

Un Mecanismo Constructivista para el Aprendizaje de Anticipaciones en Sistemas Acoplados Agente-Ambiente

Filipo Studzinski Perotto¹, Jean-Christophe Buisson², Luis Otávio Alvares¹

¹ Universidade Federal do Rio Grande do Sul – Instituto de Informática
Porto Alegre, Brasil
{fsperotto, alvares}@inf.ufrgs.br

² Institut de Recherche en Informatique – INPT– ENSSEIHT
Toulouse, France
{jean-christophe.buisson}@enseiht.fr

Abstract. This paper presents CALM (Constructivist Anticipatory Learning Mechanism), an agent learning mechanism based on a constructivist approach. It is designed to deal dynamically and interactively with environments which are at the same time partially deterministic and partially observable. CALM can discover environmental regularities by constructing more abstract or high-level representational concepts. The created world model is represented as the structure of a factored POMDP, and it is used to define a police of actions, what increases the agent performance. We also present CAES (*Coupled Agent Environment System*), an agent architecture based on the notions of situatedness, embodiment, and intrinsic motivation.

Palabras Clave: IA Constructivista, Aprendizaje de Estructura en POMDP factorizado, Construcción de Modelos de Mundo.

1. Introducción

Este artículo presenta el CALM, un mecanismo de aprendizaje para agentes artificiales, basada en el paradigma constructivista. La concepción constructivista de la inteligencia fue definitivamente traída al campo de la IA a principios de los años 90 por el trabajo pionero de Gary Drescher (1991), incorporando conceptos de la teoría psicológica de Jean Piaget (1957).

CALM es desarrollado para descubrir regularidades en el caso parcialmente determinista, esto significa que él identifica transformaciones deterministas dentro de mundos no deterministas. CALM representa el conocimiento por *esquemas* (estructuras de anticipación elementales), que tienen la forma $[contexto]+[acción]=[expectativa]$. El mecanismo funciona incrementalmente, entonces el agente aprende al mismo tiempo que tiene que relacionarse con el ambiente. CALM también es capaz de tratar con características ambientales no observables, induciendo propiedades ocultas. Esta característica constituye un paso más hacia la inteligencia abstracta.

De otra parte, la investigación en IA hoy en día está más dirigida a los modelos de cognición situada, y de motivación intrínseca, utilizando afectividad y emociones. Por eso, este artículo presenta también la arquitectura CAES (*Coupled Agent Environment*

System), que reúne las dos características, y soporta el mecanismo CALM como su sistema cognitivo.

El mundo real es un ambiente relativamente estructurado, pero muy complejo, y la transición de la inteligencia sensoriomotor a la inteligencia simbólica es el aspecto principal para explicar como los seres humanos son capaces de tratarlo con éxito (Piaget, 1957). El problema es el mismo para un agente artificial situado, que tiene que aprender incrementalmente las regularidades relacionándose con el mundo.

En ambientes complejos, las propiedades 'macroscópicas' especiales surgen de las interacciones funcionales de elementos 'microscópicos', y generalmente estas características emergentes no están presentes en cualquiera de las partes que lo generan. Los fenómenos salientes en esta clase de ambiente tienden a estar relacionados con objetos o procesos de alto nivel, y en este caso es claramente inadecuado representar el mundo solo en términos de elementos sensoriomotores primitivos (Drescher, 1991).

Un agente inteligente (humano o artificial) que vive en estas condiciones tiene que tener la posibilidad de sobrepasar los límites de la percepción sensorial directa, organizando el universo en términos de conceptos más abstractos. El agente tiene que ser capaz de descubrir regularidades de alto nivel en la dinámica del ambiente, pero eso no es posible si está encerrado en un vocabulario representativo rígido.

CALM es capaz de construir un modelo para describir su universo, y crear elementos abstractos para representar propiedades inobservables.

2. Sistema Acoplado Agente-Ambiente

Trabajos importantes, como (Varela et al., 1991), (Beer, 1995), (Barandiaran; Moreno, 2006), discuten la necesidad del uso, por parte de la IA, de arquitecturas más estrechamente relacionadas a la naturaleza, reajustando las relaciones entre agente y ambiente. El problema del paradigma clásico de la IA simbólica es su excesivo cognitivismo, distante de la noción de agente situado y encarnado, desconsiderando la posibilidad de regulación dinámica por *feedback* sensoriomotor, y cayendo en el conocido problema del enraizamiento de los símbolos (*symbol grounding problem*).

