

Una Gramática Libre de Contexto para el Lenguaje del Ambiente de Visualización Molecular - AVISMO

F. Narciso¹, A. Ríos-Bolívar¹, F. Hidrobo², O. Vicuña¹

¹Universidad de Los Andes, Facultad de Ingeniería, Escuela de Ingeniería de Sistemas, Venezuela

²Universidad de Los Andes, Facultad de Ciencias, Departamento de Física, Venezuela

RESUMEN

A objeto de generar gráficos en 2D y 3D de modelos moleculares de compuestos orgánicos a partir de un programa escrito en el Lenguaje del Ambiente de Visualización Molecular (AVISMO) utilizando un traductor, en este artículo se presenta la definición de una gramática, la cual está conformada por la unión de dos gramáticas, una que genera el Lenguaje de Especificación de Modelos Moleculares (LEMM), y la otra que genera el Lenguaje de Operación y Manejo de Modelos Moleculares (LOMMM). Estas gramáticas están definidas en función de los elementos que conforman una gramática formal de tipo 2 o libre de contexto (símbolos terminales, símbolos no terminales, axioma y reglas de producción). Se presenta como resultado la gramática libre de contexto resultante de la unión de estas dos gramáticas, una vez sometida a un proceso de transformación de sus reglas de producción de notación EBNF a notación BNF-no ampliada, seguido de un proceso de limpieza de la misma, con el fin de obtener reglas de producción fáciles de manipular por el traductor desarrollado para AVISMO bajo el paradigma de software libre.

ABSTRACT

In order to generate 2D and 3D graphics of molecular models of organic chemical compounds from a program written in the Molecular Visualization Environment Language (AVISMO, spanish acronym) by using a translator, the definition of a grammar is presented in this paper, which is formed by the union of two grammars, one of these generates the Specification of Molecular Models Language (LEMM, spanish acronym), while the other generates the Operation and Management of Molecular Model Language (LOMMM, spanish acronym). Both grammars are defined in function of the components of a type-2 grammar or context-free formal grammar (terminal symbols, non terminal symbols, axiom and production rules). As a result, a context-free grammar resulting from the union of these two grammars is presented, once subjected to a transformation process of its production rules from EBNF notation to not extended BNF notation. Also, this grammar has been cleaned up, in order to obtain production rules easy to manipulate by the translator developed for AVISMO under the free software paradigm.

Keywords: Modelos moleculares, Visualización Molecular, Gramáticas Libres de contexto.

1. Introducción

La visualización molecular puede ser considerada como una de las áreas más importantes dentro de la bioinformática. Entre sus aplicaciones más relevantes se destacan el diseño de nuevos fármacos [Dür02], así como el uso de herramientas de visualización molecular en el proceso de enseñanza-aprendizaje en cursos de biología, química y geología, y en todos aquellos cursos en los que se requiera el uso de modelos moleculares. Para visualizar una molécula es necesario conocer su información estructural, la cual se representa como una dupla de conjuntos conformada por átomos y enlaces atómicos. Cada elemento de estos conjuntos debe estar especificado por datos, tales como las coordenadas atómicas, su tipo y la cadena química a la cual pertenece, entre otros.

Diversos productos de *software*, tanto de licencias libres (GPL o similares) como de licencias privativas, han sido

desarrollados para satisfacer la necesidad de visualizar la estructura molecular de un compuesto químico. Tras evaluar las bondades de los recursos presentados por [MK08], entre los paquetes de licencia libre destacan: Jmol, un visor de moléculas gratuito y de código abierto para estudiantes, profesores e investigadores en química y bioquímica, es multi-plataforma, compatible con sistemas Windows, Mac OS X y Linux/Unix. Protein Explorer, versión mejorada de RasMol, *software* gratuito creado por Eric Martz de la Universidad de Massachusetts con objetivos educacionales y de investigación, ampliamente extendido en la comunidad universitaria. En cuanto a *software* privativo se pueden mencionar, entre otros, Gaussian y Accelrys. Este último produce gráficos de notable calidad. La gran mayoría de estos productos de *software* utilizan una base de datos, en la cual se tiene registrada la información estructural de los compuestos químicos comunes. De esta manera, el usuario se-

lecciona un compuesto químico y puede visualizarlo gráficamente. El uso de estas bases de datos limita el número de moléculas que pueden ser visualizadas. Además, tiene un alto costo de almacenamiento por cada molécula en la base de datos. Para solucionar este problema se crearon bancos de modelos moleculares en la Web, entre los que se pueden mencionar PDB Lite (<http://oca.ebi.ac.uk/ocabin/pdblite>), RCSB (<http://www.rcsb.org/pdb>) y NCBI (<http://www.ncbi.nlm.nih.gov/>). Una limitación presente en estos bancos de datos es que contienen tantas estructuras moleculares que se dificulta encontrar la estructura molecular deseada. Otra limitación, según [MK08], es que estos bancos de datos no contienen coordenadas atómicas de moléculas pequeñas como la glucosa, un ácido graso o un aminoácido. Una técnica menos usada provee al usuario una Interfaz Gráfica de Usuario (IGU) que le permite enlazar átomos y moléculas de forma gráfica, manteniendo este proceso alejado de las propiedades químicas y estructurales de una molécula en particular. Esto exige que el usuario tenga un conocimiento previo de las moléculas que desea visualizar.

Como puede observarse, las herramientas de visualización molecular actualmente empleadas exigen del usuario un conocimiento previo de las moléculas que desea visualizar. Además, tienen serias limitaciones con respecto a cuántas o cuáles moléculas pueden manejar. Surge entonces la necesidad de desarrollar una técnica de generación de modelos moleculares, con menos limitaciones y de más fácil manejo. Esta necesidad da lugar al desarrollo del traductor propuesto en [Vic09], denominado Ambiente de Visualización Molecular (AVISMO), que permite la visualización en dos dimensiones (2D) y en tres dimensiones (3D) de modelos moleculares de compuestos orgánicos a partir de su definición estructural, como consecuencia de un proceso de traducción. Previamente al desarrollo del traductor, se definieron dos Gramáticas Libres de Contexto (\mathcal{GLC}) [ALSU07], las cuales son descritas en este artículo. La primera genera el lenguaje de especificación de modelos moleculares y la segunda genera el lenguaje de operación y manejo de modelos moleculares. De la unión de estas \mathcal{GLC} se obtiene una \mathcal{GLC} que genera el lenguaje de AVISMO. Esta última se transforma, modificando las reglas de producción de la notación EBNF a la notación BNF-no ampliada, y aplicando el proceso de limpieza de una gramática [SG86], para así obtener una \mathcal{GLC} equivalente que contiene el menor número de imperfecciones posibles, es decir, escrita de forma correcta, lo cual facilita, entre otras cosas, su manipulación por parte del traductor que toma como entrada una cadena generada por esta gramática, calcula las propiedades químicas y estructurales de una molécula y permite su visualización. Para el desarrollo del traductor se utilizó el paradigma del *software* libre, buscando con esto un producto de *software* robusto, seguro, confiable y de licencia libre para el uso educacional.

2. Ambiente de Visualización Molecular

Por definición, un traductor es un *software* que lee un texto escrito en un lenguaje, llamado lenguaje fuente (LF), y lo traduce a un texto equivalente escrito en otro lenguaje, llamado lenguaje destino o lenguaje objeto (LO). Adicionalmente, el traductor informa al usuario la presencia de errores en el texto escrito en el lenguaje fuente [ALSU07].

Para el desarrollo del traductor propuesto en [Vic09], cuyo nombre es AVISMO, se requirió la definición de una \mathcal{GLC}

para generar el LF del traductor. Este LF permite especificar y manipular modelos moleculares de compuestos orgánicos (ver Figura 1). En el caso de AVISMO el LO no es una representación textual, sino que consiste en representaciones visuales (gráficos) de los modelos moleculares de compuestos orgánicos en 2D (ver Figura 2) y en 3D. Una vez generado un gráfico en 2D o en 3D, éste puede ser almacenado para su posterior uso y visualización. En AVISMO, el principio de visualización molecular yace en la generación de gráficos a partir de especificaciones moleculares. Cabe destacar que AVISMO provee una IGU implementada utilizando el entorno integrado de desarrollo (IDE) de Lazarus. Esta IGU ofrece controles que manejan la traducción, un área de trabajo que permite editar el texto escrito en LF (programa fuente), una consola de errores y herramientas que permiten el uso y manejo del traductor. Además, desde esta IGU se instancia el manejador gráfico GraphViz, el cual permite visualizar y manipular los gráficos generados por el traductor. Para implementar los analizadores léxico, sintáctico y semántico del traductor, escritos en lenguaje de programación Java, se utilizó la herramienta ANTLR (*ANother Tool for Language Recognition*). AVISMO se desarrolló utilizando el método de desarrollo de *software* basado en componentes denominado *WATCH-Component* [MB03].

