

---

# Desensamblado automático no destructivo para la reutilización de componentes. Aplicación al desensamblado de PC's

Santiago Timoteo Puente Méndez

## Tesis de Doctorado

**Facultad:** Escuela Politécnica Superior

**Directores:** Dr. Fernando Torres Medina  
Dr. Rafael Aracil Santonja

**2002**

---



Universitat d'Alacant  
Universidad de Alicante

Departament de Física, Enginyeria de Sistemes i Teoria del Senyal  
Departamento de Física, Ingeniería de Sistemas y Teoría de la Señal

Programa  
INFORMÁTICA APLICADA (2998)

# Desensamblado automático no destructivo para la reutilización de componentes. Aplicación al desensamblado de PC's.

TESIS DOCTORAL

Autor:

D. Santiago Timoteo Puente Méndez

Directores:

D. Fernando Torres Medina

D. Rafael Aracil Santonja

Alicante, 2002

A mi padre y mi madre

# Agradecimientos

Agradezco a mis directores, Fernando Torres y Rafael Aracil, la ayuda y guía que me han prestado para llevar a buen término esta tesis doctoral, así como para mejorar y superarme como investigador.

Mi reconocimiento a los miembros del departamento de Física, Ingeniería de Sistemas y Teoría de la Señal de la Universidad de Alicante y muy especialmente, al director del departamento Augusto Beléndez.

Inestimable ha sido la colaboración prestada por los miembros del Grupo de Automática, Robótica y Visión Artificial: Francisco Candelas, Jorge Pomares, Pablo Gil y Francisco Ortiz.

Quiero dar las gracias a mis padres, Abilio y Magdalena, por darme su apoyo incondicional en tantas horas dedicadas a la realización de esta tesis, así como a mi hermano Abilio, que ha compartido conmigo ideas, sugerencias y muchas sesiones de fructífero trabajo.

Por último, y no por ello menos importante, deseo mostrar mi gratitud a los amigos que en todo momento me han animado, principalmente durante los últimos períodos de realización de mi tesis doctoral.



# Índice general

[Agradecimientos](#)

[Resumen](#)

[Abstract](#)

[Índice general](#)

[Índice de figuras](#)

## 1 Introducción

1.1. Contexto

1.2. Marco de la tesis

1.3. Estructura de la tesis

## 2 Planificación de secuencia de desensamblado

2.1. Introducción

2.2. Estado del arte

2.2.1. Grafos

2.2.2. Características geométricas

2.2.3. Árboles

2.2.4. Redes de Petri

2.2.5. Otros métodos

2.2.5.1. Redes neuronales

2.2.5.2. Sistemas expertos

2.2.5.3. Algoritmos genéticos

2.2.6. Comparativa

- 2.3. Modelo relacional
    - 2.3.1. Definiciones y representación
      - 2.3.1.1. Definiciones
      - 2.3.1.2. Representación
  - 2.4. Acciones de desensamblado
  - 2.5. Estrategias de planificación
    - 2.5.1. Desensamblado total de un producto
    - 2.5.2. Desensamblado parcial de un producto
      - 2.5.2.1. Desensamblado de un componente
      - 2.5.2.2. Desensamblado de un montaje
  - 2.6. Ejemplo de planificación
    - 2.6.1. Desensamblado total de un producto
    - 2.6.2. Desensamblado parcial de un producto
      - 2.6.2.1. Desensamblado de un componente
      - 2.6.2.2. Desensamblado de un montaje
  - 2.7. Evaluación del algoritmo
- 3 Generador de movimientos de desensamblado**
- 3.1. Introducción
  - 3.2. Representación geométrica del producto
    - 3.2.1. Estado del arte
    - 3.2.2. Representación utilizada
  - 3.3. Determinación de la dirección de separación de un componente
    - 3.3.1. Estado del arte
      - 3.3.1.1. Esfera unidad
      - 3.3.1.2. Representaciones cinemáticas
    - 3.3.2. Conjunto de direcciones de separación
    - 3.3.3. Determinación de una dirección de separación
  - 3.4. Generación de una trayectoria de desensamblado
    - 3.4.1. Estado del arte
      - 3.4.1.1. Búsqueda automática de trayectorias
      - 3.4.1.2. Simulación interactiva
    - 3.4.2. Solución utilizada
  - 3.5. Detección de colisiones
    - 3.5.1. Planteamiento utilizado
      - 3.5.1.1. Restricciones debidas al movimiento del componente

- 3.5.1.2. Detección de colisiones mediante un problema de programación lineal
- 3.6. Cálculo de la trayectoria de desensamblado de una fuente de alimentación
- 3.7. Evaluación del algoritmo
- 4 Fusión de la planificación y del generador de movimientos
  - 4.1. Introducción
  - 4.2. Fusión de la planificación con el generador de movimientos
  - 4.3. Fusión considerando un robot
    - 4.3.1. Introducción de una herramienta
    - 4.3.2. Introducción del robot completo
    - 4.3.3. Utilización de diversas herramientas
  - 4.4. Desensamblado virtual
  - 4.5. Desensamblado real
- 5 Resultados: Aplicación al desensamblado de PC's
  - 5.1. Base de datos
    - 5.1.1. Productos
    - 5.1.2. Componentes
    - 5.1.3. Enlaces
    - 5.1.4. Herramientas
  - 5.2. Secuencia de desensamblado de un ordenador completo
  - 5.3. Desensamblado de un PC genérico
  - 5.4. Arquitectura del sistema de desensamblado flexible
  - 5.5. Desensamblado de PC's
    - 5.5.1. Desensamblado de un PC semitorre
    - 5.5.2. Desensamblado de un PC sobremesa
- 6 Conclusiones
  - 6.1. Aportaciones
  - 6.2. Trabajo futuro
- Anexo A Entorno desarrollado
  - A.1. Funcionalidades
    - A.1.1. Añadir componentes
    - A.1.2. Añadir elementos de enlace
    - A.1.3. Añadir uniones
    - A.1.4. Añadir elementos a un montaje nuevo
    - A.1.5. Añadir elementos a un montaje existente

A.1.6. Añadir montajes a otro montaje

A.2. Secuencia de desensamblado

A.3. Simulación de la secuencia de desensamblado

Anexo B      [Cinemática del brazo robot utilizado](#)

Anexo C      [Posicionamiento del producto](#)

[Bibliografía](#)

# Índice de figuras

Figura 1.1: Interacción entre las distintas actividades del ciclo de vida de un producto (Fuente [92]).....	2
Figura 1.2: Componentes que necesitan un desensamblado destructivo para poder separarlos en la dirección indicada (Fuente [243]).....	4
Figura 1.3: Respuesta a los problemas medio ambientales (Fuente [92]).	5
Figura 1.4: Arquitectura del proceso de desensamblado automático. ....	7
Figura 2.1: PC desensamblado parcialmente y su grafo de componentes-enlaces (Fuente: [300]).....	14
Figura 2.2: Árbol de desensamblado para el grafo de la Figura 2.1 (Fuente: [300]).....	14
Figura 2.3: Representación de un producto para generar un grafo de componentes-enlaces (Fuente: [308]).	15
Figura 2.4: Grafo de relaciones, grafo a nivel de producto y grafo del producto (Fuente: [308]).	15
Figura 2.5: Producto y grafo dirigido correspondiente (Fuente: [214]).	16
Figura 2.6: Estructura y grafo and/or de un tubo de rayos catódicos (Fuente: [206]).	16
Figura 2.7: Superficies de contacto del producto ensamblado (Fuente: [243]).	17
Figura 2.8: Grafo de topología del producto ensamblado de la Figura 2.7 (Fuente: [243]).	18
Figura 2.9: Transformación de un polígono no convexo en convexo (Fuente: [243]).	18
Figura 2.10: Método de propagación de ondas, ondas desde el componente $c_9$ (Fuente: [248]).	18

Figura 2.11: Restricciones geométricas para calcular el grafo and/or (Fuente: [89]).	19
Figura 2.12: Componentes de una picadora eléctrica y árboles para el desensamblado (Fuente: [180]).	20
Figura 2.13: Producto y su red de Petri de desensamblado (Fuente: [310]).	21
Figura 2.14: Base de datos de un sistema experto después de la introducción de varios productos (Fuente: [202]).	22
Figura 2.15: Componentes de diversos productos.	25
Figura 2.16: Elementos de enlace.	25
Figura 2.17: Componentes que requieren un desensamblado destructivo.	26
Figura 2.18: Unión de cierre generada por un tornillo.	26
Figura 2.19: Unión de cierre mediante rosca.	26
Figura 2.20: Unión de contacto entre dos chapas.	27
Figura 2.21: Componente con distintos niveles de granularidad.	27
Figura 2.22: Ejemplos de montajes. a) es indistinto desensamblar primero $m_i$ y después $m_j$ que al revés. b) hay que desensamblar primero $m_i$ antes de poder desensamblar $m_j$ .	29
Figura 2.23: Acción 1, unión de cierre entre componente y montaje, precedencia entre montajes. a) Grafo de ejemplo con tres componentes y una estructura jerárquica entre los montajes. b) Representación compacta del grafo de este tipo de acción.	30
Figura 2.24: Componentes que implican una acción de tipo 1.	30
Figura 2.25: Acción 2a, unión de contacto entre dos componentes. a) Grafo de ejemplo. b) Representación compacta del grafo de este tipo de acción.	31
Figura 2.26: Acción 2b, unión de contacto entre más de dos componentes. a) Grafo de ejemplo. b) Representación compacta del grafo de este tipo de acción.	31
Figura 2.27: Componentes que implican una acción de tipo 2. Componentes juntos y componentes separados.	32
Figura 2.28: Acción 3, montajes sin precedencia entre ellos. a) Grafo de ejemplo con tres componentes y dos montajes sin precedencia. b) Representación compacta del grafo de esta acción.	32
Figura 2.29: Componentes que implican una acción de tipo 3.	33
Figura 2.30: Acción 4, unión de cierre entre montaje y elemento de enlace. a) Grafo de ejemplo con un montaje y un elemento de enlace. b) Representación compacta del grafo de este tipo de acción.	33
Figura 2.31: Componentes que implican una acción de tipo 4.	33
Figura 2.32: Acción 5, montajes y/o componentes unidos por un elemento de enlace. a) Grafo de ejemplo con tres componentes, un montaje y un elemento de enlace que une un componente con el montaje. b) Representación compacta del grafo de esta acción.	34

Figura 2.33: Componentes que implican una acción de tipo 5.....	34
Figura 2.34: Acción 6, montajes unidos por un elemento externo cuyo orden de desensamblado es intercambiable. a) Grafo de ejemplo con tres componentes, dos montajes y un elemento de enlace que une los dos montajes. b) Representación compacta del grafo de esta acción.....	35
Figura 2.35: Componentes que implican una acción de tipo 6.....	35
Figura 2.36: Grafo parcial de un ordenador con sus montajes.....	38
Figura 2.37: Grafo dirigido con las acciones a realizar para desensamblar totalmente el ordenador de la Figura 2.36.....	41
Figura 2.38: Grafo dirigido con las acciones a realizar para desensamblar el componente "tarjeta 1" del ordenador de la Figura 2.36.....	43
Figura 2.39: Grafo del producto para desensamblar el montaje $m_1$ . Primer paso del algoritmo. ....	43
Figura 2.40: Grafo del producto para desensamblar el montaje $m_1$ . Segundo paso del algoritmo. El grafo definitivo del producto considerando como componente "montaje <sub>1</sub> " al montaje $m_1$ . ....	44
Figura 2.41: Grafo dirigido con las acciones a realizar para desensamblar el montaje $m_1$ ("montaje1") del ordenador de la Figura 2.36.....	47
Figura 3.1: Representación de un componente 2-desensamblable. ....	50
Figura 3.2: Disco duro y su representación simplificada mediante polígonos. ....	51
Figura 3.3: Representación de un componente utilizando elementos finitos (Cortesía de HEC). ....	51
Figura 3.4: Modelo poliédrico de un cilindro utilizando 5, 8, 14 y 26 polígonos. ....	52
Figura 3.5: Componente poliédrico $C_i$ de cuatro caras ( $P_1, P_2, P_3, P_4$ ) y el grafo de caras de adyacencia.....	53
Figura 3.6:Producto compuesto por tres componentes.....	53
Figura 3.7:Grafo de adyacencia para el componente $C_1$ del producto de la Figura 3.6.....	54
Figura 3.8:Grafo de caras de contacto para los componentes $C_1$ y $C_2$ del producto de la Figura 3.6.....	54
Figura 3.9:Grafo de caras de contacto para el producto de la Figura 3.6. ....	55
Figura 3.10: Ejemplo de aplicación de la técnica "freedom matrix". ....	57
Figura 3.11: Combinación de movimientos. a) Se necesitan dos movimientos lineales para desensamblar $c_2$ . b) Se necesita un movimiento de rotación y uno lineal para desensamblar el componente $c_2$ . ....	57
Figura 3.12: Los componentes $c_1$ y $c_4$ poseen una restricción que impide su movimiento en una determinada dirección (zona gris), pero no existe un contacto directo en esa dirección que indique la restricción.....	58
Figura 3.13: Representación del componente grupo residual para desensamblar el componente $c_1$ .....	58

Figura 3.14: Representación de los contactos entre las caras del producto....	59
Figura 3.15: Esfera gaussiana para un componente libre en el espacio tridimensional.....	59
Figura 3.16: Semiesfera que define las posibles direcciones de desensamblado de un componente que posee un contacto plano con otro. ....	60
Figura 3.17: Unión de las direcciones de desensamblado del componente $c_1$ para obtener el conjunto de direcciones de separación. ....	60
Figura 3.18: Probabilidad de colisión en un determinado conjunto de direcciones de desensamblado. ....	61
Figura 3.19: Discretización del entorno de trabajo en celdas cúbicas.....	61
Figura 3.20: Detección de colisiones e incremento del mundo discretizado....	63
Figura 3.21: Mundo discretizado tras un haz de rayos.....	63
Figura 3.22: Transformación a coordenadas esféricas. ....	64
Figura 3.23: Determinación de la dirección de separación a partir de la obtención de la máxima fuerza de repulsión.....	65
Figura 3.24: Técnicas de búsqueda automática de trayectorias (Fuente: [218]). ....	66
Figura 3.25: Secuencia del algoritmo para determinar la trayectoria de desensamblado de un componente en un producto formado por tres componentes . ....	70
Figura 3.26: Bucle al desplazar totalmente un componente. ....	71
Figura 3.27: Colisión de tipo-A entre dos objetos. ....	72
Figura 3.28: Colisión de tipo-B entre dos objetos. ....	73
Figura 3.29: Colisión de tipo-C entre dos objetos. ....	73
Figura 3.30: Espacio de búsqueda de colisiones, teniendo en cuenta las restricciones, por la dirección de desensamblado para la cara inferior de un componente. ....	75
Figura 3.31: Componentes sobre los que se aplica el problema de programación lineal. ....	78
Figura 3.32: Restricciones por los planos paralelos a la dirección de movimiento de la cara del componente móvil. ....	79
Figura 3.33: Restricciones por los planos paralelos a la dirección de movimiento de la cara del componente estático.....	79
Figura 3.34: Restricción por la cara del componente móvil. ....	80
Figura 3.35: Restricción por la cara del componente estático. ....	80
Figura 3.36: Zona del espacio donde se puede producir colisión y la colisión. ....	81
Figura 3.37: Modelo geométrico de la caracas interior y la fuente de alimentación.....	81
Figura 3.38: Identificación de las caras en la fuente de alimentación.....	82
Figura 3.39: Identificación de las caras en la carcasa interior.....	83



---

Figura 3.40: Grafo de relaciones.....	84
Figura 3.41: Direcciones de desensamblado para la fuente de alimentación..	84
Figura 3.42: Secuencia de desplazamiento de la fuente de alimentación según la dirección (0, -0.7071, 0.7071). ....	87
Figura 3.43: Coste del algoritmo de cálculo de colisiones según las caras existentes en el producto. ....	89
Figura 4.1: Entorno de trabajo el producto sobre la mesa de trabajo. ....	92
Figura 4.2: Entorno de trabajo formado por un robot y el producto sobre la mesa de trabajo. ....	93
Figura 4.3: Varias posibilidades de desensamblar el disco duro.....	96
Figura 4.4: Relación entre el robot y la mesa de trabajo. ....	97
Figura 4.5: Modelo geométrico de una pinza paralela de dos dedos. ....	98
Figura 4.6: Modelo geométrico de la herramienta enganchada al componente. ....	99
Figura 4.7: Necesidad de considerar el modelo completo del robot para realizar el desensamblado. ....	100
Figura 4.8: Esquema del sistema de desensamblado virtual. ....	102
Figura 4.9: Esquema del sistema de desensamblado real. ....	105
Figura 4.10: Ordenador con disquetera de 5¼ y sin disquetera de 5¼.....	106
Figura 5.1: Esquema genérico del modelo entidad-relación de la base de datos. ....	110
Figura 5.2: Tablas de la base de datos para el almacenamiento del modelo relacional de los productos. ....	111
Figura 5.3: Tablas de la base de datos para el almacenamiento de los componentes así como las herramientas necesarias para desensamblarlos. ....	113
Figura 5.4: Varios tipos de herramientas (Cortesía de Sckunk y Norgren). .	114
Figura 5.5: Ventosas de distintos tamaños (Cortesía de Norgren). ....	114
Figura 5.6: Modelo relacional de un ordenador completo.....	115
Figura 5.7: Ordenador completo utilizado para generar el modelo relacional de la Figura 5.6. ....	115
Figura 5.8: Grafo de tareas a realizar para desensamblar el disco duro del PC de la Figura 5.6. ....	119
Figura 5.9: Secuencia de desensamblado del disco duro del PC de la Figura 5.6; vista del entorno.....	119
Figura 5.10: Grafo de tareas a realizar para desensamblar el montaje $m_{15}$ de la Figura 5.6. ....	122
Figura 5.11: Secuencia de desensamblado del montaje $m_{15}$ del PC de la Figura 5.6, vista del entorno. ....	123
Figura 5.12: Modelo relacional genérico de un PC. ....	124

Figura 5.13: Modelo relacional extendido de un PC con una disquetera.....	125
Figura 5.14: Modelo relacional extendido de un PC; una tarjeta, dos periféricos y fuente de alimentación. ....	126
Figura 5.15: Arquitectura utilizada para el desensamblado. ....	127
Figura 5.16: Tipos de PC: semitorre y sobremesa.....	128
Figura 5.17: Sistema de visión adquiriendo información sobre el tipo de producto; imágenes de la cámara izquierda y derecha del par estéreo. .	129
Figura 5.18: Modelo relacional genérico de un PC.....	130
Figura 5.19: Modelo relacional extendido del PC semitorre con las tarjetas. ....	131
Figura 5.20: Grafo de acciones a llevar a cabo por el sistema para desensamblar el componente "tarjeta de vídeo". ....	132
Figura 5.21: Acciones a llevar a cabo por el sistema para desensamblar el componente "tarjeta de vídeo". ....	132
Figura 5.22: Secuencia simulada de desensamblado para la carcasa exterior de un PC semitorre. ....	133
Figura 5.23: Sistema de visión adquiriendo información sobre el tipo de producto, PC semitorre; imágenes de la cámara izquierda y derecha del par estéreo.....	134
Figura 5.24: PC semitorre con los componentes internos visibles. ....	135
Figura 5.25: Modelo relacional extendido del PC semitorre. ....	135
Figura 5.26: Modelo geométrico extendido del PC semitorre con los componentes visibles. ....	136
Figura 5.27: Secuencia simulada de la trayectoria de desensamblado del componente "tarjeta de vídeo" desde distintos puntos de vista. ....	137
Figura 5.28: Secuencia real de desensamblado del componente "tarjeta de vídeo".....	138
Figura 5.29: Modelo relacional del PC una vez desensamblado el componente "tarjeta de vídeo". ....	139
Figura 5.30: Grafo de acciones para desensamblar el componente "disco duro". ....	139
Figura 5.31: Acciones a llevar a cabo por el sistema para desensamblar el componente "disco duro". ....	140
Figura 5.32: Secuencia simulada de la trayectoria de desensamblado del componente "disco duro". ....	141
Figura 5.33: Secuencia real de desensamblado del componente "disco duro". ....	142
Figura 5.34: Sistema de visión adquiriendo información sobre el tipo de producto, PC sobremesa; imágenes de la cámara izquierda y derecha del par estéreo.....	143
Figura 5.35: Secuencia simulada de la trayectoria de desensamblado del componente "carcasa exterior" de un PC sobremesa (Parte 1). ....	144

---

Figura 5.36: Secuencia simulada de la trayectoria de desensamblado del componente "carcasa exterior" de un PC sobremesa (Parte 2). .....	145
Figura 5.37: Sistema de visión adquiriendo información sobre el tipo de producto, PC sobremesa; imágenes de la cámara izquierda y derecha del par estéreo (Parte 1).....	145
Figura 5.38: Sistema de visión adquiriendo información sobre el tipo de producto, PC sobremesa; imágenes de la cámara izquierda y derecha del par estéreo (Parte 2).....	146
Figura 5.39: PC sobremesa con los componentes internos visibles. ....	146
Figura 5.40: Modelo relacional extendido del PC sobremesa. ....	147
Figura 5.41: Grafo de acciones a llevar a cabo por el sistema, para desensamblar el componente "tarjeta de vídeo" del PC sobremesa. ....	148
Figura 5.42: Acciones a llevar a cabo por el sistema para desensamblar el componente "tarjeta de vídeo" del PC sobremesa.....	149
Figura 5.43: Modelo geométrico extendido del PC sobremesa. ....	149
Figura 5.44: Herramienta colisionando con el componente "fuente de alimentación". .....	150
Figura 5.45: Grafo de acciones a realizar para desensamblar el componente "fuente de alimentación" del PC sobremesa. ....	150
Figura 5.46: Modelo geométrico utilizado para la herramienta "ventosa"...	151
Figura 5.47: Secuencia simulada de la trayectoria de desensamblado del componente "fuente de alimentación" de un PC sobremesa (Parte 1)...	151
Figura 5.48: Secuencia simulada de la trayectoria de desensamblado del componente "fuente de alimentación" de un PC sobremesa (Parte 2)...	152
Figura 5.49: Producto sobre la mesa de trabajo en la posición original y tras trasladarlo y rotarlo para que el robot pueda desensamblar la "tarjeta de vídeo". .....	152
Figura 5.50: Secuencia simulada de la trayectoria de desensamblado del componente "tarjeta de vídeo" de un PC sobremesa. ....	153
Figura 5.51: Secuencia real de desensamblado del componente "tarjeta de vídeo" en el PC sobremesa. ....	154
Figura A.1: Entorno de desarrollo para la introducción de los modelos relacionales.....	160
Figura A.2: Elementos de un mando a distancia.....	160
Figura A.3: Diseño del modelo relacional de un mando a distancia; componentes. ....	161
Figura A.4: Diseño del modelo relacional de un mando a distancia; elementos de enlace.....	162
Figura A.5: Diseño del modelo relacional de un mando a distancia; uniones de cierre y de contacto. ....	163
Figura A.6: Diseño del modelo relacional de un mando a distancia; introducción de componentes en los montajes. ....	164

Figura A.7: Diseño del modelo relacional de un mando a distancia; modelo final. ....	165
Figura A.8: Secuencia de desensamblado de la carcasa exterior de un mando a distancia. ....	166
Figura A.9: Entorno de simulación, modelo de un PC sobremesa. ....	167
Figura B.1: Ejes de coordenadas del robot Scrobot ER-IX. ....	170
Figura B.2: Esquema del plano de trabajo del robot.....	172

# 1

## Introducción

### 1.1. Contexto

Durante los primeros años de la revolución industrial lo más importante era producir, sin preocuparse de los problemas medio ambientales que pudiera ocasionar un producto o su proceso de fabricación. Sin embargo, en los últimos años, la preocupación por el medio ambiente ha crecido y los gobiernos se ven en la necesidad de regular aspectos medio ambientales de la producción. Debido a ello la Comisión Europea introduce en 1998 un borrador para una directiva sobre la producción de residuos en productos eléctricos y electrónicos, *Waste Electrical and Electronical Equipment (WEEE)* [212], cuya última modificación se realizó en abril del 2002.

Estas normativas implican una fabricación que tenga en cuenta el medio ambiente, denominada en la bibliografía como *environmentally conscious manufacturing (ECM)*, lo que supone una política de producción que involucre todo el ciclo de vida del producto, desde la etapa de diseño, pasando por la entrega al consumidor, sin olvidarse del producto una vez que ha terminado su vida útil. Lo que se pretende con esta política es eliminar la cantidad de residuos generados al finalizar la vida útil de un producto así como causar un menor daño al medio ambiente. Mediante el reciclado y la reutilización de los productos o de algunas partes de ellos se consigue reducir la cantidad de residuos producidos.

Lo primero a considerar son las actividades en las que se puede dividir el ciclo de vida de un producto, en las cuales habrá que tomar consideraciones distintas para el tratamiento del producto (Figura 1.1). Las etapas son las siguientes [92]:

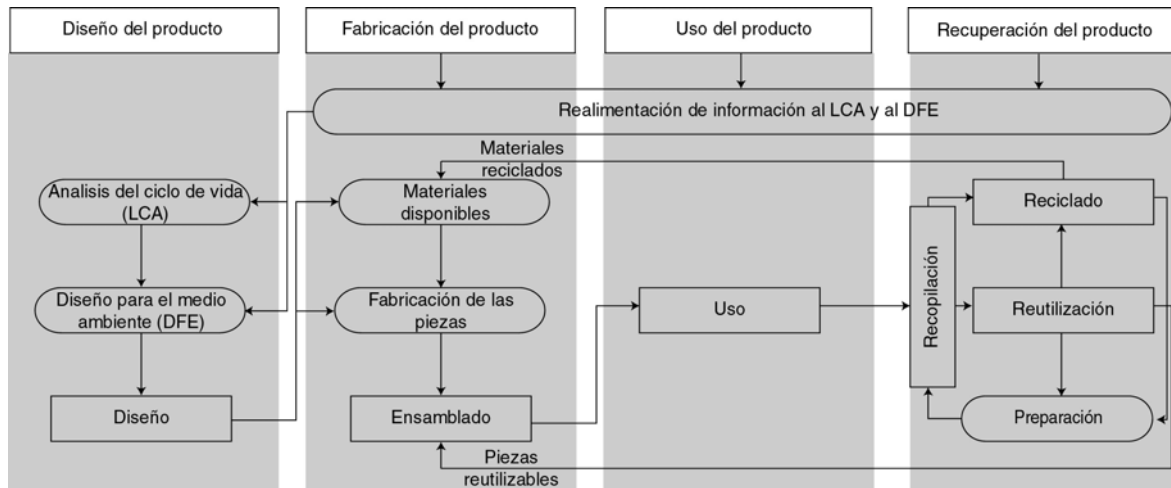


Figura 1.1: Interacción entre las distintas actividades del ciclo de vida de un producto (Fuente [92]).

- **Diseño del producto:** En esta etapa se realiza el diseño del producto en base a lo que se desea obtener y a la información que se pueda tener sobre los distintos componentes que formarán parte de él, teniendo en cuenta su impacto ambiental y su facilidad de reciclado y reutilización.
- **Fabricación del producto:** Esta fase es la encargada, como indica su nombre, de construir el producto propiamente dicho; teniendo en cuenta el diseño realizado se genera el producto para su uso.
- **Uso del producto:** Esta fase se corresponde con la utilización del producto por el usuario al que este destinado.
- **Recuperación del producto:** Es la última fase, y se aplica una vez que el producto ha dejado de ser utilizado. Consiste en el reciclado y la reutilización de las distintas partes del producto.

Este proceso no es en una única dirección, sino que, según se realizan las distintas etapas, éstas proporcionan información sobre el producto, que puede ser utilizada para realizar un mejor diseño de nuevas versiones del mismo así como de los nuevos productos, para que cada vez estén diseñados de una manera más consciente con el medio ambiente. Para ello se utiliza información para realizar un análisis del ciclo de vida del producto, denominado en la bibliografía como *life cycle analysis (LCA)*, que representa un estudio de todo lo que pasa con el producto una vez que ha sido diseñado [1] [3] [8] [13] [18] [22] [26] [37] [43] [49] [51] [57] [58] [69] [71] [72] [73] [96] [98] [99] [105] [111] [117] [118] [119] [120] [121] [122] [127] [130] [132] [141] [144] [149] [159] [183] [184] [197] [201] [221] [226] [228] [241] [253] [257] [261] [265] [283] [284] [285] [286] [294] [311].

La información recopilada también se utiliza para realizar nuevos diseños más conscientes con el medio ambiente, denominado en la bibliografía como *design for environment (DFE)*, para diseñar los nuevos productos teniendo en cuenta las ventajas e inconvenientes que se dan durante la vida del producto [7] [12] [16] [17] [19] [32] [33] [35] [36] [40] [47] [56] [63] [67] [68] [74] [75] [96] [101] [103] [104] [106] [120] [125] [139] [145] [151] [154] [157] [168] [169] [170] [179] [186] [187] [198] [215] [224] [226] [227] [230]

[231] [232] [234] [236] [237] [238] [239] [246] [251] [252] [261] [280]  
[295] [305] [310].

Como consecuencia de la gran cantidad de residuos generados por los productos de las industrias, así como de las diferentes formas en que pueden aparecer (residuos sólidos, polución en el aire y en el agua, etc.), se contemplan dos iniciativas principales a la hora de luchar contra la existencia de estos residuos. La primera consiste en crear productos que no dañen el medio ambiente, lo que se denomina "ingeniería verde". La segunda consiste en desarrollar técnicas para recuperar los componentes que forman los productos así como para tratar los residuos.

Para poder llevar a cabo un diseño del producto teniendo en cuenta una estrategia que no dañe el medio ambiente, hay que tener un conocimiento de todo el ciclo de vida del producto [253]. Para ello no sólo basta con realizar decisiones que consideren el medio ambiente en la fase de diseño, sino que durante todo el ciclo de vida, hay que tomar decisiones respecto a aspectos medio ambientales del producto [177] [178]. En este sentido se pueden señalar tres maneras de diseñar un producto según los objetivos que se persigan.

- La primera de ellas sería la realización de un diseño para el reciclado, denominado en la bibliografía como *design for recycling (DFR)*, que plantea unos objetivos para que el producto sea fácilmente reciclable.
- La segunda manera de enfocar el diseño consiste en realizar un diseño para el medio ambiente, conocido como *design for environment (DFE)*, que dañe lo menos posible minimizando los componentes peligrosos.
- La tercera manera consiste en realizar un diseño para el desensamblado, conocido como *design for disassembly (DFD)*, que plantea una estrategia en la cual la separación de los componentes del producto, una vez finalizado su período de vida útil, puedan separarse para su posterior reciclado o reutilización.

Estas tres estrategias no son incompatibles entre sí, sino que se pueden combinar para conseguir un mejor diseño del producto.

El análisis del ciclo de vida de un producto como se ha comentado resulta muy importante para evitar problemas medioambientales con los nuevos productos. Sin embargo hay que tener en cuenta que normalmente el mayor daño para el medio ambiente se da cuando un producto se deja de utilizar, termina su vida útil, es por esto que los procesos de recuperación de materiales al término de su vida útil son muy importantes.

En estos momentos resulta de vital importancia el reciclado de productos al final de su vida útil, ya que existen millones de unidades que han sido diseñados sin tener en cuenta al medio ambiente, con lo que al llegar éstos al final de su vida útil hay que llevar a cabo una de las opciones posibles, el reciclado o la reutilización.

El reciclado por su parte trata de recuperar los materiales utilizados durante la fabricación, mediante el desensamblado y la separación de los materiales, utilizando procesos químicos para su recuperación una vez

desensamblados del resto [27] [30] [38] [45] [76] [100] [102] [114] [115] [116] [122] [123] [137] [143] [166] [191] [193] [194] [195] [209] [217] [233] [242] [304] [310]. Estas técnicas implican el uso de desensamblado destructivo, ya que se puede dañar un componente para conseguir separar una determinada materia prima del resto del producto; es decir, se trata de desensamblar el producto, no separándolo en sus componentes individuales, sino separando los distintos materiales que lo forman.

La reutilización del producto intenta en lugar de recuperar los distintos materiales que lo componen, hacerlo con las piezas que lo forman, para ello se necesitan técnicas de desensamblado, así como para clasificar y restaurar las distintas piezas [1] [11] [15] [19] [29] [31] [34] [54] [55] [59] [60] [93] [108] [109] [113] [122] [138] [175] [192] [197] [203] [208] [217] [220] [254] [271] [274] [286]. Para poder llevar a cabo la reutilización de los componentes se requieren técnicas de desensamblado no destructivo ya que para volver a utilizar un componente se requiere que no se haya dañado durante su desensamblado.

El desensamblado es el proceso clave que permite la recuperación de las partes deseadas de un producto así como la separación de los materiales que lo forman. Esto implica una reducción en el tiempo necesario para fabricar nuevos productos gracias a la reutilización de las partes desensambladas en buen estado [20] [191], ya que así no se tienen que volver a fabricar esas partes.

Teniendo en cuenta estos criterios se define el tipo de desensamblado utilizado como: destructivo o no destructivo [124] [158]:

- Se denomina desensamblado destructivo a aquel en el que se debe romper algún componente, es decir si para realizar el desensamblado hay que deteriorar algún componente (Figura 1.2).

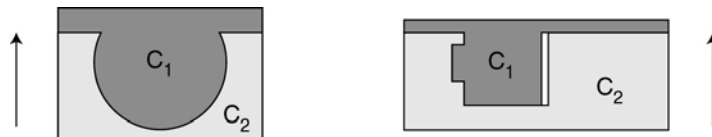


Figura 1.2: Componentes que necesitan un desensamblado destructivo para poder separarlos en la dirección indicada (Fuente [243]).

Un claro ejemplo de este tipo de desensamblado se presenta cuando se desea separar dos elementos unidos mediante una soldadura, es decir siempre que para acceder a un elemento hay que cortar otro que impide el paso se está trabajando con un desensamblado destructivo [64] [290].

- Se conoce como desensamblado no destructivo al que realiza un desensamblado sin dañar ni romper ningún componente, este tipo de desensamblado es el más indicado para tareas de mantenimiento ya que no se debe de dañar un componente para poder sustituir otro. También es más indicado para tareas que requieran una reutilización de los productos completos, ya que recupera los distintos elementos que lo forman sin dañarlos. Por el contrario no siempre es posible realizar el desensamblado completo de un producto mediante desensamblado no destructivo, especialmente si se trata de productos antiguos que no han sido diseñados para su



recuperación. El algoritmo que aporta esta tesis plantea una nueva solución al problema del desensamblado no destructivo.

Todos estos conceptos se encuentran relacionados entre sí, y son las soluciones que se plantean para los problemas medio ambientales que surgen (Figura 1.3).

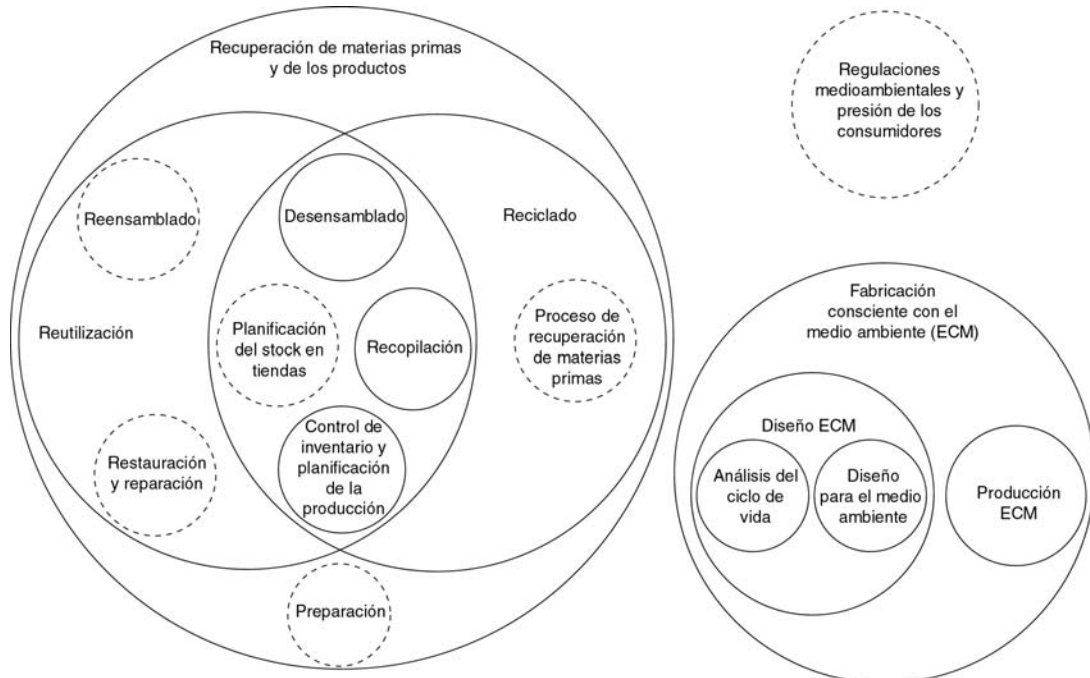


Figura 1.3: Respuesta a los problemas medio ambientales (Fuente [92]).

Además de realizar consideraciones a la hora de diseñar un producto, se deben también utilizar estrategias para la recuperación una vez utilizados, esta etapa en la bibliografía se denomina como *collection*, [41] [59] [60] [66] [70] [87] [108] [122] [128] [129] [134] [146] [153] [172] [219] [286].

Además, en base a estas técnicas, se deben generar estrategias para realizar un control del inventario de manera que se evite un exceso de producción que llevaría a un aumento de los residuos producidos [1] [11] [29] [31] [60] [113] [129] [138] [140] [142] [176] [208] [222] [223] [245] [254] [272] [273] [274].

Muy ligada a una correcta estrategia para el control del inventario está el control de la producción así como una correcta planificación de ella para poder tener un control del inventario apropiado [4] [20] [34] [40] [60] [77] [78] [79] [80] [81] [82] [83] [84] [85] [86] [94] [108] [109] [123] [131] [135] [138] [141] [142] [143] [147] [198] [203] [204] [205] [220] [229] [244] [254] [259] [260] [261] [264] [271].

Teniendo en consideración todas estas técnicas se plantean métodos para realizar un adecuado proceso de desensamblado de los productos, denominado en la bibliografía como *disassembly process planning (DPP)* [5] [6] [10] [28] [48] [88] [89] [90] [91] [93] [95] [110] [124] [133] [136] [152] [157] [160] [165] [185] [188] [189] [190] [200] [202] [213] [214] [240] [244] [248] [256] [261] [263] [268] [270] [275] [276] [277] [278] [279] [281] [290] [291] [292] [293] [296] [297] [298] [299] [304] [309] [310].

El objetivo perseguido con estas técnicas consiste en estudiar como desensamblar un producto completamente para aprovechar sus piezas, o como separar las piezas o materias primas para que resulte rentable su reciclado o reutilización.

## **1.2. Marco de la tesis**

La presente tesis se enmarca dentro de la búsqueda de sistemas de desensamblados flexibles, planteando un sistema para la planificación y realización de un proceso de desensamblado automático destinado a la reutilización de componentes empleando técnicas no destructivas.

Dentro del ámbito del reciclado y la "ingeniería verde", el desensamblado y más aún el desensamblado automático es un factor muy importante debido a las posibilidades que ofrece.

De cara a la reutilización de los componentes que forman los productos se tiene que el uso de técnicas adecuadas de desensamblado permite la correcta separación de los componentes sin que estos sufran daño.

Es por tanto, otro de los factores a tener en consideración para un sistema automático orientado a la reutilización, el llevar a cabo un desensamblado no destructivo, ya que si se utilizan técnicas destructivas se dañarían los componentes a recuperar y por tanto no se podrían utilizar para llevar a cabo una reutilización de los componentes.

Hay que destacar que un proceso de desensamblado no puede verse simplemente como el inverso del proceso de ensamblado sino que hay que ir más allá dado que pueden existir modificaciones, debido al uso, o cambios de los componentes que forman un producto.

Por todo ello se plantea esta tesis relacionada con el proyecto de investigación financiado por la CICYT (TAP 1999-0436), "Sistema robotizado de desensamblado automatizado basado en modelos y visión artificial".

La arquitectura que se plantea para la resolución del problema de desensamblado se muestra en la Figura 1.4.

Esta arquitectura se plantea teniendo en mente un sistema orientado a la reutilización de los componentes, es por ello que las técnicas que se van a plantear son técnicas no destructivas.

Según la arquitectura planteada se parte de la información sobre lo que se quiere desensamblar, bien sea un componente individual, un conjunto de ellos o el desensamblado total del producto.

Esta información se suministra al "planificador global", el cual considerando la información suministrada decide obtener el modelo genérico del producto correspondiente que se encuentra almacenado en la base de datos.

Dicho modelo se basa en las relaciones de jerarquía, modelo relacional, existentes entre los componentes, que permite definir la secuencia de acciones que se deben realizar para poder separar el componente deseado.

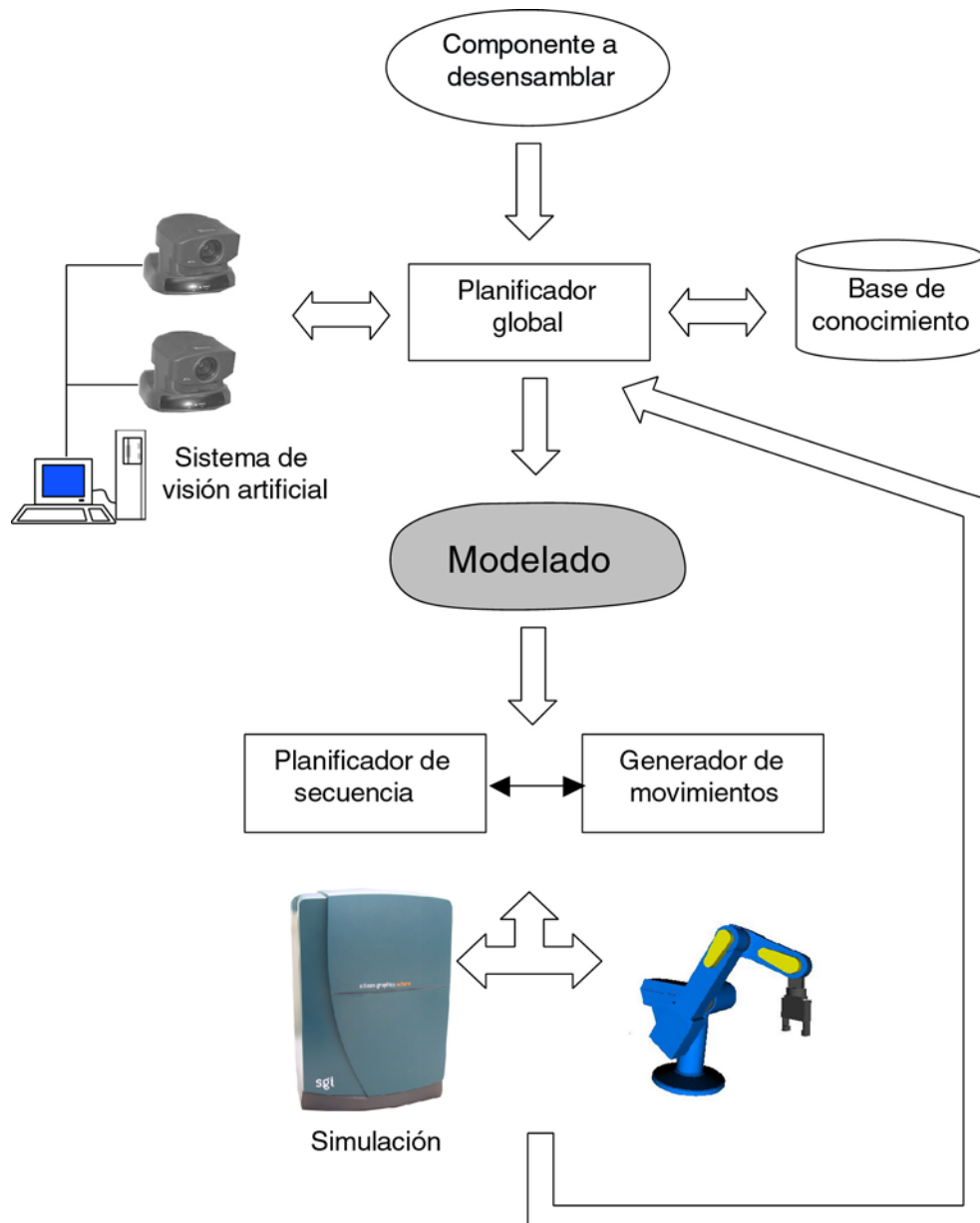


Figura 1.4: Arquitectura del proceso de desensamblado automático.

Por su parte el "sistema de visión artificial" se encarga de obtener, mediante un par estereoscópico información precisa sobre el producto que realmente se tiene para desensamblar. Este sistema debe ser capaz de reconocer los componentes del producto que se está desensamblando así como obtener su localización.

Para ello el "sistema de visión artificial" utiliza la información, contenida en la "base de conocimiento", sobre los modelos de los distintos componentes, así como información relativa a sus posibles ubicaciones en el producto.

Por tanto el planificador global utilizando un modelo relacional genérico del producto a desensamblar obtendrá un modelo preciso del producto que realmente se tiene para desensamblar, mediante el "sistema de visión artificial".

Este modelado que se realiza del producto se componen de dos partes: un modelo relacional y un modelo geométrico.

El modelo relacional es utilizado por el "planificador de secuencia" para generar la secuencia de operaciones a seguir para llevar a cabo la tarea especificada, teniendo en cuenta el modelo actual.

Esta secuencia se compone de la lista de componentes que hay que separar para poder alcanzar, y por tanto desensamblar, el componente objetivo.

El modelo geométrico se basa en un modelo poliédrico del producto. Este es utilizado por el "generador de movimientos" para calcular para cada uno de los componentes que se deben separar, información que proporciona el "planificador de secuencia", cuáles son los movimientos que tiene que seguir para separarse del resto del producto, teniendo en cuenta un manipulador determinado y evitando posibles colisiones.

Estos dos últimos módulos, el "planificador de secuencia" y el "generador de movimientos", intercambian información ya que si el "generador de movimientos" detecta que el componente indicado por el "planificador de secuencia" no puede ser desensamblado en un determinado instante, debido a restricciones geométricas del producto, entonces el "planificador de secuencia" debe reestructurar la secuencia de componentes a desensamblar para tener en consideración dicha eventualidad.

Una vez se tienen generados los movimientos de un componente, se llevan a cabo por el sistema de simulación y también por el manipulador que se está utilizando.

Por último existe en la arquitectura del sistema un bucle de retorno de información, que sirve, para una vez se ha desensamblado un componente, actualizar mediante el "sistema de visión artificial" el modelo del producto, realizando el planificador global un nuevo modelado más preciso del producto disponible en estos momentos.

En esta tesis se realizan los módulos correspondientes al planificador global, a la base de conocimiento, al planificador de secuencia, al generador de movimientos así como la simulación y su posterior realización por el sistema real, de tal manera que se abordan todas las partes del sistema mostradas en la Figura 1.4 excepto la correspondiente al sistema de visión artificial que no se lleva a cabo en esta tesis.

El módulo correspondiente al "sistema de visión artificial" no se desarrolla en esta tesis, ya que se parte del hecho que dicho sistema realiza correctamente las tareas que le corresponden y se trabaja con dicha información.

### **1.3. Estructura de la tesis**

La presente tesis se estructura en una serie de capítulos en los que se van a ir tratando los distintos temas planteados para la resolución del problema de desensamblado.

En el capítulo 1 se presenta un estado del arte actual de los métodos de planificación para sistemas de desensamblado; tras esta introducción se

presenta el método original aportado en esta tesis para realizar la planificación de desensamblado, ya sea parcial o total.

En el capítulo 2 se presenta un método que permite calcular la trayectoria de movimientos que debe realizar un componente para poder desensamblarse del resto del producto. Esta trayectoria se plantea desde el punto de vista de una secuencia de movimientos cartesianos lineales del componente.

En el capítulo 3 se explica como se lleva a cabo la fusión de la información proveniente del capítulo 1 y del capítulo 2, añadiendo un robot que realiza el desensamblado. Además se presentan cuales son las diferencias entre realizar un desensamblado virtual, y un desensamblado real. En el primero no se requiere un brazo robot real, sólo se realiza una simulación y en el segundo se tiene el sistema completo y se deben realizar los movimientos con el robot, realizando correctamente la fusión entre la información obtenida en los capítulos 2 y 3.

En el capítulo 4 se aplican los conocimientos aportados en la presente tesis a un sistema real, se realiza el desensamblado de varios PCs, mostrando los pasos seguidos así como secuencias del sistema real llevando a cabo la tarea. También se incluye una descripción de la base de datos utilizada para la generación de los ejemplos.

En el capítulo 5 se presentan las conclusiones de la tesis así como un resumen de las principales aportaciones que realiza. En este capítulo también se describen posibles temas de trabajo futuro que plantea esta tesis.

Seguidamente se incluyen las referencias bibliográficas que se han utilizado.

Por último se incluyen tres anexos: El primero de ellos describe la aplicación creada para el sistema flexible de desensamblado, que permite la introducción de los modelos relacionales y la generación de los movimientos de desensamblado.

El segundo anexo, incluye los cálculos seguidos para obtener el modelo cinemático del robot utilizado en las aplicaciones de desensamblado de PCs.

El último anexo contiene el desarrollo de los cálculos necesarios, tal como se verá en el capítulo 4, para adaptar la cinemática inversa del robot utilizado en la aplicación, para que tenga en consideración una mesa de trabajo que permite rotar y desplazar el producto.

# 2

## Planificación de secuencia de desensamblado

### 2.1. Introducción

En este capítulo se tratan los métodos de planificación de secuencias de desensamblado existentes en la bibliografía. Además se presenta un método de planificación original aportado en esta tesis. El cual se basa en la generación de un modelo relacional del producto que permite mediante una jerarquización de los componentes que lo forman, calcular la secuencia de desensamblado.

De cara a planificar el desensamblado de un producto hay que tener en cuenta si se realiza un desensamblado parcial o total [165], dado que según la opción elegida se deben seguir unas u otras estrategias. Esta tesis presenta aportaciones tanto en desensamblado parcial como total.

El desensamblado parcial consiste en desensamblar un determinado componente o conjunto de ellos, sin necesidad de separar todos los componentes del producto.

El desensamblado total pretende separar todos los componentes del producto obteniendo cada uno de los componentes por separado.

A la hora de realizar una planificación para desensamblar un producto un aspecto a tener en cuenta es la representación que se va a realizar de él. Ésta determina el tipo de operario necesario para poder definirla: cualificado o no cualificado. La representación expresa el producto de manera simbólica, las distintas piezas que lo forman así como las relaciones entre ellas. Para ello existen en la bibliografía múltiples métodos para llevarlo a cabo, ya sea por medio de representaciones que reflejan distintas materias primas que

componen un producto, aquellas que intentan recuperar ciertas piezas, otras que detallan todos y cada uno de los componentes y sus características, etc.

Una determinada representación está preparada para un operario en concreto de manera que hay esquemas que intentan reflejar el producto desde el punto de vista del diseñador, otros desde el punto del usuario, otros desde las relaciones entre los distintos componentes, otros desde las características de los componentes, etc. Esto no quiere decir que una forma de expresar un producto excluya a la otra, pero a la hora de realizar el diseño se debe tener en cuenta a quién va dirigida la representación, y como se dijo anteriormente, en la fase de diseño del producto, se deben reflejar las características del mismo, y cuando sea posible, considerar también los métodos para su desensamblado. Esta manera de ver la representación de un producto es muy útil si se realiza desde el primer momento del ciclo de vida, pero en la actualidad muchos productos no tienen una representación pensada para el desensamblado con lo que se debe buscar una manera de reflejar sus características. Dicha representación interesa que pueda ser realizada por un operario que tenga conocimientos del producto en particular, sin necesidad de tener grandes conocimientos técnicos, esta es una de las características que se plantea en la representación aportada por esta tesis. Dado que al tratarse de productos ya diseñados interesa poder introducirlos fácilmente en un programa que calcule la secuencia de desensamblado.

A la hora de seleccionar una manera para representar un producto a desensamblar hay que realizar un estudio sobre cuáles son las características del mismo que se quieren reflejar en el esquema, el tipo de desensamblado a efectuar, a quién va dirigida dicha representación, etc.

Las características que se quieran representar del producto influyen también en la representación, en el sentido de reflejar hasta que nivel de detalle se quiere llegar a definir el mismo, es decir, qué componentes forman el producto. Por ejemplo en un ordenador se pueden considerar como componentes las tarjetas, los discos duros pero no los chips que componen dichas tarjetas y discos duros. Con esta representación no se pueden desensamblar los distintos chips, pero si no se desea llegar a ese nivel de desensamblado el esquema es correcto y además más sencillo que si se tienen en cuenta dichas características. Esto quiere decir que para un mismo producto se dan distintas representaciones según el objetivo perseguido en el desensamblado. Esta profundidad se puede seleccionar en base a múltiples criterios, tanto económicos como ecológicos, según el objetivo final perseguido.

Además de definir el nivel de detalle en la representación hay que dejar claro hasta que punto es rentable el desensamblado de un producto, ya que su desensamblado total puede no serlo, pero uno parcial sí. Para calcular hasta donde es rentable realizar el desensamblado se utilizan técnicas que maximizan el beneficio obtenido en un proceso de desensamblado. Estas técnicas, que permiten calcular hasta qué punto es rentable desensamblar un producto, se denominan en la bibliografía como *disassembly leveling* [20] [27] [37] [38] [39] [53] [65] [146] [152] [157] [172] [187] [194] [195] [205] [210] [301] [306] [307].

La información que se necesita de cada componente así como de las relaciones entre ellos dependen del tipo de desensamblado a realizar. Si se

quiere realizar un desensamblado para el reciclado del producto entonces lo que interesara reflejar en la representación son los distintos materiales de los componentes ya que el desensamblado de dos componentes conjuntamente es correcto si están compuestos por el mismo material, puesto que no habría ningún problema para su reciclado. Si por el contrario se desea realizar un desensamblado para mantenimiento o para la reutilización de los componentes, interesa en vez de una representación que refleje los distintos materiales, una representación que tenga en cuenta los componentes que pueden ser sustituidos o recuperados así como los enlaces que tienen entre sí y cómo se pueden eliminar.

Lo que se persigue con una estrategia de planificación del proceso de desensamblado es obtener una secuencia de las operaciones a realizar para quitar los componentes de un producto necesarios para acceder a un determinado componente o para separarlos todos. La cantidad de posibles soluciones crece exponencialmente con el número de componentes que posee el producto, con lo cual calcular la solución óptima requiere de técnicas adecuadas que van apareciendo paulatinamente en la literatura que consiguen mejorar la generación de las secuencias de desensamblado. Estas técnicas se describen en el siguiente apartado.

Este incremento en la cantidad de soluciones existentes en la generación de secuencias de desensamblado es debido a que se trata de un problema NP-completo [191]. Por ejemplo, un producto con 3 componentes puede tener hasta  $3! = 6$  secuencias distintas de desensamblado, mientras que un producto con 10 componentes podría tener hasta  $10! = 3.228.800$  secuencias de desensamblado distintas. Aunque, considerando las restricciones existentes entre los componentes, se reduce el número de secuencias a analizar, para obtener la óptima, la cantidad de soluciones se supone que sigue siendo grande para productos complejos, lo que implica el desarrollo de heurísticas y métodos para calcular las secuencias de desensamblado de un producto [90].

## 2.2. Estado del arte

En este apartado se describen las maneras de representar un producto presentes en la bibliografía [92] [158] [199] [262] [302], realizando una clasificación según el método de representación del producto utilizado.

### 2.2.1. Grafos

Una de las maneras más extendidas para calcular la planificación de desensamblado consiste en la utilización de grafos que expresan las relaciones entre los componentes, donde los nodos se corresponden con los componentes o agrupaciones de ellos y las aristas representan las relaciones existentes entre ellos.

La representación de productos mediante grafos se puede clasificar en tres estrategias generales: Las que utilizan grafos de componentes-enlaces [148] [149] [150] [299] [300] [301] [308], las que utilizan grafos dirigidos [106] [213] [214] y las que utilizan grafos and/or [62] [89] [106] [107] [163] [206] [303].

La representación mediante grafos de componentes-enlaces la encontramos en [148] [150] [299] [300] [301] donde se plantea la



representación del producto mediante un grafo como el mostrado en la Figura 2.1. en el que cada nodo representa a un componente y las aristas expresan relaciones, uniones, entre los componentes.

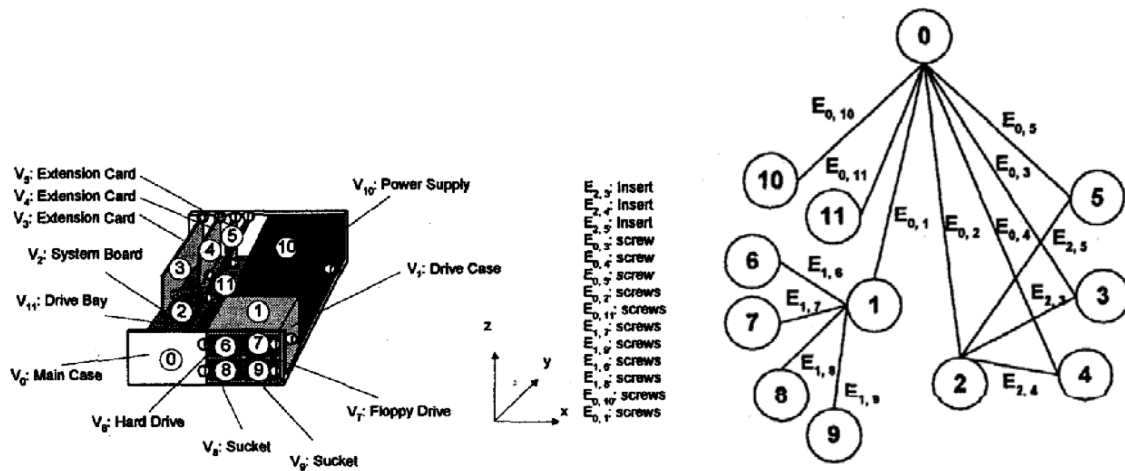


Figura 2.1: PC desensamblado parcialmente y su grafo de componentes-enlaces (Fuente: [300]).

El método que plantea para calcular la secuencia de desensamblado consiste en reducir el grafo a un árbol (Figura 2.2) de tal manera que se tiene una representación más sencilla para calcular cual es la secuencia de desensamblado de un determinado componente.

La secuencia de desensamblado consiste en eliminar los nodos padre, desde el componente que se quiere desensamblar, hasta llegar al nodo raíz del árbol. De manera que se van eliminando las uniones correspondientes a cada una de las aristas.

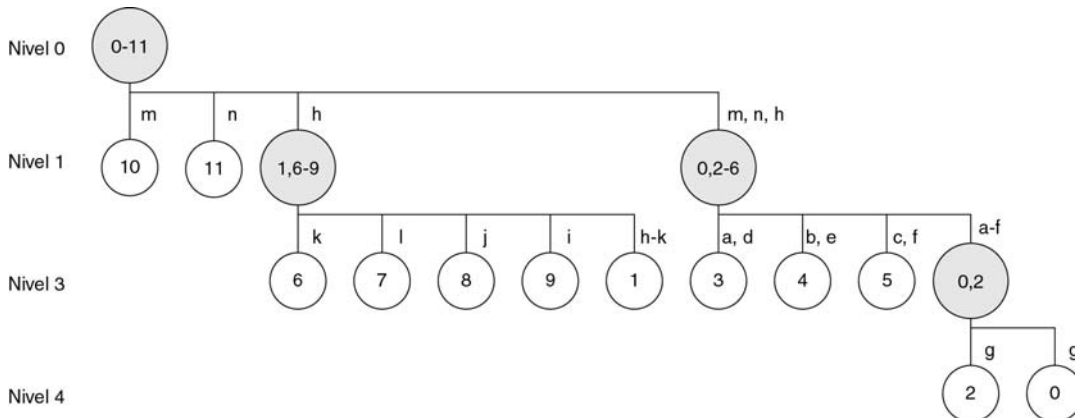


Figura 2.2: Árbol de desensamblado para el grafo de la Figura 2.1 (Fuente: [300])<sup>\*</sup>.

Otra representación mediante un grafo de componentes-enlaces la encontramos en [308], donde el grafo consiste en una combinación de datos geométricos, topológicos y funcionales del producto.

<sup>\*</sup> Donde:  $m=E_{0,10}$ ;  $n=E_{0,11}$ ;  $h=E_{0,1}$ ;  $k=E_{1,6}$ ;  $l=E_{1,7}$ ;  $j=E_{1,8}$ ;  $i=E_{1,9}$ ;  $a=E_{0,3}$ ;  $b=E_{0,4}$ ;  $c=E_{0,5}$ ;  $d=E_{2,3}$ ;  $e=E_{2,4}$ ;  $f=E_{2,5}$ ;  $g=E_{0,2}$ .

La construcción del grafo se realiza en función de las superficies de contacto entre cada par de componentes así como del tipo de contacto existente entre ellas. En la Figura 2.3 se muestra un producto compuesto por tres componentes así como la identificación de las superficies de cada uno de los componentes que lo forman.

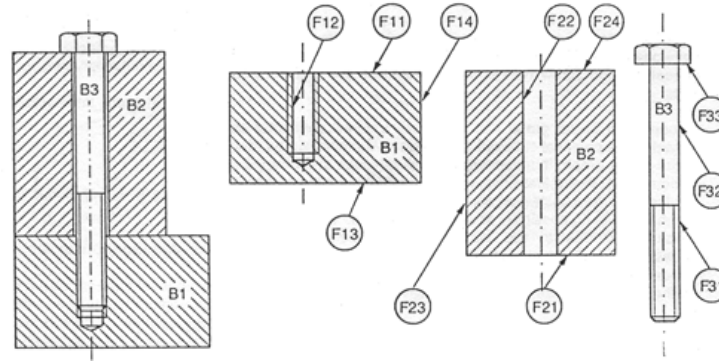


Figura 2.3: Representación de un producto para generar un grafo de componentes-enlaces (Fuente: [308]).

Utilizando esta información se ha generado el grafo de relaciones de la Figura 2.4.

El algoritmo planteado de esta manera realiza una simplificación del grafo obtenido originalmente, esta simplificación se realiza en dos etapas. En la etapa de análisis se obtiene la estructura topológica, representada por medio de relaciones en el grafo entre características de los componentes. La segunda etapa consiste en la simplificación del grafo combinando las referencias topológicas del producto con los componente físicos que se consideran para crear el producto (Figura 2.4). Tras esto realiza una búsqueda en el grafo del componente a desensamblar, y para desensamblarlo hay que eliminar las uniones que posee con el resto de componentes, es decir, eliminar las aristas que lo unen al resto del grafo.

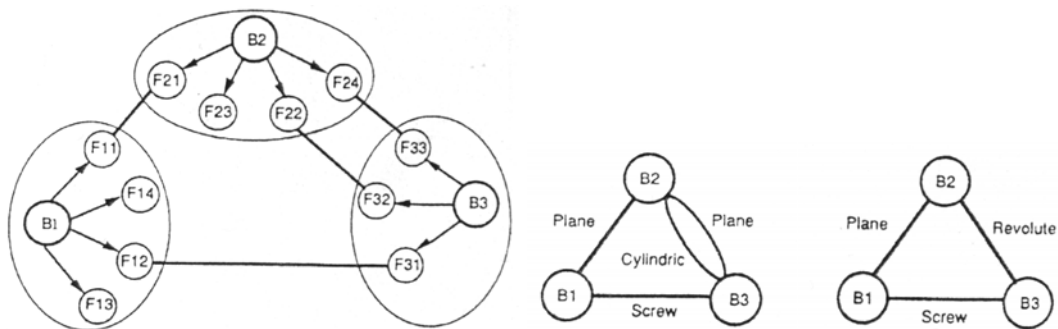


Figura 2.4: Grafo de relaciones, grafo a nivel de producto y grafo del producto (Fuente: [308]).

El uso de grafos dirigidos para representar los productos [106] [213] [214], donde los nodos son los componentes y las aristas se corresponden con restricciones que vienen dadas por la precedencia entre los componentes, (Figura 2.5) es otro método utilizado para calcular secuencias de desensamblado.

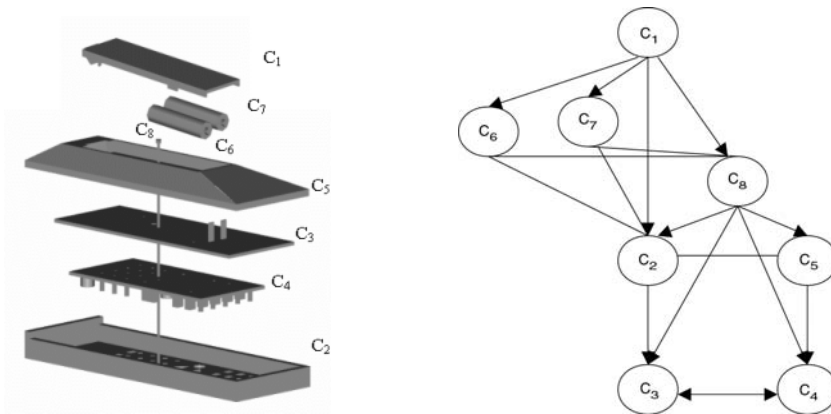


Figura 2.5: Producto y grafo dirigido correspondiente (Fuente: [214]).

En base a la información contenida en el grafo, así como a los atributos de cada componente: operaciones necesarias para desensamblarlo, valor económico, toxicidad, etc., se calcula mediante unas ecuaciones matemáticas la secuencia óptima de desensamblado de un componente [214] o un conjunto de ellos [213].

Este método resulta eficaz a la hora de maximizar una función de coste para obtener el mayor beneficio en el desensamblado del producto.

La utilización de grafos and/or [62] [89] [106] [107] [163] [206] [303] es una de las técnicas más extendidas ya que permite una gran cantidad de posibilidades de desensamblado.

Esta representación expresa de forma compacta las relaciones entre los componentes de un producto así como las opciones de ensamblar cada uno de los componentes con los demás. Un grafo and/or tiene como nodos agrupaciones de componentes y sus hojas corresponden a componentes individuales. Juntando dos hojas se consigue un nodo intermedio que representa un subensamblado del producto, combinando estos nuevos subensamblados con el resto de nodos del grafo se van teniendo nodos con una mayor cantidad de componentes, este proceso sigue hasta que se llega al nodo raíz que representa el componente completamente ensamblado (Figura 2.6).

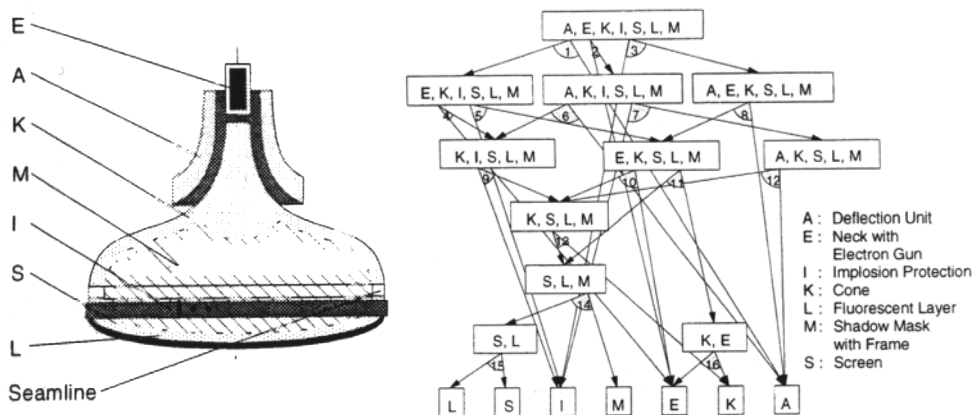


Figura 2.6: Estructura y grafo and/or de un tubo de rayos catódicos (Fuente: [206]).

Las relaciones entre los componentes del grafo pueden ser de dos tipos, representadas mediante las aristas "and" u "or" como indica el nombre del grafo. Un conjunto de aristas se definen como "and" si para poder llevar a cabo el desensamblado deben expandirse, realizar el desensamblado de todos los nodos que conectan dichas aristas. Un conjunto de aristas se definen como "or" si para realizar el desensamblado hay que seguir una cualquiera de ellas.

Cada autor presenta un método distinto de resolver el problema del cálculo de la secuencia de desensamblado basándose en el grafo and/or, pero las estrategias se pueden resumir en recorrer el grafo desde el nodo que representa al componente o al conjunto de componentes que se desea desensamblar y buscar el mejor camino para llegar al nodo raíz con lo que la secuencia de desensamblado será la obtenida en este recorrido realizándola desde el nodo raíz hasta el nodo a desensamblar.

Una extensión de los grafos and/or consiste en la incorporación de la función "xor", lo que amplía la información de los productos que se pueden expresar con sólo relaciones "and" y "or" [190] para calcular las correspondientes secuencias de desensamblado.

### 2.2.2. Características geométricas

La utilización de información geométrica para realizar una planificación para el desensamblado de un producto resulta una técnica cada vez más extendida, especialmente a la hora de realizar el diseño de nuevos productos, ya que permite comprobar si un producto tendrá un fácil desensamblado cuando termine su ciclo de vida útil.

Los planteamientos más actuales basados en características geométricas, utilizan un modelo CAD para generar un grafo, gracias a las restricciones que se pueden obtener directamente del diseño del producto, sin la necesidad de intervención de un operador [62] [160] [248]. En base al grafo obtenido se calcula la secuencia de desensamblado [276] [279].

Un primer planteamiento consiste en utilizar una representación basada en polígonos, de tal manera que cada componente se describe como un conjunto de polígonos; además un componente cuando esté junto a otro los polígonos que los forman tendrán una relación de contacto (Figura 2.7) [161] [243] [247] [248] [249] [250].

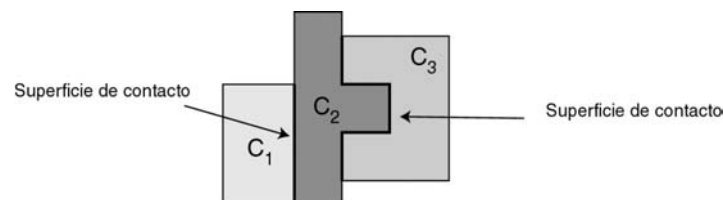


Figura 2.7: Superficies de contacto del producto ensamblado (Fuente: [243]).

Partiendo de la información geométrica, generan un grafo con la topología del producto (Figura 2.8), que permite calcular cual es la secuencia de desensamblado a llevar a cabo para un determinado componente.

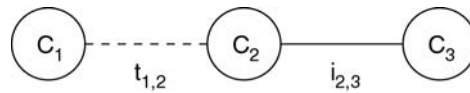


Figura 2.8: Grafo de topología del producto ensamblado de la Figura 2.7 (Fuente: [243]).

Estos algoritmos se basan en la utilización de polígonos convexos [24] [25], transformando aquellos que no lo son tal y como se muestra en la Figura 2.9 para el componente  $c_2$  del producto de la Figura 2.7.

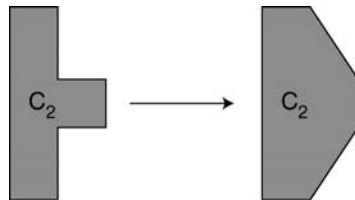


Figura 2.9: Transformación de un polígono no convexo en convexo (Fuente: [243]).

Teniendo en cuenta las características que se obtienen de los algoritmos basados en características geométricas se puede decir que obtienen secuencias óptimas de desensamblado para sistemas 1-desensamblado. Un sistema de desensamblado es 1-desensamblado cuando se utiliza un sólo movimiento lineal para desensamblar un componente.

En [248] se presenta un algoritmo para calcular las secuencias óptimas de desensamblado, basado en la propagación de ondas. Este método consiste en simular la propagación de una onda desde el componente a desensamblar hasta llegar a un componente que se encuentre accesible, de tal manera que, realizando la secuencia calculada en orden inverso, se puede desensamblar el componente correspondiente (Figura 2.10).

Este mismo método ha sido ampliado en [62] para el cálculo de una secuencia de desensamblado conjunta de varios componentes del producto.

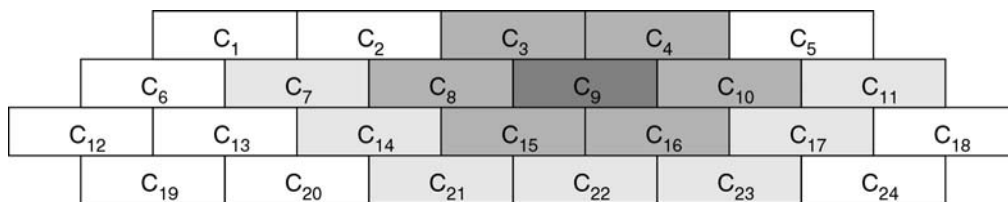


Figura 2.10: Método de propagación de ondas, ondas desde el componente  $c_9$  (Fuente: [248]).

Una de las ventajas que proporciona este método se basa la posibilidad de utilizar los modelos geométricos que proporcionan programas CAD comerciales para definir los componentes del producto. Como principal desventaja hay que señalar que sólo contempla movimientos lineales, en una única dirección, aunque puede considerar otros tipos de movimientos como casos especiales, por ejemplo, en el caso de los tornillos, los considera como elementos especiales que requieren un tipo de movimiento particular, de manera que, utilizando consideraciones adicionales, puede incluir componentes que requieran movimientos de desensamblado no lineales.

En esta tesis se aporta una ampliación de estos sistemas de desensamblado ya que se amplían para eliminar las restricciones que llevan

impuestos los sistemas 1-desensamblado, permitiendo realizar más de un movimiento lineal para desensamblar un componente.

Otra manera de utilizar las características geométricas para calcular la secuencia de desensamblado se plantea en [89] donde en base a la representación geométrica del producto y a restricciones de precedencia (Figura 2.11) se calcula un grafo and/or desde el que se va generando la secuencia de desensamblado.

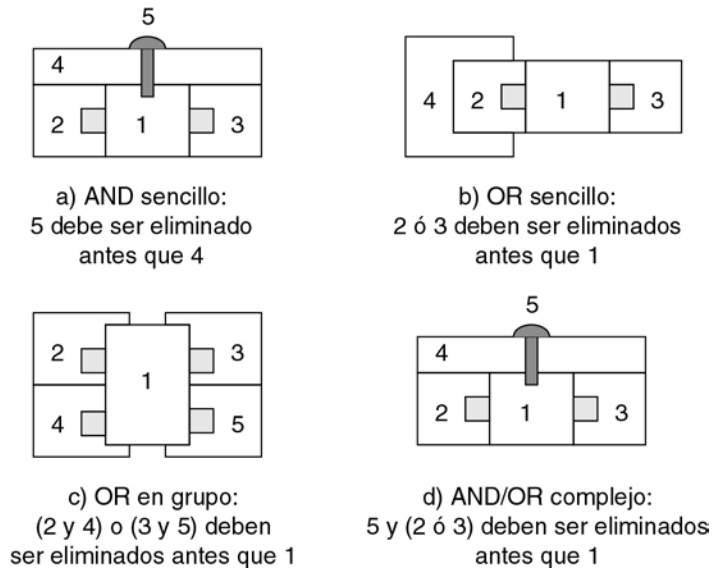


Figura 2.11: Restricciones geométricas para calcular el grafo and/or (Fuente: [89]).

El grafo and/or se calcula teniendo en cuenta las restricciones que definen los movimientos posibles a realizar por un componente, de tal manera que el grafo tiene la información sobre los movimientos a realizar para poder separar un componente del resto.

### 2.2.3. Árboles

Otro método para realizar el cálculo de una secuencia de desensamblado consiste en partir de una representación arborescente del producto [48] [180].

La utilización de estrategias arborescentes permite una paralelización de las tareas a realizar por el sistema de desensamblado [48] considerando las distintas posibilidades de separación de las piezas [28] de un producto.

Una representación arborescente se plantea en [180] con la idea de realizar un desensamblado en tiempo real de un producto. La estrategia planteada permite, por medio de pesos asignados a cada rama del árbol, ir eligiendo en tiempo real cual es la estrategia a seguir para desensamblar un componente, teniendo en cuenta posibles diferencias durante el proceso real respecto a la planificación inicialmente planteada (Figura 2.12).

