Negociación en sistemas de agentes

Sistemas Multi-Agente y Sistemas Autónomos

Juan A. Botía

Departamento de Ingeniería de la Información y las Comunicaciones, Universidad de Murcia

November 22, 2007

Negociación, ¿por qué?, ¿para qué?

- Favorece la coordinación y la cooperación entre agentes
- Necesaria tanto cuando los agentes son egoistas como cooperativos.
- Podemos encontrar una definición de negociación entre agentes como

  > *Negociación entre agentes es el proceso mediante el cual un grupo de agentes llegan a un acuerdo mutuamente aceptable, sobre algún asunto.(Jennings, 2001 [1])*

- Los agentes son autónomos → si necesitan algo de los demás puede que sea necesario negociar
  - realizar propuestas,
  - intercambiar opciones,
  - ofrecer concesiones y, si es posible,
  - llegar a un acuerdo mutuo aceptable.

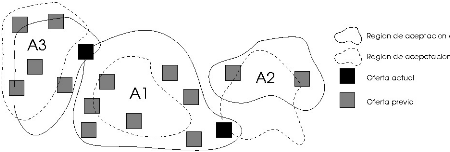
Elementos a considerar en la negociación

Para definir un marco negociador, se deben definir formalmente los siguientes elementos

- Protocolos de negociación
  - roles de agentes,
  - estados de la negociación,
  - eventos de transiciones y
  - mensajes intercambiados
- Objetos sobre los que negociar (i.e. precio de artículo, QoS, términos contractuales, etc)
- Modelos usados para la toma de decisiones

Negociación como un proceso de búsqueda

- el agente A_1 negocia con A_2 y A_3
- A_1 puede llegar a acuerdo con A_3 al interseccionar sus regiones de aceptación actuales.
- A_1 no puede llegar a un acuerdo con A_2 al no existir intersección



Negociación y búsqueda (II)

- Cada uno de los puntos en regiones de aceptación debe tener asignada una puntuación
- Las regiones de aceptación pueden estar sujetas a cambios (desplazamiento, contracción o expansión)
- Proceso negociador más sencillo: Dutch auction
  - el subastador comunica el precio de subasta del artículo
  - si no recibe ninguna respuesta, intentará reducir el precio
  - termina con una respuesta
  - la única información que llega al subastador para la elaboración de una nueva propuesta es la ausencia de información!!!

Realimentación en la negociación

- El proceso negociador se agiliza si se incluye realimentación (crítica o contraoferta)
- Crítica, ejemplo A

  A: Te propongo que me proveas del servicio X bajo ciertas condiciones
  B: El precio de X me parece bueno pero el tiempo de suministro es muy pequeño

  y ejemplo B

  A: Te propongo proporcionarte el servicio Y si tu me proporcionas el servicio X
  B: No me interesa el servicio Y

- Contraoferta, ejemplo A

  A: Te propongo de que me proveas del servicio X
  B: Te propongo proveerte del servicio X si tu me provees del servicio Z

  y ejemplo B

  A: Te propongo proporcionarte el servicio Y si tu me proporcionas el servicio X
  B: Te propongo proporcionarte del servicio X si tu me proporcionas el servicio Z

Negociación desde la óptica de la teoría de juegos

- Cuando los agentes negocian hacen uso de dos conceptos básicos
  - Preferencia: orden sobre los estados del mundo
  - Utilidad: función que define la preferencia
- Sea un conjunto de estados $\Omega = \{\omega_1, \ldots, \omega_n\}$ en los que cada agente (i y j) manifiestan una preferencia que definimos así

$$u_k : \Omega \rightarrow \Re,\ k \in \{i, j\}$$

- u_k induce una preferencia sobre las salidas, tal que si $u_i(\omega) \geq u_i(\omega')$, entonces el agente i tiene una preferencia sobre ω al menos tan alta como la que tiene sobre ω'
- Lo denotamos con

$$\omega \succeq_i \omega'.$$

- Y cuando la preferencia es estricta, $u_i(\omega) > u_i(\omega')$, usaremos

$$\omega \succ_i \omega'$$

Escenarios para la negociación

- En un proceso de negociación, agentes (i, j) realizan simultáneamente una acción cada vez sobre el entorno, generándose un nuevo estado en la negociación $\in \Omega$
- La utilidad de cada agente depende de la combinación de las acciones que ambos han realizado
- Podemos modelar el entorno como una función

$$\tau : Ac \times Ac \rightarrow \Omega,$$

  siendo Ac es el conjunto de acciones posibles
- Ejemplo, si los agentes puede, bien cooperar, C, bien no cooperar, D,
  - $\tau(D, D) = \omega_1$, $\tau(D, C) = \omega_1$, $\tau(C, D) = \omega_1$, $\tau(C, C) = \omega_1$
  - $\tau(D, D) = \omega_1$, $\tau(D, C) = \omega_2$, $\tau(C, D) = \omega_1$, $\tau(C, C) = \omega_2$

Un ejemplo más a fondo

- Sea la definición del entorno que vamos a utilizar en el ejemplo

$$\tau(D, D) = \omega_1,\ \tau(D, C) = \omega_2,\ \tau(C, D) = \omega_3,\ \tau(C, C) = \omega_4$$

