

Aplicación de procesos Markovianos para recomendar acciones pedagógicas óptimas en tutores inteligentes

Hermilo Victorio Meza¹, Manuel Mejía-Lavalle¹,
Alicia Martínez Rebollar¹, Obdulia Pichardo², Grigori Sidorov²

¹ Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET),
Tecnológico Nacional de México, Cuernavaca, Morelos,
México

² Instituto Politécnico Nacional (IPN), Ciudad de México,
México

{hvictoriom, mlavalle, amartinez}@cenidet.edu.mx
ayilina@hotmail.com, sidorov@cic.ipn.mx

Resumen. Se fundamenta y describe un proyecto de investigación en desarrollo en el área de Sistemas Tutores Inteligentes que busca automatizar la recomendación de acciones pedagógicas óptimas para que un estudiante lleve a cabo de la mejor manera su aprendizaje. Para lograr lo anterior se están empleando procesos markovianos, específicamente los llamados Procesos de Decisión de Markov Parcialmente Observables, debido a que ellos han permitido generar automáticamente políticas óptimas exitosas en gran variedad de dominios del mundo real. Sin embargo, este tipo de procesos markovianos tienen la problemática de ser de alta complejidad computacional, por lo que, antes de poder ser empleados en problemas reales del área de Tutores Inteligentes, se deberá trabajar en reducir tal complejidad. En el presente artículo se presentan una serie de experimentos que buscan evaluar la complejidad de cuatro métodos o algoritmos exactos y tres aproximados disponibles en la Web para la solución de modelos markovianos. De la evaluación así realizada proponemos varias alternativas para reducir la complejidad de estos algoritmos, pero buscando no afectar las ventajas relevantes que ofrecen los Procesos de Decisión de Markov Parcialmente Observables.

Palabras clave: Sistemas tutores inteligentes, modelo del estudiante, modelo del tutor, procesos de decisión de Markov parcialmente observables.

Application of Markovian Processes for Recommendation of Optimal Educational Actions for Intelligent Tutoring Systems

Abstract. We describe a project still in development about Intelligent Tutoring Systems and optimal educational actions. Good pedagogical actions are key components in all learning-teaching schemes. Automate that is an important

Intelligent Tutoring Systems objective. We propose apply Partially Observable Markov Decision Process (POMDP) in order to obtain automatic and optimal pedagogical recommended actions in benefit of human students, in the context of Intelligent Tutoring System. To achieve that goal, we need previously create an efficient POMDP solver framework with the ability to work with real world tutoring cases. At present time, there are several Web available POMDP open tool solvers, but their capacity is limited, as experiments showed in this paper exhibit. In this work, we describe and discuss several design ideas toward obtain an efficient POMDP solver, useful in our problem domain

Keywords: Intelligent tutoring systems, student model, tutor model, partially observable Markov decision processes.

1. Introducción

Los Sistemas Tutores Inteligentes (STI) son el equivalente digital de los maestros humanos, dado que ofrecen servicios para apoyar al aprendizaje; y, de manera similar a como lo haría un tutor humano experimentado, los STI tienen la capacidad de adaptar las actividades de aprendizaje de acuerdo con la situación de cada estudiante en particular, de tal manera que el proceso se desarrolle en las mejores condiciones que propicien un aprendizaje eficiente. Estos tutores monitorean y entrenan a los estudiantes basados en representaciones tanto de contenido como de aspectos sociales y razonan acerca de la probabilidad de las acciones del estudiante. El campo de la Inteligencia Artificial en la educación investiga sobre la naturaleza del conocimiento y su representación, la forma en que se puede ayudar a aprender a un estudiante y sobre cuáles estilos de interacción de enseñanza son efectivos y cuándo deberían usarse [1].

En el presente artículo se fundamenta y describe un proyecto de investigación, actualmente en desarrollo, en el área de STI que busca automatizar la recomendación de acciones pedagógicas óptimas para que un estudiante lleve a cabo de la mejor manera posible su aprendizaje en línea. Para lograr lo anterior se están empleando los Procesos de Decisión de Markov Parcialmente Observables (POMDP por sus siglas en inglés), debido a que existen múltiples trabajos publicados que reportan que usando estos algoritmos markovianos se han podido generar automáticamente políticas óptimas exitosas en gran variedad de dominios del mundo real como la robótica. Sin embargo, este tipo de procesos markovianos tienen como problema el ser de alta complejidad computacional, es decir, para casos relativamente pequeños el tiempo de procesamiento es alto y conforme el caso crece, el tiempo de respuesta se incrementa mucho más, haciendo a estos procesos prohibitivos para casos del mundo real, donde el número de variables asciende a algunas centenas de variables. Es por eso que, antes de poder ser empleados los POMDP en problemas reales del área de STI, se deberá trabajar en reducir tal complejidad. En el presente artículo se presentan una serie de experimentos que buscan evaluar la complejidad de cuatro métodos o algoritmos (en inglés se les nombra como *solvers*) exactos y tres aproximados que se obtuvieron de la Web y que sirven para la solución de estos modelos markovianos. Con estos experimentos realizados, y que mostramos en el presente artículo, comprobamos las limitaciones que actualmente tienen estas herramientas computacionales. Observando estas limitaciones

nosotros proponemos varias alternativas para reducir la complejidad de estos algoritmos, pero buscando no afectar las ventajas relevantes que ofrecen los POMDP.

