



Universidad Popular Autónoma del Estado de Puebla  
Centro Interdisciplinario de Posgrados  
Investigación y Consultoría  
Departamento de Ingeniería Industrial y  
Tecnologías de Información  
Doctorado en Tecnologías de Información y Análisis de  
Decisiones

## UN MODELO DE METAPLANEACIÓN BASADO EN MEMORIA ORGANIZACIONAL

Tesis que para obtener el Grado de Doctor en Tecnologías de  
Información y Análisis de Decisiones

Presenta  
Bernardo Parra

Puebla, México.

Agosto 2007



Universidad Popular Autónoma del Estado de Puebla

Centro Interdisciplinario de Posgrados

Investigación y Consultoría

Departamento de Ingeniería Industrial y

Tecnologías de Información

Doctorado en Tecnologías de Información y Análisis de  
Decisiones

Se aprueba la Tesis:

Doctorado en Tecnologías de Información y Análisis de  
Decisiones

Comité Doctoral

Rúbrica

---

Raúl Morales Carrasco

Director de Tesis

Rúbrica

---

Vittorio Zanella Palacios

Asesor

Rúbrica

---

Luis Alejandro Fabre Bandini

Asesor

Puebla, México.

agosto de 2007

## AGRADECIMIENTOS

Me es muy grato agradecer a muchas personas que me han apoyado en la terminación de mis estudios.

Doy gracias a Dios por permitirme llegar hasta aquí.

A mis padres con total cariño que al igual que todos los padres de estudiantes siempre dan su mayor esfuerzo para ver a sus hijos concluir sus estudios. Con todo cariño agradezco a mi familia, esposa y mis hijos Malú, Eli, Cesar y José que me han brindado su apoyo incondicional.

Asimismo agradezco a mi director de tesis Raúl Morales y mis asesores Vittorio Zanella y Luis Alejandro Fabre por su trabajo. A mi jefe Laboral, Efrén A. Osorio. A los que ayudaron en la realización del trabajo: A Alberto Hernández por la programación del prototipo.

También doy gracias a mis detractores porque impulsaron la demostración de que es posible realizar este modelo de toma de decisiones.

# Abstract

## Metaplaning Model Based in an Organizational Memory

This thesis describes the research to develop a method of Metaplaning to obtain solutions for critical problems in any organization. First, it is necessary to build an organizational memory where the experts register good problem solutions in a specific domain. In this way, when a new problem is presented, the model searches the desired goals in the organizational memory data base and compares them to the past actions registered as good solutions based on experience. If an action has been registered by the majority of the experts then this action will be part of the solution.

This point of view is equivalent to a vote in a meeting to obtain agreement for a consensual solution.

The metaplaning model was implemented with Visual FoxPro in a simple domain, to facilitate the understanding of the proposed solution.

We did this because in an organization we don't have the availability of confidential information from enterprises. However, it is easy to extrapolate the model behavior to any domain.

The two main areas that back the project position up are the following: Planners with prestablished actions and it's what supports half of the model. The other area is Decisions aided by technology applied to decisions supported systems. We experimented with a model of (GDSS) Group Decisions Support Systems using typified interventions.

The objectives that we state are completeness, avoids forgetfulness, secret solution methods, mutual influence in meetings. Include bias in the solution and add high pondering actions.

At the end we made a comparison of the results obtained with two different tools:

The solution in the agents presence and the solution with the registered experiences.

The metrics used let us know:

- i) How many goals of the proposed problem were found in the group of experiences.
- ii) The percentage of actions which were included in the solution.
- iii) The more it gets closer to the majority number it represents that there was a consensus in the actions found,
- iv) How much the agents influence the solution; the closer it gets to the number representing majority of experiences it tells us that the solution is structured by actions of all the agents.

The benefits are: i) A prototype which is portable to any computer with VisualFoxPro, ii) A model which reduce expert's participations when we search a solution, iii) The model helps decision making of transcendent problems, and iv) A new formal method of mixture by majority

# Resumen

## Un Modelo de Metaplaneación basado en una Memoria Organizacional

El presente trabajo describe la investigación para desarrollar un método para obtener soluciones de problemas críticos en cualquier organización. Primero es necesario construir una memoria organizacional donde los expertos registren soluciones correctas de problemas en un dominio específico. De esta forma cuando un nuevo problema se presente, el modelo busca las metas deseadas en la base de datos de la memoria organizacional y las compara con las acciones pasadas registradas como soluciones adecuadas basadas en experiencias. Si una acción ha sido registrada por la mayoría de los expertos, entonces esta acción será parte de la solución.

Este punto de vista es equivalente a votar en una reunión para obtener acuerdo en una solución condensada.

El modelo de metaplaneación fue implementado con el lenguaje Visual FoxPro en un dominio sencillo para facilitar el entendimiento de la solución propuesta.

Se hizo lo anterior debido a que en una organización no tenemos la disponibilidad de información confidencial. Sin embargo, es fácil extrapolar el comportamiento del modelo a cualquier dominio.

Las dos áreas principales que respaldan la posición del proyecto son las siguientes: Planeadores con acciones preestablecidas y Decisiones asistidas por tecnología aplicadas para sistemas de soporte de decisiones. Nosotros experimentamos con un modelo de Sistemas de Soporte de Decisiones Grupales, usando intervenciones tipificadas.

Los objetivos que establecimos para el proyecto son: conseguir completez, evitar olvidos, evitar métodos secretos de solución, influencia mutua en reuniones. Asimismo se buscó incluir sesgos en la solución y agregar acciones con alta ponderación.

Al final hicimos una comparación de resultados obtenidos con dos herramientas diferentes: La solución con presencia de agentes y la solución con las experiencias registradas

Las métricas usadas nos permitieron saber:

- i) Cuántas metas del problema propuesto fueron encontradas con mayoría en el grupo de experiencias,
- ii) Qué porcentaje de acciones fueron incluidas en la solución,
- iii) Qué tanto se acercó el número de acciones de la solución a la mayoría que representa el consenso,
- iv) Cuántos de los agentes influenciaron la solución a través de sus acciones seleccionadas.

Los beneficios son: i) Un prototipo portable a cualquier computadora con Visual Fox Pro, ii) Un modelo que reduce las participaciones de los expertos cuando se busca una solución, iii) Ayuda en la toma de decisiones de problemas trascendentes y iv) Un nuevo método formal de mezcla por mayoría.

## INDICE GENERAL

<b>Cap. 1 Introducción . . . . .</b>	<b>1</b>
1.1 Planteamiento del problema . . . . .	1
1.2 Propósito de la investigación . . . . .	2
1.3 Objetivo de la investigación . . . . .	3
1.4 Preguntas de Investigación . . . . .	5
1.5 Justificación de la investigación . . . . .	5
1.6 Alcances y limitaciones . . . . .	5
1.7 Viabilidad de la investigación . . . . .	6
1.8 Impacto esperado. . . . .	6
1.9 Organización del Estudio. . . . .	7
<b>Cap. 2 Marco Teórico . . . . .</b>	<b>8</b>
2.1 Modelos relacionados a la planeación . . . . .	8
2.2 Modelos de toma de decisiones . . . . .	11
2.3 Modelos de toma de decisiones asistidas por tecnología . . . . .	13
2.4 Modelos de Sistemas Expertos . . . . .	15
<b>Cap. 3 Diseño Conceptual del prototipo . . . . .</b>	<b>17</b>
3.1 Modelo de la investigación . . . . .	17
3.2 Descripción de términos usados. . . . .	18
3.2.1 Paradigmas generales . . . . .	21
3.3 Área de oportunidad . . . . .	22
3.4 Definición de equipos de trabajo para solución de problemas . . . . .	23
3.5 Modelo de metaplaneador propuesto . . . . .	24

3.6	Descripción del algoritmo de metaplaneación (MeCePCA)	28
3.7	Descripción del contexto del modelo	29
3.8	Modelo conceptual.	29
3.9	Descripción formal de operadores mezclables	37
3.9.1	Concepto de operadores mezclables	37
3.9.2	Ejemplo	39
4.1	Fundamentos del prototipo	42
4.2	Casos de uso	43
4.3	Clases candidatas	45
4.4	Formularios	46
4.5	Pruebas	48
4.6	Métricas	50
4.7	Resultados y discusión	50
4.8	Aportaciones	51
4.9	Líneas de investigación futuras	52
4.10	Análisis de cumplimiento de los objetivos propuestos	52

## INDICE DE FIGURAS

Figura 1. Ubicación de MeCePCA . . . . .	25
Figura 2. Mapeo resultante de aplicar el plan $\pi$ . . . . .	27
Figura 3. Condición para la ejecución del plan $\pi$ . . . . .	27
Figura 4. Modificación del estado del mundo por el agente $\alpha$ . . . . .	28
Figura 5. Descripción del contexto del modelo . . . . .	30
Figura 6. Grupos de experiencias . . . . .	30
Figura 7. Conjunto de metas en cada experiencia. . . . .	30
Figura 8. Grupo de acciones en cada meta. . . . .	31
Figura 9. Grupo de experiencias con metas y acciones. . . . .	31
Figura 10. Esquema de comparaciones entre acciones. . . . .	32
Figura 11. Ejemplo numérico del modelo . . . . .	32
Figura 12. Comparación de acciones y plantilla para caso 0 . . . . .	33
Figura 13. Comparación de acciones y plantilla para caso 1. . . . .	34
Figura 14. Comparación de acciones y plantilla para caso 2. . . . .	34
Figura 15. Comparación de acciones y plantilla para caso 3. . . . .	34
Figura 16. Comparación de acciones y plantilla para caso 4 . . . . .	35
Figura 17. Mayoría relativa. . . . .	35
Figura 18. Tipos de comparaciones promedio. . . . .	36
Figura 19. Modelo de composición de operadores mezclables . . . . .	38
Figura 20. Casos de uso del Metaplaneador . . . . .	43
Figura 21. Diagrama de clases candidatas. . . . .	45
Figura 22. Formulario para captura de datos de los agentes expertos . . . . .	46
Figura 23. Formulario para captura de acciones. . . . .	46
Figura 24. Muestra de la solución . . . . .	47
Figura 25. Tiempos de ejecución para 16 pruebas con 33 recetas agrupadas acumulativamente . . . . .	49
Figura 26. Comparación de soluciones . . . . .	51

## INDICE DE TABLAS

Tabla 1. Tipos de Modelos de Decisión Asistidos por Tecnología . . . . .	13
Tabla 2. Definición de operador . . . . .	39
Tabla 3. Formato de Solución . . . . .	48



# 1 Introducción

---

## **Resumen**

En este capítulo se presenta un panorama general de las etapas iniciales del proyecto y se da el punto focal de la investigación junto con los conceptos que se encuentra alrededor de este punto. Se da un panorama general del contenido y que se desarrolla más adelante en la presente tesis. También de manera importante se menciona lo que se espera del trabajo de investigación. Los puntos tratados son: Planteamiento del problema o de la oportunidad de investigación, Propósito de la investigación, Objetivo de la investigación, Preguntas de Investigación, Justificación de la investigación, Alcances y limitaciones, Viabilidad de la investigación, Impacto esperado y también la Organización de

## **1.1 Planteamiento del problema**

Los planificadores computacionales han sido extendidos para actuar en ambientes multi-agente, donde ellos pueden razonar acerca de los planes e intenciones de otros agentes. Por ejemplo en el trabajo de Parsons (Parsons et al 98) se propone un marco basado en un sistema de argumentación que permite a los agentes negociar para establecer formas aceptables de resolver problemas. Una técnica reciente conocida como Metaplaneación permite planear acerca de cómo conformar los planes. De manera más acertada podemos decir que:

Metaplaneación es una técnica que puede usarse para generar ideas en un dominio con los métodos de solución de problemas que contribuyan a mejorar la manera de hacer los planes. En un contexto más amplio podría usarse para generar ideas para casi cualquier situación. Puede verse como una forma de lluvia de ideas estructurada.

Un ejemplo de cómo trabaja es el siguiente: Una idea o proposición se le da a un grupo, éste genera sus propias ideas, las cuales se intercalan y se concentran. Las ideas con más de una igualdad en un conjunto, se examinan con más detalle. De tal manera que por proceso de eliminación, una solución a un problema puede encontrarse que tiene la mayoría de aceptación del grupo.

Uno de los proyectos más avanzados en Metaplaneación es el de AI Center, SRI International (Wilkins 01), en donde en el futuro piensan integrar mezcla de planes. Actualmente intentan explorar algoritmos para contar con planeación distribuida.

La problemática existente en el ámbito de solución de problemas de planeación se plantea de la siguiente manera:

*No se ha diseñado a la fecha un modelo que, permita mezclar el conocimiento registrado de varias personas expertas para plantear la solución de problemas de planeación.*

Por lo anterior expuesto creemos que es factible proponer un modelo que resuelva algunas de las carencias existentes y que pueda ayudar a las personas que enfrentan problemas en equipos o procesos fundamentales en sus organizaciones.

## **1.2 Propósito de la investigación**

Analizando modelos como los arriba descritos, así como el modelo de Veloso de planeación jerárquica, el de Tsamardinos y A-Teams de Talukdar (Talukdar et al 98), aquí se propone el Modelo de Metaplaneador mediante Células de Planeación por Composición de Acciones (MeCePCA), que satisface un área de oportunidad en mezcla de planes.

MeCePCA tiene como finalidad crear un modelo de Metaplaneación basado en una memoria organizacional donde se registran las soluciones a todos los problemas de un dominio dado, que permita proporcionar el plan de solución para problemas en equipos o procesos estratégicos de cualquier organización

El propósito que se busca en este proyecto de investigación es **Diseñar un modelo que permita mezclar el conocimiento registrado de varias personas expertas para plantear la solución de problemas de planeación.**

Describiendo un concepto básico de planeación manejado en el modelo, diremos que en MeCePCA los planes están conformados por metas y las metas por acciones, llamados también operadores.

## **1.3 Objetivo de la investigación**

El Objetivo principal que se plantea para la presente tesis es:

***Definir un modelo de Metaplaneador, formado por la composición de acciones de experiencias de agentes.***

*El modelo aquí propuesto permitirá aprovechar las experiencias retenidas en una Memoria Organizacional de los expertos de un dominio para realizar una composición de las acciones de cada meta de las experiencias. La composición comprende el análisis para decidir, si una acción es más adecuada porque la mayoría de los agentes realizó esa acción en el logro de las mismas metas. Asimismo el criterio de negociación utilizado puede decidir entre dos o más acciones iguales. De igual manera dicho criterio afecta el orden de secuencia de las acciones en la solución obtenida.*

Para alcanzar el objetivo principal, se plantean los siguientes objetivos específicos:

## **Objetivos Específicos**

### ***I Conseguir completez para cada meta del plan de solución***

La composición de acciones tiene como alcance dar refinamiento en el logro de una meta. El refinamiento da un mayor grado de completez a la consecución de cada meta debido a que detalla qué acciones harán que la realización de una meta no pase por alto aquellas acciones que pueden ser críticas, relevantes o complementarias. La completez en este modelo incorpora todas las acciones que son iguales en el conjunto de acciones de cada meta en las experiencias de los agentes y asegura que para cada acción se hicieron las comparaciones con todas las acciones de metas iguales y se obtuvo mayoría en dichas acciones (consenso).

### ***II Que el modelo propuesto contenga la facilidad de manejar Soluciones con sesgos***

Para poder manejar un rango de soluciones deseadas se requiere analizar las implicaciones que tiene el uso de sesgos en la forma de realizar las acciones (argumentos).

Las soluciones planteadas por el modelo, descritas a través de un plan pueden ser analizadas por los responsables de ejecutar los planes, quienes pueden adecuar la solución para no incluir determinados argumentos. Las observaciones serán alimentadas como sesgos a una o más acciones. Los sesgos implican no tomar en cuenta determinados tipos de argumentos en acciones de las metas.

### ***III Implementar un prototipo que permita evaluar la completez en la conformación de planes.***

Un prototipo del modelo es presentado para evaluar los resultados que se pueden obtener con ejemplos de datos ficticios y reales. Se tiene la necesidad de desarrollar el prototipo porque no existe un modelo similar; aunque es factible aprovechar los sistemas libres en internet que permiten la comunicación entre agentes, esta parte no es la más importante en el modelo sino la forma de integrar las acciones de cada meta en la solución. Por lo anterior se optó por desarrollar un prototipo que simula la comunicación entre agentes pero que garantiza la composición de acciones tomando en cuenta todas las experiencias de los agentes.

### ***IV Presentar y discutir aplicaciones reales***

Para describir el funcionamiento del prototipo del modelo propuesto se explicará mediante un ejemplo de datos donde se maneja información de experiencias. Algunas áreas de la industria buscan tener una manera de aprovechar la experiencia acumulada a través de su Memoria Organizacional en diferentes áreas en donde requieren planear soluciones firmes que no

busquen reducir costos, sino asegurar funcionamiento permanente. Por ejemplo en el mantenimiento de equipos estratégicos.

## ***V Comparar el modelo propuesto con otro modelo que plantee***

### ***soluciones***

Se analizará un modelo que se asemeja al modelo propuesto en el tipo de trabajo que realiza para plantear soluciones a problemas en forma de planes. Esto permitirá ubicar y contrastar el alcance porque las soluciones deben ser comparadas no sólo en el plan obtenido sino en la manera de alcanzarlo. Así se podrán medir características como tiempo de obtención, dispersión de participantes que conforman las acciones en la solución, entre otros.

## ***VI Presentar y discutir posibles líneas de investigación que pueden surgir a partir de este modelo.***

El futuro de los sistemas multiagentes para planeación bajo esta línea de investigación, parece ser la colaboración con agentes en empresas del mismo ramo. La colaboración entre agentes ya existe pero no bajo el esquema de compartir experiencias a nivel de composición de acciones en problemas de planeación de soluciones. Se plantean como posibles líneas el uso de composición de planes mediante reducciones previas en las experiencias de un mismo agente, para llegar a un proceso que únicamente contenga planes previamente filtrados y se reduzca el conjunto de metas a analizar.

## **1.4 Preguntas de Investigación**

Algunas de las preguntas que han fundamentado la realización del presente trabajo, surgen de un trabajo previo de groupware realizado en el Centro de Investigación en Computación (CIC.), las cuales se presentan en seguida:

- ¿Es posible generar soluciones con los mejores posibles participantes en un dominio?
- ¿Se podrá juntar el conocimiento de los expertos de un dominio, tanto actuales, pasados o de otros lugares?
- ¿Las soluciones de un problema en discusión se pueden tomar por las mejores?
- ¿Las soluciones garantizan que no existen expertos celosos que se hayan guardado conocimiento como de su propiedad?
- ¿De dónde se puede obtener el conocimiento de un dominio?
- ¿Las soluciones pueden excluir tipos de acciones indeseadas?

