



UNIVERSIDAD  
DE LOS ANDES  
MÉRIDA VENEZUELA



**UTPL**  
UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

# Inteligencia Artificial Distribuida

# La IAD

- En diferentes dominios se han hecho esfuerzos por plantearse formas para la solución de problemas cooperativamente.
- La Inteligencia Artificial Distribuida (IAD) es uno de los dominios que ha estudiado ese problema, en este caso, usando conceptos provenientes de:
  - La Inteligencia Artificial (IA)
  - Los Sistemas Distribuidos.

# Generalidades

- La IAD involucra *comunidades de agentes* donde los agentes resuelven (sub)tareas
- **Áreas principales:**
  - Sistemas Multi-agentes (SMA)
  - Resolución Distribuida de Problemas (RDP)
- **La IAD tiene que ver con:**
  - Granularidad de los agentes
  - Heterogeneidad de los agentes
  - Métodos para distribuir las tareas, el control, etc., entre los agentes
  - Protocolos de comunicación entre los agentes
- **La IAD no tiene que ver con:**
  - Coordinación de procesos concurrentes
  - Arquitecturas paralelas, Lenguajes de programación paralela, Sistemas operativos distribuidos

# La IAD

## Técnicas de resolución de problemas que mezclan tanto la IA como los Sistemas Distribuidos.

- Los sistemas basados en IAD integran al menos dos agentes, los cuales interactúan para la resolución de un problema dado.
- Los agentes son frecuentemente heterogéneos y deben tener cierto grado de *autonomía, capacidades de razonamiento, planificación, etc.*
- Los agentes deben ser capaces de *interactuar, negociar, cooperar* y hasta *competir* con otros para llevar a cabo sus tareas, las cuales pueden ser individuales o grupales.

# La IAD

## Objetivos IAD

Desarrollar mecanismos y métodos que permiten a los agentes interactuar, así como lo hacen los seres humanos (o incluso mejor), comprender la interacción entre entidades inteligentes, ya sean computacionales, humanas, o ambas.

Estos objetivos plantean una serie de retos que se centran en torno a la cuestión elemental de:  
**cuándo y cómo interactuar y con quién.**

# La IAD

## Dos razones principales de la IAD

1. Tiene un papel clave en las plataformas informáticas y entornos de información actuales: distribuidos, amplios, abiertos y heterogéneos.
  - Las computadoras están clásicamente conectadas.
  - La creciente complejidad de los sistemas informáticos y de información a menudo excede el nivel de la computación convencional, centralizada, ya que requieren, por ejemplo, el procesamiento de grandes cantidades de datos, o los datos están en lugares geográficamente distintas.
- Para hacer frente a estas situaciones, las computadoras tienen que actuar más como "individuos" o "agentes", en lugar de sólo "partes".

**IAD proporciona una gestión de la interacción de alto nivel requeridas para las aplicaciones computacionales complejas modernas.**

# La IAD

## Dos razones principales de la IAD

2. IAD tiene la capacidad de desempeñar un papel importante en el desarrollo y análisis de modelos y teorías de la interactividad en las sociedades humanas.
  - Los seres humanos interactúan de diversas maneras y en muchos niveles: por ejemplo, observan y modelan unos a otros, solicitan y proporcionan información, negocian y discuten, desarrollan puntos de vista compartidos de su entorno, detectan y resuelven conflictos, y se forman y se disuelven estructuras organizacionales, tales como equipos, comités y economías.
  - Muchos procesos interactivos entre los seres humanos son todavía poco conocidos, a pesar de que son una parte integral de nuestra vida cotidiana.

**IAD permite explorar fundamentos sociológicos y psicológicos**

# La IAD

El papel que el concepto de SMAs juega en IAD es comparable al papel que el concepto de agente desempeña en la IA.

**Tanto IAD e IA consideran aspectos computacionales de la inteligencia, pero lo hacen desde diferentes puntos de vista y bajo diferentes supuestos:**

- IA se centra en agentes como "sistemas autónomos inteligentes" y la inteligencia como una propiedad del sistema que actúa aisladamente, la IAD se centra en agentes como *"sistemas inteligentes conectados"* y la inteligencia como una *propiedad de los sistemas que interactúan*.
- IA se centra en los "procesos cognitivos" de los individuos, la IAD se centra en los *"procesos sociales"* de grupos de individuos.

# La IAD

El papel que el concepto de SMAs juega en IAD es comparable al papel que el concepto de agente desempeña en la IA.

**Tanto IAD e IA consideran aspectos computacionales de la inteligencia, pero lo hacen desde diferentes puntos de vista y bajo diferentes supuestos:**

- IA considera que los sistemas tienen un único foco de razonamiento y de control interno y requieren sólo una ayuda mínima de otros para actuar, la IAD considera sistemas en los que se *distribuye el razonamiento, el control y las actividades*.
- IA usa ideas, inspiraciones y metáforas de la psicología y el conductismo , la IAD utiliza *la sociología y la economía*.

