RLA. Revista de Lingüística Teórica y Aplicada
Concepción (Chile), 53 (2), II Sem. 2015, pp. 171-191.
CL ISSN 0033 - 698X

¿POR QUÉ SE USA 'RECURSIÓN' CUANDO SE QUIERE SIGNIFICAR 'AUTO-INCLUSIÓN'?: CLARIFICACIONES CONCEPTUALES SOBRE LA RECURSIÓN EN EL PROGRAMA CHOMSKIANO[1]

WHY 'RECURSION' IS USED WHEN 'SELF-EMBEDDING' IS MEANT?: CONCEPTUAL CLARIFICATIONS ON RECURSION IN THE CHOMSKYAN PROGRAM

SERGIO MOTA
Departamento de Psicología Básica
Universidad Autónoma de Madrid, España
sergio.mota.v@gmail.com

RESUMEN

Este trabajo pretende presentar de forma clara y concisa la (doble) disociación conceptual entre 'recursión' y 'auto-inclusión' tan frecuentemente confundidos en la literatura especializada. Se mostrará que la recursión no implica auto-inclusión y que la auto-inclusión no implica tampoco recursión. El ámbito de discusión será el programa chomskiano, dado que ha sido uno de los ámbitos donde la recursión ha sido peor interpretada, fundamentalmente por sus críticos.

Palabras clave: Recursión, auto-inclusión, regla, estructura, Chomsky.

ABSTRACT

This work intends to show in a clear and concise way that there is a double conceptual dissociation between 'recursion' and 'self-embedding', two terms which are frequently conflated in the specialized literature. It will show that recursion does not imply self-embedding and that self-embedding does not imply recursion either. The discussion

[1] Mi gratitud al profesor Noam Chomsky por su dedicación e interés por los análisis presentados, especialmente en el último punto de este trabajo. Asimismo, quiero expresar mi agradecimiento a los profesores José Manuel Igoa y Luis Eguren, quienes han leído detenidamente el manuscrito y han hecho inteligentes comentarios que han contribuido a mejorar la calidad de este trabajo. Este trabajo se enmarca parcialmente bajo el proyecto PSI2012-37623, financiado por el Ministerio de Economía y Competitividad del Gobierno de España, dada mi estrecha colaboración con el profesor José Manuel Igoa.

will be focused on the Chomskyan program, since this has been one particular domain where recursion has been worse understood, especially by Chomsky's critics.

Keywords: Recursion, self-embedding, rule, structure, Chomsky.

Recibido: 04.12.2014. Aceptado: 19.10.2015.

1. INTRODUCCIÓN

El presente artículo, lejos de querer presentar una detallada exégesis de la obra de Noam Chomsky, se limitará a hacer una breve exposición de la importancia que la noción de recursión ha tenido y tiene en su obra. Pero antes de presentar estos breves apuntes, conviene advertir que la noción de recursión ha recibido diversas interpretaciones en la bibliografía especializada, incluso en aquellos trabajos que tratan de precisar su significado. En esta introducción se expondrá el significado original del concepto de recursión en la Lógica Matemática y la Teoría de la Computabilidad, y se hará un breve repaso del desarrollo de diferentes clases de funciones recursivas, avances que están, como el propio Chomsky reconoce en sus obras, en la base de la formulación del procedimiento generativo/computacional que subyace a la facultad del lenguaje. Esta presentación está motivada porque autores reconocidos en el ámbito de la Lingüística que han trabajado sobre este tópico, como es el caso de Tomalin (2007, 2011), tienden a confundir la noción de recursión con las diferentes clases de funciones formuladas. Por tanto, se entiende que lo que varía son las clases, pero no la noción de recursión como definición por inducción (ver *infra*)[2]. En el caso particular de este autor, esta confusión le ha llevado a formular hasta nueve significados diferentes del concepto de recursión, de los cuales seis tienen que ver con los avances en el campo de la Teoría de la Computabilidad (2011, p. 307). Sin embargo, en dicho campo, el significado original y primario de recursión se aplica exactamente en el mismo sentido a diferentes clases de funciones (cf. Kleene, 1943).

[2] Como Kleene (1943) muestra, entre otros, todas las clases de funciones recursivas –i.e., las primitivas, las generales y las parciales– hacen uso del principio de recursión. Así, el esquema V, que hace uso de tal principio, es empleado para definir las distintas clases de funciones. Además, se emplean los esquemas I-IV –que aluden, respectivamente, a la función sucesor, a las funciones constantes, a las funciones identidad como funciones iniciales, y al esquema de substitución– y el esquema VI en el caso de las generales y las parciales –que alude al esquema de *minimalización*–. La clase de las funciones recursivas generales se cierra bajo I-VI, teniendo que cumplir la función definida mediante el esquema VI la exigencia de que tiene que tener al menos una solución. Al relajar esta exigencia tenemos las funciones recursivas parciales, también definidas mediante I-VI, pero, en este caso, una función definida mediante VI no tiene que tener necesariamente una solución. Un apunte interesante es que el esquema V presenta diversas *versiones*: una versión en la que la inducción se aplica sobre una variable, otra en la que se emplean parámetros, otra en la que se aplica la inducción sobre, al menos, dos variables simultáneamente.

Como es bien sabido, la noción de recursión encuentra su origen en el siglo XIX, cuando Richard Dedekind y Giuseppe Peano emplearon tal noción en su sentido primario para hacer referencia a una definición de funciones y predicados que, como indica Soare (1996, 1999, 2009), se puede expresar del siguiente modo: una función está técnicamente definida por recursión si y sólo si ésta se define para un argumento X haciendo uso de sus propios valores previamente computados (para argumentos menores que X); pudiendo emplearse también funciones previamente definidas (véase también Gödel, 1931; Kleene, 1952, 2002; Cutland, 1980).

En este sentido, en un texto clásico de este campo, Epstein y Carnielli (1989) indican que, en su forma más simple, la definición recursiva de una función f sería como sigue (siendo g una función previamente definida): $f(0) = m$; $f(n+1) = g(f(n))$ (cf. Kleene, 1952, 2002; Boolos y Jeffrey, 1974). Otro ejemplo clásico, en donde se hace uso del principio de recursión con parámetros, es el de la función suma:

$a+0 = a/a+1 = a'$ Def. [caso base]
$a+(b+1) = (a+b)+1$ Def. [paso recursivo]

Una función como ésta es una función que pertenece a la clase de las funciones recursivas *primitivas*, definidas y empleadas por Skolem (1923) y Gödel (1931, 1934). Dado que éstas no agotan todas las clases de funciones totales que pueden definirse por recursión, Gödel (1934) amplió la clase de las funciones recursivas primitivas formulando la clase de las funciones recursivas denominadas *generales*, que incluyen a las primitivas. Una diferencia fundamental entre la clase de las funciones recursivas generales y la clase de las funciones recursivas primitivas es que en las primeras la inducción se aplica, al menos, sobre dos variables simultáneamente. A continuación, se presenta una función recursiva general en donde las funciones ψ y χ están previamente definidas y la función φ es la que se define por recursión (véase Gödel, 1934, p. 69).

$$\varphi(0,y) = \psi(y),$$
$$\varphi(x+1, 0) = \chi(x),$$
$$\varphi(x+1, y+1) = \varphi(x, \varphi(x+1, y)).$$

Seguidamente, se presenta una clase de funciones recursivas que Tomalin no menciona, pero que tienen una enorme relevancia en la Teoría de la Computabilidad, a saber, las funciones recursivas *parciales*.

