

ASISTENCIA INTELIGENTE EN AMBIENTES DE APRENDIZAJE COLABORATIVOS

Heinz Ulrich HOPPE

RESUMEN

Algunas críticas al concepto de sistemas de tutoría inteligentes (STI) indican que se debería definir un marco social y organizacional para el uso de mecanismos inteligentes para apoyar el aprendizaje. Hoy poseemos nuevas tecnologías que permiten la introducción de asistencia automática junto con comunicación electrónica en ambientes existentes como el aula. En este documento se describen herramientas de construcción de materiales interactivos, mecanismos de comunicación y un ejemplo de asistencia inteligente. Además, se discuten varias formas de brindar ayuda inteligente en ambientes de aprendizaje colaborativos.

REVISIÓN CRÍTICA DE SISTEMAS DE TUTORÍA INTELIGENTES

Recientemente se han formulado críticas fundamentales con respecto al desarrollo y a la investigación de sistemas de tutoría inteligentes (STI). Se ha argumentado que esta línea de trabajos científicos podría estar mal dirigida porque se basa en un concepto de aprendizaje individualizado y además reduce el contenido de lo enseñado a lo que se puede modelar con técnicas existentes de Inteligencia Artificial (AI). Esto implicaría, en particular, dificultades para modelar el razonamiento de "sentido común", lo que podría dificultar el entendimiento mutuo entre el tutor inteligente y el estudiante en un sentido integral. Junto con la falta de interacción social, el concepto de tutoría inteligente reduciría la riqueza y variedad indispensables para un aprendizaje verdaderamente profundo y significativo. Este punto de vista coincide por ejemplo con la posición adoptada por Clancey [1].

Aceptar esta posición en su forma apodíctica nos llevaría a rechazar y descartar completamente lo que se ha logrado en el campo de STI. Por varias razones esto no parece justo: Primero, el enfoque STI no pretende reemplazar cada tipo de enseñanza tradicional. En particular, no trata de reemplazar la interacción social que se da entre tutor y estudiante, por ejemplo en el aula. Segundo, no hay que olvidar las ventajas de sistemas de tutoría inteligente individualizados, 28 *Informática Educativa* 7 (1), 1994 especialmente su alta disponibilidad sin mucha infraestructura y su potencial de soportar trayectorias diferentes, en términos de velocidad y selección del nivel de contenido, adaptadas a las necesidades del estudiante. Además, la crítica fundamental de los STI tiene puntos débiles en cuanto a su fundamentación empírica e histórica [2].

El aspecto social es muy importante en cada concepto integral de enseñanza-aprendizaje. Y es cierto que la interacción entre personas (estudiante-estudiante o estudiante-profesor) no fue definida en forma suficientemente precisa para el uso de STI. En la práctica el tutor computarizado se integra de una forma u otra en la infraestructura del sistema educativo. Es necesario definir claramente el o los modos de integración del tutor inteligente en el aula, así como su relación con el resto del sistema educativo.

Así mismo, es necesario aceptar que pocos sistemas inteligentes de tutoría han llegado a un alto nivel de desarrollo que les permita realmente ocupar un espacio substancial en el curriculum (en cobertura de la materia, en robustez, en facilidad de interacción, etc.) y ser usados de forma práctica fuera del laboratorio.

Estos problemas explican, pero no justifican, la negligencia hacia todos los aspectos de la integración del STI con el sistema educativo existente y de la definición de su papel en el contexto social del aprendizaje.

Si la negligencia de la interacción social y la poca aplicabilidad directa son seguramente puntos débiles en los estudios de sistemas de tutoría inteligentes, también hay logros significativos:

- el uso de mecanismos de razonamiento experto en varias áreas para guiar, contrastar y juzgar las soluciones del estudiante;
- el desarrollo de representaciones sofisticadas tanto de los conocimientos ideales (*domain knowledge*) como del estado de conocimiento real del estudiante incluyendo deficiencias y conceptos falsos (modelo del estudiante).
- la descripción e implementación de algoritmos de diagnóstico para el análisis profundo y detallado de errores, algunos de ellos aplicables a una gran variedad de problemas.

El problema del diagnóstico en informática educativa, junto con el de la representación de conocimientos, ha sido un área de investigación aplicando métodos de Inteligencia Artificial. Empezando con modelos de tipo "overlay" [3] y 29 Asistencia inteligente en ambientes de aprendizaje colaborativos, [2]. con librería de errores en forma de reglas de producción [4], se ha llegado a modelos basados en métodos de diagnóstico generales [5], en el uso de meta-interpretación en programación lógica [6, 7] y en la síntesis inductiva de programas lógicos combinado con técnicas de razonamiento hipotético [89]

Estas técnicas son teóricamente bien fundadas y, con excepción del último enfoque, no necesariamente muy costosas en términos computacionales.