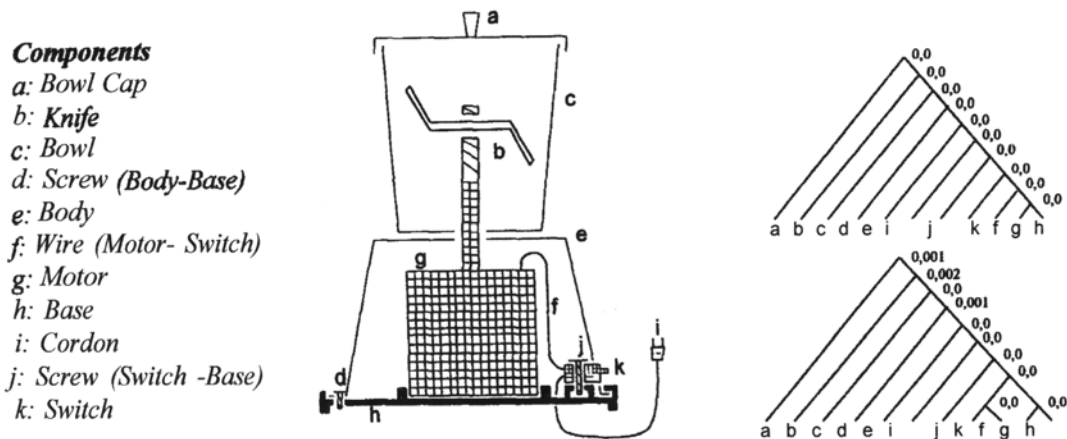


Figura 2.12: Componentes de una picadora eléctrica y árboles para el desensamblado (Fuente: [180]).

En el árbol se tiene contemplada una estrategia de planificación con un coste determinado; por tanto existe un árbol para cada posible planificación de desensamblado. De esta manera el algoritmo selecciona la mejor inicialmente y si no puede seguir realizando el desensamblado según dicho árbol debido a algún componente dañado, la operación prevista no se puede realizar, etc., entonces selecciona otro árbol que sí que permita continuar desde el punto donde ocurrió el problema. Esta estrategia sirve para aplicarla en tiempo real, siempre que se disponga de productos no muy complejos, ya que ello implicaría una gran cantidad de estrategias posibles con el consiguiente coste de almacenarlas.

#### 2.2.4. Redes de Petri

La utilización de redes de Petri tiene una similitud con la representación usada para los grafos and/or, permitiendo generar a partir de un grafo and/or la correspondiente red de Petri para realizar el desensamblado del producto [188] [189] [190] [261] [290] [298] [310].

Donde las transiciones representan acciones de desensamblado y por tanto deben tener una o más entradas, y al menos dos salidas, ya que el producto completo se representa por el estado inicial y el resto de estados se corresponden con subensamblados del producto. Un ejemplo de un producto diseñado y la correspondiente red de Petri de desensamblado se muestra en la Figura 2.13.

El uso de redes de Petri permite utilizar las implementaciones existentes de las mismas para realizar los diseños de los procesos de desensamblado. Además, mediante el ajuste del instante de disparo de las transiciones, se pueden crear estrategias de desensamblado adaptativas.

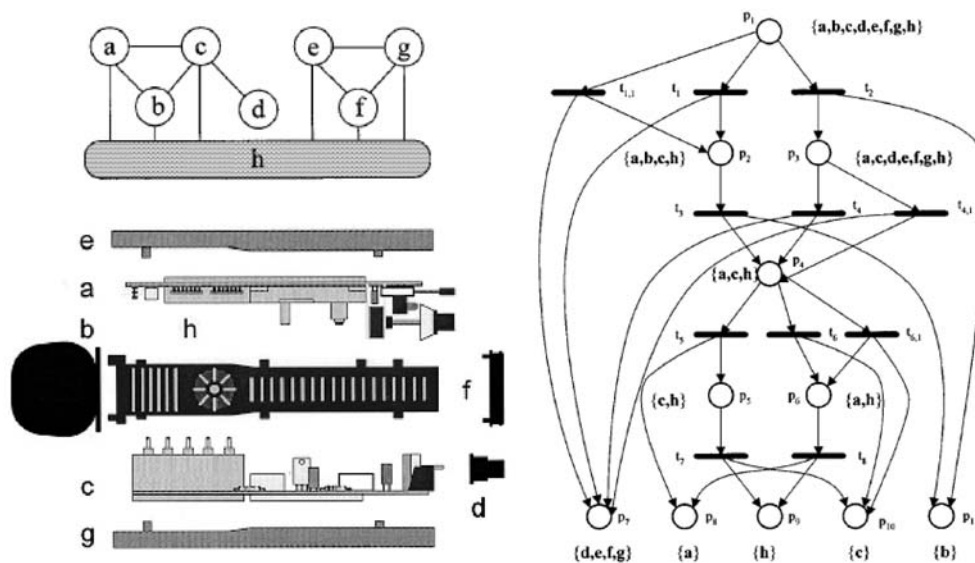


Figura 2.13: Producto y su red de Petri de desensamblado (Fuente: [310]).

### 2.2.5. Otros métodos

En este apartado se presentan otros métodos para calcular las secuencias de desensamblado que no están tan extendidos como los anteriores, entre los que se incluyen las redes neuronales, los sistemas expertos, los algoritmos genéticos, etc.

#### 2.2.5.1. Redes neuronales

Las redes neuronales también pueden utilizarse para generar secuencias de desensamblado [110]. Esta perspectiva busca maximizar una función de beneficio para los materiales recuperados, pudiendo utilizar redes de Hopfield, que es uno de los métodos más extendidos para resolver problemas de optimización con redes neuronales.

Este método permite realizar un planteamiento que resuelve en paralelo los problemas de desensamblado.

#### 2.2.5.2. Sistemas expertos

Los sistemas expertos se basan en la utilización de información para la generación de la secuencia de desensamblado. La información se encuentra en una base de datos y permite, según se definen nuevos productos, utilizar la información existente para su definición, de tal manera que, si para desensamblar un determinado producto no existe una estrategia en la base de datos, se busca una que corresponda a un producto similar, y si dicha estrategia funciona correctamente entonces se aumenta la información de la base de datos para contemplar el nuevo planteamiento [202].

En la Figura 2.14 se muestra un ejemplo de cómo quedaría la base de datos después de introducir varios productos en un sistema experto para planificación de secuencias de desensamblado.



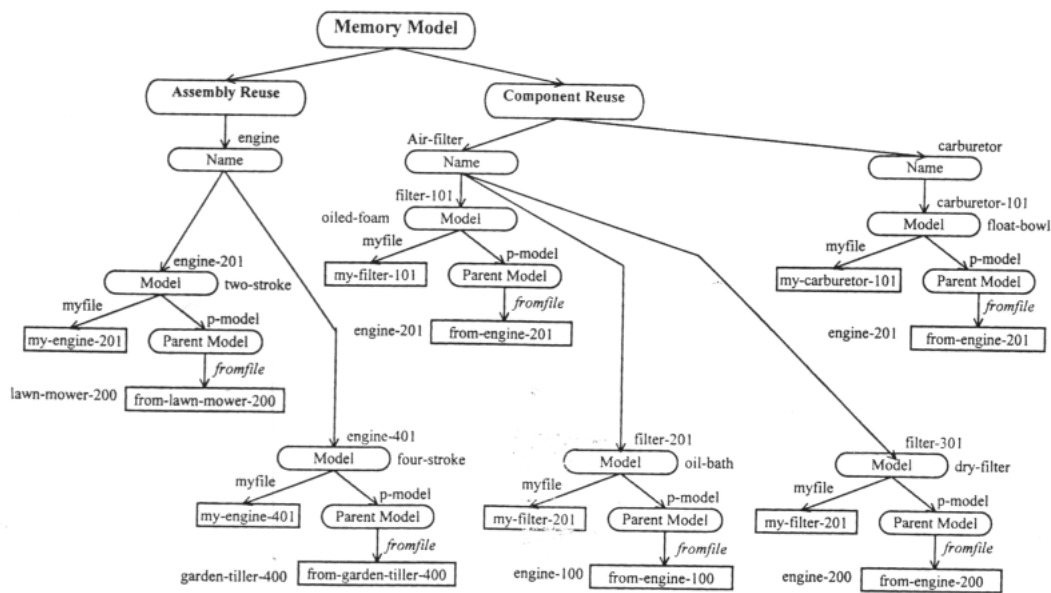


Figura 2.14: Base de datos de un sistema experto después de la introducción de varios productos (Fuente: [202]).

### 2.2.5.3. Algoritmos genéticos

Los algoritmos genéticos plantean una modalidad que permite de una manera interesante realizar el problema del desensamblado.

El planteamiento propuesto en [240] propone partir de un grafo and/or del producto y generar secuencias de desensamblado de manera aleatoria. A continuación se calcula el coste de cada una de ellas. El algoritmo genético crea nuevas generaciones de estrategias de desensamblado partiendo de las mejores estrategias obtenidas en la generación anterior.

Para llegar a una buena solución, el número de generaciones necesarias se incrementa rápidamente según se va aumentando la complejidad de los productos.

Otro planteamiento de los algoritmos genéticos [133] consiste en considerar cada dirección de desensamblado como un cromosoma de manera que la secuencia de movimientos a realizar sobre cada componente se calcula asignando a cada uno de los componentes del producto una dirección y se hace evolucionar al sistema hasta conseguir una secuencia que minimice el coste.

### 2.2.6. Comparativa

En la tabla siguiente se ha realizado una comparativa entre los métodos expuestos indicando las ventajas e inconvenientes de cada uno de ellos.

Método	Ventajas	Inconvenientes
Grafos de componentes-enlaces	<ul style="list-style-type: none"> <li>Representación intuitiva.</li> <li>Representación sencilla de todo tipo de enlaces.</li> </ul>	<ul style="list-style-type: none"> <li>Complejidad en el cálculo de la secuencia de desensamblado, se realiza previamente una simplificación del grafo.</li> </ul>

<b>Método</b>	<b>Ventajas</b>	<b>Inconvenientes</b>
Grafos dirigidos	<ul style="list-style-type: none"> <li>• Rápido cálculo de la secuencia de desensamblado.</li> <li>• Facilidad de aplicar funciones de coste para el cálculo de la secuencia de desensamblado.</li> </ul>	<ul style="list-style-type: none"> <li>• Requiere un conocimiento detallado de las relaciones entre los componentes del producto.</li> </ul>
Grafos and/or	<ul style="list-style-type: none"> <li>• Representación intuitiva.</li> <li>• Expresa relaciones entre subensamblados del producto.</li> </ul>	<ul style="list-style-type: none"> <li>• Necesidad de recorrer un grafo para obtener la secuencia de desensamblado.</li> </ul>
Características geométricas	<ul style="list-style-type: none"> <li>• Utiliza modelos CAD.</li> <li>• Requiere poca o nula intervención de un operador.</li> <li>• Buen sistema para aplicar a productos de nuevo diseño.</li> </ul>	<ul style="list-style-type: none"> <li>• Requiere de casos especiales para representar restricciones de movimiento no lineales.</li> <li>• Aumenta la complejidad si se usan objetos no poliédricos.</li> </ul>
Árboles	<ul style="list-style-type: none"> <li>• Rápida búsqueda de la solución.</li> <li>• Útiles para aplicar en sistemas de tiempo real.</li> </ul>	<ul style="list-style-type: none"> <li>• Requieren gran cantidad de almacenamiento para contemplar todas las posibilidades.</li> </ul>
Redes de Petri	<ul style="list-style-type: none"> <li>• Fácil representación partiendo del grafo and/or del producto.</li> <li>• Buenas para sistemas de tiempo real.</li> </ul>	<ul style="list-style-type: none"> <li>• Complejidad de las redes de Petri obtenidas.</li> </ul>
Redes neuronales	<ul style="list-style-type: none"> <li>• Fácil optimización de funciones de coste.</li> </ul>	<ul style="list-style-type: none"> <li>• Dificultad de definición de la red.</li> </ul>
Sistemas expertos	<ul style="list-style-type: none"> <li>• Representación modular de las tareas de desensamblado.</li> <li>• Permite desensamblar productos similares conociendo sólo como desensamblar uno de ellos.</li> </ul>	<ul style="list-style-type: none"> <li>• Requiere de información específica para conjunto de productos.</li> <li>• Con poca información no consiguen buenos resultados.</li> </ul>
Algoritmos genéticos	<ul style="list-style-type: none"> <li>• Reducen el tiempo de cómputo de las secuencias de desensamblado.</li> </ul>	<ul style="list-style-type: none"> <li>• Requieren una precisa definición de cada uno de los genes.</li> <li>• No garantiza alcanzar una solución óptima.</li> </ul>

## 2.3. Modelo relacional

En esta tesis se ha optado por realizar una nueva representación original de los productos más intuitiva y sencilla de definir que las anteriores. Esta nueva manera de representación distingue dos tipos de componentes: los elementos que forman el producto y aquellos elementos que sólo sirven para unir dos o más componentes. Además de considerar estos dos tipos de componentes se consideran tres tipos distintos de relaciones entre ellos: la primera representa que dos componentes están unidos físicamente de alguna manera, la segunda indica que tienen una relación, pero dicha relación no implica una unión directa entre ambos componentes; y el tercer tipo de relación representa la existente entre un grupo de componentes, los cuales se pueden considerar a su vez como un único componente, a la hora de desensamblar el producto, que denominaremos montaje. Cada montaje tiene asociada una serie de acciones para desensamblar un componente que pertenezca a él.

Con esta información se construye una representación jerárquica del producto que permite realizar un correcto desensamblado ya sea de un componente, de un montaje o del producto completo [268]. Esta representación es una de las aportaciones de esta tesis.

Considerando la relación jerárquica entre los componentes, la estrategia a seguir para el desensamblado se basa en recorrer dicha jerarquía desde el componente que se desea desensamblar hasta llegar al montaje que representa el producto completo, de manera que se almacenan una serie de acciones relacionadas con cada uno de los montajes, que, realizadas en sentido contrario al obtenido, indicarán las acciones a realizar, y en que orden, para desensamblar el componente seleccionado del producto.

Esta estrategia planteada se basa, en todo momento, en realizar un desensamblado no destructivo, de manera que pueda ser utilizada en tareas de mantenimiento y reciclado de componentes para su reutilización. Esto implica que todo componente del producto debe existir de manera individual y no asociado mediante uniones que impliquen una destrucción de las mismas para poder separarlo.

A continuación se describe con mayor detalle y de manera formal la representación del producto; tras ésta se exponen las acciones de desensamblado asociadas a cada tipo de montaje. Después se plantea como realizar el desensamblado total de un producto siguiendo las reglas dadas y posteriormente cómo desensamblar un componente individual y cómo desensamblar un conjunto de componentes, un montaje.

### 2.3.1. Definiciones y representación

En este apartado inicialmente se definen de manera formal los distintos conceptos utilizados en la planificación para dejar claro su significado de ahora en adelante. Tras estos conceptos se describe como realizar la representación gráfica de un producto así como el significado de los términos utilizados para dicha representación.

### 2.3.1.1. Definiciones

- **Componentes:** Se entiende por un componente a todas aquellas piezas que forman parte de un producto y poseen entidad física independiente e individual, por ejemplo: pastillas de freno, llantas, chips, etc. Es decir un componente se corresponde con cada una de las partes que se pueden considerar de manera individual en un producto (Figura 2.15).



Figura 2.15: Componentes de diversos productos.

- **Elementos de enlace:** Se entiende por un elemento de enlace a todas las piezas que tengan entidad física independiente e individual que se utilizan para enlazar dos o más componentes (Figura 2.16), por ejemplo: tornillos, conectores, etc.



Figura 2.16: Elementos de enlace.

Como puede deducirse de esta definición no se considera como elemento de enlace a soldaduras ni aquellos métodos de unión que no posean una existencia independiente e individual, por tanto si se dispone de dos piezas unidas mediante una soldadura, se consideran como un solo componente y no como dos componentes unidos mediante un elemento de enlace (la soldadura). Esto se debe, como se comentó anteriormente, a que se está trabajando con una planificación para desensamblado no destructivo y la eliminación de una soldadura implica técnicas de desensamblado destructivo. En la Figura 2.17 se muestra una placa de circuito que posee varios chips. Como los chips se encuentran soldados a la placa, no se les puede considerar como componentes individuales, ya que

están unidos mediante una soldadura; se considera como un todo a la placa con los chips.



*Figura 2.17: Componentes que requieren un desensamblado destructivo.*

- Elementos: Se entiende como elementos tanto a los componentes como a los elementos de enlace. Cuando aparezca este término en el texto es porque se está refiriendo indistintamente a los componentes y a los elementos de enlace. Se trata de una manera de referirse a ambos términos sin tener que utilizar los dos explícitamente y de esta manera mejorar la legibilidad del texto.
- Uniones: Se consideran uniones a las relaciones entre los distintos elementos. Se clasifican en dos tipos diferentes de uniones:
  - *De cierre*: Este tipo de uniones realizan la unión entre varios elementos formando juntos un todo. Son relaciones que permanecen mientras no se eliminen directamente. Este tipo de unión es el que existe entre un tornillo y los componentes que está atornillando. Por ejemplo el existente entre la carcasa de un ordenador y una unidad de CD (Figura 2.18).



*Figura 2.18: Unión de cierre generada por un tornillo.*

En la Figura 2.19 se muestra un ejemplo de unión mediante rosca, la correspondiente a la tapadera de la bola de un ratón.



*Figura 2.19: Unión de cierre mediante rosca.*

También se puede poner como ejemplo la unión entre la tapa de las pilas de un mando a distancia y la carcasa, unidas mediante una unión a presión.

- *De contacto*: Este tipo de uniones realizan la unión entre varios componentes mientras existan uniones de cierre de un nivel jerárquico superior. En el momento en que desaparecen dichas uniones de nivel superior, también desaparece la relación entre ellos. Este tipo de unión es el que existe entre dos chapas unidas por un tornillo; las dos chapas entre sí tienen una relación, puesto que están una junto a la otra y no pueden estar independientemente, pero en el momento en que el tornillo desaparece las chapas dejan de tener relación ya que pueden tratarse individualmente sin ningún elemento propio o ajeno que las una (Figura 2.20).



Figura 2.20: Unión de contacto entre dos chapas.

- **Montaje**: Se entiende como un montaje a un conjunto de elementos combinados entre sí mediante uniones, al menos una de las cuales ha de ser una unión de cierre, de forma que configuran un todo, que puede ser considerado como un nuevo componente con un menor grado de granularidad. Un montaje es el constituido por los chips, condensadores, resistencias, etc., de un circuito impreso, que individualmente se pueden considerar como componentes; pero si se consideran en su conjunto, están unidos a la placa del circuito impreso, y se tendría un nuevo componente que sería el circuito impreso, que poseería un nivel de granularidad menor que los chips.

En la Figura 2.21 se muestra una placa de circuito impreso montada con sus componentes y junto a ella se muestran un conjunto de posibles componentes.

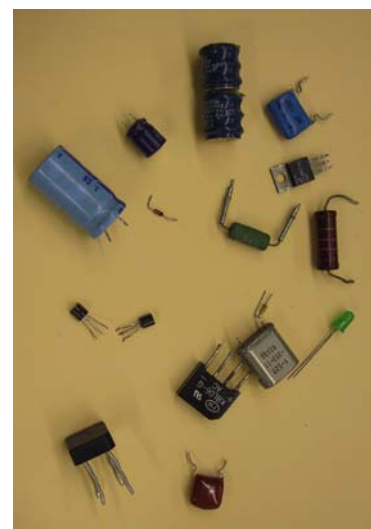
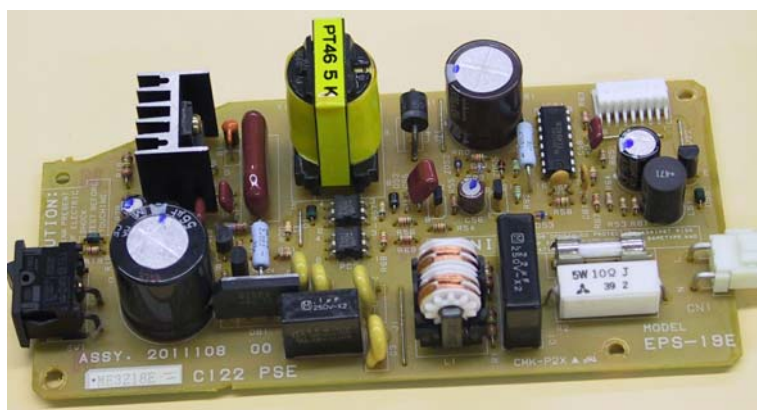


Figura 2.21: Componente con distintos niveles de granularidad.

La placa mostrada es un montaje con un nivel de granularidad menor que si se considera a los chips, condensadores y demás componentes individualmente. De esta manera se pueden definir montajes con el nivel de granularidad deseado según la aplicación a llevar a cabo; y si se dispone de modelos con mayor granularidad, se podrá trabajar sólo con el nivel deseado, considerando el montaje como si de un componente individual se tratara.

- **Producto:** Se entiende por un producto a todos los componentes y elementos de enlace que lo forman así como las relaciones entre ellos. Otra manera de entender un producto es como un montaje que posee una granularidad pequeña, un montaje que puede ser considerado como algo utilizable por ejemplo: un coche, un ordenador, el sistema de frenos de un coche, etc. Como puede apreciarse la definición de producto es relativa según la tarea que se quiera realizar, ya que un producto podrá también ser considerado como un montaje de otro producto con menor granularidad.

### *2.3.1.2. Representación*

Los elementos definidos en el apartado anterior se representan gráficamente utilizando la nomenclatura siguiente:

- **Grafo:** Representación gráfica mediante nodos y aristas de un montaje o de un producto.
- **Nodos:** Representación gráfica de cada uno de los componentes y elementos de enlace que forman el producto. Hay dos tipos de nodos que se representan por tanto de dos maneras distintas. Los componentes mediante una circunferencia, y los elementos de enlace utilizando una circunferencia doble.
- **Arista:** Representación gráfica de las uniones. Como existen dos tipos: de cierre y de contacto, hay dos representaciones distintas. Se representan utilizando una línea continua las de cierre, y una línea discontinua las de contacto.
- **Montaje:** Se representa mediante un conjunto de nodos y aristas que deben estar unidos, al menos, por una unión de cierre. Existen dos maneras gráficas de representar un montaje; esto es debido a que una de ellas puede resultar confusa para productos con muchos montajes. La primera de las representaciones consiste en dibujar una línea cerrada que contenga los nodos y montajes que forman el montaje; ya que un montaje puede ser considerado como un componente, podrá a su vez formar parte de otro montaje. La segunda forma de representación consiste en indicar qué componentes y/o montajes pertenecen a otro montaje mediante una enumeración de los mismos, sin representación gráfica en el dibujo.

Además para definir un montaje hay que tener en cuenta las relaciones de jerarquía que se describen a continuación entre dos montajes, considerando que: un nodo que forma parte de un montaje *i*, puede formar parte de otro montaje *j*, de una de las dos maneras que se van a describir, teniendo en consideración las relaciones de precedencia existentes entre ambos, tal y como se describe a continuación:

- Si ambos montajes,  $i$  y  $j$ , puede alterar su orden de montaje (Figura 2.22a); es decir, si podemos montar primero el montaje  $i$  y a continuación el  $j$  o se puede montar primero el  $j$  y después el  $i$ , no existe precedencia entre ambos montajes; se estaría ante una relación jerárquica que implica un mismo nivel de jerarquía entre ambos montajes ya que su orden de desensamblado es indistinto.

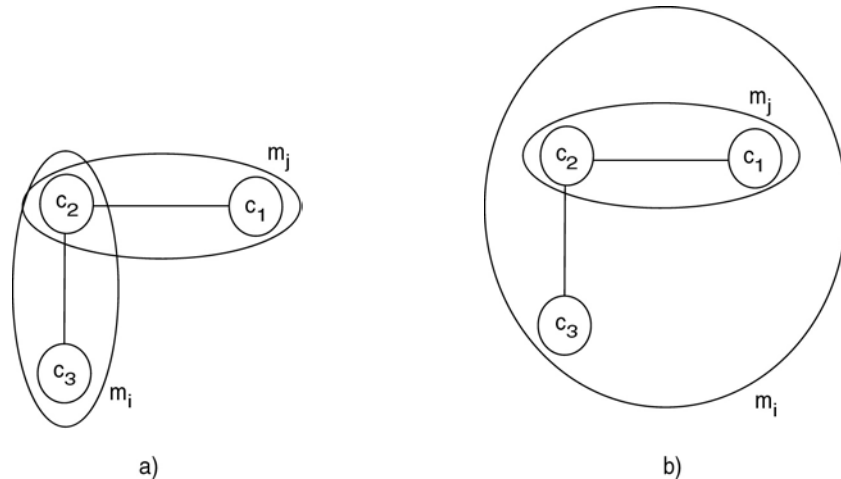


Figura 2.22: Ejemplos de montajes. a) es indistinto desensamblar primero  $m_i$  y después  $m_j$  que al revés. b) hay que desensamblar primero  $m_i$  antes de poder desensamblar  $m_j$ .

- Por otra parte si ambos montajes,  $i$  y  $j$ , no pueden alterar su orden de montaje, hay una precedencia del montaje  $i$  sobre el  $j$ . El nodo común formará parte del montaje,  $j$ , de menor precedencia; el resto de componentes del montaje  $i$  se combinarán con el montaje  $j$  considerando a este como un componente individual, de manera que todos ellos quedarán incluidos dentro del montaje  $i$  (Figura 2.22b). Por lo tanto existe una relación jerárquica entre ambos montajes, uno de ellos, el montaje  $i$ , debe ser desensamblado antes de poder acceder a desensamblar el montaje  $j$ .

## 2.4. Acciones de desensamblado

Como se ha dicho, para cada tipo de montaje existen una serie de acciones asociadas, según la relación existente entre los elementos que forman parte del mismo. Estas acciones forman parte del algoritmo de planificación aportado en esta tesis y se resumen en seis tipos diferentes de montajes posibles. A continuación se describen dichas acciones, explicando primero la relación existente entre los distintos elementos que forman parte de ella y seguidamente, la acción a llevar a cabo para esa situación:

1. Unión de cierre entre componentes y/o montajes (Figura 2.23). Este tipo de relación se da cuando existe una precedencia entre componentes y/o montajes, lo que implica la necesidad de eliminar primero la unión existente entre ellos y después separarlos antes de poder seguir. Las acciones a realizar son:
  - i. Eliminar la unión entre los componentes y/o montajes.



ii. Separar los componentes y/o montajes.

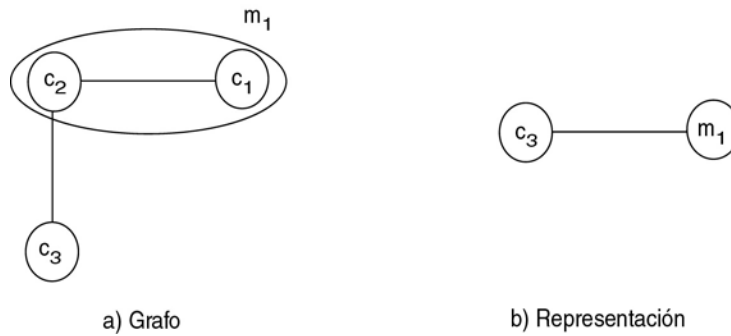


Figura 2.23: Acción 1, unión de cierre entre componente y montaje, precedencia entre montajes. a) Grafo de ejemplo con tres componentes y una estructura jerárquica entre los montajes. b) Representación compacta del grafo de este tipo de acción.

Como ejemplo de este tipo de montaje se tiene la relación existente entre una tarjeta de expansión de un ordenador y la placa base sobre la que va conectada por medio del bus de datos. Este tipo de conector entre dos componentes, que pertenece a los propios componentes, es lo que define la unión entre ellos.

En la Figura 2.24 se muestra un ejemplo de componentes que implican este tipo de acción: se ve un chip insertado en un zócalo; para desensamblar el chip del zócalo hay que utilizar una acción de tipo 1.

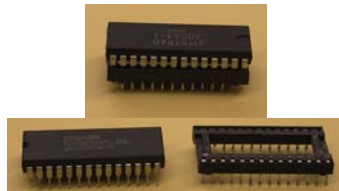


Figura 2.24: Componentes que implican una acción de tipo 1.

2. Unión de contacto entre componentes. Este tipo de relación se da cuando sólo existen uniones de contacto entre un conjunto de componentes. Respecto de este tipo de montaje hay que indicar que aunque no cumple lo dicho anteriormente, todo montaje debe tener al menos una unión de cierre. Es importante considerarlo como tal a la hora de definir las acciones de desensamblado, debido a que, esta situación aparece al eliminar la unión de cierre que unía los componentes, con lo que, considerándolos como un montaje, se simplifican las acciones asociadas a cada montaje, ya que se diferencia la acción de eliminar la unión de cierre de la eliminación de las uniones de contacto.

Pueden aparecer dos casos distintos según el número de componentes implicados:

- a. Dos componentes: Se encuentran relacionados mediante una unión de contacto sólo dos componentes (Figura 2.25), en este caso la acción a realizar es:
  - i. Separar los dos componentes.



Figura 2.25: Acción 2a, unión de contacto entre dos componentes. a) Grafo de ejemplo. b) Representación compacta del grafo de este tipo de acción.

- b. Más de dos componentes: En este caso hay más de dos componentes relacionados mediante uniones de contacto. La acción a realizar en este caso consiste en separar los componentes desde uno de los componentes accesibles hasta llegar al deseado. En el caso mostrado en la Figura 2.26 para desensamblar el componente  $c_i$  hay dos posibilidades. La primera consiste en separar los componentes  $c_1$  de  $c_2$  y a continuación separar los componentes  $c_2$  de  $c_i$  con lo que ya se tendría el componente deseado  $c_i$  desensamblado. La segunda posibilidad consiste en separar  $c_n$  de  $c_{n-1}$  y seguidamente separar  $c_{n-1}$  de  $c_i$  con lo que ya estaría desensamblado el componente  $c_i$ .

- i. Separar los componentes hasta llegar al deseado.

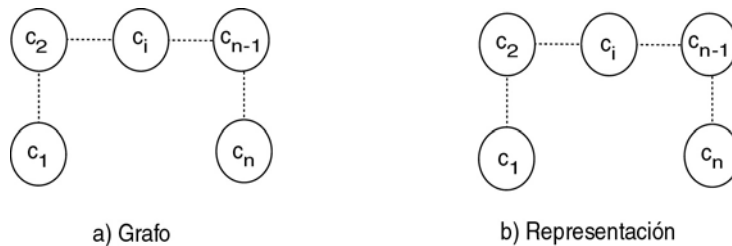


Figura 2.26: Acción 2b, unión de contacto entre más de dos componentes. a) Grafo de ejemplo. b) Representación compacta del grafo de este tipo de acción.

Un ejemplo de este tipo de montaje se tiene cuando se desmonta el auricular de un teléfono; al eliminar el tornillo que une los distintos componentes, éstos dejan de encontrarse relacionados, lo que correspondería a un montaje de este tipo. En la Figura 2.27 se muestran dos componentes de un mando a distancia que poseen este tipo de relación; juntos y por separado.

3. Montajes sin precedencia entre ellos (Figura 2.28). Este tipo de relación existe cuando es indistinto el orden para desensamblar los montajes que forman la relación, por lo tanto la acción a realizar será no hacer nada. Ya que la secuencia de desensamblado a realizar dependerá de los montajes que contengan a éste y no directamente de la relación entre ellos.

- i. No hacer nada.



Figura 2.27: Componentes que implican una acción de tipo 2. Componentes juntos y componentes separados.

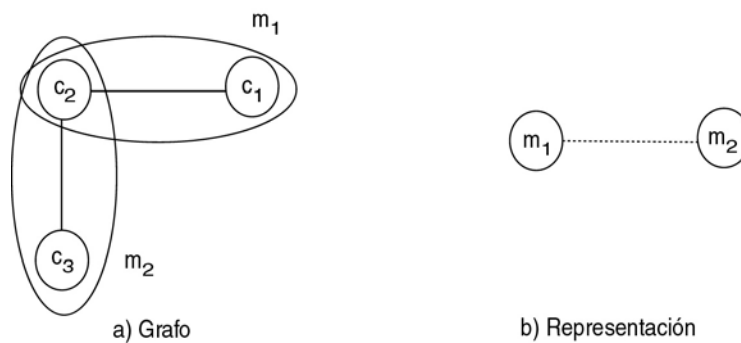


Figura 2.28: Acción 3, montajes sin precedencia entre ellos. a) Grafo de ejemplo con tres componentes y dos montajes sin precedencia. b) Representación compacta del grafo de esta acción.

Un claro ejemplo de este tipo de montajes lo tenemos cuando se tienen en una placa base de un ordenador varias tarjetas; cada tarjeta con la placa base formará un montaje independiente, y entre ellos hay un elemento común, la placa base, y además otros elementos que se pueden montar en el orden que se desee, las tarjetas, que se pueden montar primero una y luego otra, sin que el orden de conexión de las mismas con la placa base afecte al montaje final.

En la Figura 2.29 se muestra un conjunto de chips insertados en zócalos que forman parte de una tarjeta; en este caso el orden en que se desensamblen los chips no afecta al resto, luego se tiene un montaje que implica una acción de tipo 3 para el desensamblado de los chips.



Figura 2.29: Componentes que implican una acción de tipo 3.

4. Unión de cierre entre montaje y elemento de enlace. Este tipo de relación se da cuando un elemento de enlace está uniendo los componentes de un montaje (Figura 2.30). Por lo tanto la acción a realizar en este caso consiste es:

- i. Quitar el elemento de enlace.



Figura 2.30: Acción 4, unión de cierre entre montaje y elemento de enlace. a) Grafo de ejemplo con un montaje y un elemento de enlace. b) Representación compacta del grafo de este tipo de acción.

El más claro ejemplo de este tipo de montaje lo tenemos en un tornillo que está uniendo un conjunto de componentes, como sería el caso de los tornillos que, en una mando a distancia, mantienen unidas los distintos componentes entre sí (Figura 2.31).



Figura 2.31: Componentes que implican una acción de tipo 4.

5. Montaje y/o componentes unidos por un elemento de enlace (Figura 2.32). Esta relación existe cuando un elemento de enlace une varios componentes que no forman un montaje entre ellos y además no se trata de montajes cuyo orden de desensamblado sea intercambiable. En este caso las acciones a realizar son:

- i. Quitar el elemento de enlace.
- ii. Separar los montajes que estaban unidos por el elemento de enlace.

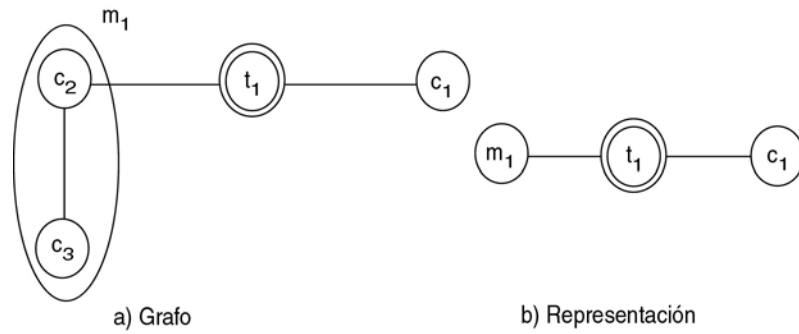


Figura 2.32: Acción 5, montajes y/o componentes unidos por un elemento de enlace. a) Grafo de ejemplo con tres componentes, un montaje y un elemento de enlace que une un componente con el montaje. b) Representación compacta del grafo de esta acción.

El ejemplo que se puede utilizar para describir este tipo de montajes lo encontramos en el caso de una televisión, en donde hay una tapa que cubre el tubo de rayos, y está unida al mismo mediante unos tornillos; los tornillos son los elementos de enlace que mantienen unidos al componente, la tapa, y el montaje, el resto del aparato de televisión. En la Figura 2.33 se muestra una placa de protección de un ordenador, unida mediante un tornillo a la carcasa, de manera que, una vez eliminado el tornillo, hay que separar la placa de la carcasa para conseguir desensamblar ambos componentes, lo que se corresponde con una acción de tipo 5.



Figura 2.33: Componentes que implican una acción de tipo 5.

6. Montajes unidos por un elemento externo cuyo orden de desensamblado es intercambiable (Figura 2.34). Este tipo de relación entre los montajes se da cuando existe un conjunto de montajes que tienen algún elemento común y es indistinto el orden en que separemos los montajes, una vez eliminado el elemento de enlace. Por lo tanto la acción que habrá que realizar es:
  - i. Quitar el elemento de enlace.

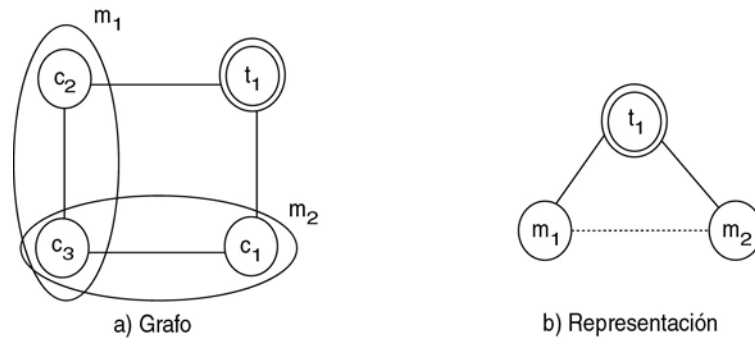


Figura 2.34: Acción 6, montajes unidos por un elemento externo cuyo orden de desensamblado es intercambiable. a) Grafo de ejemplo con tres componentes, dos montajes y un elemento de enlace que une los dos montajes. b) Representación compacta del grafo de esta acción.

Este caso se plantea claramente en los tornillos que sujetan una tarjeta a la carcasa del ordenador. Hay que eliminarlos para poder desensamblar el montaje formado por la tarjeta y la placa base, y también es necesario para poder desensamblar el montaje formado por la placa base con la carcasa; pero, una vez eliminado el tornillo, para seguir desensamblando, no nos afecta si queremos desensamblar un montaje u otro; en el momento en que el tornillo se elimina, dicha unión desaparece y quedan como dos montajes independientes a desensamblar. En la Figura 2.35 se muestra un montaje formado por una placa con los botones de "reset" y "turbo" así como otros indicadores, unidos mediante cuatro tornillos a la carcasa del ordenador. Con esta configuración se tendría entre el montaje formado por la placa y el resto del ordenador que implicaría una acción de tipo 6.



Figura 2.35: Componentes que implican una acción de tipo 6.

## 2.5. Estrategias de planificación

Una vez visto cómo realizar la representación de un producto y definir las acciones asociadas con él, se plantea la estrategia a seguir para conseguir desensamblar el producto. Es importante señalar que se pueden considerar dos objetivos distintos a la hora de realizar el desensamblado: Un desensamblado total del producto o un desensamblado parcial. En esta tesis se plantean aportaciones para ambos objetivos, destacando la aportación en el desensamblado parcial ya que se pueden dar dos tipos de desensamblado

parcial que son tratados en esta tesis y se considera una de sus aportaciones más importantes. En el desensamblado parcial se puede querer desensamblar directamente un componente individual o por el contrario se puede querer desensamblar un montaje; ambas estrategias son comentadas en detalle en los siguientes apartados. La principal diferencia entre desensamblar un componente o un montaje se basa en el hecho de redefinir el grafo del producto para considerar el montaje como un componente individual.

### 2.5.1. Desensamblado total de un producto

Para realizar el desensamblado total de un producto se parte de la representación del mismo mediante sus elementos y montajes; además se dispone de las acciones asociadas a cada montaje. De esta manera para poder desensamblar el producto, se siguen las acciones asociadas al montaje que representa el producto completo; tras estas acciones, hay que realizar aquellas asociadas a cada uno de los montajes que pertenezcan al montaje anterior, y así sucesivamente hasta tener desensamblado el producto por completo.

Por lo tanto, si se considera el producto  $p$ , compuesto por  $n$  montajes:  $m_1, m_2, \dots, m_n$ ; donde el montaje  $m_i$  tiene asociada la acción de desensamblado  $a_i$ . Además el conjunto de montajes que forman parte del montaje  $m_i$  se denomina  $M_i$ . Se tiene que la secuencia de desensamblado total se obtiene siguiendo los siguientes pasos:

- 1) Se parte del montaje  $m_p$ , que se corresponde con el producto completo.
- 2) Se realiza la acción  $a_p$  asociada al montaje  $m_p$ .
- 3)  $\forall m_j/m_j \in M_p$ .
  - a) Realizar la acción  $a_j$ .
  - b) Si  $M_j \neq \emptyset$  entonces ir al paso 3 del algoritmo considerando como montaje  $p$  el  $j$ .

### 2.5.2. Desensamblado parcial de un producto

A la hora de realizar el desensamblado parcial de un producto, se dan dos situaciones que requieren un tratamiento diferente. La primera se da cuando se quiere desensamblar un componente del producto, y la segunda cuando lo que se desea desensamblar es un montaje. Se ha dicho anteriormente que un montaje se considera como un componente de menor granularidad, por lo cual esta diferencia podría parecer que no existe realmente, pero la diferencia consiste en reestructurar el grafo de un producto para considerar el montaje completo como un componente individual y no como un montaje.

#### 2.5.2.1. Desensamblado de un componente

En este apartado se van a explicar cuáles son los pasos a seguir para realizar el desensamblado de un componente individual  $c_i$  del resto de componentes del producto.

Inicialmente se parte de la representación del producto; es decir, del grafo, que representa las relaciones entre los distintos elementos, así como los montajes que forman el producto. En base a esta información se calculan

cuáles son las acciones asociadas a cada montaje del producto. Una vez se tiene esto, se procede con los siguientes pasos:

- 1) Se buscan todos aquellos montajes  $m_j$  que contienen el componente  $c_i$ .
- 2) Para cada uno de los montajes  $m_j$  almacenar la acción asociada con el montaje para separar el componente  $c_i$  del montaje  $m_j$ .
- 3) Para cada uno de los montajes  $m_j$  hay dos opciones:
  - c) Si  $m_j$  se corresponde con el montaje del producto completo: Entonces se sigue con el paso 4.
  - d) Si  $m_j$  es un montaje intermedio del producto: Entonces se considera al montaje  $m_j$  como si de un componente individual se tratara, el componente  $c_i$  a desensamblar, y se sigue con el paso 1.
- 4) Se realizan las acciones almacenadas en el paso 2 pero en sentido contrario, es decir realizando primero las acciones asociadas con el montaje del producto completo y se sigue hasta llegar a las acciones asociadas con los montajes a los que pertenece directamente el componente a desensamblar. Hay que tener en cuenta para la realización de este paso, que pueden existir un conjunto de acciones que se puedan realizar en paralelo. La información sobre si dos acciones se pueden realizar en paralelo aparece en el paso 2 del algoritmo; si existen varios montajes  $m_j$  que realizan este paso en la misma iteración del algoritmo, las acciones asociadas a dichos montajes podrán ser realizadas en paralelo.

#### 2.5.2.2. Desensamblado de un montaje

Visto ya en el apartado anterior cómo realizar el desensamblado de un componente, en este apartado se trata cómo desensamblar un montaje en lugar de un componente. Como ya se ha comentado la principal diferencia entre el desensamblado de un componente y de un montaje, consiste en realizar la simplificación del grafo para que considere al montaje como un componente individual. En esta simplificación hay que considerar lo que sucede cuando distintos componentes de un montaje pertenecen a varios montajes a la vez y lo que se desea es desensamblar uno de esos montajes, pero no el montaje compuesto por todos ellos.

Para realizar el desensamblado del montaje  $m_i$  hay que seguir los siguientes pasos:

- 1) Cambiar todos los componentes  $c_j$  que forman parte del montaje  $m_i$  por el nuevo componente  $c_k$ , donde este nuevo componente  $c_k$  representa el montaje  $m_i$  junto en un solo componente de menor granularidad  $c_k$ . Al realizar esto, hay que cambiar también las relaciones existentes entre un componente que pertenezca a  $m_i$  y otro que no, su relación se transforma en una relación con el componente  $c_k$ . Y si lo que existe es una relación entre dos componentes internos del montaje  $m_i$ , entonces dicha relación desaparece, ya que no tiene sentido una relación  $c_k$  con  $c_k$ .



- 2) Este paso consiste en eliminar componentes redundantes, es decir, si existen dos componentes  $c_k$  que pertenecen al mismo montaje entonces se dejará sólo uno, y si un componente  $c_k$  pertenece a otro componente  $c_k$  (existencia de un montaje  $m_j$  que pertenece al montaje  $m_i$ ) sólo se dejará el componente  $c_k$  que representa una menor granularidad. Así mismo todos los componentes que representa el montaje  $m_i$  se sustituyen por un único componente  $c_k$ .
- 3) Calcular las acciones asociadas a cada uno de los montajes del nuevo grafo ya que, al haber variado los elementos de los montajes y los propios montajes, estas acciones pueden ser distintas de las asignadas originalmente a cada uno de ellos.

Una vez llevados a cabo estos pasos, se dispone de un grafo que define un producto con sus correspondientes elementos y montajes, donde habrá que desensamblar el componente individual  $c_k$  para conseguir el desensamblado del montaje deseado  $m_i$ .

## 2.6. Ejemplo de planificación

Para ilustrar las distintas estrategias, tras realizar una descripción de los pasos a seguir se ha modelado parcialmente un ordenador (Figura 2.36).

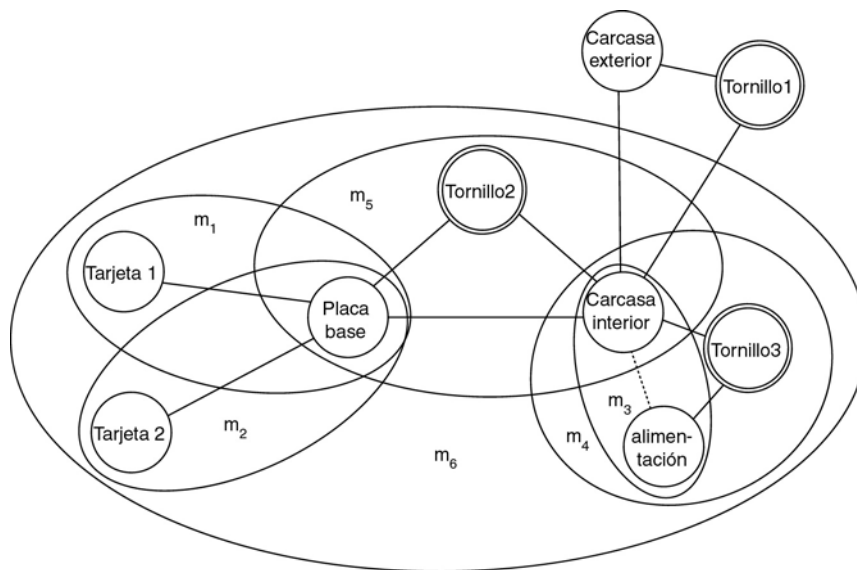


Figura 2.36: Grafo parcial de un ordenador con sus montajes.

El ordenador posee seis montajes más el producto completo, que es considerado como el montaje 7 ( $m_7$ ); los montajes son los siguientes, expresados como una enumeración de los distintos componentes o montajes que incluyen:

- $m_1$ : Está compuesto por los componentes: "Placa base" y "tarjeta 1".
- $m_2$ : Está compuesto por los componentes: "Placa base" y "tarjeta 2".
- $m_3$ : Está compuesto por los componentes: "Carcasa exterior" y "alimentación". Este montaje, como puede verse, no posee ninguna unión de cierre; la unión de cierre esta considerada en el montaje

$m_4$ . El considerar estos componentes como un montaje, tal como se dijo en el apartado anterior, se debe a que, una vez eliminada la unión que mantiene unidos los componentes, en el montaje  $m_4$  se generan las acciones que se contemplan en el montaje  $m_3$ . Al considerarlo de esta manera, permite separar las acciones en dos montajes, en lugar de tener que definir las todas en un único montaje.

- $m_4$ : Está compuesto por el montaje:  $m_3$ ; y por el elemento de enlace: "tornillo 3".
- $m_5$ : Está compuesto por los componentes: "Placa base" y "carcasa interior"; y por el elemento de enlace: "tornillo 2".
- $m_6$ : Está compuesto por los montajes:  $m_1$ ,  $m_2$ ,  $m_4$  y  $m_5$ .
- $m_7$ : Está compuesto por el componente: "Carcasa exterior"; el elemento de enlace: "tornillo 1"; y el montaje  $m_6$ .

### 2.6.1. Desensamblado total de un producto

Teniendo en cuenta el producto definido anteriormente en la Figura 2.36 las acciones asociadas a cada uno de los montajes son las siguientes:

- 1)  $m_1$ : Es un montaje de tipo 1, ya que está formado por dos componentes relacionados directamente entre sí; luego las acciones asociadas con este montaje son:
  - Eliminar la unión entre los dos componentes, en este caso la unión es el bus de datos.
  - Separar los componentes. Dado que si eliminamos la unión, separación por el bus de datos, ya se tienen separados. La acción de eliminar la unión, permite tener directamente separados los dos componentes.
- 2)  $m_2$ : Este montaje también es de tipo 1, ya que está formado por elementos similares que el montaje 1. Luego las acciones relacionadas con este montaje son equivalentes:
  - Eliminar la unión entre los componentes.
  - Separar los componentes.
- 3)  $m_3$ : En este caso el montaje es del tipo 2, para ser más exactos, es un montaje de tipo 2a, dado que la carcasa interior y la fuente de alimentación se encuentran unidas, la una a la otra, mediante una relación de contacto. Con lo que en este caso la acción asociada es:
  - Separar los dos componentes.
- 4)  $m_4$ : Este montaje es de tipo 4, ya que el elemento de enlace "tornillo 3" mantiene unidos a los elementos del montaje  $m_3$ . Con lo que la acción asociada con este montaje es:
  - Quitar el elemento de enlace ("tornillo 3").
- 5)  $m_5$ : Este montaje se corresponde con un montaje de tipo 5, es decir dos componentes unidos mediante un elemento de enlace. Por ello las acciones asociadas con este montaje son:

- Quitar el elemento de enlace ("tornillo 2").
  - Separar los componentes que estaban unidos por el elemento de enlace ("placa base" y "carcasa interior").
- 6)  $m_6$ : Este montaje es de tipo 3, dado que los distintos montajes se pueden ensamblar en cualquier orden. Con lo que la acción asociada a este montaje es:
- No hacer nada.
- 7)  $m_7$ : Este montaje es de tipo 5, lo que quiere decir un montaje ( $m_6$ ) y un componente ("carcasa exterior") unidos mediante un elemento de enlace ("tornillo 1"). Lo que implica las siguientes acciones:
- Quitar el elemento de enlace ("tornillo 1"):
  - Separar el componente ("carcasa exterior") del montaje ( $m_6$ ).

Utilizando estas acciones y las normas para la definición de la secuencia de desensamblado total del producto (Apartado 2.5.1), ésta queda de la siguiente manera:

- 1) Como que el montaje  $m_7$  representa al producto hay que:
  - a. Quitar el elemento de enlace "tornillo 1".
  - b. Separar el componente "carcasa exterior" del montaje  $m_6$ .
  - c. Seguir las acciones de desensamblado para el montaje  $m_6$ .
- 2) Las acciones asociadas con el montaje  $m_6$  son:
  - a. No hacer nada.
  - b. Como los montajes  $m_1$ ,  $m_2$ ,  $m_4$  y  $m_5$  pertenecen al montaje  $m_6$  hay que seguir con las acciones asociadas con estos montajes, que además algunas de ellas podrán ser realizadas en paralelo: aquellas que no impliquen el desensamblado del mismo componente.
- 3) Las acciones asociadas con  $m_5$  son:
  - a. Quitar el elemento de enlace "tornillo 2".
  - b. Separar los componentes "placa base" y "carcasa interior".
- 4) Las acciones asociadas con  $m_4$  son:
  - a. Quitar el elemento de enlace "tornillo 3".
  - b. Como  $m_3$  pertenece al montaje  $m_4$ , se sigue con las acciones para desensamblar dicho montaje.
- 5) La acción asociada con  $m_3$  es:
  - a. Separar los componentes "carcasa interior" de "alimentación".
- 6) Las acciones asociadas con el montaje  $m_2$  son:
  - a. Eliminar la unión entre los componentes "placa base" y "tarjeta 2".
  - b. Separar los dos componentes.
- 7) Las acciones asociadas con el montaje  $m_1$  son:

- a. Eliminar la unión entre los componentes "placa base" y "tarjeta 1".
- b. Separar los dos componentes.

Esta serie de acciones se pueden expresar en forma de grafo dirigido de tareas a realizar, que expresa la precedencia y el paralelismo posible entre las acciones de desensamblado (Figura 2.37).

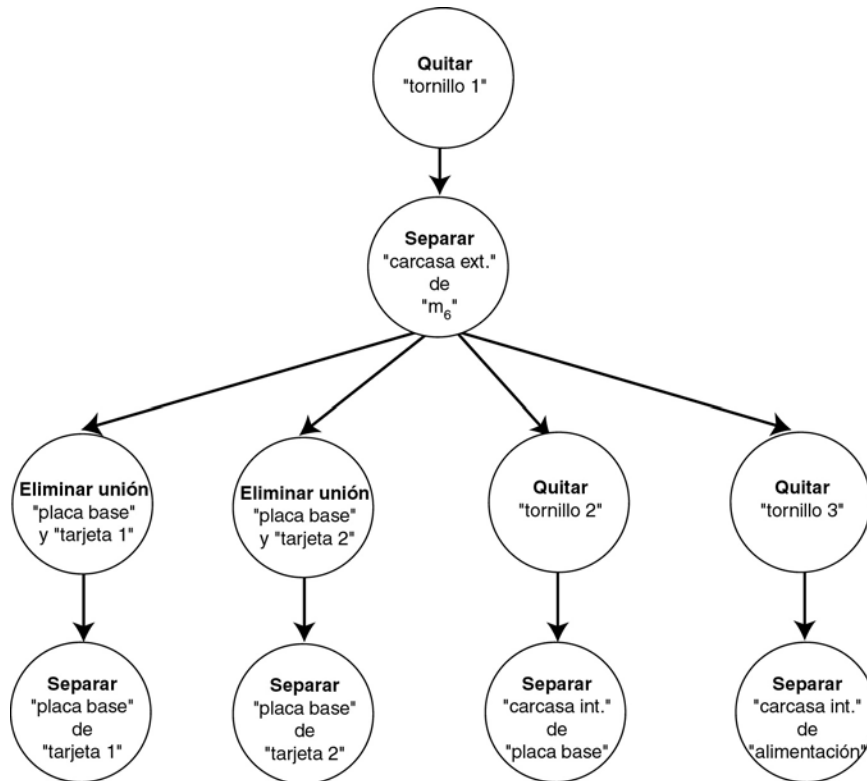


Figura 2.37: Grafo dirigido con las acciones a realizar para desensamblar totalmente el ordenador de la Figura 2.36.

De esta manera, y siguiendo las acciones indicadas se lograría el desensamblado total del producto.

### 2.6.2. Desensamblado parcial de un producto

El desensamblado parcial de un producto se puede realizar, tal y como se dijo anteriormente, desde dos perspectivas distintas: el desensamblado de un único componente o el desensamblado de un montaje. Visto en el apartado 2.5.2 la descripción teórica del algoritmo a utilizar para generar las correspondientes secuencias de desensamblado, en este apartado se van a ilustrar dichos algoritmos basándose en el desensamblado parcial de un ordenador, primero se desensambla un componente y posteriormente se ilustra para el desensamblado de un montaje.

#### 2.6.2.1. Desensamblado de un componente

Como ejemplo del desensamblado de un determinado componente se va a desensamblar el componente "tarjeta 1" del ordenador de la Figura 2.36. Para llevarlo a cabo, primeramente hay que conocer las acciones asociadas a cada montaje, que en este caso, como es el mismo grafo, son las expuestas

en el apartado anterior para el desensamblado total del producto. Seguidamente se siguen los pasos indicados para el desensamblado de un componente:

- 1) Paso 1 del algoritmo: Buscar los montajes a los que pertenece el componente "tarjeta 1".
  - Este componente pertenece solamente al montaje  $m_1$ .
- 2) Paso 2 del algoritmo: Almacenar las acciones asociadas con el montaje encontrado en el paso 1, es decir las acciones asociadas con el montaje  $m_1$ .
  - Eliminar la unión entre los componentes "placa base" y "tarjeta 1".
  - Separar los componentes "placa base" y "tarjeta 1".
- 3) Paso 3 del algoritmo: Como el montaje  $m_1$  no es el producto completo se sigue con el paso 1 del algoritmo, considerando como componente a desensamblar el montaje  $m_1$ .
- 4) Paso 1 del algoritmo: Buscar los montajes a los que pertenece el componente "montaje  $m_1$ ".
  - El montaje  $m_1$  pertenece al montaje  $m_6$ .
- 5) Paso 2 del algoritmo: Almacenar las acciones asociadas con el montaje encontrado en el paso anterior, las acciones asociadas con el montaje  $m_6$ .
  - No hacer nada.
- 6) Paso 3 del algoritmo: Como el montaje  $m_6$  no es el producto completo, entonces se sigue con el paso 1 del algoritmo, considerando como componente a desensamblar el montaje  $m_6$ .
- 7) Paso 1 del algoritmo: Buscar los montajes a los que pertenece el componente "montaje  $m_6$ ".
  - El montaje  $m_6$  pertenece sólo al montaje  $m_7$ .
- 8) Paso 2 del algoritmo: Almacenar las acciones asociadas con el montaje encontrado en el paso anterior del algoritmo, es decir las acciones asociadas con el montaje  $m_7$ .
  - Quitar elemento de unión "tornillo 1".
  - Separar el componente "carcasa exterior" del montaje " $m_6$ ".
- 9) Paso 3 del algoritmo: Como el montaje  $m_7$  sí que se corresponde con el producto completo entonces se sigue con el paso 4 del algoritmo.
- 10) Paso 4 del algoritmo: Realizar las acciones almacenadas en los sucesivos "pasos 2" del algoritmo, en sentido contrario, para llevar a cabo el desensamblado. Es decir las acciones a seguir para desensamblar el componente "tarjeta 1" son:
  - Quitar elemento de unión "tornillo 1".
  - Separar el componente "carcasa exterior" del montaje " $m_6$ ".
  - No hacer nada.

- Eliminar la unión entre los componentes "placa base" y "tarjeta 1".
- Separar los componentes "placa base" y "tarjeta 1".

Estas acciones se pueden expresar en forma de grafo dirigido, como se ve en la Figura 2.38, donde se ha eliminado la acción de "no hacer nada" ya que no aporta información útil a la hora de realizar el desensamblado, pues consiste en no hacer nada, es decir, pasar a la siguiente acción.

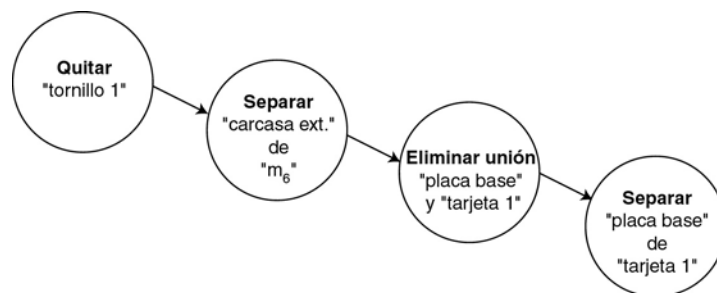


Figura 2.38: Grafo dirigido con las acciones a realizar para desensamblar el componente "tarjeta 1" del ordenador de la Figura 2.36.

#### 2.6.2.2. Desensamblado de un montaje

Para ilustrar más claramente cómo se lleva a cabo el desensamblado de un montaje, se va a realizar el desensamblado del montaje  $m_1$  del producto de la Figura 2.36. Para ello se siguen los siguientes pasos:

- 1) Cambiar todos los componentes que pertenezcan al montaje  $m_1$  (en este caso los componentes "tarjeta 1" y "placa base") por un nuevo componente, que denominaré montaje<sub>1</sub>. Realizando esta sustitución, el grafo del producto queda como se puede ver en la Figura 2.39.

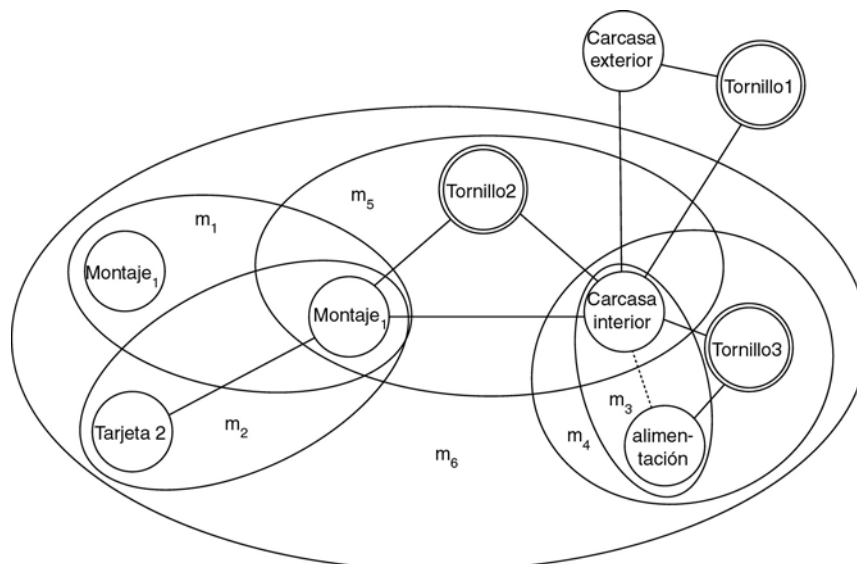


Figura 2.39: Grafo del producto para desensamblar el montaje  $m_1$ . Primer paso del algoritmo.

- 2) Eliminar los componentes redundantes y simplificar el montaje  $m_1$ . Para ello se transforma el montaje  $m_1$  en un único componente denominado "montaje<sub>1</sub>", que mantenga las relaciones existentes

entre los distintos componentes "montaje<sub>1</sub>" que pertenecen a él y los componentes que no pertenecen al montaje. El grafo del producto, tras esta simplificación, se muestra en la Figura 2.40.

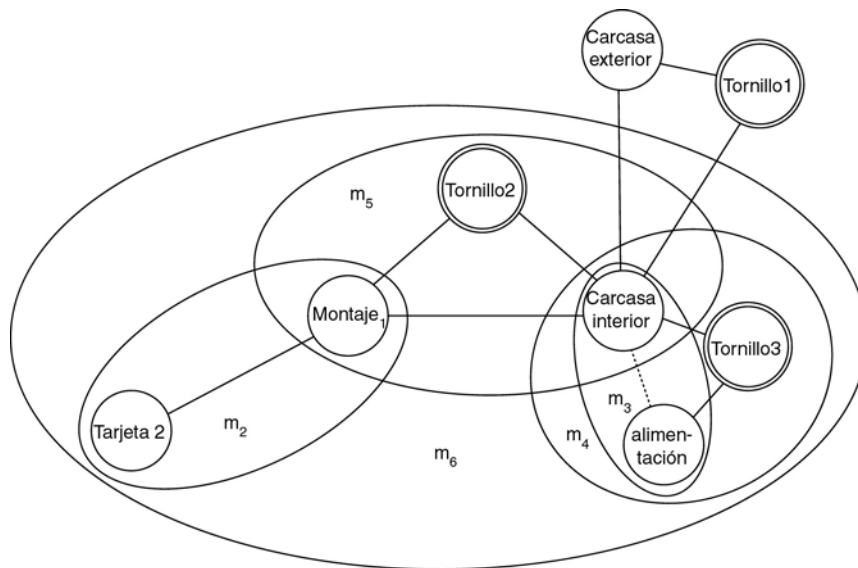


Figura 2.40: Grafo del producto para desensamblar el montaje  $m_1$ . Segundo paso del algoritmo. El grafo definitivo del producto considerando como componente "montaje<sub>1</sub>" al montaje  $m_1$ .

- 3) Calcular las nuevas acciones asociadas a cada uno de los montajes teniendo en cuenta el nuevo grafo, el grafo de la Figura 2.40. Las acciones quedan de la siguiente manera:
- $m_2$ : Este montaje es de tipo 1, porque esta formado por dos componentes relacionados directamente entre sí, el componente "tarjeta 2" y el componente "montaje<sub>1</sub>". Luego las acciones relacionadas con este montaje son:
    1. Eliminar la unión entre los componentes.
    2. Separar los componentes.
  - $m_3$ : Este montaje es de tipo 2, debido a que la carcasa interior y la fuente de alimentación se encuentran unidas la una a la otra mediante una relación de contacto. Con lo que en este caso la acción asociada es:
    1. Separar los dos componentes.
  - $m_4$ : Este montaje es de tipo 4, ya que el elemento de enlace "tornillo 3" mantiene unidos a los elementos del montaje  $m_3$ . Con lo que la acción asociada con este montaje es:
    1. Quitar el elemento de enlace ("tornillo 3").
  - $m_5$ : Este montaje se corresponde con un montaje de tipo 5, es decir dos componentes unidos mediante un elemento de enlace. Por ello las acciones asociadas con este montajes son:
    1. Quitar el elemento de enlace ("tornillo 2").

2. Separar los componentes que estaban unidos por el elemento de enlace ("montaje<sub>1</sub>" y "carcasa interior").
  - m<sub>6</sub>: Este montaje es de tipo 3, debido a que los distintos montajes se pueden ensamblar en cualquier orden. Con lo que la acción asociada a este montaje es:
    1. No hacer nada.
  - m<sub>7</sub>: Este montaje corresponde al producto completo, y es de tipo 5, lo que quiere decir un montaje (m<sub>6</sub>) y un componente ("carcasa exterior") unidos mediante un elemento de enlace ("tornillo 1"). Lo que implica las siguientes acciones:
    1. Quitar el elemento de enlace ("tornillo 1"):
    2. Separar el componente ("carcasa exterior") del montaje (m<sub>6</sub>).

Con esto, ya se tienen preparado el grafo del producto para realizar el desensamblado del montaje m<sub>1</sub>; a continuación hay que seguir los pasos correspondientes al desensamblado de un componente individual (componente "montaje<sub>1</sub>") para calcular la secuencia de desensamblado. Ésta se calcula de la siguiente manera:

- 1) Paso 1 del algoritmo: Buscar los montajes a los que pertenece el componente "montaje<sub>1</sub>".
  - Este componente pertenece al montaje m<sub>2</sub> y m<sub>5</sub>.
- 2) Paso 2 del algoritmo: Almacenar las acciones asociadas con los montajes encontrados en el paso 1, es decir las acciones asociadas con el montaje m<sub>2</sub>:
  - Eliminar la unión entre los componentes.
  - Separar los componentes ("tarjeta 2" y "montaje<sub>1</sub>").
 Y las acciones asociadas con el montaje m<sub>5</sub> son:
  - Quitar el elemento de enlace ("tornillo 2").
  - Separar los componentes que estaban unidos por el elemento de enlace ("montaje<sub>1</sub>" y "carcasa interior").
- 3) Paso 3 del algoritmo: Como el montaje m<sub>2</sub> no es el producto completo, se sigue con el paso 1 del algoritmo, considerando como componente a desensamblar el montaje m<sub>2</sub>. Y como m<sub>5</sub> tampoco es el producto completo, hay que realizar lo mismo que para m<sub>2</sub>: seguir con el paso 1 del algoritmo. Para simplificar primero se desarrollan los pasos correspondientes al montaje m<sub>2</sub> y después se continúa con los del montaje m<sub>5</sub>.
- 4) Paso 1 del algoritmo: Buscar los montajes a los que pertenece el componente "montaje m<sub>2</sub>".
  - El montaje m<sub>2</sub> pertenece al montaje m<sub>6</sub>.
- 5) Paso 2 del algoritmo: Almacenar las acciones asociadas con el montaje encontrado en el paso anterior, las acciones asociadas con el montaje m<sub>6</sub>.



- No hacer nada.
- 6) Paso 3 del algoritmo: Debido a que el montaje  $m_6$  no es el producto completo entonces se sigue con el paso 1 del algoritmo, considerando como componente a desensamblar el montaje  $m_6$ .
  - 7) Paso 1 del algoritmo: Buscar los montajes a los que pertenece el componente "montaje  $m_6$ ".
    - El montaje  $m_6$  pertenece sólo al montaje  $m_7$ .
  - 8) Paso 2 del algoritmo: Almacenar las acciones asociadas con el montaje encontrado en el paso anterior del algoritmo, es decir las acciones asociadas con el montaje  $m_7$ .
    - Quitar elemento de unión "tornillo 1".
    - Separar el componente "carcasa exterior" del montaje " $m_6$ ".
  - 9) Paso 3 del algoritmo: Como el montaje  $m_7$  sí que se corresponde con el producto completo, entonces se sigue con el paso 4 del algoritmo.
  - 10) Antes de continuar con el paso 4 del algoritmo, hay que seguir la secuencia de desensamblado que se había dejado a medias en el punto 3 de este desarrollo: la secuencia para desensamblar  $m_5$ , ya que hasta ahora, se había calculado la de  $m_2$ , olvidándonos de la de  $m_5$ .
  - 11) Paso 1 del algoritmo: Buscar los montajes a los que pertenece el componente "montaje  $m_5$ ".
    - El montaje  $m_5$  pertenece al montaje  $m_6$ .
  - 12) Debido a que el montaje  $m_5$  pertenece al montaje  $m_6$  se dan las mismas acciones de desensamblado para este montaje  $m_6$  descritas anteriormente desde el punto 5 hasta el punto 9, con lo que ya se puede seguir con el último paso del algoritmo:
  - 13) Paso 4 del algoritmo: Realizar las acciones almacenadas en los sucesivos "pasos 2" del algoritmo, en sentido contrario, para llevar a cabo el desensamblado. Es decir las acciones a seguir para desensamblar el componente "montaje<sub>1</sub>" son:
    - Quitar elemento de unión "tornillo 1".
    - Separar el componente "carcasa exterior" del montaje " $m_6$ ".
    - No hacer nada.
    - Quitar el elemento de enlace ("tornillo 2").
    - Separar los componentes que estaban unidos por el elemento de enlace ("montaje<sub>1</sub>" y "carcasa interior").
    - Eliminar la unión entre los componentes.
    - Separar los componentes ("tarjeta 2" y "montaje<sub>1</sub>").

Estas acciones se pueden expresar en forma de grafo dirigido tal como se ve en la Figura 2.41, donde se ha eliminado la acción de "no hacer nada".

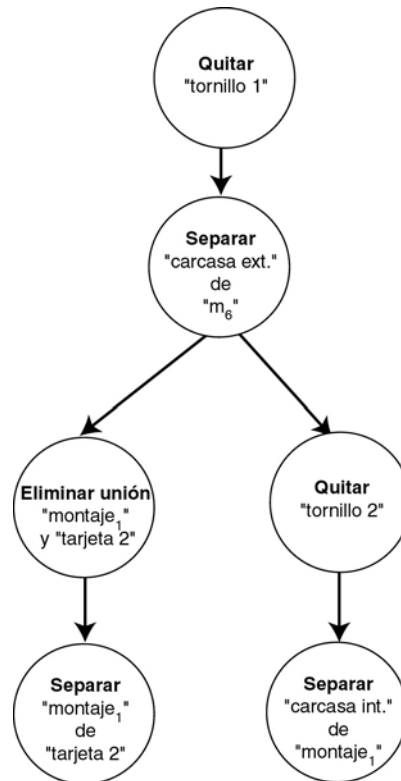


Figura 2.41: Grafo dirigido con las acciones a realizar para desensamblar el montaje  $m_1$  ("montaje1") del ordenador de la Figura 2.36.

## 2.7. Evaluación del algoritmo

En este apartado se contemplan las distintas posibilidades del algoritmo, teniendo en cuenta cuando es posible obtener una solución y cuando no.

El algoritmo planteado presenta solución siempre que el grafo de tareas este definido conforme a las reglas definidas anteriormente, es decir:

Todo montaje  $m_i$ , perteneciente al producto  $p$ , tiene una acción asociada  $a_i$ , que permite separar los componentes que lo forman.

Por lo tanto para todo montaje del producto se puede definir una secuencia de montajes, desde el que se desea desensamblar, hasta el montaje que representa el producto  $p$ . Si se tiene en consideración la premisa anterior, dicha secuencia tiene asociadas una serie de acciones que permite separar los componentes implicados en los montajes, y por tanto realizar la operación de desensamblado deseado.

Para demostrarlo se parte de que para que no se cumpla lo anterior hay dos opciones:

- El montaje  $m_k$ , perteneciente a la secuencia de desensamblado, no tiene asociada una acción de desensamblado.
- El montaje  $m_k$ , perteneciente a la secuencia de desensamblado, tiene asociada una acción que no permite separar los componentes que lo forman.

Como puede observarse claramente, la primera condición no se puede dar ya que ello implicaría que el producto no se ha definido conforme a las reglas correspondientes, por lo que no existiría una acción asociada al montaje  $m_k$ .

La segunda condición tampoco puede darse debido a que por definición todo montaje tiene asociada una acción de desensamblado que permite separar todos los componentes que lo forman.

Con lo que se llega a la conclusión que siempre que se pueda definir el producto utilizando las reglas dadas se puede calcular la secuencia de desensamblado del mismo.

Por otra parte, el coste del algoritmo viene definido por el número de montajes que forman el producto, ya que en el peor de los casos hay que desensamblar todos los montajes para llegar al componente objetivo, lo que implica una acción por cada uno de ellos.

# 3

## Generador de movimientos de desensamblado

### 3.1. Introducción

Un aspecto muy importante a la hora de realizar un proceso de desensamblado consiste en determinar cuál es la trayectoria a seguir por un componente para poder separarlo del resto del producto.

En este capítulo se tratan las distintas maneras de abordar este problema, para, en base a ellas, definir el método aportado en esta tesis para calcular los movimientos de desensamblado de un componente.

A la hora de realizar el desensamblado de un producto existen distintos planteamientos sobre cuál debe ser la secuencia de movimientos, el camino a seguir, por un componente para que se separe del resto del producto.

Una primera aproximación es la utilizada por los autores que sólo plantean la generación de la secuencia de planificación, por lo tanto la información sobre los movimientos a realizar por cada componente se encuentra almacenada en la base de datos [48] [106] [107] [162] [180] [202] [206] [213] [214].

Dentro de este tipo de métodos se encuentran principalmente aquellos que sólo intentan minimizar una función de coste sencilla [213] [214]; también se incluyen entre este tipo los que utilizan sistemas expertos [202]. Por consiguiente, debido a que estos métodos no planean una aproximación a la generación de movimientos de desensamblado, no se van a tratar en este apartado ya que dichos métodos se han tratado con mayor profundidad en el capítulo 2.

Otro planteamiento se presenta cuando se calcula automáticamente la dirección de separación de un componente en base a información sobre sus contactos con otros componentes, con lo que si un componente está unido a otro tiene una restricción de movimiento en una dirección determinada, con lo que según se desensamblan los componentes se obtienen nuevas direcciones libres en las que se puede desplazar un componente para separarlo del resto [133] [148] [150] [299] [300] [301] [308].

Otro conjunto de métodos son aquellos métodos que permiten generar la secuencia de desensamblado sin intervención del usuario, calculan las posibles direcciones de separación utilizando un modelo CAD del producto; es decir, utilizan un modelo geométrico del producto sobre el que se aplican una serie de operaciones que permiten obtener las direcciones de desensamblado [62] [160] [161] [243] [247] [248] [249] [250].

Llegados a este punto hay que realizar la definición del concepto de sistema  $n$ -desensamblado, que es:

*«Un sistema de desensamblado se denomina  $n$ -desensamblado si requiere, para separar un componente del resto,  $n$  movimientos en diferentes direcciones. Si un componente requiere  $n+1$  movimientos para separarse del resto entonces dicho componente se considera no desensamblable.»*

En la Figura 3.1 se muestra un ejemplo en el que el componente  $c_2$  es 2-desensamblable ya que requiere de dos traslaciones para poder separarse del resto del producto.

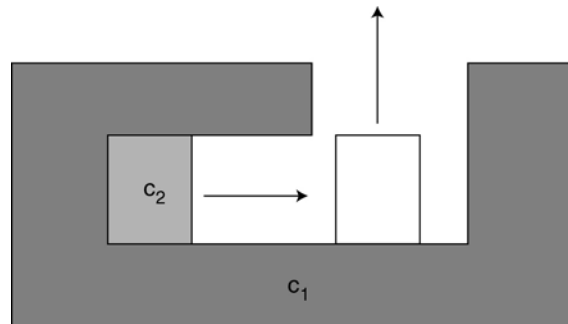


Figura 3.1: Representación de un componente 2-desensamblable.

Teniendo en cuenta esta definición la mayoría de los sistemas de desensamblado, son del tipo 1-desensamblado, es decir, se requiere un solo movimiento para desensamblar un componente [62] [89] [90] [91] [133] [240] [247] [248] [249] [250].