## **1.5 Justificación de la investigación**

Las suposiciones de conocimiento perfecto son una idealización del contexto de la planeación que tiende a simplificar los procesos de planeación. Los procesos de planeación permiten el desarrollo de algoritmos de planeación que han demostrado propiedades tales como completez y correctez. Desafortunadamente existen pocos dominios en los cuales puedan ser reales la mayoría de las veces el mundo es impredecible. Dependiendo de las suposiciones de conocimiento perfecto en un mundo impredecible puede probarse el costo de efectividad si la incertidumbre del planeador es pequeña acerca del dominio o si el costo de recuperarse de una falla es bajo.

Los resultados que se puedan obtener darán, de acuerdo a lo esperado, amplias posibilidades a los usuarios planeadores de plantear soluciones sin incurrir en gastos de soluciones con un fuerte respaldo de los expertos.

## **1.6 Alcances y limitaciones**

Los alcances del proyecto están determinados por la información dada como experiencias de un dominio en la Memoria Organizacional. Es decir, la Memoria Organizacional tendrá la calidad de las soluciones que el usuario pueda alimentar correctamente. Asimismo cada nueva experiencia puede limitar o engrandecer el contenido de soluciones para nuevos problemas planteados en el dominio.

El agregar una experiencia influye al dar mayoría a las metas o a las acciones que se busquen para una nueva solución. De esta forma cada nueva experiencia hará que una nueva solución se vea afectada con la entrada de las últimas experiencias registradas.

## **1.7 Viabilidad de la investigación**

El proyecto es viable debido a que se cuenta con las herramientas necesarias para realizarlo. Se tiene disponible un software de Sistemas de Soporte de Decisiones Grupales que ha permitido generar las pruebas necesarias que fundamentan la mitad del proyecto de investigación. La otra mitad que respalda el proyecto es el estudio de generación y combinación de planes que han desarrollado otros autores para mejorar los planes de solución en cualquier dominio.

El método de validar el funcionamiento del modelo propuesto es:

1. Recolectar datos del dominio de ejemplo
2. Alimentar los datos en el prototipo del modelo
3. Calcular los resultados que se deben producir
4. Ejecutar el prototipo para obtener resultados de planes de solución
5. Comparar ambos resultados
6. Anotar las diferencias, si las hay
7. Repetir los pasos 1 a 6, n veces hasta verificar que los resultados no tienen diferencias.

## **1.8 Impacto esperado**

Las contribuciones del presente proyecto son las siguientes:

- Un modelo de generación de planes de solución de problemas de planeación en un dominio.

Es una lástima que las personas que van adquiriendo experiencia en un área dentro de una organización, porque se vuelven indispensables. En algunas ocasiones se ignora que los conocimientos formales o descubrimientos de los empleados, pertenecen legalmente a la organización. Pero todavía es más trágico que las organizaciones no sepan como aprovechar el conocimiento de las personas. El modelo ayudará a la organización y a los expertos de un dominio.

- Un prototipo de software capaz de instalarse para cualquier organización que lo requiera, sin importar su tamaño.

El prototipo es realizado con el manejador de base de datos Visual Fox Pro 9.0 de Visual Studio, que por la popularidad de software bajo Windows lo hace muy portable y de fácil uso.

- Un modelo que reduce el número de intervenciones de los expertos de un dominio en la solución de problemas.

Por ser un modelo que requiere el registro de experiencias, no es necesaria la presencia de los expertos del dominio para poder generar soluciones inmediatas y eficaces mientras los problemas pueden estar deteniendo el buen funcionamiento de la empresa.

- Un modelo que ayuda en la toma de decisiones de problemas trascendentes.

Generalmente los problemas con más costo para la empresa, son los que detienen su buen funcionamiento. Vale la pena construir una Memoria Organizacional de las experiencias de solución de problemas en el área más influyente dentro de la empresa. De igual manera otra área es la que le puede dar mayor y más rápido crecimiento a la organización, de esta forma no sólo los problemas que detienen el funcionamiento, sino también se plantean como problemas, las decisiones que involucran el crecimiento.

- Un nuevo modelo formal para mezcla de muestras por mayoría

Es posible que no existiera desde antes este método, porque no había una aplicación necesaria para justificar la inclusión de la mezcla de arreglos bajo el criterio de mayoría. Como en este proyecto se justifica su uso, también se analiza formalmente la forma y número de comparaciones necesarias para obtener un arreglo de solución.

## 1.9 Organización del Estudio

La metodología de análisis incluirá investigación documental e investigación de campo.

Estas dos herramientas se usarán para buscar tanto bibliográficamente como a través de internet los diferentes modelos de metaplaneadores que existen.

Se hará una investigación de campo usando el modelo de Toma de decisiones grupales mediante el software creado para tal fin por el autor, que permita plantear problemas reales en los que se busquen planes de solución que se basen en experiencia acumulada.

Las dos herramientas se usarán en virtud de que el problema planteado del proyecto es de tipo Teórico-Práctico

Se empleará el método inductivo, el cual crea leyes a partir de la observación de los hechos, mediante la generalización del comportamiento observado; por medio de la lógica matemática se puede conseguir una demostración de las citadas leyes o conjunto de conclusiones.

Dichas conclusiones no podrían ser falsas pero si podrían tener rasgos de incompletez, debido a que las experiencias no generen mayoría en las comparaciones, por eso, el método inductivo necesita una condición adicional, su aplicación se considera válida *mientras no se encuentre ningún caso que no cumpla el modelo propuesto*

Una vez realizado el análisis de las comparaciones necesarias en la búsqueda de mayoría, es necesario crear el software del modelo planteado. En seguida se podrá comparar el funcionamiento del software con las soluciones obtenidas de manera presencial mediante el software de Toma de decisiones grupales creado por el autor.

Se requiere una vez probado el funcionamiento que se haga una prueba estadística para medir el comportamiento en la generación de soluciones.

## 2 Marco Teórico

---

### **Resumen**

Este capítulo muestra los modelos de planeación tradicional más cercanos al propuesto, también los modelos de decisión asistidos por tecnología que apoyan el desarrollo del proyecto y finalmente se describen los sistemas expertos porque su funcionamiento tiene una estructura parecida al modelo que más adelante se plantea. Aunque los modelos de planeación tradicional no se asemejan, si tratan temas que todo modelo de planeación debe incluir, tales como la definición de metas, operadores o acciones, recursos, precedencia y consecuencia, entre otros. Los modelos de decisión asistida por tecnología se describen en su estructura y su utilización como apoyo en las organizaciones. Y los sistemas expertos tienen una forma de trabajo que se asemeja al modelo propuesto.

### **2.1 Modelos relacionados a la planeación**

Este trabajo trata de un método de metaplaneación mediante la experiencia de agentes, por lo que se manejarán temas relacionados como son: planeación computacional, metaplaneación, planeadores más recientes y justificación del modelo propuesto. Primeramente se describe la evolución de los planeadores con la finalidad de observar como se fueron desarrollándose nuevos modelos que mejoraban los anteriores.

Evolutivamente se tiene primero GPS (General Problem Solving) (Fikes & Nilsson 71) que consiste fundamentalmente en descomponer de manera sistemática un problema en otros. GPS requiere describir el estado del sistema, lo que obliga a describir los objetos y sus relaciones. Lo anterior puede requerir una enorme cantidad de descripciones de estado. Esto hizo surgir la versión STRIPS (Stanford Research Institute Problem Solver) de GPS en donde un estado se representa por fórmulas lógicas de primer orden que constituyen los axiomas del estado; sin embargo algunas veces no se podía llegar a soluciones porque la resolución de un objetivo parcial daba lugar a la destrucción de información necesaria para resolver objetivos subsiguientes. Una versión mejorada, RSTRIPS, en algunos casos considera la posibilidad de generación de un nuevo estado por aplicación de operadores a pesar de que el estado actual sirviera ya para aplicar la acción que se pretendía.

Por otra parte ABSTRIPS y NOAH (Sacerdoti 98) hicieron la resolución de problemas complejos con operadores elementales lo cual condujo a la generación de planificadores jerarquizados que resolvieran el plan operando a distintos niveles de definición de objetivos.

Asimismo SIPE (Wilkins 01) y algunos otros modelos concibieron sistemas jerarquizados e interactivos que permitieran mezclar la concepción automática con la intervención humana.



Las aplicaciones recientes han ilustrado el poder de la planeación, en áreas como Cooperación en sistemas Multi-Agentes, donde se tienen trabajos como:

- Descubrimiento y explotación de sinergia entre agentes de planeación jerárquica (Cox & Durfee 02).
- Búsqueda de un plan alternativo (Felner et al 02).
- Programación de agentes en Dribble: en la cual se pasa de creencias a metas usando planes (Riemsdijk et al 02).

Los planificadores computacionales han sido extendidos para actuar en ambientes multi-agente, donde ellos pueden razonar acerca de los planes e intenciones de otros agentes. Por ejemplo en el trabajo de Parsons (Parsons et al 98) se propone un marco basado en un sistema de argumentación que permite a los agentes negociar para establecer formas aceptables de resolver problemas. Una técnica reciente conocida como Metaplaneación permite planear acerca de cómo conformar los planes

Actualmente se cuenta con modelos tales como Planeación Jerárquica (Veloso y Carbonell 1994), Planes compartidos (Grosz & Krauz 1996) y Refinamiento en Planeación (Kambhampati 1997) que realizan la composición de acciones basados en el enfoque reduccionista de tomar en cuenta las acciones que tengan asociado un menor costo. Otro enfoque de mezcla de planes es dado por Yang (Yang 97), este último mejorado por Cox y Durfee (Cox y Durfee 03), en los cuales la mezcla se hace resolviendo problemas de coordinación de acciones de los multiagentes y mejorando el rendimiento al compartir acciones entre agentes; sin embargo no se ha tomado en cuenta que:

- Tener el menor costo de un conjunto de acciones no asegura que se tomen en cuenta las acciones más adecuadas para lograr la meta, ya que algunas veces el menor costo incluye operaciones que no garantizan que sean confiables.
- Cuando se hace un análisis de acciones en algún modelo como Tsemardinos (Tsemardinos et al 00) se lleva a cabo una unión uno a uno buscando que esté libre de conflictos; sin embargo no toman en cuenta que no es lo mismo unir planes que analizar cada acción mediante el enfoque individual de cada agente
- La solución no puede ser única, el uso de sesgos nos ayuda a que la solución no tome en cuenta determinados contextos de acciones; o bien las restricciones apoyan las soluciones que cuentan con valores acotados en el manejo de recursos.

Algunos otros modelos de planeación fuertemente relacionados son:

- Planeadores Lineales no-Jerárquicos. Buscan una secuencia de acciones para alcanzar una meta
- Planeadores Oportunistas. Usan un control de pizarrón
- Planeadores Basados en casos. Dependen de un plan esqueleto que siempre tienen disponible.

Una vez descritos algunos de los modelos representativos más conectados con el modelo propuesto, se describirán en seguida otra parte fundamental que da respaldo al modelo propuesto.

## 2.2 Modelos de toma de decisiones

Ahora se describirá la importancia que tienen los modelos de decisión en las empresas, para justificar como influyen en el funcionamiento de las organizaciones.

Hoy en día la existencia de las organizaciones depende de dos elementos importantes: su infraestructura y de su soporte tecnológico, por ello es importante decir que:

El cómputo organizacional está relacionado con crear nuevas clases de infraestructura tecnológica, estudiando las relaciones entre la infraestructura tecnológica y la infraestructura organizacional. La parte de cómputo organizacional incluye Sistemas de Soporte de Decisiones (SSD) para apoyar a múltiples personas involucradas en la toma de decisiones (en contraste con los SSD para individuos que toman decisiones). Las características de Cómputo Organizacional vienen de:

i) La Tecnología Groupware, ii) Tecnología de trabajo cooperativo soportado por computadora, iii) Tecnología de comunicación mediada por computadora y iv) Tecnología de coordinación.

Estos modelos de apoyo dan fortaleza a la organización. Ampliando esta información, los tipos de SSDs identificados que actualmente usan las corporaciones son: Sistemas de planeación corporativa, Sistemas de soporte de decisiones funcionales, Sistemas de información ejecutiva y Sistemas de soporte de decisiones locales.

Por otra parte, los sistemas que hoy en día dan apoyo a las organizaciones que se actualizan tecnológicamente y organizacionalmente, incluyen cuatro tipos de SSD:

- 1.- Sistemas de planeación corporativa. Incluyen algunos conceptos como:
  - Manejo de conocimiento relevante para planear decisiones de altos directivos,
  - Decisiones que afectan a departamentos y divisiones,
  - Decisiones que afectan todas las funciones de la organización,
  - Diseño de planeadores corporativos que responden a preguntas “Que pasa si” (“what if” en inglés), entre otros
- 2.- Sistemas de Soporte de Decisiones funcionales. Comprenden ideas como:
  - Manejo de conocimiento relevante para decisiones a cerca de alguna función de rendimiento en la organización desplegando el efectivo disponible,
  - Presupuesto en una campaña de publicidad, entre algunas que se pueden mencionar.
- 3.- Sistemas de Información ejecutiva. Agrupan temas como:
  - Información relevante para decisiones hechas por ejecutivos de alto nivel,
  - Soporte de decisiones inmediatas y a corto plazo para ejecutivos,
  - Satisface adecuadamente las necesidades de información a cerca de rendimiento actual y actividades anticipadas, por mencionar algunos temas.
- 4.- Sistemas de Soporte de Decisiones Locales. Conjunta conocimientos de:
  - Información construida y mantenida por usuarios finales,
  - Se asocia con funciones más que con sistemas complejos.

Por otro lado los modelos de Sistemas de Soporte de Decisiones Multiparticipante (SSDM), se pueden resumir como Sistemas de Soporte de Decisiones: Grupales, Organizacionales y en Equipo; estos últimos todavía en investigación. De manera más corta a los SSDM se los llaman Sistemas de Soporte de Negociaciones. Dentro de los sistemas de Soporte de Negociaciones en forma reciente se han incluido los sistemas de agentes inteligentes, los cuales tienen un amplio campo de aplicaciones y ahora definimos su alcance:

Sistemas de Agentes Inteligentes.- están constituidos por software que emplea algunos mecanismos de IA para llevar a cabo un cierto conjunto de tareas. Por ejemplo aquellos que hacen tareas como: ser asistentes personales, realizar tareas complejas, tareas en lugar del usuario, monitoreo de eventos, notificación al usuario a cerca de situaciones; además pueden los agentes inteligentes: funcionar como participantes en un grupo, y ayudar en la colaboración de los participantes

## 2.3 Modelos de toma de decisiones asistidas por tecnología

En la taxonomía de las decisiones asistidas por tecnología se cuenta con varios modelos como son:

Tipo de Tomador de decisiones Multiparticipante	Facilidades y hardware	Software	Consideraciones Específicas
Sala de Pizarrón Electrónico	Sala de conferencias con capacidades de proyección audiovisual en pantalla controlada por computadora	Aplicación para almacenamiento y recuperación de presentaciones	Interacción sincrónica en el mismo tiempo y lugar. Requiere la presencia de un técnico
Sala de Teleconferencias	Sala de conferencias con transmisión audiovisual controlada por computadora entre localidades	Aplicación para controlar transmisión digital de audio, video y datos	Interacción mismo lugar tiempo diferente. Requiere la presencia de un técnico de teleconferencias
Red grupal	Separa las facilidades de oficina conectadas via una red de computadoras	Aplicación para permitir conferencias ya sea en tiempo real o desktop asincrónico	Mismo o diferente tiempo diferente lugar en interacción con un participante sirviendo como coordinador o chair
Centro de Información	Sala de conferencias con video proyector. Computadoras individuales como terminales	Aplicaciones para manejo de Base de datos, análisis estadístico, generación de gráficas y procesador de textos	Interacción en el mismo tiempo y lugar. Requiere especialistas en modelado y software de aplicación específica
Laboratorio de Colaboración	Sala de conferencias con pizarrón electrónico y computadoras en red	Aplicaciones para interacción colaborativa e intercambio de información	Mismo o diferente tiempo, mismo lugar de interacción. Requiere la presencia de un facilitador de procesos MDM
Sala de Decisión	Sala de conferencias con video proyector y computadoras en red	Aplicación para soportar lluvia de ideas, comentarios, votación, modelado, análisis de decisiones, interacción colaborativa, interacción e intercambio de datos	Mismo o diferente tiempo, mismo lugar de interacción. Requiere la presencia de un facilitador de procesos MDM
Chat Tipificado	Computadoras en red	Aplicación para actualización de datos por internet. Invitar a expertos para la solución de un problema, Menú para: Proponer una solución, dar aportaciones a favor o en contra, intercambio de información, votar. Se analiza la solución propuesta y su factibilidad	Mismo tiempo, diferente lugar. Requiere la presencia virtual de un facilitador de procesos MDM

Tabla 1. Tipos de Modelos de Decisión Asistidos por Tecnología

Fuente: Adaptado de ACM Computing Surveys 20, No 2 “Computer-Based Systems for cooperative Work and Group Decision Making” by Kraemer, K. F. and King, J.L.

En la tabla 1, se muestran los tipos de modelos de decisiones asistidos por Tecnología. En el último renglón, el cual no pertenece a la clasificación de los autores, es un modelo creado por el autor del presente trabajo para modelar los tipos de interacción que permiten obtener soluciones en reuniones virtuales. Este último modelo

respalda el planteamiento de planes de solución por mayoría o por consenso mostrado más adelante en el modelo de investigación.

Las clases de groupware se pueden observar en la tabla 1, dentro de estas clases, un modelo que es parecido al planteado, es el de Autoría Colaborativa, el cual permite al grupo la creación y revisión de un documento. Este documento se compone de secciones y en este proceso cualquier participante puede ver y comentar a cerca de cualquier sección en cualquier momento, pero sólo uno puede modificar una sección a la vez. En este modelo el documento describe una decisión colectiva.