Esta clase de funciones, definida por Kleene (1938, 1943, 1952) –quien, además, identificó correctamente la clase de funciones computables (Kleene, 1952, pp. 323-348)– debe su nombre al hecho de que una función no necesita estar definida para todas las *n*-tuplas de números naturales que toma como argumentos[3]. Por esta razón, las funciones recursivas parciales incluyen a las primitivas y a las generales, identificándolas como aquellas funciones parciales definidas para *todas* la *n*-tuplas de números naturales tomados como argumentos; esto es, como *funciones recursivas totales*[4]. Conviene subrayar que esta clase de funciones recursivas está más relacionada con los posteriores trabajos en la teoría de la gramática de lo que a simple vista pudiera parecer, dado que como recientemente ha traído Chomsky (2012) a colación, desde un punto de vista conceptual se asume que la recursión es necesariamente infinita, pero las funciones recursivas parciales ponen de manifiesto que una función recursiva puede dar un único valor o incluso ninguno, en caso de que no esté definida para un argumento dado.

Para terminar este breve recorrido por la Teoría de la Computabilidad y la Lógica Matemática, es aconsejable aludir a Wittgenstein, cuyas reflexiones acerca de la noción de regla gramatical se hallan –a juicio del autor del presente trabajo–, en la base misma del concepto de recursión (Mota, 2015)[5]. Así, la definición recur-

[3] Un ejemplo de función parcial es: $f(x) = (x\text{-}1)/2$. Esta función sólo está definida para los números naturales impares tomados como argumentos (cf. Boolos y Jeffrey, 1974; Mota, 2015 –corrigiendo en este ejemplo una errata que aparece allí–). Para una definición más técnica véase Kleene (1938, 1943, 1952).

[4] En Mota (2013, 2014, 2015) se define explícitamente la *Tesis de Kleene*, la cual expone que "una función es computable si y sólo si es una función recursiva parcial (o está definida por el formalismo de Kleene)" (2014, p. 24, nota 3), y la *Tesis de Turing-Kleene* que reza así: "*Una función es computable si y sólo si es una función recursiva parcial (o está definida por el formalismo de Kleene) o, de forma equivalente, si es computable/calculable por una máquina de Turing*" (*Ibid*.). Esta última bien podría substituir a la *Tesis de Church-Turing*, dado que, como se señala, fue Kleene, y no Church (véase Soare, 2009), quien identificó correctamente la clase de las funciones computables.

[5] En Mota (2015) se realiza un tratamiento algo más extenso en relación con la noción de recursión: "Así, desde la lógica matemática, la matemática y la ciencia de la computación, se puede distinguir entre las definiciones inductivas y co-inductivas (que proporcionan los principios duales a los de las primeras). Las primeras proporcionan, o están en la base de, dos principios diferentes que, *a priori*, no hay que confundir, los principios de inducción (que permiten probar ciertas propiedades internas de los objetos tratados, por ejemplo de los números naturales o de las fórmulas proposicionales), y de recursión (que, como se verá, garantizan la buena definición de las funciones recursivas), las segundas proporcionan los principios de co-inducción (que también permiten probar ciertas propiedades internas de los objetos tratados) y de co-recursión (que garantizan la buena definición de funciones a través de procesos). Desde un punto de vista formal, estas definiciones se pueden contextualizar de varias maneras: en la Teoría de Conjuntos, en la Teoría de categorías, entre otras. Sin embargo, no es el objetivo de este trabajo entrar en cada una de estas aplicaciones" (p. 156). En ese trabajo se eligió la teoría axiomática de conjuntos de Zermelo-Fraenkel (ZF) con axioma de elección. Por otro lado, conviene no confundir la definición inductiva, la cual sirve para definir términos como 'número natural', 'objeto', etc., con una definición por recursión (o 'recursiva'). Sólo la segunda, y no la primera, es la empleada en la Lógica Matemática y la Teoría de la Computabilidad para definir funciones y/o predicados (Kleene, 1952, p. 217). Dado que aquí se proporcionan varios ejemplos de definiciones recursivas, también es apropiado presentar un ejemplo de definición inductiva. Así, en el marco de ZF con axioma de elección, una definición inductiva de conjunto inductivo es como sigue: un conjunto I se dice inductivo si (a) el conjunto vacío '$\varnothing$' pertenece a I y (b) siempre que un conjunto X pertenece a I, entonces también el sucesor S(X), obtenido de la unión de X con el conjunto que tiene como único elemento a X, esto es, $S(X) = X \cup \{X\}$. Así, pueden definirse cómodamente el conjunto de los números naturales. Es importante apuntar que en esta definición,

siva, expresada como '$a+0 = a$; $a+(b+1) = (a+b)+1$', es, como Wittgenstein (1975, §163) señala en las *Observaciones Filosóficas*, una regla fundamental del sistema que indica cómo proceder y, como tal, no se puede aseverar o negar, es decir, no son descripciones y, por ello, carece de valor de verdad. Abundando en esta misma idea, Wittgenstein (1974, 36, p. 851) indica en la *Gramática Filosófica* que la definición recursiva "es una regla para la construcción de reglas de sustitución, o también el término general de una serie de definiciones [formas]".

Si bien es verdad que Chomsky no ha reconocido explícitamente su coincidencia con las ideas de Wittgenstein acerca del concepto de regla gramatical (véase por ejemplo Mota, 2014, 2015, para una discusión más amplia sobre este asunto), no es menos cierto que nunca ha sido ajeno al desarrollo de la Teoría de la Computabilidad y la Lógica Matemática y de ello dejó constancia en sus primeros escritos. Así, en *The logical structure of linguistic theory* (1955/1975), Chomsky indica que la Lógica Matemática, en particular la teoría de las funciones recursivas y la metamatemática, fue siendo cada vez más accesible a diferentes disciplinas y los desarrollos en estas áreas proporcionaban herramientas apropiadas para un estudio más preciso del lenguaje natural (p. 39).

Chomsky insiste en varios de sus trabajos, tanto de la primera época como de tiempos recientes, en la importancia de la Teoría de la Computabilidad y la Lógica Matemática. Así, en otro de sus primeros trabajos indica que, al menos desde un punto de vista conceptual, la teoría de la gramática puede ser vista como el estudio de una clase especial de funciones recursivas (1959, p. 149). Una función recursiva no es sino el procedimiento generativo/computacional que subyace a la facultad del lenguaje en la concepción de Chomsky. Mucho más recientemente (Chomsky, 2007a), indica que uno de los principales factores que han impulsado el desarrollo de la Biolingüística ha sido precisamente el trabajo realizado dentro de la Teoría de la Computabilidad, que ha permitido estudiar más seriamente los mecanismos formales de la gramática generativa. En esta misma línea, ha señalado recientemente (Chomsky, 2012) que el estudio del lenguaje-I, entendido como un sistema dotado de infinitud discreta, cae dentro de la Teoría de la Computabilidad, siendo, de este modo, contemplado como un procedimiento computacional que genera recursivamente una cantidad potencialmente infinita de expresiones jerárquicamente estructuradas.

En el siguiente apartado se mostrará cómo la noción de recursión también se ha venido empleando para referirse a las *estructuras* lingüísticas que presentan una determinada organización interna caracterizada por la 'auto-inclusión' (*self-embedding*). No obstante, se argumentará que la noción de recursión aplicada a

un conjunto *no* puede contenerse a sí mismo, con lo que la posibilidad de la auto-inclusión, que caracteriza la auto-referencia tan típica de la paradoja de Russell, queda neutralizada. Ello también es señalado por Boolos (1971, p. 222), cuando indica que ningún conjunto pertenece a sí mismo, puesto que cada conjunto formado en cada fase, contiene sólo a conjuntos formados en fases previas.

las estructuras no se ajusta al significado original del término. Posteriormente, en el punto 3, se discutirá la noción de recursión en el actual Programa Minimista (Chomsky, 1995a), a objeto de dejar claro que la recursión es sólo una propiedad de una regla, de una definición, si se entiende en su sentido original, y que la auto-inclusión no justifica la recursión. En el punto 4 se presentan algunas consecuencias de esta propuesta. Finalmente, se presentarán brevemente las conclusiones finales.