En esta tesis se aporta una solución al problema del  $n$ -desensamblado, utilizando para ello una representación geométrica del producto basada en un modelo CAD, ya que ello permite el cálculo automático de los movimientos de desensamblado.

Este capítulo se divide de tal manera que: en el siguiente apartado se explica el método de representación utilizado para llevar a cabo el cálculo de la trayectoria de desensamblado para un solo componente. A partir de esta, se presenta cómo calcular la dirección de desensamblado de un componente. Tras esto, se explica cómo calcular la trayectoria completa de desensamblado del componente y como detectar las colisiones con otros componentes. Por último se muestra un ejemplo del cálculo de la trayectoria de desensamblado

para un producto en concreto así como una evaluación en cuanto a costes computacionales del algoritmo planteado.

## 3.2. Representación geométrica del producto

La representación que se utiliza del producto debe permitir la generación automática de los movimientos de desensamblado, es por ello que se presenta a continuación un breve estado del arte en lo que respecta a representación geométrica de los productos. Tras él se plantea el método utilizado para convertir la información geométrica en un grafo, que se corresponde con el método utilizado en esta tesis.

### 3.2.1. Estado del arte

Las posibles representaciones de un producto mediante un programa CAD se pueden resumir en tres:

1. Utilización de un modelo basado en sólidos, lo que implica el manejo de modelos geométricos complejos y diferentes para cada componente del producto.
2. Un modelo poliédrico del producto, en el que cada componente se define como un poliedro. Este método simplifica el tratamiento de los distintos componentes ya que todos se simplifican en última instancia a un conjunto de polígonos [248] (Figura 3.2).

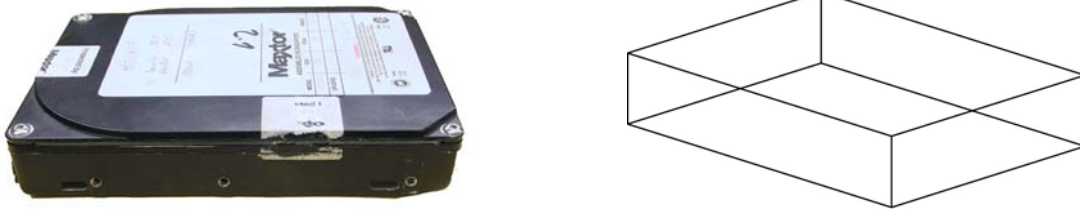


Figura 3.2: Disco duro y su representación simplificada mediante polígonos.

3. Un modelo basado en elementos finitos. Este modelo planea una solución bastante precisa en cuanto al diseño de los componentes, pero conlleva un cálculo complejo [42] (Figura 3.3).

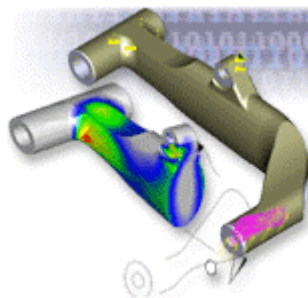


Figura 3.3: Representación de un componente utilizando elementos finitos (Cortesía de HEC).

Para que estas representaciones estén correctamente especificadas habría que incluir dos definiciones, la de polígono y la de poliedro [44]:

Un polígono se define como:

«Figura plana limitada por una línea poligonal cerrada. Los segmentos de la línea poligonal se llaman lados. Según el número de lados, los polígonos se

*clasifican en: triángulos, cuadriláteros, pentágonos, hexágonos, heptágonos, octógonos, etc.»*

Basándose en esta definición, se puede definir un poliedro como:

*«Sólido limitado por polígonos llamados caras. La intersección de dos caras es una arista. Los puntos de intersección de 3 o más caras forman un vértice.»*

Teniendo en cuenta estas características hay que indicar que la más utilizada consiste en la utilización del modelo poligonal del producto ya que permite una generalización mayor a la hora de definir las operaciones para el cálculo de la secuencia de movimientos.

Esto es debido a que todo programa CAD actual puede convertir cualquier modelo por complejo que sea a un conjunto de polígonos, normalmente triángulos, que definan los distintos componentes.

El problema que aparece a la hora de utilizar un modelo poliédrico de los productos consiste en la pérdida de resolución, ya que las superficies que no son planas: tubos, tornillos, etc., se tienen que transformar en superficies planas, realizando una aproximación de sus contornos. Esta aproximación se puede ajustar, en la mayoría de programas CAD, según el error que se pueda tolerar en el modelo, de manera que para obtener una mayor precisión simplemente se aumenta el número de polígonos utilizados para convertir el modelo CAD a un modelo poliédrico (Figura 3.4).

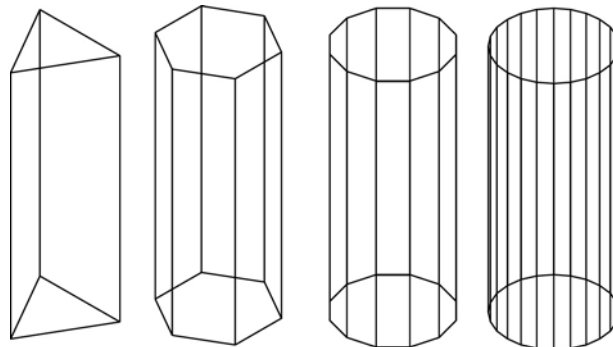


Figura 3.4: Modelo poliédrico de un cilindro utilizando 5, 8, 14 y 26 polígonos.

### 3.2.2. Representación utilizada

El planteamiento que se utiliza en esta tesis consiste en usar un modelo geométrico del producto. Este modelo se basa en un modelo CAD de cada uno de los componentes del producto a desensamblar, así como de las relaciones de posición existentes entre ellos. Utilizando dicho modelo se calcula la mejor dirección de desensamblado, y se realiza un desplazamiento lineal del componente, tras lo que se actualiza el modelo y se permite realizar al componente una secuencia de  $n$  movimientos lineales hasta que se consigue desensamblarlo, con lo que está planteando un sistema  $n$ -desensamblado, el cuál es una de las aportaciones originales de esta tesis.

Además se dispone de varios componentes que forman un producto, por lo tanto, no basta con tener el modelo individual de cada uno de ellos por separado, sino que hay que conocer cuál es la localización espacial de unos respecto a otros para tener un correcto modelado del producto.

Para representar, mediante un grafo, las caras existentes en un componente, se utiliza un grafo de adyacencia (Figura 3.5), donde se representa la relación de un componente con las caras que lo forman [161].

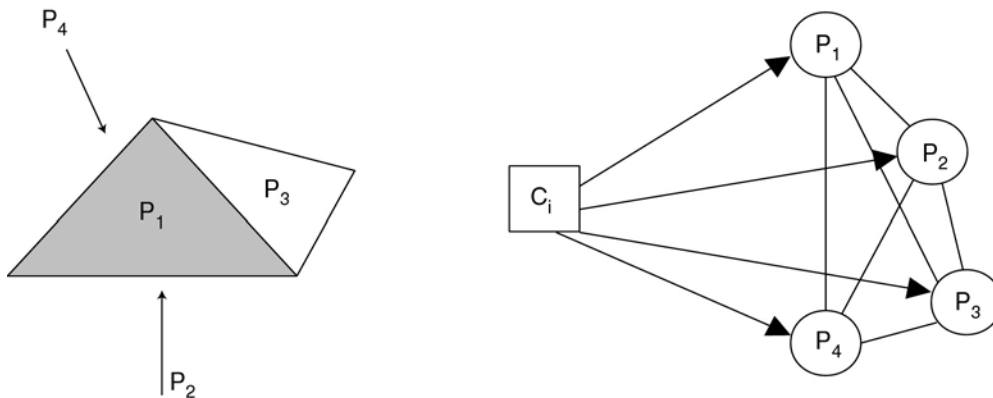


Figura 3.5: Componente polédrico  $C_i$  de cuatro caras ( $P_1, P_2, P_3, P_4$ ) y el grafo de caras de adyacencia.

Una vez se ha definido el concepto de grafo de adyacencia se definen los grafos de adyacencia correspondientes a cada uno de los componentes del producto.

Con estos grafos hay que crear el grafo de caras de contacto, el cual representa las relaciones existentes entre las caras de los distintos componentes; es decir, si dos componentes están en contacto dicha información se verá reflejada en el grafo de caras de contacto.

Este grafo se genera partiendo de los grafos de adyacencia de todos los componentes del producto y añadiendo una arista entre cada par de nodos, los nodos que representan las caras, que poseen una relación de contacto entre ellos.

De esta manera en el nuevo grafo aparecen reflejadas las relaciones, ya no de un solo componente, sino entre todos los componentes del producto.

Para ilustrar estos conceptos se utiliza el producto compuesto por los tres componentes polédricos de la Figura 3.6.

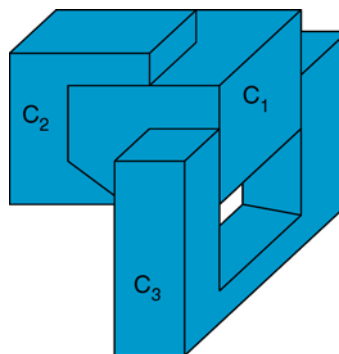


Figura 3.6: Producto compuesto por tres componentes.

En la Figura 3.7 se puede ver el grafo de adyacencia para el primer componente del producto de la Figura 3.6. En el que se ha simplificado el grafo, para una mejor visualización, eliminando las aristas que representan un contacto entre las caras del propio componente.



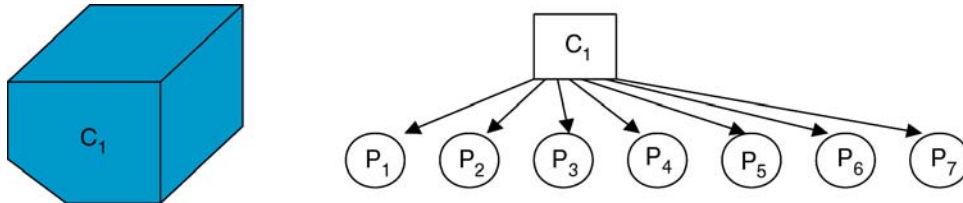


Figura 3.7: Grafo de adyacencia para el componente  $C_1$  del producto de la Figura 3.6.

La Figura 3.8 presenta el grafo de caras de contacto que resulta de unir el componente de la Figura 3.7 con el del segundo componente del producto.

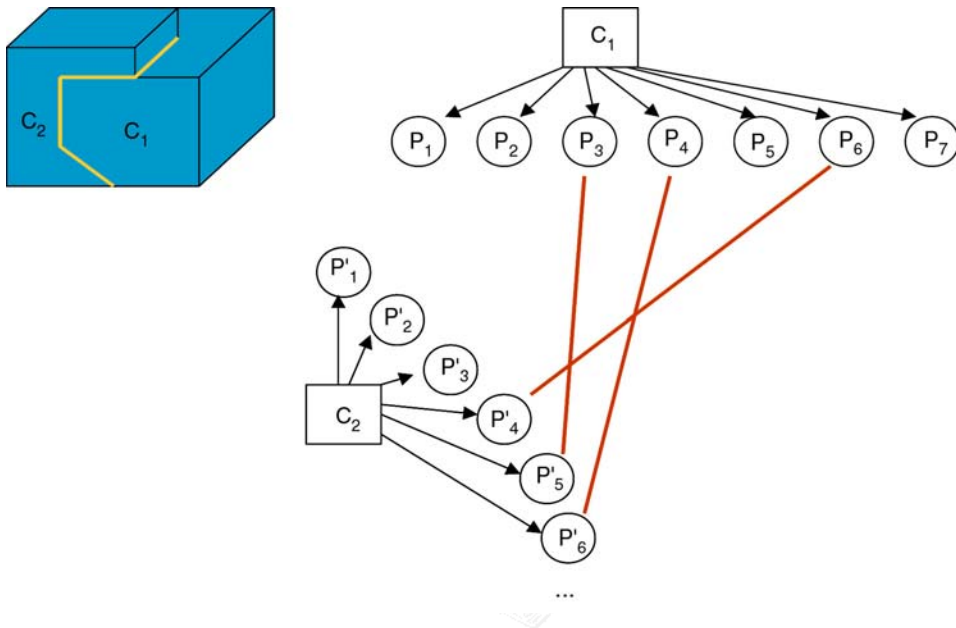


Figura 3.8: Grafo de caras de contacto para los componentes  $C_1$  y  $C_2$  del producto de la Figura 3.6.

En la Figura 3.9 se presenta el grafo de caras de contacto resultante de añadir a los dos componentes de la Figura 3.8 el tercer componente, con lo que se obtiene el grafo de caras de contacto completo del producto.

Cada arista que representa un contacto posee un descriptor que indica en cada momento qué tipo de contacto se está produciendo. En el esquema descrito (Figura 3.9) se observa como el tipo de contacto existente entre las caras de los componentes, es de tipo plano.

También se pueden producir otros tipos de contacto como de rosca entre un tornillo y un componente, de tal manera, que el tipo de contacto implica una u otra secuencia de movimientos de desensamblado que se pueden almacenar en una base de datos. Por ejemplo el contacto plano requiere un desplazamiento, un contacto de rosca requiere un desplazamiento y simultáneamente una rotación, etc.

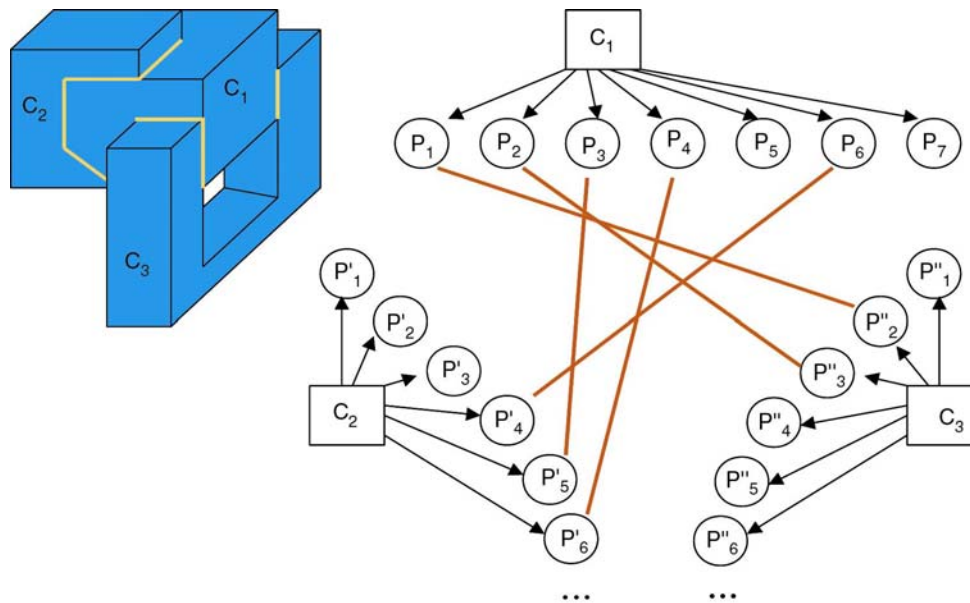


Figura 3.9: Grafo de caras de contacto para el producto de la Figura 3.6.

En el estudio que se va a realizar, las direcciones de desensamblado se calculan basándose en la existencia de contactos planos; si el contacto es de otro tipo, entonces la dirección de desensamblado vendrá indicada por el tipo de contacto, dado que no implicará desplazamientos lineales de los componentes.

### 3.3. Determinación de la dirección de separación de un componente

En este apartado se determina la dirección de separación de un componente respecto del resto del producto. En primer lugar, a partir de la información disponible en el grafo de caras de contacto, se establece un conjunto de posibles direcciones de separación, desde el que, finalmente, se determina una dirección concreta.

#### 3.3.1. Estado del arte

A partir del grafo de caras de contacto entre los componentes del producto, hay que determinar los movimientos relativos de los componentes para su desensamblado. Estos desplazamientos corresponden, principalmente, con traslaciones que son modeladas a partir de relaciones entre componentes. Para abordar este problema existen dos métodos principales: el concepto de esfera unidad y las representaciones cinemáticas.

##### 3.3.1.1. Esfera unidad

El concepto de esfera unidad, o esfera de Gauss es utilizado ampliamente y presentado en detalle en [244], [288] y [289]. Permite determinar los desplazamientos de traslación posibles de un componente en función de sus contactos. Los componentes son modelados utilizando una representación poliédrica, que permite definir una representación polinomial de los planos de contacto, que son necesarios para este método. La idea básica consiste en que

cada cara de contacto entre dos superficies divide la esfera en dos hemisferios.

Una característica importante a tener en cuenta es que este método no tiene en cuenta explícitamente los movimientos de rotación de los componentes.

Otro planteamiento [156] [225] [287] consiste en: a partir de una representación poliédrica de los componentes y del análisis de los contactos planos generar un grafo denominado como: *Non-Directional Blocking Graph*; que representa las direcciones en las que se puede desplazar cada componente.

También existen extensiones de esta idea que consideran los contactos cilíndricos y las dos direcciones de traslación asociadas a este tipo de contacto [244].

Una aproximación más detallada incluye, además, la posibilidad de tener en cuenta los ejes de rotación admisibles para los contactos planos [182]. Las rotaciones son determinadas a partir de la geometría del contacto, es decir, de las dimensiones de las superficies planas y son representadas con una esfera unidad.

### 3.3.1.2. Representaciones cinemáticas

Dentro de un sistema mecánico, la puesta en contacto de dos componentes implica condiciones o limitaciones de los valores posibles de sus parámetros de desplazamiento relativo. La unión así constituida viene caracterizada por la superficie o superficies de contacto entre los sólidos modelados de los componentes.

Los modelos de uniones suelen ser definidos por un modelo de representación de los grados de libertad correspondientes a los movimientos autorizados por la unión. La representación de los grados de libertad utilizando una notación cinemática permite distinguir explícitamente la orientación y la dirección de los movimientos relativos de traslación y de rotación de un contacto que son necesarios para romper este contacto.

En este caso se representan las uniones en forma parecida a una matriz de transformación homogénea que se denomina "pseudomatriz" o "freedom matrix". Esta matriz representa las transformaciones relativas necesarias para la eliminación de un determinado componente [155] [181]. Las traslaciones representadas en la pseudomatriz se pueden determinar automáticamente a partir del análisis de los modelos CAD mientras que las rotaciones son completadas manualmente si son necesarias para realizar el desensamblado [155] (Figura 3.10).

El concepto de esfera unidad es el más general y permite determinar las direcciones de traslación posibles para el desensamblado. Sin embargo, es difícil de extender para todos los tipos de contacto y de tener en cuenta los movimientos de rotación.

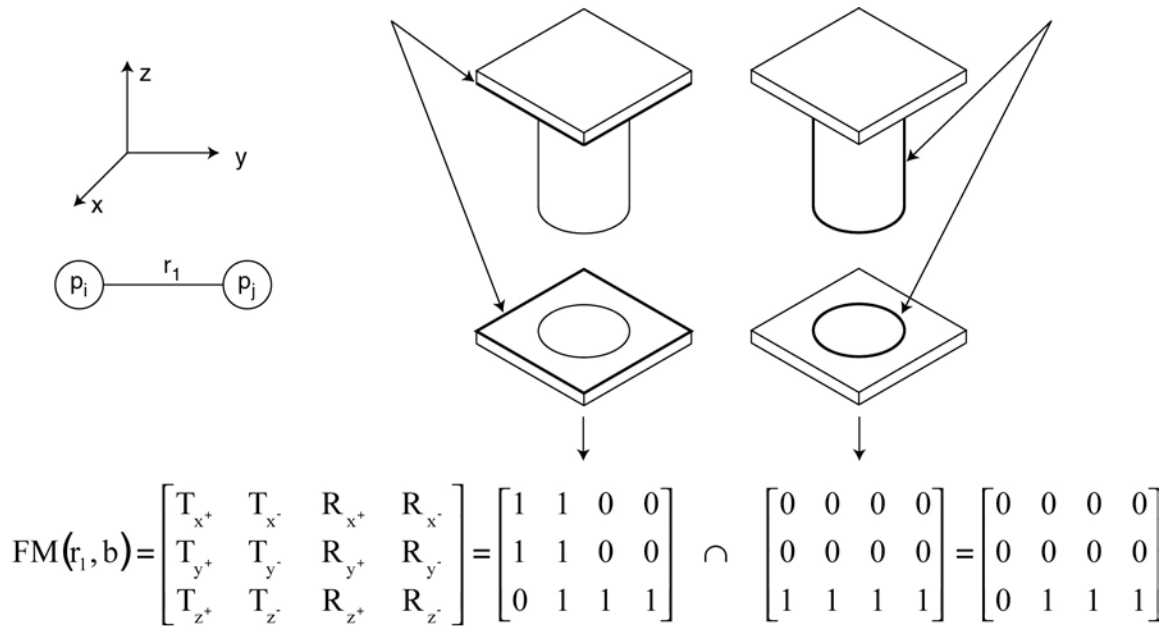


Figura 3.10: Ejemplo de aplicación de la técnica "freedom matrix".

### 3.3.2. Conjunto de direcciones de separación

La dirección de desensamblado de un componente consiste en calcular cuál es la dirección en la que se puede desplazar, para conseguir separarlo del resto. Este concepto presupone que, con un solo movimiento lineal, se puede separar el componente deseado; si se necesita más de un movimiento o movimientos no lineales (Figura 3.11), la dirección obtenida no sirve directamente para desensamblar el componente.

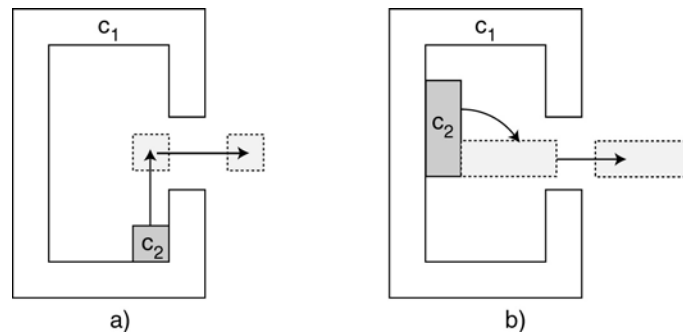


Figura 3.11: Combinación de movimientos. a) Se necesitan dos movimientos lineales para desensamblar  $c_2$ . b) Se necesita un movimiento de rotación y uno lineal para desensamblar el componente  $c_2$ .

También se presupone que todas las caras de los poliedros que pueden impedir un movimiento, se encuentran adyacentes unas a otras. Una cara que no tenga contacto con otra, no impide el movimiento (Figura 3.12).

Con estos dos conceptos, el sistema calcula una dirección de desensamblado para un determinado componente.

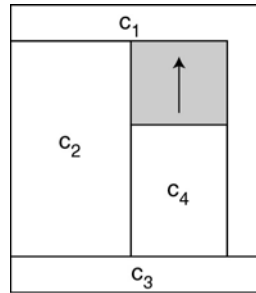


Figura 3.12: Los componentes  $c_1$  y  $c_4$  poseen una restricción que impide su movimiento en una determinada dirección (zona gris), pero no existe un contacto directo en esa dirección que indique la restricción.

Para poder llevar a cabo el desensamblado del componente  $c_1$  de la Figura 3.6, al resto de componentes que forman el producto ( $c_2$  y  $c_3$ ), se les considera como si de un único componente se tratara, denominado componente grupo residual (Figura 3.13), conocido en la bibliografía como *residual component group (RCG)* [248].

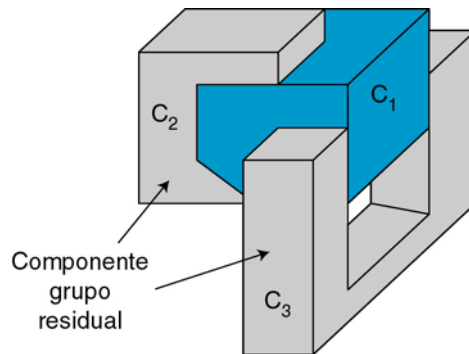


Figura 3.13: Representación del componente grupo residual para desensamblar el componente  $c_1$ .

Para separar un componente del grupo residual, lo primero a realizar consiste en calcular cuáles son las caras de contacto entre ambos componentes, denominadas en la bibliografía como *interacting surfaces (IS)*. Las caras de contacto restringen las direcciones en las que pueden desensamblarse dos componentes.

Un aspecto importante a tener en cuenta es que las caras de contacto siempre aparecen por pares, es decir, si una cara del componente  $c_1$  tiene contacto con una cara del componente  $c_2$ , entonces, el componente  $c_2$  tiene el contacto conjugado con el componente  $c_1$ . La cara de contacto conjugada se expresa como  $IS^*$ . Cada par de caras adyacentes  $IS/IS^*$  recibe el nombre de grupo de superficies con interacción, denominada en la bibliografía como *set of interacting surfaces (SIS)*.

Las caras de contacto entre los distintos componentes del producto de la Figura 3.6 se muestran en la Figura 3.14.

Un grupo de superficies con interacción, expresa una restricción en el movimiento de un componente respecto a otro, de manera que define una serie de direcciones posibles de separación para el componente, denominado en la bibliografía como *set of possible directions of separation (SDS)*.

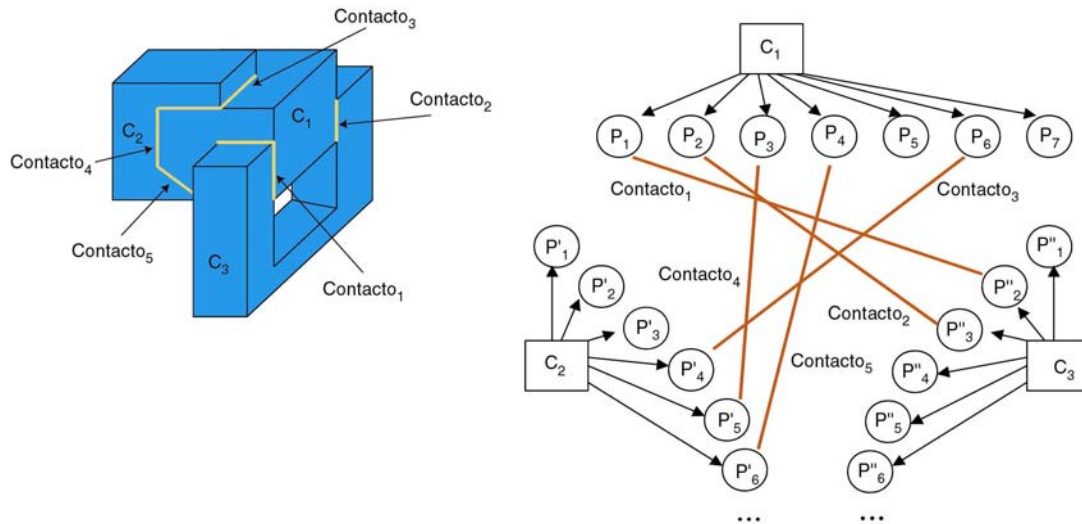


Figura 3.14: Representación de los contactos entre las caras del producto.

Las posibles direcciones de separación de un componente se pueden expresar en el espacio tridimensional como una esfera gaussiana [61], que representa todas las direcciones posibles de movimiento si el componente estuviese libre en el espacio 3D. Esta esfera representa todos los posibles vectores del espacio; cada vector es una dirección de desensamblado, para los que es posible realizar un movimiento del componente a desensamblar. De esta manera, y según las restricciones impuestas por el grafo de caras de contacto, se va cortando la esfera, hasta dejar las direcciones válidas para desensamblar un determinado componente (Figura 3.15).

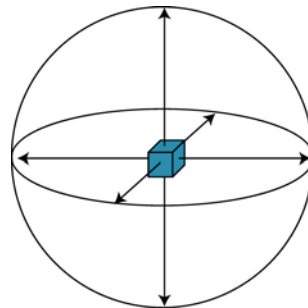


Figura 3.15: Esfera gaussiana para un componente libre en el espacio tridimensional.

Si se tiene en cuenta un contacto del componente por su cara inferior con una superficie plana, otro componente, una mesa de trabajo, etc.; entonces las direcciones posibles de desensamblado se reducen a una semiesfera (Figura 3.16).

Aplicando las definiciones anteriores se obtiene que el grupo de direcciones de separación de dos componentes, que tienen un contacto plano entre dos de sus caras, se reduce a una semiesfera.

De esta manera se pueden definir matemáticamente las posibles direcciones de desensamblado de un componente que tiene un contacto plano con otro de la siguiente manera:

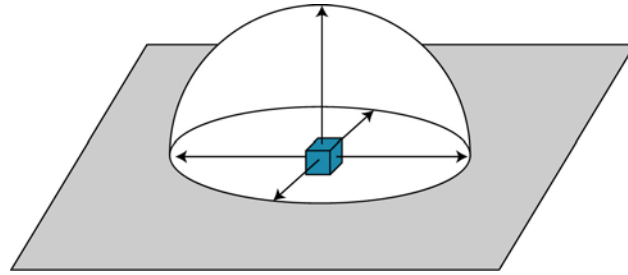


Figura 3.16: Semiesfera que define las posibles direcciones de desensamblado de un componente que posee un contacto plano con otro.

Considérese  $\mathbf{n}=(n_1, n_2, n_3)$  como el vector normal a la superficie de contacto, cuyo sentido se corresponde a ir desde el punto de contacto en la superficie hacia el componente; y si se representa por  $\mathbf{x}=(x_1, x_2, x_3)$  al vector que define una posible dirección de desensamblado, entonces se tiene que, para que una dirección sea válida, debe cumplir:

$$\mathbf{n} \times \mathbf{x} = n_1x_1 + n_2x_2 + n_3x_3 > 0 \quad (3.1)$$

Si en lugar de tener un único contacto plano existen múltiples contactos planos, para que un vector  $\mathbf{x}$  sea una dirección de desensamblado válida, debe de cumplir:

$$\sum_{j=1}^3 n_{ij}x_j > 0 \quad \text{para } i = 1, 2, \dots, n \quad (3.2)$$

Donde  $n_{ij}$  son las componentes de los vectores  $\mathbf{n}_i$  normales a cada uno de los planos de contacto.

De esta manera, para que un determinado componente tenga una solución válida, debe de tener al menos una solución para el vector  $\mathbf{x}$  en la ecuación 3.2.

El conjunto de posibles direcciones en las que un componente puede ser desensamblado, se denomina como conjunto de direcciones de separación, denominado en la bibliografía como *set of directions of removal (SDR)*. Este conjunto de direcciones se calcula realizando la intersección de todas las direcciones de separación del componente.

En la Figura 3.17 se muestra la unión de las direcciones de desensamblado obtenidas para cada contacto, según se va reduciendo la esfera por las direcciones posibles de desensamblado.

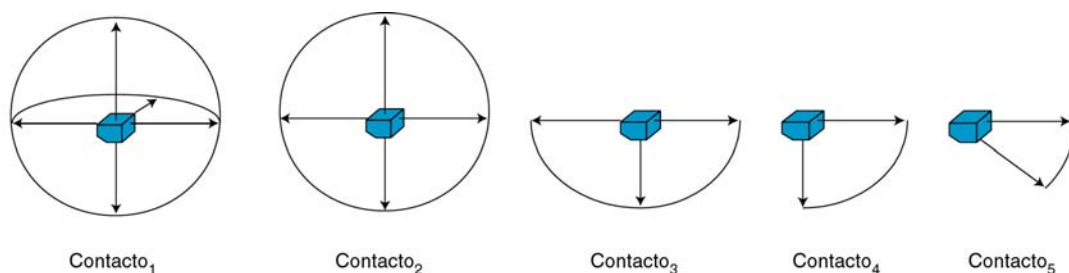


Figura 3.17: Unión de las direcciones de desensamblado del componente  $c_1$  para obtener el conjunto de direcciones de separación.

Aplicando estos resultados al desensamblado del componente  $c_1$  se obtiene que, para desensamblarlo, se puede desplazar hacia la derecha, hacia la parte inferior derecha o en cualquier dirección comprendida entre ambas; según el rango de direcciones posibles que han quedado de la esfera de la Figura 3.17

### 3.3.3. Determinación de una dirección de separación

De entre todas las posibles direcciones de separación, determinadas a partir del grafo de caras de contacto, se selecciona una de ellas, considerando las localizaciones del espacio en las que hay menor probabilidad de que el componente a desensamblar colisione con el resto del producto, con el que no mantiene contactos físicos.

Como se puede apreciar en la Figura 3.18 el algoritmo determina unas direcciones de desensamblado posibles; pero dado que el obstáculo no posee una cara de contacto con el componente a desensamblar, entonces existen direcciones de la solución que plantean una mayor probabilidad de colisión que las demás.

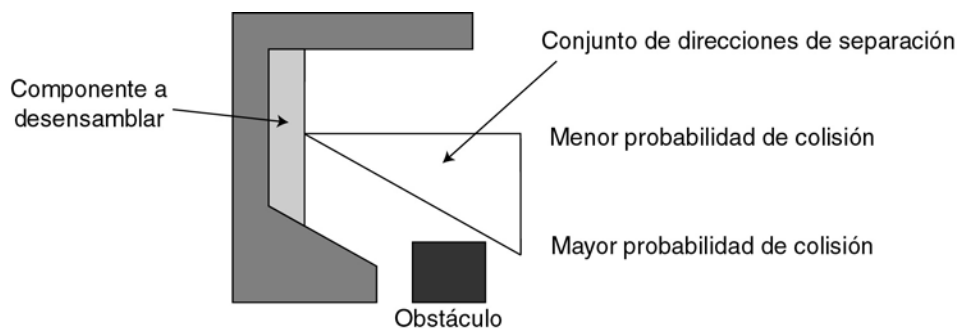


Figura 3.18: Probabilidad de colisión en un determinado conjunto de direcciones de desensamblado.

Al llegar a este punto hay que determinar cuál es la dirección en la que se desplaza el componente para realizar su desensamblado, evitando las colisiones con los objetos con los que no posee un contacto.

Para solucionar este problema, se plantea un algoritmo que permite el cálculo de dónde se pueden producir las colisiones. Para ello, el entorno de trabajo se discretiza en celdas cúbicas (Figura 3.19), de tal manera que, en cada celda se indica la probabilidad de encontrar un obstáculo; así es posible determinar cuál es la dirección en la que existe una menor probabilidad de encontrar otro componente.

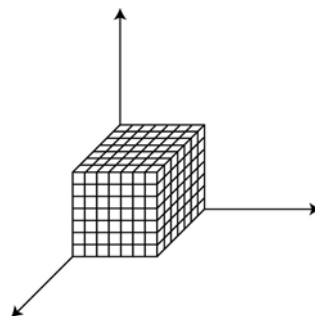


Figura 3.19: Discretización del entorno de trabajo en celdas cúbicas.



En cada celda se acumula información acerca de la existencia de posibles obstáculos en esa posición, conteniendo el valor 1 cuando haya certeza de que se encuentre un obstáculo en esa posición, y 0 cuando no la haya.

Para determinar la existencia de un obstáculo, se simula una técnica similar a la empleada por ultrasonidos o láser en robótica móvil. Es decir, a partir del modelado geométrico existente en ese momento, se simula la existencia de un haz esférico de rayos, que parte del centro del componente a desensamblar, indicando el valor de la detección para cada rayo en las coordenadas correspondientes del mundo discretizado.

Con la información almacenada, se crea un histograma que indica, para cada dirección del conjunto determinado a partir de las caras de contacto, la probabilidad de que exista un obstáculo. El histograma es generado a partir de la información obtenida dentro de una esfera de un cierto radio preestablecido, de tal manera que, a mayor radio, más información se tendrá en cuenta; pero por el contrario, mayor será el coste computacional para realizar los cálculos.

Para generar el haz de rayos, se ha seleccionado hacerlo desde el centro geométrico del componente. Dicho punto de referencia puede ser perfectamente cualquier punto del componente, ya que lo que se pretende con él es aumentar el conocimiento del entorno. Dicha información será más o menos precisa según la elección del punto de referencia, pero siempre válida. Una alternativa interesante consiste en generar una semiesfera de rayos desde el centro de cada una de las caras del componente, con lo que se obtiene una mayor información del entorno, pero se incurre en un mayor coste computacional, que se incrementa rápidamente con la utilización de componentes complejos.

Ya que si se posee un componente con  $n$  caras, y se lanza un haz de  $m$  rayos desde el centro del componente, en caso de lanzar el mismo haz desde cada cara, se tendrían  $n \cdot m$  rayos, lo que incrementa claramente el coste computacional y está claro que se tendrá una mayor información del entorno, debido a que se han lanzado mayor cantidad de rayos.

A continuación se emplea una estrategia similar a la empleada en [216]. Se generan unas fuerzas de repulsión y atracción sobre el componente que se está separando. Las fuerzas de repulsión en una dirección resultan proporcionales a los valores de probabilidad almacenados en el histograma para esa dirección e inversamente proporcional a la distancia del obstáculo. Las fuerzas de atracción se generan desde las direcciones libres de obstáculos. Combinando ambas, se determina la dirección de menor repulsión, dentro de las comprendidas en el conjunto de direcciones de separación, obtenidas por el grafo de caras de contacto.

A continuación se detallan los pasos a seguir para calcular la dirección en la que existe una menor probabilidad de encontrar un obstáculo; es decir, menor repulsión:

1. Una vez lanzados los rayos desde el componente, hay que determinar las casillas del mundo discretizado donde existe un obstáculo, es decir, donde el rayo colisiona con otro componente. La posición del punto de choque se expresa en coordenadas, respecto al centro del componente que se está considerando. De tal manera que se asigna un valor de 1 a aquellas casillas en las que existe un

obstáculo. Un ejemplo de estos conceptos se ilustra en la Figura 3.20 en representación bidimensional.

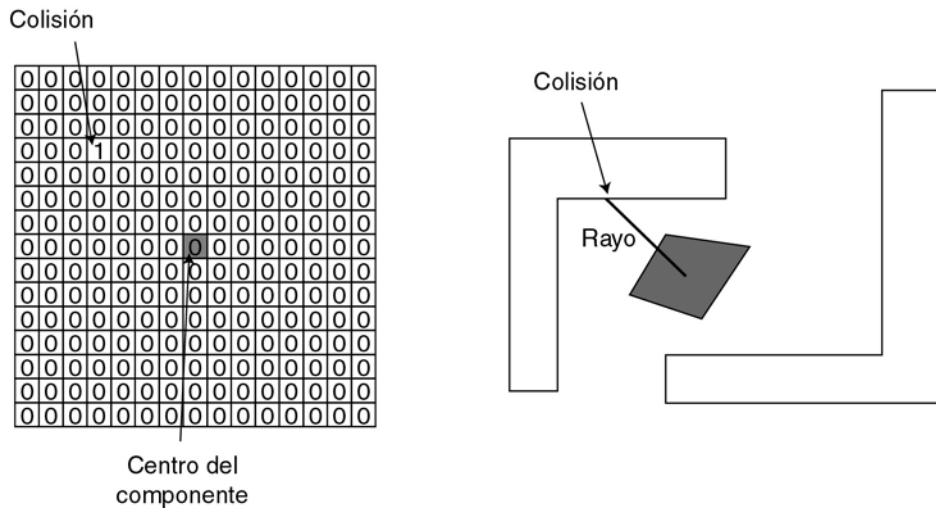


Figura 3.20: Detección de colisiones e incremento del mundo discretizado.

Tras enviar un haz de rayos se obtiene para el mundo discretizado una mayor información sobre el entorno del componente. Además hay que tener en cuenta que la zona a considerar para generar la trayectoria, se reduce a una distancia máxima de  $r'$  (Figura 3.21)

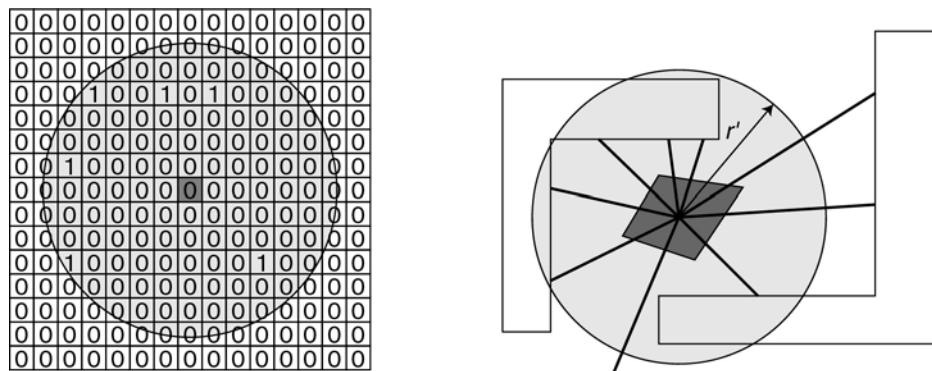


Figura 3.21: Mundo discretizado tras un haz de rayos.

2. El siguiente paso del algoritmo consiste en la generación del histograma del mundo discretizado, que indica la distancia y posición de los objetos en el mundo discretizado. Para generar el histograma, las coordenadas cartesianas del mundo discretizado  $(x, y, z)$  se convierten a coordenadas esféricas  $(r, \theta, \phi)$  (Figura 3.22), considerando el origen de coordenadas en el punto que representa el centro del componente.

$$r = \sqrt{x^2 + y^2 + z^2} \tag{3.3}$$

$$\theta = \arctan \frac{y}{x} \tag{3.4}$$

$$\phi = \arctan \frac{\sqrt{x^2 + y^2}}{z} \quad (3.5)$$

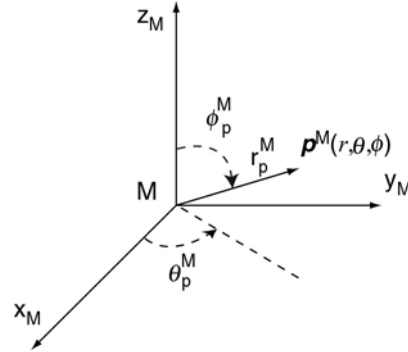


Figura 3.22: Transformación a coordenadas esféricas.

En base a esta información en coordenadas esféricas, se genera información sobre los obstáculos existentes en una determinada dirección, es decir para cada par  $(\theta, \phi)$  se calcula la fuerza de repulsión del objeto en la dirección correspondiente, que es inversamente proporcional a la distancia; a mayor distancia menor es la fuerza de repulsión. Por lo tanto la fuerza de repulsión viene definida como:

$$\mathbf{f}(\theta, \phi) = \begin{bmatrix} f_x(\theta, \phi) \\ f_y(\theta, \phi) \\ f_z(\theta, \phi) \end{bmatrix} = - \sum_{\forall p / p(r, \theta, \phi) \neq 0} \frac{\text{hist}(\theta, \phi)}{r} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (3.6)$$

Donde el  $\text{hist}(\theta, \phi)$  representa el valor del histograma para dichos ángulos.

- Una vez se han calculado las fuerzas de repulsión, se añade un conjunto de fuerzas de atracción, que se corresponden con cada dirección en la que un rayo del haz no ha colisionado con otro componente. Por lo tanto estas fuerzas se suman a las anteriores de repulsión para establecer una dirección de desensamblado de entre el conjunto dado previamente por el grafo de caras de contacto.

Una vez seguidos estos pasos, se tiene generada la dirección de desensamblado a realizar, que se corresponde con la resultante de sumar todas las fuerzas obtenidas.

En la Figura 3.23 se muestra un ejemplo bidimensional de la determinación de la dirección de separación de un componente respecto al resto, en el que se representa el mundo discretizado y el histograma obtenido, a partir del cual es posible determinar la fuerza de repulsión y la dirección de separación.

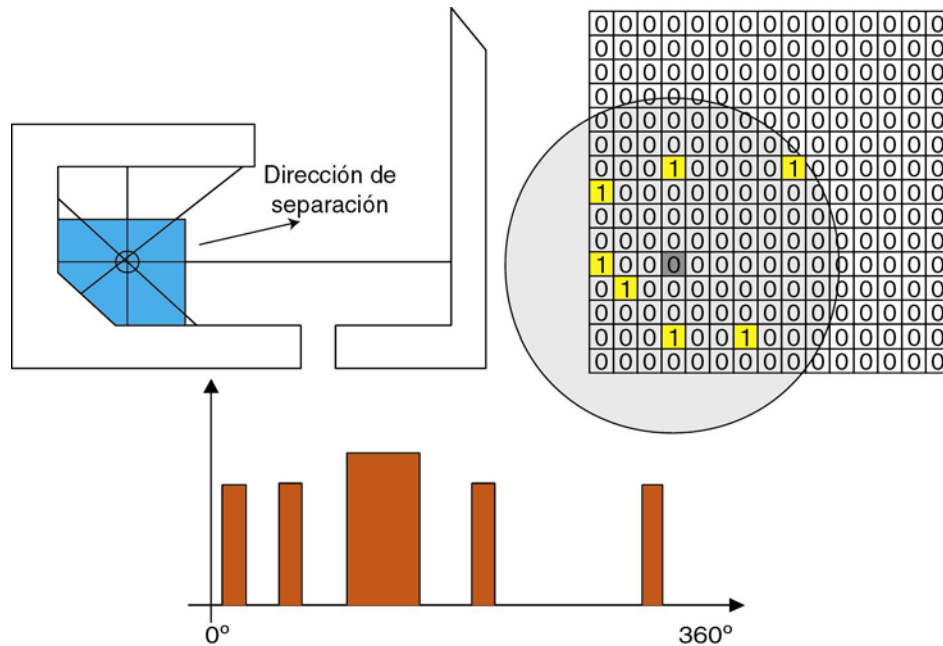


Figura 3.23: Determinación de la dirección de separación a partir de la obtención de la máxima fuerza de repulsión.

### 3.4. Generación de una trayectoria de desensamblado

Una vez determinada la dirección de separación de un componente del resto del montaje, ésta será válida si no existe ningún obstáculo en esa dirección. Por el contrario si existen obstáculos que impidan un desensamblado del componente en esa dirección, (Figura 3.23) se necesita determinar un conjunto de movimientos de separación simples, según la metodología explicada en el anterior apartado, que constituyan una trayectoria cartesiana para desensamblar un componente [211].

#### 3.4.1. Estado del arte

En los estudios de accesibilidad de un componente en el contexto de desensamblado o de mantenimiento se emplean principalmente dos técnicas: búsqueda automática de trayectorias y la simulación interactiva con la ayuda de herramientas de robótica o de realidad virtual.

##### 3.4.1.1. Búsqueda automática de trayectorias

Estas técnicas están basadas en el conocimiento del entorno de los componentes a manipular. Pueden ser de tres tipos:

- Globales: basadas en un conocimiento global del entorno.
- Locales: utilizan sólo una percepción local del entorno.
- Mixtas: usando principios locales y globales.

El entorno es denominado espacio de configuraciones. Este espacio, denominado  $C$ , es el conjunto de todas las configuraciones y contiene, por tanto:

- Espacio libre, denominado  $C$ -libre, representa el conjunto de las configuraciones sin colisión entre los componentes y los obstáculos.

- El espacio de los obstáculos, denominado C-obstáculos, representa el conjunto de configuraciones que provocan colisiones. C-obstáculos es el complementario de C-libre en C.

En la Figura 3.24 se puede ver una división de las búsquedas automáticas de trayectorias [218]:

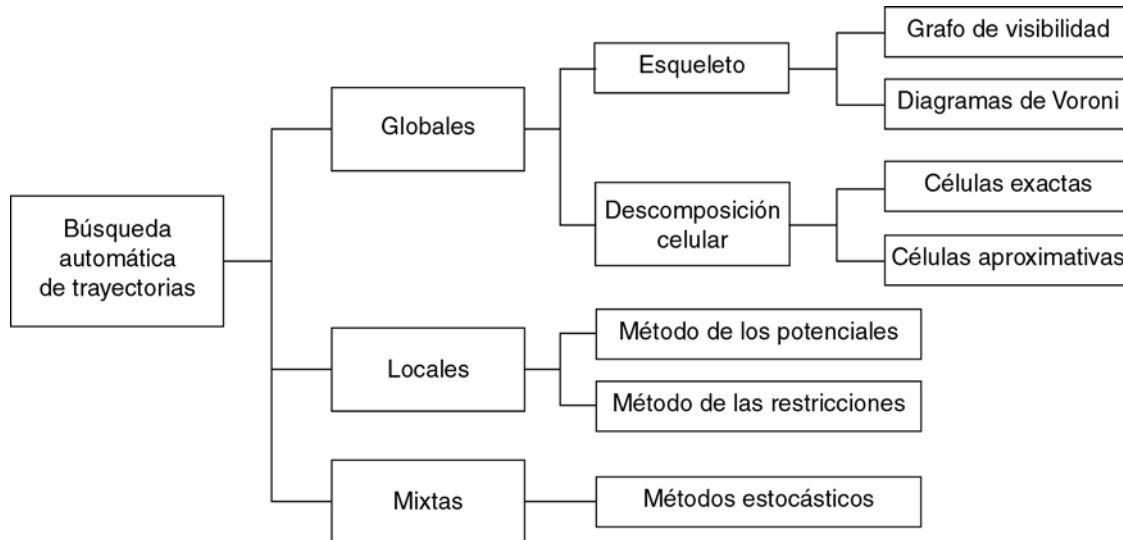


Figura 3.24: Técnicas de búsqueda automática de trayectorias (Fuente: [218]).

## Globales

Las técnicas globales están basadas en la discretización del espacio de configuraciones. Una solución global construye una representación del espacio de configuraciones a partir del modelo completo del entorno. En general, este tipo de aproximación transforma el problema de búsqueda de trayectorias en un problema de búsqueda en un grafo. Se pueden clasificar en dos categorías:

- Esqueleto: intenta construir una representación del espacio de configuraciones con subespacios de dimensión más pequeña que el espacio de configuraciones. Las propiedades de conexión son guardadas en una nueva representación llamada esqueleto. En este planteamiento se utiliza una representación explícita de los espacios libres conexos. El conjunto de trayectorias en el espacio libre es almacenado en un grafo de manera que este grafo representa todas las regiones del espacio libre. De esta manera toda trayectoria en el espacio libre está representada por un conjunto de arcos y de nodos. Para determinar una trayectoria de una configuración inicial a otra final, basta con proyectar las configuraciones inicial y final sobre dos nodos del grafo. A continuación se utiliza un método de búsqueda para recorrer el grafo y encontrar el conjunto de nodos y arcos que describen la trayectoria. Se tienen, principalmente, dos planteamientos para el cálculo de los espacios libres:
  - Grafos de visibilidad: utilizados para entornos compuestos por obstáculos poligonales. El grafo es entonces construido con los nodos de los componentes y las configuraciones inicial y final [282].
  - Diagramas de Voronoi: construyendo el grafo libre a partir de puntos equidistantes entre los obstáculos [258].

- Descomposición celular: consiste en dividir el espacio de búsqueda en subespacios denominados células. La búsqueda de una trayectoria consiste entonces en la construcción de un camino entre dos subespacios. Cada célula libre de obstáculos representa un nodo. Dos nodos están conectados entre ellos si las células son adyacentes [173]. Las descomposiciones en células son exactas o aproximadas:
  - Para las exactas: hay que obtener una representación exacta de todas las células completamente libres de obstáculos.
  - En las aproximadas: el conocimiento de un subespacio del espacio libre puede ser representado, por ejemplo, en la forma de un quadtree. Para este tipo de representación, se consideran tres tipos de células: células libres, células C-obstáculos y células mixtas. Las células libres son subespacios del espacio libre, las células C-obstáculos son subespacios de C-obstáculos y las células mixtas contienen un subespacio del espacio libre y un subespacio de los C-obstáculos.

Las técnicas globales construyen una representación del espacio de configuraciones a partir de un modelo completo del entorno. Son eficaces para un pequeño número de grados de libertad pero, en general, son costosos en tiempo de cálculo. Uno de los inconvenientes de la utilización del grafo de visibilidad o bien de los diagramas de Voronoi es que se aplican solamente a dos dimensiones.

### Locales

Este planteamiento utiliza una representación local del entorno para intentar generar una trayectoria hacia la configuración final. Esta basado en la optimización de una función orientada al guiado del manipulador hacia la configuración final. En general, se utiliza cuando la configuración inicial y final son vecinas. Las aproximaciones locales son también utilizadas en combinación con los métodos globales para alcanzar las configuraciones intermedias con el fin de mejorar la búsqueda de una trayectoria completa. Las aproximaciones locales principales son:

- Método de los potenciales: consideran el componente como una partícula bajo la influencia de fuerzas de atracción y repulsión. Estas fuerzas son generadas respectivamente por la configuración final y los obstáculos. Estas fuerzas pueden ser representadas por un campo de curvas equipotenciales [217]. La partícula se encuentra sometida a una energía potencial. Si se considera que la partícula está en un punto más alto que la posición final, la energía potencial obliga entonces a la partícula a descender por un camino libre de colisiones debido a la fuerza de repulsión de los obstáculos.
- Método de las restricciones: representa localmente el espacio de configuraciones por una lista de restricciones [52]. El problema de búsqueda de trayectorias es entonces transformado en un problema de optimización bajo restricciones o de minimización de una función objetivo. Esta función es definida a partir de la especificación de la configuración a alcanzar. La minimización bajo restricciones permite entonces determinar localmente la dirección de movimiento del

componente con el fin de alcanzar la configuración final sin colisión con los obstáculos.

Las técnicas locales son rápidas y eficaces. Sin embargo, es difícil definir una función potencial o restricciones que determine que un mínimo local representa la configuración final.

### **Mixtas**

Las técnicas mixtas utilizan principios locales y globales. Tienen por objetivo: evitar los mínimos locales, permitir un número importante de grados de libertad, utilizar entornos complejos; así como, disminuir el tiempo de cálculo.

El método estocástico permite una descomposición de la búsqueda de la trayectoria [9]. Esta técnica propone la colocación de marcas intermedias definidas aleatoriamente. El módulo encargado de calcular una trayectoria sin colisiones para el transporte de objetos manipulados debe elegir un conjunto de posiciones intermedias que garanticen esquivar los obstáculos durante el desplazamiento. La complejidad de este problema está unida directamente al número de grados de libertad del componente.

Otro planteamiento propone muestrear aleatoriamente el espacio de configuraciones del robot [126]. En este caso, las configuraciones libres de colisiones, son alcanzadas con trayectorias simples. El resultado es un grafo denominado: *Probabilistic roadmap for Path Planning*. Las configuraciones inicial y final del robot son alcanzadas por las configuraciones libres de obstáculos en este mapa.

Otra solución consiste en la utilización de facultades humanas. En efecto, el hombre tiene la facultad de visualizar y de analizar rápidamente un entorno siguiendo la representación que se le propone. Entonces, el hombre gracias a su visión global ve que camino debe seleccionar. Un estudio presentado en [112] está basado en esta aproximación y utiliza muestras intermedias elegidas por el hombre a lo largo de un recorrido. La búsqueda de trayectorias es entonces realizada por sistemas autónomos que intentan alcanzar las configuraciones de paso sucesivas. La utilización de la visión global del hombre conduce a una división coherente del problema inicial de búsqueda de trayectorias.

Las técnicas mixtas utilizan principios locales y globales. Los métodos propuestos anteriormente son básicamente secuenciales. La búsqueda global está simplificada por la inserción de muestras intermedias. De esta manera, estos métodos confirman que las intervenciones secuenciales del hombre o una aproximación estocástica mejoran los tiempos de búsqueda.

#### *3.4.1.2. Simulación interactiva*

Actualmente, existen herramientas de robótica que permiten la simulación de robots. Estas herramientas permiten el desplazamiento interactivo de un objeto, es decir, un desplazamiento de manera secuencial siguiendo los seis grados de libertad del objeto manipulado. Este desplazamiento, generalmente largo y delicado, se realiza por el operador gracias a su facultad de visión global. De esta manera, una funcionalidad de detección de colisiones permite al operador definir una trayectoria que evite colisiones.

La manipulación interactiva es igualmente accesible en realidad virtual [269]. Con este propósito se utilizan periféricos de comando especiales. El operador manipula entonces el componente en un entorno virtual. De todas maneras, a pesar de la funcionalidad de detección de colisiones que permite al operador definir una trayectoria libre de colisiones, es necesario señalar la ausencia de evitación automática de colisiones con el entorno, ausencia de periféricos que hagan "retroceder el esfuerzo".

#### 3.4.2. Solución utilizada

Para calcular cual es la trayectoria de desensamblado que debe realizar un determinado componente para separarse del resto, sigue el siguiente algoritmo, el cual se corresponde con una aproximación local que utiliza el método de los potenciales para su cálculo:

1. Determinar la dirección de desensamblado según las caras de contacto, tal y como se explicó en el apartado 3.3.
2. Determinar el punto de colisión más cercano al componente según la dirección elegida. Para ello se hace uso de un detector de colisiones que se explica en el siguiente apartado.
3. Una vez encontrado el punto de colisión más cercano y determinada la distancia a la que se encuentra dicho punto, se realiza el desplazamiento del componente según dicha dirección. De esta manera desaparecen algunas de las caras de contacto del componente actual y aparecen otras, información que es necesario actualizar en el grafo que representa el producto.
4. Una vez se ha desplazado el componente, se vuelve al paso 1 del algoritmo teniendo en cuenta la nueva configuración del producto y el entorno de trabajo discretizado con la información actual.

De esta manera se obtiene la secuencia de desplazamientos que conforman la trayectoria de desensamblado.

En la Figura 3.25 se muestra la secuencia que se sigue para obtener la trayectoria de desensamblado de un componente del producto de la Figura 3.23. La primera dirección de desensamblado mostrada en la Figura 3.25a, se corresponde con la obtenida por el algoritmo inicialmente, que permite desplazar el componente una cierta distancia (Figura 3.25b y Figura 3.25c) según la dirección de desensamblado obtenida. En ese momento el modelo se actualiza con las nuevas posibles colisiones y determina una nueva dirección de desplazamiento (Figura 3.25d); en la Figura 3.25e se muestra como se desplaza el componente hasta llegar a la posición representada en la Figura 3.25f, donde se calcula de nuevo la dirección de desensamblado, que en este caso, ya permite realizar el desensamblado del componente, con lo que se tendría calculada la trayectoria de desensamblado para el componente.



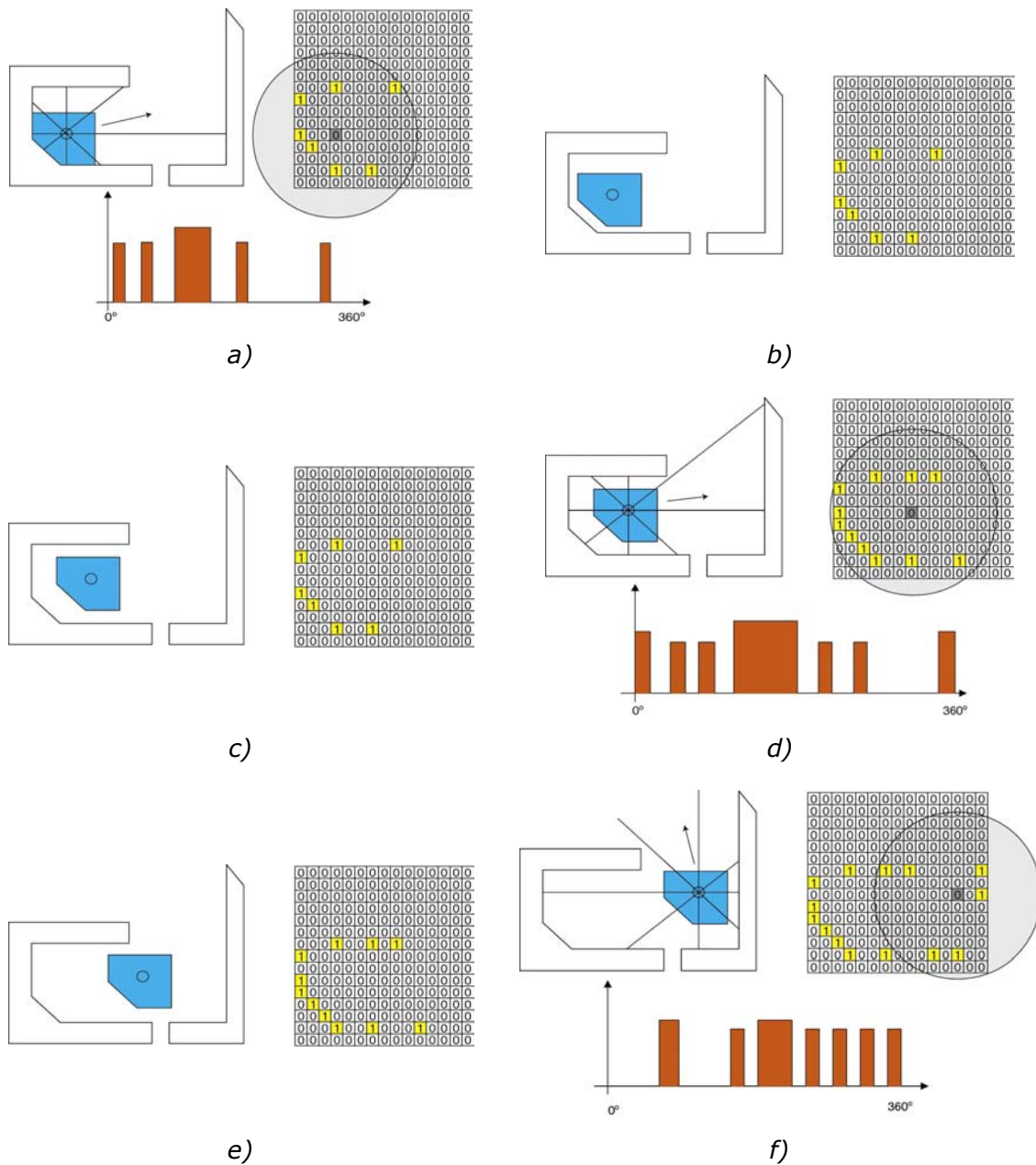


Figura 3.25: Secuencia del algoritmo para determinar la trayectoria de desensamblado de un componente en un producto formado por tres componentes .

Del algoritmo explicado para calcular la trayectoria de desensamblado, cabe destacar un dato importante sobre el que no se ha especificado un valor; este dato es la distancia que se desplaza el componente. Una primera aproximación podría pensar en desplazarlo lo máximo posible hasta que colisionara, en cuyo caso se podrían plantear bucles como el de la Figura 3.26.

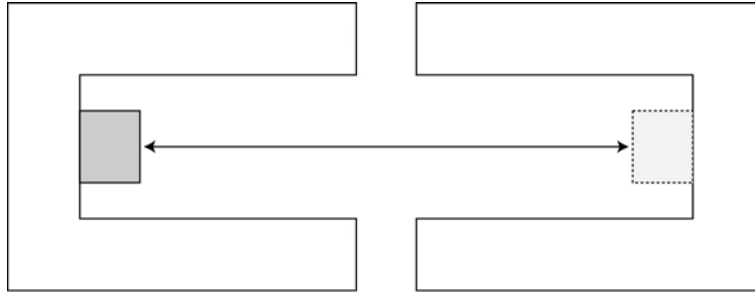


Figura 3.26: Bucle al desplazar totalmente un componente.

Una manera de evitar este problema, consiste en prohibir que un componente se desplace en la misma dirección, pero en sentido opuesto, al que se ha desplazado en el movimiento inmediatamente anterior.

Otro método sería desplazarlo sólo una celda del entorno discretizado. Este método plantea un costo computacional elevado, dado que se pueden generar una gran cantidad de desplazamientos en la misma dirección; pero obtiene la trayectoria de desensamblado que implica un menor desplazamiento total del componente.

### 3.5. Detección de colisiones

Otro aspecto de interés en la determinación de movimientos de desensamblado, es la detección de colisiones y la representación eficiente de la posición y orientación de los componentes. Una parte esencial, a la hora de determinar movimientos de desensamblado, es un sistema de razonamiento geométrico que pueda detectar colisiones entre los componentes en el espacio de trabajo, así como planificar caminos libres de colisiones a lo largo de este espacio de trabajo. Las restricciones en el movimiento del componente son causadas por los obstáculos correspondientes a superficies en el espacio de configuración del componente.

Bibliografía relacionada con la computación gráfica y la realidad virtual a menudo hace referencia a métodos para la determinación de colisiones entre objetos en movimiento [97].

Un objeto sólido tridimensional tiene tres grados de libertad de traslación y otros tres de rotación. Su orientación es típicamente especificada usando ángulos de Euler, pero esto provoca que las restricciones en el espacio de configuración contenga funciones trascendentales [174].

Hay algoritmos de planificación 3-D que utilizan distintos conceptos para representar la orientación: una representación angular [46], una representación usando los coeficientes de una matriz de rotación de tamaño  $3 \times 3$ , dando las restricciones algebraicas [235]. Esta última aproximación implica un espacio de configuración de 12 dimensiones, por lo que se tienen que introducir 6 restricciones adicionales para asegurar que todas las configuraciones corresponden a los coeficientes válidos de una matriz de rotación.

La orientación de un componente puede ser representada usando un cuaternio con cuatro componentes. Esto aumenta el espacio de configuración a 7 dimensiones, con una restricción adicional para asegurar que el cuaternio

tiene magnitud unidad. Debido a que la rotación de un vector por un cuaternio implica sólo multiplicación y suma, las restricciones serán algebraicas. De hecho es posible eliminar las variables extra usando las propiedades de escalado de los cuaternios.

Existen métodos para detectar si un componente poliédrico en movimiento colisiona con obstáculos también poliédricos [23]. En esta solución, los componentes se mueven con un movimiento de traslación uniforme así como con una rotación también uniforme a lo largo del espacio de configuración, que corresponde aproximadamente con una rotación uniforme en el espacio real. Expresando cada coordenada del objeto como una función polinomial de varios parámetros que varían a lo largo del camino, es posible detectar colisiones a lo largo de los caminos más generales. Así mismo si la rotación se representa mediante cuaternios las restricciones son puramente algebraicas en un espacio de mayor dimensión. Por simple manipulación, las restricciones pueden ser proyectadas en un espacio de seis dimensiones sin incrementar la complejidad. Usando esta formulación, se deriva un test eficiente y exacto para la intersección de un componente que se está trasladando y rotando entre medio de obstáculos.

### 3.5.1. Planteamiento utilizado

Como se ha comentado, se necesita calcular las posibles colisiones de un determinado componente con el resto del producto; para ser más exactos, se necesita conocer el punto de colisión de cada cara del componente que se está desensamblando (componente móvil), con cada una de las caras pertenecientes al resto de componentes del producto (componente estático).

Para ello se considera que el componente a desensamblar está en movimiento y el resto está estático, lo cual se corresponde con la notación presentado por Lozano-Pérez [174] para determinar las interacciones entre un componente poliédrico y obstáculos poliédricos, donde al componente en movimiento se le denomina A y al componente estático, el obstáculo, como B.

Teniendo en cuenta esta notación, existen tres tipos diferentes de colisiones que dan lugar a ciertas restricciones [23]:

- Tipo-A: Se producen cuando un vértice de A colisiona con una cara de B (Figura 3.27).

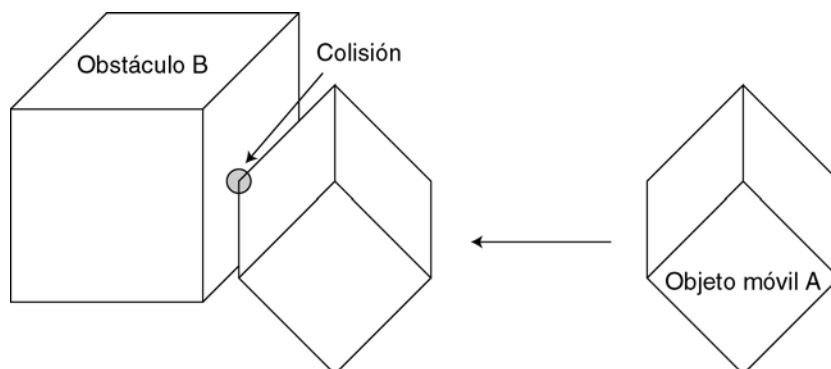


Figura 3.27: Colisión de tipo-A entre dos objetos.

- Tipo-B: Se producen cuando una cara de A colisiona con un vértice B (Figura 3.28).

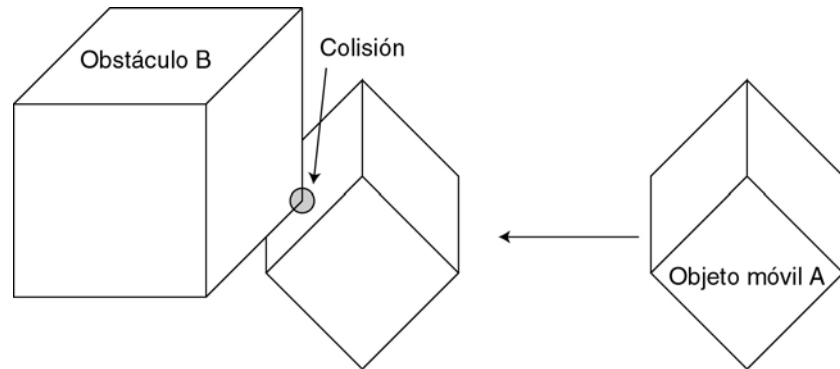


Figura 3.28: Colisión de tipo-B entre dos objetos.

- Tipo-C: Se producen cuando una arista de A colisiona con un arista B (Figura 3.29).

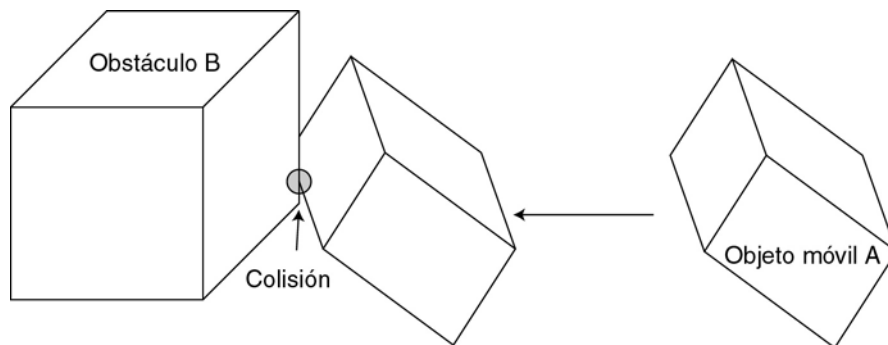


Figura 3.29: Colisión de tipo-C entre dos objetos.

De manera general, la localización de un objeto viene dada por un par  $(\mathbf{x}, \mathbf{Q})$ , donde  $\mathbf{x}$  es un vector de posición, que indica el desplazamiento que ha sufrido el componente respecto del origen de coordenadas, y  $\mathbf{Q}$  es un cuaternino, que representa la rotación del componente respecto al origen. De tal manera que, la localización del componente se obtiene realizando su rotación respecto del origen, según el cuaternino  $\mathbf{Q}$  y posteriormente trasladándolo según  $\mathbf{x}$ .

Para cada uno de los contactos indicados existen una serie de restricciones entre ambos componentes, de tal manera, que se puede definir una región del espacio, en la que existe una interferencia entre el objeto en movimiento y los obstáculos.

Si el componente A se encuentra en el origen, entonces un punto  $\mathbf{y}$  en una cara de A debe satisfacer:

$$\mathbf{y} \cdot \mathbf{n}_A - d_A = 0 \quad (3.7)$$

Donde:

- $\mathbf{n}_A$  es el vector unitario normal a la superficie de la cara dirigido hacia el exterior del componente.
- $d_A$  es la distancia desde el origen a la cara en la dirección normal.

Además la rotación del componente por un cuaternino  $\mathbf{Q}$  viene dada por:

$$\mathbf{y} \cdot \mathbf{Rot}(\mathbf{n}_A, \mathbf{Q}) - d_A = 0 \quad (3.8)$$

Donde:

- $\mathbf{Rot}(\mathbf{n}_A, \mathbf{Q})$  es la rotación del vector  $\mathbf{n}_A$  por el cuaternino  $\mathbf{Q}$ .

Si se traslada el componente A según el vector  $\mathbf{x}$ , es el mismo efecto que una traslación de  $\mathbf{y}$  por  $-\mathbf{x}$ , ya que un punto perteneciente a la cara de A rotada y desplazada debe satisfacer:

$$(\mathbf{y} - \mathbf{x}) \cdot \mathbf{Rot}(\mathbf{n}_A, \mathbf{Q}) - d_A = 0 \quad (3.9)$$

De esta manera para calcular las colisiones de tipo-A se parte de la ecuación de una cara de B:

$$\mathbf{y} \cdot \mathbf{n}_B - d_B = 0 \quad (3.10)$$

Donde:

- $\mathbf{n}_B$  es el vector unitario normal a la superficie de la cara dirigido hacia el exterior del componente.
- $d_B$  es la distancia desde el origen a la cara en la dirección normal.

En la ecuación 3.10 se sustituye  $\mathbf{y}$  por un vértice de A denominado  $\mathbf{p}_A$ , que se encuentra rotado  $\mathbf{Q}$  y desplazado  $\mathbf{x}$ , de tal manera que la restricción queda:

$$(\mathbf{Rot}(\mathbf{p}_A, \mathbf{Q}) + \mathbf{x}) \cdot \mathbf{n}_B - d_B = 0 \quad (3.11)$$

Por otra parte para las colisiones de tipo-B, donde un vértice del obstáculo  $\mathbf{p}_B$  colisiona con una cara de A, la restricción corresponde a sustituir en la ecuación 3.9 el valor de  $\mathbf{y}$  por  $\mathbf{p}_B$  es:

$$(\mathbf{p}_B - \mathbf{x}) \cdot \mathbf{Rot}(\mathbf{n}_A, \mathbf{Q}) - d_A = 0 \quad (3.12)$$

Para calcular las colisiones de tipo-C se aplica el hecho de que dos rectas no colineales, que se intersectan en el espacio tridimensional, forman un plano. De tal manera que, definiendo las aristas como rectas, se puede calcular el plano perpendicular en el que se intersectan. En base a lo que se puede decir:

$$(\mathbf{y} - \mathbf{p}_B) \cdot (\mathbf{e}_A \times \mathbf{e}_B) = 0 \quad (3.13)$$

Donde:

- $\mathbf{e}_A$  es el vector que define la dirección de la arista a considerar del componente A.
- $\mathbf{e}_B$  es el vector que define la dirección de la arista a considerar del obstáculo B.
- $\mathbf{p}_B$  es un punto del obstáculo.

Condición que cumple el plano que pasa por la arista de B y es paralelo a la arista de A. Para comprobar la intersección entre ambas aristas se sustituye  $\mathbf{y}$  por las coordenadas de un punto en la arista A, el punto  $\mathbf{p}_A$ . Dado que el objeto A ha sido rotado según  $\mathbf{Q}$  y trasladado según  $\mathbf{x}$  se tiene:

$$y = \mathbf{Rot}(p_A, Q) + x \quad (3.14)$$

Y además se puede ver que la dirección de la arista de A viene dada por:

$$\mathbf{Rot}(e_A, Q) \quad (3.15)$$

Con lo cual, sustituyendo en la ecuación 3.13 las ecuaciones 3.14 y 3.15 se obtiene que la restricción de tipo-C es la siguiente:

$$(\mathbf{Rot}(p_A, Q) + x - p_B) \cdot (\mathbf{Rot}(e_A, Q) \times e_B) = 0 \quad (3.16)$$

Estas expresiones permiten determinar si un punto está obtenido realmente por una colisión y además saber que tipo de colisión lo ha provocado.

### 3.5.1.1. Restricciones debidas al movimiento del componente

Cuando un componente se encuentra en movimiento en una determinada dirección, se generan una serie de restricciones que dependen de la propia geometría del objeto, así como del movimiento realizado.

En este apartado, así como en el resto de la tesis, se va a considerar que los componentes no pueden rotar.

Estas restricciones se aplican sólo a una parte del espacio, la zona que se puede ver afectada por el desplazamiento del componente; para ser más exactos, se realiza un estudio de las colisiones de cada una de las caras del componente en la dirección del movimiento, de manera que, el espacio de búsqueda, se reduce a los lugares donde dicha cara puede colisionar con otra (Figura 3.30).

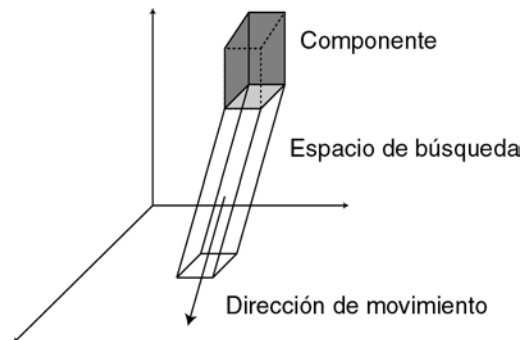


Figura 3.30: Espacio de búsqueda de colisiones, teniendo en cuenta las restricciones, por la dirección de desensamblado para la cara inferior de un componente.