Los Sistemas de Soporte de Decisiones Grupales (SSDG) incluyen asistencia para: remover barreras de comunicación, facilitar la estructura de flujo de deliberaciones, genera ideas, organiza, analiza, categoriza alternativas, despliega resultados colectivos, permite a los participantes estar en diferentes lugares, muestra a cada participante el estado de las acciones propias y de otros, publica alertas y recordatorios etc. (Marakas 06)

Las características de Groupware que es la herramienta que ayuda a los SSDG en su desarrollo, son las siguientes:

Es un software diseñado para soportar colaboración, incluyendo captura y almacenamiento de información intercambiada. Los líderes en el mercado actual son: Lotus Notes and Domino, Microsoft Exchange, Novell GroupWise y Oracle Office.

Las herramientas individuales dentro del software completo incluyen un administrador de reuniones (Lotus Sametime) e intercambio de mensajes (Lotus Notes Mail).

Algunas de los resultados obtenidos en el desarrollo de una sesión son:

- Incremento de productividad
- Reducción del número de reuniones
- Incremento en la automatización del flujo de trabajo (workflow) de rutina
- Necesidad de una mejor coordinación global
- Disponibilidad de extender las redes de trabajo

Dentro de los métodos más comunes de coordinación en actividades de toma de decisiones grupales tenemos: i) Técnica de Grupo Nominal, ii) Técnica Delphi, iii) Arbitraje, iv) Sistema de información basado en publicación y v) Nemawashi

### **Técnica de Grupo Nominal**

- Cada participante escribe ideas a cerca de cual debe ser la solución
- A su vez cada participante presenta sus ideas, las cuales se registran en un pizarrón. Aquí no ocurre discusión.
- Después de que todas las ideas se presentan, los participantes pueden cuestionar a los demás
- Cada participante vota por cada idea

### **Técnica Delphi**

- Esencialmente es la misma técnica de grupo nominal, excepto que los participantes nunca se reúnen
- Un instrumento de medición se usa para recolectar la entrada inicial de cada miembro
- Una segunda revisión es enviada con un resumen de los resultados colectivos. Estos pasos se repiten hasta que se alcance mayoría o consenso

### **Arbitraje**

- Esta técnica es la más apropiada cuando los miembros de la reunión representan factores opuestos
- Los participantes acuerdan que si las alternativas no son mutuamente aceptadas, un arbitro externo se deberá incluir en la reunión
- El arbitro selecciona entonces la alternativa que el cree más apropiada

### **Sistema de Información basado en resultados (SIBR)**

Es un método de argumentación estructurado

- Un SIBR se representa como una gráfica con nodos y ligas
- El SIBR comienza con la selección de un nodo raíz, entonces los nodos-posición son ligados a la raíz
- Los nodos posición son entonces evaluados con los argumentos unidos a ellos

### **Nemawashi**

- Uno o más miembros de la reunión son designados como coordinadores. Los coordinadores seleccionan entonces a los participantes restantes.
- Los coordinadores construyen un conjunto de opciones y los expertos evalúan las opciones
- El coordinador selecciona una opción basado en resultados del paso anterior
- La alternativa es difundida y el coordinador busca consenso a través de persuasión y negociación.
- Si se alcanza consenso, los coordinadores circulan un documento que cada miembro firma

## **2.4 Modelos de Sistemas Expertos**

La estructura básica de un Sistema Experto (SE) está en relación con la estructura genérica de un SSD. La base de conocimientos, la cual es específica para un dominio de problema particular asociada con el SE, es un componente distinto para los algoritmos de solución de problemas que puede ser empleado a través de diferentes dominios o tareas. La principal diferencia entre un SSD y un SE está en el contenido del conocimiento adquirido de los expertos en el dominio de aplicación. Esto último constituye la Base de Conocimiento. El conocimiento consiste de todos los tipos de conocimiento usado por el experto del dominio durante el curso de la solución de problemas relacionados en un dominio: objetos, descripciones y relaciones, comportamientos durante la solución, restricciones, heurísticas e incertidumbres. El éxito de un SE está relacionado con la completez y exactitud de su base de conocimientos.

El SE usa en su comportamiento de solución la Máquina de Inferencia (MI). La MI realiza deducciones o inferencias basada en reglas o hechos. Además es capaz de realizar razonamiento inexacto o difuso basado en probabilidad o empate de patrones.

El proceso básico del SE es llamado Ciclo de Control. Tres pasos se identifican en el ciclo de control de inferencia: i) Empate de reglas con hechos dados ii) Seleccionar la regla que va a ser ejecutada y iii) Ejecutar la regla.

El Espacio de trabajo de Pizarrón, es el área de trabajo de un SE; es el lugar donde se realizan cálculos, donde se almacenan hipótesis intermedias y se realizan decisiones durante el proceso de solución del problema. A través del Sistema de

Pizarrón el SE puede comunicar soluciones candidatas o requerimientos adicionales de entrada. Una vez que el sistema ha empleado su trabajo, el área de pizarrón es borrada y queda lista para la próxima sesión de solución de problemas.

Los SE aportan beneficios en la solución de problemas como los siguientes:

- Decrementan el tiempo de toma de decisiones
- Incrementan la productividad de los expertos organizacionales
- Mejoran la consistencia en las decisiones
- Mejoran el entendimiento y explicaciones
- Mejoran el tratamiento de la incertidumbre
- Formalizan el conocimiento organizacional

Pero los SE también tienen asociadas limitaciones como son:

- El conocimiento necesario no está siempre disponible
- Los expertos usan sentido común. El sentido común en Programación no es todavía una realidad
- La experiencia es difícil de extraer y codificar
- Los expertos pueden reconocer un problema que este fuera del dominio de conocimiento mucho más rápido que un SE
- Los SE no pueden eliminar las limitaciones cognitivas del usuario
- Un SE es funcional solamente funcional en un dominio pequeño de conocimiento
- El vocabulario experto es a menudo limitado y no es fácil de entender por otros
- Los expertos humanos se adaptan a su medio ambiente de manera natural, mientras un SE debe ser explícitamente actualizado
- Un SE tiene una experiencia limitada comparada a los expertos humanos.

Los SE tienen ahora herramientas de desarrollo de aplicaciones construidas de acuerdo al dominio, por ejemplo: EMYCIN (para diagnosis médica), de igual modo han sido construidos muchos ejemplos de “shells” en otros dominios.

Resumiendo las áreas de conocimiento expuestas, podemos decir que los Modelos de Planeación, la Toma de Decisiones Asistida por Tecnología y Los Sistemas Expertos forman parte de un área de la Inteligencia artificial que busca optimizar el proceso de planteamiento de soluciones (toma de decisiones) de problemas en un dominio a través de diferentes tecnologías. Las tecnologías van desarrollándose y ahora describimos como Marco teórico estas tecnologías para plantear más adelante el modelo que involucra las tecnologías hasta ahora descritas.

### 3 Diseño conceptual del prototipo

---

#### **Resumen**

En este capítulo se analiza la ubicación del modelo propuesto, revisando los puntos de vista de modelos presenciales así como modelos que generan u obtienen planes a partir de registros anteriores de otros planes o acciones. Se ubica el área de surgimiento del modelo propuesto y se comparan las características de modelos existentes con el modelo propuesto.

#### **3.1 Modelo de la investigación**

El enfoque de la investigación a realizar es cualitativa; es decir, se usará una metodología de análisis que incluirá investigación documental e investigación de campo. Estas dos herramientas se usarán para buscar tanto bibliográficamente como a través de internet los diferentes modelos de metaplaneadores que existen así como las áreas en que esta ubicado el proyecto

Se hará una investigación de campo que permita plantear problemas reales en los que se busquen planes de solución que se basen en experiencia acumulada.

Las dos herramientas (investigación documental y de campo) se usarán en virtud de que el problema planteado del proyecto es de tipo Teórico-Práctico. Se empleará el método inductivo, el cual crea leyes a partir de la observación de los hechos, mediante la generalización del comportamiento observado; en realidad, lo que realiza es una generalización, sin que por medio de la lógica pueda conseguir una **demonstración** de las citadas leyes o conjunto de conclusiones.

Dichas conclusiones podrían ser falsas y, al mismo tiempo, la aplicación parcial efectuada de la lógica podría mantener su validez; por eso, el método inductivo necesita una condición adicional, su aplicación se considera válida *mientras no se encuentre ningún caso que no cumpla el modelo propuesto*.

El método de validar el funcionamiento del modelo propuesto es:

1. Recolectar datos del dominio de ejemplo
2. Alimentar los datos en el prototipo del modelo
3. Calcular los resultados que se deben producir
4. Ejecutar el prototipo para obtener resultados de planes de solución
5. Comparar ambos resultados
6. Anotar las diferencias, si las hay
7. Repetir los pasos 1 a 6, n veces hasta verificar que los resultados no tienen diferencias o se obtiene consistencia

Los resultados esperados serán:

- El planteamiento de un modelo que sea considerado en la solución de problemas de un dominio.
- Planes de solución de problemas a partir de un banco de datos de experiencias de otros problemas solucionados en un dominio.
- Crear un software que permita ayudar a los tomadores de decisiones en problemas críticos y/o estratégicos dentro de su organización

### 3.2 Descripción de términos usados

Para comprender el ámbito de los modelos de Planeación y Metaplaneación se describe en seguida un glosario y además se plantean los paradigmas generales que existen en la planeación.

**Abstracción:** Una forma de reducir la complejidad de planeación es hacer caso omiso de las características no esenciales.

**Complejidad:** La complejidad de la planeación está relacionada al número de planes viables en el espacio-plan. Los planes alternativos se pueden crear por operadores no ordenados u operadores más específicos o por operadores equivalentes. Se ha desarrollado un número de enfoques para tratar con la complejidad descrita. En general esos métodos se enfocan en dividir o restringir la búsqueda de tal forma que el problema llega a ser tratable.

**Completez de planes:** Una composición de operadores para plantear la solución de un plan puede ser una tarea ardua; por lo que para simplificarla se asumen en planeación lo siguiente:

- . existe solamente un operador para realizar cada actividad
- . los operadores seleccionados son los óptimos (menores o mayores costos menores o mayores cantidades de tiempo empleado, o bien mayores cantidades de estados aportados).
- . el orden de los operadores es el mejor de acuerdo a un criterio compuesto
- . los modelos del mundo están limitados a problemas de planeación
- . los planes son secuenciales
- . las acciones son concurrentes (no paralelas)

**Enfoque de compromiso mínimo.-** Es un enfoque constructivo en el cual no se hacen compromisos a menos que sea necesario hacerlos. Las decisiones son hechas solamente cuando el planeador puede determinar que dichas decisiones no interferirán con decisiones pasadas o futuras. La meta es evitar el retroceso (backtracking) tanto como sea posible. Este enfoque es usado en planeadores no-lineales (por ejemplo NOAH, MOLGEN y SNLP).

**Estados:** Son las condiciones del modelo del mundo.

**Interacción.-** Este término describe cómo las submetas pueden interferir unas con otras.

Existen tres clases de interacción:

- Interacción débil.- Las metas deben ser alcanzadas en un cierto orden para no interferir unas con otras.



- **Interacción fuerte.-** No hay un orden en las submetas, las cuales alcanzarán la meta, pero después ellas son ejecutadas en el mismo orden.
- **Interacción muy fuerte.-** No se utiliza ningún orden. Ejemplo existen dos cuartos, derecho e izquierdo. El cuarto derecho contiene un buzón y el izquierdo contiene un robot con una llave. La meta es poner la llave en el buzón y cerrar la puerta con candado del cuarto izquierdo desde dentro. Si la llave está puesta en el buzón, la puerta no puede ser cerrada; si la puerta esta cerrada, el robot no puede alcanzar el buzón

**Meta.-** Comúnmente especificado por un estado o un conjunto de estados del mundo. Existen tres clases de metas: Las que mantienen alguna condición, las que previenen de alguna condición de ocurrencia y las de secuencia de actividades

**Modelo del Mundo:** Es una serie de condiciones almacenados en una lista. Así como las condiciones cambian en el mundo, condiciones en el modelo del mundo se agregarán o removerán de la lista.

**Modelo de Solución (plan):** Los operadores a aplicar pueden actualizar el modelo del mundo. La solución formará una secuencia lineal de estados sobre el eje del tiempo.

**Operadores:** Para describir acciones se usan los operadores. Un operador tiene un conjunto de precondiciones, las cuales deben existir en el modelo del mundo antes de que una acción pueda realizarse. Si las precondiciones de una acción son satisfechas, los resultados de una acción son simulados por agregar y borrar condiciones en el modelo del mundo. Nótese que un operador puede tener como resultado, violar las precondiciones de otros operadores.

**Operador STRIPS.-** Un operador STRIPS consiste de un nombre de operador junto con una lista de prerequisites (o lista de precondiciones), una lista de adiciones y una lista de borrado. Los elementos de esas listas son todas expresiones de proposición.

**Planeación.-** Decide sobre un curso de acción antes de actuar. Un plan es una representación de un curso de acción. Un plan terminado es una secuencia lineal o parcialmente ordenada de operadores. La planeación es una técnica de resolver problemas. Planear es razonar a cerca de eventos futuros con el fin de verificar la existencia de una serie razonable de acciones a tomar con la finalidad de alcanzar una meta. Los mayores beneficios de la planeación son: la reducción de búsquedas, el resolver conflictos entre las metas y poner las bases para solucionar los errores.

**Problema en Planeación:** Un problema se describe como un conjunto de metas que necesitan alcanzar una solución aceptable.

**Plan jerárquico.-** Es un plan que describe como tomar acciones (operadores) en niveles de incremento de refinamiento y especificidad (por ejemplo: "Hacer la tarea x" es más específico que "ir a trabajar", "Hacer trabajo" o "Ir a casa"). Muchos planes son jerárquicos por naturaleza

**Planeación Jerárquica.-** Planeación que usa una jerarquía de abstracciones de un plan para resolver el problema. El plan base es la forma que lista las operaciones ejecutables; los niveles que están por arriba del plan base se incrementan en abstracción y

simplicidad. La meta de los planeadores jerárquicos es simplificar la búsqueda y procesos de razonamiento al encontrar soluciones vagas en niveles donde los detalles no son computacionalmente abrumadores y entonces refinarlas. Existen dos formas para conformar jerarquías: abstracción de planes por medio del uso de conceptos de alto nivel como operadores (por ejemplo en NOAH, NONLIN y SIPE): abstracción de espacio conforma clases de equivalencia de estados al ignorar precondiciones menos críticas y submetas

**Planeación No-Jerárquica.-** Consiste en encontrar una secuencia de operadores para alcanzar cada una de las metas. Comúnmente los planeadores no distinguen entre metas importantes y las menos críticas; de tal forma que invierten mucho tiempo en encontrar soluciones para partes no críticas de un plan; algunos ejemplos incluyen STRIPS, HACKER y WARPLAN.

**Planeación No-Lineal.-** Compromiso mínimo con respecto al tiempo. Si el resultado del plan es no-lineal, se puede usar linearización para un agente o ejecución para múltiples agentes. Las decisiones de orden son solamente realizadas cuando es necesario. Los planes parciales son ordenamientos parciales en operadores, los cuales son dejados en desorden hasta que se detecta un conflicto, en ese punto se agregan restricciones. Como ejemplos de planeadores no-lineales, tenemos NOAH (Sacerdotti 77), NONLIN (Tate 77), DEVISER (Vere 83), SIPE (Wilkins 84) TWEAK (Chapman 84).

**Planeación Oportuna.-** Planeación en tiempo real en la cual los operadores son invocados cuando son útiles al estado actual. Analizado en el sistema OPM de Hayes-Roth.

**Problema de independencia.-** Cuando se incluyen múltiples agentes, no se quiere especificar todos los eventos externos que deben ocurrir, ni se quiere una solución que asuma que cualquier cosa no especificada no sucederá.

**Reflexión.-** Es la diferencia entre actuar y pensar acerca de actuar. Planeación es la reflexión a cerca de actuar; Meta-Planeación es reflexionar a cerca de la planeación.

**Robustez.-**Un plan robusto es aquel que puede tratar con cambios inesperados que pueden alcanzarse durante la ejecución; los planes robustos tratan de evitar el tomar cursos de acción que ofrezcan pocas alternativas si fallan (Por ejemplo la programación de conexiones apretadas de vuelos).

**Representación de metas:** En todo momento el modelo del mundo contiene un conjunto de condiciones. Cuando algunas metas se introducen, ellas deben describir el estado del modelo del mundo. Por lo tanto, las metas son descritas con una expresión conjuntiva de todas las condiciones deseadas en el modelo del mundo.

**Suposición de linealidad.-** Todas las submetas son independientes una de otra y entonces pueden ser alcanzadas en un orden arbitrario.

**Suposición de linealidad fuerte.(transitividad)-** No siempre un plan puede ser totalmente ordenado, si un operador  $S$  alcanza la meta  $G$  y el operador  $T$  alcanza la meta

$H$  y  $S$  está antes que  $T$ , entonces todos los pasos que alcancen precondiciones de  $S$  están todos antes que todos los pasos que alcanzan las precondiciones de  $T$ .

**Solución de problemas.**- Es el proceso de desarrollar una secuencia de acciones para alcanzar una meta. Aunque este es un concepto muy general, abarca todas las metas de programas de Inteligencia Artificial (IA).

Después de ver un glosario de términos usados en el ambiente de la planeación y la metaplaneación, se describe a continuación los paradigmas que existen en este ambiente

### 3.2.1 Paradigmas generales

Todos los modelos de planeación requieren hacer búsquedas para ello, existen dos formas básicas de ver a la planeación como búsqueda. En la primera forma la planeación es una operación de búsqueda de un espacio de estados del mundo en el cual las transiciones de un estado a otro son acciones que el agente puede realizar. En la segunda forma, el espacio consiste de planes parciales y cada nodo es un plan terminado parcialmente. Las transiciones en este espacio son realizadas al agregar acciones a los planes.

Existen sistemas los cuales razonan a cerca de los recursos consumidos como parte de la ejecución de un plan o sistemas que reconocen que la actividad de planear por si misma tiene un costo.

Una forma de manejar la fluidez de recursos (tales como dinero o agua por ejemplo) en planeación es usar métricas valuadas numéricamente. Una métrica es una cantidad de algo, tal como dinero o volumen.

Los planeadores que usan métricas, usualmente requieren que sean declaradas inicialmente de acuerdo con los límites del rango. Algunas medidas son controladas por el medio ambiente (tal como el precio del gas) mientras que otras son recursos que pueden ser producidos y consumidos.

Muchos planeadores clásicos hacen las siguientes suposiciones:

- Las acciones de los agentes cambian el mundo de un estado estático a otro.
- Los efectos de las acciones son predecibles.
- El mundo cambia solamente como resultado de las acciones de los agentes.

