Análisis sintáctico ascendente de TAGs guiado por la esquina izquierda

Vicente Carrillo y **Víctor J. Díaz**
Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla
{carrillo,vjdiaz}@lsi.us.es

Miguel A. Alonso
Departamento de Computación. Universidad de La Coruña
alonso@dc.fi.udc.es

Resumen Definimos un nuevo analizador para Gramáticas de Adjunción de Árboles (TAGs, Tree Adjoining Grammars) que es una extensión del analizador ascendente guiado por la esquina izquierda para Gramáticas Incontextuales (CFGs, Context Free Grammars). La complejidad temporal teórica del nuevo analizador permanece en la cota del análisis de TAGs, siendo ésta de $O(n^6)$ en el peor de los casos, donde n es la longitud de la cadena de entrada. Sin embargo, mostraremos que el nuevo analizador aumenta las prestaciones en casos prácticos, reduciendo de manera significativa el número de ítems deducidos respecto a un analizador ascendente sin ningún tipo de filtro.

1 Introducción

Las Gramáticas de Adjunción de Árboles (TAGs) [1] constituyen un formalismo gramatical que utiliza árboles como elementos de composición básicos, frente a las producciones usadas en formalismos como las gramáticas incontextuales (CFG). Asimismo utiliza como operación de composición la operación de adjunción, la cual permite una potencia expresiva superior a la de las gramáticas incontextuales. La importancia de las gramáticas de adjunción de árboles viene dada porque todas sus estructuras se encuentran lexicalizadas de manera natural y aporta un dominio de localidad extendido, con los beneficios tanto lingüísticos como computacionales que estas características conllevan.

Los analizadores para TAGs que se encuentran en la literatura habitualmente son adaptaciones de analizadores estudiados para gramáticas incontextuales. En concreto, y por la relevancia que tiene para este trabajo, podemos citar los analizadores ascendentes basados en **CYK** y **buE** (*bottom-up Earley*) para TAGs que se describen en [4]. En este artículo presentamos un analizador que utiliza una estrategia ascendente guiada por la esquina izquierda para TAGs como una adaptación para este formalismo del analizador **buLC**[1] (*bottom-up Left Corner*) para CFGs descrito en [2]. Para especificar los analizadores usaremos los *esquemas de análisis sintáctico* [2].

En esta sección introduciremos los conceptos básicos tanto de las gramáticas de adjunción de árboles como de los *esquemas de análisis* necesarios para la comprensión de los algoritmos descritos en el resto del artículo. También mostraremos el esquema **buLC** para gramáticas incontextuales, indicando cómo soluciona ciertas redundancias que presenta un esquema estrictamente ascendente como el **buE**. En la sección 2 se describe el esquema **buE** para TAGs. En la sección 3 analizamos las posibles mejoras que se pueden introducir a **buE**, y cómo desde éstas se deriva el nuevo esquema **buLC**. En la sección 4 haremos una comparativa, en base a resultados empíricos, entre los dos analizadores unidireccionales con estrategia ascendente para TAGs citados: **buE** y el propuesto en este trabajo, **buLC**. No consideraremos el analizador **CYK** porque establece restricciones importantes en cuanto a la forma que deben tener los árboles elementales de la gramática.

1.1 Las Gramáticas de Adjunción de Árboles

Formalmente una TAG es una quíntupla $(V_N, V_T, S, \boldsymbol{I}, \boldsymbol{A})$, donde V_N es el conjunto finito de símbolos no terminales, V_T es el conjunto finito de símbolos terminales, $S \in V_N$ es

[1] Utilizaremos el subrayado para indicar que un determinado esquema está definido para CFGs y permitir de este modo distinguirlo del esquema del mismo nombre denifido para TAGs.

el axioma de la gramática, $\boldsymbol{I} \cup \boldsymbol{A}$ es un conjunto finito de árboles finitos denominados *árboles elementales*. A los árboles del conjunto $\boldsymbol{I}$ se les denomina *árboles iniciales*, y se caracterizan porque todos sus nodos interiores están etiquetados con símbolos de V_N, mientras los nodos de su frontera se etiquetan con símbolos de V_T o la cadena vacía ϵ. A los árboles del conjunto $\boldsymbol{A}$ se les denomina *árboles auxiliares*, y se caracterizan porque todos sus nodos interiores están etiquetados con símbolos de V_N, mientras los nodos de su frontera se etiquetan con símbolos de V_T o la cadena vacía ϵ, salvo uno, denominado *pie*, que está etiquetado con el mismo símbolo que la raíz del árbol. El camino que va desde la raíz hasta el pie se denomina *espina*.

Vamos a denotar con M^γ un nodo interior perteneciente a un árbol elemental γ. Nos referiremos a la raíz de un árbol elemental γ como $\mathbf{R}^\gamma$ y al pie de un árbol auxiliar β como $\mathbf{F}^\beta$. Los otros nodos frontera los denotaremos con sus etiquetas.

