



UNIVERSIDAD
COMPLUTENSE
MADRID

Facultad de Informática

Modelando la variabilidad: Métodos formales para la representación de líneas de productos software

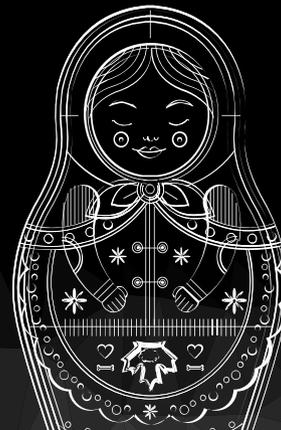
Autor:

Carlos D. Camacho González

Directores:

Dr. Luis Llana

Dr. Alberto Núñez



20 de Junio de 2017

ÍNDICE

- **Introducción**

- Modelos de variabilidad
- Objetivos
- Estructura del proyecto

- **spla** - Software product lines algebra (capítulo 3)
- **splaC** - Cost-related extension for SPLs (capítulo 4)
- **splaP** - Probabilistic product lines (capítulo 5)
- Conclusiones y trabajo futuro

ÍNDICE

- **Introducción**
 - Modelos de variabilidad
 - Objetivos
 - Estructura del proyecto
- **spla** - Software product lines algebra (capítulo 3)
- **splaC** - Cost-related extension for SPLs (capítulo 4)
- **splaP** - Probabilistic product lines (capítulo 5)
- Conclusiones y trabajo futuro

ÍNDICE

- **Introducción**

- Modelos de variabilidad
- **Objetivos**
- Estructura del proyecto

- **spla** - Software product lines algebra (capítulo 3)
- **splaC** - Cost-related extension for SPLs (capítulo 4)
- **splaP** - Probabilistic product lines (capítulo 5)
- Conclusiones y trabajo futuro

ÍNDICE

- **Introducción**

- Modelos de variabilidad
- Objetivos
- Estructura del proyecto

- **spla** - Software product lines algebra (capítulo 3)
- **splaC** - Cost-related extension for SPLs (capítulo 4)
- **splaP** - Probabilistic product lines (capítulo 5)
- Conclusiones y trabajo futuro

ÍNDICE

- **Introducción**
 - Modelos de variabilidad
 - Objetivos
 - Estructura del proyecto
- **spla - Software product lines algebra (capítulo 3)**
- splaC - Cost-related extension for SPLs (capítulo 4)
- splaP - Probabilistic product lines (capítulo 5)
- Conclusiones y trabajo futuro

ÍNDICE

- Introducción
 - Modelos de variabilidad
 - Objetivos
 - Estructura del proyecto
- **spla** - Software product lines algebra (capítulo 3)
- **splaC** - Cost-related extension for SPLs (capítulo 4)
- **splaP** - Probabilistic product lines (capítulo 5)
- Conclusiones y trabajo futuro

ÍNDICE

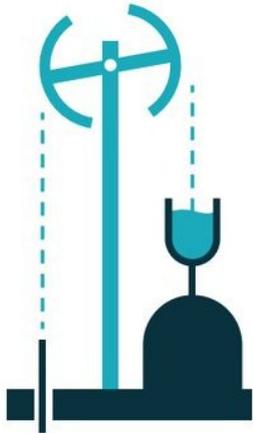
- **Introducción**
 - Modelos de variabilidad
 - Objetivos
 - Estructura del proyecto
- **spla** - Software product lines algebra (capítulo 3)
- **splaC** - Cost-related extension for SPLs (capítulo 4)
- **splaP** - Probabilistic product lines (capítulo 5)
- Conclusiones y trabajo futuro

ÍNDICE

- **Introducción**
 - Modelos de variabilidad
 - Objetivos
 - Estructura del proyecto
- **spla** - Software product lines algebra (capítulo 3)
- **splaC** - Cost-related extension for SPLs (capítulo 4)
- **splaP** - Probabilistic product lines (capítulo 5)
- **Conclusiones y trabajo futuro**

INTRODUCCIÓN

Etapas de la revolución industrial.



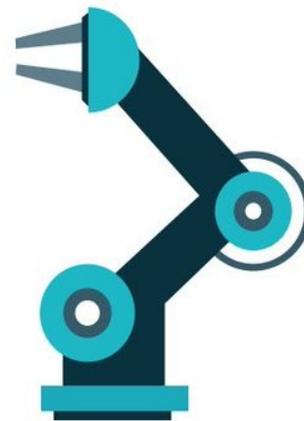
1784

Energía a vapor.



1870

Producción en serie.



1969

Control electrónico.



Actualidad

Sistemas informáticos.

INTRODUCCIÓN

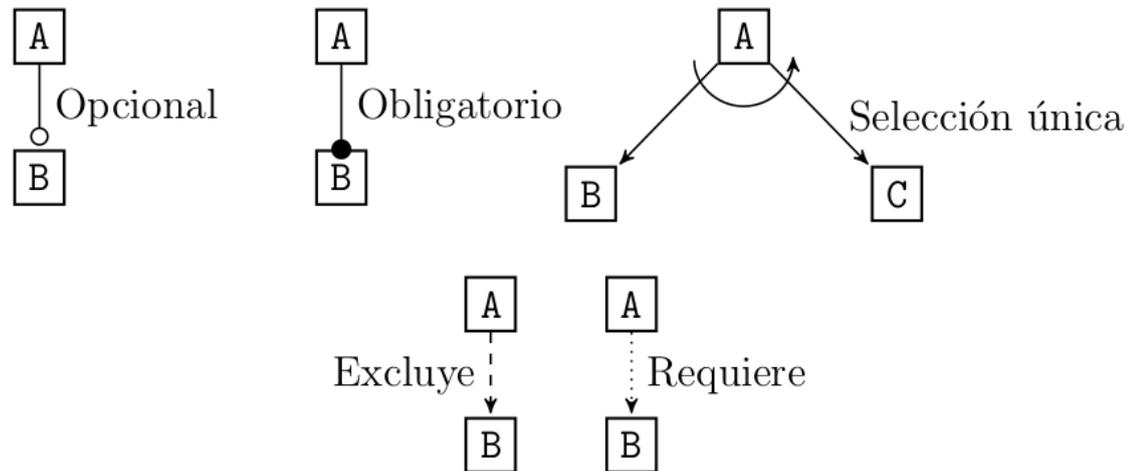
Las líneas de productos (PLs, del inglés Product Lines) permiten construir plataformas comunes para la creación de productos, reutilizando componentes previamente desarrollados.

INTRODUCCIÓN

En el campo de las PLs existen dos conceptos clave que permiten modelar productos en función de sus componentes comunes y variables (commonality y variability).

FODA

(Feature Oriented Domain Analysis)



Relaciones de FODA

OBJETIVOS

- El objetivo principal de esta tesis doctoral consiste en **definir nuevos métodos formales para representar SPLs**.
 - Definir una representación formal para FODA que permita realizar análisis automatizado sobre su estructura.
 - Definir el coste dentro del proceso de construcción de los productos válidos.
 - Representar probabilidades dentro de los términos.
 - Mostrar distintas implementaciones del formalismo descrito, en función de la información que sea necesario generar.

OBJETIVOS

- El objetivo principal de esta tesis doctoral consiste en **definir nuevos métodos formales para representar SPLs**.
 - ▶ Definir una representación formal para FODA que permita realizar análisis automatizado sobre su estructura.
 - ▶ Definir el coste dentro del proceso de construcción de los productos válidos.
 - ▶ Representar probabilidades dentro de los términos.
 - ▶ Mostrar distintas implementaciones del formalismo descrito, en función de la información que sea necesario generar.

OBJETIVOS

- El objetivo principal de esta tesis doctoral consiste en **definir nuevos métodos formales para representar SPLs**.
 - Definir una representación formal para FODA que permita realizar análisis automatizado sobre su estructura.
 - **Definir el coste dentro del proceso de construcción de los productos válidos.**
 - Representar probabilidades dentro de los términos.
 - Mostrar distintas implementaciones del formalismo descrito, en función de la información que sea necesario generar.

OBJETIVOS

- El objetivo principal de esta tesis doctoral consiste en **definir nuevos métodos formales para representar SPLs**.
 - Definir una representación formal para FODA que permita realizar análisis automatizado sobre su estructura.
 - Definir el coste dentro del proceso de construcción de los productos válidos.
 - **Representar probabilidades dentro de los términos.**
 - Mostrar distintas implementaciones del formalismo descrito, en función de la información que sea necesario generar.

OBJETIVOS

- El objetivo principal de esta tesis doctoral consiste en **definir nuevos métodos formales para representar SPLs**.
 - Definir una representación formal para FODA que permita realizar análisis automatizado sobre su estructura.
 - Definir el coste dentro del proceso de construcción de los productos válidos.
 - Representar probabilidades dentro de los términos.
 - **Mostrar distintas implementaciones del formalismo descrito, en función de la información que sea necesario generar.**

ESTRUCTURA DE LA TESIS

Cost-related extension for SPLs

Componente teórico

- Sintaxis
- Función de coste
- Extensión de las reglas de transición

Componente práctico

- Caso de estudio
- Implementación

Probabilistic product lines

Componente teórico

- Sintaxis
- Extensión de la semántica operacional y denotacional
- Equivalencia entre las semánticas
- Ocultando conjuntos de características

Componente práctico

- Implementación

Software product lines algebra

Componente teórico

- Sintaxis
- Método de traducción de FODA a splA
- Semántica operacional, denotacional y axiomática
- Consistencia en la traducción
- Factibilidad de los términos

Componente práctico

- Módulo de factibilidad
- Módulo denotacional
- Caso de estudio

ESTRUCTURA DE LA TESIS

Software product lines algebra (SPLA)

Componente teórico

- Sintaxis
- Método de traducción de FODA a SPLA
- Semántica operacional, denotacional y axiomática
- Consistencia en la traducción
- Factibilidad de los términos

Componente práctico

- Módulo de factibilidad
- Módulo denotacional
- Caso de estudio

ESTRUCTURA DE LA TESIS

Cost-related extension for SPLs (SPLAC)

Componente teórico

- Sintaxis
- Función de coste
- Extensión de las reglas de transición

Componente práctico

- Caso de estudio
- Implementación

ESTRUCTURA DE LA TESIS

Probabilistic product lines (SPLAP)

Componente teórico

- Sintaxis
- Extensión de la semántica operacional y denotacional
- Equivalencia entre las semánticas
- Ocultando conjuntos de características

Componente práctico

- Implementación

SPLA

Esta sección presenta la definición e implementación del lenguaje formal SPLA.

