

An Efficient Normalisation Procedure for Linear Temporal Logic: Isabelle/HOL Formalisation

Salomon Sickert

May 9, 2020

Abstract

In the mid 80s, Lichtenstein, Pnueli, and Zuck proved a classical theorem stating that every formula of Past LTL (the extension of LTL with past operators) is equivalent to a formula of the form $\bigwedge_{i=1}^n \mathbf{GF}\varphi_i \vee \mathbf{FG}\psi_i$, where φ_i and ψ_i contain only past operators [3, 6]. Some years later, Chang, Manna, and Pnueli built on this result to derive a similar normal form for LTL [2]. Both normalisation procedures have a non-elementary worst-case blow-up, and follow an involved path from formulas to counter-free automata to star-free regular expressions and back to formulas. We improve on both points. We present an executable formalisation of a direct and purely syntactic normalisation procedure for LTL yielding a normal form, comparable to the one by Chang, Manna, and Pnueli, that has only a single exponential blow-up.

Contents

1	Overview	2
2	A Normal Form for Linear Temporal Logic	2
2.1	LTL Equivalences	3
2.2	$\psi[M]_1^{\Pi}$, $\psi[N]_1^{\Sigma}$, $\psi[M]_2^{\Sigma}$, and $\psi[N]_2^{\Pi}$	3
2.3	Main Theorem	6
3	Size Bounds	6
3.1	Inequalities and Identities	6
3.2	Length	7
3.3	Proper Subformulas	8
4	Code Export	10

1 Overview

This document contains the formalisation of the central results appearing in [4, Sections 4-6]. We refer the interested reader to [4] or to the extended version [5] for an introduction to the topic, related work, intuitive explanations of the proofs, and an application of the normalisation procedure, namely, a translation from LTL to deterministic automata.

The central result of this document is the following theorem:

Theorem 1. *Let φ be an LTL formula and let Δ_2 , Σ_1 , Σ_2 , and Π_1 be the classes of LTL formulas from Definition 2. Then φ is equivalent to the following formula from the class Δ_2 :*

$$\bigvee_{\substack{M \subseteq \mu(\varphi) \\ N \subseteq \nu(\varphi)}} \left(\varphi[M]_2^\Sigma \wedge \bigwedge_{\psi \in M} \mathbf{GF}(\psi[N]_1^\Sigma) \wedge \bigwedge_{\psi \in N} \mathbf{FG}(\psi[M]_1^\Pi) \right)$$

where $\psi[M]_2^\Sigma$, $\psi[N]_1^\Sigma$, and $\psi[M]_1^\Pi$ are functions mapping ψ to a formula from Σ_2 , Σ_1 , and Π_1 , respectively.

Definition 2 (Adapted from [1]). *We define the following classes of LTL formulas:*

- The class $\Sigma_0 = \Pi_0 = \Delta_0$ is the least set containing all atomic propositions and their negations, and is closed under the application of conjunction and disjunction.
- The class Σ_{i+1} is the least set containing Π_i and is closed under the application of conjunction, disjunction, and the **X**, **U**, and **M** operators.
- The class Π_{i+1} is the least set containing Σ_i and is closed under the application of conjunction, disjunction, and the **X**, **R**, and **W** operators.
- The class Δ_{i+1} is the least set containing Σ_{i+1} and Π_{i+1} and is closed under the application of conjunction and disjunction.

2 A Normal Form for Linear Temporal Logic

theory *Normal-Form* **imports**

LTL-Master-Theorem.Master-Theorem

begin

2.1 LTL Equivalences

Several valid laws of LTL relating strong and weak operators that are useful later.

lemma *ltln-strong-weak-2*:

$$\begin{aligned} w \models_n \varphi U_n \psi &\longleftrightarrow w \models_n (\varphi \text{ and}_n F_n \psi) W_n \psi \text{ (is ?thesis1)} \\ w \models_n \varphi M_n \psi &\longleftrightarrow w \models_n \varphi R_n (\psi \text{ and}_n F_n \varphi) \text{ (is ?thesis2)} \end{aligned}$$

<proof>

lemma *ltln-weak-strong-2*:

$$\begin{aligned} w \models_n \varphi W_n \psi &\longleftrightarrow w \models_n \varphi U_n (\psi \text{ or}_n G_n \varphi) \text{ (is ?thesis1)} \\ w \models_n \varphi R_n \psi &\longleftrightarrow w \models_n (\varphi \text{ or}_n G_n \psi) M_n \psi \text{ (is ?thesis2)} \end{aligned}$$

