Análisis eficiente de Gramáticas de Inserción de Árboles

Vicente Carrillo†*, Miguel A. Alonso*‡*, Víctor J. Díaz*†

†Departamento de Lenguajes y Sistemas Informáticos
Universidad de Sevilla
{carrillo,vjdiaz}@lsi.us.es

‡Departamento de Computación
Universidade da Coruña
alonso@udc.es

Resumen

El formalismo *Tree Insertion Grammar* (TIG) es un compromiso entre *Context Free Grammar* (CFG) y *Tree Adjoining Grammar* (TAG) que puede ser analizada con un coste temporal de $\mathcal{O}(n^3)$. En la literatura podemos encontrar algunos analizadores para TIG derivados a partir de los existentes para CFG, como los populares CYK y Earley. En este trabajo extendemos para las gramáticas de inserción de árboles el concepto de *left-corner* conocido para CGF, y a partir de él, definimos un nuevo analizador para TIG que mejora las prestaciones del clásico analizador tipo earley mediante una reducción en el número de predicciones.

1. Introducción

Las gramáticas de adjunción de árboles (*Tree Adjoining Grammar*, TAG) [1] constituyen un formalismo naturalmente lexicalizado muy adecuado para la descripción de la sintaxis de los lenguajes naturales. Como contrapartida, el proceso de análisis para este formalismo implica mayores costes computacionales que el mismo proceso para las gramáticas independientes del contexto (*Context Free Grammar*, CFG): la complejidad temporal en el caso peor de los analizadores para TAG es de $\mathcal{O}(n^6)$, donde n es la longitud de la cadena de entrada, frente a la complejidad $\mathcal{O}(n^3)$ que presentan los analizadores para CFG. En los últimos años, se han descrito muchas aproximaciones que intentan mejorar las prestaciones de los analizadores para TAG: unas basadas en la compilación de los árboles elementales en autómatas de estados finitos [2], otras que aplican ciertos filtros a los algoritmos de análisis [3, 4, 5] y otras, como la empleada en este trabajo, basadas en restricciones sobre el formalismo [6].

Las gramáticas de inserción de árboles (*Tree Insertion Grammar*, TIG) [6] constituyen un compromiso entre CFG y TAG que combina la eficiencia de análisis de las primeras con la fuerte lexicalización de las segundas, ya que, al igual que ocurre con las CFGs, cualquier TIG se puede analizar con un coste temporal de $O(n^3)$ en el peor caso y, por otra parte, al ser las TIGs una subclase de las TAGs, se encuentran naturalmente lexicalizadas. La importancia del formalismo TIG se fundamenta en el hecho de que la mayoría de las gramáticas de adjunción de árboles de amplia cobertura se corresponden en su mayor parte con dicho formalismo. Esta afirmación se puede comprobar en la gramática del inglés XTAG [7], donde el 99 % de los árboles y adjunciones posibles son compatibles con el formalismo TIG.

La mayoría de los analizadores para TAG y TIG son extensiones de analizadores bien conocidos para CFG. En la literatura podemos encontrar multitud de analizadores para TAG, algunos usan una estrategia ascendente [8, 9, 10], otros utilizan estrategias ascendentes predictivas de manera similar al algoritmo de Earley para CFG [8, 11] y otros, como [5, 4], usan una adaptación para TAG del conocido filtro *left corner* para CFG con objeto de aumentar las prestaciones de los analizadores predictivos.

Podría pensarse que los analizadores para TIG pueden ser derivados directamente de los analizadores para TAG, dadas las similitudes entre ambos formalismos. Sin embargo, los aspectos que los diferencian son lo suficientemente significativos como para hacer que tal adaptación no sea sencilla de realizar. Como ilustración, podemos considerar la enorme diferencia existente entre el analizador de tipo earley para TAG y el analizador de tipo earley para TIG definido en [6].En [12] se presentan un conjunto de analizadores para TIG, concretamente cuatro, tres que usan estrategia ascendente y uno ascendente predictivo. En este trabajo ampliamos esta red de analizadores para TIG, introduciendo un nuevo analizador que aplica un filtro de tipo left corner a una variante del analizador ascendente predictivo presentado en [12].

El artículo se encuentra estructurado de la siguiente manera. La sección 2 introduce la notación necesaria para el resto del artículo. En la sección 3 se presenta una variante del analizador para TIG basado en earley de [12]. En la sección 4 se establece el concepto de relación de left corner (esquina izquierda) para TIG, mediante una extensión de la conocida definición de relación de left corner para las reglas de producción en CFG a los árboles elementales en TIG. La sección 5 presenta el nuevo analizador obtenido mediante la aplicación de la relación de esquina izquierda definida en la sección 4 al analizador tipo earley introducido en la sección 3. La sección 6 presenta las conclusiones finales.

2. Notación

2.1. Gramáticas de inserción de árboles

Una TIG es una 5-tupla $(V_N, V_T, S, \boldsymbol{I}, \boldsymbol{A})$, donde V_N es un conjunto de símbolos no terminales, V_T es un conjunto de símbolos terminales, $S \in V_N$ es el axioma, $\boldsymbol{I}$ es un conjunto finito de *árboles iniciales* finitos y $\boldsymbol{A}$ es un conjunto finito de *árboles auxiliares* finitos. Al conjunto $\boldsymbol{I} \cup \boldsymbol{A}$ se le denomina *árboles elementales*. Nos referiremos a la raíz de un árbol elemental γ como $\mathbf{R}^\gamma$. En cada árbol elemental, los nodos de la frontera se etiquetan con símbolos terminales, la palabra vacía (ε) o símbolos no terminales marcados para sustitución, excepto un nodo en cada árbol auxiliar, cuya etiqueta es la misma que la de la raíz y que se denomina nodo *pie*. Denotaremos como $\mathbf{F}^\beta$ al nodo pie de un árbol auxiliar β. Denominamos *espina* al camino de la raíz al pie de un árbol auxiliar. Denominaremos *nodos izquierdos* (resp. *nodos derechos*) a aquellos nodos situados a la izquierda (resp. a la derecha) de la espina. Usaremos $label(M^\gamma)$ para denotar la etiqueta asociada al nodo M^γ.