**IAD no es una especialización de la IA, sino una generalización de la misma.**

# La IAD

## IAD como sistema distribuido, tiene las siguientes propiedades:

- **Velocidad y eficiencia** - Los agentes pueden funcionar de forma asíncrona y en paralelo, aumentando la velocidad en general (si la sobrecarga por la coordinación no penalice esa ganancia).
- **Robustez**- El fallo de un o varios agentes no significa necesariamente que el sistema en general quede inútil, ya que otros agentes disponibles en el sistema pueden hacerse cargo de su parte.
- **Escalabilidad y flexibilidad** - El sistema puede soportar el aumento del tamaño del problema mediante la adición de nuevos agentes, y esto no necesariamente afecta la operatividad del sistema.
- **Costos** - puede ser mucho más rentable que un sistema centralizado, ya que podría estar compuesto por subsistemas simples de bajo costo unitario.

# La IAD

## IAD como sistema distribuido, tiene las siguientes propiedades:

- **Desarrollo y reutilización** - agentes individuales pueden ser desarrollados por separado por especialistas (ya sea desde cero o basado en cosas ya está disponibles), el sistema global se puede probar y mantener con más facilidad, y se puede reconfigurar y reutilizar agentes en diferentes escenarios de aplicación.
- **La tecnología informática y de red disponible constituye una plataforma sólida para la realización de estos sistemas.** En particular, la evolución reciente de la computación paralela y distribuida, la computación móvil, etc.

# Generalidades

- **En IAD:**
  - No es obligatorio pensar en un *control* central; el control puede ser distribuido
  - El *conocimiento* o las *fuentes de información* pueden también estar distribuidos
- **Un RDP** considera la tarea de resolver un problema en particular, para lo cual lo descompone en módulos (división de tareas) que serán resueltos cooperativamente por un grupo de agentes.
- **Un SMA** tiene que ver con el comportamiento de una comunidad de agentes autónomos tratando de resolver problemas, por lo cual comparten conocimiento acerca de los problemas y sus soluciones.

# RDP

Estudia los sistemas inteligentes distribuidos con una funcionalidad global. Los agentes cumplen con características mínimas de:

- *Benevolencia*
- *Objetivo Global Compartido*
- *El Diseño es Centralizado:*
  - Todos los agentes son diseñados para su integración en un sistema inteligente capaz de resolver un problema.
  - El diseñador debe asegurar que los agentes utilicen el mismo lenguaje, que cada agente influya en la consecución del objetivo global, etc.
- El procedimiento de un sistema RDP es:
  1. Descomposición de tareas.
  2. Asignación de tareas y recursos a los agentes.
  3. Resolución de sub-problemas.
  4. Integración de soluciones.

# SMA

Un sistema multiagente es un sistema informático formado por *un grupo de agentes que interactúan entre sí* utilizando protocolos y lenguajes de comunicación de *alto nivel*, para resolver problemas que pueden estar *más allá de* las capacidades o del conocimiento de cada uno.

Es una *red de entidades que trabajan conjuntamente* para encontrar respuesta a problemas que pueden estar más allá de la capacidad y el conocimiento individual de cada entidad.

# SMA

- Son sistemas compuestos por múltiples componentes *autónomos* que poseen las siguientes características:
  - Cada agente tiene ciertas capacidades.
  - Necesitan de formas de coordinarse
  - Los datos no están centralizados
  - La computación es asíncrona.
- Los SMA buscan responder a preguntas, tales como:
  - ¿Por qué y cómo cooperan los agentes?
  - ¿Cómo los agentes pueden reconocer y resolver conflictos?
  - ¿Cómo los agentes pueden negociar?.

# SMA

sistemas en los que varios **agentes inteligentes interactúan**, siguiendo un conjunto de metas o realizar un conjunto de tareas.

- "*Agentes*" son entidades computacionales autónomas que perciben su entorno a través de sensores y actúan sobre él a través de efectores.
- "*Inteligente*" indica que los agentes persiguen sus objetivos y ejecutan sus tareas optimizando algunas medidas de rendimiento. Funcionan flexible y racionalmente en una variedad de circunstancias. Para ello, en IAD los procesos de resolución de problemas, planificación, toma de decisiones, y aprendizaje que hacen posible que los agentes sean flexibles y racionales en su comportamiento, se llevan a cabo en escenarios multiagente.