<proof>

2.2 $\psi[M]_1^\Pi$, $\psi[N]_1^\Sigma$, $\psi[M]_2^\Sigma$, and $\psi[N]_2^\Pi$

The following four functions use "promise sets", named M or N , to rewrite arbitrary formulas into formulas from the class Σ_{1-} , Σ_{2-} , Π_{1-} , and Π_2 , respectively. In general the obtained formulas are not equivalent, but under some conditions (as outlined below) they are.

no-notation *FG-advice* $(-[-]_\mu$ [90,60] 89)

no-notation *GF-advice* $(-[-]_\nu$ [90,60] 89)

notation *FG-advice* $(-[-]_{\Sigma 1}$ [90,60] 89)

notation *GF-advice* $(-[-]_{\Pi 1}$ [90,60] 89)

fun *flatten-sigma-2*:: 'a ltl \Rightarrow 'a ltl set \Rightarrow 'a ltl $(-[-]_{\Sigma 2})$

where

$$\begin{aligned} (\varphi U_n \psi)[M]_{\Sigma 2} &= (\varphi[M]_{\Sigma 2}) U_n (\psi[M]_{\Sigma 2}) \\ | (\varphi W_n \psi)[M]_{\Sigma 2} &= (\varphi[M]_{\Sigma 2}) U_n ((\psi[M]_{\Sigma 2}) \text{ or}_n (G_n \varphi[M]_{\Pi 1})) \\ | (\varphi M_n \psi)[M]_{\Sigma 2} &= (\varphi[M]_{\Sigma 2}) M_n (\psi[M]_{\Sigma 2}) \\ | (\varphi R_n \psi)[M]_{\Sigma 2} &= ((\varphi[M]_{\Sigma 2}) \text{ or}_n (G_n \psi[M]_{\Pi 1})) M_n (\psi[M]_{\Sigma 2}) \\ | (\varphi \text{ and}_n \psi)[M]_{\Sigma 2} &= (\varphi[M]_{\Sigma 2}) \text{ and}_n (\psi[M]_{\Sigma 2}) \\ | (\varphi \text{ or}_n \psi)[M]_{\Sigma 2} &= (\varphi[M]_{\Sigma 2}) \text{ or}_n (\psi[M]_{\Sigma 2}) \\ | (X_n \varphi)[M]_{\Sigma 2} &= X_n (\varphi[M]_{\Sigma 2}) \\ | \varphi[M]_{\Sigma 2} &= \varphi \end{aligned}$$

fun *flatten-pi-2*:: 'a ltl \Rightarrow 'a ltl set \Rightarrow 'a ltl $(-[-]_{\Pi 2})$

where

$$\begin{aligned} (\varphi W_n \psi)[N]_{\Pi 2} &= (\varphi[N]_{\Pi 2}) W_n (\psi[N]_{\Pi 2}) \\ | (\varphi U_n \psi)[N]_{\Pi 2} &= (\varphi[N]_{\Pi 2} \text{ and}_n (F_n \psi[N]_{\Sigma 1})) W_n (\psi[N]_{\Pi 2}) \\ | (\varphi R_n \psi)[N]_{\Pi 2} &= (\varphi[N]_{\Pi 2}) R_n (\psi[N]_{\Pi 2}) \\ | (\varphi M_n \psi)[N]_{\Pi 2} &= (\varphi[N]_{\Pi 2}) R_n ((\psi[N]_{\Pi 2}) \text{ and}_n (F_n \varphi[N]_{\Sigma 1})) \end{aligned}$$

$$\begin{aligned}
& | (\varphi \text{ and}_n \psi)[N]_{\Pi 2} = (\varphi[N]_{\Pi 2}) \text{ and}_n (\psi[N]_{\Pi 2}) \\
& | (\varphi \text{ or}_n \psi)[N]_{\Pi 2} = (\varphi[N]_{\Pi 2}) \text{ or}_n (\psi[N]_{\Pi 2}) \\
& | (X_n \varphi)[N]_{\Pi 2} = X_n (\varphi[N]_{\Pi 2}) \\
& | \varphi[N]_{\Pi 2} = \varphi
\end{aligned}$$

lemma *GF-advice-restriction:*

$$\begin{aligned}
& \varphi[\mathcal{GF} (\varphi W_n \psi) w]_{\Pi 1} = \varphi[\mathcal{GF} \varphi w]_{\Pi 1} \\
& \psi[\mathcal{GF} (\varphi R_n \psi) w]_{\Pi 1} = \psi[\mathcal{GF} \psi w]_{\Pi 1} \\
& \langle \text{proof} \rangle
\end{aligned}$$

lemma *FG-advice-restriction:*

$$\begin{aligned}
& \psi[\mathcal{FG} (\varphi U_n \psi) w]_{\Sigma 1} = \psi[\mathcal{FG} \psi w]_{\Sigma 1} \\
& \varphi[\mathcal{FG} (\varphi M_n \psi) w]_{\Sigma 1} = \varphi[\mathcal{FG} \varphi w]_{\Sigma 1} \\
& \langle \text{proof} \rangle
\end{aligned}$$