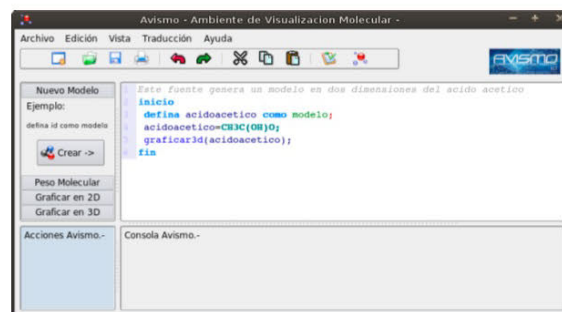


Figura 1: Texto fuente escrito en lenguaje fuente de AVISMO

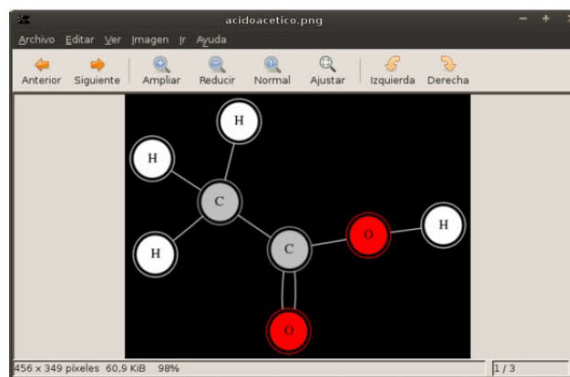


Figura 2: Gráfico en 2D generado por AVISMO

Como se mencionó anteriormente, para el desarrollo de AVISMO fue necesario definir una \mathcal{GLC} o Gramática de tipo 2 [ALSU07, Cho59], la cual se utiliza para definir formalmente la sintaxis de los lenguajes naturales, los lenguajes de programación y de otros lenguajes formales, en función del conjunto T de símbolos terminales, el conjunto N de símbolos no terminales, el axioma S y el conjunto P de reglas

de producción, denotada $GLC = (\mathcal{N}, \mathcal{T}, \mathcal{S}, \mathcal{P})$. Siendo que el proceso de traducción contempla las fases de análisis léxico, sintáctico y semántico del texto fuente escrito en LF, la GLC definida para AVISMO se utiliza en la construcción de su analizador sintáctico. Gráficamente, la relación entre los conceptos tratados en esta sección, y que han sido utilizados para el desarrollo de AVISMO, se presenta en la Figura 3.

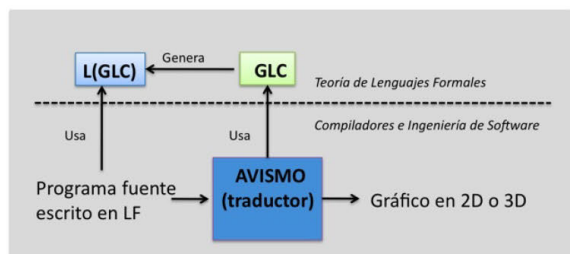


Figura 3: Relación entre los conceptos utilizados para el desarrollo de AVISMO

Atendiendo a la visión con que se concibe el LF de AVISMO, éste se divide en dos partes que trabajan en conjunto como un todo. Una parte, el Lenguaje de Especificación de Modelos Moleculares (LEMM), permite definir estructuralmente un compuesto químico orgánico. Este lenguaje ha sido diseñado tomando como base las formulaciones de compuestos orgánicos propuestas por la IUPAC (del inglés *International Union of Pure and Applied Chemistry*) [PPI94, Int79], con el fin de lograr una mayor estandarización en la notación utilizada. La parte restante del lenguaje de AVISMO, que trabaja en conjunto con LEMM, es el Lenguaje de Operación y Manejo de Modelos Moleculares (LOMMM). Este lenguaje permite manipular cada compuesto químico orgánico bajo la representación lógica de una variable. Además, define estructuralmente el macro lenguaje de AVISMO.

3. Lenguaje de Especificación de Modelos Moleculares

Para la definición de la GLC que genera LEMM, se estudiaron las nomenclaturas químicas y las formulaciones de compuestos químicos utilizados en la actualidad. Esto con el fin de analizar las fortalezas y debilidades de cada uno de ellos y, obtener así, la formulación para dicho lenguaje. En particular, se realizó una comparación entre las formulaciones propuestas por IUPAC, a fin de obtener la formulación más adecuada para LEMM. Los requisitos que se tomaron en consideración, para tomar una elección entre estas formulaciones, fueron:

- Alto grado de adaptabilidad:** Capacidad de ser adaptada a los requisitos de AVISMO.
- Identificación unívoca de compuestos químicos:** Que una fórmula química permita identificar y definir a un compuesto en particular sin ambigüedades.
- Facilidad de formular compuestos químicos en una línea de texto:** Dada la estructura de AVISMO, la entrada al traductor es un archivo de texto, por lo que la formulación debe permitir describir un compuesto químico orgánico en una línea de texto.

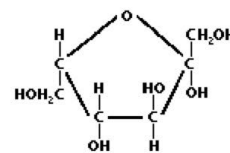


Figura 4: Fórmula estructural de la fructosa

3.1. Formulaciones químicas para el Lenguaje de Especificación de Modelos Moleculares

De acuerdo al objetivo establecido para LEMM, en la sección 2 se analizaron las formulaciones de IUPAC. El primer caso considerado, la *fórmula molecular*, representa los compuestos químicos mediante el número total de átomos que forman la molécula; por ejemplo, la fructosa es $C_6H_{12}O_6$. A pesar de satisfacer dos de los requisitos anteriores (a y b), existe un problema con el segundo requisito, debido a que con esos elementos y ese número de átomos (6 de Carbono, 12 de Hidrógeno y 6 de Oxígeno) se pueden formar distintos compuestos, o lo que es lo mismo, muchas moléculas de estructura distinta pueden tener la misma fórmula molecular, lo cual genera ambigüedad en el lenguaje. De manera similar, en la *fórmula empírica*, está presente esta ambigüedad, ya que expresa sólo el número relativo de átomos perteneciente a cada elemento presente en la molécula, por ejemplo en la fórmula empírica de la fructosa (CH_2O). El último caso considerado, la *fórmula estructural*, indica la unión entre los átomos de una molécula, mediante una especie de representación gráfica plana de una molécula (ver Figura 4). Esta representación resuelve el problema de ambigüedad, puesto que encierra en ella misma la estructura de la molécula; sin embargo, no satisface completamente el tercer requisito (c), por la dificultad para describirla en una línea de texto.

Una esquema para resolver los problemas que se presentan con los tres requisitos considerados, implica unificar la formulación molecular y la estructural, obteniendo así una formulación que se pueda escribir en una línea de texto, representando adecuadamente la estructura de una molécula. La fórmula estructural puede representarse mediante un grafo no dirigido [Joy09], donde los átomos son los nodos del grafo y los enlaces son los arcos del grafo. Un recorrido sobre la cadena química más larga del grafo, da como resultado una representación lineal del compuesto químico orgánico.

Para definir un lenguaje de especificación de compuestos orgánicos utilizando grafos, es primordial solucionar los detalles de representación necesarios para apoyar la visualización molecular, tales como definición y orientación de los grupos funcionales, cardinalidad de los enlaces (dobles, triples y cuádruples) y cadenas cíclicas. La representación de los grupos funcionales se logra a través de un artificio, signos de agrupación que encierran la cadena química que compone el grupo funcional, posicionados inmediatamente después del átomo de la cadena principal al que están enlazados. Para representar la orientación de los grupos funcionales, se usan paréntesis “()” para grupos funcionales ubicados por debajo de la cadena principal, y corchetes “[]” para grupos funcionales ubicados por arriba de la cadena química principal. De esta manera, se logra expresar en una cadena de texto la fórmula estructural del compuesto químico orgánico.

El recorrido del grafo permite identificar la ubicación de

los enlaces múltiples, por consiguiente, permite especificar la multiplicidad de cada enlace entre los átomos en la representación lineal. Esto conlleva a definir cuatro símbolos que identifiquen la multiplicidad del enlace:

- – para un enlace simple, cuando no se coloca explícitamente la cardinalidad, se toma 1 por omisión.
- = para enlaces dobles.
- : para enlaces triples.
- :: para enlaces cuádruples.

Cuando un átomo de la cadena principal se conecta con un grupo funcional, a través de un enlace que no es simple, se deberá anteponer el símbolo que represente la cardinalidad del enlace a la cadena química del grupo funcional.

Por último, para resolver la representación de la estructura cíclica de algunos compuestos, se escribe, al final de la cadena cíclica un enlace que indique que el último átomo de la cadena se enlaza al primero por medio de éste.

3.2. Gramática Libre de Contexto para el Lenguaje de Especificación de Modelos Moleculares

Según [Bre03], diseñar una \mathcal{GLC} consiste en proponer, dado un lenguaje \mathcal{L} , una gramática tal que su lenguaje generado es exactamente \mathcal{L} . Así, una \mathcal{GLC} es correcta con respecto a \mathcal{L} , cuando el lenguaje generado por dicha gramática no contiene palabras que estén fuera de \mathcal{L} . Por otra parte, una \mathcal{GLC} es completa cuando es capaz de generar al menos las palabras de \mathcal{L} . Al diseñar gramáticas es posible cometer dos clases de errores:

- a) “sobran palabras”: la gramática genera algunas palabras que no debería generar. Así, la gramática sería incorrecta.
- b) “faltan palabras”: Existen palabras en el lenguaje considerado para las que no hay derivación. En este caso, la gramática sería incompleta.

Tomando en consideración lo anterior, se presenta a continuación la definición de la \mathcal{GLC} para LEMM, en términos de sus conjuntos de símbolos terminales \mathcal{T} y símbolos no terminales \mathcal{N} , el axioma S y el conjunto de sus reglas de producción \mathcal{P} en notación EBNF [Ken04].