Así pues, podemos constatar logros substanciales en lo que se puede clasificar como tareas analíticas y reactivas con respecto a lo que entrega el estudiante-usuario en términos del área. Aunque todavía predominan sistemas inteligentes que no permiten la libre definición de problemas por parte del estudiante, como es característico de micromundos o ambientes interactivos de aprendizaje, hay una tendencia fuerte a combinar sistemas de alta interactividad y flexibilidad con mecanismos de ayuda inteligentes. El denominador común podrían ser los ambientes de aprendizaje adaptivos [10]. Para no sólo responder a lo que hizo el estudiante-usuario en un tal ambiente altamente interactivo sino para inducir también una cierta trayectoria de aprendizaje en forma flexible, los ambientes de aprendizaje adaptivos deberían organizar un curriculum individualizado en forma dinámica. En este aspecto, según el "estado de arte" actual, encontramos dificultades en organizar el curriculum, es decir la selección y secuenciación de las materias presentadas, en forma dinámica para materias amplias y no solo muy especializadas compare [11, 12].

Otro problema para el cual no existen soluciones convincentes en forma de mecanismos inteligentes es el control adaptivo del estado de motivación del estudiante. Si comparamos la información disponible para el sistema en este aspecto con la que usa, por ejemplo, un profesor ante un estudiante en el aula, encontramos que los sistemas actuales carecen de "sentidos" muy

importantes (por ejemplo la observación de gestos y mímica) para realmente evaluar el estado de motivación. En las actividades de un profesor humano el control adaptativo del estado de motivación es un parámetro muy importante para la organización dinámica del curriculum. Dadas estas limitaciones, parece razonable delegar ciertas tareas demasiado complejas o no suficientemente accequibles por el sistema a instancias externas: en primer lugar al profesor, pero también a la interacción entre estudiantes. Esto es viable en ambientes de aprendizaje colaborativos que permiten y soportan la comunicación y cooperación entre varios participantes humanos.

En resumen, la propuesta que vamos a elaborar consiste en ampliar la noción de ambientes inteligentes/adaptivos de aprendizaje para incluir trabajo y comunicación en grupo. Eso permite la delegación de ciertas tareas críticas a participantes humanos y crea una situación en la cual los mecanismos inteligentes 30 Informática Educativa 7 (1), 1994 no son responsables por cada aspecto del proceso de enseñanza-aprendizaje, una meta muy alta que en parte puede haber causado la impracticabilidad de los STI.

Antes de presentar un proyecto práctico, que constituye un primer paso hacia la realización de ambientes de aprendizaje colaborativos que integran mecanismos inteligentes, hay que mencionar un problema de terminología: El término de sistemas o ambientes de aprendizaje colaborativos es también usado para denominar sistemas educativos de múltiples agentes artificiales (pero todavía con un estudiante). Ese tipo de sistemas fue concebido según los principios de inteligencia artificial distribuida, sin embargo confronta las mismas limitaciones mencionadas con respecto al control de motivación y a la organización dinámica del curriculum.

En adelante el termino "ambiente/sistema de aprendizaje colaborativo" se refiere a ambientes/sistemas de aprendizaje en grupo que incluyen mecanismos de comunicación y cooperación pero no son necesariamente inteligentes. Algunos ejemplos de trabajos en la misma dirección [13, 14]. Se discutirá también cómo y dónde se pueden integrar mecanismos inteligentes con estos ambientes.

EL PROYECTO COSOFT TECNOLOGÍA BÁSICA Y PRINCIPIOS GENERALES

La extensión de ambientes de aprendizaje computarizados hacia el soporte de aprendizaje en grupo es facilitada por recientes desarrollos técnicos, particularmente por redes locales de tipo LAN que conectan computadores en un aula, pizarrones electrónicos que pueden desempeñar el papel de la pizarra como medio de exposición y presentación, así como por computadores compactos que se integran en un aula para el uso local por los estudiantes sin impedir el contacto cara a cara en el sentido tradicional.

El proyecto COSOFT (COmputer SuppOrt for Face-to-face Teaching) se basa en estas tecnologías usándolas para enriquecer la interacción en el aula en el sentido de "ubiquitous computing" [15]. Es decir, el enfoque COSOFT no quiere definir o crear una infraestructura nueva para la educación sino apunta a mejorar la calidad de la enseñanza en su estructura existente. Así se evita que el sistema se tenga que responsabilizar de todos los aspectos relevantes de la situación. En cambio, el sistema puede complementar el ambiente tradicional siempre y cuando se pueda esperar un valor agregado.