Cabe hacer notar que estas suposiciones se dan de alguna forma en sistemas de planeación recientes.

Existen cinco enfoques básicos para planeación:

- No jerárquica
- Jerárquica
- Basada en casos
- Oportuna
- Planeación de orden parcial

#### Planeadores Lineales No-Jerárquicos

Los primeros sistemas de planeación fueron no-jerárquicos; esto significa que no hicieron distinción entre las partes importantes y las no importantes del plan, de tal forma que podrían estar atascados tratando con detalles no importantes.

#### **Planeadores Lineales Jerárquicos**

Los planeadores jerárquicos crean planes vagos y refinan los detalles incrementalmente hasta que el plan construido está completamente especificado.

#### **Planeadores Basados en Casos**

Los sistemas de planeación basados en casos dependen de planes tipo esqueleto que están siempre disponibles. La premisa es que se requiere la experiencia para formular nuevas clases de planes y depurar los viejos planes. Se generan muy pocos planes completamente nuevos, pero existen muchas nuevas instanciaciones de los tipos de planes. La planeación consiste en usar y depurar planes existentes y almacenar los resultados de nuevas creaciones de planes útiles. Este tipo de planeación se puede resumir como:

Casos + transformaciones = planes + estrategias de refinamiento.

#### **Planeación Oportuna**

Este tipo de planeación tiene un enfoque basado en la planeación humana. Este enfoque usa una estructura de control de pizarrón en el cual las sugerencias son hechas por especialistas para los pasos de los planes. El orden se desarrolla poniendo las piezas de grupos concretos de operadores. La planeación oportuna construye superestructuras en el espacio del plan e intenta conectarlas, mientras los planeadores jerárquicos tratan de desarrollar un plan completo a la vez e intentan refinarlo.

#### **Planeación de Orden Parcial (POP)**

La POP nos habilita para tomar ventaja de la descomposición del problema. El algoritmo de planeación trabaja en diferentes submetas de manera independiente y las resuelve con diferentes subplanes. Al final combina los subplanes (Russell 03). Además, se tiene la ventaja de la flexibilidad en el orden en el cual se construye el plan. Esto permite al planeador trabajar primero en decisiones importantes o sencillas, en lugar de estar forzado a trabajar en los pasos (acciones) del orden cronológico.

### **3.3 Área de oportunidad**

El ámbito del tipo de problemas que se solucionan mediante planeadores está en las organizaciones que deben tomar decisiones a cerca de problemas inesperados relacionados con: equipos, inversiones, problemas de transporte, programación de producción, entre los principales tipos.

Este tipo de problemas se solucionan de manera intuitiva, basados en el conocimiento de expertos. Algunas veces las soluciones se llevan varias horas de

análisis o días, cuando hablamos de modelos presenciales. Como apoyo se han creado algunos modelos como los mencionados en la introducción del presente trabajo.

La idea principal gira en torno a crear un modelo que sea práctico en el sentido de que aprovecha la experiencia acumulada por los expertos en una organización y que sea capaz de mezclar las soluciones anteriores mediante un algoritmo novedoso que se probado y que algunas organizaciones están esperando todavía desarrollar.

Una vez planteados las características de solución de problemas por algunos métodos más conocidos es importante describir la otra vertiente que existe en la solución de problemas como lo son las soluciones en forma presencial. Esta última forma tiene algunas variantes como se describe desde la integración de grupos que participan en el planteamiento de soluciones hasta las soluciones influenciadas por la preparación (background) de los que intervinieron.

### **3.4 Definición de equipos de trabajo para solución de problemas**

El modelo planteado en el presente trabajo se basa en el trabajo en equipo para la solución de problemas. Algunos de los resultados efectivos que se buscan en el trabajo en equipo se logran a través de los principios siguientes:

- Obtener participación de todos los miembros del grupo
- Obtener información del grupo
- Obtener una solución democrática del grupo ante un problema

(Andueza 97)

Un requisito inobjetable es crear las mejores condiciones para la integración del grupo, como son:

- Ambiente adecuado
- Reducción de la intimidación
- Liderazgo distribuido
- Formulación del objetivo
- Flexibilidad
- Consenso
- Comprensión del proceso
- Evaluación continua

(Gibb 84)

Sin embargo el grupo toma determinada dirección movido por una serie de fuerzas complejas, a veces difíciles de precisar: energía, cambios, reacciones, actividades, retrocesos, aciertos, equivocaciones, etc.

Para poder conseguir un resultado adecuado se requiere que quien coordine este tipo de actividades sea experto en manejar equipos de trabajo y conocer el área de planeación. Por otra parte para tener una solución que minimice la cantidad de riesgos, es mejor buscar el consenso. El consenso se basa en la creencia de que cada persona tiene parte de la verdad y que nadie tiene toda la verdad

Un grupo multidisciplinario no puede ser el más apropiado para desarrollar consensos, sino grupos de profesionales con experiencias o habilidades similares. (Gellerman 81).

La forma directa de obtener soluciones acertadas a problemas inesperados es evitar los conflictos que se presentan en la conformación de grupos de trabajo y aprovechar la enorme experiencia acumulada en la solución de problemas anteriores en la organización.

Hasta ahora, lo anterior se ha logrado mediante la creación de Memorias Organizacionales que guardan la pista de solución de problemas presentados con anterioridad. Esto generó un amplio campo de acción en la teoría de solución de problemas como es el Razonamiento Basado en Casos (RBC). Este enfoque busca el caso más cercano al problema nuevo y lo modifica para adaptarlo como modelo de solución. El RBC deja de lado el proceso de tomar en cuenta todos los componentes de algún método que pueden ser mejores para una parte de la solución del problema.

En la figura 9 se observa la ubicación del modelo propuesto, basado en los dos modelos principales que son: los presenciales y los basados en experiencia escrita.

### **3.5 Modelo propuesto de metaplaneador**

De manera directa el presente trabajo es parte de los modelos que plantean solucionar problemas mediante la composición de acciones de todas las experiencias registradas en una memoria organizacional. Esto es obtenido a partir de la manera en que se trabaja en Sistemas de Soporte de Decisiones Grupales (GDSS en inglés).

Para generar soluciones en este campo, el modelo Metaplaneador mediante Células de Planeación por Composición de Acciones (MeCePCA) trabaja con las experiencias de agentes, teniendo como limitaciones y requisitos: contar con una memoria organizacional y que los problemas a solucionar se planteen en términos de metas que se desean lograr. Asimismo se determina un criterio de trabajo para los agentes como es seleccionar las acciones con mayores o menores costos durante el proceso de integración de la solución. Esto constituye la base de negociación entre agentes para seleccionar las acciones más adecuadas cuando se encuentran acciones iguales o equivalentes.

En el modelo de metaplaneador Cassandra (Pryor & Collins 96) y algunos modelos más no han considerado algunas características como:

- El problema de determinar si su modelo sirve para una salida particular (MeCePCA si, mediante el uso de un grupo de experiencias predeterminado).
- No es un planeador probabilístico; es decir no hace uso de cualquier información de probabilidad o de cualquier otro evento que mejore o le de ponderación a las acciones resultado (MeCePCA maneja sesgos e integración de acciones con alta ponderación.)
- Se ignora la posibilidad de mezclar la planeación y la ejecución. (MeCePCA también porque es un metaplaneador "Off-line").
- No maneja eventos exógenos (MeCePCA si porque puede agregar cambios de estado en las metas y volver a ejecutarse).

De manera similar existen otros modelos que manejan otras formas de generar planes como se observa en la figura 1.

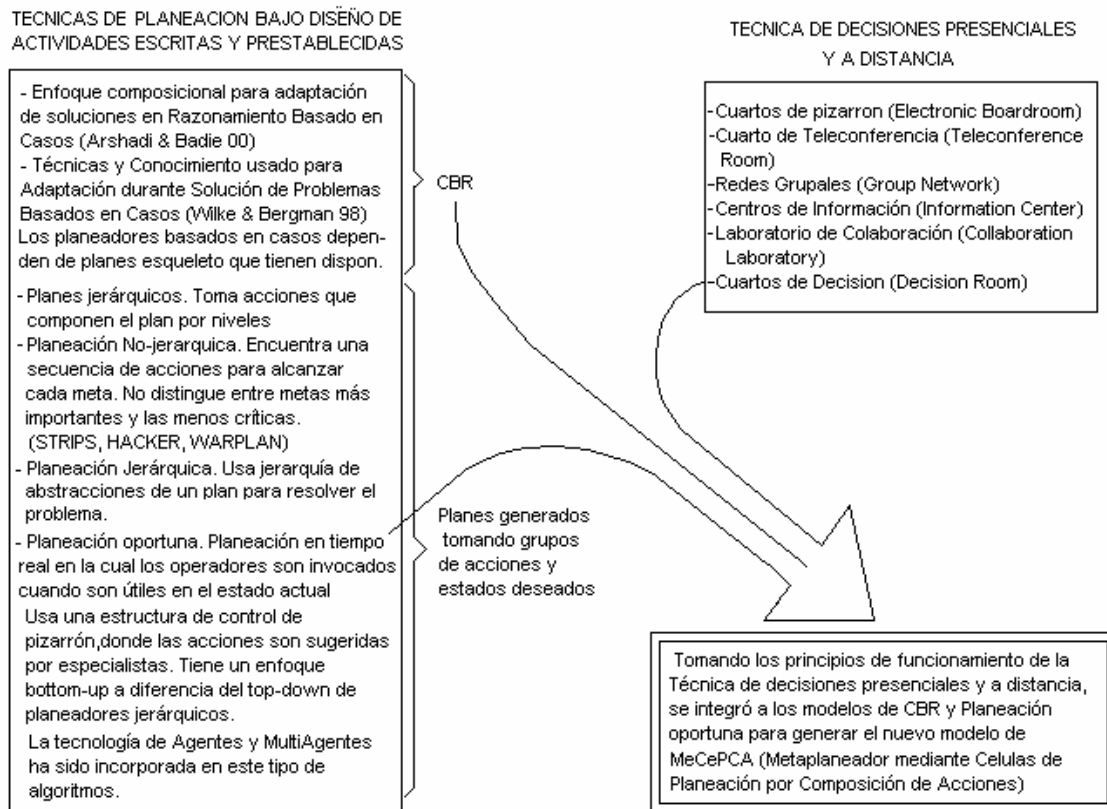


Figura 1. Ubicación del Modelo Propuesto

También se observa en la figura 1, que MeCePCA se genera de uno de los modelos presenciales que se puede describir como un “Chat tipificado” porque las intervenciones de los participantes están en forma distribuida y la forma de identificarse es a través de sobrenombres o “nicks” para no reconocer a las personas. Por otro lado el tipo de intervenciones no es abierto sino seleccionado de un menú de tipos de intervenciones y bajo ciertas reglas de participación.

Lo anterior dio pie a tratar de evitar todos los problemas de intervenciones personales. Para facilitar el acceso al conocimiento se les da a los expertos una herramienta para registrar las distintas soluciones lineales que han realizado en los planes para resolver problemas, como más adelante se describirá.

Antes de explicar con detalle el modelo es conveniente indicar que se obtuvo de manera colateral un nuevo modelo de mezcla de arreglos por mayoría el cual no estaba registrado en el libro donde se resumen los modelos de ordenamientos, mezclas y búsquedas (Knuth 72).

Todos los Metaplaneadores buscan en forma inherente encontrar a partir de su base de conocimientos (la memoria organizacional) todas y las mejores soluciones obtenidas de la composición de acciones. Por esto, el algoritmo a usar debe garantizar

que no se están dejando de lado algunas soluciones posibles; esta factibilidad se elimina asegurando que el modelo tiene la característica de Completez .

. La completez así como su consistencia dependen de la suposición de conocimiento perfecto. La completez en general depende de tres extensiones para esta suposición:

- Todas las fuentes de incertidumbre se especifican (a través de los estados alcanzados);
- Las salidas especificadas son exhaustivas (por la comparación combinatorial de todas las acciones de una meta);
- Existen acciones disponibles que permiten la determinación de la resolución de cualquier incertidumbre (bajo el algoritmo cualquier acción que tenga mayoría elimina la posibilidad de ser excluida)

Estas condiciones son necesarias pero no suficientes. El modelo propuesto puede solamente conformar planes si las acciones que contienen las experiencias de los agentes usados para determinar la contingencia, no interfieren con el alcance de la meta.

Formalmente una acción que se compara con otras acciones para buscar agregarla a la solución, la llamamos operador mezclable  $\mu$ . En la composición de operadores del plan solución  $\mu$  debe ser comparado con:

- Las acciones que puedan ser iguales con las acciones en las metas de otros agentes, denominadas IG.
- Los argumentos, denominado AR y
- El conjunto de sesgos, denominado como SE

En la comparación con las demás acciones del conjunto de metas  $m$  de las experiencias de los agentes se espera que la búsqueda se realice cumpliendo que las acciones sean iguales en su descripción y argumentos y que no pertenezcan a los sesgos (acciones con argumentos no deseados).

En la búsqueda de operadores que sean iguales, se selecciona solamente un operador mediante un criterio compuesto. Esta comparación debe encontrar el operador más adecuado que cumpla con IG, AR,  $\neg$  SE, es decir

$$\forall m(\forall \mu \in \Sigma, \exists \mu_{may} \{ \mu \in (IG, AR, \neg SE) \}) \quad (4.1)$$

Donde  $\mu_{may}$  es el operador que tiene mayoría en el proceso de búsqueda, es decir, se encontró al menos  $N/2 + 1$  veces en el conjunto de acciones de las metas de otros agentes.

Cuando se realizan las comparaciones, se espera que no hayan acciones que no sean comparadas, mientras no se obtenga mayoría. Este proceso exhaustivo se le llama completez, denominada CMPLTZ.

La completez se define aquí como una función de la composición denominada como  $C$ , de operadores  $op$ , en las metas de todas las experiencias de los agentes susceptibles de integrar parte de la solución. Formalmente:

$$CMPLTZ = \{ \{ C_{i=1}^{i=n} (op) \} \in (IG, AR, \neg SE) \} \quad (4.2)$$

La completez es la selección de la combinación de todos los operadores mezclables en las metas de los agentes expertos que cumplan la selección de operador más adecuado descrita en (4.1).



Es importante hacer notar que las experiencias de los agentes contienen planes y que esos planes necesariamente pueden relacionarse por las secuencias que muestran al querer alcanzar estados de meta definidos previamente.

## El Mundo de un agente simple

Formalmente se definen las operaciones realizadas por un simple agente en un mundo.

Sea  $S$  un conjunto de estados del mundo y  $\Pi$  un conjunto finito de planes. La ejecución de un plan  $\pi$  cambia el estado del mundo el cual es modelado como un mapeo definido parcialmente denominado *res*, formalmente se tiene:

$$res: \Pi \times S \rightarrow S$$

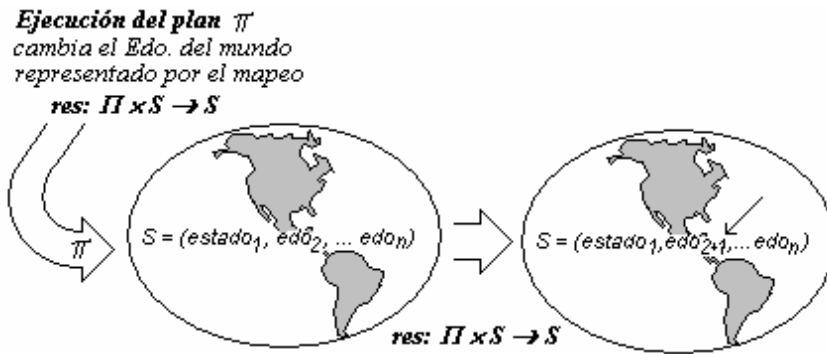


Figura 2. Mapeo resultante de aplicar el plan  $\pi$

Un plan es *ejecutable* en  $S$ , si y solo si  $\pi$  está definido para un cierto estado del mundo  $S$ , hecho que expresamos formalmente por el predicado  $exec(\Pi, S)$



Si  $\pi$  está definido en un estado de  $S$   
entonces  $\pi$  se puede ejecutar

Figura 3. Condición para la ejecución del plan  $\pi$

Un agente  $\alpha$  actúa en el mundo, por lo tanto modifica su estado. Esto está definido por las siguientes nociones:

- un conjunto  $\Pi_\alpha \subset \Pi$ , determinando los *planes individuales* que  $\alpha$  es capaz de ejecutar. Si  $\alpha$  es capaz de ejecutar el plan  $\pi$  en una situación  $s$ , también podemos escribir  $prep_s(\alpha, \Pi)$ ;
- un conjunto  $I_\alpha \subset S$  de estados ideales de  $\alpha$ , expresando los estados que el agente podría idealmente conseguir
- una *métrica*  $C_\alpha$  que mapea dos estados a un número real, representa la estimación del agente  $\alpha$  de qué tan lejos está un estado de los otros. Esto usualmente modela la noción de dificultad para traer los cambios entre los estados del mundo.



Figura 4. Modificación del estado del mundo por el agente  $\alpha$

Aun cuando un agente usualmente no puede alcanzar el estado ideal, los estados ideales  $I_\alpha$ , constituyen su *motivación*.

La distancia entre dos estados  $s_1$  y  $s_2$  está dado para otros modelos como la longitud del plan más corto que transforma  $s_1$  en  $s_2$ . Aquí bajo el criterio de consenso, el cual disminuye los riesgos al incluir sólo acciones que al menos dos agentes las tienen establecidas; se busca que el plan esté dado por lo que la mayoría de los agentes tiene guardado en sus experiencias.

### Un mundo multiagente

Un mundo multiagente está habitado por un conjunto de agentes  $A$ .

Cada  $\alpha \in A$  tiene la estructura arriba definida

En tal mundo multiagente los agentes actúan al mismo tiempo y en el mismo medio ambiente, de tal manera que necesitamos introducir una cantidad de acciones interdependientes. El conjunto de *multiplanes*  $k$ ,  $M_k$  es la unión disjunta del conjunto de planes individuales de los  $k$  agentes:

$$M_k = \Pi_1 \cup \dots \cup \Pi_n$$

Un **multiplan-k**  $\mu \in M_k$  modela la ejecución simultánea de los planes individuales de los agentes involucrados. Usamos el operador conmutativo  $\circ$  para denotar la creación de un multiplan. La función parcial *res* es extendida a multiplanes-k

$$\text{res}: M_k \times S \rightarrow S$$

Un **multiplan-k**  $\mu$  es ejecutable en la situación  $s$  (formalmente:  $\text{exec}(\mu, s)$ ), si y sólo si *res* está definido en  $s$ . Por otra parte algunos de los planes componentes son incompatibles, es decir son físicamente imposibles de ejecutarse simultáneamente.