A diferencia de las gramáticas incontextuales, en las cuales se usa la *sustitución* de reglas como operación de composición, en las TAGs la composición de estructuras más complejas se lleva a cabo mediante la operación de *adjunción*. Esta operación, que dota a las TAGs de una potencia expresiva superior a las CFGs, consiste en lo siguiente: dado un nodo M^γ etiquetado con el mismo símbolo que la raíz de un árbol auxiliar $\mathbf{R}^\beta$, la adjunción de β en M^γ escinde el subárbol que pende de M^γ, pega el árbol auxiliar β en M^γ y, por último, pega el subárbol escindido en el nodo pie $\mathbf{F}^\beta$. Denotaremos mediante $\beta \in Adj(M^\gamma)$ que el árbol auxiliar β pueda ser adjuntado en el nodo M^γ. Si un nodo no tiene adjunción obligatoria entonces $\mathbf{nil} \in Adj(M^\gamma)$, donde **nil** es un símbolo vacío que no pertenece al conjunto de árboles auxiliares.

Para usar *esquemas de análisis* como método de especificación es habitual representar mediante reglas el reconocimiento parcial de los árboles elementales. Por tanto, es necesario traducir cada árbol elemental γ en un conjunto de producciones $\mathcal{P}(\gamma)$ de la siguiente manera:

$$\mathcal{P}(\gamma) = \{N^\gamma \rightarrow N_1^\gamma ... N_g^\gamma\}$$

donde N^γ es un nodo interior de γ y $N_1^\gamma ... N_g^\gamma$ es el conjunto ordenado de sus nodo hijos.

Por razones técnicas, y siguiendo el enfoque de [3], vamos a introducir dos reglas adicionales: (1) $\top \rightarrow \mathbf{R}^\gamma$ para cada árbol elemental γ y, (2) $\mathbf{F}^\gamma \rightarrow \bot$ para cada árbol auxiliar β. Los nuevos nodos $\top$ y $\bot$ presentan una restricción de adjunción nula con objeto de no modificar la capacidad generativa de la gramática.

1.2 Esquemas de análisis sintáctico

Los *esquemas de análisis sintáctico* [2] constituyen un método general para la especificación de algoritmos de análisis sintáctico que aporta como ventajas fundamentales:

- Definición de los analizadores sin tener en cuenta las estructuras de datos y de control que se usarán en su implementación.
- Permite establecer de una manera fácil las relaciones entre distintos algoritmos mediante el análisis de ciertas relaciones formales.

Definición 1 *Sistema de análisis*
Un sistema de análisis $\mathbb{P}$ *para una gramática* G *y una cadena de entrada* $a_1 ... a_n$ *es una tripleta* $< \mathcal{I}, \mathcal{H}, \mathcal{D} >$*, donde:*

- $\mathcal{I}$ *es un conjunto de ítems, denominado dominio;*
- $\mathcal{H}$ *es un conjunto finito de ítems, llamados hipótesis.* $\mathcal{H}$ *no tiene que ser un subconjunto de* $\mathcal{I}$*;*
- $\mathcal{D} \subseteq \wp(\mathcal{H} \cup \mathcal{I}) \times \mathcal{I}$ *es un conjunto de pasos deductivos. Con* $\wp$ *denotamos el conjunto potencia de conjuntos finitos.*

La notación que vamos a usar para especificar los pasos deductivos, mediante los cuales se derivan nuevos ítems ξ a partir de los ítems η_i existentes, es $\frac{\eta_1,...,\eta_k}{\xi}$ *cond*. El analizador sintáctico añadirá el consecuente $\xi \in \mathcal{I}$ si todos los antecedentes del paso deductivo $\eta_i \in \mathcal{H} \cup \mathcal{I}$ existen y la condición *cond* se cumple.

Definición 2 *Ítems válidos*
El conjunto de ítems válidos para un sistema de análisis $\mathbb{P} =< \mathcal{I}, \mathcal{H}, \mathcal{D} >$ *se define como*

$$\mathcal{V}_{\mathbb{P}} = \{\xi \in \mathcal{I} \mid \mathcal{H} \vdash^* \xi\}$$

donde $\vdash^*$ *es una secuencia de pasos deductivos.*

Definición 3 *Sistema de análisis no instanciado*
Un sistema de análisis no instanciado para una gramática G es una tripleta $< \mathcal{I}, \mathcal{H}, \mathcal{D} >$, donde $\mathcal{H}$ es una función que asigna un conjunto de hipótesis a cada cadena de entrada $a_1...a_n$, tal que $< \mathcal{I}, \mathcal{H}(a_1...a_n), \mathcal{D} >$ es un sistema de análisis.

La función $\mathcal{H}$ que se usará en este trabajo es: $\mathcal{H}(a_1...a_n) = \{[a, i-1, i] \mid a = a_i \wedge 1 \leq i \leq n\}$

Definición 4 *Esquema de análisis*
Un esquema de análisis para una clase de gramáticas es una función que asigna un sistema de análisis no instanciado a cada gramática de dicha clase.

Los esquemas de análisis permiten definir variantes, extensiones y optimizaciones de analizadores. Se pueden obtener mejoras cualitativas mediante *generalizaciones* y optimizaciones cuantitativas mediante el *filtrado*, como se detalla en [2].

1.3 El esquema buLC para gramáticas incontextuales

Vamos a mostrar el esquema del analizador ascendente guiado por la esquina izquierda para CFGs que nos servirá como base para definir el nuevo esquema **buLC** para TAGs que se describe en la sección 3.

