Sistemas Multi-Agentes

Cooperación Distribuida

Dr. Alejandro Guerra-Hernández

Instituto de Investigaciones en Inteligencia Artificial
Universidad Veracruzana

Campus Sur, Calle Paseo Lote II, Sección Segunda No 112,
Nuevo Xalapa, Xalapa, Ver., México 91097

mailto:aguerra@uv.mx
https://www.uv.mx/personal/aguerra/sma

Maestría en Inteligencia Artificial 2024

SMA vs Sistemas Distribuidos

- El término cooperación suele usarse al hablar de sistemas distribuidos tradicionales, pero existen diferencias importantes con su uso en los SMA:
  - Los agentes pueden estar diseñados por diferentes individuos y tener metas distintas. Podrían incluso no tener metas comunes. Sus encuentros se parecen más a un juego donde actúan estratégicamente para alcanzar sus resultados preferidos.
  - Los agentes deben ser capaces de coordinar sus actividades y cooperar con los demás agentes en tiempo de ejecución. En los sistemas distribuidos tradicionales, los mecanismos de coordinación y cooperación se establecen estáticamente durante su diseño.

Universidad Veracruzana

Solución Cooperativa Distribuida de Problemas (CDPS)

- Planteada en los trabajos de Durfee, Lesser y Corkill [3].
- Estudia como una red débilmente acoplada de programas capaces de resolver problemas, puede proponer soluciones más allá de las capacidades individuales de sus miembros.
- La cooperación es necesaria porque ninguno de estos programas es capaz, por si mismo, de solucionar los problemas que se le plantean al sistema.
- Esto se debe a su falta de experiencia, de recursos, o de información.
- Las metas se asumen compatibles.

Universidad Veracruzana

Benevolencia

- La mayoría de los trabajos en CDPS asumen la benevolencia de los agentes.
- Los componentes del sistema comparten una meta común y, por lo tanto, no hay conflictos potenciales entre ellos.
- Esto supone que los agentes son diseñados para ayudar cuando es necesario. Lo que importa es resolver el problema planteado al sistema como un todo.
- La benevolencia se puede aceptar, cuando todos los agentes son diseñados y pertenecen a una misma organización o individuo.
- Asumir benevolencia simplifica enormemente el diseño de un SMA.

Universidad Veracruzana

Agenda propia

- Los trabajos en SMA asumen que los agentes involucrados en el sistema tienen sus propios intereses.
- Los agentes en un SMA no asumen metas comunes y son diseñados normalmente por diferentes organizaciones o individuos.
- Por lo tanto, los intereses de un agente pueden entrar en conflicto con los intereses de otros agentes.
- A pesar de estos conflictos, al igual que en un sistema CDPS, los agentes de un SMA deberán cooperar para resolver los problemas que enfrentan.

Universidad Veracruzana

Solución Paralela de Problemas

- Los sistemas de solución de problemas en paralelo fueron propuestos por Bond y Gasser [2].
- Sus componentes son simples procesos y un solo componente es responsable de descomponer la tarea principal en subproblemas asignados a estos procesos.
- Los componentes del sistema se asumen homogéneos, en el sentido que no tienen experiencia diferenciada.
- Aunque los sistemas de solución paralela se consideraban sinónimos de los CDPS, actualmente suelen considerarse como dos áreas de estudio diferentes.

Universidad Veracruzana

Evaluación de éxito (o fracaso)

Coherencia. ¿Qué tan bien el SMA se comporta como una unidad, sobre alguna dimensión de evaluación? Medida en términos de calidad de la solución, eficiencia en el uso de recursos, claridad conceptual de la operación, o qué tan bien el sistema se degrada en presencia de ruido y falta de información.

Coordinación. ¿En qué grado los agentes pueden evitar procesos relacionados con la sincronización y alineamiento de sus actividades. La presencia de conflictos donde los agentes interfieren entre si, es un indicador de coordinación pobre.