### Relaciones entre planes:

En el modelo, en una situación  $s$  un plan  $\pi$  puede estar en cuatro relaciones cualitativas mutuamente exclusivas para un **multiplan-k**  $\mu$ :

$$\text{indiferente}(\pi, \mu) \Leftrightarrow (\text{exec}(\pi, s) \wedge \text{exec}(\pi \circ \mu, s) \wedge \text{res}(\pi \circ \mu, s) = \text{res}(\pi, s)) \vee$$

$$(\neg \text{exec}(\pi, s) \wedge \neg \text{exec}(\pi \circ \mu, s))$$

$$\text{interferente}(\pi, \mu) \Leftrightarrow (\text{exec}(\pi, s) \wedge \text{exec}(\pi \circ \mu, s) \wedge \text{res}(\pi \circ \mu, s) \neq \text{res}(\pi, s))$$

$$\text{complementaria}(\pi, \mu) \Leftrightarrow (\neg \text{exec}(\pi, s) \wedge \text{exec}(\pi \circ \mu, s))$$

$$\text{inconsistente}(\pi, \mu) \Leftrightarrow \text{exec}(\pi, s) \wedge \neg \text{exec}(\pi \circ \mu, s)$$

El multiplan  $\mu$  es *indiferente* con respecto a  $\pi$  si la ejecución de  $\mu$  no afecta  $\pi$  en todo.

El multiplan  $\mu$  es *interferente* con  $\pi$  si  $\pi$  es ejecutable sólo en conjunción con  $\mu$ , pero las dos alternativas llevan a estados diferentes del mundo.

El multiplan  $\mu$  es *complementario* con respecto a  $\pi$  cuando  $\pi$  no es ejecutable solo, sino en conjunción con  $\mu$ . La idea es que existe un hueco en el plan  $\pi$ , es decir, alguna acción se olvidó o las precondiciones de alguna acción no son alcanzadas con el plan y  $\mu$  llena el hueco por ejecutar de la acción olvidada o contiene las carencias de los estados del mundo.

Finalmente, el plan  $\mu$  es *incompatible* con  $\pi$  si  $\pi$  es ejecutable solo pero no en conjunción con  $\mu$  (Ossowski y .García-Serrano 97)

Las relaciones entre planes prevalecen al considerar las relaciones entre las acciones de uno o más agentes; es decir dos o más acciones son *indiferentes* si su ejecución no interfiere una con otra. Son *interferentes* si la ejecución de una sólo es realizable en conjunción con la otra pero llevan a estados diferentes. Aquí conviene la mezcla de otra actividad que elimine la interferencia. Las acciones *complementarias* se dan como secuencias naturales en las acciones de cada agente o en la mezcla con otros agentes que ayuden a conseguir los estados deseados, siempre y cuando esos agentes sean mayoría.

Las acciones incompatibles deben incluir a la acción que otorgue más estados a la meta y eliminar a la otra.

### 3.6 Descripción del algoritmo de metaplaneación (MeCePCA)

Suposiciones iniciales:

Se asume que la ontología de un dominio existe previamente (dada por el conjunto de metas, de acciones y parámetros manejados en ese dominio)

1.- Captura de experiencias de agentes, donde se detectan la forma de describir la realización de metas y acciones

2.- Las acciones requieren tener adicionalmente la lista de atributos (precondiciones, poscondiciones estados y argumentos) y también el atributo ponderación que permite diferenciar entre aquellas acciones Necesarias o Complementarias, atributo dado por los agentes. La ponderación de una acción es una valoración que permite posteriormente en la solución saber si fue sugerida (complementaria) o tiene el carácter de necesaria

Se requiere una comparación para eliminar acciones de acuerdo al criterio de negociación, realizándose de la siguiente manera:

Si una acción de ponderación complementaria al compararse con una acción de ponderación necesaria resulta seleccionada (desde luego son acciones iguales) la primera es seleccionada porque contribuye con sus estados al logro de la meta; entonces se convierte en una acción de ponderación necesaria.

3.- Pedir metas del nuevo problema

- 4.- Se le pide al usuario el Criterio de Negociación (búsqueda) que servirá para la comparación de acciones; el cual puede ser, la búsqueda del menor o mayor tiempo o bien, menor o mayor cantidad de recursos.
- 5.- Buscar en las experiencias de los agentes expertos, las metas que sean iguales a las metas del problema a resolver. Se almacenan las metas que hayan sido encontradas.
- 6.- Con el grupo de metas iguales, se buscan las acciones que sean iguales en la misma meta de cada agente
- 7.- Se busca que las acciones iguales en una meta de diferentes agentes tenga mayoría absoluta (la mitad de los agentes mas uno).
- 8.- De cada acción con mayoría absoluta se selecciona la más adecuada por criterio de negociación
- 9.- Si se tuvo mayoría relativa (la acción moda pero sin ser mayoría absoluta) se selecciona la acción más adecuada de acuerdo al criterio de negociación.
- 10.- Los estados que se tienen en cada meta de la solución son acumulados por las acciones que alcanzan dichos estados.
- 11.- Se despliega la solución.
- 12.- Si se desea aplicar sesgos para eliminar acciones, se seleccionan el tipo de argumento no deseado y se obtiene una solución adicional.
- 13.- Si se desea que en la solución aparezcan acciones que no fueron incluidas pero que tienen alta ponderación, se genera una nueva solución.
- 14.- Se respaldan los archivos de trabajo para analizar la solución obtenida

#### NOTAS:

Este algoritmo garantiza que:

Siempre se tendrá una solución con la seguridad que se compararon todas las acciones de las metas iguales a las del problema planteado

Se pueden obtener soluciones distintas de acuerdo a los criterios de negociación (búsqueda)

Las acciones incluidas contribuyen a obtener los estados pedidos para la meta con la característica de ser la máxima cantidad de estados conseguidos.

### **3.7 Descripción del contexto del modelo**

Para poder hacer uso de la experiencia que se tiene en una empresa de los problemas de planeación, es necesario que las bitácoras de solución de problemas, constituyan una Memoria Organizacional, la cual significa: "Un conjunto estructurado de conocimiento relacionado a la experiencia de la empresa en un dominio dado" (Ackerman 96).

Durante el proceso de integración de una Memoria Organizacional se requiere homogenizar los documentos de bitácora de operaciones de soluciones de problemas, por lo cual se sintetizarán los documentos de bitácoras, aclarando los defectos o errores encontrados con los expertos del área en cuestión. Para no perder la actualización de la Memoria Organizacional la solución de cada nuevo problema se documentará con el formato necesario para su reutilización

### **3.8 Modelo conceptual**

El modelo aquí planteado comprende lo siguiente:

Se parte del planteamiento de un nuevo problema, en el cual se describen las *Metas* que se desean alcanzar, estas *Metas* se establecen mediante los *Estados* que se desean obtener. Obtener una solución es describir el *Plan* que permita alcanzarla.

Los pasos que constituyen cada meta serán conformados por la composición de Operadores tomados de experiencias anteriores, que a partir de un Estado Inicial se determina mediante el Modelo aquí planteado, cuál es el operador más adecuado que integre un paso de la meta en cuestión como puede verse en la figura 5.

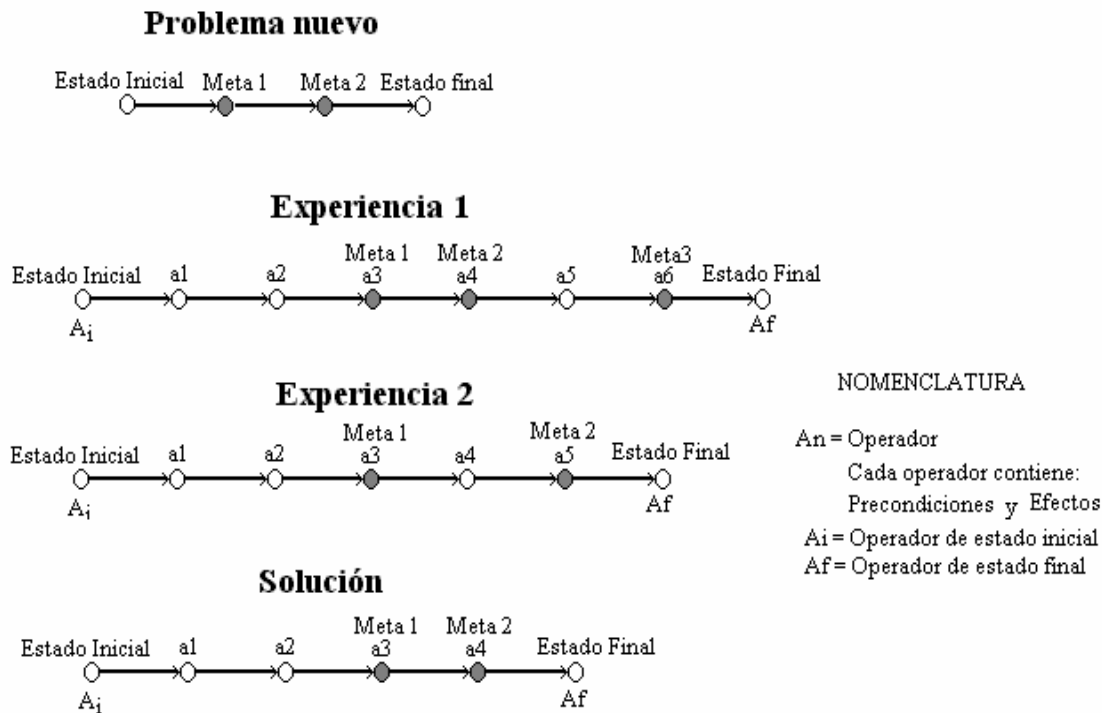


Figura 5. Descripción del contexto del modelo

Este modelo se basa en la mezcla de acciones que a continuación se describe formalmente

Se tiene un número variable de muestras de 1 a w (experiencias de agentes)

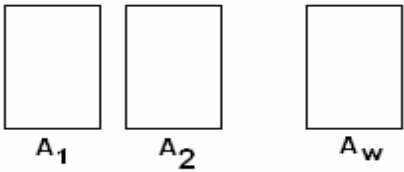


Figura 6. Grupos de experiencias

Cada muestra tiene una cantidad x de grupos m, no ordenados (metas de la experiencia de un agente)

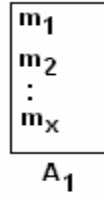


Figura 7. Conjunto de metas en cada experiencia.

Cada grupo tiene  $y$  elementos de  $a$  (acciones de una meta)

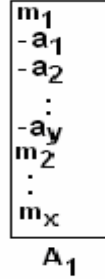


Figura 8. Grupo de acciones en cada meta.

Por lo que las  $w$  muestras, cada una con  $m$  grupos y cada grupo con  $y$  elementos se puede representar de la siguiente forma:

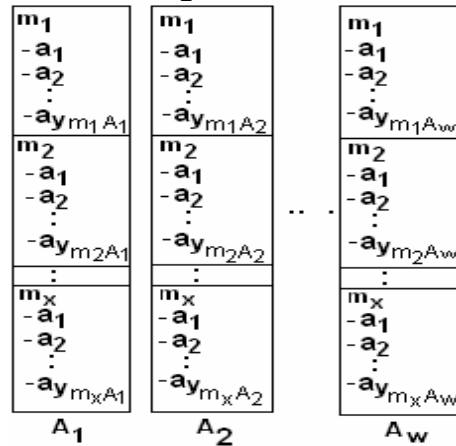


Figura 9. Grupo de experiencias con metas y acciones.

Se quiere comparar cada elemento  $a$  (acción de una meta) del grupo  $m1$  (**grupo 1 de metas de la experiencia de un agente**) de la muestra  $A1$  (**grupo de experiencias del agente 1**) con cada elemento del grupo  $m1$  de la siguiente muestra ( $A2$ ). La comparación se hace también con todas las muestras. Cuando los elementos sean iguales a la mayoría ( $[(W/2)+1]$  veces) de las muestras  $A$ ; entonces se guardarán en la muestra de salida **B**.

Para evitar comparar con elementos que ya fueron tomados para la muestra resultante **B**, y mejorar la eficiencia del proceso, se marcan los elementos iguales. El proceso se repite tomando como base de comparación cada elemento no marcado de cada grupo, de cada muestra; para compararlo con los elementos restantes no marcados de las demás muestras. Como se muestra en la siguiente figura.

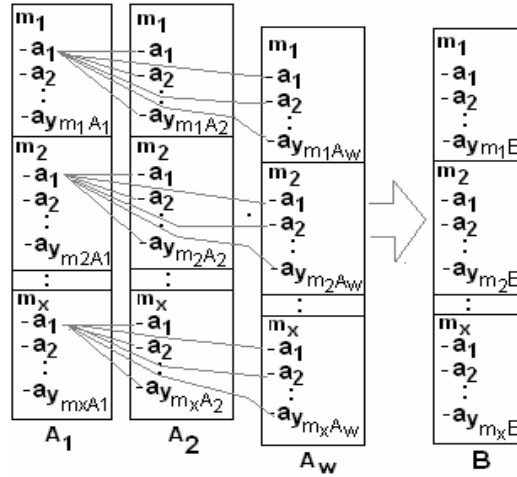


Figura 10. Esquema de comparaciones entre acciones.

La función de comparación puede expresarse de la siguiente manera:

$$\left[ \left[ [a]_{i=1}^{i=y} \right]_{j=1}^{j=w-1} \right]_{k=1}^{k=m} \xrightarrow{\text{compare}} \left[ \left[ [a']_{i=1}^{i=y'} \right]_{j=j+1}^{j=w} \right]_{k=1}^{k=m'} = \left[ [B]_{i=1}^{i=y} \right]_{j=1}^{j=m}$$

**sys may**  $[a = a']$

donde **may** =  $[(w/2) + 1]$  veces o más (son iguales en la mayoría de las comparaciones)

Por ejemplo, considerando las siguientes muestras:

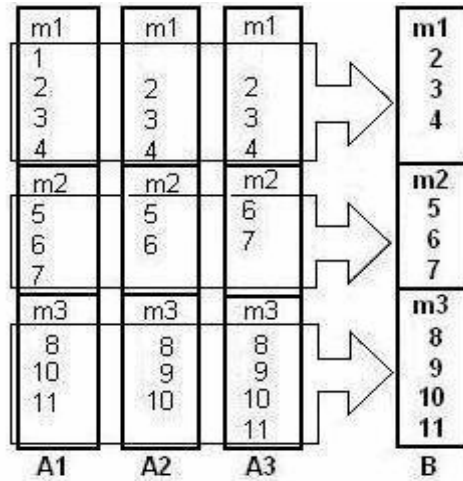


Figura 11. Ejemplo numérico del modelo

En la muestra de resultado **B** se tiene en el primer grupo m1, los elementos 2,3 y 4. Esto se debe a que 1 no está en la mayoría de las muestras. Del grupo m2, los elementos 5, 6 y 7 están en la mayoría de las muestras. En el grupo m3 los elementos 8, 9, 10 y 11 están en la mayoría de las muestras.

Para definir formalmente el número de comparaciones realizadas en un grupo se tiene la siguiente fórmula, representando MaxComparaciones como  $\mu$ :

$$\mu = (w-1)(y-y') + [w(w-1)]y' + y'w(y'-1)(w-1)$$

donde:

**y-y'** renglones con mayoría

**w-1** los posibles comparaciones en un renglón con mayoría

**[w(w-1)]y'** comparación de cada renglón sin mayoría con los demás elementos de su renglón

**y'w(y'-1)(w-1)** comparación de cada elemento de un renglón sin mayoría con los elementos de los demás renglones, excepto los de su misma columna (muestra)

Los posibles casos de comparación con  $w \geq 3$  tienen las actividades siguientes:

**φ** Cuando hay mayoría absoluta en un renglón se hacen w-1 comparaciones y se marcan los elementos iguales para ya no compararlos posteriormente

**β** Cada elemento de cada renglón se compara con los demás renglones, excepto con los elementos de la misma columna.

**Γ** Los renglones que no tienen mayoría, requieren compararse en el mismo renglón w(w-1) veces

En seguida se muestran los posibles casos de comparación:

**Caso 0** Ningún elemento tiene mayoría

Cada elemento se compara con los demás elementos del renglón

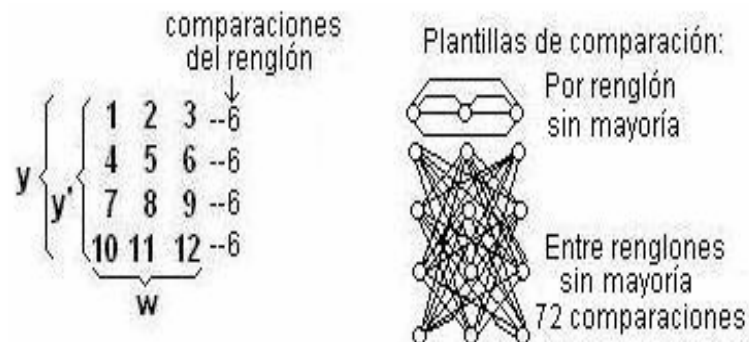


Figura 12. Comparación de acciones y plantilla para caso 0.

$$w = 3$$

$$y = 4$$

$$y' = 4$$

$$\mu = 96$$

**Caso 1:** Un renglón tiene mayoría



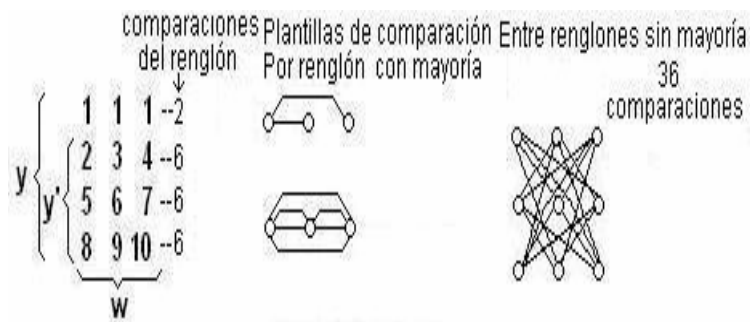


Figura 13. Comparación de acciones y plantilla para caso 1.

$w = 3$   
 $y = 4$   
 $y' = 3$   
 $\mu = 56$

**Caso 2: Dos renglones tienen mayoría**



Figura 14. Comparación de acciones y plantilla para caso 2.