El analizador **buLC** para CFGs es un filtro del analizador **buE**[2] [2] que elimina ítems que no juegan un papel significativo en el reconocimiento ascendente. Concretamente, los ítems de la forma $[A \rightarrow \bullet B\nu, i, i]$, donde B es un símbolo terminal o no terminal, son introducidos por el paso que inicia el reconocimiento ascendente. Estos ítems sólo pueden aparecer como antecedentes en los pasos deductivos de lectura de terminal y compleción y no aportan nada significativo. Son sólo válidos por definición, por tanto, se pueden eliminar.

Definición 5 *Esquina izquierda*
La esquina izquierda de una producción es el símbolo más a la izquierda del lado derecho de la producción. $A \rightarrow B\nu$ tiene como esquina izquierda a B; una producción vacía tiene como esquina izquierda ϵ.

[2] **buE** es una versión ascendente del método de Earley.

Basándonos en el razonamiento y la definición anteriores, se define un nuevo analizador **buLC** que sustituye el paso que inicia el reconocimiento ascendente en **buE** por los tres pasos deductivos $\mathcal{D}^{\mathrm{LC}_\epsilon}_{\underline{\mathrm{buLC}}}$, $\mathcal{D}^{\mathrm{LC}_\mathrm{n}}_{\underline{\mathrm{buLC}}}$ y $\mathcal{D}^{\mathrm{LC}_\mathrm{t}}_{\underline{\mathrm{buLC}}}$, con objeto de eliminar los ítems de la forma descrita anteriormente.

Al desaparecer los ítems con el punto a la izquierda de la esquina izquierda de la producción, el dominio del esquema buLC para CFG viene dado por:

$$\mathcal{I}_{\underline{\mathrm{buLC}}} = \mathcal{I}^{\epsilon}_{\underline{\mathrm{buLC}}} \cup \mathcal{I}^{(\mathrm{i})}_{\underline{\mathrm{buLC}}}$$

$$\mathcal{I}^{\epsilon}_{\underline{\mathrm{buLC}}} = \{[A \rightarrow \bullet, j, j]\}$$

donde $A \rightarrow \epsilon$ es una regla de producción de la gramática y $j \geq 0$ es un índice en la cadena de entrada.

$$\mathcal{I}^{(\mathrm{i})}_{\underline{\mathrm{buLC}}} = \{[A \rightarrow B\nu \bullet \omega, i, j]\}$$

donde $A \rightarrow B\nu\omega$ es una regla de producción de la gramática, $B \in V_T \cup V_N$ y $0 \leq i \leq j$.

Los pasos deductivos del esquema son:

$$\mathcal{D}_{\underline{\mathrm{buLC}}} = \mathcal{D}^{\mathrm{LC}_\mathrm{t}}_{\underline{\mathrm{buLC}}} \cup \mathcal{D}^{\mathrm{LC}_\epsilon}_{\underline{\mathrm{buLC}}} \cup \mathcal{D}^{\mathrm{LC}_\mathrm{n}}_{\underline{\mathrm{buLC}}} \cup \mathcal{D}^{\mathrm{Sc}}_{\underline{\mathrm{buLC}}} \cup \mathcal{D}^{\mathrm{Cmp}}_{\underline{\mathrm{buLC}}}$$

El paso $\mathcal{D}^{\mathrm{LC}_\mathrm{t}}_{\underline{\mathrm{buLC}}}$ lanza el reconocimiento de las producciones cuya esquina izquierda sea un símbolo terminal, entre todas las posiciones de la cadena de entrada donde aparece dicho símbolo.

$$\mathcal{D}^{\mathrm{LC}_\mathrm{t}}_{\underline{\mathrm{buLC}}} = \frac{[a, j, j+1]}{[A \rightarrow a \bullet \omega, j, j+1]}$$

El paso $\mathcal{D}^{\mathrm{LC}_\epsilon}_{\underline{\mathrm{buLC}}}$ comienza el reconocimiento de las producciones que derivan la cadena vacía.

$$\mathcal{D}^{\mathrm{LC}_\epsilon}_{\underline{\mathrm{buLC}}} = \frac{}{[A \rightarrow \bullet, j, j]}$$

El paso $\mathcal{D}^{\mathrm{LC}_\mathrm{n}}_{\underline{\mathrm{buLC}}}$ inicia el reconocimiento de las producciones que cumplen que: (1) la esquina izquierda sea un símbolo no terminal y, (2) la producción que domina dicho símbolo haya sido completamente reconocida. Es decir, continúa el reconocimiento de la producción una vez reconocido el subárbol dominado por la esquina izquierda.

$$\mathcal{D}^{\mathrm{LC}_\mathrm{n}}_{\underline{\mathrm{buLC}}} = \frac{[A \rightarrow \nu \bullet, j, k]}{[B \rightarrow A \bullet \omega, j, k]}$$

El paso deductivo $\mathcal{D}^{\text{Sc}}_{\underline{\text{buLC}}}$ realiza la lectura del símbolo actual si coincide con el símbolo situado entre las posiciones j y $j+1$ de la cadena de entrada. A diferencia del paso $\mathcal{D}^{\text{LC}_\text{t}}_{\underline{\text{buLC}}}$, el símbolo actual no puede ser esquina izquierda en la producción.

