Lógica computacional

Tema: Lógica de Primer Orden. Conceptos sintácticos importantes.

Pilar Selene Linares Arévalo

Facultad de Ciencias
Universidad Nacional Autónoma de México

marzo 2018

Material desarrollado bajo el proyecto UNAM-PAPIME PE102117.

Ligado y alcance

La presencia de los cuantificadores $\forall x$ y $\exists x$ en Lógica de Predicados dan pie al fenómeno de *ligado* que también surge en la mayoría de los lenguajes de programación: ciertas presencias de variables son presencias *ligadas*, cada una de las cuales se asocia con una expresión llamada *alcance*.

Ligado y alcance

Alcance

Dada una fórmula cuantificada $\forall x \varphi$ o $\exists x \varphi$, la presencia de x en $\forall x \varphi$ o $\exists x \varphi$ es la variable (artificial) que liga el cuantificador correspondiente, mientras que la fórmula φ se llama el **alcance**, de este cuantificador.

Ligado y alcance

Alcance

Dada una fórmula cuantificada $\forall x\varphi$ o $\exists x\varphi$, la presencia de x en $\forall x\varphi$ o $\exists x\varphi$ es la variable (artificial) que liga el cuantificador correspondiente, mientras que la fórmula φ se llama el **alcance**, de este cuantificador.

Presencias libres y ligadas

Una presencia de la variable x en la fórmula φ está **ligada** si es la variable que liga a un cuantificador de φ o si figura en el alcance de un cuantificador $\forall x$ o $\exists x$ de φ.
Si una presencia de la variable x en la fórmula φ no está ligada, decimos que está **libre** en φ.

Ligado y alcance

Sea $\varphi = \forall x \exists z (Q(y,z) \vee R(z,x,y)) \wedge P(z,x)$

Ligado y alcance

Sea $\varphi = \forall x \exists z(Q(y,z) \vee R(z,x,y)) \wedge P(z,x)$

- El alcance del cuantificador $\forall$ es la fórmula $\exists z(Q(y,z) \vee R(z,x,y))$.

Ligado y alcance

Sea $\varphi = \forall x \exists z(Q(y,z) \vee R(z,x,y)) \wedge P(z,x)$

- El alcance del cuantificador $\forall$ es la fórmula $\exists z(Q(y,z) \vee R(z,x,y))$.
- El alcance del cuantificador $\exists$ es la fórmula $Q(y,z) \vee R(z,x,y)$.

Ligado y alcance

Sea $\varphi = \forall x \exists z(Q(y,z) \vee R(z,x,y)) \wedge P(z,x)$

- El alcance del cuantificador $\forall$ es la fórmula $\exists z(Q(y,z) \vee R(z,x,y))$.
- El alcance del cuantificador $\exists$ es la fórmula $Q(y,z) \vee R(z,x,y)$.
- En φ hay tres presencias de x: la primera es la variable asociada al cuantificador, la segunda es ligada y la tercera libre.

Ligado y alcance

Sea $\varphi = \forall x \exists z(Q(y, z) \vee R(z, x, y)) \wedge P(z, x)$

- El alcance del cuantificador $\forall$ es la fórmula $\exists z(Q(y, z) \vee R(z, x, y))$.
- El alcance del cuantificador $\exists$ es la fórmula $Q(y, z) \vee R(z, x, y)$.
- En φ hay tres presencias de x: la primera es la variable asociada al cuantificador, la segunda es ligada y la tercera libre.
- Las presencias de z son cuatro: la primera es la variable asociada al cuantificador, ligadas segunda y terceras y libre la última.

Ligado y alcance

Sea $\varphi = \forall x \exists z(Q(y,z) \vee R(z,x,y)) \wedge P(z,x)$

- El alcance del cuantificador $\forall$ es la fórmula $\exists z(Q(y,z) \vee R(z,x,y))$.
- El alcance del cuantificador $\exists$ es la fórmula $Q(y,z) \vee R(z,x,y)$.
- En φ hay tres presencias de x: la primera es la variable asociada al cuantificador, la segunda es ligada y la tercera libre.
- Las presencias de z son cuatro: la primera es la variable asociada al cuantificador, ligadas segunda y terceras y libre la última.
- Finalmente las dos presencias de y son libres.

