Verifying Featured Transition Systems using Variability Parity Games

Sjef van Loo

May 7, 2019

1 Definitions

1.1 Transition systems

Similar to [1].

Definition 1.1. *An LTS is a tuple $M = (S, Act, trans, s_0)$, where:*

- *S is a set of states,*
- *Act a set of actions,*
- *$trans \subseteq S \times Act \times S$ is the transition relation with $(s, a, s') \in trans$ denoted by $s \xrightarrow{a} s'$,*
- *$s_0 \in S$ is the initial state.*

Definition 1.2. *An FTS is a tuple $M = (S, Act, trans, s_0, N, P, \gamma)$, where:*

- *$S, Act, trans, s_0$ are defined as in an LTS,*
- *N is a set of features,*
- *$P \subseteq \mathcal{P}(N)$ is a set of products, ie. feature assignments, that are valid,*
- *$\gamma : trans \to \mathbb{B}(N)$ is a total function, labelling each transition with a Boolean expression over the features. A product $p \in \mathcal{P}(N)$ satisfying the Boolean expression of transition t is denoted by $p \models \gamma(t)$, $\gamma(t)(p) = 1$ or $p \in [\![\gamma(t)]\!]$.*

  A transition $s \xrightarrow{a} s'$ and $\gamma((s, a, s')) = f$ is denoted by $s \xrightarrow{a/f} s'$.

Definition 1.3. *The projection of an FTS M to a product $p \in P$, noted $M_{|p}$, is the LTS $M' = (S, Act, trans', s_0)$, where $trans' = \{t \in trans \mid p \models \gamma(t)\}$.*

Definition 1.4. *[3] A modal μ-calculus formula over the set of actions $\mathcal{A}$ and a set of variables $\mathcal{X}$ is defined by*

$$\varphi = \top \mid \bot \mid X \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \langle a \rangle \varphi \mid [a]\varphi \mid \mu X.\varphi \mid \nu X.\varphi$$

with $a \in \mathcal{A}$ and $X \in \mathcal{X}$.

No negations in the language because negations can be pushed inside to the propositions, ie. the $\top$ and $\bot$ elements..

A fixed point formula φ with variable X can be unfolded; the occurrences of X are replaced by φ. A fixed point formula is equivalent to its unfolding, ie. $\mu X.\varphi$ is equivalent to $\mu X.\varphi[X := \mu X.\varphi]$. [3]

Definition 1.5. *Given LTS $M = (S, Act, trans, s_0)$, state $s \in S$ and mu-calculus formula φ we write $M, s \models \varphi$ if and only if φ is satisfied in state s for LTS M. If $M, s_0 \models \varphi$ we write $M \models \varphi$.*

2 Goal

Similar to [2].

Given an FTS $M = (S, Act, trans, s_0, N, P, \gamma)$ and a modal μ-calculus formula φ we want to find the set $P_s \subseteq P$ such that:

- for every $p \in P_s$ we have $M_{|p} \models \varphi$,
- for every $p \in P \backslash P_s$ we have $M_{|p} \not\models \varphi$.

A counterexample for every $p \in P \backslash P_s$ is preferred.

If $P_s = P$, ie. all products satisfy φ, we write $M \models \varphi$.

3 Parity Games

3.1 Parity games

Definition 3.1. *[3] A parity game (PG) is a tuple $G = (V, V_0, V_1, E, \rho)$, where:*

- $V = V_0 \cup V_1$ *and* $V_0 \cap V_1 = \emptyset$,
- V_0 *is the set of vertices owned by player* 0,
- V_1 *is the set of vertices owned by player* 1,
- $E \subseteq V \times V$ *is the edge relation,*
- $\rho : V \to \mathbb{N}$ *is a priority assignment.*

We write $\alpha \in \{0, 1\}$ to denote an arbitrary player. We write $\overline{\alpha}$ to denote α's opponent, ie. $\overline{0} = 1$ and $\overline{1} = 0$.

A parity game is played by players 0 and 1. A play starts with placing a token on vertex $v \in V$. Player α moves the token if the token is on a vertex owned by α, ie. $v \in V_\alpha$. The token can be moved to $w \in V$, with $(v, w) \in E$. A series of moves results in a sequence of vertices, called path. For path π we write π_i to denote the i^{th} vertex in path π. A play ends when the token is on vertex $v \in V_\alpha$ and α can't move the token anywhere, in this case player $\overline{\alpha}$ wins the play. If the play results in an infinite path then we determine the highest priority that occurs infinitely often in $\{\rho(\pi_i) | \pi_i \in \pi\}$, if the highest priority is odd then player 1 wins, if it is even player 0 wins.

