INVESTIGACION OPERATIVA

INTRODUCCION

Estructura del Curso

1. Introducción y modelamiento: 2 semanas
2. Programación Lineal: 2 semanas
3. Programación Lineal Entera: 1 semana
4. Teoría de Grafos: 2 semanas
5. Flujo en Redes: 2 semanas
6. Aplicaciones al mundo real de IO: 5 semanas

Evaluaciones

- 3 parcialitos (Modelamiento, PL y PLE, Grafos y Redes)
- 3 TP's
- Examen Final

Introducción

Qué es la Investigación de Operaciones?

- Enciclopedia Británica: “Es la aplicación de métodos científicos a la adminstración y gestión de organizaciones gubernamentales, industriales, comerciales y militares.”
- Gass (1983): “Es la ciencia de la toma de decisiones.”

Área de Aplicación Fundamental: Gestión en Diversas Organizaciones.

La I.O. se apoya en diferentes disciplinas científicas, pertenecientes la mayoría al ámbito de la Matemática Aplicada.

Una de las disciplinas principales de la I.O. es la Programación Matemática.

Problema Básico de la Programación Matemática

"Encontrar el mejor valor de alguna medida de desempeño (llamada Función Objetivo), siempre que las variables de decisión cumplan ciertas restricciones."

1. Problemas del Estilo

$$\max f(x)$$

$$\text{s. a. } x \in S$$

con S es cualquier conjunto, por ejemplo $S = [1, 8]$. Gráficamente:
Ej:

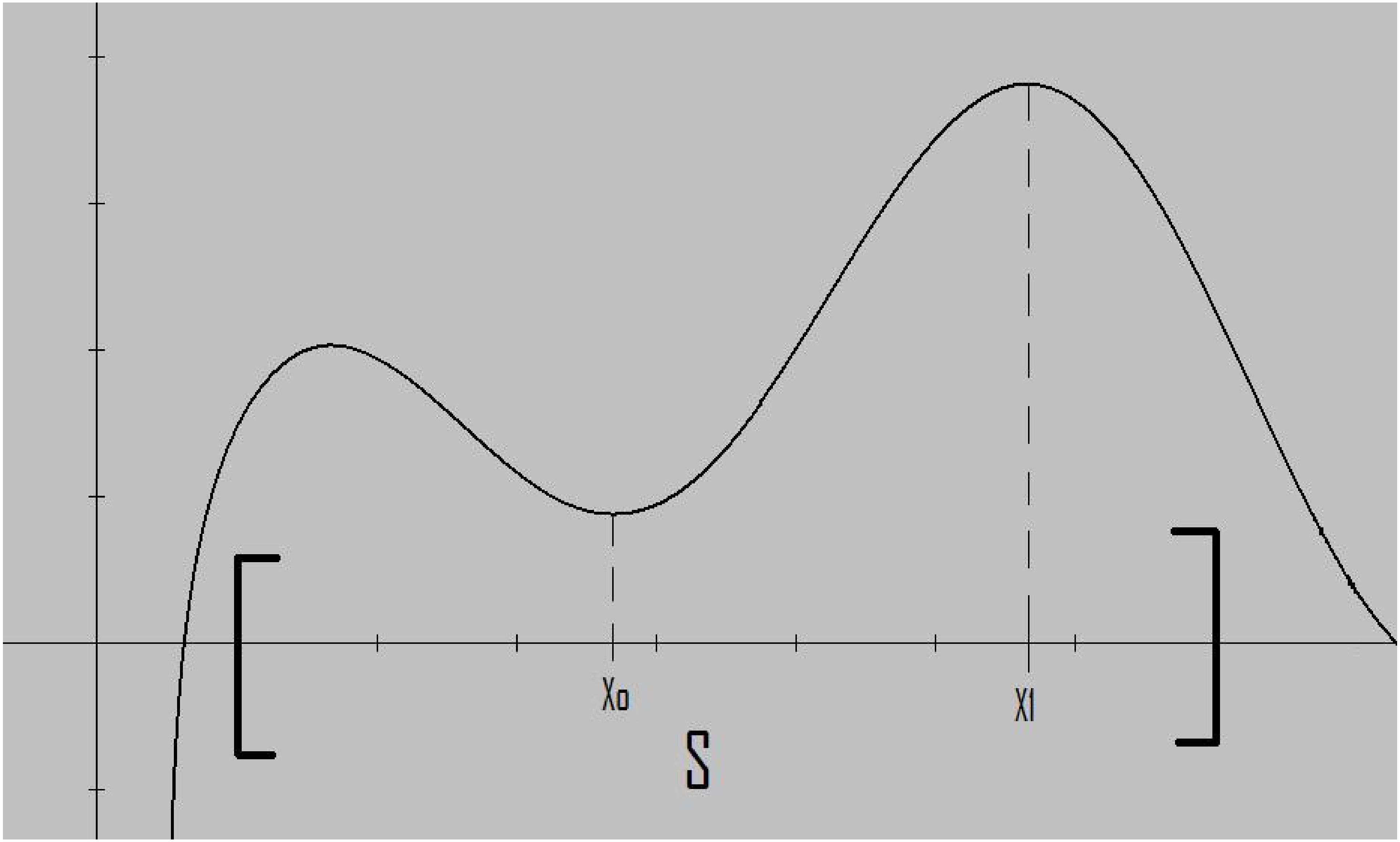


Figura 1: función f

Algo de Historia

- Siglo XVIII: Primeras formalizaciones matemáticas de este tipo de problemas realizadas por Fourier y Lagrange.
- 1945: El trabajo de Von Neumann da origen a la Programación Lineal.
- Segunda Guerra Mundial: La guerra obliga a solucionar problemas muy grandes, destinados a resolver la asignación óptima de recursos y la correcta puesta en práctica de la logística. Varios de estos problemas se engloban en lo que hoy conocemos como "problemas de transporte".
- Década del 50: Dantzig desarrolla un algoritmo para problemas de programación lineal generales: el método SIMPLEX.

Metodología para enfrentar un problema con el enfoque de I.O.