# SMA

**sistemas en los que varios agentes inteligentes interactúan, siguiendo un conjunto de metas o realizar un conjunto de tareas.**

- "Interacción" indica que los agentes pueden verse afectados por otros agentes en la consecución de sus objetivos y ejecución de sus tareas. Interacción puede ser:
  - Indirecta a través del entorno (por ejemplo, mediante la observación de los otros o llevando a cabo una acción que modifica el ambiente)
  - Directa a través de un lenguaje compartido (por ejemplo, intercambiando información en la que otros agentes están interesados o que confunde otros agentes).
- DAI se centra en la *coordinación* como una forma de interacción para la consecución de objetivos y la realización de tareas.
- La finalidad de la coordinación es lograr estados deseables o evitar estados indeseables para varios agentes.
- Dos patrones básicos de coordinación: cooperación y competencia

# SMA

## SMA

Se enfocan primordialmente en la coordinación o negociación entre los agentes

Son autónomos con la posibilidad de objetivos individuales

Permite la emergencia

## RPD

Se enfocan en la resolución de un pb. dado, su descomposición, y síntesis de la soluciones

Necesitan sincronizarse

Es determinista

# RDP

Es un subcampo de la inteligencia artificial distribuida (IAD) en el que el énfasis está en conseguir que los agentes trabajen juntos para resolver problemas que requieren un esfuerzo colectivo.

Debido a una distribución inherente de los recursos como el conocimiento, la capacidad, la información y experiencia entre los agentes, un agente en es incapaz de cumplir sus propias tareas por sí solo.

RDP requiere

- **Coherencia** (es decir, los agentes necesitan querer trabajar juntos): se asume un cierto grado de coherencia, ya que los agentes son diseñados para trabajar juntos; o la ingeniería social introduce desincentivos para el individualismo de los agentes; etc.
- **competencia** (es decir, los agentes necesitan saber cómo trabajar juntos): La tarea entonces consiste en como se distribuyen las competencias; como cualquiera que haya jugado en un equipo, trabajado en un proyecto de grupo, o participado en una orquesta musical,  
**simplemente tener el deseo de trabajar juntos no asegura un resultado colectivo exitoso**

# RDP

- Presume la existencia de problemas que necesitan ser resueltos y expectativas acerca de lo que constituyen las soluciones.
  - Por ejemplo, un problema a resolver podría ser para un equipo de agentes diseñar un artefacto (por ejemplo, un coche).
  - La solución que formulen deberá cumplir los requisitos generales (debe tener cuatro ruedas, el motor debe encajar dentro del compartimiento del motor y ser potente como para mover el coche, etc.),
  - debe existir en una forma particular (documento de especificaciones para el montaje del carro).
- Los agentes formulan soluciones por cada subproblema (uno o más) y sintetizan estas soluciones de los subproblema en soluciones globales.

# RDP

**A veces el problema de los agentes es construir un plan.**

- Incluso, si los agentes están resolviendo otro tipo de problemas, tienen que resolver problemas de planificación también. Es decir, los agentes deben planear cómo trabajar juntos:

descomponer problemas en subproblemas, asignar estos subproblemas, intercambiar soluciones de los sub-problemas, y sintetizar soluciones globales-

**Son en sí un problema que los agentes tienen que resolver!!**

**Planificación distribuida está así estrechamente entrelazada con la resolución de problemas distribuida, siendo a la vez un problema en sí mismo y un medio para resolver un problema.**

# RDP

## Motivaciones para la resolución de problemas distribuida

### Los conocimientos o capacidades de resolución de problemas se pueden distribuir

Por ejemplo, en la ingeniería concurrente, un problema podría implicar el diseño y fabricación de un artefacto (por ejemplo, un coche) con agentes especializados que hagan individualmente componentes y procesos, y la combinación de éstos en una solución colectiva. (ejemplos son los procesos de automatización de fábrica, etc.) puede implicar una interacción entre las piezas separadas para monitorear los eventos, evaluar la situación, generar la respuestas, etc.

- El problema es emplear diversas capacidades para resolver problemas que no sólo son de gran tamaño, pero también multifacéticos.
- Ejemplo de la capacidad distribuida, es una **red de sensores distribuido** en un estacionamiento para monitorizar los movimientos de los vehículos. La tarea general de vigilancia no se puede hacer desde una ubicación central. El problema es, pues, descomponer la tarea de monitoreo en subtarear que se pueden asignar adecuadamente a agentes distribuidos geográficamente.

# RDP

## Motivaciones para la resolución de problemas distribuida.

**El uso de los recursos distribuidos simultáneamente puede permitir una aceleración de la resolución de problemas gracias a paralelismo.**

Las posibles mejoras debido al paralelismo dependen, por supuesto, en el grado de paralelismo inherente en un problema.