El resto de este trabajo está organizado de la siguiente manera: en la Sección 2 fundamentamos esta investigación, describiendo los conceptos relevantes; en la Sección 3 revisamos el trabajo previo que se ha realizado a nivel internacional para el tema poco explorado del uso de POMDP en el área de los STI; en la Sección 4 describimos la forma en que nosotros estamos tratando el problema de la recomendación automatizada de acciones pedagógicas usando POMDP; la Sección 5 detalla los experimentos realizados y los resultados obtenidos; finalmente en la sección 6 se ofrecen las conclusiones y el trabajo a realizar en el futuro inmediato.

2. Contexto teórico

En esta Sección presentamos los fundamentos teóricos en torno a nuestro trabajo de investigación, como lo es una descripción del Modelo del Estudiante sin y con POMDP, la definición formal detrás de los POMDP y la manera en que se obtiene una solución, así como el problema de la complejidad en los POMDP.

2.1. Modelo del estudiante

Las características individuales pueden incluir desde los conceptos y/o habilidades dominadas, estilo de aprendizaje y estado emocional, entre otras. Estas características se concentran en el llamado Modelo del Estudiante, componente esencial de los STI, dado que la adaptabilidad del sistema depende de la información contenida en este modelo.

Dado que la característica distintiva de los STI es su adaptabilidad a la situación personal de cada estudiante, la representación del conocimiento se considera un elemento de especial importancia. En particular, el Modelo del Estudiante es el componente del STI que contiene la información necesaria para una adecuada adaptabilidad reflejada en actividades de aprendizaje aplicables a la situación actual. El Modelo del Tutor recibe información de los Modelos del Estudiante y de Dominio y toma decisiones acerca de las estrategias tutoriales y acciones pedagógicas. Basado en el conocimiento indicado debe tomar decisiones tales como si intervenir o no, y de ser así, cuándo y cómo. La planificación y distribución de contenidos son también parte de las funciones del Modelo del Tutor [2].

El Modelo del Estudiante es el componente clave de un STI. Idealmente, debe contener tanto conocimiento como sea posible acerca de los estados cognitivo y afectivo del estudiante y su evaluación conforme avanza el proceso de aprendizaje. El modelo del estudiante incorpora varias funciones: a) debe reunir información sobre el estudiante; b) debe crear una representación del conocimiento del estudiante y de su proceso de aprendizaje a partir de esa información; y c) debe tomar en cuenta esos datos para realizar algún tipo de diagnóstico, tanto del estado de conocimiento del estudiante y en términos de seleccionar estrategias pedagógicas óptimas para presentar subsecuente información de dominio para el estudiante [2].

La información contenida en el Modelo del Estudiante determina el comportamiento del sistema, generando una estrategia de adaptación basada en las características personales del estudiante que atiende, en un momento determinado.

De los diversos métodos que permiten modelar las características del estudiante se seleccionó a los Procesos de Decisión de Markov Parcialmente Observables (Partially Observable Markov Decision Processes o POMDP). Se considera que no se tiene acceso directo al conocimiento del estudiante (procesos y planes mentales) [3], sino que se deduce a partir de ciertas evidencias como lo son las evaluaciones que se le apliquen, por lo que se tiene cierto grado de incertidumbre. Entre los métodos de modelado que consideran el manejo de la incertidumbre se cuentan a la Lógica Difusa, Redes Bayesianas y similares como las Redes Bayesianas Dinámicas, Redes de Decisión, etc. y los modelos de Markov, como las Cadenas de Markov, los Modelos Ocultos de Markov, los Procesos de Decisión de Markov (completamente observables) y los Procesos de Decisión de Markov Parcialmente Observables ó POMDP.

2.2. Modelado del estudiante con POMDP

Los métodos de modelado citados, con excepción de los dos tipos de procesos de decisión de Markov, requieren de componentes adicionales que determinen las acciones que deba realizar el sistema una vez determinado el estado de conocimiento del estudiante. Los Procesos de Decisión de Markov (completa y parcialmente observables) incluyen en su estructura las acciones que puede ejecutar el sistema, siendo guiadas por las recompensas indicadas en el modelo. Al no ser completamente observable el estado de conocimiento del estudiante se descartan los Procesos de Decisión de Markov Completamente Observables (MDPs o COMDPs).

El proceso de enseñanza puede modelarse como un problema de toma de decisiones secuenciales bajo incertidumbre. Un STI debe seleccionar las acciones pedagógicas, tales como acciones remediales y recomendaciones, sin tener un conocimiento absoluto de los estados mentales del estudiante y sin garantía de que las acciones tendrán el efecto deseado [3].

Al enfrentarse con un ambiente parcialmente observable, se justifica el uso de un modelo general para problemas de decisión secuencial como los POMDP. Un POMDP es un modelo para planeación bajo incertidumbre, lo que da al agente la capacidad de estimar efectivamente el resultado de sus acciones aún cuando no puede observar el ambiente con exactitud.

2.3. Definición formal de POMDP

Un POMDP es un formalismo para representar problemas que involucran la planeación bajo incertidumbre. Se puede usar para computar iterativamente una política para optimizar una función de recompensas sobre horizontes de tiempo progresivamente largos. Han probado ser útiles en muchas tareas de planeación diferentes a los STI tales como el control de robots, planeación de tratamientos médicos e interpretación de voz (*spoken dialogue*) o video. El propósito del POMDP es describir cómo cambian los estados mediante la ejecución de acciones, de tal manera que en el proceso de su solución se pueda estimar el estado y cuál acción se aplica en el momento y se propongan planes para mejorar el estado a lo largo del tiempo [4].