- Y la función de utilidad o preferencias sobre los estados

$$\begin{array}{llll} u_i(\omega_1) = 1 & u_i(\omega_2) = 1 & u_i(\omega_3) = 4 & u_i(\omega_4) = 4 \\ u_j(\omega_1) = 1 & u_j(\omega_2) = 4 & u_j(\omega_3) = 1 & u_j(\omega_4) = 4, \end{array}$$

- Así, para el agente i

$$C, C \succeq_i C, D \succ_i D, C \succeq_i D, D.$$

- ¿Si fueramos el agente i el este escenario, cooperaríamos o no?
  - i cooperará haga lo que haga el agente j
  - j hará lo mismo

Un ejemplo más a fondo (y II)

- Supogamos ahora que u está definida como sigue

$$\begin{array}{llll} u_i(\omega_1) = 4 & u_i(\omega_2) = 4 & u_i(\omega_3) = 1 & u_i(\omega_4) = 1 \\ u_j(\omega_1) = 4 & u_j(\omega_2) = 1 & u_j(\omega_3) = 4 & u_j(\omega_4) = 1, \end{array}$$

- i tiene el siguiente orden para sus preferencias

$$D, D \succeq_i D, C \succ_i C, D \succeq_i C, C$$

- Ambos agentes no pueden hacer otra cosa mejor que no cooperar !!!
- Las cosas no son tan fáciles en el mundo real (e.g. dilema del prisionero)

Dominancia en conjuntos de estados

Para responder a *qué acción tomar* en entornos algo más complejos se usa el concepto de dominancia

- Sean Ω_1 y Ω_2 una partición disjunta de Ω
- Ω_1 domina a Ω_2 para el agente *i* si i prefiere cada resultado de Ω_1 sobre cada resultado de Ω_2 (i.e. la utilidad es siempre mayor en los estados de Ω_1)
- Ejemplo: si $\Omega_1 = \{\omega_1, \omega_2\}$ y $\Omega_2 = \{\omega_3, \omega_4\}$

$$\omega_1 \succ_i \omega_2 \succ_i \omega_3 \succ_i \omega_4,$$

  entonces Ω_1 domina fuertemente a Ω_2

Formalmente, Ω_1 domina fuertemente a Ω_2 si se cumple que $\forall \omega_1 \in \Omega_1$, $\forall \omega_2 \in \Omega_2$, tenemos que $\omega_1 \succ_i \omega_2$.

Estrategias

- Una estrategia s, es el mecanismo que usan los agentes para decidir qué acciones de Ac realizar
- s_1 domina a s_2 si el conjunto de salidas posibles al emplear s_1 dominan al conjunto de salidas posibles al emplear s_2
- Si queremos encontrar una buena estrategia, podemos
  1. Si la estrategia s domina a s', no será racional para un agente el seguir s' (s' se rechazará)
  2. Si, iteratívamente, se van eliminando estrategias dominadas y al final solamente queda una estrategia, problema resuelto
  3. Si no , se eliminan estrategias débilmente dominadas
     - ★ s_1 domina débilmente a s_2 si toda salida obtenida mediante el empleo de s_1 es preferida al menos de igual forma como la más alta obtenida usando s_2

Equilibrio Nash

Equilibrio

Un sistema de agentes está en equilibrio cuando todos los agentes están ejecutando la estrategia adecuada, de tal forma que si uno de ellos cambia de estrategia, independientemente de lo que hagan los demás, verá reducido su beneficio (utilidad)

Equilibrio Nash

dos estrategias s_1 y s_2 están en equilibrio Nash si se cumple que

1. dado que el agente i emplea s_1, el agente j no puede hacer nada mejor que emplear s_2 y
2. dado que el agente j emplea s_2, el agente i no puede hacer nada mejor que emplear s_1.

Equilibrio Nash, Observaciones

El equilibrio Nash es una situación deseable (no necesariamente la mejor)

- Si existe un equilibrio de este tipo entre un par de agentes, los dos estarán enganchados a sus estrategias respectivas y ambos se beneficiarán de ello
- Sus respectivas estrategias no tienen por que ser las óptimas para ambos por separado (i.e. solo lo son si el otro está jugando la correspondiente al equilibrio)

Problemas

1. No todo escenario de negociación tiene un equilibrio de este tipo
2. Algunos escenarios de interacción tienen más de un equilibrio de este tipo

El dilema del prisionero

Definición del problema

dos hombres están acusados del mismo crimen, y mantenidos encerrados en la cárcel en celdas separadas e incomunicadas, proporcionándosele a ambos la siguiente información:

- Si uno de los dos confiesa el crimen y el otro no, el confesor será dejado libre y el otro será encarcelado durante tres años;
- si los dos confiesan el crimen, cada uno será encarcelado por dos años;
- si ninguno confiesa, los dos serán encarcelados por un año.

¿Qué debe hacer cada uno de los prisioneros?