### **Los datos pueden ser distribuidos.**

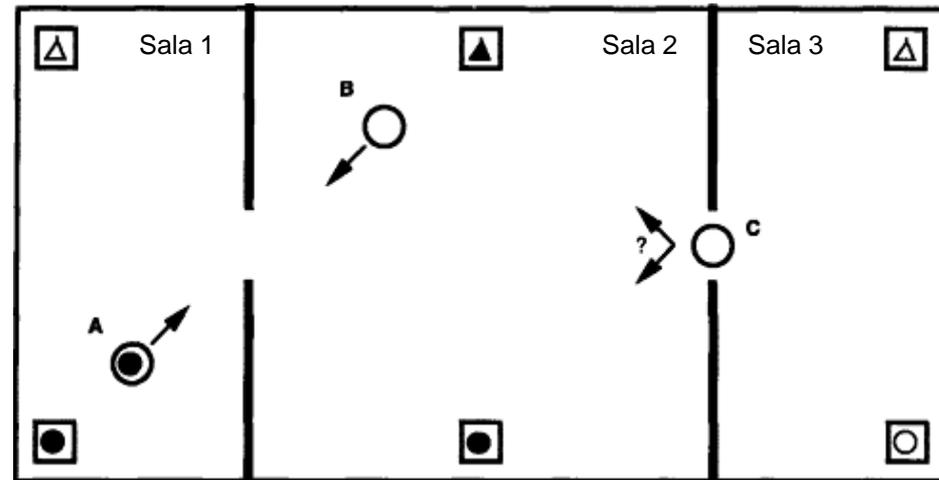
Para el ejemplo de la red de sensores distribuidos, hacer el seguimiento de vehículos distribuido en principio, podría ser centralizado: cada uno de los agentes de sensores distribuidos podría transmitir los datos en bruto a un sitio central para su interpretación. Esta estrategia centralizada, sin embargo, podría implicar enormes cantidades de comunicación innecesaria en comparación si se permite a los agentes sensores interpretar localmente qué debería ser transmitido de manera selectiva.

# RDP

## Motivaciones para la resolución de problemas distribuida.

### Los resultados de la resolución de problemas o la planificación son distribuidos.

Por ejemplo, en una tarea que implica la entrega de objetos en diferentes ubicaciones, agentes de entrega distribuidos pueden actuar en paralelo. La formulación de planes se podría hacer en un sitio centralizado (despachador) o puedan implicar distribuir su realización entre ellos. Por otra parte, durante la ejecución de sus planes, características del ambiente que no se conoce en tiempo de planificación, o que inesperadamente cambian, puede desencadenar cambios en lo que los agentes deben hacer.



Una vez más, todas estas decisiones podrían ser realizadas por un coordinador central, pero por diversas razones (que explotan el paralelismo, la disponibilidad coordinador esporádica, canales de comunicación lentos, etc.) podría ser preferible para los agentes modificar sus planes de manera unilateral o con comunicación limitada entre ellos

# RDP

## Estrategias de resolución de problemas distribuidos

**Reparto de tareas:** Cuando un agente tiene muchas tareas que hacer, se debe contar con la ayuda de agentes con pocas o ninguna tarea. Los pasos principales en el reparto de tareas son:

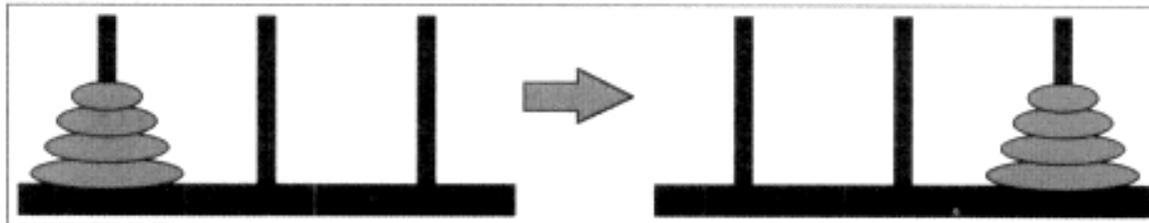
- *Descomposición de tareas:* Generar el conjunto de tareas que potencialmente serán pasadas a otros. En general, esto podría implicar la descomposición de tareas grandes en tareas parciales que podrían ser abordados por diferentes agentes.
- *Asignación de tareas:* Asignar tareas parciales a los agentes apropiados.
- *Logro de Tareas:* Los agentes logran sus subtareas, que podrían incluir la descomposición adicional y asignación de las subsubtask, de forma recursiva, hasta el punto de que un agente pueda realizar la tarea por si solo.
- *Síntesis de Resultado:* Cuando un agente logra su subtarea, pasa el resultado al agente apropiado (por lo general el agente original, ya que conoce las decisiones de descomposición y por lo tanto es más probable saber cómo componer los resultados en una solución global).

Los diferentes pasos pueden ser más o menos difíciles. Por ejemplo, a veces un agente sobrecargado comienza con un conjunto de tareas separadas, por lo que la descomposición es innecesaria; A veces, el agente puede pasar tareas a una serie de agentes idénticos, por lo que la asignación es trivial; y, a veces las tareas no dan ningún resultado que necesite ser sintetizado

# RDP

Un problema clásico de la IA que muestra lo anterior es el problema de la Torre de Hanoi (ToH)

- ToH: El objetivo es mover los discos de la clavija de inicio a otra clavija, moviendo sólo un disco a la vez, sin poder colocar un disco más grande en la parte superior de un disco más pequeño. El problema es encontrar una secuencia de movimientos que permitan alcanzar el estado final.