La composición de un POMDP se define como una tupla $(S, A, P(s'|s,a), R(a,s), O, P(o/s,a))$ que comprende: un conjunto de estados ocultos S ; un conjunto de acciones A ; un conjunto de probabilidades de transición $P(s'|s,a)$ que definen cómo las acciones cambian los estados del sistema; un conjunto de recompensas $R(a,s)$ que asignan valores para los estados alcanzados o acciones completadas; un conjunto de observaciones O y un conjunto de probabilidades de emisión $P(o/s,a)$ que definen cuáles observaciones aparecen cuando el sistema se encuentra en un estado dado. La Figura 1 ilustra estas ideas.

2.4. Solución de un POMDP

La solución de un POMDP consiste en determinar la política o conjunto de acciones a ejecutar que producirá la mayor recompensa, aplicables a lo que se perciben como los diferentes estados del sistema. Como consecuencia de una acción aplicada, el sistema hará probablemente una transición a otro estado.

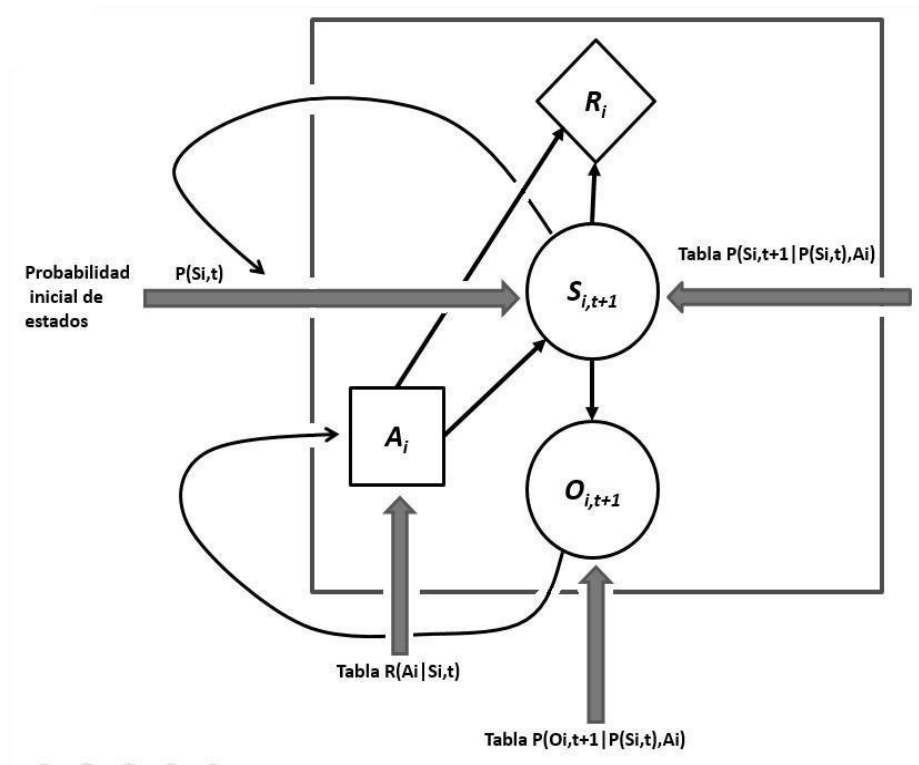


Fig. 1. Representación gráfica de un POMDP.

Para resolver los POMDP hay una amplia diversidad de algoritmos que, al mismo tiempo que pretenden obtener una solución de calidad aceptable (que no siempre es la óptima) buscan aliviar el problema de complejidad en espacio de almacenamiento y tiempo de procesamiento que tiene este método de representación del conocimiento [5, 6, 7].

2.5. Problemática de los POMDP

Muchos problemas pueden modelarse con POMDP, pero muy pocos pueden resolverse hasta alcanzar el óptimo debido a su complejidad computacional. La Teoría de la Computación clasifica los problemas de acuerdo con su dificultad computacional y denomina de clase P a los problemas fáciles o tratables que pueden resolverse en una cantidad de tiempo polinomial; en la clase NP , más amplia, y que incluye a P , se encuentran los problemas que se consideran difíciles o intratables; los problemas NP -**completos** son los problemas más difíciles de la clase NP y se asume que tardan una cantidad exponencial de tiempo. La cualidad de completitud se refiere al conjunto de problemas más difícil de resolver en la clase. $PSPACE$ es otra clase aún más amplia que se refiere a los problemas que pueden resolverse en una cantidad de espacio polinomial. La clase P se incluye en $PSPACE$ dado que cualquier algoritmo que corre en un tiempo polinomial sólo puede consumir una cantidad polinomial de recursos [5]. Aún cuando los problemas modelados como Procesos de Decisión de Markov Completamente Observables (COMDPs) se clasifican como problemas P -**completos**[19], no es el caso de los POMDP, que se clasifican como $PSPACE$ -**completos** [8] aún para horizontes finitos muy reducidos [19] por lo que sólo queda el recurso de buscar soluciones aproximadas [5]. Algunos problemas con solución exacta difícil tienen un correspondiente algoritmo de aproximación fácil y hay otros cuya aproximación es tan difícil como la solución exacta, siendo desafortunadamente éste el caso para los POMDP, aún para horizontes finitos [5].

El principal problema con los POMDP es que su complejidad los hace aplicables, aparentemente, sólo en ambientes pequeños [7, 9]. Para los POMDP sólo pueden encontrarse soluciones óptimas (estrategias o políticas que indican acciones a ejecutar según el estado actual) para problemas con unos pocos estados (menos de 20, contra 100 o más necesarios para un STI, según [3]) por sus requerimientos de almacenamiento y tiempo de procesamiento. Por lo tanto, la complejidad de los procedimientos de solución del problema se convierte en el aspecto clave de la aplicación exitosa del modelo en problemas del mundo real, aún a expensas de la optimalidad. Sin embargo, la mayoría de los problemas de interés tienen un espacio de estados enorme, lo que motiva a la búsqueda de métodos aproximados [8, 9].

