

Aspectos técnicos de los robots co-trabajadores con operadores humanos

Raúl Suárez, Jan Rosell

Recibido: 8 de Marzo de 2019 / Aceptado: 10 de Abril de 2019

Resumen

La robótica colaborativa, en la que los robots comparten el espacio de trabajo con los humanos e interactúan con ellos, toma un papel cada vez más relevante. En este contexto, el artículo presenta desarrollos realizados en diferentes aspectos técnicos necesarios para que estos robots funcionen adecuadamente y cumplan las expectativas en cuanto a su integración. En particular, se enfatizan las capacidades de planificación autónoma de tareas y movimientos, de manipulación diestra, de manipulación bimanual y de interacción con los humanos. Todos los desarrollos presentados han sido validados experimentalmente, y en el artículo se incluyen ejemplos ilustrativos de los experimentos realizados.

Palabras clave

Robótica, manipuladores móviles, robots co-trabajadores, cooperación humano-robot.

1. Introducción

El campo de aplicación de la robótica crece constantemente a medida que los avances tecnológicos van permitiendo nuevas prestaciones. Entre las nuevas aplicaciones destaca, por su potencial trascendencia tanto desde el punto de vista productivo como desde el social, aquellas en las que los robots trabajan conjuntamente con los seres humanos, dando origen al concepto de “robot co-trabajador”. En estas aplicaciones los robots requieren características particulares. Por un lado, han de tener cierta autonomía y capacidad de decisión, ya que su función no es la de realizar tareas de forma repetitiva en el sentido clásico, sino que deberán adaptarse con cierta celeridad a condiciones cambiantes, sobretodo generadas por los operarios con los que comparten el entorno de actuación. Esta adaptación implica una capacidad de interacción con los operarios que condiciona los movimientos y forma de actuar de los robots, a fin de evitar posibles daños a la integridad física de los humanos, pero evidentemente manteniendo la eficiencia de sus acciones. Por otra parte, para que los robots co-trabajadores sean realmente útiles es necesario, además, que tengan un nivel de destreza suficiente que les permita ejecutar diferentes acciones usualmente requeridas en las actividades humanas.

En este contexto, se presentan aquí resultados de trabajos orientados al desarrollo de robots autónomos diestros como co-trabajadores con operadores humanos. Estos resultados pretenden incrementar las prestaciones de los robots co-tra-

bajadores para acercarlos a su implantación definitiva en nuestra sociedad, y facilitar también su aceptación por parte de los humanos.

Concretamente, se plantean avances en temas tan relevantes como:

- El aumento de la capacidad del robot para gestionar autónomamente las tareas y los movimientos específicos para llevarla a cabo, representando el conocimiento mediante ontologías.
- El incremento de la capacidad de manipulación diestra y bimanual, monitorizando las acciones para prevenir potenciales fallos.
- La mejora en la interacción con el operario humano, en referencia a movimientos reactivos, de cooperación y de transferencia de objetos.

Todos los desarrollos teóricos aquí presentados se han probado y validado experimentalmente mediante sistemas específicamente preparados para tal fin. Por otra parte, los problemas mencionados se abordan con la intención de aportar soluciones generales que sean válidas tanto en la robótica industrial como en la de servicios, previendo tanto la influencia puramente industrial de los resultados como la componente social de los mismos.

Tras esta introducción, el resto del artículo está estructurado de la siguiente manera. La sección 2 presenta la motivación y el alcance del trabajo. A continuación se presentan las diferentes capacidades que se requieren en los robots para conseguir los objetivos marcados: capacidad de gestión autónoma de tareas y movimientos (Sección 3), capacidad de manipulación diestra (Sección 4), capacidad de manipulación bimanual (Sección 5) y capacidad de interacción con



Raúl Suárez
raul.suarez@upc.edu

Jan Rosell
jan.rosell@upc.edu

Institut d'Organització i Control de Sistemes Industrials
(IOC - Divisió de Robòtica)
Universitat Politècnica de Catalunya (UPC)

los operadores humanos (Sección 6). Finalmente, la Sección 7 presenta las conclusiones.

2. Motivación

Aunque la introducción de robots avanza de manera imparable en distintos ámbitos, la disponibilidad de robots humanoides con capacidades “casi humanas” y que realicen tareas avanzadas en el trabajo y en el hogar no es inmediata. Sin embargo, la existencia de robots manipuladores móviles con capacidades de manipulación diestra que puedan asistir al ser humano de manera versátil en distintas tareas simples es ya una realidad, por ejemplo, como transportadores logísticos o adaptaciones de manipuladores clásicos, y algunos fabricantes ya comercializan modelos que cumplen las normativas de seguridad requeridas para compartir espacio de trabajo con los seres humanos. En cualquier caso, lograr una adecuada efectividad, depende de mejoras de hardware y de software que permitan un grado elevado de adaptabilidad. El uso de robots como compañeros de trabajo es un salto evolutivo en la robótica industrial. La disponibilidad de robots que puedan realizar distintas tareas, autónomamente o en cooperación, y compartir de forma segura el espacio con los humanos puede permitir incrementar la producción y los puestos de trabajo. Los robots co-trabajadores necesitan la interacción y supervisión humanas, pueden considerarse como herramientas avanzadas y es posible establecer con ellos una relación simbiótica, permitiendo realizar más tareas en menos tiempo, con mayor calidad y de forma más comfortable. La propuesta de introducir en el entorno de trabajo robots co-trabajadores tiene un carácter disruptivo que puede tener efectos importantes sobre el modelo productivo actual, incluyendo el comportamiento y la convivencia laboral. Por ello, en los desarrollos en este campo hay que tener en cuenta la importancia de los comportamientos y percepciones sociales que esta tecnología pueda generar, ampliando sus ventajas y minimizando los riesgos asociados para que estos comportamientos propicien el proceso de innovación, que debe conllevar un aumento de la productividad y de la calidad, así como una mejora de las condiciones de trabajo. En este sentido, deben implementarse sistemas en los que los robots co-trabajadores den soporte y se integren en las tareas de los operarios de manera fluida, natural, mínimamente invasiva y segura. Ahora bien, para que el concepto de robots co-trabajadores se consolide, tanto en el ambiente industrial como en el de servicios, aún se deben realizar avances científicos y tecnológicos en determinados temas afines. Por ejemplo, aunque ya existen propuestas y ensayos experimentales, la manipulación diestra y la cooperación bimanual en su sentido más riguroso aún son temas de trabajo en centros de investigación, así como la capacidad de cooperación con el operador humano en cuanto a planificación simultánea de movimientos y tareas para una adecuada interacción, desde el punto de seguridad, de eficiencia, y también de confort para el humano. Por lo tanto, resulta razonable y pertinente trabajar en estos aspectos.

Así, entre las motivaciones más relevantes para abordar los temas que se describen en las secciones siguientes podemos mencionar:

- Los robots co-trabajadores se presentan actualmente como una de las herramientas de alto nivel más versátiles y útiles;
- Se dispone en el mercado de robots manipuladores móviles de prestaciones mecánicas aceptables, pero que tienen fuertes limitaciones en cuanto a destreza y capacidad de interacción con el operario humano;
- Entre las causas de estas limitaciones están las carencias en la capacidad de manipulación diestra, por ejemplo explotando inadecuadamente el uso de información táctil, y en la capacidad de cooperación segura y eficiente con el humano, por ejemplo no planificando adecuadamente las tareas y movimientos en concordancia con los del humano.

3. Capacidad de gestión autónoma de tareas y movimientos

Muchos retos en robótica, como las tareas de manipulación en entornos desordenados, requiere de la planificación a nivel de tarea y a nivel de movimientos. La planificación a nivel de movimientos se centra en la búsqueda de caminos libres de colisión entre dos configuraciones del robot, y usualmente se realiza en el espacio de configuraciones mediante métodos basados en muestreo (García et al., 2017) y pueden requerir realimentación tanto de configuración como de fuerza cuando se prevé contacto (Suárez et al. 1995). Diferentes variantes permiten la inclusión de distintos tipos de restricciones, e incluso permiten la interacción con objetos móviles del entorno, como los planificadores basados en la física que consideran las interacciones dinámicas (robot-objeto y objeto-objeto) durante la planificación. La planificación a nivel de tarea, por el contrario, se centra en buscar una secuencia de acciones que permita la ejecución de una determinada tarea. Usualmente se modela como un sistema de transición de estados y puede resolverse, por ejemplo, usando métodos de satisfacción de restricciones o métodos heurísticos de búsqueda en el espacio de estados, como el método Fast Forward (FF) (Hoffmann y Nebel, 2001).

Las tareas a abordar pueden tener diferentes grados de dificultad, tanto a nivel simbólico (dimensión del espacio de estados) como geométrico (conectividad de las zonas libres del espacio de configuraciones), así como de interacción entre ambos niveles. Los estados y acciones a nivel simbólico deben implementarse a nivel geométrico, siendo las acciones ejecutables solo si las implementaciones geométricas satisfacen las precondiciones de la acción y si la acción es geoméricamente factible. Por ello, es necesaria una combinación eficiente entre los niveles de planificación para hacer que el proceso sea eficiente.

Asimismo, el uso del conocimiento y de técnicas de razonamiento pueden ayudar a las capacidades de planificación a ambos niveles, dando más autonomía a los robots. El conocimiento puede modelarse usando ontologías, que lo representan mediante conceptos y relaciones. Las ontologías pueden implementarse mediante el Web Ontology Language (OWL) (McGuinness et al., 2004), haciéndolo una base de datos accesible a través de la red, que puede ser usada para el razonamiento mediante predicados escritos en lenguaje Prolog.

Unas de las tareas de manipulación de gran interés en el marco de los robots co-trabajadores es la manipulación de objetos usando un robot bi-brazo. En estas tareas se debe hacer frente a la existencia de objetos que bloquean los movimientos del robot, requiriendo una correcta selección de las configuraciones de prensión y de regiones para depositar los objetos manipulados, y garantizando que las acciones sean realizables. Asimismo, en función del tipo de problema a resolver, el orden en que se resuelven los objetivos parciales debe ser analizado cuidadosamente, ya que es posible la generación de espacios de estados muy grandes y soluciones que impliquen mover varias veces los mismos objetos para conseguir el objetivo final deseado.

En esta sección se describe un marco basado en ontologías para modelar tareas de manipulación y un planificador simultáneo de tareas y de movimientos basado en el conocimiento llamado k-TMP. Este método combina un planificador de tareas heurístico Fast Forward con un planificador de movimientos basado en la física (que permite la realización de tareas de empuje de objetos sin sujetarlos) y con ontologías. Tanto el planificador de movimientos como el conocimiento modelado con las ontologías se usan en procesos de razonamiento para determinar la factibilidad de las acciones y guiar al planificador de tareas en la búsqueda del plan de acciones. El planificador de movimientos basado en la física también se usa en acciones de prensión en entornos desordenados cuando las posiciones finales de los otros objetos no son relevantes, ya que este planificador permite de manera natural apartar objetos que estorban en el camino hacia la prensión de un objeto predeterminado.