Los árboles auxiliares en los cuales todo nodo frontera distinto de ε se encuentra a la izquierda (resp. derecha) del nodo pie se denominan *árboles auxiliares izquierdos (resp. derechos)*. El resto de árboles auxiliares se denominan *árboles wrapping*. Usaremos $\boldsymbol{A_L}$ y $\boldsymbol{A_R}$ para denotar los conjuntos de árboles auxiliares izquierdos y derechos, respectivamente.

Una derivación TIG comienza con un árbol inicial cuya raíz está etiquetada por S. Este árbol se extiende repetidamente usando las operaciones de *adjunción* y *sustitución*. La *adjunción* inserta un árbol auxiliar β en el nodo M^γ de un árbol γ que tenga la misma etiqueta que $\mathbf{R}^\beta$. En concreto, M^γ es reemplazado por β y $\mathbf{F}^\beta$ es reemplazado por el subárbol dominado por M^γ. Usaremos $\beta \in adj(M^\gamma)$ para denotar que un árbol $\beta \in \boldsymbol{A}$ puede ser adjuntado en un nodo M^γ, es decir, M^γ es un nodo de adjunción. Si la adjunción no es obligatoria en M^γ entonces $\mathbf{nil} \in adj(M^\gamma)$, donde $\mathbf{nil}$ es un símbolo vacío. La adjunción de un árbol auxiliar izquierdo (resp. derecho) se denomina *adjunción izquierda (resp. derecha)*. Usaremos $\beta \in \mathrm{ladj}(M^\gamma)$ (resp. $\beta \in \mathrm{radj}(M^\gamma)$) para denotar que $\beta \in \boldsymbol{A_L}$ (resp. $\beta \in \boldsymbol{A_R}$) se puede adjuntar en el nodo M^γ, es decir, M^γ es un nodo de adjunción izquierda (resp. derecha). Si una adjunción izquierda (resp. derecha) no es obligatoria en el nodo M^γ entonces $\mathbf{nil} \in \mathrm{ladj}(M^\gamma)$ (resp. $\mathbf{nil} \in \mathrm{radj}(M^\gamma)$). La *sustitución* es una operación obligatoria y reemplaza un nodo marcado para sustitución M^γ con una copia de un árbol inicial α cuya raíz esté etiquetada igual que M^γ. Usamos $\alpha \in \mathrm{subst}(M^\gamma)$ para indicar que el nodo M^γ puede ser sustituido por el árbol $\alpha \in \boldsymbol{I}$.

TIG no permite: (1) árboles auxiliares *wrapping*, (2) la adjunción de un árbol auxiliar izquierdo (resp. derecho) en la espina de un árbol auxiliar derecho (resp. izquierdo), (3) la adjunción en los nodos izquierdos (resp. derechos) de los árboles auxiliares derechos (resp. izquierdos) y (4) la adjunción en los nodos raíz y pie de los árboles auxiliares. Para incrementar los árboles que se pueden generar, TIG permite un número arbitrario de adjunciones simultáneas sobre un mismo nodo. La *adjunción simultánea* es una operación esencialmente ambigua y provoca la creación de muchos árboles diferentes. Fácilmente se pueden imaginar variantes de TIG donde la adjunción simultánea esté más limitada. Para no incrementar la ambiguedad de la derivación, hemos elegido la variante de TIG presentada en [13] que como máximo permite una adjunción izquierda y otra derecha sobre un nodo. Además, para mantener los árboles que se pueden generar mediante adjunción simultánea, permitiremos la adjunción en los nodos raíz y pie de los árboles auxiliares.

Con objeto de representar los árboles de análisis parciales, definimos una producción $N^\gamma \rightarrow N_1^\gamma \ldots N_g^\gamma$ para cada nodo N^γ y sus secuencia ordenada de g hijos $N_1^\gamma \ldots N_g^\gamma$ en un árbol elemental. Denotaremos el conjunto de producciones asociado a un árbol elemental γ como $\mathcal{P}(\gamma)$. Por razones técnicas,consideramos las producciones adicionales $\top \rightarrow \mathbf{R}^\alpha$, $\top \rightarrow \mathbf{R}^\beta$ y $\mathbf{F}^\beta \rightarrow \bot$ para cada árbol inicial α y cada árbol auxiliar β. Para mantener la capacidad generativa de la gramática, se prohíbe la adjunción y sustitución en los nodos $\top$ y $\bot$.

2.2. Esquemas de análisis

Los algoritmos de análisis sintáctico se pueden definir como sistemas deductivos [14], donde las fórmulas, llamadas ítems, son conjuntos de constituyentes completos o incompletos. En este trabajo, describiremos los algoritmos de análisis mediante esquemas de análisis sintáctico [15]. Un *sistema de análisis sintáctico* para una gramática $\mathcal{G}$ y una cadena de entrada $a_1 \ldots a_n$ es un triple $\mathbb{P} = \langle \mathcal{I}, \mathcal{H}, \mathcal{D} \rangle$, donde $\mathcal{I}$ es un conjunto de *ítems* que representan resultados intermedios del proceso de análisis, $\mathcal{H}$ es un conjunto inicial de ítems denominado *hipótesis* que representan la cadena que va a ser analizada y $\mathcal{D}$ es un conjunto de reglas de inferencia denominadas *pasos deductivos*, mediante las cuales se derivan nuevos ítems ξ a partir de los ítems η_i existentes. Estos pasos deductivos tienen la forma $\frac{\eta_1,\ldots,\eta_k}{\xi}\, cond$ y su significado es el siguiente: si todos los antecedentes $\eta_i \in \mathcal{H} \cup \mathcal{I}$ de un paso deductivo ya existen y se satisfacen las condiciones *cond*, entonces el consecuente ξ deberá ser generado por el analizador sintáctico. La presencia de un conjunto $\mathcal{F} \subseteq \mathcal{I}$ de *ítems finales* indica el reconocimiento de la cadena de entrada. Un *esquema de análisis sintáctico* es un sistema de análisis sintáctico parametrizado para una gramática y una cadena de entrada.

3. Esquema tipo earley

Aquí presentamos un algoritmo eficiente de análisis *left-to-right* para TIGs que combina predicciones descendentes con reconocimiento ascendente, como el algoritmo de Earley para CFGs. Una variante de este esquema , al que denominaremos $\mathbf{Earley^i}$, lo introducimos en [12] y se obtiene mediante una adaptación del esquema presentado en [6] a la restricción impuesta en este trabajo que obliga a que como máximo se efectúe una adjunción izquierda y otra derecha sobre un nodo.