1. Definición del Problema
2. Construcción del Modelo
3. Resolución del Modelo
4. Validación del Modelo
5. Implementación y Control del Modelo

1. Definición del Problema

a) Identificar el ámbito del sistema en estudio
- Identificar componentes y relaciones existentes entre ellas.
- Definir límites o fronteras del sistema.
- Definir el medio ambiente y las posibles influencias de éste sobre el sistema.

b) Establecer los objetivos del sistema
- Proponer diferentes cursos de acción conducentes a obtener el mejoramiento deseado.
- Identificar las decisiones que son tomadas en el ámbito del sistema y que pueden ser modificadas.
- Cuantificación de los objetivos específicos a fin de poder evaluar la eficiencia y/o eficacia de las soluciones propuestas.

c) Identificar las alternativas de decisión
- Generación de alternativas de decisión.
- Seleccionar un número reducido de las alternativas más efectivas.
- Definir lineas gruesas de acción consistentes con los objetivos del estudio.

Para los puntos a y b es importante considerar los objetivos de la organización solicitante del estudio y de los participantes involucrados en las decisiones consideradas.

Aspectos a tener en cuenta en la definición del problema

a) **Dimensión Espacial del Sistema:** Límites entre el sistema y el marco de referencia.

Ej: Producción

- Una planta de la empresa
- La totalidad de la empresa
- El sector productivo nacional

b) **Dimensión Temporal del Sistema u Horizonte:** Período para el cual se hará el diseño y establecimiento de unidades temporales dentro del período.

Ej: Planificación de Producción

- Período: 1 mes
- Unidades Temporales: 1 día

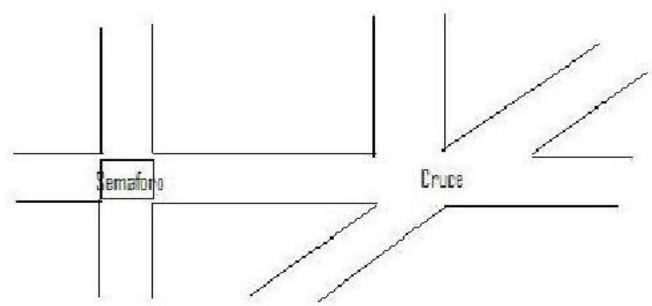
  Notar que los subperíodos en los cuales se divide un período no necesariamente deben ser de la misma extensión que las unidades temporales.

Aspectos a tener en cuenta en la definición del problema

c) **Nivel de las Decisiones:** Decisiones que aborda el estudio.
Ej:

- Estratégicas ⟶ Largo Plazo ⟶ Agregar una planta
- Operativas ⟶ Corto Plazo ⟶ Programación de máquinas

d) **Separabilidad de las Decisiones:** Grado de interrelación que existe entre las diferentes instancias abordadas en el estudio.
Cuando es posible separar componentes de un sistema?
Cuando el nivel de influencia de las reacciones a las decisiones tomadas en una componente es débil.
Ej:



La congestión causada por el semáforo afecta el cruce.
Conclusión: No estudiar la congestión en el cruce independiente del semáforo.

Aspectos a tener en cuenta en la definición del problema

e) **Grado de Precisión Numérica:** Error que se está dispuesto a aceptar en los valores numéricos de la solución.

$$\text{Error en los datos} \longrightarrow \text{Error en las soluciones.}$$

f) **Tiempo y Recursos Humanos Disponibles:** Plazos establecidos para el estudio y características del equipo de trabajo.

g) **Tecnología Disponible:** Metodologías con las que se cuenta y características de los equipos computacionales.

Ej: Metodologías
- Software Comercial
- Software Desarrollado Ad-Hoc

Nota: Todos los aspectos anteriores están fuertemente interrelacionados.

2. Construcción del Modelo

Característica de la I.O.: Utilización de modelos matemáticos en la resolución de los problemas.

Modelo $\longrightarrow$ Aproximación de la realidad

La calidad de un modelo depende de la percepción, creatividad, intuición, conocimiento e imaginación del modelador.

En la práctica no pueden considerarse todos los factores y todas las relaciones existentes entre ellos, por lo que se seleccionan los aspectos más relevantes.

Elementos Característicos de un Modelo Matemático:

- **Variables:**
  - **Variables de Decisión o Endógenas:** Decisiones cuantificables abordadas por el estudio, cuyos valores se intenta determinar por medio de la resolución del modelo.
  - **Variables Exógenas o Parámetros:** Decisiones que han sido tomadas fuera del ámbito del sistema. Desde el punto de vista del modelo son datos.
  - **Variables de Estado:** Variables que caracterizan la situación en la que se encuentra el sistema en un instante dado. Su valor depende de las variables de decisión y los parámetros.
- **Restricciones:** Limitaciones que deben imponerse al sistema.
- **Medida de Efectividad:** Criterio que se utilizará para comparar distintas opciones.

Restricciones y medidas de efectividad expresan las relaciones entre las variables.

Tipos de Modelos

1. **Modelos de Programación Matemática u Optimización**
   - Resolver un modelo de optimización implica determinar los valores de las variables de decisión de modo que se cumplan las restricciones y se optimice la función objetivo.
   - Existen muchas situaciones donde las características de las decisiones no permiten construir un modelo matemático adecuado. Otras veces el grado de complejidad y el tamaño del modelo pueden hacer muy dificultosa su resolución.

2. **Teoría de Juegos**
   - Se utilizan cuando en el sistema considerado existen varios componentes que toman decisiones interrelacionadas. El modelo permite evaluar una alternativa propuesta por uno de los actores considerando la participación del resto. Ej: Una empresa que quiere lanzar al mercado un nuevo producto (las empresas de la competencia son los otros actores).
   - Esta herramienta permite comprender el funcionamiento de sistemas complejos.
   - Aplicaciones en el campo de la Economía (más detalles en el curso in41a).

Tipos de Modelos

3. **Simulación**

   - Permite replicar (en un computador) el comportamiento del sistema bajo condiciones dadas.
   - Los resultados vienen dados por un conjunto de indicadores de desempeño que permiten evaluar el comportamiento del sistema.
   - El espíritu no es optimizar sino reproducir el comportamiento del sistema.
     Ej: Call Center y diferentes alternativas de cambios.