El dilema del prisionero (y II)

Si asociamos no confesión con no cooperación y confesión con cooperación, podemos representar

	i no coopera	i coopera
j no coopera	2 / 2	0 / 5
j coopera	5 / 0	3 / 3

De lo que se derivan

$$u_i(D,D) = 2, u_i(D,C) = 5, u_i(C,D) = 0, u_i(C,C) = 3,$$

$$u_j(D,D) = 2, u_j(D,C) = 0, u_j(C,D) = 5, u_i(C,C) = 3,$$

$$D,C \succ_i C,C \succ_i D,D \succ_i C,D,$$

$$C,D \succ_j C,C \succ_j D,D \succ_j D,C.$$

El dilema del prisionero (y III)

- No existe una estrategia dominante

  El cooperar y no cooperar deberá depender de lo que haga el otro

- Pongámonos en el lugar de i
  - Supongamos que coopero: entonces si j coopera ambos obtendremos una recompensa de 3 mientras que si j no coopera obtendré una recompensa de 0. La recompensa que se me garantiza si coopero es 0.
  - Supongamos que no coopero: entonces si j coopera, obtendré una recompensa de 5 mientras que si no lo hace, obtendré una de 2. La mejor recompensa garantizada es de 2.
  - Desde luego, preferiré una recompensa garantizada de 2 a una de 0, por lo tanto debería no cooperar.

El dilema y el equilibrio Nash

- El escenario es simétrico (los agentes son intercambiables y los resultados no variarían si a los dos se les supone racionalidad)
- Es razonable pensar que ambos no cooperarán
- Solamente hay una estrategia Nash, (D, D)
  - Suponiendo que i no va a cooperar, el agente j no puede hacer nada mejor que no cooperar
  - Suponiendo que j no va a cooperar, el agente i no puede hacer nada mejor que no cooperar

Negociación basada en Teoría de Juegos

- Los agentes egoistas pueden verse como oponentes en un juego
- los resultados obtenidos en teoría de juegos son aplicables a la negociación entre agentes de este tipo
- Se trata de diseñar sistemas que respondan a ¿qué acción es la mejor que un agente puede ejecutar actuando racionalmente?
- Aplicarse a los dos problemas de diseño derivados de la construcción de un **entorno para negociación**
  - Los protocolos definen las reglas del juego: conjunto de normas que constriñen las propuestas que pueden hacer los agentes participantes.
  - La estrategia es el modelo de decisión usado para que el agente maximice su función de utilidad.

Propiedades deseables de un entorno para la negociación [4]

- Bienestar social (i.e. *social welfare*)
  - Def: la suma de los beneficios obtenidos por los agentes en sus acciones en el entorno
  - Aplicación: comparar diferentes soluciones obtenidas con diferentes protocolos.
- Eficiencia pareto
  - Usado también como indicador global en un SMA
  - Un protocolo va a ser *Pareto eficiente* si no existe ningún otro protocolo tal que este haga a un agente más feliz ejecutándolo, sin hacer a ninguno de los otros agentes más infeliz.
- Racionalidad individual
  - Imprescindible que se de en un protocolo de negociación.
  - Se da cuando el negociar bajo las reglas del protocolo implica obtener un bienestar mejor que al no participar en la negociación.
- Estabilidad
  - Los protocolos deben ser estables
    - ★ deben motivar a los agentes a comportarse bajo sus reglas (i.e. no manipular al contrario) ya que si un agente obtiene más bienestar comportándose de otra forma, lo hará
  - El diseño ideal implica la existencia de una estrategia dominante (se obtiene mayor bienestar con ella, independientemente del resto de estrategias que use el resto)
- Simplicidad, uso de ancho de banda limitado y robustez

Críticas a la teoría de juegos

- ¿Cómo asociar las preferencias de los agentes con las posibles ofertas recibidas?
  - TJ asume que cada situación es perfectamente comparable con otras (i.e. piénsese en dos posiciones en un tablero de ajedrez)
  - ¿Qué pasa cuando hay mútliples criterios (i.e. niveles semánticos para las preferencias)
- No existe un modelo genérico que gobierne el proceso racional de elección de alternativas
  - Modelos específicos muy sensibles a cambios
- la teoría de juegos asume que cada agente conoce la totalidad del espacio de búsqueda en el que se mueven las posibles soluciones a la negociación (problemas computacionalmente intratables)

El regateo, un escenario concreto de negociación
Modelo formal

- Dos agentes con acceso compartido a todas las alternativas de un conjunto denominado **de aceptabilidad** (i.e. *feasible set*)
- Sus funciones de utilidad están definidas sobre ese conjunto
- Si los dos están de acuerdo en una alternativa la escogen (*agreement point*) y el proceso finaliza
- Si no, el proceso finaliza en otra alternativa denominada el punto de desacuerdo (i.e. *disagreement point*)

