = S$.

For each assignment h and formula A of UYF we will define a subset $h(A)$ of $\mathbb{N}$, the interpretation of A in $\mathbb{N}$. Intuitively, $h(A) = \{n \in \mathbb{N}: A \text{ is true at } n \text{ under } h\}$. We will ensure that, whenever φqA is well-formed,

$$h(\varphi qA) \text{ is the unique } S \subseteq \mathbb{N} \text{ such that } S = h_{q/S}(A). \tag{$*$}$$

Notation 2.2. If $S \subseteq \mathbb{N}$, we write $S+1$ (or $1+S$) for $\{s+1: s \in S\}$.

Definition 2.3. We define the semantics of UYF by induction on the structure of formulas. Let h be an assignment. If A is atomic then $h(A)$ is already defined. We set the following.

- $h(\top) = \mathbb{N}$.
- $h(\neg A) = \mathbb{N} \setminus h(A)$.
- $h(YA) = h(A) + 1$.
- $h(A \wedge B) = h(A) \cap h(B)$.
- $h(U(A,B)) = \{n \in \mathbb{N}: \exists m > n(m \in h(A) \wedge \forall m'(n < m' < m \rightarrow m' \in h(B)))\}$.
- Finally, assume that φqA is well-formed, and (inductively) that $g(A)$ is defined for all assignments g. We will define $h(\varphi qA)$.
  First define assignments h_n ($n \in \mathbb{N}$) by induction: $h_0 = h$, $h_{n+1} = (h_n)_{q/h_n(A)}$. We now define
  $$h(\varphi qA) \stackrel{\text{def}}{=} \{n \in \mathbb{N}: n \in h_n(A)\} = \{n \in \mathbb{N}: n \in h_{n+1}(q)\}.$$

To establish $(*)$ we need some definitions and lemmas.

Definition 2.4. (1) If $n \in \mathbb{N}$, we say that subsets $S_1, S_2 \subseteq \mathbb{N}$ *agree before* n if for all $m < n$, $m \in S_1$ if and only if $m \in S_2$. We say that S_1 and S_2 *agree up to* n if they agree before $n+1$.

(2) Assume that A is a formula of UYF. If $n \in \mathbb{N}$, we say that assignments g, h are *A-similar up to n* if for all atoms q,

- if all free occurrences of q in A are pure past, then $g(q)$ and $h(q)$ agree before n;
- if not all free occurrences of q in A are pure past, but still no free occurrence of q in A is within the scope of a U, then $g(q)$ and $h(q)$ agree up to n;
- otherwise, $g(q)=h(q)$.

(3) A UYF formula A is said to be *local* if $g(A)$ and $h(A)$ agree up to n whenever g, h are assignments that are A-similar up to n.

Remark 2.5. From the definitions, if g, h are φqA-similar up to n, and $g(q)$, $h(q)$ agree before n, then g, h are A-similar up to n.

Lemma 2.6. *Assume that A is a local UYF formula, and that φqA is well-formed. Then:*
(1) *If g, h are assignments with $g =_q h$, then $g(\varphi qA)=h(\varphi qA)$.*
(2) *If $S \subseteq \mathbb{N}$ and h is an assignment, then $h_{q/S}(A)=S$ if and only if $S=h(\varphi qA)$.*
(3) *φqA is local.*

Proof. (1) By Definition 2.3 it suffices to show that for all $n \in \mathbb{N}$, $g_n(q)$ and $h_n(q)$ (as in the definition) agree before n. We do so by induction on n. If $n=0$ there is nothing to prove. Assume the statement for n. Clearly, $g_n =_q h_n$. By Remark 2.5 and the inductive hypothesis, g_n and h_n are A-similar up to n. As A is local, $g_n(A)$ and $h_n(A)$ agree up to n: i.e., $g_{n+1}(q)$ and $h_{n+1}(q)$ agree before $n+1$. This completes the induction.

(2) By (1) we can replace h by $h_{q/S}$. So it is enough to show that for any h, $h(A)=h(q) \Leftrightarrow h(\varphi qA)=h(q)$.

$\Rightarrow$: Suppose that $h(A)=h(q)$. First observe that $h_1(q)=h(A)=h(q)$, so that $h=h_1$. It follows by the definition of the h_n and induction that $h=h_n$ for all n. But now for all $n, n \in h(\varphi qA)$ if and only if $n \in h_{n+1}(q)=h(q)$, so that $h(\varphi qA)=h(q)$ as required.

$\Leftarrow$: Assuming that $h(\varphi qA)=h(q)$, we show that for all n,
$(\text{a})_n$ $h(A)$ and $h_n(A)$ agree up to n,
$(\text{b})_n$ $h(q)$ and $h_n(A)$ agree up to n.

It will clearly follow that $h(A)=h(q)$. We proceed by induction on n. $(\text{a})_0$ holds because $h_0=h$. We now show that $((\text{a})_m\colon m \leqslant n) \Rightarrow (\text{b})_n$. Let $m \leqslant n$. Then $m \in h(q)=h(\varphi qA)$ if and only if $m \in h_m(A)$ by Definition 2.3, if and only if $m \in h(A)$ by $(\text{a})_m$, if and only if $m \in h_n(A)$ by $(\text{a})_n$. This proves $(\text{b})_n$.

We now show that $(\text{b})_n \Rightarrow (\text{a})_{n+1}$. $(\text{b})_n$ says that $h(q)$ and $h_{n+1}(q)$ agree before $n+1$. Thus, h and h_{n+1} are A-similar up to $n+1$. As A is local, $h(A)$ and $h_{n+1}(A)$ agree up to $n+1$, proving $(a)_{n+1}$.

(3) To prove φqA local, we need to show that whenever g, h are assignments that are φqA-similar up to n, then $g(\varphi qA)$ and $h(\varphi qA)$ agree up to n. By (1) and (2) we can assume that $g(q)=g(A)=g(\varphi qA)$ and $h(q)=h(A)=h(\varphi qA)$; the hypothesis and what we have to prove are unchanged.

The proof is by induction on n. Assume the result for all $m<n$ and suppose that g, h are φqA-similar up to n. If $m<n$ then g and h are φqA-similar up to m, so by the inductive hypothesis, $g(\varphi qA)$ and $h(\varphi qA)$ agree up to m. Hence, $g(A)$ and $h(A)$ agree

before n. By Remark 2.5, g and h are A-similar up to n, so as A is local, $g(A)$ and $h(A)$ agree up to n, as required. This completes the proof. □

Proposition 2.7. *Every formula A of UYF is local, and for any h, $h(A)$ depends only on $h(q)$ for those atoms q that have free occurrences in A.*

Proof. We show by induction on A that A is local, and that whenever g, h are assignments agreeing on the atoms occurring free in A, then $g(A)=h(A)$. In the atomic case and the cases of the boolean connectives the proof is simple, and the case of φqA is covered by the lemma. For $A=U(B,C)$, if assignments g, h are A-similar up to n then $g(q)=h(q)$ for all atoms q with free occurrences in A. Clearly, g and h agree on the free atoms of B and of C, so by the inductive hypothesis, $g(B)=h(B)$ and $g(C)=h(C)$, yielding $g(A)=h(A)$ – so these two sets certainly agree up to n. This proves both claims for A.

Now consider the case of $A=YB$. Pick $n\in\mathbb{N}$ and a pair g, h of assignments that are YB-similar up to n. We claim that g and h are B-similar up to $n-1$. Let q be an atom. If all free occurrences of q in B are pure past, then the same holds for YB, so that $g(q)$, $h(q)$ already agree up to $n-1$. Otherwise, if no free occurrence of q in A is under a U, then all free occurrences of q in YB are pure past. Hence again, $g(q)$, $h(q)$ agree up to $n-1$. If none of these apply to q then $g(q)=h(q)$. It follows that g and h are B-similar up to $n-1$, as claimed. By the inductive hypothesis, $g(B)$ and $h(B)$ agree up to $n-1$. So by definition of the semantics of Y, $g(YB)$ and $h(YB)$ agree up to n. The proof that $h(A)$ depends only on $h(q)$ for q occurring free in A is straightforward. □

Combining Lemma 2.6 and Proposition 2.7 yields the following theorem.

Theorem 2.8 (Fixed point theorem). (1) *Suppose that A is any UYF formula and φqA is well formed. Then if h is any assignment, there is a unique subset $S=h(\varphi qA)$ of $\mathbb{N}$ such that $S=h_{q/S}(A)$. Thus, regarding $S\mapsto h_{q/S}(A)$ as a map $\alpha:\wp\mathbb{N}\to\wp\mathbb{N}$ (depending on h, A), α has a unique fixed point $S\subseteq\mathbb{N}$, and we have $S=h(\varphi qA)$. For any h, $h(A)=h(q)\Leftrightarrow h(\varphi qA)=h(q)$.*

(2) *If q has no free occurrence in a formula A and $g=_q h$, then $g(A)=h(A)$.*

(3) *if φqA is well-formed and r is an atom not occurring in A, then for all assignments h, $h(\varphi qA)=h(\varphi r A(q/r))$, where $A(q/r)$ denotes substitution by r for all free occurrences of q in A.*

Proof. By the proposition, A is local, so (1) follows from the lemma. (2) is proved in the proposition, and (3) is clear from (1). □

Remark 2.9. We should mention two technical differences between our system and the original logic USF of Gabbay. In [6], Gabbay defines USF using the first-order connectives *Until* and *Since* as well as the fixed point operator. *Since* is the temporal dual of *Until*; its semantics are given by $h(S(A,B))=\{n\in\mathbb{N}: \exists m<n\,(m\in h(A)\wedge$

$\forall m'(m<m'<n \rightarrow m' \in h(B)))\}$. We stress that UYF is just as expressive as USF: Yq is definable in USF by the formula $S(q, \perp)$, whilst $S(p,q)$ is definable in UYF by $\varphi r Y(p \vee (q \wedge r))$. Using UYF allows easier proofs and stronger results. Also, we admit rather more well-formed formulas than does Gabbay in [6]. For $\varphi q A$ to be well-formed, Gabbay requires that all atoms have only pure past occurrences in A, whilst we only need this for the atom q. As an example, $\varphi r(U(p,q) \wedge Yr)$ is well-formed for us, whilst $\varphi r(U(p,q) \wedge S(r, \perp))$ is not a formula in USF as defined in [6].