lemma *flatten-sigma-2-intersection:*

$$M \cap \text{subformulas}_\mu \varphi \subseteq S \implies \varphi[M \cap S]_{\Sigma 2} = \varphi[M]_{\Sigma 2} \\
\langle \text{proof} \rangle$$

lemma *flatten-sigma-2-intersection-eq:*

$$M \cap \text{subformulas}_\mu \varphi = M' \implies \varphi[M']_{\Sigma 2} = \varphi[M]_{\Sigma 2} \\
\langle \text{proof} \rangle$$

lemma *flatten-sigma-2-monotone:*

$$w \models_n \varphi[M]_{\Sigma 2} \implies M \subseteq M' \implies w \models_n \varphi[M']_{\Sigma 2} \\
\langle \text{proof} \rangle$$

lemma *flatten-pi-2-intersection:*

$$N \cap \text{subformulas}_\nu \varphi \subseteq S \implies \varphi[N \cap S]_{\Pi 2} = \varphi[N]_{\Pi 2} \\
\langle \text{proof} \rangle$$

lemma *flatten-pi-2-intersection-eq:*

$$N \cap \text{subformulas}_\nu \varphi = N' \implies \varphi[N']_{\Pi 2} = \varphi[N]_{\Pi 2} \\
\langle \text{proof} \rangle$$

lemma *flatten-pi-2-monotone:*

$$w \models_n \varphi[N]_{\Pi 2} \implies N \subseteq N' \implies w \models_n \varphi[N']_{\Pi 2} \\
\langle \text{proof} \rangle$$

lemma *ltln-weak-strong-stable-words-1:*

$$w \models_n (\varphi W_n \psi) \iff w \models_n \varphi U_n (\psi \text{ or}_n (G_n \varphi[\mathcal{GF} \varphi w]_{\Pi 1})) \text{ (is ?lhs} \\
\iff \text{?rhs)} \\
\langle \text{proof} \rangle$$

lemma *ltln-weak-strong-stable-words-2:*

$w \models_n (\varphi R_n \psi) \longleftrightarrow w \models_n (\varphi \text{or}_n (G_n \psi[\mathcal{GF} \psi w]_{\Pi 1})) M_n \psi$ (is ?lhs
 \longleftrightarrow ?rhs)

<proof>

lemma *ltln-weak-strong-stable-words:*

$w \models_n (\varphi W_n \psi) \longleftrightarrow w \models_n \varphi U_n (\psi \text{or}_n (G_n \varphi[\mathcal{GF} (\varphi W_n \psi) w]_{\Pi 1}))$

$w \models_n (\varphi R_n \psi) \longleftrightarrow w \models_n (\varphi \text{or}_n (G_n \psi[\mathcal{GF} (\varphi R_n \psi) w]_{\Pi 1})) M_n \psi$

<proof>

lemma *flatten-sigma-2-IH-lifting:*

assumes $\psi \in \text{subfrmlsn } \varphi$

assumes $\text{suffix } i w \models_n \psi[\mathcal{GF} \psi (\text{suffix } i w)]_{\Sigma 2} = \text{suffix } i w \models_n \psi$

shows $\text{suffix } i w \models_n \psi[\mathcal{GF} \varphi w]_{\Sigma 2} = \text{suffix } i w \models_n \psi$

<proof>

lemma *flatten-sigma-2-correct:*

$w \models_n \varphi[\mathcal{GF} \varphi w]_{\Sigma 2} \longleftrightarrow w \models_n \varphi$

<proof>

lemma *ltln-strong-weak-stable-words-1:*

$w \models_n \varphi U_n \psi \longleftrightarrow w \models_n (\varphi \text{and}_n (F_n \psi[\mathcal{FG} \psi w]_{\Sigma 1})) W_n \psi$ (is ?lhs
 \longleftrightarrow ?rhs)

<proof>

lemma *ltln-strong-weak-stable-words-2:*

$w \models_n \varphi M_n \psi \longleftrightarrow w \models_n \varphi R_n (\psi \text{and}_n (F_n \varphi[\mathcal{FG} \varphi w]_{\Sigma 1}))$ (is ?lhs \longleftrightarrow
 ?rhs)

<proof>

lemma *ltln-strong-weak-stable-words:*

$w \models_n \varphi U_n \psi \longleftrightarrow w \models_n (\varphi \text{and}_n (F_n \psi[\mathcal{FG} (\varphi U_n \psi) w]_{\Sigma 1})) W_n \psi$

$w \models_n \varphi M_n \psi \longleftrightarrow w \models_n \varphi R_n (\psi \text{and}_n (F_n \varphi[\mathcal{FG} (\varphi M_n \psi) w]_{\Sigma 1}))$