Una descripción de los compuestos orgánicos de un modelo molecular, está definida por la siguiente regla de producción:

```
<MODELO_MOLECULAR> ::= <CADENA_ATOMICA> | <CADENA_MOLECULAR>
```

donde <MODELO_MOLECULAR> representa al axioma S de la gramática, a partir del cual se derivan los modelos moleculares. Como puede observarse, en esta regla de producción se generan por separado los modelos que contienen sólo un átomo como cadena principal, y los modelos que contienen más de un elemento atómico en su cadena principal, debido a que estos últimos necesitan enlaces entre los distintos elementos. Por ejemplo, el argón (Ar) por sí solo, no necesita de enlaces moleculares, mientras que el propano ($CH_3CH_2CH_3$) necesita 12 enlaces simples. La definición de los no terminales <CADENA_ATOMICA> y <CADENA_MOLECULAR> viene dada por las siguientes reglas de producción:

```
<CADENA_ATOMICA> ::= <ELEMENTO> | <GRUPO_FUNCIONAL>
```

```
<CADENA_MOLECULAR> ::= <COMPUESTO> | <ELEMENTO>
```

```
<GRUPO_FUNCIONAL> ::= <ENLACE> | <COMPUESTO> | <COMPUESTO>
```

donde un <COMPUESTO>, que representa un compuesto químico, se define mediante la siguiente regla de producción:

```
<COMPUESTO> ::= <ELEMENTO> | <GRUPO_FUNCIONAL> | <ENLACE>
```

Esencialmente, tres símbolos no terminales componen estas tres reglas de producción:

- <ELEMENTO>: Representa los elementos químicos. A menudo, un elemento químico se escribe seguido de un número natural que indica la cantidad de átomos de la misma naturaleza que se están enlazados al mismo átomo destino; por ejemplo, en CH_4 , el número 4 indica que cuatro átomos de hidrógeno se combinan con un átomo de carbono. Las reglas de producción que definen este no terminal son las siguientes:

```
<ELEMENTO> ::= <ELEMENTO_QUIMICO> | <VALENCIA>
```

```
<ELEMENTO_QUIMICO> ::= "H" | "Li" | "Na" | "K" | "Rb" | "Cs" | "Fr" | "Be" | "Mg" | "Ca" | "Sr" | "Ba" | "Ra" | "Sc" | "Y" | "Tl" | "Zr" | "Hf" | "Db" | "V" | "Nb" | "Ta" | "U" | "Cr" | "Mo" | "W" | "Rf" | "Mn" | "Tc" | "Re" | "Bh" | "Fe" | "Ru" | "Os" | "Hn" | "Co" | "Rh" | "Ir" | "Mt" | "Ni" | "Pd" | "Pt" | "Cu" | "Ag" | "Au" | "Zn" | "Cd" | "Hg" | "B" | "Al" | "Ga" | "In" | "Tl" | "C" | "Si" | "Ge" | "Sn" | "Pb" | "N" | "P" | "As" | "Sb" | "Bi" | "O" | "S" | "Se" | "Te" | "Po" | "F" | "Cl" | "Br" | "I" | "At" | "He" | "Ne" | "Ar" | "Kr" | "Xe" | "Rn"
```

```
<VALENCIA> ::= "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
```

Por ejemplo, el oxígeno molecular, que esta compuesto por dos átomos unidos, se expresa como: O_2

- <GRUPO_FUNCIONAL>: Permite definir los grupos funcionales, es decir, un conglomerado de átomos, por lo general de pequeñas dimensiones, que se unen a una cadena química, aportándole ciertas propiedades y características específicas. Los grupos funcionales pueden ubicarse por encima o por debajo del elemento de la cadena a la que están unidos. Este no terminal está definido por las siguientes reglas de producción:

```
<GRUPO_FUNCIONAL> ::= <GRUPO_FUNCIONAL_INFERIOR> | <GRUPO_FUNCIONAL_SUPERIOR> | <GRUPO_FUNCIONAL_SUPERIOR> | <GRUPO_FUNCIONAL_INFERIOR>
```

```
<GRUPO_FUNCIONAL_INFERIOR> ::= "[" | <MODELO_GRUPO_FUNCIONAL> "]" | <GRUPO_FUNCIONAL_SUPERIOR> ::= "(" | <MODELO_GRUPO_FUNCIONAL> ")"
```

```
<MODELO_GRUPO_FUNCIONAL> ::= <ENLACE>
```

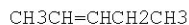
```
<MODELO_MOLECULAR>
```

De esta manera, el grupo funcional OH existente en el etanol, se expresa en LEMM como: $CH_2(OH)CH_3$

- <ENLACE>: Representa los enlaces químicos. En la química orgánica, los enlaces suelen tener al menos un carbono entre sus extremos. Un carbono, para lograr su estabilidad en el último nivel de electrones, necesita combinarse con cuatro electrones de otros átomos, estos pueden pertenecer a otros átomos de su misma naturaleza. El carbono entonces rige el tipo de enlaces que se pueden darse en un compuesto químico orgánico, los cuales pueden ser simples, dobles, triples y cuádruples. La regla de producción que define este no terminal es:

```
<ENLACE> ::= "-" | "=" | ":" | "::"
```

Por ejemplo, el 2-penteno, que contiene un doble enlace que une al segundo carbono con el tercer carbono, según esta regla de producción se expresa de la siguiente forma:



La \mathcal{GLC} para LEMM se define a continuación:

$$\mathcal{GLC}_{lemm} = (\mathcal{N}_{lemm}, \mathcal{T}_{lemm}, \mathcal{P}_{lemm}, \langle \text{MODELO_MOLECULAR} \rangle)$$

El conjunto de los símbolos no terminales \mathcal{N}_{lemm} se define como:

$$\mathcal{N}_{lemm} = \{ \langle \text{MODELO_MOLECULAR} \rangle, \langle \text{CADENA_ATOMICA} \rangle, \langle \text{CADENA_MOLECULAR} \rangle, \langle \text{ELEMENTO} \rangle, \langle \text{ELEMENTO_QUIMICO} \rangle, \langle \text{VALENCIA} \rangle, \langle \text{GRUPO_FUNCIONAL} \rangle, \langle \text{COMPUESTO} \rangle, \langle \text{ENLACE} \rangle, \langle \text{GRUPO_FUNCIONAL_INFERIOR} \rangle, \langle \text{GRUPO_FUNCIONAL_SUPERIOR} \rangle, \langle \text{MODELO_GRUPO_FUNCIONAL} \rangle \}$$

El conjunto de símbolos terminales \mathcal{T}_{lemm} se define como:

$$\mathcal{T}_{lemm} = \{ \text{“H”, “Li”, “Na”, “K”, “Rb”, “Cs”, “Fr”, “Be”, “Mg”, “Ca”, “Sr”, “Ba”, “Ra”, “Sc”, “Y”, “Ti”, “Zr”, “Hf”, “Db”, “V”, “Nb”, “Ta”, “Jl”, “Cr”, “Mo”, “W”, “Rf”, “Mn”, “Tc”, “Re”, “Bh”, “Fe”, “Ru”, “Os”, “Hh”, “Co”, “Rh”, “Ir”, “Mt”, “Ni”, “Pd”, “Pt”, “Cu”, “Ag”, “Au”, “Zn”, “Cd”, “Hg”, “B”, “Al”, “Ga”, “In”, “Tl”, “C”, “Si”, “Ge”, “Sn”, “Pb”, “N”, “P”, “As”, “Sb”, “Bi”, “O”, “S”, “Se”, “Te”, “Po”, “F”, “Cl”, “Br”, “I”, “At”, “He”, “Ne”, “Ar”, “Kr”, “Xe”, “Rn”, “1”, “2”, “3”, “4”, “5”, “6”, “7”, “8”, “9”, “.”, “=”, “-”, “.”, “:”, “(”, “)” \}$$

Por último, el conjunto de reglas de producción \mathcal{P}_{lemm} se define como:

$$\begin{aligned} \mathcal{P}_{lemm} = \{ & \langle \text{MODELO_MOLECULAR} \rangle ::= \langle \text{CADENA_ATOMICA} \rangle | \\ & \langle \text{CADENA_MOLECULAR} \rangle, \\ & \langle \text{CADENA_ATOMICA} \rangle ::= \langle \text{ELEMENTO} \rangle [\langle \text{GRUPO_FUNCIONAL} \rangle], \\ & \langle \text{CADENA_MOLECULAR} \rangle ::= \langle \text{COMPUESTO} \rangle \langle \text{ELEMENTO} \rangle \\ & [\langle \text{GRUPO_FUNCIONAL} \rangle] [\langle \text{ENLACE} \rangle] \langle \text{COMPUESTO} \rangle [\langle \text{COMPUESTO} \rangle], \\ & \langle \text{COMPUESTO} \rangle ::= \langle \text{ELEMENTO} \rangle [\langle \text{GRUPO_FUNCIONAL} \rangle] [\langle \text{ENLACE} \rangle], \\ & \langle \text{ELEMENTO} \rangle ::= \langle \text{ELEMENTO_QUIMICO} \rangle [\langle \text{VALENCIA} \rangle], \\ & \langle \text{ELEMENTO_QUIMICO} \rangle ::= \text{“H” | “Li” | “Na” | “K” | “Rb” | “Cs” | “Fr” | “Be” | “Mg” | “Ca” | “Sr” | “Ba” | “Ra” | “Sc” | “Y” | “Ti” | “Zr” | “Hf” | “Db” | “V” | “Nb” | “Ta” | “Jl” | “Cr” | “Mo” | “W” | “Rf” | “Mn” | “Tc” | “Re” | “Bh” | “Fe” | “Ru” | “Os” | “Hh” | “Co” | “Rh” | “Ir” | “Mt” | “Ni” | “Pd” | “Pt” | “Cu” | “Ag” | “Au” | “Zn” | “Cd” | “Hg” | “B” | “Al” | “Ga” | “In” | “Tl” | “C” | “Si” | “Ge” | “Sn” | “Pb” | “N” | “P” | “As” | “Sb” | “Bi” | “O” | “S” | “Se” | “Te” | “Po” | “F” | “Cl” | “Br” | “I” | “At” | “He” | “Ne” | “Ar” | “Kr” | “Xe” | “Rn”}, \\ & \langle \text{VALENCIA} \rangle ::= \text{“1” | “2” | “3” | “4” | “5” | “6” | “7” | “8” | “9”}, \\ & \langle \text{GRUPO_FUNCIONAL} \rangle ::= \langle \text{GRUPO_FUNCIONAL_INFERIOR} \rangle \\ & [\langle \text{GRUPO_FUNCIONAL_SUPERIOR} \rangle] | \langle \text{GRUPO_FUNCIONAL_SUPERIOR} \rangle \\ & [\langle \text{GRUPO_FUNCIONAL_INFERIOR} \rangle], \\ & \langle \text{GRUPO_FUNCIONAL_INFERIOR} \rangle ::= \text{“[”} \langle \text{MODELO_GRUPO_FUNCIONAL} \rangle \text{”} \text{“]”}, \\ & \langle \text{GRUPO_FUNCIONAL_SUPERIOR} \rangle ::= \text{“(”} \langle \text{MODELO_GRUPO_FUNCIONAL} \rangle \text{“)”}, \\ & \langle \text{MODELO_GRUPO_FUNCIONAL} \rangle ::= [\langle \text{ENLACE} \rangle] \langle \text{MODELO_MOLECULAR} \rangle, \\ & \langle \text{ENLACE} \rangle ::= \text{“-” | “=” | “.” | “:”} \end{aligned}$$