En este artículo presentamos CAES, una arquitectura en la cual agente y ambiente son sistemas dinámicos parcialmente abiertos y acoplados, inspirada en (Beer, 1995). Un sistema puede deformar continuamente la otra trayectoria del otro, pero sin controlarlo de una manera absoluta. El ambiente es solo parcialmente observable por la percepción sensorial del agente, y solo parcialmente sujeto a ser transformado por sus acciones. El agente (Δ) se relaciona con el ambiente (ξ), formando un sistema global (\mathcal{A}). Del mismo modo, el agente es formado por dos subsistemas: cuerpo (β) y mente (μ). El cuerpo realiza la mediación entre la mente y el ambiente. El ambiente impone una situación (s) al agente, que responde con una actuación (a). La situación es percibida por sensores del cuerpo y comunicada a la mente a través de una señal de percepción (p). Algunas sensaciones que vienen del propio cuerpo son también añadidas a la percepción, informando la mente sobre el estado interno del agente. La mente administra en parte las acciones que serán ejecutadas sobre el ambiente a través

de una señal de control (c), que puede también influir sobre las propiedades internas del cuerpo. Esta arquitectura es presentada en la figura 1.

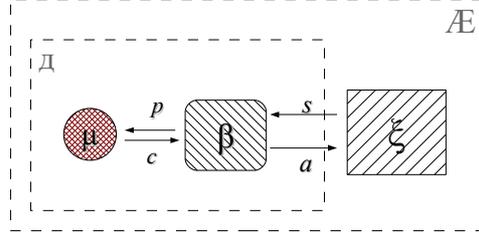


Fig. 1: el cuerpo entre la mente y el ambiente.

3. Representación del Ambiente

Se supone que un agente tiene motivaciones, que de algún modo representan sus objetivos. Clásicamente, un agente tiene que construir de manera autónoma una política para maximizar la obtención de sus objetivos. La estrategia basada en modelos separa el problema en dos partes: (a) la construcción de un modelo de mundo, en el cual se apoya (b) la construcción de una política de acciones.

Hay dos modos comunes de describir un ambiente: basado en estados, o basado en propiedades. Para un problema de control, el modelo es descrito como un *Proceso de Decisión de Markov* (MDP). Cuando no se revelan directamente los estados al agente, el modelo se vuelve un MDP parcialmente observable (POMDP).

Un MDP se define como $\mathcal{G} = \{Q, A, \delta, r\}$ donde Q es un conjunto de estados, A es un conjunto de acciones del agente, $\delta: Q \times A \rightarrow Q$ es una función de transición, que describe como los estados cambian según las acciones, y $r: Q \rightarrow \mathfrak{R}$ es la función de recompensa.

En un POMDP, diferentemente, los estados no están directamente accesibles para el agente, en este caso $\mathcal{G} = \{Q, A, O, \delta, \gamma, r\}$, donde O es un conjunto de observaciones, y $\gamma: Q \rightarrow O$ es una función de observación, que aporta alguna información perceptiva relacionada con el estado subyacente actual del ambiente, pero que no lo identifica por completo. Las funciones de observación, de recompensa y de transición pueden ser estocásticas.

Sin embargo, cuando un agente situado tiene que aprender un modelo de mundo basado únicamente en su experiencia, este tipo de representación extensiva se hace problemática porque requiere la enumeración completa de los estados, y su tamaño crece exponencialmente según el número de sensores del agente. Por eso, trabajos más recientes usan representaciones factorizadas, (Boutilier et al., 2000), (Guestrin et al., 2003), (Degris et al., 2006, 2008), (Strehl, 2007) donde los estados son definidos por un conjunto de propiedades o variables.

Un ambiente basado en propiedades puede ser expresado por un FPOMDP, como $\xi = \{F, A, \tau\}$ donde F es un conjunto de propiedades, formadas por $F_{(p)}$, el subconjunto de propiedades observables, y $F_{(h)}$, el subconjunto de propiedades ocultas, A es un conjunto de acciones del agente, y $\tau: F_1 \times F_2 \times \dots \times F_{|F|} \times A \rightarrow F_i$ es un conjunto de

funciones de transformación, una para cada propiedad F_i en F , describiendo las transiciones según la situación y las acciones que el agente ejecuta.