<proof>

lemma *flatten-pi-2-IH-lifting:*

assumes $\psi \in \text{subfrmlsn } \varphi$

assumes $\text{suffix } i w \models_n \psi[\mathcal{FG} \psi (\text{suffix } i w)]_{\Pi 2} = \text{suffix } i w \models_n \psi$

shows $\text{suffix } i w \models_n \psi[\mathcal{FG} \varphi w]_{\Pi 2} = \text{suffix } i w \models_n \psi$

<proof>

lemma *flatten-pi-2-correct:*

$w \models_n \varphi[\mathcal{FG} \varphi w]_{\Pi 2} \longleftrightarrow w \models_n \varphi$

<proof>

2.3 Main Theorem

Using the four previously defined functions we obtain our normal form.

theorem *normal-form-with-flatten-sigma-2:*

$w \models_n \varphi \longleftrightarrow$
 $(\exists M \subseteq \text{subformulas}_\mu \varphi. \exists N \subseteq \text{subformulas}_\nu \varphi.$
 $w \models_n \varphi[M]_{\Sigma 2} \wedge (\forall \psi \in M. w \models_n G_n (F_n \psi[N]_{\Sigma 1})) \wedge (\forall \psi \in N. w$
 $\models_n F_n (G_n \psi[M]_{\Pi 1}))$) (**is** ?lhs \longleftrightarrow ?rhs)
 ⟨proof⟩

theorem *normal-form-with-flatten-pi-2:*

$w \models_n \varphi \longleftrightarrow$
 $(\exists M \subseteq \text{subformulas}_\mu \varphi. \exists N \subseteq \text{subformulas}_\nu \varphi.$
 $w \models_n \varphi[N]_{\Pi 2} \wedge (\forall \psi \in M. w \models_n G_n (F_n \psi[N]_{\Sigma 1})) \wedge (\forall \psi \in N. w \models_n$
 $F_n (G_n \psi[M]_{\Pi 1}))$) (**is** ?lhs \longleftrightarrow ?rhs)
 ⟨proof⟩

end

3 Size Bounds

We prove an exponential upper bound for the normalisation procedure. Moreover, we show that the number of proper subformulas, which correspond to states very-weak alternating automata (A1W), is only linear for each disjunct.

theory *Normal-Form-Complexity imports*

Normal-Form

begin

3.1 Inequalities and Identities

lemma *inequality-1:*

$y > 0 \implies y + 3 \leq (2 :: \text{nat}) \wedge (y + 1)$
 ⟨proof⟩

lemma *inequality-2:*

$x > 0 \implies y > 0 \implies ((2 :: \text{nat}) \wedge (x + 1)) + (2 \wedge (y + 1)) \leq (2 \wedge (x$
 $+ y + 1))$
 ⟨proof⟩

lemma *size-gr-0:*

$\text{size } (\varphi :: 'a \text{ tln}) > 0$
 ⟨proof⟩

lemma *sum-associative*:

$$\text{finite } X \implies (\sum x \in X. f x + c) = (\sum x \in X. f x) + \text{card } X * c$$

<proof>

3.2 Length

We prove that the length (size) of the resulting formula in normal form is at most exponential.

lemma *flatten-sigma-1-length*:

$$\text{size } (\varphi[N]_{\Sigma 1}) \leq \text{size } \varphi$$

<proof>

lemma *flatten-pi-1-length*:

$$\text{size } (\varphi[M]_{\Pi 1}) \leq \text{size } \varphi$$

<proof>

lemma *flatten-sigma-2-length*:

$$\text{size } (\varphi[M]_{\Sigma 2}) \leq 2 \wedge (\text{size } \varphi + 1)$$

<proof>

lemma *flatten-pi-2-length*:

$$\text{size } (\varphi[N]_{\Pi 2}) \leq 2 \wedge (\text{size } \varphi + 1)$$

<proof>

definition *normal-form-length-upper-bound*

where *normal-form-length-upper-bound* φ

$$\equiv (2 :: \text{nat}) \wedge (\text{size } \varphi) * (2 \wedge (\text{size } \varphi + 1) + 2 * (\text{size } \varphi + 2) \wedge 2)$$

definition *normal-form-disjunct-with-flatten-pi-2-length*

where *normal-form-disjunct-with-flatten-pi-2-length* $\varphi M N$

$$\equiv \text{size } (\varphi[N]_{\Pi 2}) + (\sum \psi \in M. \text{size } (\psi[N]_{\Sigma 1}) + 2) + (\sum \psi \in N. \text{size } (\psi[M]_{\Pi 1}) + 2)$$