Debido a que los POMDP consideran el manejo de la incertidumbre en la representación del conocimiento y deciden, a largo plazo, las acciones a ejecutar en base a las recompensas, parece el método de modelado natural para los STI y sin embargo tienen una muy escasa aplicación en éstos, la cual se ha dado sólo en tiempos recientes [3, 10, 12, 13, 14].

3. Trabajo relacionado con la aplicación de los POMDP en educación

Aunque los POMDP ofrecen ventajas frente a otros métodos de modelado del estudiante, en los STI han sido poco investigados en esta área y eso sólo recientemente [3, 12, 10, 14]. Las investigaciones relacionadas con POMDP generalmente experimentan con solución de problemas relacionados con la navegación robótica [8, 9, 7, 5], aunque se menciona que también pueden aplicarse en problemas relacionados

con la operación, mantenimiento y reparación de maquinaria industrial y en medicina en el tratamiento de las enfermedades [15]. La causa principal de esta escasa aplicación se deriva del gran problema de los POMDP: la complejidad de éstos al tener grandes requerimientos de almacenamiento y tiempo de procesamiento. La solución óptima para un POMDP sólo puede determinarse para problemas pequeños de 10 a 20 estados como máximo, lo cual dejaría fuera a los problemas del mundo real con una cantidad mucho mayor de estados, como sería el caso de los STI. Existen métodos que encuentran soluciones aproximadas, que se espera sean aceptables y cercanas a la óptima, pero aún así siguen siendo complejas y, posiblemente, no aplicables a todo tipo de problemas [9, 7, 8, 10].

Las investigaciones relacionadas con POMDP en STI han considerado la complejidad del modelo y han aplicado métodos que hagan tratable la solución. Aunque se llega a mencionar que los métodos utilizados pueden tratar con POMDP de tamaño significativo, en su investigación han abordado sólo modelos de tamaño relativamente pequeño.

Por ejemplo, Folsom-Kovarik [3, 4] combina un simulador de adiestramiento militar con un STI basado en POMDP. El STI tiene el objetivo de resolver problemas de aprendizaje (*knowledge gaps*) derivados de la operación del simulador. Para reducir la complejidad del POMDP se plantean dos estrategias: colas de prioridad y cadenas de observación.

Theocharous [12] desarrolló un STI para enseñar a ordenar un conjunto de monedas virtuales. Considera que la enseñanza es un dominio interesante para la Inteligencia Artificial porque deben resolverse tres problemas al mismo tiempo: el maestro debe inferir variables sociales (atención, aburrimiento, confusión, conocimientos, aptitudes) y variables físicas (progreso de la tarea, objetos usados, actividad actual) y debe combinar este conocimiento para planear estrategias de interacción efectivas momento-a-momento. El algoritmo cambia entre estados prototípicos de estudiantes basándose en la probabilidad de las variables sociales. Es decir, se crean varios POMDP pequeños que pueden ser seleccionados por un segundo POMDP, un POMDP jerárquico.

Rafferty [10, 11] considera tres modelos de aprendizaje del estudiante (modelo sin memoria, modelo discreto con memoria, modelo continuo) y se presentan métodos aproximados para encontrar las acciones de enseñanza óptimas dados los grandes espacios de estados y acciones que se presentan en la enseñanza. Para determinar la política (que seleccione la mejor acción dada una distribución sobre el estado de conocimiento actual del estudiante, el *beliefstate*) se seleccionó una técnica *online* de búsqueda hacia adelante (*forward*), dado que este tipo de técnicas ha demostrado tener un buen desempeño en otros grandes dominios.

Zhang [13] y Wang [14] presentan un marco de POMDP para construir un STI en el que se puede hacer una reducción del espacio de estados posiblemente exponencial a un espacio manejable; la reducción presentada se basa en que en un sistema de aprendizaje de conceptos hay un orden de prerrequisitos, de tal manera que el dominio de un concepto cualquiera generalmente depende de conceptos previos; para llegar al estado de dominio del concepto considerado, se debió pasar, previamente, por los estados anteriores a él (prerrequisitos) con lo que se evitarían estados (combinaciones) imposibles.

4. Dominio de la investigación: Automatizar la recomendación de acciones pedagógicas óptimas

Para realizar la experimentación se diseñaron modelos aleatorios de POMDP de tamaño $2 \times 2 \times 2$ (es decir: 2 estados, 2 acciones y 2 observaciones), $3 \times 3 \times 3$, $4 \times 4 \times 4$, $5 \times 5 \times 5$, $6 \times 6 \times 6$, $10 \times 10 \times 10$ y $20 \times 20 \times 20$. Estos modelos los llamamos aleatorios porque generamos previamente y con un programa de computadora, valores aleatorios que usamos como las probabilidades de transición de un estado al siguiente.

También se diseñaron modelos de POMDP que consideran 50% de eficiencia en el aprendizaje, de tamaño $2 \times 2 \times 2$, $3 \times 3 \times 3$, $4 \times 4 \times 4$, $5 \times 5 \times 5$ y $6 \times 6 \times 6$.

Además se diseñó un modelo de POMDP de $10 \times 10 \times 10$ que considera la eliminación de probabilidades de transición y observaciones para probabilidades menores a 0.05 (umbral de 5%), es decir, se traducen a 0.0. Después de la modificación de estas probabilidades se normalizan el resto de los valores de la fila, para que sumen 1.0.