Universidad Veracruzana

Temas de Investigación

- ¿Cómo puede un problema dividirse en tareas más pequeñas que puedan distribuirse entre un grupo de agentes?
- ¿Cómo puede una solución sintetizarse efectivamente a partir de los resultados obtenidos para los sub-problemas?
- ¿Cómo pueden ser optimizadas las todas actividades para obtener soluciones que maximicen la medida de coherencia?
- ¿Qué técnicas pueden usarse para coordinar la actividad de los agentes, de forma que se eviten las interacciones destructivas y se maximice la efectividad al explotar cualquier interacción positiva?

Universidad Veracruzana

Paso 1: La Descomposición de un problema

- La descomposición en sub-problemas es típicamente jerárquico. Los diferentes niveles de descomposición corresponden a diferentes niveles de abstracción.
- La granularidad es importante: En extremo, la descomposición continua hasta que los sub-problemas representan acciones atómicas, e.g., ACTOR [1].
- ¿Quién llevará a cabo la descomposición de tareas?

Centralizada. Un sólo agente lleva a cabo esta tarea, necesitando tener experiencia sobre la estructura de la tarea en cuestión.

Descentralizada Todos los agentes tienen cierto conocimiento de la estructura del problema.

Universidad Veracruzana

Paso2: Solución de los sub-problemas

- Los problemas identificados en la fase de descomposición son resueltos individualmente por los agentes del SMA.
- Esto involucra una asignación de tareas.
- La fase involucra normalmente intercambio de información entre los agentes.

Universidad Veracruzana

Paso 3: Síntesis de la solución

- Las soluciones individuales a los sub-problemas identificados en la primera fase, son integradas en una solución global.
- Al igual que la descomposición, esta tarea puede ser jerarquizada: Las soluciones parciales se ensamblan en diferentes niveles de abstracción.

Universidad Veracruzana

Actividades en CDPS

- De acuerdo a Smith y Davis [6] hay dos actividades a considerar, la compartición de:

  Tareas. Cuando el problema se descompone en sub-problemas más pequeños que son asignados a los agentes, e.g., subastas y negociaciones.

  Resultados. Puede ser pro-activa si un agente comparte resultados que cree importantes para los demás agentes; o reactiva lo hace a solicitud expresa de otros agentes, e.g., actos de habla.

Universidad Veracruzana

Protocolo de Red de Contratos (CNP) I

- Propuesta por Smith [5], usa la metáfora de cómo las compañías ponen contratos a concurso.
- Para proponer una una tarea, se debe emitir un anuncio:
  - Si no conoce las capacidades de los demás, el administrador utilizará un *broadcasting* para anunciar la tarea.
  - Si conoce las competencias de otros agentes en la red, puede hacer un *broadcasting* limitado a los agentes de su interés.
  - Si tiene conocimientos suficientes, puede hacer convocatorias puntuales.
- El agente que anuncia una tarea se comprometerse a actuar como administrador de la misma. Conforme la solución del problema original progrese, diferentes administradores generarán diferentes anuncios de tarea.

Universidad Veracruzana

Protocolo de Red de Contratos (CNP) II

- Los agentes escuchan los anuncios y los evalúan con respecto a sus capacidades y recursos.
- Cuando un agente se reconoce capaz de llevar a cabo una tarea anunciada, emite una oferta que indica las capacidades del agente con respecto a la tarea a resolver.
- El administrador de la tarea recibe diversas ofertas y selecciona a los agentes más apropiados.
- La selección es comunicada a los agentes que tuvieron éxito, mediante un mensaje de reconocimiento.

Universidad Veracruzana

Protocolo de Red de Contratos (CNP) III

- Los agentes ganadores asumen la responsabilidad de ejecutar la tarea y se asumen como contratistas para la tarea que les ha sido asignada.
- Una vez que la tarea ha sido completada, el contratista envía un reporte al administrador.

Universidad Veracruzana

Simplificaciones al CNP

- Cuando el administrador sabe cual es el agente apropiado para la ejecución de una tarea dada, puede efectuar una contratación directa.
- En ese caso, los contratistas tienen la posibilidad de rechazar la tarea propuesta.

Universidad Veracruzana

Actos de habla vs anuncios de tareas

- Observen que para aquellas tareas que simplemente solicitan información, un contrato no es lo más apropiado. Una secuencia de actos de habla *ask-tell* es más adecuada en esos casos.
- Un contrato debe usarse para distribuir el control en la ejecución de una tarea, no simplemente datos.
- En Jason esto se puede hacer con actos de habla *achieve*.