4. Lenguaje de Operación y Manejo de Modelos Moleculares

Este lenguaje, denominado LOMMM, determina la estructura sintáctica de los programas escritos en el LF de AVISMO. Además, permite definir la sintaxis del manejo de los modelos moleculares, así como las operaciones con los mismos, una vez que han sido definidos utilizando LEMM, lo cual se logra manipulando las especificaciones escritas en LEMM como *instancias del objeto modelo molecular*. Trabajar con el paradigma de Programación Orientada a Objetos (POO) permite definir operaciones aplicables a todas las instancias de este objeto [Joy98]. Es así como el lenguaje provee la sintaxis que permite al traductor graficar modelos moleculares y computar propiedades físicas del compuesto químico orgánico representado por el modelo molecular.

El lenguaje debe permitir al usuario:

- Documentar el programa fuente.
- Marcar el inicio y fin de las operaciones sobre los modelos moleculares.
- Manejar los modelos moleculares.
- Realizar operaciones con modelos moleculares.

Para satisfacer los dos primeros puntos (a y b), se escribe la palabra *inicio* como primera sentencia del programa fuente (de aquí en adelante se usará la palabra *programa* en lugar de *programa fuente*) y la palabra *fin* como última sentencia del programa escrito en LOMMM. Cualquier texto escrito antes del inicio y después del fin del programa será tomado como comentario y no tendrá significado para el tratamiento de modelos moleculares. Cabe destacar que el lenguaje no contempla comentarios escritos dentro del bloque delimitado por *inicio-fin*. Así, la estructura sintáctica básica de un programa escrito en LOMMM es la siguiente:

Este es un comentario

inicio

fin

Este es el primer programa escrito en LOMMM

Para manejar modelos moleculares, y satisfacer el punto (c), LOMMM permite crear instancias de modelos moleculares. Para esto, el lenguaje provee la siguiente sentencia:

defina <nombre_variable> como modelo;

Cabe destacar que en la definición de la sintaxis de las sentencias de LOMMM, el uso de los caracteres “<” y “>” indica que la palabra escrita dentro de los mismos no es una palabra reservada de este lenguaje, a diferencia de la notación EBNF, en la cual estos caracteres se utilizan para denotar símbolos no terminales.

La especificación de un modelo molecular escrita en LEMM es asignada a una variable mediante la siguiente sentencia, constituyendo el punto de unión entre los lenguajes LEMM y LOMMM:

<nombre_variable> = <modelo_escrito_en_LEMM>;

Por ejemplo, la definición del 2 metil-propano como modelo molecular manipulable, utilizando el lenguaje de AVISMO se presenta a continuación:

El compuesto 2-metil-propano tiene al propano como cadena principal y un metil como grupo funcional.

Nótese que las sentencias terminan en punto y coma (;)

```
inicio
defina comp1 como modelo;
comp1 = CH3CH(CH3)CH3;
fin
```

Como se puede observar, el grupo metilo se conecta al segundo carbono

Una vez que la representación de un compuesto químico orgánico se crea como una instancia del objeto modelo molecular, se pueden realizar operaciones con ésta (graficar en 2D, graficar en 3D, obtener el peso molecular), con lo cual se satisface el último punto en consideración (d).

Toda operación con modelos moleculares se define sintácticamente de la siguiente forma:

```
nombre_funcion(<nombre_variable>)
```

En LOMMM, las operaciones que se pueden realizar con modelos moleculares son:

- **graficar2d**: Genera la representación en 2D de un modelo molecular.
- **graficar3d**: Genera una representación dinámica en 3D de un modelo molecular.
- **pesomolecular**: Retorna el peso molecular del compuesto químico que representa el modelo molecular.

Además, se han tomado técnicas usadas en lenguajes que manejan gráficas matemáticas como MAPLE® y MATLAB®, los cuales permiten que el delimitador que indica el final de una sentencia pueda variar. Así, para el lenguaje de AVISMO, si una operación que implica generar una gráfica termina en punto y coma “;” se genera la gráfica solicitada; sin embargo, si el final de la sentencia está indicado por el carácter dos puntos “:”, entonces no se genera la gráfica. De acuerdo a esto para generar una gráfica en 3D que represente al modelo molecular del 2 metil-pentano se escribe:

```
inicio
defina comp1 como modelo;
comp1 = CH3CH(CH3)CH3;
graficar3d(comp1);
fin
```

si se escribe:

```
inicio
defina comp1 como modelo;
comp1 = CH3CH(CH3)CH3;
graficar3d(comp1);
graficar2d(comp1);
fin
```

se genera sólo la gráfica en 2D, puesto que la sentencia que indica graficar en 3D finaliza con el caracter “;”.

4.1. Gramática Libre de Contexto para el Lenguaje de Operación y Manejo de Modelos Moleculares

Al igual que la gramática definida para LEMM, esta gramática debe ser una \mathcal{GLC} completa, donde no sobren ni falten palabras en el lenguaje generado, en el que se escriben los programas que son traducidos por AVISMO. La primera regla de producción a considerar es la siguiente:

$$\langle S \rangle ::= \text{"inicio"} \{ \langle \text{SENTENCIA} \rangle \langle \text{FIN_DE_LINEA} \rangle \} \text{"fin"}$$

Donde $\langle s \rangle$ es el axioma de la gramática, a partir del cual se genera cualquier programa válido para AVISMO. Esta regla de producción utiliza un par de símbolos no terminales, $\langle \text{SENTENCIA} \rangle$ y $\langle \text{FIN_DE_LINEA} \rangle$, que combinados como muestra la regla de producción, generan todo el cuerpo de un programa. La regla de producción que define $\langle \text{FIN_DE_LINEA} \rangle$ se usa para generar el símbolo de fin de una sentencia. Como se explicó anteriormente, éste puede ser un punto y coma “;” o dos puntos “:”:

$$\langle \text{FIN_DE_LINEA} \rangle ::= \text{";"} \mid \text{":"}$$

La regla de producción:

$$\langle \text{SENTENCIA} \rangle ::= \langle \text{DECLARACION_DE_VARIABLE} \rangle \mid$$

$$\langle \text{DEFINICION_DE_MODELO} \rangle \mid \langle \text{OPERACION_CON_MODELO} \rangle$$

permite generar el bloque de sentencias que posibilita el proceso de manejo y operación de los modelos moleculares. Como puede observarse, tres símbolos no terminales componen esta regla de producción:

- $\langle \text{DECLARACION_DE_VARIABLE} \rangle$: Una declaración de variable permite crear una nueva instancia del objeto modelo molecular. Está definida por las siguientes reglas de producción:

$$\langle \text{DECLARACION_DE_VARIABLE} \rangle ::= \text{"defina"} \langle \text{ID} \rangle \text{" como"} \langle \text{TIPO} \rangle$$

$$\langle \text{ID} \rangle ::= \langle \text{LETRA} \rangle \{ [\langle \text{LETRA} \rangle] [\langle \text{DIGITO} \rangle] \}$$

$$\langle \text{LETRA} \rangle ::= \text{"A"} \mid \text{"B"} \mid \text{"C"} \mid \text{"D"} \mid \text{"E"} \mid \text{"F"} \mid \text{"G"} \mid \text{"H"} \mid \text{"I"} \mid \text{"J"} \mid \text{"K"} \mid \text{"L"} \mid \text{"M"} \mid \text{"N"} \mid \text{"O"} \mid \text{"P"} \mid \text{"Q"} \mid \text{"R"} \mid \text{"S"} \mid \text{"T"} \mid \text{"U"} \mid \text{"V"} \mid \text{"W"} \mid \text{"X"} \mid \text{"Y"} \mid \text{"Z"} \mid \text{"a"} \mid \text{"b"} \mid \text{"c"} \mid \text{"d"} \mid \text{"e"} \mid \text{"f"} \mid \text{"g"} \mid \text{"h"} \mid \text{"i"} \mid \text{"j"} \mid \text{"k"} \mid \text{"l"} \mid \text{"m"} \mid \text{"n"} \mid \text{"o"} \mid \text{"p"} \mid \text{"q"} \mid \text{"r"} \mid \text{"s"} \mid \text{"t"} \mid \text{"u"} \mid \text{"v"} \mid \text{"w"} \mid \text{"x"} \mid \text{"y"} \mid \text{"z"}$$

$$\langle \text{DIGITO} \rangle ::= \text{"0"} \mid \text{"1"} \mid \text{"2"} \mid \text{"3"} \mid \text{"4"} \mid \text{"5"} \mid \text{"6"} \mid \text{"7"} \mid \text{"8"} \mid \text{"9"}$$

$$\langle \text{TIPO} \rangle ::= \text{"modelo"}$$

- $\langle \text{DEFINICION_DEL_MODELO} \rangle$: Para definir un modelo molecular, se hace uso de LEMM, siendo la definición del modelo el punto de unión entre los dos lenguajes, divididos desde el inicio por su diferencia de objetivos. La regla de producción que define un modelo molecular es la siguiente:

$$\langle \text{DEFINICION_DEL_MODELO} \rangle ::= \langle \text{ID} \rangle \text{"="} \langle \text{MODELO_MOLECULAR} \rangle$$

donde la regla de producción que define $\langle \text{MODELO_MOLECULAR} \rangle$ pertenece al conjunto de reglas de producción de la \mathcal{GLC} generado para LEMM.

- $\langle \text{OPERACION_CON_MODELO} \rangle$: Las operaciones con modelos moleculares, están definidas por las siguientes reglas de producción:

$$\langle \text{OPERACION_CON_MODELO} \rangle ::= \langle \text{OPERACION} \rangle \text{"("} \langle \text{ID} \rangle \text{")"}$$

$$\langle \text{OPERACION} \rangle ::= \langle \text{GRAFICAR} \rangle \mid \langle \text{PROPIEDADES_FISICAS} \rangle$$

$$\langle \text{GRAFICAR} \rangle ::= \text{"graficar2d"} \mid \text{"graficar3d"}$$

$$\langle \text{PROPIEDADES_FISICAS} \rangle ::= \text{"pesomolecular"}$$

La \mathcal{GLC} para LOMMM se define a continuación:

$$\mathcal{GLC}_{lommm} = (\mathcal{N}_{lommm}, \mathcal{T}_{lommm}, \mathcal{P}_{lommm}, \langle S \rangle)$$

El conjunto de símbolos no terminales \mathcal{N}_{lommm} se define como:

$$\mathcal{N}_{lommm} = \{ \langle S \rangle, \langle \text{SENTENCIA} \rangle, \langle \text{FIN_DE_LINEA} \rangle, \langle \text{DECLARACION_DE_VARIABLE} \rangle, \langle \text{DEFINICION_DE_MODELO} \rangle, \langle \text{OPERACION_CON_MODELO} \rangle, \langle \text{ID} \rangle, \langle \text{TIPO} \rangle, \langle \text{LETRA} \rangle, \langle \text{DIGITO} \rangle, \langle \text{MODELO_MOLECULAR} \rangle, \langle \text{OPERACION} \rangle, \langle \text{GRAFICAR} \rangle, \langle \text{PROPIEDADES_FISICAS} \rangle \}$$

El conjunto de símbolos terminales \mathcal{T}_{lommm} se define como:

$$\mathcal{T}_{lommm} = \{ \text{"inicio"}, \text{"fin"}, \text{"defina"}, \text{"como"}, \text{"("}, \text{"), "}, \text{"A"}, \text{"B"}, \text{"C"}, \text{"D"}, \text{"E"}, \text{"F"}, \text{"G"}, \text{"H"}, \text{"I"}, \text{"J"}, \text{"K"}, \text{"L"}, \text{"M"}, \text{"N"}, \text{"O"}, \text{"P"}, \text{"Q"}, \text{"R"}, \text{"S"}, \text{"T"}, \text{"U"}, \text{"V"}, \text{"W"}, \text{"X"}, \text{"Y"}, \text{"Z"}, \text{"a"}, \text{"b"}, \text{"c"}, \text{"d"}, \text{"e"}, \text{"f"}, \text{"g"}, \text{"h"}, \text{"i"}, \text{"j"}, \text{"k"}, \text{"l"}, \text{"m"}, \text{"n"}, \text{"o"}, \text{"p"}, \text{"q"}, \text{"r"}, \text{"s"}, \text{"t"}, \text{"u"}, \text{"v"}, \text{"w"}, \text{"x"}, \text{"y"}, \text{"z"}, \text{"0"}, \text{"1"}, \text{"2"}, \text{"3"}, \text{"4"}, \text{"5"}, \text{"6"}, \text{"7"}, \text{"8"}, \text{"9"}, \text{"modelo"}, \text{"="}, \text{"("}, \text{"), "}, \text{"graficar2d"}, \text{"graficar3d"}, \text{"pesomolecular"} \}$$

Por último, el conjunto de reglas de producción \mathcal{P}_{lommm} se define como:


```

 $\mathcal{P}_{iommm} = \{ \langle S \rangle ::= \text{"inicio"} \{ \langle \text{SENTENCIA} \rangle \langle \text{FIN\_DE\_LINEA} \rangle \} \text{"fin"},$ 
 $\langle \text{FIN\_DE\_LINEA} \rangle ::= \text{";"} | \text{"."},$ 
 $\langle \text{SENTENCIA} \rangle ::= \langle \text{DECLARACION\_DE\_VARIABLE} \rangle |$ 
 $\langle \text{DEFINICION\_DE\_MODELO} \rangle | \langle \text{OPERACION\_CON\_MODELO} \rangle ,$ 
 $\langle \text{DECLARACION\_DE\_VARIABLE} \rangle ::= \text{"defina"} \langle \text{ID} \rangle \langle \text{"como"} \rangle \langle \text{TIPO} \rangle ,$ 
 $\langle \text{ID} \rangle ::= \langle \text{LETRA} \rangle \{ \langle \text{LETRA} \rangle \langle \text{DIGITO} \rangle \} ,$ 
 $\langle \text{LETRA} \rangle ::= \text{"A"} | \text{"B"} | \text{"C"} | \text{"D"} | \text{"E"} | \text{"F"} | \text{"G"} | \text{"H"} | \text{"I"} | \text{"J"} | \text{"K"} | \text{"L"} | \text{"M"} |$ 
 $\text{"N"} | \text{"O"} | \text{"P"} | \text{"Q"} | \text{"R"} | \text{"S"} | \text{"T"} | \text{"U"} | \text{"V"} | \text{"W"} | \text{"X"} | \text{"Y"} | \text{"Z"} | \text{"a"} | \text{"b"} |$ 
 $\text{"c"} | \text{"d"} | \text{"e"} | \text{"f"} | \text{"g"} | \text{"h"} | \text{"i"} | \text{"j"} | \text{"k"} | \text{"l"} | \text{"m"} | \text{"n"} | \text{"o"} | \text{"p"} | \text{"q"} | \text{"r"} |$ 
 $\text{"s"} | \text{"t"} | \text{"u"} | \text{"v"} | \text{"w"} | \text{"x"} | \text{"y"} | \text{"z"} ,$ 
 $\langle \text{DIGITO} \rangle ::= \text{"0"} | \text{"1"} | \text{"2"} | \text{"3"} | \text{"4"} | \text{"5"} | \text{"6"} | \text{"7"} | \text{"8"} | \text{"9"} ,$ 
 $\langle \text{TIPO} \rangle ::= \text{"modelo"} ,$ 
 $\langle \text{DEFINICION\_DEL\_MODELO} \rangle ::= \langle \text{ID} \rangle \langle \text{"="} \rangle \langle \text{MODELO\_MOLECULAR} \rangle ,$ 
 $\langle \text{OPERACION\_CON\_MODELO} \rangle ::= \langle \text{OPERACION} \rangle \langle \text{"("} \langle \text{ID} \rangle \text{"}")} ,$ 
 $\langle \text{OPERACION} \rangle ::= \langle \text{GRAFICAR} \rangle | \langle \text{PROPIEDADES\_FISICAS} \rangle ,$ 
 $\langle \text{GRAFICAR} \rangle ::= \text{"graficar2d"} | \text{"graficar3d"} ,$ 
 $\langle \text{PROPIEDADES\_FISICAS} \rangle ::= \text{"pesomolecular"} \}$ 

```

5. Gramática Libre de Contexto para el Ambiente de Visualización Molecular

La $\mathcal{G}LC$ para AVISMO requiere de la unión de los conjuntos de símbolos no terminales, de símbolos terminales y de las reglas de producción definidos previamente para las $\mathcal{G}LC$ que generan los lenguajes LEMM y LOMMM, $\mathcal{G}LC_{lemm}$ y $\mathcal{G}LC_{iommm}$ respectivamente. Como se mencionó anteriormente, el axioma de esta gramática será el axioma de $\mathcal{G}LC_{iommm}$, es decir, $\langle S \rangle$. Entonces, la $\mathcal{G}LC$ para AVISMO se define a continuación:

$$\begin{aligned}
 \mathcal{N}_{AVISMO} &= \mathcal{N}_{lemm} \cup \mathcal{N}_{iommm} \\
 \mathcal{T}_{AVISMO} &= \mathcal{T}_{lemm} \cup \mathcal{T}_{iommm} \\
 \mathcal{P}_{AVISMO} &= \mathcal{P}_{lemm} \cup \mathcal{P}_{iommm} \\
 \mathcal{G}LC_{AVISMO} &= (\mathcal{N}_{AVISMO}, \mathcal{T}_{AVISMO}, \mathcal{P}_{AVISMO}, \langle S \rangle)
 \end{aligned}$$

En [Vic09], esta $\mathcal{G}LC$ fue sometida a una serie de transformaciones con el fin de obtener una gramática limpia [Oso07, SG86]. Primeramente se realizó la transformación de las reglas de producción de AVISMO, expresadas en notación EBNF, a reglas más simples, expresadas en notación BNF-no ampliada, con el fin de obtener reglas de producción más fáciles de leer y comprender por una persona, así como más fáciles de utilizar por aplicaciones generadoras de analizadores léxicos y sintácticos, además de presentarse de mejor forma como entrada para los algoritmos de limpieza de gramáticas y transformación a formas normales estandarizadas como las de Chomsky y Greichbach [SG86].