Ligado y alcance

Conjunto de variables libres

Sea φ una fórmula. El **conjunto de variables libres** de φ, se denota $FV(\varphi)$. Es decir, $FV(\varphi) = \{x \in Var \mid x \text{ figura libre en } \varphi\}$.
La notación $\varphi(x_1, \ldots, x_n$ quiere decir que $\{x_1, \ldots, x_n \subseteq FV(\varphi)\}$.

Ligado y alcance

Conjunto de variables libres

Sea φ una fórmula. El **conjunto de variables libres** de φ, se denota $FV(\varphi)$. Es decir, $FV(\varphi) = \{x \in Var \mid x \text{ figura libre en } \varphi\}$.
La notación $\varphi(x_1, \ldots, x_n$ quiere decir que $\{x_1, \ldots, x_n \subseteq FV(\varphi)\}$.

Fórmula cerrada

Una fórmula φ es **cerrada** si no tiene variables libres, es decir, si $FV(\varphi) = \varnothing$. Una fórmula cerrada también se conoce como enunciado o sentencia.

Ligado y alcance

Conjunto de variables libres

Sea φ una fórmula. El **conjunto de variables libres** de φ, se denota $FV(\varphi)$. Es decir, $FV(\varphi) = \{x \in Var \mid x \text{ figura libre en } \varphi\}$.
La notación $\varphi(x_1, \ldots, x_n$ quiere decir que $\{x_1, \ldots, x_n \subseteq FV(\varphi)\}$.

Fórmula cerrada

Una fórmula φ es **cerrada** si no tiene variables libres, es decir, si $FV(\varphi) = \varnothing$. Una fórmula cerrada también se conoce como enunciado o sentencia.

Cerradura universal / existencial

Sea $\varphi(x_1, \ldots, x_n$ una fórmula con $FV(\varphi) = \{x_1, \ldots, x_n\}$. La cerradura universal de φ, denotada $\forall x \varphi$, es la fórmula $\forall x_1 \ldots \forall x_n \varphi$. La cerradura existencial de φ, denotada $\exists \varphi$ es la fórmula $\exists x_1 \ldots \exists x_n \varphi$.

Sustitución

A diferencia de lo que sucede en lógica proposicional, donde una sustitución es sólo una operación textual sobre las fórmulas, en la lógica de predicados las sustituciones son operaciones sobre los términos y sobre las fórmulas y en este último caso deben respetar el ligado de variables, por lo tanto no son operaciones textuales.

Sustitución

Sustitución

Una **sustitución** en un lenguaje de predicados $\mathcal{L}$ es una tupla de variables y términos denotada como $[x_1, x_2, \ldots, x_n := t_1, \ldots, t_n]$ donde

Por lo general denotaremos a esta sustitución con $[\overrightarrow{x} := \overrightarrow{t}]$

Sustitución

Sustitución

Una **sustitución** en un lenguaje de predicados $\mathcal{L}$ es una tupla de variables y términos denotada como $[x_1, x_2, \ldots, x_n := t_1, \ldots, t_n]$ donde

- $x_1, \ldots, x_n$ son variables distintas.

Por lo general denotaremos a esta sustitución con $[\overrightarrow{x} := \overrightarrow{t}]$

Sustitución

Sustitución

Una **sustitución** en un lenguaje de predicados $\mathcal{L}$ es una tupla de variables y términos denotada como $[x_1, x_2, \ldots, x_n := t_1, \ldots, t_n]$ donde

- $x_1, \ldots, x_n$ son variables distintas.
- $t_1, \ldots, t_n$ son términos de $\mathcal{L}$.

