

Inteligencia Artificial

Informe Final

Botta Mariano
Campostrini Esteban

13 de Septiembre del 2013

1 Introducción

El proyecto consiste en diseñar y desarrollar una inteligencia artificial capaz de jugar al juego de cartas conocido como Truco. Una descripción completa del juego puede verse en [1]. El diseño del juego fue pensado para permitir partidas entre un humano y una máquina. No se tuvieron en cuenta todas las características del juego, por ejemplo el canto de "Flor" no está soportado.

Luego de un análisis del juego optamos por representar la inteligencia artificial mediante un sistema basado en conocimientos, más precisamente un sistema experto con reglas de producción de tipo deductivo (Forward Chaining). Si bien no se utilizó un motor deductivo en particular, creemos que los pasos seguidos por el sistema a la hora de tomar las decisiones representan el ciclo base de un motor deductivo.

2 Lenguaje y Desarrollo

El juego junto con la inteligencia fueron desarrollados desde cero en una interfaz web.

La interfaz gráfica está hecha con HTML y CSS. La lógica del juego está escrita en Javascript y está separada en 2 módulos principales.

- El módulo **truco.js** provee las estructuras de datos necesarias para la representación de la información utilizada durante el juego, controla la interacción entre el jugador y la máquina, es decir, determina quien puede jugar en un momento determinado y lleva cuenta del estado global del juego.

- El modulo **ia.js** contiene la inteligencia artificial de la máquina. Aquí se encuentra codificado el razonamiento realizado por la máquina a la hora de tomar una decisión.

A modo de dar una visión general, el sistema funciona de la siguiente manera:

Si el usuario es humano, cualquier acción que este haga será procesada por el modulo **truco.js**. Este se encargará de contextualizar la acción (por ejemplo, determinar en que mano se realizó el canto y/o si hubo un canto anterior a ese) y llamará al correspondiente método del módulo **ia.js** que usando esta información como conocimiento de dominio, determinará con qué acción responder.

3 Conocimiento de Dominio

3.1 Sobre el engaño en el truco

El truco es un tipo de juego con información parcial en el sentido en que al menos uno de los jugadores conoce solo un estado parcial del juego (en caso de un desarrollo "normal" del juego, esto debería pasar para todos los jugadores involucrados). Esto implica una complejidad mayor a la hora de tomar decisiones ya que es necesario tener en cuenta las jugadas pasadas como las posibles jugadas. El hecho de que los jugadores no conozcan exactamente el estado del resto de los jugadores puede ser aprovechado de manera beneficiosa para intentar inducir estados falsos que den lugar a malas decisiones de juego por parte de otro de los jugadores. En el truco este engaño puede manifestarse de dos formas (que osamos en llamar activa y pasiva)

Activa: el jugador toma la iniciativa mediante algún canto (envido o truco) con el fin de hacerle creer al oponente que tiene cartas o puntos que en realidad no tiene.

Pasiva: el jugador plantea un escenario (generalmente mediante el orden en el que juega las cartas) de manera tal de que la hipotetización hecha por el oponente sobre el estado sea errónea llevándolo a tomar acciones 'activas' poco favorables para este (basicamente que el jugador que esta intentando ser engañado piense que el otro jugador tiene mas o menos puntos o cartas de mayor o menor valor que en las que realidad tiene).

Cabe destacar que un intento de engaño tanto activo como pasivo esta influenciado tambien por la información que uno tiene del otro jugador, que en la mayoría de los casos sigue siendo parcial, por lo cual hay una especie de ciclo de mentiras.

De aquí, que el engaño juega un rol importante en el truco.

3.2 Funciones principales

Como se explica en la sección anterior, el engaño es parte del juego y este está generalmente influenciado por la información disponible, lo cual también implica que no hay una estrategia ganadora. A continuación haremos una descripción de las funciones principales de la IA comentando cómo fue tomada en cuenta la información disponible y de qué manera se intentó darle "picardía" a la máquina.

Luego de analizar la metodología de juego, se llegó a una división en 3 partes diferentes:

- Truco: se encarga de decidir los cantos del truco
- Envido: decide el canto del envido
- Jugar Carta: decide qué carta jugar

3.2.1 Truco

: Esta es la información que consideramos pertinente a la hora de decidir el canto del truco:

- Información sobre si el humano cantó algo y la máquina debe contestar o si esta puede cantar (variable **resp**)
- El último canto hasta el momento en la ronda actual (variable **ultimo**)
- El número de la mano actual (primera, segunda o tercera)
- Resultado de las manos anteriores
- Cartas jugadas en la mano actual (propias o del oponente)
- Clasificación de las cartas en mano (cantidad de cartas altas, medias o bajas que tengo).
- Posibles cartas del oponente según los puntos cantados en el envido y las cartas que jugó.