3. Elementary results

Here we establish some simple results on the way the fixed point operator interacts with the other connectives of the logic. They are proved using the fixed point theorem, and some will be needed later.

Definition 3.1. Two UYF-formulas A, B are said to be *equivalent* if for all assignments h we have $h(A)=h(B)$. We write $A \equiv B$ if A and B are equivalent.

Proposition 3.2. *Let $\varphi q A(q)$ be a UYF-formula. Then $\neg \varphi q A(q)$ is equivalent to $\varphi q \neg A(\neg q)$. Here, $A(\neg q)$ denotes the result of replacing each free occurrence of q by $\neg q$ throughout A.*

Proof. Let h be any assignment and assume that $h(\varphi q A(q))=S \subseteq \mathbb{N}$. We wish to show that $h(\varphi q \neg A(\neg q))=\mathbb{N} \setminus S$. By the fixed point theorem, it suffices to show that $h_{q/\mathbb{N}\setminus S}(\neg A(\neg q))=\mathbb{N}\setminus S$. But $h_{q/\mathbb{N}\setminus S}(\neg A(\neg q))=h_{q/S}(\neg A(q))=\mathbb{N}\setminus h_{q/S}(A(q))$, and this last is equal to $\mathbb{N}\setminus S$ by the fixed point theorem again. □

In a similar way we can show the following proposition.

Proposition 3.3. *Let q, r be distinct atoms and suppose that $\varphi q \varphi r A$ is well-formed. Then $\varphi q \varphi r A \equiv \varphi r \varphi q A \equiv \varphi q A(r/q)$. Here, $A(r/q)$ denotes substitution of q for all free occurrences of r in A.*

Proof. Choose any assignment h. By the fixed point theorem, for all $S \subseteq \mathbb{N}$ we have $h(\varphi q \varphi r A)=S$ if and only if $h_{q/S}(\varphi r A)=S$. Using the theorem again, this is if and only if $h_{q/S,r/S}(A)=S$. So by symmetry, $\varphi q \varphi r A \equiv \varphi r \varphi q A$. Moreover, as clearly $h_{q/S,r/S}(A)=S$ if and only if $h_{q/S}(A(r/q))=S$, the last part follows. □

Now we examine how φ interacts with the *yesterday* connective.

Proposition 3.4. *Let $B(q)$ be any formula and write $B(Yq)$ for the result of replacing every free occurrence of q in B by the formula Yq. Then $Y\varphi q B(Yq) \equiv \varphi q Y B(q)$.*

Proof. Let h be given, and let $S = h(\varphi q B(Yq))$, so that the interpretation of the left-hand side under h is just $S+1$. By the fixed point theorem, it suffices to show that $h_{q/S+1}(YB(q)) = S+1$. But $h_{q/S+1}(YB(q)) = h_{q/S}(YB(Yq)) = 1 + h_{q/S}(B(Yq))$, and the latter is equal to $1+S$, by choice of S and using the fixed point theorem once more. $\square$

This result allows us to normalise UYF-formulas, by pushing all Y's inwards until they are next to atoms.

Definition 3.5. (1) If A is any UYF-formula, and $n < \omega$, we define $Y^n A$ by induction: $Y^0 A = A$ and $Y^{n+1} A = Y(Y^n A)$. Formulas of the form $Y^n q$ for atomic q, or $Y^n \top$, are called *basic*.

(2) A UYF-formula A is said to be *normal* if it is built up from basic formulas (the *basic subformulas* of A) using only $\neg$, $\wedge$, U and φ (no Y). A subformula B of A is said to be a *normal subformula* of A if every basic subformula of B is a basic subformula of A.

So a "basic subformula" of A is a subformula that is maximal amongst those subformulas of A that are basic. For example, the basic subformulas of $A = \neg Yq \rightarrow YYq$ are the first Yq and the YYq; neither the two occurrences of q nor the subformula Yq of the YYq are basic subformulas of A. The normal subformulas of A are just the first Yq, $\neg Yq$, YYq and A.

Lemma 3.6. *Let A be any normal formula. Then YA is equivalent to a normal formula B.*

Proof. We will show in addition that for any atom q, if all free occurrences of q in YA are pure past then the same holds for B.

We go by induction on the number of $\neg$'s, $\wedge$'s, U's and φ's in A. If this is 0 then A is basic, so YA is already normal. The condition on atoms is trivially valid. It is easily seen that $Y\neg A \equiv Y\top \wedge \neg YA$, $Y(A \wedge B) \equiv YA \wedge YB$, and $YU(A,B) \equiv Y\top \wedge (A \vee (B \wedge U(A,B)))$, so the result follows immediately from the inductive hypothesis in these cases.

Now assume the result for all formulas with no more $\wedge$'s, $\neg$'s, U's and φ's than A, and consider $\varphi q A$ (assumed well-formed and normal). By Theorem 2.8(3) we can rename bound atoms of A if necessary, so we can suppose that all occurrences of q in A are free and (necessarily) pure past. Now A is normal, so it has the form $B(Yq)$ where $B(q)$ is normal and with the same number of $\neg$'s, $\wedge$'s, U's and φ's as A. By the inductive hypothesis, $YB(q)$ is equivalent to a normal formula C. As all free occurrences of q in $YB(q)$ are pure past, the same holds for C, so that $\varphi q C$ is well-formed. It is clearly normal, and by Proposition 3.4 is equivalent to $Y\varphi q A$. Finally, note that if r is any atom all of whose free occurrences in $Y\varphi q A$ are pure past, then the same holds for YB and so (inductively) for C. Hence, all free occurrences of r in $\varphi q C$ are pure past, as required. $\square$

Theorem 3.7. *Any UYF-formula A is equivalent to a normal formula.*

Proof. By induction on A. The only hard case is YA, which is dealt with by the preceding lemma. □

4. Recursive systems

The semantics of the fixed point operator were defined by nested recursion. We will now see how to unravel the nesting, replacing it by simultaneous recursion. This will be a main step in our proof of elimination of nested φ's for YF.

Definition 4.1. Let $n>0$. A *recursive system* (*of width n*) is a pair $\rho=(\bar{r},\bar{B})$, where $\bar{r}=(r_1,\dots,r_n)$ and $\bar{B}=(B_1,\dots,B_n)$, the r_i are distinct atoms and the B_i are UYF-formulas. We require that for all $i,j\leqslant n$, every free occurrence of r_i in B_j is pure past.

A formula A is said to be *φ-free* if it contains no φ-operator. The recursive system $(\bar{r},\bar{B})$ is said to be *φ-free* if each B_j is φ-free.

4.1. Semantics of recursive systems

We give them a fixed point semantics, as for φ. Let $\rho=(\bar{r},\bar{B})$ be a recursive system. It can be shown using the technique of Lemma 2.6 that for any assignment h there is a unique assignment h_ρ with $h_\rho=_{\bar{r}}h$ and $h_\rho(r_i)=h_\rho(B_i)$ for all $i\leqslant n$. h_ρ can be defined by recursion as before.

Definition 4.2. (1) Let $\rho=((r_1,\dots,r_n),(B_1,\dots,B_n))$ be a recursive system, and let A be a YF-formula. We say that A and ρ are *equivalent* if for all h we have $h(A)=h_\rho(B_1)$ $(=h_\rho(r_1))$.

(2) If $\rho'=((r'_1,\dots,r'_m),(B'_1,\dots,B'_m))$ is another recursive system, we say that ρ and ρ' are *equivalent* if for all assignments h, $h_\rho(r_i)=h_{\rho'}(r'_i)$ for all $i\leqslant\min(n,m)$. Note that in general, equivalence is not transitive; but the definition is no less useful for that.

We begin by showing, in the following proposition, that UYF is at least as expressive as recursive systems. The idea of the proof is well-known; see, for example, Bekić's theorem [19, Theorem 10.1], a similar result on fixed points of continuous functions on domains.

Proposition 4.3. *Let $\rho=((r_1,\dots,r_n),(B_1,\dots,B_n))$ be any recursive system. Then there is a UYF-formula A that is equivalent to ρ. If the B_i are YF-formulas then such an A can be found in YF also.*

Proof. By induction on the width n. If $n=1$ we let $A=\varphi r_1 B_1$. Assume the result for n and let $\rho=((r_1,\dots,r_{n+1}),(B_1,\dots,B_{n+1}))$ be given. Let $\rho^*=((r_1,\dots,r_n),(B_1^*,\dots,B_n^*))$, where $B_i^* = B_i(r_{n+1}/\varphi r_{n+1} B_{n+1})$ for each $i \leq n$. We claim that ρ and ρ^* are equivalent; the result will then follow by induction.