2. LOS SISTEMAS DE PRODUCCIÓN: RECURSIÓN VERSUS AUTO-INCLUSIÓN

En los años 60, Chomsky entendía por 'gramática generativa' un conjunto de reglas que generan cadenas de símbolos (esto es, las sentencias de un lenguaje) a las que se puede dar una descripción estructural (Chomsky, 1965, p. 10). Asimismo, Chomsky y Miller (1963, p. 284) establecen que por 'gramática' entienden un conjunto de reglas que recursivamente especifican las oraciones de un lenguaje. Una definición recursiva formal de tal caracterización es, tal y como ha sido definida en Mota (2014, p. 39), como sigue:

$$R^{(0)'}(s) = s \text{ Def.},$$
$$R^{(n)}{}'(s) = R'R^{(n-1)}{}'(s) \text{ Def.}$$

R está por un conjunto de reglas, *s* por una estructura generada/especificada por esas reglas y *R*(*s*) por una nueva estructura generada/especificada por medio de la aplicación de *R*( ) sobre una estructura previamente generada/especificada *s*.

Esta definición recursiva guarda un gran parecido con los sistemas de producción de Post (1921, 1943, 1944), en los que Chomsky reconoce abiertamente que se apoya (véase 1965). Muy brevemente, el formalismo de Post consiste en un sistema capaz de generar todas las proposiciones de la lógica de enunciados. Así, sea *g* un enunciado de la lógica proposicional y *P* una variable operacional aplicada a dicho enunciado, el sistema produce un enunciado *g'* que sustituye a *g*. Esto se expresa en su sistema de producción normal como sigue: $gP \rightarrow Pg'$ (Post, 1943, p. 199). Su tesis, conocida como la Tesis de Post, establece que un conjunto no vacío (como el de las proposiciones de la lógica de enunciados) es efectivamente numerable si y sólo si es derivado de un sistema de producción (normal) [derivado de su sistema canónico (normal)] (Davis, 1982; Soare, 2009)[6]. Una definición recursiva de un sistema tal puede verse en Mota (2014, p. 31):

[6] Post (1944) también se centró en los conjuntos recursivamente numerables, distinguiendo entre éstos y los conjuntos recursivos. Un conjunto *S* es recursivo si se puede establecer un método efectivo –un procedimiento mecánico finito– para determinar si, por ejemplo, un número entero positivo *n* pertenece o no al conjunto *S*. Un conjunto *S* es recesivamente numerable si existe un procedimiento mecánico efectivo para numerar efectivamente los elementos de *S*.

$P^{(0)}\text{'}(g) = g$ Def.,
$P^{(n)}\text{'}(g) = P\text{'}P^{(n-1)}\text{'}(g)$ Def.

Sin embargo, dentro de la Ciencia Cognitiva del Lenguaje, no sólo se predica recursión de un sistema como el que se presentó más arriba, sino de reglas específicas. Así, siguiendo a Chomsky y Miller (1963), un sistema como $\Sigma \rightarrow \varepsilon\Sigma$; $\Sigma \rightarrow \varepsilon$ no expresa sino reglas de un cálculo que permite generar Σ, ε, $\varepsilon\varepsilon$, $\varepsilon\varepsilon\varepsilon$, lo que equivale a Σ, $R(\Sigma)$, $R(R(\Sigma))$, $R(R(R(\Sigma)))$. Así, para generar $\varepsilon\varepsilon\varepsilon$, el sistema aplica $\varepsilon\Sigma \rightarrow \varepsilon\varepsilon\Sigma$ (lo cual quiere decir que las expresiones de lado izquierdo se rescriben como las del lado derecho), lo que puede formularse como $\varepsilon\varepsilon\Sigma = R(\varepsilon\Sigma)$ (Mota, 2014). Esto último son reglas del tipo ‘$a+b' = (a+b)+1$’ (Ibíd.). Luuk y Luuk (2011) caracterizan de manera un tanto general las reglas recursivas como aquellas en las que todo aquello que aparece en el lado izquierdo de la flecha reaparece en el lado derecho. Sea como fuere, bajo estas premisas, se entiende que una regla como AB→AABB es una regla recursiva.

Hasta aquí, la recursión se ha mostrado como una propiedad de una definición, referida específicamente a sistemas de reglas o de cómputo en su totalidad. Sin embargo, desde hace unos años, el término ‘recursión’ se comenzó a emplear para significar una propiedad de la organización interna de las estructuras lingüísticas. Así, una estructura con *auto-inclusión* se define como aquella en la cual un constituyente (o una estructura A) contiene otro constituyente (o estructura B) del mismo tipo (Pinker y Jackendoff, 2005; Moro, 2008; Karlsson, 2010; Kinsella, 2010). Ejemplos generales de estructuras con auto-inclusión (las llamadas estructuras recursivas) son aquellas que exhiben sintagmas dentro de sintagmas del mismo tipo (por ejemplo, un sintagma nominal incluido en otro cf. Karlsson, 2010, p. 51) o también oraciones de relativo dentro de oraciones de relativo (Ibíd.)[7]. Así, Jackendoff (2011, p. 592) afirma que tanto [A[AB]B] como [[[AB]C]D] son estructuras recursivas, entendiendo siempre que la estructura interna y la externa son del mismo tipo, pues éste es el criterio definitorio del término (cf. Luuk y Luuk, 2011). Sin embargo, recursión y auto-inclusión hacen referencia a propiedades muy diferentes y la recursión no está justificada por la auto-inclusión. A continuación se verá por qué.

Como se acaba de señalar, una regla como AB→AABB es una regla recursiva. Tal regla genera cadenas o secuencias de símbolos como AABB, AAABBB,…, (Luuk y Luuk, 2011, p. 5). Por otro lado, diversos autores (Corballis, 2007; Luuk y Luuk, 2011) señalan que una secuencia como AABB no presenta necesariamente auto-inclusión, dado que puede describirse alternativamente como una

[7] Un ejemplo del primer tipo es “*[la casa de [la sierra]$_{SN}$]$_{SN}$ es de piedra*”, en la que un sintagma nominal (SN) incluye a otro SN. Un ejemplo del segundo tipo, y que se empleará más abajo, es “*el ratón [que el gato [que el perro perseguía]$_{OR}$ mordió]$_{OR}$ corría*”, en donde una oración de relativo (OR) incluye a otra OR.

sucesión de unidades de un tipo (A) seguida del mismo número de unidades de otro tipo (B) (así, [AA][BB]), sin que haya dependencias entre unidades no adyacentes dispuestas jerárquicamente (como en [A[AB]B]). Además, estos autores señalan que una secuencia como AABB, aun cuando presente una estructura con auto-inclusión, no tiene por qué haberse generado necesariamente mediante reglas recursivas. Y a la inversa, aunque la regla AB→AABB sí sea una regla recursiva, la secuencia AAABBB, resultante de su aplicación, no puede juzgarse como recursiva o con auto-inclusión (Ibíd.). Esto compromete, por ejemplo, el gran salto inferencial llevado a cabo por Jackendoff (2011, p. 592) y que queda expresado en los siguientes términos: si se puede establecer el dominio de las estructuras con auto-inclusión (lo que Jackendoff llama *recursión estructural*), entonces se podrá concluir que las reglas que definen este dominio son formalmente recursivas. Sin embargo, tanto [A[AB]B] como [AA][BB] se pueden generar mediante una regla del tipo AB→AABB, de tal manera que si se establece un dominio con estructuras sin auto-inclusión, se podrá concluir que las reglas que definen dicho dominio son recursivas. De ello se desprende que la auto-inclusión no justifica la recursión, y, por tanto, se puede establecer un dominio de recursión formal independientemente de que las estructuras exhiban o no auto-inclusión. En esta misma línea, Fitch (2010, p. 87) afirma que secuencias del tipo AB, ABAB,…, esto es, cadenas de símbolos de las que comúnmente se considera que no presentan inclusión, sino que consisten en secuencias repetidas de AB, son generadas por un sistema como S→AB; S→AB+S, en donde S→AB+S es una regla recursiva; y lo mismo podría decirse del sistema que se presentó más arriba basado en el sistema Σ→εΣ; Σ→ε, el cual genera secuencias o cadenas de símbolos como ε, εε, εεε, de las cuales no se puede decir que presenten auto-inclusión.