$$\mathcal{D}^{\text{Sc}}_{\underline{\text{buLC}}} = \frac{[a, j, j+1] \quad [A \to B\nu \bullet a\omega, i, j]}{[A \to B\nu a \bullet \omega, i, j+1]}$$

El paso $\mathcal{D}^{\text{Cmp}}_{\underline{\text{buLC}}}$ continúa el reconocimiento de una producción una vez que que el subárbol dominado por un símbolo no terminal haya sido completamente reconocido. La diferencia con el paso $\mathcal{D}^{\text{LC}_\text{n}}_{\underline{\text{buLC}}}$ radica en que el símbolo no terminal no puede ser la esquina izquierda de la producción.

$$\mathcal{D}^{\text{Cmp}}_{\underline{\text{buLC}}} = \frac{[B \to \delta\bullet, j, k] \quad [A \to C\nu \bullet B\omega, i, j]}{[A \to C\nu B \bullet \omega, i, k]}$$

2 *Análisis ascendente de TAGs. El esquema* buE

El esquema **buE**, presentado en [4], se obtiene a partir de una generalización del esquema **CYK** para TAGs. El interés de este esquema radica en que se trata de un reconocedor con estrategia ascendente que elimina la restricción impuesta por el esquema **CYK** sobre la forma que deben tener los árboles elementales. Como se muestra en los resultados experimentales obtenidos en [7], el comportamiento de **buE** es, en general, peor que otros analizadores que usan estrategias ascendentes con algún tipo de filtro.

El dominio del esquema $\mathcal{I}_{\text{buE}}$ se define mediante:

$$[N^\gamma \to \nu \bullet \omega, i, j, p, q]$$

donde $N^\gamma \to \nu\omega$ es una producción en $\mathcal{P}(\gamma)$, siendo γ un árbol elemental. Los índices $0 \leq i \leq j$ establecen las posiciones dentro de la cadena de entrada que delimitan el fragmento reconocido por ν. Si p y q presentan un valor conocido, entonces γ es un árbol auxiliar y se cumple $i \leq p \leq q \leq j$.

Los pasos deductivos del esquema vienen dados por:

$$\mathcal{D}_{\text{buE}} = \mathcal{D}^{\text{Ini}}_{\text{buE}} \cup \mathcal{D}^{\text{Foot}}_{\text{buE}} \cup \mathcal{D}^{\text{Sc}}_{\text{buE}} \cup \mathcal{D}^{\text{Cmp}}_{\text{buE}} \cup \mathcal{D}^{\text{Cad}}_{\text{buE}}$$

El paso deductivo $\mathcal{D}^{\text{Ini}}_{\text{buE}}$ establece de partida la predicción de todos los subárboles participantes en los árboles elementales. Debido a la estrategia ascendente pura del esquema, esta regla comienza el reconocimiento de los subárboles sobre cualquier posición i de la cadena de entrada. Esto va a generar una gran cantidad de ítems espurios que con una adecuada técnica de filtrado podemos suprimir en parte, como veremos en la siguiente sección.

$$\mathcal{D}^{\text{Ini}}_{\text{buE}} = \frac{}{[N^\gamma \to \bullet\delta, i, i, -, -]}$$

El paso deductivo $\mathcal{D}^{\text{Foot}}_{\text{buE}}$ completa el reconocimiento de los subárboles que dominan los nodos pie de los árboles auxiliares. Al igual que el paso anterior, el análisis ascendente obliga a suponer que el nodo pie dominará cualquier subcadena válida de la cadena de entrada ($0 \leq k \leq l \leq n$). Es en este paso donde se asignan valores para los dos últimos índices, ya que éstos establecen el fragmento de la cadena de entrada que domina el subárbol que pende del nodo pie del árbol auxiliar.

$$\mathcal{D}^{\text{Foot}}_{\text{buE}} = \frac{}{[\mathbf{F}^\beta \to \bot\bullet, k, l, k, l]}$$

El paso deductivo $\mathcal{D}^{\text{Sc}}_{\text{buE}}$ realiza la lectura del símbolo actual si coincide con el símbolo situado entre las posiciones j y $j+1$ de la cadena de entrada.

$$\mathcal{D}^{\text{Sc}}_{\text{buE}} = \frac{[a, j, j+1] \quad [N^\gamma \to \nu \bullet a\omega, i, j, p, q]}{[N^\gamma \to \nu a \bullet \omega, i, j+1, p, q]}$$

El paso deductivo $\mathcal{D}^{\text{Cmp}}_{\text{buE}}$ continúa el reconocimiento del superárbol respecto a M^γ una vez que el subárbol dominado por él ha sido completamente reconocido. Este paso es equivalente a una operación de compleción en gramáticas incontextuales, por tanto, solo se puede aplicar cuando la adjunción no sea obligatoria ($\mathbf{nil} \in Adj(M^\gamma)$).

$$\mathcal{D}^{\text{Cmp}}_{\text{buE}} = \frac{[M^\gamma \to \delta\bullet, j', j, p, q] \quad [N^\gamma \to \nu \bullet M^\gamma\omega, i, j', p', q']}{[N^\gamma \to \nu M^\gamma \bullet \omega, i, j, p \cup p', q \cup q']}$$

donde la operación parcial de unión entre índices $p \cup p'$ se define como: p si el valor de p' está indefinido, p' si el valor de p está indefinido y - si los valores de p y p' están indefinidos.