4. **Prototipos**

   - Opera directamente con el sistema real bajo estudio.
   - Es más ajustado pero también más costoso que un modelo de simulación.
     Ej: Call Center y el análisis acerca de un cambio realizado.

Tipos de Modelos

Modelo Matemático T. de Juegos Simulación Prototipo Sistema Real

$\longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow \longleftarrow$

Crece Grado de Abstracción

La selección del tipo de modelo que se ha de utilizar depende de la complejidad de la situación planteada, del tiempo y de la tecnología disponible para su resolución.

3. Resolución del Modelo

Qué es resolver un modelo?

Es encontrar los valores de las variables de decisión de manera que se cumplan las restricciones y se optimice la función objetivo.

Muchas veces esto puede ser sencillo porque sólo consiste en aplicar un paquete computacional comercial.

Otras veces puede ser necesario desarrollar metodologías especializadas de solución.

Tipos de Soluciones

1. **Soluciones Analíticas:** El valor de las variables de decisión se obtiene por medio de operaciones algebraicas.

2. **Soluciones Numéricas:** El valor de las variables de decisión se obtiene por medio de procedimientos, generalmente iterativos.

Procedimientos para Encontrar Soluciones Numéricas

- **Algoritmos Exactos:** Obtiene soluciones óptimas a través de iteraciones.

- **Heurísticas:** Procedimientos iterativos que tratan de determinar buenas soluciones pero no garantizan optimalidad. Comúnmente utilizadas ante la convergencia muy lenta de un algoritmo exacto.

Además de la solución del modelo, es conveniente proveer información acerca del comportamiento de éste ante variaciones pequeñas de los parámetros, esto se logra a través del **Análisis de Sensibilidad**

4. Validación del Modelo

Es necesario verificar que una solución es razonable.
Un modelo puede resultar no válido debido a:

- Se omitieron variables o relaciones relevantes.
- Se incluyeron variables o relaciones no relevantes.
- Se consideraron valores de los datos estimados incorrectamente, etc.

Se suele partir analizando la solución para instancias pequeñas del problema, si los resultados son coherentes se pasa al análisis de soluciones para instancias más grandes.

Para los sistemas que están operando una forma de validar es una prueba retrospectiva, es decir, estudiar el modelo con datos de períodos anteriores.

Para los sistemas que no están operando pueden utilizarse datos obtenidos de un modelo de simulación.

5. Implementación y Control del Modelo

Implementación:
Una vez desarrollado el estudio y validado el modelo, las soluciones deben transformarse en herramientas de apoyo a la toma de decisiones, esto implica:

- Establecer procedimientos, manuales y/o computacionales, al interior de la organización que aseguren la disponibilidad de los datos que el modelo requiere.

- Establecer procedimientos que permitan, una vez que se tienen los datos, resolver el modelo y obtener la solución. Esto corresponde fundamentalmente a la puesta en marcha del paquete computacional utilizado.

- Establecer procedimientos que permitan transformar las soluciones en acciones específicas en la organización.

$$\text{Datos} \longrightarrow \text{Algoritmos} \longrightarrow \text{Acciones}$$

5. Implementación y Control del Modelo

Control:
Es necesario establecer un sistema de control que permita detectar cualquier desviación de los supuestos. Esto permitirá efectuar oportunamente las modificaciones que correspondan cuando los cambios sean significativos.

Es importante notar que la solución del modelo **debe** ser analizada por el tomador de decisiones, quien podrá modificarla incorporando en la decisión final elementos que no se han considerado en el modelo.

Principales Causas del Fracaso de Aplicaciones de I.O.

1. Falta de integración e interacción entre el equipo de desarrollo y las personas que estarán a cargo de la operación del sistema en la organización. Para evitar esto es importante la participación del usuario/operador durante el desarrollo del proyecto.

2. Problemas de diseño, particularmente en la interfaz computacional con el usuario. Para evitar esto es necesario tener un medio fácil, rápido y amistoso para ingresar los datos del modelo y para interpretar sus resultados.

Disciplinas de la I.O.

1. **Programación Matemática u Optimización:**
   Incluye un conjunto de conceptos y técnicas para abordar el problema de determinar los valores de las variables de decisión de un modelo, de modo que se obtenga el mejor valor de una medida de rendimiento y se cumplan las restricciones impuestas en el modelo.
   El hecho de incorporar o no la incertidumbre en los modelos identifica dos áreas específicas bien diferenciadas: Programación Matemática Estocástica y **Programación Matemática Determinística**

2. **Programación Dinámica:**
   Técnica que aborda la resolución de los problemas identificando estructuras dinámicas entre las variables de decisión. Por ejemplo, problemas que describen la evolución en el tiempo de un sistema cuyas condiciones en un período dependen del estado del sistema en períodos anteriores.

Disciplinas de la I.O.

3. **Teoría de Colas:**
   Se ocupa de técnicas y conceptos que permiten abordar problemas cuyo modelo básico es el de una cola o fila de individuos que esperan ser atendidos por un servidor.
   Ej: Call Center, Bancos, etc.

4. **Simulación:**
   Disciplina que permite el estudio de sistemas complejos mediante el uso del computador con el objeto de reproducir su comportamiento.

5. **Teoría de Juegos:**
   Aborda problemas donde dos o más agentes se enfrentan a situaciones de conflicto e intentan alcanzar cierto estado de equilibrio.

El curso se centra fundamentalmente en los problemas de tipo 1, de tipo determinístico. Por el final del curso se abordarán problemas de tipo 2.

Impacto de la I.O. en la Gestión Moderna

- Se ha cuestionado la verdadera utilidad de la I.O. ya que muchos de los grandes modelos desarrollados en las décadas de los 60 y 70 no pudieron ser resueltos pues los recursos computacionales resultaron inficientes.

- Hoy el gran desarrollo de herramienas computacionales y la introducción de nuevos métodos de solución exacta (o al menos aproximada) ha permitido resolver problemas que hace 30 o 40 años eran intratables.