De todo lo anterior cabe concluir, por tanto, que la recursión es una propiedad de las reglas y no de las estructuras, en contraste con la extendida opinión de que la recursión es una propiedad de las estructuras. Por consiguiente, cuando se predica recursión de las estructuras, sólo se quiere decir auto-inclusión, esto es, la existencia de constituyentes dentro de constituyentes del mismo tipo.

En relación con este último punto, conviene diferenciar también entre inclusión central (*center-embedding*) y auto-inclusión (*self-embedding*). Así, una estructura exhibe inclusión central cuando un constituyente A está totalmente incluido en B, con elementos no nulos a izquierda y derecha (cf. Chomsky, 1965, p. 12). A este respecto, hay que señalar que para algunos autores (v.g. Karlsson, 2010, p. 51) la inclusión central es la recursión por excelencia. Sin embargo, esto tampoco es correcto, dado que según la concepción "estructural" de la recursión, sólo cabe atribuir esta propiedad a las estructuras que incluyan constituyentes del mismo tipo (véase, por ejemplo, Arsenijević y Hinzen, 2012 o Vicari y Adenzato, 2014, p. 173; trabajos en los que la auto-inclusión se considera una propiedad central de la recursión). Así, una estructura [A[AB]B] no indicaría, para este último grupo de autores, una estructura recursiva *necesariamente*.

Un análisis similar a éste se puede efectuar desde la Teoría de la Computabilidad y la Lógica Matemática. Así, una ecuación como ‘$\varphi(a,x) = \psi(\varphi(a,x\text{-}1))$’ es un paso recursivo formal en el que no hay rastro de auto-inclusión. Esta ecuación muestra que la expresión del lado izquierdo ($\varphi(a,x)$) se puede *reemplazar/substituir/intercambiar* (pero no auto-incluir) por la del lado derecho ($\psi(\varphi(a,x\text{-}1))$), en donde ψ y φ son dos funciones diferentes. En este caso, la recursión se muestra en el hecho de que para definir un nuevo valor (de φ) se hace uso de un valor previamente computado por esa función para un argumento menor, es decir, se hace uso de la definición por recursión, pero ninguna de las expresiones por separado es recursiva, sino, más bien, la regla. De ahí que una función no pueda contenerse a sí misma, esto es, que no pueda ser ella misma su propio argumento (Mota, 2015).

En el siguiente apartado se va a afianzar esta diferencia entre recursión y auto-inclusión desde la perspectiva del actual Programa Minimista, en el que los sistemas de producción han sido reemplazados por un procedimiento generativo/computacional basado en una única operación: *Merge* (traducida como “ensamble”; véase Eguren y Fernández, 2004).

3. LA RECURSIÓN EN EL ACTUAL PROGRAMA MINIMISTA: UNA PROPIEDAD DEL PROCEDIMIENTO MECÁNICO FINITO

Como ha señalado Chomsky en varios textos (véase por ejemplo, Chomsky, 1995a, 2007b), los sistemas de producción basados en reglas de rescritura (ver *supra*) empleados en los años 50 y 60 para explicar la generación de expresiones del lenguaje natural, se fueron eliminando de la teoría hasta que en los años 80, bajo el modelo de *Principios y Parámetros*, la gramática generativa se centró más en los principios y condiciones de buena formación de las expresiones estructuradas (véase, por ejemplo Chomsky, 1980), que en los procedimientos que dan cuenta de su generación.

Años después, y dentro del *Programa Minimista*, Chomsky (1995a) expurga conceptualmente sus propuestas anteriores y establece que un lenguaje-I es un procedimiento generativo que genera recursivamente una ordenación (o serie) infinita de expresiones jerárquicamente estructuradas mediante la operación denominada *Merge*. Así, lo que hace un procedimiento mecánico finito de este tipo (también llamado sistema computacional o procedimiento generativo) es construir recursivamente objetos sintácticos (Chomsky, 1995a, p. 226). Como es bien sabido, las expresiones u objetos sintácticos generados por *Merge* son interpretados por dos sistemas, a saber: el sensorio-motor y el conceptual-intencional (véase para más detalles Hauser, Chomsky y Fitch, 2002; Chomsky, 2006, 2007b, 2008, 2010, 2011).

Lo que interesa aquí es analizar qué hace *Merge*, que como ya se señaló, es generar recursivamente objetos sintácticos. Así, se puede expresar la forma general de *Merge* de la siguiente manera: $[N, O_s, M(O_s)]$ (Mota, 2013, 2014, 2015). N hace referencia a las piezas léxicas (o ítems léxicos) y objetos sintácticos que configuran la numeración (o repertorio seleccionado de objetos que entran en una derivación sintáctica), O_s hace referencia a un objeto sintáctico cualquiera de la serie generada (que puede tomar n-valores: si $n = 1$, entonces $O_s = X$; si $n = 2$, entonces $O_s = X, Y$, y así sucesivamente) y $M(O_s)$ hace referencia a un objeto sintáctico nuevo generado a partir de O_s mediante la aplicación de $M(\,)$ –esto es, *Merge*–.

Como se ha indicado en otros trabajos (Mota, 2013, 2014, 2015), este procedimiento genera/define recursivamente objetos sintácticos, independientemente de su organización interna –esto es, presenten o no auto-inclusión–.

Así, sea la numeración $N = \{$*Juan, canta, muchas, canciones*$\}$, lo que *Merge* hace es tomar dos ítems léxicos (definida aquí como una operación binaria) 'muchas' y 'canciones' y forma el objeto sintáctico {muchas, canciones} = {X,Y}. Mediante otra aplicación, *Merge* genera un nuevo objeto sintáctico {canta, {muchas, canciones}}, formado a partir de un objeto previamente computado ({muchas, canciones}); por último, *Merge* genera {Juan, {canta, {muchas, canciones}}}. Así, y de manera esquemática, se puede ver que lo que *Merge* hace (esto es, generar/definir recursivamente objetos sintácticos, independientemente de su estructura interna, tal como se muestra en la definición formal) queda expresado mediante la siguiente serie: O_s, $M'O_s$, $M'M'O_s$…; en el ejemplo: $O_{s\{muchas,\ canciones\}}$; $M_{canta}'Os_{\{muchas,\ canciones\}}$; $M_{Juan}'M_{\text{canta}}'Os_{\{muchas,\ canciones\}}$. Esto puede definirse recursivamente de manera formal como sigue (Mota, 2013, nota 2; 2014, p. 38; 2015, p. 167):

$$M^{(0)}{}'(O_s) = O_s,\ \text{Def.}$$
$$M^{(n)}{}'(O_s) = M'M^{(n-1)}{}'(O_s)\ \text{Def.}$$

Pero el procedimiento empleado por Chomsky y definido por medio de este sistema de ecuaciones recursivas, no sólo genera, como acaba de hacerse patente, expresiones sin auto-inclusión (como *Juan canta muchas canciones*), sino que también genera estructuras con auto-inclusión como *el ratón que el gato que el perro perseguía mordió corría*, la cual exhibe inclusión central y, además, auto-inclusión, dado que una cláusula de relativo se incluye dentro de otra del mismo tipo. Antes de continuar, conviene advertir que según lo que se acaba de exponer, un dominio sin auto-inclusión también puede ser definido por recursión y, por tanto, como se mostró en el apartado anterior, la auto-inclusión no justifica la recursión y, por ello, ambos conceptos no deberían identificarse.