Por lo general denotaremos a esta sustitución con $[\overrightarrow{x} := \overrightarrow{t}]$

Sustitución

Sustitución

Una **sustitución** en un lenguaje de predicados $\mathcal{L}$ es una tupla de variables y términos denotada como $[x_1, x_2, \ldots, x_n := t_1, \ldots, t_n]$ donde

- $x_1, \ldots, x_n$ son variables distintas.
- $t_1, \ldots, t_n$ son términos de $\mathcal{L}$.
- $x_i \neq t_i$ para cada $1 \leqslant y \leqslant n$.

Por lo general denotaremos a esta sustitución con $[\overrightarrow{x} := \overrightarrow{t}]$

Sustitución

La aplicación de una sustitución $[\overrightarrow{x} := \overrightarrow{t}]$ a un término r, denotada $r[\overrightarrow{x} := \overrightarrow{t}]$, se define como el término obtenido al reemplazar simultáneamente todas las presencias de x_i en r por t_i . Este proceso de define recursivamente como sigue:

$$
\begin{array}{rcll}
x_i[\vec{x} := \vec{t}\,] & = & t_i & 1 \leqslant i \leqslant n \\
z[\vec{x} := \vec{t}\,] & = & z & \text{si } z \neq x_i \ 1 \leqslant i \leqslant n \\
c[\vec{x} := \vec{t}\,] & = & c & \text{si } c \in \mathcal{C}\text{, es decir, } c \text{ constante} \\
f(t_1, \ldots, t_m)[\vec{x} := \vec{t}\,] & = & f(t_1[\vec{x} := \vec{t}\,], \ldots, t_m[\vec{x} := \vec{t}\,]) & \text{con } f^{(m)} \in \mathcal{F}.
\end{array}
$$

Sustitución

La aplicación de sustituciones a fórmulas, necesita de ciertos cuidados debido a la presencia de variables ligadas mediante cuantificadores. La aplicación de una sustitución textual a una fórmula puede llevar a situaciones problemáticas con respecto a la sintaxis y a la semántica de la lógica. En particular deben evitarse los siguientes problemas:

- Generación de expresiones que no son fórmulas.

Sustitución

La aplicación de sustituciones a fórmulas, necesita de ciertos cuidados debido a la presencia de variables ligadas mediante cuantificadores. La aplicación de una sustitución textual a una fórmula puede llevar a situaciones problemáticas con respecto a la sintaxis y a la semántica de la lógica. En particular deben evitarse los siguientes problemas:

- Generación de expresiones que no son fórmulas.
- Captura de variables libres.

Sustitución

Aplicación sustitución a fórmulas

La aplicación de una sustitución $[\vec{x} := \vec{t}]$ a una fórmula φ, denotada $\varphi[\vec{x} := \vec{t}]$ se define como la fórmula obtenida al reemplazar *simultáneamente* todas las presencias libres de x_i en φ por t_i, verificando que este proceso no capture posiciones de variables libres.

La aplicación de una sustitución a una fórmula $\varphi[\vec{x} := \vec{t}\,]$ se define **recursivamente** como sigue:

$$\begin{array}{rcl} \bot[\vec{x} := \vec{t}\,] & = & \bot \\ \top[\vec{x} := \vec{t}\,] & = & \top \\ P(t_1, \ldots, t_m)[\vec{x} := \vec{t}\,] & = & P\big(t_1[\vec{x} := \vec{t}\,], \ldots, t_m[\vec{x} := \vec{t}\,]\big) \\ (t_1 = t_2)[\vec{x} := \vec{t}\,] & = & t_1[\vec{x} := \vec{t}\,] = t_2[\vec{x} := \vec{t}\,] \end{array}$$