SPLA

- Sintaxis del lenguaje

Está compuesta por los principios y reglas necesarias para construir términos propios del álgebra.

$$P ::= \checkmark \mid \text{nil} \mid A; P \mid \bar{A}; P \mid \\ P \vee Q \mid P \wedge Q \mid A \not\Rightarrow B \text{ in } P \mid \\ A \Rightarrow B \text{ in } P \mid P \setminus A \mid P \Rightarrow A$$

Elementos sintácticos de SPLA

SPLA

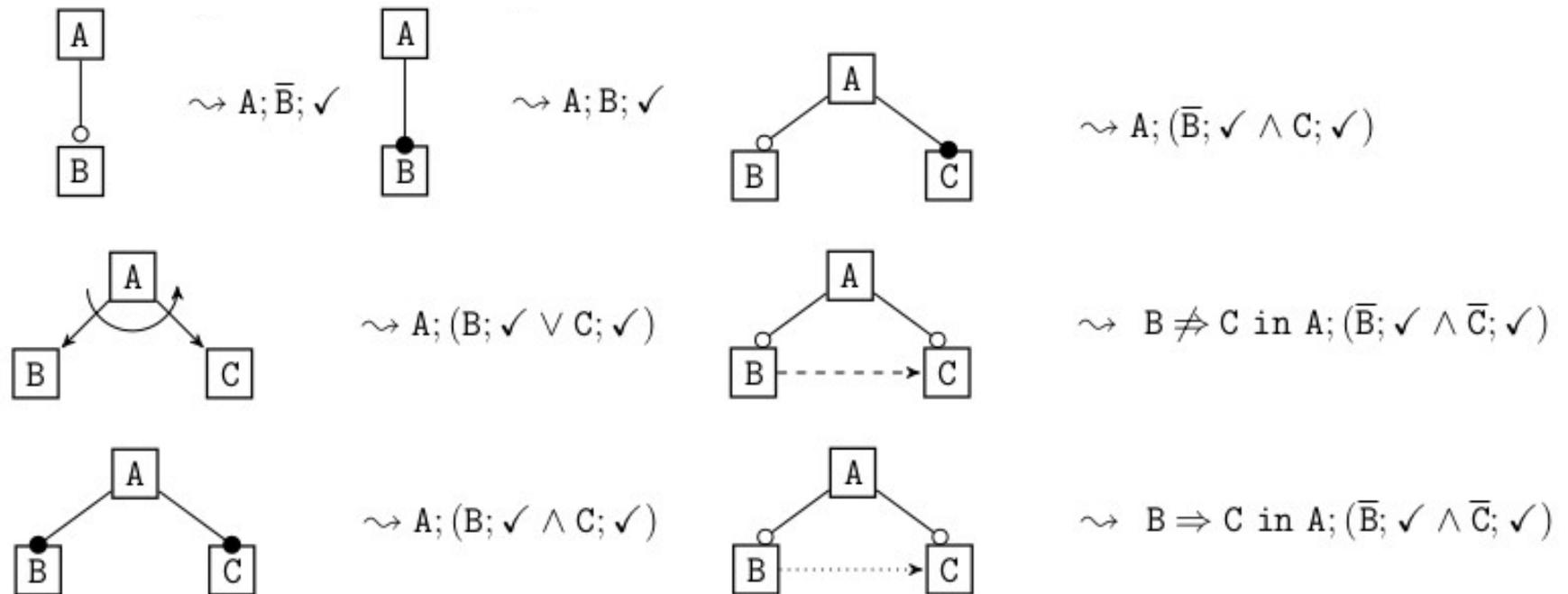
- Traducción de FODA a SPLA

	Diagrama FODA	Término SPLA
Característica		$A; \checkmark$
Paralelo		$A; (B1; \triangle_{P_1} \wedge \dots \wedge BN; \triangle_{P_n} \wedge \overline{C1}; \triangle_{Q_1} \wedge \dots \wedge \overline{CN}; \triangle_{Q_n})$
Selección única		$A; (\triangle_P \vee \triangle_Q)$
Requerimiento		$A \Rightarrow B \text{ in } \triangle_P$
Exclusión		$A \not\Rightarrow B \text{ in } \triangle_P$

Proceso de traducción FODA \rightarrow SPLA

SPLA

Ejemplo de conversión de diagramas FODA a SPLA



Diagramas FODA → Términos SPLA

SPLA

- Reglas de la semántica operacional

[tick]	$\checkmark \xrightarrow{\checkmark} \text{nil}$	[feat]	$A; P \xrightarrow{A} P$
[ofeat1]	$\bar{A}; P \xrightarrow{A} P$	[ofeat2]	$\bar{A}; P \xrightarrow{\checkmark} \text{nil}$
[cho1]	$\frac{P \xrightarrow{A} P_1}{P \vee Q \xrightarrow{A} P_1}$	[cho2]	$\frac{Q \xrightarrow{A} Q_1}{P \vee Q \xrightarrow{A} Q_1}$
[con1]	$\frac{P \xrightarrow{A} P_1}{P \wedge Q \xrightarrow{A} P_1 \wedge Q}$	[con2]	$\frac{Q \xrightarrow{A} Q_1}{P \wedge Q \xrightarrow{A} P \wedge Q_1}$
[con3]	$\frac{P \xrightarrow{\checkmark} \text{nil}, Q \xrightarrow{\checkmark} \text{nil}}{P \wedge Q \xrightarrow{\checkmark} \text{nil}}$	[req1]	$\frac{P \xrightarrow{C} P_1, C \neq A}{A \Rightarrow B \text{ in } P \xrightarrow{C} A \Rightarrow B \text{ in } P_1}$
[req2]	$\frac{P \xrightarrow{A} P_1}{A \Rightarrow B \text{ in } P \xrightarrow{A} P_1 \Rightarrow B}$	[req3]	$\frac{P \xrightarrow{\checkmark} \text{nil}}{A \Rightarrow B \text{ in } P \xrightarrow{\checkmark} \text{nil}}$
[excl1]	$\frac{P \xrightarrow{C} P_1, C \neq A \wedge C \neq B}{A \not\Rightarrow B \text{ in } P \xrightarrow{C} A \not\Rightarrow B \text{ in } P_1}$	[excl2]	$\frac{P \xrightarrow{A} P_1}{A \not\Rightarrow B \text{ in } P \xrightarrow{A} P_1 \setminus B}$
[excl3]	$\frac{P \xrightarrow{B} P_1}{A \not\Rightarrow B \text{ in } P \xrightarrow{B} P_1 \setminus A}$	[excl4]	$\frac{P \xrightarrow{\checkmark} \text{nil}}{A \not\Rightarrow B \text{ in } P \xrightarrow{\checkmark} \text{nil}}$
[forb1]	$\frac{P \xrightarrow{B} P_1, B \neq A}{P \setminus A \xrightarrow{B} P_1 \setminus A}$	[forb2]	$\frac{P \xrightarrow{\checkmark} \text{nil}}{P \setminus A \xrightarrow{\checkmark} \text{nil}}$
[mand1]	$\frac{P \xrightarrow{\checkmark} \text{nil}}{P \Rightarrow A \xrightarrow{A} \checkmark}$	[mand2]	$\frac{P \xrightarrow{A} P_1}{P \Rightarrow A \xrightarrow{A} P_1}$
[mand3]	$\frac{P \xrightarrow{B} P_1, A \neq B}{P \Rightarrow A \xrightarrow{B} P_1 \Rightarrow A}$		

SPLA

- Reglas de la semántica operacional

$$[\text{ofeat1}] \bar{A}; P \xrightarrow{A} P$$

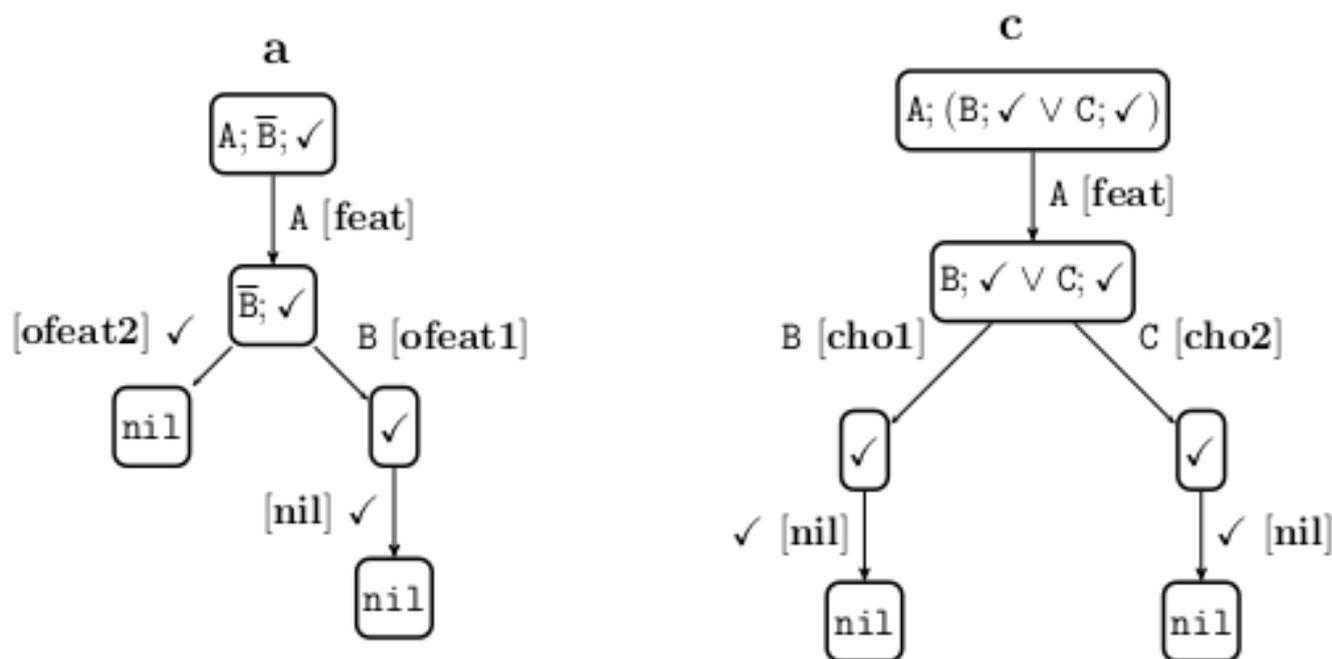
$$[\text{ofeat2}] \bar{A}; P \xrightarrow{\checkmark} \text{nil}$$

$$[\text{cho1}] \frac{P \xrightarrow{A} P_1}{P \vee Q \xrightarrow{A} P_1}$$

$$[\text{cho2}] \frac{Q \xrightarrow{A} Q_1}{P \vee Q \xrightarrow{A} Q_1}$$

SPLA

Procesamiento de las reglas de la semántica operacional



$$\text{a } \text{prod}(A; \bar{B}; \checkmark) = \{ [A], [AB] \}$$

$$\text{c } \text{prod}(A; (B; \checkmark \vee C; \checkmark)) = \{ [AB], [AC] \}$$

Ejemplos del procesamiento de términos SPLA

SPLA

• Reglas de la semántica denotacional

- $\llbracket \text{nil} \rrbracket = \emptyset$
- $\llbracket \checkmark \rrbracket = \{\emptyset\}$
- $\llbracket A; \cdot \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket A; \cdot \rrbracket(P) = \{\{A\} \cup p \mid p \in P\}$
- $\llbracket \bar{A}; \cdot \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket \bar{A}; \cdot \rrbracket(P) = \{\emptyset\} \cup \{\{A\} \cup p \mid p \in P\}$
- $\llbracket \cdot \vee \cdot \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \times \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket \cdot \vee \cdot \rrbracket(P, Q) = P \cup Q$
- $\llbracket \cdot \wedge \cdot \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \times \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket \cdot \wedge \cdot \rrbracket(P, Q) = \{p \cup q \mid p \in P, q \in Q\}$
- $\llbracket A \Rightarrow B \text{ in } \cdot \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket A \Rightarrow B \text{ in } \cdot \rrbracket(P) = \{p \mid p \in P, A \notin p\} \cup \{p \cup \{B\} \mid p \in P, A \in p\}$
- $\llbracket A \not\Rightarrow B \text{ in } \cdot \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket A \not\Rightarrow B \text{ in } \cdot \rrbracket(P) = \{p \mid p \in P, A \notin p\} \cup \{p \mid p \in P, B \notin p\}$
- $\llbracket \cdot \Rightarrow A \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket \cdot \Rightarrow A \rrbracket(P) = \{p \cup \{A\} \mid p \in P\}$
- $\llbracket \cdot \setminus A \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket \cdot \setminus A \rrbracket(P) = \{p \mid p \in P, A \notin p\}$