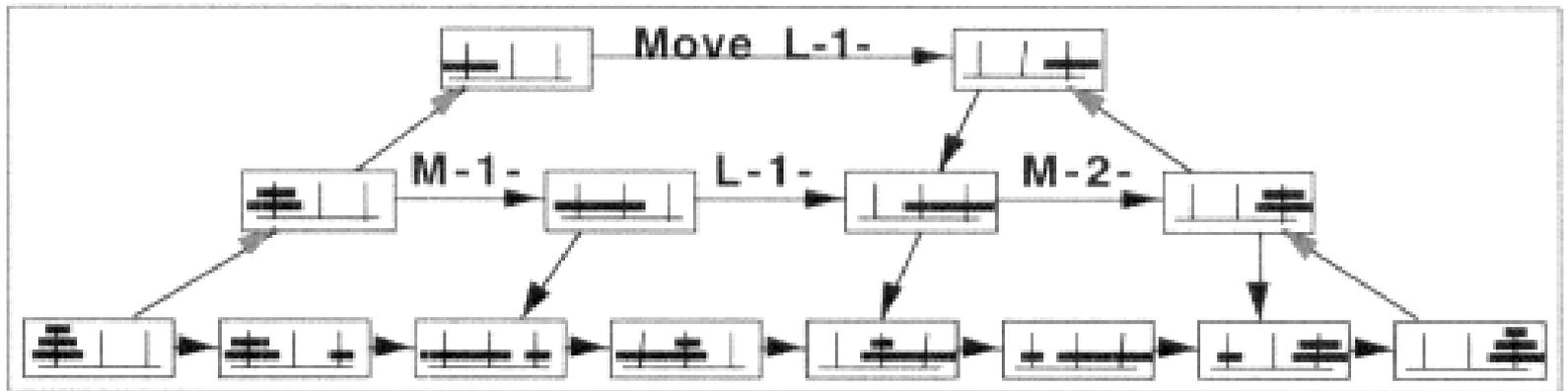
# RDP

## Reparto de tareas en ese problema

1. *Tarea de descomposición*: mover el disco más grande que no está en su clavija de destino conduce a una descomposición recursiva: resolver el problema de llegar al estado en el que más grande de disco se puede mover. Este subproblema se puede descomponer aún más en el problema de mover el segundo disco mayor, etc.
2. *Asignación de tareas*: Si suponemos un número indeterminado de agentes desocupados idénticos capaces de resolver el problema, la asignación se reduce a sólo asignar una tarea al azar a uno de estos agentes.
3. *Logro de Tareas*: En general, un agente debe encontrar la diferencia más significativa entre los estados de inicio y el objetivo y descomponer el problema en base a éstos. Si los estados de inicio y el objetivo son los mismos, entonces la descomposición recursiva termina.
4. *Síntesis de los Resultados*: Cuando un agente ha solucionado su problema, pasa la solución de nuevo hacia arriba. Cuando un agente ha recibido soluciones a todos los subproblemas que paso hacia abajo, los puede componer en una secuencia más completa de movimientos, y luego pasar esto como su solución.

# RDP

## Reparto de tareas en ese problema



# RDP

## Reparto de tareas en sistemas heterogéneos

Una de las motivaciones para la RDP es que es difícil construir artefactos para ser competente en cada tarea posible. Además, incluso si es factible construir un agente capaz a menudo es excesivo porque la mayoría de esas capacidades se desperdicia. La estrategia en los sistemas humanos es combinar especialistas en diferentes áreas para resolver problemas.

- Es posible que un agente tenga una tabla que identifica las capacidades de los agentes, por lo que simplemente selecciona un agente apropiado y envía el subproblema

# RDP

## Reparto de tareas en sistemas heterogéneos

Pero por lo general la información es más dinámica. Por ejemplo, si hay varios agentes candidatos , algunos ya se han comprometido,

**Cómo se descubre eso? Una forma es utilizar el protocolo Net-Contrat**

- Un gerente difunde su anuncio para llegar a los agentes de que es actualmente desconocen también.
- Una estrategia muy simple es volver a intentar el anuncio periódicamente, en el supuesto de que, finalmente, un contratista va a liberar.
- Podría ser que el gerente estaba siendo demasiado exclusiva en los que sería entretener a las ofertas de. Por lo tanto, el gerente podría participar en revisión iterativa de su anuncio, se relajan los requisitos de elegibilidad hasta que comience a recibir ofertas.
- El administrador puede intentar descomponer el problema de tal manera que los contratistas están disponibles para los subproblemas alternativas.<sup>30</sup>

# RDP

## Compartiendo Resultado

Los resultados de la tarea de un solucionador del problema podría diferir de los resultados de la misma tarea por otro solucionador de problemas. Por ejemplo, los alumnos de una clase tienen a menudo la misma tarea (problema), pero sus soluciones no lo son (no deberían ser idénticas). Al compartir los resultados, se puede mejorar el desempeño del grupo:

- **Confianza:** generación independientemente de resultados para la misma tarea pueden utilizarse para corroborar el uno al otro, produciendo un resultado colectivo con mayor confianza de ser correcta. Por ejemplo, al estudiar para un examen, los estudiantes pueden trabajar por separado un ejercicio y luego comparar las respuestas para aumentar la confianza.
- **Integridad:** Cada agente formula resultado para la subtarea que puede cumplir, y estos resultados cubren por completo una parte más completa de la tarea global. Por ejemplo, en el control de vehículos distribuido, un mapa más completo de los movimientos de vehículos es posible cuando los agentes comparten sus mapas.