De tal manera que cada arista de la cara que se está analizando, genera un plano paralelo a la dirección del movimiento, cuyo vector director apunta hacia el exterior del volumen encerrado por el movimiento de la cara.

Para calcular cada uno de los planos que definen las restricciones, se parte de los dos vértices que forman una arista ( $\mathbf{v}_1$  y  $\mathbf{v}_2$ ) de la cara, de la dirección de desensamblado que se está considerando  $\mathbf{d}$  y de otro vértice de la cara  $\mathbf{v}_3$ ; este último vértice servirá para determinar el sentido del vector director del plano. Teniendo en cuenta esta información, se determina la ecuación del plano que pasa por dos puntos y contiene la dirección dada por un vector como:

$$\det \begin{vmatrix} 1 & 1 & 0 & 1 \\ x_{v_1} & x_{v_2} & x_d & x \\ y_{v_1} & y_{v_2} & y_d & y \\ z_{v_1} & z_{v_2} & z_d & z \end{vmatrix} = 0 \quad (3.17)$$

En donde expresado el plano como:

$$Ax + By + Cz + D = 0 \quad (3.18)$$

Se obtendría el siguiente valor para los parámetros A, B, C y D, teniendo en cuenta que se pretende que el vector director del plano sea unitario:

$$A' = y_{v_2}z_d - z_{v_2}y_d + z_{v_1}y_d - y_{v_1}z_d \quad (3.19)$$

$$B' = -x_{v_2}z_d + x_{v_2}x_d - z_{v_1}x_d + x_{v_1}z_d \quad (3.20)$$

$$C' = x_{v_2}y_d - y_{v_2}x_d + y_{v_1}x_d - x_{v_1}y_d \quad (3.21)$$

$$\text{Mod} = \sqrt{A'^2 + B'^2 + C'^2} \quad (3.22)$$

$$A = \frac{A'}{\text{Mod}}; \quad B = \frac{B'}{\text{Mod}}; \quad C = \frac{C'}{\text{Mod}} \quad (3.23)$$

$$D = \frac{-x_{v_1}(y_{v_2}z_d - z_{v_2}y_d) + x_{v_2}(y_{v_1}z_d - z_{v_1}y_d) - x_d(y_{v_1}z_{v_2} - z_{v_1}y_{v_2})}{\text{Mod}} \quad (3.24)$$

Teniendo en cuenta que todo punto  $(x, y, z)$  que se encuentre en la zona de búsqueda delimitada por el plano debe cumplir:

$$Ax + By + Cz + D \leq 0 \quad (3.25)$$

El tercer punto del polígono  $v_3$  debe cumplir la condición 3.25 con lo que:

$$Ax_{v_3} + By_{v_3} + Cz_{v_3} + D \leq 0 \quad (3.26)$$

Luego si no se cumple, el resultado obtenido en las ecuaciones 3.23 y 3.24 se debe multiplicar por  $-1$  para obtener el plano que posea el sentido, de su vector director, adecuado para el resto de cálculos.

### 3.5.1.2. Detección de colisiones mediante un problema de programación lineal

Una vez definidos los distintos tipos de restricciones que aparecen a la hora de realizar el movimiento de una cara según una determinada dirección, se van a calcular los puntos de colisión con las caras de otros componentes del producto, mediante un problema de programación lineal [21].

Este método resulta muy útil para representar las restricciones, debido a que todas ellas se pueden expresar de manera sencilla mediante ecuaciones lineales, que van limitando el espacio de búsqueda de la solución.

## Programación lineal

El método de programación lineal busca maximizar la función  $z$  para  $n$  variables independientes  $\mathbf{x}=(x_1, x_2, \dots, x_n)$ .

$$z = a_{01}x_1 + a_{02}x_2 + \dots + a_{0n}x_n \quad (3.27)$$

Donde se tiene como restricciones principales:

$$x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0 \quad (3.28)$$

Y además la función 3.27 se encuentra sometida a  $M = m_1 + m_2 + m_3$  restricciones extras. De las que  $m_1$  son de la forma:

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \leq b_i \quad (b_i \geq 0) \quad i = 1, \dots, m_1 \quad (3.29)$$

Y  $m_2$  restricciones son de la forma:

$$a_{j1}x_1 + a_{j2}x_2 + \dots + a_{jn}x_n \geq b_j \quad (b_j \geq 0) \quad j = m_1 + 1, \dots, m_1 + m_2 \quad (3.30)$$

Y por último  $m_3$  restricciones son de la forma:

$$a_{k1}x_1 + a_{k2}x_2 + \dots + a_{kn}x_n = b_k \geq 0 \quad k = m_1 + m_2 + 1, \dots, m_1 + m_2 + m_3 \quad (3.31)$$

Donde los coeficientes  $a_{ij}$  pueden ser positivos, negativos o cero.

Existen dos clases de métodos para la solución de problemas de programación lineal: el método denominado en la bibliografía como *simplex*, y los métodos interiores. El método *simplex* [2] considera el problema como un espacio  $n$ -dimensional donde cada vector  $\mathbf{x}$  es un punto en el espacio, y cada restricción define un hiperplano que divide el espacio en dos. De esta manera las restricciones definen una región cerrada del espacio que contiene las soluciones del problema [255]. El algoritmo comienza con un vértice en una superficie  $n$ -dimensional y se mueve en la dirección que consigue una mejora de la solución hasta encontrar la óptima.

Los métodos interiores comienzan en un punto dentro del conjunto de soluciones válidas y van aproximando el punto hacia el óptimo [2].

Como se puede ver para el caso que nos ocupa, interesa el método *simplex* en lugar de un método interior, ya que no se conoce a priori ningún punto del conjunto de soluciones.

Existen dos casos en los que no se puede encontrar una solución:

- Cuando no hay solución posible, las restricciones son incompatibles.
- No existe un máximo. Este caso se da cuando una o más variables pueden tomar un valor infinito y seguir cumpliendo las restricciones, de tal manera que la función objetivo no estaría acotada.

## Generación del problema de programación lineal

Para describir el problema de programación lineal e identificar claramente cada una de las restricciones existentes se va a realizar el diseño esquemático de las restricciones utilizando un par de componentes, tal y como se muestran en la Figura 3.31.



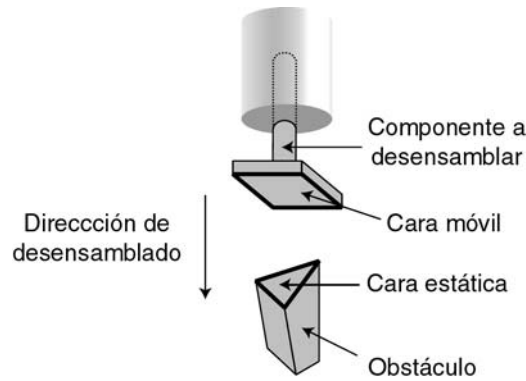


Figura 3.31: Componentes sobre los que se aplica el problema de programación lineal.

Si se considera la cara del componente móvil implicada en la detección de la colisión como:

$$f(\mathbf{x}) = Ax + By + Cz + D \quad (3.32)$$

Se tendría un problema ya que se tiene un término independiente,  $D$ , que en la ecuación objetivo no existe (Ecuación 3.27). Para solventar este problema se ha optado por aumentar la dimensión del espacio de tal manera que se introduce una nueva variable  $v$  cuyo valor es 1 con lo que la ecuación objetivo queda:

$$f(\mathbf{x}) = Ax + By + Cz + Dv \quad (3.33)$$

Y por tanto hay que introducir una primera restricción, para que el sistema no varíe con la introducción de la nueva variable, que es:

$$v = 1 \quad (3.34)$$

Teniendo esto en cuenta, la función a maximizar es  $-f(\mathbf{x})$  que vale:

$$\text{Maximizar} \equiv -Ax - By - Cz - Dv \quad (3.35)$$

Donde además de la restricción comentada, se deben aplicar otras restricciones que dependen de los componentes implicados y la dirección del movimiento. En estas restricciones se ha hecho directamente la extrapolación de ellas, para que incluyan la variable  $v$ :

- Restricciones por los planos paralelos a la dirección de movimiento que pasan por cada par de vértices de las aristas de la cara del componente móvil (Figura 3.32). Si se considera que la ecuación del plano obtenido es de la forma:

$$f_m(\mathbf{x}) = A_m x + B_m y + C_m z + D_m v \quad (3.36)$$

Entonces la restricción tiene la forma:

$$A_m x + B_m y + C_m z + D_m v \leq 0 \quad (3.37)$$

Esta restricción limita el espacio de búsqueda de soluciones basándose en la geometría de la cara móvil y en la dirección en la que se está realizando el movimiento.

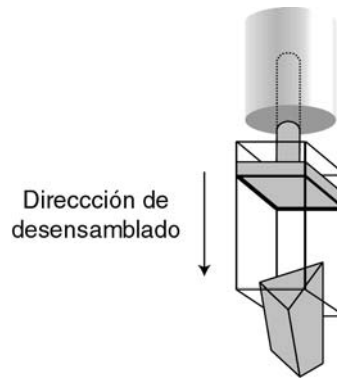


Figura 3.32: Restricciones por los planos paralelos a la dirección de movimiento de la cara del componente móvil.

- Restricciones por los planos paralelos a la dirección de movimiento que pasan por cada par de vértices de las aristas de la cara del componente estático que se está considerando (Figura 3.33).

Si se considera que la ecuación del plano obtenido es de la forma:

$$f_e(\mathbf{x}) = A_e x + B_e y + C_e z + D_e v \quad (3.38)$$

La restricción es de la forma:

$$A_e x + B_e y + C_e z + D_e v \leq 0 \quad (3.39)$$

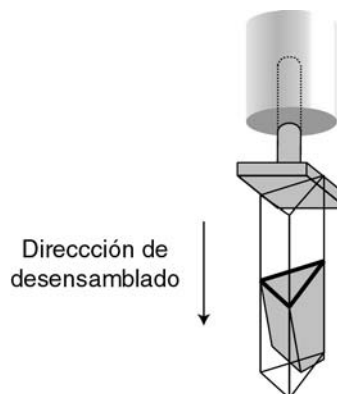


Figura 3.33: Restricciones por los planos paralelos a la dirección de movimiento de la cara del componente estático.

Esta restricción sirve para limitar el volumen del espacio por el que tiene que desplazarse la cara del componente móvil para colisionar con la cara estática.

- Restricción por la cara que se está considerando del componente móvil. (Figura 3.34). Si se considera la ecuación del plano de la cara como:

$$f_{cm}(\mathbf{x}) = A_{cm} x + B_{cm} y + C_{cm} z + D_{cm} v \quad (3.40)$$

La restricción es de la forma:

$$A_{cm} x + B_{cm} y + C_{cm} z + D_{cm} v \geq 0 \quad (3.41)$$

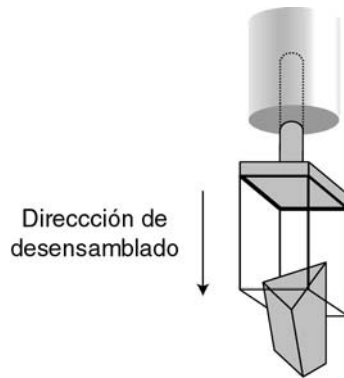


Figura 3.34: Restricción por la cara del componente móvil.

Esta restricción representa el espacio que se encuentra delante de la cara móvil.

- Restricción por la cara que se está considerando del componente estático. (Figura 3.35). Si se considera que la ecuación del plano de la cara como:

$$f_{ce}(x) = A_{ce}x + B_{ce}y + C_{ce}z + D_{ce}v \quad (3.42)$$

Con lo que la restricción tiene la forma:

$$A_{ce}x + B_{ce}y + C_{ce}z + D_{ce}v = 0 \quad (3.43)$$

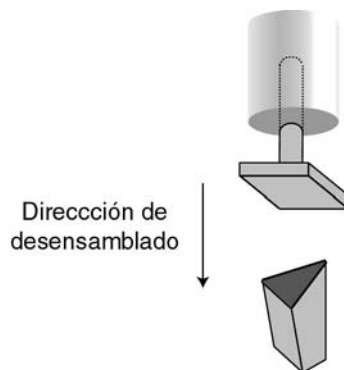


Figura 3.35: Restricción por la cara del componente estático.

Esta restricción implica que la colisión tiene que producirse en el plano que genera la cara estática, ya que se colisiona con él.

De esta manera se tiene definida la zona del espacio donde se puede producir la colisión entre ambas caras (Figura 3.36).

Con lo que se plantea el problema de programación lineal correspondiente, el cual obtiene el punto de colisión entre la cara móvil y la cara del componente estático (Figura 3.36).

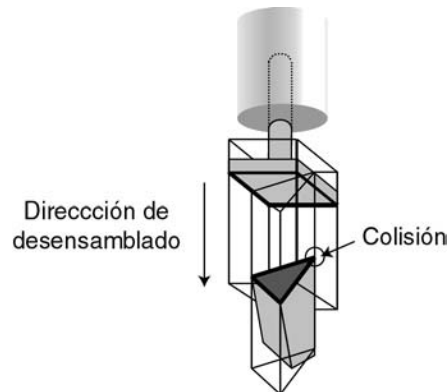


Figura 3.36: Zona del espacio donde se puede producir colisión y la colisión.

### 3.6. Cálculo de la trayectoria de desensamblado de una fuente de alimentación

En este apartado se ilustran los conceptos vistos durante el presente capítulo, aplicándolos al cálculo de la trayectoria de desensamblado de la fuente de alimentación de un ordenador, considerando solamente la existencia de dos componentes: la fuente de alimentación y la carcasa interior del mismo.

En la Figura 3.37 se muestran ambos componentes así como la relación de posición existente entre ellos.

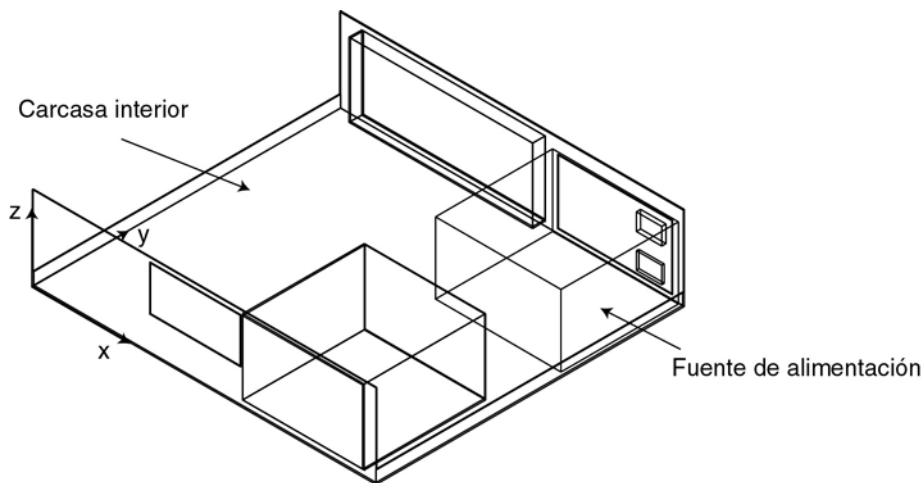


Figura 3.37: Modelo geométrico de la carcasa interior y la fuente de alimentación.

En base a la información geométrica de cada uno de los componentes, se genera el grafo de adyacencia correspondiente.

Para la fuente de alimentación, se tiene un grafo de adyacencia formado por 16 caras rectangulares, denominadas como  $p_1, p_2, p_3, \dots, p_{16}$  (Figura 3.38).

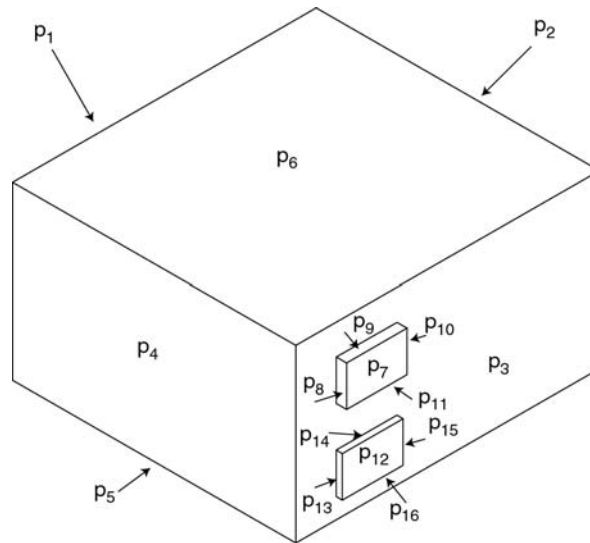


Figura 3.38: Identificación de las caras en la fuente de alimentación.

Como puede observarse cada una de las caras está formada por cuatro puntos, cuyas coordenadas así como el plano que forman, se detallan a continuación. Para ello se supone que el origen de coordenadas se encuentra en la esquina inferior izquierda del ordenador, tal y como muestra la Figura 3.37.

Cara	Punto 1	Punto 2	Punto 3	Punto 4	Plano
P <sub>1</sub>	(254, 225, 85)	(404, 225, 85)	(404, 225, 0)	(254, 225, 0)	$-y + 225 = 0$
P <sub>2</sub>	(254, 225, 85)	(254, 225, 0)	(254, 366, 0)	(254, 366, 85)	$-x + 254 = 0$
P <sub>3</sub>	(254, 366, 0)	(254, 366, 85)	(404, 366, 85)	(404, 366, 0)	$y - 366 = 0$
P <sub>4</sub>	(404, 366, 85)	(404, 366, 0)	(404, 225, 0)	(404, 225, 85)	$x - 404 = 0$
P <sub>5</sub>	(404, 225, 0)	(254, 225, 0)	(254, 366, 0)	(404, 366, 0)	$-z = 0$
P <sub>6</sub>	(254, 225, 85)	(404, 225, 85)	(404, 366, 85)	(254, 366, 85)	$z - 85 = 0$
P <sub>7</sub>	(354, 370, 50)	(384, 370, 50)	(384, 370, 70)	(354, 370, 70)	$-y - 370 = 0$
P <sub>8</sub>	(384, 366, 50)	(384, 370, 50)	(384, 370, 70)	(384, 366, 70)	$x - 384 = 0$
P <sub>9</sub>	(384, 370, 70)	(384, 366, 70)	(354, 366, 70)	(354, 370, 70)	$z - 70 = 0$
P <sub>10</sub>	(354, 366, 70)	(354, 370, 70)	(354, 370, 50)	(354, 366, 50)	$-x + 354 = 0$
P <sub>11</sub>	(354, 370, 50)	(354, 365, 50)	(384, 366, 50)	(384, 370, 50)	$-z + 50 = 0$
P <sub>12</sub>	(354, 368, 10)	(384, 368, 10)	(384, 368, 30)	(354, 368, 30)	$-y - 368 = 0$
P <sub>13</sub>	(384, 368, 10)	(384, 368, 30)	(384, 366, 30)	(384, 366, 10)	$x - 384 = 0$
P <sub>14</sub>	(384, 368, 30)	(384, 366, 30)	(354, 366, 30)	(354, 368, 30)	$z - 30 = 0$
P <sub>15</sub>	(354, 366, 30)	(354, 368, 30)	(354, 368, 10)	(354, 368, 10)	$-x + 354 = 0$
P <sub>16</sub>	(354, 368, 10)	(354, 366, 10)	(384, 366, 10)	(384, 368, 10)	$-z + 10 = 0$

Por otro lado, para la carcasa interior del ordenador, se genera un grafo de adyacencias que indique las caras que posee, así como la información relevante a las mismas. Para este componente se han especificado sólo 17 caras, que son las que se consideran más relevantes para el resto de la resolución, ya que la definición de la totalidad de la carcasa poseería muchas más y su desglose es innecesario para los cálculos subsiguientes. En la Figura 3.39 se muestran las caras más relevantes identificadas sobre el modelo geométrico del componente.

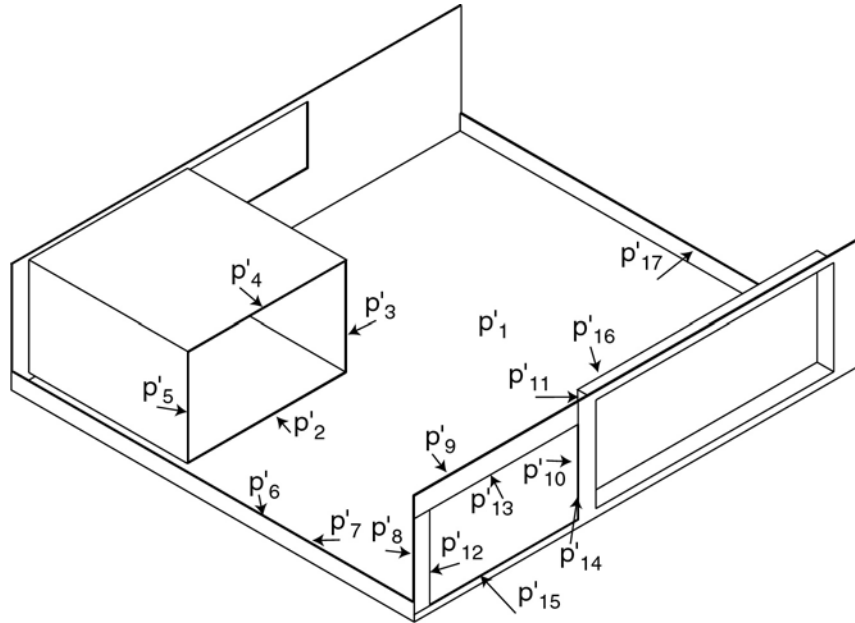


Figura 3.39: Identificación de las caras en la carcasa interior.

En la siguiente tabla se muestra la información relativa a las coordenadas de cada una de las caras de la carcasa interior, así como la información sobre los planos que forman.

Cara	Punto 1	Punto 2	Punto 3	Punto 4	Plano
P' <sub>1</sub>	(0, 0, 0)	(0, 366, 0)	(408, 366, 0)	(408, 0, 0)	$z = 0$
P' <sub>2</sub>	(394, 145, 6)	(250, 145, 6)	(250, 145, 7)	(394, 145, 7)	$y - 145 = 0$
P' <sub>3</sub>	(249, 145, 7)	(250, 145, 7)	(250, 145, 94)	(249, 145, 94)	$y - 145 = 0$
P' <sub>4</sub>	(250, 145, 94)	(250, 145, 95)	(394, 145, 95)	(394, 145, 94)	$y - 145 = 0$
P' <sub>5</sub>	(394, 145, 94)	(393, 145, 94)	(393, 145, 7)	(394, 145, 7)	$y - 145 = 0$
P' <sub>6</sub>	(409, 366, 14)	(408, 366, 14)	(408, 1, 14)	(409, 1, 14)	$z - 14 = 0$
P' <sub>7</sub>	(408, 366, 14)	(408, 1, 14)	(408, 1, 0)	(408, 366, 0)	$-x + 408 = 0$
P' <sub>8</sub>	(409, 366, 81)	(395, 366, 81)	(395, 366, 5)	(395, 366, 81)	$-y + 366 = 0$
P' <sub>9</sub>	(409, 366, 99)	(409, 366, 81)	(244, 366, 81)	(244, 366, 99)	$-y + 366 = 0$
P' <sub>10</sub>	(259, 366, 81)	(259, 366, 5)	(244, 366, 5)	(244, 366, 81)	$-y + 366 = 0$
P' <sub>11</sub>	(244, 351, 93)	(244, 351, 5)	(244, 366, 5)	(244, 366, 93)	$x - 244 = 0$
P' <sub>12</sub>	(395, 366, 81)	(395, 366, 5)	(395, 367, 5)	(395, 367, 81)	$-x + 395 = 0$
P' <sub>13</sub>	(395, 366, 81)	(395, 367, 81)	(259, 367, 81)	(259, 366, 81)	$-z + 81 = 0$
P' <sub>14</sub>	(259, 367, 81)	(259, 366, 81)	(259, 366, 5)	(259, 367, 5)	$x - 259 = 0$
P' <sub>15</sub>	(259, 366, 5)	(259, 367, 5)	(395, 367, 5)	(395, 366, 5)	$z - 5 = 0$
P' <sub>16</sub>	(244, 351, 93)	(244, 351, 5)	(25, 351, 5)	(25, 351, 93)	$-y + 351 = 0$
P' <sub>17</sub>	(0, 0, 0)	(0, 0, 14)	(0, 366, 14)	(0, 366, 0)	$x = 0$

Utilizando esta información, se calcula el grafo de caras de contacto, que genera las relaciones mostradas en la Figura 3.40. Que expresa los contactos existentes entre la cara correspondiente a la base de la fuente de alimentación ( $p_5$ ) con la correspondiente de la carcasa interior ( $p'_1$ ); y además las relaciones entre la parte posterior de la carcasa ( $p'_8$ ,  $p'_9$ ,  $p'_{10}$ ) con la parte posterior de la fuente de alimentación ( $p_3$ ).

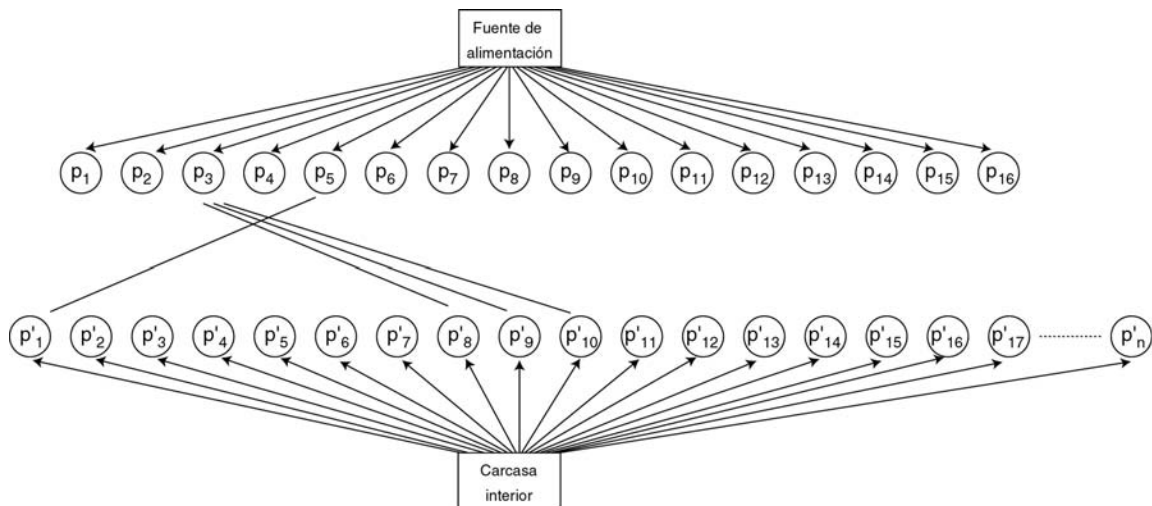


Figura 3.40: Grafo de relaciones.

Utilizando esta información se genera el conjunto de posibles direcciones de desensamblado para el componente "fuente de alimentación", mediante la intersección de las superficies esféricas correspondientes a cada cara de contacto. Cabe destacar al respecto que las caras  $p_8$ ,  $p_9$  y  $p_{10}$  de la carcasa interior pertenecen al mismo plano, luego tendrán la misma esfera de direcciones de desensamblado.

En la Figura 3.41 se muestra cada una de las esferas que representan las posibles direcciones de desensamblado del componente así como el resultado de su combinación.

De entre el conjunto de direcciones de desensamblado, ahora se tiene que calcular cuál es la que posee una menor probabilidad de colisión con el entorno; para ello se procede a la generación de un haz de rayos desde el centro del objeto, con lo que se obtiene información acerca de la existencia de otros componentes en las cercanías y a qué distancia se encuentra cada uno de ellos. Con esta información se construye un histograma que indica las direcciones con mayor probabilidad de colisión.

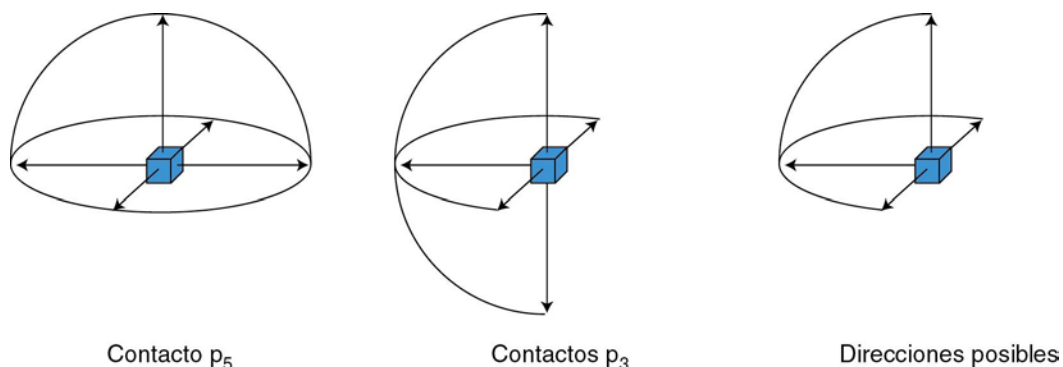


Figura 3.41: Direcciones de desensamblado para la fuente de alimentación.

En este caso se ha considerado un haz de rayos desde el centro de la fuente de alimentación con un ángulo de  $45^\circ$  entre cada rayo. De esta manera sólo existen fuerzas de atracción, en el cuarto de esfera considerado, porque no colisiona ningún rayo con otro componente. Por lo tanto la dirección de desensamblado viene definida por la suma de dichas direcciones:

$$d = (0, -0.7071, 0.7071) \quad (3.44)$$

Una vez se ha determinado la dirección de desplazamiento, hay que aplicar el método de programación lineal para calcular la distancia a la que se encuentra un obstáculo según dicha dirección, por lo tanto, se aplica el problema de programación lineal a todas aquellas caras de la fuente de alimentación cuyo vector director, del plano correspondiente, forme un ángulo comprendido entre  $-90^\circ$  y  $90^\circ$ , que serían las caras  $p_1$ ,  $p_6$ ,  $p_9$  y  $p_{14}$ . Por otra parte, los puntos de colisión con las caras del componente estático a considerar son aquellos que formen con la dirección de desensamblado un ángulo entre  $90^\circ$  y  $-90^\circ$ . Por lo tanto se utilizan las caras  $p'_2$ ,  $p'_3$ ,  $p'_4$ ,  $p'_5$ ,  $p'_{13}$ .

De esta manera hay que plantear un problema de programación lineal para cada posible cara del componente móvil con cada una de las posibles caras del componente estático. El problema de programación lineal, para la cara  $p_1$  y la cara  $p'_2$ , queda definido por las siguientes ecuaciones:

Función objetivo:

$$\text{Maximizar} \equiv y - 225v \quad (3.45)$$

Las restricciones provenientes de los planos paralelos a la dirección del movimiento, tanto de la cara del componente móvil como de la cara del componente estático son:

$$-0.707107y - 0.707107z + 159.0990v \leq 0 \quad (3.46)$$

$$x - 394v \leq 0 \quad (3.47)$$

$$0.707107y + 0.707107z - 107.4802v \leq 0 \quad (3.48)$$

$$-x + 254v \leq 0 \quad (3.49)$$

Hay que considerar que, en este caso, deberían ser 8 las ecuaciones correspondientes a los planos paralelos a la dirección del movimiento, 4 correspondientes a las caras del componente móvil y los otros 4 a las caras del componente estático. Estas ecuaciones se han reducido a 4 porque las otras 4 definen planos paralelos menos restrictivos, con lo que se puede simplificar el problema, utilizando, en caso de planos paralelos y con el mismo vector director, aquellos que son más restrictivos.

Por otro lado la restricción por la cara móvil es:

$$-y + 225v \geq 0 \quad (3.50)$$

Por último queda la restricción debida a la cara estática:

$$y - 145v = 0 \quad (3.51)$$

Y la correspondiente a  $v$ , que se muestra en la ecuación 3.34.

El algoritmo de programación lineal, teniendo en cuenta estas ecuaciones informa al sistema que no hay punto de colisión en la dirección del movimiento entre las caras implicadas, por lo tanto, dicha cara del componente estático no impide el desplazamiento en la dirección seleccionada. Esta característica se



puede observar fijándose en la Figura 3.37, donde se aprecia que las cara  $p_1$  y  $p'_2$  no pueden llegar a afectarse, teniendo en cuenta la actual dirección de desensamblado. Esto mismo sucede con la cara  $p_1$  y la  $p'_{13}$ .

A continuación hay que seguir resolviendo el problema de programación lineal para el resto de caras del componente estático respecto de la cara actual del componente móvil.

Si consideramos  $p_1$  respecto a la cara estática  $p'_4$  el problema de programación lineal planteado es el siguiente:

Función objetivo:

$$\text{Maximizar} \equiv y - 225v \quad (3.52)$$

Las restricciones correspondientes de los planos paralelos a la dirección del movimiento, tanto de la cara del componente móvil como de la cara del componente estático son:

$$-0.707107y - 0.707107z + 168.9985v \leq 0 \quad (3.53)$$

$$x - 394v \leq 0 \quad (3.54)$$

$$0.707107y + 0.707107z - 169.7056v \leq 0 \quad (3.55)$$

$$-x + 254v \leq 0 \quad (3.56)$$

La restricción por la cara móvil es:

$$-y + 225v \geq 0 \quad (3.57)$$

Por último queda la restricción debida a la cara estática:

$$y - 145v = 0 \quad (3.58)$$

En donde se obtiene como punto de corte más cercano entre ambas caras el punto:

$$(x, y, z) = (254, 145, 94) \quad (3.59)$$

Lo que implica una distancia máxima permitida en la dirección del movimiento de 113mm.

Por otra parte, si se calcula el problema de programación lineal de la cara móvil  $p_1$  con la cara estática  $p'_3$  se obtiene que no existe solución, es decir dichas caras no se cortan en el desplazamiento a lo largo de la dirección seleccionada, tal y como pasaba con la cara  $p'_2$  y  $p'_{13}$ .

Sin embargo para la cara estática  $p'_5$  sí que existe solución al problema de programación lineal, que viene definido por las siguiente restricciones:

Restricciones por los planos paralelos a la dirección del movimiento:

$$-0.707107y - 0.707107z + 159.0990v \leq 0 \quad (3.60)$$

$$x - 394v \leq 0 \quad (3.61)$$

$$0.707107y + 0.707107z - 168.9985v \leq 0 \quad (3.62)$$

$$-x + 393v \leq 0 \quad (3.63)$$

La restricción correspondiente a la cara móvil es la misma que en los casos anteriores, ya que se trata de la misma cara móvil. Por la misma razón la función a maximizar también es la misma.

La última restricción a considerar es la correspondiente a la cara estática:

$$y - 145v = 0 \quad (3.64)$$

En donde se obtiene, como punto de corte más cercano entre ambas caras, el punto:

$$(x, y, z) = (393, 145, 80) \quad (3.65)$$

que se encuentra a una distancia de 113mm según la dirección del movimiento. Esta distancia era de esperar si se observa la estructura del producto, ya que es el mismo valor obtenido para  $p'_4$ . Esto es debido a que ambas caras definen el mismo plano, pero se diferencian en las aristas que lo delimitan.

Considerando como cara móvil  $p_6$ , se define un nuevo problema de programación lineal con cada una de las caras correspondientes de la carcasa interior. En este caso se puede apreciar directamente en la Figura 3.37 que no existe colisión de dicha cara con ninguna de las del componente estático.

Por lo tanto hay que seguir calculando las colisiones correspondientes a  $p_9$  y  $p_{14}$ . Estas caras tampoco colisionan con ninguna otra en la dirección seleccionada; de esta manera ya se tiene resuelto el problema, que implica un desplazamiento máximo de 113mm.

El componente se desplaza una unidad en dicha dirección; en este caso se ha tomado el valor de una unidad como 50mm. Con lo que se desplaza en la dirección del movimiento dicha cantidad, quedando, el producto, tal y como muestra la Figura 3.42.

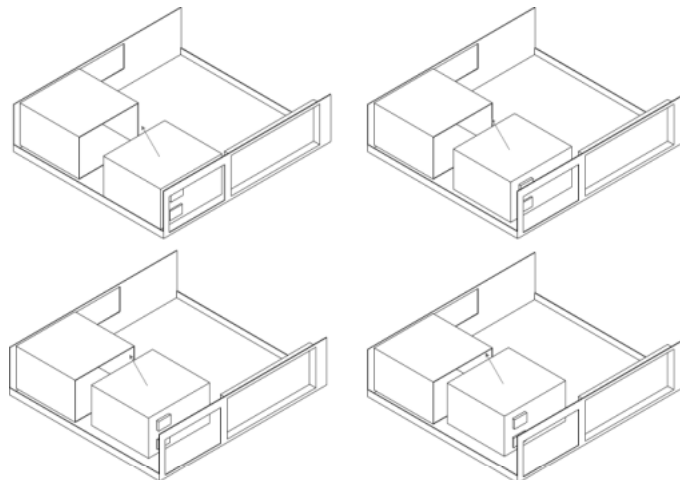


Figura 3.42: Secuencia de desplazamiento de la fuente de alimentación según la dirección  $(0, -0.7071, 0.7071)$ .

Tras realizar el desplazamiento del objeto, hay que actualizar el grafo de relaciones del producto. En este caso no existiría ninguna relación entre las caras de los dos componentes, ya que no existe ningún contacto. Por lo tanto, no se eliminaría ninguna dirección para el cálculo de las trayectoria de desensamblado.

Además hay que actualizar los valores correspondientes a los polígonos que forman el componente "fuente de alimentación" para que contemplen el desplazamiento, con lo que las caras quedarían definidas de la siguiente manera:

Cara	Punto 1	Punto 2	Punto 3	Punto 4	Plano
P <sub>1</sub>	(254, 190, 120)	(404, 190, 120)	(404, 190, 35)	(254, 190, 35)	-y + 190 = 0
P <sub>2</sub>	(254, 190, 120)	(254, 190, 35)	(254, 331, 35)	(254, 331, 120)	-x + 254 = 0
P <sub>3</sub>	(254, 331, 35)	(254, 331, 120)	(404, 331, 120)	(404, 331, 35)	y - 331 = 0
P <sub>4</sub>	(404, 331, 120)	(404, 331, 35)	(404, 190, 35)	(404, 190, 120)	x - 404 = 0
P <sub>5</sub>	(404, 190, 35)	(254, 190, 35)	(254, 331, 35)	(404, 331, 35)	-z + 35 = 0
P <sub>6</sub>	(254, 190, 120)	(404, 190, 120)	(404, 331, 120)	(254, 331, 120)	z - 120 = 0
P <sub>7</sub>	(354, 335, 85)	(384, 335, 85)	(384, 335, 105)	(354, 335, 105)	-y - 335 = 0
P <sub>8</sub>	(384, 331, 85)	(384, 335, 85)	(384, 335, 105)	(384, 331, 105)	x - 384 = 0
P <sub>9</sub>	(384, 335, 105)	(384, 331, 105)	(354, 331, 105)	(354, 335, 105)	z - 105 = 0
P <sub>10</sub>	(354, 331, 105)	(354, 335, 105)	(354, 335, 85)	(354, 331, 85)	-x + 354 = 0
P <sub>11</sub>	(354, 335, 85)	(354, 360, 85)	(384, 331, 85)	(384, 335, 85)	-z + 85 = 0
P <sub>12</sub>	(354, 333, 45)	(384, 333, 45)	(384, 333, 65)	(354, 333, 65)	-y - 333 = 0
P <sub>13</sub>	(384, 333, 45)	(384, 333, 65)	(384, 331, 60)	(384, 331, 45)	x - 384 = 0
P <sub>14</sub>	(384, 333, 65)	(384, 331, 65)	(354, 331, 65)	(354, 333, 65)	z - 65 = 0
P <sub>15</sub>	(354, 331, 65)	(354, 333, 65)	(354, 333, 45)	(354, 333, 45)	-x + 354 = 0
P <sub>16</sub>	(354, 333, 45)	(354, 331, 45)	(384, 331, 45)	(384, 333, 45)	-z + 45 = 0

Tras generar un nuevo haz de rayos, se obtiene un histograma más detallado del entorno, y se selecciona la dirección de desensamblado como:

$$d = (0.7071, 0, 0.7071) \quad (3.66)$$

Con este resultado hay que volver a aplicar el problema de programación lineal para cada una de las caras del componente móvil, la "fuente de alimentación", con las caras del componente estático, la "carcasa interior".

Una vez realizado este cálculo, se observa que no existe ninguna colisión entre ambos componentes, con lo que, ya se tiene definida la trayectoria de desensamblado del componente, como un desplazamiento según la dirección definida en la ecuación 3.44 de 50mm, seguido de un desplazamiento según la dirección 3.66, que le permite desplazarse hasta el infinito, es decir, hasta el punto en el que se considera que el componente ya no interfiere con el resto, y de esa posición, trasladarlo hasta el almacén correspondiente.

### 3.7. Evaluación del algoritmo

En este apartado se presenta un estudio sobre el coste del algoritmo planteado a lo largo del capítulo.

Este método implica un coste en función del número de polígonos que compongan cada componente. Si se supone que el componente móvil posee n polígonos y el resto del producto esta compuesto por m polígonos entonces el coste del algoritmo viene determinado, una vez que se selecciona la dirección de desensamblado, por el coste de calcular a que distancia se puede desplazar el componente en la dirección señalada sin que colisione con otro componente. Para ello hay que plantear para cada cara del componente móvil un problema

de programación lineal con cada una de las caras del resto del producto. Lo que implicaría  $n \cdot m$  problemas de programación lineal, que pueden ser reducidos a una cuarta parte si se consideran sólo las caras móviles cuyo vector director esté en el plano de la dirección de movimiento con lo que se tendría  $n/2$  caras; y de igual manera si se consideran sólo las caras estáticas cuyo vector director se encuentre en la dirección opuesta a la dirección del movimiento tendremos que considerar  $m/2$  caras.

En la Figura 3.43 se aprecia el coste de calcular las colisiones con otros componentes según la dirección de desensamblado para un número caras variable.

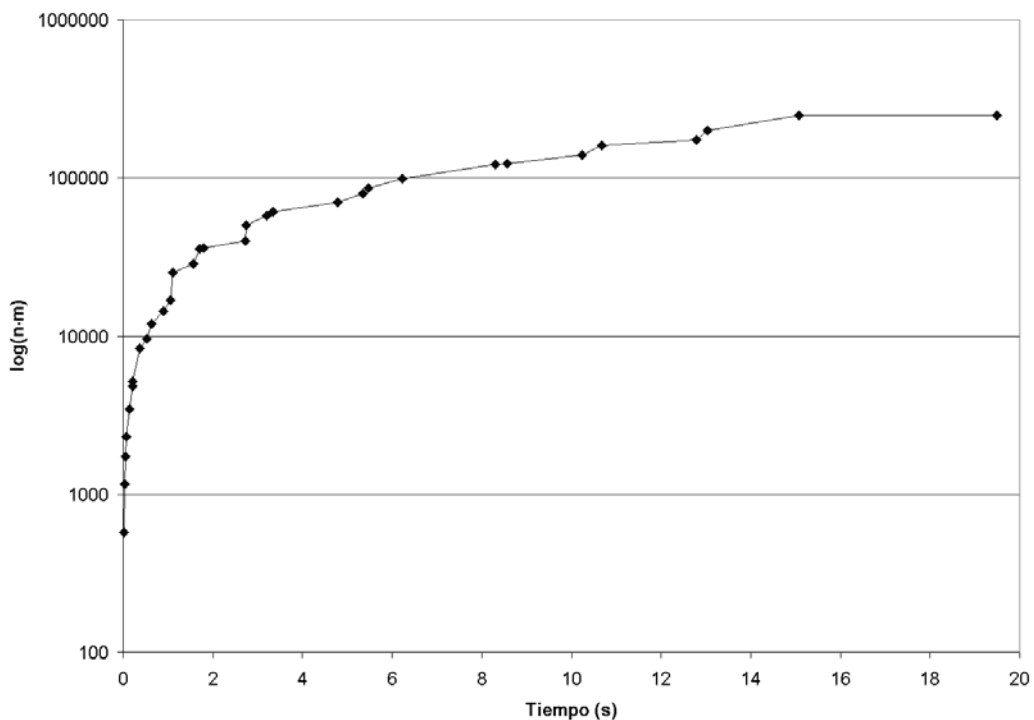


Figura 3.43: Coste del algoritmo de cálculo de colisiones según las caras existentes en el producto.

Este valor viene definido en el eje de ordenadas por el producto  $n \cdot m$ , correspondiente a la cantidad total de veces que hay que ejecutar el algoritmo de programación lineal para realizar la detección de las posibles colisiones. Se ha expresado dicha magnitud en valor logarítmico para una mayor claridad en la representación gráfica.

En el eje de abscisas se ha representado el tiempo empleado en la realización de los problemas de programación lineal en segundos.

Este tiempo se ha calculado utilizando un ordenador Pentium III a 450Mhz con 128Mbs de memoria RAM para realizar la ejecución de los problemas de programación lineal.

Como puede apreciarse, en la Figura 3.43, al aumentar el número de caras a analizar aumenta rápidamente el coste computacional necesario para llevarlo a cabo.

# 4

## **Fusión de la planificación y del generador de movimientos**

Hasta ahora en los capítulos 2 y 3, se ha visto cómo calcular independientemente la secuencia de componentes a separar, para conseguir desensamblar un determinado componente, así como la trayectoria de movimientos que deben realizarse para poder separar un componente del resto.

En este capítulo se unen ambos conceptos, lo que supone una importante aportación original de esta tesis, para realizar el desensamblado de un componente en particular del producto. Para ello se plantea una introducción en la que se expone el problema a resolver, tras esta se plantea la fusión de la secuencia de desensamblado con el generador de movimientos. Después se presenta como se tiene que operar para, en base a la fusión realizada, considerar el robot que debe llevar a cabo la tarea así como los distintos aspectos que esta fusión plantea.

Por último se presenta como se fusiona toda esta información con el sistema real de desensamblado automático, considerando los distintos módulos que intervienen y el sistema de visión artificial.

### **4.1. Introducción**

Este apartado plantea el problema que se pretende resolver mediante la fusión de la información, así como las consideraciones a tener en cuenta en esta fusión.

El problema a resolver consiste en realizar el desensamblado automático del componente deseado de un producto, para ello hay que decidir en que orden se deben separar los distintos componentes para alcanzar el objetivo, y

conjuntamente se tiene que calcular cual es la trayectoria de movimientos que debe seguir cada uno de esos componentes para separarlos del resto del producto. Así mismo hay que considerar que estas operaciones se llevan a cabo por un robot, teniendo por tanto restricciones extras por sus características. Además al disponer de un robot y de un producto, compuesto por diversos componentes, cada uno de ellos necesita de una herramienta distinta para desensamblarlo, pero puede darse el caso que varios componentes utilicen la misma herramienta, por lo que habrá que considerar el intercambio de herramientas que debe realizar el robot para poder llevar a cabo la totalidad de la tarea requerida minimizando los cambios a realizar, ya que cada cambio de herramienta conlleva un gasto de tiempo.

Por último cabe considerar las diferencias entre una simulación del sistema y la realización de las operaciones por un sistema real, incluyendo en esta parte la fusión de la información proveniente del sistema de visión artificial, mediante el uso del planificador global. En el primer caso la fusión se realiza basándose en la información contenida en la base de datos, mientras que en el otro caso dicha información se actualiza con datos provenientes del sistema sensorial.

## 4.2. Fusión de la planificación con el generador de movimientos

En este caso la fusión de la información que se pretende llevar a cabo consiste en unir la proveniente del planificador de desensamblado, la secuencia de acciones de desensamblado, con la del generador de movimientos para cada uno de los componentes implicados.

La fusión se lleva a cabo tomando cada una de las acciones implicadas en la secuencia de desensamblado y generando para cada una de ellas la secuencia de movimientos que tiene que realizar el componente o los componentes implicados en ella.

Antes de llevar a cabo esta fusión de la información hay que considerar que en el entorno real el proceso de desensamblado consta de una mesa de trabajo sobre la que se llevan a cabo las tareas de desensamblado (Figura 4.1).

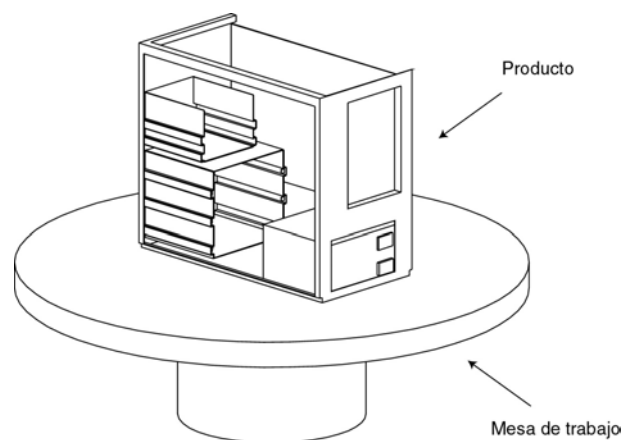


Figura 4.1: Entorno de trabajo el producto sobre la mesa de trabajo.

Con lo que a la hora de utilizar el generador de movimientos hay que incluir el modelo geométrico de la mesa de trabajo, ya que implica una serie de restricciones a tener en cuenta según su geometría.

Por lo tanto si se quiere desensamblar un determinado componente o un montaje, entonces la fusión de esta información se realiza de la siguiente forma:

- Calcular la secuencia de desensamblado para el componente deseado. Obteniendo las secuencia de acciones:  $a_1, a_2, \dots, a_n$ .
- Para  $i=1$  hasta  $n$  hacer:
  - Calcular los movimientos de desensamblado correspondientes a la acción  $a_i$ .
  - Realizar los movimientos de separación obtenidos en el paso anterior.
- Fin para.

### 4.3. Fusión considerando un robot

Este planteamiento tiene en consideración que el desensamblado lo va a realizar un robot en concreto, con las limitaciones que ello impone al sistema, y además el entorno de desensamblado incluye una mesa de trabajo para realizar las tareas de separación de los componentes (Figura 4.2).

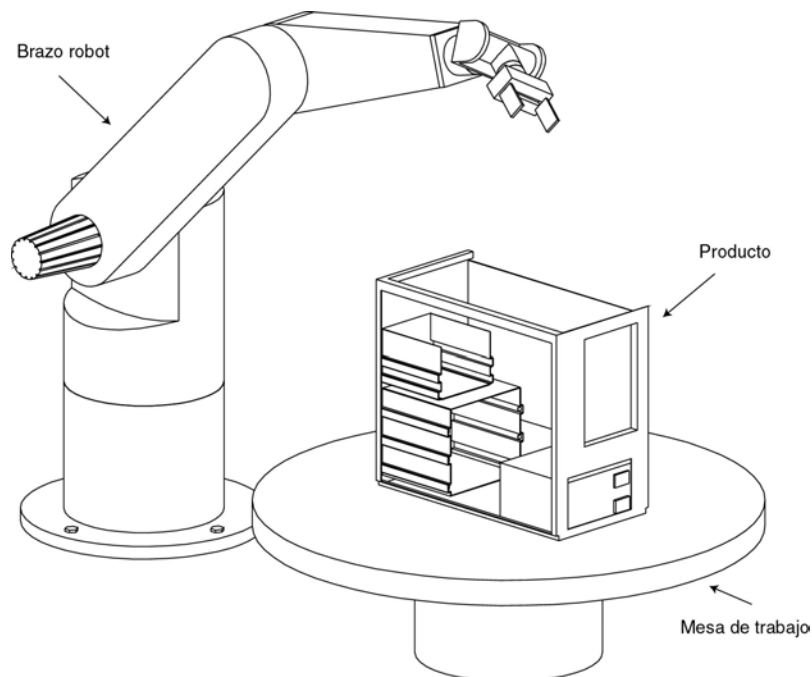


Figura 4.2: Entorno de trabajo formado por un robot y el producto sobre la mesa de trabajo.

Por lo tanto es necesario considerar que el extremo del robot tiene que realizar una determinada trayectoria lineal que viene definida por el generador de movimientos de desensamblado.

A la hora de calcular la trayectoria a seguir por el robot se pueden tener en cuenta diferentes aspectos [14] [164] [167] [171] [196] [266] como: minimizar el tiempo, no forzar las articulaciones, realizar un movimiento suave, mantener una fuerza constante, etc. El método que se utiliza en esta

tesis consiste en comprobar si la trayectoria de desensamblado es correcta cinemáticamente y si se puede llevar a cabo en el entorno de trabajo existente.

La consideración del robot se puede realizar de dos maneras distintas: la primera de ellas, la más sencilla de realizar, consiste en suponer el producto a desensamblar fijo en una posición, por ejemplo sobre la mesa de trabajo, y el robot ubicado en una posición conocida. Con esta estrategia, cada vez que se calcula un desplazamiento a realizar por el generador de movimientos de desensamblado, hay que comprobar si el robot puede realizarlo, lo que implica que pueda llegar al punto origen del desplazamiento con una orientación determinada y pueda desplazarse al punto destino manteniendo la orientación ya que se trabaja con trayectorias cartesianas lineales.

A la hora del cálculo de las trayectorias lineales se pueden seguir distintas aproximaciones según los parámetros y la forma en que se realiza la trayectoria. Además también influye el tipo de interpolador que se utilice, con lo que se consigue una trayectoria más o menos ajustada a la línea recta deseada.

En esta tesis se ha optado por utilizar una trayectoria punto a punto, a la hora de realizar las comprobaciones cinemáticas, lo que quiere decir que el robot debe poder posicionarse en el origen del movimiento y en el destino, con lo que se reducen los cálculos necesarios ya que sólo hay que comprobar dos posiciones. Esta aproximación es buena si se realizan desplazamientos pequeños para cada movimiento, con lo que se asegura, parcialmente, la linealidad de la trayectoria entre ambos puntos.

Otra manera de plantear el problema es con el robot en una posición conocida y el producto sobre la mesa de trabajo cuya posición también es conocida, de tal manera que el producto se puede trasladar y rotar sobre la mesa de trabajo. Este último planteamiento es el que se utiliza en esta tesis.

El problema de introducir un brazo robot en el entorno de trabajo se considera como un problema cinemático, sin considerar las relaciones dinámicas que aparecen.

Para poder resolver este problema se necesita conocer la cinemática inversa del brazo robot que se va a utilizar para realizar el desensamblado, ya que ésta permite calcular si se puede posicionar en un lugar y con una orientación determinada o no. De tal manera que la posición y orientación del extremo, para coger el componente, podemos expresarla mediante una matriz de transformación homogénea como:

$${}^{\text{base}}\mathbf{T}_{\text{extremo}} = \begin{bmatrix} x_{\text{extremo}}^{\text{base}} & y_{\text{extremo}}^{\text{base}} & z_{\text{extremo}}^{\text{base}} & x_{\text{extremo}}^{\text{base}} \\ y_{\text{extremo}}^{\text{base}} & y_{\text{extremo}}^{\text{base}} & y_{\text{extremo}}^{\text{base}} & y_{\text{extremo}}^{\text{base}} \\ z_{\text{extremo}}^{\text{base}} & z_{\text{extremo}}^{\text{base}} & z_{\text{extremo}}^{\text{base}} & z_{\text{extremo}}^{\text{base}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.1)$$

Donde se ha utilizado la siguiente notación:

$(x_{\text{destino}}^{\text{origen}}, y_{\text{destino}}^{\text{origen}}, z_{\text{destino}}^{\text{origen}}) \rightarrow$  se corresponden con las coordenadas  $(x, y, z)$  del vector que va desde el origen del sistema de coordenadas "origen" hasta el origen del sistema de coordenadas "destino".



$(x_{x_{destino}^{origen}}, y_{y_{destino}^{origen}}, z_{z_{destino}^{origen}}) \rightarrow$  se corresponden con las coordenadas  $(x, y, z)$  del vector unitario que representa la dirección del eje  $x$  del sistema de coordenadas "destino" respecto del sistema de coordenadas "origen".

${}^{origen}T_{destino} \rightarrow$  representa la matriz de transformación homogénea para convertir las coordenadas de un vector respecto del sistema "destino" en las correspondientes respecto al sistema "origen".

Si se utilizara el planteamiento en el que el producto se encuentra estático en el centro de la mesa de trabajo y con una orientación de cero grados y además se ha planteado el modelo del entorno para realizar el cálculo de la trayectoria de movimientos incluyendo la mesa, entonces, se está incluyendo las restricciones correspondientes a la mesa en el modelo, ya que añade direcciones en las que es imposible realizar un movimiento.

Por otra parte a la hora de desensamblar un determinado componente, hay que decidir por qué punto debe engancharlo el brazo robot, para lo cual el sistema se basa en información de la base de datos y del sistema de visión que le indicarán cuál es el punto óptimo de enganche así como la orientación. Para ello se utiliza información existente en la base de datos sobre dónde debería encontrarse dicho punto en el componente. Hay que indicar que el punto de agarre dependerá de la herramienta que se esté utilizando para realizar el desensamblado del componente, ya que para un mismo componente pueden existir distintos puntos de agarre según la herramienta que se utilice para llevar a cabo el desensamblado.

Como se acaba de mencionar se incluye una herramienta en el problema, que corresponderá con el extremo del robot, de tal manera que dicha herramienta y por consiguiente el robot entero, debe de ser incluido en el modelo geométrico para poder calcular las posibles colisiones de éste con el resto del producto ya que existirán configuraciones en las que un robot no pueda llegar a coger un componente, porque no pueda introducir su extremo por una determinada parte del producto, de tal manera que habrá que buscar una vía alternativa para poder desensamblar el componente. Este caso se presenta con mayor detalle en el apartado 4.3.1.

Como puede verse en la Figura 4.3 existen configuraciones en las que no es posible enganchar un componente, debido a que la herramienta intersecta con alguna parte del producto, con lo cual de las posibles configuraciones habrá que seleccionar la que no intersecte el resto del producto.

La introducción de todo el robot a la hora de realizar los cálculos de colisiones incrementa sustancialmente el coste computacional de los mismos con lo que, si se conoce que algunos eslabones no influyen en las colisiones, se pueden eludir, como sería en caso de los primeros eslabones en la mayoría de los robots, e incluso dependiendo del producto a desensamblar, se podría considerar sólo la herramienta que debe realizar la tarea.

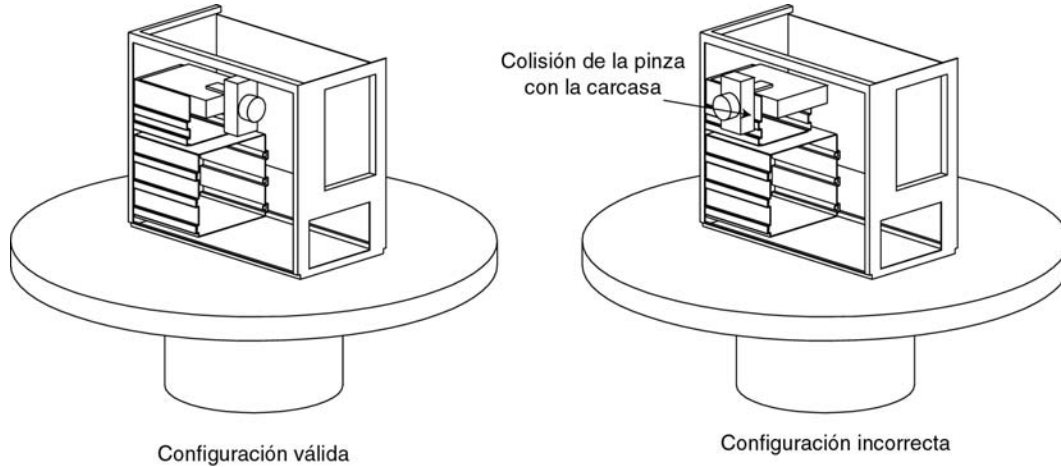


Figura 4.3: Varias posibilidades de desensamblar el disco duro.

Una de las aportaciones que plantea esta tesis consiste en un método para calcular la trayectoria teniendo en cuenta la posibilidad de trasladar y rotar el producto. Estas posibilidades amplían enormemente la capacidad de desensamblado de un robot ya que le añaden al sistema grados de libertad adicionales. En este caso el número de grado de libertad que se van a añadir al sistema es de tres, permitiendo una traslación en el eje  $x$  ( $d_x$ ), una traslación en el eje  $y$  ( $d_y$ ), así como una rotación a lo largo del eje  $z$  ( $\theta$ ), definidas por la matriz de transformación  $\mathbf{T}$  siguiente:

$$\mathbf{T} = \begin{bmatrix} \cos \theta & -\text{sen } \theta & 0 & d_x \\ \text{sen } \theta & \cos \theta & 0 & d_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

En la Figura 4.4 se muestran las relaciones existentes entre el extremo del robot, la base del mismo y la mesa de trabajo.

En donde la matriz de transformación  ${}^{\text{base}}\mathbf{T}_{\text{mesa}}$  se corresponde con la relación existente entre el sistema de la base del robot y el sistema en la mesa de trabajo.

$${}^{\text{base}}\mathbf{T}_{\text{mesa}} = \begin{bmatrix} x_{\text{mesa}}^{\text{base}} & y_{\text{mesa}}^{\text{base}} & z_{\text{mesa}}^{\text{base}} & x_{\text{mesa}}^{\text{base}} \\ y_{\text{mesa}}^{\text{base}} & y_{\text{mesa}}^{\text{base}} & y_{\text{mesa}}^{\text{base}} & y_{\text{mesa}}^{\text{base}} \\ z_{\text{mesa}}^{\text{base}} & z_{\text{mesa}}^{\text{base}} & z_{\text{mesa}}^{\text{base}} & z_{\text{mesa}}^{\text{base}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.3)$$

También se tiene que la matriz de transformación  ${}^{\text{mesa}}\mathbf{T}_{\text{extremo}}$  se corresponde con la matriz que define la posición del extremo del robot para poder coger un componente respecto al sistema de coordenadas de la mesa. En base a estas relaciones se deduce, combinando las ecuaciones 4.1 y 4.3, que:

$${}^{\text{mesa}}\mathbf{T}_{\text{extremo}} = {}^{\text{base}}\mathbf{T}_{\text{mesa}}^{-1} \cdot {}^{\text{base}}\mathbf{T}_{\text{extremo}} \quad (4.4)$$

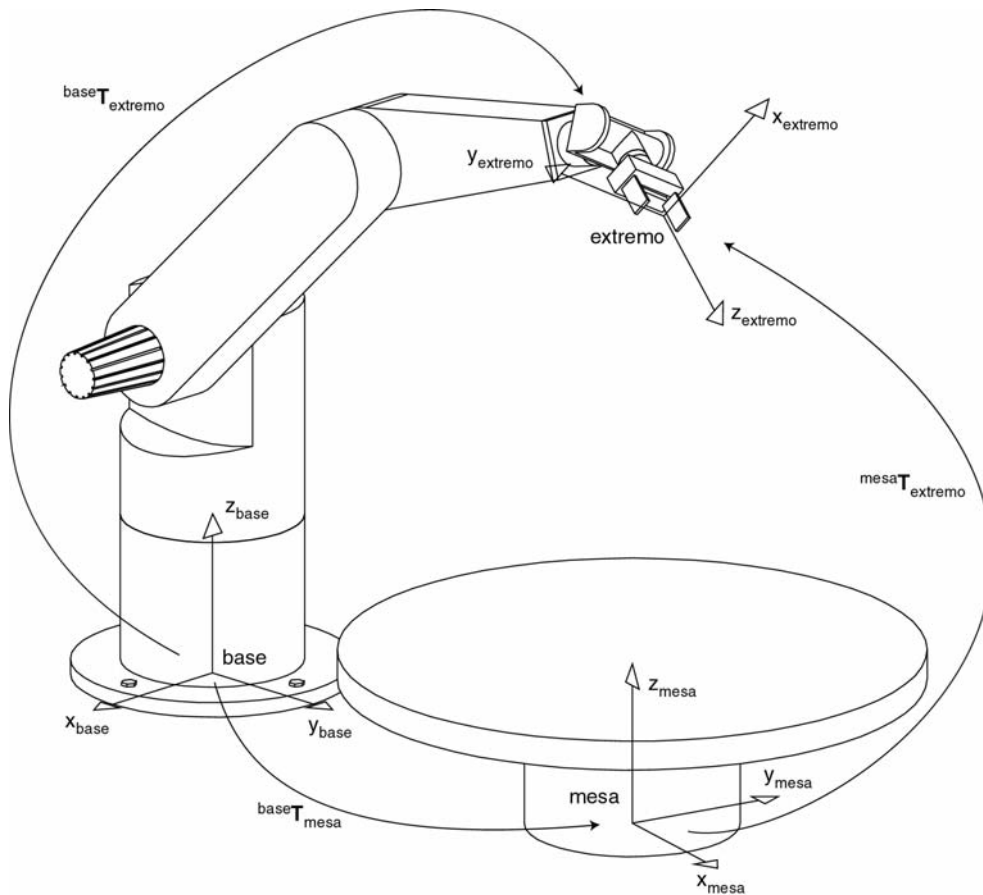


Figura 4.4: Relación entre el robot y la mesa de trabajo.

Una vez se tiene esta relación, se aplica el desplazamiento y rotación definidos por la matriz  $\mathbf{T}$  (Ecuación 4.2), de tal manera que se obtiene la nueva posición del extremo respecto de la mesa,  ${}^{\text{mesa}}\mathbf{T}'_{\text{extremo}}$  como:

$${}^{\text{mesa}}\mathbf{T}'_{\text{extremo}} = \mathbf{T} \cdot {}^{\text{mesa}}\mathbf{T}_{\text{extremo}} = \mathbf{T} \cdot {}^{\text{base}}\mathbf{T}_{\text{mesa}}^{-1} \cdot {}^{\text{base}}\mathbf{T}_{\text{extremo}} \quad (4.5)$$

Una vez obtenida la nueva posición del extremo del robot respecto a la mesa, hay que calcular cuál es dicha posición en función del sistema de la base del robot (Ecuación 4.3), para poder aplicarle la cinemática inversa; con lo que se obtiene la matriz  ${}^{\text{base}}\mathbf{T}'_{\text{extremo}}$  como:

$${}^{\text{base}}\mathbf{T}'_{\text{extremo}} = {}^{\text{base}}\mathbf{T}_{\text{mesa}} \cdot {}^{\text{mesa}}\mathbf{T}'_{\text{extremo}} = {}^{\text{base}}\mathbf{T}_{\text{mesa}} \cdot \mathbf{T} \cdot {}^{\text{base}}\mathbf{T}_{\text{mesa}}^{-1} \cdot {}^{\text{base}}\mathbf{T}_{\text{extremo}} \quad (4.6)$$

Con la nueva matriz, que define la localización del extremo del robot, hay que resolver la cinemática inversa para esa posición y orientación, considerando siempre que es preferible mantener los desplazamientos con un valor de cero, si se dispone de varias posibilidades, así como que el giro deberá ser nulo, si es posible, para evitar movimientos extras del producto una vez posicionado sobre la mesa de trabajo.

Con los cálculos realizados hasta este momento se tiene posicionado el robot en el punto correcto y se pueden calcular los movimientos de desensamblado a realizar para ser realizados por el robot.

Además de considerar el posicionamiento del producto sobre la mesa de trabajo hay que introducir información geométrica sobre el robot que tiene que realizar el desensamblado, de manera que se incluya dicha información en el cálculo de los movimientos de desensamblado y así evitar las posibles colisiones del robot con el producto.

La introducción del modelo del robot produce un incremento en el coste computacional ya que se introducen nuevos poliedros, que representan las distintas partes del robot, es por ello que se puede simplificar el cálculo de la secuencia de desensamblado si en lugar de considerar el modelo completo del robot sólo se considera la herramienta.

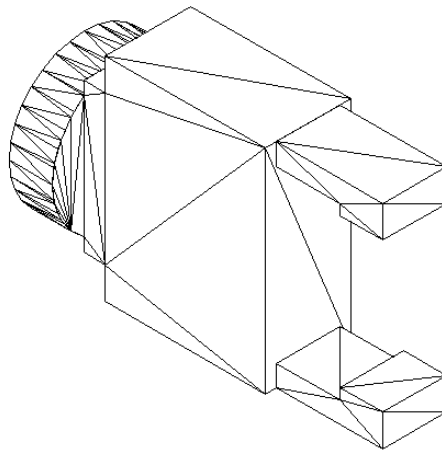
El hecho de considerar solamente la herramienta depende del entorno de desensamblado que se disponga y del producto a desensamblar.

En el siguiente apartado se describen los pasos a seguir si se considera sólo la herramienta en el modelo geométrico y tras éste se presenta el caso en el que es necesario introducir el robot completo y como se debe realizar.

#### 4.3.1. Introducción de una herramienta

La introducción de un robot implica como se ha dicho anteriormente la consideración de su modelo geométrico para el cálculo de la trayectoria de desensamblado.

Esta primera aproximación consiste en introducir sólo la herramienta a utilizar, junto con el eslabón correspondiente al extremo del robot, en el modelo geométrico (Figura 4.5).



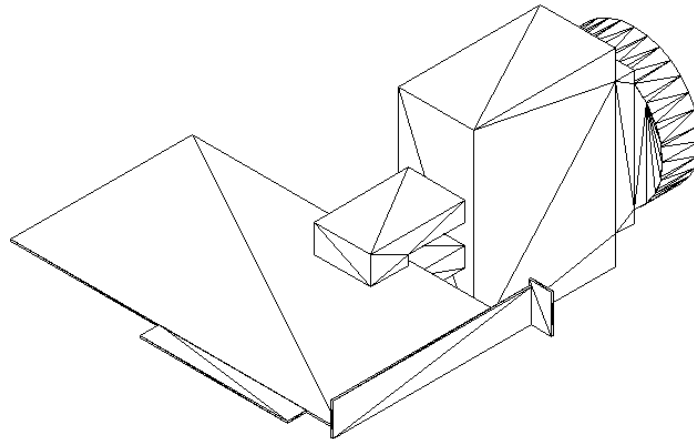
*Figura 4.5: Modelo geométrico de una pinza paralela de dos dedos.*

Esta aproximación, como puede verse en el modelo de la Figura 4.5, introduce un nuevo poliedro en el sistema, lo que implica una mayor cantidad de cálculos a la hora de generar la trayectoria de movimientos de desensamblado.

Una vez se tiene el modelo geométrico de la herramienta que va a utilizar el sistema para realizar el desensamblado del componente, hay que localizarla correctamente en el modelo geométrico del producto.

Teniendo en cuenta que la herramienta debe enganchar a un componente por un punto determinado, el punto de agarre, se considera la herramienta

como parte del propio componente a desensamblar, posicionada en el punto de enganche (Figura 4.6).



*Figura 4.6: Modelo geométrico de la herramienta enganchada al componente.*

Una vez realizada esta fusión, entre la herramienta y el componente, hay que comprobar que el nuevo modelo del componente no intersecta con el modelo de otro componente, ya que de ser así dicha posición de enganche para la herramienta no sería válida y habría que seleccionar otro punto de enganche o utilizar una herramienta distinta. Si estas opciones no fuesen posibles entonces el componente seleccionado no puede ser desensamblado mientras no se desensamble el componente que produce la colisión. Y en consecuencia habría que informar al planificador de desensamblado para que seleccione otro componente a desensamblar o decida que el componente deseado no se puede desensamblar utilizando las herramientas disponibles y con el producto actual.

Si por el contrario no hay ninguna intersección entre los componentes se puede proceder al cálculo de la trayectoria de desensamblado tal y como se ha explicado en el capítulo 3.

#### 4.3.2. Introducción del robot completo

En esta aproximación se debe considerar que se introduce el modelo geométrico del robot en el sistema. Por lo que se va a incrementar el coste computacional debido al incremento de poliedros.

La introducción del modelo completo del robot no siempre es necesaria; tal y como se ha comentado anteriormente, en muchos casos valdría con incluir sólo el modelo correspondiente a la herramienta. Se debe introducir cuando se disponga de un entorno donde se requiera que parte del robot, más allá de la herramienta, se introduzca dentro de lo que se podría considerar como el volumen ocupado por el producto.

Por ejemplo en la Figura 4.7 se muestra la necesidad de considerar todo el robot para poder desensamblar el ladrillo objetivo ya que se debe realizar el desensamblado por la zona libre de ladrillos, la que limita los movimientos del robot. Como se aprecia, hay configuraciones que son imposibles de realizar debido a que se intersectan con la zona ocupada por el resto de componentes.

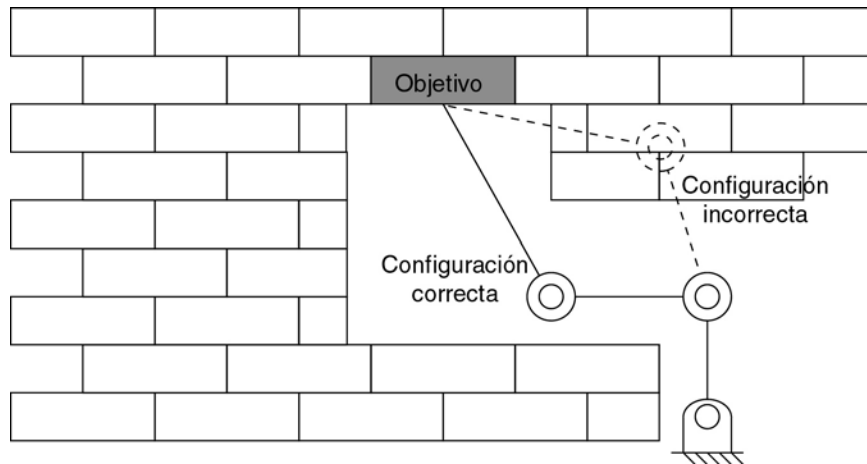


Figura 4.7: Necesidad de considerar el modelo completo del robot para realizar el desensamblado.

La manera de proceder consiste en unir la herramienta al componente a desensamblar por su punto de enganche, tal y como se realizaba en el caso de incluir sólo la herramienta, pero en este caso, además se introduce el modelo completo del robot unido a ésta, no sólo la herramienta, teniendo en cuenta el modelo cinemático correspondiente para asignarle una posición a las articulaciones que sea válida, para evitar problemas como el representado en la Figura 4.7.

En este caso pueden existir varias soluciones para posicionar el robot, por lo que se opta por aquella en la que ninguna parte del producto interseca algún eslabón del robot, ya que dicha configuración no se puede utilizar.

La manera de proceder para realizar el cálculo de la trayectoria de desensamblado varía respecto a la manera explicada en el capítulo 3 de la siguiente manera:

1. Se calcula la dirección de desensamblado a utilizar.
2. Se desplaza el componente según dicha dirección.
3. Se actualiza el valor de todas las articulaciones del robot, para que alcance la nueva posición. Si no es posible dicha posición no es una dirección de desensamblado válida.
4. Se actualiza el modelo geométrico del robot según la nueva posición
5. Se comprueba que no existe intersección del robot con el resto del producto.
  - a. Si existe intersección entonces hay que seleccionar otra solución de la cinemática inversa para dicha posición.
  - b. Si no existe intersección es un movimiento válido.
6. Si se aún no se tiene desensamblado el componente se vuelve al paso 1 para realizar otro desplazamiento.

De esta manera se consigue calcular la trayectoria que debe seguir el componente para separarse del resto del producto, teniendo en cuenta que existe un robot concreto en el sistema para realizar la tarea.

### 4.3.3. Utilización de diversas herramientas

A la hora de obtener una planificación de desensamblado, ésta está compuesta por una secuencia de acciones, las cuales implican el desensamblado de una serie de componentes. Así mismo si observamos la secuencia, tal y como se ha dicho anteriormente, existen operaciones que se pueden realizar en paralelo, por lo tanto si se dispone sólo de un robot entonces hay que tener en cuenta que el sistema las tiene que realizar secuencialmente, pero ¿cuál es la mejor secuencia?

Para poder decir que una secuencia es mejor que otra se utiliza la información relativa a la herramienta necesaria para llevar a cabo el desensamblado de cada uno de los componentes; de manera que se va a minimizar el número de cambios de herramientas necesarios para llevar a cabo las operaciones de desensamblado.

En base a esta información se ordenan, las operaciones de desensamblado en las que se ha dividido cada acción, en una lista, de manera que se puedan realizar secuencialmente. Para configurar la lista de operaciones se lleva a cabo la ejecución de un algoritmo de ordenación que tenga en consideración un determinado coste a la hora de realizar un intercambio de herramientas, de manera que se opte por realizar las operaciones que requieran la misma herramienta seguidas y no alternándolas.

Un ejemplo en el que se ve claramente la pérdida de eficacia del sistema si se esta cambiando continuamente de herramienta aparece cuando se desea desensamblar de un ordenador una serie de tarjetas, las cuales se encuentran unidas a la carcasa mediante un tornillo, por lo tanto se puede optar por eliminar el tornillo de la tarjeta 1, después desensamblar la tarjeta 1, a continuación eliminar el tornillo de la tarjeta 2, tras esto desensamblar la tarjeta 2, etc.