En las reglas de producción de la gramática de AVISMO, expresada en EBNF, los elementos se relacionan por medio de cuatro operadores: llaves ({ , }), corchetes ([,]), paréntesis ((,)) y el operador O lógico (!). Considerando que el último operador es el más sencillo de analizar por una aplicación cualquiera, se expresó la $\mathcal{G}LC$ de AVISMO en BNF-no ampliada, utilizando sólo este operador y obviando los otros tres. Para ello se estableció una equivalencia entre los operadores que se consideran complejos (llaves, corchetes y paréntesis) y el operador O lógico y, de esta manera, se transformó cada regla de producción de la gramática. El resultado de esta transformación es presentando detalladamente en [Vic09].

Seguidamente se procedió a realizar la limpieza de la $\mathcal{G}LC$

de AVISMO, aplicando los algoritmos definidos para cada una de las acciones que se deben llevar a cabo para limpiar una gramática [Oso07, SG86]. Estas acciones consisten en eliminar producciones ϵ (cadena vacía), producciones no generativas, símbolos no terminales, símbolos no alcanzables, recursividad directa y recursividad indirecta.

El proceso completo realizado para la limpieza de la $\mathcal{G}LC$ de AVISMO también es descrito en [Vic09]. La gramática resultante que corresponde a la $\mathcal{G}LC$ limpia de AVISMO, denotada $\mathcal{G}LC'_{AVISMO}$, por definición es equivalente a $\mathcal{G}LC_{AVISMO}$ puesto que ambas generan el mismo lenguaje. Esta gramática se define a continuación:

$$\mathcal{G}LC'_{AVISMO} = (\mathcal{N}'_{AVISMO}, \mathcal{T}'_{AVISMO}, \mathcal{P}'_{AVISMO}, \langle S \rangle)$$

El conjunto de símbolos no terminales \mathcal{N}'_{AVISMO} es:

```

 $\mathcal{N}'_{AVISMO} = \{ \langle \text{MODELO\_MOLECULAR} \rangle, \langle \text{ELEMENTO} \rangle, \langle \text{ENLACE} \rangle,$ 
 $\langle \text{ELEMENTO\_QUIMICO} \rangle, \langle \text{VALENCIA} \rangle, \langle \text{GRUPO\_FUNCIONAL} \rangle,$ 
 $\langle \text{GRUPO\_FUNCIONAL\_INFERIOR} \rangle, \langle \text{GRUPO\_FUNCIONAL\_SUPERIOR} \rangle,$ 
 $\langle \text{COMPUESTO} \rangle, \langle \text{COMPUESTOS} \rangle, \langle \text{MODELO\_GRUPO\_FUNCIONAL} \rangle,$ 
 $\langle S \rangle, \langle \text{SENTENCIA} \rangle, \langle \text{SENTENCIAS} \rangle, \langle \text{FIN\_DE\_LINEA} \rangle, \langle \text{ID} \rangle, \langle \text{TIPO} \rangle,$ 
 $\langle \text{LETRA} \rangle, \langle \text{IDCONT} \rangle, \langle \text{DIGITO} \rangle, \langle \text{OPERACION} \rangle \}$ 

```

El conjunto de símbolos terminales \mathcal{T}'_{AVISMO} es:

```

 $\mathcal{T}'_{AVISMO} = \{ \text{"H"}, \text{"Li"}, \text{"Na"}, \text{"K"}, \text{"Rb"}, \text{"Cs"}, \text{"Fr"}, \text{"Be"}, \text{"Mg"}, \text{"Ca"}, \text{"Sr"}, \text{"Ba"},$ 
 $\text{"Ra"}, \text{"Sc"}, \text{"V"}, \text{"Ti"}, \text{"Zr"}, \text{"Hf"}, \text{"Db"}, \text{"Vn"}, \text{"Nb"}, \text{"Ta"}, \text{"Jr"}, \text{"Cr"}, \text{"Mo"}, \text{"W"}, \text{"Rf"},$ 
 $\text{"Mn"}, \text{"Tc"}, \text{"Re"}, \text{"Bh"}, \text{"Fe"}, \text{"Ru"}, \text{"Os"}, \text{"Hn"}, \text{"Co"}, \text{"Rh"}, \text{"Ir"}, \text{"Mt"}, \text{"Ni"}, \text{"Pd"},$ 
 $\text{"Pt"}, \text{"Cu"}, \text{"Ag"}, \text{"Au"}, \text{"Zn"}, \text{"Cd"}, \text{"Hg"}, \text{"B"}, \text{"Al"}, \text{"Ga"}, \text{"In"}, \text{"Tl"}, \text{"C"}, \text{"Si"}, \text{"Ge"},$ 
 $\text{"Sn"}, \text{"Pb"}, \text{"N"}, \text{"P"}, \text{"As"}, \text{"Sb"}, \text{"Bi"}, \text{"O"}, \text{"S"}, \text{"Se"}, \text{"Te"}, \text{"Po"}, \text{"F"}, \text{"Cl"}, \text{"Br"}, \text{"I"},$ 
 $\text{"At"}, \text{"He"}, \text{"Ne"}, \text{"Ar"}, \text{"Kr"}, \text{"Xe"}, \text{"Rn"}, \text{"."}, \text{";"}, \text{"("}, \text{")"}, \text{"["}, \text{"]"}, \text{"["}, \text{"]"}, \text{"inicio"}, \text{"fin"}, \text{"defina"},$ 
 $\text{"como"}, \text{";"}, \text{"."}, \text{"A"}, \text{"B"}, \text{"C"}, \text{"D"}, \text{"E"}, \text{"F"}, \text{"G"}, \text{"H"}, \text{"I"}, \text{"J"}, \text{"K"}, \text{"L"}, \text{"M"}, \text{"Q"}, \text{"R"}, \text{"T"}, \text{"U"}, \text{"X"},$ 
 $\text{"Z"}, \text{"a"}, \text{"b"}, \text{"c"}, \text{"d"}, \text{"e"}, \text{"f"}, \text{"g"}, \text{"h"}, \text{"i"}, \text{"j"}, \text{"k"}, \text{"l"}, \text{"m"}, \text{"n"}, \text{"o"}, \text{"p"}, \text{"q"}, \text{"r"}, \text{"s"},$ 
 $\text{"t"}, \text{"u"}, \text{"v"}, \text{"w"}, \text{"x"}, \text{"y"}, \text{"z"}, \text{"0"}, \text{"1"}, \text{"2"}, \text{"3"}, \text{"4"}, \text{"5"}, \text{"6"}, \text{"7"}, \text{"8"}, \text{"9"}, \text{"modelo"},$ 
 $\text{"="}, \text{"("}, \text{")"}, \text{"graficar2d"}, \text{"graficar3d"}, \text{"pesomolecular"}, \}$ 