Let h be any assignment, and let $S_i = h_\rho(r_i)$ for each $i \leq n+1$. Since $h_\rho(r_{n+1}) = h_\rho(B_{n+1})$, it follows from the fixed point theorem (2.8) that $h_\rho(\varphi r_{n+1} B_{n+1}) = h_\rho(r_{n+1})\,(= S_{n+1})$. Hence, from h_ρ's point of view, replacing r_{n+1} by $\varphi r_{n+1} B_{n+1}$ in B_i makes no difference, and we have $h_\rho(B_i^*) = h_\rho(B_i)$ for each i. But r_{n+1} does not occur free in the B_i^*, so letting $h^* = h_{(r_i/S_i : i \leq n)}$, we see that $h^*(B_i^*) = h_\rho(B_i^*)$ for each i. Thus, $h^*(B_i^*) = h_\rho(B_i^*) = h_\rho(B_i) = S_i = h^*(r_i)$ for each i. It follows by uniqueness of the fixed point that $h^* = h_{\rho^*}$, so that ρ and ρ^* are equivalent. $\square$

The fact that formulas B_i in a recursive system may contain U will be needed later, but our current aim is to show that nesting of fixed point operators in YF can be eliminated. So from now until the end of Section 5 we restrict attention to YF: all formulas will be YF-formulas. We first prove a converse to the previous proposition.

Theorem 4.4. *Let A be any YF-formula. Then there is a φ-free recursive system $\rho = ((r_1, \ldots, r_n), (B_1, \ldots, B_n))$ for some n, that is equivalent to A.*

Proof. By Theorem 3.7 we can assume that A is normal. By renaming bound atoms if need be (cf. Theorem 2.8(3)), we can also assume that for any subformula $\varphi q B$ of A, the only occurrences of q in A are in B.

If A is a formula, we write $\hat{A}$ for the formula obtained from A by omitting all φ's. Formally, $\hat{q} = q$ for any atom q, $\hat{\top} = \top$, $\widehat{\neg A} = \neg \hat{A}$, $(A \wedge B)\hat{} = \hat{A} \wedge \hat{B}$, $\widehat{YA} = Y\hat{A}$, and $\widehat{\varphi q A} = \hat{A}$. (Later we will also use $\hat{A}$ for UYF-formulas; we then include the clause $U(A, B)\hat{} = U(\hat{A}, \hat{B})$.) We will find ρ as above, with the additional property:

$$\text{No } r_i \text{ occurs free in } A\text{, and each } B_i \text{ is a normal subformula of } \hat{A}\text{, and } B_1 = \hat{A}. \quad (*)$$

We go by induction on A. If A is basic we let $\rho = (r, A)$ where r is any atom not occurring in A. Clearly, ρ is equivalent to A, and $(*)$ holds. If the recursive system $(\bar{r}, \bar{B}) = ((r_1, \ldots, r_n), (B_1, \ldots, B_n))$ is equivalent to A, and $(*)$ holds, then $((r_0, \bar{r}), (\neg B_1, \bar{B}))$ is equivalent to $\neg A$, where r_0 is a new atom; and $(*)$ still holds.

Assume that $A \wedge A'$ satisfies the condition on bound atoms, and that the recursive systems $((r_1, \ldots, r_n), (B_1, \ldots, B_n))$ and $((r'_1, \ldots, r'_m), (B'_1, \ldots, B'_m))$ are equivalent to A, A', respectively, $(*)$ holding for each. Then no r_i occurs free in A. If some r_i occurs free in A', then by the condition on bound atoms it cannot occur bound in A. Hence, it does not occur at all in A, nor in any B_j (since they are subformulas of $\hat{A}$). In consequence, the functionality of $(\bar{r}, \bar{B})$ is unaffected if we replace r_i by a new atom not occurring at all in $A \wedge A'$. If we do this for all r_i where necessary, and undertake similar modifications for the r'_i, then the recursive system $((r_0, \bar{r}, \bar{r}'), (B_1 \wedge B'_1, \bar{B}, \bar{B}'))$, where r_0 is a new atom, satisfies $(*)$ for $A \wedge A'$; and it is certainly equivalent to $A \wedge A'$. This completes the case of $\wedge$.

Finally, we consider the case φqA (as A is normal, the case YA does not arise). Assume $\rho=((r_1,\dots,r_n),\ (B_1,\dots,B_n))$ is equivalent to A, and that (*) holds. We let $\rho^*=((q,r_2,\dots,r_n), (B_1,\dots,B_n))$. By (*), all occurrences of q in all the B_i are pure past, so ρ^* is well-formed. (This is where the assumption that A is normal is used.)

Evidently, (*) holds for ρ^*, φqA. We claim that ρ^* is equivalent to φqA. Let h be an arbitrary assignment, suppose that $h(\varphi qA)=S$, say, and let $h'=h_{q/S}$. Obviously, $h'_\rho(B_i)=h'_\rho(r_i)$ for each $i\geqslant 2$. But further, $h'_\rho(B_1)=h'(A)$ since A and ρ are equivalent; and by the fixed point theorem, $h'(A)=S=h'(q)$. If q does occur in A then by (*), $q\notin\{r_1,\dots,r_n\}$, and if not, then we can certainly assume this; so $h'(q)=h'_\rho(q)$. We have shown that $h'_\rho(B_1)=h'_\rho(q)$. Hence, h'_ρ yields a fixed point of ρ^*, and by uniqueness of fixed points, $h_{\rho^*}(q)=h'_\rho(q)=S=h(\varphi qA)$. So φqA and ρ^* are equivalent, as claimed. This completes the induction, and with it the proof of the theorem. □

We can restrict recursive systems further without losing any of their expressive power.

Definition 4.5. A recursive system $\rho=((r_1,\dots,r_n), (B_1,\dots,B_n))$ is said to be *simple* if (a) it is φ-free, (b) the B_i are normal, and (c) all occurrences of the r_j in the B_i lie under the same number of Y's – i.e., there is $d>0$ such that every basic subformula of any B_i of the form $Y^k r_j$ (for any j) is such that $k=d$. This unique d is called the *depth* of ρ.

Proposition 4.6. *Let $\rho=((r_1,\dots,r_n), (B_1,\dots,B_n))$ be a φ-free recursive system. Then ρ is equivalent to a simple recursive system of depth* 1.

Proof. By Theorem 3.7 we can assume that the B_i are normal. Let k be maximal such that for some j, $Y^k r_j$ occurs as a basic subformula of some B_i. Introduce new atoms s_j^l for $j\leqslant n$ and $1\leqslant l\leqslant k$, and define S_j^1 to be the (normal) formula obtained from B_j by replacing each basic subformula $Y^l r_j$ by Ys_j^l (all l), and $S_j^l=Ys_j^{l-1}$ for each $l\geqslant 2$. Then let $\rho'=(\bar{s},\bar{S})$, where $\bar{s}=((s_j^l{:}\,j\leqslant n){:}\ l\leqslant k)$, and similarly for $\bar{S}$.

Certainly, ρ' is well-formed and simple of depth 1. We claim that it is equivalent to ρ. Let h be given, and define h' by

$$h'=_{\bar{s}}h,\qquad h'(s_j^l)=h_\rho(Y^{l-1}r_j)\quad\text{for each } j,l.$$

Then for each j, $h'(s_j^l)=h'(Ys_j^{l-1})=h'(S_j^l)$ if $l\geqslant 2$, and

$$h'(S_j^1)=h_\rho(S_j^1(S_i^l/Y^{l-1}r_i{:}\ i\leqslant n, l\leqslant k))=h_\rho(B_j)=h_\rho(r_j)=h'(s_j^1).$$

Hence, h' is a fixed point assignment for ρ', and by uniqueness of fixed points, $h'=h_{\rho'}$. As $h_{\rho'}(s_j^1)=h_\rho(r_j)$ for each j, ρ' and ρ are equivalent, as claimed. □

4.2. Unfolding

We need a final result on syntactic manipulation of recursive systems.

Definition 4.7. Let $\rho=((r_1,\ldots,r_n),(B_1,\ldots,B_n))$ be a simple recursive system of depth 1. Define formulas B_i^k for $k<\omega$ by induction:
- $B_i^0=r_i$ for all i,
- B_i^{k+1} is obtained by *normalizing* $B_i^k(r_j/B_j\colon j\leqslant n)$ (i.e., replacing it by a normal equivalent).

Also define recursive systems $\rho^k=((r_1,\ldots,r_n),(B_1^k,\ldots,B_n^k))$ for each $k\geqslant 1$.

Lemma 4.8. *Let $\rho=((r_1,\ldots,r_n),(B_1,\ldots,B_n))$ be a simple recursive system of depth* 1, *as above. Then for each $k\geqslant 1$, the system ρ^k is simple, of depth k, and equivalent to ρ – we have $h_\rho=h_{\rho^k}$ for all h,k.*

Proof. By induction on k. Assume the result for $k\geqslant 1$ and let h be any assignment. Clearly, ρ^{k+1} is φ-free. To normalise a φ-free formula we move all Y's in through the $\wedge$'s and $\neg$'s using the rules $Y(A\wedge B)\equiv YA\wedge YB$ and $Y\neg A\equiv Y\top\wedge\neg YA$. These rewrite rules clearly preserve the total number of Y's above each atom, and it follows that ρ^{k+1} is simple and of depth $k+1$.

We claim that h_ρ gives a fixed point of ρ^{k+1}; the lemma will then follow by uniqueness of fixed points. But as $h_\rho(r_j)=h_\rho(B_j)$ for each j, the substitution of B_j for r_j in B_i^k makes no difference from h_ρ's point of view, so that $h_\rho(B_i^{k+1})=h_\rho(B_i^k)$. By the inductive hypothesis this last is equal to $h_\rho(r_i)$, which completes the proof. □

5. Elimination of fixed point operators

We now prove our first main result, that any formula of YF is equivalent to one with depth of nesting of φ's of at most 1.