3.1. Planificación combinada de tareas y movimientos

Considérese un robot bi-brazo capaz de manipular objetos (de distintas formas y tamaños) en una mesa, es decir de sujetarlos (posiblemente con diferentes agarres) y de dejarlos (en diferentes sitios como bandejas, estantes, o zonas intermedias), así como de desplazarlos empujándolos. En este escenario se pueden plantear distintas tareas, que van desde la simple prensión de un objeto determinado hasta la reordenación de varios objetos. Para encontrar la secuencia de acciones factible que permita solventar la tarea deben superarse distintos retos, como la detección de los objetos bloqueantes,

la selección de los agarres que son cinemáticamente posibles y libres de colisiones tanto al sujetar como al dejar un objeto, la determinación de dónde dejar los objetos teniendo en cuenta la alcanzabilidad y las posibles limitaciones de espacio, y la determinación de caminos libres de colisión del robot, moviéndose sin carga o en el proceso de manipulación de un objeto.

Para abordar estos problemas se propone un marco para combinar los niveles de planificación basado en:

- La definición de ontologías de manipulación: La formalización del conocimiento de manipulación que incluye información de los objetos, robots, sensores y acciones, el entorno y el contexto, así como un mecanismo de inferencia para razonar sobre este conocimiento.
- La planificación para la prensión: Un planificador de movimientos basado en la física capaz de buscar trayectorias que incluyan la interacción con objetos móviles, permitiendo apartar aquellos que interfieren en el camino al objeto a sujetar, permitiendo prensiones simples y eficientes.
- Planificación de la manipulación: Un planificador de tareas basado en la búsqueda heurística en el espacio de estados, en el que se guía la búsqueda en base a procesos de razonamiento geométrico que determinan la factibilidad y aplicabilidad de las acciones.

En las siguientes subsecciones se detallan estos puntos.

3.2. Ontologías para la manipulación

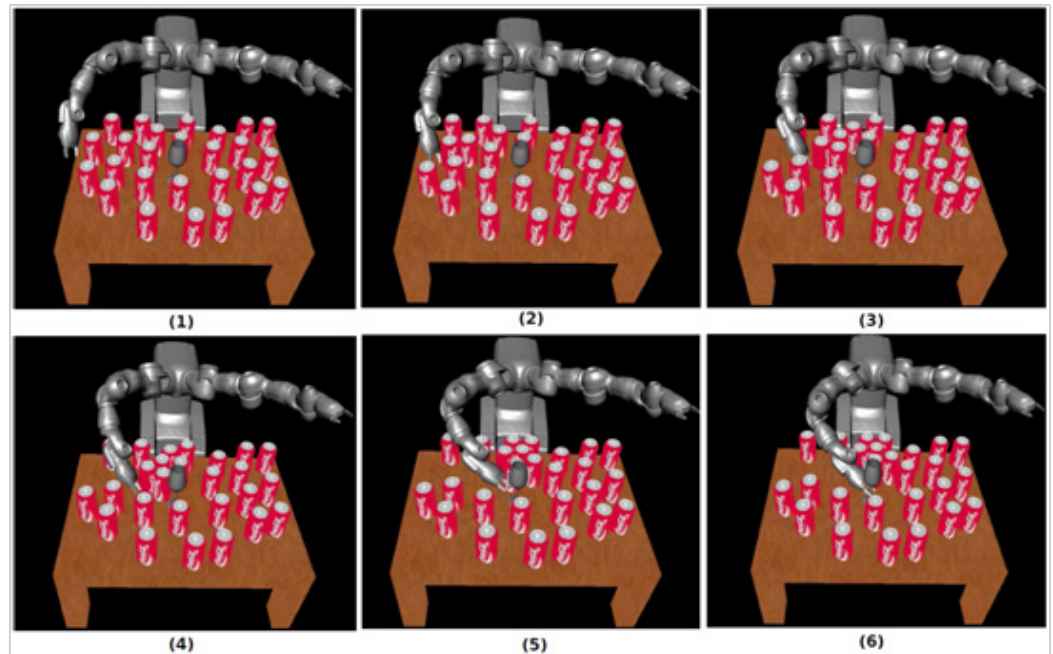
El marco propuesto para la representación del conocimiento necesario para tareas de manipulación usa un planificador de movimientos basado en la física, que automáticamente permite la categorización de los objetos en diferentes tipos en base a las restricciones de los objetos y de la tarea (Diab et al., 2017). La propuesta usa vocabulario común dado por el Ontologies for Robotics and Automation Working Group (ORA WG), auspiciado por IEEE Robotics & Automation Society (Olszewska et al., 2017), basado en una jerarquía de ontologías de tres niveles: metaontology (para representar información general, como el concepto de objeto rígido), ontology schema (para representar conocimiento de un dominio específico, como un entorno de oficina) y ontology instance (para almacenar la información de objetos particulares). Estos niveles se subdividen en clases para representar el conocimiento relativo a: Características (propiedades de los objetos relevantes para la manipulación), Objeto (objetos físicos y sus partes y agrupaciones), Actor (robots y sus componentes), Espacio (espacio de trabajo, relaciones topológicas), Contexto (conocimiento relativo a la situación temporal y espacial) y Acción (algoritmos de planificación).

3.3. Planificación de movimientos de prensión

Para la planificación de movimientos de prensión se presenta un planificador basado en la física que permite abordar la búsqueda de trayectorias robustas del robot en entornos donde puede haber interacciones con los objetos y donde hay incertidumbre en las posiciones de los objetos y en los parámetros de interacción (Muhayyuddin et al., 2018). La propuesta se basa en el planificador KPIECE (Sucan y Kavraki, 2012), un planificador basado en muestreo que explora el espacio de configuraciones mediante un árbol guiado mediante una proyección en un subespacio discretizado donde se evalúa la exploración (el proceso de expansión escoge con mayor probabilidad aquellas ramas cuyas celdas asoci-

adas tienen un mayor índice de calidad). Se trabaja en dos fases: muestreo y cálculo de certeza. Una vez seleccionada la rama del árbol que debe crecer, se seleccionan aleatoriamente direcciones de crecimiento generando potenciales extensiones. En la fase de evaluación de la certeza se considera incertidumbre (en el posicionamiento de los objetos, en las direcciones de crecimiento y en los parámetros de interacción) y se estima la tasa de éxito que la ejecución de cada extensión reportaría, seleccionando la mejor. La certeza de la nueva rama se usa para actualizar el índice de la celda del subespacio asociado, promoviendo un sesgo hacia zonas con más certeza. Este método se ha probado con diferentes robots (Kuka LWR y ABB YuMi) y con diferentes escenas con muchos objetos (Figura 1).

Figura 1 Ejemplo de tarea de prensión en un entorno desordenado.



3.4. Planificación de la manipulación

Para la planificación de tareas de manipulación se propone un planificador simultáneo de tareas y movimientos basado en el planificador de tareas Fast Forward (FF), que realiza una búsqueda heurística en el espacio de estados, en el que se incorpora información geométrica en el cálculo de la heurística y se incluyen llamadas al planificador de movimientos en las acciones que llevan a los estados explorados (Akbari et al., 2018). El enfoque propuesto tiene tres partes principales: el cálculo de la heurística, la búsqueda en el espacio de estados y el proceso de selección de las acciones.

La heurística usada en el planificador FF se estima como la distancia al estado final, calculada como el número de acciones de un plan obtenido mediante una versión relajada del Grafo de Planificación (Blum y Furst, 1997), llamada RPG. La modificación propuesta incluye un análisis geométrico de las acciones del plan relajado basado en averiguar, por ejem-

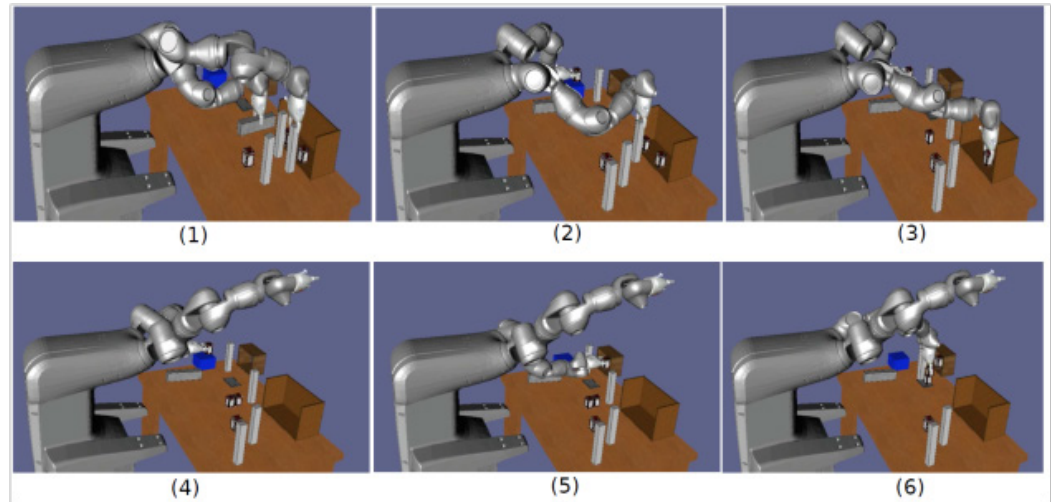
plo, si existen soluciones de la cinemática inversa para las configuraciones de agarre, si el robot colisiona o no con el entorno en dichas configuraciones, o si existe espacio suficiente para poder dejar los objetos. Si todas las acciones del plan relajado son factibles, entonces el valor de la heurística es el número de acciones de dicho plan. En caso contrario, se identifica la razón y se actualiza el estado (por ejemplo, etiquetando un objeto como obstáculo) y se recalcula un nuevo plan relajado. Este proceso de razonamiento permite una búsqueda informada en términos geométricos.

La búsqueda en el espacio de estados se hace, igual que el FF estándar, mediante el algoritmo Enhanced Hill Climbing (EHC) y alternativamente con el algoritmo Best First Search (BFS). Desde cada estado el módulo selecciona el estado sucesor que tiene un valor de la heurística menor.

El proceso de selección de la acción busca un camino para una acción usando el planificador de movimientos adecuado (un planificador geométrico para encontrar caminos libres de colisión y un planificado basado en la física (physics based) para movimientos de empuje). Si la planificación de movimientos falla, el estado se actualiza y recomienza la búsqueda. En caso de éxito se almacena el camino juntamente con la acción.

La propuesta se ha validado con éxito en tareas de sobremesa ejecutadas con el robot bi-brazo YuMi (Figura 2). La tarea requiere quitar diferentes objetos que bloquean el paso, seleccionar con tino el brazo que realiza cada acción, seleccionar adecuadamente las configuraciones de agarre de las piezas, y ordenar adecuadamente la consecución de objetivos.

Figura 2 Ejemplo de una tarea de manipulación ejecutada con el robot YuMi.