Dicho esto, la descripción de cómo genera *Merge* la oración de relativo especificativa *El ratón que el gato que el perro perseguía mordió corría* sería, en términos generales y de manera algo más compleja que en el caso anterior, como sigue:

Ensamble de *el* y *perro*: {el, perro}
$[O_{s\{\text{el, perro}\}}]$

Ensamble de *el perro* y *perseguía*: {{el, perro}, perseguía}
$[O_{s\{\{\text{el, perro}\}, \text{perseguía}\}}]$

Ensamble de *que* y *el perro perseguía*: {que, {{el, perro}, perseguía}}
$[M_{\text{que}}{}'O_{s\{\{\text{el, perro}\}, \text{perseguía}\}}]$

Ensamble de *gato* y *que[…]*: {gato, {que, {{el, perro}, perseguía}}}
$[M_{\text{gato}}{}'M_{\text{que}}{}'O_{s\{\{\text{el, perro}\}, \text{perseguía}\}}]$

Ensamble de *el* y *gato que[…]*: {el, {gato, {que, {{el, perro}, perseguía}}}}
$[M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'O_{s\{\{\text{el, perro}\}, \text{perseguía}\}}]$

Ensamble de *el gato que[…]* y *mordió*: {{el, {gato, {que, {{el, perro}, perseguía}}}}, mordió}
$[M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'M_{\{\{\text{el, perro}\}, \text{perseguía}\}}{}'O_{s\ \text{mordió}}]$

Ensamble de *que* y *el gato[…]*: {que, {{el, {gato, {que, {{el, perro}, perseguía}}}}, mordió}}
$[M_{\text{que}}{}'M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'M_{\{\{\text{el, perro}\}, \text{perseguía}\}}{}'O_{s\ \text{mordió}}]$

Ensamble de *ratón* y *que el gato[…]*: {ratón, {que, {{el, {gato, {que, {{el, perro}, perseguía}}}}, mordió}}}
$[M_{\text{ratón}}{}'M_{\text{que}}{}'M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'M_{\{\{\text{el, perro}\}, \text{perseguía}\}}{}'O_{s\ \text{mordió}}]$

Ensamble de *el* y *ratón que el gato[…]*: {el, {ratón, {que, {{el, {gato, {que, {{el, perro}, perseguía}}}}, mordió}}}}
$[M_{\text{el}}{}'M_{\text{ratón}}{}'M_{\text{que}}{}'M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'M_{\{\{\text{el, perro}\}, \text{perseguía}\}}{}'O_{s\ \text{mordió}}]$

Ensamble de *el ratón[…]* y *corría*: {{el, {ratón, {que, {{el, {gato, {que, {{el, perro}, perseguía}}}}, mordió}}}}, corría}
$[M_{\text{el}}{}'M_{\text{ratón}}{}'M_{\text{que}}{}'M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'M_{\{\{\text{el, perro}\}, \text{perseguía}\}}{}'M_{\text{mordió}}{}'O_{s\ \text{corría}}]$

Por otra parte, Chomsky (2007b, p. 6; 2008, p. 139) ha insistido en que el modo de operar (cómo procede o puede implementarse de manera abstracta) de este procedimiento generativo consiste en aplicar iterativamente la operación

Merge[8]. Así, la aplicación iterativa queda descrita mediante la siguiente serie, que muestra, en cada paso, las *n* veces que *Merge* se aplica de forma sucesiva, lo cual puede generar todo tipo de expresiones (con auto-inclusión o no) del lenguaje natural: $M^{(0)}, M^{(1)}, M^{(2)}, \ldots, M^{(n)}$. Vemos, una vez más, que las estructuras con auto-inclusión no justifican la propiedad de la recursión.

Así, en el caso de la oración *Juan canta muchas canciones* (una oración que no presenta constituyentes dentro de constituyentes del mismo tipo y, por tanto, sin auto-inclusión a tal nivel), la aplicación iterativa es como sigue:

$M_{\text{muchas}}{}'O_{s\ \text{canciones}} = Os_{\{muchas,\ canciones\}}$ lo que equivale a $\{A,B\} = C$;
$[M_{\text{canta}}{}'Os_{\{muchas,\ canciones\}}] = \{C,D\} = E$;
$M_{Juan}{}'M_{\text{canta}}{}'Os_{\{muchas,\ canciones\}} = \{E,F\} = G$
En donde la secuencia completa es:
$\{A,B\} = C$; $\{C,D\} = E$; $\{E,F\} = G$; que se resume en $M^{(3)}{}'(A)$

Esto indica que la operación *Merge* se ha aplicado tres veces sucesivamente (iterativamente) sobre su último resultado generado.

Por su parte, en el caso de la oración *el ratón que el gato que el perro perseguía mordió corría* (la cual presenta auto-inclusión) la generación iterativa sería como sigue (se da por hecho el ensamble de *el* y *perro*):

$[O_{s\{\{\text{el, perro}\},\ \text{perseguía}\}}]$; $\{A,B\} = C$

$[M_{\text{que}}{}'O_{s\{\{\text{el, perro}\},\ \text{perseguía}\}}]$; $\{C,D\} = E$

$[M_{\text{gato}}{}'M_{\text{que}}{}'O_{s\{\{\text{el, perro}\},\ \text{perseguía}\}}]$; $\{E,F\} = G$

$[M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'O_{s\{\{\text{el, perro}\},\ \text{perseguía}\}}]$; $\{G,H\} = I$

$[M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'M_{\{\{\text{el, perro}\},\ \text{perseguía}\}}{}'O_{s\ \text{mordió}}]$; $\{I,J\} = K$

$[M_{\text{que}}{}'M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'M_{\{\{\text{el, perro}\},\ \text{perseguía}\}}{}'O_{s\ \text{mordió}}]$; $\{K,L\} = M$

$[M_{\text{ratón}}{}'M_{\text{que}}{}'M_{\text{el}}{}'M_{\text{gato}}{}'M_{\text{que}}{}'M_{\{\{\text{el, perro}\},\ \text{perseguía}\}}{}'O_{s\ \text{mordió}}]$]; $\{M,N\} = P$

[8] Es interesante apuntar cómo Wittgenstein distingue entre un procedimiento recursivo e iterativo. Así, siguiendo a Wittgenstein (1974, 32), éste dice que es claro que debe haber una "prueba" iterativa que "transmita la idea de que así debe ocurrir con todos los números"; esto es, una regla que concuerda con la inducción: $f(1) = a+b$; $f(1) \times (a+b) = (a+b)^2 = f(2)$; $(a+b)^n = f(n)$; $f(n) \times (a+b) = f(n+1)$. Así "una vez que se tiene la inducción, todo ha terminado" (Ibíd.). De este modo, la inducción matemática puede expresarse mediante una prueba iterativa.

$[M_{\text{el}}\,{}'M_{\text{ratón}}\,{}'M_{\text{que}}\,{}'M_{\text{el}}\,{}'M_{\text{gato}}\,{}'M_{\text{que}}\,{}'M_{\{\{\text{el, perro}\},\text{ perseguía}\}}\,{}'O_{s\ \text{mordió}}]$; $\{P,Q\} = R$

$[M_{\text{el}}\,{}'M_{\text{ratón}}\,{}'M_{\text{que}}\,{}'M_{\text{el}}\,{}'M_{\text{gato}}\,{}'M_{\text{que}}\,{}'M_{\{\{\text{el, perro}\},\text{ perseguía}\}}\,{}'M_{\text{mordió}}\,{}'O_{s\ \text{corría}}]$; $\{R,S\} = T$

Cuya secuencia completa es:

$\{A,B\} = C$; $\{C,D\} = E$; $\{E,F\} = G$; $\{G,H\} = I$; $\{I,J\} = K$; $\{K,L\} = M$; $\{M,N\} = P$; $\{P,Q\} = R$; $\{R,S\} = T$; que se resume en $M^{(9)}{}'(A)$

En este caso, *Merge* se aplica nueve veces (dando por hecho el ensamble de *el* y *perro*) de manera sucesiva sobre el último resultado producido, generando iterativamente una expresión jerárquicamente estructurada como la empleada en el ejemplo.