METAS PEDAGÓGICAS Y AREAS DE SOPORTE 31 Asistencia inteligente en ambientes de aprendizaje colaborativos Definiendo la orientación del proyecto COSOFT nos focalizamos explícitamente en la interacción en el aula dirigida por el profesor [16]. Eso incluye no sólo la fases de exposición por parte del profesor sino también discusiones más o menos guiadas y también ejercicios supervisados. Conscientes que la pedagogía moderna favorece un papel activo del estudiante y critica la dominancia del profesor en la interacción educativa, aceptamos, sin embargo, que el profesor tiene un rol destacado en la interacción en el aula, un rol que también se asocia con una responsabilidad particular en cuanto a garantizar que las diferentes formas de interacción sean productivas y para evitar cualquier forma de discriminación en el uso de los recursos educativos. Dentro de este marco hay diferentes modos de involucrar en forma mas activa a los estudiantes, como por ejemplo:

(a) Diálogos Socráticos

En un diálogo socrático [17] el profesor guía al estudiante (o a los estudiantes) en forma indirecta a través de preguntas o ejemplos contrarios, facilitando así el descubrimiento de un concepto nuevo o la solución creativa de un problema. En este sentido, el diálogo socrático se basa en los propios recursos del estudiante e incentiva una forma autónoma de pensar y razonar. Esto requiere mucha flexibilidad por parte del profesor, porque tiene que evaluar y responder en el instante a proposiciones idiosincráticas del estudiante . El estilo socrático no es compatible con el uso de material que incluye resultados previamente preparados, pues requiere un manejo muy flexible de todos los medios de presentación usados.

(b) Ejercicios preparados

Ejercicios preparados son una forma muy común de incentivar las actividades de los estudiantes. Típicamente se presentan luego de una fase de exposición introductoria por parte del profesor. El espectro de diferentes tipos de ejercicios se extiende de simples "tests" (memorización) a la resolución de problemas abiertos en la cual los estudiantes tienen que aplicar sus conocimientos a situaciones nuevas. Los estudiantes pueden trabajar en grupos o separados, y en muchos casos se usa material preparado ("hojas de trabajo"). Dadas las cifras reales de estudiantes por clase, es muy difícil supervisar y retroalimentar los trabajos individuales en forma adecuada.

(c) Reflexión del proceso de aprendizaje

En general, se habla de reflexión en el proceso de aprendizaje si una actividad del primer nivel (nivel objeto, como por ejemplo la discusión de un concepto o la solución de un problema) se toma como objeto en un proceso de segundo grado (*meta-learning*). Este aspecto es bastante complejo, ya que 32 Informática Educativa 7 (1), 1994 actividades reflectivas se pueden referir a aspectos muy diferentes. Desde el punto de vista de las ciencias educativas domina el aspecto de la comunicación interhumana o de la interacción en el grupo. Esta perspectiva fue inicialmente inspirada por la teoría de la comunicación pragmática de Watzlawick [18]. Por el otro lado, estudios cognitivos tienden a focalizarse en la elaboración de la materia, particularmente en los procesos de resolución de problemas como por ejemplo [19]. Para facilitar la reflexión, se trata de hacer disponible y accesible el respectivo material, por ejemplo en forma de un protocolo de interacción o una huella de la resolución de un problema.

VALOR AGREGADO DE LA NUEVA TECNOLOGÍA EN EL AULA

La introducción de la nueva tecnología en el aula se tiene que justificar por el valor agregado que se puede esperar. Por el momento, este valor agregado es una hipótesis que debe ser sometida al examen empírico. Las siguientes características definen el posible valor agregado:

Presentación multimedial

El pizarrón electrónico une la flexibilidad y reactividad del tablero tradicional con el acceso fácil a material preparado, incluyendo gráficas animadas, simulaciones, audio y video. En este aspecto presenta toda la flexibilidad para facilitar diálogos socráticos. No obstante, para integrar material preparado también en forma imprevista (por ejemplo casos contrarios surgiendo del diálogo) es necesario proporcionar mecanismos de recuperación de información muy flexibles.

Memoria

Todas las presentaciones, sean preparadas o hechas durante la clase (usando editores de texto y gráfica), se pueden fácilmente archivar y volver a utilizar. Esta capacidad ayuda mucho a las actividades de reflexión. Para hacer uso óptimo de la capacidad de protocolar la secuencia de la clase, se tienen que inventar representaciones flexibles a diferentes niveles de granularidad y herramientas gráficas para la navegación.

Transferencia de información

La red local permite la transferencia de información instantánea entre los puestos de trabajo de los estudiantes y el profesor en ambas direcciones, así como también entre estudiantes. Eso facilita la distribución de materiales de trabajo y la discusión de los resultados. 33 Asistencia inteligente en ambientes de aprendizaje colaborativos

Trabajo simultáneo

Usando protocolos de comunicación para la cooperación en grupo, varios estudiantes o también el profesor y estudiante(s) puedan compartir un espacio de trabajo, por ejemplo para solucionar un problema geométrico, en sus respectivos computadores. Si uno de los puestos es el pizarrón electrónico, eso facilita la discusión de ejercicios dentro de la clase.