A strategy for player α is a function $\sigma : V^* V_\alpha \to V$ that maps a path ending in a vertex owned by player α to the next vertex. Parity games are positionally determined [3], therefore a strategy $\sigma : V_\alpha \to V$ that maps the current vertex to the next vertex is sufficient.

A strategy σ for player α is winning from vertex v if and only if any play that results from following σ results in a win for player α. The graph can be divided in two partitions $W_0 \subseteq V$ and $W_1 \subseteq V$, called winning sets. If and only if $v \in W_\alpha$ then player α has a winnings strategy from v. Every vertex in the graph is either in W_0 or W_1 [3]. Furthermore finite parity games are decidable [3].

3.2 Featured parity games

Definition 3.2. *A featured parity game (FPG) is a tuple $G = (V, V_0, V_1, E, \rho, N, P, \gamma)$, where:*

- $V = V_0 \cup V_1$ *and* $V_0 \cap V_1 = \emptyset$,
- V_0 *is the set of vertices owned by player* 0,
- V_1 *is the set of vertices owned by player* 1,
- $E \subseteq V \times V$ *is the edge relation,*
- $\rho : V \to \mathbb{N}$ *is a priority assignment,*
- N *is a set of features,*
- $P \subseteq \mathcal{P}(N)$ *is a set of products, ie. feature assignments, for which the game can be played/*
- $\gamma : E \to \mathbb{B}(N)$ *is a total function, labelling each edge with a Boolean expression over the features.*

An FPG is played similarly to a PG, however the game is played for a specific product $p \in P$. Player α can only move the token from $v \in V_\alpha$ to $w \in V$ if $(v, w) \in E$ and $p \models \gamma(v, w)$.

A game played for product $p \in P$ results in winnings sets W_0^p and W_1^p, which are defined similar to the W_0 and W_1 winning sets for parity games.

Definition 3.3. *The projection from FPG $G = (V, V_0, V_1, E, \rho, N, P, \gamma)$ to a product $p \in P$, noted $G_{|p}$, is the parity game (V, V_0, V_1, E', ρ) where $E' = \{e \in E | p \models \gamma(e)\}$.*

Playing FPG G for a specific product $p \in P$ is the same as playing the PG $G_{|p}$. Any path that is valid in G for p is also valid in $G_{|p}$ and vice versa. Therefore the strategies are also interchangeable, furthermore the winning sets W_α for $G_{|p}$ and W_α^p for G are identical. Since parity games are positionally determined so are FPGs. Similarly, since finite parity games are decidable, so are finite FPGs.

3.3 Variability parity games

Definition 3.4. *A variability parity game (VPG) is a tuple $G = (V, V_0, V_1, E, \rho, \mathfrak{C}, \theta)$, where:*

- $V = V_0 \cup V_1$ *and* $V_0 \cap V_1 = \emptyset$,
- V_0 *is the set of vertices owned by player* 0,
- V_1 *is the set of vertices owned by player* 1,
- $E \subseteq V \times V$ *is the edge relation; we assume that* E *is total, i.e. for all* $v \in V$ *there is some* $w \in V$ *such that* $(v, w) \in E$,
- $\rho : V \to \mathbb{N}$ *is a priority assignment,*
- $\mathfrak{C}$ *is a finite set of configurations,*
- $\theta : E \to \mathcal{P}(\mathfrak{C}) \setminus \{0\}$ *is the configuration mapping, satisfying for all* $v \in V$, $\bigcup\{\theta(v, w) | (v, w) \in E\} = \mathfrak{C}$.

A VPG is played similarly to a PG, however the game is played for a specific configuration $c \in \mathfrak{C}$. Player α can only move the token from $v \in V_\alpha$ to $w \in V$ if $(v, w) \in E$ and $c \in \theta(v, w)$. Furthermore VPGs don't have deadlocks, every play results in an infinite path.

A game played for configuration $c \in \mathfrak{C}$ results in winning sets W_0^c and W_1^c, which are defined similar to the W_0 and W_1 winning sets for parity games.