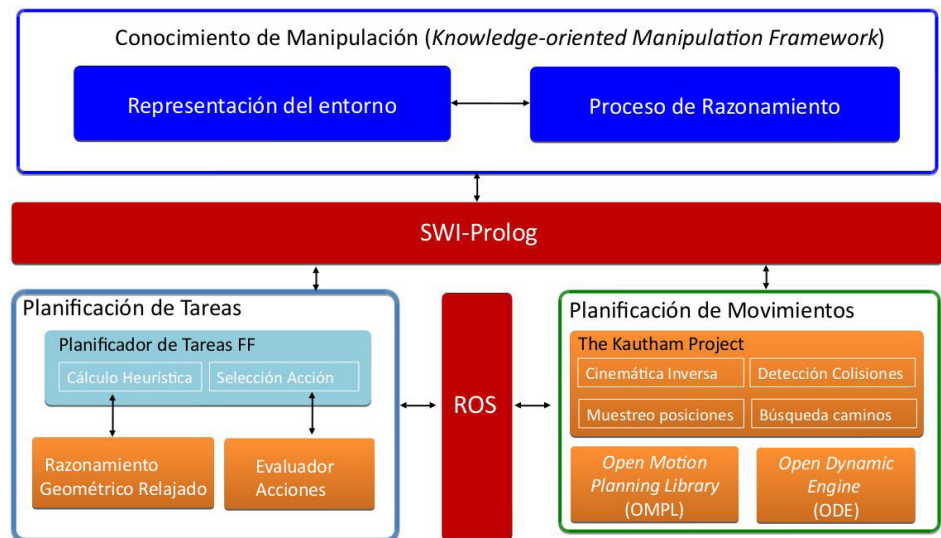


3.5. Implementación

El marco de planificación propuesto se ha implementado en tres partes (Figura 3): planificación de la tarea, planificación de movimientos y procesado del conocimiento. La planificación de la tarea se ha implementado modificando una versión de planificador FF en C++ para incluir la información geométrica y el procesado del conocimiento. La planificación de movimientos se ha hecho usando The Kautham

Project (Rosell et al., 2014), que permite la planificación de movimientos basándose en la librería Open Motion Planning Library (Sucan et al., 2012). El procesado del conocimiento se ha realizado en Prolog, y la generación de las ontologías mediante el editor Protege. Todas las comunicaciones se han hecho usando el Robotic Operating System (ROS) (Quigley et al., 2009).

Figura 3 Implementación: componentes y comunicaciones.



4. Capacidad de manipulación diestra

Un robot co-trabajador debe tener en sí mismo capacidades avanzadas de manipulación, lo que incluye aspectos tales como el uso coordinado de dos brazos para llevar a cabo una tarea (sea a cadena cinemática abierta o cerrada cuando dos brazos sujetan un único objeto) o la capacidad de manipular objetos de forma diestra. En este contexto, el concepto de manipulación diestra se refiere a la manipulación de objetos dentro del espacio de trabajo de la mano usando solamente los dedos (in-hand manipulation).

En entornos de trabajo reales generalmente el modelo del objeto a manipular solo es parcialmente conocido, o incluso totalmente desconocido, lo que significa que no se dispone de su modelo para precalcular los movimientos de manipulación. En estos casos, una solución útil consiste en utilizar sensores táctiles que ayudan al reconocimiento del objeto en los puntos de contacto y por lo tanto a construir su modelo geométrico.

Se describe a continuación un enfoque propuesto para realizar manipulación diestra con dos dedos (Montaño y Suárez, 2015, 2018), cuyas principales asunciones son:

- La mano robótica tiene un sistema de sensores táctiles que permite obtener información del punto de contacto y de la fuerza ejercida sobre el objeto manipulado, y no se dispone de otros sensores externos como, por ejemplo, realimentación visual. Si se dispone de información sobre la cinemática de la mano y de la posición de sus articulaciones (información propioceptiva).
- Se utilizan dos dedos para manipular el objeto emulando la sujeción mediante los dedos pulgar e índice de la mano humana.
- No se tiene información previa sobre la forma del objeto, pero se asume que es un sólido rígido.
- No se conoce el coeficiente de fricción, pero se asume que está por encima de un valor crítico prefijado en función del material de la yema del dedo, que es conocido.

- La mano robótica utilizada tiene un sistema de control de posición de las articulaciones de los dedos, por lo que el enfoque propuesto cierra un bucle de control de más alto nivel que genera comandos de posición para este sistema de control.

La manipulación diestra persigue normalmente tres objetivos, individual o conjuntamente:

- Desde el punto de vista de la mano, la optimización de su configuración, es decir, que la configuración sea lo más próxima posible a una configuración predeterminada. Frecuentemente, se considera como configuración predeterminada la definida por la posición media de las articulaciones a fin de dar, potencialmente, máxima capacidad de movimiento. Para esta optimización se minimiza el siguiente índice (Liégeois 1977):

$$I_{hc} = \sum_{i=1}^2 \sum_{j=1}^{n_i} \left(\frac{q_{ij} - q_{oij}}{q_{maxij} - q_{minij}} \right)^2 \quad [1]$$

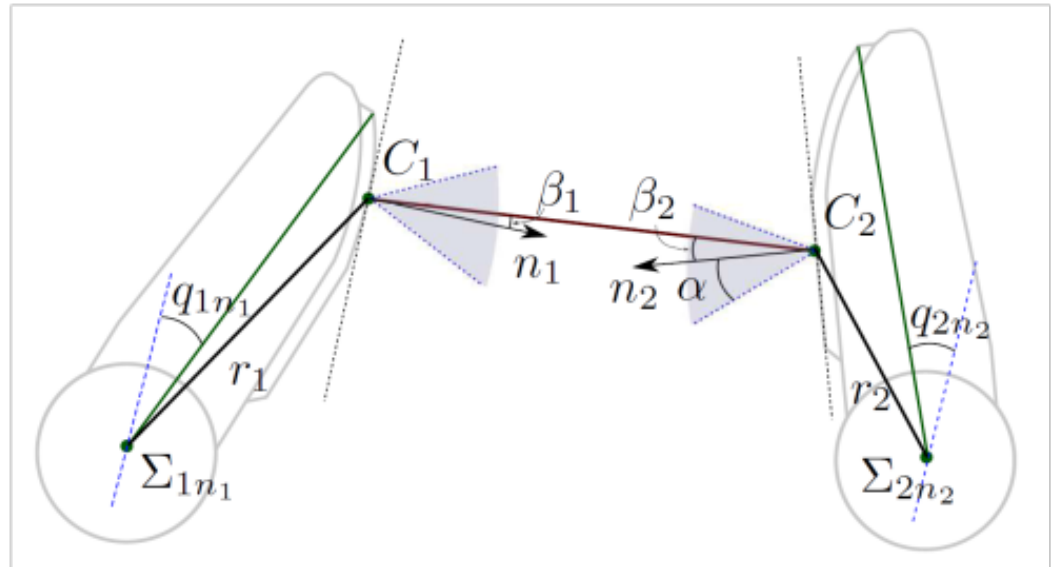
donde q_{ij} es la posición de la articulación j del dedo i , q_{oij} su valor deseado (predefinido), y q_{maxij} y q_{minij} sus límites máximo y mínimo.

- Desde el punto de vista de la presión (relación objeto-mano), la optimización de la calidad de la presión, es decir, la búsqueda de presiones seguras en las que la mano pueda resistir mayores fuerzas externas aplicadas al objeto. Existen diferentes medidas de calidad para evaluar la presión de un objeto (Roa et al., 2008; Roa y Suárez, 2014). En este trabajo, para esta optimización se minimiza el siguiente índice:

$$I_{gq} = \frac{1}{2} \sum_{i=1}^2 |\beta_i| \quad [2]$$

donde β_i , $i \in \{1, 2\}$ es el ángulo entre el segmento que une los dos puntos de contacto y las normales n_i , en cada uno de ellos (ver Figura 4). Cuanto menores son los ángulos β_i mayor es el margen de seguridad entre la fuerza aplicada y el borde del cono de fricción.

Figura 4 Extremos de los dedos donde se puede apreciar las normales a las superficies de contacto y los ángulos β_i que indican cuán lejos están las fuerzas aplicadas de los extremos del cono de fricción.



- Desde el punto de vista del objeto, la optimización de su configuración, es decir, la búsqueda de una posición y orientación que satisfagan los requisitos de una tarea dada. En este trabajo los dedos se encargarán de orientar adecuadamente el objeto, mientras que el brazo es el encargado de corregir la posición. Para esta optimización se minimiza el siguiente índice:

$$I_{oe} = \left| \frac{\gamma_d - \gamma}{\gamma_d - \gamma_i} \right| \quad [3]$$

donde γ_d es la orientación deseada del objeto, γ es su orientación actual y γ_i es la orientación inicial.

El procedimiento general usado en cada iteración k para calcular las posiciones de los dedos deseadas en el paso $k+1$ puede resumirse como (véanse las Figuras 5 y 6):

Etapa de medición. Se mide la posición de las articulaciones de los dedos de ambas manos incluyendo la posición

de los respectivos puntos de contacto sobre los sensores táctiles, Q_k , así como la fuerza aplicada por los dedos sobre el objeto, F_k . Con esta información es posible calcular la posición absoluta de los puntos de contacto en el espacio C_{ik} y la distancia entre ellos d_k .

Aplicación de la estrategia deseada. Conocido el vector Q_k , se calculan unos nuevos puntos de contacto, C_{ik+1}^* , de forma que al mover los dedos hacia estas posiciones se produzca el efecto deseado (Montaño y Suárez, 2018).

Ajuste final. Conocida la fuerza actual F_k y la distancia actual entre los contactos en cada dedo d_k , se calcula la nueva distancia deseada d_{k+1} (el cálculo se detalla a continuación) apretando o relajando la presión sobre el objeto según corresponda, y, finalmente, la distancia entre los puntos C_{ik+1}^* se ajusta al valor d_{k+1} moviéndolos de forma simétrica respecto a su punto medio para obtener así los puntos finales C_k que realmente servirán de consigna para el movimiento de los dedos al pasar de la iteración k a la $k+1$.