- Aplicaciones Interesantes:
  - Líneas Aéreas
    - Sistemas de Reservaciones
    - Asignación de Flotas y Tripulación
  - Logística
    - Problemas de Despacho
    - Ruteo de Vehículos

Impacto de la I.O. en la Gestión Moderna

- Aplicaciones realizadas desde la Universidad de Chile:
  - Sistemas de despacho de camiones aplicados a la industria forestal
  - Licitación de alimentos para comedores escolares
  - Sistema de optimización de la planificación de largo plazo en minería subterránea y de rajo.
  - Diseño del fixture del fútbol chileno, desde el año 2005.
  - Seleccion de postulantes a un Master de la Universidad con restricciones de equidad.
  - Optimización de la logística y la producción en empresas salmoneras chilenas.

- Aplicaciones realizadas desde la Universidad de Buenos Aires:
  - Diseño del fixture de la Liga de voley en las últimas 3 temporadas.
  - Distribución eficiente de censistas para el censo de la Provincia de Buenos Aires que se realizará en octubre de este año.
  - Ruteo de camiones para optimizar la recolección de basura en la zona sur de la ciudad de Buenos Aires.
  - Modelos matemáticos para gestionar la licitación de Internet en las escuelas públicas de la ciudad de Buenos Aires.

Problema General de Optimización

- **Variables de Decisión:** $x_1, \ldots, x_n$
- **Restricciones:** Se debe cumplir que $(x_1, \ldots, x_n) \in S \subseteq R^n$, el conjunto S se denomina **conjunto de soluciones factibles**.
- **Función Objetivo:** Corresponde a la medida de efectividad $f : R^n \longrightarrow R$

Luego, el problema a resolver es:

$$\begin{gathered} \text{mín(o máx)} f(\overrightarrow{x}) \\ \text{s. a. } \overrightarrow{x} \in S \end{gathered}$$

Con $\overrightarrow{x} = x_1, \ldots, x_n$.

Si $S = R^n$ se dice que el problema es irrestricto. Si no es un problema con restricciones.

Problema General de Optimización

Si se tiene un problema con restricciones se representará a S por un conjunto finito de desigualdades, es decir:

$$\min f(\overrightarrow{x})$$
$$\text{s. a. } g_i(\overrightarrow{x}) \leq 0, \text{ con } i = 1, \ldots, m$$

Donde cada $g_i : R^n \longrightarrow R$ representa alguna característica o condición que se desea incorporar.

Clasificación de los Problemas

- **Problemas Lineales:** f y g_i son funciones lineales (de la forma $a^T x + b$, con $a \in R^n$ y $b \in R$). Dentro de este tipo de problemas tenemos:
  - **Problemas Enteros:** Todas las variables toman valores enteros.
  - **Problemas Binarios:** Todas las variables pueden tomar solo uno de dos valores, en general 0 o 1 (este caso es un caso particular del punto anterior).
  - **Problemas Mixtos:** Algunas variables deben ser enteras y otras pueden tomar cualquier valor en los reales.
- **Problemas No Lineales:** Al menos una restricción g_i o la función objetivo f es no lineal. Existen distintas variantes de estos problemas que se verán más adelante.

Clasificación de los Problemas

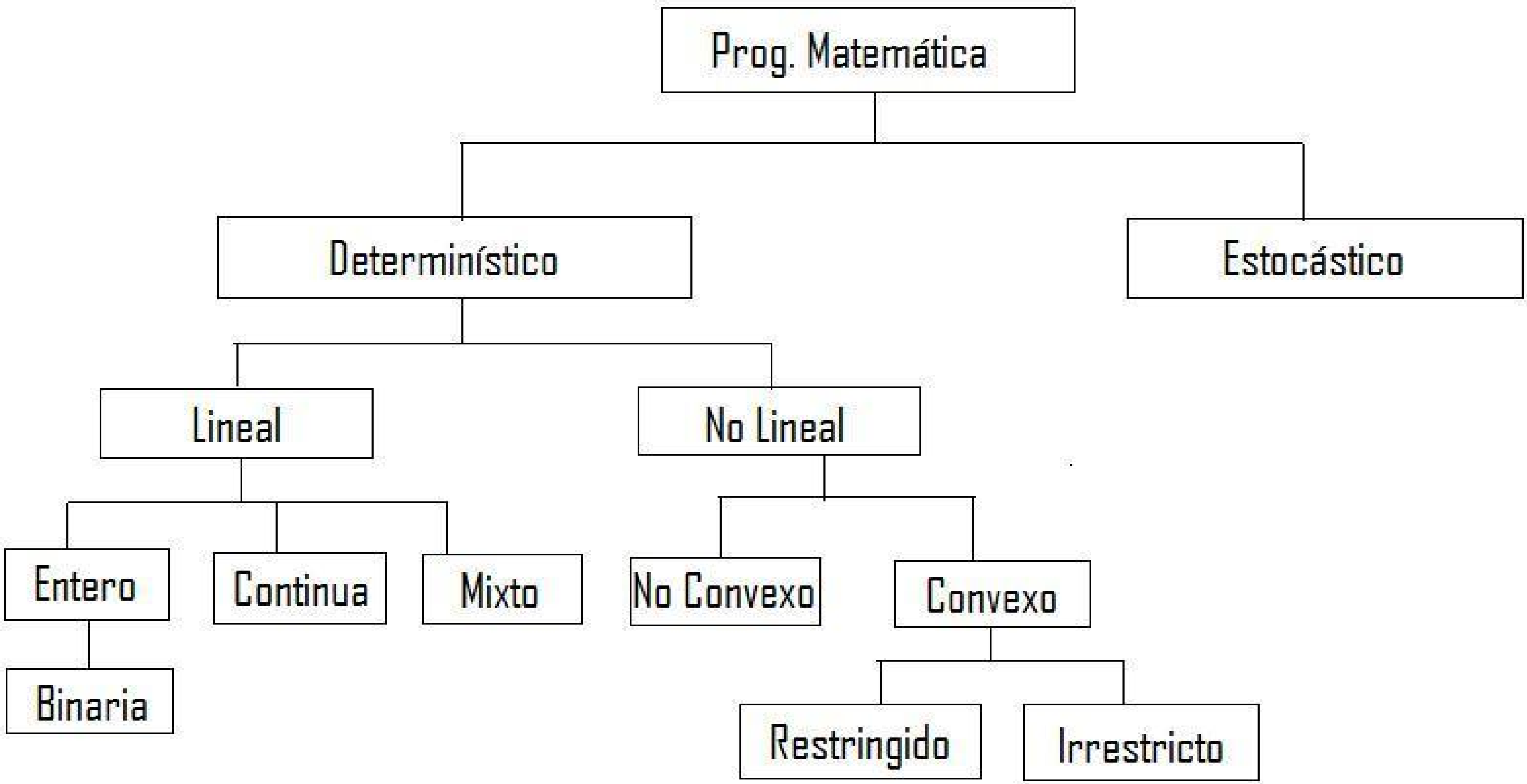


Figura 2: Tipos de Problemas

Métodos, Algoritmos y Heurísticas

La solución de un problema de optimización se encuentra aplicando procedimientos computacionales denominados métodos, algoritmos o heurísticas.

