

Práctica 1. Sistemas de Producción

Curso 2020-2021

1 Introducción

La Robótica Social Asistida, SAR por sus siglas en inglés *Socially Assistive Robotics*, comprende el conjunto de plataformas robóticas que ofrecen asistencia a las personas mediante interacción social. Con el reciente crecimiento de la robótica, este campo está siendo aplicado actualmente a múltiples ámbitos de la vida humana. Así surge el desarrollo de plataformas capaces de ayudar a personas en tareas típicamente conducidas por un especialista, aplicándose por ejemplo en el tratamiento del trastorno del espectro autista (TEA) o en sesiones de rehabilitación física, donde el robot adopta el papel del especialista. Los beneficios del uso de este tipo de aplicaciones, capaces de mejorar la motivación y seguimiento de las actividades, han sido ampliamente demostrados. Un ejemplo de ellas es el proyecto NAOTherapist para rehabilitación ¹.

El desarrollo de estos sistemas requiere del diseño de las acciones que debe realizar el robot en cada sesión para conseguir el objetivo buscado. Por ejemplo, en el caso de un sistema de rehabilitación física, incluiría los ejercicios o poses que debe realizar el paciente junto con las acciones de interacción social tales como saludar, explicación de los ejercicios, alentar y corregir al paciente, etc. Este tipo de actividades ha sido estudiadas y documentadas en la literatura [1], donde puede encontrarse el esquema de desarrollo típico de las sesiones.

Dado que estos sistemas se ejecutan en entornos reales y, por tanto, dinámicos, deben ser flexibles frente a situaciones o reacciones que puedan romper la secuencia lógica de la sesión. De esta forma, uno de los objetivos fundamentales de las plataformas SAR es conseguir una interacción fluida con el usuario. Para ello, se espera que los robots sean capaces de “tomar decisiones” y cambiar su comportamiento dependiendo de las circunstancias, haciendo que el diseño de las sesiones sea una tarea complicada que implica conocimiento experto, tanto para el diseño de las actividades típicas de cada sesión, como para la creación de un sistema robusto frente a cambios.

El objetivo de la Práctica 1 es implementar un sistema de producción (SP) en CLIPS capaz de diseñar y simular la ejecución de una sesión de un especialista con un paciente en la que interactúan un humano y un robot. Para ello, el SBC deberá contener toda la información referente a la definición de los personajes y al entorno, teniendo en cuenta cómo el robot sería capaz de adaptar la sesión frente a posibles cambios y reacciones de los pacientes.

2 Requisitos

En este trabajo nos limitaremos a sesiones donde los pacientes son niños y las acciones que deben realizar se plantean como juegos de distinto tipo. Por ejemplo, en NAOTherapist cada sesión se puede considerar como un juego de imitación donde el niño tiene que copiar las posturas que realice el robot.

¹<https://www.youtube.com/user/NAOTherapist>

El objetivo consiste en plantear un escenario en el que están involucrados un paciente y un robot, cuyo propósito es que ambos interactúen mediante el desarrollo de juegos. Estos pueden ser directamente extraídos de la literatura proporcionada o inspirados en ella, teniendo como requisito que sean colaborativos entre el niño y el robot. Los niños estarán definidos por diferentes personalidades, en función de las cuales reaccionarán de una forma u otra ante los ejercicios de la sesión. Por ejemplo, un niño distraído puede necesitar que se le repita la explicación del ejercicio, uno impaciente que las sesiones sean más cortas, etc. Así, el sistema deberá adaptarse a las respuestas de cada niño en función de su personalidad.

El objetivo es la creación de un SBC completo que contenga toda la información referente al desarrollo de una sesión (definición de personajes y personalidades, acciones para llevar a cabo la sesión, posibles reacciones por parte de los niños, etc.), teniendo en cuenta posibles desviaciones en la sesión a las que el robot se verá obligado a reaccionar y modificar sus acciones hasta conseguir finalizar la sesión.

Para el diseño de los juegos que debe incluir la sesión podéis consultar en Aula Global el documento *juegos.pdf*, de entre los que tendréis que escoger dos de ellos para su implementación en el SP. Para los juegos seleccionados se tendrán que definir, además del funcionamiento normal, dos posibles alteraciones debidas a distintas personalidades de los niños. En cualquier caso, se tiene que intentar que las reglas del SP sean lo más genéricas posibles para que puedan tratar cualquier juego y alteración.