El regateo y la aproximación de Nash

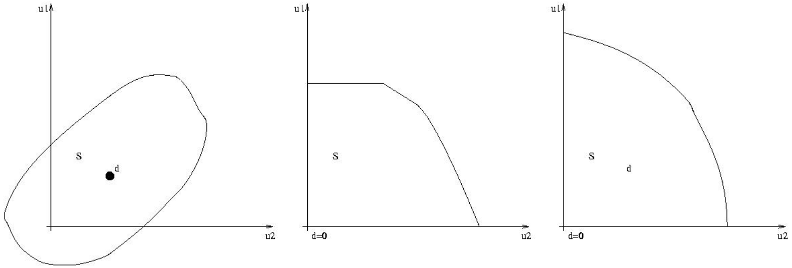
- La teoría axiomática del regateo (i.e. *bargaining*) se originó a raiz de un artículo de J.F. Nash de 1950 denominado "The Bargaining Problem"
  - a partir de una representación idealizada del problema, se garantiza la existencia de solución
- Tipo de problema: Dada la patronal y los trabajadores de una empresa, ha llegado el tiempo de la renegociación de los contratos en la que se debe decidir cuáles van a ser los beneficios de la empresa. Si no se llega a un acuerdo, se irá a la huelga

Modelo Nash para el regateo

- Formalmente, un problema de bargaining de n agentes es un par (S, d) en donde S es un subconjunto del espacio euclídeo n-dimensional y d es un punto de S
- Σ_d^n es la clase de los problemas que cumplen
  1. S es convexo, limitado y cerrado (i.e. contiene a los puntos del límite)
  2. Existe al menos un punto en S que domina estrictamente a d.

Representación de los tipos de problemas

1. Un problema en Σ_d^2
2. Un problema en Σ_0^2
3. Un problema estrictamente comprendido en Σ_0^2



- Los problemas más comunes son aquellos con $d = 0$

Modelo Nash para el regateo

- La convexidad se exige debido a la posibilidad de negociación aleatoria.
- El espacio S será limitado si las funciones de utilidad lo son
- La cerradura se asume por conveniencia en los cálculos matemáticos
- La existencia de un punto $x \in S$ tal que $x > d$ se exige para evitar el caso degenerado en el que uno solo de los agentes gana en el acuerdo

Soluciones al problema del regateo

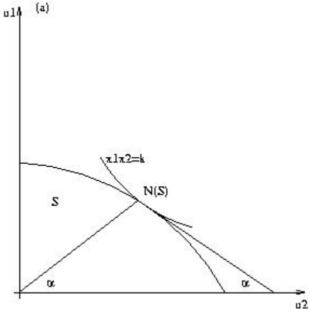
Definición informal de solución al problema del bargaining

Una **solución** en un dominio de problemas determinado, asocia a cada elemento de (S, d) un único punto de S interpretado como una predicción (o recomendación) para ese problema.

- En la teoría del bargaining hay varias soluciones, dependiendo de los axiomas que se exige que cumplan
- Se formulan para un $S \in \Sigma_0^n$ arbitrario.
- Soluciones más conocidas
  1. La solución Nash
  2. La solución Kalai-Smorodinsky
  3. La solución igualitaria

Soluciones al problema del regateo: Nash

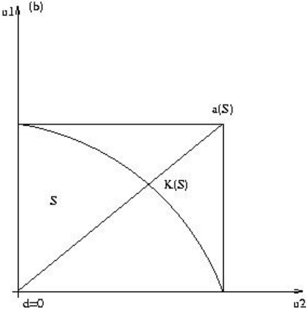
- La solución Nash, N: $N(S)$ es el máximo del producto $\prod x_i$ sobre S o análogamente $N(S, d)$ es el máximo de $\prod(x_i - d_i)$ para $x \in S$ donde $x \geq d$.



La solución (compromiso) se obtiene maximizando el producto de las ganacias que se pueden obtener partiendo de d (*disagreement point*)

Soluciones al problema del regateo: Kalai-Smorodinsky

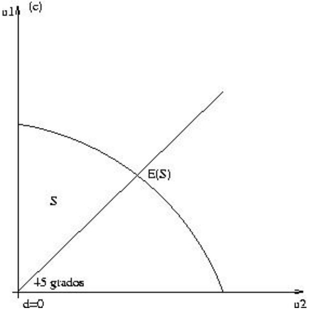
- La solución Kalai-Smorodinsky, K: $K(S)$ es el punto máximo de S en el segmento que conecta el origen a $a(S)$, el punto ideal de S definido como $a_i(S) \equiv max\{x_i | x \in S\}$ para todo i.



  - Solución proporcional a las expectativas más optimistas de los agentes.

Soluciones al problema del regateo: Igualitaria

- La solución igualitaria, E: $E(S)$ es el punto máximo de S de iguales coordenadas. Dicho de otra forma, $E_i(S,d) - d_i = E_j(S,d) - d_j$ para todo i, j.

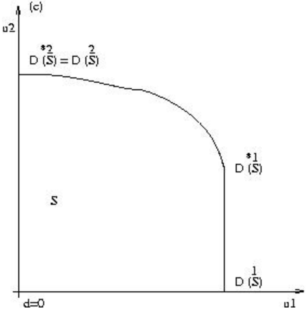


  - Solución inevitable ya que lo ideal en muchos sistemas económicos es favorecer la igualdad

Soluciones al problema del regateo: Dictatorial

Esta solución es un ejemplo extremo del favorecimiento a un agente con respecto al resto

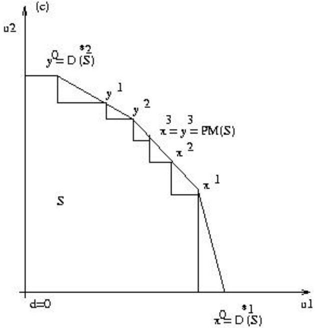
- Solución dictatorial, D^i y D^{*i}: $D^i(S)$ es el punto máximo x de S con $x_j = 0$ para todo $j \neq i$. Si $n = 2$, $D^{*i}(S)$ es el punto del conjunto $\{x \in S | \nexists x' \in S \text{ con } x' \geq x\}$.