El dominio del esquema $\mathbf{Earley^i}$ es:

$$\mathcal{I}_{\mathrm{Earley^i}} = \mathcal{I}^{(\mathrm{i})}_{\mathrm{Earley^i}} \cup \mathcal{I}^{(\mathrm{ii})}_{\mathrm{Earley^i}}$$

$$\mathcal{I}^{(\mathrm{i})}_{\mathrm{Earley^i}} = \{[M^\gamma \rightarrow \delta \bullet \nu, i, j, false] \mid M^\gamma \rightarrow \delta\nu \in \mathcal{P}(\gamma)\,,$$

$$\gamma \in \boldsymbol{I} \cup \boldsymbol{A}\,, 0 \leq i \leq j\,, \nu \neq \varepsilon\}$$

$$\mathcal{I}^{(\mathrm{ii})}_{\mathrm{Earley^i}} = \{[M^\gamma \rightarrow \nu \bullet, i, j, radj] \mid M^\gamma \rightarrow \nu \in \mathcal{P}(\gamma)\,,$$

$$\gamma \in \boldsymbol{I} \cup \boldsymbol{A}\,, 0 \leq i \leq j\,, radj \in \{true, false\}\}$$

El parámetro booleano $radj$ se usa para controlar si se ha efectuado una adjunción derecha sobre el nodo M^γ. De manera que $radj = false$ si no se completó ninguna adjunción derecha sobre el nodo M^γ y $radj = true$ si se completó una adjunción derecha sobre M^γ.

El conjunto de pasos deductivos del esquema viene dado por:

$$\mathcal{D}_{\mathrm{Earley^i}} = \mathcal{D}^{\mathrm{Ini}}_{\mathrm{Earley^i}} \cup \mathcal{D}^{\mathrm{Scan}}_{\mathrm{Earley^i}} \cup \mathcal{D}^{\varepsilon}_{\mathrm{Earley^i}} \cup \mathcal{D}^{\mathrm{Pred}}_{\mathrm{Earley^i}} \cup$$

$$\mathcal{D}^{\mathrm{Comp}}_{\mathrm{Earley^i}} \cup \mathcal{D}^{\mathrm{LAdjPred}}_{\mathrm{Earley^i}} \cup \mathcal{D}^{\mathrm{LAdjComp}}_{\mathrm{Earley^i}} \cup \mathcal{D}^{\mathrm{RadjPred}}_{\mathrm{Earley^i}} \cup$$

$$\mathcal{D}^{\mathrm{RAdjComp}}_{\mathrm{Earley^i}} \cup \mathcal{D}^{\mathrm{SubsPred}}_{\mathrm{Earley^i}} \cup \mathcal{D}^{\mathrm{SubsComp}}_{\mathrm{Earley^i}}$$

Inicio

El reconocimiento comienza con la predicción de todo árbol inicial cuya raíz sea el axioma:

$$\mathcal{D}^{\text{Ini}}_{\text{Earley}^{\text{i}}} = \frac{}{[\top \to \bullet\mathbf{R}^{\alpha}, 0, 0, false]} \quad \begin{array}{l} \alpha \in \boldsymbol{I} \\ label(\mathbf{R}^{\alpha}) = S \end{array}$$

Reconocimiento

Los pasos de reconocimiento son los que efectúan la lectura de símbolos de la cadena de entrada. Obsérvese como el paso $\mathcal{D}^{\varepsilon}_{\text{Earley}^{\text{i}}}$ ignora tanto la cadena vacía como los nodos $\perp$ de los árboles auxiliares, ya que éstos no subsumen nada.

$$\mathcal{D}^{\text{Scan}}_{\text{Earley}^{\text{i}}} = \frac{\begin{array}{c}[a, j, j+1] \\ [N^{\gamma} \to \delta \bullet M^{\gamma}\nu, i, j, false]\end{array}}{[N^{\gamma} \to \delta M^{\gamma} \bullet \nu, i, j+1, false]} \; label(M^{\gamma}) = a$$

$$\mathcal{D}^{\varepsilon}_{\text{Earley}^{\text{i}}} = \frac{[N^{\gamma} \to \delta \bullet M^{\gamma}\nu, i, j, false]}{[N^{\gamma} \to \delta M^{\gamma} \bullet \nu, i, j, false]} \quad \begin{array}{l} label(M^{\gamma}) = \varepsilon \\ \text{ó } label(M^{\gamma}) = \perp \end{array}$$

Los pasos deductivos que establecen la estrategia ascendente predictiva del algoritmo de Earley son los correspondientes a predicciones y compleciones. Para el caso de las TIGs, vamos a distinguir cuatro tipos de predicciones con sus correspondientes pasos de compleción asociados: subárbol, adjunción izquierda, adjunción derecha y sustitución.

Predicción de subárbol

Si el reconocimiento alcanza un nodo que no presenta adjunción izquierda obligatoria, el análisis debe continuar el reconocimiento descendente del subárbol dominado por dicho nodo.

$$\mathcal{D}^{\text{Pred}}_{\text{Earley}^{\text{i}}} = \frac{[N^{\gamma} \to \delta \bullet M^{\gamma}\nu, i, j, false]}{[M^{\gamma} \to \bullet\nu, j, j, false]} \; \mathbf{nil} \in \text{ladj}(M^{\gamma})$$

Compleción de subárbol

Una vez completado el reconocimiento de un subárbol, el análisis debe continuar el reconocimiento ascendente.

$$\mathcal{D}^{\text{Comp}}_{\text{Earley}^{\text{i}}} = \frac{\begin{array}{c}[M^{\gamma} \to \nu\bullet, j, k, radj] \\ [N^{\gamma} \to \delta \bullet M^{\gamma}\nu, i, j, false]\end{array}}{[N^{\gamma} \to \delta M^{\gamma} \bullet \nu, i, k, false]}$$

Para continuar el reconocimiento ascendente será necesario comprobar que el nodo M^{γ} no presente adjunción derecha obligatoria o, en el caso de presentarla, que se haya efectuado la misma. Por tanto, se debe cumplir la condición de que si $radj = false$ entonces $\mathbf{nil} \in \text{radj}(M^{\gamma})$.