Una **instancia** de un problema se obtiene especificando valores particulares para todos sus parámetros.
Ej: Sea el problema $\min f(x) = ax^2 + bx + c$
Una instancia del problema se obtendrá fijando los valores de a, b y c.

1. **Método**
   Secuencia de pasos, que puede ser finita o infinita, cuyo objetivo es determinar la solución de un problema.
   Ej: Determinar una raíz de $g(x) = x^2 - 2$
   Usamos el método iterativo de Newton-Raphson:

$$x_{k+1} = x_k - \frac{g(x_k)}{g'(x_k)}$$

   A partir de un $x_0 \neq 0$.

Métodos, Algoritmos y Heurísticas

Cuando el número de iteraciones tiende a infinito el método garantiza convergencia (en este caso a $\sqrt{2}$).

Como obviamente no se alcanza la solución exacta, se modifica el método buscando un x tal que $|x^2 - 2| \leq \varepsilon$, donde $\varepsilon > 0$ es dado como dato.

2. **Algoritmo**

   Secuencia finita de pasos que garantiza el encontrar una solución de un problema para una instancia dada.

   Ej:

   - Algoritmo de Gauss para invertir matrices.
   - Algoritmo de Euclides para determinar máximo común divisor
   - Algoritmo de Newton-Raphson para encontrar una raíz aproximada de una funcio diferenciable.

Métodos, Algoritmos y Heurísticas

3. **Heurística**
   Procedimientos que en un tiempo $razonable$ dan una $buena$ aproximación a la solución del problema.

   Se han desarrollado para problemas que no se pueden resolver con algoritmos, o para problemas cuyos algoritmos se demoran mucho en resolverlos.

   Ej: Vendedor Viajero
   - Algoritmos Exactos
     - Fuerza Bruta: $\frac{(n-1)!}{2}$ posibles tours
     - Algoritmos más inteligentes
   - Heurística Greedy

Complejidad de Problemas y Algoritmos

La dificultad de un problema está relacionada con la estructura del problema y con el tamaño de la instancia.

Para indicar el tamaño de un problema se identifican uno o más parámetros que permitan tener una idea del tamaño de la instancia considerada.

Se suele estudiar el número de operaciones aritméticas elementales (comparaciones, sumas, multiplicaciones) en función del tamaño de la instancia que un algoritmo realiza para resolver un problema.

Teoría de la Complejidad:(Cook 1971, Karp 1972)
Disciplina que estudia formalmente los temas anteriores. Está basada en conceptos de Ciencias de la Computación y de Matemática.
Dado los alcances del curso, se hará una sencilla introducción respecto a este tema.
Para un estudio más profundo se debe partir por el modelo de computador conocido como Máquina de Turing (por Alan Turing, uno de los fundadores de la Teoría de Computación).

Complejidad de un Algoritmo

Se define como el número máximo de operaciones que realiza el algoritmo para determinar la solución de una instancia de tamaño n del problema:

1. Si un algoritmo que ordena una lista de elementos de tamaño n realiza $k \cdot n^2$ operaciones (k una constante) diremos que el algoritmo tiene complejidad $O(n^2)$.
   Observación: El mejor algoritmo posible para ordenar tiene complejidad $O(n \log_2 n)$.
2. Si un algoritmo simple que multiplica 2 matrices de tamaño $n \times n$ realiza $k \cdot n^3$ operaciones diremos que tiene complejidad $O(n^3)$.
   Observación: El mejor algoritmo posible para multiplicar 2 matrices tiene complejidad mejor a $O(n^3)$ (pero esto excede el marco de este curso).

La complejidad computacional de un algoritmo se denota $O(f(n))$ (orden de $f(n)$) donde f es la relación entre el número de operaciones y el tamaño de la instancia.

La notación $O(f(n))$ indica que el número de operaciones está acotado por $k \cdot f(n)$, donde k es una constante independiente de n.

Complejidad de un Algoritmo

Análisis del Peor Caso: Dado que existen muchas instancias del mismo tamaño y no todas requieren del mismo número de operaciones, se establece una cota superior de la complejidad de un algoritmo para un tamaño de instancia.

Algoritmo Polinomial: El número de operaciones está acotado por un polinomio en el tamaño del problema. Estos algoritmos se consideran eficientes.

Algoritmo Exponencial: El número de operaciones no está acotado por un polinomio. Estos algoritmos se consideran ineficientes.

Complejidad \ Tamaño	10	30	50	60
n	0,00001	0,00003	0,00005	0,00006
n^3	0,001	0,027	0,125	0,216
2^n	0,001	17,9 min	35,7 años	366 siglos
3^n	0,059	6,5 años	2×10^8	$1,3 \times 10^{13}$

Complejidad de un Problema

La complejidad de un problema viene dada por la complejidad del **mejor algoritmo** de entre todos los que lo resuelven. Así, un algoritmo particular da una **cota superior** para la complejidad de ese problema.

Se puede determinar una **cota inferior** de la complejidad de un problema probando matemáticamente que cualquier posible algoritmo deberá tener, por lo menos, la complejidad establecida por la cota inferior.