La otra opción consiste en eliminar primero el tornillo de la tarjeta 1, el tornillo de la tarjeta 2, etc., y a continuación desensamblar las distintas tarjetas, primero la 1, luego la 2, etc.

Como puede apreciarse en este ejemplo en el primer caso para cada operación se está realizando un cambio de herramienta, luego si suponemos que inicialmente se dispone de la herramienta correcta se tendría que se han realizado, para un total de  $n$  tarjetas,  $2n-1$  cambios de herramientas.

Por otra parte para el segundo ejemplo, sólo se ha realizado 1 cambio de herramienta, independientemente del número de tarjetas que se estén desensamblando; siempre que se suponga que todas las tarjetas se pueden desensamblar utilizando la misma herramienta. En caso contrario, si se requieren  $m$  herramientas distintas ( $m \leq n$ ) se tendría que hay que realizar un total de  $m$  cambios de herramientas, ya que se podría ordenar la extracción de las tarjetas para desensamblar seguidas aquellas que requieran de la misma herramienta.

Observando estos resultados, se aprecia claramente que el realizar una ordenación previa de las operaciones a realizar, teniendo en consideración las herramientas que se deben utilizar para el desensamblado de cada uno de los componentes, permite un importante ahorro de tiempo.

El algoritmo seguido para realizar la ordenación de las operaciones de desensamblado es:

1. Para cada uno de los nodos raíces del grafo de operaciones obtenido por el planificador de desensamblado.
  - a. Para cada uno de los nodos hijos del nodo raíz:
    - i. Suponer que se ejecuta dicha operación y calcular el coste del resto del grafo añadiendo a la lista de nodos raíces los hijos del nodo raíz y eliminar éste de la lista.
2. Volver al paso 1 si la lista de nodos raíces no es vacía.
3. Seleccionar la secuencia de que posee un menor coste, en este caso el coste se calcula como la cantidad de cambios de herramientas.

#### 4.4. Desensamblado virtual

Este apartado describe como se fusiona la información de los apartados anteriores para simular una secuencia completa de desensamblado de un producto. Esta simulación se ha denominado desensamblado virtual y constituye una aportación original de esta tesis.

La realización del desensamblado virtual se divide en un conjunto de tareas (Figura 4.8), en donde se realiza una planificación de la secuencia de desensamblado en base a la información contenida en la base de datos y al tipo de desensamblado que se quiera realizar.

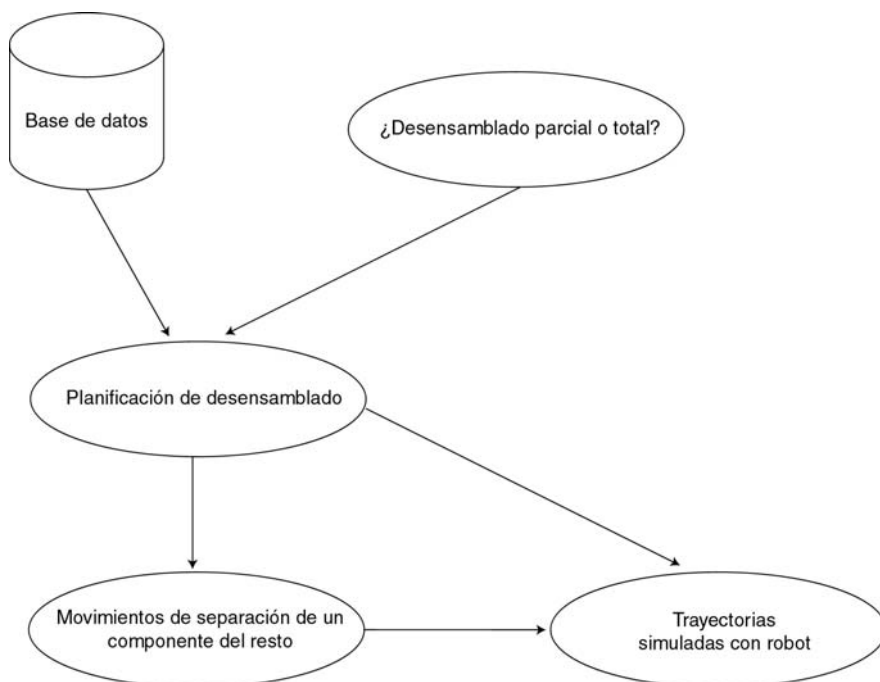


Figura 4.8: Esquema del sistema de desensamblado virtual.

Considerando esta planificación, se calcula tal y como se ha dicho en los apartados precedentes la secuencia de desensamblado correspondiente, teniendo en cuenta si se está utilizando un robot, sólo la herramienta o simplemente calculando la secuencia de desensamblado del producto sin utilizar ningún robot.



Por tanto para generar las distintas trayectorias de desensamblado se utiliza el siguiente algoritmo:

1. Calcular la secuencia de acciones de desensamblado  $a_1, a_2, \dots, a_n$  para desensamblar el componente objetivo.
2. Para  $i=1$  hasta  $n$  hacer
  - Calcular los movimientos de desensamblado correspondientes a la acción  $i$ .
  - Si existe un problema al desensamblar el componente:
    - Si el problema se debe a que la herramienta utilizada intersecta el volumen correspondiente al resto del producto se buscan en la base de datos si existen herramientas alternativas que permitan desensamblar el componente.
    - Si existen herramientas alternativas: se selecciona la siguiente, según criterios incluidos en la base de datos e incluso teniendo en cuenta cual es el punto de enganche para cada una de ellas, y se vuelve al comienzo del bucle.
    - Si no existen herramientas alternativas: entonces se parte del hecho de que el componente no puede ser desensamblado en estos momentos debido a restricciones geométricas del producto y por tanto hay que ver si existe la posibilidad de desensamblar otro componente en estos momentos, si no existe tal posibilidad entonces el desensamblado se da por terminado incluyendo un mensaje de error debido a que resulta imposible seguir desensamblando por las características geométricas del producto.
    - Si el problema se debe a que no existe ninguna trayectoria de desensamblado que permita separar el componente del resto entonces hay que ver si se puede desensamblar otro componente, para posteriormente intentar desensamblarlo otra vez, ya que en estos momentos aunque el desensamblado tendría que ser intercambiable, por cuestiones de la geometría de los componentes no se puede llevar a cabo.
  - Sino hay ningún problema entonces se desensambla el componente correspondiente.

### 3. Fin Para

Con esto se tendría generada la secuencia para llevar a cabo el desensamblado del componente o del montaje deseado, con lo que se podría simular rápidamente tras haber realizado todos los cálculos.

## 4.5. Desensamblado real

Además de realizar un desensamblado virtual un sistema flexible de desensamblado automático debe de poder realizar sobre el sistema real las

operaciones correspondientes, por lo tanto en este apartado se explica como se lleva a cabo la fusión de toda la información generada hasta estos momentos en la presente tesis para que un sistema real pueda llevar a cabo el desensamblado de los productos que se le presenten. Siendo este apartado una importante aportación original de esta tesis.

A la hora de plantear el desensamblado real se pueden optar por distintas estrategias:

- Realizar un desensamblar virtual inicial y después el desensamblado real sin ningún tipo de supervisión ni control sobre lo que se tiene realmente.
- Llevar a cabo un desensamblado real que se realiza simultáneamente al virtual, de manera que la información utilizada para el desensamblado virtual se actualiza conforme se van separando los componentes del producto y por tanto permite una mayor flexibilidad a la hora de realizar un desensamblado.

La primera aproximación plantea un sistema en el que se supone que se dispone de un modelo perfecto del producto a desensamblar, lo cual no es realmente viable ya que un producto a desensamblar puede haber cambiado durante el transcurso de su ciclo de vida y por tanto no es correcto realizar dicha suposición sobre un producto a desensamblar, con lo que sólo es factible para pruebas de laboratorio, ya que permite comprobar la eficacia de los algoritmos de planificación de desensamblado, el generador de movimientos así como la fusión de información entre ambos.

Por otra parte, el segundo planteamiento sí que permite realizar un sistema flexible de desensamblado ya que se podrá adaptar a posibles errores en el diseño y también permite trabajar en un entorno que no se encuentre totalmente estructurado ya que va a utilizar un sistema sensorial para reconocer y localizar los componentes del producto que se desea desensamblar.

Este sistema sensorial puede ser de múltiples tipos, según las características del producto que se desee desensamblar y de la precisión que se requiera para ello. En esta tesis se plantea el uso de un sistema de visión artificial.

Las tareas a seguir para poder realizar el desensamblado real del producto se pueden esquematizar tal y como se muestra en la Figura 4.9.

Los pasos que debe seguir para llevar a cabo el desensamblado del producto se detallan como:

1. Partiendo del modelo del producto existente en la base de datos, y del tipo de desensamblado a realizar. Se realiza el reconocimiento del producto a desensamblar, utilizando algoritmos de visión artificial, que permiten seleccionar el modelo correcto de la base de datos, además se utilizan los algoritmos de visión artificial para ajustar los tamaños y posiciones reales del producto y de sus componentes.

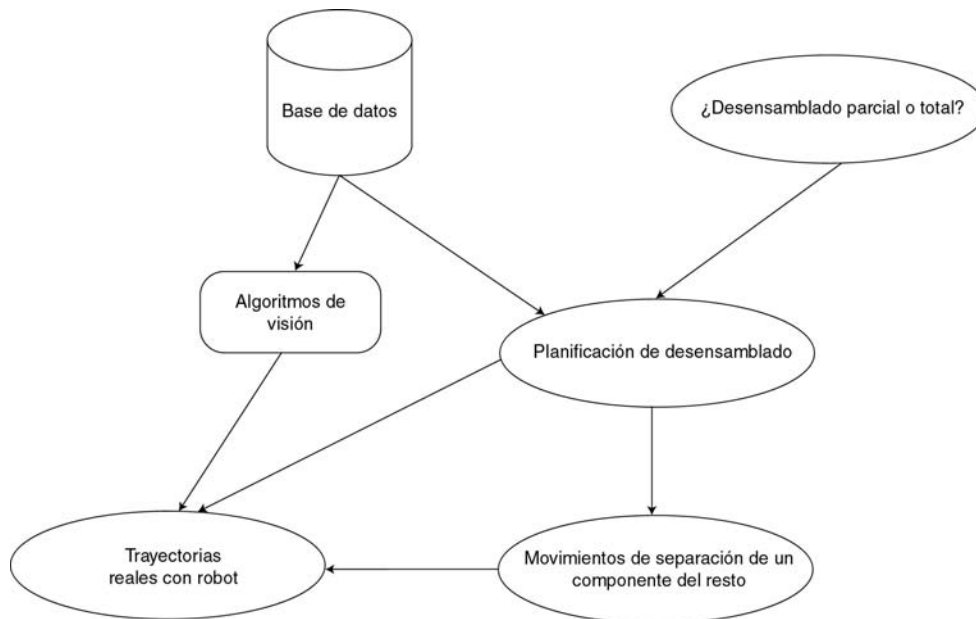


Figura 4.9: Esquema del sistema de desensamblado real.

Se tiene en cuenta que el modelo relacional seleccionado es un modelo genérico y que no posee información detallada sobre los componentes internos del producto que se esté analizando. A este respecto el modelo contendrá información sobre a que nivel de desensamblado es necesario llegar para poder reconocer la existencia de los distintos componentes.

2. Utilizando la información del sistema de visión, se ajusta el modelo a la realidad añadiendo los componentes que sean visibles y eliminando aquellos que no existen en el producto.
3. Una vez se tiene el modelo del producto el sistema calcula la secuencia de desensamblado a seguir.
4. En base al resultado obtenido se calcula la trayectoria de movimientos de desensamblado que debe realizar el componente a desensamblar.

La trayectoria se calcula teniendo en cuenta información tanto de la base de datos, modelo geométrico genérico, así como la información obtenida por el sistema de visión, tamaño y localización de los distintos componentes. Además la trayectoria se calcula teniendo en cuenta el método de fusión requerido entre la planificación y el generador de movimientos para que considere además el robot o en su defecto que considere la herramienta que va a llevar a cabo el desensamblado.

5. Si el componente desensamblado es el componente objetivo ya se ha terminado el proceso de desensamblado.
6. Una vez desensamblado un componente, se actualiza el modelo mediante los algoritmos de visión, para conseguir mejorar la información existente de manera que sea más exacta. Y para detectar aquellos componentes que no se podían detectar por la existencia del que ha sido desensamblado. Para ello, tal y como se

ha dicho, el modelo relacional de la base de datos debe incluir información que especifique al sistema de visión desde que momento del desensamblado se puede ver un componente, de manera que se aumente la información en el instante adecuado.

7. Cuando ya se ha actualizado el nuevo modelo del producto se sigue con el paso 3 del algoritmo para continuar desensamblando el resto de componentes requeridos.

Tal y como se ha dicho hay que actualizar el modelo del producto según se van desensamblando los distintos componentes, esto es debido a que un mal modelo geométrico produciría colisiones entre los componentes que implicarían un desensamblado erróneo del producto.

Por otra parte la actualización de los componentes permite obtener información sobre la existencia de ciertos componentes sin necesidad de realizar el desensamblado de algunos que sería necesario eliminar para poder acceder hasta ellos.

Un ejemplo de este tipo de situación aparece cuando se intenta desensamblar la disquetera de 5¼ de un ordenador. El montaje correspondiente a los periféricos no está accesible mientras no se elimine el componente "carcasa exterior"; no se puede desensamblar la disquetera mientras no se abra el ordenador, pero el saber si un ordenador tiene o no disquetera de 5¼ se puede realizar sin necesidad de desensamblar la carcasa exterior. Con lo que el reconocimiento de la existencia del componente, disquetera 5¼ se puede realizar inicialmente, con el ordenador sin desensamblar ningún componente. Esto sirve para no realizar desensamblados inútiles, ya que el desensamblado de la carcasa exterior en un ordenador que no disponga de disquetera de 5¼ no tendría sentido ya que al no poseer el componente no se puede desensamblar, con lo que se habría realizado una operación innecesaria (Figura 4.10).



Figura 4.10: Ordenador con disquetera de 5¼ y sin disquetera de 5¼.

Por otro lado si lo que se pretende desensamblar es un disco duro, para poder reconocer su existencia es necesario desensamblar el componente "carcasa exterior", ya que se trata de un componente que no es directamente visible desde el montaje "PC completo"; es decir hay que abrir el ordenador para comprobar si tiene o no un disco duro. Con lo que se podrían realizar

operaciones de separación de componentes en productos para los cuales no se va a poder desensamblar el objetivo, pero en este caso no se pueden evitar ya que no se puede conocer la existencia dichos componentes mientras no se realice el desensamblado de los que los ocultan.

# 5

## Resultados: Aplicación al desensamblado de PC's

En el presente capítulo se aplican los algoritmos planteados en esta tesis para realizar el desensamblado automático de varios PC's.

Este capítulo se divide en varias partes, entre las que se incluye inicialmente una descripción de la base de datos utilizada para la generación de las secuencias de desensamblado y almacenamiento de los modelos relacionales y geométricos de los productos.

Tras esto se presenta el cálculo de la secuencia de desensamblado para un ordenador completo, indicando en el modelo relacional todos los elementos que lo forman, incluyendo todos los elementos de enlace. A continuación se presenta como opera el sistema para fusionar las informaciones del planificador de desensamblado con el generador de trayectorias para llevar a cabo el desensamblado de un PC genérico.

Después se describe la arquitectura del sistema de desensamblado real disponible en el laboratorio y sobre el que se han realizado las pruebas del sistema flexible de desensamblado. Por último se incluyen dos ejemplos de desensamblado automático: uno sobre un PC sobremesa y el otro sobre un PC semitorre.

### 5.1. Base de datos

La información contenida en la base de datos se puede distribuir según el esquema relacional mostrado en la Figura 5.1 donde se aprecian las cuatro tablas principales: productos, componentes, enlaces y herramientas; así como los atributos más relevantes de cada una de ellas [267].

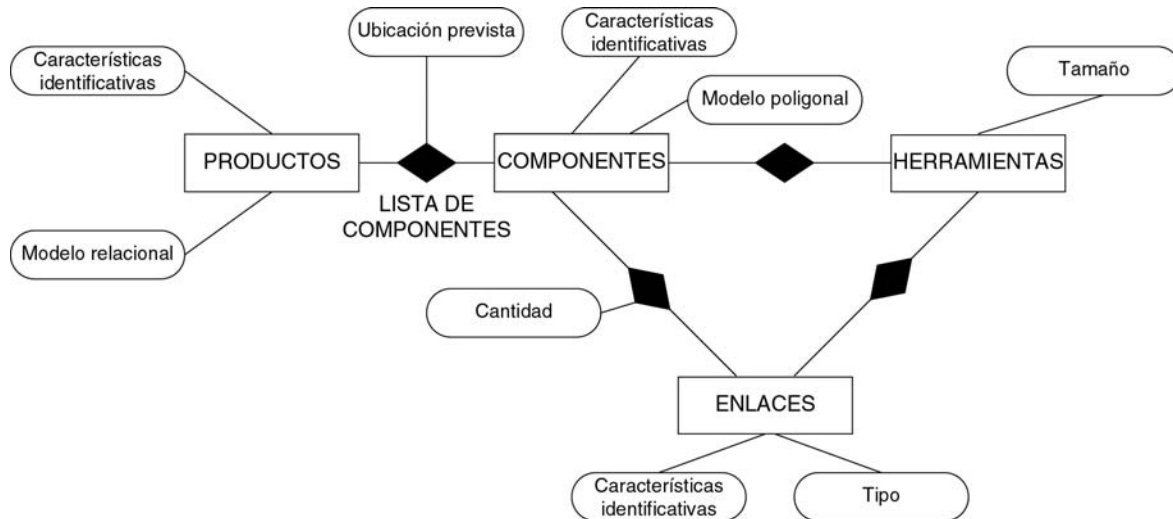


Figura 5.1: Esquema genérico del modelo entidad-relación de la base de datos.

Partiendo de esta información, a continuación se describe cada una de las tablas en mayor detalle, así como las relaciones que tienen con las demás.

En este apartado se añade el concepto de enlace en la base de datos; esto es debido a que por enlace se entiende, además de los elementos de enlace definidos en el apartado 2.3.1, a todos los elementos que se utilicen para unir dos componentes, ya que la base de datos está diseñada para incluir información para realizar un desensamblado destructivo, de manera que permite una definición más genérica de los productos de la que se utiliza en el modelo relacional.

### 5.1.1. Productos

La base de datos contiene la información sobre los productos a desensamblar, esta información se almacena en la tabla productos, en la que se incluyen las características más importantes que se necesiten conocer, tales como: características que permitan identificar el producto, el modelo relacional del producto, los componentes que lo forman, etc.

De esta manera los atributos principales de los productos son:

- Características identificativas del producto: se trata de una lista de características sobre las que es posible llegar a identificar de forma unívoca un determinado producto. Pueden ser características geométricas, fotométricas o estructurales. Esta información sirve para que el sistema de visión artificial reconozca los componentes que conforman el producto real.
- Modelo relacional: mantiene información sobre la relación jerarquizada de montaje de los componentes de un determinado producto. Se representa según el modelado relacional visto en el capítulo 2.
- Lista de componentes: mantiene información de todos los componentes que potencialmente pueden formar parte del producto, independientemente de que en el proceso de fabricación, el ensamblado del producto, estuvieran presentes o no. En un

ordenador la lista de componentes que pueden formar parte del mismo contendría: tarjetas, discos duros, cds, fuente de alimentación, etc. Esta información es necesaria para que el proceso de desensamblado pueda buscar, tanto componentes existentes en el proceso de fabricación, como otros que posteriormente se hayan incorporado, por ejemplo una tarjeta en un ordenador.

- Ubicación prevista de los componentes: Esta información sirve para que el sistema de visión pueda localizar más fácilmente un determinado componente; indica dónde es más probable encontrar el componente; por ejemplo, para las tarjetas en un ordenador, la ubicación prevista es en la parte posterior.

Esta información se distribuye en la base de datos sobre varias tablas, de manera que resulta fácil su utilización. En la Figura 5.2 se muestran los campos más relevantes así como el conjunto de tablas involucradas en el almacenamiento del modelo relacional de los productos. También se muestran algunas de las relaciones existentes entre las tablas.

Según la Figura 5.2 para almacenar el modelo relacional del producto se guardan las relaciones existentes entre los componentes, así como información sobre qué componentes forman el modelo, y también se almacenan los montajes involucrados en el modelo incluyendo los componentes que lo forman así como otros montajes que puedan estar incluidos.

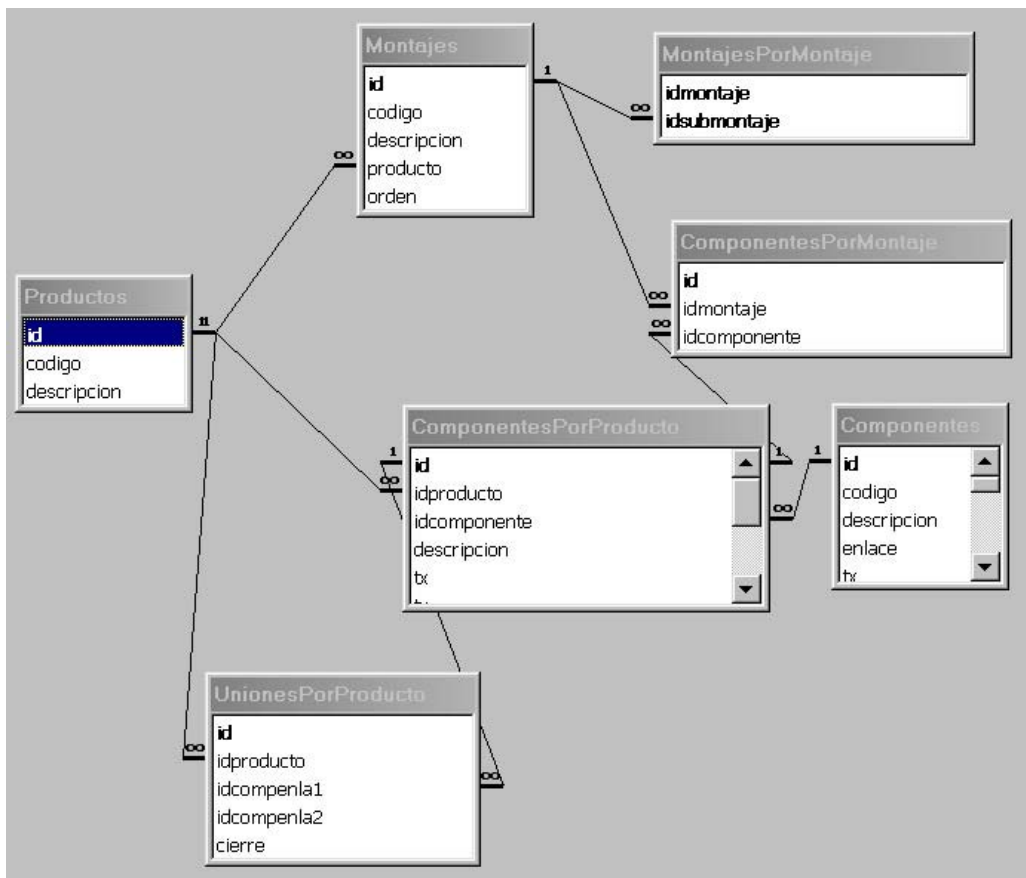


Figura 5.2: Tablas de la base de datos para el almacenamiento del modelo relacional de los productos.



### 5.1.2. Componentes

Esta tabla de la base de datos mantiene la información sobre los componentes que pueden llegar a formar parte de cualquier producto almacenado en la base de datos; todo posible componente debe estar reflejado aquí.

Además de contener información relativa a cada componente hay que mantener información sobre las herramientas necesarias para desensamblarlo, y sobre las relaciones existentes entre cada componente y los enlaces que pueda tener.

Teniendo en cuenta estas características, los principales atributos de los componentes son:

- Características identificativas del componente: se trata de una lista de características sobre las que es posible llegar a identificar, de forma unívoca, un determinado componente. Pueden ser características geométricas, fotométricas o estructurales, indicando, cuando sea posible, la localización de las mismas; por ejemplo una disquetera se puede identificar por una serie de descriptores de su cara frontal. Esta información sirve al sistema de visión artificial para identificar qué componentes existen realmente en el producto.
- Modelo poliédrico: se trata del modelo geométrico del componente, basado en un poliedro de caras planas que lo encierra. La información se almacena según el modelado geométrico de componentes del capítulo 3. El modelo de los componentes se ha realizado mediante prismas rectangulares y mediante modelos CAD.
- Tipos de enlace: mantiene información sobre los distintos tipos de enlaces que se pueden utilizar para conectar el componente con otro. Un componente puede requerir de más de un tipo de enlace, por ejemplo una tarjeta de un ordenador tiene generalmente un tornillo y la conexión en el bus de datos, lo que implica dos enlaces distintos.
- Cantidad de enlaces de cada tipo: almacena el máximo número de enlaces de un determinado tipo que potencialmente pueden existir. Esta información servirá para saber cuántos enlaces hay que eliminar; por ejemplo en el caso de los tornillos que unen la carcasa de un ordenador, se sabrá la cantidad máxima de ellos que se pueden encontrar. Así mismo en el caso de los tornillos que unen una tarjeta con la carcasa del ordenador, sólo existe como máximo uno, de manera que, a la hora de reconocerlos, sólo habrá que buscar un tornillo.
- Herramienta necesaria: almacena el tipo de herramienta que se necesita para poder separar el componente.
- Herramientas alternativas: contiene una lista de posibles herramientas alternativas a usar, en caso de que no fuese posible la eliminación por la herramienta prevista.

En la base de datos, las relaciones correspondientes a los componentes se expresan tal y como muestra la Figura 5.3, donde se ven las relaciones existentes entre los componentes y las herramientas para su desensamblado.

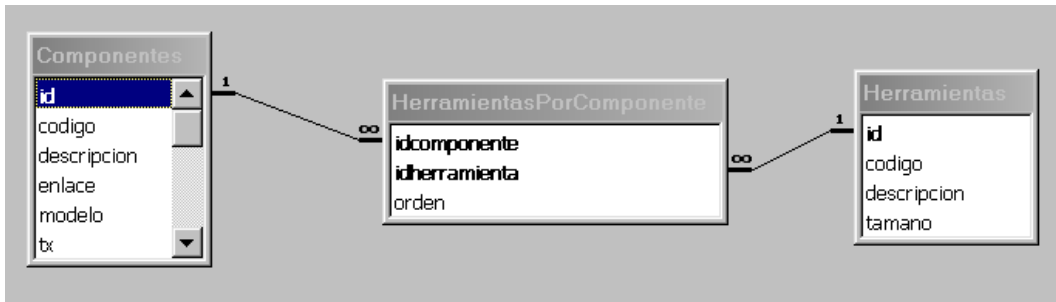


Figura 5.3: Tablas de la base de datos para el almacenamiento de los componentes así como las herramientas necesarias para desensamblarlos.

### 5.1.3. Enlaces

En esta tabla de la base de datos se guarda la información relativa a los enlaces. Dicha información incluye, aparte de las características identificativas del enlace, información sobre si es reversible o no, la herramienta necesaria para su eliminación, los movimientos requeridos para eliminarlo, etc.

Los atributos más importantes de los enlaces se detallan a continuación:

- Características identificativas del enlace: se trata de una lista de características sobre las que es posible llegar a identificar de forma unívoca un determinado enlace. Pueden ser características geométricas, fotométricas o estructurales. Esta información sirve al sistema de visión artificial para identificar un enlace en el producto.
- Tipo de enlace: mantiene información sobre si el enlace es reversible o irreversible. Para enlaces irreversibles habrá que utilizar métodos de desensamblado destructivo. En esta tesis se utilizan enlaces reversibles, ya que se parte de una planificación de desensamblado no destructiva.
- Herramienta necesaria: almacena el tipo de herramienta que se necesita para poder eliminar el enlace.
- Movimientos de eliminación: contiene información sobre los movimientos que han de realizarse para eliminar el enlace. Por ejemplo, giro y desplazamiento simultáneos en un tornillo.
- Herramientas alternativas: contiene una lista de posibles herramientas alternativas a usar, en caso de que no fuese posible la eliminación por la herramienta prevista; por ejemplo, un tornillo deteriorado tendrá, como herramienta alternativa, un proceso de eliminación destructivo por corte.

### 5.1.4. Herramientas

Esta parte de la base de datos contiene información sobre cada una de las herramientas que se pueden utilizar en el desensamblado de los productos. La tabla de herramientas puede contener: pinzas de dedos paralelos, ventosas, herramientas de corte, etc. (Figura 5.4).



Figura 5.4: Varios tipos de herramientas (Cortesía de Sckunk y Norgren).

Así mismo se requiere cierta información sobre cada herramienta, ya que el mismo tipo de herramienta no tiene por qué ser válido para desensamblar los mismos componentes; no es la misma ventosa la que se necesita para desensamblar la carcasa de un ordenador que la necesaria para desensamblar una fuente de alimentación; deben de ser de distinto tamaño, aunque sean la misma herramienta.

Es por ello que, de las herramientas se destaca, como el atributo más importante a considerar en la base de datos, el tamaño:

- **Tamaño:** describe el tamaño de la herramienta. Un tipo de herramienta puede tener varios tamaños, por ejemplo un destornillador plano según los tornillos a eliminar necesita un tamaño del extremo u otro; una ventosa dependiendo del componente a enganchar requiere un tamaño determinado; etc. (Figura 5.5).



Figura 5.5: Ventosas de distintos tamaños (Cortesía de Norgren).

## 5.2. Secuencia de desensamblado de un ordenador completo

En este apartado se presenta un ejemplo de un ordenador completo, así como el cálculo de distintas secuencias de desensamblado mediante el entorno desarrollado (Anexo A).

En la Figura 5.6 se muestra el modelo relacional de un ordenador completo, considerando para el mismo todos los elementos que lo forman.

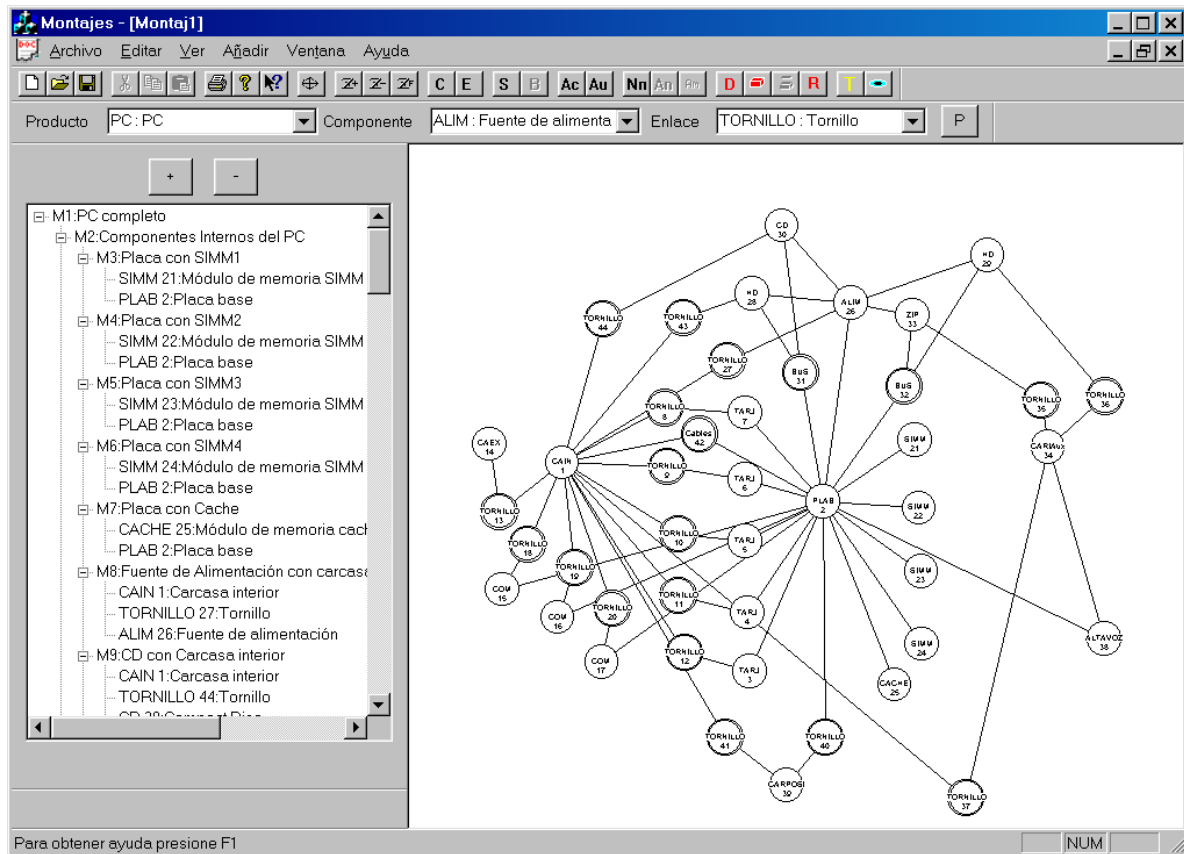


Figura 5.6: Modelo relacional de un ordenador completo.

El modelo relacional propuesto se corresponde con el ordenador mostrado en la Figura 5.7.



Figura 5.7: Ordenador completo utilizado para generar el modelo relacional de la Figura 5.6.

Teniendo esto en cuenta, se puede generar la tabla siguiente en la que se muestra una lista de los componentes y elementos de enlace existentes en el modelo:

C <sup>1</sup>	Descripción	E <sup>2</sup>	C	Descripción	E
1	Carcasa interior	No	23	Módulo de memoria, SIMM 3	No
2	Placa base	No	24	Módulo de memoria, SIMM 4	No
3	Tarjeta de TV	No	25	Módulo de memoria, cache	No
4	Tarjeta de red	No	26	Fuente de Alimentación	No
5	Tarjeta SCSI	No	27	Tornillos	Sí
6	Tarjeta de sonido	No	28	Disco duro (HD)	No
7	Tarjeta de vídeo	No	29	Disco duro (HD)	No
8	Tornillos	Sí	30	Disco compacto (CD)	No
9	Tornillos	Sí	31	Bus de datos IDE 1	Sí
10	Tornillos	Sí	32	Bus de datos IDE 2	Sí
11	Tornillos	Sí	33	Unidad de discos extraíbles, ZIP	No
12	Tornillos	Sí	34	Carcasa auxiliar para unidades de disco	No
13	Tornillos	Sí	35	Tornillos	Sí
14	Caja exterior	No	36	Tornillos	Sí
15	Puerto de comunicaciones COM1	No	37	Tornillos	Sí
16	Puerto de comunicaciones COM2	No	38	Altavoz interno del PC	No
17	Puerto de comunicaciones LPT1	No	39	Carcasa posterior	No
18	Tornillos	Sí	40	Tornillos	Sí
19	Tornillos	Sí	41	Tornillos	Sí
20	Tornillos	Sí	42	Cables	Sí
21	Módulo de memoria, SIMM 1	No	43	Tornillos	Sí
22	Módulo de memoria, SIMM 2	No	44	Tornillos	Sí

De igual manera los montajes que se definen para este producto se detallan en la tabla siguiente, indicando para cada componente a que montaje o montajes pertenece.

M <sup>3</sup>	Descripción	C/M <sup>4</sup>	M	Descripción	C/M
1	PC completo	13	24	Carcasa interior con COM1	1
		14			15
		m <sub>2</sub>			18

<sup>1</sup> C: Número del elemento.

<sup>2</sup> E: ¿Es un elemento de enlace? Sí => Elemento de enlace. No => Componente.

<sup>3</sup> M: Número del montaje.

<sup>4</sup> C/M: Indica el número del elemento que pertenece al montaje (por ejemplo 5, para el elemento 5, "tarjeta SCSI") o el número del montaje al que pertenece (por ejemplo m<sub>2</sub>, para el montaje 2).

M <sup>3</sup>	Descripción	C/M <sup>4</sup>	M	Descripción	C/M
2	Componentes Internos del PC	m <sub>3</sub> m <sub>4</sub> m <sub>5</sub> m <sub>6</sub>	25	Carcasa interior con COM2	1
		m <sub>7</sub> m <sub>8</sub> m <sub>9</sub> m <sub>10</sub>			16
		m <sub>11</sub> m <sub>12</sub> m <sub>16</sub> m <sub>18</sub>			19
		m <sub>20</sub> m <sub>22</sub> m <sub>24</sub> m <sub>25</sub> m <sub>26</sub> m <sub>27</sub> m <sub>28</sub> m <sub>33</sub> m <sub>34</sub> m <sub>35</sub> m <sub>36</sub> m <sub>37</sub> m <sub>38</sub> m <sub>39</sub> m <sub>40</sub> m <sub>41</sub> m <sub>42</sub> m <sub>43</sub> m <sub>44</sub> m <sub>45</sub> m <sub>46</sub>			
3	Placa con SIMM1	2	26	Carcasa interior con LPT1	1
		21			17
4	Placa con SIMM2	2	27	Carcasa interior con la placa base por medio de los cables	20
		22			1
5	Placa con SIMM3	3	28	Carcasa auxiliar con sus componentes adjuntos	2
		23			42
6	Placa con SIMM4	2	29	HD2 con la carcasa	m <sub>29</sub>
		24			m <sub>30</sub>
7	Placa con cache	2	30	ZIP con la carcasa	m <sub>31</sub>
		25			m <sub>32</sub>
8	Fuente de Alimentación con carcasa interior	1	31	Altavoz con carcasa	29
		26			34
9	CD con Carcasa interior	1	32	Carcasa interior con carcasa auxiliar	38
		30			1
10	HD1 con la carcasa	44	33	Placa base con un bus de datos	34
		1			30
11	Altavoz con placa base	28	34	Placa base con un bus de datos	2
		43			32
12	Carcasa interior con tarjeta de video	2	35	CD con el bus de datos	8
		7			m <sub>13</sub>
13	Placa con tarjeta de video	m <sub>14</sub>	36	HD 1 con el bus de datos	31
		2			28
14	Carcasa interior con la placa base	1	37	HD 2 con el bus de datos	31
		41			29
15	Placa base con la placa de la carcasa posterior	m <sub>15</sub>	38	ZIP con el bus de datos	32
		2			33
		39			
		40			

M <sup>3</sup>	Descripción	C/M <sup>4</sup>	M	Descripción	C/M
16	Carcasa interior con la tarjeta de sonido	9	39	CD con la alimentación	26
		m <sub>14</sub>			30
		m <sub>17</sub>			
17	Placa con tarjeta de sonido	2	40	HD 1 con la alimentación	26
		6			28
18	Carcasa interior con tarjeta SCSI	10	41	HD 2 con la alimentación	26
		m <sub>14</sub>			29
		m <sub>19</sub>			
19	Placa con tarjeta SCSI	2	42	ZIP con la alimentación	26
		5			33
20	Carcasa interior con tarjeta de red	11	43	Placa base con el cable de alimentación	2
		m <sub>14</sub>			26
		m <sub>21</sub>			
21	Placa con tarjeta de red	2	44	Placa base con el bus del COM1	2
		4			15
22	Carcasa interior con tarjeta de TV	12	45	Placa base con el bus del COM2	2
		m <sub>14</sub>			16
		m <sub>23</sub>			
23	Placa con tarjeta de TV	2	46	Placa base con el bus del LPT1	2
		3			17

Utilizando el modelo relacional del ordenador planteado, se calcula la secuencia de desensamblado del componente 28, el disco duro. En este caso, el programa devuelve como resultado la siguiente secuencia de acciones a realizar:

1. Como  $c_{28} \in m_{40}$  entonces hay que:
  - a. Eliminar la unión y separar el componente  $c_{28}$ , "disco duro", del componentes  $c_{26}$ , "fuente de alimentación".
2. Como  $c_{28} \in m_{36}$  entonces hay que:
  - a. Quitar  $c_{31}$ , "bus de datos IDE 1".
3. Como  $c_{28} \in m_{10}$  entonces hay que:
  - a. Quitar el componente  $c_{43}$ , "tornillos".
  - a. Separar el componente  $c_1$ , "carcasa interior", del componente  $c_{28}$ , "Disco duro".
2. Como  $m_{10}, m_{36}$  y  $m_{40} \in m_2$  entonces hay que:
  - a. No hacer nada.
3. Como  $m_2 \in m_1$  entonces hay que:
  - a. Quitar el componente  $c_{13}$ , "tornillos".
  - b. Separar el montaje  $m_2$ , "componentes internos del PC", del componente  $c_{14}$ , "caja exterior".

Con lo que se tiene la secuencia de desensamblado del disco duro del ordenador. Además de esta información el programa suministra un gráfico con las tareas a realizar, de manera que resulta fácil visualizar el paralelismo existente entre ellas (Figura 5.8).

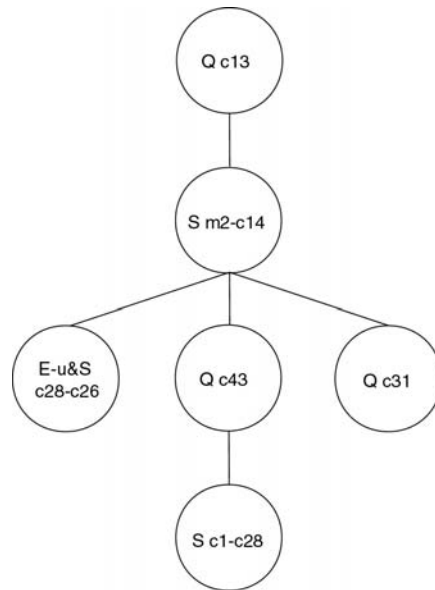


Figura 5.8: Grafo de tareas a realizar para desensamblar el disco duro del PC de la Figura 5.6.

La pantalla que presenta el entorno con la información se muestra en la Figura 5.9.

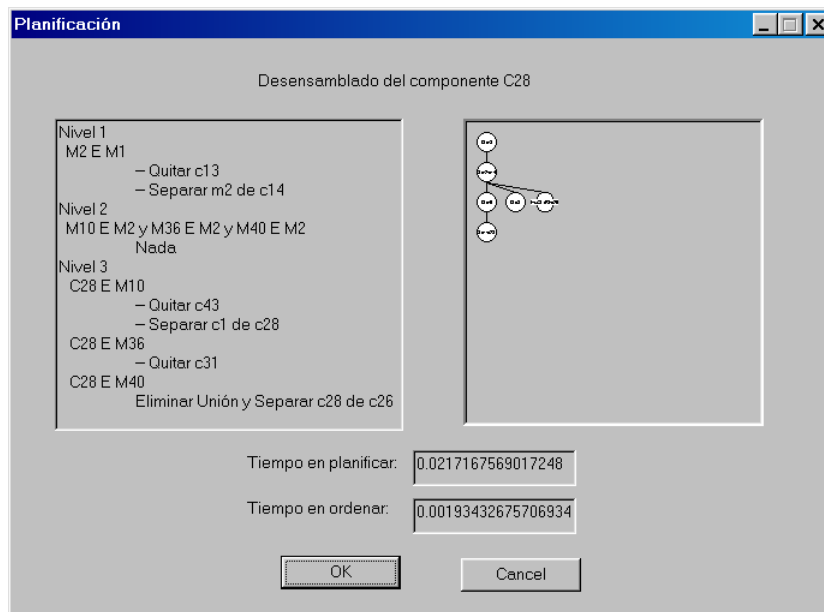


Figura 5.9: Secuencia de desensamblado del disco duro del PC de la Figura 5.6; vista del entorno.

La secuencia de acciones a realizar por un solo brazo robot, para minimizar los cambios de herramientas es:

1. Quitar  $c_{13}$ .
2. Separar  $m_2$  de  $c_{14}$ .
3. Quitar  $c_{43}$ .
4. Separar  $c_1$  de  $c_{28}$ .



5. Quitar  $c_{31}$ .
6. Eliminar unión y separar  $c_{28}$  de  $c_{26}$ .

De igual manera, si en lugar de un elemento del producto se desea desensamblar todo un montaje, se procede de manera similar, salvo que, en lugar de seleccionar el componente en la zona del modelo relacional, se selecciona el montaje en la zona de montajes.

Para ilustrar la secuencia que se obtiene al realizar el desensamblado de un montaje, se va a desensamblar el montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior", donde la secuencia de desensamblado obtenida es:

1. Como  $m_{15} \in m_{13}$  entonces hay que:
  - a. Eliminar unión y separar el componente  $c_7$ , "tarjeta de vídeo", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
2. Como  $m_{15} \in m_{17}$  entonces hay que:
  - a. Eliminar unión y separar el componente  $c_6$ , "tarjeta de sonido", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
3. Como  $m_{15} \in m_{19}$  entonces hay que:
  - a. Eliminar unión y separar el componente  $c_5$ , "tarjeta SCSI" del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
4. Como  $m_{15} \in m_{21}$  entonces hay que:
  - a. Eliminar unión y separar el componente  $c_4$ , "tarjeta de red", del componente  $m_{15}$ , "placa base con la placa de la carcasa posterior".
5. Como  $m_{15} \in m_{23}$  entonces hay que:
  - a. Eliminar unión y separar el componente  $c_3$ , "tarjeta de TV", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
6. Como  $m_{15} \in m_3$  entonces hay que:
  - a. Eliminar unión y separar el componente  $c_{21}$ , "módulo de memoria, SIMM 1", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
7. Como  $m_{15} \in m_4$  entonces hay que:
  - a. Eliminar unión y separar el componente  $c_{22}$ , "módulo de memoria, SIMM 2", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
8. Como  $m_{15} \in m_5$  entonces hay que:
  - a. Eliminar unión y separar el componentes  $c_{23}$ , "módulo de memoria, SIMM 3", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
9. Como  $m_{15} \in m_6$  entonces hay que:

- a. Eliminar unión y separar el componente  $c_{24}$ , "módulo de memoria, SIMM 4", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
10. Como  $m_{15} \in m_7$  entonces hay que:
- a. Eliminar unión y separar el componente  $c_{25}$ , "módulo de memoria, cache", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
11. Como  $m_{15} \in m_{11}$  entonces hay que:
- a. Eliminar unión y separar el montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior", del componente  $c_{38}$ , "altavoz interno del PC".
12. Como  $m_{15} \in m_{14}$  entonces hay que:
- a. Quitar el elemento de enlace  $c_{41}$ , "tornillos".
  - b. Separar el montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior", del componente  $c_1$ , "carcasa exterior".
13. Como  $m_{15} \in m_{27}$  entonces hay que:
- a. Quitar el elemento de enlace  $c_{42}$ , "cables".
  - b. Separar el montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior", del componente  $c_1$ , "carcasa exterior".
14. Como  $m_{15} \in m_{33}$  entonces hay que:
- a. Quitar el elemento de enlace  $c_{31}$ , "bus de datos IDE 1".
15. Como  $m_{15} \in m_{34}$  entonces hay que:
- a. Quitar el elemento de enlace  $c_{32}$ , "bus de datos IDE 2".
16. Como  $m_{15} \in m_{43}$  entonces hay que:
- a. Eliminar unión y separar el componente  $c_{26}$ , "fuente de alimentación", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
17. Como  $m_{15} \in m_{44}$  entonces hay que:
- a. Eliminar unión y separar el componente  $c_{15}$ , "puerto de comunicaciones COM1", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
18. Como  $m_{15} \in m_{45}$  entonces hay que:
- a. Eliminar unión y separar el componente  $c_{16}$ , "puerto de comunicaciones COM2", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
19. Como  $m_{15} \in m_{46}$  hay que:
- a. Eliminar unión y separar el componente  $c_{17}$ , "puerto de comunicaciones LPT1", del montaje  $m_{15}$ , "placa base con la placa de la carcasa posterior".
20. Como  $m_{13}$  y  $m_{14} \in m_{12}$  entonces hay que:

- a. Quitar el elemento de enlace  $c_8$ , "tornillos".
21. Como  $m_{17}$  y  $m_{14} \in m_{16}$  entonces hay que:
- a. Quitar el elemento de enlace  $c_9$ , "tornillos".
22. Como  $m_{19}$  y  $m_{14} \in m_{18}$  entonces hay que:
- a. Quitar el elemento de enlace  $c_{10}$ , "tornillos".
23. Como  $m_{21}$  y  $m_{14} \in m_{20}$  entonces hay que:
- a. Quitar el elemento de enlace  $c_{11}$ , "tornillos".
24. Como  $m_{23}$  y  $m_{14} \in m_{22}$  entonces hay que:
- a. Quitar el elemento de enlace  $c_{12}$ , "tornillos".
25. Como  $m_3, m_4, m_5, m_6, m_7, m_{11}, m_{27}, m_{33}, m_{34}, m_{43}, m_{44}, m_{45}, m_{46}, m_{12}, m_{16}, m_{18}, m_{20}$  y  $m_{22} \in m_2$  entonces hay que:
- a. No hacer nada.
26. Como  $m_2 \in m_1$  entonces hay que:
- a. Quitar el elemento de enlace  $c_{13}$ , "tornillos".
- b. Separar el montaje  $m_2$ , "componentes internos del PC", del componente  $c_{14}$ , "caja exterior".

Con lo que esas serían las acciones a llevar a cabo para poder desensamblar el montaje  $m_{15}$ . Estas acciones se pueden resumir en un grafo tal y como muestra la Figura 5.10.

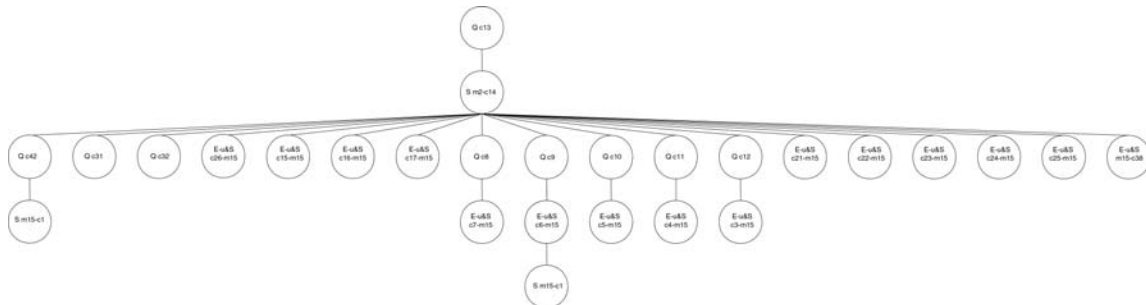


Figura 5.10: Grafo de tareas a realizar para desensamblar el montaje  $m_{15}$  de la Figura 5.6.

De manera que, la pantalla que presenta el entorno para dar la información, es la mostrada en la Figura 5.11.

La secuencia de acciones a realizar por un solo brazo robot, minimizando los cambios de herramientas, es la siguiente:

1. Quitar  $c_{13}$ .
2. Separar  $m_2$  de  $c_{14}$ .
3. Quitar  $c_9$ .
4. Quitar  $c_{10}$ .
5. Quitar  $c_{11}$ .
6. Quitar  $c_{12}$ .
7. Quitar  $c_8$ .

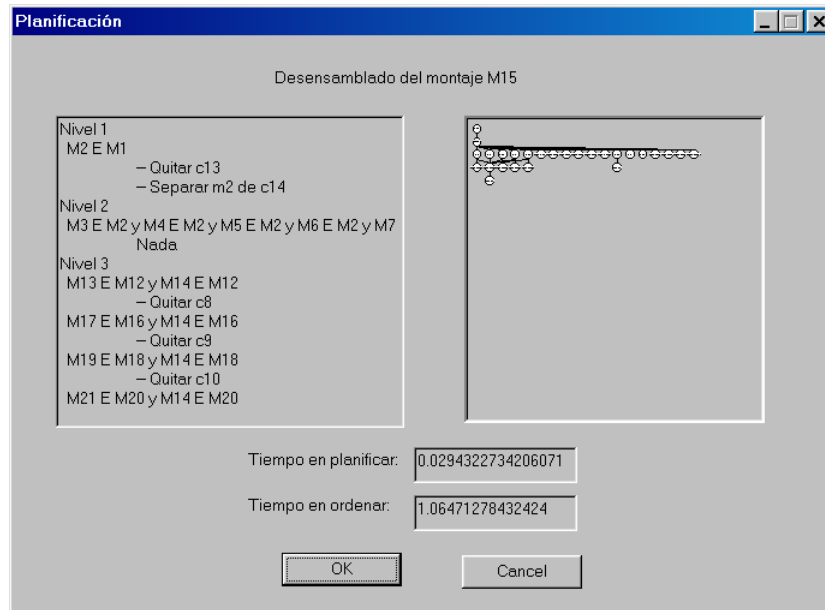


Figura 5.11: Secuencia de desensamblado del montaje  $m_{15}$  del PC de la Figura 5.6, vista del entorno.

8. Quitar  $c_{41}$ .
9. Eliminar unión y separar  $c_6$  de  $m_{15}$ .
10. Eliminar unión y separar  $c_5$  de  $m_{15}$ .
11. Eliminar unión y separar  $c_4$  de  $m_{15}$ .
12. Eliminar unión y separar  $c_3$  de  $m_{15}$ .
13. Eliminar unión y separar  $c_{21}$  de  $m_{15}$ .
14. Eliminar unión y separar  $c_{22}$  de  $m_{15}$ .
15. Eliminar unión y separar  $c_{23}$  de  $m_{15}$ .
16. Eliminar unión y separar  $c_{24}$  de  $m_{15}$ .
17. Eliminar unión y separar  $c_{25}$  de  $m_{15}$ .
18. Quitar  $c_{42}$ .
19. Quitar  $c_{31}$ .
20. Quitar  $c_{32}$ .
21. Eliminar unión y separar  $c_{15}$  de  $m_{15}$ .
22. Eliminar unión y separar  $c_{16}$  de  $m_{15}$ .
23. Eliminar unión y separar  $c_{17}$  de  $m_{15}$ .
24. Eliminar unión y separar  $c_7$  de  $m_{15}$ .
25. Separar  $m_{15}$  de  $c_1$ .
26. Eliminar unión y separar  $c_{26}$  de  $m_{15}$ .
27. Eliminar unión y separar  $m_{15}$  de  $c_{38}$ .

De esta manera se tiene calculada la secuencia de desensamblado de un componente o de un montaje.

### 5.3. Desensamblado de un PC genérico

A la hora de llevar a cabo el desensamblado de un producto el sistema solicita al operario que componente o montaje desea desensamblar, para buscarlos en el modelo, y calcular una secuencia de desensamblado básica. Por ejemplo, en el caso de un PC se podría querer desensamblar la tarjeta de red.

Una vez se conocen los componentes a desensamblar el sistema tiene que reconocer el tipo de producto que se va a desensamblar, esto quiere decir que si el sistema esta buscando ordenadores para quitarles la tarjeta de red, debe ser capaz de distinguir entre un ordenador y otros productos que se presenten al sistema. Además de reconocer los distintos diseños de ordenadores: sobremesa, semitorre, etc. Una vez determinado el tipo de producto y siendo éste el que se busca, se parte de un modelo relacional genérico (Figura 5.12) del mismo, que proporciona información sobre que componentes puede tener el producto y de que manera están distribuidos los montajes, sin entrar en detalles sobre cuantos elementos de cada tipo dispone.

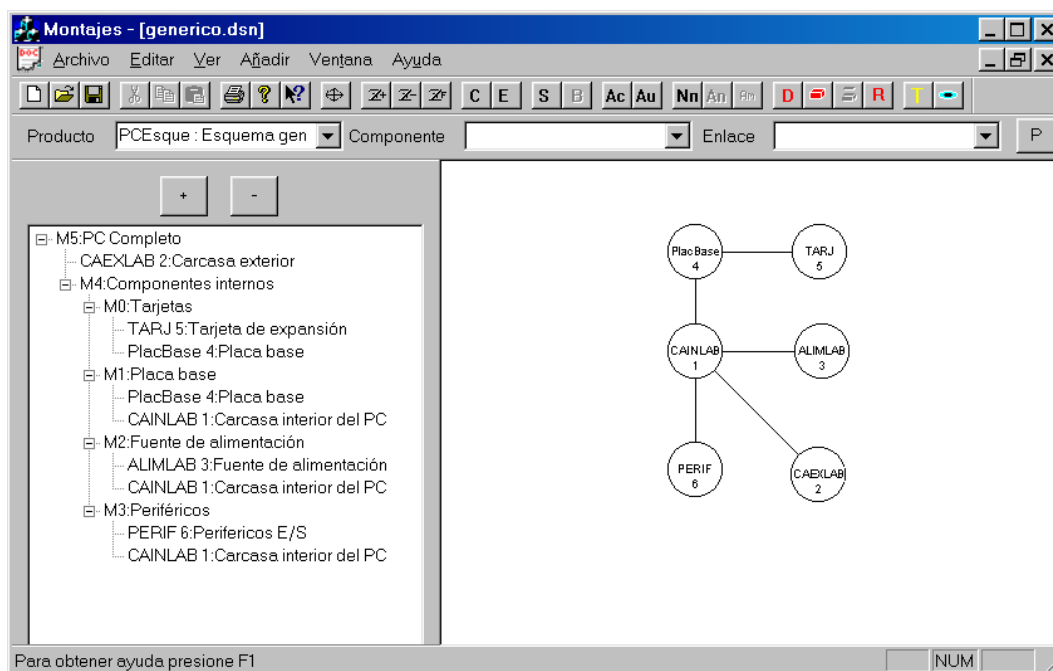


Figura 5.12: Modelo relacional genérico de un PC.

Utilizando la información proveniente del modelo genérico del producto, el planificador de desensamblado determina cual es la secuencia a seguir, solicitando al sistema de visión información sobre la existencia de componentes que puedan existir; y de esta manera conocer su presencia en el producto, tal y como se ha explicado anteriormente.

De esta manera el planificador determina que componente se debe desensamblar. Esta información es utilizada por el módulo encargado de generar los movimientos de desensamblado para el componente. Este módulo genera la secuencia de movimientos a realizar, incluyendo en la misma posibilidades de orientación de la mesa de trabajo, posicionamiento del brazo robot, etc.; es decir, incluye todos los aspectos reales a tener en cuenta para realizar el desensamblado, para generar la trayectoria.

Este módulo solicita al sistema de visión información sobre los componentes del producto para generar un modelo geométrico preciso acerca de cada uno de los componentes. La información que solicita al sistema de visión, depende del modelo que posea el planificador de desensamblado, es decir de los componentes que se sepa que existen en el producto.

Una vez se tiene calculada la secuencia de movimientos a llevar a cabo por el brazo robot para desensamblar el componente se generan las órdenes correspondientes al brazo robot para que se posicione en los lugares indicados para realizarlos, consiguiendo así el desensamblado del componente.

Además se incluye en el sistema información sobre la posición inicial del robot y de la posición final, de manera que se considera en los cálculos, un movimiento del robot desde la posición inicial hasta el punto de agarre del componente; y también desde el punto en donde el componente se esta desensamblado hasta el almacén correspondiente para depositarlo.

En el caso de un PC y basándose en el modelo relacional genérico de la Figura 5.12 se obtiene que para desensamblar el componente "disquetera 3,5" hay que expandir el componente "periféricos". Como este componente es visible desde el montaje actual, el "PC completo", se reconoce en este punto la existencia de la disquetera; con lo que se expande el modelo relacional tal y como muestra la Figura 5.13 suponiendo que se ha encontrado la disquetera.

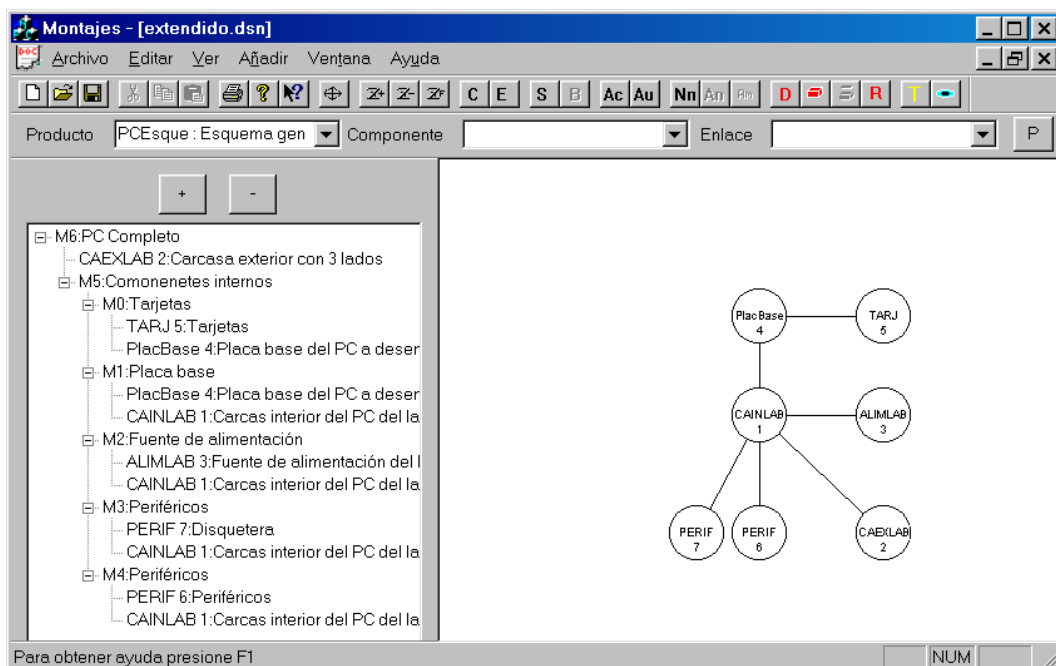


Figura 5.13: Modelo relacional extendido de un PC con una disquetera.

Basándose en esta información se determina la secuencia de acciones a seguir, que en este caso se corresponde con el desensamblado del componente "carcasa exterior" y posteriormente el del componente "disquetera".

Utilizando la información existente en el sistema se generan los movimientos de desensamblado para el componente "carcasa exterior" y se llevan a cabo por el brazo robot. Tras esto, el siguiente componente a desensamblar es la "disquetera", pero antes de seguir, ya se dispone de la

secuencia de desensamblado para este componente, hay que expandir aquellos componentes que tengan la opción de existir al mismo nivel que el componente "periféricos", esto es debido a que si no se expandieran estos componentes el modelo geométrico del producto sería incompleto y podría llevar a errores en la generación de los movimientos con las posibles colisiones que ello provocaría. Por lo tanto se expande el componente "tarjetas", obteniendo la existencia de una "tarjeta de vídeo". También se tiene que reconocer la existencia de la fuente de alimentación. Además el componente "periféricos" no se expandió totalmente en la iteración anterior del algoritmo, por lo que se debe buscar la existencia de nuevos componentes que pertenezcan a éste; con lo que se encuentra en este caso la existencia de un "disco duro" (Figura 5.14).

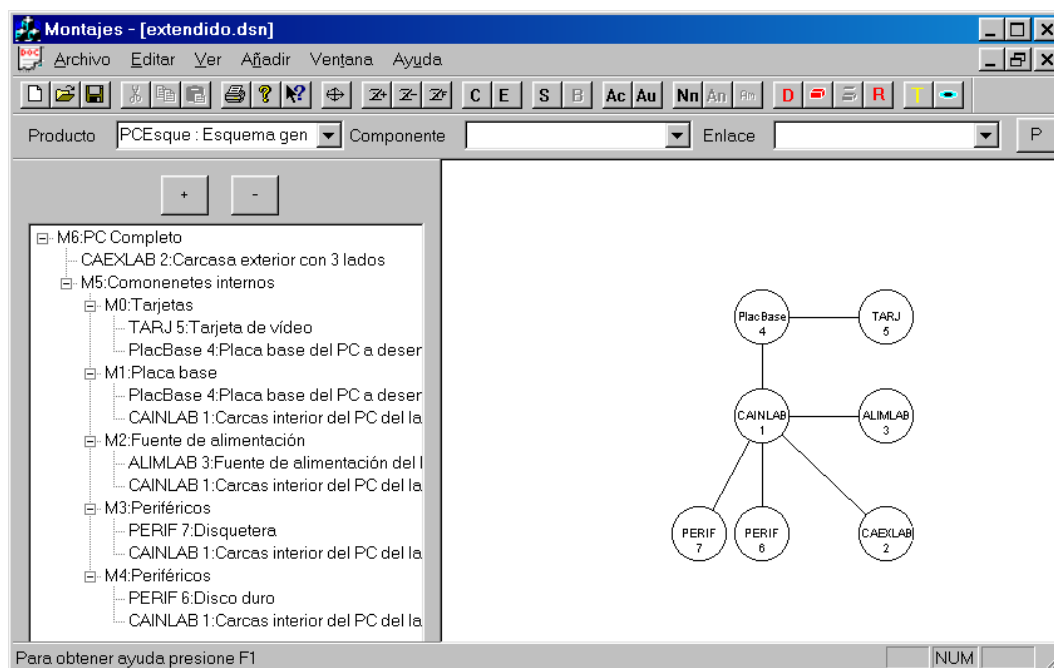


Figura 5.14: Modelo relacional extendido de un PC; una tarjeta, dos periféricos y fuente de alimentación.

Con esta información el generador de movimientos de separación solicita al sistema de visión una localización precisa de cada uno de los componentes, para actualizar el modelo geométrico, y una corrección de los tamaños si fuera necesario, por ejemplo la longitud de las tarjetas.

Tras realizar esta actualización se generan los movimientos de separación del componente "disquetera" y se realizan por el robot real, con lo que se ha alcanzado el objetivo de desensamblado buscado.

## 5.4. Arquitectura del sistema de desensamblado flexible

En este apartado se describe el sistema de desensamblado flexible disponible en el laboratorio para la realización de las pruebas de desensamblado así como los valores de los parámetros necesarios para llevar a cabo el proceso de desensamblado.

La arquitectura utilizada para el desensamblado está formada por un robot Scorbobot ER-IX de cinco grados de libertad, con una pinza de dedos paralelos como herramienta. También se dispone de una mesa de trabajo giratoria; esta mesa permite orientar el producto dando al sistema un grado de libertad extra. Por último el sistema utiliza, como sistema sensorial, un par estereoscópico montado sobre un robot cartesiano y-z para reconocer y localizar los componentes del producto (Figura 5.15).

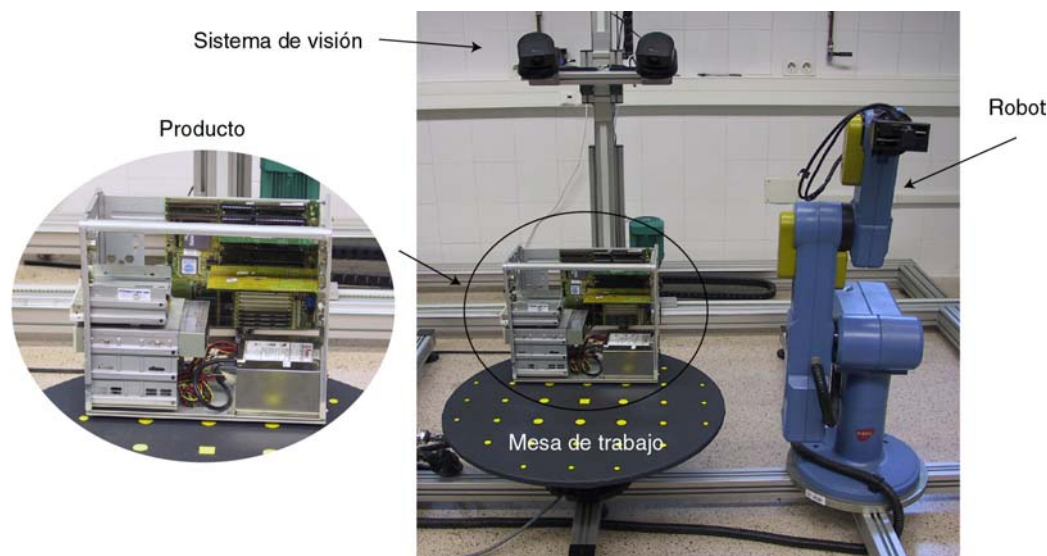


Figura 5.15: Arquitectura utilizada para el desensamblado.

El modelo cinemático del robot se presenta en el Anexo B, donde se ha resuelto en función de los valores articulares el valor de la posición del extremo y viceversa.

En la configuración utilizada, el centro de la mesa de trabajo se encuentra a una distancia de  $-700$  mm según el eje y del robot. Además, para evitar los problemas que se presentan con el robot, debido a su reducida área de trabajo, se permite realizar desplazamientos del producto sobre la mesa de trabajo, tal como se explicó en el capítulo anterior.

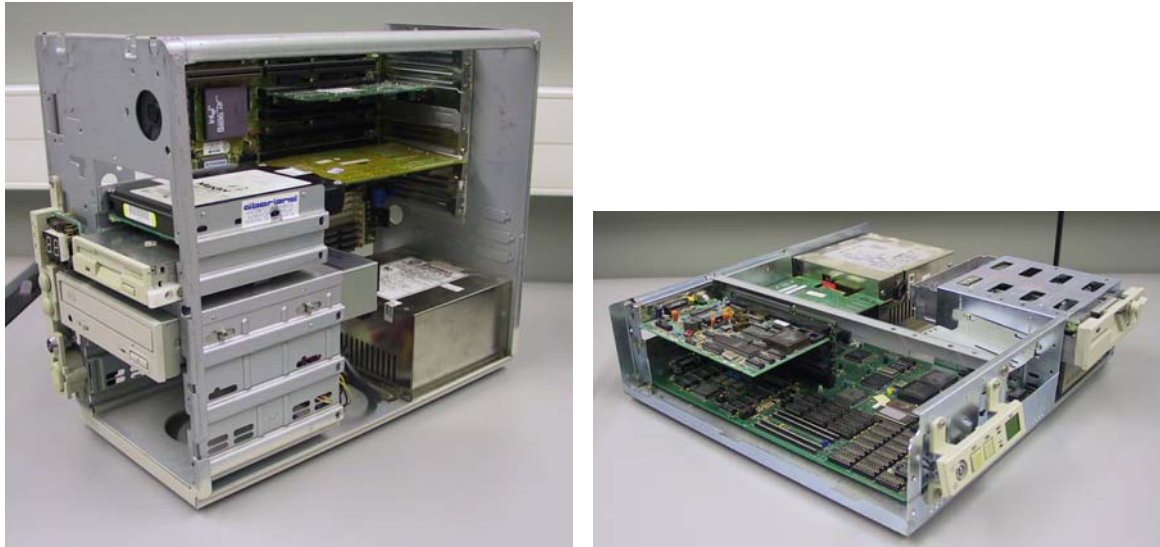
El desarrollo correspondiente al cálculo de la nueva matriz de transformación así como de las ecuaciones que definen los valores de giro y de desplazamiento del producto sobre la mesa de trabajo se encuentran desarrollados en el Anexo C.

Los desplazamientos calculados se tienen que llevar a cabo manualmente, debido a que la mesa no permite realizarlos de manera automática. El rango que se ha fijado para los desplazamientos es de  $\pm 100$  mm según los ejes x e y.

## 5.5. Desensamblado de PC's

En este apartado se va a realizar el desensamblado mediante el sistema flexible de desensamblado explicado y utilizando todos los conceptos vistos en esta tesis para desensamblar componentes de dos modelos distintos de PC, uno de tipo semitorre y otro sobremesa (Figura 5.16).





*Figura 5.16: Tipos de PC: semitorre y sobremesa.*

Para ver el funcionamiento del sistema, se va a realizar el desensamblado de la "tarjeta de vídeo" y del "disco duro" de los ordenadores mostrados en la Figura 5.16. Dado que ambos componentes pertenecen a montajes distintos, no forman un montaje entre ellos; el sistema busca desensamblar primero uno, y después, en base a los componentes que queden en el producto, calcula la secuencia para desensamblar el otro.

Para realizar los cálculos, se parte del PC con su "carcasa exterior", pero el robot utilizado no puede realizar su desensamblado, debido al área de trabajo necesaria para llevarlo a cabo. Por lo tanto se elimina, del cálculo de los movimientos de desensamblado para dicho componente, el brazo robot, y en consecuencia, su desensamblado se realiza de manera manual.

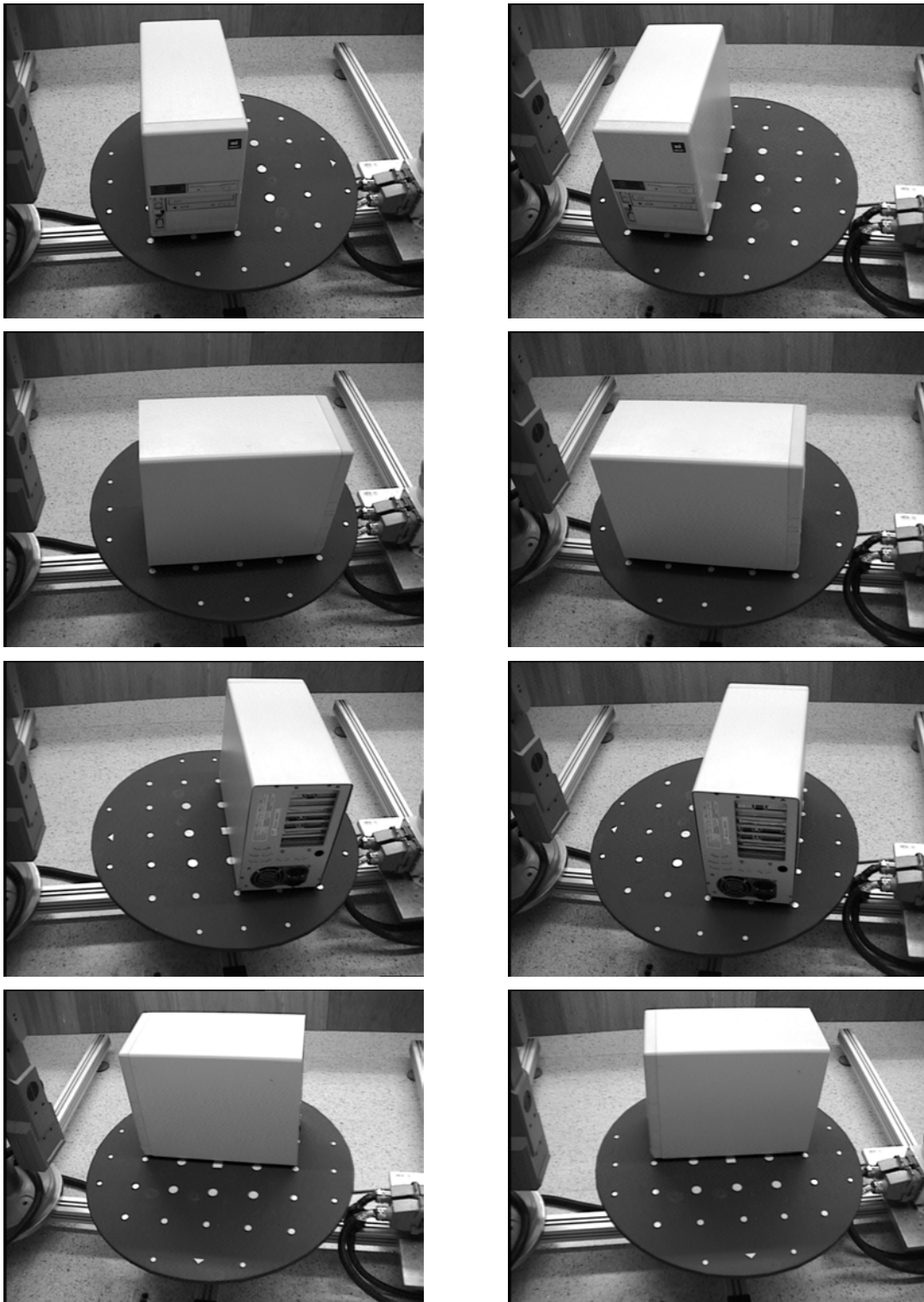
De igual manera aquellos componentes que requieren una herramienta especial, distinta de una pinza paralela de dos dedos, se muestran en simulación como se realizaría el desensamblado pero en el sistema real debido a la falta de dichas herramientas su desensamblado se realiza de manera manual.

De cara a continuar con el desensamblado se va a proceder a realizar primero el correspondiente al PC semitorre y posteriormente el del PC sobremesa.

#### 5.5.1. Desensamblado de un PC semitorre

Una vez se tiene el PC sobre la mesa de trabajo, lo primero que tiene que realizar el sistema, es reconocer el tipo de producto. Para ello el sistema de visión captura varias vistas del producto, utilizando la mesa de trabajo para orientarlo (Figura 5.17) y así poder reconocer el producto de que se trata.

El sistema de visión utiliza cuatro vistas del producto, cada una de ellas girada 90° respecto de la anterior, como información para el reconocimiento del producto.