Definition 3.5. *The projection from VPG $G = (V, V_0, V_1, E, \rho, \mathfrak{C}, \theta)$ to a configuration $c \in \mathfrak{C}$, noted $G_{|c}$, is the parity game (V, V_0, V_1, E', ρ) where $E' = \{e \in E | c \in \theta(e)\}$.*

Playing VPG G for a specific configuration $c \in \mathfrak{C}$ is the same as playing the PG $G_{|c}$. Any path that is valid in G for c is also valid in $G_{|c}$ and vice versa. Therefore the strategies are also interchangeable, furthermore the winning sets W_α for $G_{|c}$ and W_α^c for G are identical. Since parity games are positionally determined so are VPGs. Similarly, since finite parity games are decidable, so are finite VPGs.

3.4 Creating parity games

Originating from an FTS and a modal μ-calculus we can create an FPG (from which we can create a PG by projection) and from the FPG we can create a VPG. For a specific product we can project the FTS to an LTS, from which we can create a PG. The relation between the transition systems and games is displayed in the following diagram.

$$\begin{tikzcd}
\text{FTS} \arrow[rr, "\varphi"] \arrow[d, Rightarrow, "\Pi"'] & & \text{FPG} \arrow[r] \arrow[d, Rightarrow, "\Pi"] & \text{VPG} \\
\text{LTS} \arrow[r, "\varphi"] & \text{PG} \arrow[r, phantom, "\subseteq"] & \text{PG} &
\end{tikzcd}$$

The projections are defined in the previous section. In this section we will define the horizontal arrows in the diagram. First we show how to create a PG from an LTS and a modal μ-calculus formula, this part is well studied and the approach is based on [3].

Definition 3.6. *[3] LTS2PG(M, φ) converts LTS $M = (S, Act, trans, s_0)$ and closed formula φ to a PG (V, V_0, V_1, E, ρ).*
A vertex in the parity game is represented by a pair (s, ψ) where $s \in S$ and ψ is a modal μ-calculus formula.
We create the parity game with the smallest sets V, V_0, V_1, E such that:

- $V = V_0 \cup V_1$,
- $V_0 \cap V_1 = \emptyset$,
- $(s_0, \varphi) \in V$ *and*
- *for every* $v = (s, \psi) \in V$ *we have:*
  - *If* $\psi = \top$ *then* $v \in V_1$.
  - *If* $\psi = \bot$ *then* $v \in V_0$.
  - *If* $\psi = \psi_1 \vee \psi_2$ *then:*
    $v \in V_0$,
    $(s, \psi_1) \in V$,
    $(s, \psi_2) \in V$,
    $(v, (s, \psi_1)) \in E$ *and*
    $(v, (s, \psi_2)) \in E$.

– *If* $\psi = \psi_1 \wedge \psi_2$ *then:*
  $v \in V_1$,
  $(s, \psi_1) \in V$,
  $(s, \psi_2) \in V$,
  $(v, (s, \psi_1)) \in E$ *and*
  $(v, (s, \psi_2)) \in E$.
– *If* $\psi = \langle a \rangle \psi_1$ *then* $v \in V_0$ *and for every* $s \xrightarrow{a} s'$ *we have* $(s', \psi_1) \in V$ *and* $(v, (s', \psi_1)) \in E$.
– *If* $\psi = [a]\psi_1$ *then* $v \in V_1$ *and for every* $s \xrightarrow{a} s'$ *we have* $(s', \psi_1) \in V$ *and* $(v, (s', \psi_1)) \in E$.
– *If* $\psi = \mu X.\psi_1$ *then* $(s, \psi_1(\mu X.\psi_1[X := \mu X.\psi_1])) \in V$ *and* $(v, (s, \psi_1(\mu X.\psi_1[X := \mu X.\psi_1]))) \in E$.
– *If* $\psi = \nu X.\psi_1$ *then* $(s, \psi_1(\nu X.\psi_1[X := \nu X.\psi_1])) \in V$ *and* $(v, (s, \psi_1(\nu X.\psi_1[X := \nu X.\psi_1]))) \in E$.