SPLA

- Reglas de la semántica denotacional

\mathcal{F} = conjunto de características

$\mathcal{P}(\mathcal{F})$ = superconjunto de \mathcal{F}

$\mathcal{P}(\mathcal{P}(\mathcal{F}))$ = superconjunto de $\mathcal{P}(\mathcal{F})$

- $\llbracket \bar{A}; \cdot \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket \bar{A}; \cdot \rrbracket(P) = \{\emptyset\} \cup \{\{A\} \cup p \mid p \in P\}$
- $\llbracket \cdot \wedge \cdot \rrbracket : \mathcal{P}(\mathcal{P}(\mathcal{F})) \times \mathcal{P}(\mathcal{P}(\mathcal{F})) \mapsto \mathcal{P}(\mathcal{P}(\mathcal{F}))$ como
 $\llbracket \cdot \wedge \cdot \rrbracket(P, Q) = \{p \cup q \mid p \in P, q \in Q\}$

SPLA

Procesamiento de las reglas de la semántica denotacional

$$\begin{array}{l} [\checkmark] \\ [\bar{B}; \checkmark] \\ [A; \bar{B}; \checkmark] \end{array} \quad \begin{array}{l} \text{a} \\ = \{\emptyset\} \\ = [\bar{B}; \cdot](\{\emptyset\}) = \{\emptyset, \{B\}\} \\ = [A; \cdot](\{\emptyset, \{B\}\}) = \{\{A\}, \{A, B\}\} \end{array}$$

$$\begin{array}{l} [B; \checkmark \wedge C; \checkmark] \\ [\cdot \wedge \cdot](\{\{B\}\}, \{\{C\}\}) \\ [A; (B; \checkmark \wedge C; \checkmark)] \end{array} \quad \begin{array}{l} \text{d} \\ = [\cdot \wedge \cdot](\{\{B\}\}, \{\{C\}\}) = \{\{B, C\}\} \\ = [A; \cdot](\{\{B, C\}\}) = \{\{A, B, C\}\} \end{array}$$

Ejemplos del procesamiento de términos SPLA

SPLA

• Reglas de la semántica axiomática (complejos)

$$[\text{REQ1}] \quad A \Rightarrow B \text{ in } (C; P) =_E C; (A \Rightarrow B \text{ in } P)$$

$$[\text{REQ2}] \quad A \Rightarrow B \text{ in } (A; P) =_E A; (P \Rightarrow B)$$

$$[\text{REQ3}] \quad A \Rightarrow B \text{ in } (B; P) =_E B; P$$

$$[\text{REQ4}] \quad A \Rightarrow B \text{ in } P \vee Q =_E (A \Rightarrow B \text{ in } P) \vee (A \Rightarrow B \text{ in } Q)$$

$$[\text{REQ5}] \quad A \Rightarrow B \text{ in } \checkmark =_E \checkmark$$

$$[\text{REQ6}] \quad A \Rightarrow B \text{ in } \text{nil} =_E \text{nil}$$

$$[\text{MAND1}] \quad (A; P) \Rightarrow A =_E A; P$$

$$[\text{MAND2}] \quad (B; P) \Rightarrow A =_E B; (P \Rightarrow A)$$

$$[\text{MAND3}] \quad \checkmark \Rightarrow A =_E A; \checkmark$$

$$[\text{MAND4}] \quad \text{nil} \Rightarrow A =_E \text{nil}$$

$$[\text{MAND5}] \quad (P \vee Q) \Rightarrow A =_E (P \Rightarrow A) \vee (Q \Rightarrow A)$$

$$[\text{EXCL1}] \quad A \not\Rightarrow B \text{ in } (C; P) =_E C; (A \not\Rightarrow B \text{ in } P)$$

$$[\text{EXCL2}] \quad A \not\Rightarrow B \text{ in } (A; P) =_E A; (P \setminus B)$$

$$[\text{EXCL3}] \quad A \not\Rightarrow B \text{ in } (B; P) =_E B; (P \setminus A)$$

$$[\text{EXCL4}] \quad A \not\Rightarrow B \text{ in } P \vee Q =_E (A \not\Rightarrow B \text{ in } P) \vee (A \not\Rightarrow B \text{ in } Q)$$

$$[\text{EXCL5}] \quad A \not\Rightarrow B \text{ in } \checkmark =_E \checkmark$$

$$[\text{EXCL6}] \quad A \not\Rightarrow B \text{ in } \text{nil} =_E \text{nil}$$

$$[\text{FORB1}] \quad (A; P) \setminus A =_E \text{nil}$$

$$[\text{FORB2}] \quad (B; P) \setminus A =_E B; (P \setminus A)$$

$$[\text{FORB3}] \quad \checkmark \setminus A =_E \checkmark$$

$$[\text{FORB4}] \quad \text{nil} \setminus A =_E \text{nil}$$

$$[\text{FORB5}] \quad (P \vee Q) \setminus A =_E (P \setminus A) \vee (Q \setminus A)$$

SPLA

- Reglas de la semántica axiomática (básicos)

$$[\text{PRE1}] \quad A; B; P =_E B; A; P.$$

$$[\text{PRE2}] \quad \bar{A}; P =_E (A; P) \vee \checkmark.$$

$$[\text{PRE3}] \quad (A; P) \vee (A; Q) =_E A; (P \vee Q)$$

$$[\text{PRE4}] \quad A; \text{nil} =_E \text{nil}$$

$$[\text{PRE5}] \quad A; A; P =_E A; P$$

$$[\text{CHO1}] \quad P \vee Q =_E Q \vee P.$$

$$[\text{CHO2}] \quad (P \vee Q) \vee R =_E P \vee (Q \vee R).$$

$$[\text{CHO3}] \quad P \vee \text{nil} =_E P.$$

$$[\text{CHO4}] \quad P \vee P = P.$$

$$[\text{CON1}] \quad (A; P) \wedge Q =_E A; (P \wedge Q)$$

$$[\text{CON2}] \quad P \wedge Q =_E Q \wedge P.$$

$$[\text{CON3}] \quad P \wedge (Q \vee R) =_E (P \wedge Q) \vee (P \wedge R).$$

$$[\text{CON4}] \quad P \wedge \text{nil} =_E \text{nil}$$

$$[\text{CON5}] \quad P \wedge \checkmark =_E P$$

SPLA

- Reducción de los términos del álgebra

Se muestra que el modelo desarrollado es totalmente equivalente. Dado cualquier conjunto s de productos, existe un término SPLA cuya semántica es exacta al conjunto s .

Este conjunto de términos es conocido como $SPLA_b$

$$P ::= \surd \mid \text{nil} \mid A; P \mid P \vee Q$$

SPLA

Ejemplo de reducción de términos

d

$$\begin{aligned} A; (B; \checkmark \wedge C; \checkmark) &=_E \text{ [CON1]} \\ A; B; (\checkmark \wedge C; \checkmark) &=_E \text{ [CON2], [CON5]} \\ A; B; C; \checkmark & \end{aligned}$$

e

$$\begin{aligned} A; (\bar{B}; \checkmark \wedge C; \checkmark) &=_E \text{ [CON2], [CON1]} \\ A; C; (\checkmark \wedge \bar{B}; \checkmark) &=_E \text{ [CON2], [CON5]} \\ A; C; \bar{B}; \checkmark &=_E \text{ [PRE2]} \\ A; C; (B; \checkmark \vee \checkmark) &=_E \text{ [PRE3]} \\ A; (C; B; \checkmark \vee C; \checkmark) &=_E \text{ [PRE1]} \\ A; (B; C; \checkmark \vee C; \checkmark) & \end{aligned}$$

SPLA

Factibilidad de los términos del álgebra

La comprobación de la factibilidad de cualquier término $P \in \text{SPLA}$ puede llevarse a cabo procesando todos sus productos, haciendo uso de las reglas definidas en la semántica operacional o en la semántica denotacional.

SPLA

Factibilidad de los términos del álgebra (cont.)

$$P : A \Rightarrow C \text{ in } A; \checkmark \wedge B; \checkmark$$

$$Q : C \not\Rightarrow B \text{ in } P$$

$$\begin{aligned} \phi(P) &= (\neg A_1 \rightarrow \neg A_0) \wedge (\neg C_0 \rightarrow \neg \perp) \wedge (A_1 \rightarrow C_0) \wedge A_0 \wedge B_0 \equiv \\ &\quad (\neg A_1 \rightarrow \neg A_0) \wedge (A_1 \rightarrow C_0) \wedge A_0 \wedge B_0 \end{aligned}$$

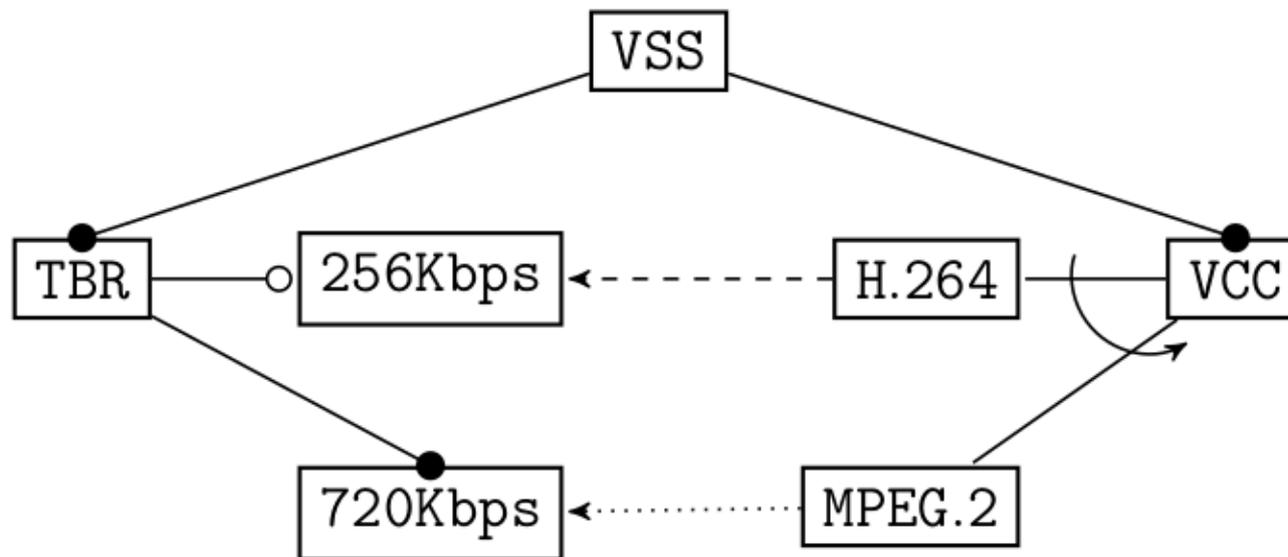
$$\begin{aligned} \phi(Q) &= (\neg C_1 \rightarrow \neg C_0) \wedge (\neg B_1 \rightarrow \neg B_0) \wedge (\neg C_1 \vee B_1) \wedge \phi(P) \equiv \\ (\neg C_1 \rightarrow \neg C_0) \wedge (\neg B_1 \rightarrow \neg B_0) \wedge (\neg C_1 \vee \neg B_1) \wedge (\neg A_1 \rightarrow \neg A_0) \wedge (A_1 \rightarrow C_0) \wedge A_0 \wedge B_0 &\equiv \end{aligned}$$

La fórmula es falsa por:

- 1) Comenzando A_0 y B_0 deben ser ciertas.
- 2) $\neg A_0$ es falso y por lo tanto $\neg A_1$ es falso.
- 3) A_1 es cierto y por lo tanto C_0 es cierto.
- 4) $\neg C_0$ es falso y por lo tanto $\neg C_1$ es falso.
- 5) Por lo tanto $\neg B_1$ es cierto, de lo que se deduce que $\neg B_0$ es cierto.
- 6) Contradicción, ya que B_0 es cierto.