Archivos individuales

Los estudiantes pueden directamente copiar a un medio externo (diskette) la información presentada o ejemplos elaborados, para su uso privado. El problema técnico que se plantea aquí es la máxima mantención de la estructura original de las unidades potencialmente heterogéneas. Se puede solucionar este problema usando estándares de intercambio de documentos.

La primera fase de realización del proyecto COSOFT se enfocó hacia los aspectos de presentación multimedial, transferencia de información y trabajo simultáneo [16], facilitando así tanto la exposición flexible por parte del profesor como ejercicios preparados.

La figura 1 muestra el escenario de COSOFT centrado en el pizarrón electrónico de tipo "LiveBoard" (un producto de Xerox - cf. [18]). El *LiveBoard* es una pantalla larga de aproximadamente 1m x 1.4m que se puede manipular con un lápiz electrónico directamente en su

superficie similar a un "lápiz óptico". El lápiz tiene todas las funciones del ratón pero también sirve para escribir a mano. El *LiveBoard* se encuentra en una posición que es visible para todos los estudiantes en el aula. Se distingue de un proyector no sólo por la interacción, sino también por la relación física con el que lo usa, normalmente el profesor. Aquí se perciben todos los gestos en forma natural y directa. 34 *Informática Educativa* 7 (1), 1994 Figura 1. Escenario de aprendizaje en COSOFT

Manejo de material interactivo

Como apoyo a la presentación flexible y multimedial extendimos una herramienta para la creación interactiva de interfaces gráficas [20]. Esta herramienta (CENTER) se basa en los productos estandar X-Windows y OSF/Motif. Ofrece objetos y funciones que son o especializaciones de objetos con muchos parámetros o agregaciones de funcionalidad que no existen en esta forma en las herramientas básicas. Lo importante es que para crear una interfaz compuesta de tales objetos se ofrece un editor interactivo (TOE) que requiere únicamente la programación para la provisión de funciones particulares a la aplicación. Hay clases de objetos para elementos tales como imágenes o botones asociados con sonido. Según las primeras experiencias se pueden producir "marcos de presentación" multimediales con esfuerzo mínimo.

Las mismas herramientas (CENTER y TOE) se usan también para crear materiales multimediales interactivos para ejercicios. Este material es organizado en forma de marcos similares a los marcos de presentación, y puede incluir simulaciones interactivas. Mientras el material de presentación usado por el profesor en la exposición típicamente ofrece toda la flexibilidad (por ejemplo de ajustar los parámetros de una simulación), el material de ejercicios viene con ciertas restricciones o instanciaciones de valores que definen los problemas particulares que el estudiante tiene que reflexionar. El uso del editor TOE hace 35 Asistencia inteligente en ambientes de aprendizaje colaborativos posible la fácil transformación de marcos de presentación en marcos de ejercicio u "hojas de trabajo electrónicas". Hasta el momento se crearon aplicaciones en áreas como economía (modelos de inversión), idiomas, matemáticas (discusión de funciones y derivadas con soporte gráfico y simbólico) y geografía.

Soporte a la cooperación

En cuanto al soporte a la comunicación y cooperación, es decir a la transferencia de información y el trabajo simultáneo, había que definir un modelo apropiado a las necesidades de la situación en el aula, particularmente a los cambios entre la exposición por el profesor y ejercicios individuales o en grupo. Para transferir en el instante información entre los ambientes del profesor y de los estudiantes, es necesario copiar información parcial (no necesariamente todos los elementos y parámetros del ambiente). Además se tiene que tomar en cuenta que los ambientes de los estudiantes y el del profesor pueden ser bastante diferentes, lo que implica que la transferencia tiene que ser posible también entre ambientes que son heterogéneos en términos de la estructura de sub-espacios y de los objetos que contienen. Para formular los requisitos acerca del trabajo simultáneo, partimos de una situación característica en la discusión de ejercicios: El profesor quiere discutir un resultado particular en la clase. Por eso tiene que copiar los respectivos datos de un estudiante e integrarlos en su ambiente de trabajo sobre la estación del *LiveBoard*. Los objetos que contienen estos datos en los ambientes del estudiante y del profesor tienen que ser sincronizados y compartidos entre ambos, hasta que se termina la sesión conjunta. Esto significa que ambos tienen el derecho de modificar los datos u objetos y dichas modificaciones se aplican a la vez a los dos ambientes. Cuando se termina la sesión se desconectan los objetos y quedan disponibles para ser modificados en forma individual.