Note that since φ *is closed and we use unfolding the case where* $\psi = X$ *does not happen.*

Finally we have $\rho(s, \psi) = \begin{cases} 2\lfloor adepth(X)/2 \rfloor & \text{if } \psi = \nu X.\psi' \\ 2\lfloor adepth(X)/2 \rfloor + 1 & \text{if } \psi = \mu X.\psi' \\ 0 & \text{otherwise} \end{cases}$

Next we define the transformation from FTS to FPG.

Definition 3.7. *FTS2FPG(M, φ) converts FTS* $M = (S, Act, trans, s_0, N, P, \gamma)$ *and closed formula* φ *to FPG* $(V, V_0, V_1, E, \rho, N, P, \gamma')$
We have $(V, V_0, V_1, E, \rho) = $ *LTS2PG(*$(S, Act, trans, s_0), \varphi$*) and*

$$\gamma'((s, \psi), (s', \psi')) = \begin{cases} \gamma(s, a, s') & \text{if } \psi = \langle a \rangle \psi' \text{ or } \psi = [a]\psi' \\ \top & \text{otherwise} \end{cases}$$

Finally we define how to create a VPG from an FPG. This transformation abstracts from the notion of products and uses configuration for a syntactically more pleasant representation. Furthermore in VPGs deadlocks are removed, this is done by creating two losing vertices l_0 and l_1 such that player α loses when the token is in vertex l_α. Any vertex that can not move for a configuration will get an edge that is admissible for that configuration towards one of the losing vertices.

Definition 3.8. *FPG2VPG(*G^F*) converts FPG* $G^F = (V^F, V_0^F, V_1^F, E^F, \rho^F, N, P, \gamma)$ *to VPG* $G = (V, V_0, V_1, E, \rho, \mathfrak{C}, \theta)$.
Let P *be defined as* $\{p_0, p_1, \ldots, p_m\}$*, we define* $\mathfrak{C} = \{c_0, c_1, \ldots, c_m\}$.
We create vertices l_0 *and* l_1 *and define* $V_0 = V_0^F \cup \{l_0\}$, $V_1 = V_1^F \cup \{l_1\}$ *and* $V = V_0 \cup V_1$.
We construct E *by first making* $E = E^F$ *and adding edges* (l_0, l_0) *and* (l_1, l_1) *to* E*. Simultaneously we construct* θ *by first making* $\theta(e) = \bigcup\{c_i \in \mathfrak{C} | p_i \models \gamma(e)\}$ *for every* $e \in E^F$*. Furthermore* $\theta(l_0, l_0) = \theta(l_1, l_1) = \mathfrak{C}$.
Next, for every vertex $v \in V_\alpha$ *with* $\alpha = \{0, 1\}$*, we have* $C = \mathfrak{C} \backslash \bigcup\{\theta(v, w) | (v, w) \in E\}$*. If* $C \neq \emptyset$ *then we add* (v, l_α) *to* E *and make* $\theta(v, l_\alpha) = C$*. Finally we have*

$$\rho(v) = \begin{cases} m_o & \text{if } v = l_0 \\ m_e & \text{if } v = l_1 \\ \rho^F(v) & \text{otherwise} \end{cases}$$

where m_o *is the highest odd priority given by* ρ^F *or* 1 *if no odd priorities occur. And* m_e *is the highest even priority given by* ρ^F *or* 0 *if no even priorities occur.*

3.5 Correctness

Lemma 3.1. *Given:*

- *FTS* $M = (S, Act, trans, s_0, N, P, \gamma)$,
- *a closed modal mu-calculus formula* φ,
- *a product* $p \in P$,
- *a player* $\alpha \in \{0, 1\}$.

For the winning sets W_α *in LTS2PG(*$M_{|p}, \varphi$*) and* W'_α *in FTS2FPG(*M, φ*)*$_{|p}$ *it holds that* $W_\alpha \subseteq W'_\alpha$.