SPLA

Caso de uso Video Streaming Software



$P_1 := \text{MPEG.2} \Rightarrow 720\text{Kbps}$ in $(\text{H.264} \not\Rightarrow 256\text{Kbps}$ in $P_2)$

$P_2 := \text{VSS}; (P_{21} \wedge P_{22})$

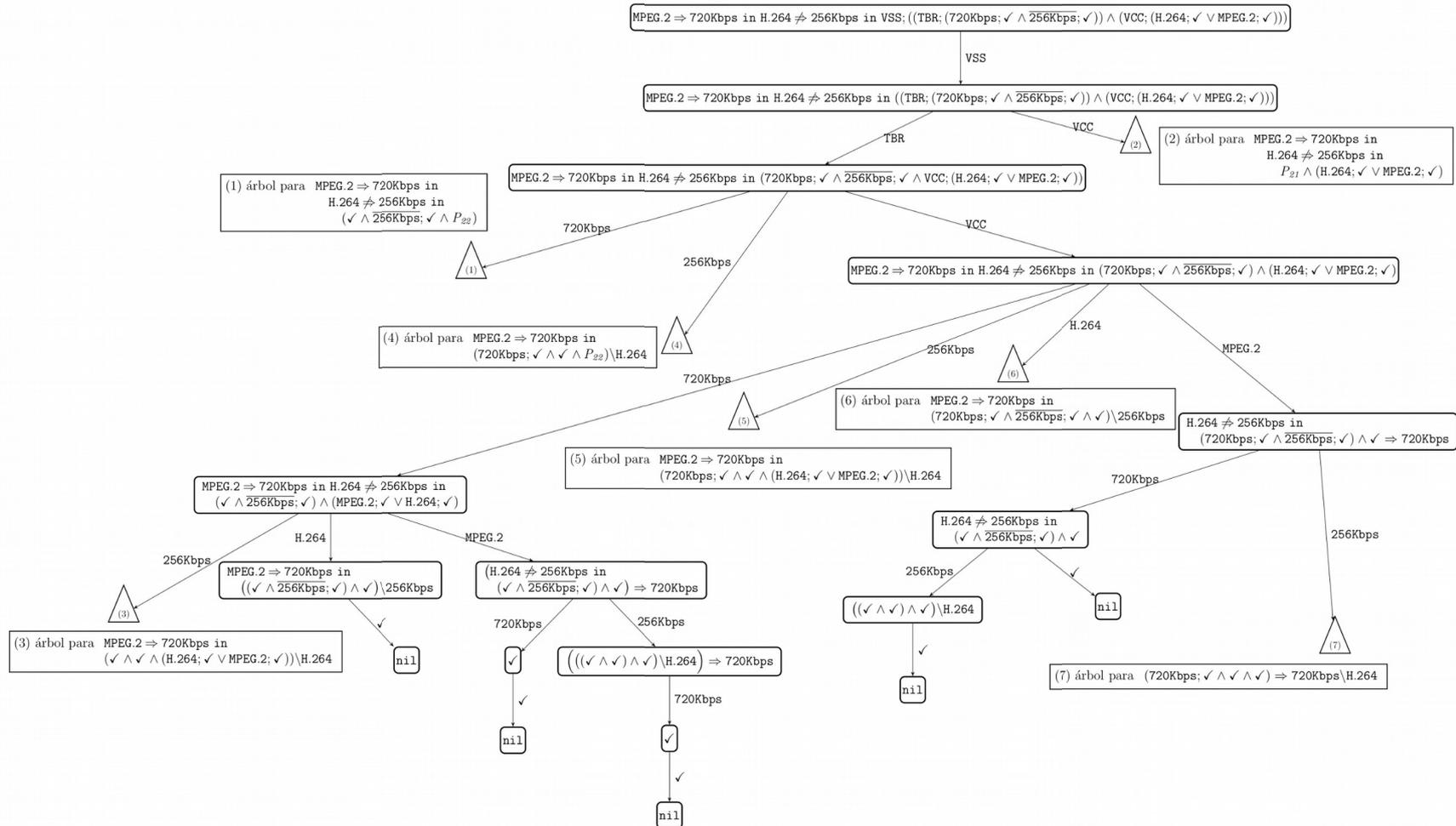
$P_{21} := \text{TBR}; (720\text{Kbps}; \checkmark \wedge \overline{256\text{Kbps}}; \checkmark)$

$P_{22} := \text{VCC}; (\text{H.264}; \checkmark \vee \text{MPEG.2}; \checkmark)$

Diagrama FODA de VSS → Término SPLA

SPLA

Caso de uso Video Streaming Software



Procesamiento del término VSS (sem. operacional)

SPLA

Implementación de SPLA

Modelos en XML de SPLs utilizando el generador de modelos de características BeTTY. Los valores utilizados en los parámetros de configuración se detallan a continuación:

- El porcentaje de restricciones es de un 30 %.
- La probabilidad de que exista una característica obligatoria es de 0.25.
- La probabilidad de que una característica esté dentro de una relación de selección única es de 0.5.

SPLA

Implementación de SPLA (benchmark de factibilidad)

Características	Tiempo (ms.)	Características	Tiempo (ms.)
01000	67	07500	529
01500	71	08000	556
02000	105	08500	682
02500	140	09000	428
03000	164	09500	639
03500	153	10000	620
04000	193	10500	396
04500	410	11000	513
05000	331	11500	575
05500	339	12000	552
06000	441	12500	413
06500	289	13000	513
07000	369	13500	503

Tiempos de procesamiento en función del número de características del término

SPLA

Implementación de SPLA (benchmark denotacional)

Características	Tiempo (ms.)	Productos	Características	Tiempo (ms.)	Productos
50	6	48	180	744	6384
60	25	108	190	1390	7232
70	15	104	200	960000	-
80	8	31	210	97770	800544
90	38	0	220	263	51
100	55	1404	230	47	8
110	13	12	240	65	29
120	2139	1	250	191	5920
130	511	24802	260	205	7296
140	7	6	270	250	4301
150	786126	1312848	280	960000	-
160	136	5670	290	65	3
170	42	398	300	960000	-

Tiempos de procesamiento en función del número de características del término

SPLAC

Esta sección presenta la descripción formal de la extensión de costes para SPLA.

SPLAC

Función de coste

Agregar un componente de software a un producto podría, o no, suponer un coste. Sin embargo, este coste no es constante y dependerá de su relación con otros componentes del producto.

s = Traza con características procesadas.

x = Característica actual a ser procesada.

$$c(s, x) = \begin{cases} \perp & \text{si } x = B \text{ y } A \notin s \\ 2 & \text{si } x = D \text{ y } E \notin s \\ 1 & \text{en caso contrario} \end{cases}$$

SPLAC

Transiciones con coste

[ini] $(P, \epsilon, 0) \in N$.

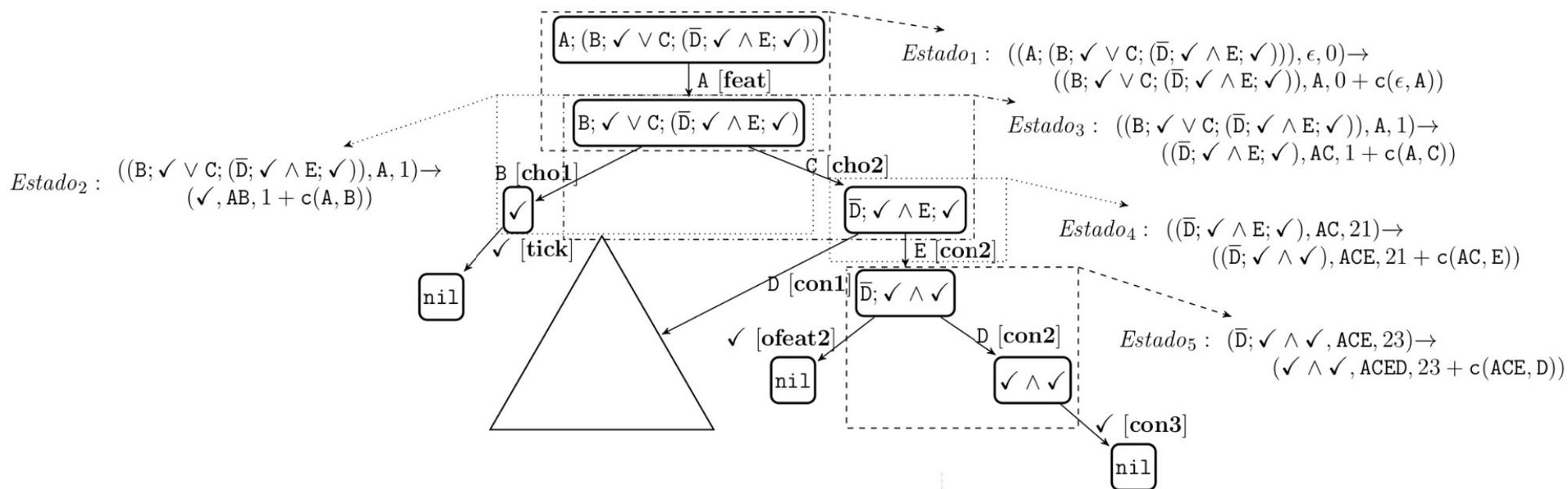
[cost1] Si $(Q, s, n) \in N$, $Q \xrightarrow{A} Q_1$ y $A \notin s$, entonces
 $(Q_1, sA, n + c(s, A)) \in N$ y $(Q, s, n) \rightarrow (Q_1, sA, n + c(s, A))$.

[cost2] Si $(Q, s, n) \in N$, $Q \xrightarrow{A} Q_1$ y $A \in s$, entonces
 $(Q_1, sA, n) \in N$ y $(Q, s, n) \rightarrow (Q_1, sA, n)$.