Los aspectos más importantes de este modelo de comunicación y cooperación es, entonces, la posibilidad de copiar parámetros entre objetos y de sincronizar objetos en diferentes ambientes, potencialmente heterogéneos, en forma dinámica y parcial. Lo que no es tan importante en nuestro escenario es la composición de diferentes resultados intermedios para formar un producto común permanente. Este último aspecto es central, por ejemplo, en sistemas de autoría cooperativos, en los cuales se trata de combinar diferentes partes o versiones elaboradas por diferentes participantes [21].

El modelo usado en COSOFT se basa en la noción de objetos compartidos a nivel de la interfaz ("shared UI objects" - [20]). Fue implementado como extensión de la biblioteca de tipos de objetos y funciones que usa el generador de interfaces CENTER. Se definió un protocolo de comunicación realizando los siguientes principios: Llamemos una "aplicación" un programa interactivo, y una "instancia 36 Informática Educativa 7 (1), 1994 de una aplicación" una versión de un programa interactivo corriendo como un proceso en un ambiente de procesos múltiples. Para que sean "conectables" dos aplicaciones (o mejor dicho, sus respectivas instancias) sus interfaces tienen que ser definidas en términos de objetos y funciones de CENTER. Para poder comunicarse entre las diferentes instancias de un conjunto, cada instancia cuando es llamada tiene que registrarse con un servidor central. Este servidor recibe y mantiene en forma dinámica la información sobre los objetos "conectados" entre diferentes instancias de aplicaciones como también ciertas restricciones en los derechos de acceso etc. La sincronización se realiza a través de una modificación del procesamiento de eventos y acciones activadas (*callbacks*) que ya no reside en las aplicaciones individuales sino que se delega al servidor. Es decir, un evento recibido por un objeto O en una instancia A1 de una aplicación A, antes de disponer una reacción dentro de A se transmite al servidor, el cual multiplica la acción para todas los objetos O', O", ... conectados a O. Estos objetos pueden residir tanto en instancias de la misma aplicación A como en instancias de otras aplicaciones. Antes de establecer una conexión entre dos o más objetos, se tiene que definir cuáles son los atributos que se tienen que sincronizar y cuáles pueden ser diferentes (e.g, entre células que contienen textos o cadenas modificables, es esencial sincronizar el contenido pero no necesariamente el tamaño o el tipo de letra). Estas relaciones son definidas previamente para diferentes tipos de objetos como botones, menús, y otros. En un conjunto de objetos que se conectan se tiene que identificar un objeto especial que "domina" los demás, lo que significa que en el momento inicial los valores de los atributos esenciales son copiados a los otros en forma destructiva.

Más detalles del mecanismo de comunicación se encuentran en [20]. Una característica esencial de este mecanismo es que no requiere la reprogramación de aplicaciones basadas en CENTER para convertirlos del tipo individual al tipo multi-usuario, con excepción de la orden de registrar la aplicación con el servidor. Esto hace que todos los mecanismos de comunicación estén automáticamente disponibles. También tenemos mecanismos interactivos para establecer la comunicación entre objetos, por ejemplo uno en el espacio de trabajo del estudiante y el otro en el ambiente de presentación sobre el *LiveBoard*. En este sentido, el mecanismo de comunicación es realmente independiente de la aplicación.

Sin embargo, dado que el mecanismo está localizado a nivel de la interfaz, la transferencia y sincronización de información se limita a lo que está representado en forma explícita en los objetos de la superficie. La sincronización de estructuras internas requiere mecanismos particulares. En realidad, el mismo protocolo de comunicación también permite el intercambio de mensajes arbitrarios de diferentes instancias de aplicaciones (en forma de un "*remote procedure call*"). Pero en este caso se tiene que definir la semántica de estos mensajes para cada caso especial. 37

Las experiencias con este mecanismo de comunicación son muy positivas. Normalmente los marcos interactivos de presentación, o las hojas de trabajo electrónicas, contienen toda la información relevante en objetos de la superficie. También descubrimos que puede ser muy eficiente compartir objetos en forma indirecta: Por ejemplo, para compartir dos funciones y sus grafos, es suficiente con conectar los campos que contienen el término simbólico que define la función y los botones que activan la generación del grafo. Entonces los grafos son prácticamente sincronizados sin hacer la conexión directa (la cual sería técnicamente mucho más costosa).

Supervisión inteligente en COSOFT

En la versión actual del sistema de COSOFT, se integró un mecanismo de diagnóstico inteligente para la supervisión de ejercicios a la aplicación de "funciones y derivadas". Este mecanismo se basa en el principio de "diagnóstico deductivo" descrito en detalle en [22]. En el ambiente colaborativo, este mecanismo es una función de la hoja de trabajo del estudiante. Es invocado siempre cuando el estudiante confirma una solución dada. Si se encuentra un error, se identifica el sub-problema que fue mal solucionado y se explica el error y su corrección en un diálogo con el estudiante.