Soluciones al problema del regateo: Perles-Maschler

Ejemplo típico de concesiones balanceadas, los agentes realizan un recorrido por sus alternativas preferidas (soluciones dictatoriales respectivas) hasta alcanzar una posición final de compromiso en compromiso

- Si el límite de S es poligonal, PM(S) es punto límite común de las secuencias $\{x^t\}, \{y^t\}$ definido mediante $x^0 = D * 1(S)$, $y^0 = D * 2(S)$ y para cada $t \in \mathcal{N}$, x^t, $y^t \in PO(S)$ son tales que $xt \geq y^t$, los segementos $[x^{t-1}, x^t]$ y $[y^{t-1}, y^t]$ están contenidos en $PO(S)$ y $|(x_1^{t-1} - x_1^t)(x_2^{t-1} - x_2^t)|$ y $|(y_1^{t-1} - y_1^t)(y_2^{t-1} - y_2^t)|$ son iguales y máximos.



Coordinación mediante leyes sociales

- Se debe a Shoham y Tennenholtz [3]
- La idea principal es que la coordinación en un sistema de entidades se genera a partir de la imposición de *leyes sociales*
- Argumentación: dadas las dos soluciones extremas que existen para habilitar coordinación
  1. que un único programador se encargue de la construcción de todos los robots y piense en los posibles conflictos que puedan surgir entre ellos (impracticable)
  2. que cada robot se programe de manera independiente y, admitiendo que pueden surgir conflictos entre ellos, dotarlos de los mecanismos necesarios para resolverlos como el acudir a un árbitro o mediante negociación (poco restringido)

  se proponen leyes sociales a cumplir por
    - programador y
    - artefactos participantes del sistema

Escenario de referencia

- Considérese una malla de $n \times n$ con m robots móviles.
- cada robot está localizado en una coordenada de la malla
- Conflictos: varios robots ocupan la misma coordenada, a evitar
- Dinámica
  - Un robot tienen la facultad de moverse a cada una de las coordenadas vecinas o permanecer inmóvil en cada instante de tiempo
  - Cada una de estas operaciones conlleva un gasto de tiempo y energía apreciables
  - el sistema es síncrono, i.e. el movimiento a una celda vecina conlleva una unidad de tiempo y una unidad de energía
- Tareas a desempeñar por los robots
  - Objetivo: alcanzar una localización en la malla (minimizando t, *power* o ambos)
  - Los objetivos se acumulan

Una primera aproximación a la coordinación

Aproximación 1

- Supuestos
  1. los m robots no pueden detectarse entre sí
  2. malla $n \times n$ donde n es par, filas
  3. $m \geq n$, m es un número alto
- Pregunta: ¿es posible el diseño de un sistema de tráfico mediante el cual no se den colisiones haciendo uso únicamente de normas sociales?

Una primera aproximación a la coordinación
Aproximación 1

Aplicamos la siguiente Ley

> **Ley de tráfico I**: *cada robot ha de moverse con una velocidad constante. La dirección del movimiento ha de fijarse así: en las filas pares cada robot debe moverse a la izquierda, mientras que en las filas impares ha de hacerlo hacia la derecha. Cuando se llega a la columna más a la derecha, se ha de mover hacia arriba. Si la columna es la que está más a la izquierda estando en filas pares, o la segunda más a la izquierda en filas impares, se debe mover hacia abajo.*

Una primera aproximación a la coordinación

Aproximación 1

Análisis

- movimiento de los agentes es como el de una serpiente y define lo que se denomina un ciclo Hamiltoniano
- De esa ley se extrae el siguiente resultado:

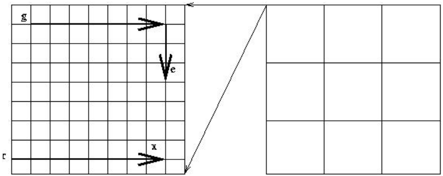
  Resultado I: *La ley de tráfico I garantiza que no se van a producir colisiones y que cada agente va a alcanzar en un tiempo $O(n^2)$ desde el instante que recibe el objetivo.*

- La ley de tráfico I es un claro ejemplo de cómo una ley dicta totalmente el comportamiento de los agentes anulando cualquier rastro de autonomía
- la manera en la que los agentes desempeñan sus tareas mediante esta ley no es demasiado eficiente (no perciben)