Theorem 5.1 (Elimination of fixed point nesting). *Let A be any YF-formula. Then A is equivalent to a formula A′ without nested φ's – a boolean combination of YF-formulas of the form φqB, where B is φ-free. Moreover, A′ is obtainable effectively from A (in time polynomial in the length of A).*

The proof is rather technical. Nesting of the fixed point operator corresponds to recursive systems of width greater than 1. The idea of the proof is (roughly) to express A as a recursive system of width n, reduce n to 1 by coding the truth values of its formulas at regular intervals by a single atom, and obtain the value of A at intermediate times by interpolation.

Notation 5.2. If $k,p\in\mathbb{N}$, and $p>0$, we write $k \bmod p$ for the unique i with $0\leqslant i<p$ and $i\equiv k \pmod p$. We write k_p for $k-(k \bmod p)$; k_p is the largest multiple of p not greater than k.

Let A be any YF-formula. By Theorem 4.4 there is a φ-free recursive system $\rho=((r_1,\dots,r_n),(B_1,\dots,B_n))$ that is equivalent to A. By Proposition 4.6, ρ can be taken to be simple of depth 1. Study of the proofs in Sections 3–4 shows that ρ is obtainable effectively from A in polynomial time.

Let $p=2n+3$. To prove Theorem 5.1 it suffices to show the following lemma.

Lemma 5.3. *For each $j\leqslant n$ there is a YF-formula D_j of the form φqC, where C is φ-free, such that for any assignment h and any multiple m of p, we have $m\in h(D_j)\Leftrightarrow m\in h_\rho(B_j)$. The formula D_j is obtainable effectively from ρ in polynomial time.*

For assuming the lemma, we can interpolate to get the values of A at any time. To see this, observe first that the "clock" formula $\tau=\varphi q\neg Y\neg Y^{p-1}q$ satisfies $h(\tau)=\{mp: m\in\mathbb{N}\}$ for any assignment h. We will show the following claim.

Claim. *A is equivalent to $A'=\bigwedge_{0\leqslant i<p} Y^i\tau\to B_1^i(r_j/D_j: j\leqslant n)$, where B_1^i is as in Definition* 4.7.

Proof. Let h be any assignment, let $m\in\mathbb{N}$ and let $i=m \bmod p$. Then $m\in h(A')$ if and only if $m\in h(B_1^i(r_j/D_j: j\leqslant n))$. Now by Definition 4.7 and Lemma 4.8, B_1^i is normal and every occurrence of each r_j in it is in a basic subformula of the form $Y^i r_j$. Moreover, as m_p is divisible by p, $m_p\in h(D_j)$ if and only if $m_p\in h_\rho(B_j)=h_\rho(r_j)$. So as $h=_{\bar{r}}h_\rho$, it follows that $m\in h(B_1^i(r_j/D_j: j\leqslant n))$ if and only if $m\in h_\rho(B_1^i)$. But by Lemma 4.8, ρ^i is equivalent to ρ and hence to A, so this is if and only if $m\in h(A)$, proving the claim. □

Evidently, A' has no nested fixed point operators. Moreover, it is obtainable effectively from ρ in polynomial time, assuming that the D_j are. So Theorem 5.1 follows from the lemma. □

Proof of Lemma 5.3. Assume for simplicity of notation that $j=1$; the proof for other j is the same. By Lemma 4.8 we can replace ρ by ρ^p (effectively and in polynomial time), and thus assume that ρ is a simple recursive system of depth p. Choose a surjective function $\chi:\mathbb{N}\to\{\top,\bot,B_i: i\leqslant n\}$ with the following properties:

- $\chi(m)=\chi(m+p)$ for all $m\in\mathbb{N}$;
- $\chi(0)=B_1$;
- if $k<p-3$ and k is odd, then $\chi(k)=\bot$;
- $\chi(p-3)=\chi(p-2)=\top$;
- $\chi(p-1)=\bot$.

We can find such a χ since p is large enough. χ has the important property that for any assignment h, the sets $h(\chi(m))$, $h(\chi(m+1))$ are both nonempty if and only if $m\equiv p-3 \pmod p$. We will design a φ-free recursive system $\sigma=(s,C)$ satisfying, for every assignment h,

$$\forall m\in\mathbb{N}\,(m\in h_\sigma(s)\ \Leftrightarrow\ m\in h_\rho(Y^{m \bmod p}\chi(m))). \qquad (*)$$

In particular, if m is a multiple of p then $m \in h_\sigma(s) \Leftrightarrow m \in h_\rho(B_1)$. The formula $D_1 = \varphi s C$ then satisfies the conclusion of the lemma, as by the fixed point theorem, $h(D_1) = h_\sigma(s)$ for any h.

We let the formula C be:

$$\bigwedge_{0 \leqslant i < p} Y^i[\neg Y\top \vee (Y^2 s \wedge Y^3 s)] \to Y^i[\chi(i)(Y^p r_j / Y^{p-l_j} s : j \leqslant n)].$$

Here, $0 \leqslant l_j < p$ is such that $\chi(l_j) = B_j$; l_j exists because χ is onto. It is clear that C is obtainable effectively in polynomial time from the B_j. Hence, the lemma will be established if we can prove that $(*)$ holds for all m.

We do this by induction on m. Fix an assignment h. Let $m \in \mathbb{N}$ and assume that $(*)$ holds for all $m' < m$. We will show that it holds for m.

Claim. *For all $i < p$, $Y^i[\neg Y\top \vee (Y^2 s \wedge Y^3 s)]$ is true at m under the assignment h_σ if and only if $i = m \bmod p$.*

Proof. If $i > m$ then $Y^i A$ is false at m for any A under any assignment, and $i \neq (m \bmod p)$. So we can assume $i \leqslant m$, in which case the left-hand side holds if and only if

$$m - i \in h_\sigma(\neg Y\top \vee (Y^2 s \wedge Y^3 s)). \tag{†}$$

But clearly $h'(\neg Y\top) = \{0\}$ for any assignment h', so (†) holds if and only if $m = i$ or $m-i-2, m-i-3 \in h_\sigma(s)$. By the inductive hypothesis, this holds if and only if $m = i$, or $m-i-2 \in h_\rho(Y^{(m-i-2) \bmod p}\chi(m-i-2))$ and similarly for $m-i-3$: i.e., if and only if $m = i$ or $((m-i-2)_p \in h_\rho(\chi(m-i-2))$ and $(m-i-3)_p \in h_\rho(\chi(m-i-3)))$. But by choice of χ, this holds if and only if $m = i$ or $m-i-3 \equiv p-3 \pmod p$ – i.e., if and only if $m \equiv i \pmod p$. This proves the claim. □

Proof of Lemma 5.3 (*continued*). We now prove $(*)$ for m. Let $i = m \bmod p$. Now $m \in h_\sigma(s)$ if and only if $m \in h_\sigma(C)$, so by the claim, we see that we must prove

$$m \in h_\sigma(Y^i[\chi(i)(Y^p r_j / Y^{p-l_j} s : j \leqslant n)]) \Leftrightarrow m \in h_\rho(Y^i \chi(m)),$$

or equivalently,

$$m_p \in h_\sigma(\chi(m)(Y^p r_j / Y^{p-l_j} s : j \leqslant n)) \Leftrightarrow m_p \in h_\rho(\chi(m)).$$

Now $\chi(m)$ is $\top, \bot$ or a normal formula B_k in which all occurrences of the r_j are in normal subformulas of the form $Y^p r_j$, which in C are replaced by $Y^{p-l_j} s$. So as $h_\sigma =_{\bar{r}, s} h_\rho$, we need only check that

$$m_p \in h_\sigma(Y^{p-l_j} s) \Leftrightarrow m_p \in h_\rho(Y^p r_j) \quad \text{for each } j \leqslant n. \tag{‡}$$

So let $1 \leqslant j \leqslant n$. First assume that $m < p$, so that $m_p = 0$. Now $p - l_j \geqslant 1$, so that $0 \notin h_\sigma(Y^{p-l_j} s)$; and clearly $0 \notin h_\rho(Y^p r_j)$. Hence, (‡) holds in this case. Now assume $m \geqslant p$, so that $m_p \geqslant p$. Then $m_p \in h_\sigma(Y^{p-l_j} s)$ if and only if $m_p - p + l_j \in h_\sigma(s)$. As $l_j < p$, $m_p - p + l_j < m$, so (using $(*)$ inductively) this holds if and only if

$m_p - p + l_j \in h_\rho(Y^{l_j}\chi(m_p - p + l_j))$, if and only if $m_p - p \in h_\rho(\chi(l_j)) = h_\rho(B_j) = h_\rho(r_j)$, if and only if $m_p \in h_\rho(Y^p r_j)$. Thus (‡) is proved.

So (∗) holds for m, and hence by induction it holds for all $m \in \mathbb{N}$, completing the proof of the lemma. □

6. Decidability and expressive power

Now we return to the full UYF. We will prove that it has the same expressive power as the monadic second-order logic $S1S$, *the second-order theory of one successor function*. Our argument uses automata and is reminiscent of that of McNaughton [13], though much less sophisticated. Decidability of UYF will then follow from the known decidability of $S1S$. Gurevich [10] surveys the necessary general knowledge on $S1S$ and automata.

We first explain the need to invoke second-order logic. An older temporal logic, "US", introduced by Kamp in [12], involves in addition to the boolean connectives the binary temporal connective *Until* and its dual *Since*. There is no fixed point operator in US. Now consider the monadic first-order logic over $\mathbb{N}$. Its signature consists of the first-order signature $\{=, <\}$, augmented with monadic predicate variables $Q(x)$, $R(x)$, ... associated with the atoms $q, r, \ldots$ of Section 2. The semantics are those of first-order logic in the structure $(\mathbb{N}, <, h)$. The assignment h provides the semantics of the unary predicate variables, so that if $Q(x)$ is associated with the atom q, and $n \in \mathbb{N}$, then $(\mathbb{N}, <, h) \models Q(n)$ if and only if $n \in h(q)$.