Como en el caso expuesto en la sección precedente, encontramos que la recursión es independiente de la auto-inclusión, en tanto en cuanto se puede definir por recursión todo objeto sintáctico con independencia de su organización interna. Asimismo, se puede implementar (de manera abstracta) el procedimiento generativo/computacional propuesto por Chomsky de manera iterativa, tal como ha quedado expuesto aquí, de tal modo que procediendo iterativamente se pueden generar estructuras sin y con auto-inclusión. En suma, la auto-inclusión no es una propiedad central de la recursión, como se ha venido proponiendo de forma reiterada[9].

4. ALGUNAS CONSECUENCIAS DE ESTA PROPUESTA

En este apartado se busca exponer algunas de las consecuencias que se pueden derivar al asumir esta propuesta, la cual adopta un punto de vista wittgensteiniano. En primer lugar, se va a analizar si la eventual existencia de lenguas sin estructuras recursivas (cf. Everett, 2005, 2009) no sería un argumento en contra de la recursividad como parte de la Gramática Universal (GU) (o como propiedad esencial de la facultad estrecha del lenguaje). Según el análisis realizado, que una lengua, como puede ser el Pirahã, no exhiba estructuras recursivas, esto es, estructuras con auto-inclusión, no es un argumento en contra de la recursión como parte de la GU, o como propiedad de la facultad del lenguaje (en sentido estrecho). Lo que el análisis precisamente muestra es que tales estructuras son, *simplemente*, estructuras con auto-inclusión, y, por tanto, no tienen que ver con la propiedad de la recursión, en tanto en cuanto esta última caracteriza el procedimiento

[9] Un trabajo muy reciente que sostiene que la auto-inclusión sí es una propiedad central de la recursión es el de Vicari y Adenzato (2014).

generativo/computacional. El problema surge –a juicio de quien escribe– de la confusión entre propiedades y niveles de análisis. Nótese que aunque se reconozca un esquema universal aplicable a las estructuras, a saber, el conocido esquema de [especificador-[núcleo-complemento]], el análisis no cambia nada, pues la recursión sigue aplicándose al procedimiento, mientras que tal esquema se aplica a las estructuras. Por tanto, desde el análisis efectuado no se deriva ninguna confusión a este respecto.

Con todo, esto puede conllevar un problema: qué hecho en el mundo podría contar como argumento en contra de la recursividad; en otras palabras, cuáles son las condiciones de falsabilidad de la idea de recursividad, si es que las hay. Esta cuestión, sumamente interesante, está lejos de presentar un problema real. Así, lo primero que se debe plantear es hasta qué punto el problema de la recursión es un problema empírico. Desde la perspectiva que se defiende en este trabajo, no se trata de un problema eminentemente empírico, sino matemático o formal. En este sentido, se considera que Kenny (1990) expuso la cuestión de manera magistral al proponer que la mente es "la capacidad de adquirir habilidades intelectuales" (Kenny, 1990, p. 204). Para este autor, hay que distinguir entre poseedores y vehículos. El poseedor de la habilidad lingüística es un ser humano y el vehículo "es la parte de su poseedor en virtud de la cual éste es capaz de ejercer la habilidad… algo concreto y más o menos tangible" (*op.cit*., p. 205). Se puede hacer una distinción entre las personas (el poseedor), la mente (conjunto de capacidades) y el cerebro (el vehículo de tales capacidades). Evidentemente, las personas y sus cerebros son objetos físicos, pero no así la mente, que es un conjunto de capacidades. Conviene tener presente también que tal distinción no es una distinción metafísica/ontológica, sino conceptual, en virtud de la cual se definen 'mente' y 'vehículo'. Para ilustrar la cuestión, Kenny propone que pensemos en una calculadora como analogía de lo que se acaba de exponer: el fisiólogo estaría a la par con el ingeniero electrónico, mientras que el psicólogo está en una posición análoga a la del matemático, pues su papel consiste en intentar descubrir el algoritmo que usa la calculadora (1990, p. 206). La calculadora tiene la habilidad de computar un algoritmo y su análisis exige una investigación matemática, más que electrónica. El aparataje electrónico sería el vehículo.

Sin embargo, si se atiende al *naturalismo metodológico* que defiende Chomsky, el lenguaje, entendido como un órgano *mental* o un sistema biológico, ha de ser considerado como un objeto real, esto es, como el resto de objetos del mundo, y su estudio ha de ser abordado desde una perspectiva naturalista, esto es, como las ciencias naturales estudian otros objetos del mundo. Bajo este naturalismo metodológico, tal y como se acaba de exponer, Chomsky hace una aproximación a la mente bajo la cual considera el lenguaje como un elemento del mundo natural (1995b, p. 1). Chomsky subraya aquí que usa el término 'mente' y 'mental' sin ninguna connotación metafísica, estando a la par de términos como 'químico', 'óptico' o 'eléctrico'. Tales términos seleccionan ciertos aspectos del mundo como foco de investigación, entendiendo por 'mente' los aspectos mentales del mundo.

Desde la perspectiva biolingüística, nos dice Chomsky (2006), el lenguaje es un componente de la mente, entendiendo 'mente' en el sentido de los científicos del siglo XVIII, de tal manera que, no siendo posible postular un problema mente/cuerpo coherente, sólo se pueden considerar los aspectos del mundo denominados mentales como el resultado de una organización orgánica tal como la del cerebro (p. 173; véase también 2005, p. 2).

Estas consideraciones generales han sido algo más precisadas en un interesante trabajo recientemente publicado. En dicho trabajo, Berwick, Friederici, Chomsky y Bolhuis (2013) señalan que el lenguaje humano está asentado sobre un mecanismo computacional particular realizado neurológicamente (p. 90). En este primer apunte, es interesante notar que el mecanismo computacional no es sólo una herramienta construida para caracterizar el lenguaje de manera abstracta, sino que se postula que tal procedimiento mecánico finito está realizado neurológicamente. Es decir, tal procedimiento está a la par que, por ejemplo, las neuronas; son objetos del mundo. Por decirlo de otra manera, dicho procedimiento no sólo es una clase especial de funciones recursivas (ver *supra*), sino que también es un objeto del mundo.

Como se señaló unas líneas más arriba, el procedimiento mecánico finito basado en *Merge* genera objetos sintácticos que son interpretados por dos sistemas: el sensorio-motor (encargado entre otras cosas de la externalización) y el intencional-conceptual (que hace referencia, entre otras cuestiones, al pensamiento y a la planificación de la acción) (Chomsky, 2011, p. 269). Obviamente, en el plano neurológico, tal mecanismo generativo/computacional debe diferenciarse de ambos sistemas de interfaz. Berwick et al. (2013, p. 93) presentan diferentes regiones identificadas tanto con el mecanismo computacional como con los sistemas de interfaz. Así, dicho mecanismo está sustentado (o realizado neurológicamente) por el área de Brodmann 44 y el córtex temporal superior posterior. El sistema sensorio-motor, por su parte, estaría sustentado por el córtex premotor y el córtex temporal superior. Finalmente, el procesamiento semántico (propio del sistema conceptual-intencional) está sustentado por el área de Brodmann 45 y por el córtex frontal inferior, así como por porciones del córtex temporal. Por tanto, cada uno de estos sistemas consisten en las mencionadas regiones particulares del cerebro, conectadas vía tractos específicos (véase Berwick et al., 2013 para más detalles).