# RDP

## Compartiendo Resultado

- **Precisión:** Para refinar su propia solución, un agente necesita saber más acerca de las soluciones que otros han formulado. Por ejemplo, en una aplicación de la ingeniería concurrente, cada agente por separado podría llegar a las especificaciones para una parte de un artefacto, al compartir estas especificaciones se puede afinar más como ellos encajaran precisión.
- **Puntualidad:** Incluso si un agente podría resolver, en principio, una gran tarea por sí sola, la solución de subtareas en paralelo puede dar una solución global más rápido. Acumular los beneficios de compartir resultado obviamente significa que los agentes tienen que compartir los resultados.

### **Pero hacer este trabajo es más difícil de lo que piensas!:**

- Los agentes deben saber qué hacer con los resultados compartidos:
- Dado que la asimilación podría ser no trivial, que la comunicación de grandes volúmenes de los resultados puede ser costoso, y que la gestión de muchos resultados asimilados incurre en gastos generales, los agentes deben tratar de ser lo más selectivos posible acerca de lo que intercambien.

# RDP

## Compartiendo Resultado: Repositorios compartidos y búsqueda negociado

Una de las estrategias para reducir el potencial aluvión de mensajes de multidifusión es en concentrar resultados parciales en un único repositorio compartido. La arquitectura de pizarra, por ejemplo, permite colocar fuentes de conocimiento diferentes para intercambiar resultados.

En esencia, el uso de un repositorio compartido puede soportar búsqueda a través de diseños alternativos, donde los agentes con diferentes criterios de diseño pueden revisar y criticar las alternativas.

Por ejemplo, los agentes involucrados en la búsqueda negociado tienen a su disposición una variedad de operadores para la RDP:

- iniciar solución (proponer un nuevo punto de partida para una solución);
- extender-solución (Revisar una solución parcial ya existente);
- crítica-solución (dar feedback sobre la viabilidad de una solución parcial ya existente);
- relajar-requisito -solución (cambiar los requisitos locales para la solución de aceptabilidad).

# Planificación Distribuida

- Puede pensarse como una especialización de la RDP, donde el problema es alrededor de un plan y su ejecución .
- La planificación distribuida es un término ambiguo, porque no está claro lo que se "distribuye".
  - El plan está formulado y se distribuye su ejecución.
  - El proceso de planificación es distribuido.
  - Ambas cuestiones son distribuidas.
  - por supuesto no se considera el caso donde ninguno lo es (ya que es la planificación centralizada tradicional)

# Planificación Distribuida

- Planificación Centralizada para Planes Distribuidos
- Planificación Distribuida para Planes Centralizados
- Planificación Distribuida para Planes Distribuidos

# Planificación Distribuida para Planes Centralizados

- La formulación de un plan complejo podría requerir de la colaboración de varios especialistas, al igual que generar la solución a cualquier problema complejo. Por lo tanto, el proceso de planificación distribuida podría ser realizado por varios agentes, cada uno contribuye al plan, hasta crear un plan global.
- La formulación del problema puede ser pensado en como descomponer la planificación y distribuirla entre varios especialistas, cada uno genera partes del plan
- Tanto las estrategias de compartir resultados o compartir tareas se pueden usar en este caso.

# Planificación Centralizada para Planes Distribuidos

Planes que se ejecutan de manera distribuida, no obstante se formulan de manera centralizada.

- Un **planificador de orden parcial** puede generar planes que no tienen un orden estricto.
- Esas acciones se pueden ejecutar en **paralelo**.
- Un **agente coordinador** centralizado un plan de este tipo y lo descompone en hilos separados, posiblemente con algunas acciones de sincronización.
- Estas **partes planas separadas se pueden pasar a los agentes** que pueden ejecutarlas.
- Los **agentes operan en paralelo** y logran alcanzar un estado del mundo consistente con los objetivos del plan.