La información sobre los errores no sólo es de interés para el estudiante sino también para el profesor. Se pueden definir ciertas clases de errores de interés general, que al encontrarlos son transmitidos (en forma de un mensaje) al ambiente del profesor, donde son archivados en una lista que se puede acceder a través de un botón iconizado. La llegada de mensajes se indica por la forma del botón y por un sonido característico. El profesor puede reaccionar a estos mensajes cuando le convenga. Entonces el procedimiento es el siguiente: (1) Abrir la lista de mensajes; cada elemento indica el estudiante y el tipo de error encontrado. (2) Seleccionando un elemento de la lista, el profesor recibe más detalles sobre el error concreto. (3) Si le parece adecuado, el profesor usa el mecanismo interactivo para conectarse con el ambiente del estudiante y entrar en una sesión sincronizada.

Mecanismos de ese tipo pueden substancialmente mejorar la calidad de la supervisión de ejercicios. Su función principal es la selección de casos interesantes para ser discutidos en público. En el ejemplo, el criterio son ciertos tipos de errores que requieren técnicas bastante sofisticadas para ser reconocidas y distinguidas. Pero tanto el criterio como la naturaleza del mecanismo de selección son parámetros completamente variables. Hay criterios también razonables que requieren mucho menos "inteligencia", como por ejemplo la frecuencia de entradas o simples estadísticas de soluciones que difieren de lo que se anticipó. También se puede considerar el rendimiento promedio del estudiante para evitar un desequilibrio entre la capacidad individual y la dificultad del problema.

En el futuro queremos integrar más mecanismos inteligentes al ambiente COSOFT. La próxima sección describe y sistematiza algunas ideas generales a este respecto.

EL USO DE TECNICAS INTELIGENTES EN AMBIENTES DE APRENDIZAJE PARA GRUPOS

Recordemos que el uso de técnicas inteligentes en ambientes de aprendizaje para grupos, no necesariamente pone problemas más exigentes que el caso individual: tareas demasiado complejas, actualmente fuera del alcance de mecanismos computacionales, se pueden delegar a los participantes humanos. Sin embargo, en el diseño del ambiente hay que definir un modelo integral de las posibles interacciones, aun cuando espera que "comprenda" y controle todas estas

interacciones. En la primera versión del ambiente COSOFT nos concentramos en el aprendizaje en el aula, dirigido por el profesor. Otros modelos podrían ser la interacción de grupos independientes sin coordinación central, la interacción entre un estudiante y un profesor remoto en casos de necesidad de ayuda, etc.

En el ambiente actual de COSOFT se emplea un mecanismo inteligente para la supervisión local de ejercicios. Este mecanismo puede contactar al profesor si un error encontrado cumple con ciertos criterios predefinidos. Así se evita una avalancha de noticias que sería casi de la misma complejidad que el problema de supervisar el grupo sin ayuda técnica. En un sentido general, este mecanismo actúa como un agente local que proporciona consejos para el estudiante y selecciona información relevante para el profesor. En COSOFT no hay ningún tipo de coordinación central entre los mecanismos o agentes locales; este tipo de coordinación podría ayudar a una mejor selección de la información o también a determinar grupos o individuos entre los alumnos que podrían proporcionarse ayuda entre ellos (vea abajo). Un modelo de coordinación central se podría realizar recurriendo a la teoría de sistemas de inteligencia artificial distribuida.

Otra extensión del mecanismo de supervisión sería la introducción de un modelo del estudiante. Como cada instancia del mecanismo trabaja en un ambiente individual, eso todavía no implicaría la creación de un modelo de múltiples participantes. Ya se mencionaron decisiones simples basadas, por ejemplo, en información estadística sobre el rendimiento del estudiante. También podría ser interesante detallar el modelo con respecto a ciertas clases de problemas. Pero, como en cualquier ensayo de modelaje de estudiantes, se tiene que considerar la 39 Asistencia inteligente en ambientes de aprendizaje colaborativos precaución formulada por Self [23] cuando señala que sólo se debería incluir en el modelo la información que el sistema realmente sea capaz de explotar para modificar su comportamiento.