El estudio del lenguaje en este nivel neurológico es, siguiendo a Kenny, el estudio del vehículo, del cerebro. El mecanismo computacional (cuyo estudio apropiado es una investigación matemática) no consiste en estas o aquellas regiones cerebrales. Tal procedimiento es una construcción matemática que no supone el descubrimiento de ningún objeto mental del mundo natural. El estudio del cerebro, por su parte, sí es el estudio de un objeto del mundo natural, susceptible de una metodología naturalista. Eso sí, el estudio se realiza en otro nivel de análisis.

A modo de analogía, se puede pensar que en una ciencia como la física, que describe y predice con cierto éxito los acontecimientos de la realidad circundante, ha de considerarse que si una teoría es verdadera, entonces *deben* existir las entidades referidas por los términos de la teoría. De ahí que, de acuerdo con Putnam (1972, p. 57), algunos autores, como por ejemplo Quine, se hayan comprometido con la existencia de entidades matemáticas, dado que la cuantificación sobre tales entidades (i.e., números, funciones, conjuntos) es indispensable tanto en la ciencia física como en la ciencia formal. Por otro lado, el paso de la cuantificación sobre objetos matemáticos a la afirmación de su existencia (un paso nada obvio) es un ejemplo de la aplicación del criterio ontológico de Quine (véase Alemán, 2011 para más detalles).

Así pues, una cosa es que se construya un sistema formal, un cálculo, y se lleve a cabo un análisis matemático del mismo, y otra cosa completamente diferente es que (I) se lo postule como un objeto de una realidad (empírica o platónica), y (II) se entienda que ambos niveles de análisis, a saber, el lógico-matemático y el empírico, en este caso, no sólo están al mismo nivel, sino que son lo mismo; esto es, las entidades físicas y matemáticas existen en la misma realidad espacio-temporal y, por tanto, se las aborda mediante una metodología naturalista. Esta concepción realista del mecanismo formal propuesto por Chomsky parece estar detrás de la idea de que dicho mecanismo es un objeto del mundo natural.

Sin embargo, una cosa es decir que el lenguaje es un sistema biológico y otra muy diferente decir que la caracterización abstracta que se deriva de la formalización planteada por Chomsky sea un objeto real del mundo natural; es decir, que tal formalización describa un objeto del mundo natural. Tal formalismo es, en cierto sentido, autónomo con respecto a cualquier realidad, en el sentido de que no es descriptivo de la misma. En este sentido, en el momento en el que se construye un formalismo se están constituyendo tanto los signos lógico-matemáticos empleados en éste como las reglas para su uso dentro del sistema (por ejemplo, definidos unos axiomas, un sistema formal o procedimiento mecánico finito puede calcular en un número finito de pasos otras verdades lógicas mediante la aplicación de operaciones siguiendo un conjunto de reglas, pudiendo ordenarse tales verdades en series; cf. supra). De este modo, los objetos mentales como el mecanismo generativo/computacional propuesto por Chomsky no se descubren, sino que se crean, se construyen y se constituyen mediante las reglas gramaticales (en sentido wittgensteiniano). Por ello, el carácter esencial del sistema formal definido por Chomsky no es su carácter descriptivo, sino constitutivo o constructivo, como se acaba de indicar.

Adicionalmente, surge la duda de cuáles serían las condiciones en que podría asignarse validez o realidad psicológica a la recursividad. Esta propuesta es compatible con un enfoque no psicológico del lenguaje, esto es, con la tesis de que el lenguaje no es un *objeto* mental. En este caso, ¿cuál sería el estatus de la recursividad? Relacionado con esto último, conviene recordar que una regla gramatical no niega ni asevera nada y, por tanto, no describe ninguna realidad. Así, por ejemplo, se podría distinguir entre:

(a) La expresión "2+2" se puede intercambiar con la expresión "(2+1)+1".
(b) El sujeto S usa la regla 'R'.
(c) El sujeto S usa la regla 'R′'.

(a) no es un enunciado empírico y, por tanto, las propiedades constituidas por dicha regla gramatical no dependen de ninguna realidad; son autónomas frente a ésta y se constituyen por la regla y, por tanto, dentro del sistema. Por su parte, los enunciados (b) y (c) son enunciados empíricos y, por tanto, lo que se confirma o desconfirma son tales enunciados, no las reglas 'R' o 'R′'. Dichas reglas, así como (a), no *dicen* nada, no describen ninguna realidad.

Continuando con el análisis mostrado unas líneas más arriba, la recursión no tiene realidad psicológica en el sentido de que su investigación es matemática y, por tanto, no hay que postular la existencia de un mundo matemático poblado de entidades independientes de nosotros. Dicho de otra manera, la recursión no es una propiedad surgida de la relación entre entidades matemáticas existentes independientemente. Como se indicó antes, la investigación formal del procedimiento mecánico finito propuesto por Chomsky no implica necesariamente ningún tipo de criterio ontológico por medio del cual sea necesario postular la existencia de determinadas entidades. Por el contrario, el carácter constitutivo o constructivo del sistema generativo/computacional pone de manifiesto que es una herramienta o instrumento útil para realizar una caracterización abstracta de la facultad del lenguaje, proponiendo un procedimiento que permite computar en un número finito de pasos un resultado u *output* lingüístico dado un *input* lingüístico. Pero, hay que insistir, que su investigación no es psicológica, si por 'investigación psicológica' se entiende una investigación experimental, pues tal procedimiento no es un descubrimiento del mundo empírico (ni, por supuesto, es un descubrimiento de un mundo platónico). Antes bien, es una creación o invención matemática.

De esto se desprende que el estatus de la recursión no es el de una propiedad del mundo empírico o platónico, sino una propiedad interna a un cálculo creado o inventado. Dicho de otra manera, es una propiedad de una técnica diseñada para definir y calcular y, por tanto, operar de una determinada manera. Entendida así, un experimento no debería proponerse descubrirla, sino aclarar cómo se aplica para caracterizar un comportamiento lingüístico. En suma, la recursión no es un objeto que descubrir en el mundo, sino una herramienta que, en el mejor de los casos, se puede aplicar.

5. CONCLUSIONES

En el presente trabajo se ha tratado de poner de relieve que la literatura especializada confunde de manera recurrente la recursión con la auto-inclusión. Tal como se analizó, el sentido original y primario de 'recursión', entendida como propiedad de una regla, no puede identificarse con la auto-inclusión y esta última no justifica la primera. Una vez que Chomsky tomó el concepto de recursión de la Teoría de la Computabilidad y la Lógica Matemática, y propuso que tal propiedad era constitutiva de su procedimiento generativo, muchos autores comenzaron a aplicar la recursión como una propiedad de las estructuras. Sin embargo, tal como se espera haber demostrado, la recursión es una propiedad de una definición, de una regla, y la auto-inclusión, propiedad que caracteriza a las estructuras, no justifica tampoco en la Ciencia Cognitiva la propiedad de la recursión, siendo ambas propiedades claramente independientes.

REFERENCIAS

Alemán, A. (2011). *Lógica, matemáticas y realidad*. Madrid: Tecnos.

Arsenijević, B. & Hinzen, W. (2012). On the abscence of X-within-X recursion in human grammar, *Linguistic Inquiry*, 43, 423-440.

Berwick, R., Friederici, A.D., Chomsky, N. & Bolhuis, J. (2013). Evolution, brain, and the nature of language, *Trends in Cognitive Science*, 17, 89-98.

Boolos, G. (1971). The iterative conception of set, *The Journal of Philosophy*, 68, 215-231.