La aplicación de una sustitución a una fórmula $\varphi[\vec{x} := \vec{t}\,]$ se define **recursivamente** como sigue:

$$
\begin{array}{rcl}
\bot[\vec{x} := \vec{t}\,] & = & \bot \\
\top[\vec{x} := \vec{t}\,] & = & \top \\
P(t_1, \ldots, t_m)[\vec{x} := \vec{t}\,] & = & P\big(t_1[\vec{x} := \vec{t}\,], \ldots, t_m[\vec{x} := \vec{t}\,]\big) \\
(t_1 = t_2)[\vec{x} := \vec{t}\,] & = & t_1[\vec{x} := \vec{t}\,] = t_2[\vec{x} := \vec{t}\,]
\end{array}
$$

$$
\begin{array}{rcl}
(\neg\varphi)[\vec{x} := \vec{t}\,] & = & \neg\big(\varphi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \wedge \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \wedge \psi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \vee \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \vee \psi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \to \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \to \psi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \leftrightarrow \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \leftrightarrow \psi[\vec{x} := \vec{t}\,]\big)
\end{array}
$$

La aplicación de una sustitución a una fórmula $\varphi[\vec{x} := \vec{t}\,]$ se define **recursivamente** como sigue:

$$
\begin{array}{rcl}
\bot[\vec{x} := \vec{t}\,] & = & \bot \\
\top[\vec{x} := \vec{t}\,] & = & \top \\
P(t_1, \ldots, t_m)[\vec{x} := \vec{t}\,] & = & P\big(t_1[\vec{x} := \vec{t}\,], \ldots, t_m[\vec{x} := \vec{t}\,]\big) \\
(t_1 = t_2)[\vec{x} := \vec{t}\,] & = & t_1[\vec{x} := \vec{t}\,] = t_2[\vec{x} := \vec{t}\,] \\
\\
(\neg\varphi)[\vec{x} := \vec{t}\,] & = & \neg\big(\varphi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \wedge \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \wedge \psi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \vee \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \vee \psi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \to \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \to \psi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \leftrightarrow \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \leftrightarrow \psi[\vec{x} := \vec{t}\,]\big) \\
\\
(\forall y\varphi)[\vec{x} := \vec{t}\,] & = & \forall y\big(\varphi[\vec{x} := \vec{t}\,]\big) \ \ \textbf{si}\ y \notin \vec{x} \cup Var(\vec{t})
\end{array}
$$

La aplicación de una sustitución a una fórmula $\varphi[\vec{x} := \vec{t}\,]$ se define **recursivamente** como sigue:

$$
\begin{array}{rcl}
\bot[\vec{x} := \vec{t}\,] & = & \bot \\
\top[\vec{x} := \vec{t}\,] & = & \top \\
P(t_1, \ldots, t_m)[\vec{x} := \vec{t}\,] & = & P\big(t_1[\vec{x} := \vec{t}\,], \ldots, t_m[\vec{x} := \vec{t}\,]\big) \\
(t_1 = t_2)[\vec{x} := \vec{t}\,] & = & t_1[\vec{x} := \vec{t}\,] = t_2[\vec{x} := \vec{t}\,] \\
 & & \\
(\neg\varphi)[\vec{x} := \vec{t}\,] & = & \neg\big(\varphi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \wedge \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \wedge \psi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \vee \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \vee \psi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \to \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \to \psi[\vec{x} := \vec{t}\,]\big) \\
(\varphi \leftrightarrow \psi)[\vec{x} := \vec{t}\,] & = & \big(\varphi[\vec{x} := \vec{t}\,] \leftrightarrow \psi[\vec{x} := \vec{t}\,]\big) \\
 & & \\
(\forall y\varphi)[\vec{x} := \vec{t}\,] & = & \forall y\big(\varphi[\vec{x} := \vec{t}\,]\big) \;\; \textbf{si} \;\; y \notin \vec{x} \cup Var(\vec{t}) \\
(\exists y\varphi)[\vec{x} := \vec{t}\,] & = & \exists y\big(\varphi[\vec{x} := \vec{t}\,]\big) \;\; \textbf{si} \;\; y \notin \vec{x} \cup Var(\vec{t})
\end{array}
$$

Sustitución

Propiedades de la sustitución

La operación de sustitución tiene las siguientes propiedades, considerando $x \notin \vec{x}$.

Propiedades de la sustitución

La operación de sustitución tiene las siguientes propiedades, considerando $x \notin \vec{x}$.