```

El conjunto de reglas de producción \mathcal{P}'_{AVISMO} es:

```

 $\mathcal{P}'_{AVISMO} = \{ \langle S \rangle ::= \text{"inicio"} \langle \text{SENTENCIAS} \rangle \text{"fin"},$ 
 $\langle \text{SENTENCIAS} \rangle ::= \langle \text{SENTENCIA} \rangle \langle \text{FIN\_DE\_LINEA} \rangle \langle \text{SENTENCIAS} \rangle |$ 
 $\langle \text{SENTENCIA} \rangle \langle \text{FIN\_DE\_LINEA} \rangle ,$ 
 $\langle \text{FIN\_DE\_LINEA} \rangle ::= \text{";"} | \text{"."},$ 
 $\langle \text{SENTENCIA} \rangle ::= \text{"defina"} \langle \text{ID} \rangle \langle \text{"como"} \rangle \langle \text{TIPO} \rangle | \langle \text{ID} \rangle \langle \text{"="} \rangle$ 
 $\langle \text{MODELO\_MOLECULAR} \rangle | \langle \text{OPERACION} \rangle \langle \text{"("} \langle \text{ID} \rangle \text{"}")} ,$ 
 $\langle \text{ID} \rangle ::= \langle \text{"A"} \rangle | \langle \text{"B"} \rangle | \langle \text{"C"} \rangle | \langle \text{"D"} \rangle | \langle \text{"E"} \rangle | \langle \text{"F"} \rangle | \langle \text{"G"} \rangle | \langle \text{"H"} \rangle | \langle \text{"I"} \rangle | \langle \text{"J"} \rangle | \langle \text{"K"} \rangle | \langle \text{"L"} \rangle | \langle \text{"M"} \rangle |$ 
 $\langle \text{"N"} \rangle | \langle \text{"O"} \rangle | \langle \text{"P"} \rangle | \langle \text{"Q"} \rangle | \langle \text{"R"} \rangle | \langle \text{"S"} \rangle | \langle \text{"T"} \rangle | \langle \text{"U"} \rangle | \langle \text{"V"} \rangle | \langle \text{"W"} \rangle | \langle \text{"X"} \rangle | \langle \text{"Y"} \rangle | \langle \text{"Z"} \rangle | \langle \text{"a"} \rangle | \langle \text{"b"} \rangle |$ 
 $\langle \text{"c"} \rangle | \langle \text{"d"} \rangle | \langle \text{"e"} \rangle | \langle \text{"f"} \rangle | \langle \text{"g"} \rangle | \langle \text{"h"} \rangle | \langle \text{"i"} \rangle | \langle \text{"j"} \rangle | \langle \text{"k"} \rangle | \langle \text{"l"} \rangle | \langle \text{"m"} \rangle | \langle \text{"n"} \rangle | \langle \text{"o"} \rangle | \langle \text{"p"} \rangle | \langle \text{"q"} \rangle | \langle \text{"r"} \rangle |$ 
 $\langle \text{"s"} \rangle | \langle \text{"t"} \rangle | \langle \text{"u"} \rangle | \langle \text{"v"} \rangle | \langle \text{"w"} \rangle | \langle \text{"x"} \rangle | \langle \text{"y"} \rangle | \langle \text{"z"} \rangle | \langle \text{LETRA} \rangle \langle \text{IDCONT} \rangle ,$ 
 $\langle \text{IDCONT} \rangle ::= \langle \text{"A"} \rangle | \langle \text{"B"} \rangle | \langle \text{"C"} \rangle | \langle \text{"D"} \rangle | \langle \text{"E"} \rangle | \langle \text{"F"} \rangle | \langle \text{"G"} \rangle | \langle \text{"H"} \rangle | \langle \text{"I"} \rangle | \langle \text{"J"} \rangle | \langle \text{"K"} \rangle | \langle \text{"L"} \rangle |$ 
 $\langle \text{"M"} \rangle | \langle \text{"N"} \rangle | \langle \text{"O"} \rangle | \langle \text{"P"} \rangle | \langle \text{"Q"} \rangle | \langle \text{"R"} \rangle | \langle \text{"S"} \rangle | \langle \text{"T"} \rangle | \langle \text{"U"} \rangle | \langle \text{"V"} \rangle | \langle \text{"W"} \rangle | \langle \text{"X"} \rangle | \langle \text{"Y"} \rangle | \langle \text{"Z"} \rangle | \langle \text{"a"} \rangle$ 
 $| \langle \text{"b"} \rangle | \langle \text{"c"} \rangle | \langle \text{"d"} \rangle | \langle \text{"e"} \rangle | \langle \text{"f"} \rangle | \langle \text{"g"} \rangle | \langle \text{"h"} \rangle | \langle \text{"i"} \rangle | \langle \text{"j"} \rangle | \langle \text{"k"} \rangle | \langle \text{"l"} \rangle | \langle \text{"m"} \rangle | \langle \text{"n"} \rangle | \langle \text{"o"} \rangle | \langle \text{"p"} \rangle | \langle \text{"q"} \rangle |$ 
 $\langle \text{"r"} \rangle | \langle \text{"s"} \rangle | \langle \text{"t"} \rangle | \langle \text{"u"} \rangle | \langle \text{"v"} \rangle | \langle \text{"w"} \rangle | \langle \text{"x"} \rangle | \langle \text{"y"} \rangle | \langle \text{"z"} \rangle | \langle \text{LETRA} \rangle \langle \text{IDCONT} \rangle | \langle \text{"0"} \rangle | \langle \text{"1"} \rangle |$ 
 $\langle \text{"2"} \rangle | \langle \text{"3"} \rangle | \langle \text{"4"} \rangle | \langle \text{"5"} \rangle | \langle \text{"6"} \rangle | \langle \text{"7"} \rangle | \langle \text{"8"} \rangle | \langle \text{"9"} \rangle | \langle \text{DIGITO} \rangle \langle \text{IDCONT} \rangle ,$ 
 $\langle \text{LETRA} \rangle ::= \langle \text{"A"} \rangle | \langle \text{"B"} \rangle | \langle \text{"C"} \rangle | \langle \text{"D"} \rangle | \langle \text{"E"} \rangle | \langle \text{"F"} \rangle | \langle \text{"G"} \rangle | \langle \text{"H"} \rangle | \langle \text{"I"} \rangle | \langle \text{"J"} \rangle | \langle \text{"K"} \rangle | \langle \text{"L"} \rangle | \langle \text{"M"} \rangle |$ 
 $\langle \text{"N"} \rangle | \langle \text{"O"} \rangle | \langle \text{"P"} \rangle | \langle \text{"Q"} \rangle | \langle \text{"R"} \rangle | \langle \text{"S"} \rangle | \langle \text{"T"} \rangle | \langle \text{"U"} \rangle | \langle \text{"V"} \rangle | \langle \text{"W"} \rangle | \langle \text{"X"} \rangle | \langle \text{"Y"} \rangle | \langle \text{"Z"} \rangle | \langle \text{"a"} \rangle | \langle \text{"b"} \rangle |$ 
 $\langle \text{"c"} \rangle | \langle \text{"d"} \rangle | \langle \text{"e"} \rangle | \langle \text{"f"} \rangle | \langle \text{"g"} \rangle | \langle \text{"h"} \rangle | \langle \text{"i"} \rangle | \langle \text{"j"} \rangle | \langle \text{"k"} \rangle | \langle \text{"l"} \rangle | \langle \text{"m"} \rangle | \langle \text{"n"} \rangle | \langle \text{"o"} \rangle | \langle \text{"p"} \rangle | \langle \text{"q"} \rangle | \langle \text{"r"} \rangle |$ 
 $\langle \text{"s"} \rangle | \langle \text{"t"} \rangle | \langle \text{"u"} \rangle | \langle \text{"v"} \rangle | \langle \text{"w"} \rangle | \langle \text{"x"} \rangle | \langle \text{"y"} \rangle | \langle \text{"z"} \rangle ,$ 
 $\langle \text{DIGITO} \rangle ::= \langle \text{"0"} \rangle | \langle \text{"1"} \rangle | \langle \text{"2"} \rangle | \langle \text{"3"} \rangle | \langle \text{"4"} \rangle | \langle \text{"5"} \rangle | \langle \text{"6"} \rangle | \langle \text{"7"} \rangle | \langle \text{"8"} \rangle | \langle \text{"9"} \rangle ,$ 
 $\langle \text{TIPO} \rangle ::= \text{"modelo"} ,$ 
 $\langle \text{OPERACION} \rangle ::= \text{"graficar2d"} | \text{"graficar3d"} | \text{"pesomolecular"} ,$ 
 $\langle \text{MODELO\_MOLECULAR} \rangle ::= \text{"H"} | \text{"Li"} | \text{"Na"} | \text{"K"} | \text{"Rb"} | \text{"Cs"} | \text{"Fr"} | \text{"Be"} |$ 
 $\text{"Mg"} | \text{"Ca"} | \text{"Sr"} | \text{"Ba"} | \text{"Ra"} | \text{"Sc"} | \text{"V"} | \text{"Ti"} | \text{"Zr"} | \text{"Hf"} | \text{"Db"} | \text{"Vn"} | \text{"Nb"} |$ 
 $\text{"Ta"} | \text{"Jr"} | \text{"Cr"} | \text{"Mo"} | \text{"W"} | \text{"Rf"} | \text{"Mn"} | \text{"Tc"} | \text{"Re"} | \text{"Bh"} | \text{"Fe"} | \text{"Ru"} | \text{"Os"} |$ 
 $\text{"Hn"} | \text{"Co"} | \text{"Rh"} | \text{"Ir"} | \text{"Mt"} | \text{"Ni"} | \text{"Pd"} | \text{"Pt"} | \text{"Cu"} | \text{"Ag"} | \text{"Au"} | \text{"Zn"} |$ 
 $\text{"Cd"} | \text{"Hg"} | \text{"B"} | \text{"Al"} | \text{"Ga"} | \text{"In"} | \text{"Tl"} | \text{"C"} | \text{"Si"} | \text{"Ge"} | \text{"Sn"} | \text{"Pb"} | \text{"N"} |$ 