Para evaluar el potencial de modelos de múltiples estudiantes, es importante recordar diferentes tipos de modelos de estudiantes individuales: los modelos de tipo pasivo funcionan como recipientes de datos sobre el estudiante que otros componentes del sistemas pueden consultar directamente para recuperar estos datos. Un modelo de tipo activo simula el comportamiento del estudiante ante una clase de problemas. Su primera función es la predicción, pero se tiene que considerar que el resultado de la simulación constituye una hipótesis que puede divergir el comportamiento real. En el enfoque de Kono [9] se analizan estas divergencias para mantener el modelo consistente. Los modelos pasivos se pueden extender a situaciones con múltiples estudiantes como simple suma de las partes. Técnicamente solo se tiene que proveer el acceso a todos los modelos parciales desde todos los componentes que utilizan sus datos. Eso también permite la computación de parámetros relativos, e.g, del rendimiento individual en relación con el promedio del grupo. La extensión de modelos activos para el caso múltiple es mas exigente, porque se tienen que considerar hipótesis del tipo "qué pasaría si el estudiante A recibiera ayuda del estudiante B?", siempre considerando las respectivas bases de conocimientos individuales. Es decir, se trataría de simular también la interacción entre participantes, lo que otra vez nos lleva a considerar modelos distribuidos con múltiples agentes.

La expectativa asociada con modelos múltiples es la coordinación de interacciones entre estudiantes y grupos de trabajo sin la intervención necesaria por parte del profesor. Para no entrar directamente en problemas de sistemas inteligentes distribuidos, se puede seguir una estrategia partiendo de modelos pasivos, agregando más y más detalles individuales relevantes y relaciones interindividuales para finalmente considerar los modelos activos.

Todavía falta un factor muy importante en este escenario: el profesor. Como en la situación colaborativa el sistema no asume (normalmente) el rol del profesor, éste es otro factor externo. El aspecto más importante en este sentido es el progreso de la elaboración de la materia o de la secuencia de ideas expuestas, lo que depende primordialmente de las decisiones del profesor. Aspectos que tienen que ver con la personalidad del profesor son de menor relevancia, porque influyen en forma menos directa y son difíciles, y hasta imposibles de evaluar. En cuanto a la elaboración de la materia, supongamos que el material de presentación sea organizado en forma de hipermedia (cómo es realmente el caso de COSOFT). Entonces no es difícil mantener un modelo global de la secuencia de exposición en la forma de una trayectoria a través de los marcos de hipermedia. Una 40 Informática Educativa 7 (1), 1994 representación formal de lo que contiene cada marco ya permite ciertas formas de razonamiento sobre lo que fue expuesto y, en un sentido ideal, debería haber sido adquirido por el estudiante. Un "modelo de exposición" en este sentido puede ayudar a crear el modelo de un estudiante ideal, el cual puede por ejemplo servir para seleccionar problemas (ó descartar ciertos ejercicios que no corresponden a los acontecimientos actuales). Una descripción del posible uso de un modelo ideal en el caso individual se encuentra en Hoppe [22].

El modelo de exposición puede también retroalimentar el proceso de presentación: Se pueden detectar inconsistencias o la falta de ciertos elementos que son necesarios para introducir un concepto nuevo. En estos casos se pueden generar mensajes de advertencia para el profesor. La última decisión sobre qué hacer está en manos del profesor; estas advertencias no tienen que ser siempre correctas y basadas en conocimientos completos e inferencias infalibles.

Otra aplicación muy interesante es la ayuda a la recuperación de información de una base de material de instrucción, con el fin de reaccionar a problemas o preguntas imprevistas. En este caso, el modelo de exposición puede definir un marco contextual para preformular la pregunta, de tal modo que el profesor sólo tenga que agregar uno pocos términos claves muy especiales. Por ejemplo, si en una clase de geografía se habla del desarrollo económico de ciertos países latinoamericanos y se quiere hacer una comparación con los "países tigres" en Asia, podría ser suficiente entregar los nombres de los países (o una descripción más general) para encontrar la información análoga. La analogía sería extraída del modelo de exposición.

CONCLUSION

La integración de mecanismos inteligentes con ambientes colaborativos de aprendizaje pone de manifiesto una gran variedad de problemas y define así todo un programa de investigación de los sistemas de tutoría inteligentes individualizados. No se trata de resolver todos los problemas a la vez para tener un sistema práctico y funcional, sino de usar estrategias incrementales que de por sí pueden ir produciendo sistemas prácticos.

Con el proyecto de COSOFT creamos un punto de partida en forma de un ambiente práctico y operacional que permita la extensión hacia los diferentes aspectos discutidos. En este contexto, quiero mencionar con agradecimiento las contribuciones de Jian Zhao, quien proporcionó las herramientas básicas para la interfaz e implementó el modelo de comunicación, Nelson Baloian, quien es el responsable de las herramientas de tutoría, la representación del curriculum y el 41 Asistencia inteligente en ambientes de aprendizaje colaborativos

modelo de exposición. Y finalmente, Frank Tewissen, quien implementó varios ejemplos y extendió las herramientas básicas. El proyecto de COSOFT forma parte de la división de "Ergonomía Cognitiva en Interfaces al Usuario" del Instituto de Sistemas de Información y Publicación del Centro Nacional de Investigación en Informática en Alemania (GMD).