Universidad Veracruzana

Recepción de mensajes I

Anuncio de tarea. El agente decide si es elegible para tal tarea, revisando la especificación que el mensaje incluye. Si es elegible, almacena los detalles de la oferta y lanza una oferta.

Propuesta. Los detalles de las ofertas de los posibles contratistas son almacenados por los administradores hasta que un plazo limite se cumple. Entonces el administrador premia al ganador.

Universidad Veracruzana

Recepción de mensajes II

Reconocimiento. Los agentes que no ganaron, borran los detalles de la tarea en cuestión. El ganador tratará de llevar a cabo la tarea, posiblemente generando nuevas sub-tareas.

Solicitudes e informes. Actos de habla *ask* y *tell*.

Universidad Veracruzana

Costo Marginal

- Supongamos que al tiempo t, un potencial contratista i tiene asignado un conjunto de tareas τ_i^t y que tiene recursos totales r_i.
- Entonces i recibe un anuncio de tarea con la especificación ts.
- Denotaremos con $\tau(ts)$ el conjunto de tareas especificado por ts.
- Sea $c_i^t(\tau)$ el costo para el agente i de llevar a cabo las tareas especificadas en τ al tiempo t.
- Entonces, el costo marginal de llevar a cabo las tareas es:

$$\mu_i(\tau(ts) \mid \tau_i^t) = c_i(\tau(ts) \cup \tau_i^t) - c_i(\tau_i^t)$$

Universidad Veracruzana

¿Ofertar o no ofertar?

- Si $\mu_i(\tau(ts)|\tau_i^t) = 0$, entonces el costo marginal (extra) de llevar a cabo $\tau(ts)$ es 0. Estas tareas pueden hacerse gratis.
- Si $\mu_i(\tau(ts)|\tau_i^t) < r_i$, el costo marginal es menor a los recursos a disposición del agente i y sería racional proponer una oferta; en cualquier otro caso no lo es.
- Si la especificación de la tarea ts incluye un posible pago $r(ts)$, entonces la decisión dependerá de si:

$$\mu_i(\tau(ts)|\tau_i^t) < (r(ts) + r_i)$$

Universidad Veracruzana

Mejoras al compartir resultados

Confianza. Soluciones derivadas de manera independiente, pueden someterse a una validación cruzada, señalando posibles errores e incrementando la confianza de la solución global.

Completitud. Los agentes pueden compartir su perspectiva local para lograr una mejor visión global.

Precisión. Los agentes pueden compartir resultados para incrementar la precisión de sus soluciones globales.

Tiempo. Aunque un agente pueda resolver una tarea por si mismo, al compartirse resultados parciales, podría terminar su tarea más rápido.

Universidad Veracruzana

Un CNP en Jason

```
1  MAS cnp {
2
3    infrastructure: Centralised
4
5    agents:
6      nuncaContesta;
7      siempreRechaza;
8      ofertaAleatoria #3;
9      admin;
10
11   aslSourcePath: "src/asl";
12 }
```

Universidad Veracruzana

Agente administrador I

- Este agente puede iniciar procesos CNP:

```
13  !iniciarCNP(1,componer(computadora)).
14
15  /* Plans */
16
17  +!iniciarCNP(Id,Tarea) <-
18      .print("Esperando a los participantes...");
19      .wait(2000);
20      +estadoCNP(Id,proponer);   // recordar estado CNP
21      .findall(Nombre,intro(participante,Nombre),LP);
22      .print("Enviando CFP a ",LP);
23      .send(LP,tell,cfp(Id,Tarea));
24       // el plazo del cfp es dentro de 4 segs.
25       .at("now +4 seconds", { +!contrato(Id) }).
```