3 Se pide

Para completar la práctica hay que realizar los siguientes pasos:

1. Identificar el problema a resolver. Identificar todas las posibles acciones y recursos implicados en el escenario. Identificar los posibles perfiles de paciente o niño y sus respuestas ante el desarrollo de la sesión. Las diferentes entradas al sistema se introducirán mediante instancias iniciales en la Base de Hechos. En la parte de evaluación del sistema se deben simular varias entradas diferentes (distintos pacientes y tipo de sesión) y analizar las soluciones (cómo se desarrolla la sesión en función a sus reacciones) propuestas por el Sistema de Producción.
2. Adquirir el conocimiento necesario para poder implementar el sistema.
3. Hacer la conceptualización.
4. Implementar en CLIPS la **jerarquía de clases** necesarias para realizar el sistema.
5. Crear las reglas necesarias para simular las sesiones siguiendo las siguientes pautas:
 - Hacer una regla inicial en la que se establezca la estrategia de control *random* y se ejecute el comando (`dribble-on <fichero>`) para que todos los *printout* del programa se salven en un fichero. Antes de acabar la ejecución hay que ejecutar el comando (`dribble-off`).
 - Las reglas deben tener los *printout* necesarios para poder hacer las trazas de cada ejecución y comprobar los resultados obtenidos.
 - Las reglas deben ser lo más generales posibles para tratar con cualquier escenario y configuración.
 - En las reglas no se puede utilizar ninguna función de CLIPS, salvo las mencionadas anteriormente para hacer las trazas. Es decir, la parte derecha de las reglas solo puede contener comandos de CLIPS para añadir, modificar o borrar la base de hechos.
6. Hacer 5 pruebas del sistema generado. Para ello definir las instancias necesarias para simular distintos escenarios (en función de las personalidades de cada niño y las respuestas que estos puedan tener). Ejecutar el SP 5 veces, con distintas entradas, guardando las trazas en ficheros diferentes. Estas deben proporcionar información del razonamiento seguido y mostrar la salida de la sesión obtenida.

4 Ejemplo

Un ejemplo de fichero de salida con las trazas de ejecución del diseño y simulación de una sesión podría ser el que se puede encontrar a continuación:

```
El juego elegido para la sesión es <nombre-juego>
El niño se caracteriza por ser <personalidad-niño>
El robot ejecuta la acción <saludar-niño>
....
El robot realiza la acción <acción-del-ejercicio>
El niño <personalidad-niño> puede realizar:
    <opción-1>
    <opción-2>
El niño realiza <opción-1>
EL robot responde ejecutando la acción <corregir-niño>
...
```

5 Entrega

Los tres primeros pasos referentes a la identificación, adquisición y conceptualización se entregan en los respectivos foros habilitados en Aula Global.

El resto del trabajo se debe entregar por AG subiendo un único archivo por cada pareja llamado **practica1.zip**, que al descomprimirlo **genere un directorio llamado practica1** que contenga los siguientes ficheros:

- Un fichero pdf llamado **practica1.pdf**, con la memoria explicativa de los desarrollos realizados que, en general, no deberá contener listados de código salvo los imprescindibles para la correcta explicación del mismo. La estructura de la memoria será la siguiente:
 - Portada: indicando la titulación, la asignatura, el curso, el número de práctica y para cada uno de los integrantes de la pareja, su nombre, dirección de correo, grupo y campus.
 - Índice: tabla de contenidos del documento.
 - Introducción: breve presentación del documento y su estructura.
 - Manual técnico: descripción (no el código) de la implementación, explicando las reglas y ontología. Se deben justificar las decisiones tomadas durante la implementación.
 - Manual de usuario: descripción del uso del programa realizado.
 - Pruebas realizadas: descripción de las pruebas que se han realizado para verificar el correcto funcionamiento del SP y **análisis de los resultados**.
 - Conclusiones: conclusiones **técnicas** extraídas durante la realización de la práctica.
 - Comentarios personales, críticas constructivas, problemas encontrados, etc.
- Fichero **ontologia.clp** con la jerarquía de clases. En caso de utilizar plantillas y hechos añadirlas a este fichero.
- Fichero **reglas.clp** con las reglas que compondrán el Sistema de Producción.
- Ficheros **prueba-X.clp** que contendrán diferentes estados iniciales utilizados para probar el sistema. Se deberán incluir al menos 5 ficheros de este tipo conteniendo varios estados iniciales, procurando que difieran lo más posible entre ellos. X representará un número del 1 al 5 que identificará la prueba realizada.
- Ficheros **salida-prueba-X.txt** con las trazas de las pruebas realizadas.

Las prácticas que no sigan estos criterios estrictamente serán penalizadas con 2 puntos menos en su calificación.

References

- [1] Misra Turp, José Carlos Pulido, José Carlos González, and Fernando Fernández. Playing with robots: an interactive simon game. In *Proceedings of the 16th Conference of the Spanish Association for Artificial Intelligence (CAEPIA), RSIM workshop*, pages 1085–1095, 2015.