It is easily seen by induction on A that for any formula $A(q_1, \ldots, q_k)$ of US there is an *equivalent* monadic first-order formula $\psi_A(x, Q_1, \ldots, Q_k)$: for all h and $n \in \mathbb{N}$, $n \in h(A)$ if and only if $(\mathbb{N}, <, h) \models \psi_A(n)$. Kamp proved that the converse also holds: for each monadic first-order formula there is an ($\mathbb{N}$-) equivalent US-formula. Other proofs are in [6, 7, 9]. Thus, we say that US is *fully expressive* with respect to monadic first-order logic over $\mathbb{N}$.

The need for second-order logic when treating the fixed point operator is prompted by the following observation of Wolper.

Proposition 6.1. (1) *Assume that for every atom q, either $h(q)$ is finite or $\mathbb{N} \setminus h(q)$ is finite (in the latter case we say $h(q)$ is cofinite). Then for all formulas A of US, $h(A)$ is either finite or cofinite.*

(2) *If $\psi(x)$ is a monadic first-order formula, and $h(q)$ is finite or cofinite for each Q occurring in ψ, then so is $\{n \in \mathbb{N}\colon (\mathbb{N}, <, h) \models \psi(n)\}$.*

(3) *There is no first-order formula $\varepsilon(x)$ in the signature $\{=, <\}$ such that for all $n \in \mathbb{N}$, $(\mathbb{N}, <) \models \varepsilon(n)$ if and only if n is even.*

Proof. (1) By induction on the complexity of A. For atomic A we are given the result. The set of finite and cofinite subsets of $\mathbb{N}$ is closed under the boolean operations, so the only remaining cases are $U(A, B)$ and $S(A, B)$. Assume inductively that $h(A)$ and

$h(B)$ are finite or cofinite. Inspection of the semantics of U shows that if $n \in h(U(A,B))$ then there is $m > n$ in $h(A)$, and that $n+1 \in h(A)$ implies $n \in h(U(A,B))$. Hence, $h(U(A,B))$ is finite or cofinite according as $h(A)$ is. Now assume, for contradiction, that $h(S(A,B))$ is neither finite nor cofinite. Hence, there are infinitely many $n \in \mathbb{N}$ such that $n \notin h(S(A,B))$ but $n+1 \in h(S(A,B))$. Inspection of the semantics of S shows that $n \in h(A)$ for each such n. Hence $h(A)$ is infinite, and so, by the inductive hypothesis, cofinite. But $h(S(A,B)) \supseteq h(A)+1$, so $h(S(A,B))$ is also cofinite, a contradiction. This completes the proof.

(2) This is immediate from (1) and Kamp's result.

(3) This is a special case of (2). □

But the UYF-formula $\tau = \varphi q \neg Yq$ satisfies $h(\tau) = \{0, 2, 4, \ldots\}$ for all h, so the expressive power of UYF goes beyond that of first-order logic over $\mathbb{N}$.

Recall the definition of $S1S$. The signature of this logic is as for monadic first-order logic (above). The formation rules for $S1S$-formulas are also as for first-order logic, but with the additional clause: if ψ is a formula and Q a monadic predicate variable then $\exists Q\psi$ is a formula. The semantics of $S1S$ are as for monadic first-order logic in $(\mathbb{N}, <, h)$, with the additional second-order clause: if $\psi(x_1, \ldots, x_n)$ is a formula, and $\bar{a} = (a_1, \ldots, a_n) \in \mathbb{N}^n$, then $(\mathbb{N}, <, h) \models \exists Q\psi(\bar{a})$ if and only if $(\mathbb{N}, <, h') \models \psi(\bar{a})$ for some $h' =_q h$.

We want to compare the expressive power of UYF and $S1S$. The natural definition to make is the following.

Definition 6.2. Let A be a UYF-formula and $\psi(x)$ a formula of $S1S$ with a single free variable, x. We say that A and ψ are *equivalent* if for all assignments h, $h(A) = \{n \in \mathbb{N} : (\mathbb{N}, <, h) \models \psi(n)\}$.

It is easy to show that $S1S$ is at least as expressive as UYF.

Proposition 6.3. *Let A be any UYF-formula. Then there is a formula $\psi_A(x)$ of $S1S$ that is equivalent to A. There is an algorithm that constructs ψ_A from A.*

Proof. By induction on A. If A is the atom q we let $\psi_A(x)$ be $Q(x)$, and we let $\psi_\top(x)$ be $x = x$. The boolean clauses are as expected, $\psi_{YA}(x)$ is $\exists y < x(\psi_A(y) \wedge \neg\exists z(y < z < x))$, and $\psi_{U(A,B)}(x)$ is $\exists y > x(\psi_A(y) \wedge \forall z(x < z < y \rightarrow \psi_B(z)))$. Finally, assume that φqA is well-formed and suppose that we have defined $\psi_A(x)$. The fixed point theorem shows that for any $h, h(\varphi qA)$ is the unique $S \subseteq \mathbb{N}$ such that $h_{q/S}(A) = S$. So we can define $\psi_{\varphi qA}(x)$ as $\exists Q[Q(x) \wedge \forall y(Q(y) \leftrightarrow \psi_A(y))]$. A standard induction on A now shows that ψ_A is always equivalent to A, and clearly the construction of ψ_A from A is effective. □

Corollary 6.4. *UYF is decidable: there is an algorithm that, given a formula A of UYF, decides whether or not there is an assignment h such that $h(A) \neq \emptyset$.*

Proof. This is because there is an algorithm that decides whether or not a sentence of $S1S$ has a model (see [10]). We can apply this algorithm to the sentence $\exists x\psi_A(x)$, which by the proposition is effectively constructible from A. $\square$

In Section 7 we will show that the decision problem for UYF is PSPACE-complete.

The remainder of this section is devoted to proving the converse of Proposition 6.3.

Theorem 6.5. *For any formula of $S1S$ with a single free variable, there is an equivalent UYF-formula, which is effectively constructible.*

This will establish that UYF is "fully expressive" with respect to monadic second-order logic. The proof will use automata.

Definition 6.6. (1) Let L be a finite set of atoms. A *(Muller) L-automaton* is a 4-tuple $M=(S, s_0, T, F)$ where S is a finite nonempty set (of *states*), $s_0 \in S$ is the *initial state*, $T \subseteq S \times \wp L \times S$ is the *transition table*, and $F \subseteq \wp S$ is the set of *accepting conditions*.

(2) M (as above) is said to be *deterministic* if for all $s \in S$ and $X \subseteq L$ there is a unique $s' \in S$ with $(s, X, s') \in T$. In this case we will often regard T as a function: $S \times \wp L \to S$.

(3) If M is an automaton as above, and h is an assignment, a *run* of M over h is a sequence $(s_n\colon n<\omega)$ of states, such that for all n,

$$(s_n, \{q \in L\colon n \in h(q)\}, s_{n+1}) \in T.$$

The run is said to be *accepting* if

$$\{s \in S\colon s = s_n \text{ for infinitely many } n < \omega\} \in F.$$

Otherwise, it is said to be *rejecting*.

(4) M is said to *accept* an assignment h if there exists an accepting run of M over h.

(5) Two L-automata are said to be *equivalent* if they accept exactly the same assignments.

An L-automaton is thought of as running along $\mathbb{N}$: at each $n \in \mathbb{N}$ it "reads" which atoms of L are true at n under h, and chooses its next state in the light of this and its current state. It then advances to $n+1$ and the process repeats.

The main lemma that we need follows after a definition.

Definition 6.7. If h is an assignment and $m \in \mathbb{N}$, we write $h_{\geqslant m}$ for the assignment given by $h_{\geqslant m}(q) = \{n \in \mathbb{N}\colon m+n \in h(q)\}$.

Intuitively, an automaton M accepts $h_{\geqslant m}$ if and only if M would accept h if it started its run at m instead of at 0.

Lemma 6.8. *Let $M=(S,s_0,T,F)$ be a deterministic automaton. Then there is a formula A_M of UYF such that for any assignment h and $n\in\mathbb{N}$, $n\in h(A_M)$ if and only if M accepts $h_{\geqslant n}$. The formula A_M is obtainable effectively from M.*

Proof. The idea is taken from [13], and we will only sketch it. Suppose that M has k states. We first describe a deterministic automation M^* involving $k+1$ "copies" $M_0,\ldots,M_k$ of M. We describe the run of M^*. At each time $t\in\mathbb{N}$, M^* releases a dormant copy of M. Then for each $s\in S$ it checks to see if more than one currently active copy of M is in state s; if so, it renders dormant all but the longest-running copy (the one that was released first). (Thus, M^* must keep track of the order of launch of the currently active copies of M – it can do this with finitely many extra states.) After this check, at most k copies of M can be active, so there will always be at least one dormant copy. M^* then advances to $t+1$ by allowing all surviving copies of M so to advance, and the process repeats.