Figura 5 Etapas en el cálculo de la configuración de los dedos Q_{k+1} en cada iteración

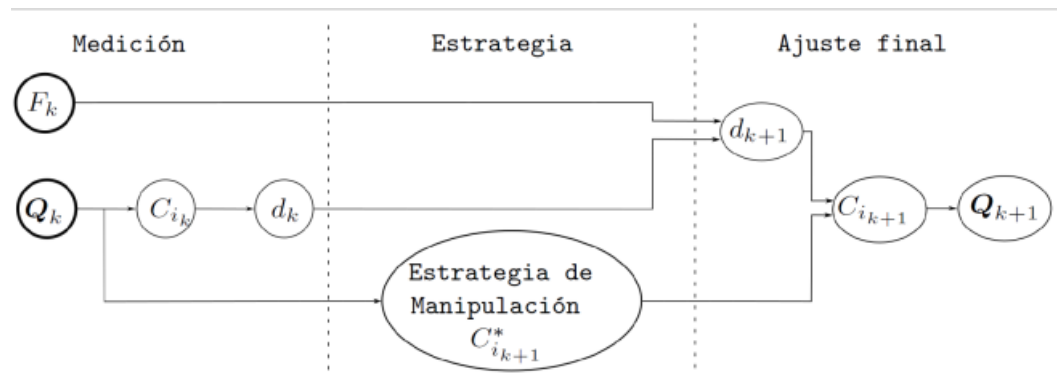
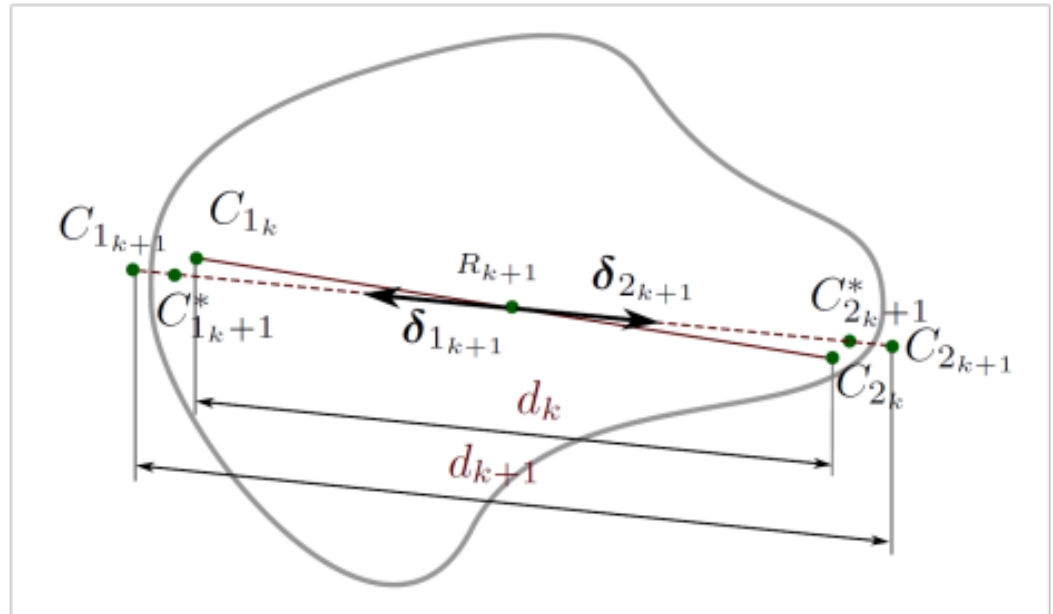


Figura 6 Ejemplo del cálculo de C_{ik+1} usando C_{ik+1}^* ajustando la distancia d_k a d_{k+1} cuando la fuerza de contacto F_k es mayor que la cota máxima prefijada F_{max} .



Dado que la forma del objeto desconocida, durante su manipulación las fuerzas pueden variar saliendo del rango de seguridad deseado, para evitar este efecto la distancia d se ajusta en cada iteración en función de la diferencia entre el valor de la fuerza de presión actual F_k y la fuerza deseada F_d , i.e. el error $e=F_k-F_d$. Así,

$$d_{k+1} = d_k + \Delta d \tag{4}$$

donde

$$\Delta d = \begin{cases} f_1(e) & \text{si } e \leq 0 \\ f_2(e) & \text{si } e > 0 \end{cases} \tag{5}$$

siendo $f_1(e)$ y $f_2(e)$ dos funciones arbitrariamente definidas por el usuario. En este trabajo se ha usado $f_1(e)=2\lambda(e+e^2)$ y $f_2(e) = \lambda e$, siendo λ un valor constante predefinido. La razón por la que $f_1(e)$, tiene una mayor ganancia según aumenta $|e|$, es que una potencial caída del objeto ($F_k \rightarrow 0$) se considera más una situación más crítica que una potencial aplicación de fuerzas de presión superiores a las deseadas ($F_k \gg F_d$), pero esto puede ajustarse fácilmente dependiendo de la aplicación.

Finalmente, debe destacarse que el proceso de manipulación iterativo puede finalizar por distintas causas:

- El índice de calidad utilizado alcanza su valor óptimo.

- El valor máximo actual del índice de calidad utilizado no ha mejorado pasado un número prefijado de iteraciones (no es necesario que el índice mejore monóticamente, puede decrecer u oscilar localmente).
- La presión esperada con los nuevos puntos de contacto calculados no satisface las restricciones de fricción y el objeto se puede caer.
- Los nuevos puntos de contacto calculados no pertenecen al espacio de trabajo de la mano y por lo tanto no son alcanzables.

En las Figuras 7 y 8 se ilustran dos ejemplos de manipulación diestra. En el primer caso se optimiza la calidad de la presión I_{gq} y la manipulación finaliza después de 5.460 s y 81 iteraciones debido a que I_{gq} ya no mejora su valor. En el segundo ejemplo se varía la orientación del objeto en ambos sentidos siguiendo unas consignas γ_d igual a 5, -5, 10, -10, 15, -15 y 20 grados; el cambio de consigna se realiza cada vez que se alcanza una condición de finalización, en el caso de las primeras 6 consignas I_{oe} alcanzó el valor óptimo, i.e. $\gamma_k=\gamma_d$ y en la última consigna el procedimiento se detuvo cuando en el próximo paso se esperaba que el ángulo β_2 excediera el límite del cono de fricción antes de llegar a $\gamma_k=20$ grados, lo que implica un riesgo de deslizamiento del objeto en los dedos con su consiguiente caída.

Figura 7 Optimización de la calidad de la prensión: a) Prensión inicial. b) Prensión final. c) Evolución del índice I_{gq} . d) Evolución de los valores de las articulaciones (en grados), q_{11} en azul, q_{12} en rojo, q_{21} en verde, q_{22} en magenta. e) Fuerza F_k (en Newtons, la línea a trazos indica F_d). f) Ángulos β_i (en grados), β_1 en azul y β_2 en rojo (la línea a trazos indica el valor óptimo $\beta_i=0$).

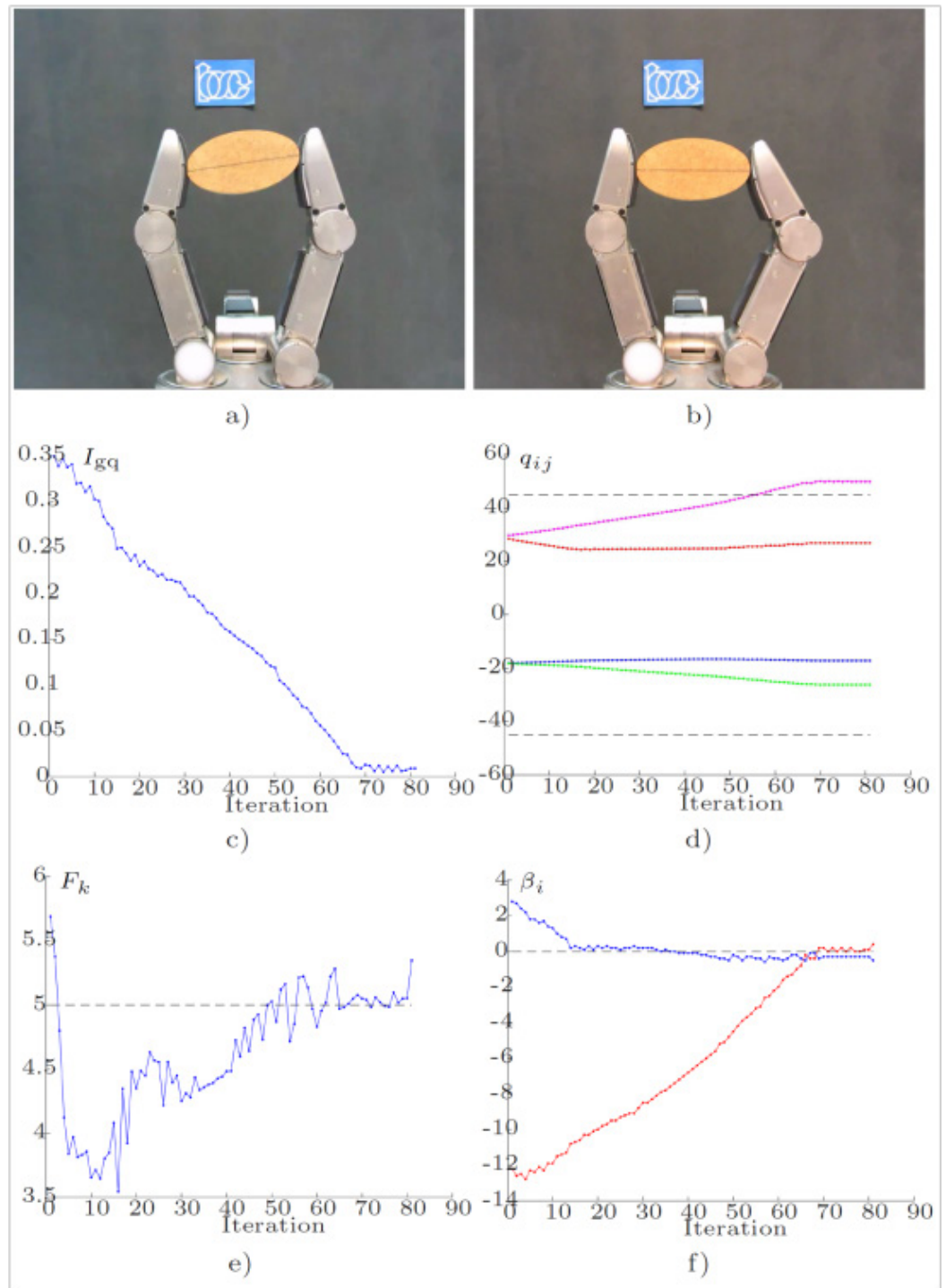
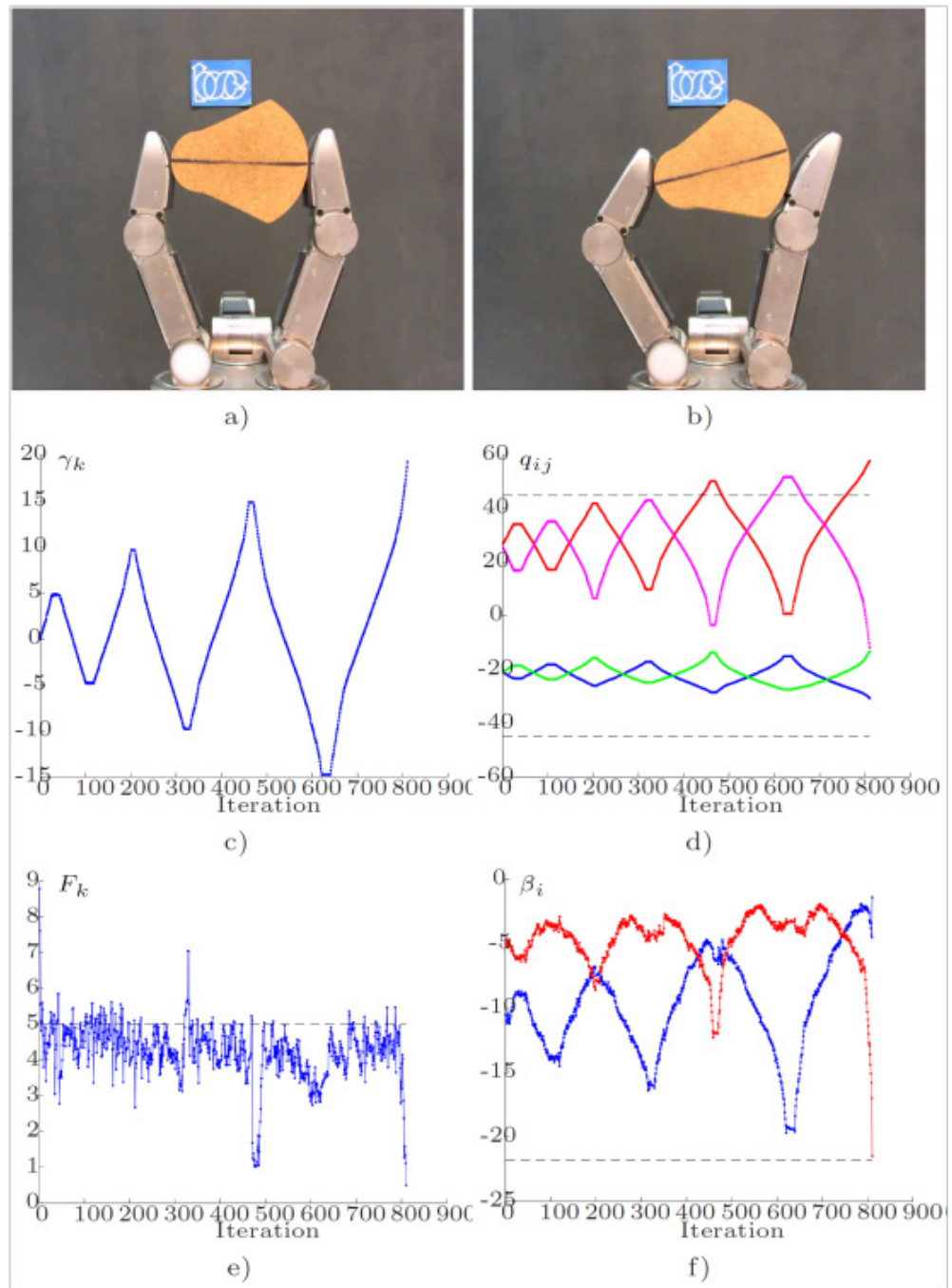