Segunda aproximación a la coordinación

Asumimos

- los robots tienen determinadas capacidades de percepción
- Vamos a superponer a la malla otra con menos densidad de cuadrículas (i.e. filas y columnas más anchas)
- cuando los agentes se mueven en la malla más abstracta, las leyes de tráfico simples serán suficientes
- el tiempo de espera para entrar en una malla interna estará acotado convenientemente
- En la malla interna, se asegura el movimiento sin bloqueos



Segundar aproximación a la coordinación (y II)

Ley de Tráfico II - Primera parte

- Superponer a la malla otra con menos filas y columnas. Estará dividida en submallas de tamaño $2m \times 2m$
- En filas impares de la superpuesta, los robots solamente pueden moverse a la derecha y hacia abajo, en las pares hacia la izquierda y hacia arriba.
- En los cruces de la malla superpuesta, cuando un robot está a una distancia de 1 de entrar en uno de los cruces de la malla superpuesta, no debería entrar en el cruce, debería esperar siempre que hubiera otro robot a la misma distancia y estuviera allí antes.
- A un robot en la malla superpuesta le está permitido cambiar su dirección solamente k veces antes de dejar la malla y entrar en una malla interna.

Segunda aproximación a la coordinación (y III)

Ley de Tráfico II - Segunda parte

- Dentro de cada malla interna, cada robot se moverá de la siguiente forma:
  - El punto de entrada a la malla ha de ser por la coordenada más a la izquierda de la fila marcada con r en la figura
  - Una vez dentro, el robot ha de moverse hasta la coordenada denotada con x en la misma figura
  - En una fila si se percibe la presencia de otro robot en la siguiente coordenada a la derecha o bien llega hasta x, debe parar y esperar $2n + cmk$ unidades de tiempo
  - Después ha de continuar, si es necesario, hasta x a lo largo de r.
  - Una vez ahí, para salir debe hacerlo desde la coordenada g, y siguiendo el camino marcado hasta e en menos de $4m - 2$ instantes de tiempo y sin pisar las áreas con línea gruesa.
- Los robots no pueden parar de moverse al menos que exista un robot enfrente y no se mueva o se requiera por la ley social.
- Los puntos de retorno desde la malla fina a la gruesa se considerarán cruces adicionales de esta segunda malla.

Segunda aproximación a la coordinación (y IV)

Ley de Tráfico II - Resultado Con esta segunda ley tenemos también el resultado

> **Resultado II**: *La ley de tráfico I garantiza que no se van a producir colisiones.*

- ahora los robots tienen más autonomía (malla fina)
- La planificación del movimiento es independiente de las leyes sociales.

Conclusiones iniciales a la coordinación mediante leyes sociales

- la imposición de leyes que todo agente debe cumplir (sociales), manifestando autonomía cuando se permite, induce un sistema sin conflictos
- La cuestión de la eficiencia depende de la habilidad a la hora de plantear las leyes
  - Leyes más restrictivas generarán sistemas más seguros y menos eficientes

Modelo formal para la coordinación mediante leyes sociales

Ideas principales

- Un agente es una entidad abstracta con estado, que realiza acciones que modifican ese estado.
- Dado que el alcanzar un estado depende eventualmente de las acciones de varios agentes el modelo general se basa en estados formados por estados particulares de cada agente

Ley Social

Dado un conjunto de estados S, un lenguaje de primer orden $\mathcal{L}$, y un conjunto de acciones A, una restricción es un par (a, φ) en donde $a \in A$ y $\varphi \in \mathcal{L}$ es una sentencia. Entonces, definimos una ley social como un conjunto de restricciones (a_i, φ_i), conteniendo al menos una restricción para cada $a_i \in A$.

Modelo formal (cont.)

- Dados un par $s \in S$ y $\varphi \in \mathcal{L}$, cuando s satisface φ escribimos $s \models \varphi$
- (a_i, φ_i) significa que φ_i es la condición más general sobre estados de S que prohibe ejecutar a_i
- dado un par de leyes sociales sl_1 y sl_2, usamos $sl_1 < sl_2$ para referirnos al hecho de que para cada $(a_i, \varphi_i) \in sl_2$ existe $(a_i, \varphi_j) \in sl_1$ tal que $\varphi_j \models \varphi_i$ (sl_1 más restrictiva que sl_2)
-

Modelo formal (cont.)

Agente Social

Un agente social es una quíntupla $(S, \mathcal{L}, A, SL, T)$ en donde S, $\mathcal{L}$ y A el conjunto de estados, el lenguaje de primer orden y el conjunto de acciones. SL es un conjunto de leyes sociales, y $T : S \times A \times SL \to 2^S$ es la función de transición definida como sigue:

- para cada $s \in S$, $a \in A$, $sl \in SL$, si $s \models \varphi$ se cumple y $(a, \varphi) \in sl$, entonces $T(s, a, sl) = \emptyset$.
- para cada $s \in S$, $a \in A$, $sl_1 \in SL$, $sl_2 \in SL$, si $sl_2 < sl_1$ entonces $T(s, a, sl_1) \subseteq T(s, a, sl_2)$.