Predicción de adjunción izquierda

Cuando el reconocimiento alcanza un nodo adjuntable por la izquierda, el análisis debe lanzar el reconocimiento de todos los árboles auxiliares izquierdos que se pueden adjuntar en él.

$$\mathcal{D}^{\text{LAdjPred}}_{\text{Earley}^{\text{i}}} = \frac{[N^{\gamma} \to \delta \bullet M^{\gamma}\nu, i, j, false]}{[\top \to \bullet\mathbf{R}^{\beta}, j, j, false]} \; \beta \in \text{ladj}(M^{\gamma})$$

Compleción de adjunción izquierda

Una vez que se ha completado el reconocimiento de un árbol auxiliar izquierdo, debemos continuar el reconocimiento del árbol donde se ha efectuado la adjunción.

$$\mathcal{D}^{\text{LAdjComp}}_{\text{Earley}^{\text{i}}} = \frac{\begin{array}{c}[\top \to \mathbf{R}^{\beta}\bullet, j, k, false] \\ [N^{\gamma} \to \delta \bullet M^{\gamma}\nu, i, j, false]\end{array}}{[M^{\gamma} \to \bullet\nu, j, k, false]} \; \beta \in \text{ladj}(M^{\gamma})$$

Este paso incluye una diferencia reseñable respecto al mismo del analizador original presentado en [6]: el antecedente de la forma $[N^{\gamma} \to \delta \bullet M^{\gamma}\nu, i, j, false]$, el cual evita que se puedan producir más de una adjunción izquierda sobre el nodo M^{γ}.

Predicción de adjunción derecha

Cuando se completa el reconocimiento de un subárbol dominado por un nodo adjuntable por la derecha, el análisis debe lanzar el reconocimiento de todos los árboles auxiliares derechos que se pueden adjuntar en él.

$$\mathcal{D}^{\text{RAdjPred}}_{\text{Earley}^{\text{i}}} = \frac{[M^{\gamma} \to \nu\bullet, i, j, false]}{[\top \to \bullet\mathbf{R}^{\beta}, j, j, false]} \; \beta \in \text{radj}(M^{\gamma})$$

Compleción de adjunción derecha

Cuando se ha completado el reconocimiento de un árbol auxiliar derecho β que se puede adjuntar en un nodo M^{γ}, se actualizan las posiciones de la cadena de entrada reconocida por M^{γ} incluyendo la parte reconocida por β y se establece a $true$ el último parámetro para evitar otras adjunciones derechas en dicho nodo.

$$\mathcal{D}^{\text{RAdjComp}}_{\text{Earley}^{\text{i}}} = \frac{\begin{array}{c}[\top \to \mathbf{R}^{\beta}\bullet, j, k, false] \\ [M^{\gamma} \to \nu\bullet, i, j, false]\end{array}}{[M^{\gamma} \to \nu\bullet, i, k, true]} \; \beta \in \text{radj}(M^{\gamma})$$

Predicción de sustitución

Cuando el reconocimiento alcanza un nodo marcado para sustitución, el análisis debe lanzar el reconocimiento de todos los árboles iniciales que se pueden sustituir en él:

$$\mathcal{D}^{\text{SubsPred}}_{\text{Earley}^{\text{i}}} = \frac{[N^{\gamma} \to \delta \bullet M^{\gamma}\nu, i, j, false]}{[\top \to \bullet\mathbf{R}^{\alpha}, j, j, false]} \; \alpha \in \text{subst}(M^{\gamma})$$

Compleción de sustitución

Una vez completado el reconocimiento de un árbol inicial que se puede sustituir en un nodo, debemos continuar el reconocimiento del árbol donde se ha efectuado la sustitución.

$$\mathcal{D}^{\text{SubsComp}}_{\text{Earley}^{\text{i}}} = \frac{\begin{array}{c}[\top \to \mathbf{R}^{\alpha}\bullet, j, k, false] \\ [N^{\gamma} \to \delta \bullet M^{\gamma}\nu, i, j, false]\end{array}}{[N^{\gamma} \to \delta M^{\gamma} \bullet \nu, i, k, false]} \; \alpha \in \text{subst}(M^{\gamma})$$

El conjunto de ítems finales se define mediante:

$$\mathcal{F}_{\text{Earley}} = \{[\top \to \mathbf{R}^{\alpha}\bullet, 0, n, false] \mid \alpha \in \boldsymbol{I}, label(\mathbf{R}^{\alpha}) = S\}$$

Si no tenemos en cuenta el factor multiplicativo que introduce el parámetro $radj$, la complejidad espacial del analizador con respecto a la longitud n de la cadena de entrada es de $\mathcal{O}(n^2)$. La complejidad temporal del algoritmo es de $\mathcal{O}(n^3)$, y viene dada por la complejidad de los pasos deductivos que presentan el número máximo de variables de entrada: $\mathcal{D}^{\text{Comp}}_{\text{Earley}}$,$\mathcal{D}^{\text{RAdjComp}}_{\text{Earley}}$ y $\mathcal{D}^{\text{SubsComp}}_{\text{Earley}}$.

4. La relación left-corner en las TIGs

En la siguiente sección vamos a definir un esquema que usa una extensión del concepto de relación de esquina izquierda (left-corner,LC), conocido sobre las CFGs, para filtrar las predicciones en el analizador basado en el algoritmo de Earley para TIGs. La complejidad temporal de este algoritmos se mantiene en la cota de $\mathcal{O}(n^3)$, pero mejora sus prestaciones mediante una reducción en el número de predicciones. Antes de describir el nuevo analizador necesitamos definir el concepto de relación de esquina izquierda en las TIGs.

Definición 4.1 *Relación de esquina izquierda (LC) en los árboles elementales de TIGs*
La esquina izquierda de un nodo O^γ es su hijo izquierdo P^γ si y sólo si $ladj(P^\gamma) = \{\mathbf{nil}\}$.
La relación esquina izquierda $>_\ell$ en $V_N \times \{V_N \cup V_T \cup \{\varepsilon, \bot\}\}$ se define como
$O^\gamma >_\ell P^\gamma$ si hay una producción $O^\gamma \rightarrow P^\gamma \nu \in \mathcal{P}(\gamma)$ y $ladj(P^\gamma) = \{\mathbf{nil}\}$.
La clausura reflexiva y transitiva de $>_\ell$ la denotaremos como $>_\ell^*$.