Si ambas cotas coinciden tenemos determinada la complejidad del problema.
Ej: Ordenar un arreglo

- Heap-Sort: $O(n \log_2 n)$ (cota superior)
- Existe una prueba de que al menos hace falta $O(n \log_2 n)$ tiempo para ordenar el arreglo (cota inferior)

$\therefore$ El problema tiene complejidad $O(n \log_2 n)$

Tipos de Problema

La teoría de Complejidad está diseñada para problemas de decisión, donde la respuesta es si o no.

En los problemas de optimización el objetivo consiste en encontrar una estructura que satisfaga algún criterio de optimización.

Ej: El vendedor viajero y sus dos variantes.

- **Optimización:** Encontrar un tour óptimo.
- **Decisión:** Dado un grafo G y un entero k, existe en G algún tour de longitud $\leq k$.

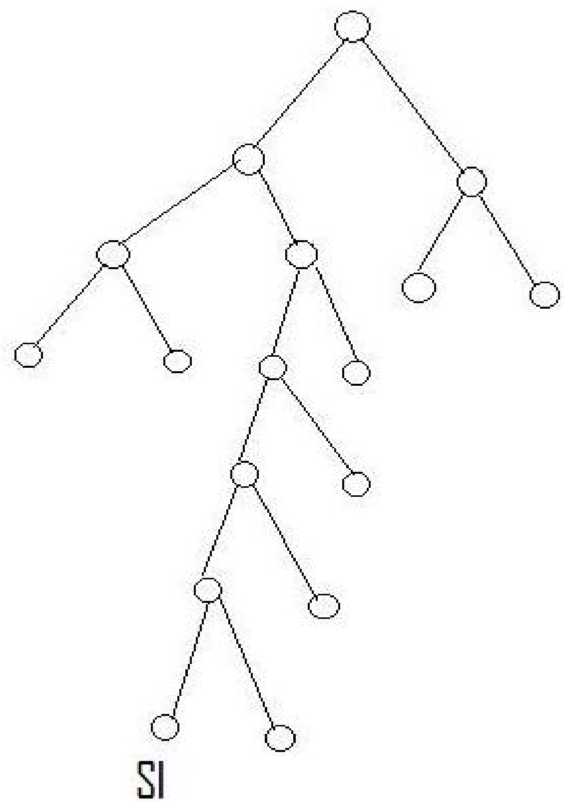
Tipos de Problema

- **P:** Problemas en los cuales existe un algoritmo polinomial que los resuelve (problemas fáciles).
- **NP:** (non-deterministic polinomial) Problemas en los cuales existe un algoritmo no determinístico que lo resuelve en tiempo polinomial.

  Como entender esto?

  1. Supongamos una máquina que tiene la propiedad de dividirse en varias copias idénticas de sí misma cada vez que se ve enfrentada a una alternativa, y que las copias continúan trabajando en paralelo. Representemos este proceso como un árbol de búsqueda:

Tipos de Problema



Si la altura es polinomial $\longrightarrow$ NP

- Si una de las copias responde afirmativamente tengo resuelto mi problema de decisión.
- Si el tiempo máximo que se requiere para recorrer una rama está acotado polinomialmente, el problema está en NP.

Tipos de Problema

2. Si alguien me da una solución para una instancia cuya respuesta es SI de mi problema y yo puedo verificar en tiempo polinomial que esa solución es correcta, el problema está en NP. (certificado polinomial)
   Ej: Vendedor Viajero.

Observación: P $\subseteq$ NP

Entre los problemas NP existe una subclase formada por los problemas difíciles aún no resueltos eficientemente, los **problemas NP-completos**.

Un problema A es **NP-completo** si:

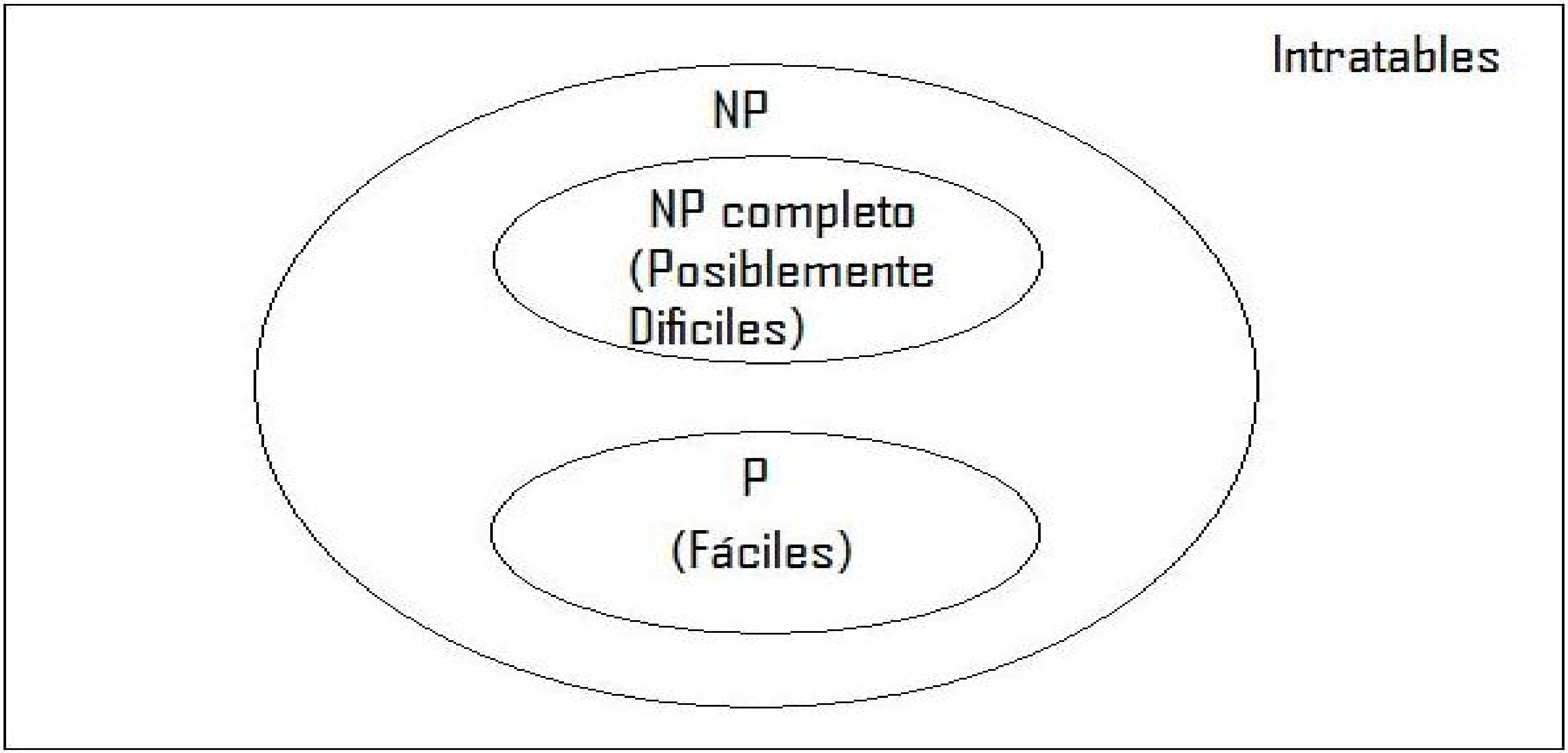
1. $A \in$ NP
2. Todo problema en NP se puede transformar polinomialmente a A