Reglas para procesar los costes en las transiciones

SPLAC

Ejemplo de ejecución



c	A	B	C	D	E
ϵ	1	1	1	1	1
A	\perp	22	20	125	134
AC	12	15	\perp	13	2
ACE	\perp	11	\perp	332	\perp
ACD	\perp	132	\perp	\perp	581

Procesamiento de un término tomando en cuenta los costes en las transiciones

SPLAC

chef.io – Diagrama FODA

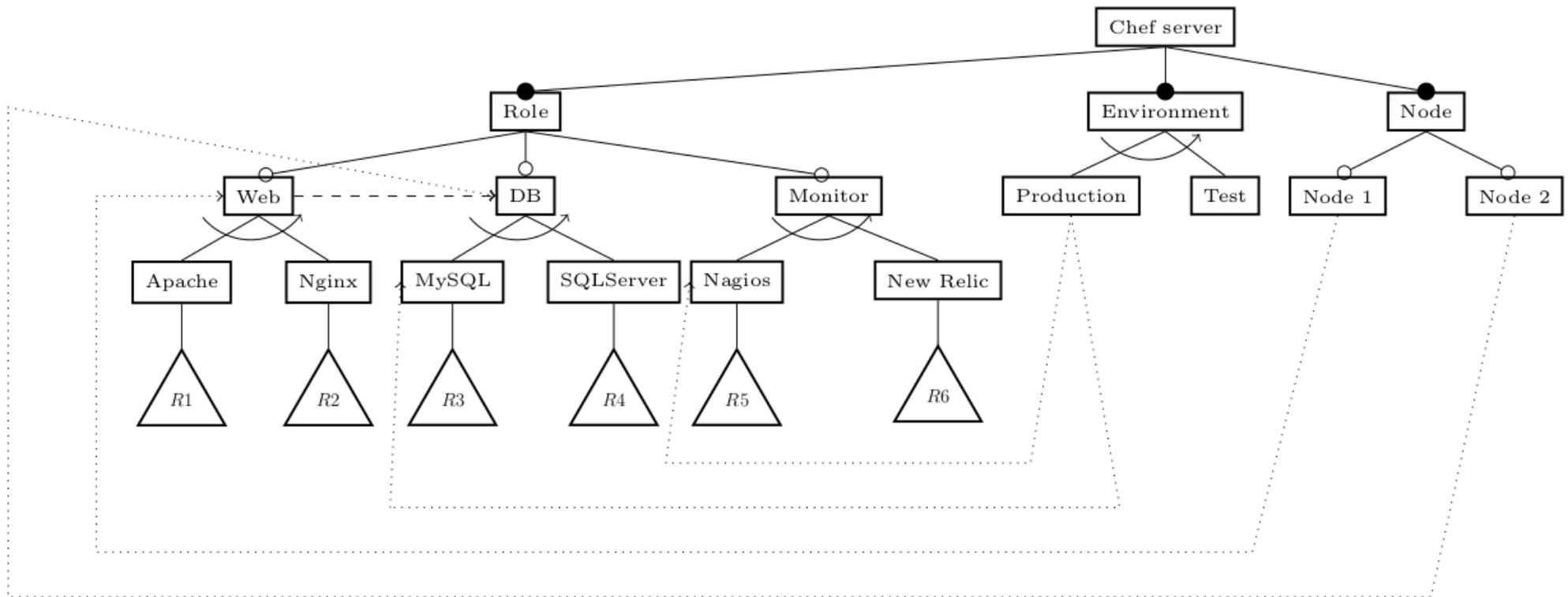


Diagrama FODA para modelar el caso de estudio

SPLAC

chef.io – Término SPLA y función de coste

run-list =

```

N1 => W in (
  N2 => D in (
    P => MS in (
      P => NA in (
        W ≠ D in (
          S; (
            R; (
              W̄; (AP; ✓ ∨ NG; ✓)
              ^
              D̄; (MS; ✓ ∨ SS; ✓)
              ^
              M̄; (NA; ✓ ∨ NR; ✓)
            )
            ^
            E; (P; ✓ ∨ T; ✓)
            ^
            N; (N̄1; ✓ ∨ N̄2; ✓)
          )
        )
      )
    )
  )
)

```

Término SPLA

S : Chef server
 R : Role
 E : Environment
 N : Node
 P : Production
 T : Test
 W : Web server
 D : Database server
 M : Monitoring
 AP : Apache
 NG : Nginx
 MS : MySQL
 SS : SQLServer
 NA : Nagios
 NE : New Relic
 N1 : Node 1
 N2 : Node 2

Leyenda

$$c(s, x) = \begin{cases}
 \text{Regla1} : 7 & \text{si } x = \text{MySQL y Nagios} \in s \\
 \text{Regla2} : 7 & \text{si } x = \text{Apache y Nagios} \in s \\
 \text{Regla3} : 4 & \text{si } x = \text{Nagios y MySQL} \in s \\
 \text{Regla4} : 4 & \text{si } x = \text{Nagios y Apache} \in s \\
 \text{Regla5} : \perp & \text{si } x = \text{Apache y Mysql} \in s \\
 \text{Regla6} : \perp & \text{si } x = \text{Mysql y Apache} \in s \\
 \text{Regla7} : 0 & \text{en cualquier otro caso}
 \end{cases}$$

Función de coste

SPLAC

chef.io – Productos y costes

```
tr(run-list) =  
  {  
    ...  
    ['Node2', 'Prod', 'DB', 'MySQL', 'Monitor', 'NewRelic', 'Nagios']  
    ['Node2', 'Prod', 'Monitor', 'DB', 'Nagios', 'MySQL']  
    ...  
    ['Node2', 'Prod', 'Monitor', 'DB', 'NewRelic', 'MySQL', 'Nagios']  
    ...  
  }
```

Resultado de las trazas del término

p	$C_{SPLA}(P, p)$
...	...
{Node2, Prod, DB, MySQL, Monitor, NewRelic, Nagios}	{4, 7}
{Node2, Prod, Monitor, DB, Nagios, MySQL}	{4, 7}
...	...

Resultado de la función de coste

SPLAC

Implementación

La herramienta está desarrollada en Python y genera una máquina de estados finitos que modela los estados válidos del sistema, es decir, genera los productos válidos. Mientras las características son procesadas, el coste y las trazas son almacenadas en cada término intermedio para su posterior uso.

SPLAC

Consideraciones

Optimización del modelo de características

run-list genera 97648 productos en 2340.49 segundos.

run-list_{op} genera 10572 productos en 491.41 segundos.

Al eliminar los componentes no funcionales, el tiempo de ejecución mejoró un 79 % y el número de productos producidos se redujo un 89,1 %.

```
run-list =
  N1 ⇒ W in (
    N2 ⇒ D in (
      P ⇒ MS in (
        P ⇒ NA in (
          W ≠ D in (
            S; (
              R; (
                W̄; (AP; ✓ ∨ NG; ✓)
                ^
                D̄; (MS; ✓ ∨ SS; ✓)
                ^
                M̄; (NA; ✓ ∨ NR; ✓)
              )
            ^
            E; (P; ✓ ∨ T; ✓)
            ^
            N; (N1; ✓ ∨ N2; ✓)
          )
        )
      )
    )
  )
```

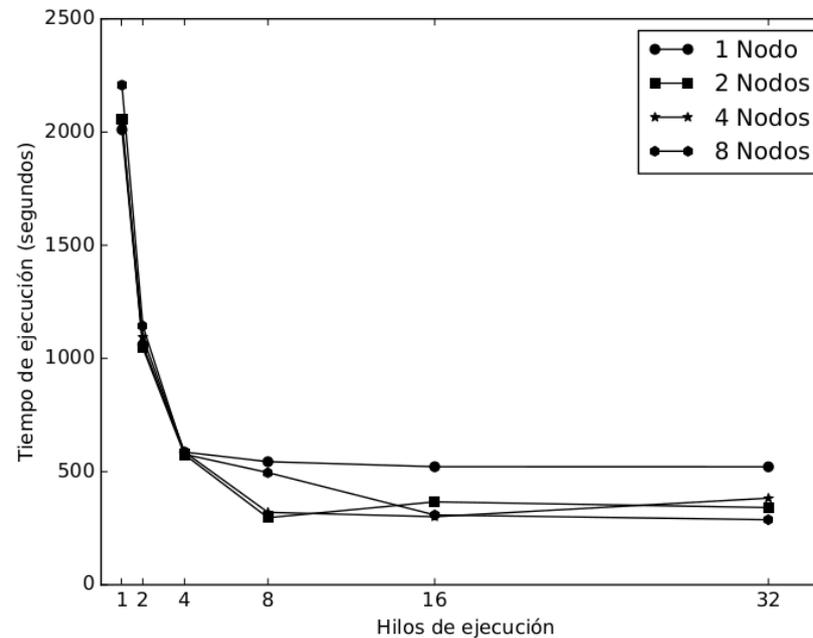
```
run-listop =
  N1 ⇒ W in (
    N2 ⇒ DB in (
      P ⇒ MS in (
        P ⇒ NA in (
          W ≠ D in (
            W̄; (AP; ✓ ∨ NG; ✓)
            ^
            D̄; (MS; ✓ ∨ SS; ✓)
            ^
            M̄; (NA; ✓ ∨ NR; ✓)
            ^
            P; ✓ ∨ T; ✓
            ^
            N1; ✓ ∨ N2; ✓
          )
        )
      )
    )
  )
```

SPLAC

Consideraciones

Cómputo distribuido

	1 Hilo	2 Hilos	4 Hilos	8 Hilos	16 Hilos	32 Hilos
1 Nodo	2010.44763303	1060.80047798	586.014445066	543.712262154	521.616870165	521.292215109
2 Nodos	2059.06947899	1046.87119007	575.115453959	296.285589933	366.384442091	341.007520199
4 Nodos	2031.25184584	1093.52087283	586.344302893	320.010899067	300.745616913	382.748430014
8 Nodos	2207.35427213	1143.951792	576.370896101	495.507214785	308.374300957	287.340030909