# Planificación Centralizada para Planes Distribuidos

- Esquema:
  - Planificación en orden parcial que puede ser ejecutada en paralelo (diferentes agentes hacen ramas diferentes).
  - Se debe garantizar no condiciones de dependencia.
- Algoritmo:
  1. Diseñar un plan general teniendo en cuenta una descripción meta, un conjunto de operadores y un inicial estado (el planificador)
  2. identificación ramificaciones que se pueden realizar en paralelo. (POP).
  3. Descomponer el plan en tales ramas-subplanes (ramificaciones)
  4. Insertar acciones de sincronización entre los subplanes
  5. Asignar subplanes a los agentes (asignacion (estatica o dinamica) a los agentes)
  6. Iniciar el plan de ejecución y seguimiento de los avances

# Planificación Distribuida para Planes Centralizados

Formular planes requiere colaboración de especialistas

1. Distribuir la tarea de construir planes entre los especialistas
2. Mezclar los planes obtenidos en paralelos verificando la coherencia entre ellos

Debe haber una coordinación centralizada para unir los planes parciales

- CADA AGENTE HACE SU PLAN PARCIAL
- EL COORDINADOR FUSIONA LOS PLANES (RELACION ENTRE LAS ACCIONES)

# Planificación Distribuida para Planes Distribuidos

- La más desafiante en planificación es cuando se distribuye tanto el proceso de planificación como sus resultados.
- Podría ser innecesario tener un plan total en cualquier parte del sistema, y sin embargo, las partes del plan deben ser coherentes en un mínimo **no hay conflictos entre sí cuando se ejecutan los planes**
- Pero además, se pueden ayudar para alcanzar sus Planes, unos a otros cuando sea racional

# Planificación Distribuida para Planes Distribuidos

- Caso General
  1. Asignar objetivos a cada agente
  2. C/u formula planes locales
  3. C/u ejecuta sus planes
- Problemas
  - Resolución de Conflictos: negociación
  - Suposiciones de tiempo de ejecución de tareas

# Planificación Distribuida para Planes Distribuidos

- Otro Algoritmo:
  1. Intercambiar entre agentes descripción de los planes
  2. Aprobar planes sin conflictos (si todos lo cumplen, resuelto el problema)
  3. Refinar planes con conflictos (cambios en los planes locales para eliminarlos)
  4. Ejecutar localmente planes sin conflicto

# Planificación Distribuida para Planes Distribuidos

## Fusión del Plan

- múltiples agentes formulan planes para individuos, y deben asegurar que sus planes se pueden ejecutar sin conflictos.
- El reto es identificar y resolver los posibles conflictos. Hay varios enfoques:
  - **Coordinación plan centralizado**: un agente recoge planes individuales y los analiza para descubrir secuencias de acciones que podrían dar conflictos, y para modificarlos para eliminar los conflictos.
  - **Qué se vayan sincronizando durante la ejecución**, tal que se determina si el siguiente paso es seguro (no genera conflicto), siguiendo las ideas de estados seguros aplicados en evitar bloqueos de los procesos