El paso deductivo $\mathcal{D}^{\text{Cad}}_{\text{buE}}$ continúa el reconocimiento del superárbol respecto a M^γ

donde se ha efectuado la adjunción del árbol auxiliar β una vez que éste ha sido completamente reconocido, por tanto, se debe cumplir que $\beta \in Adj(M^\gamma)$. El movimiento del punto en el consecuente, que se sitúa detrás de M^γ, evita la posibilidad de múltiples operaciones de adjunción sobre M^γ.

$$\mathcal{D}_{\text{buE}}^{\text{Cad}} = \frac{\begin{array}{c}[\top \rightarrow \mathbf{R}^\beta \bullet, j, m, k, l] \\ [M^\gamma \rightarrow \delta \bullet, k, l, p, q] \\ [N^\gamma \rightarrow \nu \bullet M^\gamma \omega, i, j, p', q']\end{array}}{[N^\gamma \rightarrow \nu M^\gamma \bullet \omega, i, m, p \cup p', q \cup q']}$$

El conjunto de ítems válidos $\mathcal{V}_{\text{buE}}$ del esquema es

$$[N^\gamma \rightarrow \nu \bullet \omega, i, j, p, q]$$

tal que si p y q presentan un valor conocido entonces ν deriva $a_{i+1} \ldots a_p F^\beta a_{q+1} \ldots a_j$ y en otro caso ν deriva $a_{i+1} \ldots a_j$.

El conjunto de ítems finales viene dado por:

$$\mathcal{F}_{\text{buE}} = \{[\top \rightarrow \mathbf{R}^\alpha \bullet, 0, n, -, -] \mid \alpha \in \boldsymbol{I}\}$$

3 *El esquema* buLC

El esquema **buLC** (*bottom-up Left Corner*) para TAGs se obtiene mediante la aplicación de un filtro al esquema **buE** visto en la sección anterior. Como vimos en la definición del esquema **buLC** para gramáticas incontextuales, el objetivo es mejorar el comportamiento práctico de **buE** disminuyendo el número de ítems que se generan en el proceso de análisis. Para alcanzar dicha mejora en el análisis de TAGs vamos a eliminar aquellos ítems que en el esquema **buE** no aportan nada significativo en el proceso constructivo y que, al igual que en el esquema **buLC**, son los ítems de la forma $[N^\gamma \rightarrow \bullet M^\gamma \omega, j, j, -, -]$.

Puesto que a cada nodo de los árboles elementales le asociamos una producción como vimos en la sección 1, podemos extender la definición de esquina izquierda a los subárboles elementales que representa cada regla de producción.

El dominio del esquema $\mathcal{I}_{\text{buLC}}$ se define mediante:

$$\mathcal{I}_{\text{buLC}} = \mathcal{I}_{\text{buLC}}^{\epsilon} \cup \mathcal{I}_{\text{buLC}}^{(\text{i})}$$

$$\mathcal{I}_{\text{buLC}}^{\epsilon} = [N^\gamma \rightarrow \bullet, j, j, -, -]$$

donde $N^\gamma \rightarrow \epsilon$ es una regla de producción que pertenece a $\mathcal{P}(\gamma)$ y $0 \leq j$ es un índice en la cadena de entrada.

$$\mathcal{I}_{\text{buLC}}^{(\text{i})} = \{[N^\gamma \rightarrow P^\gamma \nu \bullet \omega, i, j, p, q]$$

$\}$ donde $N^\gamma \rightarrow P^\gamma \nu \omega$ es una regla de producción que pertenece a $\mathcal{P}(\gamma)$. Los índices $0 \leq i \leq j$ establecen las posiciones dentro de la cadena de entrada que delimitan el fragmento reconocido por $P^\gamma \nu$. Si p y q presentan un valor conocido, entonces γ es un árbol auxiliar y se cumple $i \leq p \leq q \leq j$.

Para eliminar los ítems con el punto delante de la esquina izquierda, vamos a sustituir el paso $\mathcal{D}_{\text{buE}}^{\text{Ini}}$, que es el que introduce ítems de esta forma, por los cuatro siguientes: $\mathcal{D}_{\text{buE}}^{\text{LC}_\text{t}}$, $\mathcal{D}_{\text{buLC}}^{\text{LC}_\epsilon}$, $\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{n}}$ y $\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{cad}}$. Los cuales llevarán a cabo el reconocimiento de la esquina izquierda de cada subárbol.