Figura 8 Optimización de la orientación del objeto: a) Prensión inicial. b) Prensión final. c) Evolución de la orientación γ_k . d) Evolución de los valores de las articulaciones (en grados), q_{11} en azul, q_{12} en rojo, q_{21} en verde, q_{22} en magenta. e) Fuerza F_k (en Newtons, la línea a trazos indica F_d). f) Ángulos β_i (en grados), β_1 en azul y β_2 en rojo (la línea a trazos indica el valor óptimo $\beta_i=0$).



Aparte de la manipulación diestra autónoma de objetos, el enfoque descrito se ha usado también en tareas de teleoperación de una mano robótica (Montaño y Suárez, 2016a, 2017). En esta aplicación se tiene un sistema de control compartido entre el operador y el sistema robótico remoto, en el que el operador solo envía comandos de alto nivel indicando en qué sentido se debe rotar el objeto, y el sistema remoto usa un grafo de estados y la información táctil y cinemática de la mano para controlar localmente las fuerzas de prensión y los movimientos de los dedos, generando así manipulaciones robustas y estables que evitan caídas del objeto independientemente de los posibles retardos que pudieran existir en el canal de comunicación.

5. Capacidad de manipulación bimanual

La manipulación bimanual puede entenderse de dos modos, uno en que las dos manos son parte de una cadena cinemática cerrada, por ejemplo, cuando sujetan un mismo objeto, y otro en que cada mano es el elemento terminal de una cadena cinemática independiente pero trabajando coordinadamente, por ejemplo, cuando las dos manos mueven diferentes objetos cooperando tras un mismo fin.

5.1. Sujeción bimanual de objetos voluminosos

Uno de los principales problemas abordados en manipulación bimanual es la sujeción robusta de objetos voluminosos que no pueden sujetarse con una sola mano. Se muestra

aquí un ejemplo de planificador de prensiones bimanuales basado en una segmentación del objeto (Rojas-de-Silva y Suárez, 2016; Rodríguez et al., 2016).

El objeto se representa mediante una nube de puntos 3D cuya información asociada incluye la dirección normal al objeto en cada punto. En el cálculo de la prensión se consideran contactos puntuales con fricción y la implementación realizada puede utilizar tres o cuatro dedos de cada mano, lo que da redundancia y mayor seguridad a la prensión.

Se describe a continuación el procedimiento de planificación particularizado para el uso de tres dedos de cada mano. El planificador busca los tres puntos de contacto alcanzables por cada mano dividiendo la nube de puntos que representa la superficie del objeto en dos conjuntos de rebanadas ortogonales a uno de los ejes de inercia del objeto. Se comienza por el eje de inercia que forma el menor ángulo con respecto al segmento definido por las bases de los brazos, es decir el segmento que va de "hombro a hombro", y si con este no se encuentra una solución se prueba con otro eje de inercia elegido con el mismo criterio. Para definir cada rebanada se usan planos ortogonales al eje de inercia usado y a una distancia predeterminada entre ellos, siendo el primero de estos planos tangente al objeto (Figura 9) y el último a una distancia del primero equivalente a la máxima distancia alcanzable por los dedos sobre un objeto cualquiera sin que éste penetre en palma de la mano (Figura 10).

Figura 9 Objeto representado por una nube de puntos y el primer plano ortogonal a un eje de inercia de los usados para dividir el objeto en rebanadas.

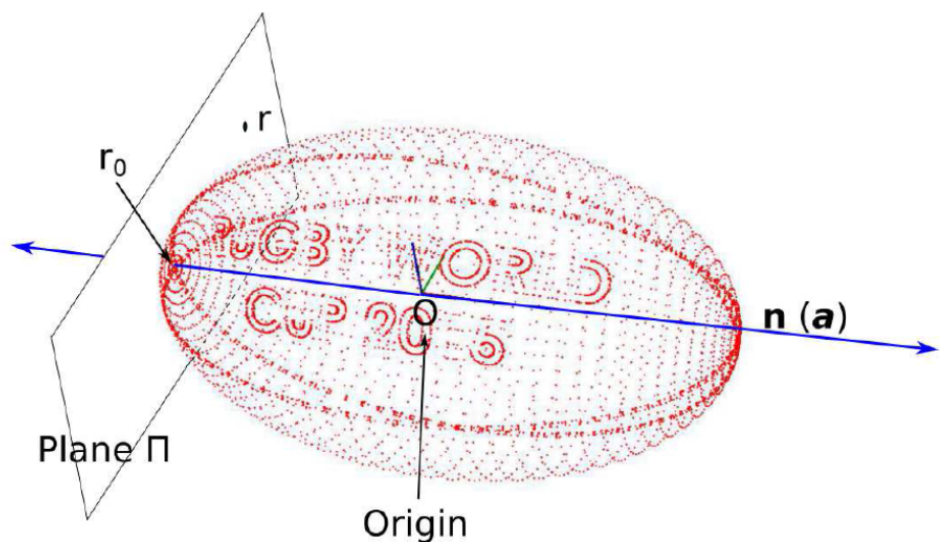
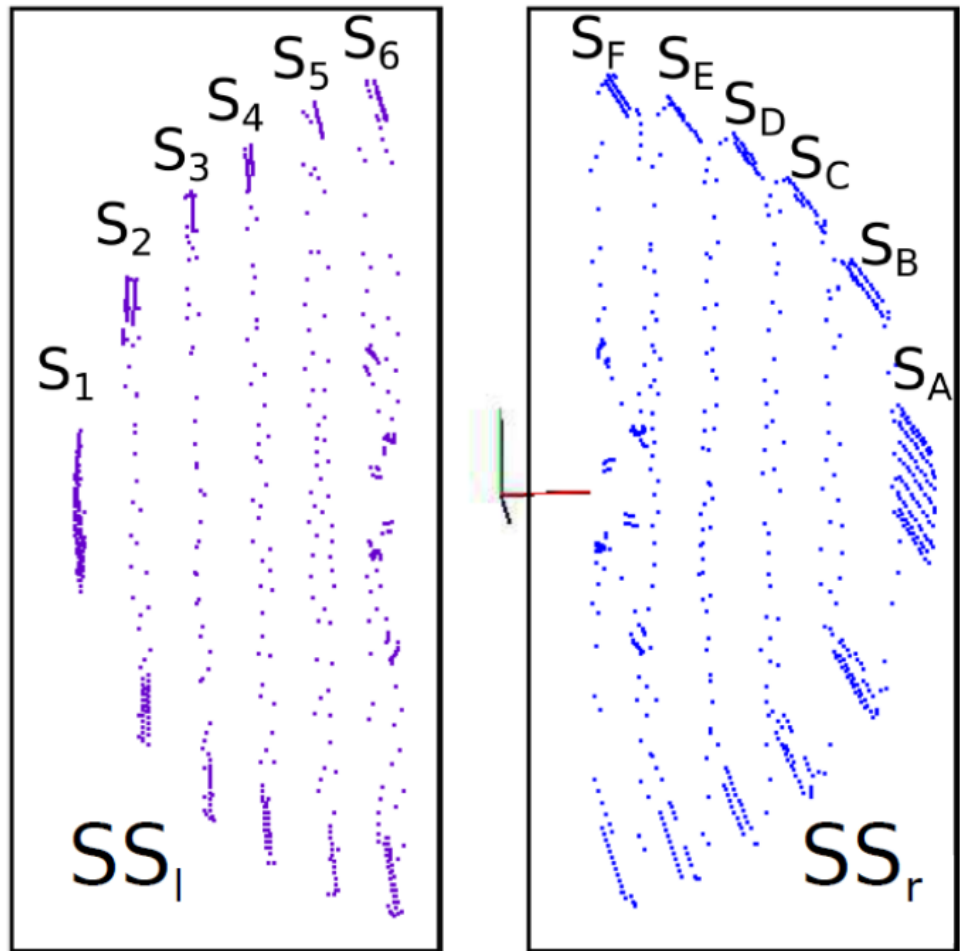


Figura 10 Conjunto de rebanadas resultantes del objeto mostrado en la Figura 9. Cada rectángulo incluye las rebanadas de un extremo del objeto



Una vez definidas las rebanadas, se selecciona iterativamente una tripleta de puntos de una rebanada de cada extremo del objeto. Las rebanadas se van explorando desde el extremo hacia el centro del objeto avanzando secuencialmente de un lado y del otro, asegurando de este modo, dentro de lo posible, una cierta coherencia en las posiciones de las manos al sujetar el objeto. A fin de eliminar tripletas de puntos claramente no alcanzables por los dedos, se usan los siguientes tres filtros:

- El área del triángulo formado por los tres puntos de la tripleta debe ser menor que el área del triángulo formado por los puntos centrales de las yemas de los tres dedos usados de cada mano cuando estos están extendidos;
- La distancia entre los puntos debe ser menor que la máxima distancia posible entre las yemas de los dedos de la mano usada;
- Un índice Q que indica que tan cerca está el triángulo definido por los tres puntos de ser equilátero debe estar por encima de un umbral dado. Q se calcula como

$$Q = \frac{(b+c-a)(c+a-b)(a+b-c)}{abc} \quad [6]$$

siendo a, b y c las longitudes de los lados del triángulo. Q es igual a 1 para triángulos equiláteros e igual a 0 para triángulos degenerados de área nula.