Los modelos propuestos en las pruebas actuales sólo consideran una acción por estado. En una situación ideal se supone que la acción $a1$ aplicada en el estado $s1$ genera la transición al estado $s2$.

Se considera una distribución uniforme de recompensas (todos los estados tienen el mismo valor o proporción). De lo anterior, la recompensa (calculada como porcentaje) de aplicar la acción i en el estado i de un modelo de n estados es $r = 100 / (n - 1)$. Se divide entre $(n - 1)$ porque se asume que el último estado es el objetivo y no hay recompensas por aplicar acciones adicionales en este estado. Cualquier acción x diferente de i aplicada en el estado i no generará recompensa (0). Aunque los modelos podrían considerar castigos o costos (recompensas negativas) por aplicar acciones que generen transiciones a estados no deseables.

Con la presente investigación se espera encontrar un procedimiento de solución de POMDP que supere a los hasta ahora usados para STI, sin afectar de manera importante la representatividad del modelo.

5. Resultados de la experimentación

Al momento de redactar el presente artículo se han hecho pruebas iniciales que consideran algunos de los algoritmos de solución aplicados a modelos de POMDP de tamaño relativamente reducido. Se utilizó el software *Pomdp-solve v5.4* [16] para calcular las soluciones exactas (*Enumeration*, *Twopass*, *Witness*, *Incremental Pruning*) de los modelos generados. También se utilizó el paquete de software *ZMDP v1.1.7* [17] que ejecuta los algoritmos aproximados *HSVI* y *FRTDP*. Además, se utilizó el paquete *APPL v0.96* [18] que aplica el algoritmo aproximado *SARSOP*. Todas las sesiones de experimentación se dejaron en ejecución por un máximo aproximado de una hora (3,600 segundos). El software se ejecutó bajo el sistema *Linux Ubuntu 15*. Se utilizó un equipo de cómputo portátil con procesador *Intel CORE i5* a 2.40GHz y 4GB en RAM. En la Tabla 1 se muestran los resultados obtenidos aplicando métodos de solución exacta.

Tabla 1. Solución de modelos de POMDP mediante métodos exactos.

MODELO	MÉTODOS EXACTOS											
	ENUMERATION			TWO PASS			WITNESS			INCREMENTAL PRUNING		
	t(seg)	obs.**	Lim. Inf.	t(seg)	obs.**	Lim. Inf.	t(seg)	obs.**	Lim. Inf.	t(seg)	obs.**	Lim. Inf.
2aleat	0.098		3481.403000	<u>0.092</u>		3481.403000	0.100		3481.403000	0.113		3481.403000
2ef	0.305		1753.302000	0.090		1753.302000	<u>0.072</u>		1753.302000	0.105		1753.302000
3aleat	0.319		1108.095000	0.138		1108.095000	0.130		1108.095000	<u>0.108</u>		1108.095000
3ef	4.430		1282.275000	21.731		1282.275000	3.918		1282.275000	<u>2.903</u>		1282.275000
4aleat	2.634		712.418000	7.787		712.418000	2.832		712.418000	<u>2.478</u>		712.418000
4ef	404.950		168.117000	428.390		168.117000	53.040		168.117000	<u>23.960</u>		168.117000
5aleat	*	3 epocas	42.257000	*	4 epocas	49.636000	*	4 epocas	49.636000	*	<u>15 epocas</u>	122.051000
5ef	*	3 epocas	50.659000	*	3 epocas	50.659000	*	4 epocas	60.291000	*	<u>5 epocas</u>	69.499000
6aleat	*	2 epocas	27.691942	*	2 epocas	27.691942	*	<u>3 epocas</u>	33.449471	*	<u>3 epocas</u>	33.449471
6ef	*	2 epocas	31.700000	*	2 epocas	31.700000	*	<u>3 epocas</u>	39.205633	*	<u>3 epocas</u>	39.205633
10aleat	*	<u>2 epocas</u>	2.650088	*	<u>2 epocas</u>	2.650088	*	<u>2 epocas</u>	2.650088	*	<u>2 epocas</u>	2.650088
20aleat	*	<u>1 epoca</u>	0.525263	*	<u>1 epoca</u>	0.525263	*	<u>1 epoca</u>	0.525263	*	<u>1 epoca</u>	0.525263

* No se obtuvo solución completa en 3600 segundos (1 hora)

** Iteraciones resueltas en tiempo límite de 3600 segundos (1 hora)

Nótese que sólo se obtuvieron soluciones exactas completas para los modelos de tamaño 2 a 4. Para los restantes modelos se obtuvieron soluciones parciales. Se observan los mejores tiempos de solución (subrayados en la Tabla 1) para el método *Incremental Pruning*. Los peores tiempos se obtuvieron con el método *Two Pass*. La Figura 2 ilustra los resultados obtenidos para los modelos de tamaño 2 a 4.