Los pasos deductivos del esquema vienen dados por:
$\mathcal{D}_{\text{buLC}} = \mathcal{D}_{\text{buE}}^{\text{LC}_\text{t}} \cup \mathcal{D}_{\text{buLC}}^{\text{LC}_\epsilon} \cup \mathcal{D}_{\text{buLC}}^{\text{LC}_\text{n}} \cup \mathcal{D}_{\text{buLC}}^{\text{LC}_\text{cad}} \cup$
$\mathcal{D}_{\text{buLC}}^{\text{Foot}} \cup \mathcal{D}_{\text{buLC}}^{\text{Sc}} \cup \mathcal{D}_{\text{buLC}}^{\text{Cmp}} \cup \mathcal{D}_{\text{buLC}}^{\text{Cad}}$

El paso deductivo $\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{t}}$ lanza el reconocimiento de los subárboles cuya esquina izquierda sea un símbolo terminal, entre todas las posiciones de la cadena de entrada donde aparece dicho símbolo. De esta forma eliminamos la posibilidad de que se generen producciones de la forma $N^\gamma \rightarrow \bullet a \omega$. $\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{t}}$ sustituye al paso $\mathcal{D}_{\text{buLC}}^{\text{Sc}}$ en los símbolos de la esquina izquierda.

$$\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{t}} = \frac{[a, j, j+1]}{[O^\gamma \rightarrow a \bullet \nu, j, j+1, -, -]}$$

El paso $\mathcal{D}_{\text{buLC}}^{\text{LC}_\epsilon}$ lanza el reconocimiento de los subárboles vacíos para todas las posiciones de entrada.

$$\mathcal{D}_{\text{buLC}}^{\text{LC}_\epsilon} = \frac{}{[O^\gamma \rightarrow \bullet, j, j, -, -]}$$

En principio, el paso $\mathcal{D}_{\text{buLC}}^{\text{Foot}}$ sería igual a $\mathcal{D}_{\text{buE}}^{\text{Foot}}$. Sin embargo, podemos aprovechar la estrategia ascendente del reconocimiento para filtrar dinámicamente los ítems que se generan mediante $\mathcal{D}_{\text{buLC}}^{\text{Foot}}$. Puesto que este paso completa el reconocimiento de los subárboles que dominan los nodos pie de los árboles auxiliares, su lanzamiento puede estar limitado a que previamente se haya reconocido el subárbol que cuelga de un nodo O^γ donde sea adjuntable un árbol auxiliar β, es decir, $\beta \in Adj(O^\gamma)$. Con este planteamiento, el paso deductivo $\mathcal{D}_{\text{buLC}}^{\text{Foot}}$ quedaría de la

siguiente forma:

$$\mathcal{D}_{\text{buLC}}^{\text{Foot}} = \frac{[O^\gamma \rightarrow \nu\bullet, k, l, p, q]}{[\mathbf{F}^\beta \rightarrow \bot\bullet, k, l, k, l]}$$

El paso $\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{n}}$ inicia el reconocimiento de los subárboles que cumplen que: (1) su esquina izquierda sea un símbolo no terminal y, (2) el subárbol que domina dicho símbolo haya sido completamente reconocido. Es decir, continúa el reconocimiento una vez completo el subárbol que domina la esquina izquierda. $\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{n}}$ es equivalente a la operación *completor* en gramáticas incontextuales, por tanto, solo se puede aplicar cuando la adjunción no sea obligatoria en la esquina izquierda, esto es, $\textbf{nil} \in Adj(O^\gamma)$.

$$\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{n}} = \frac{[O^\gamma \rightarrow \nu\bullet, j, k, p, q]}{[Q^\gamma \rightarrow O^\gamma \bullet \omega, j, k, p, q]}$$

Cuando la esquina izquierda de un subárbol sea un nodo adjuntable ($\beta \in Adj(O^\gamma)$) se aplica el paso deductivo $\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{cad}}$, el cual requiere que se hayan reconocido completamente el árbol auxiliar β y el subárbol que domina el nodo pie de dicho árbol. El paso $\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{cad}}$ sustituye a $\mathcal{D}_{\text{buLC}}^{\text{Cad}}$ en los símbolos de la esquina izquierda.

$$\mathcal{D}_{\text{buLC}}^{\text{LC}_\text{cad}} = \frac{\begin{array}{c}[\top \rightarrow \mathbf{R}^\beta\bullet, j, m, k, l] \\ [O^\gamma \rightarrow \nu\bullet, k, l, p, q]\end{array}}{[Q^\gamma \rightarrow O^\gamma \bullet \omega, j, m, p, q]}$$

El paso deductivo $\mathcal{D}_{\text{buLC}}^{\text{Sc}}$ tiene la misma función que $\mathcal{D}_{\text{buE}}^{\text{Sc}}$, pero solo se aplica a símbolos terminales que no sean esquinas izquierdas en sus subárboles.

$$\mathcal{D}_{\text{buLC}}^{\text{Sc}} = \frac{\begin{array}{c}[a, j, j+1] \\ [N^\gamma \rightarrow P^\gamma\nu \bullet a\omega, i, j, p, q]\end{array}}{[N^\gamma \rightarrow P^\gamma\nu a \bullet \omega, i, j+1, p, q]}$$

El paso $\mathcal{D}_{\text{buLC}}^{\text{Cmp}}$ tiene la misma función que $\mathcal{D}_{\text{buE}}^{\text{Cmp}}$, pero solo se aplica a símbolos no terminales sin restricción de adjunción obligatoria ($\textbf{nil} \in Adj(M^\gamma)$) que no sean esquinas izquierdas en sus subárboles.