*Figura 5.17: Sistema de visión adquiriendo información sobre el tipo de producto; imágenes de la cámara izquierda y derecha del par estéreo.*

Una vez que el sistema ha reconocido el producto como un PC semitorre, se selecciona el modelo relacional genérico del producto, que es el mostrado en la Figura 5.18, el mismo que aparecía en la Figura 5.12.

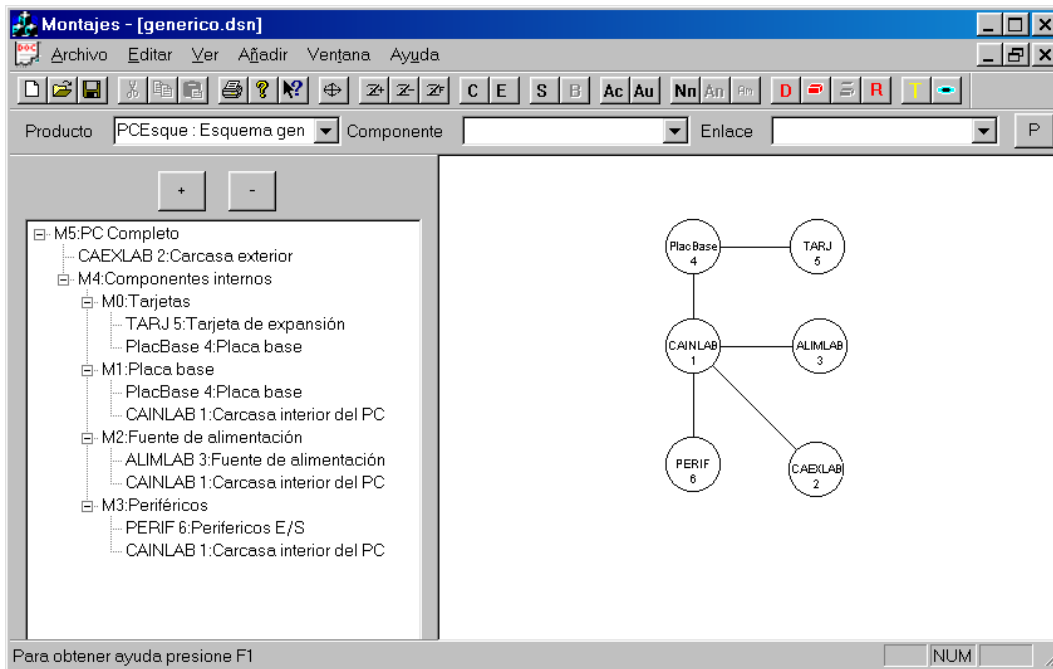


Figura 5.18: Modelo relacional genérico de un PC.

Una vez se ha seleccionado el modelo, el planificador global del sistema decide que el componente "tarjetas", se puede expandir en este nivel del modelo; también se pueden expandir algunos "periféricos" (disquetera, cd), e incluso se puede saber si existe el componente "fuente de alimentación". Utilizando esta información y debido a que, lo que se busca es desensamblar el componente "tarjeta de vídeo" y el "disco duro", el planificador global decide que la existencia de la "tarjeta de vídeo" se puede resolver en este nivel, con lo que, si no existe, no es necesario seguir el proceso de desensamblado. Por otro lado, la existencia de un "disco duro" no puede ser conocida en este nivel de desensamblado, lo que implica la necesidad de desensamblar los componentes correspondientes, hasta llegar a un nivel en el que el "disco duro" sea reconocible, hasta que el componente "periféricos" se pueda expandir para obtener dicha información.

El planificador global decide, en este momento, expandir el componente "tarjetas", con lo que obtiene información sobre la existencia de dos tarjetas, una "tarjeta de vídeo" y una "tarjeta de red". Por lo tanto, el nuevo modelo relacional queda tal y como muestra la Figura 5.19.

Por otra parte, el modelo geométrico del producto no se puede extender con información sobre las tarjetas, ya que se conoce su existencia pero no sus dimensiones exactas, con lo que esta información tendrá que adquirirse en una iteración posterior del algoritmo.

Según este modelo, el planificador busca la secuencia de acciones necesarias para desensamblar el componente "tarjeta de vídeo". La secuencia obtenida viene definida por las siguientes acciones:

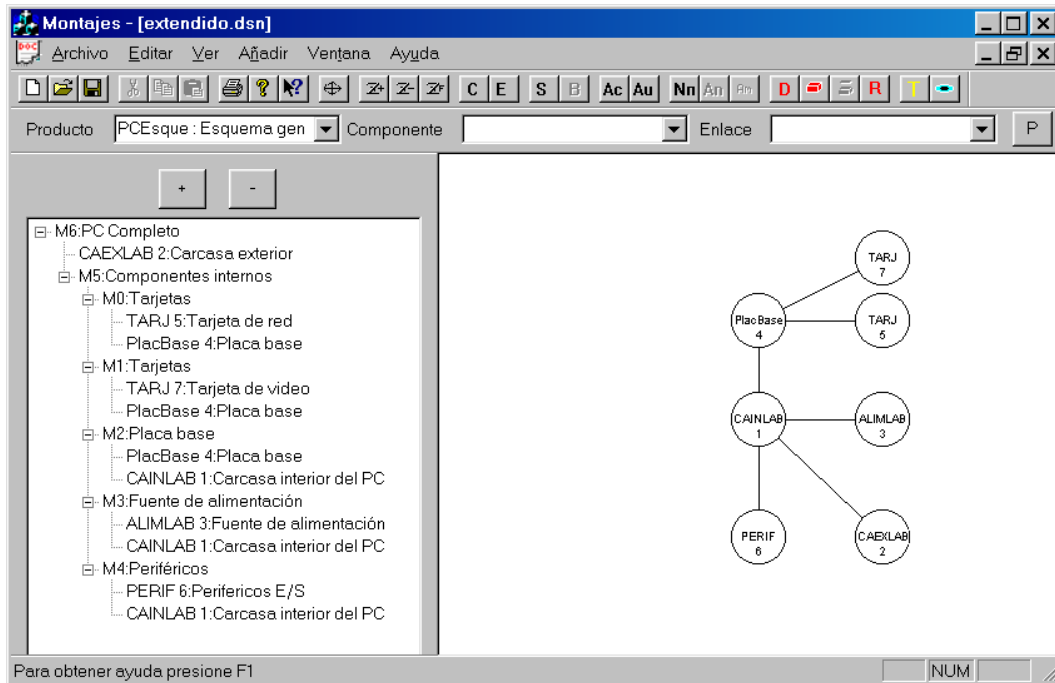


Figura 5.19: Modelo relacional extendido del PC semitorre con las tarjetas.

- 1) Como el componente "tarjeta de vídeo" pertenece al montaje  $m_1$  hay que:
  - a. Eliminar la unión entre los componentes "placa base" y "tarjeta de vídeo".
  - b. Separar los dos componentes.
  - c. Como el montaje  $m_1$  pertenece al montaje  $m_5$  hay que seguir con las acciones de desensamblado para el montaje  $m_5$ .
- 2) Las acciones asociadas con el montaje  $m_5$  son:
  - a. No hacer nada.
  - b. Como el montajes  $m_5$  pertenece al montaje  $m_6$  hay que seguir con las acciones asociadas este montaje.
- 3) Las acciones asociadas con  $m_6$  son:
  - a. Eliminar la unión entre el componente "carcasa exterior" y el montaje  $m_5$ .
  - b. Separar el componente "carcasa exterior" del montaje  $m_5$ ; "componentes internos".
  - c. Como ya se ha llegado al montaje que representa el producto completo, para obtener la secuencia de desensamblado hay que seguir las acciones obtenidas en orden inverso.

Esta serie de acciones se expresan en forma de grafo dirigido de tareas a realizar, tal como muestra la Figura 5.20.

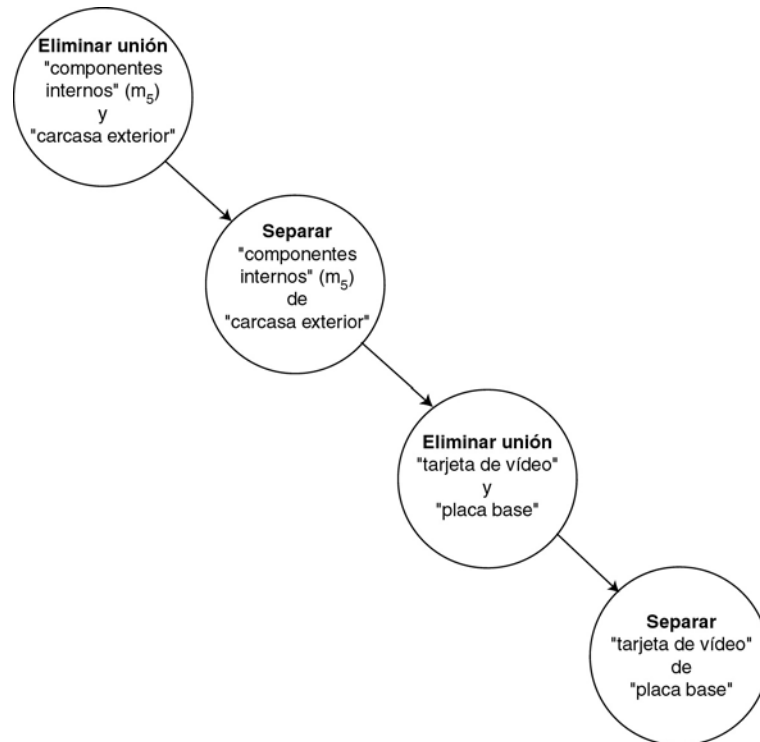


Figura 5.20: Grafo de acciones a llevar a cabo por el sistema para desensamblar el componente "tarjeta de vídeo".

Esta secuencia se obtiene en el programa mediante la ventana de información de la Figura 5.21.

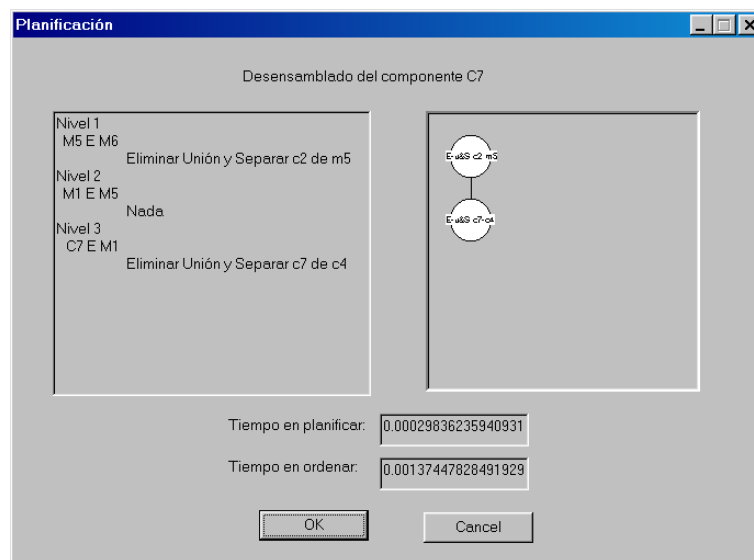


Figura 5.21: Acciones a llevar a cabo por el sistema para desensamblar el componente "tarjeta de vídeo".

En base a esta secuencia, el planificador de desensamblado decide eliminar la "carcasa exterior" del PC, por lo que, esta información se comunica al generador de movimientos, para que éste genere la trayectoria de movimientos necesarios para desensamblar el componente.

Teniendo en cuenta esta información, se calcula la trayectoria de movimientos de desensamblado, que como se ha dicho anteriormente, omite la existencia de un brazo robot. La trayectoria de movimientos obtenida consiste en realizar una traslación a lo largo del eje x del componente "carcasa exterior" (Figura 5.22).

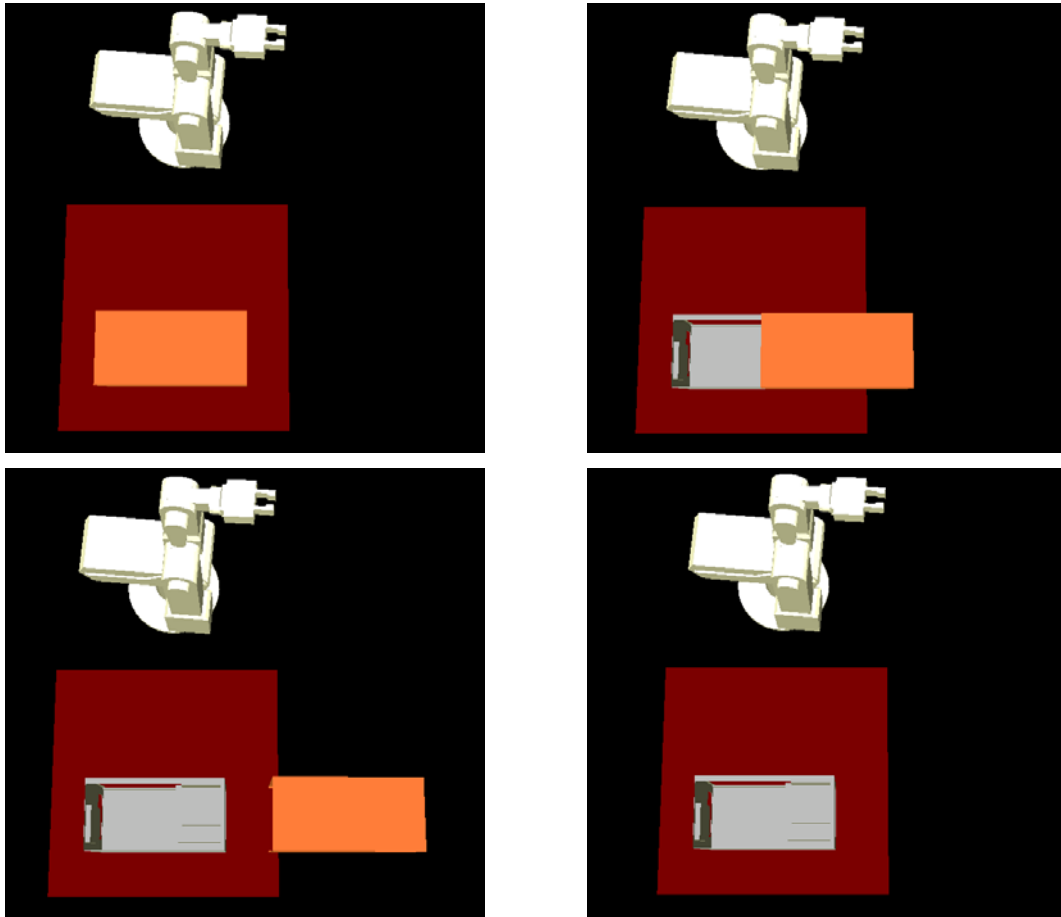
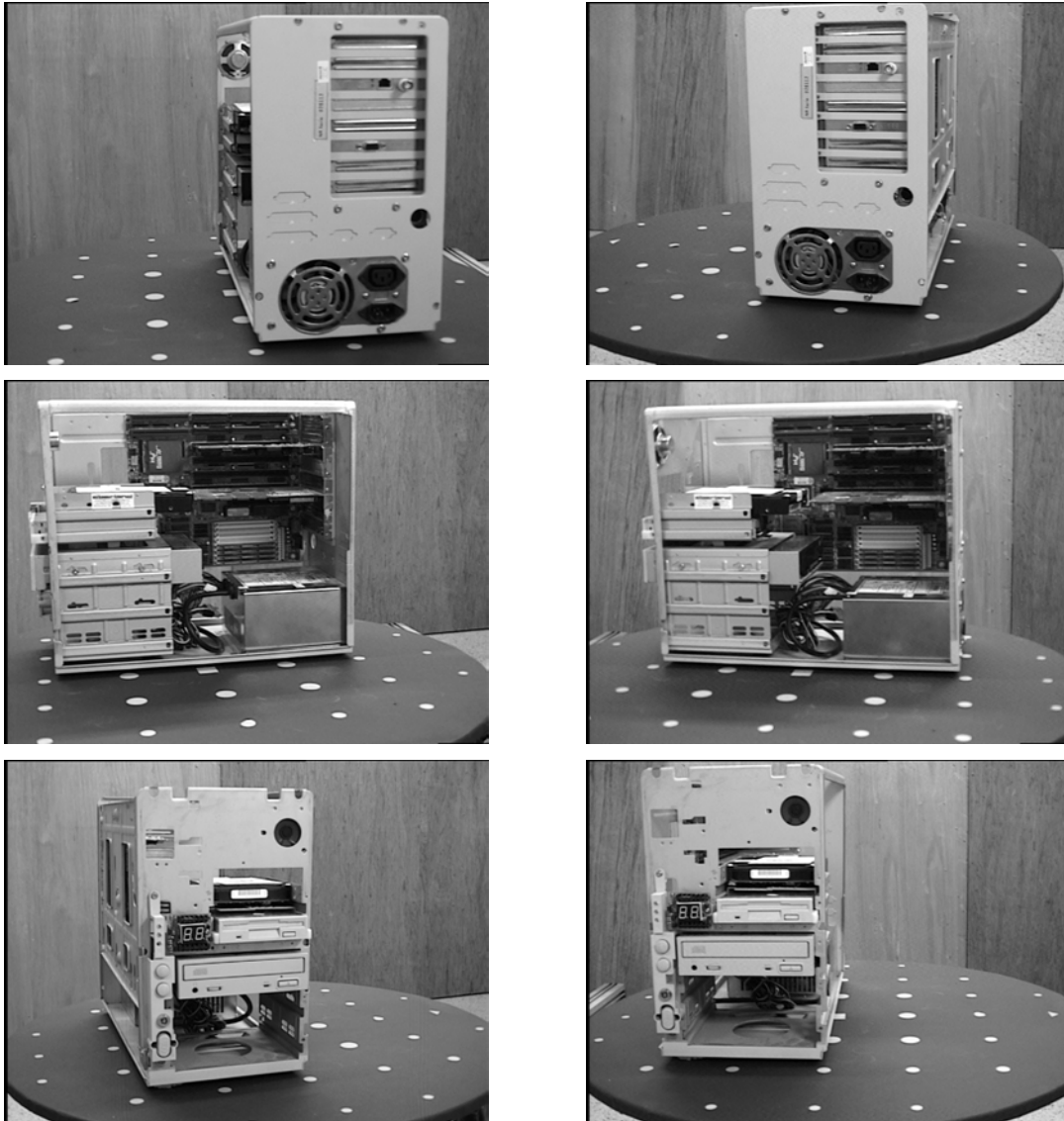


Figura 5.22: Secuencia simulada de desensamblado para la carcasa exterior de un PC semitorre.

Una vez desensamblada la carcasa exterior, el planificador global tiene la posibilidad de expandir y obtener mayor información sobre los componentes "tarjeta de vídeo", "tarjeta de red", "periféricos", "fuente de alimentación" y "placa base". En este caso el planificador decide buscar la existencia de "periféricos", de la "fuente de alimentación" y de la "placa base", así como solicitar al sistema de visión información más precisa sobre los componentes "tarjeta de vídeo" y "tarjeta de red". Por lo tanto se realiza una nueva captura de imágenes del producto rotándolo sobre la mesa de trabajo (Figura 5.23).

En este caso, el sistema de visión reconoce la existencia de tres componentes dentro de la categoría de "periféricos", que son un "disco duro", una "disquetera de 3,5" y una unidad de "CD". Además de estos componentes en la categoría de "periféricos", el sistema reconoce la existencia de la "fuente de alimentación" y de la "placa base" (Figura 5.24).

Con esta información, el planificador global actualiza el modelo relacional del producto, considerando los nuevos componentes y sin incluir los componentes ya desensamblados del producto.



*Figura 5.23: Sistema de visión adquiriendo información sobre el tipo de producto, PC semitorre; imágenes de la cámara izquierda y derecha del par estéreo.*

En la Figura 5.25 se muestra el modelo relacional correspondiente a la introducción de los nuevos componentes.

Por otro lado, además de modificar el modelo relacional, se tiene que modificar el modelo geométrico del producto, debido a la existencia de los nuevos componentes e información más precisa de los componentes "tarjetas de vídeo" y "tarjeta de red". Por lo tanto utilizando la información proveniente del sistema de visión, se actualizan los valores de posición, orientación y tamaño de los componentes, quedando el modelo geométrico del producto mostrado en la Figura 5.26.

Con esta información, el planificador de desensamblado tiene que calcular la nueva secuencia de desensamblado del componente "tarjeta de vídeo".



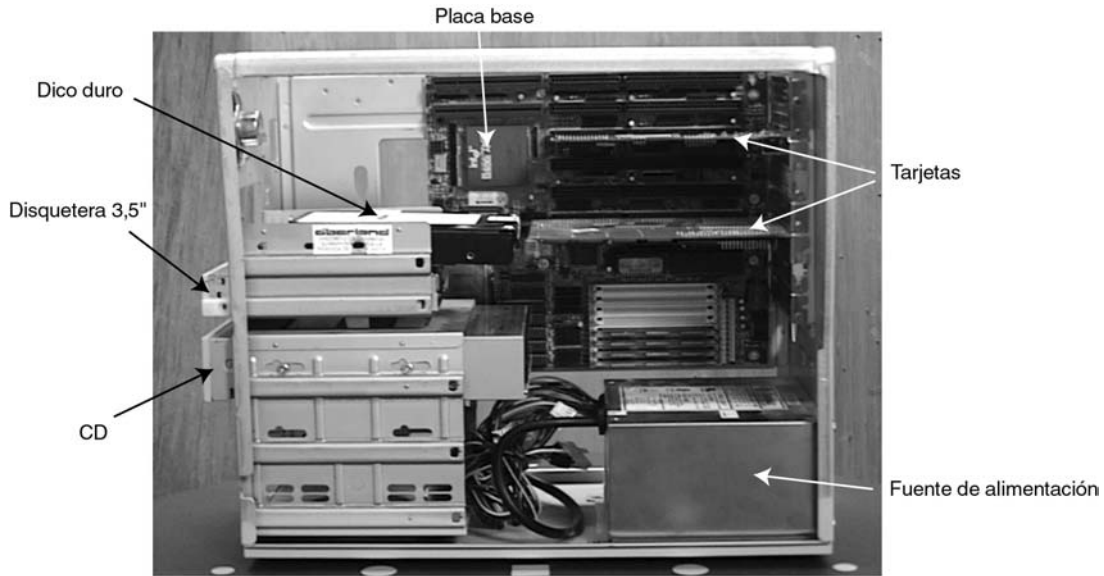


Figura 5.24: PC semitorre con los componentes internos visibles.

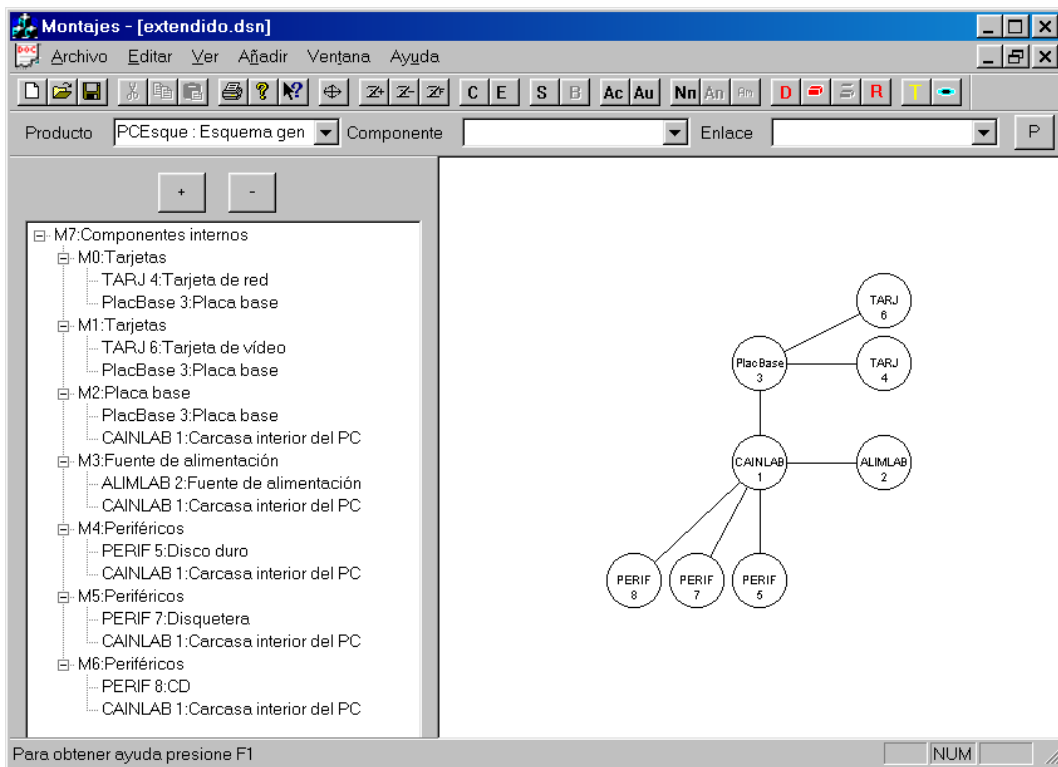


Figura 5.25: Modelo relacional extendido del PC semitorre.

Para ello, el planificador global no requiere ejecutar otra vez el planificador de desensamblado, debido a que los montajes y los componentes involucrados en el desensamblado del componente "tarjeta de vídeo", no han modificado las relaciones existentes entre ellos.

Por lo tanto, el componente a desensamblar es el correspondiente a la "tarjeta de vídeo".



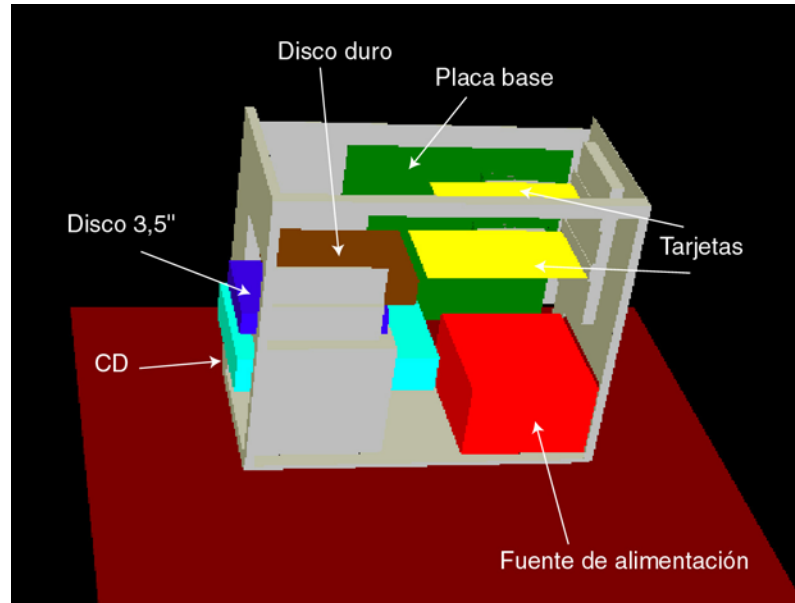


Figura 5.26: Modelo geométrico extendido del PC semitorre con los componentes visibles.

Esta información se comunica al generador de movimientos de desensamblado, por parte del planificador de desensamblado.

Con esta información, el generador de movimientos, calcula la trayectoria del componente a desensamblar, la "tarjeta de vídeo". Para ello, considera en el modelo geométrico, tanto los componentes del producto como el modelo geométrico del robot. Obteniendo la secuencia de movimientos lineales definida por los vectores siguientes:

$$\mathbf{d}_1 = (0, 30, 0) \quad (5.1)$$

$$\mathbf{d}_2 = (28.284, 20.8284, 0) \quad (5.2)$$

$$\mathbf{d}_3 = (0, 160, 0) \quad (5.3)$$

De tal manera, que realizando las acciones definidas por las ecuaciones 5.1, 5.2 y 5.3, se tiene desensamblado el componente "tarjeta de vídeo".

La simulación de la trayectoria de los movimientos de desensamblado del componente "tarjeta de vídeo" obtenida, se muestra en la Figura 5.27.

Una vez simulado el desensamblado del componente "tarjeta de vídeo", las acciones correspondientes deben ser llevadas a cabo por el robot, antes de proseguir con el desensamblado del componente "disco duro". Esta secuencia se muestra en la Figura 5.28, donde se puede apreciar el brazo robot realizando los movimientos correspondientes a la trayectoria de desensamblado.

Los valores articulares del robot, en los puntos de control, para realizar los movimientos, se detallan a continuación, incluyendo como puntos de control el comienzo de la trayectoria y la posición donde depositar el componente una vez desensamblado.

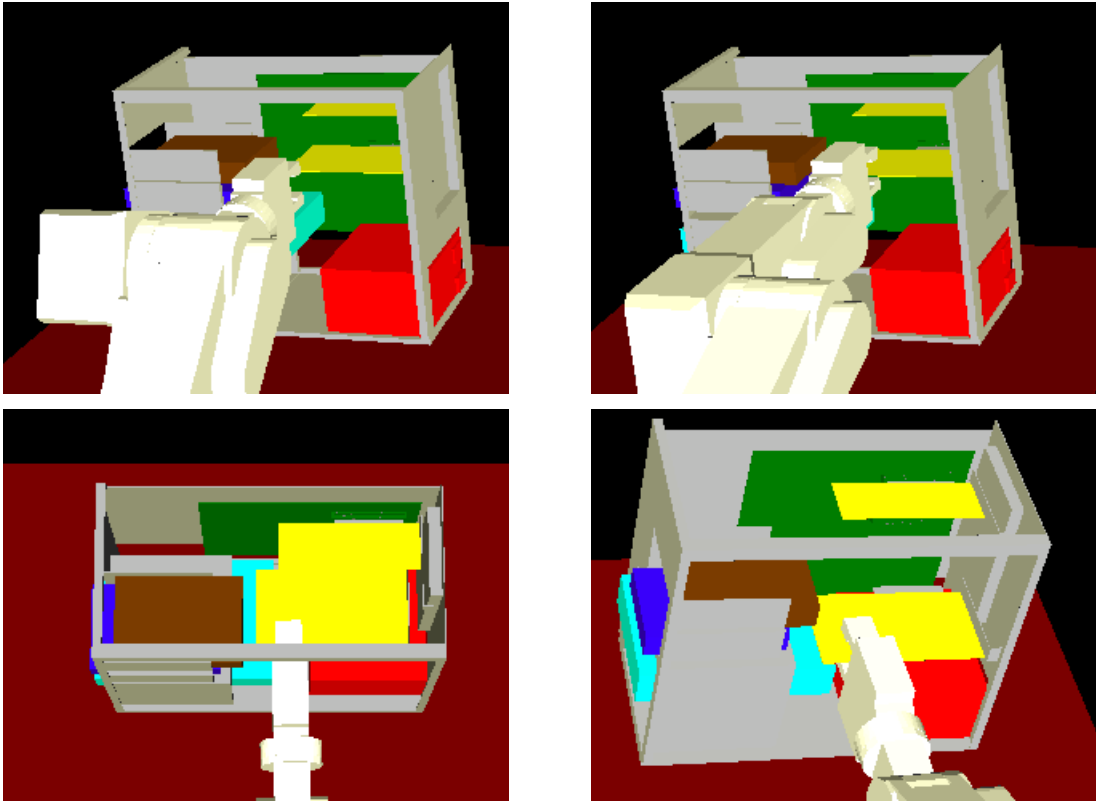


Figura 5.27: Secuencia simulada de la trayectoria de desensamblado del componente "tarjeta de vídeo" desde distintos puntos de vista.

Los puntos de control vienen dados por los siguientes valores articulares:

$$d_1 \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (89039, 28702, -35097, 125462, 52387) \quad (5.4)$$

$$d_2 \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (86891, 21064, -49476, 131955, 52388) \quad (5.5)$$

$$d_3 \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (86890, -15214, -115748, 162928, 52387) \quad (5.6)$$

Para realizar los movimientos de desensamblado, se parte del punto definido por los valores articulares siguientes:

$$\text{origen} \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (89039, 53203, 11462, 104266, 52388) \quad (5.7)$$

Y el componente desensamblado se deposita en el almacén situado en la posición articular:

$$\text{almacén} \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (-114877, -34027, -115748, 148878, 52383) \quad (5.8)$$

Una vez desensamblado el componente "tarjeta de vídeo", el modelo relacional del PC queda reducido, eliminando dicho componente y los montajes que implica, tal y como se muestra en la Figura 5.29.

Por lo tanto, ahora el planificador global decide: que se debe continuar solicitando al planificador de desensamblado que genere la secuencia de desensamblado para el componente "disco duro", según el nuevo modelo relacional.

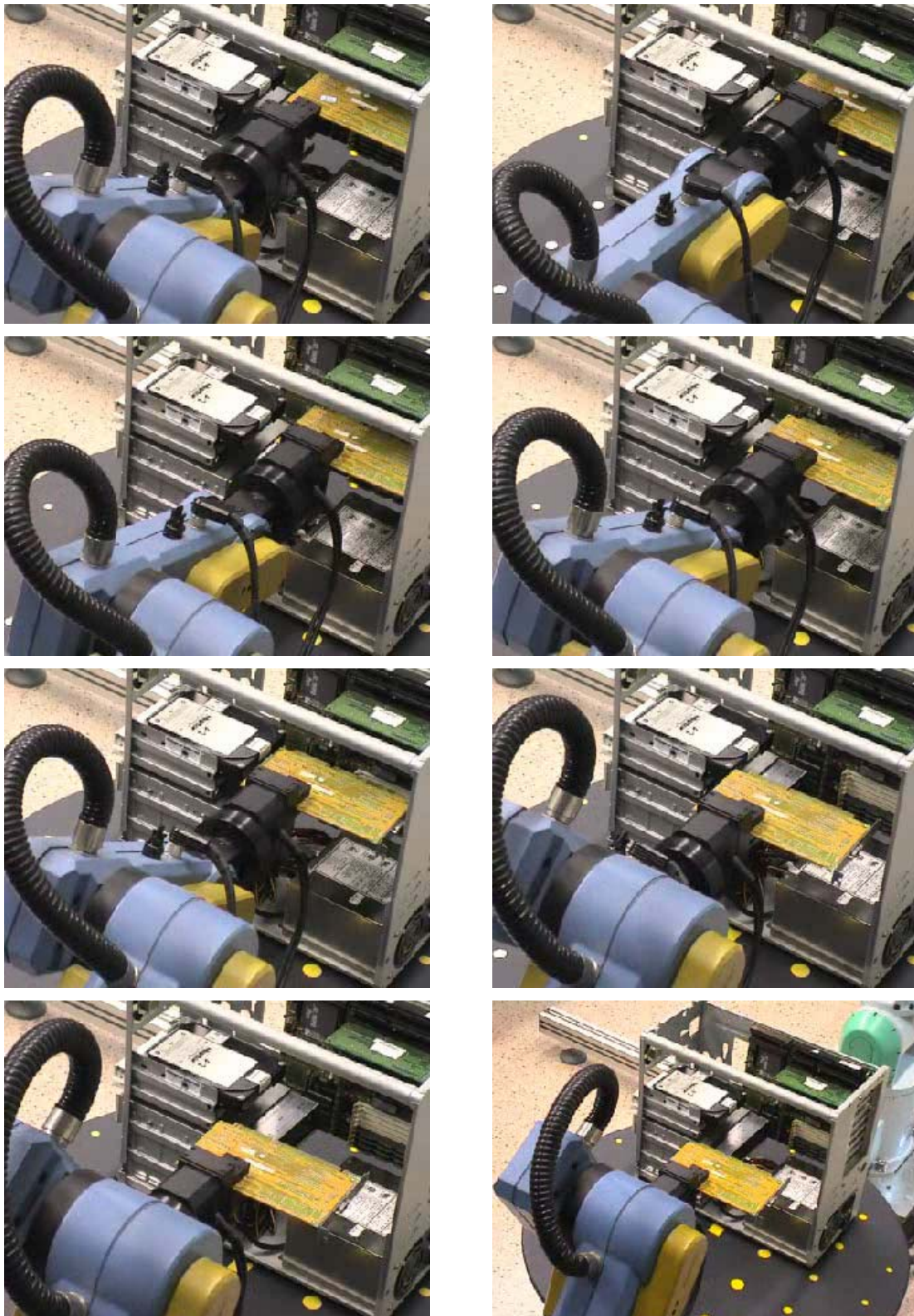


Figura 5.28: Secuencia real de desensamblado del componente "tarjeta de vídeo".

Según esta información, el planificador de desensamblado genera la siguiente secuencia de acciones para desensamblar el componente "disco duro":

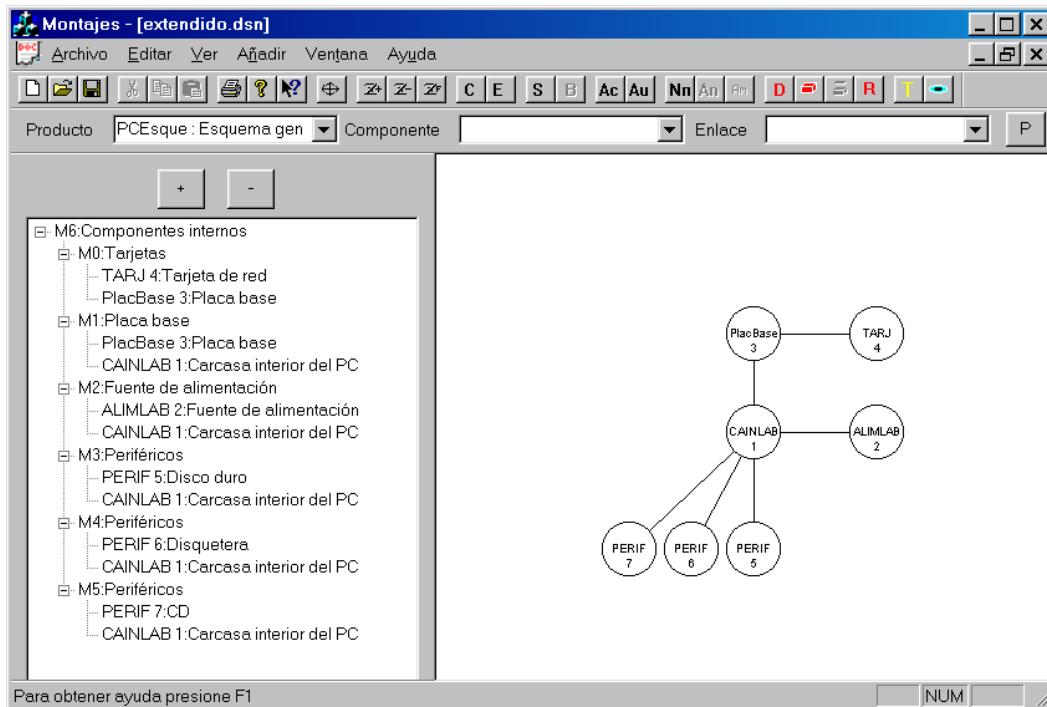


Figura 5.29: Modelo relacional del PC una vez desensamblado el componente "tarjeta de vídeo".

- 1) Como el componente "disco duro" pertenece al montaje  $m_3$  hay que:
  - a. Eliminar la unión entre los componentes "carcasa interior" y "disco duro".
  - b. Separar los dos componentes.
  - c. Como que el montaje  $m_3$  pertenece al montaje  $m_6$  hay que seguir con las acciones de desensamblado para el montaje  $m_6$ .
- 2) Las acciones asociadas con el montaje  $m_6$  son:
  - a. No hacer nada.
  - b. Como el montaje  $m_6$  es el montaje final, el que representa a todos los componentes existentes en estos momentos en el producto, ya se tiene la secuencia de acciones a realizar, para desensamblar el componente "disco duro".

La secuencia de acciones a seguir se muestra en el grafo de la Figura 5.30.

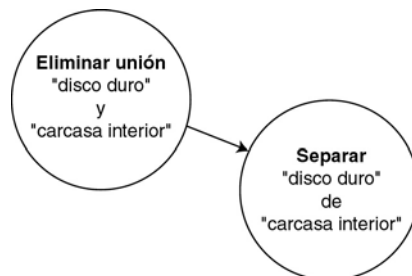


Figura 5.30: Grafo de acciones para desensamblar el componente "disco duro".

Esta información es la misma que nos proporciona el sistema mediante la ventana de información correspondiente (Figura 5.31).

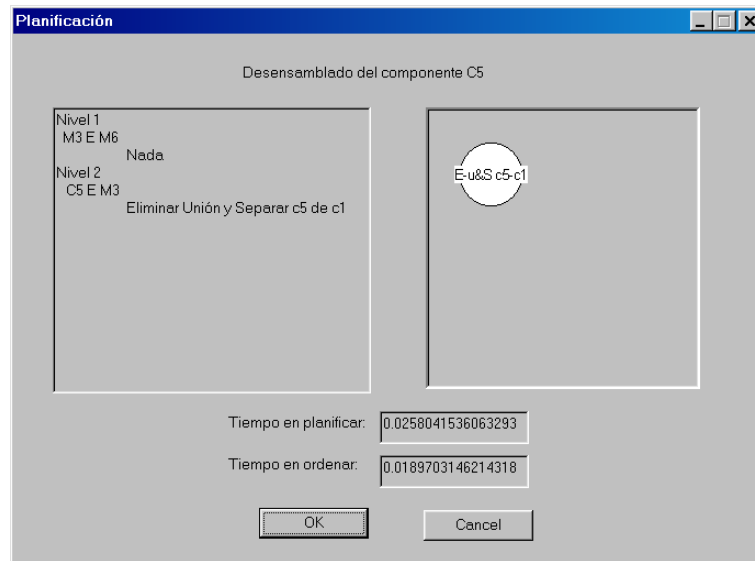


Figura 5.31: Acciones a llevar a cabo por el sistema para desensamblar el componente "disco duro".

Esta información, obtenida por el planificador de desensamblado, se comunica al generador de movimientos, para que, en función del modelo geométrico del PC y del modelo del robot, calcule la secuencia de movimientos necesarios para desensamblar el componente "disco duro".

La trayectoria obtenida viene definida por los siguientes vectores:

$$\mathbf{d}_1 = (0, 0, 60) \quad (5.9)$$

$$\mathbf{d}_2 = (0, 180, 0) \quad (5.10)$$

El sistema, además de estos desplazamientos, informa que es necesario un giro de la mesa de trabajo de  $5^\circ$ , con lo que será necesario realizar primero el giro, antes de poder trasladar el componente según las direcciones, teniendo en cuenta que también sufren el giro. Con lo que las nuevas direcciones a seguir por el robot vienen definidas por:

$$\mathbf{d}_1 = (0, 0, 60) \quad (5.11)$$

$$\mathbf{d}_2 = (-15.688, 179.315, 0) \quad (5.12)$$

La trayectoria obtenida con estos valores se simula tal y como muestra la Figura 5.32.



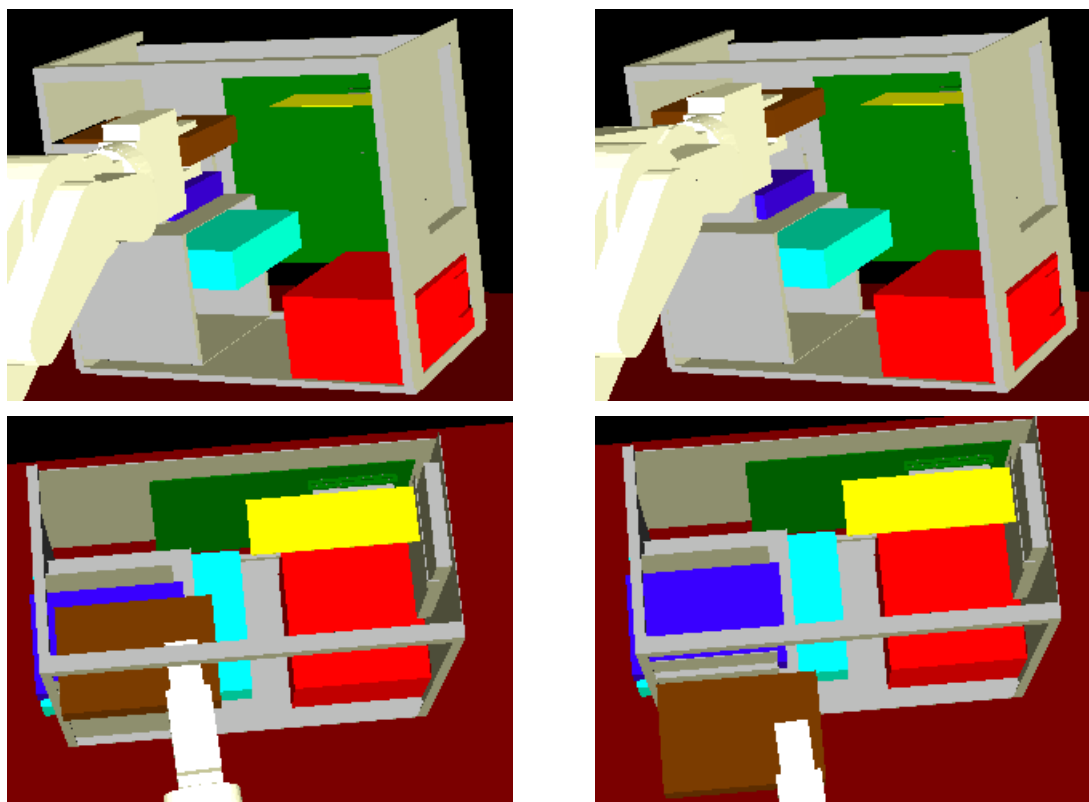


Figura 5.32: Secuencia simulada de la trayectoria de desensamblado del componente "disco duro".

Una vez simulado el desensamblado del "disco duro" se realiza la ejecución sobre el sistema real, tal como muestra la Figura 5.33.

En este caso los valores articulares del robot en los puntos de control, de las trayectorias lineales, son:

$$d_1 \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (79878, 17865, -38018, 119907, 52383) \quad (5.13)$$

$$d_2 \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (79878, -34027, -115748, 148878, 52383) \quad (5.14)$$

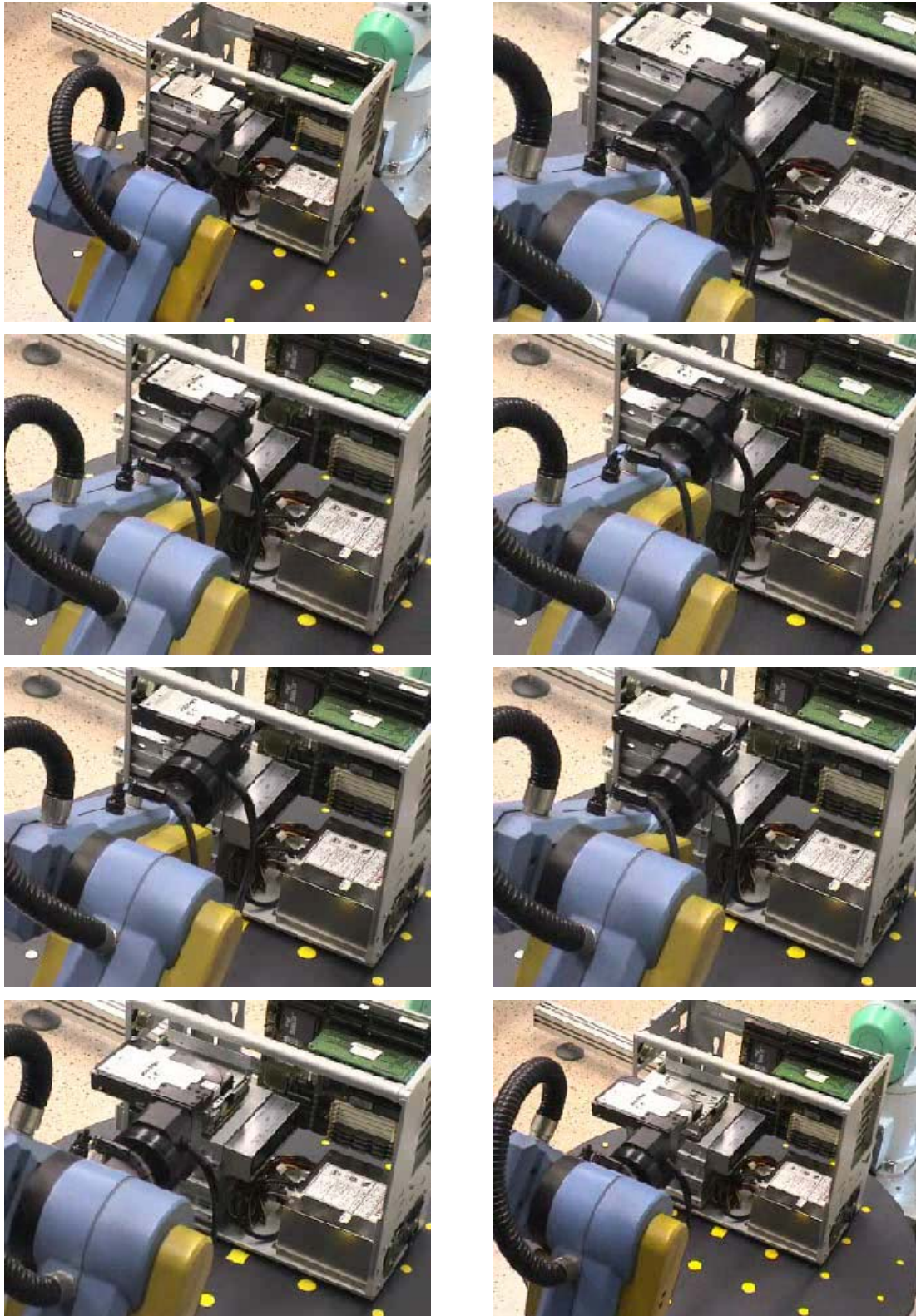
Donde se parte del punto dado en coordenadas articulares, como:

$$\text{origen} \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (80140, 25121, -42816, 129312, 52388) \quad (5.15)$$

Y el componente, una vez desensamblado, se deposita en el almacén situado en la posición articular:

$$\text{almacén} \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (-114877, -34027, -115748, 148878, 52383) \quad (5.16)$$

Con lo que se ha conseguido desensamblar, de manera automática, la "tarjeta de vídeo" y el "disco duro" del PC, tal y como se deseaba.



*Figura 5.33: Secuencia real de desensamblado del componente "disco duro".*

### 5.5.2. Desensamblado de un PC sobremesa

En este apartado se realiza el desensamblado de un PC sobremesa, llevando a cabo el desensamblado de la "tarjeta de vídeo". Para ello se parte,

igual que para el PC semitorre, de un reconocimiento del producto que se encuentra en la mesa de trabajo. Esto se lleva a cabo por el sistema de visión, que adquiere varias vistas del producto, utilizando la mesa para orientarlo, (Figura 5.34) y así poder reconocer el producto de que se trata.

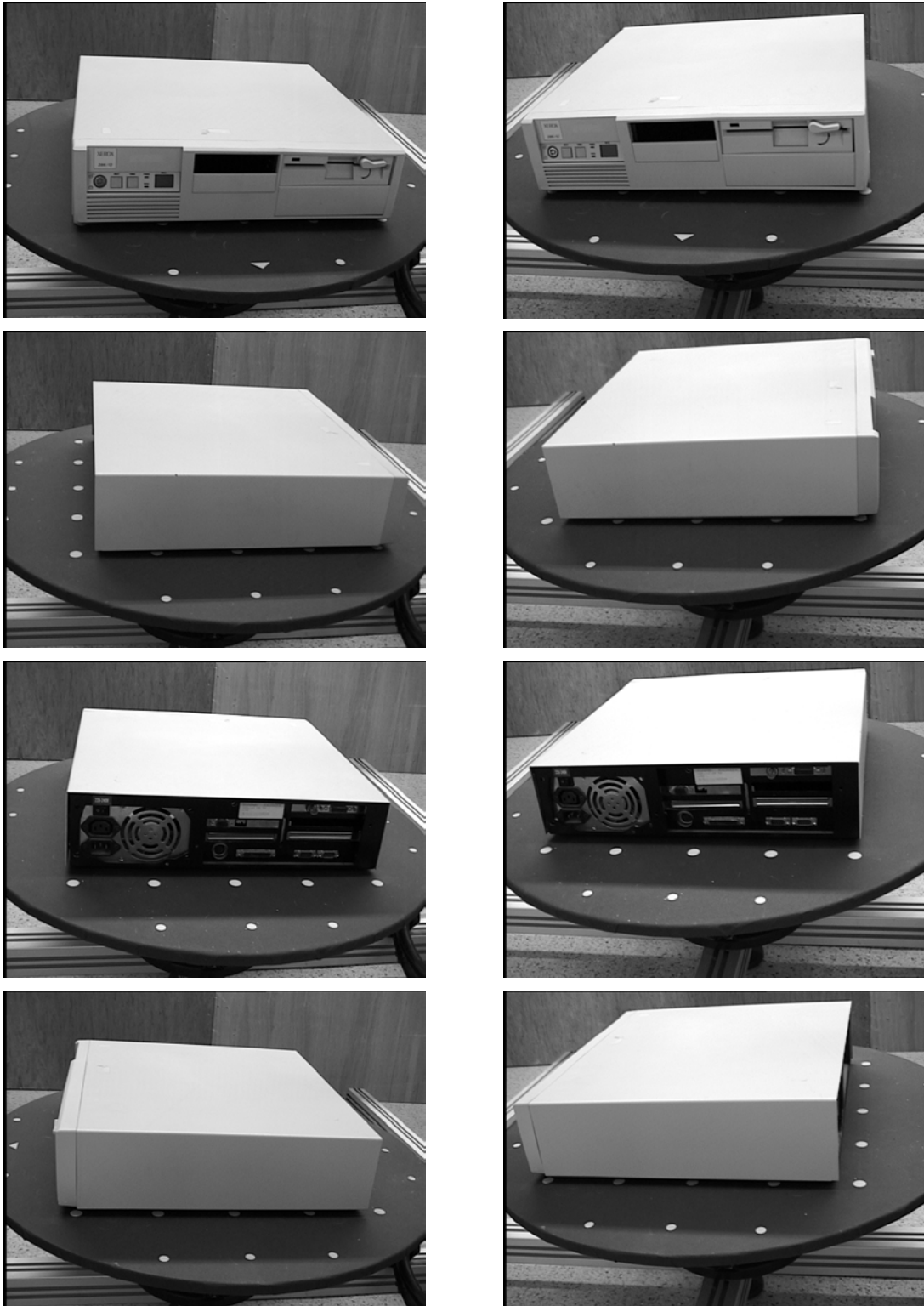


Figura 5.34: Sistema de visión adquiriendo información sobre el tipo de producto, PC sobremesa; imágenes de la cámara izquierda y derecha del par estéreo.



Una vez reconocido y seleccionado el tipo de producto, el planificador global escoge el modelo relacional apropiado, que en este caso, se corresponde con el de un PC genérico, el mismo del caso anterior (Figura 5.18).

El planificador global debe decidir en estos momentos si expandir o no algún componente; por ello decide que hay que expandir el componente "tarjetas", ya que se requiere realizar un desensamblado de una tarjeta y por tanto, se debe reconocer si posee o no la tarjeta buscada.

Con esta información el planificador global indica, al planificador de desensamblado, que el nuevo modelo debe poseer dos tarjetas: una "tarjeta de vídeo" y otra "tarjeta de red". Por lo que coincide con el modelo extendido del PC semitorre de la Figura 5.19.

Teniendo estos datos en cuenta, para llevar a cabo el desensamblado de la "tarjeta de vídeo" se debe proceder de igual manera que se siguió para el PC semitorre, es decir, hay que desensamblar el componente "carcasa exterior". Para ello el planificador de desensamblado informa al generador de movimientos de desensamblado, para que calcule la trayectoria a seguir por el componente "carcasa exterior" y pueda ser desensamblado.

Para el cálculo de la trayectoria de desensamblado, el planificador global elimina, como objeto geométrico, al robot, dado que, en este caso, no es posible realizar el desensamblado de la carcasa exterior mediante su uso, ya que no posee un volumen de trabajo adecuado.

La trayectoria obtenida consiste en realizar un desplazamiento del componente "carcasa exterior" según el eje x, tal y como se muestra en la simulación (Figura 5.35 y Figura 5.36)

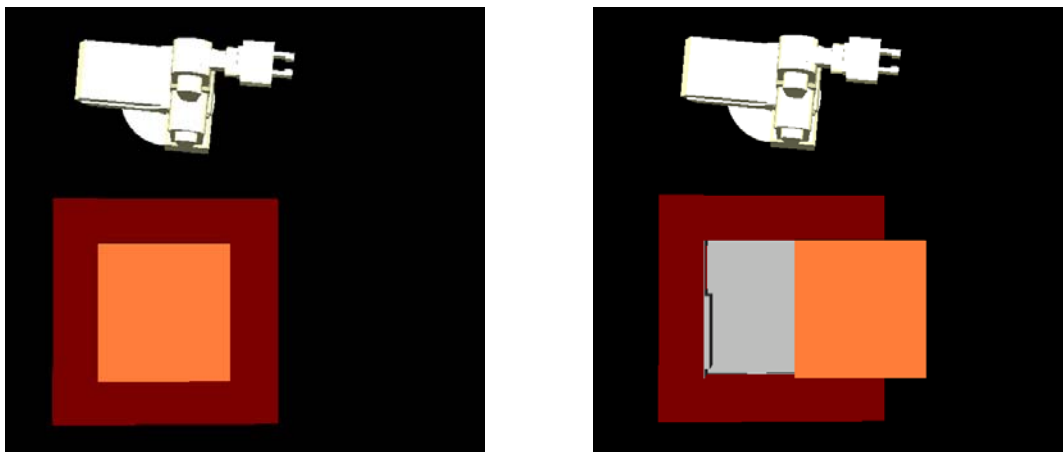


Figura 5.35: Secuencia simulada de la trayectoria de desensamblado del componente "carcasa exterior" de un PC sobremesa (Parte 1).

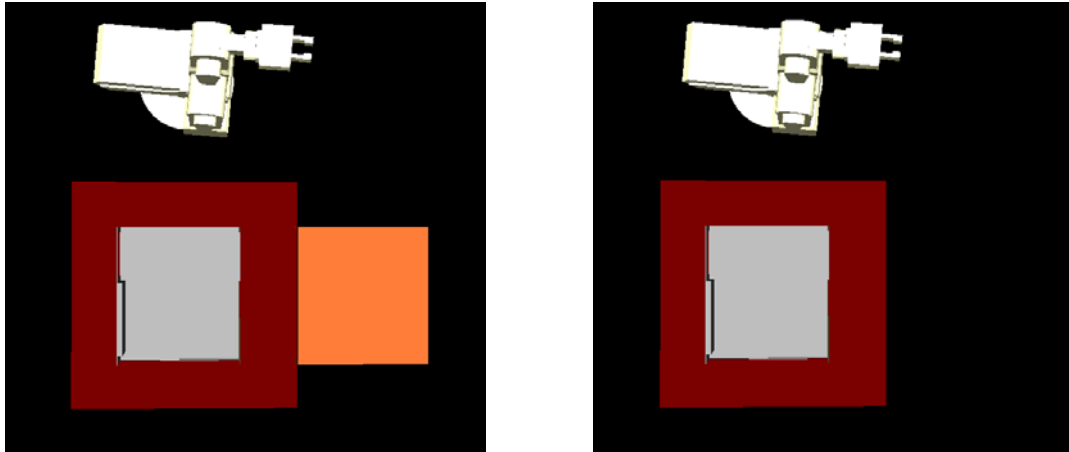


Figura 5.36: Secuencia simulada de la trayectoria de desensamblado del componente "carcasa exterior" de un PC sobremesa (Parte 2).

Una vez realizado el desensamblado de la "carcasa exterior", el planificador global toma la decisión de expandir varios componentes. En este caso se realiza la expansión del componente "periféricos", con lo que se obtiene la existencia de una "disquetera de 5¼" y un "disco duro". Por otro lado se comprueba la existencia de la "fuente de alimentación" y de la "placa base", así como se obtiene información más precisa sobre las tarjetas.

Para ello, el sistema realiza una nueva captura de imágenes del PC sobre la mesa de trabajo (Figura 5.37 y Figura 5.38).

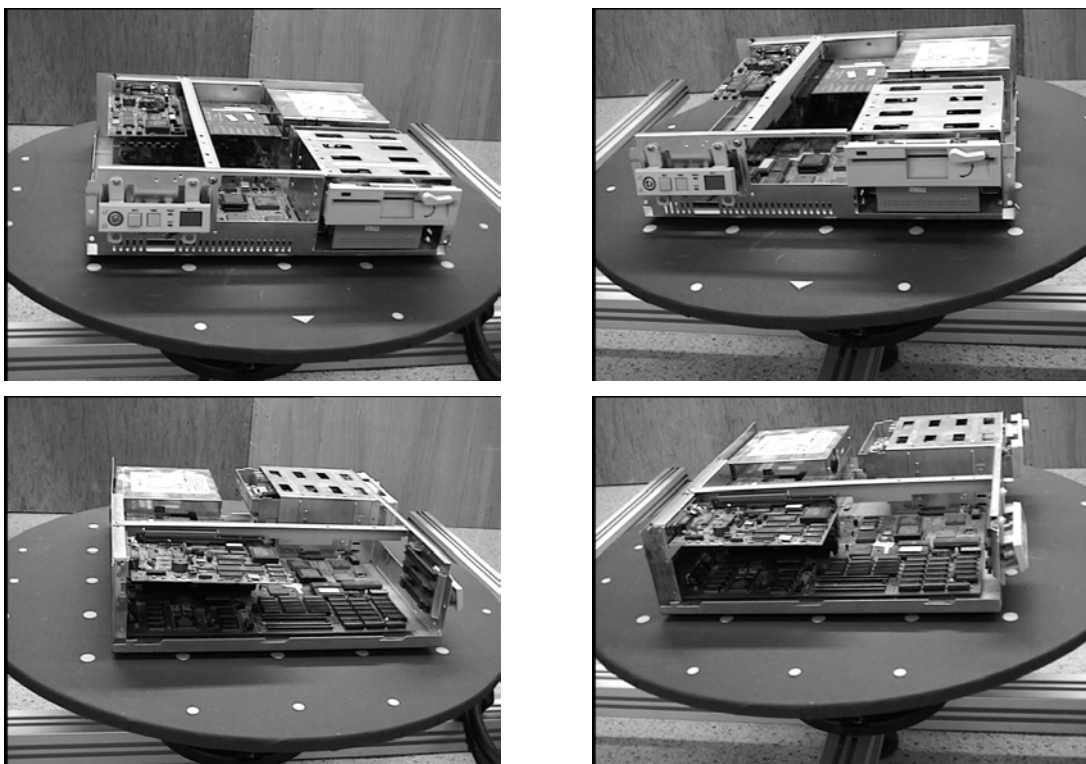


Figura 5.37: Sistema de visión adquiriendo información sobre el tipo de producto, PC sobremesa; imágenes de la cámara izquierda y derecha del par estéreo (Parte 1).

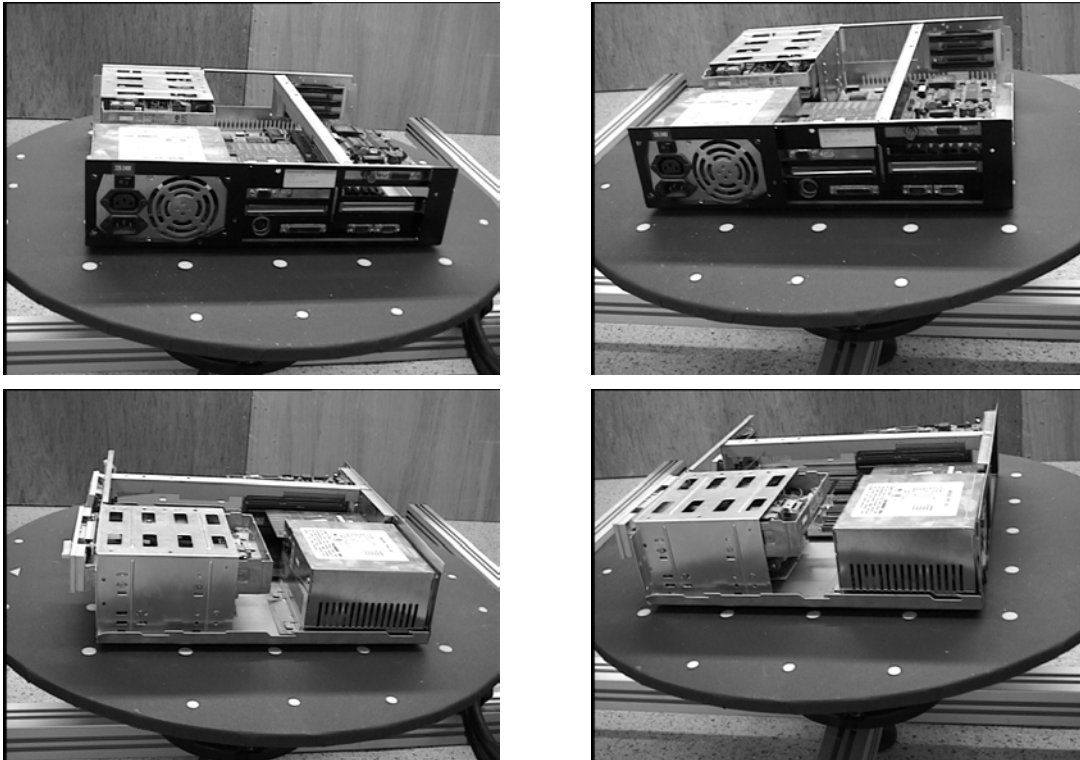


Figura 5.38: Sistema de visión adquiriendo información sobre el tipo de producto, PC sobremesa; imágenes de la cámara izquierda y derecha del par estéreo (Parte 2).

En este caso, por tratarse de un PC sobremesa, existe la posibilidad de tener componentes unidos a la "carcasa interior", placas auxiliares, por lo que se solicita su detección al sistema de visión. En este caso, se busca el reconocimiento del componente auxiliar, que existe como una barra sobre la placa base.

De esta manera, se detecta la existencia de la "placa base" que no está formada por una única placa, sino que incluye una tarjeta, con lo que la placa se compone de la "placa base" propiamente dicha, y de una "tarjeta auxiliar".

Los componentes detectados se indican claramente en la Figura 5.39.

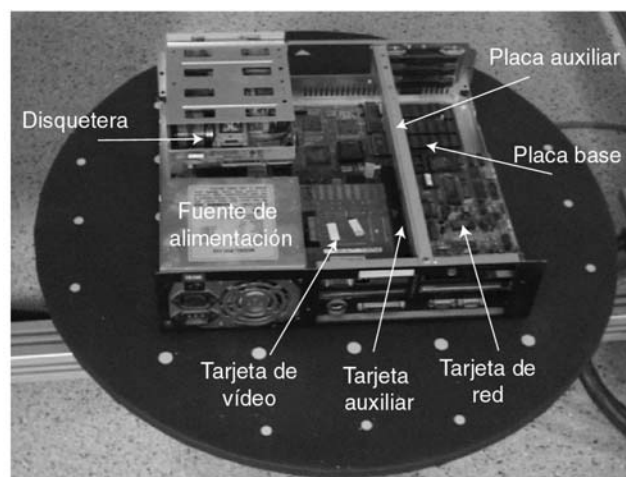


Figura 5.39: PC sobremesa con los componentes internos visibles.

Con toda esta información, se genera un modelo relacional extendido del PC sobremesa; dicho modelo se muestra en la Figura 5.40.

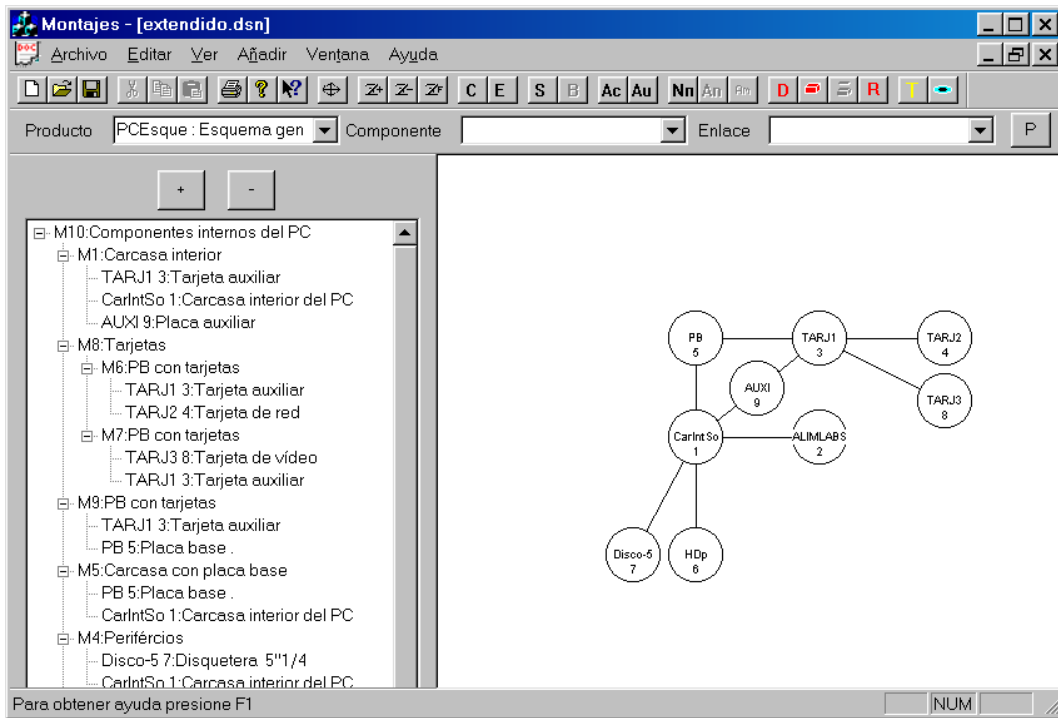


Figura 5.40: Modelo relacional extendido del PC sobremesa.

Donde los montajes contienen los siguientes componentes:

- $m_1$ : Está compuesto por los componentes: "carcasa interior" ( $c_1$ ), "tarjeta auxiliar" ( $c_3$ ) y "placa auxiliar" ( $c_9$ ).
- $m_2$ : Está formado por los componentes: "carcasa interior" ( $c_1$ ) y "fuente de alimentación" ( $c_2$ ).
- $m_3$ : Está compuesto por los componentes: "carcasa interior" ( $c_1$ ) y "disco duro" ( $c_6$ ).
- $m_4$ : Está compuesto por los componentes: "carcasa interior" ( $c_1$ ) y "disquetera 5¼" ( $c_7$ ).
- $m_5$ : Está compuesto por los componentes: "carcasa interior" ( $c_1$ ) y "placa base" ( $c_5$ ).
- $m_6$ : Está compuesto por los componentes: "tarjeta auxiliar" ( $c_3$ ) y "tarjeta de red" ( $c_4$ ).
- $m_7$ : Está compuesto por los componentes: "tarjeta auxiliar" ( $c_3$ ) y "tarjeta de vídeo" ( $c_8$ ).
- $m_8$ : Está compuesto por los montajes:  $m_6$  y  $m_7$ .
- $m_9$ : Está formado por los componentes: "tarjeta auxiliar" ( $c_3$ ) y "placa base" ( $c_5$ ).
- $m_{10}$ : Está compuesto por los montajes:  $m_1$ ,  $m_2$ ,  $m_3$ ,  $m_4$ ,  $m_5$ ,  $m_8$  y  $m_9$ .

Utilizando este nuevo modelo relacional del producto el planificador de desensamblado determina la nueva secuencia para desensamblar el componente "tarjeta de vídeo".

- 1) Como el componente "tarjeta de vídeo" pertenece al montaje  $m_7$  hay que:
  - a. Eliminar la unión entre los componentes "tarjeta de vídeo" y "tarjeta auxiliar".
  - b. Separar los dos componentes.
  - c. Como el montaje  $m_7$  pertenece al montaje  $m_8$  se tienen que seguir las acciones de desensamblado para el montaje  $m_8$ .
- 2) Las acciones asociadas con el montaje  $m_8$  son:
  - a. No hacer nada.
  - b. Como el montaje  $m_8$  pertenece al montaje  $m_{10}$  hay que continuar con las acciones de desensamblado correspondientes al montaje  $m_{10}$ .
- 3) Las acciones asociadas al montaje  $m_{10}$  son:
  - a. No hacer nada.
  - b. Como el montaje  $m_{10}$  es el montaje final, ya se tienen definidas las acciones para desensamblar el componente "tarjeta de vídeo".

Esta secuencia se corresponde con la mostrada en el grafo de la Figura 5.41.

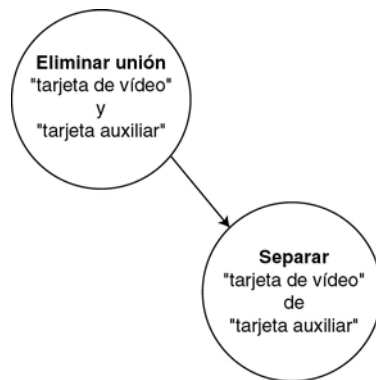


Figura 5.41: Grafo de acciones a llevar a cabo por el sistema, para desensamblar el componente "tarjeta de vídeo" del PC sobremesa.

Esta información es proporcionada por el programa para conocimiento del usuario, tal como la muestra la Figura 5.42.

El planificador de desensamblado, remite la información al planificador de movimientos, para que calcule cuales son los movimientos de desensamblado para el componente "tarjeta de vídeo".

El planificador de movimientos antes de realizar los cálculos debe actualizar el modelo del producto que posee, según los nuevos componentes introducidos y la información relativa a ellos que le proporcione el sistema de visión. Con esta información, el planificador de movimientos genera el modelo del producto mostrado en la Figura 5.43.

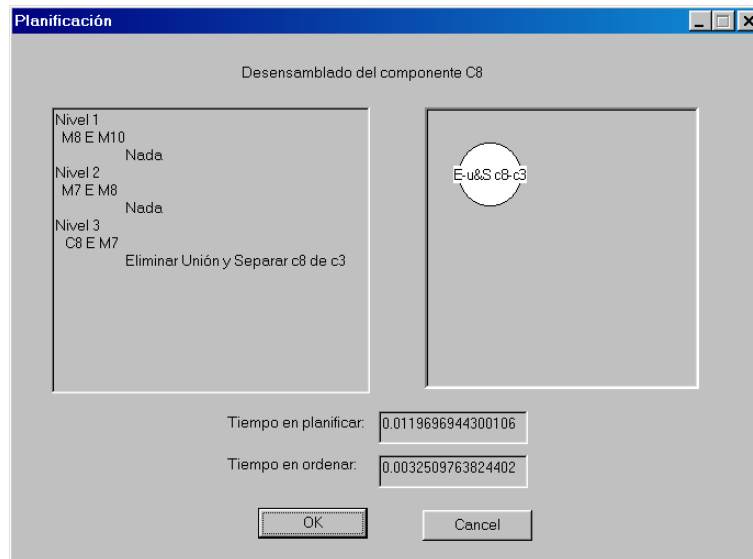


Figura 5.42: Acciones a llevar a cabo por el sistema para desensamblar el componente "tarjeta de vídeo" del PC sobremesa.

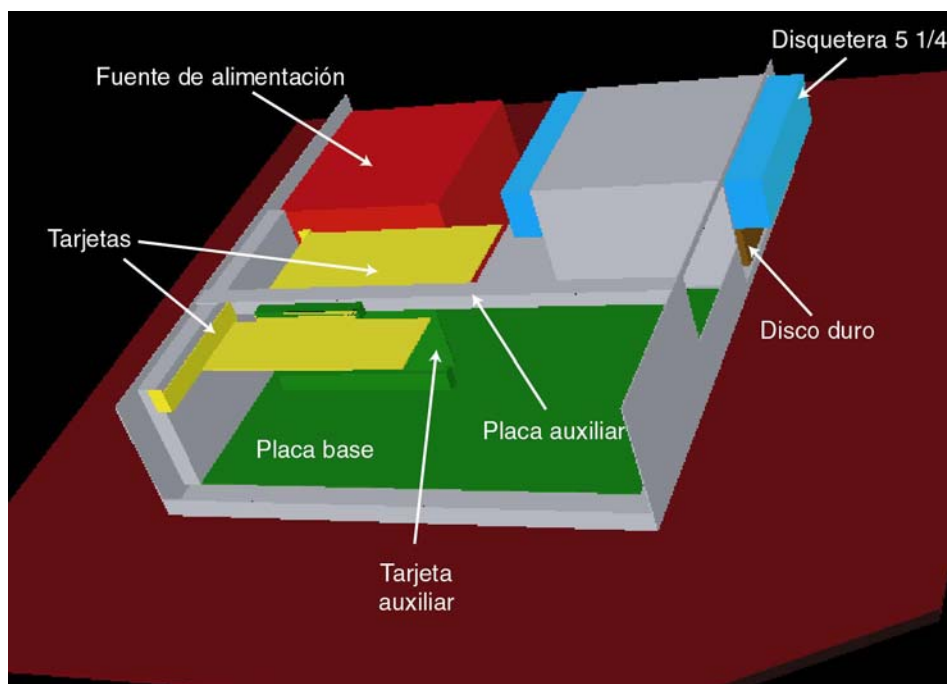


Figura 5.43: Modelo geométrico extendido del PC sobremesa.