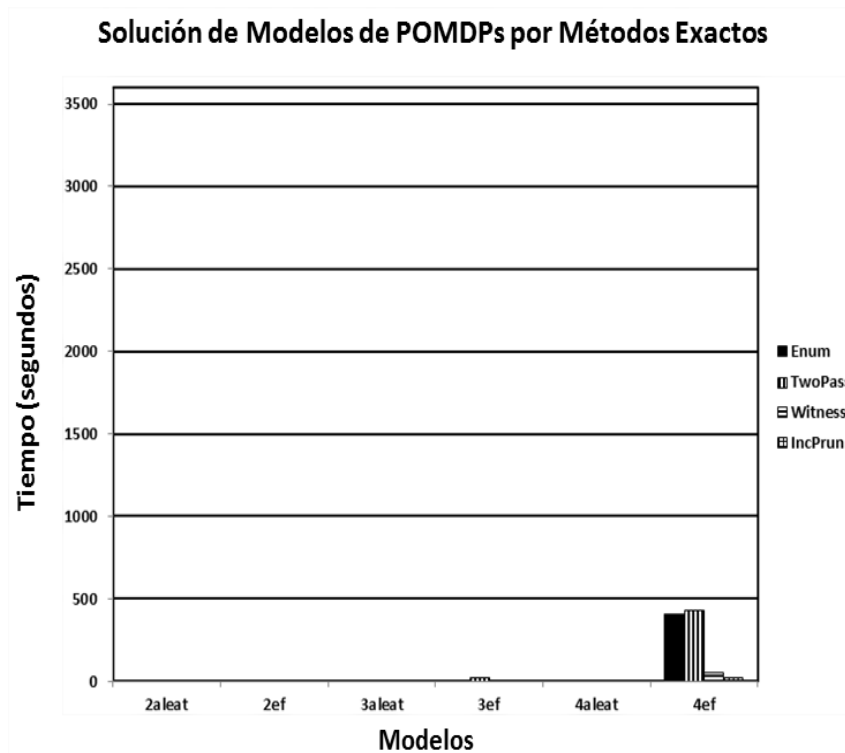


Fig. 2. Resultados con métodos exactos (escala lineal).

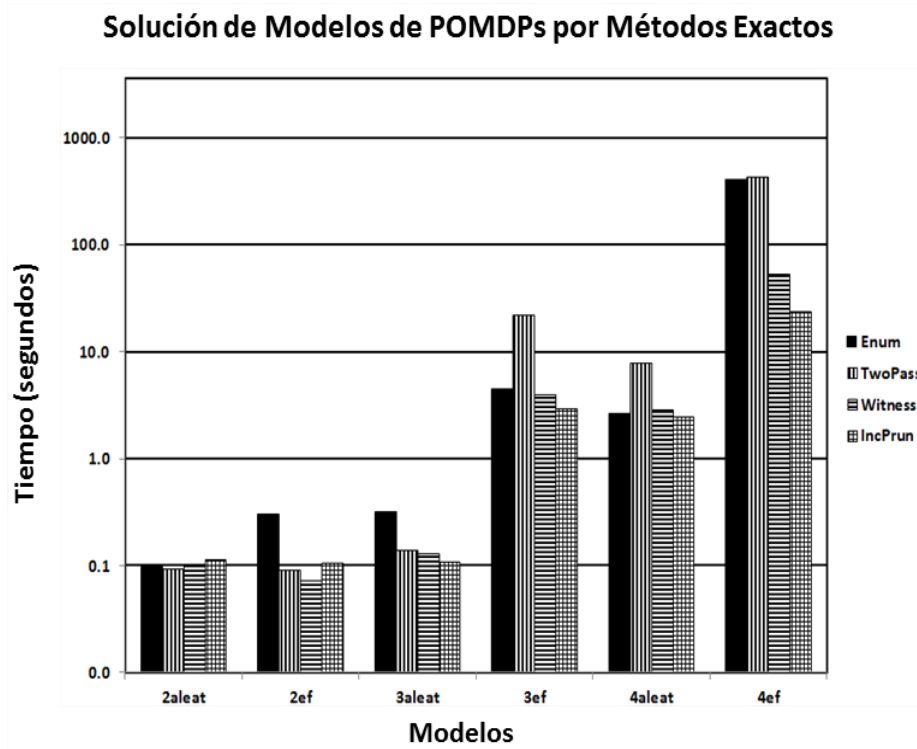


Fig. 3. Resultados con métodos exactos (escala logarítmica)

La Figura 3 muestra los mismos resultados pero con una escala logarítmica en el eje y, para observar mejor los resultados.

La Tabla 2 muestra los resultados obtenidos aplicando métodos de solución aproximados.

Tabla2. Solución de modelos de POMDP mediante métodos aproximados.

MODELO	MÉTODOS APROXIMADOS											
	SARSOP				FRTDP				HSVI			
	t(seg)	Lim. Inf.	Lim. Sup.	Precision	t(seg)	Lim. Inf.	Lim. Sup.	Precision	t(seg)	Lim. Inf.	Lim. Sup.	Precision
2aleat	0.009	3484.860000	3484.860000	0.000476	0.000	3484.857700	3484.858100	0.000476	0.000	3484.857700	3484.858100	0.000439
2ef	0.010	1754.970000	1754.970000	0.000475	0.000	1754.966400	1754.966900	0.000475	0.000	1754.966400	1754.966900	0.000438
3aleat	212.190	1109.170000	1109.170000	0.000990	2196.0000	1109.170700	1109.171600	0.000865	1221.0000	1109.170700	1109.171700	0.001000
3ef	3600.0000	1283.530000	1283.610000	0.083941	*** error desconocido ***				3600.0000	1283.528700	1283.612900	0.084127
4aleat	3600.0000	713.105000	714.704000	1.599830	3600.0000	713.104500	715.840600	2.736080	3600.0000	713.104500	714.297700	1.193130
4ef	3600.0000	168.118000	168.147000	0.029339	3600.0000	168.117700	168.301800	0.184090	3600.0000	168.117700	168.147100	0.029417
5aleat	3600.0000	412.872000	420.809000	7.936890	3600.0000	412.871800	421.218700	8.346880	3600.0000	412.871800	416.356700	3.484870
5ef	3600.0000	507.242000	534.347000	27.105000	3600.0000	507.242100	535.308100	28.066000	3600.0000	507.242200	526.118500	18.876400
6aleat	3600.0000	313.586000	346.178000	32.591900	3600.0000	313.585700	349.568400	35.982700	3600.0000	313.586000	337.602700	24.016700
6ef	3600.0000	95.507200	96.874200	1.366960	3600.0000	95.507200	98.076600	2.569340	3600.0000	95.507200	97.257400	1.750190
10aleat	3600.0000	3.113280	3.264020	0.150742	3600.0000	3.112900	3.486400	0.373573	3600.0000	3.113300	3.294400	0.181095
20aleat	3600.0000	2.153080	2.800530	0.647453	*** error desconocido ***				3600.0000	2.163100	2.703000	0.539856