```

```

"P" | "As" | "Sb" | "Bi" | "O" | "S" | "Se" | "Te" | "Po" | "F" | "Cl" | "Br" | "I" | "At" |
"He" | "Ne" | "Ar" | "Kr" | "Xe" | "Rn" | <ELEMENTO_QUIMICO> <VALENCIA>
| <ELEMENTO> <GRUPO_FUNCIONAL> | <COMPUESTO> <ELEMENTO> |
<COMPUESTO> <ELEMENTO> <GRUPO_FUNCIONAL> | <COMPUESTO>
<COMPUESTO> <COMPUESTOS>,
<COMPUESTO> ::= "H" | "Li" | "Na" | "K" | "Rb" | "Cs" | "Fr" | "Be" | "Mg" |
"Ca" | "Sr" | "Ba" | "Ra" | "So" | "Y" | "Tl" | "Zr" | "Hf" | "Db" | "V" | "Nb" | "Ta"
| "Jl" | "Cr" | "Mo" | "W" | "Rf" | "Mn" | "Tc" | "Re" | "Bh" | "Fe" | "Ru" | "Os"
| "Hn" | "Co" | "Rh" | "Ir" | "Mt" | "Ni" | "Pd" | "Pt" | "Cu" | "Ag" | "Au" | "Zn"
| "Cd" | "Hg" | "B" | "Al" | "Ga" | "In" | "Tl" | "C" | "Si" | "Ge" | "Sn" | "Pb" |
"N" | "P" | "As" | "Sb" | "Bi" | "O" | "S" | "Se" | "Te" | "Po" | "F" | "Cl" | "Br" |
"I" | "At" | "He" | "Ne" | "Ar" | "Kr" | "Xe" | "Rn" | <ELEMENTO_QUIMICO>
<VALENCIA> | <ELEMENTO> <GRUPO_FUNCIONAL> | <ELEMENTO>
<GRUPO_FUNCIONAL> <ENLACE> | <ELEMENTO> <ENLACE>,
<COMPUESTOS> ::= <COMPUESTO> <COMPUESTOS> | <COMPUESTO>,
<ELEMENTO> ::= "H" | "Li" | "Na" | "K" | "Rb" | "Cs" | "Fr" | "Be" | "Mg" | "Ca" |
"Sr" | "Ba" | "Ra" | "So" | "Y" | "Tl" | "Zr" | "Hf" | "Db" | "V" | "Nb" | "Ta" | "Jl"
| "Cr" | "Mo" | "W" | "Rf" | "Mn" | "Tc" | "Re" | "Bh" | "Fe" | "Ru" | "Os" | "Hn" |
"Co" | "Rh" | "Ir" | "Mt" | "Ni" | "Pd" | "Pt" | "Cu" | "Ag" | "Au" | "Zn" | "Cd" | "Hg"
| "B" | "Al" | "Ga" | "In" | "Tl" | "C" | "Si" | "Ge" | "Sn" | "Pb" | "N" | "P" | "As" |
"Sb" | "Bi" | "O" | "S" | "Se" | "Te" | "Po" | "F" | "Cl" | "Br" | "I" | "At" | "He" | "Ne"
| "Ar" | "Kr" | "Xe" | "Rn" | <ELEMENTO_QUIMICO> <VALENCIA>,
<ELEMENTO_QUIMICO> ::= "H" | "Li" | "Na" | "K" | "Rb" | "Cs" | "Fr" | "Be" |
"Mg" | "Ca" | "Sr" | "Ba" | "Ra" | "So" | "Y" | "Tl" | "Zr" | "Hf" | "Db" | "V" | "Nb" |
"Ta" | "Jl" | "Cr" | "Mo" | "W" | "Rf" | "Mn" | "Tc" | "Re" | "Bh" | "Fe" | "Ru" | "Os"
| "Hn" | "Co" | "Rh" | "Ir" | "Mt" | "Ni" | "Pd" | "Pt" | "Cu" | "Ag" | "Au" | "Zn"
| "Cd" | "Hg" | "B" | "Al" | "Ga" | "In" | "Tl" | "C" | "Si" | "Ge" | "Sn" | "Pb" | "N" |
"P" | "As" | "Sb" | "Bi" | "O" | "S" | "Se" | "Te" | "Po" | "F" | "Cl" | "Br" | "I" | "At" |
"Ar" | "He" | "Ne" | "Ar" | "Kr" | "Xe" | "Rn",
<VALENCIA> ::= "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9",
<GRUPO_FUNCIONAL> ::= <GRUPO_FUNCIONAL_INFERIOR>
<GRUPO_FUNCIONAL_SUPERIOR> | <GRUPO_FUNCIONAL_SUPERIOR>
<GRUPO_FUNCIONAL_INFERIOR> | (" <MODELO_GRUPO_FUNCIONAL>
") | (" <MODELO_GRUPO_FUNCIONAL> ") |
<GRUPO_FUNCIONAL_INFERIOR> ::= (" <MODELO_GRUPO_FUNCIONAL> ") |
<GRUPO_FUNCIONAL_SUPERIOR> ::= (" <MODELO_GRUPO_FUNCIONAL> ") |
<MODELO_GRUPO_FUNCIONAL> ::= <ENLACE> <MODELO_MOLECULAR>
| "H" | "Li" | "Na" | "K" | "Rb" | "Cs" | "Fr" | "Be" | "Mg" | "Ca" | "Sr" | "Ba" |
"Ra" | "So" | "Y" | "Tl" | "Zr" | "Hf" | "Db" | "V" | "Nb" | "Ta" | "Jl" | "Cr" | "Mo"
| "W" | "Rf" | "Mn" | "Tc" | "Re" | "Bh" | "Fe" | "Ru" | "Os" | "Hn" | "Co" | "Rh" |
"Ir" | "Mt" | "Ni" | "Pd" | "Pt" | "Cu" | "Ag" | "Au" | "Zn" | "Cd" | "Hg" | "B" | "Al"
| "Ga" | "In" | "Tl" | "C" | "Si" | "Ge" | "Sn" | "Pb" | "N" | "P" | "As" | "Sb" | "Bi"
| "O" | "S" | "Se" | "Te" | "Po" | "F" | "Cl" | "Br" | "I" | "At" | "He" | "Ne" | "Ar" |
"Kr" | "Xe" | "Rn" | <ELEMENTO_QUIMICO> <VALENCIA> | <ELEMENTO>
<GRUPO_FUNCIONAL> | <COMPUESTO> <ELEMENTO> | <COMPUESTO>
<ELEMENTO> <GRUPO_FUNCIONAL> | <COMPUESTO> <COMPUESTO>
<COMPUESTOS>,
<ENLACE> ::= "-" | "=" | "." | ":" | ";"

```

6. Conclusiones

Se ha presentado la definición de una *GLC* que genera el LF del traductor denominado AVISMO, en función de los conjuntos de símbolos no terminales y terminales, el axioma y el conjunto de reglas de producción. Tras un proceso de transformación de las reglas de producción seguido de la limpieza de la gramática definida inicialmente, se logró obtener una *GLC* fácilmente legible e implementable, lo cual facilitó el desarrollo de AVISMO.

La importancia de la definición de esta gramática para el desarrollo de AVISMO, el cual permite generar gráficos en 2D y 3D a partir de un programa escrito en el LF de AVISMO, radica en que a partir de la misma es posible el dise-

ño e implementación tanto del analizador léxico del traductor, debido a que las expresiones regulares que describen un lenguaje son un caso particular de una *GLC*, así como del analizador sintáctico del traductor, el cual determina si una sentencia pertenece o no a un lenguaje. Mientras que para la química orgánica, AVISMO constituye una herramienta visual alternativa de licencia libre que contribuye en el proceso de enseñanza-aprendizaje en dicha área.

Además, se propuso un nuevo lenguaje de especificación de compuestos orgánicos que elimina las ambigüedades presentes en las formulaciones propuestas por la IUPAC.

Agradecimientos

Se agradece el financiamiento del CDCHTA-ULA bajo el proyecto No. I-1237-10-02-AA.

Referencias

- [ALSU07] AHO A. V., LAM M. S., SETHI R., ULLMAN J. D.: *Compilers: Principles, Techniques, and Tools*. Addison-Wesley, 2007. 2
- [Bre03] BRENNAN R.: *Autómatas y Lenguajes*. Tecnológico de Monterrey, 2003. 4
- [Cho59] CHOMSKY N.: On certain formal properties of grammars. *Information and Control* 2, 2 (1959), 137-167. 2
- [Dür02] DÜRSTELER J. C.: Visualización molecular. *La revista Digital de InfoVis.Net* (Feb 2002). (Inf@Vis!) <http://www.infovis.net/printMag.php?num=76&lang=1> 1
- [Int79] INTERNATIONAL UNION OF PURE AND APPLIED CHEMISTRY: *Nomenclature of Organic Chemistry, Sections A, B, C, D, E, F, and H*. Pergamon Press; 4th edition, 1979. 3
- [Joy98] JOYANES L.: *Programación Orientada a Objetos*. McGraw-Hill; 2da edición, 1998. 5
- [Joy09] JOYANES L.: *Fundamentos de Programación*. McGraw-Hill; 4th edición, 2009. 3
- [Ken04] KENNETH L.: *Construcción de compiladores: principios y prácticas*. Thomson, 2004. 4
- [MB03] MONTILVA J., BARRIOS J.: A component-based method for developing web applications. *Revista Colombiana de Computación* 4, 1 (July 2003), 21-34. 2
- [MK08] MARTZ E., KRAMER T.: World index of molecular visualization resources, 2008. <http://molvis.sdsc.edu/visres/index.html>. 1, 2
- [Oso07] OSORIO C. I. G.: Limpieza de gramáticas y formas normales. Universidad de Burgos, España, 2007. pierrezga.inf.ubu.es/cgosorio/ALeF/UD5/opGram.pdf. 7
- [PPI94] PANICO R., POWELL W. H., IUPAC: *A Guide to Iupac Nomenclature of Organic Compounds Recommendations 1993 (International Union of Pure and Applied Chemistry Organic Chemistry Division)*. Blackwell Science Inc; 2nd edition, 1994. 3
- [SG86] SANCHIS F., GALÁN C.: *Compiladores Teoría y Construcción*. Paraninfo, 1986. 2, 7
- [Vic09] VICUÑA O.: *Desarrollo de un Traductor de Lenguaje de Descripción de Compuestos Químicos a Lenguaje de Visualización Gráfica utilizando el Paradigma de Software Libre*. Proyecto de grado, Universidad de Los Andes, Facultad de Ingeniería, Escuela de Ingeniería de Sistemas, Mérida, 2009. 2, 7