Mediante todos estos atributos se conforman las reglas de inferencia que se dividen (a grandes rasgos) de la siguiente forma:

- 1) Contestar un canto o Cantar
- 2) Se determina cuál es la mano actual

- 3) Se determina el resultado de la mano anterior (para las manos 2 y 3)
- 4) Se clasifican las cartas que poseo
- 5) En funcin de la situacin en la que se encuentre y teniendo en cuenta los datos correspondientes al dominio se decide que cantar (Si, No, Re-Truco, Vale 4)

De manera más ilustrativa veamos un extracto del código de la función truco

```

1
2 if (resp) { // Me cantaron, tengo que responder
3   switch(nroMano){
4     case 0:
5       var ran = getRandomInt(0,100);
6       switch(ultimo){
7         case 'T':
8           if (e2.jugador.puntosGanadosEnvido < 2 &&
9             (mediaalta) >= 2 && clasif.alta >= 1)
10            return 'S';
11           if (clasif.baja == 3 && ran <= 50) return 'RT';
12           if (mediaalta >= 2 && diff < 0 ) return 'RT';
13           // Te trato de correr
14           if (mediaalta >= 1 && diff > 0 ) return 'S';
15           return ( ran < 66 ? 'N' : 'S' ); //
16           (*) Rompe con la reglas staticas
17           case 'RT':
18           case 'V':
19             if (clasif.alta >= 2) return 'S';
20             if ((mediaalta) >= 2 && diff > 3 ) return 'S';
21             if (e2.jugador.puntosGanadosEnvido >= 2) return 'N';
22             return 'N';
23             break;
24         }
25       }
26     }
27   }
28 }
```

Listing 1: Extracto de la función truco

En este caso vemos qué sucede cuando el oponente canta algo (la variable **resp** toma el valor true). Primero determinamos en que número de mano

nos encontramos. Luego en función de lo que fue cantado por el oponente (truco - T, re-truco - RT, o vale 4 - V) y en función de las cartas en mano así como de los puntos obtenidos en el envío se decide que acción tomar. Se puede ver también una componente aleatorio. Si bien no es un recurso ideal, nos encontramos con el hecho de que una vez colectada toda la información posible (que es incompleta) siempre hay un factor no determinista (siempre hay más de un camino a seguir) que podría ser ignorado poniendo reglas fijas pero esto haría que el comportamiento de la máquina sea bastante "rígido".

3.2.2 Envío

El conocimiento de dominio que utiliza comprende:

- ultimo: Ultimo canto
- acumulado: La cantidad de puntos en juego
- ultimaCarta: Carta jugada por el oponente(si es que existe). Deducion de puntos
- puntos: Puntos de Envío en mano
- Puntos restantes para terminar el partido
- Puntos con lo que el oponente canto en otras manos

Para decidir el canto del envío, primero se analiza los puntos restantes para terminar la partida debido a que la Falta Envío es diferente al resto de los cantos. Si faltan dos o menos puntos para ganar, se responde a un canto con la Falta Envío o si se canta se elige esta. En caso de estar a mediados del juego se analiza si hubo cantos previos, si el oponente ya jugó una carta y los puntos que tengo en manos. Todos estos valores generan una decisión de juego, que por último es avalada por un poco de no-determinismo, para generar la sensación de cambios en la mentalidad de juego.

3.2.3 Jugar Carta

El conocimiento de dominio que utiliza comprende:

- Juego primero o el oponente ya jugo
- Clasificación de las cartas
- Numero de Mano

- Carta jugada por el oponente

Para determinar que carta jugar primero se tiene en cuenta si el oponente jugo. Si es así elige la carta en función de la que esta en la mesa. Depende de si le puede ganar o no. Si la máquina es mano, por lo general opta por la siguiente fórmula: Ganar primera, dejar pasar segunda y ganar tercera. Por supuesto hay reglas , que capturan situaciones especiales, provocando una variación al juego de la máquina. Por ejemplo, si es mano y posee dos cartas altas.

El conocimiento de dominio necesario fue aportado por los integrantes del grupo

References

- [1] *Descripción y Reglas del Truco* . http://es.wikipedia.org/wiki/Truco_argentino