# Planificación Distribuida para Planes Distribuidos

## Plan de Formación iterativo

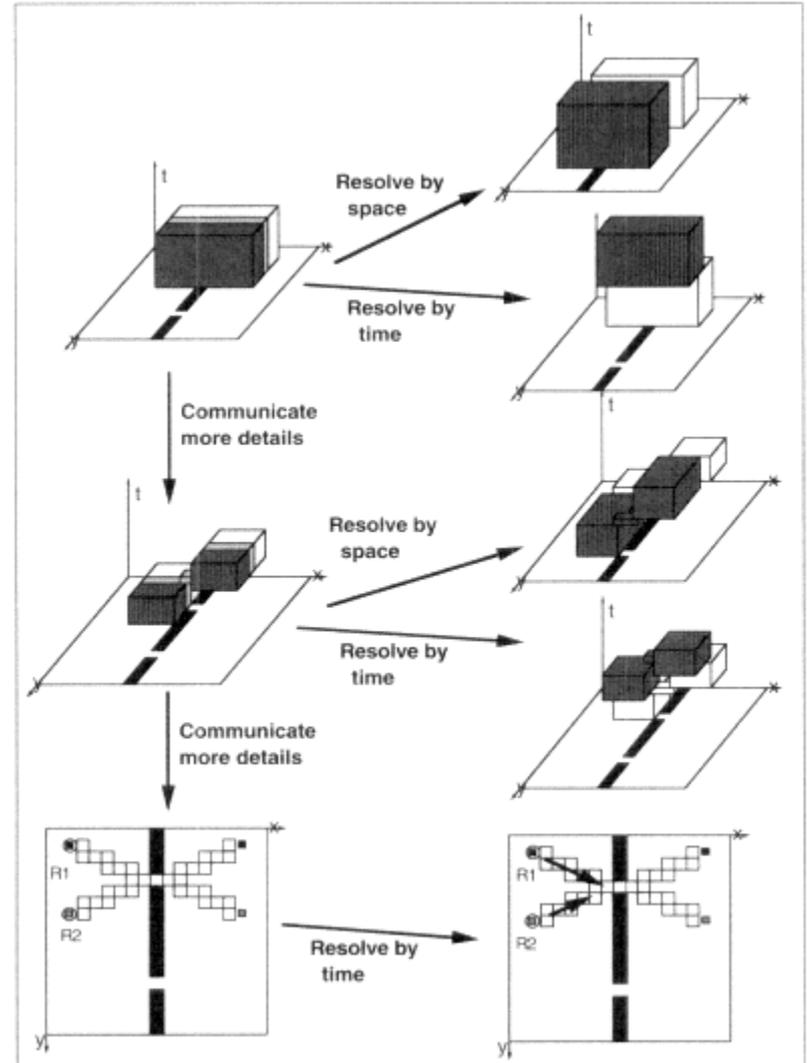
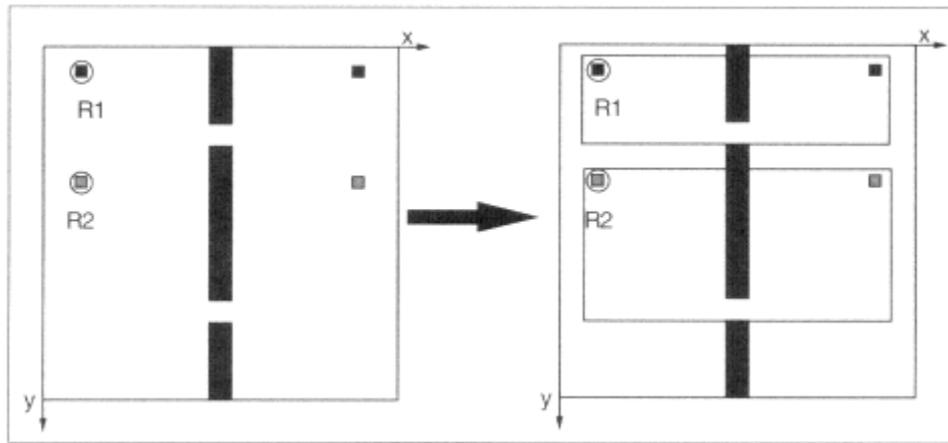
- Es una fusión técnica de gran alcance para el Incremento de paralelismo en el proceso de planificación, así como durante la ejecución.
- Los algoritmos de sincronización y programación pueden llevarse a cabo centralizada y descentralizada
- El flujo es generalmente
  1. Asignar objetivos a los agentes;
  2. Agentes locales formulen planes;
  3. Se cambian y combinan los planes locales;
  4. Se imponen compromisos para resolver interacciones negativas.

# Planificación Distribuida para Planes Distribuidos

## Planificación distribuida jerárquica

- Agente representa cada uno de sus planes locales en múltiples niveles de abstracción, para resolver todos los conflictos por cada nivel.
- El bucle externo del protocolo identifica un nivel de abstracción especial para trabajar, y si los conflictos deben resolverse a este nivel o se pasa a los niveles más detallados.
- El bucle interno del protocolo lleva a cabo lo que se puede considerar como una búsqueda de satisfacción de restricciones distribuido para resolver los conflictos.
  1. Inicialice el nivel de abstracción actual
  2. Agentes intercambian descripciones del plan y los objetivos de interés en el nivel actual
  3. Retire los planes sin conflictos. Si el conjunto está vacío, entonces listo; De lo contrario, determinar si se debe resolver los conflictos en el nivel actual o en un nivel más profundo.
  4. Si se resuelven los conflictos en un nivel más profundo, especifique el nivel profundo y establezca los planes/objetivos que interesan. Vaya al paso 2.
  5. Si no, se procede a resolver los conflictos en este nivel

# Planificación Distribuida para Planes Distribuidos



# Planificación Distribuida y Ejecución

**La planificación distribuida no ocurre en un vacío. El plan distribuido debe ser ejecutado.**

- La complejidad de coordinar planes
- *Coordinación Post-Planificación*: El enfoque de planificación distribuida permite que los agentes planifiquen, pero luego debe hacerse la coordinación de la ejecución, y puede ser:
  - *Planificación de contingencia*. Cada agente que hizo plan responde a las contingencias que se pueden surgir en tiempo de ejecución. Esto implica planes más grandes con ramas condicionales coordinadas. El proceso de coordinación es complicado por las diversas combinaciones de hilos de ejecución del plan.
  - *Un segunda forma es a través de la supervisión y la replanificación*: Un agente supervisa planes de ejecución y si hay una desviación se detiene el progreso de todos los agentes, y se regresa al ciclo de los planes donde la coordinación debe rehacerse.

# Planificación Distribuida y Ejecución

## Coordinación Pre-Planificación

- Antes de que un agente de planificación comience, el agente se coordinará con los demás (para definir restricciones a cumplir).
- Un ejemplo es usando leyes sociales. (Una ley social es una prohibición de acciones en contextos particulares).
- Estas Leyes se pueden derivar trabajando desde estados indeseables y encontrando combinaciones de esos estados, para imponer las restricciones sobre acciones de manera de evitarlos (para prevenir)