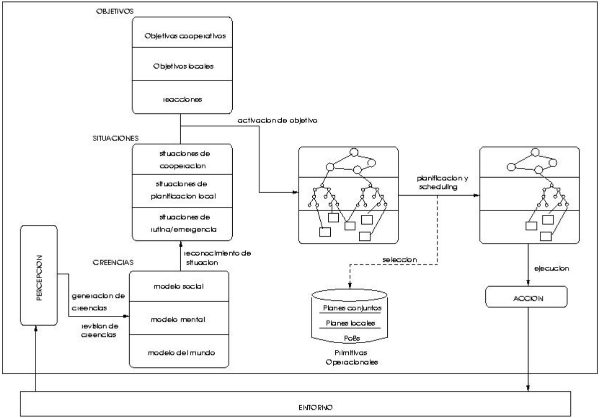
Sistema multiagente

Un Sistema multiagente social es una colección de agentes sociales que comparten el conjunto de estados, el lenguaje que describe los estados, el conjunto de acciones, el conjunto de leyes sociales y la función de transición.

Consideraciones finales

- El problema es encontrar leyes sociales (NP-completo) válidas, aunque se puede
  - restringir el número de transiciones entre estados
  - dividir el problema en conjuntos de estados modulares agrupados por funcioalidad
- No existe una metodología
- Poca aplicabilidad
- Continuación en el trabajo se Sasscha Ossowski, con la coordinación mediante cooperación estructural

La arquitectura InteRRaP
(INTEgration of Reactive behaviour and Rational Planning)



Negociación en los agentes InteRRaP

Los agentes de esta arquitectura se ciñen al siguiente esquema procedimental para realizar un proceso negociador:

1. Intercambiar información de estatus y/o objetivos entre los agentes
2. Determinar el tópico de la negociación. Si no se requiere, o no es posible la negociación, entonces ir al paso 7. Si no, continuar.
3. Determinar el protocolo de negociación y asignar roles a los agentes
4. Elegir un lider que calcule el conjunto sobre el que negociar
5. Enviar el conjunto de negociación al resto de los agentes
6. Negociar; si se obtiene una solucion salir; si no continuar;
7. Abortar negociación

Intercambio de negociación y tópico a negociar

- Los dos primeros pasos del proceso negociador determinan el problema a resolver
- En InteRRaP se asume que los agentes comunican sus objetivos de manera confiable y por un canal de transporte seguro
- el protocolo de negociación a usar es parte del conocimiento común

Asignación de roles

- Dados un tópico sobre el que negociar y un protocolo de negociación
  - ¿cómo asignamos los roles?
- Se basa en la elección de un lider que los asigna
  - todos los agentes tienen un identificador numérico que los distingue
  - El lider es el que tiene el ID más alto

```
M = I;
enviar I a todos los vecinos
al recibir mensaje con identificador J
  si M < J entonces
    M = J;
    enviar J al resto de vecinos;
  end
al haber recibido un mensaje de cada vecino
  si M = I el agente es lider, si no es seguido
```

El conjunto de negociación

¿Cómo realizar el cálculo del conjunto de negociación?

- En una negociación bilateral comprador-vendedor
  - el conjunto de negociación es un intervalo continuo $[o_v, o_c]$ en donde o_v es la primera oferta emitida por el vendedor y o_c la primera oferta emitida por el comprador
- En una negociación basada en el protocolo de red de contratos para planificación cooperativa
  - El número de planes posibles a realizar puede ser exponencial con relación al número de agentes!!!
  - se recurre a la abstracción de planes (librerías) que son planes no especificados del todo y que se corresponden con un conjunto de situaciones típicas

Negociar

Sea D un dominio sobre el que negociar; un negociación se define mediante la tupla

$$NEG = (A, R, \rho, N, U, P, S)$$

en donde

- $A = \{a_1, \ldots, a_k\}$, $k \geq 2$, los agentes a_i con estados mentales B_i
- $R = \{r_1, \ldots, r_l\}$, $l \leq k$ es un conjunto de roles;
- ρ es una función que asigna roles a los agentes, $rho(a) = r$ con $a \in A$, $r \in R$
- $N \neq \emptyset$, $N \subseteq D$ es el conjunto de negociación.
- $U = \{u_1, \ldots, u_k\}$, donde $u_i : N \rightarrow \Re$ es la función de utilidad para el agente a_i

Negociar, sigue

- $P = (K, \{\pi : R \times K \rightarrow 2^K | r \in R\})$ es un protocolo de negociación.
  - K es un conjunto finito de performativas
  - π mapea primitivas de comunicación en un conjunto de reacciones admisibles con respecto a un rol específico r
- $S = \{\sigma_i : P \times R \times K \times 2^D \times U \rightarrow K \times N | 1 \leq i \leq k\}$ son estrategias de negociación, una por agente
  - Como entradas protocolo, rol, primitiva recibida, conjunto de negociación, función de utilidad y estado interno
  - Como salida la réplica a la primitiva, el conjunto de negociación y el estado modificado del agente
- Las primitivas usadas en InteRRaP son `INFORM`, `ACCEPT`, `ANNOUNCE`, `PROPOSE`, `REJECT`, `REQUEST`, `GRANT`, `MODIFY` y `CONFIRM`.