La descripción factorizada del ambiente (ξ) tiene algunas ventajas sobre la descripción basada en estados (\mathcal{E}). En primer lugar porque promueve una relación más clara entre el ambiente y la percepción del agente. En general asumimos que el agente tiene un sensor para cada propiedad observable del ambiente. En segundo lugar, porque la identidad de los estados queda distribuida en las propiedades. De esta manera, es posible representar estados generalizados y, por consiguiente, hacer descripciones compactas de las funciones de transformación. Representaciones factorizadas evitan la enumeración explícita de estados.

4. Tipos de Ambiente

Adoptamos la descripción factorizada (ξ), basada en propiedades, definiendo 3 ejes para caracterizar tipos diferentes de ambientes. El primero eje representa el nivel de determinismo (∂), y es equivalente a la proporción de transformaciones deterministas en τ sobre el número total de transformaciones. Un ambiente es dicho parcialmente determinista si es situado entre estos dos extremos del eje ($0 < \partial < 1$). Por ejemplo, cuando $\partial = 0.5$, la mitad de las transformaciones en τ es determinista.

El segundo eje es el nivel de accesibilidad (ω), que representa el grado de acceso perceptivo al ambiente. Es equivalente a la proporción de propiedades observables en F sobre el número total de propiedades. Si $\omega = 1$ entonces el ambiente es dicho completamente observable, lo que significa que el agente tiene sensores para observar directamente todas las propiedades del ambiente. En este caso no hay ninguna confusión perceptiva, y el agente siempre sabe cual es la situación actual. Si $\omega < 1$, entonces el ambiente es dicho parcialmente observable, y el agente ya no es capaz de identificar la situación actual solo en términos de su percepción.

En otras palabras, los ambientes parcialmente observables presentan algunas propiedades determinantes a un modelo de mundo coherente que no pueden ser directamente percibidas por el agente. Tales ambientes pueden parecer arbitrariamente complejos y no deterministas en la superficie, pero ellos realmente pueden ser deterministas y previsibles con respecto a elementos subyacentes inobservables (Holmes; Isbell, 2006).

Finalmente, el nivel de concentración de relevancia (ϕ) indica cuán estructurado es el ambiente. Es equivalente al inverso del número medio de propiedades necesarias para representar cada regularidad del ambiente, dividido por el número total de propiedades en F . Cuán mayor es ϕ , más compacto se puede representar la función de transformación (τ). En otras palabras, los niveles más altos de ϕ significan que cada fenómeno en la dinámica del ambiente se explica por unas pocas causas (solo un pequeño subconjunto de propiedades muy relevantes para cada predicción).

5. Mecanismo de Aprendizaje

En esta sección presentamos el mecanismo CALM, que es desarrollado para modelar la interacción de un agente con su ambiente, mismo cuando este ambiente es parcialmente observable y parcialmente determinista, o sea, donde $\omega < 1$, $\partial < 1$ y $\phi < 1$.

CALM trata de construir un conjunto de esquemas para representar regularidades percibidas que ocurren en el ambiente durante sus interacciones. Cada esquema Ξ es formado por tres partes, representando una regularidad del tipo $p \wedge c \rightarrow p'_i$. El vector de contexto (p) tiene cada uno de sus elementos unidos con un sensor, y corresponde a la señal de percepción en la arquitectura CAES (sección 4). El vector de acción (c) está conectado con los efectores, y corresponde al señal de control. La expectativa (p'_i) representa el valor esperado para un sensor específico en el próximo instante.

En un esquema específico, el vector de contexto representa el conjunto de situaciones equivalentes donde el esquema es aplicable. El vector de acción representa un conjunto de acciones similares que el agente puede realizar en el ambiente. El vector de expectativa representa el resultado esperado después de ejecutar la acción en el contexto dado. Cada vector de elemento puede asumir cualquier valor en un intervalo distinto definido por el respectivo sensor o efector.

Algunos elementos en estos vectores pueden tomar el valor indeterminado (#). Por ejemplo, un elemento conectado con un sensor binario debe tener uno de tres valores: verdadero, falso o indeterminado ('1', '0', '#'). Tanto en los vectores de acción como de contexto, '#' representa algo ignorado, o sea, una condición irrelevante para la anticipación. Pero en la expectativa, '#' significa que el elemento no es previsible.