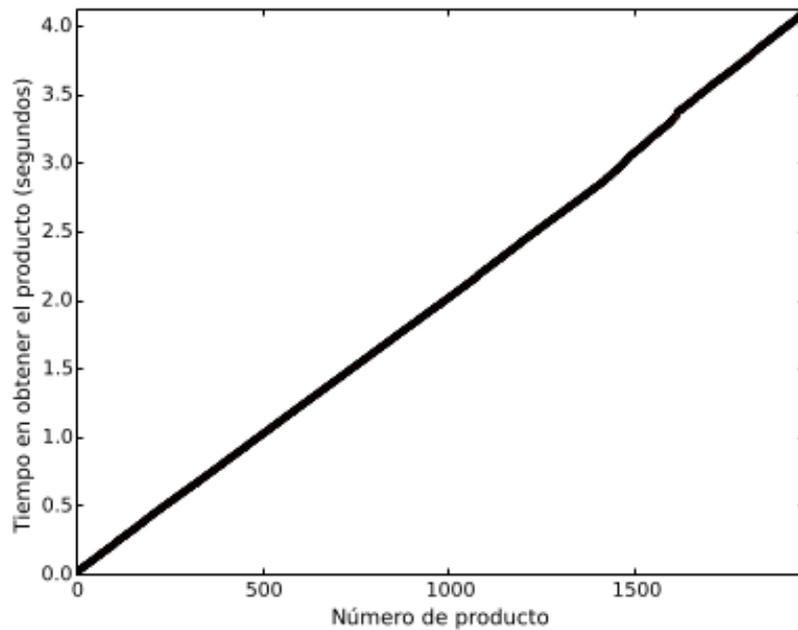


Tiempos de ejecución en función del número de hilos de ejecución del cluster

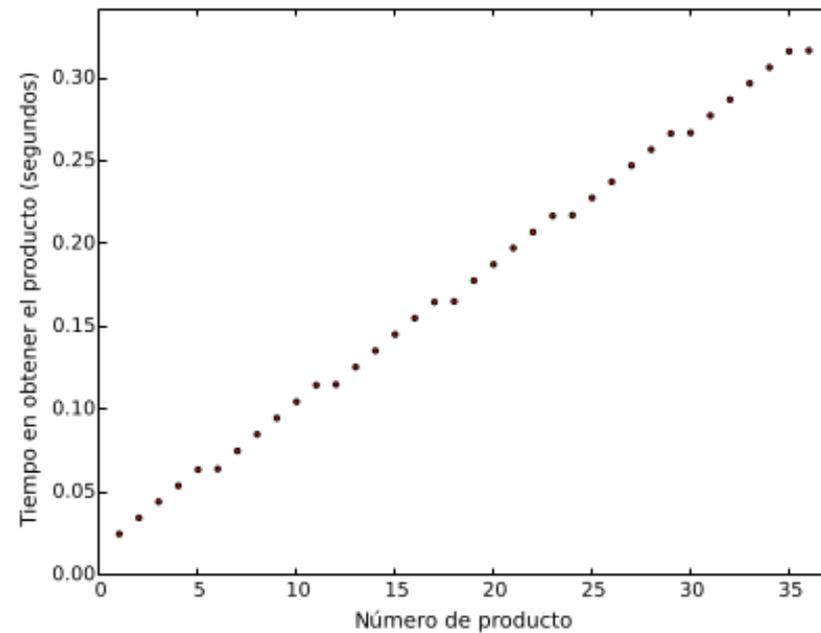
SPLAC

Consideraciones

Umbral de coste



Tiempos de procesamiento sin umbrales de coste (1957 productos válidos procesados en 4.02 segs.).



Tiempos de procesamiento con umbrales de coste (umbral entre de 1 y 2, 36 productos válidos en 0.45 segs.)

SPLA^P

Esta sección presenta la definición e implementación de la extensión probabilística de SPLA.

SPLA^P

Sintaxis

La extensión probabilística permite representar probabilidades en los términos del álgebra.

$$P ::= \checkmark \mid \text{nil} \mid A; P \mid \bar{A};_p P \mid P \vee_p Q \mid P \wedge Q \mid \\ A \not\Rightarrow B \text{ in } P \mid A \Rightarrow B \text{ in } P \mid P \setminus A \mid P \Rightarrow A$$

Elementos sintácticos representando probabilidades (opcional y selección única)

SPLA^P

Reglas de la semántica operacional

[tick]	$\checkmark \xrightarrow{1} \text{nil}$	[feat]	$A; P \xrightarrow{A} P$
[ofeat1]	$\bar{A}; P \xrightarrow{A} P$	[ofeat2]	$\bar{A}; P \xrightarrow{(1-p)} \text{nil}$
[cho1]	$\frac{P \xrightarrow{A} P_1}{P \vee_q Q \xrightarrow{A} P_1}$	[cho2]	$\frac{Q \xrightarrow{A} Q_1}{P \vee_p Q \xrightarrow{A} Q_1}$
[con1]	$\frac{P \xrightarrow{A} P_1}{P \wedge Q \xrightarrow{\frac{A}{2}} P_1 \wedge Q}$	[con2]	$\frac{Q \xrightarrow{A} Q_1}{P \wedge Q \xrightarrow{\frac{A}{2}} P \wedge Q_1}$
[con3]	$\frac{P \xrightarrow{A} \text{nil}, Q \xrightarrow{A} \text{nil}}{P \wedge Q \xrightarrow{A} \text{nil}}$	[con5]	$\frac{P \xrightarrow{A} \text{nil}, Q \xrightarrow{A} Q_1}{P \wedge Q \xrightarrow{\frac{A}{2}} Q_1}$
[con4]	$\frac{P \xrightarrow{A} P_1, Q \xrightarrow{A} \text{nil}}{P \wedge Q \xrightarrow{\frac{A}{2}} P_1}$	[req2]	$\frac{P \xrightarrow{A} P_1}{A \Rightarrow B \text{ in } P \xrightarrow{A} P_1 \Rightarrow B}$
[req1]	$\frac{P \xrightarrow{C} P_1, C \neq A}{A \Rightarrow B \text{ in } P \xrightarrow{C} P_1 \Rightarrow A \Rightarrow B \text{ in } P_1}$	[req3]	$\frac{P \xrightarrow{A} \text{nil}}{A \Rightarrow B \text{ in } P \xrightarrow{A} \text{nil}}$
[excl1]	$\frac{P \xrightarrow{C} P_1, C \neq A, C \neq B}{A \not\Rightarrow B \text{ in } P \xrightarrow{C} P_1 \not\Rightarrow A \not\Rightarrow B \text{ in } P_1}$	[excl2]	$\frac{P \xrightarrow{A} P_1}{A \not\Rightarrow B \text{ in } P \xrightarrow{A} P_1 \not\Rightarrow B}$
[excl3]	$\frac{P \xrightarrow{B} P_1}{A \not\Rightarrow B \text{ in } P \xrightarrow{B} P_1 \not\Rightarrow A}$	[excl4]	$\frac{P \xrightarrow{A} \text{nil}}{A \not\Rightarrow B \text{ in } P \xrightarrow{A} \text{nil}}$
[forb1]	$\frac{P \xrightarrow{B} P_1, B \neq A}{P \setminus A \xrightarrow{B} P_1 \setminus A}$	[forb2]	$\frac{P \xrightarrow{A} \text{nil}}{P \setminus A \xrightarrow{A} \text{nil}}$
[mand1]	$\frac{P \xrightarrow{A} \text{nil}}{P \Rightarrow A \xrightarrow{A} \checkmark}$	[mand2]	$\frac{P \xrightarrow{A} P_1}{P \Rightarrow A \xrightarrow{A} P_1}$
[mand3]	$\frac{P \xrightarrow{B} P_1, A \neq B}{P \Rightarrow A \xrightarrow{B} P_1 \Rightarrow A}$		

SPLA^P

Reglas de la semántica operacional

$$[\text{ofeat1}] \quad \bar{A};_p P \xrightarrow{A}_p P$$

$$[\text{ofeat2}] \quad \bar{A};_p P \xrightarrow{\checkmark}_{(1-p)} \text{nil}$$

$$[\text{con1}] \quad \frac{P \xrightarrow{A}_p P_1}{P \wedge Q \xrightarrow{A}_{\frac{p}{2}} P_1 \wedge Q}$$

$$[\text{con2}] \quad \frac{Q \xrightarrow{A}_q Q_1}{P \wedge Q \xrightarrow{A}_{\frac{q}{2}} P \wedge Q_1}$$

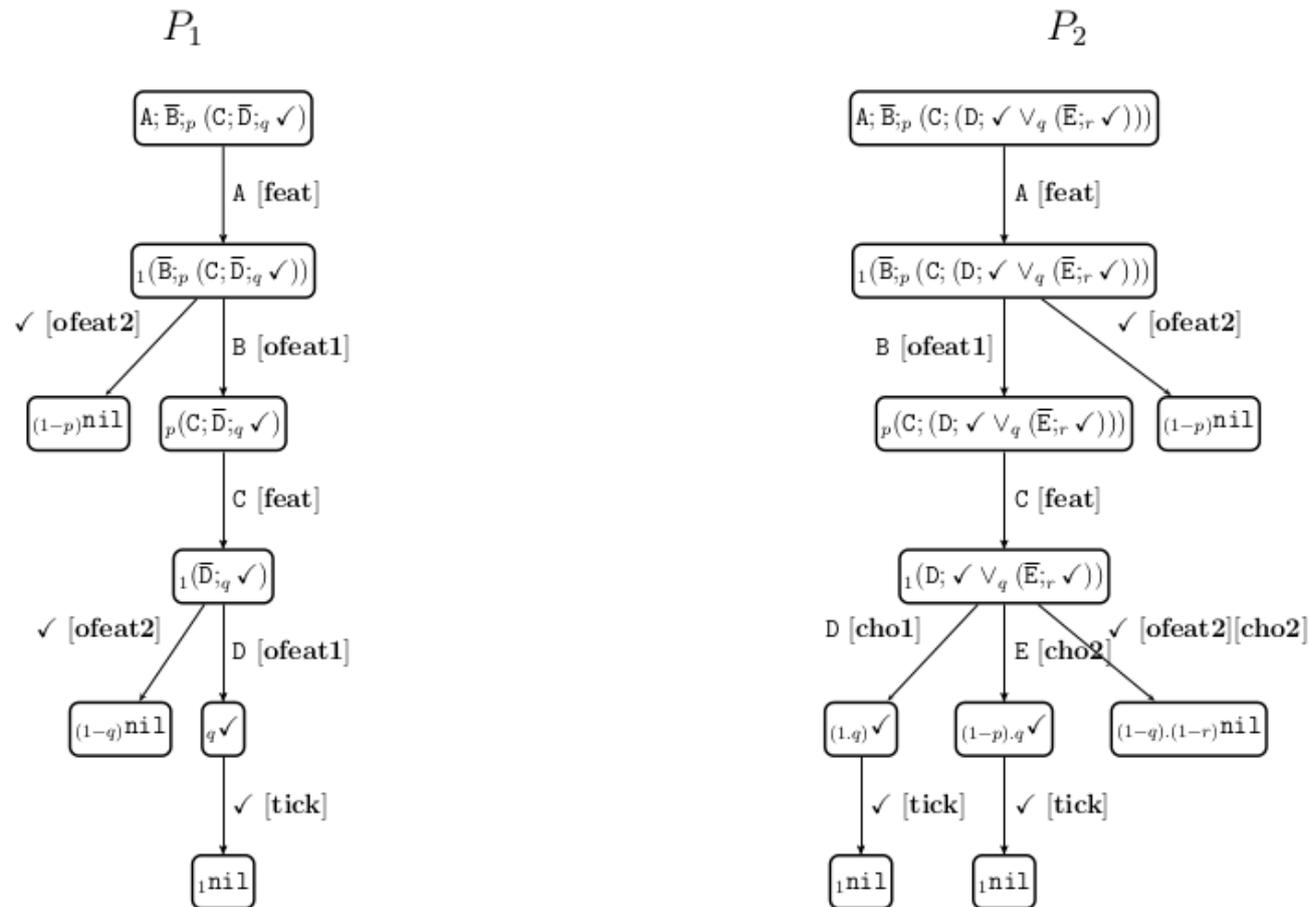
$$[\text{con3}] \quad \frac{P \xrightarrow{\checkmark}_q \text{nil}, Q \xrightarrow{\checkmark}_p \text{nil}}{P \wedge Q \xrightarrow{\checkmark}_{p \cdot q} \text{nil}}$$

$$[\text{con4}] \quad \frac{P \xrightarrow{A}_p P_1, Q \xrightarrow{\checkmark}_q \text{nil}}{P \wedge Q \xrightarrow{A}_{\frac{p \cdot q}{2}} P_1}$$

$$[\text{con5}] \quad \frac{P \xrightarrow{\checkmark}_p \text{nil}, Q \xrightarrow{A}_q Q_1}{P \wedge Q \xrightarrow{A}_{\frac{p \cdot q}{2}} Q_1}$$