Universidad Veracruzana

Agente administrador II

- Y puede procesar el contrato que ha anunciado:

```
35  @lc1[atomic]
36  +!contrato(CNPId)
37     :  estadoCNP(CNPId,proponer)
38     <- -+estadoCNP(CNPId,contratar);
39        .findall(oferta(O,A),oferta(CNPId,O)[source(A)],L);
40        .print("Las ofertas son ",L);
41        L \== []; // constraint the plan execution to at least one offer
42        .min(L,oferta(WOf,WAg)); // sort offers, the first is the best
43        .print("El ganador es ",WAg," con ",WOf);
44        !anunciarResultado(CNPId,L,WAg);
45        -+estadoCNP(CNPId,fin).
46
47  +!contrato(_).
48
49  -!contrato(CNPId)
50     <- .print("CNP ",CNPId," ha fallado.").
```

Universidad Veracruzana

Agente administrador III

- La recepción de ofertas y rechazos también pude activar el procesamiento de contratos:

```
27  +oferta(CNPId,_)
28     :  estadoCNP(CNPId,proponer) & todasPropuestasRecibidas(CNPId)
29     <- !contrato(CNPId).
30
31  +rechazo(CNPId)
32     :  estadoCNP(CNPId,proponer) & todasPropuestasRecibidas(CNPId)
33     <- !contrato(CNPId).
```

- La regla para computar si todas las propuestas han sido recibidas es:

```
5   todasPropuestasRecibidas(CNPId) :-
6       .count(intro(participante,_),NP) &  // No participantes
7       .count(oferta(CNPId,_), NO) &       // No ofertas recibidas
8       .count(rechazo(CNPId), NR) &        // No rechazos recibidos
9       NP = NO + NR.
```

Universidad Veracruzana

Agente administrador IV

- El anuncio de resultados se lleva a cabo de la siguiente manera:

```
52  +!anunciarResultado(_,[],_).
53
54  +!anunciarResultado(CNPId,[oferta(_,WAg)|T],WAg)
55      <- .send(WAg,tell,ofertaAceptada(CNPId));
56         !anunciarResultado(CNPId,T,WAg).
57
58  +!anunciarResultado(CNPId,[oferta(_,LAg)|T],WAg)
59      <- .send(LAg,tell,ofertaRechazada(CNPId));
60         !anunciarResultado(CNPId,T,WAg).
```

Universidad Veracruzana

Agentes que ofertan aleatoriamente I

- Estos agentes calculan su precio aleatoriamente:

```
3  // Obtener un precio para el producto como un valor aleatorio entre 100 y
   ↪  110
4  precio(_,X) :- .random(R) & X = (10*R)+100.
```

- Y saben quien es el administrador para registrarse ante él:

```
6  rol(inicia,admin).
7
8  // Presentarse ante el agente que inicia en CNP
9  +rol(inicia,Ag)
10    :  .my_name(Yo)
11    <- .send(Ag,tell,intro(participante,Yo)).
```

Universidad Veracruzana

Agentes que ofertan aleatoriamente II

- Responden a los CFP de la siguiente manera:

```
13  // responder a un cfp
14  +cfp(CNPId,Tarea)[source(Ag)]
15     :  rol(inicia,Ag) & precio(Tarea,Oferta)
16     <- +oferta(CNPId,Tarea,Oferta); // recordar mi oferta
17        .send(Ag,tell,oferta(CNPId,Oferta)).
```

- Y a los anuncios del ganador:

```
19  +ofertaAceptada(CNPId)
20     :  oferta(CNPId,Tarea,Oferta)
21     <- .print("Mi oferta ",Oferta," ganó el CNP ",CNPId,
22               " para la tarea ",Tarea,").
23        // Hacer la tarea y reportar los resultados a admin
24
25  +ofertaRechazada(CNPId)
26     <- .print("Perdí el CNP ", CNPId, ".");
27        -oferta(CNPId,_,_).
```

Universidad Veracruzana

Agente que siempre rechaza los cfp

```
1  // Agente siempreRechaza en proyecto cnp
2
3  rol(inicia,admin).
4
5  +rol(inicia,Ag)
6     :  .my_name(Yo)
7     <- .send(Ag,tell,intro(participante,Yo)).
8
9  +cfp(CNPId,_)[source(Ag)]
10    :   rol(inicia,Ag)
11    <- .send(Ag,tell,rechazar(CNPId)).
```