El valor indeterminado generaliza el esquema porque permite desconsiderar algunas propiedades y así representar un conjunto de situaciones. Por ejemplo, un esquema que tiene el vector de contexto '100#' es capaz de asimilar las situaciones compatibles '1000' y '1001'. El uso de valores indeterminados hace posible la construcción de un árbol de esquemas (ψ). Relaciones de generalización y especialización dirigen su topología (similar a árboles de decisión o árboles de discriminación). El nodo de la raíz representa la situación más generalizada, en la cual los vectores de contexto y acción son completamente indeterminados. La adición de un nivel en el árbol especializa un elemento, creando una rama donde el valor indeterminado es sustituido por valores definidos diferentes. Esta especialización ocurre en el vector de contexto o en el vector de acción. Las hojas del árbol son los esquemas, que hacen la anticipación.

El contexto en el cual el agente se encuentra en un momento dado (percibido por sus sensores) es aplicado en el árbol, excitando todos los esquemas que tienen contextos compatibles. Este proceso define un conjunto de esquemas excitados, cada uno aconsejando una acción diferente para hacer en la situación actual. CALM elegirá uno para activar, y entonces la acción definida por el esquema activado será ejecutada por los efectores del agente. Los esquemas son aquellos nodos del árbol que tienen el contexto y acción más específicos, en las hojas de cada rama.

6. Métodos de Aprendizaje

El proceso de aprendizaje es un refinamiento del conjunto de esquemas. El agente se hace más adaptado a su ambiente como una consecuencia de esto. Después de cada situación experimentada, CALM verifica si el resultado (contexto percibido en el instante después de la acción) está en conformidad con la expectativa del esquema activado. Si la anticipación falla, el error entre el resultado y la expectativa sirve como parámetro para corregir el árbol o ajustar el esquema.

En la topología del árbol de esquemas, los vectores de contexto y acción son considerados juntos. Este vector concatenado identifica el nodo en el árbol, que crece sus ramas por sucesivas diferenciaciones. El árbol tiene un sólo esquema inicial. Este esquema de raíz tiene el vector de contexto completamente general (sin cualquier diferenciación, '###') y la expectativa determinada, creado en la primera situación experimentada, como un espejo del resultado directamente observado después de la acción.

El vector de contexto será gradualmente especializado por la diferenciación. En ambientes más complejos, el número de características es enorme, y, en general, solo algunas de ellas son relevantes para identificarse una situación ($1 > \phi \gg 0$).

La expectativa puede ser vista como una etiqueta en cada esquema, y esto representa la anticipación predicha para una propiedad cuando el esquema es activado. El agente tiene dos alternativas cuando la expectativa falla, en un modo de hacer el conocimiento compatible con la experiencia, la primera alternativa trata de dividir el dominio del esquema, creando nuevos esquemas, con contextos más especializados. A veces eso no es posible y la solución es indeterminar la expectativa del esquema.

Cinco métodos básicos realizan el aprendizaje en el mecanismo CALM: diferenciación, diferenciación abstracta, ajuste, anticipación abstracta e integración. Cuando se crea un nuevo esquema, su expectativa empieza totalmente determinada (como un espejo del resultado de su primera activación).

La *diferenciación* es un mecanismo necesario porque a un esquema responsable de un contexto demasiado general le cuestan hacer anticipaciones precisas. Si un esquema general no trabaja bien, el mecanismo lo divide en nuevos esquemas. De hecho, el método de diferenciación toma un esquema inestable y lo cambia en subárboles de dos niveles. El esquema original se transforma en el nodo padre de este subárbol, conservando el contexto del esquema original. Los niños, que son los nuevos esquemas, tienen sus identificadores un poquito más especializados que su padre. Ellos atribuyen un valor a algún elemento indeterminado, dividiendo el alcance del esquema original. Cada uno de estos nuevos esquemas se encarga de una parte del conjunto de situaciones. De esta manera, el conocimiento correcto anterior permanece conservado, distribuido en los nuevos esquemas, y la situación discordante es aislada y tratada como un contexto específico. La diferenciación es el método responsable para hacer crecer el árbol, donde cada nivel representa la introducción de alguna restricción en la definición del esquema.