Proof. Let G^F =FTS2FPG(M, φ)= $(V^F, V_0^F, V_1^F, E^F, \rho^F, N, P, \gamma')$, using definition 3.7, and $G^F_{|p} = (V^F, V_0^F, V_1^F, E^{F'})$, using definition 3.3. Furthermore we have $M_{|p} = (S, Act, trans', s_0)$ and we let $G = \text{LTS2PG}(M_{|p}, \varphi) = (V, V_0, V_1, E, \rho)$. We depict the different transition systems and games in the following diagram.

$$\begin{tikzcd}
\text{FTS } M \arrow[rr, "\varphi"] \arrow[d, Rightarrow, "\Pi"'] & & \text{FPG } G^F \arrow[d, Rightarrow, "\Pi"] \\
\text{LTS } M_{|p} \arrow[r, "\varphi"] & \text{PG } G \;\subseteq & \text{PG } G^F_{|p}
\end{tikzcd}$$

Finally let W_α be the winning set of G and W'_α the winning set of $G^F_{|p}$. We will show that $W_\alpha \subseteq W'_\alpha$ by showing that the game G is a subset of the game $G^F_{|p}$ and that every vertex in $v \in V$ has no outgoing vertex in game $G^F_{|p}$ that does not exist in game G. Therefore the game behaves the exact same for every vertex that is in both games. Specifically we have to show that:

- $V_0 \subseteq V_0^F$,
- $V_1 \subseteq V_1^F$,
- $E \subseteq E^{F'}$,
- $\rho \subseteq \rho^F$ and
- for all $v \in V_\alpha$ we have $\forall (v, w) \in E^{F'} \mid (v, w) \in E$.

Game G is created by

$$(V, V_0, V_1, E, \rho) = LTS2PG((S, Act, trans', s_0), \varphi)$$

and game $G^F_{|p}$ is created by

$$(V^F, V_0^F, V_1^F, E^F, \rho^F) = LTS2PG((S, Act, trans, s_0), \varphi)$$

using definitions 3.6 and 3.7. Using definition 1.3 we find that $trans' \subseteq trans$. Using definition 3.6 it is trivial that $LTS2PG$ is monotonic in the $trans$ parameter. So given that $trans' \subseteq trans$ we get $V_0 \subseteq V_0^F$, $V_1 \subseteq V_1^F$, $E \subseteq E^F$ and $\rho \subseteq \rho^F$.

To show that $E \subseteq E^{F'}$ we let $((s, \psi), (s', \psi')) \in E$. Since $E \subseteq E^F$ we have $((s, \psi), (s', \psi')) \in E^F$. We distinguish two cases:

- If $\psi = \langle a \rangle \psi_1$ or $\psi = [a]\psi_1$ then there exists an $a \in Act$ such that $(s, a, s') \in trans'$. Using definition 1.3 we get $(s, a, s') \in trans$ and $p \models \gamma(s, a, s')$. Using definition 3.7 we find that $\gamma'((s, \psi), (s', \psi')) = \gamma(s, a, s')$ and therefore $p \models \gamma'((s, \psi), (s', \psi'))$. Now using definition 3.3 we find $((s, \psi), (s', \psi')) \in E^{F'}$.
- Otherwise the existence of the edge does not depend on the $trans$ parameter and therefore $((s, \psi), (s', \psi')) \in E^{F'}$ if $(s, \psi) \in V^F$, since $V^F \subseteq V$ we have $(s, \psi) \in V^F$.

Let $(s, \psi) \in V$ and $((s, \psi), (s', \psi')) \in E^{F'}$, we will show $((s, \psi), (s', \psi')) \in E$. We distinguish two cases:

- If $\psi = \langle a \rangle \psi_1$ or $\psi = [a]\psi_1$ then there exists an $a \in Act$ such that $(s, a, s') \in trans$. Using definition 3.3 we get $p \models \gamma'(s, a, s')$. Using definition 3.7 we get $p \models \gamma(s, a, s')$. Using the projection definition 3.7 we get $(s, a, s') \in trans'$ and therefore $((s, \psi), (s', \psi')) \in E$.
- Otherwise the existence of the edge does not depend on the $trans$ parameter and therefore $((s, \psi), (s', \psi')) \in E$ if $(s, \psi) \in V$, since $(s, \psi) \in V$ was assumed we have $((s, \psi), (s', \psi')) \in E$.