Un ejemplo definido formalmente

Negociación de planes conjuntos entre un grupo de agentes

- Sean dos agentes a_1 y a_2 que han llegado a una situación de conflicto y tienen que elaborar un plan conjunto
- El conjunto de negociación es $\{p_1, p_2, p_3, p_4\}$ y los roles $R = \{leader, follower\}$
- Sea la función de asignación de roles ρ tal que $Pr(\rho(a_1) = leader) = Pr(\rho(a_2) = leader) = 0.5$
- El protocolo $P = (K, \pi_{JPN})$, donde $K = \{PROPOSE, ACCEPT, MODIFY, CONFIRM\}$ y π_{JPN} se define como

$$\begin{aligned}
\pi_{JPN} &= \{PROPOSE(p)\} \\
\pi_{JPN}(ACCEPT(p)) &= \{CONFIRM(p)\} \\
\pi_{JPN}(MODIFY(p, p')) &= \{ACCEPT(p'), MODIFY(p', p'')\} \\
\pi_{JPN}(PROPOSE(p)) &= \{ACCEPT(p), MODIFY(p, p')\} \\
\pi_{JPN}(CONFIRM(p)) &= \{done\}
\end{aligned}$$

Estrategia de negociación para el ejemplo

La definiremos para cada agente i y cada elemento en K como sigue:

$$
\begin{array}{lcl}
\sigma_1(start, N, u_1) & = & \{(PROPOSE(p), N) \text{ siendo } p \in N,\ u_1(p) = \max_{p' \in N} u_1(p')\} \\
\sigma_i(PROPOSE(p), N, u_i) & = & \begin{cases} (ACCEPT(p), N) & si \quad u_i(p) \geq \max_{p' \in N} u_i(p') \\ (MODIFY(p, p'), N - \{p\}) & \text{si no} \quad \text{siendo } p' \in N \\ & \quad \text{y } u_i(p') = \max_{p' \in N} u_1(p') \end{cases} \\
\sigma_i(ACCEPT(p), N, u_i) & = & (CONFIRM(p), \}p\}) \\
\sigma_i(MODIFY(p, p'), N, u_i) & = & \begin{cases} (ACCEPT(p'), \{p'\}) & si \quad u_i(p') \geq \max_{\hat{p} \in N'} u_i(\hat{p}) \\ & \quad N' = N - \{p\} \\ (MODIFY(p, p'), N' - \{p'\}) & \text{si no} \quad \text{siendo } \hat{p} \in N' \\ & \quad u_i(\hat{p}) = \max_{\hat{p} \in N} u_i(\hat{p}) \\ & \quad N' = N - \{p\} \end{cases} \\
\sigma_i(CONFIRM(p), N, u_i) & = & (done, N)
\end{array}
$$

equivalente al protocolo de negociación denominado “de concesión monotónica” [2]

Instanciación concreta

Dada la siguiente definición para las funciones de utilidad de a_1 y a_2 respectivamente

$$u_1(p_1) = 4,\ u_1(p_2) = 3,\ u_1(p_3) = 2,\ u_1(p_4) = 1 \text{ y}$$

$$u_2(p_1) = 1,\ u_2(p_2) = 2,\ u_2(p_3) = 3,\ u_2(p_4) = 4,$$

la traza de la negociación sería la siguiente:

$$\begin{array}{lllr}
a_1: & (start, \{p_1, p_2, p_3, p_4\}, u_1) & \rightarrow & (PROPOSE(p_1), \{p_1, p_2, p_3, p_4\}) \\
a_2: & (PROPOSE(p_1), \{p_1, p_2, p_3, p_4\}, u_2) & \rightarrow & (MODIFY(p_1, p_4), \{p_2, p_3, p_4\}) \\
a_1: & (MODIFY(p_1, p_4), \{p_2, p_3, p_4\}, u_1) & \rightarrow & (MODIFY(p_4, p_2), \{p_2, p_3\}) \\
a_2: & (MODIFY(p_4, p_2), \{p_2, p_3\}, u_2) & \rightarrow & (MODIFY(p_2, p_3), \{p_3\}) \\
a_1: & (MODIFY(p_2, p_3), \{p_3\}, u_1) & \rightarrow & \{N = \{p_3\}; (ACCEPT(p_3), \{p_3\})\} \\
a_2: & (ACCEPT(p_3), \{p_3\}, u_2) & \rightarrow & \{(CONFIRM(p_3), \{p_3\}); done\} \\
a_1: & (CONFIRM(p_3), \{p_3\}, u_1) & \rightarrow & (done, \{p_3\})
\end{array}$$

N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge.
Automated negotiation: prospects, methods and challenges.
Int. J. of Group Decision and Negotiation, 2(10):199–215, 2001.

Jeffrey S. Rosenschein and Gilad Zlotkin.
Rules of Encounter. Designing Conventions for Automated Negotiation among Computers.
MIT Press, 1994.

Yoav Shohama and Moshe Tennenholtz.
On social laws for artificial agent societies: off-line design.
Artificial Intelligence, 73:231–252, February 1995.

Gerard Weiss, editor.
Multi-Agent Systems. A Modern Approach to Distributed Artificial Intelligence.
MIT Press, 1999.