El algoritmo tiene que elegir un *elemento diferenciador*, que puede ser en el vector de contexto o en el vector de acción. Este diferenciador tiene que separar la situación

responsable por el desequilibrio de las demás, y el algoritmo lo elige conforme a su capacidad de disminuir la entropía.

Si no hay diferenciadores perceptivos, CALM puede suponer la existencia de alguna propiedad oculta del ambiente, que podría ser capaz de diferenciar la situación. Esta propiedad inobservable será representada como un *elemento sintético*. Cuando un elemento sintético es creado, se lo incluye en el vector de contexto de los esquemas como un nuevo término posible. Al mismo tiempo, se crea un nuevo árbol de esquemas para descubrir su dinámica. Los elementos sintéticos suponen la existencia de algo más allá de la percepción sensorial, que puede ser útil para explicar una situación desequilibrada.

De esta manera, tratando con ambientes parcialmente observables, CALM tiene dos desafíos adicionales: (a) deducir la existencia de propiedades inobservables, que representará por elementos sintéticos, y (b) incluir estos nuevos elementos en su modelo anticipador.

Para descubrir la dinámica de una cosa que no se puede observar directamente, tiene que reparar en la información histórica, y captar esta dinámica a través de fenómenos indirectos. Para los casos completamente deterministas, Holmes e Isbell (2006) han demostrado que es siempre posible encontrar trozos suficientemente pequeños de historia para distinguir e identificar todos los estados subyacentes en un D-POMDP.

La *diferenciación abstracta* crea un nuevo elemento sintético booleano, ampliando el vector de contexto posible de los esquemas. Inmediatamente, este elemento es usado para diferenciar la situación incoherente en el esquema desequilibrado. El método atribuye valores arbitrarios a este elemento en cada esquema diferenciado. Estos valores representan la presencia o la ausencia de alguna condición inobservable, necesaria para determinar la predicción correcta en la situación dada.

Una vez que un elemento sintético es creado, puede ser usado en siguientes diferenciaciones. Un nuevo elemento sintético será creado solo si los existentes están ya saturados. Para evitar el problema de crear nuevos elementos sintéticos infinitamente, CALM puede hacerlo solo hasta un límite predefinido.

El elemento sintético no tiene que ver con ninguna percepción sensorial. Por consiguiente, su valor no puede ser directamente observado. Este hecho puede colocar el agente en situaciones ambiguas, donde no es posible saber si una condición relevante pero inobservable (representada por un elemento sintético) está presente o ausente. Al principio, el valor de un elemento sintético es verificado solo a posteriori (es decir después de la ejecución de la acción en una situación ambigua). Una vez que la acción se pasa y el resultado consecuente es verificado, entonces el agente puede mirar hacia atrás y deducir cual era la real situación afrontada en el instante anterior (deshaciendo la ambigüedad).

Descubrir el valor de un elemento sintético después que se pasa la circunstancia donde esta información era necesaria puede parecer poco útil, pero de hecho, esta deducción retrasada es la información necesaria al método de *anticipación abstracta*. Si la propiedad inobservable representada por este elemento sintético tiene un comportamiento regular, entonces el mecanismo puede retropropagar el valor deducido para el esquema activado en el instante anterior en el árbol que trata este elemento sintético. El valor deducido será incluido como una nueva anticipación en este esquema.

Finalmente, cuando algún esquema falla y no es posible diferenciarlo, ni perceptiva, ni abstractamente, entonces CALM ejecuta el método de *ajuste*. Este método indetermina la expectativa del esquema, suponiendo que no hay ninguna regularidad determinista a partir de la situación representada con relación al elemento anticipado, y que la transformación es imprevisible.

Los ajustes sucesivos pueden revelar algunas diferenciaciones innecesarias, entonces, después de un ajuste, CALM verifica la posibilidad de reagrupar algunos esquemas. Así, el método de *integración* busca en un mismo subárbol dos esquemas que tengan expectativas equivalentes y que aborden contextos diferentes. En este caso, eliminase la diferenciación, y los esquemas son integrados en uno solo.

El trayecto realizado por cada esquema, después que está creado, es desarrollar nuevas hojas más especializadas, o sufrir una indeterminación de su expectativa, y entonces una posible integración con otros. Este movimiento sigue hasta el punto en que el árbol alcanza un estado donde solo permanecen aquellos esquemas que realmente representan las regularidades deterministas del ambiente.