Universidad Veracruzana

Agente que nunca contesta

```
1  // Agente nuncaContesta in project cnp
2
3  // El agente que tiene el rol de iniciar el CNP
4  rol(inicia,admin).
5
6  // Enviar un mensaje a ese agente anunciándose como participante
7  +rol(inicia,Ag)
8     :  .my_name(Yo)
9     <- .send(Ag,tell,intro(participante,Yo)).
```

Universidad Veracruzana

Salida en consola

```
1 [admin] Esperando a los participantes...
2 [admin] Enviando CFP a [ofertaAleatoria1, nuncaContesta,
  ↪ ofertaAleatoria2, siempreRechaza, ofertaAleatoria3]
3 [admin] Las ofertas son [oferta(103.77618324256807, ofertaAleatoria3),
4 oferta(108.985744452349, ofertaAleatoria2),
  ↪ oferta(106.33970211579945,ofertaAleatoria1)]
5 [admin] El ganador es ofertaAleatoria3 con 103.77618324256807
6 [ofertaAleatoria3] Mi oferta 103.77618324256807 ganó el CNP 1 para la
  ↪ tarea componer(computadora).
7 [ofertaAleatoria1] Perdí el CNP 1.
8 [ofertaAleatoria2] Perdí el CNP 1.
```

Universidad Veracruzana

Formas de inconsistencia en un SMA

Metas. Se debe a que se asume que los agentes son autónomos, y por lo tanto no comparten necesariamente objetivos comunes.

Creencias. Tienen fuentes diversas:

- El punto de vista de un agente es limitado, es de suponer que ningún agente tiene información completa sobre su ambiente.
- Los sensores de los agentes suelen ser ruidosos.
- Las fuentes de información de los agentes pueden ser poco confiables.

Universidad Veracruzana

Soluciones

- Evitar que esto suceda, o al menos ignorar las inconsistencias. Este es el enfoque que adopta CNP. La compartición de tareas es siempre controlada por un agente administrador, quien tiene autoridad sobre el problema en cuestión.
- Resolver las inconsistencias mediante negociación, por ejemplo, usando subastas. Mientras que esto es al menos teóricamente deseable, los costos de comunicación y computación sugieren que raramente es realizable en la práctica.
- Construir sistemas que se degradan con gracia ante la presencia de inconsistencia. Lesser et al. [4] se refieren a estos sistemas como funcionalmente precisos y cooperativos (FA/C).

Universidad Veracruzana

Sistemas FA/C

- La solución no está restringida estrictamente a una secuencia particular de eventos. Se progresa de manera oportunista e incremental.
- Los agentes comunican resultados intermedios de alto nivel, en lugar de comunicar datos simples.
- La inconsistencia se resuelven implícitamente por intercambio de soluciones para su comparación.
- Solución oportuna, ni antes, ni después.
- No restringirse a una sola ruta posible, se consideran diversas alternativas de llegar a una solución.
- Esto hace que el sistema sea robusto ante fallas locales y cuellos de botella.

Universidad Veracruzana

Referencias I

[1] GA Agha. *Actors: a Model of Concurrent Computation in Distributed Systems, Series in Artificial Intelligence*. Inf. téc. 844. Cambridge, MA, USA: MIT Artificial Intelligence Laboratory, 1985.

[2] A Bond y L Gasser. *Distributed Artificial Intelligence*. Morgan Kaufmann Publishers Inc., San Francisco, CA, 1988.

[3] EH Durfee, VR Lesser y DD Corkill. "Trends in cooperative distributed problem solving". En: *Knowledge and Data Engineering, IEEE Transactions on* 1.1 (1989), págs. 63-83.

[4] VR Lesser y DD Corkill. "Functionally Accurate, Cooperative Distributed Systems". En: *IEEE Transactions on Systems, Man, and Cybernetics* 11.1 (1981), págs. 81-96.

[5] RG Smith. "The Contract Network Protocol: Communication and Control in a DistributedProblem Solver". En: *IEEE Transactions on computers* c-29.12 (1980), págs. 1104-1113.

[6] RG Smith y R Davis. "Frameworks for Cooperation in Distributed Problem Solving". En: *IEEE Transactions on Systems, Man, and Cybernetics* 11.1 (1981), págs. 61-70. ISSN: 0018-9472. URL: http://dx.doi.org/10.1109/TSMC.1981.4308579.

Universidad Veracruzana