En un abuso de notación, vamos a denotar como $P^\gamma >_\ell \triangle$ si hay una producción $O^\gamma \rightarrow P^\gamma \nu \in \mathcal{P}(\gamma)$ y existe un β tal que $\beta \in \mathrm{ladj}(P^\gamma)$.

La relación LC en las TIGs no va más allá de los límites de un árbol elemental. Es importante señalar que toda relación de LC en las TIGs comienza en un nodo etiquetado con un símbolo no terminal y finaliza en: (1) un nodo adjuntable por la izquierda; (2) un nodo etiquetado con un símbolo terminal , ε o $\bot$; (3) ó un nodo marcado para sustitución.

Las diferencias de esta definición respecto a la establecida para TAGs en [5]son fundamentalmente dos:

- Al introducir la operación de sustitución, en las fronteras de los árboles elementales de las TIGs pueden aparecer nodos etiquetados con símbolos no terminales marcados para sustitución. Ésto nos obliga a introducir este caso como una posibilidad de finalización en las relaciones LC.
- En las TIGs distinguimos dos tipos de adjunciones: izquierdas y derechas. Las segundas no afectan a las relaciones de esquina izquierda, puesto que sólo introducen contextos derechos en los subárboles. Sin embargo, las primeras sí rompen las relaciones LC. Por esta causa, únicamente tenemos en cuenta los nodos adjuntables por la izquierda como fin de una relación LC.

5. Esquema tipo left-corner con ítems predictivos

En este esquema, al que denominaremos $\mathbf{pLC^i}$, los pasos predictivos en el esquema $\mathbf{Earley^i}$ son reemplazados por objetivos que se intentan satisfacer de forma ascendente. La fase ascendente del proceso de reconocimiento es guiada hacia el correspondiente objetivo mediante la relación LC. Por tanto, en el dominio del esquema $\mathbf{pLC^i}$ vamos a distinguir dos tipos de ítems: predictivos y *left corner*. Por la propia naturaleza del analizador estos últimos se dividen en tres subtipos.

El conjunto de ítems de $\mathbf{pLC^i}$ se define mediante:

$$\mathcal{I}_{\mathrm{pLC^i}} = \mathcal{I}_{\mathrm{pLC^i}}^{(\mathrm{i})} \cup \mathcal{I}_{\mathrm{pLC^i}}^{(\mathrm{ii})} \cup \mathcal{I}_{\mathrm{pLC^i}}^{(\mathrm{iii})} \cup \mathcal{I}_{\mathrm{pLC^i}}^{(\mathrm{iv})}$$

Los *ítems predictivos* son aquellos que lanzan la predicción (de adjunción o subárbol) de un nodo M^γ desde una posición de la cadena de entrada, por tanto, sólo es necesaria la información del propio nodo y la posición de la cadena de entrada j:

$$\mathcal{I}_{\mathrm{pLC^i}}^{(\mathrm{i})} = \{[M^\gamma, j] \mid \gamma \in \boldsymbol{I} \cup \boldsymbol{A} \,,\, 0 \le j\}$$

Los *ítems left corner* tienen la misma forma que los ítems del analizador tipo Earley para las TAGs pero se le añade un nodo delante (C^γ), que es el nodo que domina por una relación de esquina izquierda al nodo situado a la izquierda de la producción (M^γ). Esta información adicional, aunque no es necesaria, va a permitir al analizador simplificar algunos de sus pasos deductivos y el reconocimiento ascendente a través de una secuencia de relaciones de esquinas izquierdas.

$$\mathcal{I}_{\mathrm{pLC^i}}^{(\mathrm{ii})} = \{[C^\gamma; M^\gamma \rightarrow P^\gamma \delta \bullet \nu, i, j, false] \mid C^\gamma >_\ell^* M^\gamma \,,\; M^\gamma \rightarrow P^\gamma \delta\nu \in \mathcal{P}(\gamma) \,,\, \gamma \in \boldsymbol{I} \cup \boldsymbol{A} \,,\, 0 \le i \le j \,,\, \nu \neq \varepsilon\}$$

$$\mathcal{I}_{\mathrm{pLC^i}}^{(\mathrm{iii})} = \{[C^\gamma; M^\gamma \rightarrow \nu \bullet, i, j, radj] \mid C^\gamma >_\ell^* M^\gamma \,,\; M^\gamma \rightarrow \nu \in \mathcal{P}(\gamma) \,,\, \gamma \in \boldsymbol{I} \cup \boldsymbol{A} \,,\, 0 \le i \le j \,,\; radj \in \{true, false\}\}$$

donde $radj = false$ si no se completó ninguna adjunción derecha sobre M^γ y $radj = true$ si se completó una adjunción derecha sobre M^γ.

$$\mathcal{I}_{\mathrm{pLC^i}}^{(\mathrm{iv})} = \{[C^\gamma; M^\gamma \rightarrow \bullet P^\gamma \nu, j, j, false] \mid C^\gamma >_\ell^* M^\gamma \,,\; M^\gamma \rightarrow P^\gamma \nu \in \mathcal{P}(\gamma) \,,\, \gamma \in \boldsymbol{I} \cup \boldsymbol{A} \,,\, 0 \le i \le j\}$$

donde $P^\gamma >_\ell \triangle$ ó existe un α tal que $\alpha \in \mathrm{subst}(P^\gamma)$. Es decir, el punto sólo aparece al comienzo de la regla cuando el hijo izquierdo sea un nodo adjuntable por la izquierda o un nodo de sustitución, con objeto de lanzar el reconocimiento del árbol auxiliar izquierdo ó del árbol inicial, respectivamente.

Con respecto al conjunto de pasos deductivos, definimos subconjuntos para *reconocimiento* y *compleción* similares a los del esquema $\mathbf{Earley^i}$ para TIGs. La relación de esquina izquierda se aplicará a cinco casos de predicción: inicial, subárbol, pie, adjunción izquierda y adjunción derecha. Obsérvese que aparece un filtro sobre la predicciones del pie, aunque este tipo de predicción aparentemente no aparece en el esquema $\mathbf{Earley^i}$. Sin embargo, si nos fijamos con atención el paso de compleción de adjunción izquierda $\mathcal{D}_{\mathrm{Earley^i}}^{\mathrm{LAdjComp}}$ hace una doble función: completa la adjunción izquierda e inicia el reconocimiento del subárbol escindido. Por ello podemos aplicar un filtro sobre esta predicción de pie.