Boolos, G. & Jeffrey, R. (1974). *Computability and logic*. Cambridge: Cambridge University Press.

Chomsky, N. (1955/1975). *The logical structure of linguistic theory*. Nueva York: Plenum Press.

Chomsky, N. (1959). On certain formal properties of grammars. *Information and Control*, 2, 137-167.

Chomsky, N. (1965). *Aspects of the theory of syntax*. Cambridge. MA: MIT Press.

Chomsky, N. (1980). *Rules and representations*. Nueva York: Columbia University Press.

Chomsky, N. (1995a). *The minimalist program*. Cambridge. MA: MIT Press.

Chomsky, N. (1995b). Language and nature, *Mind*, 104, 1-61.

Chomsky, N. (2005). Three factors in language design, *Linguistic Inquiry*, 36, 1-22.

Chomsky, N. (2006). *Language and mind*. Cambridge: Cambridge University Press.

Chomsky, N. (2007a). Of minds and language, *Biolinguistics*, 1, 9-27.

Chomsky, N. (2007b). Approaching UG from below. En U. Sauerland & H. M. Gärtner (Eds.), *Interfaces + Recursion = Language?* (pp. 1-30). Berlin: Mouton.

Chomsky, N. (2008). On phases. En R. Freidin, C. Otero, & M. L. Zubizarreta (Eds.), *Foundational issues in linguistic theory* (pp. 133-166). Cambridge. MA: MIT Press.

Chomsky, N. (2010). Some simple evo-devo theses: How true might they be for language? En R. Larson, V. Déprez & H. Yamakido (Eds.), *The evolution of human language* (pp. 45-62). Cambridge: Cambridge University Press.

Chomsky, N. (2011). Language and other cognitive systems. What is special about language?. *Language Learning and Development,* 7, 263-278.

Chomsky, N. (2012). Some core contested concepts. *Actas de la 25th Annual CUNY Conference on Human Sentence Processing* (pp. 1-18).

Chomsky, N. & Miller, G. (1963). Introduction to the formal analysis of natural languages. En R.D. Luce, R.R. Bush & E. Galanter (Eds.), *Handbook of mathematical psychology*, Vol. II (pp. 269-322). New York: John Wiley.

Corballis, M. C. (2007). Recursion, language and starlings, *Cognitive Science*, 31, 69-704.

Cutland, N. (1980). *Computability: an introduction to recursive function theory*. Cambridge: Cambridge University Press.

Davis, M. (1982). Why Gödel didn't have Church's Thesis. *Information and Control*, 54, 3-24.

Eguren, L. & Fernández Soriano, O. (2004). *Introducción a una sintaxis minimista*. Madrid: Gredos.

Epstein, R. & Carnielli, W. (1989). *Computability: computable functions, logic, and the foundations of mathematics*. Pacific Grove, CA: Wadsworth & Brooks/Cole.

Everett, D. (2005). Cultural constraints on grammar and cognition in Pirahã, *Current Anthropology*, 46(4), 621-646.

Everett, D. (2009). Pirahã culture and grammar: a response to some criticisms, *Language*, 85(2), 405-442.

Fitch, T. (2010). Three meanings of recursion: Key distinctions for biolinguistics. En R. Larson, V. Déprez & H. Yamakido (Eds.), *The evolution of human language* (pp. 73-90). Cambridge: Cambridge University Press.

Gödel, K. (1931). On formally undecidable propositions of the Principia Mathematica and related systems. I. In M. Davis (Ed.), *The undecidable* (pp. 4-38). New York: Raven Press.

Gödel, K. (1934). On undecidable propositions of formal mathematical systems. In M. Davis (Ed.), *The undecidable* (pp. 39-74). New York: Raven Press.

Hauser, M., Chomsky, N. & Fitch, T. (2002). The faculty of language: What is, who has it, and how did it evolve?, *Science*, 298, 1569-1579.

Jackendoff, R. (2011). What is the human language faculty? Two views, *Language*, 87, 586-624.

Karlsson, F. (2010). Syntactic recursion and iteration. En H. van der Hulst (Ed.), *Recursion and human language* (pp. 43-67), Berlín: De Gruyter Mouton.

Kenny, A. (1990). *El legado de Wittgenstein*. Madrid: Siglo XXI.

Kinsella, A. (2010). Was recursion the key step in the evolution of the human language faculty? En H. van der Hulst (Ed.), *Recursion and human language* (pp. 179-191) , Berlín: De Gruyter Mouton.

Kleene, S. C. (1938). On notation for ordinal numbers, *The Journal of Symbolic Logic*, 3, 150-5.

Kleene, S. C. (1943). Recursive predicates and quantifiers, *Transactions of the American Mathematical Society*, 53, 41-73.

Kleene, S. C. (1952). *Introduction to metamathematics*. Amsterdam: North-Holland Publishing.

Kleene, S. C. (2002). *Mathematical logic*. Mineola, NY: Dover Publications.

Luuk, E. & Luuk, H. (2011). The redundancy of recursion and infinity for natural language, *Cognitive Processing*, 12, 1-11.

Moro, A. (2008). *The boundaries of Babel*. Cambridge, MA: MIT Press.

Mota, S. (2013). La propiedad de la recursión en el "Tractatus Logico-Philosophicus" de Wittgenstein y su relación con la Teoría de la Computabilidad y la Lógica Matemática, 17, *Observaciones Filosóficas.* Disponible en http://www.obervacionesfilosoficas.net/lapropiedaddelarecursion.htm

Mota, S. (2014). La historia y la gramática de la recursión: una precisión desde la obra de Wittgenstein, *Pensamiento y Cultura*, 17, 20-48.

Mota, S. (2015). Sobre el concepto de recursión y sus usos, *Praxis Filosófica*, *40*, 153-181.

Pinker, S. & Jackendoff, R. (2005). The faculty of language: What's special about it?, *Cognition*, 95, 201-236.

Post, E. (1921). Introduction to a general theory of elementary propositions, *American Journal of Mathematics*, 43, 163-185.

Post, E. (1943). Formal reductions of the general combinatorial decision problem, *American Journal of Mathematics*, 65, 197-215.

Post, E. (1944). Recursively enumerable sets of positive integers and their decision problems. I. En M. Davis (Ed.), *The undecidable* (pp. 305-337). Nueva York: Raven Press.

Putnam, H. (1972). *Philosophy of logic*. Londres: George Allen and Unwin.

Skolem, T. (1923). The foundations of elementary arithmetic established by means of the recursive mode of thought, without the use of apparent variables ranging over infinite domains. En J. Van Heijenoort (Ed.), *From Frege to Gödel. A source book in mathematical logic, 1879-1931* (pp. 302-333). Cambridge, MA: Harvard University Press.

Soare, R. (1996). Computability and recursion, *The Bulletin of Symbolic Logic*, 2, 284-321.

Soare, R. (1999). The history and concept of computability. En E.R. Griffor (Ed.), *Handbook of computability theory* (pp. 3-36,). Amsterdam: North-Holland Publishing.

Soare, R. (2009). Turing oracles machines, online computing, and three displacements in computability theory, *Annals of Pure and Applied Logic*, 160, 368-399.

Tomalin, M. (2007). Reconsidering recursion in syntactic theory, *Lingua*, 117, 1784-1800.

Tomalin, M. (2011). Syntactic structures and recursive devices: A legacy of imprecision, *Journal of Logic, Language and Information*, 20, 297-315.

Vicari, G. & Adenzato, M. (2014). Is recursive language-specific? Evidence of recursive mechanisms in the structure of intentional action, *Consciousness and Cognition*, 26, 169-188.

Wittgenstein, L. (1974). *Philosophical grammar*. Oxford: Blackwell.

Wittgenstein, L. (1975). *Philosophical remarks*. Oxford: Blackwell.