- Si $x \notin FV(\varphi)$ entonces $\varphi[\vec{x}, x := \vec{t}, r\,] = \varphi[\vec{x} := \vec{t}\,]$.

Sustitución

Propiedades de la sustitución

La operación de sustitución tiene las siguientes propiedades, considerando $x \notin \vec{x}$.

- Si $x \notin FV(\varphi)$ entonces $\varphi[\vec{x}, x := \vec{t}, r\,] = \varphi[\vec{x} := \vec{t}\,]$.
- Si $\vec{x} \cap FV(\varphi) = \varnothing$ entonces $\varphi[\vec{x} := \vec{t}\,] = \varphi$.

Propiedades de la sustitución

La operación de sustitución tiene las siguientes propiedades, considerando $x \notin \vec{x}$.

- Si $x \notin FV(\varphi)$ entonces $\varphi[\vec{x}, x := \vec{t}, r\,] = \varphi[\vec{x} := \vec{t}\,]$.
- Si $\vec{x} \cap FV(\varphi) = \varnothing$ entonces $\varphi[\vec{x} := \vec{t}\,] = \varphi$.
- $FV(\varphi[\vec{x} := \vec{t}\,]) \subseteq (FV(\varphi) - \vec{x}) \cup Var(\vec{t})$.

Sustitución

Propiedades de la sustitución

La operación de sustitución tiene las siguientes propiedades, considerando $x \notin \vec{x}$.

- Si $x \notin FV(\varphi)$ entonces $\varphi[\vec{x}, x := \vec{t}, r\,] = \varphi[\vec{x} := \vec{t}\,]$.
- Si $\vec{x} \cap FV(\varphi) = \varnothing$ entonces $\varphi[\vec{x} := \vec{t}\,] = \varphi$.
- $FV(\varphi[\vec{x} := \vec{t}\,]) \subseteq (FV(\varphi) - \vec{x}) \cup Var(\vec{t})$.
- Si $x \notin \vec{x} \cup Var(\vec{t}\,)$ entonces $x \in FV(\varphi)$ si y sólo si $x \in FV(\varphi[\vec{x} := \vec{t}\,])$.

Propiedades de la sustitución

La operación de sustitución tiene las siguientes propiedades, considerando $x \notin \vec{x}$.

- Si $x \notin FV(\varphi)$ entonces $\varphi[\vec{x}, x := \vec{t}, r\,] = \varphi[\vec{x} := \vec{t}\,]$.
- Si $\vec{x} \cap FV(\varphi) = \varnothing$ entonces $\varphi[\vec{x} := \vec{t}\,] = \varphi$.
- $FV(\varphi[\vec{x} := \vec{t}\,]) \subseteq (FV(\varphi) - \vec{x}) \cup Var(\vec{t})$.
- Si $x \notin \vec{x} \cup Var(\vec{t}\,)$ entonces $x \in FV(\varphi)$ si y sólo si $x \in FV(\varphi[\vec{x} := \vec{t}\,])$.
- Si $x \notin \vec{x} \cup Var(\vec{t}\,)$ entonces $\varphi[x := t][\vec{x} := \vec{t}\,] = \varphi\big[\vec{x} := \vec{t}\,\big]\big[x := t[\vec{x} := \vec{t}\,]\big]$.

Sustitución

Propiedades de la sustitución

La operación de sustitución tiene las siguientes propiedades, considerando $x \notin \vec{x}$.

- Si $x \notin FV(\varphi)$ entonces $\varphi[\vec{x}, x := \vec{t}, r\,] = \varphi[\vec{x} := \vec{t}\,]$.
- Si $\vec{x} \cap FV(\varphi) = \varnothing$ entonces $\varphi[\vec{x} := \vec{t}\,] = \varphi$.
- $FV(\varphi[\vec{x} := \vec{t}\,]) \subseteq (FV(\varphi) - \vec{x}) \cup Var(\vec{t})$.
- Si $x \notin \vec{x} \cup Var(\vec{t}\,)$ entonces $x \in FV(\varphi)$ si y sólo si $x \in FV(\varphi[\vec{x} := \vec{t}\,])$.
- Si $x \notin \vec{x} \cup Var(\vec{t}\,)$ entonces $\varphi[x := t][\vec{x} := \vec{t}\,] = \varphi\big[\vec{x} := \vec{t}\,\big]\big[x := t[\vec{x} := \vec{t}\,]\big]$.
- Si $x \notin \vec{x} \cup Var(\vec{t}\,)$ entonces $\varphi[\vec{x}, x := \vec{t}, t] = \varphi[\vec{x} := \vec{t}\,][x := t]$.