$$\mathcal{D}_{\text{buLC}}^{\text{Cmp}} = \frac{\begin{array}{c}[M^\gamma \rightarrow \delta\bullet, j', j, p, q] \\ [N^\gamma \rightarrow P^\gamma\nu \bullet M^\gamma\omega, i, j', p', q']\end{array}}{[N^\gamma \rightarrow P^\gamma\nu M^\gamma \bullet \omega, i, j, p \cup p', q \cup q']}$$

La función del paso deductivo $\mathcal{D}_{\text{buLC}}^{\text{Cad}}$ es la misma que la de $\mathcal{D}_{\text{buE}}^{\text{Cad}}$, pero solo se aplica a símbolos no terminales adjuntables ($\beta \in Adj(M^\gamma)$) que no sean esquinas izquierdas en sus subárboles.

$$\mathcal{D}_{\text{buLC}}^{\text{Cad}} = \frac{\begin{array}{c}[\top \rightarrow \mathbf{R}^\beta\bullet, j, m, k, l] \\ [M^\gamma \rightarrow \delta\bullet, k, l, p, q] \\ [N^\gamma \rightarrow P^\gamma\nu \bullet M^\gamma\omega, i, j, p', q']\end{array}}{[N^\gamma \rightarrow P^\gamma\nu M^\gamma \bullet \omega, i, m, p \cup p', q \cup q']}$$

El conjunto de ítems válidos $\mathcal{V}_{\text{buLC}}$ del esquema es

$$[N^\gamma \rightarrow \nu \bullet \omega, i, j, p, q]$$

tal que si p y q presentan un valor conocido entonces ν deriva $a_{i+1} \ldots a_p F^\beta a_{q+1} \ldots a_j$ y en otro caso ν deriva $a_{i+1} \ldots a_j$.

El conjunto de ítems finales viene dado por:

$$\mathcal{F}_{\text{buLC}} = \{[\top \rightarrow \mathbf{R}^\alpha\bullet, 0, n, -, -] \mid \alpha \in \boldsymbol{I}\}$$

La complejidad temporal del algoritmo con respecto a la longitud n de la cadena de entrada viene dada por la complejidad del paso deductivo $\mathcal{D}_{\text{buLC}}^{\text{Cad}}$, que presenta el número máximo de variables de entrada. Sin embargo, tal como se detalla en [7], podemos definir una regla intermedia que elimine las variables no relevantes y reducir la complejidad temporal a $O(n^6)$. La complejidad espacial del analizador es de $O(n^4)$, ya que cada ítem almacena cuatro posiciones dentro de la cadena de entrada. Los algoritmos descritos son sólo reconocedores, pero es posible obtener el bosque de análisis introduciendo información adicional en los ítems indicando el modo en que se dedujeron.

4 Resultados experimentales

El analizador **buLC** propuesto en este trabajo presenta la misma complejidad teórica que el analizador **buE**, sin embargo, en esta sección vamos a ver como el comportamiento práctico es significativamente mejor. Aunque depende de la gramática y cadena de entrada consideradas, es habitual considerar el número de ítems deducidos en el proceso de análisis como medida fiable del comportamiento de un analizador deductivo.

Para llevar a cabo el estudio comparado del comportamiento de ambos analizadores para TAGs vamos a usar: (1) una implementación de la máquina deductiva de análisis [5] usando el paradigma de programación lógica, la cual nos va a permitir animar sistemas de análisis; (2) un conjunto de siete gramáticas

	Gramática G1		Gramática G2		Gramática G3		Gramática G4	
n	**buLC**	**buE**	**buLC**	**buE**	**buLC**	**buE**	**buLC**	**buE**
1	4	18	4	20	19	33	6	32
2	12	35	13	39	45	66	20	65
3	25	58	28	65	89	117	46	116
4	44	88	50	99	159	194	90	193
5	70	126	80	142	265	307	160	306
6	104	173	119	195	419	468	266	467
7	147	230	168	259	635	691	420	690
8	200	298	228	335	929	992	636	991
9	264	378	300	424	1319	1389	930	1388
10	340	471	384	527	1825	1902	1320	1901

Tabla 1: Ítems deducidos en las gramáticas G1, G2, G3 y G4

	Gramática G5		Gramática G6		Gramática G7	
	$a^n b^n$		$a^n b^n c^n$		$a^n b^n e c^n d^n$	
n	**buLC**	**buE**	**buLC**	**buE**	**buLC**	**buE**
1	20	49	20	46	12	70
2	38	102	37	99	20	146
3	57	173	54	168	28	245
4	77	264	71	253	36	367
5	98	377	88	354	44	512
6	120	514	105	471	52	680
7	143	677	122	604	60	871
8	167	868	139	753	68	1085
9	192	1089	156	918	75	1322
10	218	1342	173	1099	83	1582

Tabla 2: Ítems deducidos en G5, G6 y G7

que recogen las características más relevantes de las gramáticas de adjunción de árboles.