Los pasos de esquina izquierda vienen en tres variedades, según el tipo de la etiqueta del nodo en que finalice la relación: terminal, cadena vacía ó nodo *bottom*, y no terminal. Los nodos etiquetados con ε y $\bot$ se incluyen en el mismo caso porque los nodos $\bot$ en las TIGs se comportan igual que una cadena vacía, debido que no subsumen nada. El último caso es necesario cuando la esquina izquierda es un nodo de adjunción izquierda o está marcado para sustitución.

Los pasos deductivos del esquema son:

$$\mathcal{D}_{\mathrm{pLC^i}} = \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LI_t}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LI_\varepsilon}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LI_{pre}}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{Scan}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\varepsilon} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LC_t}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LC_\varepsilon}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LC_{pre}}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LC_n}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{Pre}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{Comp}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LA_t}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LA_\varepsilon}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LA_{pre}}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LF_t}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LF_\varepsilon}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LF_{pre}}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{RA_\varepsilon}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{RAdjComp}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LS_t}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LS_\varepsilon}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LS_{pre}}} \cup \mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{SubsComp}}$$

Filtrado de inicio

$$\mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LI_t}} = \frac{[a, 0, 1]}{[\top; O^\alpha \rightarrow P^\alpha \bullet \nu, 0, 1, false]} \quad \begin{array}{l} \alpha \in \boldsymbol{I} \\ label(\mathbf{R}^\alpha) = S \\ label(P^\alpha) = a \end{array}$$

$$\mathcal{D}_{\mathrm{pLC^i}}^{\mathrm{LI_\varepsilon}} = \frac{}{[\top; O^\alpha \rightarrow P^\alpha \bullet \nu, 0, 0, false]} \quad \begin{array}{l} \alpha \in \boldsymbol{I} \\ label(\mathbf{R}^\alpha) = S \\ label(P^\alpha) = \varepsilon \end{array}$$

$$\mathcal{D}^{\mathrm{LI_{pre}}}_{\mathrm{pLC^i}} = \frac{}{[\top; O^\alpha \to \bullet P^\alpha \nu, 0, 0, \mathit{false}]} \quad \begin{array}{l} \alpha \in \boldsymbol{I} \\ \mathit{label}(\mathbf{R}^\alpha) = S \\ (P^\alpha >_\ell \triangle \text{ ó} \\ \alpha' \in \mathrm{subst}(P^\alpha)) \end{array}$$

En el paso $\mathcal{D}^{\mathrm{LI_\varepsilon}}_{\mathrm{pLC^i}}$ no incluimos la condición $\mathit{label}(P^\alpha) = \perp$ porque este paso se aplica exclusivamente a árboles iniciales.

Reconocimiento

Los pasos de reconocimiento son similares a los del esquema $\mathbf{Earley^i}$ pero sólo se aplican a nodos que no son hijos izquierdos.

$$\mathcal{D}^{\mathrm{Scan}}_{\mathrm{pLC^i}} = \frac{\begin{array}{l}[a, j, j+1] \\ [C^\gamma; N^\gamma \to P^\gamma \delta \bullet M^\gamma \nu, i, j, \mathit{false}]\end{array}}{[C^\gamma; N^\gamma \to P^\gamma \delta M^\gamma \bullet \nu, i, j+1, \mathit{false}]}$$

donde $\mathit{label}(M^\gamma) = a$.

$$\mathcal{D}^{\varepsilon}_{\mathrm{pLC^i}} = \frac{[C^\gamma; N^\gamma \to P^\gamma \delta \bullet M^\gamma \nu, i, j, \mathit{false}]}{[C^\gamma; N^\gamma \to P^\gamma \delta M^\gamma \bullet \nu, i, j, \mathit{false}]}$$

donde $\mathit{label}(M^\gamma) = \varepsilon$ ó $\mathit{label}(M^\gamma) = \perp$.

Filtrado de predicción de subárbol

$$\mathcal{D}^{\mathrm{LC_t}}_{\mathrm{pLC^i}} = \frac{\begin{array}{c}[M^\gamma, j] \\ [a, j, j+1]\end{array}}{[M^\gamma; O^\gamma \to P^\gamma \bullet \nu, j, j+1, \mathit{false}]}$$

donde $\mathbf{nil} \in \mathrm{ladj}(M^\gamma)$ y $\mathit{label}(P^\gamma) = a$.

$$\mathcal{D}^{\mathrm{LC_\varepsilon}}_{\mathrm{pLC^i}} = \frac{[M^\gamma, j]}{[M^\gamma; O^\gamma \to P^\gamma \bullet \nu, j, j, \mathit{false}]} \quad \begin{array}{l} \mathbf{nil} \in \mathrm{ladj}(M^\gamma) \\ (\mathit{label}(P^\gamma) = \varepsilon \text{ ó} \\ \mathit{label}(P^\gamma) = \perp) \end{array}$$

$$\mathcal{D}^{\mathrm{LC_{pre}}}_{\mathrm{pLC^i}} = \frac{[M^\gamma, j]}{[M^\gamma; O^\gamma \to \bullet P^\gamma \nu, j, j, \mathit{false}]} \quad \begin{array}{l} \mathbf{nil} \in \mathrm{ladj}(M^\gamma) \\ (P^\gamma >_\ell \triangle \text{ ó} \\ \exists \alpha \in \mathrm{subst}(P^\gamma)) \end{array}$$

Compleción en la esquina izquierda

El paso $\mathcal{D}^{\mathrm{LC_n}}_{\mathrm{pLC^i}}$ es el que lleva a cabo las compleciones en los nodos que han sido filtrados por una relación LC.

$$\mathcal{D}^{\mathrm{LC_n}}_{\mathrm{pLC^i}} = \frac{[M^\gamma; O^\gamma \to \nu \bullet, j, k, \mathit{radj}]}{[M^\gamma; Q^\gamma \to O^\gamma \bullet \omega, j, k, \mathit{false}]} \quad M^\gamma \neq O^\gamma$$

donde se debe cumplir que si $\mathit{radj} = \mathit{false}$ entonces $\mathbf{nil} \in \mathrm{radj}(O^\gamma)$.