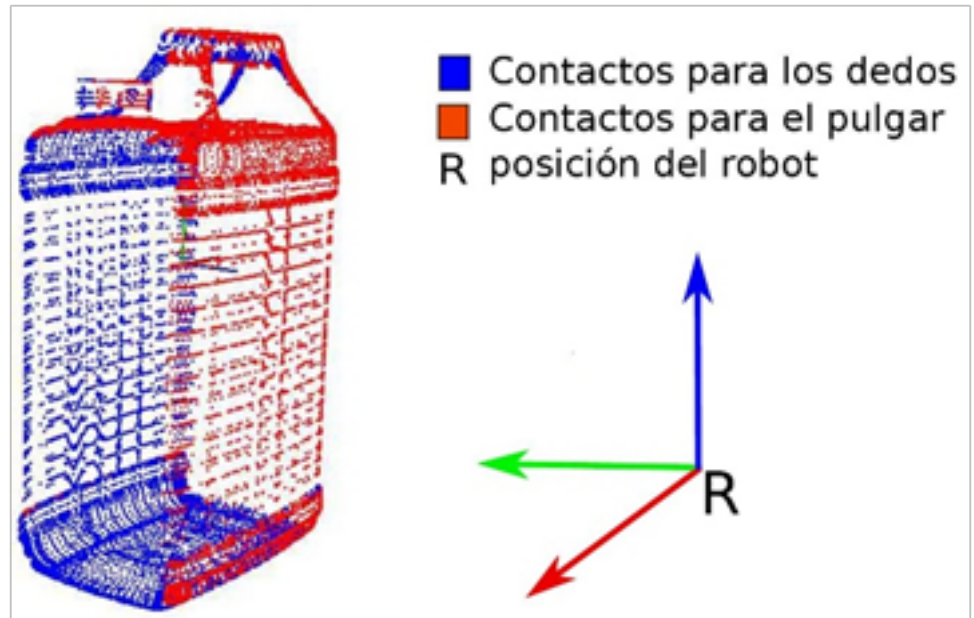
Para cada par de tripletas de puntos se verifica si haciendo contacto en cada uno de los seis puntos se puede obtener una presión que satisfaga la condición de equilibrio de fuerzas (force-closure, FC), es decir, que pueda resistir perturbaciones de fuerzas externas en cualquier dirección, y si tiene una calidad superior a un umbral deseado prefijado. La calidad de la presión está dada por el módulo de la máxima perturbación (fuerza y/o par) que la presión puede resistir en cualquier dirección (Ferrari y Canny, 1992).

Una vez encontrado un par de tripletas que permita una presión FC con la calidad deseada, se realiza un análisis de alcanzabilidad para ver si los puntos seleccionados son alcanzables por el sistema bi-brazo. La complejidad de este paso es elevada ya que se debería probar la alcanzabilidad para cada combinación posible de asignación de los seis puntos de contacto a cada uno de los seis dedos que se van a usar, y en cada caso se ha de calcular la cinemática inversa del sistema robótico completo, es decir, los dos brazos con sus respectivas manos. Para reducir la complejidad se usa una estrategia que simplifica el proceso de asignación de los

puntos a los dedos. Los puntos se agrupan en dos conjuntos dependiendo de la dirección de la normal (exterior) a la superficie del objeto: cuando la normal apunta en dirección a la posición del robot los puntos son candidatos a ser contactados por el pulgar, y, en caso contrario por el resto de

los dedos (ver Figura 11). Cuando el conjunto de puntos es cinemáticamente alcanzable por los dedos correspondientes de cada mano se procede a planificar los movimientos de los brazos y manos para llegar a ellos usando planificadores de movimientos basados en muestreo (García et al., 2017).

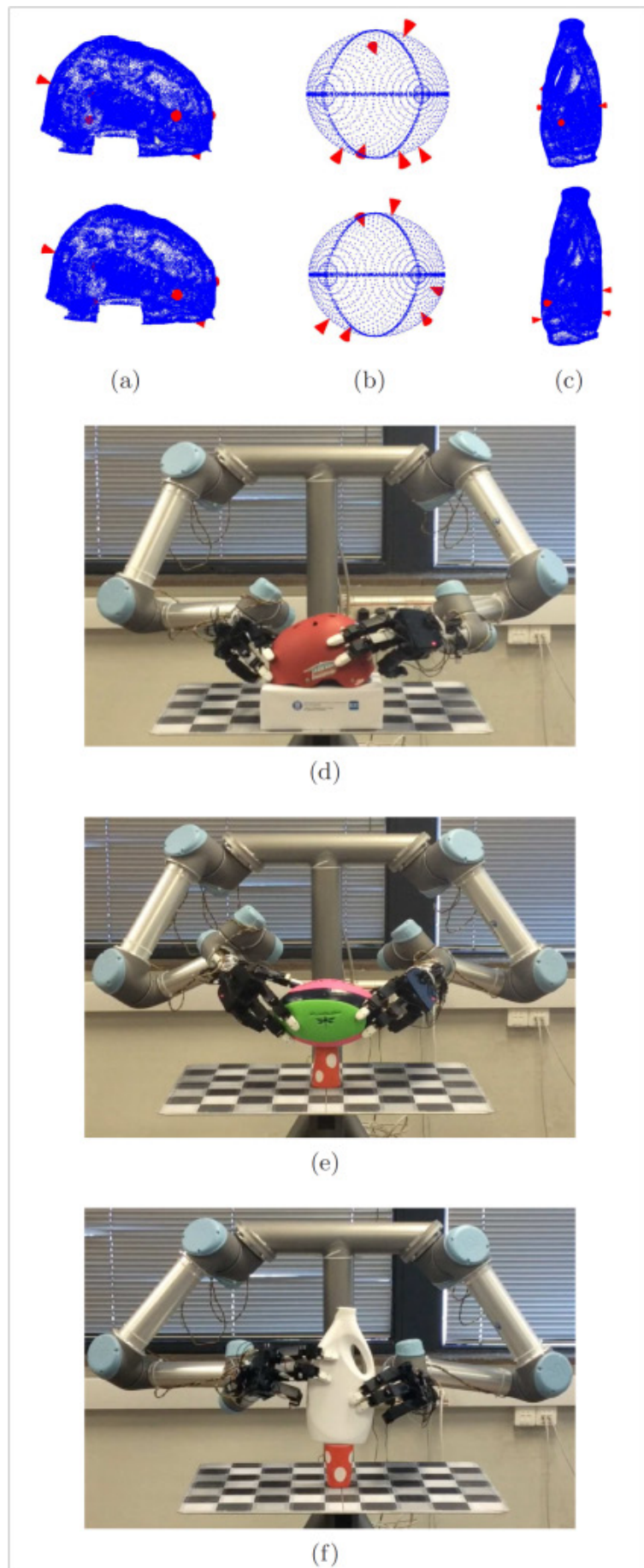
Figura 11 Puntos candidatos a ser contactados en la prensión, en rojo los puntos asignados al pulgar, en azul los asignados al resto de los dedos.



El planificador de prensiones bimanuales ha sido aplicado a distintos tipos de objetos. En la Figura 12 se muestran tres ejemplos de prensiones válidas de un casco, de un balón de rugby y de una botella de detergente, mostrándose los puntos

de contacto obtenidos sobre el modelo del objeto, así como la ejecución de la prensión con un sistema robótico bi-brazo real.

Figura 12 Ejemplos de prensiones bimanuales: a)-c) Ejemplos de puntos de contacto alcanzables por el sistema bi-braza sobre los modelos de los objetos; d)-f) Prensiones con el sistema robótico real.



La continuación de este enfoque está orientada a optimizar las fuerzas que deben aplicarse en las articulaciones de los dedos para optimizar a su vez las fuerzas resultantes sobre el objeto siguiendo diferentes criterios, como, por ejemplo, hacer que los dedos hagan las fuerzas mínimas necesarias para la sujeción, hacer que el objeto esté sometido a las menores fuerzas internas posibles, o intentar evitar equilibrar las fuerzas que hace cada mano al sujetar el objeto (Rojas-de Silva y Suárez, 2017).

5.2. Cooperación bimanual en la manipulación independiente de objetos

Otra forma de cooperación bimanual es la manipulación coordinada en la que ambos brazos trabajan en la misma tarea pero sin interacción física entre ellos. Dentro de este tipo de colaboración se han realizado desarrollos con el objetivo de sujetar dos objetos determinados, uno con cada mano, considerando que el acceso a estos objetos puede ser obstaculizado por otros objetos que deberán ser oportunamente apartados usando ambas manos (Rodríguez et al., 2014; Rodríguez y Suárez, 2016). Los objetos deseados pueden sujetarse con cualquiera de las dos manos o pueden ser pre-asignados a cada una de ellas, y lo mismo sucede con los obstáculos, que pueden ser apartados con cualquiera de las dos manos siguiendo distintas estrategias de optimización.

En estas aplicaciones se han utilizado variantes de planificadores de movimientos basados en muestreo, como PRM o RRT (García et al., 2017). En estas variantes, cuando se calculan las trayectorias de los brazos hasta cada objeto no se descartan las configuraciones en las que se producen colisiones con obstáculos que puedan ser quitados, sino que se acepta el camino, pero marcando que solo es válido si se quitan tales obstáculos. De este modo, el planificador de movimientos se utiliza de forma recursiva, comenzando por buscar caminos hasta los objetos de interés, y siguiendo con la búsqueda de caminos para quitar los obstáculos encontrados, y así sucesivamente. De esta forma se genera un grafo de precedencias G , con dos nodos raíces que representan los objetos de interés y otros que, agregados recursivamente, representan los obstáculos se han de quitar para poder acceder (y mover) cada objeto con cada brazo. El grafo G es del tipo AND/OR (Stoffel et al., 1995), cada nodo representa un

objeto O_j ($j = 1, 2, \dots, n$) y cada arco que sale de él lo liga a otro nodo que representa un obstáculo existente en el camino P_{ij} encontrado para que el brazo R_i ($i = 1, 2$) acceda a O_j . La Figura 13 muestra un ejemplo de un grafo G que se detalla más adelante.