Se observa que el método *SARSOP* obtiene los mejores tiempos de solución para el modelo *3aleat*. Los mejores valores (menores) de precisión corresponden al método *HSVI*. En el método *FRTDP*, además de presentar menos elementos favorables

(subrayados en la Tabla 2) tiene problemas con algunos modelos (*3ef* y *20 aleat*). La Figura 4 muestra los resultados obtenidos para los modelos de tamaño 2 a 4, comparables con la Figura 2.

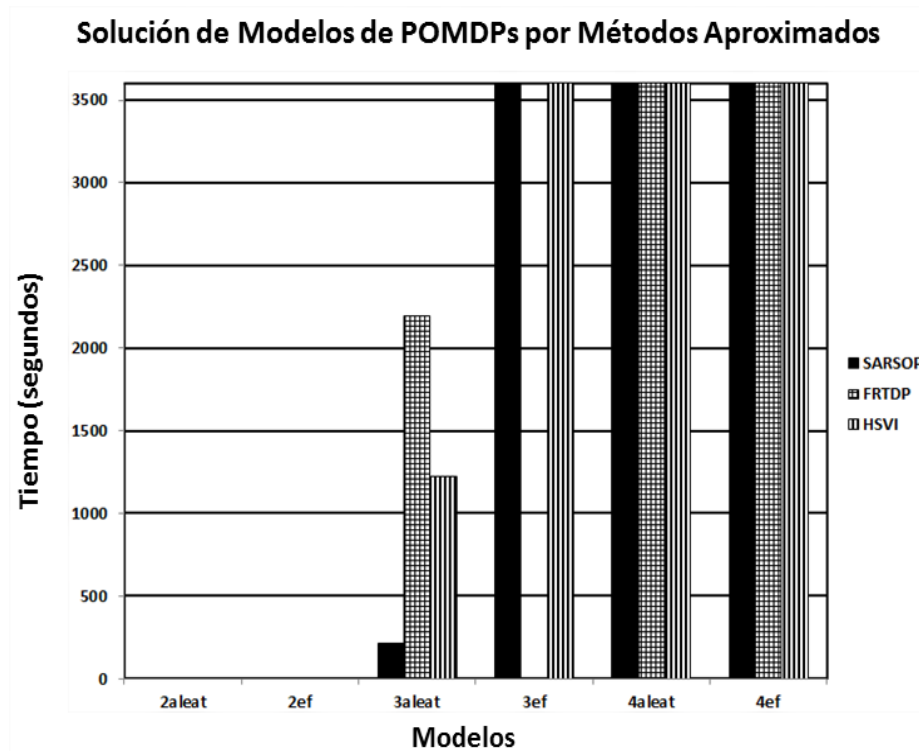


Fig. 4. Resultados con métodos aproximados.

Hacemos notar que, hasta donde podemos analizar, los métodos exactos comparan las diferencias en recompensa entre la última época (iteración) calculada y la penúltima. Si esta diferencia es menor a la precisión buscada se detiene el proceso y se genera la política resultante. Por otro lado, en los métodos aproximados se comparan las diferencias en recompensa entre la última época (iteración) calculada (a la que se llama *lower bound* o límite inferior) y un límite superior (*upper bound*) que es un máximo teórico calculado, que siempre es superior a la última época calculada. Si esta diferencia es menor a la precisión buscada se detiene el proceso y se genera la política resultante. El tiempo extra que utilizan los métodos aproximados es para la búsqueda de una mayor recompensa tendiendo hacia el límite superior cuando no se ha alcanzado la precisión establecida.

Además, la comparativa sólo se aplica a los modelos de tamaño 2 a 4, pues para modelos de tamaño 5 o mayor los métodos exactos no completan la solución y sólo resuelven unas pocas épocas. En cambio, los métodos aproximados sí generan una solución, aunque la precisión lograda varía dependiendo del tamaño del modelo y del tiempo de procesamiento.

Algo similar podría observarse con el método *Incremental Pruning* al compararlo con los demás métodos exactos: en general es el más rápido, pero lo superan todos los demás en los modelos de tamaño 2 (*Zaleat*, 2ef).

Como consecuencia de observar los resultados de la experimentación, proponemos algunas ideas de diseño tendientes a reducir la complejidad en el proceso de solución del modelo de POMDP en tres aspectos:

- Descartar transiciones de estados (o combinaciones de transición*recompensa, transición*recompensa*observación) menores a umbrales mínimos establecidos.
- Clasificar estados en un máximo de tres grandes grupos: retroceso (recompensas menores a las obtenidas en el estado actual), detenido (recompensas iguales a las obtenidas en el estado actual) y avance (recompensas mayores a las obtenidas en el estado actual) y variaciones en niveles de granularidad (subniveles de los tres grandes grupos, para ampliar abanico de búsqueda).
- Soluciones como POMDP jerárquicos de diferentes niveles comparados contra modelos monolíticos.