7. Problema Experimental

Para ejemplificar el funcionamiento del método propuesto usaremos el problema *flip*, que también es usado por (Holmes; Isbell, 2006). El agente vive en un universo de 2 estados subyacentes, con 3 acciones (*l*, *r*, *u*) y 2 percepciones (0, 1). El agente no tiene ninguna percepción directa para saber cual es el estado subyacente actual. La percepción es '1' cuando el estado se cambia, sino es '0'. La acción '*u*' no cambia el estado, mientras que las acciones '*l*' y '*r*' causan una transición determinista al estado izquierdo o derecho, respectivamente. El problema *flip* está ilustrado en la figura 2.

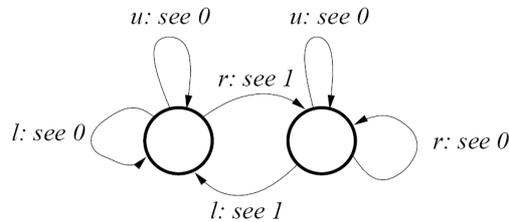


Fig. 2: el problema *flip*.

CALM es capaz de solucionar este problema. En primer lugar intenta predecir la observación siguiente en función de su acción y observación. Sin embargo, CALM rápidamente descubre que la observación perceptiva no es suficiente al modelo para hacer anticipaciones correctas. De este modo, CALM crea un nuevo elemento sintético que será capaz de representar los estados izquierdo (\clubsuit) y derecho (\diamond).

En una ejecución típica, los primeros pasos de la aprendizaje para el problema *flip* no revelan la existencia de una propiedad oculta. Estos movimientos son: "*r1, u0, l1, r1, l1, u0, r1*". Sin embargo, después de la secuencia "*r0, r0, l1, l0, l0, u0, l0, r1, u0, r0*", el elemento oculto es creado y su anticipación es definida. La figura 3 muestra la solución final.

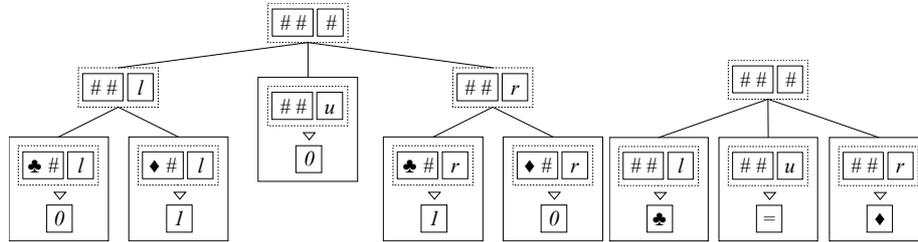


Fig. 3: árboles finales.

8. Trabajos Relacionados

CALM es un mecanismo original que permite a un agente crear incrementalmente un modelo de mundo durante el curso de su interacción. Este trabajo es la continuidad de nuestro trabajo anterior (Perotto; Alvares, 2006, 2007), ampliado para tratar con ambientes parcialmente observables.

El trabajo pionero en AI constructivista ha sido presentado por Drescher (1991). Él presentó la primera arquitectura de agente constructivista (llamado el mecanismo esquemático), que aprende un modelo de mundo por un análisis estadístico exhaustivo de la correlación entre todos los elementos de contexto observados antes de cada acción, combinada con todas las transformaciones resultantes. Drescher también ha abordado la descubierta de propiedades ocultas.

El mecanismo esquemático es un modelo reconocido, sin embargo, sin garantías teóricas de convergencia, y presentando un costo computacional muy restrictivo. La necesidad de recursos de espacio y tiempo aumenta exponencialmente con el tamaño del problema. Sin embargo, muchos otros investigadores han presentado modelos alternativos inspirados por Drescher, como (Chaput, 2004) y (Holmes; Isbell, 2005). Trabajos más recientes han cambiado los términos adecuándose al problema de la descubierta de estructura en POMDPs o en FMDPs, como (Holmes; Isbell, 2006) (Degris et al., 2006, 2008).