definition *normal-form-with-flatten-pi-2-length*

where *normal-form-with-flatten-pi-2-length* φ

$$\equiv \sum (M, N) \in \{(M, N) \mid M N. M \subseteq \text{subformulas}_\mu \varphi \wedge N \subseteq \text{subformulas}_\nu \varphi\}. \text{normal-form-disjunct-with-flatten-pi-2-length } \varphi M N$$

definition *normal-form-disjunct-with-flatten-sigma-2-length*

where *normal-form-disjunct-with-flatten-sigma-2-length* $\varphi M N$

$$\equiv \text{size } (\varphi[M]_{\Sigma 2}) + (\sum \psi \in M. \text{size } (\psi[N]_{\Sigma 1}) + 2) + (\sum \psi \in N. \text{size } (\psi[M]_{\Pi 1}) + 2)$$

definition *normal-form-with-flatten-sigma-2-length*

where *normal-form-with-flatten-sigma-2-length* φ

$\equiv \sum (M, N) \in \{(M, N) \mid M N. M \subseteq \text{subformulas}_\mu \varphi \wedge N \subseteq \text{subformulas}_\nu \varphi\}$. *normal-form-disjunct-with-flatten-sigma-2-length* $\varphi M N$

lemma *normal-form-disjunct-length-upper-bound:*

assumes

$M \subseteq \text{subformulas}_\mu \varphi$

$N \subseteq \text{subformulas}_\nu \varphi$

shows

normal-form-disjunct-with-flatten-sigma-2-length $\varphi M N \leq 2^{\wedge} (\text{size } \varphi + 1) + 2 * (\text{size } \varphi + 2)^{\wedge} 2$ (**is** *?thesis1*)

normal-form-disjunct-with-flatten-pi-2-length $\varphi M N \leq 2^{\wedge} (\text{size } \varphi + 1) + 2 * (\text{size } \varphi + 2)^{\wedge} 2$ (**is** *?thesis2*)

<proof>

theorem *normal-form-length-upper-bound:*

normal-form-with-flatten-sigma-2-length $\varphi \leq \text{normal-form-length-upper-bound } \varphi$ (**is** *?thesis1*)

normal-form-with-flatten-pi-2-length $\varphi \leq \text{normal-form-length-upper-bound } \varphi$ (**is** *?thesis2*)

<proof>

3.3 Proper Subformulas

We prove that the number of (proper) subformulas (sf) in a disjunct is linear and not exponential.

fun *sf* :: 'a ltn \Rightarrow 'a ltn set

where

$\text{sf } (\varphi \text{ and}_n \psi) = \text{sf } \varphi \cup \text{sf } \psi$

| $\text{sf } (\varphi \text{ or}_n \psi) = \text{sf } \varphi \cup \text{sf } \psi$

| $\text{sf } (X_n \varphi) = \{X_n \varphi\} \cup \text{sf } \varphi$

| $\text{sf } (\varphi U_n \psi) = \{\varphi U_n \psi\} \cup \text{sf } \varphi \cup \text{sf } \psi$

| $\text{sf } (\varphi R_n \psi) = \{\varphi R_n \psi\} \cup \text{sf } \varphi \cup \text{sf } \psi$

| $\text{sf } (\varphi W_n \psi) = \{\varphi W_n \psi\} \cup \text{sf } \varphi \cup \text{sf } \psi$

| $\text{sf } (\varphi M_n \psi) = \{\varphi M_n \psi\} \cup \text{sf } \varphi \cup \text{sf } \psi$

| $\text{sf } \varphi = \{\}$

lemma *sf-finite:*

finite (*sf* φ)

<proof>

lemma *sf-subset-subfrmlsn*:

$sf \varphi \subseteq subfrmlsn \varphi$
<proof>

lemma *sf-size*:

$\psi \in sf \varphi \implies size \psi \leq size \varphi$
<proof>

lemma *sf-sf-subset*:

$\psi \in sf \varphi \implies sf \psi \subseteq sf \varphi$
<proof>

lemma *subfrmlsn-sf-subset*:

$\psi \in subfrmlsn \varphi \implies sf \psi \subseteq sf \varphi$
<proof>

lemma *sf-subset-insert*:

assumes $sf \varphi \subseteq insert \varphi X$
assumes $\psi \in subfrmlsn \varphi$
assumes $\varphi \neq \psi$
shows $sf \psi \subseteq X$
<proof>

lemma *flatten-pi-1-sf-subset*:

$sf (\varphi[M]_{\Pi 1}) \subseteq (\bigcup \varphi \in sf \varphi. sf (\varphi[M]_{\Pi 1}))$
<proof>

lemma *flatten-sigma-1-sf-subset*:

$sf (\varphi[M]_{\Sigma 1}) \subseteq (\bigcup \varphi \in sf \varphi. sf (\varphi[M]_{\Sigma 1}))$
<proof>

lemma *flatten-sigma-2-sf-subset*:

$sf (\varphi[M]_{\Sigma 2}) \subseteq (\bigcup \psi \in sf \varphi. sf (\psi[M]_{\Sigma 2}))$
<proof>

lemma *sf-set1*:

$sf (\varphi[M]_{\Sigma 2}) \cup sf (\varphi[M]_{\Pi 1}) \subseteq (\bigcup \psi \in (sf \varphi). (sf (\psi[M]_{\Sigma 2}) \cup sf (\psi[M]_{\Pi 1})))$
<proof>

lemma *ltn-not-idempotent [simp]*:

$\varphi \text{ and}_n \psi \neq \varphi \psi \text{ and}_n \varphi \neq \varphi \varphi \neq \varphi \text{ and}_n \psi \varphi \neq \psi \text{ and}_n \varphi$
 $\varphi \text{ or}_n \psi \neq \varphi \psi \text{ or}_n \varphi \neq \varphi \varphi \neq \varphi \text{ or}_n \psi \varphi \neq \psi \text{ or}_n \varphi$
 $X_n \varphi \neq \varphi \varphi \neq X_n \varphi$

$\varphi U_n \psi \neq \varphi \varphi \neq \varphi U_n \psi \psi U_n \varphi \neq \varphi \varphi \neq \psi U_n \varphi$
 $\varphi R_n \psi \neq \varphi \varphi \neq \varphi R_n \psi \psi R_n \varphi \neq \varphi \varphi \neq \psi R_n \varphi$
 $\varphi W_n \psi \neq \varphi \varphi \neq \varphi W_n \psi \psi W_n \varphi \neq \varphi \varphi \neq \psi W_n \varphi$
 $\varphi M_n \psi \neq \varphi \varphi \neq \varphi M_n \psi \psi M_n \varphi \neq \varphi \varphi \neq \psi M_n \varphi$
 <proof>

lemma *flatten-card-sf-induct*:

assumes *finite X*

assumes $\bigwedge x. x \in X \implies sf\ x \subseteq X$

shows $card (\bigcup \varphi \in X. sf (\varphi[N]_{\Sigma 1})) \leq card\ X$

$\wedge card (\bigcup \varphi \in X. sf (\varphi[M]_{\Pi 1})) \leq card\ X$

$\wedge card (\bigcup \varphi \in X. sf (\varphi[M]_{\Sigma 2}) \cup sf (\varphi[M]_{\Pi 1})) \leq 3 * card\ X$

<proof>

theorem *flatten-card-sf*:

$card (\bigcup \psi \in sf\ \varphi. sf (\psi[M]_{\Sigma 1})) \leq card (sf\ \varphi)$ (**is** ?t1)

$card (\bigcup \psi \in sf\ \varphi. sf (\psi[M]_{\Pi 1})) \leq card (sf\ \varphi)$ (**is** ?t2)

$card (sf (\varphi[M]_{\Sigma 2}) \cup sf (\varphi[M]_{\Pi 1})) \leq 3 * card (sf\ \varphi)$ (**is** ?t3)

<proof>

corollary *flatten-sigma-2-card-sf*:

$card (sf (\varphi[M]_{\Sigma 2})) \leq 3 * (card (sf\ \varphi))$

<proof>

end

4 Code Export

theory *Normal-Form-Code-Export* **imports**

LTL.Code-Equations

LTL.Rewriting

LTL.Disjunctive-Normal-Form

HOL.String

Normal-Form

begin

fun *flatten-pi-1-list* :: *String.literal ltn* \Rightarrow *String.literal ltn list* \Rightarrow *String.literal ltn*

where

flatten-pi-1-list ($\psi_1 U_n \psi_2$) *M* = (if ($\psi_1 U_n \psi_2$) \in *set M* then (*flatten-pi-1-list* ψ_1 *M*) *W_n* (*flatten-pi-1-list* ψ_2 *M*) else *false_n*)

| *flatten-pi-1-list* ($\psi_1 W_n \psi_2$) *M* = (*flatten-pi-1-list* ψ_1 *M*) *W_n* (*flatten-pi-1-list* ψ_2 *M*)