$\square$

Theorem 3.2. *Given:*

- *FTS* $M = (S, Act, trans, s_0, N, P, \gamma)$,
- *closed modal mu-calculus formula* φ,
- *product* $p \in P$ *and*
- *state* $s \in S$

it holds that $M_{|p}, s \models \varphi$ *if and only if* $(s, \varphi) \in W_0^p$ *in FTS2FPG*(M, φ)*.*

Proof. The winning set W^p_α is equal to winning set W_α in FTS2FPG$(M,\varphi)_{|p}$ using definition 3.2. Using lemma 3.1 we find that for winning set W'_α for game LTS2PG$(M_{|p},\varphi)$ it holds that $W'_\alpha \subseteq W_\alpha = W^p_\alpha$. Using [3] we know that $M_{|p}, s \models \varphi$ if and only if $(s,\varphi) \in W'_0$. Given that (s,φ) exists in either W'_0 or W'_1 we can conclude that $(s,\varphi) \in W^p_0$ if and only if $(s,\varphi) \in W'_0$ and therefore $(s,\varphi) \in W^p_0$ if and only if $M_{|p}, s \models \varphi$. □

Theorem 3.3. *Given:*

- *FPG* $G^F = (V^F, V^F_0, V^F_1, E^F, \rho^F, N, \{p_0, p_1, \dots, p_m\}, \gamma)$,
- *product* p_i,
- *player* $\alpha \in \{0,1\}$

we have for winning sets $W^{p_i}_\alpha$ *in* G *and* $W^{c_i}_\alpha$ *in FPG2VPG(*G^F*) that* $W^{p_i}_\alpha \subseteq W^{c_i}_\alpha$.

Proof. Let $G = (V, V_0, V_1, E, \rho, \mathfrak{C}, \theta) = \text{FPG2VPG}(G^F)$. Consider finite play π that is valid in game G^F for product p_i. We have for every (π_i, π_{i+1}) in π that $(\pi_i, \pi_{i+1}) \in E^F$ and $p_i \models \gamma((\pi_i, \pi_{i+1}))$. From definition 3.8 it follows that $(\pi_i, \pi_{i+1}) \in E$ and $c_i \in \theta(\pi_i, \pi_{i+1})$. So we can conclude that path π is also valid in game G for configuration c_i. Since the play is finite the winner is determined by the last vertex v in π, player α wins such that $v \in V_{\overline{\alpha}}$. Furthermore we know, because the play is finite, that there exists no $(v,w) \in E^F$ with $p \models \gamma(v,w)$. From this we can conclude that $(v, l_{\overline{\alpha}}) \in E$ and $c_i \in \theta(v, l_{\overline{\alpha}})$. Vertex $l_{\overline{\alpha}}$ has one outgoing edge, namely to itself. So finite play π will in game G^F results in an infinite play $\pi(l_{\overline{\alpha}})^\omega$. Vertex $l_{\overline{\alpha}}$ has a priority with the same parity as player α, so player α wins the infinite play in G for configuration c_i.

Consider infinite play π that is valid in game G^F for product p_i. As shown above this play is also valid in game G for configuration c_i. Since the win conditions of both games are the same the play will result in the same winner.

Consider infinite play π that is valid in game G for configuration c_i. We distinguish two cases:

- If l_α doesn't occur in π then the path is also valid for game G^F with product p_i and has the same winner.
- If $\pi = \pi'(l_\alpha)^\omega$ then the winner is player $\overline{\alpha}$. The path π' is valid for game G^F with product p_i. Let vertex v be the last vertex of π'. Since $(v, l_\alpha) \in E$ and $c_i \in \theta(v, l_\alpha)$ we know that there is no $(v,w) \in E^F$ with $p_i \models \gamma(v,w)$ and that vertex v is owned by player α. So in game G^F player α can't move at vertex v and therefore loses the game (in which case the winner is also $\overline{\alpha}$.

We have shown that every path (finite or infinite) in game G^F with product p_i can be played in game G with configuration c_i and that they have the same winner. Furthermore every infinite path in game G with configuration c_i can be either played as an infinite path or the first part of the path can be played in G^F with product p_i and they have the same winner. From this we can conclude that the theorem holds. □

References

[1] A. Classen, M. Cordy, P.-Y. Schobbens, P. Heymans, A. Legay, and J.-F. Raskin, "Featured transition systems: Foundations for verifying variability-intensive systems and their application to ltl model checking," *IEEE Transactions on Software Engineering*, vol. 39, pp. 1069–1089, 2013.

[2] A. Classen, P. Heymans, P. Y. Schobbens, A. Legay, and J.-P. Raskin, "Model checking lots of systems: Efficient verification of temporal properties in software product lines," vol. 1, 01 2010.

[3] J. Bradfield and I. Walukiewicz, *The mu-calculus and Model Checking*, pp. 871–919. Cham: Springer International Publishing, 2018.