Consider a copy M_{i_0} of M that is released at time t_0. If it is later deactivated, this is because at some time $t_1\geqslant t_0$ it arrives in the same state as another copy M_{i_1} of M released before t_0. At all times $t\geqslant t_1$, M_{i_1} (if not itself deactivated) will be in the same state as M_{i_0} would have been in, had it survived. If M_{i_1} is later deactivated then it itself will be replaced by another copy M_{i_2}, launched earlier than M_{i_1}. The resulting sequence $M_{i_0},M_{i_1},\ldots$ of "descendants" of M_{i_0} is of length at most $k+1$, since clearly all the M_{i_j} were already active at time t_0. Let M_{i_l} be the final descendant of M_{i_0}. Then M_{i_l} is never deactivated, and as it is eventually always in the same state as M_{i_0} would have been in, we see that M accepts $h_{\geqslant t_0}$ if and only if the "run" of M_{i_0}, had it lasted, would have been accepting, if and only if the "run" of M_{i_l} is accepting. This is the condition that we have to check with UYF-formulas.

We can simulate M^* by a recursive system $\rho=(\bar{r},\bar{B})$. The atoms $\bar{r}$ involve the following.

- atoms r_{is} for each $i\leqslant k, s\in S$; r_{is} will be true at n if and only if the copy M_i of M is active and in state s at n.
- atoms o_{ij} $(i,j\leqslant k)$; o_{ij} being true at some point will mean that copies M_i and M_j of M are both currently active, and M_i was released first.
- atoms p_{ij} $(i,j\leqslant k)$; p_{ij} being true at a point will mean that at that point, copy M_i was made dormant because it was in the same state as copy M_j, which was not made dormant (i.e., M_j was the oldest copy in that state).

It is clear that, knowing the transition table of M, the values of these atoms at time $t+1$ are a fixed boolean combination of their values and of the values of the atoms of L at time t. Thus, ρ can in fact be taken to be simple of depth 1; the formulas $\bar{B}$ will not involve *Until*. By Proposition 4.3, for each atom r of $\bar{r}$ there is a YF-formula A_r such that for all h, $h(A_r)=h_\rho(r)$.

Let $D_{ii}^0=\top$, $D_{ij}^0=\bot$ if $i\neq j$, and for $d\geqslant 0$,

$$D_{ij}^{d+1}=\bigvee_{i'\neq i} U\Big((p_{i,i'}\wedge D_{i',j}^{d}),\ \bigwedge_{i''\neq i}\neg p_{i,i''}\Big).$$

D^d_{ij} says that the dth descendant of M_i is M_j. Then let

$$C_{ij}=\bigvee_{s\in S} r_{is}\wedge\bigvee_{d\leqslant k}\left[D^d_{ij}\wedge\bigwedge_{j'\neq i}\neg D^{d+1}_{i,j'}\right].$$

Then $n\in h_\rho(C_{ij})$ if and only if M_i is active at time n and its final descendant is M_j. We can also express that the run of M_j is accepting, by

$$B_j=\bigvee_{X\in F}\left(\bigwedge_{s\in X} I(r_{js})\wedge\bigwedge_{s'\in S\setminus X}\neg I(r_{j,s'})\right).$$

Here, $I(q)$ abbreviates the formula $GFq=\neg U(\neg U(q,\top),\top)$, saying that q holds infinitely often in the future. The formula A_M is now obtained from $\bigwedge_{i\leqslant k}(r_{i,s_0}\rightarrow\bigwedge_{j\leqslant k}(C_{ij}\rightarrow B_j))$ by substituting A_r for r (for each atom r of $\bar{r}$). $\square$

To complete the argument we will need the following standard results about automata.

Fact 6.9. (1) *Every automaton is equivalent to a deterministic automaton, which can be constructed effectively.* This was proved in [13].

(2) *For every sentence σ of S1S whose free monadic predicates correspond to atoms in the finite set L, there is an L-automaton M such that for all assignments h, M accepts h if and only if* $(\mathbb{N},<,h)\models\sigma$. *M can be constructed effectively from σ.* See [10], for example.

Now let $\psi(x)$ be an $S1S$-formula with a single free variable, x. We want to find a UYF-formula A equivalent to ψ. Let L be the set of atoms occurring in ψ and let Q be a monadic predicate variable not occurring in ψ. By Fact 6.9 we can find a deterministic $L\cup\{q\}$-automaton $M=(S,s_0,T,F)$ that accepts an assignment h if and only if $(\mathbb{N},<,h)\models\forall x(Q(x)\leftrightarrow\psi(x))$.

Definition 6.10. (1) If $s\in S$ we write M_s for any deterministic equivalent of the version of (S,s,T,F) that "guesses" values of $h(q)$. Formally, M_s is a deterministic equivalent of the L-automaton (S,s,T',F), where $T'=\{(s_1,X\cap L,s_2):(s_1,X,s_2)\in T\}$.

(2) We write A_s for the formula A_{M_s} of Lemma 6.8.

Clearly, for any h there is a unique $h^*=_q h$ that M accepts, namely the h^* satisfying $h^*(q)=\{n\in\mathbb{N}:(\mathbb{N},<,h)\models\psi(n)\}$. We can simulate the run of M over h^* from within h, obtaining the following result.

Lemma 6.11. *For any $s\in S$ there is a UYF-formula C_s such that for all assignments h and all $n\in\mathbb{N}$, $n\in h(C_s)$ if and only if the state of M at time n during its run on h^* is s.*

Proof. We define a recursive system $\rho=((r_s\colon s\in S), (B_s\colon s\in S))$ as follows. If $s\in S$ let $\pi(s)=\{(s',X)\in S\times \wp L\colon T(s',X)=s \text{ or } T(s',X\cup\{q\})=s\}$. We let

$$B_s=A_s\wedge \bigvee_{(s',X)\in\pi(s)} Y\Big(r_{s'}\wedge \bigwedge_{x\in X} x\wedge \bigwedge_{y\in L\setminus X} \neg y\Big)$$

for each $s\in S\setminus\{s_0\}$, and

$$B_{s_0}=\neg Y\top\vee\Big[A_{s_0}\wedge \bigvee_{(s,X)\in\pi(s_0)} Y\Big(r_s\wedge \bigwedge_{x\in X} x\wedge \bigwedge_{y\in L\setminus X}\neg y\Big)\Big]$$

Note that the B_s are formulas of UYF (and not YF). Nonetheless, by Proposition 4.3 we can find for each $s\in S$ a formula C_s such that for all h, $h(C_s)=h_\rho(r_s)$. Thus, it suffices to show that for any assignment h, if $s_0, s_1, \ldots$ is the (accepting) run of M on h^* then for all n and $s, n\in h_\rho(r_s)$ if and only if $s=s_n$.

This is clear if $n=0$. Assume the result for n. Let $X=\{p\in L\colon n\in h(p)\}$, and define $s^+=T(s_n, X\cup\{q\})$, $s^-=T(s_n,X)$. By definition of the B_s and the inductive hypothesis, for each $s\in S$ we have: $n+1\in h_\rho(r_s)$ if and only if

(1) $s=s^+$ or $s=s^-$, and

(2) $n+1\in h(A_s)$.

And (2) holds if and only if M_s accepts $h_{\geqslant n+1}$, if and only if there is $h' =_q h$ that M accepts "when started at $n+1$ in state s".

Claim. (1) *and* (2) *hold if and only if* $s=s_{n+1}$.

Proof of claim. Certainly, $s_{n+1}\in\{s^+,s^-\}$, and M obviously accepts h^* when started in state s_{n+1} at $n+1$ (its run is just $s_{n+1}, s_{n+2},\ldots$). Conversely, suppose we are given $s\in\{s^+,s^-\}$ and h' as above. We can assume that $h'(q)$ and $h^*(q)$ agree before n, and that $T(s_n, h'^{-1}(n)\cap(L\cup\{q\}))=s$. An accepting run of M on $h'_{\geqslant n+1}$ when started in state s, when preceded by $s_0,\ldots,s_n$, gives an accepting run of M on h'. By uniqueness, $h'=h^*$, and so $s=T(s_n, h^{*-1}(n)\cap(L\cup\{q\}))=s_{n+1}$, as required. This proves the claim, and with it the lemma. □

Corollary 6.12. *There is a UYF-formula D equivalent to* $\psi(x)$.

Proof. We let D *be*

$$\bigwedge_{s\in S,\, X\subseteq L}\Big(C_s\wedge \bigwedge_{x\in X} x\wedge \bigwedge_{y\in L\setminus X}\neg y\Big)\to U(A_{T(s,X\cup\{q\})},\bot).$$

Let h be given, and let $s_0, s_1,\ldots$ be the run of M on h^*, as before. Let $n\in\mathbb{N}$, and write X for $\{p\in L\colon n\in h(p)\}$. Then $n\in h(D)$ if and only if $n\in h(U(A_{T(s_n,X\cup\{q\})},\bot))$, if and only if $n+1\in h(A_{T(s_n,X\cup\{q\})})$. By the claim of Lemma 6.11, this holds if and only if $s_{n+1}=T(s_n,X\cup\{q\})$. Now $s_{n+1}=T(s_n,X\cup\{q\})$ or $T(s_n,X)$, according as $n\in h^*(q)$ or not. If we had $T(s_n,X\cup\{q\})=T(s_n,X)$, we could replace h^* by h', identical with h^*

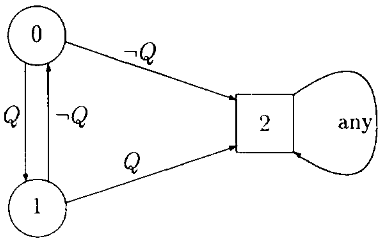

Fig. 1.

except that $n \in h'(q) \Leftrightarrow n \notin h^*(q)$; then $s_0, s_1, \ldots$ would be an accepting run of M over h', contradicting the uniqueness of h^*. Hence $T(s_n, X \cup \{q\}) \neq T(s_n, X)$. So $n \in h(D)$ if and only if $n \in h^*(q)$ – i.e., if and only if $(\mathbb{N}, <, h) \models \psi(n)$, as required. □

Thus, the proof of Theorem 6.5 is complete. We remark that the formula D of the theorem is effectively obtainable from ψ (since M and the M_s are). By Theorem 5.1, D can in fact be taken to have depth of nesting of fixed point operators of 2 (i.e., no branch of the formation tree of D contains more than two φ's). Thus, combining Proposition 6.3 and Theorem 6.5, we see that any UYF-formula is effectively equivalent to such a formula. I do not know if one can be constructed in polynomial time.