$w = 3$   
 $y = 4$   
 $y' = 2$   
 $\mu = 28$

**Caso 3: Tres renglones tienen mayoría**

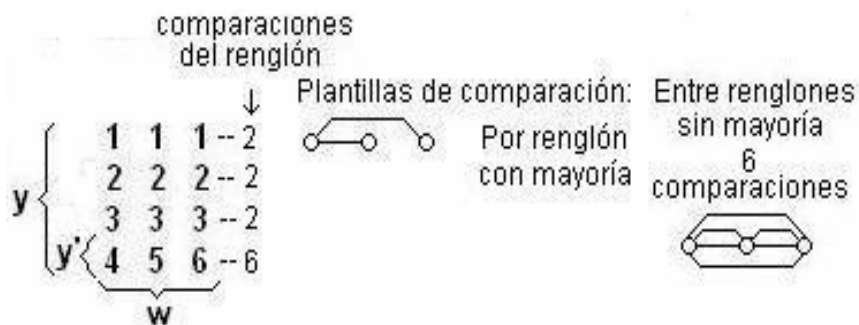


Figura 15. Comparación de acciones y plantilla para caso 3.

$w = 3$

$$y = 4$$

$$y' = 1$$

$$\mu = 12$$

**Caso 4: Cuatro renglones tienen mayoría**

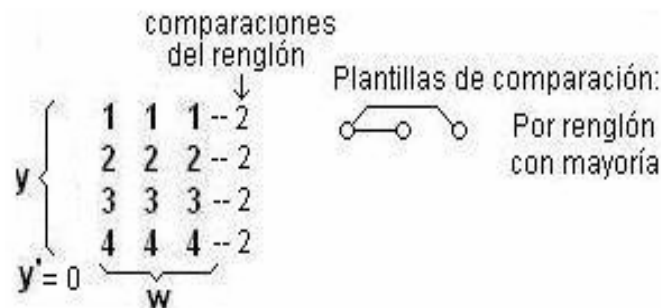


Figura 16. Comparación de acciones y plantilla para caso 4

$$w = 3$$

$$y = 4$$

$$y' = 0$$

$$\mu = 8$$

Sin embargo no están considerados todos los casos, algunas veces se tiene mayoría sin que todos los elementos sean iguales. Por ejemplo:

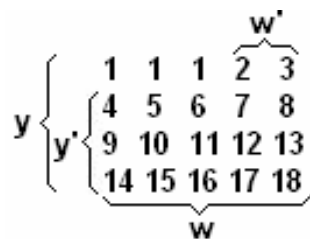


Figura 17. Mayoría relativa.

Esto hace necesario agregar un conjunto de comparaciones, cuya cantidad está dada por:

$$v = w'(w'-1) + 2w'(y^*+y'-1)$$

donde:

$w' =$  cantidad de elementos sin mayoría en un renglón con mayoría.  $w'$  cambia en cada renglón con mayoría.

$w'(w'-1) =$  comparación de elementos sin mayoría (no marcados) en el renglón a comparar.

$y^*+y'-1 =$  renglones con los que se compara  $w'$

Por lo tanto sumando  $\mu + v$

Tenemos que el número de comparaciones total  $\beta$  es:

$$\beta = (w-1)(y-y') + [w(w+1)]y' + y'w(y'-1)(w-1) + w'(w'-1) + 2w'(y^* + y'-1)$$

El significado de la fórmula de comparaciones es:

$$B = (w - 1)(y - y') + [w(w + 1)]y' +$$

comparación de elementos  
con mayoría

comparaciones en renglones  
sin mayoría

$$y'w(y' - 1)(w - 1) + w'(w' - 1) +$$

comparaciones entre renglones sin  
mayoría (excepto la columna del  
elemento a comparar en turno)

comparaciones de  
elementos residuales  
en renglones con  
mayoría

$$2w'(y^* + y' - 1)$$

comparaciones entre renglones  
con elementos residuales de  
renglones de mayoría con  
renglones sin mayoría

El conjunto de arreglos a manejar tiene la siguiente forma

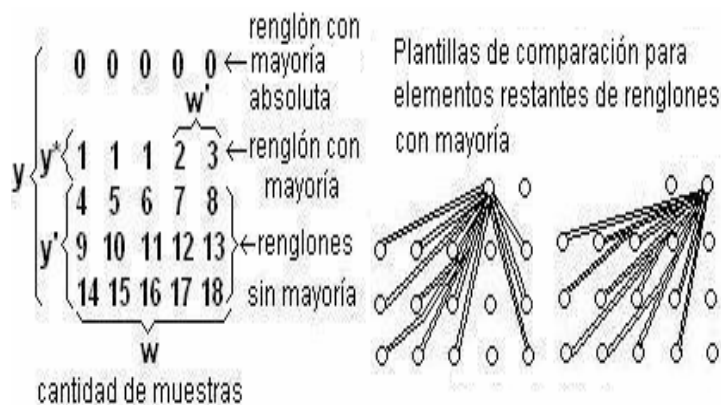


Figura 18. Tipos de comparaciones promedio.

Se ha hablado hasta ahora de cómo conformar el plan de solución; pero el modelo planteado es un metaplaneador por lo que se necesita distinguir un planeador de un metaplaneador en que:

Metaplaneador

Planeador

Usa conocimiento de planes  
Se comparte conocimiento  
Los recursos generales cons-  
truyen una o más soluciones

Usa conocimiento procedural  
Conocimiento no compartido  
Se obtiene una solución única

### 3.9 Descripción formal de operadores mezclables

Dado un conjunto de metas a alcanzar, un plan  $\pi$  es un conjunto de operadores (acciones) parcialmente ordenado, donde cada operador  $\xi$  es representado por precondiciones  $P_\xi$  y efectos  $E_\xi$ , los cuales por simplicidad se asume que son conjuntos de literales (sentencias atómicas positivas o negativas). Ninguna de las precondiciones del operador o su efecto pueden contener al mismo tiempo una literal o su negación. Un plan  $\pi$  puede ser representado como una gráfica  $\pi = (O, B)$  donde los vértices  $O$  son el conjunto de operadores en  $\pi$  y las aristas  $B$  son el conjunto de relaciones de precedencia en  $\pi$ . También se asume que dos operadores especiales existen en cada plan  $\pi$  que representan los estados iniciales y estados de meta en  $\pi$ . El operador de estado inicial  $I$  precede a los otros operadores en  $\pi$  y el operador de estado meta  $G$  es precedido por todos los otros operadores. El operador de estado inicial  $I$  tiene un conjunto vacío de precondiciones y tiene como efectos el conjunto de condiciones iniciales. De manera similar el operador de estado meta  $G$  tiene un conjunto vacío de poscondiciones y tiene como sus precondiciones las condiciones de meta a ser alcanzadas.

Se asume que cada operador  $\alpha$  tiene un costo asociado  $\text{costo}(\alpha)$ , y que el costo para un plan  $\pi$ , denotado como  $\text{costo}(\pi)$ , es la suma de los costos de los operadores en  $\pi$ .

En un plan de orden parcial, el operador  $\alpha$  necesariamente precede a  $\beta$ , si  $\alpha$  precede a  $\beta$  bajo cada orden total consistente del plan. Se usa el símbolo  $\prec$  para denotar precedencia necesaria en el orden parcial de  $\pi$ .

Se asume que el plan  $\pi$  se justifica debido a que cada operador en  $\pi$  es útil en establecer algunas precondiciones de otros operadores o los estados de meta deseados. El operador  $\alpha$  necesariamente establece una precondición  $p$  para un operador  $\beta$ , denotado por  $\text{Establece}(\xi, \beta, p)$  si, y sólo si,  $p$  es una precondición de  $\beta$  y un efecto de  $\alpha$  y no necesariamente un operador entre  $\xi$  y  $\beta$  tienen  $p$  o  $\neg p$  como efecto (Yang 97a).

Debido a que los operadores en la integración de una meta de la solución bajo el modelo MeCePCA necesitan mezclarse o eliminarse, se describen las condiciones para realizar estas operaciones.

#### 3.9.1 Concepto de operadores mezclables

Para un plan dado o propuesto, existen algunos operadores en el plan que se pueden agrupar y remplazar o complementar por un operador menos costoso o bien por un conjunto de operadores más descriptivo que alcance los mismos efectos útiles de los operadores agrupados. En tal caso se dice que los operadores son mezclables. En seguida se formaliza este concepto.

Empezaremos por definir que los operadores en un plan pueden agruparse. Un conjunto de operadores  $\Sigma$  en un plan  $\pi = (O, B)$  induce un subplan  $(\Sigma, B_\Sigma)$  dentro del plan  $\pi$ , donde  $B_\Sigma$  es un subconjunto máximo de  $B$  que son relaciones en  $\Sigma$ . Los operadores en  $\Sigma$  pueden ser agrupados si, y sólo si, ningún otro operador fuera de  $\Sigma$  está necesariamente entre cualquier par en  $\Sigma$ . Más precisamente

$$\forall \xi, \beta \in \Sigma, \neg \exists \gamma \in O - \Sigma, \text{ tal que } \xi \prec \gamma \prec \beta. \quad (4.3)$$

Sea  $\Pi_{\Sigma}$  un subplan de  $\pi = (O, B)$  inducido por  $\Sigma$ , donde  $\Sigma$  es un conjunto de operadores que pueden ser agrupados. Considere el comportamiento colectivo de  $\Pi_{\Sigma}$  en  $\pi$ . Algunos efectos de los operadores en  $\Sigma$  son útiles en  $\pi$ , debido a que ellos establecen las precondiciones de algunos otros operadores en  $\pi$  que están fuera de  $\Sigma$  o de las metas directamente. Otros efectos son: efectos laterales de los operadores o son usados para alcanzar las precondiciones de los operadores en  $\Sigma$ . Nótese que  $\Sigma$  también puede tener un sólo operador. Usamos la función Efectos-Útiles ( $\Sigma, \pi$ ) para denotar el conjunto de todos los efectos útiles de los operadores en  $\Sigma$ . Asimismo, usamos la función Red-de-Precondiciones ( $\Sigma, \pi$ ) para denotar el conjunto de todas las precondiciones de los operadores en  $\Sigma$  no alcanzadas por cualquiera de los operadores en  $\Sigma$ . Más formalmente,

$$\text{Efectos-Útiles}(\Sigma, \pi) = \bigcup_{\xi \in \Sigma} \{e \mid e \in E_{\alpha} \text{ and } \exists \beta \in (O - \Sigma) \text{ tal que } \text{Establece}(\xi, \beta, e)\} \quad (4.4)$$

$$\text{Red-de-Precondiciones}(\Sigma, \pi) = \bigcup_{\xi \in \Sigma} \{p \in P_{\xi} \mid \exists \beta \in (O - \Sigma) \text{ tal que } \text{Establece}(\xi, \beta, p)\} \quad (4.5)$$

Para dar una definición precisa para la mezcla de planes, un conjunto  $\Sigma$  de operadores es mezclable en un plan  $\pi = (O, B)$  si, y sólo si,  $\exists B', \mu$  donde  $B'$  es un conjunto de relaciones de precedencia en  $\Sigma$ , y  $\mu$  es un operador, tal que ( $\pi$  es un plan tentativo, analizable a partir de que existe una primera aportación u operador por parte de un agente).

1.  $\Sigma$  puede ser agrupado dentro del plan  $\pi$  o puede ser parte de un sub-plan tomado de cualquier meta de las experiencias de agentes.
2.  $B \cup B'$  es consistente y en un plan  $\pi' = (O, B \cup B')$   
 $P\mu \subseteq \text{Red Precondiciones}(\Sigma, \pi')$ , y  
 $\text{Efectos Útiles}(\Sigma, \pi') \subseteq E\mu$   
 Esto es los operadores  $\mu$  pueden ser usados para alcanzar todos los efectos útiles de los operadores en  $\Sigma$  mientras requiere solamente un subconjunto de sus precondiciones, después las restricciones de precedencia  $B'$  son impuestas en el plan  $\pi$ .  
 Además  $\mu \in \{ \text{Acciones (Metas Iguales (Experiencias (Agentes)))} \}$
3.  $\{\text{costo}(\mu) < \text{costo}(\Sigma)\} \vee \mu$  es consensuado (es decir  $\mu \rightarrow \in (O - \Sigma)$ )
- 4 Si no existen acciones anteriores a  $\mu$  se toma como:  
 $\mu = \min_{ID \in \text{Acciones } \Sigma \in (\text{Experiencias (Agentes)})}$
- 5.-.  $\mu \rightarrow \in \{SE\} \wedge \mu \in \{IG\} \wedge \mu \in ES$ . Donde SE = Sesgos, IG = Acciones iguales. y ES = conjunto de estados de la meta
6. El conjunto  $\Sigma$  es tomado de las experiencias de cada agente, lo cual implica, analizar los pasos 1 a 5 por cada conjunto  $\Sigma$  mezclable.

Figura 19. Modelo de composición de operadores mezclables.

El operador  $\mu$  es llamado un operador mezclable de  $\Sigma$  en el plan  $\pi$  y denotado por  $\mu = \text{merge } \pi(\Sigma)$

(o simplemente  $\text{merge}(\Sigma)$  si es claro que se trata del plan  $\pi$ ).

Ahora se explica por qué el conjunto de relaciones de precedencias  $B'$  es necesario en la definición de arriba.

Recordar que el conjunto de operadores en  $\Sigma$  puede ser agrupado. Dependiendo de diferentes relaciones de precedencia (es decir  $B'$ ) impuestas sobre  $\Sigma$ , el conjunto  $\Sigma$  de operadores requerirá diferentes conjuntos de precondiciones globales para ser alcanzadas por los operadores fuera de  $\Sigma$ . Por tanto una opción diferente de  $B'$  puede dar surgimiento a un operador mezclable diferente  $\mu$ . Seleccionamos el operador mínimo en costo o el más descriptivo para ser nuestro operador mezclable. Las precondiciones y efectos de los dos operadores están dados en la tabla 1. Las opciones en  $\mu$  se mencionan en seguida:

**Opción 1:**  $B' = \{\xi_1 < \xi_2\}$ . Entonces  $\xi_1$  alcanza la precondición  $q_1$  de  $\xi_2$ . Por lo tanto la Red-de-Precondiciones de  $\Sigma$  debe ser  $\{q_1\}$ .

**Opción 2:**  $B' = \{\xi_2 < \xi_1\}$ . Entonces  $\xi_2$  alcanza la precondición  $q_2$  de  $\xi_1$ . Por lo tanto la Red-de-Precondiciones de  $\Sigma$  debe ser  $\{q_2\}$ .

En cada opción de arriba el conjunto de precondiciones es diferente a partir de la otra opción. Así que cada opción de  $B'$  puede resultar en un operador  $\mu$  mezclable diferente.

En este caso, el del costo más bajo es un operador mezclable.

Operador	Precondición	Efectos
$\xi_1$	$q_1$	$p_1, q_2$
$\xi_2$	$q_2$	$p_2, q_1$

Tabla 2. Definición de operador

La definición para mezclar un operador claramente cubre los ejemplos dados

En seguida se describen un ejemplo en el que se utiliza la descripción de acciones mediante el lenguaje de acciones diseñado por Pednault (Pednault 88), en el cual se describe mediante fórmulas de primer orden, las acciones. El predicado corresponde a la acción y los términos a los argumentos.

Para describir los planes se usará el formato:

Plan (fórmula de primer orden) $_1 < (fórmula de primer orden)_2 < (fórmula de primer orden)_3$

Meta: condiciones del modelo del mundo

Lista de condiciones

Operadores: Precondiciones

lista de condiciones del modelo del mundo

Efectos

lista de condiciones del modelo del mundo

### 3.9.2 Ejemplo

Considere un plan para alcanzar dos metas: (1) ir a la escuela  $S$  desde casa  $H$  y regresar, (2) ir desde casa a la tienda  $G$  para comprar  $X$  y regresar. Este plan consiste de dos subplanes,

Plan1 = ir a ( $H, S$ )  $\prec$  ir a ( $S, H$ ),  
 Plan2 = ir a ( $H, G$ )  $\prec$  comprar ( $X, G$ )  $\prec$  ir a ( $G, H$ )  
 El operador inicial es: estar( $H$ ) y  
 el operador final es: estar( $H$ )

### Nuevo Plan:

Condiciones:

en ( $H$ )  $\prec$  en ( $G$ )  $\prec$  en( $S$ )  
 el costo de trasladarse a cualquier lugar es 1

Meta1:

Lista de precondiciones:

alcanzar los siguientes estados en el orden descrito:

en ( $H$ ), en ( $S$ ), en ( $H$ )

Lista de poscondiciones:

en( $H$ )

Meta2:

Lista de precondiciones

en ( $H$ ), en( $G$ ), comprar( $X$ ), en( $H$ )

Lista de poscondiciones:

comprar( $X$ )

Operadores de meta1

Operadores:	Precondiciones	Poscondiciones
Ir a( $H, S$ )	en( $H$ )	en( $S$ )
Ir a( $S, H$ )	en( $S$ )	en( $H$ )

Operadores de meta2

Operadores:	Precondiciones	Efectos
Ir a( $H, G$ )	en( $H$ )	en( $G$ )
comprar( $X$ )	en( $X$ )	comprar( $X$ ), en( $G$ )
ir a( $G, H$ )	en( $G$ )	en( $H$ )

Los dos subplanes pueden ser mezclables por (4.2) al remplazar ir a ( $S, H$ ) e ir a ( $H, G$ ) por ir a( $S, G$ ). Si  $G$  está localizada entre casa  $H$  y escuela  $S$  por (4.3), entonces el plan resultante es

Ir a( $H, S$ )  $\prec$  ir a( $S, G$ )  $\prec$  comprar ( $X, G$ )  $\prec$  ir a( $G, H$ )

con menores costos que el original.

En el ejemplo de arriba el conjunto  $B'$  de relaciones de precedencia es {ir a ( $S, H$ )  $\prec$  ir a( $H, G$ )}.