Esto signfica que si encuentro un algoritmo polinomial para un problema NP-completo, automáticamente se pueden resolver polinomialmente todos los problemas de la clase NP.

Tipos de Problema

Conjetura: P $=$ NP ó P $\neq$ NP?

En resumen:


Intratables
NP
NP completo
(Posiblemente
Dificiles)
P
(Fáciles)

Ejemplos de Complejidad de Problemas

- **P:**
  - Ordenar un arreglo
  - Ruta más corta entre 2 puntos
  - Calcular el determinante de una matriz
  - Programción Lineal
- **NP-completo:**
  - Vendedor Viajero
  - Programación Lineal Entera
- **Intratables:**
  - Determinar todos los puntos enteros que satisfacen un sistema de desigualdades lineales
  - Las Torres de Hanoi

Problema del Análisis del Peor Caso

En la práctica muchos algoritmos con mala complejidad teórica (por el peor caso) tienen un comportamiento $promedio$ aceptable, por ello, actualmente existen otros enfoques que analizan complejidad promedio (usando algún modelo estocástico apropiado).

Modelamiento Lineal Continuo

1. **Pérdida de Material en Proceso de Corte:**
   Una industria que fabrica papel y lo distribuye en rollos debe determinar la mejor forma de realizar el proceso de corte. Los rollos de papel que se producen tienen un ancho de 100 cm y un largo fijo. Los clientes demandan rollos de 30 cm, 45 cm y 50 cm de ancho. Al cortar los rollos de 100 cm se puede incurrir en pérdida de material. Se tiene un pedido de 800 rollos de 30 cm de ancho, 500 rollos de 45 cm y 1000 rollos de 50 cm.
   Se desea determinar la forma de efectuar el corte de manera que se satisfaga la demanda y se minimice la pérdida total de material.
   Existen 6 alternativas diferentes de corte de un rollo de 100 cm de ancho que tienen sentido en este caso:
   - Esquema 1: 3 cortes de 30 cm cada uno $\longrightarrow$ Pérdida $=$ 10 cm
   - Esquema 2: 1 corte de 30 cm y uno de 45 cm $\longrightarrow$ Pérdida $=$ 25 cm
   - Esquema 3: 2 cortes de 45 cm cada uno $\longrightarrow$ Pérdida $=$ 10 cm
   - Esquema 4: 1 corte de 45 cm y uno de 50 cm $\longrightarrow$ Pérdida $=$ 5 cm
   - Esquema 5: 2 cortes de 50 cm cada uno $\longrightarrow$ Pérdida $=$ 0 cm
   - Esquema 6: 1 corte de 30 cm y uno de 50 cm $\longrightarrow$ Pérdida $=$ 20 cm

Modelamiento Lineal Continuo

Solución:

Para modelar este problema se hará según un **problema general de optimización**:

- **Variables de Decisión**

  x_i: Cantidad de rollos de 100 cm de ancho que se cortarán según el esquema de corte i $(i = 1, \ldots, 6)$

- **Restricciones**

  a) Satisfacción de la Demanda:
  - Rollos de 30: $3x_1 + x_2 + x_6 = 800$
  - Rollos de 45: $x_2 + 2x_3 + x_4 = 500$
  - Rollos de 50: $x_4 + 2x_5 + x_6 = 1000$

  Nota: se puede colocar "$\geq$.en vez de "$=$"

  b) No Negatividad de las Variables

$$x_i \geq 0,\ i = 1, \ldots, 6$$

Modelamiento Lineal Continuo

c) Integralidad de las Variables
No hace falta. Estudiamos el problema lineal continuo y luego rendondeamo
Como los valores son grandes, el redondeo está dentro del error permitido.

- **Función Objetivo**
Minimizar la cantidad de rollos que se pierden:

$$\min z = 10x_1 + 25x_2 + 10x_3 + 5x_4 + 20x_6$$

Observación: Situaciones similares a este caso se pueden encontrar en la industria textil y de confección, en la industria de vidrios o madera, en la siderúrgica, etc. (se presenta siempre que las dimensiones del producto que se fabrica sean diferentes de las dimensiones que se requieren)

Modelamiento Lineal Entero

2. **El Problema de la Mochila:**
Se tienen n tipos diferentes de objetos, cada uno de ellos tiene un peso w_j y un valor v_j.
Se dispone de una mochila que soporta un peso máximo W, donde estos objetos deben ser colocados, de manera de maximizar el valor total del contenido de la mochila, sin exceder la capacidad de ésta.
Los objetos son indivisibles, por lo que solo se pueden colocar en la mochila cantidades enteras de un tipo de objeto.

Solución:

- **Variables de Decisión**
x_j: Unidades del objeto j que se que se ponen en la mochila $(j = 1, \ldots, n)$.

Modelamiento Lineal Entero

- **Restricciones**

a) Capacidad de la Mochila:

$$\sum_{j=1}^{n} w_j x_j \leq W$$

b) Naturaleza de las Variables:

$$x_j \in \mathbb{N},\ j = 1, \ldots, n$$

- **Función Objetivo**

$$\max z = \sum_{j=1}^{n} v_j x_j$$

Este problema corresponde al problema de la mochila (knapsack) entero.