Nuestro mecanismo (CALM) se diferencia de estos trabajos anteriores porque limitamos el problema al descubrimiento de regularidades deterministas en ambientes parcialmente deterministas, y de esta manera, podemos poner en práctica métodos de inducción directos y más eficientes. Este acercamiento presenta un coste computacional bajo, y esto permite que el agente aprenda incrementalmente y encuentre regularidades de alto nivel. Además, nuestro trabajo ha incluido el descubrimiento de propiedades ocultas, posibilitando su utilización en ambientes parcialmente observables, construyendo la estructura de un FPOMDP.

9. Conclusiones

El mecanismo CALM puede proporcionar a un agente la capacidad de adaptarse a su ambiente de manera autónoma, porque es capaz de construir incrementalmente el conocimiento para representar las regularidades deterministas observadas durante su

interacción, mismo tratándose de universos parcialmente deterministas. También trata con ambientes parcialmente observables, descubriendo regularidades de alto nivel. La estrategia es hacer la inducción y la predicción de propiedades inobservables, representadas por elementos sintéticos.

Los elementos sintéticos permiten al agente sobrepasar el límite de las regularidades instantáneas y sensoriomotoras. Con estas nuevas capacidades, CALM se hace capaz de construir términos más abstractos para representar el universo, y describir su propia realidad en niveles más complejos.

CALM puede ser muy eficiente para construir modelos en ambientes parcialmente pero muy deterministas ($1 > \partial \gg 0$), parcialmente pero muy observables ($1 > \omega \gg 0$), y cuando el ambiente es bien estructurado ($1 > \phi \gg 0$). Varios problemas reales presentan estas características.

Actualmente, estamos mejorando el mecanismo CALM para que pueda formar secuencias de acción encadenando esquemas, utilizando planificadores conocidos para representaciones factorizadas. Los trabajos futuros incluyen una extensión para ambientes no deterministas, ruidosos, y continuos.

Referencias

1. Barandiaran, X.; Moreno, A. On what makes certain dynamical systems cognitive. *Adaptive Behavior*, v.14, n.2, p.171-185 (2006)
2. Beer, R.D. A dynamical systems perspective on agent-environment interactions. *Artificial Intelligence*, v.72, p.173-215 (1995)
3. Boutilier, C.; Dearden, R.; Goldszmidt, M. Stochastic dynamic programming with factored representations. *Artificial Intelligence*, v.121, n.1-2, p.49-107 (2000)
4. Chaput, H. *The Constructivist Learning Architecture*. PhD Thesis. University of Texas (2004)
5. Degris, T.; Sigaud, O.; Wuillemin, P.-H. Learning the Structure of Factored Markov Decision Processes in Reinforcement Learning Problems. In: *23° ICML, proceedings*. ACM: Pittsburgh. p.257-264 (2006)
6. Drescher, G. *Made-Up Minds: A Constructivist Approach to Artificial Intelligence*. Cambridge: MIT Press (1991)
7. Guestrin, C.; Koller, D.; Parr, R.; Venkataraman, V. Efficient Solution Algorithms for Factored MDPs. *JAIR*, v.19, p.399-468 (2003)
8. Holmes M. and Isbell C. Schema Learning: Experience-based Construction of Predictive Action Models. *Advances in Neural Information Processing Systems*, v.17. (2005)
9. Holmes, M. and Isbell, C. Looping Suffix Tree-Based Inference of Partially Observable Hidden State. In: *23° ICML, proceedings*. ACM: Pittsburgh, p.409-416 (2006)
10. Perotto, F.S., Alvares, L.O. Incremental Inductive Learning in a Constructivist Agent. In: *26° SGAI, proceedings*. London: Springer-Verlag, p.129-144 (2006)
11. Perotto, F.S.; Alvares, L.O.; Buisson, J.-C. Constructivist Anticipatory Learning Mechanism (CALM): Dealing with Partially Deterministic and Partially Observable Environments. In: *7° EpiRob, proceedings*. New Jersey: Lund University. p.117-127 (2007)
12. Piaget, J. *Construction of Reality in the Child*. London: Routledge & Kegan Paul (1957)
13. Strehl, A. L., Diuk, C., & Littman, M. L. Efficient structure learning in factored-state MDPs. In: *22° AAAI, proceedings*. New York: AAAI Press. p.645-650 (2007)
14. Varela, F.; Thompson, E.; Rosch, E. *The Embodied Mind: cognitive science and human experience*. Cambridge, MA: MIT Press (1991)