Example 6.13. We give a very simple example of our construction. We will construct the formula D equivalent to the $S1S$-formula $\varepsilon(x) = \exists P[P(x) \wedge \forall y(P(y) \leftrightarrow \neg\exists z(z<y \wedge \neg\exists t(z<t<y) \wedge P(z)))]$, defining the even numbers in $\mathbb{N}$. The automaton M for $\forall x(Q(x) \leftrightarrow \varepsilon(x))$ can be taken to have three states, 0 (initial state), 1 and 2 as shown in Fig. 1.

The set F of accepting conditions is $\{\{0,1\}\}$. The version of M that "guesses" Q can accept if and only if it is initiated in state 0 or 1. Hence, we can take $A_0 = A_1 = \top$ and $A_2 = \bot$. As $L = \emptyset$, we obtain $B_0 = \neg Y\top \vee Yr_1$, $B_1 = Yr_0$, and $B_2 = \bot$. Applying Proposition 4.3 to the recursive system $(\bar{r}, \bar{B})$, we obtain (up to equivalence) $C_0 = E$, $C_1 = YE$ and $C_2 = \bot$, where $E = \varphi r_0(\neg Y\top \vee Y^2 r_0)$. Observe that E is true at even numbers only. The formula D is thus $\bigwedge_{i \leqslant 2}(C_i \rightarrow TA_{i'})$, where Tq ("tomorrow, q") abbreviates $U(q, \bot)$, and i' is the next state of M when currently it is in state i and Q is true. Since $0' = 1$ and $1' = 2' = 2$, we obtain $D = (E \rightarrow T\top) \wedge (YE \rightarrow T\bot) \wedge (\bot \rightarrow T\bot)$, which is equivalent to $\top \wedge \neg YE \wedge \top$, i.e., to E. Hence, we obtain the formula E as equivalent to $\varepsilon(x)$.

7. Eliminating fixed point operators

In this section we show how to eliminate all fixed point operators from a UYF-formula A, at the cost of restricting the values assigned to the bound atoms of A.

This will yield an axiomatisation of UYF, as well as giving the complexity of its decision problem. Our main tool is Proposition 7.2, a variant of which was also proved independently by Strulo [17].

We can assume that the bound atoms of A are distinct and do not occur free in A. We let $A\dagger = \bigwedge(q \leftrightarrow \hat{B})$, where the conjunction is taken over all subformulas of A of the form φqB, and $\hat{A}$ is as defined in the proof of Theorem 4.4. Equivalently, we can define $A\dagger$ by induction: $q\dagger = \top\dagger = \top$ for any atom q, $(\neg A)\dagger = (YA)\dagger = A\dagger$, $(A \wedge B)\dagger = (U(A,B))\dagger = A\dagger \wedge B\dagger$, and $(\varphi qA)\dagger = A\dagger \wedge (q \leftrightarrow \hat{A})$.

Lemma 7.1. *Let A be a formula of UYF, and let h be an assignment such that $h(A\dagger) = \mathbb{N}$. Then for all subformulas B of A, $h(B) = h(\hat{B})$.*

Proof. By induction on B. We need only check the case $B = \varphi qC$, as the other cases are simple. Assume that φqC is a subformula of A, and that the result holds for C. Then $q \leftrightarrow \hat{C}$ is a conjunct of $A\dagger$. Hence $h(q) = h(\hat{C})$, and $h(\hat{C}) = h(C)$ by the inductive hypothesis. By the fixed point theorem (2.8), $h(\varphi qC) = h(q)$, and $h(q) = h(\hat{C}) = h(\widehat{\varphi qC})$, as required. □

Proposition 7.2. *Let A be any UYF-formula with bound atoms $\bar{r}$. Then for any assignment h there is a unique assignment $h^A =_{\bar{r}} h$ satisfying $h^A(A\dagger) = \mathbb{N}$. We have $h^A(\hat{A}) = h(A)$ for all h.*

Proof. By induction on A. If A is atomic or $\top$, then $\bar{r}$ is empty, $A\dagger = \top$ and $\hat{A} = A$, so $h^A = h$ and the result is trivial. Assume the result for A. As YA has the same bound atoms as A, and $(YA)\dagger = A\dagger$, we must have $h^{YA} = h^A$ by the uniqueness part of the inductive hypothesis. But then, $h^{YA}(\widehat{YA}) = h^A(Y\hat{A}) = 1 + h^A(\hat{A}) = 1 + h(A)$ by the inductive hypothesis, and this last is $h(YA)$, as required. The proof for $\neg A$ is similar. Now consider $U(A,B)$. Suppose the result holds for A, B, having bound atoms $\bar{a}, \bar{b}$ respectively. There are unique $h^A =_{\bar{a}} h$, $h^B =_{\bar{b}} h$ with $h^A(A\dagger) = h^B(B\dagger) = \mathbb{N}$. As $\bar{a}$ and $\bar{b}$ are disjoint, we can define $h^* =_{\bar{a}\bar{b}} h$ by $h^* =_{\bar{b}} h^A$, $h^* =_{\bar{a}} h^B$. Then $h^*(U(A,B)\dagger) = \mathbb{N}$, and h^* is unique given this condition. Moreover, we have $h^*(u(A,B)^\wedge) = h^*(U(\hat{A},\hat{B}))$, which by the inductive hypothesis is $h(U(A,B))$, as required. The argument for $A \wedge B$ is similar.

Finally, assume the result for A and consider φqA. Let $h(\varphi qA) = S$, and $h^* = (h_{q/S})^A$. Then (i) $h^*(\widehat{\varphi qA}) = (h_{q/S})^A(\hat{A})$; by the inductive hypothesis this is $h_{q/S}(A)$, which is $h(\varphi qA)$, by the fixed point theorem. Also, (ii) $h^*(A\dagger) = \mathbb{N}$ since h^* is of the form h'^A; and it follows from (i) that (iii) $h^*(q) = S = h^*(\hat{A})$. From (ii) and (iii) we obtain (iv) $h^*((\varphi qA)\dagger) = \mathbb{N}$. Hence, by (i) and (iv) we can let $h^{\varphi qA} = h^*$.

It remains to check uniqueness in this case. Suppose h' is any assignment such that $h' =_{\bar{r},q} h$ (where $\bar{r}$ are the bound atoms of A) and $h'((\varphi qA)\dagger) = \mathbb{N}$. Then in particular, $h'(q) = h'(\hat{A})$. As $h'(A\dagger) = \mathbb{N}$, Lemma 7.1 applies, and we get $h'(\hat{A}) = h'(A)$. But $h'(q) = h'(A)$ yields $h'(q) = S$ by the uniqueness part of the fixed point theorem. Hence $h' =_{\bar{r}} h_{q/S}$, and since $h'(A\dagger) = \mathbb{N}$, we obtain $h' = (h_{q/S})^A = h^{\varphi qA}$ by the uniqueness part of the inductive hypothesis. □

We say that a formula A is *valid if* $h(A) = \mathbb{N}$ for all assignments h.

Theorem 7.3. *Let A be any UYF-formula. Then A is valid if and only if the UYS-formula $\Box(A\dagger) \to \hat{A}$ is valid, where $\Box(A\dagger)$ abbreviates $A\dagger \wedge \neg U(\neg A\dagger, \top) \wedge \neg S(\neg A\dagger, \top)$.*

Proof. Let h be any assignment, and let $t \in \mathbb{N}$. Assume first that A is valid. If $t \in h(\Box(A\dagger))$, we want to show that $t \in h(\hat{A})$. Clearly, $h(A\dagger) = \mathbb{N}$, so by Lemma 7.1, $h(\hat{A}) = h(A)$, which is $\mathbb{N}$ since A is valid. Hence $t \in h(\hat{A})$, as required.

Conversely, if $\Box(A\dagger) \to \hat{A}$ is valid, we need to show that $t \in h(A)$. By validity, $t \in h^A(\Box(A\dagger) \to \hat{A})$. By Proposition 7.2, $h^A(A\dagger) = \mathbb{N}$, so $t \in h^A(\Box(A\dagger))$. Hence $t \in h^A(\hat{A})$, and as the proposition proved that $h^A(\hat{A}) = h(A)$, we obtain the result. □

Thus, in a sense, any UYF-formula A is "equivalent" to the UYS-formula $\Box(A\dagger) \to \hat{A}$, which does not involve φ at all. We can draw two corollaries from this.

Corollary 7.4. *There is a sound and complete axiomatisation of UYF.*

Proof. We can regard any UYS-formula as a formula of US, since the formula Yq is equivalent to $S(q, \bot)$. Sound and complete axiomatisations for US over the natural numbers exist: see [14, 18] and (for U only) [9]. If we add to such an axiomatisation the inference rule

$$\frac{\Box(A\dagger) \to \hat{A}}{A},$$

we obtain a sound and complete axiomatisation of UYF over $\mathbb{N}$. For, by the theorem, the new rule is sound: if the top is valid, so is the bottom. For completeness, assume that A is a valid UYF-formula. By Theorem 7.3, $\Box(A\dagger) \to \hat{A}$ is a valid US-formula. By completeness for US-formulas, it is provable, and we obtain $\vdash A$ from the new rule.