6. Conclusiones y trabajo futuro

Se ha presentado y fundamentado una investigación que se encuentra actualmente en desarrollo y que busca automatizar la recomendación de acciones pedagógicas óptimas en el contexto de los STI y por medio del uso de POMDP.

Dada la complejidad computacional de los POMDP hemos realizado una serie de experimentos con algoritmos (*solvers*) que proporcionan soluciones exactas y aproximadas y que fueron obtenidos libremente de la Web. En la experimentación realizada estos *solvers* mostraron sus limitaciones, según reportamos en la Sección 5.

Por ejemplo, al emplear métodos exactos se obtuvieron soluciones para los modelos de tamaño de 4 estados e inferiores. Para los modelos de tamaño mayor se obtuvieron soluciones parciales (épocas resueltas) y en algunos casos se agotó la memoria disponible del equipo de cómputo.

Al emplear métodos aproximados se obtuvieron soluciones con precisión aceptable para los algoritmos *HSVI*, *FRTDP* y *SARSOP*, obteniéndose las mejores soluciones para este último. El algoritmo *FiniteGrid* sólo obtuvo soluciones para modelos de tamaño 6 e inferiores. Al observar estas limitaciones, varias ideas útiles de diseño surgen para reducir la complejidad, sin afectar la calidad de la solución.

En el futuro se deberán probar estas ideas de diseño incorporándolas en los *solvers* libremente disponibles en la Web para validarlos, a saber: a) Descartar transiciones de estados, b) Clasificar estados en un máximo de tres grandes grupos: retroceso, detenido y avance, más variaciones en subniveles de granularidad, y c) Soluciones como POMDP jerárquicos de diferentes niveles.

Referencias

1. Woolf, B.: Building Intelligent Interactive Tutors. Student-centered strategies for revolutionizing e-learning, Morgan Kaufmann, Burlington (2009)

2. Nkambou, R.: Introduction: What Are Intelligent Tutoring Systems, and Why This Book? In: R. Nkambou et al. (eds.) *Advances in Intelligent Tutoring Systems*, SCI 308, pp. 1–12 (2010)
3. Folsom-Kovarik, J.T., Sukthankar, G., Schatz, S.: Tractable POMDP representations for intelligent tutoring systems. *ACM Trans. Intell. Syst. Technol.* Vol. 4, No. 2, Article 29 (2013)
4. Folsom-Kovarik, J., Sukthankar, G., Schatz, S.: Integrating Learner Help Requests Using a POMDP in an Adaptive Training System. In: *Proceedings of the Twenty-Fourth Innovative Applications of Artificial Intelligence Conference* (2012)
5. Cassandra, A.: *Exact and Approximate Algorithms for Partially Observable Markov Decision Processes*. Ph. D. thesis. Brown University. Providence, Rhode Island (1998)
6. The POMDP Page, <http://www.pomdp.org/> (consultado en abril 2016)
7. Hauskrecht, M.: Value-Function Approximations for Partially Observable Markov Decision Processes. *Journal of Artificial Intelligence Research*, Vol. 13, pp. 33–94 (2000)
8. Paquet, S., Chaib-draa, B., Ross, S.: Hybrid POMDP Algorithms. In *Proceedings of The Workshop on Multi-Agent Sequential Decision Making in Uncertain Domains (MSDM)* (2006)
9. Ross, S., Pineau, J., Paquet, S.: Online Planning Algorithms for POMDPs. *Journal of Artificial Intelligence Research*, Vol. 32, pp. 663–704 (2008)
10. Rafferty, A., Brunskill, E., Griffiths, T., Shafto, P.: Faster Teaching via POMDP Planning. *Cognitive Science*, pp. 1–43 (2015)
11. Rafferty, A.: *Applying Probabilistic Models for Knowledge Diagnosis and Educational Game Design*. Technical Report No. UCB/EECS-2014-61, Electrical Engineering and Computer Sciences, University of California at Berkeley (2014)
12. Theocharous, G., Beckwith, R., Butko, N., Philipose, M.: Tractable POMDP Planning Algorithms for Optimal Teaching in SPAIS. In: *Workshop on Plan Activity, and Intent Recognition (PAIR), IJCAI* (2009)
13. Zhang, P.: *Using POMDP-based Reinforcement Learning for Online Optimization of Teaching Strategies in an Intelligent Tutoring System*. Master of Science thesis. University of Guelph, Guelph, Ontario, Canada (2013)
14. Wang, F.: *POMDP Framework for Building an Intelligent Tutoring System*. School of Computer Science, University of Guelph, Guelph, Canada (2014)
15. Sondik, E., Smallwood, R.: The Optimal Control of Partially Observable Markov Processes Over a Finite Horizon. *Operations Research*, Vol. 21, No. 5, pp. 1071–1088 (1973)
16. pomdp-solve, <http://www.pomdp.org/code/> (consultado en abril 2016)
17. ZMDP software for POMDP and MDP planning, <https://github.com/trey0/zmdp> (consultado en abril 2016)
18. Approximate POMDP Planning Software, <http://bigbird.comp.nus.edu.sg/pmwiki/farm/appl/> (consultado en abril 2016)
19. Papadimitriou, C., Tsitsiklis, J.: The complexity of Markov Decision Processes. *Mathematics of Operations Research*, Vol 12, No. 3, pp. 441–450 (1987)