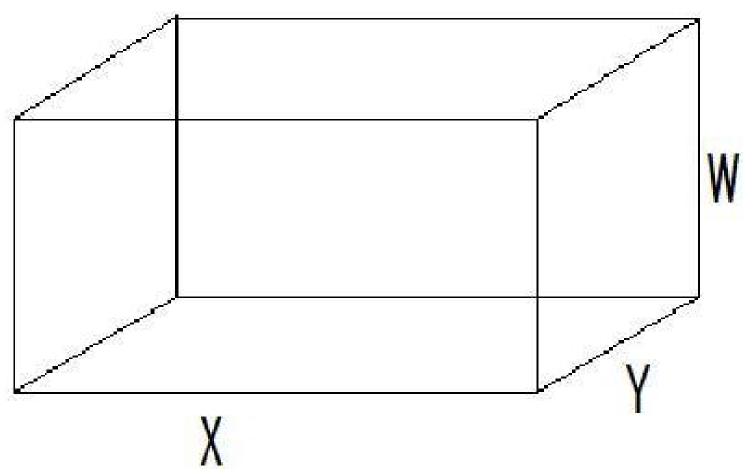
Modelamiento Lineal Entero

Observaciones:

- Una variación de este problema es cuando existe solo un objeto de cada tipo, e este caso las variables x_j toman valores 0 (no lo pongo) o 1 (lo pongo), este tipo d problemas se llaman $knapsack\ binario$.
- Algunas aplicaciones de este tipo de problemas: mercaderías que deben ser almac nadas o transportadas considerando una disponibilidad de espacio o de peso limitad (cada mercadería tiene un valor).
- Este problema se puede resolver en tiempo $O(nW)$ usando programación dinámic Esto es eficiente sólo si W está acotado por un polinomio en n.

Modelamiento No Lineal Restringido

3. **Diseño de Embalajes:**
   Una empresa exportadora debe enviar cada mes cierta cantidad del producto que se desarrolla a sus clientes en el exterior.
   El producto es embalado en cajas rectangulares, donde x corresponde al largo, y al ancho y w a la altura de la caja:



   Se quiere determinar la dimensión de las cajas que minimice el costo de armado de las mismas.

Modelamiento No Lineal Restringido

Supongamos que se dispone de hasta 60 m^2 del material necesario para armar el fondo, la cara frontal y la cara posterior de todas las cajas. El material de las otras 2 caras y la tapa deben comprarse a un costo de $5 el metro cuadrado.
Por otro lado, existe un costo de transporte de $3 por caja.
El volumen del producto que se quiere transportar es V.

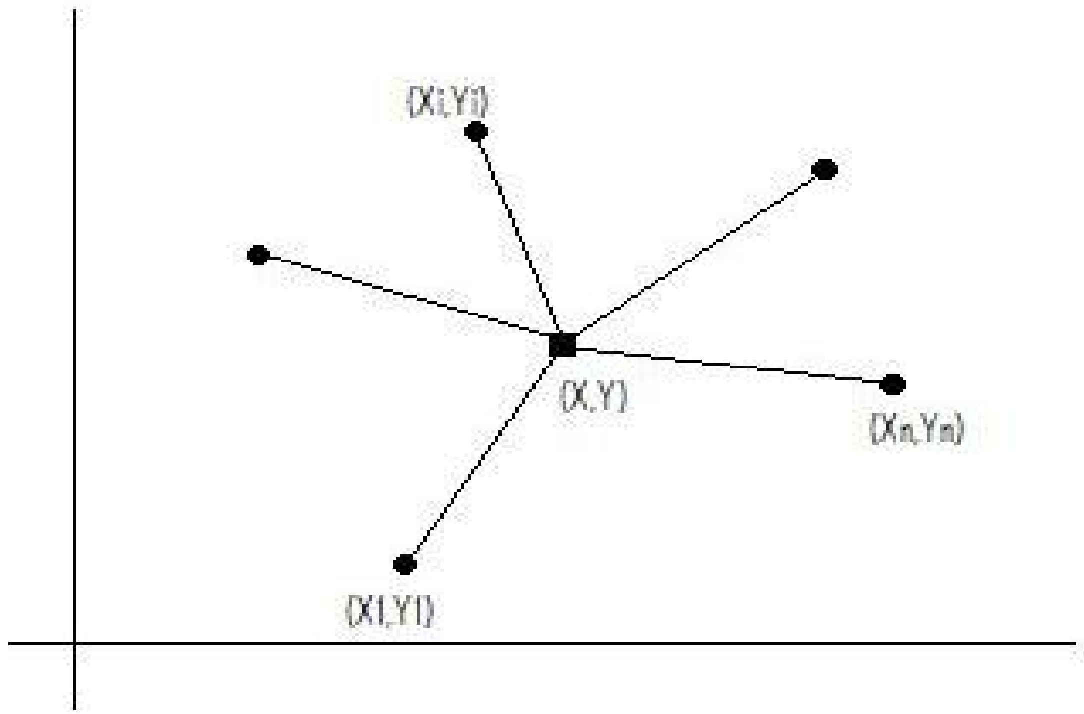
Solución:

$$\begin{gathered} \text{mín}\, z = 3Q + 10wyQ + 5xyQ \\ \text{s.a. } Qwxy = V \text{ ó } Qwxy \geq V \\ Q(2wx + xy) \leq 60 \\ w, x, y, Q \geq 0 \end{gathered}$$

Observación: La integralidad de Q puede ser obviada redondeando el resultado final (otra vez el error debería estar dentro de los valores permitidos).

Modelamiento No Lineal Irrestricto

4. **Localización de una Planta:**
   Una empresa cuenta con n centros de distribución de sus productos. Se desea instalar una planta que abastezca a los centros de tal forma que la distancia total hacia ellos sea mínima:



Solución:

Sean (x, y) las coordenadas de la planta.

$$\min z = \sum_{i=1}^{n} [(x - x_i)^2 + (y - y_i)^2]^{\frac{1}{2}}$$

Este es un problema de programación no lineal sin restricciones.