Una vez actualizado el modelo, se tiene que calcular la trayectoria de desensamblado del componente; pero cuando se añade la herramienta para el desensamblado aparece una colisión, imposible de evitar, con la "fuente de alimentación" (Figura 5.44).

Este hecho implica que se debe desensamblar antes el componente "fuente de alimentación" que el componente "tarjeta de vídeo", ya que no se puede enganchar dicho componente, mientras la fuente de alimentación se encuentre en su posición.

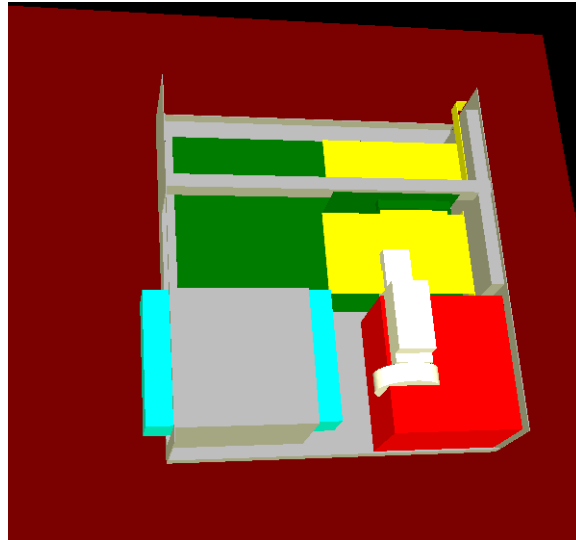


Figura 5.44: Herramienta colisionando con el componente "fuente de alimentación".

Por lo tanto el planificador global solicita al planificador de desensamblado que calcule la secuencia de desensamblado para el componente "fuente de alimentación", según el modelo relacional de la Figura 5.40.

Con lo que la secuencia obtenida es:

- 1) Como el componente "fuente de alimentación" pertenece al montaje  $m_2$  hay que:
  - a. Eliminar la unión entre los componentes "carcasa interior" y "fuente de alimentación".
  - b. Separar los dos componentes.
  - c. Como el montaje  $m_2$  pertenece al montaje  $m_{10}$  se tienen que seguir las acciones de desensamblado para el montaje  $m_{10}$ .
- 2) Las acciones asociadas al montaje  $m_{10}$  son:
  - a. No hacer nada.
  - b. Como el montaje  $m_{10}$  es el montaje que representa a todos los componentes existentes en estos momentos, en el producto se tiene definida la secuencia de desensamblado para el componente "fuente de alimentación".

La secuencia de acciones obtenida se muestra en la Figura 5.45.

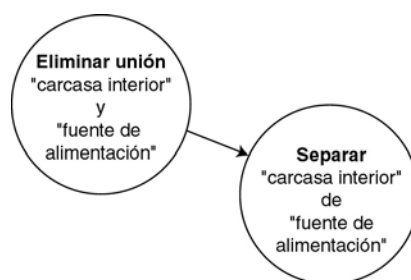


Figura 5.45: Grafo de acciones a realizar para desensamblar el componente "fuente de alimentación" del PC sobremesa.

Con esta información, el planificador global indica al generador de movimientos que debe calcular la secuencia de desensamblado para el componente "fuente de alimentación".

En este caso, se necesita una ventosa para realizar el desensamblado del componente, no sirve la pinza. Como no se dispone de una ventosa, en el sistema real, se utiliza un modelo geométrico de la misma, para el cálculo de la trayectoria (Figura 5.46). Además tampoco se incluye el robot, dado que no se le puede acoplar la herramienta para realizar el trabajo. Por lo que el desensamblado de la fuente de alimentación habrá que realizarlo manualmente.

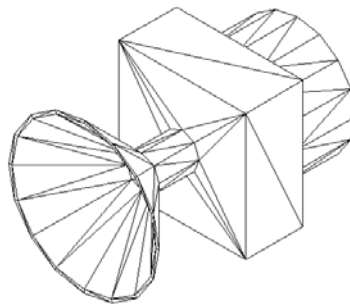


Figura 5.46: Modelo geométrico utilizado para la herramienta "ventosa".

Una vez actualizados todos los datos necesarios para el generador de movimientos, se obtiene la siguiente secuencia de movimientos para llevar a cabo el desensamblado de la fuente de alimentación:

$$d_1 = (21, 0, 21) \quad (5.17)$$

$$d_2 = (0, 0, 400) \quad (5.18)$$

Por lo que se puede pasar a realizar la simulación de la misma (Figura 5.47 y Figura 5.48).

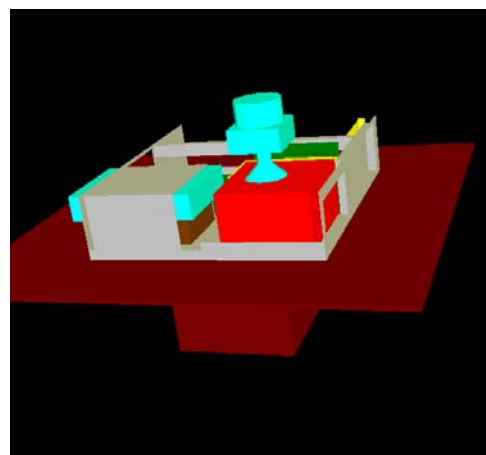
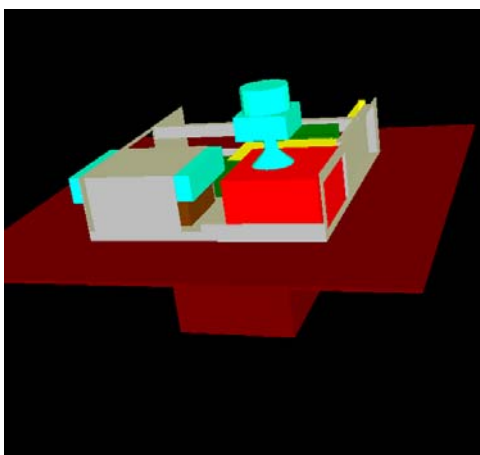


Figura 5.47: Secuencia simulada de la trayectoria de desensamblado del componente "fuente de alimentación" de un PC sobremesa (Parte 1).



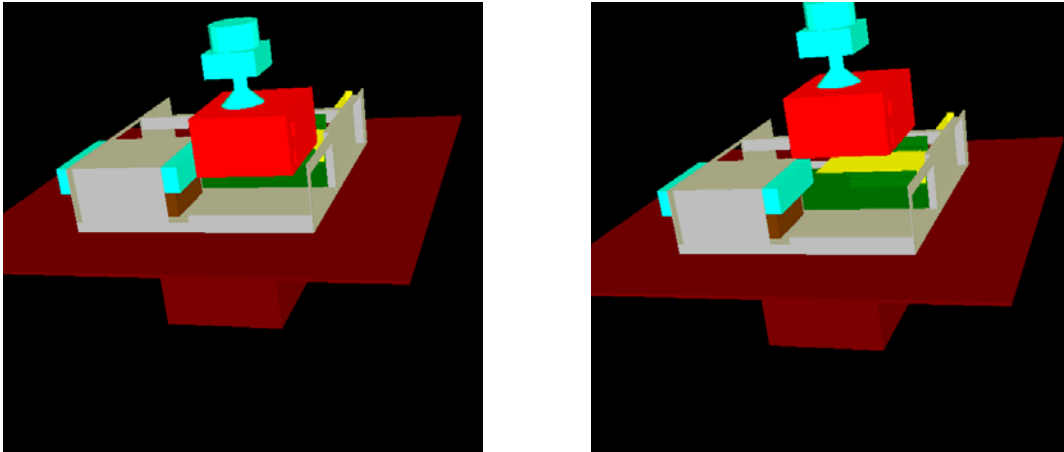


Figura 5.48: Secuencia simulada de la trayectoria de desensamblado del componente "fuente de alimentación" de un PC sobremesa (Parte 2).

Después de desensamblar el componente "fuente de alimentación", se sigue con el desensamblado del componente "tarjeta de vídeo". Para este componente el generador de movimientos calcula la secuencia de desensamblado, como:

$$d_1 = (0, 60, 0) \quad (5.19)$$

$$d_2 = (0, 0, 394) \quad (5.20)$$

Teniendo en cuenta que esta operación de desensamblado sí que puede ser realizada por el robot, hay que posicionar el producto adecuadamente sobre la mesa de trabajo, para que se pueda llevar a cabo el desensamblado del mismo. Por lo tanto el sistema, utilizando las ecuaciones del Anexo C, obtiene, para posicionar el producto, los siguientes valores:

$$d_x = 47 \quad d_y = -100 \quad \theta = 0^\circ \quad (5.21)$$

En base a esta información, se posiciona el producto sobre la mesa de trabajo, antes de proceder a su desensamblado por el robot (Figura 5.49).

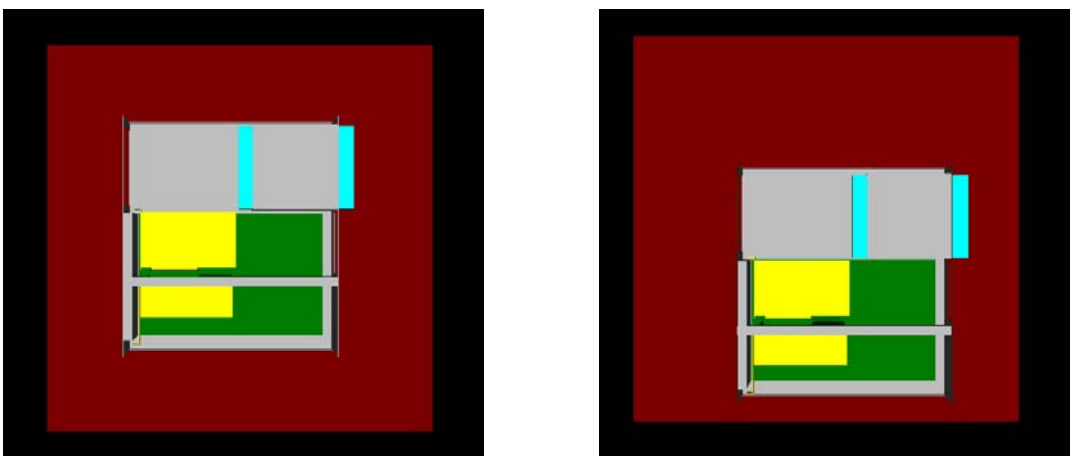


Figura 5.49: Producto sobre la mesa de trabajo en la posición original y tras trasladarlo y rotarlo para que el robot pueda desensamblar la "tarjeta de vídeo".

Una vez se tiene ya posicionado el producto en la mesa de trabajo, y dado que se ha calculado la trayectoria de movimientos de desensamblado, se realiza la simulación de la misma (Figura 5.50).

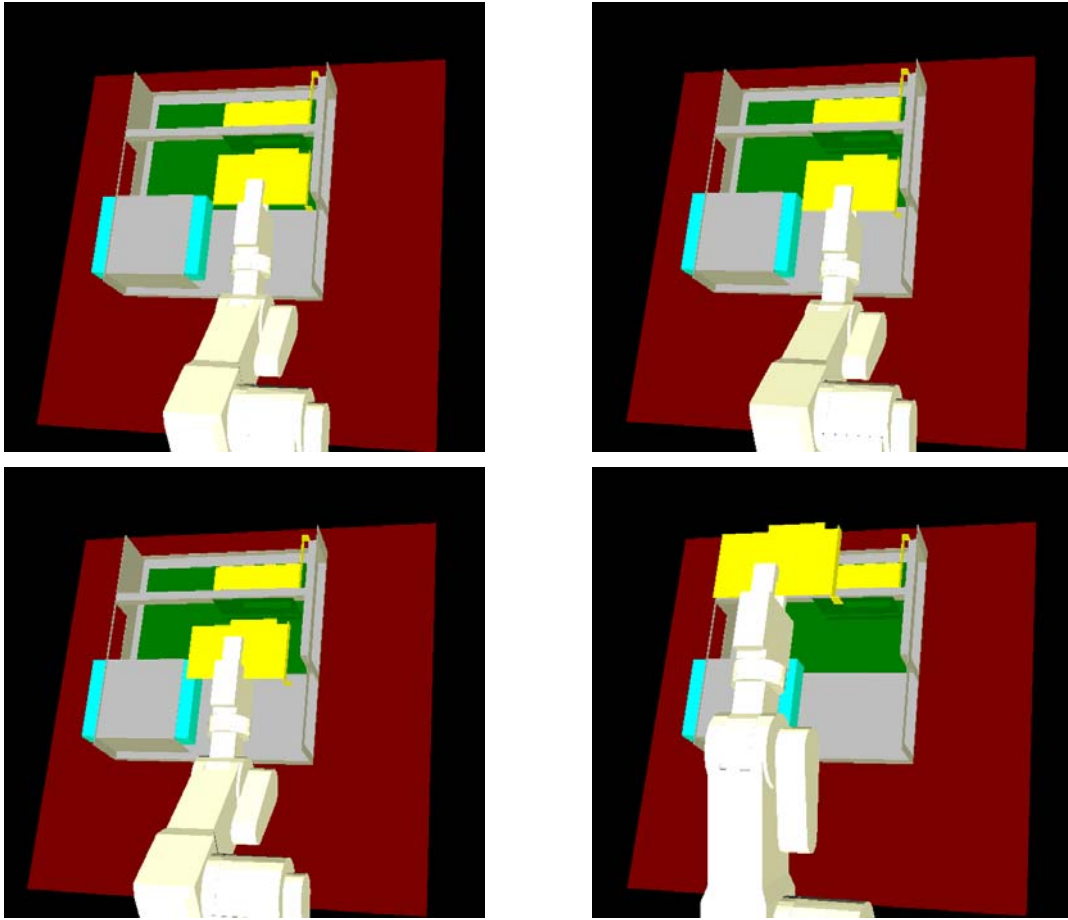


Figura 5.50: Secuencia simulada de la trayectoria de desensamblado del componente "tarjeta de vídeo" de un PC sobremesa.

Además, el sistema informa sobre los valores articulares que debe tomar el robot para realizar el desensamblado, ya que, a continuación, debe realizarlo el robot real. Teniendo esto en consideración, hay dos puntos de control principales para realizar el desensamblado, definidos por los valores articulares siguientes:

$$\mathbf{d}_1 \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (91052, 46425, -56136, 157719, 51894) \quad (5.22)$$

$$\mathbf{d}_2 \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (91052, 2585, -4896, 82118, 51894) \quad (5.23)$$

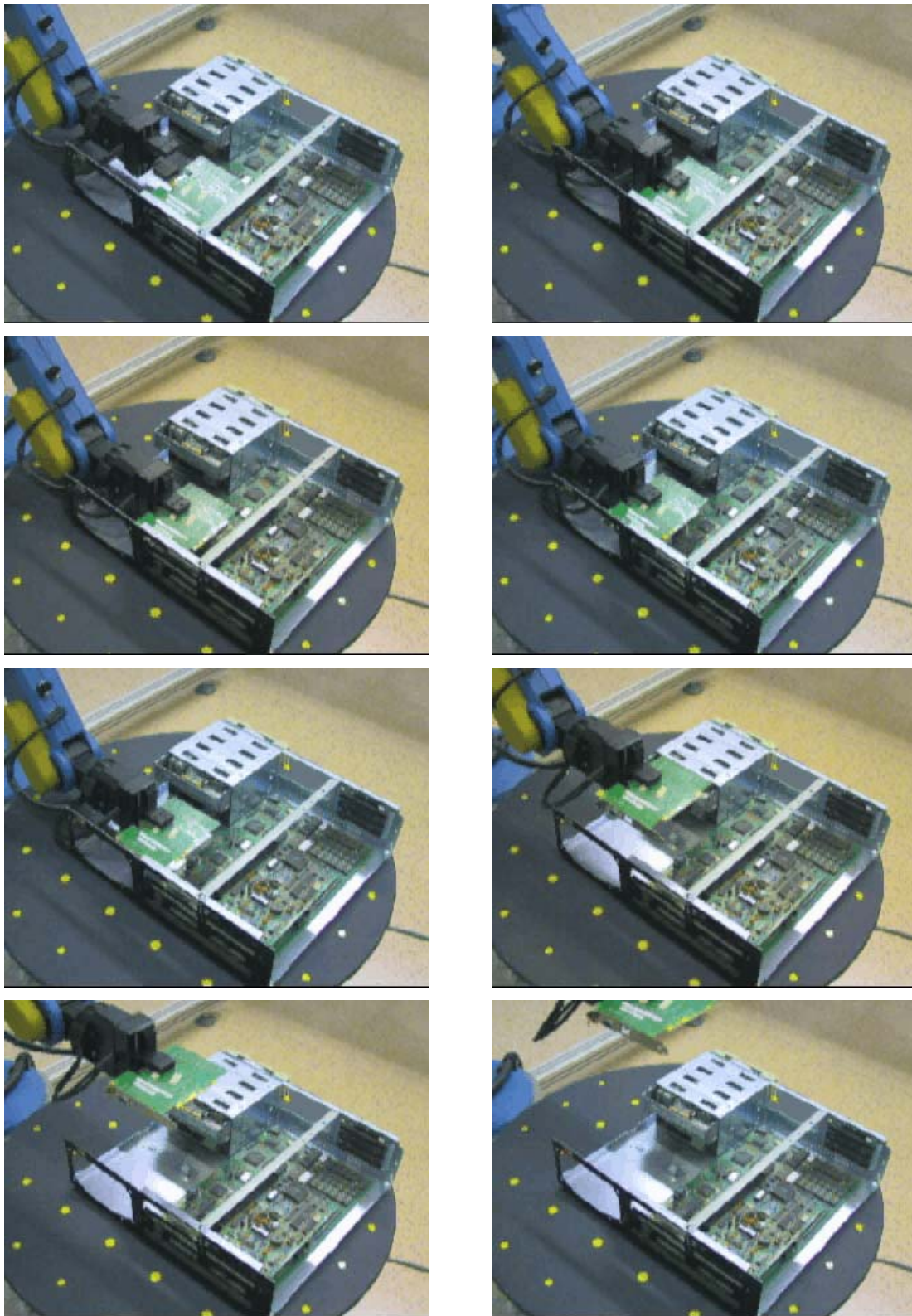
Así mismo se considera como punto de origen para realizar la trayectoria de desensamblado al punto:

$$\text{origen} \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (91052, 26485, -81593, 164476, 51894) \quad (5.24)$$

Además el componente desensamblado se deposita en el almacén situado en la posición articular:

$$\text{almacén} \Rightarrow (q_1, q_2, q_3, q_4, q_5) = (-114877, -34027, -115748, 148878, 52383) \quad (5.25)$$

Con lo que ya se tiene desensamblado el componente deseado de manera simulada. Por lo tanto, se lleva a cabo la misma operación por parte del robot real (Figura 5.51).



*Figura 5.51: Secuencia real de desensamblado del componente "tarjeta de vídeo" en el PC sobremesa.*

# 6

## Conclusiones

Esta tesis presenta un nuevo método de planificación y generación de trayectorias de movimientos para procesos de desensamblado no destructivo. Además se lleva a cabo la implementación de una célula flexible de desensamblado automático.

Teniendo en cuenta estas consideraciones, a continuación, se describen los aspectos más relevantes tratados en cada uno de los capítulos.

En el capítulo 2 se realiza una de las aportaciones originales de esta tesis; la introducción en el apartado 2.3 del modelo relacional para representar un producto; este modelo permite, de manera intuitiva y sencilla, diseñar productos existentes y posteriormente, mediante las estrategias de planificación que se explican en el apartado 2.5, llevar a cabo un desensamblado bien sea parcial o total del producto.

La generación de la secuencia de desensamblado parcial de un producto puede realizarse tanto partiendo de un único componente como del desensamblado de un conjunto de componentes, un montaje. De manera que da una versatilidad mayor al sistema de desensamblado.

Este algoritmo de planificación de secuencias de desensamblado ha dado lugar a una publicación en una revista internacional impactada:

*"Disassembly planning based on precedence relations among assemblies"*. F. Torres, S.T. Puente, R. Aracil. The International Journal of Advanced Manufacturing Technology. Aceptada y pendiente de publicación en el 2003.

Una vez presentado el método para calcular la planificación de desensamblado se explica, en el capítulo 3, cómo llevar a cabo la representación geométrica del producto basándose en poliedros, de manera

que un componente se representa como un poliedro y un producto a su vez es un conjunto de poliedros.

En base a esta representación, se aporta un método para el cálculo de los movimientos de desensamblado, el generador de movimientos, que debe seguir un componente para poder separarse de los demás, teniendo en consideración que el componente debe describir una trayectoria basada en segmentos lineales.

Una descripción del método planteado para la generación de los movimientos de desensamblado ha dado lugar a una publicación en un congreso internacional:

*"Disassembly movements for geometrical objects through heuristic methods"*. J. Pomares, F. Torres, S. Puente. Proceedings of SPIE: Environmentally Conscious Manufacturing II. Vol. 4569, pp. 71-80. Newton, USA. 28 Octubre-2 Noviembre de 2001.

Tras estos apartados, en el capítulo 4, se ha descrito cómo se lleva a cabo la fusión de la información proveniente del planificador de tareas de desensamblado con la correspondiente a la trayectoria de desensamblado. Teniendo en cuenta las posibles configuraciones del sistema, bien sea utilizando un brazo robot o no, considerando todo o parte del mismo para el cálculo y como optimizar las secuencias para aprovechar el paralelismo existente entre las tareas a la hora de minimizar la necesidad de cambiar de herramienta para realizar el proceso de desensamblado.

Seguidamente en el capítulo 5 se ha ilustrado una aplicación del sistema flexible de desensamblado automático. Presentado la base de datos utilizada, la arquitectura del sistema y la aplicación al desensamblado de varios ordenadores personales.

La arquitectura del sistema utilizado, junto con una descripción del funcionamiento del sistema de visión artificial ha sido publicada en un Workshop internacional sobre desensamblado:

*"Data fusion from multiples cameras for automatic disassembly"*. P. Gil, S.T Puente, F. Torres, J. Pomares, F.A. Candelas. IFAC Workshop on Intelligent Assembly and Disassembly - IAD'2001. Canela, Brasil. 5-7 Noviembre de 2001.

Para terminar se tiene que destacar el hecho de haber implementado una célula flexible de desensamblado automático, que permite realizar un desensamblado no destructivo de productos existentes en el mercado y por tanto reducir los costes de producción al reutilizar los componentes de dichos productos. La cual supone un beneficio para el medio ambiente ya que permite reducir la cantidad de productos desechados.

El sistema de desensamblado no destructivo resulta interesante para realizar operaciones de reutilización de productos, frente a otros sistemas que permiten el desensamblado destructivo que serían los indicados para realizar un desensamblado de cara al reciclado de las materias primas.

## **6.1. Aportaciones**

Las principales aportaciones originales de esta tesis se describen a continuación:

Una representación de los productos mediante una estructura jerárquica de sus componentes, que se basa en un modelo relacional del producto (Capítulo 2), esta es una de las aportaciones originales de esta tesis. Además esta representación permite una definición granular del producto en función de los objetivos de desensamblado perseguidos. Este método utiliza un conjunto de acciones posibles para desensamblar los componentes, que dependen de las relaciones existentes entre ellos.

El cálculo de secuencias de desensamblado no destructivo tanto parcial como total de un producto, utilizando el modelo relacional, es otra de las aportaciones originales de esta tesis. Donde el desensamblado parcial puede ser tanto de componentes individuales como de grupos de ellos, un montaje.

La utilización de información sobre el entorno cercano del componente a desensamblar así como un modelo geométrico del producto, se utilizan para la generación de los movimientos de desensamblado de cada componente. Adaptando técnicas de robótica móvil al proceso de desensamblado de los productos. Dotando, de esta manera, al sistema de capacidades de n-desensamblado. Este generador de movimientos de desensamblado es una de las aportaciones originales de esta tesis.

La fusión entre el planificador de secuencia y el generador de movimientos, incluyendo un manipulador que realice las tareas de desensamblado a la hora de generar los movimientos, es otra de las aportaciones originales de esta tesis.

Además, esta fusión con el manipulador, incluye un método para ordenar las acciones de desensamblado que permite minimizar los cambios de herramientas por parte del manipulador para realizar el desensamblado.

La arquitectura del sistema flexible de desensamblado automático que permite, gracias a un sistema de visión artificial adaptarse a los distintos productos, se puede considerar como otra de las aportaciones originales de esta tesis.

## 6.2. Trabajo futuro

Las líneas de investigación que abre la realización de esta tesis son múltiples, y a continuación se describen algunas de ellas:

Respecto de la planificación de desensamblado:

- Generación de estrategias que consideren la posibilidad de desensamblado destructivo.

Respecto de las trayectorias de movimientos:

- Incluir movimientos rotacionales de los componentes para llevar a cabo su desensamblado.
- Investigar otros métodos de generación de movimientos de desensamblado.
- Generar la zona de movilidad de un componente; (todas las trayectorias posibles según las cuales un componente puede ser desensamblado).

Respecto de la unión de la planificación de desensamblado y del generador de trayectorias de movimiento:

- Añadir nuevos grados de libertad al sistema.
- Añadir operaciones de desensamblado en paralelo, mediante robots cooperativos.

Respecto del sistema de desensamblado real:

- Generación de un módulo supervisor que compruebe la correcta realización del proceso de desensamblado planificado.
- Utilizar un robot con mayor número de grados de libertad para realizar el desensamblado.
- Ampliación del número de herramientas disponibles para poder desensamblar mayor cantidad de componentes.
- Utilizar un control de fuerza para comprobar la correcta realización de las operaciones y una mayor información al sistema.

# Anexo A

## Entorno desarrollado

El entorno que se ha desarrollado para la realización de la presente tesis permite la definición del modelo relacional de un producto en base a los componentes existentes en la base de datos. De esta manera se puede realizar una simulación para el cálculo de las estrategias de desensamblado correspondientes a cualquier producto. Cabe destacar que, para poder realizar la simulación completa de un producto, se requiere el conocimiento del producto completo, lo que quiere decir un modelo relacional y un modelo geométrico.

### A.1. Funcionalidades

Las opciones principales que posee el entorno, para la introducción de los modelos relacionales, se aprecian en la Figura A.1, donde se pueden ver los aspectos correspondientes a la introducción de componentes en el modelo, añadir elementos de enlace, añadir montajes, asignar los componentes a los montajes, etc. Seguidamente se describe cómo realizar cada una de estas acciones.

#### A.1.1. Añadir componentes

El proceso de añadir un nuevo componente, para introducirlo en el modelo consiste, en primer lugar, en seleccionar el componente a añadir de la lista de componentes disponibles; tras esto, seleccionar la opción de añadir componente. Con esta opción activa, cada vez que pulsemos sobre la zona correspondiente al modelo relacional del producto, se añade el componente seleccionado al modelo.



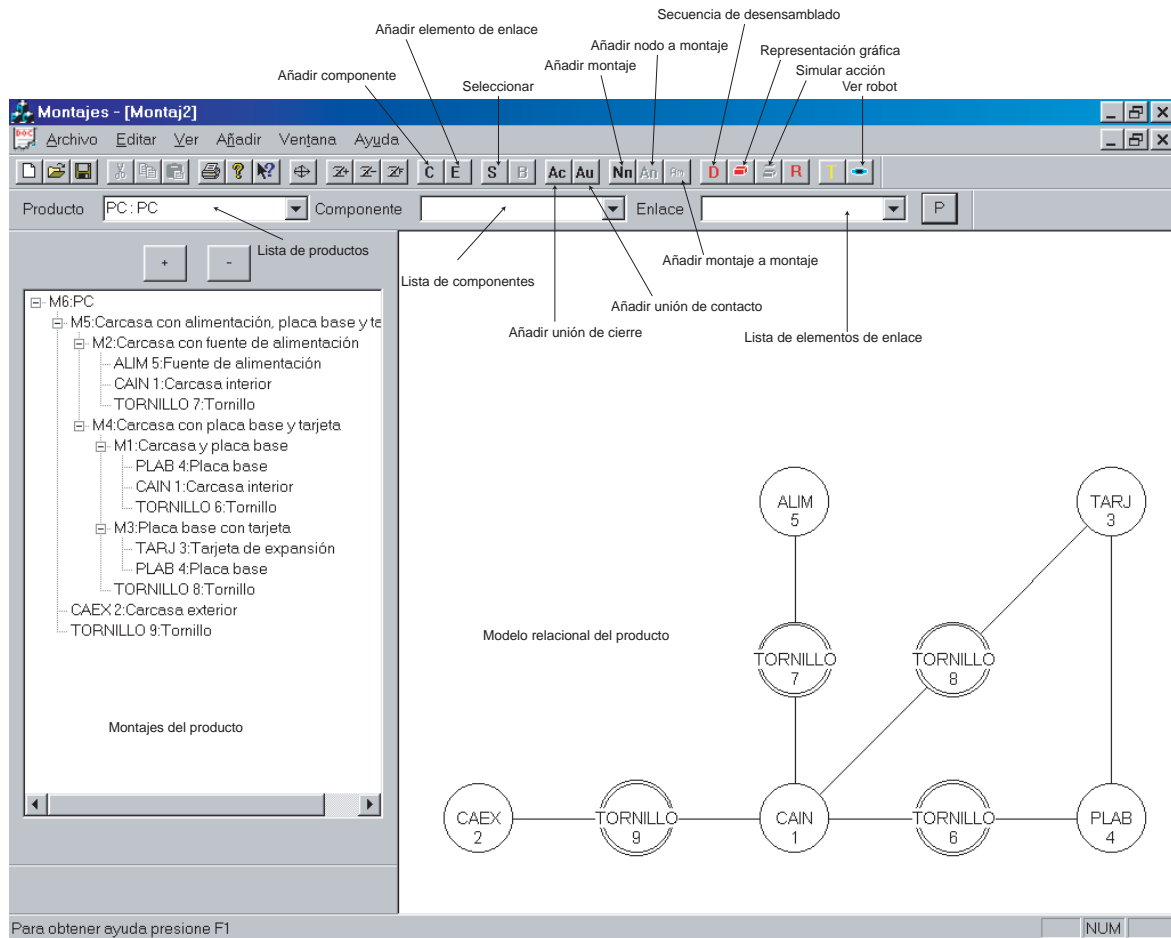


Figura A.1: Entorno de desarrollo para la introducción de los modelos relacionales.

De esta manera, pulsando sobre el área del modelo, se añaden los componentes que se desee, repitiendo estas operaciones para cada uno de los componentes que se deseen introducir en el modelo.

En la Figura A.2 se muestra una imagen de los elementos de un mando a distancia, que se va a utilizar para ilustrar las distintas fase de creación de un modelo.

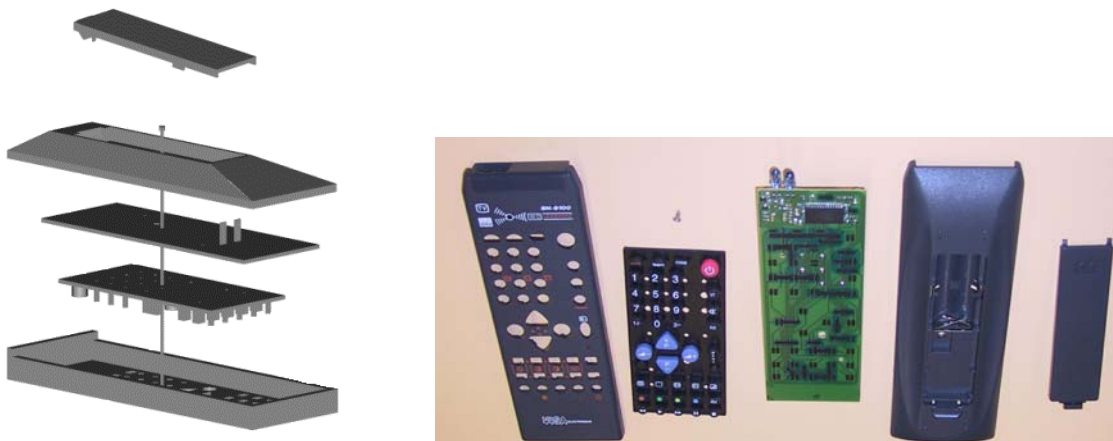


Figura A.2: Elementos de un mando a distancia.

En Figura A.3 se muestra cómo quedaría el entorno tras añadir los componentes necesarios para realizar el diseño del modelo relacional del mando a distancia.

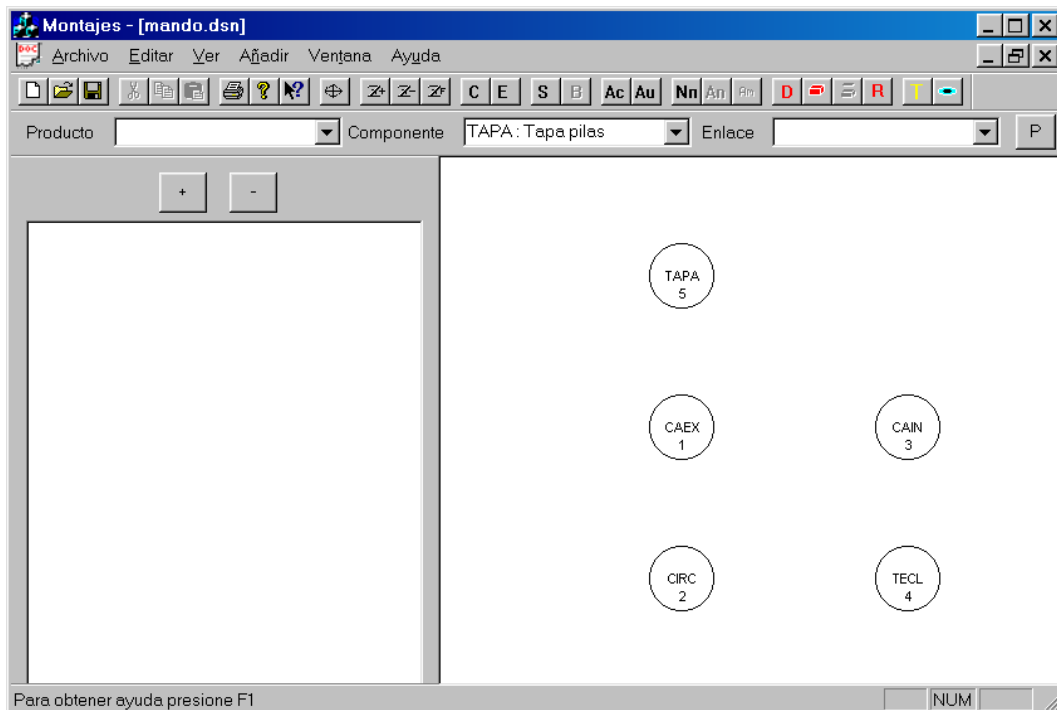


Figura A.3: Diseño del modelo relacional de un mando a distancia; componentes.

Para ello, se han añadido los cinco componentes que lo forman: la carcasa exterior, la interior, la placa con la circuitería, la placa de las teclas y por último la tapa de las pilas.

### A.1.2. Añadir elementos de enlace

Para añadir un elemento de enlace hay que proceder de manera similar a como se ha introducido un componente, es decir, primero seleccionar el elemento de enlace de la lista correspondiente; tras esto se pulsa sobre el botón de añadir elemento de enlace; se termina pulsando sobre la zona correspondiente al modelo relacional para añadir tantos elementos de enlace del tipo actual como se desee.

Igual que para los componentes, si se desea introducir otro elemento de enlace distinto, se deben repetir todos los pasos descritos para la introducción de un elemento de enlace.

Continuando con el ejemplo anterior del mando a distancia, ahora se añaden los elementos de enlace; en este caso sólo existe un tornillo que une los otros componentes excepto la tapa de las pilas (Figura A.4).

### A.1.3. Añadir uniones

En este caso se dispone de dos posibilidades: añadir una unión de cierre o de contacto, dado que la manera de realizarlo es similar en ambos casos se explica cómo introducir una unión de cierre y a continuación se explicarán las diferencias, si lo que se desea es introducir una unión de contacto.

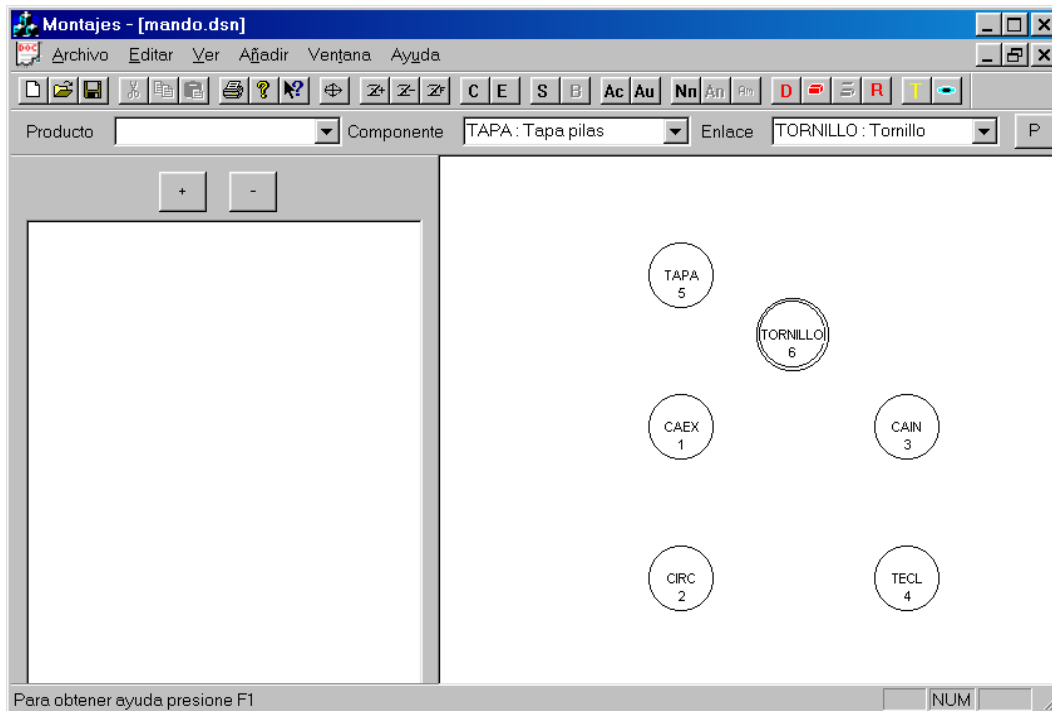


Figura A.4: Diseño del modelo relacional de un mando a distancia; elementos de enlace.

Para la introducción de una unión de cierre, lo primero a considerar es su representación en el grafo; tal como se ha visto en el apartado 2.3.1.2, se corresponde con una línea continua que une los dos nodos, los componentes, que poseen la unión de cierre. Por lo tanto lo primero es seleccionar el botón correspondiente a añadir unión de cierre; tras esto se está en disposición de añadirla, para lo cual se procede pinchando con el ratón sobre uno de los componentes que están involucrados en la unión, en la zona del modelo relacional, y posteriormente sobre el otro, de tal manera que aparecerá una línea continua uniendo ambos componentes.

Para introducir más uniones de cierre se procederá repitiendo los pasos descritos.

Si por el contrario, lo que se desea es introducir una unión de contacto, que se representa mediante una línea discontinua, se procede de igual manera, excepto que, en lugar de pulsar sobre el botón de añadir unión de cierre, se pulsará el botón de añadir unión de contacto, pinchando a continuación sobre los nodos que tienen la unión.

Siguiendo con el diseño del mando a distancia, se han añadido las correspondientes uniones de cierre y de contacto entre los componentes (Figura A.5), donde se puede apreciar que el tornillo une los cuatro elementos principales del producto, dejando la tapa de las pilas separada, unida sólo a la carcasa exterior. De igual manera la unión existente entre las teclas, la circuitería y las carcasas son de contacto ya que, en el momento en que se separa la carcasa exterior de la interior, se tienen desensamblados los componentes.

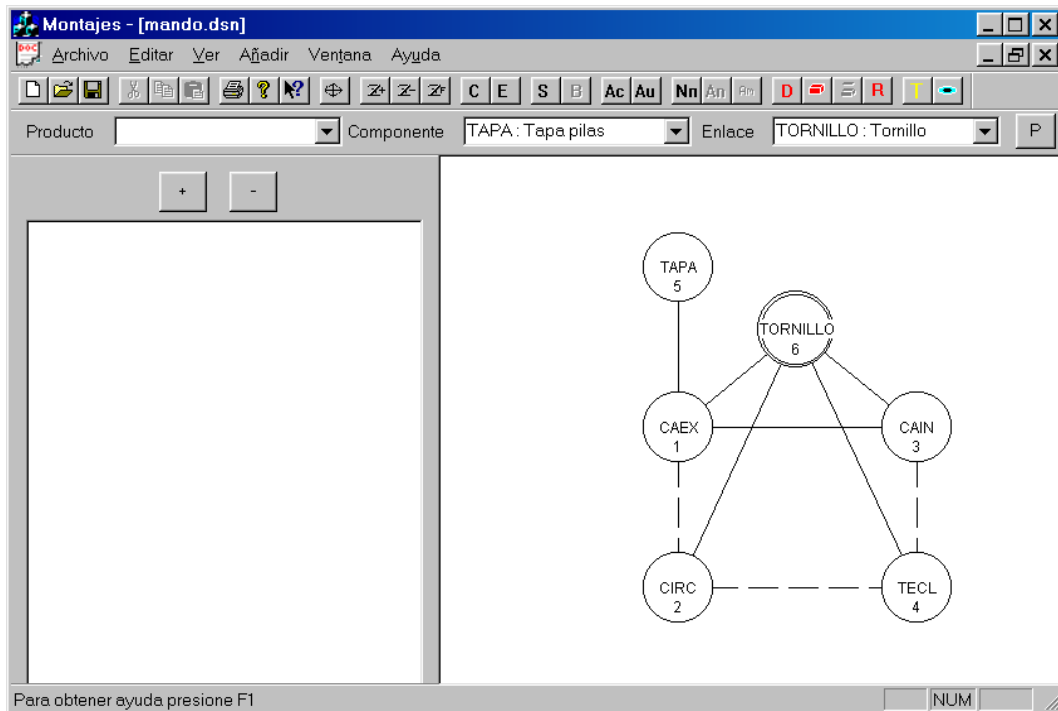


Figura A.5: Diseño del modelo relacional de un mando a distancia; uniones de cierre y de contacto.

#### A.1.4. Añadir elementos a un montaje nuevo

Para poder añadir un montaje se necesita que existan al menos dos nodos y que estén conectados mediante una unión, ya que un conjunto de nodos que no se encuentran unidos mediante uniones no pueden formar un montaje, ya que se trata de elementos independientes.

Una vez se tienen introducidos los componentes y elementos de enlace que forman parte del producto y definidas las relaciones entre ellos, se pasa a la definición de los montajes; para ello lo primero es pulsar sobre el botón de nuevo montaje, con lo que el entorno pedirá una descripción del mismo; esta descripción tiene carácter meramente informativo para el operario que está introduciendo los datos, de manera que, sea más fácil reconocer rápidamente cada uno de los montajes. Una vez seleccionada la opción de introducir un nuevo montaje, el entorno transformará todos los nodos del grafo en un color claro, activando la opción de añadir nodos a montaje, de tal manera que, según se vaya pinchando con el ratón sobre los distintos nodos, éstos se van introduciendo en el montaje y además los nodos cambian de color al introducirlos, con lo que se puede distinguir rápidamente qué nodos pertenecen o no al montaje (Figura A.6).

El entorno añade todos los nodos sobre los que se pinche al montaje actual, mientras no se seleccione otra opción.

Para introducir otro montaje hay que repetir los pasos indicados. Además, en la zona de montajes del producto, van apareciendo los nuevos montajes con sus correspondientes componentes.

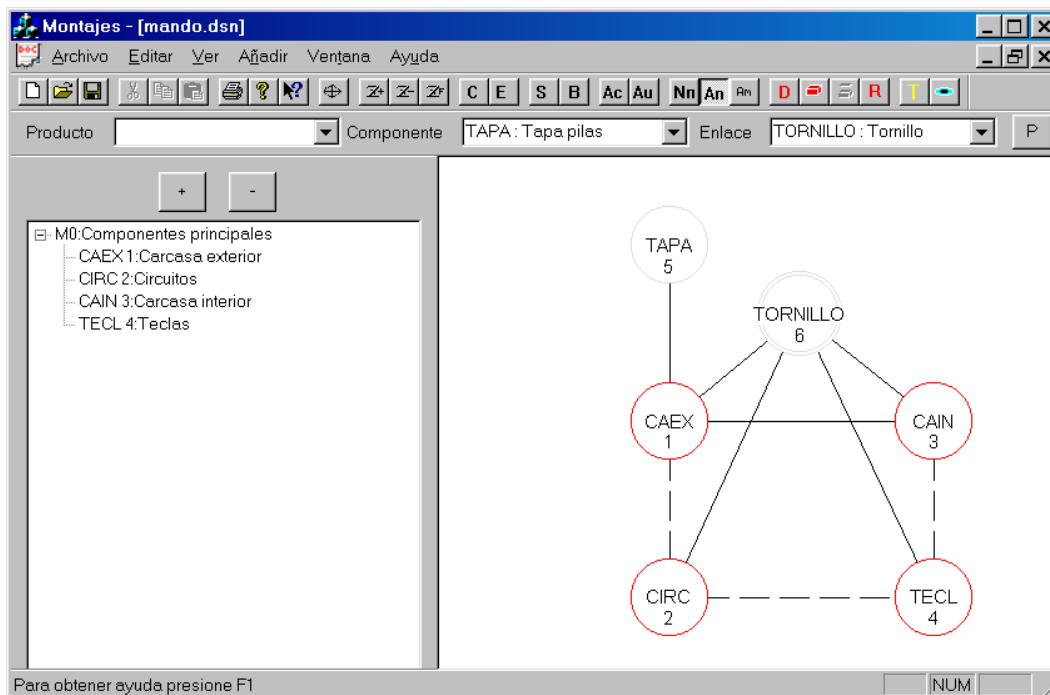


Figura A.6: Diseño del modelo relacional de un mando a distancia; introducción de componentes en los montajes.

#### A.1.5. Añadir elementos a un montaje existente

Si se dispone de un montaje ya creado y se desea añadirle nuevos elementos, ya sean componentes o elementos de enlace, bien porque se ha cambiado el modelo o por otras razones, se debe proceder de la siguiente manera:

Primero se selecciona el montaje que se quiere modificar; mediante el botón de seleccionar, se activa este modo de trabajo, y se pulsa sobre el montaje correspondiente, que se presenta en la zona de montajes; tras realizar esto, los nodos pertenecientes al montaje cambiarán de color.

Una vez seleccionado el montaje, se pulsa sobre el botón de añadir montaje; de esta manera, en lugar de añadir uno nuevo, se va a modificar el actual. Tras esto, se pulsará sobre los nodos que se desean añadir al montaje, que irán cambiando de color para indicar su pertenencia al mismo, de igual manera que se explicaba en el punto anterior.

#### A.1.6. Añadir montajes a otro montaje

Para poder añadir un montaje a otro montaje, hay que tener en cuenta que un montaje no puede contener a otro montaje en el cual esté contenido, ya que entonces, sería imposible decidir cuál es el montaje básico de los dos. Estos aspectos los tiene en cuenta el entorno, de manera que, si se pretende introducir un montaje en otro y no cumplen las condiciones adecuadas para ello, no se permite.

A la hora de añadir un montaje a otros, se puede realizar bien cuando se está añadiendo un nuevo montaje o cuando se está modificando uno existente. En ambos casos, una vez se tiene seleccionado el montaje y se

están introduciendo los elementos que lo forman, si se pulsa sobre un montaje, en la zona de montajes, aparece de un color distinto en la zona del grafo; con lo que pulsando sobre el botón de añadir montaje a montaje, el seleccionado se añade al montaje actual.

De esta manera se añaden montajes a otros montajes, repitiendo los pasos hasta conseguir las relaciones deseadas entre los montajes.

Para el ejemplo del mando a distancia, el resultado de introducir los montajes correspondientes se puede ver en la Figura A.7.

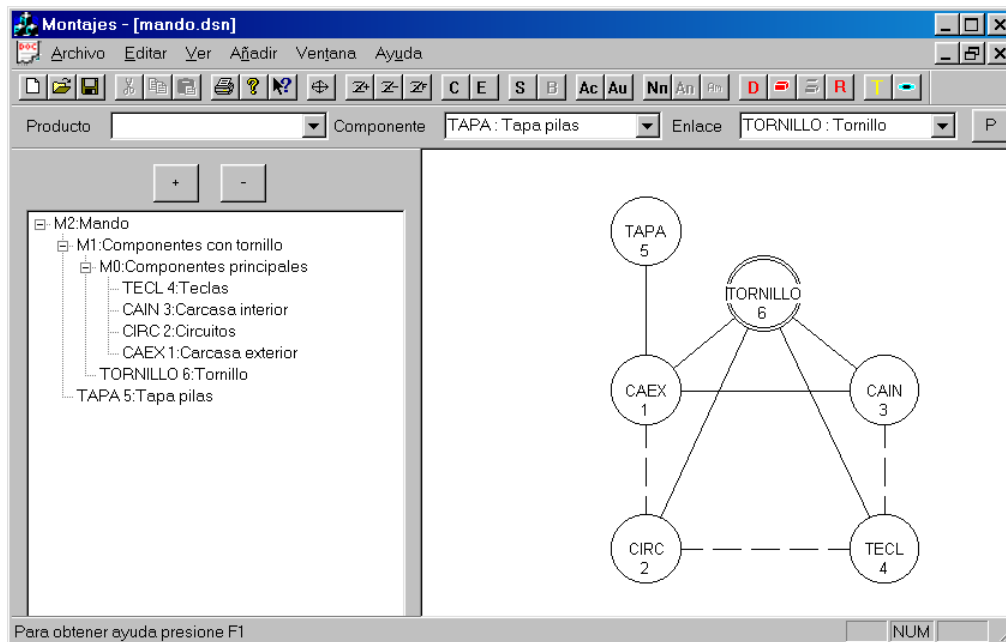


Figura A.7: Diseño del modelo relacional de un mando a distancia; modelo final.

## A.2. Secuencia de desensamblado

Una vez se ha diseñado el modelo relacional del producto, se puede proceder al cálculo de la secuencia de desensamblado deseada; para ello se selecciona el componente que se quiere desensamblar y se pulsa sobre el botón de secuencia de desensamblado, con lo que se obtiene una ventana de información sobre las acciones a realizar, expresadas según las genera el sistema y además se muestra la secuencia en un grafo ordenado, de manera que, es más sencillo observar las operaciones que se pueden realizar en paralelo.

Para el mando a distancia, se tiene que el entorno genera la ventana de información de la Figura A.8 para el componente correspondiente a la carcasa interior, componente  $c_3$ .

Esta secuencia se corresponde con las siguientes acciones:

1. Como  $m_1 \in m_2$  entonces hay que:
  - a. Eliminar unión y separar el montaje  $m_1$ , "componentes con tornillo", del componente  $c_5$ , "tapa pilas".
2. Como  $m_0 \in m_1$  entonces hay que:
  - a. Quitar el elemento de enlace  $c_6$ , "tornillo".

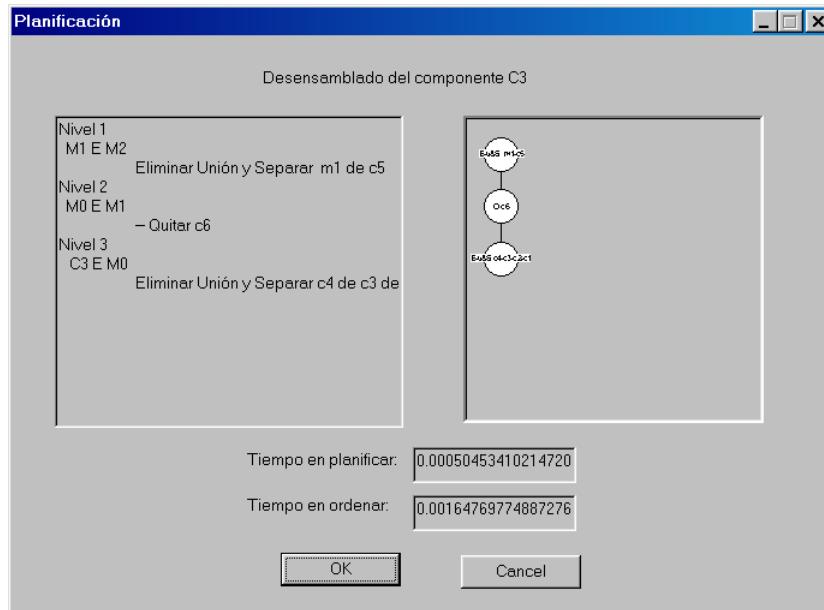


Figura A.8: Secuencia de desensamblado de la carcasa exterior de un mando a distancia.

3. Como  $c_3 \in m_0$  entonces hay que:
  - a. Eliminar unión y separar los componentes  $c_4$ , "teclas";  $c_3$ , "carcasa interior";  $c_2$ , "circuitos";  $c_1$ , "carcasa exterior".

Además de la planificación, el entorno informa sobre el tiempo empleado en calcularla, así como el tiempo empleado en ordenar las acciones para realizar el desensamblado con un solo robot, teniendo en cuenta la minimización de los cambios de herramientas entre operaciones. En este caso resulta:

1. Eliminar unión y separar  $m_1$  de  $c_5$ .
2. Quitar  $c_6$ .
3. Eliminar unión y separar  $c_4$  de  $c_3$  de  $c_2$  y de  $c_1$ .

### A.3. Simulación de la secuencia de desensamblado

Tras realizar el cálculo de la secuencia de desensamblado de un componente, se puede pasar a la simulación gráfica de la misma, en la cual se puede apreciar cómo el robot realiza el desensamblado. Para ello se selecciona la opción de ver representación gráfica, con lo que el entorno cambiará, la parte correspondiente al modelo relacional por un modelo del producto que se está intentando desensamblar (Figura A.9).

Hay que indicar que el sistema parte del objeto posicionado sobre una mesa de trabajo, lo que añade restricciones extras al generador de trayectorias de desensamblado.

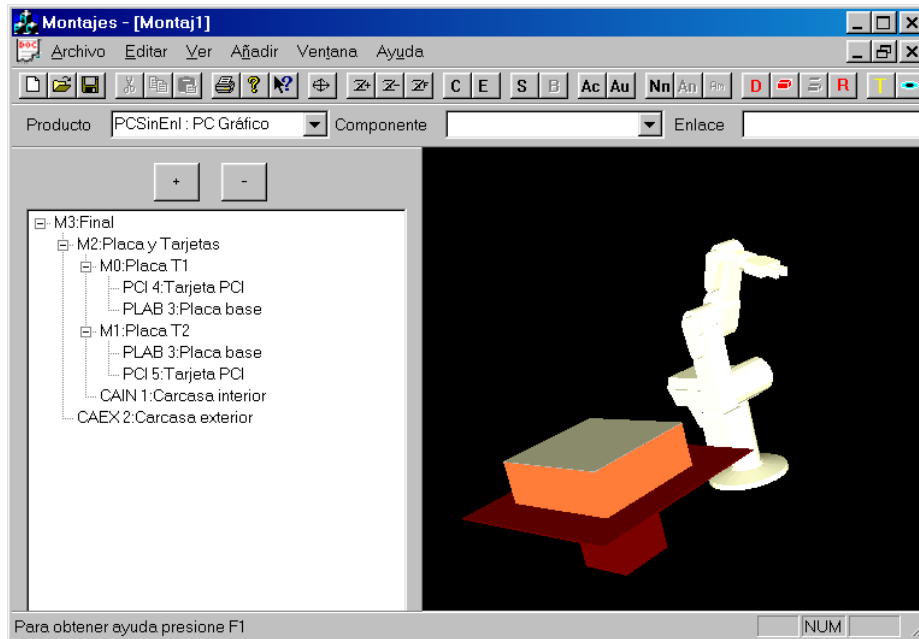


Figura A.9: Entorno de simulación, modelo de un PC sobremesa.

Una vez se tiene el modelo, pulsando sobre el botón de simular acción, el entorno mostrará como el robot accede hasta el componente y cuáles son los movimientos necesarios para realizar el desensamblado, siendo estos reproducidos en la simulación. El cálculo de los movimientos de desensamblado, para tener en cuenta las nuevas variables, se trata en el siguiente apartado.

De esta manera se simula cada movimiento a realizar por el brazo robot para desensamblar un componente, por lo tanto hay que repetir esta operación para cada uno de los componentes que se requieran desensamblar, según la secuencia de desensamblado obtenida, para ver la secuencia completa.

En este caso se está calculando cada una de las trayectorias de movimiento, según se realiza el desensamblado de cada componente, lo que implica un coste computacional elevado en caso de muchos elementos en el producto. Por tanto, una vez realizada la simulación de todos los movimientos, el sistema permite almacenarla, de tal manera que, si se desea ver la secuencia completa de movimientos de manera seguida, se puede hacer de una manera rápida evitando todo el cálculo intermedio.

Además existe otra posibilidad: si se desea ver los movimientos, evitando la aparición del brazo robot, se puede desactivar pulsando sobre el botón ver robot, que muestra o no el robot en el momento de la simulación.



## Anexo B

# Cinemática del brazo robot utilizado

En este anexo se incluyen los cálculos para la cinemática directa e inversa del robot de utilizado para las pruebas prácticas del sistema de desensamblado automático. Este robot es el modelo Scorbot ER-IX de Intelitek, el cual posee cinco grados de libertad y una pinza paralela de dos dedos. Además cabe destacar que todas sus articulaciones son de rotación.

Para el cálculo cinemático se parte del modelo del robot mostrado en la Figura B.1.

Según el esquema del robot se calculan sus parámetros de Denavit-Hartenberg que son:

	$\theta$	$d$	$a$	$\alpha$
1	$q_1$	$l_1$	$l_2$	$-\pi/2$
2	$q_2$	$-l_3$	$l_4$	0
3	$q_3$	0	$l_5$	0
4	$q_4$	0	0	$-\pi/2$
5	$q_5$	$l_6$	0	0

Donde los valores de los parámetros de longitud, correspondientes a los distintos eslabones, son:

$$\begin{aligned} l_1 &= 392.5 \text{ mm} & l_2 &= 90.0 \text{ mm} & l_3 &= 35.0 \text{ mm} \\ l_4 &= 280.0 \text{ mm} & l_5 &= 230.0 \text{ mm} & l_6 &= 245.5 \text{ mm} \end{aligned} \tag{B.1}$$

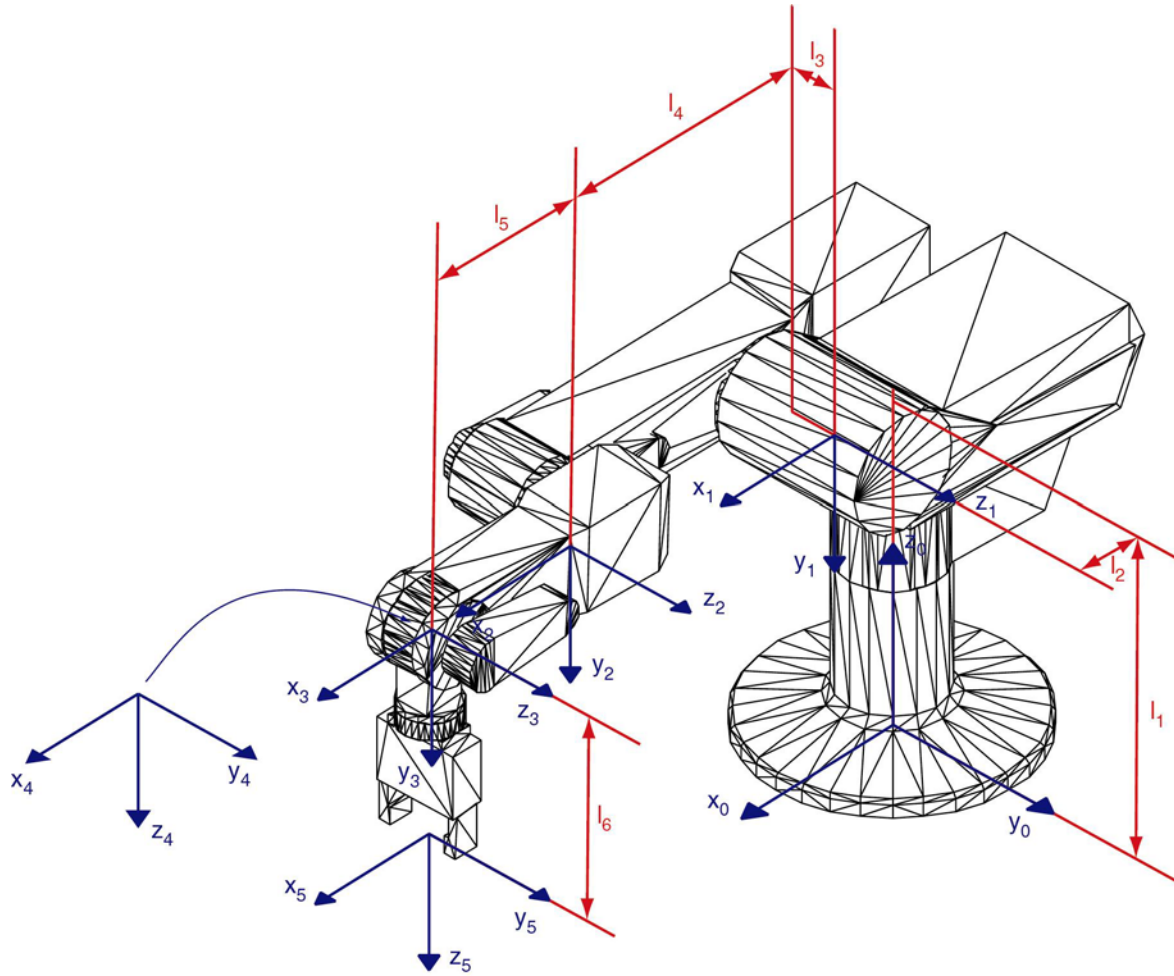


Figura B.1: Ejes de coordenadas del robot Scorbobot ER-IX.

Utilizando estos parámetros se obtiene la cinemática directa del robot, que viene dada por los siguientes valores para los elementos de la matriz  ${}_{base}T_{extremo}$ :

$$\begin{aligned} x_{x_{base}^{extremo}} = & \cos q_1 \cdot \cos q_2 \cdot \cos q_4 \cdot \cos q_5 - \cos q_1 \cdot \sin q_2 \cdot \sin q_3 \cdot \cos q_4 \cdot \cos q_5 - \\ & - \cos q_1 \cdot \cos q_2 \cdot \sin q_3 \cdot \sin q_4 \cdot \cos q_5 - \\ & - \cos q_1 \cdot \sin q_2 \cdot \cos q_3 \cdot \sin q_4 \cdot \cos q_5 + \sin q_1 \cdot \sin q_5 \end{aligned} \quad (B.2)$$

$$\begin{aligned} y_{y_{base}^{extremo}} = & \sin q_1 \cdot \cos q_2 \cdot \cos q_3 \cdot \cos q_4 \cdot \cos q_5 - \sin q_1 \cdot \sin q_2 \cdot \sin q_3 \cdot \cos q_4 \cdot \\ & \cdot \cos q_5 - \sin q_1 \cdot \cos q_2 \cdot \sin q_3 \cdot \sin q_4 \cdot \cos q_5 - \\ & - \sin q_1 \cdot \cos q_2 \cdot \cos q_3 \cdot \sin q_4 \cdot \cos q_5 - \cos q_1 \cdot \sin q_5 \end{aligned} \quad (B.3)$$

$$\begin{aligned} z_{z_{base}^{extremo}} = & \sin q_2 \cdot \cos q_3 \cdot \cos q_4 \cdot \cos q_5 + \cos q_2 \cdot \sin q_3 \cdot \cos q_4 \cdot \cos q_5 - \\ & - \sin q_2 \cdot \sin q_3 \cdot \sin q_4 \cdot \cos q_5 + \cos q_2 \cdot \cos q_3 \cdot \sin q_4 \cdot \cos q_5 \end{aligned} \quad (B.4)$$

$$\begin{aligned} x_{y_{base}^{extremo}} = & -\cos q_1 \cdot \cos q_2 \cdot \cos q_3 \cdot \cos q_4 \cdot \sin q_5 + \cos q_1 \cdot \sin q_2 \cdot \sin q_3 \cdot \\ & \cdot \cos q_4 \cdot \sin q_5 + \cos q_1 \cdot \cos q_2 \cdot \sin q_3 \cdot \sin q_4 \cdot \sin q_5 + \cos q_1 \cdot \\ & \cdot \sin q_2 \cdot \cos q_3 \cdot \sin q_4 \cdot \sin q_5 + \sin q_1 \cdot \cos q_5 \end{aligned} \quad (B.5)$$

$$y_{base}^{extremo} = -\text{sen } q_1 \cdot \text{cos } q_2 \cdot \text{cos } q_3 \cdot \text{cos } q_4 \cdot \text{sen } q_5 + \text{sen } q_1 \cdot \text{sen } q_2 \cdot \text{sen } q_3 \cdot \text{cos } q_4 \cdot \text{sen } q_5 + \text{sen } q_1 \cdot \text{cos } q_2 \cdot \text{sen } q_3 \cdot \text{sen } q_4 \cdot \text{sen } q_5 + \text{sen } q_1 \cdot \text{sen } q_2 \cdot \text{cos } q_3 \cdot \text{sen } q_4 \cdot \text{sen } q_5 - \text{cos } q_1 \cdot \text{cos } q_5 \quad (\text{B.6})$$

$$z_{base}^{extremo} = -\text{sen } q_2 \cdot \text{cos } q_3 \cdot \text{cos } q_4 \cdot \text{sen } q_5 - \text{cos } q_2 \cdot \text{sen } q_3 \cdot \text{cos } q_4 \cdot \text{sen } q_5 + \text{sen } q_2 \cdot \text{sen } q_3 \cdot \text{sen } q_4 \cdot \text{sen } q_5 + \text{cos } q_2 \cdot \text{cos } q_3 \cdot \text{sen } q_4 \cdot \text{sen } q_5 \quad (\text{B.7})$$

$$x_{base}^{extremo} = -\text{cos } q_1 \cdot \text{cos } q_2 \cdot \text{cos } q_3 \cdot \text{sen } q_4 + \text{cos } q_1 \cdot \text{sen } q_2 \cdot \text{sen } q_3 \cdot \text{sen } q_4 - \text{cos } q_1 \cdot \text{cos } q_2 \cdot \text{sen } q_3 \cdot \text{cos } q_4 - \text{cos } q_1 \cdot \text{sen } q_2 \cdot \text{cos } q_3 \cdot \text{cos } q_4 \quad (\text{B.8})$$

$$y_{base}^{extremo} = -\text{sen } q_1 \cdot \text{cos } q_2 \cdot \text{cos } q_3 \cdot \text{sen } q_4 + \text{sen } q_1 \cdot \text{sen } q_2 \cdot \text{sen } q_3 \cdot \text{sen } q_4 - \text{sen } q_1 \cdot \text{cos } q_2 \cdot \text{sen } q_3 \cdot \text{cos } q_4 - \text{sen } q_1 \cdot \text{sen } q_2 \cdot \text{cos } q_3 \cdot \text{cos } q_4 \quad (\text{B.9})$$

$$z_{base}^{extremo} = -\text{sen } q_2 \cdot \text{cos } q_3 \cdot \text{sen } q_4 - \text{cos } q_2 \cdot \text{sen } q_3 \cdot \text{sen } q_4 - \text{sen } q_2 \cdot \text{sen } q_3 \cdot \text{cos } q_4 + \text{cos } q_2 \cdot \text{cos } q_3 \cdot \text{cos } q_4 \quad (\text{B.10})$$

$$x_{base}^{extremo} = l_5 \cdot \text{cos } q_1 \cdot \text{cos } q_2 \cdot \text{cos } q_3 - l_5 \cdot \text{cos } q_1 \cdot \text{sen } q_2 \cdot \text{sen } q_3 + l_4 \cdot \text{cos } q_1 \cdot \text{cos } q_2 + l_3 \cdot \text{sen } q_1 + l_2 \cdot \text{cos } q_1 + l_6 \cdot (-\text{cos } q_1 \cdot \text{cos } q_2 \cdot \text{cos } q_3 \cdot \text{sen } q_4 + \text{cos } q_1 \cdot \text{sen } q_2 \cdot \text{sen } q_3 \cdot \text{sen } q_4 - \text{cos } q_1 \cdot \text{cos } q_2 \cdot \text{sen } q_3 \cdot \text{cos } q_4 - \text{cos } q_1 \cdot \text{sen } q_2 \cdot \text{cos } q_3 \cdot \text{cos } q_4) \quad (\text{B.11})$$

$$y_{base}^{extremo} = l_5 \cdot \text{sen } q_1 \cdot \text{cos } q_2 \cdot \text{cos } q_3 - l_5 \cdot \text{sen } q_1 \cdot \text{sen } q_2 \cdot \text{sen } q_3 + l_4 \cdot \text{sen } q_1 \cdot \text{cos } q_2 - l_3 \cdot \text{cos } q_1 + l_2 \cdot \text{sen } q_1 + l_6 \cdot (-\text{sen } q_1 \cdot \text{cos } q_2 \cdot \text{cos } q_3 \cdot \text{sen } q_4 + \text{sen } q_1 \cdot \text{sen } q_2 \cdot \text{sen } q_3 \cdot \text{sen } q_4 - \text{sen } q_1 \cdot \text{cos } q_2 \cdot \text{sen } q_3 \cdot \text{cos } q_4 - \text{sen } q_1 \cdot \text{sen } q_2 \cdot \text{cos } q_3 \cdot \text{cos } q_4) \quad (\text{B.12})$$

$$z_{base}^{extremo} = l_5 \cdot \text{sen } q_2 \cdot \text{cos } q_3 + l_5 \cdot \text{cos } q_2 \cdot \text{sen } q_3 + l_4 \cdot \text{sen } q_2 + l_1 + l_6 \cdot (-\text{sen } q_2 \cdot \text{cos } q_3 \cdot \text{sen } q_4 - \text{cos } q_2 \cdot \text{sen } q_3 \cdot \text{sen } q_4 - \text{sen } q_2 \cdot \text{sen } q_3 \cdot \text{cos } q_4 + \text{cos } q_2 \cdot \text{cos } q_3 \cdot \text{cos } q_4) \quad (\text{B.13})$$

De estas ecuaciones se obtienen los valores articulares según el modelo cinemático inverso, por lo tanto:

$$q_1 = \arctan \frac{y_{base}^{extremo}}{x_{base}^{extremo}} \quad (\text{B.14})$$

Esto será cierto siempre y cuando  $x_{base}^{extremo}$  e  $y_{base}^{extremo}$  no sean cero simultáneamente, en ese caso el valor de  $q_1$  se puede despejar de la ecuación:

$$-x_{base}^{extremo} \text{sen } q_1 + y_{base}^{extremo} \text{cos } q_1 = -l_3 \quad (\text{B.15})$$

Donde despejando  $q_1$  se obtiene:

$$q_1 = \arccos \frac{-l_3 \pm \sqrt{l_3^2 - ((y_{\text{base}}^{\text{extremo}})^2 - (x_{\text{base}}^{\text{extremo}})^2)(l_3^2 - (x_{\text{base}}^{\text{extremo}})^2)}}{(y_{\text{base}}^{\text{extremo}})^2 - (x_{\text{base}}^{\text{extremo}})^2} \quad (\text{B.16})$$

Aplicando el algoritmo de Pieper [207], para realizar el cálculo de la cinemática inversa, se desacoplan las tres primeras articulaciones del resto para obtener los valores de  $q_2$ ,  $q_3$ . Para ello la matriz del extremo de la ecuación 4.1 se varía de tal manera que la nueva posición del extremo a considerar es:

$$\mathbf{p}_{\text{extremodescoplado}} = \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \begin{bmatrix} x_{\text{extremo}}^{\text{base}} - l_6 \cdot x_{\text{extremo}}^{\text{base}} \\ y_{\text{extremo}}^{\text{base}} - l_6 \cdot y_{\text{extremo}}^{\text{base}} \\ z_{\text{extremo}}^{\text{base}} - l_6 \cdot z_{\text{extremo}}^{\text{base}} \\ 1 \end{bmatrix} \quad (\text{B.17})$$

Para calcular los valores de  $q_2$  y  $q_3$  se utiliza el método geométrico. Para ello se tiene en consideración el esquema del robot de la Figura B.2.

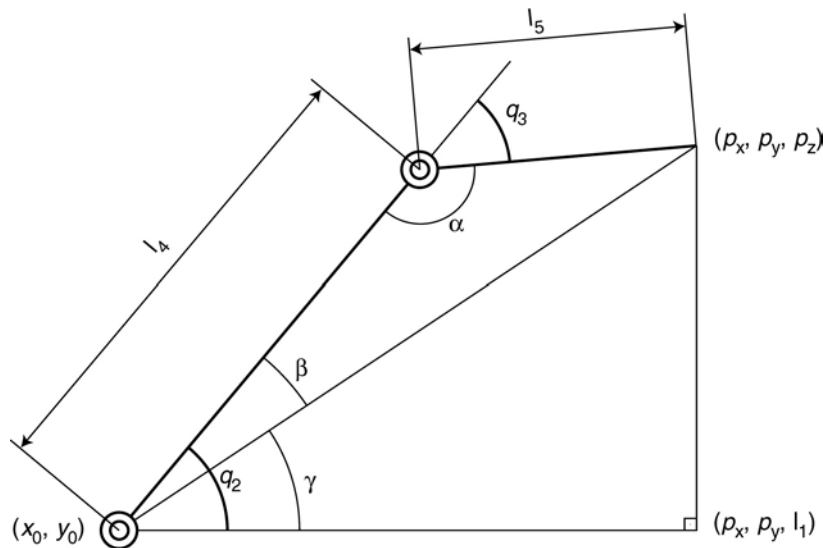


Figura B.2: Esquema del plano de trabajo del robot.

En el que el punto  $(x_0, y_0)$  toma el valor:

$$(x_0, y_0) = (l_2 \cdot \cos q_1 + l_3 \cdot \sin q_1, l_2 \cdot \sin q_1 - l_3 \cdot \cos q_1) \quad (\text{B.18})$$

De esta manera se obtiene el valor de  $q_3$  teniendo en cuenta el teorema del coseno como:

$$q_3 = \arccos \left( \frac{p_x^2 + p_y^2 + (p_z - l_1)^2 - l_4^2 - l_5^2}{2 \cdot l_4 \cdot l_5} \right) \quad (\text{B.19})$$

De manera similar se obtiene el valor de  $q_2$ , mediante la suma de los ángulos  $\beta$  y  $\gamma$ , donde  $\gamma$  se calcula en base a su tangente y  $\beta$  utilizando el seno. Con lo que el  $q_2$  queda como:

$$q_2 = \arctan\left(\frac{p_z - l_1}{\sqrt{p_x^2 + p_y^2}}\right) + \arcsen\left(\frac{l_4^2 - l_5^2 + p_x^2 + p_y^2 + (p_z - l_1)^2}{2 \cdot l_4 \cdot \sqrt{p_x^2 + p_y^2 + (p_z - l_1)^2}}\right) \quad (\text{B.20})$$

De esta manera se conocen los valores de las tres primeras articulaciones, con lo que sólo falta calcular el valor de las dos últimas  $q_4$  y  $q_5$ . Para ello se utiliza el valor de orientación del extremo así como la solución obtenida para  $q_1$ ,  $q_2$ , y  $q_3$ . Con lo que se plantea:

$${}^3\mathbf{Rot}_5 = ({}^3\mathbf{Rot}_5)^{-1} \cdot {}^0\mathbf{Rot}_5 \quad (\text{B.21})$$

Donde se calculan los valores de las articulaciones  $q_4$  y  $q_5$ . De esta manera la articulación  $q_4$  tiene el valor siguiente:

$$q_4 = \arccos\left(-x_{z_{\text{extremo}}^{\text{base}}} \cdot \cos q_1 \cdot (\cos q_2 \cdot \sin q_3 + \sin q_2 \cdot \cos q_3) - y_{z_{\text{extremo}}^{\text{base}}} \cdot \sin q_1 \cdot (\cos q_2 \cdot \sin q_3 + \sin q_2 \cdot \cos q_3) - z_{z_{\text{extremo}}^{\text{base}}} \cdot (\sin q_2 \cdot \sin q_3 - \cos q_2 \cdot \cos q_3)\right) \quad (\text{B.22})$$

Siguiendo criterios similares que en el caso anterior se obtiene el valor de  $q_5$  como:

$$q_5 = \arcsen\left(x_{x_{\text{extremo}}^{\text{base}}} \cdot \sin q_1 - y_{x_{\text{extremo}}^{\text{base}}} \cdot \cos q_1\right) \quad (\text{B.23})$$

## Anexo C

# Posicionamiento del producto.

Para considerar el posicionamiento del producto sobre la mesa de trabajo hay que utilizar como posición del extremo, en lugar de la correspondiente a la ecuación 4.1, la expresión 4.6 y calcularlo en función de  $d_x$ ,  $d_y$  y  $\theta$ .