Sea  $\Pi'$  el plan con  $B'$  impuesto en el plan original, entonces

Red-de-Precondiciones ( $\Sigma, \Pi'$ ) = {en( $S$ )} por 4.2,

Efectos-Útiles ( $\Sigma, \Pi'$ ) = {en( $G$ )} por 4.3

donde  $\Sigma = \{ \text{ir } a(\mathbf{S}, \mathbf{H}) < \text{ir } a(\mathbf{H}, \mathbf{G}) \}$ . Así  $\Sigma$  puede ser mezclada dentro del operador de mezcla ( $\Sigma$ ) = ir  $a(\mathbf{S}, \mathbf{G})$ . Nótese que existe otra forma de mezclar los operadores en este plan, es decir,  $\{ \text{ir } a(\mathbf{G}, \mathbf{H}) < \text{ir } a(\mathbf{H}, \mathbf{S}) \}$  puede ser mezclada dentro de ir  $a(\mathbf{G}, \mathbf{S})$ , pero uno tiene que decidir cual forma mezclar ya que es imposible mezclar ambos conjuntos de operadores. La habilidad para seleccionar óptimamente entre diferentes mezclas inconsistentes posibles es una característica importante del algoritmo MeCePCA



## 4 Diseño Físico y Resultados

---

### **Resumen**

En este capítulo se describen el diseño del prototipo, cuya base de programación es el manejador de Base de Datos Visual FoxPro V9.0. Asimismo el diseño se realizó tomando en cuenta el Modelo Orientado a Objetos

### **4.1 Fundamentos del prototipo**

El modelo como se describió en el capítulo 3 se basa en dos áreas fundamentales de los procesos de planeación y toma de decisiones basadas en tecnología. Este modelo resultó ser un modelo híbrido y como es común en estos casos se tienen los beneficios de ambas áreas.

Por otra parte el modelo es parte también de los Sistemas Expertos, descritos en el capítulo 2. Asimismo el modelo recibe influencia y da aportaciones en la Tecnología de los Sistemas Expertos como más adelante se describe

Del área de toma de decisiones se obtienen las intervenciones tipificadas, es decir se tienen las aportaciones de los agentes con completez, elimina en algunos participantes, factores de olvido, métodos de solución secretos, influencia de otros participantes; y del área de los planificadores se obtiene la formalidad de descripción de los planes y su conformación.

Por otra parte se seleccionó el software de desarrollo Visual FoxPro debido entre otras cosas al compararlo con Developer de Oracle:

- Oracle no permite la visualización de tablas directamente se requiere estar en el ambiente Developer para tener acceso, Visual Fox no
- Developer no permite barridos para acceso directo a las tablas, Visual Fox si
- Visual Fox incluye herramientas que otros manejadores traen por separado, tales como: Report, Graphics y Project.
- El código en VisualFox no requiere una estructura rígida, desde luego sólo requiere cumplir con las reglas semánticas y sintácticas.
- En Fox existen más funciones y procedimientos que el promedio de desarrolladores requiere.
- Fox permite manejar cursores (tablas dinámicas con datos de cualquier origen), otros no.
- Developer no cuenta con un manejador de base de datos propio, sino que requiere de sql/plus (Oracle)
- Developer no puede manipular directamente variables de un formulario
- Fox puede manipular listas desplegables con una sola herramienta, no así Developer
- Fox cuenta con un constructor de proyectos y aplicaciones, no así Developer de Oracle

## 4.2 Casos de uso

El prototipo tiene los siguientes casos de uso

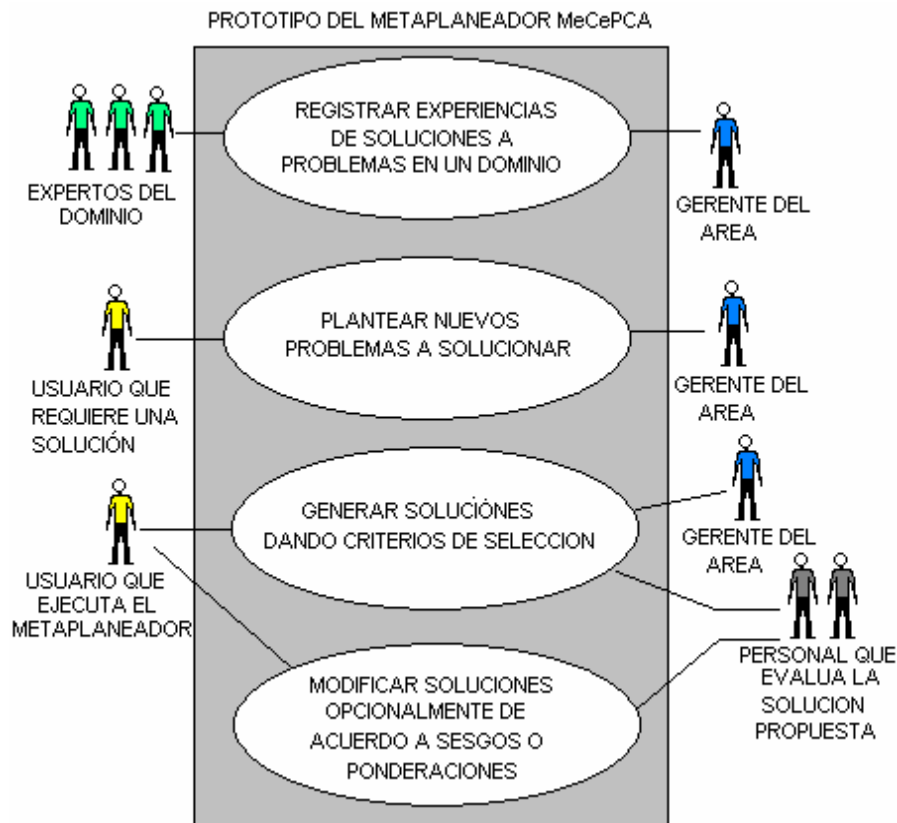


Figura 20. Casos de uso del Metaplaneador

Proceso: Registrar experiencias de soluciones a problemas en un dominio

Actores: Expertos del área, Gerente del área

El Gerente del área es la persona que tiene a su cargo un área crítica de la empresa, cuya responsabilidad es comúnmente mantener a la empresa funcionando, por ejemplo el Gerente de Producción.

Como comúnmente soluciona problemas en su área, vigila que las soluciones de estos problemas queden debidamente registrados en la Memoria Organizacional de la gerencia.

Pide que le informen los expertos la forma de solucionar el problema más reciente, verificando que esta solución haya quedado debidamente alimentada.

Los expertos van registrando las secuencias de actividades de la solución del problema presentado. Esta tarea la realiza en papel al ir dirigiendo la solución o bien directamente captura la solución en el metaplaneador.

Los datos que se guardan son: La meta que se esta cumpliendo, las actividades realizadas para esas metas. Los datos de la actividad que debe registrar son: las

precondiciones, poscondiciones, estados alcanzados, costo, duración, la ponderación de la acción (importancia para el experto), acción alternativa, y los argumentos de la acción.

Proceso: Plantear nuevos problemas a solucionar

Actores: Gerente del área, Usuario que requiere una solución

Al gerente del área cuando se le presenta un problema en su área requiere darle solución, pero los expertos de su gerencia, no se encuentran por el momento (porque renunciaron, porque están en otra ciudad, o porque ahora son parte de otra área, están de vacaciones, etc.), entonces solamente se tiene al usuario del metaplaneador y la Memoria Organizacional

El usuario plantea el problema en términos de las metas que desea alcanzar en el nuevo problema. Normalmente el usuario es el asistente del gerente o personal de logística

Proceso: Generar soluciones dando criterios de selección.

Actores: Gerente del área, Usuario que ejecuta el metaplaneador, Personal que evalúa la solución propuesta.

El Usuario ejecuta el metaplaneador y genera varias soluciones en la que cambia los criterios de selección de las acciones: por el menor costo en las acciones, por el menor tiempo de cada acción, por el mayor costo o por el mayor tiempo en las acciones. Estos criterios dan alternativas en la solución que el Gerente y el personal que evalúa las soluciones pueden analizar, de acuerdo a las circunstancias en que se presenta el problema.

El criterio de selección de menor costo en cada acción da la solución más barata.

El criterio de selección de menor tiempo obtiene la solución más rápida que se puede ofrecer.

El criterio de selección de mayor costo genera una solución que trata de emplear los mejores componentes asumiendo que el mayor costo sugiere que los materiales son de mayor calidad.

El criterio de selección de mayor tiempo, supone que no afecta la duración que tenga la solución, con tal de que la solución sea bien implantada.

El personal que evalúa la solución revisa los costos actuales de los materiales, las horas hombre de cada actividad y el personal disponible para realizar las actividades. Comúnmente este personal está constituido por el contador y/o el jefe del área

Proceso: Modificar soluciones opcionalmente de acuerdo a sesgos o ponderaciones

Actores: Usuario, Personal que evalúa las soluciones, gerente del área

El usuario cambia las soluciones de acuerdo a las modificaciones que le pide el personal que evalúa las soluciones

El Gerente recibe la o las mejores soluciones de acuerdo a las circunstancias en que se presenta el problema.

Una opción muy útil en el metaplaneador es que puede eliminar acciones ya sea porque alguno de los materiales no está disponible, o no se quiere realizar ciertas acciones que manejan argumentos que no son convenientes.

De igual manera las ponderaciones, las cuales nos dicen el grado de importancia que un experto le dio a ciertas acciones pueden ser incluidas o no dependiendo de las preferencias del usuario.

### 4.3 Clases candidatas

Las clases candidatas son obtenidas a partir del análisis textual y de los casos de uso. Para describir el análisis textual se plantea el problema de la siguiente manera:

Un gerente de área requiere tener soluciones a los problemas de su gerencia cuando surgen problemas en procesos o equipos críticos para el funcionamiento de la empresa. El gerente vigila que se registren todas las soluciones implantadas de los problemas presentados en su área. Cuando surgen nuevos problemas, se pide al asistente que utilice los registros de soluciones para obtener una probable solución. Las soluciones son secuencias de acciones que pertenecen a una meta, las acciones que se integran a la solución son tomadas por la comparación con otras acciones de las diferentes soluciones registradas. Todas las acciones se describen con una cadena de caracteres y con argumentos que describen la forma de realizar la acción.

En la siguiente figura se observan el diagrama de clases candidatas, cuyos atributos son obtenidos de los casos de uso y los nombres de las clases se obtienen del análisis textual descrito en el párrafo anterior

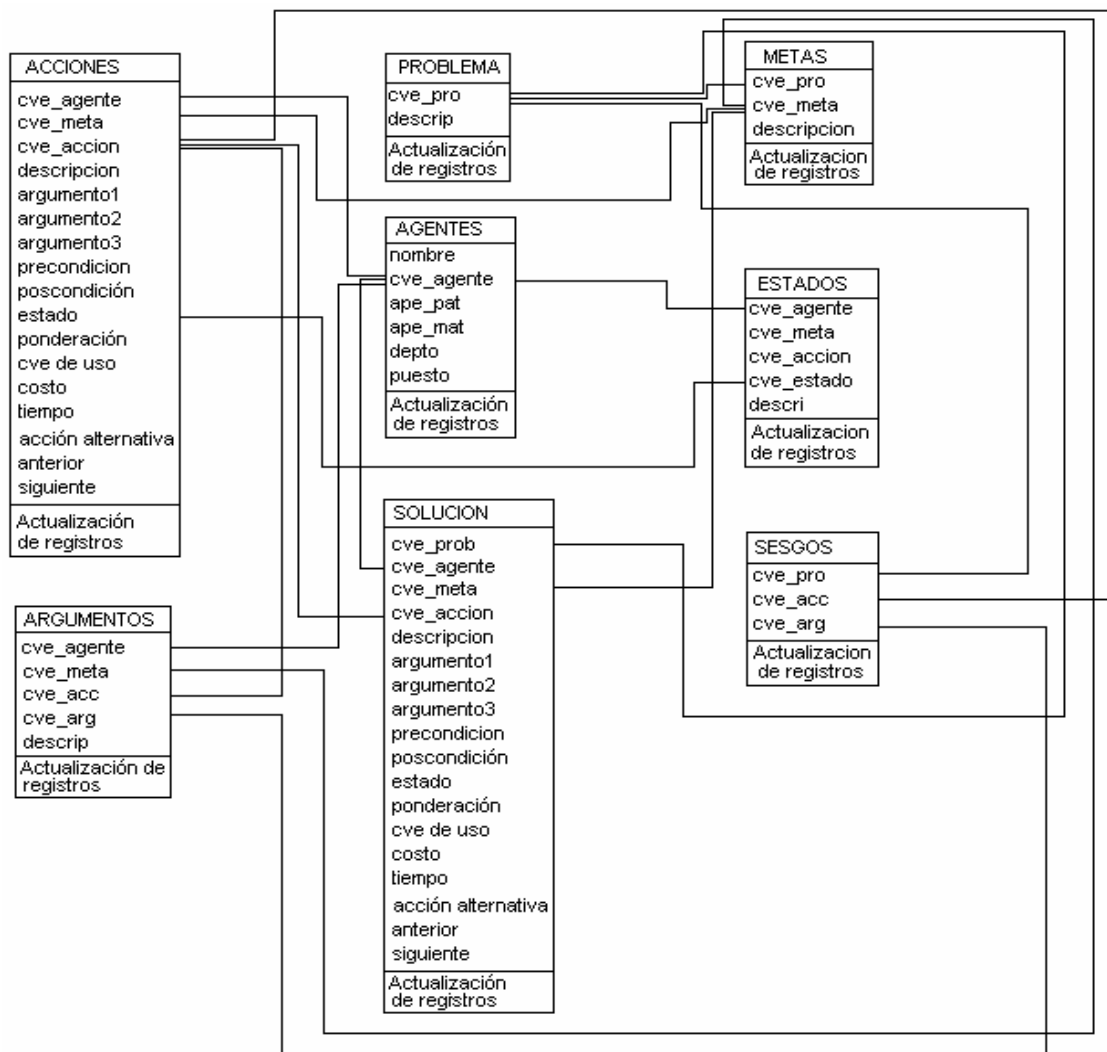


Figura 21. Diagrama de clases candidatas.

### 4.4 Formularios

Algunos de los formularios que se manejan son los siguientes:

The 'Agentes' form is a window titled 'Agentes' with a blue header bar. It contains several input fields and a list of buttons. The fields are: 'Clave del Agente' (a text box with '7' and an empty checkbox), 'Nombre' (a text box with 'JOSE ALBERTO'), 'A. Paterno' (a text box with 'HERNANDEZ'), 'A. Materno' (a text box with 'ESPINOSA'), 'Puesto' (a text box with 'DESARROLLADOR'), and 'Departamento' (a text box with 'SISTEMAS'). To the right of these fields are four buttons: 'Guardar', 'Eliminar', 'Limpiar', and 'Cerrar'. Below the fields is a 'Datos Adicionales' section with a large text area containing the word 'WERITO'.

Figura 22. Formulario para captura de datos de los agentes expertos

Otro de los formularios importantes en el sistema es el de alimentación de acciones

The 'Alta de Experiencias' form is a window titled 'Alta de Experiencias' with a blue header bar. It is divided into two main sections: 'Datos Generales' and 'Metas'. The 'Datos Generales' section contains two rows of data: 'Problema' (a dropdown menu with '1' selected) and 'Agente' (a dropdown menu with '7' selected). The 'Metas' section contains several rows of data: 'Clave Meta' (a dropdown menu with '1' selected), 'Clave Accion' (a dropdown menu with '1' selected), 'Ponderacion' (a dropdown menu with '5' selected), 'Argumento 1' (a dropdown menu with '12' selected), 'Argumento 2' (a dropdown menu), 'Argumento 3' (a dropdown menu), 'Precondición' (a dropdown menu with '1' selected), 'Accion Sig' (a dropdown menu with '2' selected), 'Tener receta' (a text box with 'GELATINA DE ROMPOPE'), 'BUSCAR RECETA' (a text box with 'JOSE ALBERTO HERNANDEZ ESPINOSA'), 'Muy Alta' (a text box), 'Ponderacion' (a dropdown menu with '3' selected), 'Ponderacion' (a dropdown menu), 'Ponderacion' (a dropdown menu), 'Poscondición' (a dropdown menu with '4' selected), 'Accion Alter' (a dropdown menu with '3' selected), and 'Recursos' (a section with 'Tiempo' and 'Dinero' text boxes). To the right of the 'Metas' section are five buttons: 'Guardar', 'Metas', 'Acciones', 'Argumentos', and 'Salir'.

Figura 23. Formulario para captura de acciones.

Las acciones constituyen el punto medular en el proceso de búsqueda de la solución. Por otro lado la solución muestra el resultado de manera explícita

The screenshot shows a window titled "Solución" with a blue border. Inside, there are several input fields and buttons. At the top left, "Clave del problema:" is followed by a text box containing "1". To its right is a small icon. Below this, "Descripción:" is followed by a text box containing "GELATINA DE ROMPOPE". To the right of the description box are three buttons: "Solucionar", "Siguiente", and "Salir". Below these, there are two columns of input fields. The left column contains: "Clave del Agente:" (1), "Clave del Problema:" (1), "Clave de la Meta:" (4), "Clave de la Acción:" (15), "Clave de argumento 1" (16), "Clave de argumento 2" (9), and "Clave de argumento 3" (0). The right column contains: "Clave Poscondición" (0), "Clave Precondición" (0), and a section titled "Recursos" with "Tiempo en minutos:" (5), "Costo en pesos:" (0.00), and "Clave del estado" (empty). All input fields are text boxes with their values displayed inside.

Figura 24. Muestra de la solución

Los criterios generales en el proceso de trabajo del prototipo son los siguientes:

Criterios de registro de Acciones:

1. Cuando se requiera agregar acciones en el catalogo se pueden incluir decimales.

Criterios de búsqueda:

1. En la búsqueda de acciones iguales se tomará como inicio de la comparación la experiencia (columna) que tenga más acciones. Porque describe con mayor detalle un plan. De ahí seguirá la experiencia con más acciones que siga a la primera.
2. Si dos acciones son iguales se toma la que cumpla con el criterio de selección (mayor \$ o tiempo / menor \$ o tiempo). Si son todavía iguales se selecciona la que consiga mas estados y si todavía no se tiene una, entonces se toma la primera acción tomada para comparar

Reglas:

1. No debe haber la misma acción en una meta (con los mismos argumentos)
2. Para que una acción se tome como igual a otra se compara la acción y los argumentos deben ser iguales, no importa el orden

Criterios para la solución:

1. Se tiene opcionalmente la capacidad de eliminar acciones que tengan argumentos que no se deseen.

Después de obtener la solución si se tienen acciones en las experiencias que no fueron incluidas en la solución y tienen una ponderación alta (la da el usuario) se pueden seleccionar para incluirlas en el lugar adecuado según el usuario.