Predicción

Este paso realiza la predicción de todos los nodos que dominan relaciones LC, excepto los nodos raíces de los árboles elementales.

$$\mathcal{D}^{\mathrm{Pre}}_{\mathrm{pLC^i}} = \frac{[C^\gamma; N^\gamma \to \delta \bullet M^\gamma \nu, i, j, \mathit{false}]}{[M^\gamma, j]} \quad \mathit{label}(M^\gamma) \in V_N$$

Compleción de subárbol

$$\mathcal{D}^{\mathrm{Comp}}_{\mathrm{pLC^i}} = \frac{\begin{array}{l}[M^\gamma; M^\gamma \to \nu \bullet, j, k, \mathit{radj}] \\ [C^\gamma; N^\gamma \to \delta \bullet M^\gamma \nu, i, j, \mathit{false}]\end{array}}{[C^\gamma; N^\gamma \to \delta M^\gamma \bullet \nu, i, k, \mathit{false}]}$$

donde se debe cumplir que si $\mathit{radj} = \mathit{false}$ entonces $\mathbf{nil} \in \mathrm{radj}(O^\gamma)$.

Filtrado de predicción de adjunción izquierda

$$\mathcal{D}^{\mathrm{LA_t}}_{\mathrm{pLC^i}} = \frac{\begin{array}{c}[M^\gamma, j] \\ [a, j, j+1]\end{array}}{[\top; O^\beta \to P^\beta \bullet \nu, j, j+1, \mathit{false}]} \quad \begin{array}{l} \beta \in \mathrm{ladj}(M^\gamma) \\ \mathit{label}(P^\beta) = a \end{array}$$

$$\mathcal{D}^{\mathrm{LA_\varepsilon}}_{\mathrm{pLC^i}} = \frac{[M^\gamma, j]}{[\top; O^\beta \to P^\beta \bullet \nu, j, j, \mathit{false}]} \quad \begin{array}{l} \beta \in \mathrm{ladj}(M^\gamma) \\ \mathit{label}(P^\beta) = \varepsilon \end{array}$$

$$\mathcal{D}^{\mathrm{LA_{pre}}}_{\mathrm{pLC^i}} = \frac{[M^\gamma, j]}{[\top; O^\beta \to \bullet P^\beta \nu, j, j, \mathit{false}]} \quad \begin{array}{l} \beta \in \mathrm{ladj}(M^\gamma) \\ (P^\beta >_\ell \triangle \text{ ó} \\ \exists \alpha \in \mathrm{subst}(P^\gamma)) \end{array}$$

En el paso $\mathcal{D}^{\mathrm{LA_\varepsilon}}_{\mathrm{pLC^i}}$ no incluimos la condición $\mathit{label}(P^\beta) = \perp$ porque en un árbol auxiliar izquierdo no se puede dar el caso $\top >^*_\ell \perp$.

Filtrado de predicción de pie

Este conjunto de pasos son el resultado de aplicar un filtro LC al paso $\mathcal{D}^{\mathrm{LAdjComp}}_{\mathrm{Earley^i}}$, el cual, como dijimos antes, también cumple una función de predicción de pie.

$$\mathcal{D}^{\mathrm{LF_t}}_{\mathrm{pLC^i}} = \frac{\begin{array}{l}[a, k, k+1] \\ [\top; \top \to \mathbf{R}^\beta \bullet, j, k, \mathit{false}] \\ [M^\gamma, j]\end{array}}{[M^\gamma; O^\gamma \to P^\gamma \bullet \nu, k, k+1, \mathit{false}]}$$

donde $\beta \in \mathrm{ladj}(M^\gamma)$ y $\mathit{label}(P^\gamma) = a$.

$$\mathcal{D}^{\mathrm{LF_\varepsilon}}_{\mathrm{pLC^i}} = \frac{\begin{array}{l}[\top; \top \to \mathbf{R}^\beta \bullet, j, k, \mathit{false}] \\ [M^\gamma, j]\end{array}}{[M^\gamma; O^\gamma \to P^\gamma \bullet \nu, k, k, \mathit{false}]}$$

donde $\beta \in \mathrm{ladj}(M^\gamma)$ y $(\mathit{label}(P^\gamma) = \varepsilon$ ó $\mathit{label}(P^\gamma) = \perp)$.

$$\mathcal{D}^{\mathrm{LF_{pre}}}_{\mathrm{pLC^i}} = \frac{\begin{array}{l}[\top; \top \to \mathbf{R}^\beta \bullet, j, k, \mathit{false}] \\ [M^\gamma, j]\end{array}}{[M^\gamma; O^\gamma \to P^\gamma \bullet \nu, k, k, \mathit{false}]}$$

donde $\beta \in \mathrm{ladj}(M^\gamma)$ y $(P^\gamma >_\ell \triangle$ ó $\exists \alpha \in \mathrm{subst}(P^\gamma))$.

Filtrado de predicción de adjunción derecha

Teniendo en cuenta que los árboles auxiliares derechos se caracterizan porque a la izquierda de la espina no se pueden adjuntar árboles auxiliares izquierdos y en esa parte de su frontera sólo pueden aparecer nodos etiquetados con ε, ésto reduce la finalización del filtro LC sobre la raíces de los árboles auxiliares derechos a dos casos: ε ó $\perp$.

$$\mathcal{D}^{\mathrm{RA_\varepsilon}}_{\mathrm{pLC^i}} = \frac{[C^\gamma; M^\gamma \to \delta \bullet, k, l, \mathit{false}]}{[\top; O^\beta \to P^\beta \bullet \nu, l, l, \mathit{false}]} \quad \begin{array}{l} \beta \in \mathrm{radj}(M^\gamma) \\ (\mathit{label}(P^\beta) = \varepsilon \text{ ó} \\ \mathit{label}(P^\beta) = \perp) \end{array}$$

Compleción de adjunción derecha

$$\mathcal{D}^{\mathrm{RAdjComp}}_{\mathrm{pLC^i}} = \frac{\begin{array}{l}[\top; \top \to \mathbf{R}^\beta \bullet, j, k, \mathit{false}] \\ [C^\gamma; M^\gamma \to \nu \bullet, i, j, \mathit{false}]\end{array}}{[C^\gamma; M^\gamma \to \nu \bullet, i, k, \mathit{true}]}$$

donde $\beta \in \mathrm{radj}(M^\gamma)$.