REFERENCIAS

- ¹ CLANCEY, W. (1992). Representation of Knowing: In defense of cognitive apprenticeship. *Journal of AI on Education*, vol. 3 (2), pp. 139-168
 - ² HOPPE, H.U. (1993). Deductive error diagnosis and inductive error generalization. *Journal of AI in Education*, vol. 4.
 - ³ CARR, B. & GOLDSTEIN, I.P. (1977). Overlays: a theory of modeling for computer-aided instruction. *AI Memo 406*. MIT, Cambridge (Mass.)
 - ⁴ BROWN, J.S. & BURTON, R.R. (1978). Diagnostic models for procedural bug in basic mathematical skills. *Cognitive Science*, vol. 2, 155-191
 - ⁵ SELF, J. (1993). Model-Based cognitive diagnosis. *User Modeling and User-Adapted Interaction*, vol. 3, pp. 89-106.
 - ⁶ HOPPE, H.U. (1992). Deductive diagnosis in Intelligent Tutoring Systems. *Proceedings of the East-West Conferences on Emerging Computer Technologies in Education*. (Moscu, Rusia, Abril 1992).
 - ⁷ BELLER, S.. & HOPPE, H.U. (1993). Deductive error reconstruction and classification in a logic programming framework. *Proceedings of the World Conference on Artificial Intelligence and Education*. (Edinburgo, Escocia, Agosto 1993).
 - ⁸ IKEDA, M.; MIZOGUCHI, R.; KAKUSHO, O. (1989). Student model description language SMDL and student modeling inference systems SMDL. *Transactions of the Institute of Electronics, Information and Communication Eng. D II*. vol. J72-D-II. pp.112 - 120.
 - ⁹ KONO, Y.; IKEDA, M.; MIZOGUCHI, R. (1993). THEMIS: a nonmonotonic inductive student modeling system. *AI Tech, Report, ISIR*, Osaka University, AI-TR-93-3.
 - ¹⁰ JONES, M. & WINNE, P.H. (1992). *Adaptive Learning Environments*. Berlin: Springer.
- 42 Informática Educativa 7 (1), 1994

¹¹ McCALLA G.I. (1992). *The search of adaptability, flexibility and individualization: approaches to curriculum in Intelligent Tutoring System*. En M. JONES & P.H. WINNE (1992). pp. 91-122

¹² HARTLEY, J.R. (1992). *The Curriculum and instructional tasks: goals, strategies, tactics for interactive learning*. En JONES & WINNE (1992), pp. 123-146.

¹³ SCARDAMALIA, M. & BEREITER, C. (1992). Technologies for Knowledge Building Discourse. *Communications of the ACM*, vol 36, pp. 37-41.

¹⁴ KOSHMANN, T.D.; FELTOVICH, P.J.; MYERS, A.C.; BARROWS, H.S. (1992). Implications of CSCL for problem-Based_Learning. *SIGCUE Outlook*, vol. 21 (3), pp. 32-35.

¹⁵ WEISER, M. (1991). The computer for the 21st century. *Scientific American*, vol. 265 (3), pp. 94-104.

¹⁶ HOPPE, H.U.; BALOIAN, N; ZHAO, J. (1993). Computer support for teacher-centered classroom interaction. *Proceedings of the International Conference on Computers in Education (ICCE '93)*, (Taipei Taiwan, Diciembre 1993).

¹⁷ WAGENSCHIN, M. (1968). *Verstehen Lehren. Weinheim (FRG); Beltz.*

¹⁸ WATZLAWICK, P.; BEAVIN, J.H.; JACKSON, D.D. (1967). *Pragmatics of Human Communication*, New York; Norton.

¹⁹ COLLINS, A. & BROWN, J.S. (1988). The computer as a tool for learning through reflection. En MANDL, H. & LESGOLD, A. (eds.). *Learning Issues for Intelligent Tutoring Systems*, New York; Springer, pp. 1-18.

²⁰ ZHAO, J. & HOPPE, H.U. (1993). Supporting flexible communication in heteroge-neous multi-user environments. *Internal paper* (sometido para publicación), GMD-IPSI, Darmstadt (FRG).

²¹ HAAKE & HAAKE (1993). Take cover: exploiting version support in cooperative systems. *Proceedings of InterCHI'93*, (Amsterdam, Holanda, Abril 1993).

²² HOPPE, H.U. (1993). Cognitive apprenticeship - The emperor's new method. *Journal of AI in Education*, vol. 4, pp. 49-54.

²³ SELF, J. (1988). Bypassing the intractable problem of student modeling. *Proceedings of STI '88*, (Montreal, Canada)