### 4.5 Pruebas

La forma en que se ve la solución es la siguiente:

cve_p	roble	cve_a	gente	nombre	paterno	materno	clave_	meta	cve_a	ccion	accion	pond_	acc	tiempo	costo	cve_	arg_1	argum_1	pond1	cve_a	argum_2	pond_2	cve_	arg_3	argum_3	pond3	pre	pos	sig	acc_	alt
	GELATINA										BUSCAR																				
	AZUL																														
2	ALIENDRAS	7	JOSE ALBERTO	HERNANDEZ	ESPINOSA		1	Tener receta	1	RECETA		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GELATINA																														
	AZUL																														
2	ALIENDRAS	7	JOSE ALBERTO	HERNANDEZ	ESPINOSA		1	Tener receta	2	LEER RECETA		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GELATINA										CONSEGUIR_INI																				
	AZUL										GREDEINTE																				
2	ALIENDRAS	7	JOSE ALBERTO	HERNANDEZ	ESPINOSA		2	Conseguir mate	3	GREDEINTE		0	0	0	0	0	0	11_TAZA	0	0	0	0	0	0	0	0	0	0	0	0	0
	GELATINA										CONSEGUIR_INI																				
	AZUL										GREDEINTE																				
2	ALIENDRAS	7	JOSE ALBERTO	HERNANDEZ	ESPINOSA		2	Conseguir mate	3	GREDEINTE		0	0	0	0	0	0	22_CUCHARA	0	0	0	0	0	0	0	0	0	0	0	0	0
	GELATINA										CONSEGUIR_INI																				
	AZUL										GREDEINTE																				
2	ALIENDRAS	7	JOSE ALBERTO	HERNANDEZ	ESPINOSA		2	Conseguir mate	3	GREDEINTE		0	0	0	0	0	0	11_TAZA	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabla 3. Formato de Solución

Las pruebas se llevaron a cabo con 16 recetas de la siguiente manera: Primero se probó con 3 recetas para obtener una solución, después se hizo con 5 recetas, Luego con 7 y así sucesivamente hasta 33 recetas diferentes, los tiempos que se obtuvieron en una computadora con 512 KB de RAM y procesador AMD

Número de experiencias	Tiempo de ejecución
3	8 seg.
5	10 seg
7	16 seg
9	28 seg.
11	40 seg.
13	68 seg.
15	88 seg.
17	102 seg
19	140 seg.
21	180 seg.
23	210 seg.
25	240 seg.
27	290 seg.
29	340 seg.
31	450 seg
33	560 seg

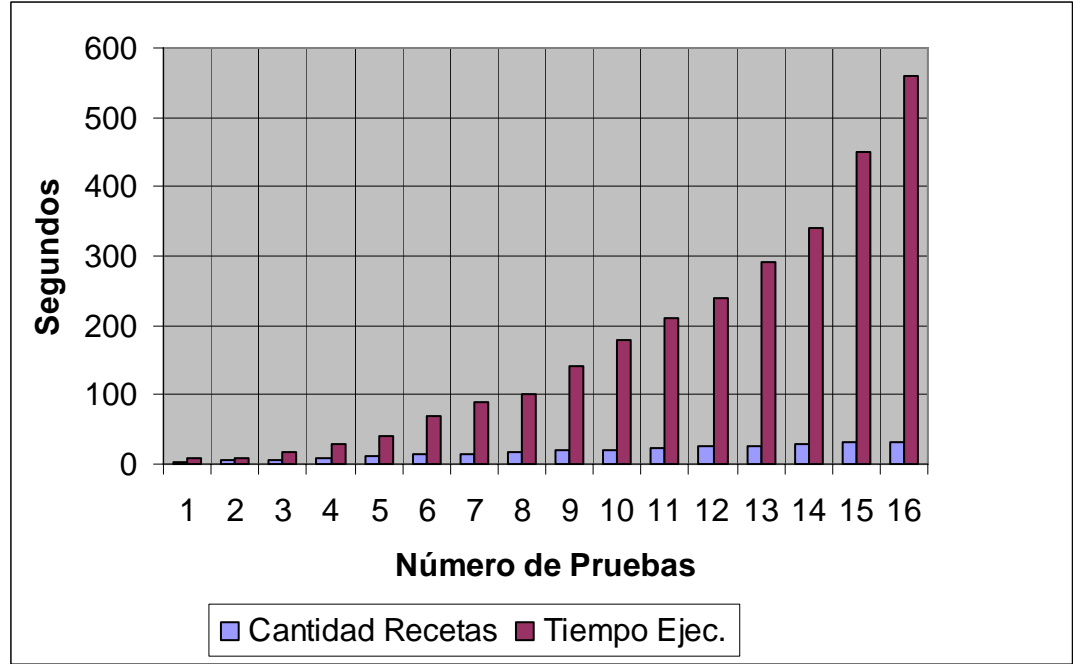


Figura 25. Tiempos de ejecución para 16 pruebas con 33 recetas agrupadas acumulativamente.

Los datos de las pruebas pueden verse en el anexo.



## 4.6 Métricas

Las métricas que se usaron se describen como razones porque nos muestran: el número de metas encontradas por experiencias disponibles, el número de acciones en la solución sobre las acciones disponibles en el catálogo y el grado de influencia en la solución por los agentes.

### Razones

Razón de metas encontradas	$y / w$
Razón de acciones diferentes en la solución	$z / x$
Razón de influencia de los agentes en la solución	$a / w$

Donde

$w$  = Número de experiencias disponibles

$x$  = Número de acciones disponibles

$y$  = Número de metas

$z$  = Número de acciones en la solución

$a$  = Número de agentes en la solución

## 4.7 Resultados y discusión

Los resultados muestran un prototipo consistente, que brinda buenos resultados conforme al planteamiento de soluciones en el dominio de la preparación de postres de gelatina. El comportamiento es fácil extrapolarlo a un dominio de solución de problemas con actividades secuenciales y que preferentemente tengan importancia para una organización, en donde se pueda dar seguimiento a las soluciones implantadas en un momento y lugar de implantación, para registrar adecuadamente dichas soluciones.

Para dar idea de la serie de obstáculos presentados, se puede mencionar entre otros: Orden en el catálogo de acciones; Pivote de búsqueda Decidir por el criterio de mayoría; Inclusión de sugerencias en la solución; Mínimo de experiencias Asegurar completez; Pasar del “backtracking” a uso de DBMS; el método de mezcla de acciones, del uso de agentes a Memoria Organizacional.

Todo esto brindó la posibilidad de producir un prototipo robusto en su funcionamiento y que depende totalmente de la buena administración de la información que suministre el usuario para obtener los resultados esperados.

Comparando con otros modelos de planeadores, podemos decir que tiene este modelo algunas ventajas y desventajas como las siguientes:

### VENTAJAS

- Los resultados se obtienen con la seguridad que la mayoría de los expertos sugieren las acciones mostradas en cada meta
- La búsqueda de acciones es exhaustiva
- Se puede modificar la solución incluyendo acciones con alta ponderación o eliminar las que tengan argumentos no deseados
- Se maneja planeación jerárquica al seleccionar primero las metas y después las acciones

- Se generan soluciones con la combinaciones de mayores o menores costos y tiempo

#### CARACTERISTICAS A INCLUIR EN EL FUTURO

- Manejo de contingencias
- Manejo de costos actualizados (con valor del dinero en el tiempo)
- Manejo de probabilidades
- Distribución de recursos para generar actividades paralelas

#### DESVENTAJAS

- No identifica el caso más parecido (razonamiento basado en casos)
- No maneja actividades paralelas
- No se tiene negociación real

El comparar los resultados del prototipo se puede hacer solamente con el modelo de obtención de resultados en reuniones presenciales como se muestra en seguida:

Solución del Metaplaneador MeCePCA					Solución del modelo Groupware Tipificado		
Metas de Problema a resolver y su solución					Solución propuesta por expertos		
Meta 1					Meta 1 (Solución de grenetina)		
	Agente	Meta	accion	parámetros	Experto	Acción	Forma de implantarla
	2	1	2	(3,5)	Alberto X	Hervir agua	En recipiente de 1 lt.
	1	1	4	(4)	Luis Y	Disolver sobre	Mover hasta disolver
	1	2	5	(2,1)			
Meta 2					Meta 2 (Vaciar en molde)		
	3	2	1	(1,2)	Carlos Z	Vaciar al molde	Preparar molde
	1	3	2	(6,8)	Alberto X	Dejar reposar	A temperatura amb. 1hr.
	2	2	4	(9,10)	Alberto X	Refrigerar	2 hr
	1	2	7	(11,14)			
	4	2	9	(3)			
Meta 3					Meta 3 (Presentarla)		
	2	4	3	(2)	Luis Y	Desmoldar	En plátón
	2	3	1	(15, 17)	Carlos Z	Mostrar a comenzal	En la mesa
					Meta 4 (Degustar)		
					Carlos Z	Cortar y servir	Cuchillo y plato postre
					Luis Y	Degustar	Con cuchara
					Luis Y	Dar opinion	verbal

Figura 26. Comparación de soluciones

La comparación registra soluciones iguales, aunque en el prototipo solo se ven claves, anteriormente se mostró el formato de la solución con las descripciones respectivas.

Algunas diferencias pueden darse en el proceso de coordinación, las cuales no se registran en el resultado. Para coordinar las reuniones se pueden manejar los métodos mostrados en el capítulo 2 (Marco Teórico)

## 4.8 Aportaciones

Algunas de las aportaciones obtenidas son:

- Un prototipo portable a cualquier computadora con Visual FoxPro 9.0
- Un modelo que reduce la participación presencial de los expertos cuando se busca una solución

- El modelo ayuda en la toma de decisiones de problemas trascendentes en la organización
- Se creó un nuevo método formal de mezcla de arreglos por mayoría

## 4.9 Líneas de investigación futuras

Las posibles líneas de investigación que surgen a partir de este modelo son:

- La Robótica para toma de decisiones de trayectorias.
- La Agricultura para disminuir el número de experimentaciones fallidas
- La Medicina para medir probabilidades de fracaso o acierto en decisiones de intervenciones quirúrgicas o diagnósticos
- La industria para decisiones de reemplazo e inversión
- El establecimiento de comercios para decisiones de selección de lugares
- La Mercadotecnia en el lanzamiento de nuevos productos
- Seguimiento de demandas legales, para decidir estrategias
- El gobierno para decidir sobre problemas ambientales, sociales, de salud, etc. entre otros.

## 4.10 Análisis de cumplimiento de los objetivos propuestos

Se cumplió el objetivo principal:

***“Definir un modelo de Metaplaneador, formado por la composición de acciones entre agentes, basados en sus experiencias”***

Los objetivos específicos:

- *Conseguir completez para cada meta del plan de solución*

Esto se consiguió porque aunque los expertos quisieran guardarse algún conocimiento, no lo pueden hacer ya que sus experiencias han quedado registradas con anterioridad. Por tanto no pueden manipular este conocimiento para afectar un plan de solución en un problema nuevo del mismo dominio.

- *Que el modelo propuesto contenga la facilidad de manejar Soluciones con sesgos*

Cuando se genera una solución es posible eliminar acciones con argumentos no deseados por el usuario del prototipo

- *Implementar un prototipo que permita evaluar la completez en la conformación de planes*

La versión del software que permitió evaluar el modelo es una prueba de que funciona el modelo propuesto

- *Presentar y discutir aplicaciones reales*

Este objetivo se cumplió parcialmente porque no se pudo probar más de una aplicación real, es decir solamente se hizo con una aplicación que es sencillo extrapolar su funcionamiento a otros dominios. Agregando que una causa importante fue que la empresa que invitamos a participar, no quiso ceder su información por considerarla confidencial y sobre todo porque pertenece a un proceso importante en el manejo de sus equipos estratégicos.

- *Comparar otros modelos de metaplaneadores con el modelo propuesto*

No existía equivalencia notablemente marcada para poder hacer comparación en el funcionamiento en otros modelos y el modelo aquí propuesto. Lo que si se pudo es comparar es el modelo de GDSS, mejor descrito como un Chat tipificado en sus intervenciones, mostrado en párrafos anteriores de este capítulo

- *Presentar y discutir posibles líneas de investigación que pueden surgir a partir de este modelo*

Mencionadas en este mismo capítulo, muestran algunos de los alcances que según el autor es factible de obtener al aplicar el modelo en diferentes áreas.

## BIBLIOGRAFIA

(Ackerman 96)

Ackerman, Mark S. Organizational Memory. .  
University of Michigan 1996

(Andueza 97)

Andueza, María. Dinámica de Grupos en Educación. Trillas. 1997

(Chapman 84)

Chapman, D. Planning for conjunctive goals. Artificial Intelligence, 32. 1984

(Chapman 87)

Chapman, D. (1987) Planning for conjunctive goals. Artificial Intelligence, 32 1987

(Cox & Durfee 02)

Cox Jeffrey S., Durfee Edmund H. Discovering and exploiting synergy between hierarchical planning agents. ACM proceedings of agents. 2002.

(Dean & McDermott 87)

Dean, T.L. & McDermott, D. Temporal data base management: Artificial Intelligence 32. 1987

(Etzioni et al 92)

Etzioni, O., Hanks, S., Weld, D., Draper, D., Lesh, N., Williamson, M. An approach to planning with incomplete information. In Proceedings of the Third International Conference on Knowledge Representation. Morgan Kaufmann Boston, MA 1992.

(Felner et al 02)

Felner, Ariel, Pomeransky, Alex, Rosenschein, Jeffrey. Búsqueda de un plan alternativo. Bar-Ilan University, Israel 2002.

(Fikes & Nilsson 71)

Fikes, R. E. & Nilsson, N. J. STRIPS: A new approach to the application of theorem proving to problem solving. Artificial Intelligence, 2, 1971

(Gellerman 81)

Gellerman, William. Group Leadership and Decision Making. [http:// www.uhc-collective.org.uk/toolbox](http://www.uhc-collective.org.uk/toolbox)

(Gibb 84)

Gibb, Jack R. Manual de Dinámica de grupos. Ed humanitas. Buenos Aires 1984

(Goldman & Boddy 94)

Goldman, R. P. & Boddy M. S.. Conditional linear planning. In proceedings of the second International conference on Artificial Intelligence Planning Systems. AAAI Press. Chicago, IL. 1994.

(Grosz & Kraus 96)

Barbara J. Grosz, Sarit Kraus. "Collaborative Plans for Complex Group Action." *Artificial Intelligence* vol. 86(2) (1996). pp. 269–357.

(Kumar et al 00)

Kumar, Sanjeev, Huber, Marcus J., Mcgee, David R. M., Cohen, Philip R., Levesque, Hector J. L. Semantics of Agent Communication Languages for Group Interaction. In *Proceedings of Aaai 2000 Conference*. Auton Tex. 2000.

(Kambhampati 00)

Subbarao Kambhampati *AI Planning and Scheduling (AIPS) Conference 2000*

(Knuth 72).

Knuth, Donald E. *The art of Computer Programming*. Adisson-Wesley Publishing Company. 1972

(Marakas 06)

Marakas, George M. *Decision Support Systems in the Twenty-First Century*. Prentice Hall 2006

(Myers 03)

Myers, K. L. and Jarvis, P. A. and Tyson, W. M. and Wolverton, M. J. *Mixed-initiative Planning in PASSAT*, in *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, 2003.

(Ossowski y García-Serrano 97)

Ossowski, Sascha & García-Serrano, Ana. Social Co-ordination among autonomous problem-solving agents. In *Proceedings of Agentes and Multi-Agents Systems, AI '97 Workshops on Commonsense Reasoning, Intelligent Agents and DAI*. Springer. Australia, 1997

(Parsons 98)

Simon Parsons *Argumentation and Multi-Agent Decision Making*  
*Proceedings of the AAAI Spring Symposium on Interactive and Mixed-Initiative Decision Making*. 1998

**(Pednault 1988 )**

Pednault, E. P. D. 1988. Extending conventional planning techniques to handle actions with context-dependent effects. In [Proceedings of the Seventh National Conference on Artificial Intelligence](#), pp. 55-59 St Paul, MN. AAAI.

(Pryor & Collins 96)

Pryor, Louise and Collins, Gregg. Planning for contingencies: A decision-based Approach. *Journal of Intelligence Research* 4 (1996). UK.

(Riemsdijk et al 02)

Riemsdijk Birna van. Computer Science Utrecht University The Netherlands. Agent Programming in Dribble from Beliefs to Goals using Plans *AAMAS'03*, 2003,

(Russell 03)

Russell, Stuart and Norvig, Peter. *Artificial Intelligence: A Modern Approach 2nd Edition*. New Jersey: Pearson Education, 2003.

(Sacerdoti 77)

Sacerdoti, E. A Structure for Plans and Behavior. Elsevier North-Holland Inc. 1977.

(Sacerdoti 98)

Earl D. Sacerdoti. THE NONLINEAR NATURE OF PLANS. Artificial Intelligence Center. Stanford Research Institute. Menlo Park California U.S.A. 1998

(Talukdar et al 98)

[Sarosh Talukdar](#), [Lars Baerentzen](#), [Andrew Gove](#), [Pedro De Souza](#),. Asynchronous Teams: Cooperation Schemes For Autonomous Agents (1998)

(Tate 77)

Tate, A. 1977. Generating Project Networks. IJCAI 1977. University of Edimburgh.

(Tsamardinos et al 00)

Tsamardinos, Ioannis, Pollack Martha E., Horty John F. Merging Plans with Quantitative Temporal Constrains, Temporally Extended Actions and Conditional Branches. American Association for Artificial Intelligence ([www.aaai.org](http://www.aaai.org)) 2000.

(Veloso y Carbonell 95)

Jaime Carbonell, Alicia Perez, Daniel Borrajo, Eugene Fink, Jim Blythe. Integrating Planning and Learning: The PRODIGY Architecture (1995) Veloso, Journal of Experimental and Theoretical Artificial Intelligence

(Vere 83)

Vere, S.A.. Planning in Time: Windows and Durations for Activities and Goals. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. PAMI-5. No. 3

(Wilkins 01)

Wilkins, D. E. and desJardins, M. *A Call for Knowledge-based Planning*. AI Magazine, vol. 22, no. 1, pp. 99-115, 2001.

(Wilkins 84)

Wilkins, D. E. . Practical Planning: Extending the Classical AI Planning Pradigm. Morgan Kaufmann, San Mateo, CA. 1984

(Yang 97)

Yang Qiang. A theory of conflict Resolution in Planning. Technical Report University of Waterloo 1997

(Yang 97a)

Yang, Q. Intelligent Planning: A Decomposition and Abstraction Based Approach. Springer. NY. 1997.