Filtrado de predicción de sustitución

$$\mathcal{D}^{\mathrm{LS_t}}_{\mathrm{pLC^i}} = \frac{\begin{array}{c}[M^\gamma, j] \\ [a, j, j+1]\end{array}}{[\top; O^\alpha \to P^\alpha \bullet \nu, j, j+1, false]}$$

donde $\alpha \in \mathrm{subst}(M^\gamma)$ y $label(P^\alpha) = a$.

$$\mathcal{D}^{\mathrm{LS_\varepsilon}}_{\mathrm{pLC^i}} = \frac{[M^\gamma, j]}{[\top; O^\alpha \to P^\alpha \bullet \nu, j, j, false]} \quad \begin{array}{l}\alpha \in \mathrm{subst}(M^\gamma) \\ label(P^\alpha) = \varepsilon\end{array}$$

$$\mathcal{D}^{\mathrm{LS_{pre}}}_{\mathrm{pLC^i}} = \frac{[M^\gamma, j]}{[\top; O^\alpha \to \bullet P^\alpha \nu, j, j, false]} \quad \begin{array}{l}\alpha \in \mathrm{subst}(M^\gamma) \\ P^\alpha >_\ell \triangle\end{array}$$

Compleción de sustitución

$$\mathcal{D}^{\mathrm{SubsComp}}_{\mathrm{pLC^i}} = \frac{\begin{array}{c}[\top; \top \to \mathbf{R}^\alpha \bullet, j, k, false] \\ [C^\gamma; N^\gamma \to \delta \bullet M^\gamma \nu, i, j, false]\end{array}}{[C^\gamma; N^\gamma \to \delta M^\gamma \bullet \nu, i, k, false]}$$

donde $\alpha \in \mathrm{subst}(M^\gamma)$.

El conjunto de ítems finales del esquema se define mediante:

$$\mathcal{F}_{\mathrm{pLC^i}} = \{[\top; \top \to \mathbf{R}^\alpha \bullet, 0, n, false] \mid \alpha \in \boldsymbol{I}, label(\mathbf{R}^\alpha) = S\}$$

La complejidad espacial del analizador con respecto a la longitud n de la cadena de entrada es de $O(n^2)$. La complejidad temporal del algoritmo es de $O(n^3)$, y viene dada por la complejidad de los pasos deductivos: $\mathcal{D}^{\mathrm{Comp}}_{\mathrm{pLC^i}}$,$\mathcal{D}^{\mathrm{RAdjComp}}_{\mathrm{pLC^i}}$ y $\mathcal{D}^{\mathrm{SubsComp}}_{\mathrm{pLC^i}}$.

6. Conclusiones

Hemos definido dos nuevos algoritmos para análisis sintáctico de TIG, uno de tipo ascendente predictivo al estilo de earley y otro de tipo left corner. El analizador tipo earley es una variante mejorada del presentado en [12]. Extendemos el concepto de relación left corner existente en CFG a los árboles elementales de TIG. Y por último, aplicamos dicha relación para construir un nuevo analizador de tipo LC. Las cotas de complejidad teórica en el caso peor del analizador LC son iguales a las del analizador tipo earley, pero las prestaciones reales mejoran significativamente debido a la reducción del número de predicciones.

7. Agradecimientos

El trabajo descrito en este artículo ha sido parcialmente financiado por el Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica (TIC2000-0370-C02-01), Ministerio de Ciencia y Tecnología (HP2001-0044, FIT-150500-2002-416) y Xunta de Galicia (PGIDT01PXI10506PN).

8. Referencias

[1] Joshi, A. K. y Schabes, Y., "Tree adjoining grammars", En G. Rozenberg y A. Salomaa, editores, *Handbook of Formal Languages. Vol 3: Beyond Words*. Springer-Verlag, páginas 69–123, 1997.

[2] Evans, R. y Weir, D., .A structure-sharing parser for lexicalized grammars", En *Proc. of COLING-ACL'98*, volumen I, páginas 372–378, Montreal, Canadá, 1998.

[3] Carrillo, V., Díaz, V.J. y Alonso, M.A., .Análisis sintáctico ascendente de TAGs guiado por la esquina izquierda", *Procesamiento del Lenguaje Natural*, 27:47–54, 2001.

[4] Carrillo, V., Díaz, V.J. y Alonso, M.A., .A predictive left-corner parser for tree adjoining grammars", *Revista electrónica de Tecnología del Habla*, por aparecer, 2002.

[5] Díaz, V., Carrillo, V. y Alonso, M.A., .A left-corner parser for tree adjoining grammars", En *Proc. of TAG+6*, Venecia, Italia, 2002.

[6] Schabes, Y. y Waters, R.C., "Tree insertion grammar: A cubic-time parsable formalism that lexicalizes context-free grammar without changing the trees produced", *Computational Linguistics*, 21(4):479–513, 1995.

[7] Doran, C., Egedi, D., Hockey, B., Srinivas, B. y Zaidel, M., "XTAG system — a wide coverage grammar for English", En *Proc. of COLING'94*, páginas 922–928, Kyoto, Japón, 1994.

[8] Alonso, M. A., Cabrero, D. , de la Clergerie, E. y Vilares, M., "Tabular algorithms for TAG parsing", En *Proc. of EACL'99*, páginas 150–157, Bergen, Noruega, 1999.

[9] Díaz, V., Carrillo, V. y Alonso, M.A., .A bidirectional bottom-up parser for TAG", En *Proc. of International Workshop in Parsing Technologies 2000*, páginas 299–300, Trento, Italia, 2000.

[10] van Noord, G., "Head corner parsing for TAG", *Computational Intelligence*, 10(4):525–534, 1994.

[11] Nederhof, M., "The computational complexity of the correct-prefix property for TAGs", *Computational Linguistics*, 25(3):345–360, 1999.

[12] Carrillo, V., Díaz, V.J. y Alonso, M.A., .Algoritmos de análisis sintáctico para gramáticas de inserción de árboles", *Procesamiento del Lenguaje Natural*, 29:89–96, 2002.

[13] Schabes, Y. y Waters, R.C., "Sthocastic lexicalized tree insertion grammar", En H. Bunt y M. Tomita, editores, *Recent advances in parsing technologies*. Kluwer Academic Publishers, páginas 281–294, 1996.

[14] Shieber, S. M., Schabes, Y. y Pereira, F.C., "Principles and implementation of deductive parsing", *Journal of Logic Programming*, 24(1–2):3–36, 1995.

[15] Sikkel, K., *Parsing Schemata — A Framework for Specification and Analysis of Parsing Algorithms*. Springer-Verlag, 1997.