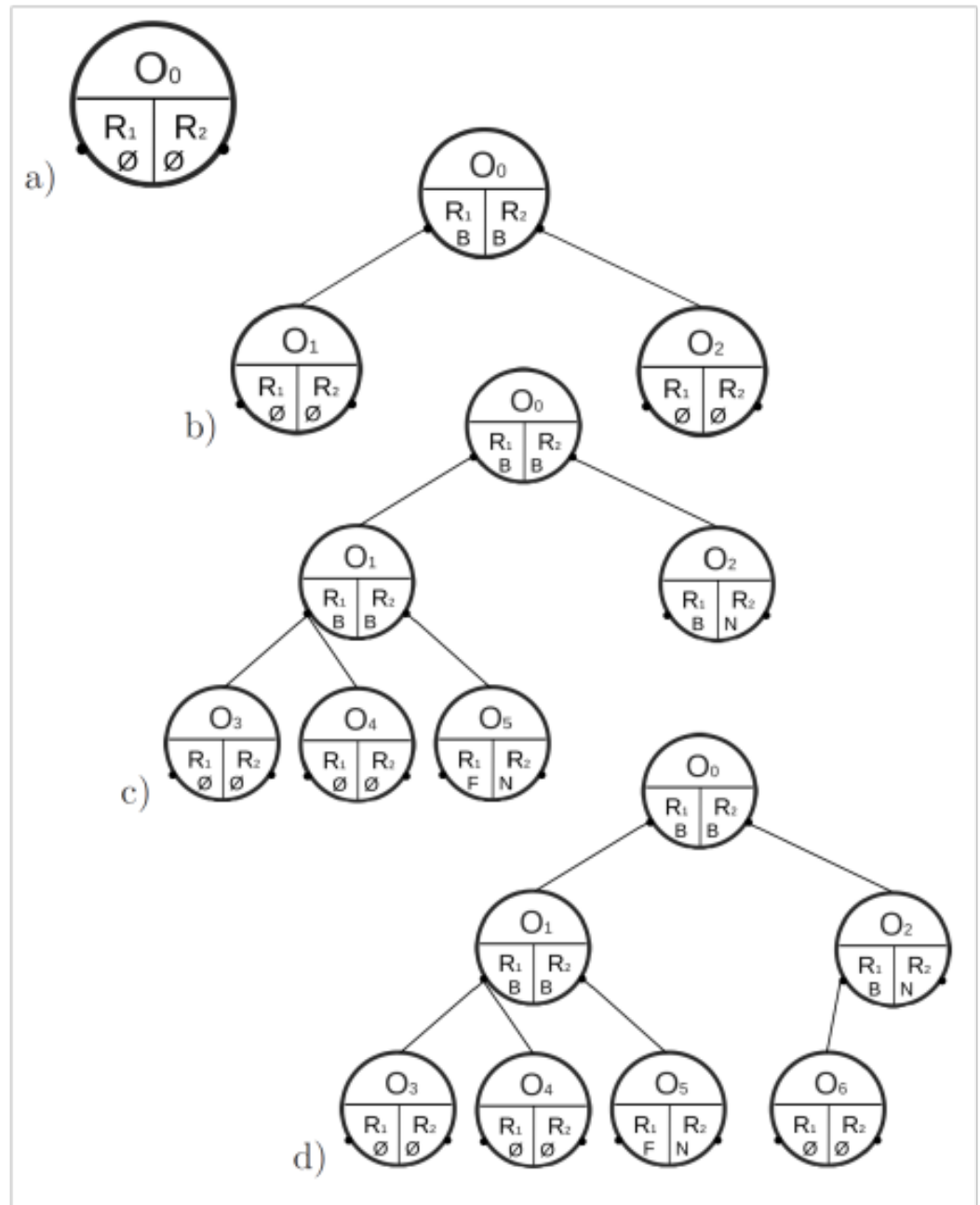
El conjunto O_{ij} agrupa a todos los obstáculos existentes en P_{ij} .

Cada nodo O_j tiene las siguientes propiedades:

- Un identificador (id) que indica el objeto representado O_j .
- Un identificador (root id) que indica cuál es el nodo padre de O_j .
- Un coste (root cost) para cada nodo raíz del grafo que representa el número de nodos (objetos) que, desde O_j , deben quitarse para alcanzar el correspondiente nodo raíz.
- Un coste local (local cost) que indica el número objetos que con un brazo dado R_i se han de quitar para acceder a O_j .
- El estado (state) de O_j , que indica para cada brazo R_i :
 - Empty (\emptyset), cuando no se ha calculado aun un camino P_{ij} para alcanzar O_j con R_i .
 - Null (N), cuando R_i no puede alcanzar O_j (i.e. O_j está fuera del alcance cinemático de R_i).
 - Free (F), cuando existe un camino P_{ij} para que R_i alcance O_j sin obstáculos ($SO_{ij} = \emptyset$).
 - Blocked (B), cuando ya se tiene un camino P_{ij} para que R_i alcance O_j pero con obstáculos ($SO_{ij} \neq \emptyset$).

La Figura 13 muestra un ejemplo de un grafo G para un único objeto de interés (i.e. con un único nodo raíz). Del análisis del grafo G se obtiene una secuencia de acciones para cada brazo que permite acceder a los objetos de interés siguiendo distintos criterios, como, por ejemplo, quitar el menor número posible de obstáculos para acceder a los objetos de interés.

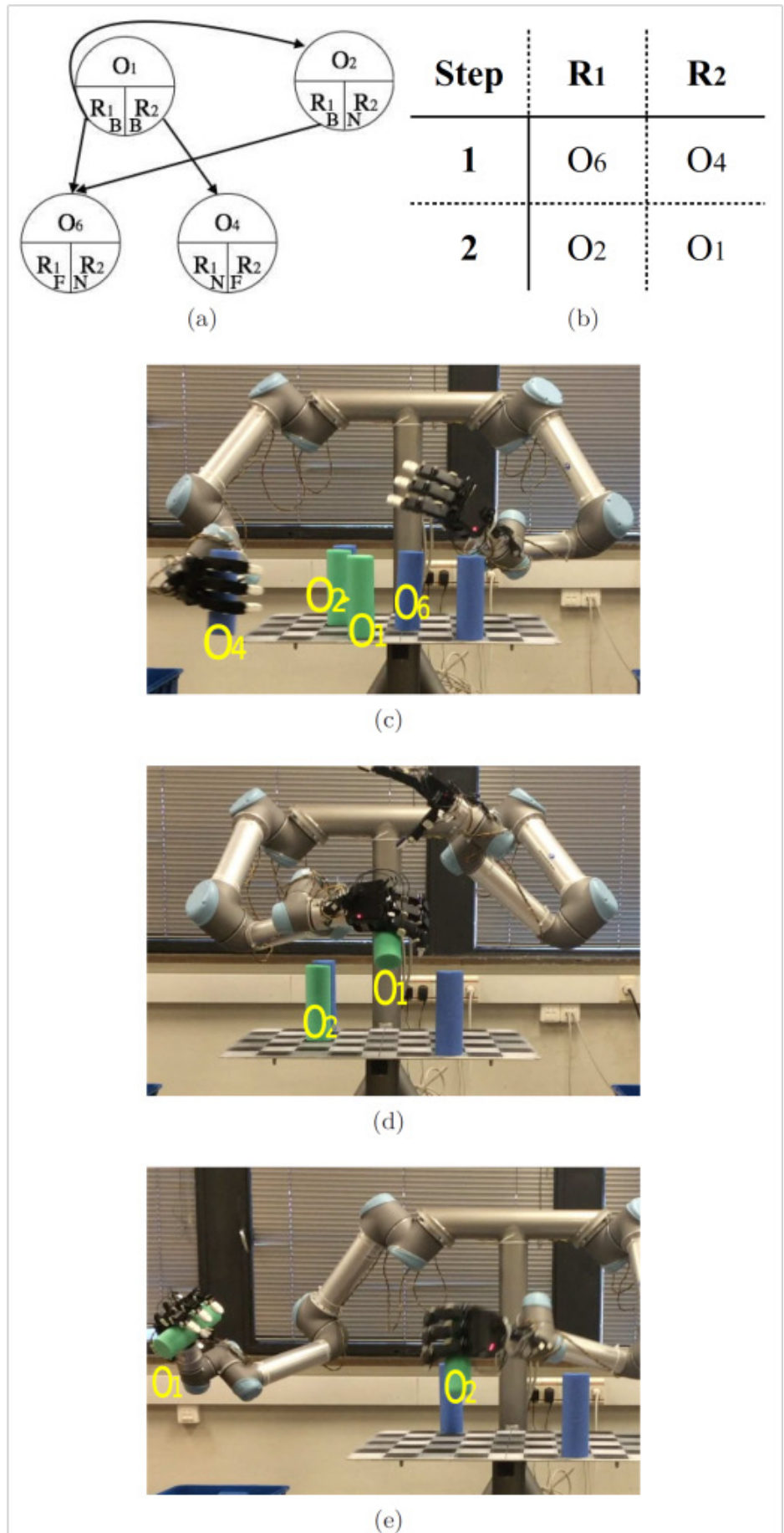
Figura 13 Ejemplo de grafo de precedencias G: (a) nodo raíz representando el objeto de interés O_0 ; (b) usar R_1 para manipular O_0 produce $SO_{1,0}=\{O_1\}$ y usar R_2 produce $SO_{2,0}=\{O_2\}$, luego O_1 y O_2 se agregan como nodos hijos de O_0 con arcos para R_1 y R_2 actuando como arcos OR; (c) quitar O_1 con R_1 produce $SO_{1,1}=\{O_3, O_4\}$ luego O_3 y O_4 se agregan como nodos hijos de O_1 con dos arcos AND para R_1 , mientras que quitar O_1 con R_2 produce $SO_{1,1}=\{O_5\}$ por lo que O_5 se agrega como hijo de O_1 , actuando en este caso como arco OR para R_2 ; (d) en la otra rama, quitar O_2 con R_1 produce $SO_{1,2}=\{O_6\}$ luego O_6 se agrega como hijo de O_2 con un arco OR para R_1 , y cuando se intenta quitar O_2 con R_2 no se encuentra un camino válido (por ejemplo debido a que no hay solución cinemática para que el brazo R_2 sujete O_2) y por lo tanto no se generan nuevos nodos hijos de O_2 para R_2 . Este procedimiento se ejecuta secuencialmente hasta que los nodos terminales de G representan objetos que pueden ser manipulados sin apartar ningún obstáculo y por lo tanto se puede buscar una secuencia de acciones para apartar los obstáculos hasta llegar a O_0 .



Los caminos geométricos obtenidos para cada brazo con este procedimiento pueden ejecutarse paralelizando algunas acciones, pero como no se ha realizado ninguna coordinación temporal podrían producirse colisiones entre los brazos. Para evitar estas colisiones, el movimiento de los brazos siguiendo los caminos geométricos obtenidos se coordina en línea (Montaña y Suárez, 2016b), priorizando las tareas de acuerdo con su precedencia en el grafo y ajustando la evolución temporal de los movimientos de los brazos sin cambiar sus rutas geométricas.

La Figura 14 ilustra el procedimiento. Cada mano debe sujetar uno de los dos objetos de interés, los cilindros verdes O_1 y O_2 , mientras que los cilindros azules $O_{i,i \neq 1,2}$ son obstáculos amovibles. La Figura 14a muestra el grafo de precedencias obtenido, de su análisis se obtuvo como resultado que, en una primera ronda de movimientos, el robot R_1 es elegido para quitar el obstáculo O_6 y el robot R_2 para quitar el obstáculo O_4 , para que posteriormente R_1 sujete el objeto deseado O_1 y R_2 el objeto deseado O_2 . Las Figuras 14(b-d) muestran algunas fotos de la ejecución real del experimento.