Notice that we only need use the new rule once, at the end of a proof of A, as the rule simply says that it suffices to prove $\Box(A\dagger) \to \hat{A}$, and this can be done (if at all) within the US proof system. □

Problem. Find a finite axiomatisation of UYF using only the conventional inference rules.

Corollary 7.5. *The decision problem for UYF is* PSPACE*-complete, as is the decision problem for USF.*

Proof. It is shown in [16] that the problem of deciding whether a formula of US is valid over $\mathbb{N}$ is PSPACE-complete. So the decision problem for USF is certainly PSPACE-hard. By Remark 2.9, this problem reduces in polynomial time to the decision problem for UYF. But by Theorem 7.3, a UYF-formula A is valid if and only if the UYS-formula $B = \Box(A\dagger) \to \hat{A}$ is valid, and B is obtainable in polynomial time from A. By replacing all subformulas YD of B by $S(D, \bot)$, as in the preceding corollary, we can obtain from B, in polynomial time, an equivalent formula C of US. By the result of [16], the validity of C over $\mathbb{N}$ is decidable in PSPACE.

Of course, the dual, satisfiability problems for USF and UYF are also PSPACE-complete. □

8. Executable temporal logic

Finally, we discuss the connection of UYF with executable temporal logics such as MetateM [3]. The MetateM computer system (see [3, 15] for details) is able to "execute" any formula A of US, building an assignment h such that $h(A)=\mathbb{N}$. A is regarded as a "specification", which the system meets by making it true at all times. The method relies on Gabbay's "separation property" for US over $\mathbb{N}$, proved in [6, 8].

Future versions of MetateM will interact with the external environment. Given a US-formula $A(\bar{q},\bar{r})$, the atoms $\bar{q}$ may be designated as under the control of the environment, whilst $\bar{r}$ remain under the auspices of the system. Given an assignment h, such a system would build a new assignment $h^* =_{\bar{r}} h$ such that $h^*(A)=\mathbb{N}$. (Here we assume that the environment does not *react* to the system, so that h can be assumed fixed during its operation. If this assumption is not valid, a logic such as $S2S$ is needed to analyse the execution; see [1, 10].)

Corollary 8.1. *Assume we have a version of MetateM as above. Then for any assignment h and subset $S\subseteq\mathbb{N}$, if $S=h(A)$ for some formula $A(\bar{q})$ of UYF, then there is a formula $A^*(\bar{q},\bar{r},s)$ of US such that if MetateM treats $\bar{q}$ as environment-controlled atoms and $\bar{r}$, s as system-controlled, and executes A^* over h to construct h^*, then $h^*(s)=S$.*

Proof. Given any formula $A(\bar{q})$ of UYF, with bound atoms $\bar{r}$, we can effectively construct the UY-formula

$$A^*(\bar{q};\bar{r},s) \stackrel{\text{def}}{=} A\dagger \wedge (s \leftrightarrow \hat{A}).$$

If MetateM constructs h^* from h as above, then we will have $h^*(A^*)=\mathbb{N}$. Hence $h^*(A\dagger)=\mathbb{N}$, so that by Proposition 7.2, $h(A)=h^*(\hat{A})$. Since also $h^*(\hat{A})=h^*(s)$, the proof is complete. □

So MetateM would in principle be able to evaluate any formula of UYF, and so by Theorem 6.5 to construct any set that is definable from the sets $h(q)$ (q *in* $\bar{q}$) by some $S1S$-formula. It is striking that the expressive power of monadic second-order logic – and of the fixed point logic UYF – would thus be achieved using only formulas of US, which have first-order definitions.

We might ask whether a converse of the corollary can be established. This depends on the execution strategy adopted by MetateM. For example, the formula $GFs=\neg U(\neg U(s,\top),\top)$ is validated by any h such that $h(s)$ is infinite; and there are 2^{ω} such h. Even if the execution strategy is assumed to be deterministic, whether it yields an h such that $h(s)$ is definable in UYF in all cases like this remains to be seen. (We believe that the answer will be positive.)

A related, speculative point concerns the so-called *uniformisation problem* for $S1S$. Given a sentence $\psi(P,Q)$ of $S1S$ such that $\mathbb{N} \models \forall P \exists Q \psi(P,Q)$, we can ask if there is another $S1S$-formula $\chi(P,Q)$ such that $\mathbb{N} \models \forall P \exists! Q \chi \wedge \forall PQ(\chi \rightarrow \psi)$. (Here, $\exists! Q$ means that there exists a unique Q.) This question was answered affirmatively in [4]; see also [10]. The corresponding problem for $S2S$ was given a negative solution in [11].

Now by Theorem 6.5 we can obtain a UYF-formula $A(p,q)$ equivalent to $\psi \wedge x = x$. Given h, we might now execute $\hat{A} \wedge A\dagger$, treating p as an environment atom and the rest as system atoms. We would obtain h^* such that $h^*(A) = \mathbb{N}$, so that $(\mathbb{N}, <, h^*) \models \psi$. Thus, a MetateM interpreter capable of handling a fixed external environment could be used to "uniformise" ψ: for any given $h(p) \subseteq \mathbb{N}$ it would construct a set $h^*(q) \subseteq \mathbb{N}$ so that $\mathbb{N} \models \psi(h(p), h^*(q))$. Again, it remains to be seen whether the execution strategy of such a system will itself be expressible in UYF or $S1S$, but if so, an alternative proof of the uniformisation result of [4] might be obtained. (Note that MetateM does involve metalanguage features, which are discussed in [5].)

Acknowledgments

This work was partially supported by Advanced Fellowship B/ITF/266 from the UK SERC. The author would like to thank Dov Gabbay and Steven Vickers for helpful conversations, and Marcelo Finger, Mark Reynolds, Ben Strulo, and the referee for carefully reading the manuscript and making many useful suggestions.

Parts of this paper are from [8], and appear by permission of Oxford University Press.

References

[1] M. Abadi, L. Lamport and P. Wolper, Realizable and unrealizable specifications of reactive systems, in: *Proc. 16th Colloq. on Automata, Languages and Programming, 1989*, Lecture Notes in Computer Science, Vol. 372 (Springer, Berlin, 1989) 1–17.

[2] B. Banieqbal and H. Barringer, A study of an extended temporal language and a temporal fixed point calculus, Tech. Report UMCS-86-10-2, Department of Computer Science, University of Manchester, 1986.

[3] H. Barringer, M. Fisher, D.M. Gabbay, G. Gough and R.P. Owens, MetateM: a Framework for programming in temporal logic, in: *REX Workshop on Stepwise Refinement of Distributed Systems: Models, Formalisms, Correctness*, Mook, Netherlands, Lecture Notes in Computer Science, Vol. 430, (Springer, Berlin, 1989) 94–129.

[4] J.R. Büchi and L.H. Landweber, Solving sequential conditions by finite state operators, *Trans. AMS* **138** (1969) 295–311.

[5] M. Fisher, T. Hunter, H. Barringer, D. Gabbay, G. Gough, I. Hodkinson, P. McBrien, R. Owens and M. Reynolds, Languages, metalanguages and MetateM, Research report, Department of Computer Science, University of Manchester, 1990.

[6] D.M. Gabbay, The declarative past and imperative future, in: B. Banieqbal et al., eds., *Proc. Colloq. on Temporal Logic and Specification*, Manchester, April 1987, Lecture Notes in Computer Science, Vol. 398 (Springer, Berlin, 1989) 67–89.

[7] D.M. Gabbay, I.M. Hodkinson and M.A. Reynolds, Temporal expressive completeness in the presence of gaps, in: *Logic Colloquium (ASL 1990 Conf. Helsinki)*, Lecture Notes in Logic, Vol. 1 (Springer, Berlin, 1993) 89–121.

[8] D.M. Gabbay, I.M. Hodkinson and M.A. Reynolds, *Temporal Logic, Vol. 1* (Oxford Univ. Press, Oxford, 1994).

[9] D.M. Gabbay, A. Pnueli, S. Shelah and J. Stavi, On the temporal analysis of fairness, in: *Proc. 7th ACM Symp. on Principles of Programming Languages*, Las Vegas (1980) 163–173.

[10] Y. Gurevich, Monadic second-order theories, in: J. Barwise and S. Feferman, eds., *Model-Theoretic Logics* (Springer, New York, 1985) 479–506.

[11] Y. Gurevich and S. Shelah, Rabin's uniformisation problem, *J. Symbolic Logic* **48** (1983) 1105–1119.

[12] J.A.W. Kamp, Tense logic and the theory of linear order, Ph.D. Thesis, University of California, 1968.

[13] R. McNaughton, Testing and generating infinite sequences by finite automata, *Inform. and Control* **9** (1966) 521–530.

[14] M. Reynolds, Axiomatizing U and S over integer time, Seminar notes, Imperial College, 1992.

[15] M. Reynolds, MetateM in intensive care, Department of Computing Technical Report, Imperial College; revised version in *J. Appl. Non-classical Logics*, to appear.

[16] A. Sistla and E. Clarke, Complexity of propositional linear temporal logics, *J. ACM* **32** (1985) 733–749.

[17] B. Strulo, Probabilistic temporal logic, M.Sc. Thesis, Imperial College, 1990.

[18] Y. Venema, Completeness via completeness, in: M. de Rijke, ed., *Colloq. on Modal Logic*, 1991, ITLI-Network Publication, Inst. for Lang., Logic and Information, University of Amsterdam, 1991.

[19] G. Winskel, *The Formal Semantics of Programming Languages* (MIT Press, Cambridge, MA, 1993).

[20] P. Wolper, Temporal logic can be more expressive, *Inform. and Comput.* **56** (1983) 72–99.