SPLA^P

Procesamiento de la semántica operacional



Ejemplos del procesamiento de términos SPLA

SPLA^P

Reglas de la semántica denotacional

- $\llbracket \text{nil} \rrbracket^{\mathcal{P}} = \emptyset$
- $\llbracket \checkmark \rrbracket^{\mathcal{P}} = \{(\emptyset, 1)\}$
- $\llbracket \mathbf{A}; \cdot \rrbracket^{\mathcal{P}} : \mathcal{M} \mapsto \mathcal{M}$ como
$$\llbracket \mathbf{A}; \cdot \rrbracket^{\mathcal{P}}(M) = \text{accum}(\lambda(\{\mathbf{A}\} \cup P, p) \mid (P, p) \in M)$$
- $\llbracket \bar{\mathbf{A}};_r \cdot \rrbracket^{\mathcal{P}} : \mathcal{M} \mapsto \mathcal{M}$ como
$$\llbracket \bar{\mathbf{A}};_r \cdot \rrbracket^{\mathcal{P}}(M) = \text{accum}(\lambda(\emptyset, 1 - r) \uplus \lambda(\{\mathbf{A}\} \cup P, r \cdot p) \mid (P, p) \in M)$$
- $\llbracket \cdot \vee_r \cdot \rrbracket^{\mathcal{P}} : \mathcal{M} \times \mathcal{M} \mapsto \mathcal{M}$ como
$$\llbracket \cdot \vee_r \cdot \rrbracket^{\mathcal{P}}(M_1, M_2) = \text{accum}(\lambda(P, r \cdot p) \mid (P, p) \in M_1 \uplus \lambda(Q, (1 - r) \cdot q) \mid (Q, q) \in M_2)$$
- $\llbracket \cdot \wedge \cdot \rrbracket^{\mathcal{P}} : \mathcal{M} \times \mathcal{M} \mapsto \mathcal{M}$ como
$$\llbracket \cdot \wedge \cdot \rrbracket^{\mathcal{P}}(M_1, M_2) = \text{accum}(\lambda((P \cup Q, p \cdot q) \mid (P, p) \in M_1, (Q, q) \in M_2))$$
- $\llbracket \mathbf{A} \Rightarrow \mathbf{B} \text{ in } \cdot \rrbracket^{\mathcal{P}} : \mathcal{M} \mapsto \mathcal{M}$ como
$$\llbracket \mathbf{A} \Rightarrow \mathbf{B} \text{ in } \cdot \rrbracket^{\mathcal{P}}(M) = \text{accum} \left(\begin{array}{l} \lambda((P, p) \mid (P, p) \in M, \mathbf{A} \notin P) \uplus \\ \lambda(\{\mathbf{B}\} \cup P, p) \mid (P, p) \in M, \mathbf{A} \in P \end{array} \right)$$
- $\llbracket \mathbf{A} \not\Rightarrow \mathbf{B} \text{ in } \cdot \rrbracket^{\mathcal{P}} : \mathcal{M} \mapsto \mathcal{M}$ como
$$\llbracket \mathbf{A} \not\Rightarrow \mathbf{B} \text{ in } \cdot \rrbracket^{\mathcal{P}}(M) = \{(P, p) \mid (P, p) \in M, \mathbf{A} \notin P\} \cup \{(P, p) \mid (P, p) \in M, \mathbf{B} \notin P\}$$
- $\llbracket \cdot \Rightarrow \mathbf{A} \rrbracket^{\mathcal{P}} : \mathcal{M} \mapsto \mathcal{M}$ como
$$\llbracket \cdot \Rightarrow \mathbf{A} \rrbracket^{\mathcal{P}}(M) = \llbracket \mathbf{A}; \cdot \rrbracket^{\mathcal{P}}(M)$$
- $\llbracket \cdot \setminus \mathbf{A} \rrbracket^{\mathcal{P}} : \mathcal{M} \mapsto \mathcal{M}$ como
$$\llbracket \cdot \setminus \mathbf{A} \rrbracket^{\mathcal{P}}(M) = \{(P, p) \mid (P, p) \in M, \mathbf{A} \notin P\}$$

SPLA^P

Ejemplo ejecución de las reglas de la semántica denotacional

$$\begin{aligned}
 P_1 &= A; \bar{B};_p \checkmark \\
 \llbracket \checkmark \rrbracket^{\mathcal{P}} &= \{(\emptyset, 1)\} \\
 \llbracket \bar{B};_p \checkmark \rrbracket^{\mathcal{P}} &= \llbracket \bar{B};_p \cdot \rrbracket^{\mathcal{P}}(\llbracket \checkmark \rrbracket^{\mathcal{P}}) = \llbracket \bar{B};_p \cdot \rrbracket^{\mathcal{P}}(\{(\emptyset, 1)\}) = \\
 &\quad \{(\emptyset, (1-p)), (B, p)\} \\
 \llbracket A; \bar{B};_p \checkmark \rrbracket^{\mathcal{P}} &= \llbracket A; \cdot \rrbracket^{\mathcal{P}}(\llbracket \bar{B};_p \checkmark \rrbracket^{\mathcal{P}}) = \llbracket A; \cdot \rrbracket^{\mathcal{P}}(\{(\emptyset, (1-p)), (B, p)\}) = \\
 &\quad \{(A, (1-p)), (AB, p)\}
 \end{aligned}$$

$$\begin{aligned}
 P_2 &= A; (\bar{B};_p \checkmark \vee_q \bar{B};_r \checkmark) \\
 \llbracket \checkmark \rrbracket^{\mathcal{P}} &= \{(\emptyset, 1)\} \\
 \llbracket \bar{B};_p \checkmark \rrbracket^{\mathcal{P}} &= \llbracket \bar{B};_p \cdot \rrbracket^{\mathcal{P}}(\llbracket \checkmark \rrbracket^{\mathcal{P}}) = \llbracket \bar{B};_p \cdot \rrbracket^{\mathcal{P}}(\{(\emptyset, 1)\}) = \\
 &\quad \{(\emptyset, (1-p)), (B, p)\} \\
 \llbracket \bar{B};_r \checkmark \rrbracket^{\mathcal{P}} &= \llbracket \bar{B};_r \cdot \rrbracket^{\mathcal{P}}(\llbracket \checkmark \rrbracket^{\mathcal{P}}) = \llbracket \bar{B};_r \cdot \rrbracket^{\mathcal{P}}(\{(\emptyset, 1)\}) = \\
 &\quad \{(\emptyset, (1-r)), (B, r)\} \\
 \llbracket \bar{B};_p \checkmark \vee_q \bar{B};_r \checkmark \rrbracket^{\mathcal{P}} &= \llbracket \cdot \vee_q \cdot \rrbracket^{\mathcal{P}}(\llbracket \bar{B};_p \checkmark \rrbracket^{\mathcal{P}}, \llbracket \bar{B};_r \checkmark \rrbracket^{\mathcal{P}}) \\
 &= \llbracket \cdot \vee_q \cdot \rrbracket^{\mathcal{P}}(\{(\emptyset, (1-p)), (B, p)\}, \{(\emptyset, (1-r)), (B, r)\}) = \\
 &= \text{accum}\left(\{(\emptyset, (q) \cdot (1-p)), (B, q \cdot p)\} \uplus \{(\emptyset, (1-q) \cdot (1-r)), (B, (1-q) \cdot r)\}\right) = \\
 &= \{(\emptyset, ((q) \cdot (1-p)) + ((1-q) \cdot (1-r))), (B, (q \cdot p) + ((1-q) \cdot r))\}
 \end{aligned}$$

SPLA^P

Consistencia en la traducción

$$\text{np}(P) = \begin{cases} \checkmark & \text{if } P = \checkmark \\ \text{nil} & \text{if } P = \text{nil} \\ A; \text{np}(P) & \text{si } P = A; P \\ \bar{A}; \text{np}(P) & \text{si } P = \bar{A};_p P \\ \text{np}(P) \vee \text{np}(Q) & \text{si } P \vee_p Q \\ \text{np}(P) \wedge \text{np}(Q) & \text{si } P \wedge Q \\ A \Rightarrow B \text{ in } \text{np}(P) & \text{si } A \Rightarrow B \text{ in } P \\ A \not\Rightarrow B \text{ in } \text{np}(P) & \text{si } A \not\Rightarrow B \text{ in } P \\ \text{np}(P) \Rightarrow A & \text{si } P \Rightarrow A \\ \text{np}(P) \setminus A & \text{si } P \setminus A \end{cases}$$

Función de traducción del modelo probabilístico al modelo no probabilístico

SPLA^P

Ocultando conjuntos de características

$$[\text{hid1}] \quad \frac{P \xrightarrow{\Lambda}_p P', \Lambda \in \mathcal{A}}{P[\mathcal{A}] \xrightarrow{\perp}_p P'[\mathcal{A}]}$$

Se oculta la característica

$$[\text{hid2}] \quad \frac{P \xrightarrow{\Lambda}_p P', \Lambda \notin \mathcal{A}}{P[\mathcal{A}] \xrightarrow{\Lambda}_p P'[\mathcal{A}]}$$

No se oculta la característica

Para calcular la probabilidad de un conjunto de características se procederá a ocultar otras. Debido a que, por razones de coste computacional no es posible procesar todos los productos de la SPL.

SPLA^P

Implementación

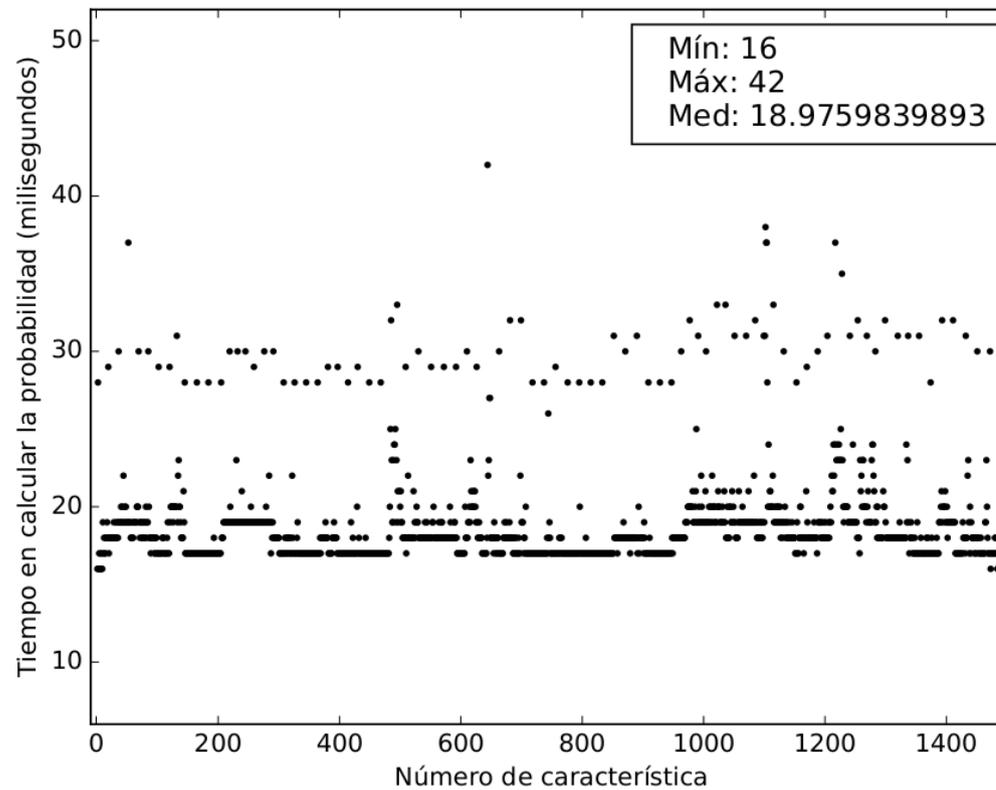
Se ha utilizado un modelo de variabilidad con 1500 características, este ha sido creado utilizando el el generador de modelos de características BeTTY. Los parámetros de configuración utilizados en BeTTY han sido los siguientes:

- La probabilidad de que exista una característica obligatoria es de 0.2.
- La probabilidad de que exista una característica opcional es de 0.3.
- La probabilidad de que una característica esté dentro de una relación de selección única es de 0.25.
- La probabilidad de que una característica esté dentro de una relación de paralelo es de 0.25.