$| \text{flatten-pi-1-list } (\psi_1 M_n \psi_2) M = (\text{if } (\psi_1 M_n \psi_2) \in \text{set } M \text{ then } (\text{flatten-pi-1-list } \psi_1 M) R_n (\text{flatten-pi-1-list } \psi_2 M) \text{ else } \text{false}_n)$
 $| \text{flatten-pi-1-list } (\psi_1 R_n \psi_2) M = (\text{flatten-pi-1-list } \psi_1 M) R_n (\text{flatten-pi-1-list } \psi_2 M)$
 $| \text{flatten-pi-1-list } (\psi_1 \text{and}_n \psi_2) M = (\text{flatten-pi-1-list } \psi_1 M) \text{and}_n (\text{flatten-pi-1-list } \psi_2 M)$
 $| \text{flatten-pi-1-list } (\psi_1 \text{or}_n \psi_2) M = (\text{flatten-pi-1-list } \psi_1 M) \text{or}_n (\text{flatten-pi-1-list } \psi_2 M)$
 $| \text{flatten-pi-1-list } (X_n \psi) M = X_n (\text{flatten-pi-1-list } \psi M)$
 $| \text{flatten-pi-1-list } \varphi - = \varphi$

fun *flatten-sigma-1-list* :: *String.literal ltl* \Rightarrow *String.literal ltl list* \Rightarrow *String.literal ltl*

where

$\text{flatten-sigma-1-list } (\psi_1 U_n \psi_2) N = (\text{flatten-sigma-1-list } \psi_1 N) U_n (\text{flatten-sigma-1-list } \psi_2 N)$
 $| \text{flatten-sigma-1-list } (\psi_1 W_n \psi_2) N = (\text{if } (\psi_1 W_n \psi_2) \in \text{set } N \text{ then } \text{true}_n \text{ else } (\text{flatten-sigma-1-list } \psi_1 N) U_n (\text{flatten-sigma-1-list } \psi_2 N))$
 $| \text{flatten-sigma-1-list } (\psi_1 M_n \psi_2) N = (\text{flatten-sigma-1-list } \psi_1 N) M_n (\text{flatten-sigma-1-list } \psi_2 N)$
 $| \text{flatten-sigma-1-list } (\psi_1 R_n \psi_2) N = (\text{if } (\psi_1 R_n \psi_2) \in \text{set } N \text{ then } \text{true}_n \text{ else } (\text{flatten-sigma-1-list } \psi_1 N) M_n (\text{flatten-sigma-1-list } \psi_2 N))$
 $| \text{flatten-sigma-1-list } (\psi_1 \text{and}_n \psi_2) N = (\text{flatten-sigma-1-list } \psi_1 N) \text{and}_n (\text{flatten-sigma-1-list } \psi_2 N)$
 $| \text{flatten-sigma-1-list } (\psi_1 \text{or}_n \psi_2) N = (\text{flatten-sigma-1-list } \psi_1 N) \text{or}_n (\text{flatten-sigma-1-list } \psi_2 N)$
 $| \text{flatten-sigma-1-list } (X_n \psi) N = X_n (\text{flatten-sigma-1-list } \psi N)$
 $| \text{flatten-sigma-1-list } \varphi - = \varphi$

fun *flatten-sigma-2-list* :: *String.literal ltl* \Rightarrow *String.literal ltl list* \Rightarrow *String.literal ltl*

where

$\text{flatten-sigma-2-list } (\varphi U_n \psi) M = (\text{flatten-sigma-2-list } \varphi M) U_n (\text{flatten-sigma-2-list } \psi M)$
 $| \text{flatten-sigma-2-list } (\varphi W_n \psi) M = (\text{flatten-sigma-2-list } \varphi M) U_n ((\text{flatten-sigma-2-list } \psi M) \text{or}_n (G_n (\text{flatten-pi-1-list } \varphi M)))$
 $| \text{flatten-sigma-2-list } (\varphi M_n \psi) M = (\text{flatten-sigma-2-list } \varphi M) M_n (\text{flatten-sigma-2-list } \psi M)$
 $| \text{flatten-sigma-2-list } (\varphi R_n \psi) M = ((\text{flatten-sigma-2-list } \varphi M) \text{or}_n (G_n (\text{flatten-pi-1-list } \psi M))) M_n (\text{flatten-sigma-2-list } \psi M)$
 $| \text{flatten-sigma-2-list } (\varphi \text{and}_n \psi) M = (\text{flatten-sigma-2-list } \varphi M) \text{and}_n (\text{flatten-sigma-2-list } \psi M)$
 $| \text{flatten-sigma-2-list } (\varphi \text{or}_n \psi) M = (\text{flatten-sigma-2-list } \varphi M) \text{or}_n (\text{flatten-sigma-2-list } \psi M)$

| *flatten-sigma-2-list* ($X_n \varphi$) $M = X_n$ (*flatten-sigma-2-list* φM)
| *flatten-sigma-2-list* $\varphi - = \varphi$

lemma *flatten-code-equations*[simp]:

$\varphi[\text{set } M]_{\Pi 1} = \text{flatten-pi-1-list } \varphi M$
 $\varphi[\text{set } M]_{\Sigma 1} = \text{flatten-sigma-1-list } \varphi M$
 $\varphi[\text{set } M]_{\Sigma 2} = \text{flatten-sigma-2-list } \varphi M$
⟨*proof*⟩

abbreviation *and-list* $\equiv \text{foldl And-ltn true}_n$

abbreviation *or-list* $\equiv \text{foldl Or-ltn false}_n$

definition *normal-form-disjunct* ($\varphi :: \text{String.literal ltn}$) $M N$

$\equiv (\text{flatten-sigma-2-list } \varphi M)$
 $\text{and}_n (\text{and-list } (\text{map } (\lambda\psi. G_n (F_n (\text{flatten-sigma-1-list } \psi N)))) M)$
 $\text{and}_n (\text{and-list } (\text{map } (\lambda\psi. F_n (G_n (\text{flatten-pi-1-list } \psi M)))) N))$

definition *normal-form* ($\varphi :: \text{String.literal ltn}$)

$\equiv \text{or-list } (\text{map } (\lambda(M, N). \text{normal-form-disjunct } \varphi M N)) (\text{advice-sets } \varphi)$

lemma *and-list-semantic*: $w \models_n \text{and-list } xs \longleftrightarrow (\forall x \in \text{set } xs. w \models_n x)$
⟨*proof*⟩

lemma *or-list-semantic*: $w \models_n \text{or-list } xs \longleftrightarrow (\exists x \in \text{set } xs. w \models_n x)$
⟨*proof*⟩

theorem *normal-form-correct*:

$w \models_n \varphi \longleftrightarrow w \models_n \text{normal-form } \varphi$
⟨*proof*⟩

definition *normal-form-with-simplifier* ($\varphi :: \text{String.literal ltn}$)

$\equiv \text{min-dnf } (\text{simplify Slow } (\text{normal-form } (\text{simplify Slow } \varphi)))$

lemma *ltn-semantics-min-dnf*:

$w \models_n \varphi \longleftrightarrow (\exists C \in \text{min-dnf } \varphi. \forall \psi. \psi \in C \longrightarrow w \models_n \psi)$ (**is** *?lhs* \longleftrightarrow *?rhs*)
⟨*proof*⟩

theorem

$w \models_n \varphi \longleftrightarrow (\exists C \in (\text{normal-form-with-simplifier } \varphi). \forall \psi. \psi \in C \longrightarrow w \models_n \psi)$ (**is** *?lhs* \longleftrightarrow *?rhs*)
⟨*proof*⟩

In order to export the code run `isabelle build -D [PATH] -e`.

`export-code normal-form in SML`

`export-code normal-form-with-simplifier in SML`

`end`

References

- [1] Ivana Černá and Radek Pelánek. Relating hierarchy of temporal properties to model checking. In Branislav Rován and Peter Vojtás, editors, *Mathematical Foundations of Computer Science 2003, 28th International Symposium, MFCS 2003, Bratislava, Slovakia, August 25-29, 2003, Proceedings*, volume 2747 of *Lecture Notes in Computer Science*, pages 318–327. Springer, 2003. doi:[10.1007/978-3-540-45138-9_26](https://doi.org/10.1007/978-3-540-45138-9_26).
- [2] Edward Y. Chang, Zohar Manna, and Amir Pnueli. Characterization of temporal property classes. In Werner Kuich, editor, *Automata, Languages and Programming, 19th International Colloquium, ICALP92, Vienna, Austria, July 13-17, 1992, Proceedings*, volume 623 of *Lecture Notes in Computer Science*, pages 474–486. Springer, 1992. doi:[10.1007/3-540-55719-9_97](https://doi.org/10.1007/3-540-55719-9_97).
- [3] Orna Lichtenstein, Amir Pnueli, and Lenore D. Zuck. The glory of the past. In Rohit Parikh, editor, *Logics of Programs, Conference, Brooklyn College, New York, NY, USA, June 17-19, 1985, Proceedings*, volume 193 of *Lecture Notes in Computer Science*, pages 196–218. Springer, 1985. doi:[10.1007/3-540-15648-8_16](https://doi.org/10.1007/3-540-15648-8_16).
- [4] Salomon Sickert and Javier Esparza. An efficient normalisation procedure for linear temporal logic and very weak alternating automata. In *Proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2020, Saarbrücken, Germany, July 8-11, 2020*. ACM, 2020. doi:[10.1145/3373718.3394743](https://doi.org/10.1145/3373718.3394743).
- [5] Salomon Sickert and Javier Esparza. An efficient normalisation procedure for linear temporal logic and very weak alternating automata. *CoRR*, abs/2005.00472, 2020. arXiv:[2005.00472](https://arxiv.org/abs/2005.00472).
- [6] Lenore D. Zuck. *Past Temporal Logic*. PhD thesis, The Weizmann Institute of Science, Israel, August 1986.