A la hora de realizar los cálculos se aplican sobre el sistema real disponible en el laboratorio, sobre el que se realizan las pruebas de desensamblado. Esto implica un valor de la matriz  ${}^{base}\mathbf{T}_{\text{mesa}}$ :

$${}^{base}\mathbf{T}_{\text{mesa}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -700 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{C.1})$$

De esta manera para la ecuación 4.6 se tiene obtiene:

$${}^{base}\mathbf{T}'_{\text{extremo}} = \begin{bmatrix} x_{\text{base}}^{\text{extremo}} \cdot \cos \theta - y_{\text{base}}^{\text{extremo}} \cdot \text{sen} \theta & x_{\text{base}}^{\text{extremo}} \cdot \cos \theta - y_{\text{base}}^{\text{extremo}} \cdot \text{sen} \theta & & \\ x_{\text{base}}^{\text{extremo}} \cdot \text{sen} \theta + y_{\text{base}}^{\text{extremo}} \cdot \cos \theta & x_{\text{base}}^{\text{extremo}} \cdot \text{sen} \theta + y_{\text{base}}^{\text{extremo}} \cdot \cos \theta & & \\ & z_{\text{base}}^{\text{extremo}} & & z_{\text{base}}^{\text{extremo}} \\ & 0 & & 0 \end{bmatrix} \quad (\text{C.2})$$

$$\left[ \begin{array}{cc} x_{\text{base}}^{\text{extremo}} \cdot \cos \theta - y_{\text{base}}^{\text{extremo}} \cdot \text{sen} \theta & x_{\text{base}}^{\text{extremo}} \cdot \cos \theta - y_{\text{base}}^{\text{extremo}} \cdot \text{sen} \theta + d_x - 700 \cdot \text{sen} \theta \\ x_{\text{base}}^{\text{extremo}} \cdot \text{sen} \theta + y_{\text{base}}^{\text{extremo}} \cdot \cos \theta & x_{\text{base}}^{\text{extremo}} \cdot \text{sen} \theta + y_{\text{base}}^{\text{extremo}} \cdot \cos \theta + d_y + 700 \cdot \cos \theta - 700 \\ z_{\text{base}}^{\text{extremo}} & z_{\text{base}}^{\text{extremo}} \\ 0 & 1 \end{array} \right]$$

En base a esta ecuación hay que calcular los valores que hacen posible el posicionamiento del robot en el lugar adecuado para coger el componente.

Para ello se parte del supuesto en que la mesa se encuentra rotada un ángulo  $\theta$ , lo que se realizará de manera iterativa para los posibles giros de la

mesa que se pueden realizar. En base a esta información se calcula el valor que debe tomar la articulación  $q_1$ , de tal manera que  $q_1$  vale:

$$q_1 = \arctan\left(\frac{x_{z_{base}^{extremo}} \cdot \text{sen } \theta + y_{z_{base}^{extremo}} \cdot \text{cos } \theta}{x_{z_{base}^{extremo}} \cdot \text{cos } \theta - y_{z_{base}^{extremo}} \cdot \text{sen } \theta}\right) \quad (\text{C.3})$$

Con esta ecuación se calcula si el  $\cos(q_1)=0$ . Si es cierto entonces utilizando la ecuación B.15 y C.2 se calcula el valor que tiene que tomar la variable  $d_x$  como:

$$d_x = -x_{base}^{extremo} \cdot \text{cos } \theta + y_{base}^{extremo} \cdot \text{sen } \theta + 700 \cdot \text{sen } \theta + \frac{l_3}{\text{sen } q_1} \quad (\text{C.4})$$

Una vez conocido el valor del desplazamiento en el eje x hay que calcular el desplazamiento en el eje y ( $d_y$ ). En este caso aparecen tres aproximaciones distintas dependiendo del valor del discriminante definido por:

$$\Delta_1 = y_2^2 - 4 \cdot y_1 \cdot y_3 \quad (\text{C.5})$$

Donde los valores de  $y_1$ ,  $y_2$  e  $y_3$  están definidos por:

$$y_1 = 1 \quad (\text{C.6})$$

$$y_2 = -2 \cdot l_6 \cdot (x_{z_{base}^{extremo}} \cdot \text{cos } \theta - y_{z_{base}^{extremo}} \cdot \text{sen } \theta) - 2 \cdot y_0 \quad (\text{C.7})$$

$$\begin{aligned} y_3 = & \left(x_{base}^{extremo} \cdot \text{cos } \theta - y_{base}^{extremo} \cdot \text{sen } \theta + d_x - 700 \cdot \text{sen } \theta\right)^2 + \\ & + l_6^2 \left(x_{z_{base}^{extremo}} \cdot \text{cos } \theta - y_{z_{base}^{extremo}} \cdot \text{sen } \theta\right)^2 - 2 \cdot l_6 \left(x_{z_{base}^{extremo}} \cdot \text{cos } \theta - y_{z_{base}^{extremo}} \cdot \text{sen } \theta\right) \cdot \\ & \cdot \left(x_{base}^{extremo} \cdot \text{cos } \theta - y_{base}^{extremo} \cdot \text{sen } \theta + d_x - 700 \cdot \text{sen } \theta\right) + x_0^2 - \\ & - 2 \cdot x_0 \cdot \left(x_{base}^{extremo} \cdot \text{cos } \theta - y_{base}^{extremo} \cdot \text{sen } \theta + d_x - 700 \cdot \text{sen } \theta\right) + \\ & + 2 \cdot l_6 \cdot x_0 \cdot \left(x_{z_{base}^{extremo}} \cdot \text{cos } \theta - y_{z_{base}^{extremo}} \cdot \text{sen } \theta\right) + l_6^2 \cdot \left(x_{z_{base}^{extremo}} \cdot \text{sen } \theta + y_{z_{base}^{extremo}} \cdot \text{cos } \theta\right)^2 + y_0^2 + \\ & + 2 \cdot l_6 \cdot y_0 \cdot \left(x_{z_{base}^{extremo}} \cdot \text{sen } \theta + y_{z_{base}^{extremo}} \cdot \text{cos } \theta\right) - l_4^2 - l_5^2 + 2 \cdot l_4 \cdot l_5 + \left(p_z - l_6 \cdot z_{z_{base}^{extremo}} - l_1\right)^2 \end{aligned} \quad (\text{C.8})$$

Las dos primeras opciones aparecen cuando el valor del discriminante  $\Delta_1$  es menor que cero, en cuyo caso puede que cualquier valor para  $d_y$  sea válido o que no exista solución posible. La diferenciación entre ambos casos se realiza considerando la ecuación de segundo grado:

$$y_1 \cdot (y_{base}^{extremo'})^2 + y_2 \cdot y_{base}^{extremo'} + y_3 = 0 \quad (\text{C.9})$$

Donde:

$$y_{base}^{extremo'} = x_{base}^{extremo} \cdot \text{sen } \theta + y_{base}^{extremo} \cdot \text{cos } \theta + d_y + 700 \cdot \text{cos } \theta - 700 \quad (\text{C.10})$$

Si el valor de  $y_3$  es mayor que cero entonces existe solución para cualquier valor que se le asigne a  $d_y$ .

Por el contrario si  $y_3$  es menor que cero no existe solución posible con ese ángulo y ese valor de  $d_x$  para posicionar el robot.

La tercera solución aparece cuando el discriminante  $\Delta_1$  (Ecuación C.5) es mayor que cero. En este caso son válidos aquellos valores que hacen que la ecuación C.9 tome valores positivos.

Por lo tanto si se considera que las soluciones de la ecuación C.9 se corresponden con los valores:

$$y_{s1}, y_{s2} = \frac{-y_2 \pm \sqrt{\Delta_1}}{2 \cdot y_1} \quad (C.11)$$

De tal manera que  $y_{s1} \leq y_{s2}$  se puede afirmar que el valor de  $d_y$  esta comprendido entre todos aquellos valores que cumplen cualquiera de las dos condiciones siguientes:

$$d_y \leq y_{s1} - x_{base}^{extremo} \cdot \text{sen } \theta - y_{base}^{extremo} \cdot \text{cos } \theta - 700 \cdot \text{cos } \theta + 700 \quad (C.12)$$

$$d_y \geq y_{s2} - x_{base}^{extremo} \cdot \text{sen } \theta - y_{base}^{extremo} \cdot \text{cos } \theta - 700 \cdot \text{cos } \theta + 700 \quad (C.13)$$

Las soluciones expuestas se utilizan, como se ha dicho, cuando  $\text{cos}(q_1)=0$ , en caso de no cumplirse esta condición hay que proceder de otra manera debido a que las ecuaciones anteriores no están definidas en dicho caso. Por lo tanto, se tiene que obtener el valor de  $d_x$  que en lugar de ser un valor exacto viene definido por un intervalo de posibles valores, según la ecuación:

$$x_1 \cdot (x_{base}^{extremo'})^2 + x_2 \cdot x_{base}^{extremo'} + x_3 = 0 \quad (C.14)$$

Que tiene que tomar un valor positivo para indicar una solución válida de la variable  $d_x$ . Los parámetros de la ecuación son:

$$x_{base}^{extremo'} = x_{base}^{extremo} \cdot \text{cos } \theta - y_{base}^{extremo} \cdot \text{sen } \theta + d_x - 700 \cdot \text{sen } \theta \quad (C.15)$$

$$x_1 = 1 + \tan^2 q_1 \quad (C.16)$$

$$x_2 = -2 \cdot l_6 \cdot \left( x_{z_{base}^{extremo}} \cdot \text{cos } \theta - y_{z_{base}^{extremo}} \cdot \text{sen } \theta \right) - 2 \cdot x_0 - 2 \cdot \frac{l_3}{\text{cos } q_1} \cdot \tan q_1 - \\ - 2 \cdot l_6 \cdot \left( x_{z_{base}^{extremo}} \cdot \text{sen } \theta + y_{z_{base}^{extremo}} \cdot \text{cos } \theta \right) \cdot \tan q_1 - 2 \cdot y_0 \cdot \tan q_1 \quad (C.17)$$

$$x_3 = l_6^2 \cdot \left( x_{z_{base}^{extremo}} \cdot \text{cos } \theta - y_{z_{base}^{extremo}} \cdot \text{sen } \theta \right)^2 + 2 \cdot x_0^2 + 2 \cdot l_6 \cdot x_0 \cdot \\ \cdot \left( x_{z_{base}^{extremo}} \cdot \text{cos } \theta - y_{z_{base}^{extremo}} \cdot \text{sen } \theta \right) + \frac{l_3^2}{\text{cos}^2 q_1} + l_6^2 \cdot \left( x_{z_{base}^{extremo}} \cdot \text{sen } \theta + y_{z_{base}^{extremo}} \cdot \text{cos } \theta \right)^2 + \\ + 2 \cdot l_7 \cdot \frac{l_3}{\text{cos } q_1} \cdot \left( x_{z_{base}^{extremo}} \cdot \text{sen } \theta + y_{z_{base}^{extremo}} \cdot \text{cos } \theta \right) + y_0^2 + 2 \cdot \frac{l_3}{\text{cos } q_1} \cdot y_0 + \\ + 2 \cdot l_6 \cdot y_0 \cdot \left( x_{z_{base}^{extremo}} \cdot \text{sen } \theta + y_{z_{base}^{extremo}} \cdot \text{cos } \theta \right) - l_4^2 - l_5^2 + 2 \cdot l_4 \cdot l_5 + \left( p_z - l_6 \cdot z_{z_{base}^{extremo}} - l_1 \right)^2 \quad (C.18)$$

Para la ecuación C.14 existen tres aproximaciones de la solución, las dos primeras aparecen cuando el discriminante  $\Delta_2$  toma valores negativos:



$$\Delta_2 = x_2^2 - 4 \cdot x_1 \cdot x_3 \quad (\text{C.19})$$

Esto implica, que puede no existir solución para  $d_x$ , o por el contrario, que cualquier valor de  $d_x$  es válido, para saber en que caso nos encontramos depende del valor de  $x_3$ , si es menor que cero no existe ninguna solución válida ya que la ecuación C.14 debe tomar valores positivos. Por el contrario, si  $x_3$  es mayor que cero entonces la condición se cumplirá para cualquier valor que se le asigne a  $d_x$ .

Si  $\Delta_2$  toma valores positivos entonces las soluciones de la ecuación C.14 vienen determinadas por:

$$x_{s1}, x_{s2} = \frac{-x_2 \pm \sqrt{\Delta_2}}{2 \cdot x_1} \quad (\text{C.20})$$

Donde se supone que  $x_{s1} \leq x_{s2}$ , por lo tanto  $d_x$  es válido si cumple alguna de las dos condiciones siguientes:

$$d_x \leq x_{s1} - x_{\text{base}}^{\text{extremo}} \cdot \cos \theta + y_{\text{base}}^{\text{extremo}} \cdot \sin \theta + d_x + 700 \cdot \sin \theta \quad (\text{C.21})$$

$$d_x \geq x_{s2} - x_{\text{base}}^{\text{extremo}} \cdot \cos \theta + y_{\text{base}}^{\text{extremo}} \cdot \sin \theta + d_x + 700 \cdot \sin \theta \quad (\text{C.22})$$

De esta manera se calcula el valor para  $d_x$ .

En base al valor asignado a  $d_x$  se calcula el valor de  $d_y$  como:

$$d_y = -\frac{l_3}{\cos q_1} + \left( x_{\text{base}}^{\text{extremo}} \cdot \cos \theta - y_{\text{base}}^{\text{extremo}} \cdot \sin \theta + d_x - 700 \cdot \sin \theta \right) \cdot \tan q_1 - \left( -x_{\text{base}}^{\text{extremo}} \cdot \cos \theta + y_{\text{base}}^{\text{extremo}} \cdot \sin \theta + 700 \cdot \sin \theta \right) \quad (\text{C.23})$$

Con esto se tiene resuelto el posicionamiento del producto sobre la mesa de trabajo para que los componentes puedan ser manipulados por el brazo robot; permitiendo para ello la traslación del producto sobre la mesa de trabajo así como la posterior rotación de la mesa.

## Bibliografía

- [1] Aksoy, HK; Gupta, SM; "Capacity and Buffer Trade-offs in a Remanufacturing System", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 167-174. 2002.
- [2] Alj, A; Fure, R; "Guide de la Recherche Operationelle", Vol. 2, pp. 206-208. 1990.
- [3] Alting, L; Jörgensen, J; "The life cycle concept as a basis for sustainable industrial production", Annals of the CIRP, Vol. 42(1), pp. 163-167. 1993.
- [4] Ammons, JC; Assavapokee, T; Newton, D; Realff, MJ; "Reverse production system design for recycling under uncertainty", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 1-12. 2002.
- [5] Arai, E; Iwata, K; "CAD system with product assembly/disassembly planning function", Robotics & Computer-Integrated Manufacturing, Vol. 10(1/2), pp. 41-48. 1993.
- [6] Arai, E; Uchiyama, N; Igoshi, M; "Disassembly path generation to verify the assemblability of mechanical products", JSME International Journal-Series C, Vol. 38(4), pp. 805-810. 1995.
- [7] Azzone, G; Noci, G; "Measuring the environmental performance of new products: an integrated approach", International Journal of Production Research, Vol. 34(11), pp. 3055-3078. 1996.
- [8] Barbier, EB; "The concept of sustainable economic development", Environment Conservation, Vol. 14(2), pp. 101-110. 1987.
- [9] Barraquand, J; Kavraki, L; "A random sampling scheme for path planning". International Journal of Robotics Research, pp.759-774. 1996.
- [10] Beasley, D; Martín, RR; "Disassembly sequences for object built from unit cubes", Computer Aided Design, Vol. 25(12), pp. 751-761. 1993.
- [11] Berg, M; "A marginal cost analysis for preventive replacement policies", European Journal of Operational Research, Vol. 4, pp. 136-142. 1980.

- [12] Billatos, SB; Nevrekar, VV; "Challenges and practical solutions to designing for the environment", ASME Design for Manufacturability Conference, Chicago, IL, pp. 49-64. 1994.
- [13] Biswas, G; Kawamura, K; Saad, A; Curtin, M; "Intelligent and environmentally conscious manufacturing systems state of the art", International Journal of Environmentally Conscious Design and Manufacturing, Vol. 4(2), pp. 1-10. 1995.
- [14] Bisztyga, B; "Robot Trajectory Tracking System Via Direct Torque Control", System Analysis - Modelling - Simulation, Vol. 33(4), pp. 385-406. 1999.
- [15] Bloemhof-Ruwaard, JM, Van Beek, P; Hordijk, L; Van Wassenhove, LN; "Interations between operational research and environmental management", European Journal of Operational Research, Vol. 85, pp. 229-243. 1995
- [16] Boothroyd, G; Alting, L; "Design for assembly and disassembly", Annals of the CIRP, Vol. 41(2), pp. 625-636. 1992.
- [17] Bras, B; "Incorporating environmental issues in product realization", Industrial and Environment, Vol. 20(1/2), pp. 7-13. 1997.
- [18] Bras, B; Emblemavag, J;"The use of activity-based costing, uncertainty and disassembly action charts in demanufacture cost assessments", Design Engineering Technical Conference, pp. 285-292. 1995
- [19] Bras, B; Hammond, R; "Design for remanufacturing metrics", Proceedings of the First International Working Seminar on Reuse, Eindhoven, The Netherlands, pp. 35-51. 1996.
- [20] Brennan, L; Gupta, SM; Taleb, KN; "Operations planning issues in an assembly/disassembly environment". International Journal of Operations and Production Management, Vol. 14(9), pp. 57-67. 1994.
- [21] Brooks, R.R.; Iyengar, S.S.; "Multi-Sensor Fusion: Fundamentals and Applications with Software", Prentice Hall. 1998.
- [22] Burke, DS; Belter, K; Ishii, K; "Life-cycle design for recyclability", 4<sup>th</sup> International Conference of Design Theory and Methodology, pp. 325-333. 1992.
- [23] Canny, J; "Collision Detection for Moving Polyhedra", Massachusetts Institute of Technology Artificial Intelligence Laboratory. A.I. Memo No 808. 1984.
- [24] Chazelle, B.; "An optimal algorithm for intersecting three-dimensional convex polyhedra". SIAM Journal of Computing, Vol. 21(4) pp. 671-696, 1992.
- [25] Chazelle, B.; "An Optimal Convex Hull Algorithm in Any Fixed Dimension". Discrete Computer Geometry, Vol. 10, pp. 377-409, 1993.
- [26] Chen, RW; Navin-Chandra, D; Nair, I; Prinz, F; Wadehra, IL; "ImSelection - an approach for material selection that integrates mechanical design and life cycle environment burdens", Proceedings of the IEEE International Symposium on Electronics and the Environment, pp. 68-74. 1995.
- [27] Chen, RW; Navin-Chandra, D; Prinz, FB; "A cost-benefit analysis model of product design for recyclability and its application", IEEE Transactions on Components, Packaging and Manufacturing Technology-Part A, Vol. 17(4), pp. 502-507. 1994.

- [28] Chen, SF; Oliver, JH; Chou, S-Y; Chen, L-L; "Parallel disassembly by onion peeling". *Journal of Mechanical Design*, Vol. 119(2), pp. 267-274. 1997.
- [29] Chikte, SD; Deshmukh, SD; "Preventive maintenance and replacement under additive damage", *Naval Research Logistics Quarterly*, Vol. 28, pp. 33-46. 1991
- [30] Ching, S; Kirby, JR; Pitts, OD; "Plastic recycling in business machines: reuse and recovery assessment", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, Arlington, VA, pp. 189. 1993.
- [31] Cho, DI; Parlar, M; "A survey of maintenance models for multi-unit systems", *European Journal of Operational Research*, Vol. 51 pp. 1-23. 1991.
- [32] Ciocci, R; "Lead-free Solder Replacement: Beyond the Material Substitution", *Proceedings of SPIE: Environmentally Conscious Manufacturing II*, Vol. 4569, pp. 100-108. 2002.
- [33] Clegg, AJ; Williams, DJ; "The strategic and competitive implications of recycling and design for disassembly in the electronics industry", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, San Francisco, CA, pp. 6-12. 1994.
- [34] Clegg, AJ; Williams, DJ; Uzsoy, R; "Production planning for companies with remanufacturing capability", *Proceedings of the IEEE Symposium on Electronics and the Environment*, Orlando, FL, pp. 186-191. 1995.
- [35] Coulter, S; Bras, B; Foley, C; "A lexicon of green engineering terms", *International Conference on Engineering Design (ICED)*, Praha, pp. 1-7. 1995.
- [36] Coulter, SL; Bras, BA; "Reducing environmental impact through systematic product evolution", *International Journal of Environmentally Conscious Design and Manufacturing*, Vol. 6(2), pp. 1-10. 1997.
- [37] Das, SK; Yedlarajiah, P; Narendra, R; "An approach for estimating the end-of-life product disassembly effort", *International Journal of Production Research*, Vol. 38(3), pp. 657-673. 2000.
- [38] De Fazio, TL; Delchambre, A; De Lit, P; "Disassembly for recycling of office electronic equipment", *European Journal Mechanical and Environmental Engineering*, Vol. 42(1), pp. 25-31. 1997.
- [39] De Ron, A; Penev, K; "Disassembly and recycling of electronic consumer products: an overview", *Technovation*, Vol. 15(6), pp. 363-374. 1995.
- [40] De Souza, RB; Ab, K; Lee, SG; Lye, SW; "An integrated decision-support environment for DFX audit", *Journal of Electronics Manufacturing*, Vol. 6(3), pp. 163-171. 1996.
- [41] Del Castillo, E; Cochran, JK; "Optimal short horizon distribution operations in reusable containers", *Journal of the Operational Research Society*, Vol. 47, pp. 48-60. 1996.
- [42] Design Optimization using Pro/MECHANICA. Disponible: <http://www.hecbv.nl>
- [43] Di Marco, P; Eubanks, CF, Ishii, K; "Compatibility analysis of products design for recycleability and reuse", *Computers in Engineering*, 1, New York: The American Society of Mechanical Engineers, pp.105-109. 1994.

- [44] Díaz Velázquez, M; "Diccionario básico de matemáticas", Ediciones Anaya S.A. - 1980.
- [45] Dillon, PS; "Mandated electronic equipment recycling in Europe: Implications for companies and US public policy", Proceedings of the IEEE International Symposium on Electronics and the Environment, San Francisco, CA, pp.15-20. 1994.
- [46] Donald, BR; "Motion Planning with Six Degrees of Freedom". MIT AI TR-791. 1984.
- [47] Dowie, T; "Green design", World Class Design to Manufacture, Vol. 1(4), pp. 32-38. 1994.
- [48] Dutta, D; Woo, TC; "Algorithm for multiple disassembly and parallel assemblies". Journal of Engineering for Industry, Vol. 117, pp. 102-109. 1995.
- [49] Emblemsvåg, J; Bras, BA; "A method for life-cycle costing using activity-based costing and uncertainty", International Journal of Engineering Design and Automation 1997.
- [50] Everett, JW; Applegate, D; "Solid waste transfer station design", Journal of Environmental Engineering, Vol. 121(1), pp. 96-106. 1996.
- [51] Faneye, O; Anderl, R; "Optimizing Product Life Cycle Processes in Design Phase", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 138-148. 2002.
- [52] Faverjon, B; Tournassoud, P; "A local based approach for path planning of manipulators with a high number of degrees of freedom". IEEE International Conference on Robotics and Automation, USA. 1987.
- [53] Feldmann, K; Scheller, H; "Disassembly of electronic products", Proceedings of the IEEE International Symposium on Electronics and the Environment, San Francisco, CA, pp. 81-86. 1994.
- [54] Ferrer, G; "The economics of personal computer remanufacturing", Resources, Conservation and Recycling, Vol. 21(2), pp. 79-108. 1997.
- [55] Ferrer, G; "The economics of tire remanufacturing", Resources, Conservation and Recycling, Vol. 19(4), pp. 221-255. 1997.
- [56] Fiksel, J; "Design for environment: creating eco-efficient products and process", McGraw-Hill, 1996.
- [57] Fiksel, J; Wapman, K; "How to design for environment and minimize life cycle cost", IEEE International Symposium on Electronics and the Environment, San Francisco, CA, pp. 75-80. 1994.
- [58] Fiksel, J; Cook, K; Roberst, S; Tsuda, D; "Design for environment at Apple Computer: A case study of the new PowerMacintosh 7200", Proceedings of the IEEE International Symposium on Electronics and the Environment, Dallas, TX, pp. 218-223. 1996.
- [59] Flapper SDP; "Logistic aspects of reuse: an overview", Proceedings of the First International Working Seminar on Reuse, Eindhoven, The Netherlands, pp. 109-118. 1996.
- [60] Fleischmann, M; Boemhof-Ruwaard, JM; Dekker, R; Van der Lann, E; Van Nunen JAE; Van Wassenhove, LN; "Quantitative models for reverse logistics: a review", European Journal of Operational Research, Vol. 103, pp. 1-17. 1997.
- [61] Gadh, R.; Hjalmarsson, H.; Prinz, F.B.; "On the recognition and classification of shape features using MANU-Facture", Issues in Design/Manufacture Integration, ASME, New York, DE-Vol. 39, pp. 53-58. 1992.

- [62] García, M. A.; Larré, A.; López, B.; Oller, A.; "Reducing the Complexity of Geometric Selective Disassembly", IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Takamatsu, Japan, Vol. 2, pp. 1474-1479, 2000.
- [63] Gatenby, DA; Foo, G; "Design for X(DFX): key to competitive, profitable products", AT&T Technical Journal 1990; 68(3):2-15.
- [64] Geider, D; Zussman, E; Lenz, E; "Probabilistic reactive disassembly planning". Annals of the CIRP, Vol. 45, pp. 49-52. 1996.
- [65] Geiger, D; Zussman, E; "Probabilistic reactive disassembly planning", Annals of the CIRP – Manufacturing Technology, Vol. 45(1), pp. 49-52. 1996.
- [66] Ginter, PM; Starling, JM; "Reverse distribution channels for recycling", California Management Review, Vol. 20(3), pp. 73-82. 1978.
- [67] Glantschnig, WJ; "Green design: a review of issues and challenges", Proceedings of the IEEE International Symposium on Electronics and the Environment, Arlington, VA, pp. 74-77. 1993.
- [68] Glantschnig, WJ; "Green design: an introduction to issues and challenges", IEEE Transactions on Components, Packaging and Manufacturing Technology-Part A, Vol. 17(4), pp. 508-513. 1994.
- [69] Glazebrook, B; Coulon, R; Abrassart, C; "Towards a product life cycle design tool", Proceedings of the IEEE International Symposium on Electronics and the Environment, pp. 81-85. 2000.
- [70] Goh, TN; Varaprasad, N; "A statistical methodology for the analysis of the life-cycle of reusable containers", IIE Transactions, Vol. 18, pp. 42-47. 1986.
- [71] González, B; Andenso-Díaz, B; Muñoz,L; Life cycle assessment in SMES: a fuzzy approach", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 149-157. 2002.
- [72] Graedel, TE, Allenby, BR; Combire, PR; "Matrix approaches to abridged life cycle assessment", Environmental Science & Technology, Vol. 29(3), pp. 134-139. 1995.
- [73] Graedel, TE; "Prioritizing impacts: the second stage of life cycle assessment", Proceedings of the IEEE International Symposium on Electronics and the Environment, San Francisco, CA, pp. 89-93. 1994.
- [74] Graedel, TE; Allenby, BR; "Design for environment", Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [75] Graedel, TE; Allenby, BR; "Industrial ecology", Englewood Cliffs, NJ: Prentice-Hall, 1995.
- [76] Grenchus, E; Keene, R; Nobs, C; "Demanufacturing of information technology equipment", Proceedings of the IEEE International Symposium on Electronics and the Environment, San Francisco, CA, pp. 157-160. 1997.
- [77] Guide Jr, VDR; "Kraus, MA; Srivastava, R; "Scheduling policies for remanufacturing", International Journal of Production Economics, Vol. 48, pp. 187-204. 1997.
- [78] Guide Jr, VDR; "Scheduling using drum-buffer-rope in a remanufacturing environment", International Journal of Production Research, Vol. 34(4), pp. 1081-1091. 1996.
- [79] Guide Jr, VDR; "Scheduling with priority dispatching rules and rum-buffer-rope in a recoverable manufacturing system", International Journal of Productions Economics, Vol. 53, pp. 101-116. 1997.

- [80] Guide Jr, VDR; Spencer, MS; "Rough-cut capacity planning for remanufacturing firms", *Production Planning & Control*, Vol. 8(3), pp. 237-244. 1997.
- [81] Guide Jr, VDR; Srivastava, R; "An evaluation of order release strategies in a remanufacturing environment", *Computers and Operations Research*, Vol. 24(1), pp. 37-47. 1997.
- [82] Guide Jr, VDR; Srivastava, R; "Buffering from material recovery uncertainty in a recoverable manufacturing environment", *Journal of the Operational Research Society*, Vol. 48, pp. 519-529. 1997.
- [83] Guide Jr, VDR; Srivastava, R; "Repairable inventory theory: models and applications", *European Journal of Operational Research*, Vol. 102, pp. 1-20. 1997.
- [84] Guide Jr, VDR; Srivastava, R; Kraus, MA; "Product structure complexity and scheduling of operations in recoverable manufacturing", *International Journal of Production Research*, Vol. 25(11), pp. 3179-3199. 1997.
- [85] Guide Jr, VDR; Srivastava, R; Kraus, ME; "Proactive expediting policies for recoverable manufacturing", *Journal of the Operational Research Society*, Vol. 48, pp. 1-13. 1997.
- [86] Guide Jr, VDR; Srivastava, R; Spencer, MS; "An evaluation of capacity planning techniques in a remanufacturing environment", *International Journal of Production Research*, Vol. 35(1), pp. 67-82. 1997.
- [87] Guiltinan, JP; Nwokoye, NG; "Developing distribution channels and systems in the emerging recycling industries", *International Journal of Physical Distribution*, Vol. 48, pp. 1-13. 1975.
- [88] Gungor, A, Gupta, SM; "An evaluation methodology for disassembly processes". *Computers and Industrial Engineering*, Vol. 33 (1-2), pp. 329-332. 1997.
- [89] Gungor, A; Gupta, SM; "Disassembly sequence plan generation using a branch-and-bound algorithm", *International Journal of Production Research*, Vol. 39(3), pp. 481-509. 2001.
- [90] Gungor, A; Gupta, SM; "Disassembly sequence planning for complete disassembly in product recovery". *Proceedings of Northeast Decision Sciences Institute -27<sup>th</sup> Annual Meeting*, Boston, Massachusetts, March 25-27, pp. 250-252. 1998.
- [91] Gungor, A; Gupta, SM; "Disassembly sequence planning for products with defective parts in product recovery", *Computers and Industrial Engineering*, Vol. 35(1-2), pp. 161-164. 1998.
- [92] Gungor, A; Gupta, SM; "Issues in environmentally conscious manufacturing and product recovery: a survey". *Computers & Industrial Engineering*, Vol. 36, pp. 811-853. 1999.
- [93] Gupta, SM; McLean, CR; "Disassembly of products", *Computers and Industrial Engineering*, Vol. 31, pp. 255-258. 1996.
- [94] Gupta, SM; Taleb, KN; "Scheduling disassembly", *International Journal of Production Research*, Vol. 32(8), pp. 1857-1866. 1994.
- [95] Gupta, SM; Veerakamolmal, P; "Profitability measure for product disassembly and recycling". *Proceedings of Production and Operations Management Society*, Cape Town, South Africa, pp. 139-147. 1998.

- [96] Hattori, M; Nomura, N; Sommer, D; Inoue, H; "Fundamentals of environmentally conscious product design", *International Journal of Environmentally Conscious Design & Manufacturing*, Vol. 4(1), pp. 3-11. 1995.
- [97] Hearn, D; Baker, MP; "Computer Graphics". Prentice Hall. Second Edition. 1994.
- [98] Hedberg, SR; "Design of a lifetime", *Byte*, pp. 103-106. 1994.
- [99] Hedge, GG; "Life cycle cost: a model and applications", *IIE Transactions*, Vol. 26(6), pp. 56-62. 1994.
- [100] Hendrix, J; Massey, KA; Whitham, E; Russel, M; Bras, BA; "Technologies for the identification, separation and recycling of automotive recycling", *International Journal of Environmentally Conscious Design and Manufacturing*, Vol. 6(2), pp. 37-50. 1997.
- [101] Henshaw, JW; "Design for recycling: new paradigm or just the latest 'design-for-X' fad?", *International Journal of Materials and Product Technology*, Vol. 9, pp. 125-138. 1994.
- [102] Hentschel, C; Seliger, G; Zussman, E; "Grouping of used products for cellular recycling system", *Annals of the CIRP*, Vol. 44(1), pp. 11-14. 1995.
- [103] Hesselbach, J; Kühn, M; "Disassembly assessment and planning for electronic consumer appliances", *Proceedings of the First International Working Seminar on Reuse*, Eindhoven, The Netherlands, pp. 136-169. 1996.
- [104] Hesselbach, J; Kühn, M; "Disassembly evaluation of electronics & electrical products", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, Oak Brook, IL, pp. 79-81. 1998.
- [105] Holloway, L; Clegg, D; Tranter, I; Cockerham, G; "Incorporating environmental principles into the design process", *Materials & Design*, Vol. 15(5), pp. 259-267. 1994.
- [106] Holt, HR; "A first step in electronic ecodesign", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, San Francisco, CA, pp. 191-195. 1994.
- [107] Homem de Mello, LS; Sanderson, A.C; "AND/OR Graph Representation of Assembly Plans", *IEEE Transaction On Robotics And Automation*, Vol. 6(2). 1990.
- [108] Hormozi, AM; "Parts remanufacturing in the automotive industry", *Production and Inventory Management Journal*, Vol. 38(1), pp. 26-31. 1997.
- [109] Hoshino, T; Yura, K; Hitomi, K; "Optimization analysis for recycle-oriented manufacturing systems", *International Journal of Production Research*, Vol. 33(8), pp. 2069-2078. 1995
- [110] Huang, HH; Wang, MH; "Optimal disassembly sequence generation using Neural Network". *Proceedings of the Fourth International Conference on Environmentally Conscious Design and Manufacturing*, Cleveland, OH, pp. 231-238. 1996.
- [111] Hundal, MS; "DFE: current status and challenges for the future", *ASME Design for Manufacturability Conference*, Chicago, IL, pp. 89-98. 1994.
- [112] Hwang, Y; Cho, K; Lee, S; Park, S; Kang, S; "Human computer cooperation in interactive motion planning". *ICAR'97*, Monterey, California, USA. 1997.



- [113] Inderfurth, K; "Simple optimal replenishment and disposal policies for a product recovery system with leadtimes", *OR Spektrum*, Vol. 19, pp. 111-122. 1997.
- [114] Isaacs, JA; Gupta, SM; "Economic consequences of increasing polymer content on the Usautomobile recycling infrastructure", *Journal of Industrial Ecology*, Vol. 1(4), pp. 19-34. 1997.
- [115] Isaacs, JA; Gupta, SM; "Effect of vehicle lightweighting on the profitability of the automobile recycling infrastructure", *Proceedings of the Julian Szekely Memorial Symposium of Materials Processing*, Cambridge, MA, pp. 669-685. 1997.
- [116] Isaacs, JA; Gupta, SM; Messac, A; "A decision tool to assess the impact of automobile design on disposal strategies", *Proceedings of the Environmentally Conscious Manufacturing and Design Conference*, Cleveland, OH, pp. 207-214. 1996.
- [117] Ishii, K; "Life-cycle engineering design", *Journal of Mechanical Design*, Vol. 117B, pp. 42-47 1995.
- [118] Ishii, K; Adler, R; Barkan, P; "Applications of design compatibility analysis to simultaneous engineering", *Journal of Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 2(1), pp. 53-65. 1998.
- [119] Ishii, K; Eubanks, CF; "Life cycle evaluation of mechanical systems", *Proceedings of the NSF Design and Manufacturing Systems Conference*, pp. 575-579. 1993.
- [120] Ishii, K; Lee, BH; Eubanks, CF; "Design for product retirement and modularity based on technology life-cycle", *Manufacturing Science and Engineering*, pp. 921-933. 1995.
- [121] Ishii, K; Eubanks, CF; Di Marco, P; "Design for product retirement and material life cycle", *Materials & Design*, Vol. 15(4), pp. 225-233. 1994.
- [122] Istvan, Z; Garamvolgyi, E; "Reverse logistics and management of end-of-life electric products", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 15-19. 2000.
- [123] Johnson, MR, Wang, MH; "Planning product disassembly for material recovery opportunities", *International Journal of Production Research*, Vol. 33(11), pp. 3119-3142. 1995.
- [124] Jovane, F; Alting, L; Armoillotta, A; Eversheim, W; Feldmann, K; Seliger, G; Roth, N; "A key issue in product life cycle: disassembly". *Annals of the CIRP*, Vol. 42(2), pp. 651-658. 1993.
- [125] Kang, HY; Lee, C; Jung, JW; "Ecodesign application to electronic products using the new software tool", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 161-165. 2000.
- [126] Kavraki, L; Svestka, P; "Probabilistic roadmaps for path planning in high-dimensional configuration spaces". *IEEE, Transactions on Robotics and Automation*. 1996.
- [127] Keldmann, R; Olsen, J; "Managing the environmental issue in product development: the environmentally oriented milestone questions and techniques", *ASME Design for Manufacturability Conference*, Chicago, IL, pp. 99-106. 1994.
- [128] Kelle, P; Silver, EA; "Forecasting the returns of reusable containers", *Journal of Operations Management* Vol. 8(1), pp. 17-35. 1989.

- [129] Kelle, P; Silver, EA; "Purchasing policy of new containers considering the random returns of previously issued containers", IIE Transactions, Vol. 21(4), pp. 349-354. 1989.
- [130] Keoleian, GA; Menerey, D; "Sustainable development by design: review of life cycle design and related approaches", Air & Waste: Journal of Air & Waste Management Association, Vol. 44(5), pp. 645-668. 1994.
- [131] Kizilkaya, EA; Gupta, SM; "Material flow control and scheduling in a disassembly environment", Computers and Industrial Engineering, Vol. 35(1-2), pp. 93-96. 1998.
- [132] Kleban, SD; Luger, GF; Watkin, RD; "Expert system support for environmental assessment of manufacturing products and facilities", Journal of Intelligent Manufacturing, Vol. 7, pp. 39-53. 1996.
- [133] Kongar, E; Gupta, SM; "A Genetic Algorithm for Disassembly Process Planning", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 54-62. 2002.
- [134] Kooi, E; Krikke, H; Schuur, P; "Physical design of a reverse logistic network: a multi-echelon model", Proceedings of the First International Working Seminar on Reuse, Eindhoven, The Netherlands, pp. 205-212. 1996.
- [135] Kopacek, B; Kopacek, P; "A semi-automatised disassembly cell for printed circuit boards", Proceedings of the 2<sup>nd</sup> IFAC Workshop on Intelligent Assembly and Disassembly IAD'2001, Caneda, Brasil. 2001.
- [136] Kopacek, P; Kronreif, G; "Semi-automated robotized disassembling of personal computers", IEEE Conference on Emerging Technologies and Factory Automation, Kauai, Hawaii, pp. 567-572. 1996.
- [137] Kopacek, P; Noe, D; "Intelligent, flexible assembly and disassembly", Proceedings of the 2<sup>nd</sup> IFAC Workshop on Intelligent Assembly and Disassembly IAD'2001, Caneda, Brasil. 2001.
- [138] Kopicky, RJ; Berg, MJ; Legg, L; Dasappa, V; Maggioni, C; "Reuse and recycling: reverse logistics opportunities", Oak Brook, IL: Council of Logistics Management. 1993.
- [139] Korpalski, T; "The role of the product steward in advancing design for environmental in Hewlett-Packard's computer product organization", Proceedings of the IEEE International Symposium on Electronics and the Environment, pp. 37-41. 1996.
- [140] Korugan, A; Gupta, SM; "A multi-echelon inventory system with returns", Computers and Industrial Engineering Vol. 35(1-2), pp. 145-148. 1998,
- [141] Korugan, A; Gupta, SM;" An Adaptative Kanban Control Mechanism for a Single Stage Hybrid System", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 175-181. 2002.
- [142] Kouikoglou, VS; "Optimal rate allocation in unreliable, assembly/disassembly production networks with blocking", IEEE Transactions on Robotics and Automation, Vol. 16(4), pp. 429-434. 2000.
- [143] Krikke, HR; Van Harten, A; Schuur, PC; "On a medium term product recovery and disposal strategy for durable assembly products", International Journal of Production Research, Vol. 36(1), pp. 111-139. 1998.

- [144] Kriwe, A; Zussman, E; Seliger, G; "Systematic integration of design for recycling into product design", *International Journal of Production Economics*, Vol. 28, pp. 15-22. 1995.
- [145] Kroll, E; Beardsley, B; Parulian, A; "A methodology to evaluate ease of disassembly for product recycling", *IIE Transactions*, Vol. 28(10), pp. 837-845. 1996.
- [146] Kroon, L; Vrijens, G; "Returnable containers: an example of reverse logistics", *International Journal of Physical Distribution & Logistics Management*, Vol. 25(2), pp. 56-68. 1995.
- [147] Krupp, J; "Structuring bills of material for automobile remanufacturing", *Production and Inventory Management Journal*, Vol. 34, pp. 46-52. 1993.
- [148] Kuo, TC; "Disassembly sequence and cost analysis for electromechanical products", *Robotics and Computer Integrated Manufacturing*, Vol. 16, pp. 43-54. 2000.
- [149] Kuo, TC; Zhang, HC; "Life cycle engineering: Design for recyclability", *Proceedings of the 1<sup>st</sup> World Conference on Integrated Design and Process Technology*, pp. 1-7. 1995.
- [150] Kuo, TC; Zhang, HC; Huang, SH; "Disassembly analysis for electromechanical products: a graph-based heuristic approach" *International Journal of Production Research*, Vol. 38(5), pp. 993-1007. 2000.
- [151] Kuuva, M; Airila, M; "Conceptual approach on design for practical product recycling", *ASME Design for Manufacturability Conference*, Chicago, IL, pp. 115-123. 1994.
- [152] Lambert, AJD; "Optimal disassembly of complex products". *International Journal of Production Research*, Vol. 35(9), pp. 2509-2523. 1997.
- [153] Lambert, AJD; Splinter, MAM; "Reuse of sophisticated product carriers", *Proceedings of the First International Working Seminar on Reuse*, Eindhoven, The Netherlands, pp. 213-222. 1996.
- [154] Laperiere, L; ElMaraghy, HA; "Planning of products assembly and disassembly", *Annals of the CIRP*, Vol. 41(1), pp. 5-9. 1992.
- [155] Laperriere, L; ElMaraghy, H; "GAPP: A generative assembly process planner". *Journal of Manufacturing Systems*, Vol. 15(4), pp. 282-293. 1996.
- [156] Latombe, JC; Wilson, R; "Assembly sequencing with toleranced parts". *Conference International Solid modelling 95*, Salt lake city, Utah USA, pp. 83-95. 1995.
- [157] Lave, LB; Hendrickson, CT; McMichael, FC; "Recycling decisions and green design", *Environmental Science & Technology*, Vol. 28(1), pp. 19A-24A. 1994.
- [158] Lee, D-H; Kang, J-G; Xirouchakis, P; "Disassembly planning and scheduling: review and further research". *Proceedings of Institution in Mechanical Engineers*, Vol. 215 Part B, pp. 695-709. 2001.
- [159] Lee, JJ; O'Callaghan, P; Allen, D; "Critical review of life cycle analysis and assessment techniques and thir application to commercial activities", *Conservation and Recycling*, Vol. 13, pp. 37-56. 1995.

- [160] Lee, K; Gadh, R; "Computer aided design-for-disassembly: a destructive approach". Joint Symposium on Concurrent Product and Process Engineering at International Mechanical Engineering Congress and Exposition '95, San Francisco, CA, pp. 1-13. 1995.
- [161] Lee, K; Gadh, R; "Destructive disassembly to support virtual prototyping". IEE Transactions, Vol. 30, pp. 959-970. 1998
- [162] Lee, KM; Van Kuren, MB; "Modeling and Supervisory Control of a Disassembly Automation Workcell Based on Blocking Topology". IEEE Transactions on Robotics and Automation, Vol. 16(1), pp. 67-67. 2000.
- [163] Lee, S; Yi, C; "Forced-Based reasoning in assembly planning", Vision Sensors and Control for Automated Manufacturing Systems, SPIE Vol. 2063, pp. 97-108. 1994.
- [164] Lee, YD; Lee, BH; "Genetic Trajectory Planner for a Manipulator with Acceleration Parametrization", Journal for Universal Computer Science, Vol. 3(9), pp. 1056-1073. 1997.
- [165] Lee, Y-Q; Kamara, SRT; "Individual and group disassembly sequence generation through freedom and interference spaces". Journal of Design and Manufacturing, Vol. 2, pp. 143-154. 1992.
- [166] Legarth, JB; Alting, L; Baldo, GL; "Sustainability issues in circuit board recycling", Proceedings of the IEEE International Symposium on Electronics and the Environment, Orlando, FL, pp. 126-131. 1995.
- [167] Leng, DY; Chen, M; "Robot trajectory planning using simulation", Robotics and Computer Integrated Manufacturing, Vol 13(2), pp. 121-130. 1997
- [168] Lenox, M; Jordan, B; Ehrenfeld, J; "The diffusion of design for environment: A survey of current practice", Proceedings of the IEEE International Symposium on Electronics and the Environment, pp. 25-30. 1996.
- [169] Leonar, L; "Design for environment", Plastics Design Forum, pp. 25-32. 1991.
- [170] Li, W; Zhang, HP, Wang, B; Awoniyi, SA; "Design for disassembly analysis for environmentally conscious design and manufacturing", Manufacturing Science and Engineering, pp. 969-976. 1995.
- [171] Liu, HHT; Mills, JK; "Robot Trajectory Control System Design for Multiple Simultaneous Specifications: Theory and Experimentation", Journal of Dynamic Systems Measurement and Control, Vol. 120(4), pp. 520-523. 1998.
- [172] Livingstone, S; Sparks, L; "The new German packaging laws: effects on firms exporting to Germany", International Journal of Physical Distribution & Logistics Management, Vol. 24(7), pp. 15-25. 1994.
- [173] Lozano-Pérez, T; "A simple motion-planning : a configuration space approach". IEEE Robotics and Automation. 1987.
- [174] Lozano-Pérez, T; "Spatial Planning: A configuration Space approach", IEEE Transactions in Computers, Vol. 32(2), pp. 108-120. 1983.
- [175] Lund, RT; "Remanufacturing", Technology Review, Vol. 87(2), pp. 19-29. 1984.
- [176] Mabini, MC; Pintelon, LM; Gelders, LF; "EOQ type formulations for controlling repairable inventories", International Journal of Production Economics, Vol. 28, pp. 21-33. 1992.
- [177] Madu, C; "A framework for environmental quality assessment". International Journal of Quality Science, Vol. 1(3), pp. 24-28. 1996.

- [178] Madu, C; Kuei, C-H; Winokur, D; "Environmental quality planning: a strategic total quality management approach". *Futures*, Vol. 27(8), pp. 839-856. 1995.
- [179] Man, E; Díez-Campo, JE, Roche, T; "DFE Workbench: a CAD integrated DFE tool", *Proceedings of SPIE: Environmentally Conscious Manufacturing II*, Vol. 4569, pp. 93-99. 2002.
- [180] Martinez, M; Pham, V-H; Favrel, J; "Dynamic Generation of Disassembly Sequences", *IEEE International Conference On Emerging Technologies and Factory Automation, ETFA'97*, pp. 177-182, 1997.
- [181] Mascle, C; "Approche méthodologique de détermination de gammes par le désassemblage". *Tesis, Ecole polytechnique fédérale de Lausanne, Suisse*, 1990.
- [182] Mattikali, RS; Khosla, PK; "Analysis of restraints to translational and rotational motion from the geometry of contact". *Proceedings ASME winter annual meeting, Issues in Design Manufacture/Integration, Atlanta, DE-Vol. 39*. 1991.
- [183] McCue, G; "Life-cycle engineering", *Management Service*, pp. 20-23. 1993.
- [184] Miettinen, P; Hämäläinen, RP; "How to benefit from decision analysis in environmental life cycle assessment (LCA)", *European Journal of Operational Research*, Vol. 102(2), pp. 279-294. 1997.
- [185] Minami, S; Pahng, KF; Jakiela, MJ; Srivastava, A; "A cellular automata representation for assembly simulation and sequence generation", *Proceedings of the IEEE International Symposium on Assembly and Task Planning, Pittsburgh, Pennsylvania*, pp. 56-65. 1995.
- [186] Mizuki, C; Sandborn, PA; Pitts, G; "Design for environment – A survey of current practices and tools", *Proceedings of the IEEE International Symposium on Electronics and the Environment, Dallas, TX*, pp. 1-6. 1996.
- [187] Mok, SM; Wu, CH; Lee, DT; "A system for analysing automatic assembly and disassembly operations", *Proceedings of the IEEE International Conference on Robotics and Automation, Vol. 4*, pp. 3695-3700. 2000.
- [188] Moore, KE, Gungor, A; Gupta, SM; "Disassembly process planning using Petri nets". *Proceedings of the IEEE International Symposium on Electronics and the Environment, Oak Brook, IL*, pp. 88-93. 1998.
- [189] Moore, KE; Gungor, A; Gupta, SM; "A Petri net approach to disassembly process planning". *Computers and Industrial Engineering, Vol. 35(1-2)*, pp. 165-168. 1998.
- [190] Moore, KE; Gungor, A; Gupta, SM; "Disassembly Petri net generation in the presence of XOR precedence relationships". *Proceedings of the IEEE International Conference on Systmes, Man and Cybernetics, San Diego, CA*, pp. 11-14. 1998.
- [191] Moyer, LK; Gupta, SM; "Environmental concerns and recycling/disassembly efforts in the electronics industry". *Journal of Electronics Manufacturing, Vol. 7(1)*, pp. 1-22. 1997.
- [192] Nasr, N; Hughson, C; Varel, E; Bauer, R; "State-of-the-art assessment of remanufacturing technology", *Rochester, NY: Rochester Institute of Technology*, 1998.

- [193] Navin-Chandra, D; "Design for environmentability", ASME Design Technical Conferences – 3<sup>rd</sup> International Conference on Design Theory and Methodology, Miami, FL, pp. 119-125. 1991.
- [194] Navin-Chandra, D; "ReStar: a design tool for environmental recovery analysis". Proceedings of ICED'93: 9<sup>th</sup> International Conference on Engineering Design, The Hague, pp. 780-787. 1993.
- [195] Navin-Chandra, D; "The recovery problem in product design". Journal of Engineering Design, Vol. 5(1), pp. 65-86. 1994.
- [196] Nearchou, AC; Aspragathos, NA; "Collision-Free Trajectory Generation Using Raster Scanning and Genetic Algorithms", Journal of Intelligent and Robotic Systems, Vol. 23(2), PP. 351-377. 1998.
- [197] Noble, JS; Lim, HH; "An Evaluation of Facility Layout Alternatives for a Remanufacturing Environment", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 158-166. 2002.
- [198] Nof, SY; Chen, J; "Assembly and disassembly: A framework for cooperation requirement planning", Proceedings of the 2<sup>nd</sup> IFAC Workshop on Intelligent Assembly and Disassembly IAD'2001, Caneda, Brasil. 2001.
- [199] O'Shea, B; Grewal, SS; Kaebernick, H.; "State of the Art Literature Survey on Disassembly Planning", Concurrent Engineering: Research and Applications, Vol. 6(4), pp. 345-357. 1998.
- [200] O'Shea, B; Kaebernick, H; Grewal, SS; Perlewitz, H; Müller, K; Seliger, G; "Method for automatic tool selection for disassembly planning", Assembly Automation, Vol. 19(1), pp. 47-54. 1999.
- [201] Overby, C; "Design for the entire life cycle: a new paradigm?", Proceedings of the ASEE Annual Conference, pp. 552-563. 1990.
- [202] Pan, L; Zeid, I; "A knowledge base for indexing and retrieving disassembly plans", Journal of Intelligent Manufacturing, Vol. 12, pp. 77-94. 2001.
- [203] Panisset, BD; "MRP II for repair/refurbish industries", Production and Inventory Management Journal, Vol. 29(4), pp. 12-15. 1988.
- [204] Pechoucek, M; Marik, V; Stepankova, O; "PROPLANT – Concept of agency in production planning", Proceedings of the 2<sup>nd</sup> IFAC Workshop on Intelligent Assembly and Disassembly IAD'2001, Caneda, Brasil. 2001.
- [205] Penev, KD; Ron, AJD; "Determination of a disassembly strategy". International Journal of Production Research, Vol. 34(2), pp. 495-506. 1996.
- [206] Perlewitz, H; Müller, K; Seliger, G.; "Usage mode and effects analysis to support disassembly process planning", IFAC Intelligent Assembly and Disassembly, Bled, Slovenia, pp. 11-16, 1998
- [207] Pieper, D.; Roth, B.; "The kinematics of manipulators under computer control". Proc. of the Second International Congress on Theory of Machines and Mechanisms, Zakopane, Polonia, Vol. 2, pp. 159-169. 1969.
- [208] Pierskalla, WP; Voelker, JA; "A survey of maintenance models: the control and surveillance of deteriorating system", Naval Research Logistics Quarterly, Vol. 23, pp. 353-388. 1976.
- [209] Pitts, G; Mizuki, C; "A view of electronic products disposition", Proceedings of the IEEE International Symposium on Electronics and the Environment, Dallas, TX, pp. 66-72. 1996.

- [210] Pnueli, Y; Zussman, E; "Evaluating the end-of-life value of a product and improving it by redesign", *International Journal of Production Research*, Vol. 35(4), pp. 921-942. 1997.
- [211] Pomares, J; Torres, F; Puente, ST; "Disassembly movements for geometrical objects through heuristic methods", *Proceedings of SPIE: Environmentally Conscious Manufacturing II*, Vol. 4569, pp. 71-80. 2002.
- [212] Propuesta de directiva del parlamento europeo y del consejo sobre restricciones a la utilización de determinadas sustancias peligrosas en aparatos eléctricos y electrónicos. Disponible: [http://europa.eu.int/prelex/detail\\_dossier\\_real.cfm?CL=es&DosId=158021](http://europa.eu.int/prelex/detail_dossier_real.cfm?CL=es&DosId=158021)
- [213] Puente, ST; Torres, F; Gil, P; "An approach to disassembly sequence generation", *Proceedings of the 2<sup>nd</sup> IFAC Workshop on Intelligent Assembly and Disassembly IAD'2001*, Caneda, Brasil. 2001.
- [214] Puente-Méndez, ST; Torres-Medina, F; Pomares-Baeza, J; "Product disassembly scheduling using graph models", *Proceedings of SPIE: Environmentally Conscious Manufacturing II*, Vol. 4569, pp. 63-70. 2002.
- [215] Rank, PG; Das, S; Caudili, RJ; "A Web-oriented, virtual product disassembly and identification method for DFE and electronic demanufacturers", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 172-177. 2000.
- [216] Reid, MB; "Path Planning Using Optically Computed Potential Fields", *Proceedings of the IEEE International Conference on Robotics and Automation*, Atlanta, NC, Vol. 2, pp. 295-300. 1993.
- [217] Reimer, B; Shodhi, MS; Knight, WA; "Optimizing electronics end-of-life disposal cost", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, pp. 342-347. 2000.
- [218] Rejnri, N; "Détermination et simulation des opérations d'assemblage lors de la conception de systèmes mécaniques". Tesis para obtener el grado de DOCTEUR DE L'INPG. Institut national Polytechnique de Grenoble.
- [219] Rembert, TC; "Package deal: The European war on waste", *The Environmental Magazine*, Vol. 8(3), pp. 38. 1997.
- [220] Reyes, W; Moore, M; Bartholomew, CRM; Currence, R; Siegel, R; "Reliability assessment of used parts: An enabler for asset recovery", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, Orlando, FL, pp. 89-94. 1995.
- [221] Rhodes, SP; "Applications of life cycle assessment in the electronics industry for product design and marketing claims", *Proceedings of the IEEE International Symposium on Electronics and the Environment*, Arlington, VA, pp. 101-105. 1993.
- [222] Richter, K; "The EOQ repair and waste disposal model with variable setup numbers", *European Journal of Operational Research*, Vol. 95(2), pp. 313-324. 1996.
- [223] Richter, K; "The extended EOQ repair and waste disposal model", *International Journal of Production Economics*, Vol. 45(1-3), pp. 443-448. 1996.

- [224] Ridder, C; Scheidt, LG; "Practical experiences in the Sony disassembly evaluation workshop", Proceedings of the IEEE International Symposium on Electronics and the Environment, Oak, Brook, IL, pp. 94-99. 1998.
- [225] Romney, B; Godard, C; "An efficient system for geometric assembly sequence generation and evaluation". Proceedings 1995 ASME. International Computers in Engineering Conference, pp. 699-712. 1995.
- [226] Rose, CM; Stevels, A; Ishii, K; "A new approach to end-of-life design advisor (ELDA)", Proceedings of the IEEE International Symposium on Electronics and the Environment, pp. 99-104. 2000.
- [227] Rose, EP; Evans, S; "A robust design for recyclability methodology", Proceedings of CIM-Europe Annual Conference, pp. 201-211. 1993.
- [228] Rosen, DW; Bras, BA; Hassenzahl, SL; Newcomb, PJ, Yu, T; "Computer-aided design for the life cycle", Journal of Intelligent Manufacturing, Vol. 7, pp. 145-160. 1996.
- [229] Rosen, DW; Bras, BA; Mistree, F; Goel, A; "A virtual design studio approach to virtual product demanufacture and service", ASME Computers in Engineering Conference, Boston, MA, pp. 951-958. 1995.
- [230] Ryberg, BA; "Design for environmental quality: Reap the benefits of closing the design loop", Proceedings of the IEEE International Symposium on Electronics and the Environment, Arlington, VA, pp. 37-42. 1993.
- [231] Sandborn, PA; McFall, G; "Performing design for environment concurrent with interdisciplinary tradeoff analysis of electronic systems", Proceedings of the IEEE International Symposium on Electronics and the Environment, Dallas, TX, pp. 167-172. 1996.
- [232] Sarkis, J; "Evaluating environmentally conscious business practices", European Journal of Operational Research, Vol. 107(1), pp. 159-174. 1998.
- [233] Scheidt, LG; Abdoolcader, N; Stadlbauer, H; Zong, S; Doelman, P; "Electronics recycling - Another dimension", Proceedings of the IEEE International Symposium on Electronics and the Environment, Orlando, FL, pp. 290-294. 1995.
- [234] Scheuring, J; Bras, BA; Lee, KM; "Significance of design for disassembly on integrated disassembly and assembly processes", International Journal of Environmentally Conscious Design and Manufacturing, Vol. 3(2), pp. 21-23. 1994.
- [235] Schwartz, JY; Sharir, M; "On the 'Piano Movers' Problem, II. General Techniques for Computing Topological Properties of Real Algebraic Manifolds". Computer Science Department, New York University report 41. 1982.
- [236] Seaver, WB; "Design considerations for remanufacturability, recyclability and reusability of user interface modules", Proceedings of the IEEE International Symposium on Electronics and the Environment, San Francisco, CA, pp. 241-245. 1994.
- [237] Seegers, HJM; "Automotive design for recycling in GE plastics", International Conference on Engineering Design, pp. 812-819. 1993.
- [238] Seliger, G; Kim, HJ; Keil, T; "Generation of Control Sequences fo a Pilot-Disassembly System", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 81-92. 2002.



- [239] Seliger, G; Zussman, E; Kriwet, A; "Integration of recycling considerations into product design - A system approach", Nof SY, editor. Information and Collaboration Models of Integration-Based on the NATO Research Workshop on Integration, Il Crocco, Italy. Dordrecht, The Netherlands: Kluwer Academic Publishers, pp. 27-41. 1994.
- [240] Senin, N; Groppetti, R; Wallace, DR; "Concurrent assembly planning with genetic algorithms", Robotics and Computer Integrated Manufacturing, Vol. 16, pp. 65-72. 2000.
- [241] Sheng, PS; Dornfeld, DA; Worhach, P; "Integration issues in green design and manufacturing", Manufacturing Review, Vol. 8(2), pp. 95-105. 1995.
- [242] Shergold, M; "Automotive materials recycling for the future", Proceedings of Institution of Mechanical Engineering, Vol. 208(2), pp. 75-82. 1994.
- [243] Shyamsundar, N; Gadh, R; "Geometric abstractions to support disassembly analysis". IIE Transactions, Vol. 31, pp. 935-946. 1999.
- [244] Siddique, Z; Rosen, DW; "A virtual prototyping approach to product disassembly reasoning", Computer-Aided Design, Vol. 29(12), pp. 847-860. 1997.
- [245] Simpson, VP; "Optimum solution structure for a repairable inventory problem", Operations Research, Vol. 26, pp. 270-281. 1978.
- [246] Sincero, AP; Sincero, GA; "Environmental engineering - A design approach", Prentice-Hall, 1995.
- [247] Srinivasan, H; Figueroa, R; Gadh R.; "Selective disassembly for virtual prototyping as applied to de-manufacturing". Robotics and Computer Integrated Manufacturing, Vol. 15 pp. 231-245. 1999.
- [248] Srinivasan, H; Gadh, R.; "A geometric algorithm for single selective disassembly using the wave propagation abstraction". Computer-Aided Design, Vol. 30(8), pp. 603-613. 1998.
- [249] Srinivasan, H; Gadh, R; "Complexity Reduction in Geometric Selective Disassembly Using the Wave Propagation Abstraction". International Conference on Robotics and Automation (ICRA), Lueven, Belgium, 1998.
- [250] Srinivasan, H; Gadh, R; "Selective disassembly of components with geometric constrains". Proc. ASME Design Engineering Technical Conferences, Las Vegas, Nevada, 1999.
- [251] Srinivasan, H; Shyamsundar, N; Gadh, R; "A framework for virtual disassembly analysis", Journal of Intelligent Manufacturing, Vol. 8, pp. 277-295. 1997.
- [252] Srinivasan, H; Shyamsundar, N; Gadh, R; "A virtual disassembly tool to support environmentally conscious product design", Proceedings of the IEEE International Symposium on Electronics and the Environment, San Francisco, CA, pp. 7-12. 1997.
- [253] Steinhilper, R; "Design for recycling and remanufacturing of mechatronic electronic products: challenges, solutions and practical examples from the European viewpoint". ASME Design for Manufacturability Conference, Chicago, IL, pp. 65-67. 1994.
- [254] Stock, JR; "Reverse Logistics", Oak Brook, IL: Council of Logistics Management, 1992.

- [255] Strang, G; "Linear Algebra and Its Applications", Academic Press, New York. 1976.
- [256] Subramani, AK; Dewhurs, P; "Automatic generation of product disassembly sequences". Annals of the CIRP 1991, Vol. 40(1), pp. 115-118. 1991.
- [257] Suga, T; Hosoda, N; "Active Disassembly and Reversible Interconnection", Proceedings of the 2000 IEEE International Symposium on Electronics and the Environment , pp. 330-334. 2000.
- [258] Takahashi, O; Schilling, R; "Motion planning in plane using generalised Voronoi diagrams". IEEE Robotics and Automation. 1989.
- [259] Taleb, KN; Gupta, SM; "Disassembly of multiple product structures", Computers and Industrial Engineering, Vol. 32(4), pp. 949-961 1997.
- [260] Taleb, KN; Gupta, SM; Brennan, L; "Disassembly of complex products with parts and material commonality", Production Planning and Control, Vol. 8(3), pp. 255-269. 1997.
- [261] Tang, Y; Zhon, MC; Caudill, RJ; "An integrated approach to disassembly planning and demanufacturing operation", Proceedings of the IEEE International Symposium on Electronics and the Environment, pp. 354-359. 2000.
- [262] Tang, Y; Zhou, M; Zussman, E; Caudill, R; "Disassembly modelling, planning, and application: A review", Proceedings of the 2000 IEEE International Conference on Robotics & Automation, San Francisco, CA, pp. 2197-2202. 2000.
- [263] Tani, K; Güner, E; "Concept of an autonomous disassembly system using behaviour-based robotics", Advanced Robotics, Vol. 11(2), pp. 187-198. 1997.
- [264] Thierry, M; Salomon, M; Van Nunen, J; Van Wassenhove, L; "Strategic issues in product recovery management", California Management Review, Vol. 37(2), pp. 114-135. 1995.
- [265] Thurston, DL; "Internalising environmental impacts in design", ASME Design for Manufacturability Conference, Chicago, IL, pp. 107-113. 1994.
- [266] Tondu, B; Bazaz, SA; "The three-cubic method: An optimal online robot joint trajectory generator under velocity, acceleration, and wandering constraints", The International Journal of Robotics Research, Vol. 28(9), pp. 893-901. 1999.
- [267] Torres, F.; Puente, ST; Gil, P; Candelas, FA; Ortiz, FG; "MOGEDA: Modelo Genérico de Desensamblado Automático", XXI Jornadas de Automática, Sevilla. 2000.
- [268] Torres, F; Puente, ST; Aracil, R; "Disassembly planning based on precedence relations among assemblies", International Journal of Advanced Manufacturing Technology, aceptado-pendiente de publicación, 2003
- [269] Torres, F; Puente, ST; Cánovas, J; Mangas, J; Martinez-Larraz, C; "Web teleoperation of robots with simulation feedback". Proceedings of the SPIE: Telemanipulator and Telepresence Technologies VIII, Vol. 4570, pp. 105-112. 2002.
- [270] Umeda, K; Arai, T; "Three-dimensional vision system for mechanical assembly/disassembly", Advanced Robotics, Vol. 11(2), pp. 147-167. 1997.

- [271] Uzsoy, R; "Production planning for companies with product recovery and remanufacturing capability", Proceedings of the IEEE International Symposium on Electronics and the Environment, San Francisco, CA, pp. 285-290. 1997.
- [272] Van Der Laan, EA; Dekker, R; Salomon, M; "Product remanufacturing and disposal: A numerical comparison of alternative control strategies", International Journal of Production Economics, Vol. 45(1-3), pp. 489-498. 1996.
- [273] Van Der Laan, EA; Dekker, R; Salomon, M; Ridder, A; "An (s,Q) inventory model with remanufacturing and disposal", International Journal of Production Economics, Vol. 46-47, pp. 339-350. 1996.
- [274] Van Der Laan, EA; Salomon, M; "Production planning and inventory control with remanufacturing and disposal", European Journal of Operational Research, Vol. 102, pp. 264-278. 1997.
- [275] Veerakamolmal, P; Gupta, SM; "Design of an integrated recovery system". Proceedings of the IEEE International Symposium on Electronics and the Environment, Oak Brook, IL, pp 264-269. 1998.
- [276] Veerakamolmal, P; Gupta, SM; "Disassembly process planning". Proceedings of the International Conference on Engineering Design and Automation, pp. 162-165, 1997.
- [277] Veerakamolmal, P; Gupta, SM; "High-mix/low-volume batch of electronic equipment disassembly". Computers and Industrial Engineering, Vol. 35(1-2), pp. 65-68. 1998.
- [278] Veerakamolmal, P; Gupta, SM; "Planning components recovery from multiple products". Northeast Decision Sciences Institute -27<sup>th</sup> Annual Meeting, Boston, MA, pp. 270-272. 1998.
- [279] Veerakamolmal, P; Gupta, SM; McLean, CR; "Disassembly process planning". Proceedings of the International Conference on Engineering Design and Automation, Bangkok, Thailand, pp. 162-165. 1997.
- [280] Veroutic, AD; Fava, JA; "Framework for the development of metrics for design for environment assessment of products", Proceedings of the IEEE International Symposium on Electronics and the Environment, pp. 13-24. 1996.
- [281] Vujosevic, R; Raskar, R; Yetukuri, NV; Jothishankar, MC; Juang, SH; "Simulation, animation and analysis of design disassembly for maintainability analysis", International Journal of Production Research, Vol. 33(11), pp. 2999-3022. 1995.
- [282] Warren, CW; "Fast path planning using modified A\* method". IEEE Robotics and Automation, Atlanta, USA. 1993.
- [283] Watkins, RD; Kleban, SD; Luger, GF; "Expert system support for environmentally conscious manufacturing", International Journal of Environmentally Conscious Design & Manufacturing, Vol. 4(2), pp. 53-66. 1995
- [284] Watson, JG; "Sustainable development by design: review of life cycle design and related approaches", Air & Waste: Journal of Air & Waste Management Association, Vol. 44(9), pp. 1083-1088. 1994.
- [285] Weule, H; "Life-cycle analysis: a strategic element for future products and manufacturing technologies", Annals of the CIRP, Vol. 42, pp. 181-184. 1993.

- [286] Williams, J; Shu, L; "Analysis of toner-cartridge remanufacturer waste stream", Proceedings of the IEEE International Symposium on Electronics and the Environment, pp. 260-265. 2000.
- [287] Wilson, RH; Latombe, JC; "Geometric reasoning about mechanical assembly". Artificial Intelligence, Vol. 71(2), pp. 371-396. 1994.
- [288] Woo, TC; "Visibility maps and spherical algorithms". Computer-Aided Design, Vol. 26(1), pp. 6-16. 1994.
- [289] Woo, TC; Dutta, D; "Automatic Disassembly and Total Ordering in Three Dimension". ASME Journal of Engineering for Industry, pp. 102-109. 1991.
- [290] Xirouchakis, P; Kiritisis, D; "Petri net modelling for disassembly process planning". Concurrent Product Design and Environmentally conscious Manufacturing-ASME, 94(5), pp. 255-262. 1997.
- [291] Xu, Y; Mattikalli, R; Khosla, P; "Generation of partial medial axis for disassembly motion planning", Journal of Design and Manufacturing, Vol. 5(2), pp. 89-102. 1995.
- [292] Yan, X; Gu, P; "A graph based heuristic approach to automated assembly planning". Flexible Assembly Systems, Vol. 73, pp. 97-106. 1994.
- [293] Yokota, K; Brough, DR; "Assembly/Disassembly Sequence Planning", Assembly Automation, Vol. 12(3), pp. 31-38. 1992.
- [294] Young, SB; Vanderburg, WH; "Applying environmental life-fycle analysis to materials", JOM: The Journal of the Minerals, Metals & Materials Society, Vol. 46(3), pp. 22-27. 1994.
- [295] Zeid, I; Gupta, SM; "A computer Algorithm to Evaluate Product Disassembly Cost Index", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 23-31. 2002.
- [296] Zeid, I; Gupta, SM; Bardasz, T; "A case-based reasoning approach to planning for disassembly", Journal of Intelligent Manufacturing, Vol. 8(2), pp. 97-106. 1997.
- [297] Zeid, I; Gupta, SM; Bardasz, T; "An analogical problem solving approach to planning for disassembly", Proceedings of the Winter Annual Meeting of ASME (Design for Manufacturing and Assembly), Atlanta, GA, pp. 79-89. 1996.
- [298] Zha, XF; Lim, SYE; "Assembly/disassembly task planning and simulation using expert Petri nets", International Journal of Production Research, Vol. 38(15), pp. 3639-3676, 2000.
- [299] Zhang, HC; Kuo, TC; "A graph-based approach to disassembly model for end-of-life product recycling". Proceedings of the IEEE/CPMT International Electronics Manufacturing Technology Symposium, pp. 247-254. 1996.
- [300] Zhang, HC; Kuo, TC; "A Graph-Based Disassembly Sequence Planning for EOL Product Recycling", IEEE/CPMT Electronics Manufacturing Technology Symposium, pp. 140-151, 1997.
- [301] Zhang, HC; Kuo, TC; Huang, SH; "Recycling model for end-of-life products", Japan/USA Symposium on Flexible Automation, pp. 1127-1130. 1996.
- [302] Zhang, HC; Kuo, TC; Lu, H; Huang, SH; "Environmentally conscious design and manufacturing: a state of the art survey", Journal of Manufacturing Systems, Vol. 16(5), pp. 252-271. 1997.

- 
- [303] Zorc, S; "Learning in assembly planning", IFAC Intelligent Assembly and Disassembly, Bled, Slovenia, pp. 17-22, 1998
- [304] Zuidwijk, R; Krikke, H; "Disassembly for Recovery under Uncertainty", Proceedings of SPIE: Environmentally Conscious Manufacturing II, Vol. 4569, pp. 44-53. 2002.
- [305] Zuo, BR; Stenzel, A; Seliger, G; "Flexible handling in disassembly with screwnail indentation", Proceedings of the IEEE International Conference on Robotics and Automation, Vol. 4, pp. 3681-3686. 2000.
- [306] Zussman, E; "Planning of disassembly systems". Assembly Automation, Vol. 15(4), pp. 20-23. 1995.
- [307] Zussman, E; Kriwet, A; Seliger, G; "Disassembly-oriented assessment methodology to support design for recycling". Annals of the CIRP, Vol. 43(1), pp. 9-14. 1994.
- [308] Zussman, E; Lenz, E; Shpitalni, M; "An Approach to the Automatic Assembly Planning Problem", Annals of the CIRP, Vol. 39(1). 1990.
- [309] Zussman, E; Zhou, M; Caudill, R; "Disassembly Petri net approach to modelling and planning disassembly process of electronics products", Proceedings of the IEEE International Symposium on Electronics and the Environment, Oak Brook, IL, pp. 331-336. 1998.
- [310] Zussman, E; Zhou, MC; "Design and implementation of an adaptative process planner for disassembly processes", IEEE Transactions on Robotics and Automation, Vol. 16(2), pp. 171-179. 2000.
- [311] Zust, R; Wagner, R; "Approach to the identification and quantification of environmental effects during product life", Annals of the CIRP, Vol. 1.41(1), pp. 473-476. 1992.

# Resumen

Esta tesis presenta una nueva perspectiva para abordar el problema del desensamblado no destructivo de un producto. Se plantea, para llevar a cabo las tareas necesarias, un sistema flexible de desensamblado automático. Este sistema utiliza una representación de los productos basada en una jerarquización de los componentes que los forman; para generar, basándose en dicha jerarquía, la secuencia de componentes a separar, que permiten desensamblar el componente o el conjunto de componentes deseado. Dicha secuencia se genera en función de una serie de acciones que dependen de las relaciones existentes entre los componentes. Para cada componente de la secuencia se calcula: cuál es la trayectoria de movimientos que debe seguir para separarse del resto; incluyendo, en el cálculo de esta trayectoria, el robot que realiza el desensamblado. Los movimientos se calculan utilizando un modelo geométrico de los componentes. Una vez generada esta información, se fusiona en un sistema real, utilizando un sistema de visión artificial para el reconocimiento y localización de los componentes, gracias al cual se aporta flexibilidad al sistema de desensamblado. Por último se presenta el funcionamiento de la célula flexible de desensamblado sobre un sistema real; basada en un brazo robot con cinco grados de libertad, y se evalúa sobre dos tipos distintos de PC's: uno sobremesa y otro semitorre.

# Abstract

This PhD thesis presents a new perspective for approaching the problem of the non-destructive disassembly of a product. A representation of the products, based on a hierarchy of its components is considered here to carry out the required tasks. This generated the disassembly sequence for the components that are to be removed, which permits the disassembly of the required component or set of components. This sequence is generated considering a series of actions that depends on the relationships that exists among the different components. For each component in the sequence, the trajectory of movements that should be follow for the removal of the remaining components is calculated, including the robot that carries out the disassembly, in the calculation of the trajectory. The movements are calculated using a geometric model of the components. Once this information has been generated, it is merged into a real system, using an artificial vision system to recognize and located the components, from which the disassembly system gets its flexibility. Finally, the behaviour of the flexible disassembly cell is presented within a real system, using this cell to carry out the disassembly of two different types of PCs, a desktop and a upright, by means of a robot arm with five degrees of freedom.