Figura 14 Ejemplo de cooperación bi-brazo, los objetos de interés son los cilindros verdes O_1 y O_2 : a) Grafo de precedencias; b) secuencia de acciones, el brazo R_1 quita O_6 y el R_2 quita O_2 en paralelo, luego R_1 sujeta O_1 y R_2 sujeta O_4 ; c), d) y e) instantáneas de la ejecución real, con R_1 yendo a sujetar O_6 y R_2 sujetando O_4 , R_2 moviendo O_1 y R_1 esperando para sujetar O_2 , y R_1 moviendo O_2 con R_2 ya en su pose final.



Actualmente se trabaja en nuevos modelos que faciliten la búsqueda de soluciones óptimas de forma sistemática, y considerando también la posibilidad de realizar la transferencia de un objeto de una mano a otra y de colaborar con el operario. En esta línea ya se han propuesto medidas de calidad que indiquen cuan buena es una prensión de cara a la posterior transferencia del objeto (Soler et al., 2018).

6. Capacidad de interacción con operarios humanos

Es necesario que los robots co-trabajadores tengan la capacidad de interacción con los operarios humanos para llevar a cabo tareas colaborativas. Por un lado, el sistema debe ser capaz de planificar tareas en las que el operario humano es uno de los actores en su ejecución y el robot co-trabajador el otro. Por otro, al compartir el mismo espacio de trabajo, es indispensable que no haya colisiones entre el robot y los operarios humanos y que éstos se puedan mover con total confianza y seguridad. Estos retos se detallan en las siguientes subsecciones.

6.1. Planificación contingente de tareas

Respecto al primer reto, se ha incorporado el operario como actor en el planificador de tareas, para realizar ciertas tareas que el robot difícilmente puede realizar, para darle información al robot sobre el estado del entorno, y para realizar acciones de transferencia de objetos al espacio alcanzable del robot. Las acciones realizadas por el operario pueden no tener un resultado único, por lo que se ha usado la variante del planificador Fast Forward, llamada Contingent Fast Forward (Brafman y Hoffmann, 2004), que genera un árbol de planes que contiene todas las posibles variantes en función de los diferentes resultados de las acciones realizadas por el operario.

6.2. Manipulación con movimientos de apariencia humana

Respecto al segundo reto, es necesario que el robot esté dotado de la capacidad de realizar movimientos con apariencia humana, para que el operario pueda predecir intuitivamente sus movimientos y pueda moverse con tranquilidad a su lado. Asimismo, el robot debe poder detectar la presencia y del operario humano y tener definidas estrategias reactivas que eviten colisiones. Esta fase está actualmente en desarrollo, y se prevé el uso de un sistema de cámaras que puedan monitorizar continuamente los movimientos del operario humano, y, acorde a estos, adaptar ligeramente de forma dinámica las trayectorias planificadas o, de ser necesario, replanificarlas.

Para planificar movimientos con apariencia humana, se plantea primero una transformación entre las articulaciones de un operador humano y las de un robot antropomorfo (aun cuando el robot pueda tener una estructura cinemática algo diferente), y se usa para trasladar al robot los movimientos que realiza un humano al ejecutar distintas tareas. A continuación, se estudian las relaciones entre las variaciones de las articulaciones del robot cuando realiza estos movimientos, tanto en lo relativo a posición como a velocidades (denominadas sinergias de orden cero y uno respectivamente). Estas sinergias permiten definir un subespacio donde realizar la planificación de movimientos, obteniéndose movimientos con apariencia humana y, además, de manera eficiente al reducirse la dimensión del espacio de planificación (García et al., 2017, 2018).

7. Conclusiones

La tendencia de la robótica industrial se dirige hacia la robótica colaborativa, en la que los robots comparten el espacio de trabajo con los operadores humanos e interactúan con ellos, tomando un papel de co-trabajadores activos. En este artículo se han presentado diferentes aspectos técnicos que este reto conlleva. Por un lado, aspectos que permiten una convivencia segura en el mismo entorno, como la ejecución de movimientos con apariencia humana. Por otro, aspectos que permiten a los robots ser más autónomos, como las capacidades avanzadas de manipulación, incluyendo manipulación diestra y cooperación bimanual, así como la capacidad de planificación simultánea de movimientos y tareas. Estos desarrollos deben permitir a los robots trabajar en entornos menos estructurados, facilitando su integración en la pequeña y mediana empresa, permitiendo un aumento de la producción y mejorando la calidad de los puestos de trabajo.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el Gobierno Español mediante el proyecto DPI2016-80077-R.

Los autores agradecen la colaboración de Aliakbar Akbari, Mohammed Diab, Néstor García, Andrés Montaña, Muhayyuddin, Carlos Rodríguez, Abiud Rojas-de-Silva y Leopold Palomo.

8. Referencias

Akbari, A., Laggrifoul, F., y Rosell, J. (2018). A sampling-based tree planner for systems with complex dynamics. *Autonomous Robots*.

- Blum, A.L. y Furst, M.L. (1997). Fast planning through planning graph analysis. *Artificial intelligence*, 90(1), 281–300.
- Brafman, R. y Hoffmann, J. (2004). Conformant planning via heuristic forward search: A new approach. In S. Koenig, S. Zilberstein, y J. Koehler (eds.), *Proceedings of the 14th International Conference on Automated Planning and Scheduling (ICAPS-04)*, 355–364. Whistler, Canada.
- Diab, M., Muhayyuddin, Akbari, A., y Rosell, J. (2017). An ontology framework for physics-based manipulation planning. In *ROBOT 2017: Third Iberian Robotics Conference. Advances in Intelligent Systems and Computing*, volume 693, 452–464. Springer, Cham.
- Ferrari, C. y Canny, J. (1992). Planning optimal grasps. In *Proceedings of IEEE International Conference on Robotics and Automation*, 2290–2295. IEEE Comput. Soc. Press.
- García, N., Rosell, J., y Suárez, R. (2017). Aplicación de algoritmos RRT en la planificación de movimientos óptimos en robótica. In *Proc. of the 12th Metaheuristics International Conference (MIC'17) - Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB'17)*, 953 – 962.
- García, N., Rosell, J., y Suárez, R. (2018). Motion planning by demonstration with human-likeness evaluation for dual-arm robots. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 1–10. doi: 10.1109/TSMC.2017.2756856.
- García, N., Suárez, R., y Rosell, J. (2017). Task-dependent synergies for motion planning of an anthropomorphic dual-arm system. *IEEE Transactions on Robotics*, 33(3), 756–764. doi: 10.1109/TRO.2017.2676131.
- Hoffmann, J. y Nebel, B. (2001). The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research*, 253–302.
- Liégeois, A. (1977). Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms. *IEEE Transactions on Systems, Man, and Cybernetics*, 7(12), 868–871.
- McGuinness, D., Van Harmelen, F., et al. (2004). Owl web ontology language overview. *W3C recommendation*, 10(10), 2004.
- Montaño, A. y Suárez, R. (2015). Unknown object manipulation based on tactile information. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 5642–5647.
- Montaño, A. y Suárez, R. (2016a). Commanding the object orientation using dexterous manipulation. In *Advances in Intelligent Systems and Computing*, volume 418, 69–79. Springer Verlag.
- Montaño, A. y Suárez, R. (2016b). Coordination of several robots based on temporal synchronization. *Robotics and Computer-Integrated Manufacturing*, 42, 73–85.
- Montaño, A. y Suárez, R. (2017). Robust dexterous tele-manipulation following object-orientation commands. *Industrial Robot: the international journal of robotics research and application*, Vol. 44, Issue 5, 2017, pp. 648–657 (ISSN: 0143-991X)..
- Montaño, A. y Suárez, R. (2018). Manipulation of unknown objects to improve the grasp quality using tactile information. *Sensors*, 18(5). doi:10.3390/s18051412.
- Muhayyuddin, Moll, M., Kavradi, L., y Rosell, J. (2018). Randomized physics-based motion planning for grasping in cluttered and uncertain environments. *IEEE Robotics and Automation Letters*, 3(2), 712–719. doi: 10.1109/LRA.2017.2783445.
- Olszewska, J.I., Barreto, M., Bermejo-Alonso, J., Carbonera, J., Chibani, A., Fiorini, S., Goncalves, P., Habib, M., Khamis, A., Olivares, A., de Freitas, E.P., Prestes, E., Ragavan, S.V., Redfield, S., Sanz, R., Spencer, B., y Li, H. (2017). Ontology for autonomous robotics. In *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*.
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., y Ng, A.Y. (2009). ROS: an open-source robot operating system. In *ICRA Workshop on Open Source Software*, volume 3, 5.
- Roa, M.A., Suárez, R., y Cornellà, J. (2008). Medidas de calidad para la prensión de objetos. *Revista iberoamericana de Automática e Informática Industrial*, 5(1), 66–82.
- Roa, M.A. y Suárez, R. (2014). Grasp quality measures: review and performance. *Autonomous Robots*, 38(1), 65–88.
- Rodríguez, C., Rojas-de-Silva, A., y Suárez, R. (2016). Dual-arm framework for cooperative applications. In *IEEE 21st Int. Conf. on Emerging Technologies and Factory Automation (ETFA)*.
- Rodríguez, C. y Suárez, R. (2016). Combining motion planning and task planning for a dual-arm system. In *IEEE-RSJ Int. Conf. on Intelligent Robots and Systems (IROS 2016)*, 4238–4243.

- Rodríguez, C., Montaña, A., y Suárez, R. (2014). Planning manipulation movements of a dual-arm system considering obstacle removing. *Robotics and Autonomous Systems*, 12(12), 1816–1826.
- Rojas-de-Silva, A. y Suárez, R. (2016). Grasping bulky objects with two anthropomorphic hands. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 877–884.
- Rojas-de-Silva, A. y Suárez, R. (2017). Contact Force Computation for Bimanual Grasps. In *Proceedings of 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2017)*, 1–6. IEEE Comput. Soc. Press.
- Rosell, J., Pérez, A., Aliakbar, A., Muhayyuddin, Palomo, L., y García, N. (2014). The Kautham Project: A teaching and research tool for robot motion planning. In *IEEE Int. Conf. on Emerging Technologies and Factory Automation*.
- Soler, F., Rojas-de Silva, A., y Suárez, R. (2018). *Grasp Quality Measures for Transferring Objects*, 28–39. Springer International Publishing.
- Stoffel, D., Kunz, W., y Gerber, S. (1995). *AND/OR graphs*. Technical Report - Max-Planck Institut für Informatik, Germany.
- Suárez R., Basañez, L., y Rosell J. (1995). Using Configuration and Force Sensing in Assembly Task Planning and Execution. In *IEEE Int. Symp. on Assembly and Task Planning, ISATP'95*, (IEEE Pres., ISBN 0-8186-6995-0), Pittsburgh, Pennsylvania, EEUU, 1995, pp. 273-279.
- Sucan, I. y Kavraki, L.E. (2012). A sampling-based tree planner for systems with complex dynamics. *IEEE Transactions on Robotics*, 28(1), 116–131.
- Sucan, I., Moll, M., y Kavraki, L.E. (2012). The open motion planning library. *Robotics & Automation Magazine, IEEE*, 19(4), 72–82.