α equivalencia

La definición de sustitución en fórmulas cuenta con una restricción aparente en el caso de los cuantificadores, por ejemplo, la sustitución

α equivalencia

La definición de sustitución en fórmulas cuenta con una restricción aparente en el caso de los cuantificadores, por ejemplo, la sustitución

$$\forall x(Q(x) \to R(z, x))[z := f(x)]$$

α equivalencia

La definición de sustitución en fórmulas cuenta con una restricción aparente en el caso de los cuantificadores, por ejemplo, la sustitución

$$\forall x(Q(x) \to R(z, x))[z := f(x)]$$

no está definida, puesto que x figura en $f(x)$ es decir $x \in Var(f(x))$, con lo que no se cumple la condición necesaria para aplicar la sustitución, a saber que las variables en la sustitución son ajenas a las de la fórmula. Por lo tanto, la aplicación de cualquier sustitución a una fórmula hasta ahora es una función parcial.

α equivalencia

Esta aparente restricción desaparece al notar que los nombres de las variables ligadas no importan: por ejemplo, las fórmulas $\forall x P(x)$ y $\forall y P(y)$ significan exactamente lo mismo, a saber que todos los elementos del universo dado cumplen la propiedad P.

α equivalencia

Esta aparente restricción desaparece al notar que los nombres de las variables ligadas no importan: por ejemplo, las fórmulas $\forall x P(x)$ y $\forall y P(y)$ significan exactamente lo mismo, a saber que todos los elementos del universo dado cumplen la propiedad P.

Por lo tanto convenimos en identificar fórmulas que sólo difieren en sus variables ligadas.

α equivalencia

Decimos que dos fórmulas φ_1, φ_2 son α-**equivalentes** lo cual escribimos $\varphi_1 \sim_\alpha \varphi_2$ si y sólo si φ_1 y φ_2 difieren a lo más en los nombres de sus variables ligadas.

α equivalencia

α equivalencia

Decimos que dos fórmulas φ_1, φ_2 son α-**equivalentes** lo cual escribimos $\varphi_1 \sim_\alpha \varphi_2$ si y sólo si φ_1 y φ_2 difieren a lo más en los nombres de sus variables ligadas.

α equivalencia

α equivalencia

Decimos que dos fórmulas φ_1, φ_2 son α-**equivalentes** lo cual escribimos $\varphi_1 \sim_\alpha \varphi_2$ si y sólo si φ_1 y φ_2 difieren a lo más en los nombres de sus variables ligadas.

Las siguientes expresiones son α-equivalentes.

$$\forall x P(x, y) \to \exists y R(x, y, z) \quad \sim_\alpha \quad \forall w P(w, y) \to \exists v R(x, v, z)$$

$$\forall w P(w, y) \to \exists v R(x, v, z) \quad \sim_\alpha \quad \forall z P(z, y) \to \exists u R(x, u, z)$$

α equivalencia

Usando la α-equivalencia, la operación de sustitución en fórmulas se vuelve una función *total* por lo que siempre está definida.

α equivalencia

Usando la α-equivalencia, la operación de sustitución en fórmulas se vuelve una función *total* por lo que siempre está definida.

Por ejemplo se tiene que

$$
\begin{aligned}
\forall x(Q(x) \to R(z,x))[z := f(x)] &= \forall y(Q(y) \to R(z,y))[z := f(x)] \\
&= \forall y(Q(y) \to R(f(x),y))
\end{aligned}
$$