*Probabilidades generadas siguiendo una distribución uniforme [0, 1]

SPLA^P

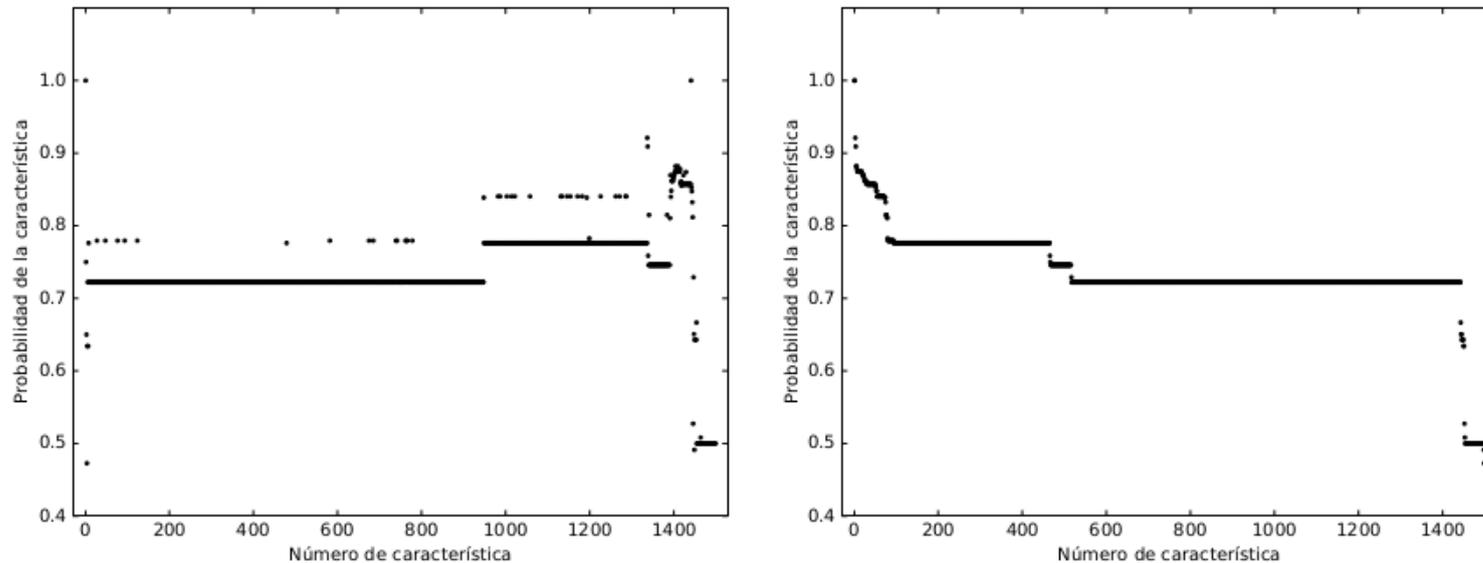
Implementación



Tiempo de procesamiento para un modelo con 1500 características

SPLA^P

Implementación



Probabilidad de las características del modelo de variabilidad.

Es posible identificar que las características con probabilidad mayor a 0.75 son sólo 450 características de las 1500 características existentes en el modelo.

Al realizar pruebas unitarias sobre ese 28 % de las características con probabilidad mayor a 0.75, se incluirían pruebas que abarcarían los componentes de al menos el 75 % de los productos generados.

CONCLUSIONES

Esta sección presenta las conclusiones y trabajo futuro de la tesis doctoral.

CONCLUSIONES

- En este trabajo de investigación se ha definido un marco teórico para modelar líneas de productos software, utilizando las álgebras de procesos como base teórica y FODA como modelo de variabilidad de referencia.
 - Marco teórico para representar diagramas FODA.
 - Extensión para representar costes (SPLAC).
 - Extensión para representar probabilidades (SPLAP).

CONCLUSIONES

- En este trabajo de investigación se ha definido un marco teórico para modelar líneas de productos software, utilizando las álgebras de procesos como base teórica y FODA como modelo de variabilidad de referencia.
- Marco teórico para representar diagramas FODA.
 - Extensión para representar costes (SPLAC).
 - Extensión para representar probabilidades (SPLAP).

CONCLUSIONES

- En este trabajo de investigación se ha definido un marco teórico para modelar líneas de productos software, utilizando las álgebras de procesos como base teórica y FODA como modelo de variabilidad de referencia.
- Marco teórico para representar diagramas FODA.
- Extensión para representar costes (SPLAC).
- Extensión para representar probabilidades (SPLAP).

CONCLUSIONES

- En este trabajo de investigación se ha definido un marco teórico para modelar líneas de productos software, utilizando las álgebras de procesos como base teórica y FODA como modelo de variabilidad de referencia.
- Marco teórico para representar diagramas FODA.
- Extensión para representar costes (SPLAC).
- Extensión para representar probabilidades (SPLAP).

PUBLICACIONES

- Tesis doctoral compuesta por los artículos:
 - C. Andres, C. Camacho y L. Llana. A formal framework for software product lines. *Information and Software Technology* 55 (11) (2013) 1925–1947. doi:10.1016/j.infsof.2013.05.005.
 - C. Camacho, L. Llana y A. Núñez. Cost-related interface for software product lines. *Journal of Logical and Algebraic Methods in Programming* 85 (2016) 227–244. doi:10.1016/j.jlamp.2015.09.009.

TRABAJO FUTURO

- Estudiar diferentes heurísticas que permitan eliminar aquellas características que no pertenezcan a ningún producto.
- Adaptar el formalismo presentado en esta tesis a herramientas de análisis como Uppaal Cora.
- Modelar sistemas de control para sistemas de computación en la nube (del inglés cloud computing) utilizando las herramientas y formalismos ya definidos.
- Análisis en tiempo real modelos de variabilidad que cambien de forma dinámica. Por ejemplo, al modelar los componentes en uso de una nube de cómputo, si la probabilidad de procesar un componente es 0, no sería utilizado.
- Desarrollar herramientas con interfaces gráficas para proveer al formalismo de una mejor usabilidad.

Future Work

- Estudiar diferentes heurísticas que permitan eliminar aquellas características que no pertenezcan a ningún producto.
- Adaptar el formalismo presentado en esta tesis a herramientas de análisis como Uppaal Cora.
- Modelar sistemas de control para sistemas de computación en la nube (del inglés cloud computing) utilizando las herramientas y formalismos ya definidos.
- Análisis en tiempo real modelos de variabilidad que cambien de forma dinámica. Por ejemplo, al modelar los componentes en uso de una nube de cómputo, si la probabilidad de procesar un componente es 0, no sería utilizado.
- Desarrollar herramientas con interfaces gráficas para proveer al formalismo de una mejor usabilidad.

Future Work

- Estudiar diferentes heurísticas que permitan eliminar aquellas características que no pertenezcan a ningún producto.
- Adaptar el formalismo presentado en esta tesis a herramientas de análisis como Uppaal Cora.
- **Modelar sistemas de control para sistemas de computación en la nube (del inglés cloud computing) utilizando las herramientas y formalismos ya definidos.**
- Análisis en tiempo real modelos de variabilidad que cambien de forma dinámica. Por ejemplo, al modelar los componentes en uso de una nube de cómputo, si la probabilidad de procesar un componente es 0, no sería utilizado.
- Desarrollar herramientas con interfaces gráficas para proveer al formalismo de una mejor usabilidad.

Future Work

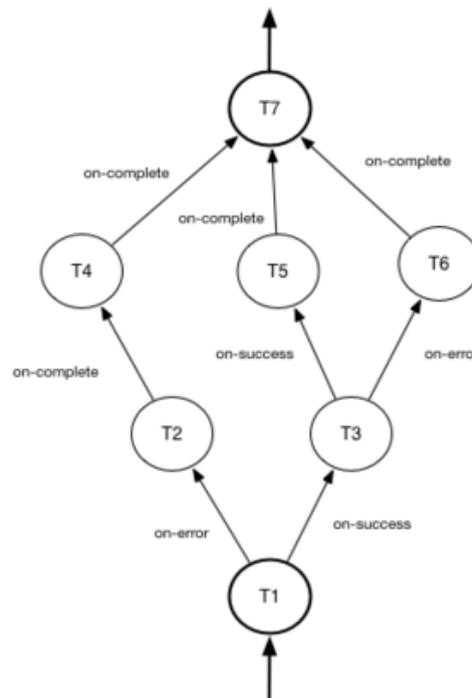
- Estudiar diferentes heurísticas que permitan eliminar aquellas características que no pertenezcan a ningún producto.
- Adaptar el formalismo presentado en esta tesis a herramientas de análisis como Uppaal Cora.
- Modelar sistemas de control para sistemas de computación en la nube (del inglés cloud computing) utilizando las herramientas y formalismos ya definidos.
- **Análisis en tiempo real modelos de variabilidad que cambien de forma dinámica. Por ejemplo, al modelar los componentes en uso de una nube de cómputo, si la probabilidad de procesar un componente es 0, no sería utilizado.**
- Desarrollar herramientas con interfaces gráficas para proveer al formalismo de una mejor usabilidad.

Future Work

- Estudiar diferentes heurísticas que permitan eliminar aquellas características que no pertenezcan a ningún producto.
- Adaptar el formalismo presentado en esta tesis a herramientas de análisis como Uppaal Cora.
- Modelar sistemas de control para sistemas de computación en la nube (del inglés cloud computing) utilizando las herramientas y formalismos ya definidos.
- Análisis en tiempo real modelos de variabilidad que cambien de forma dinámica. Por ejemplo, al modelar los componentes en uso de una nube de cómputo, si la probabilidad de procesar un componente es 0, no sería utilizado.
- **Desarrollar herramientas con interfaces gráficas para proveer al formalismo de una mejor usabilidad.**

Future Work

- Para las extensiones planteadas SPLA se deben buscar aplicaciones prácticas reales, como por ejemplo, representar workflows de Mistral como modelos de variabilidad para verificar errores antes de ejecutarlos.



Ejemplo de la descripción de un Workflow de Mistral