El conjunto de gramáticas que se va a emplear como banco de pruebas lo podemos dividir en dos grupos atendiendo a su forma:

- Un conjunto de cuatro gramáticas que reconocen lenguajes regulares y que constan de árboles auxiliares recursivos por la izquierda y por la derecha. Este tipo de árboles elementales es muy frecuente en la definición de gramáticas para lenguajes naturales, como se puede comprobar en [6]. Los árboles elementales de estas cuatro gramáticas, descritos con notación de paréntesis anidados, son los siguientes:
  - Gramática G1: α = S(e) ; β = S(e,S*)
  - Gramática G2: α = S(e) ; β = S(S*,e)
  - Gramática G3: α = S(ϵ) ; $\beta 1$ = S(S*,e) ; $\beta 2$ = S(e,S*)
  - Gramática G4: α = S(e) ; $\beta 1$ = S(S*,e) ; $\beta 2$ = S(e,S*)

Hemos probado estas gramáticas con cadenas de entrada de longitudes entre 1 y 10 símbolos, obteniendo los resultados que se muestran en la tabla 1. La media de reducción en el número de ítems deducidos es: 46% en las gramáticas G1 y G2, 16% en la gramática G3, y 49% en la gramática G4. Se observa una reducción muy significativa en el número de ítems deducidos, especial-

mente en la gramática que presenta recursión tanto por la izquierda como por la derecha. Por otra parte, también hemos observado una sensible disminución en el número de operaciones de compleción de adjunción.

- Un conjunto de tres gramáticas que reconocen lenguajes que muestran la capacidad generativa de las TAGs, incluyendo lenguajes dependientes e independientes del contexto. Las tres incluyen árboles auxiliares con espinas de longitud tres y cuyos nodos son adjuntables. Los árboles elementales de estas gramáticas, descritos con notación de paréntesis anidados, son los siguientes:

  - Gramática G5: α = S(ϵ) ; $\beta 1$ = S(a,T(S*,b); $\beta 2$ = T(a,S(T*,b))
  - Gramática G6: α = S(ϵ) ; β = S(a,S(b,S*,c))
  - Gramática G7: α = S(ϵ) ; β = S(a,S(b,S*,c),d)

  Aquí las cadenas de entrada tendrán una longitud entre 2 y 41 símbolos, obteniendo los resultados mostrados en la tabla 2. La media de reducción de ítems es muy elevada en los tres casos, siendo de un 74% en las gramáticas G5 y G6, y de un 91% en la gramática G7. En las gramáticas G5 y G6 también se produce una reducción significativa en el número de compleciones de adjunción, aunque este dato no se muestra en la tabla.

5 Conclusión

Definimos un nuevo analizador para TAGs a partir de una extensión del analizador *bottom-up Left Corner* para gramáticas incontextuales. Aunque el nuevo analizador mantiene las cotas teóricas de complejidad temporal y espacial de los analizadores para TAGs, supone una disminución muy elevada en el número de ítems generados en el proceso de análisis respecto al analizador *bottom-up Earley*. Esta reducción es consecuencia del filtro sobre la esquina izquierda que hemos establecido en los subárboles de los árboles elementales, y hace que el comportamiento de **buLC** en casos prácticos sea muy superior a **buE**. Como trabajos futuros, estamos investigando la posibilidad de extender la relación de esquina izquierda para mejorar el comportamiento de los analizadores predictivos basados en Earley.

Referencias

[1] Aravind K. Joshi y Yves Schabes. Tree-adjoining grammars. In Grzegorz Rozenberg and Arto Salomaa, editors, *Handbook of Formal Languages. Volumen 3: Beyond Words*, capítulo 2, páginas 69–123. Springer-Verlag, Berlín/Heidelberg/Nueva York, 1997.

[2] Klaas Sikkel. *Parsing Schemata — A Framework for Specification and Analysis of Parsing Algorithms*. Texts in Theoretical Computer Science — An EATCS Series. Springer-Verlag, Berlín/Heidelberg/Nueva York, 1997.

[3] Mark-Jan Nederhof. Solving the correct-prefix property for TAGs. In *Proc. of Fifth Meeting on Mathematics of Language*, páginas 124–130, Scholoss Dagstuhl, Saarbruecken, Alemania, agosto de 1997.

[4] Miguel A. Alonso, David Cabrero, Eric de la Clergerie, y Manuel Vilares. Tabular algorithms for TAG parsing. In *Proc. of EACL'99, Ninth Conference of the European Chapter of the Association for Computational Linguistics*, páginas 150–157, Bergen, Noruega, junio de 1999. ACL.

[5] Stuart M. Shieber, Yves Schabes, y Fernando C. N. Pereira. Principles and implementation of deductive parsing. *Journal of Logic Programming*, 24(1–2):3–36, julio-agosto de 1995.

[6] The XTAG Research Group. A lexicalized tree adjoining grammar for English. http://www.cis.upenn.edu/∼xtag. Technical Report IRCS 95-03, IRCS, Institute for Research in Cognitive Science, University of Pennsylvania, Filadelfia PA, EE.UU. 1999.

[7] Víctor J. Díaz. Gramáticas de Adjunción de Árboles: Un enfoque deductivo en el análisis sintáctico Tesis doctoral, Universidad de Sevilla, octubre de 2000.