

Teleoperación académica en ambiente educativo del robot RV-M1 con Matlab® en red RENATA

Academic Remote Operation in Educational Environment of the RV-M1 robot with Matlab® with RENATA net

Javier ROLDÁN Mckinley [1](#); Eugenio YIME Rodríguez [2](#); José SOSA Rodríguez [3](#)

Recibido: 09/08/2018 • Arobado: 20/12/2018 • Publicado: 21/01/2019

Contents

- [1. Introducción](#)
 - [2. Metodología](#)
 - [3. Resultados: interfaz gráfica para facilitar la teleoperación](#)
 - [4. Conclusiones](#)
- [Agradecimientos](#)
- [Referencias bibliográficas](#)

RESUMEN:

Se explican los principales comandos de Matlab que permiten la comunicación remota y teleoperación de un robot de amplio uso en la academia, los cuales se integran en la interfaz de teleoperación más sencilla que desempeña la tarea cinemática de pick-and-place. El controlador del robot impone la necesidad de un mínimo de seis funciones básicas del robot para la teleoperación en la tarea propuesta. Se propone una interfaz gráfica de usuario que permita la teleoperación del robot desde cualquier computador con conexión a internet, simplemente escribiendo la dirección IP y el puerto de datos del servidor en el computador del cliente. Mediante esta interfaz se puede aprovechar un poco más la limitada funcionalidad que brinda la arquitectura de la mayoría de los controladores de los robots académicos suministrados por los fabricantes.

Palabras clave: Teleoperación; RV-M1; RENATA

ABSTRACT:

The main Matlab commands that address the teleoperation components of a robot that is being widely used in academia are presented. They are integrated in the simplest graphical interface that performs the typical pick-and-place kinematic task. The robot controller requires that a minimum of six basic robot functions are to be integrated in the GUI in order to perform the task. A GUI that allows the remote operation of the robot from any computer with internet connection is proposed. The GUI allows the client to input the host IP address and data port to establish communication. This GUI allows taking advantage of the limited functionality of the most of the manufacturers' controller box.

Keywords: Remote operation; RV-M1; RENATA

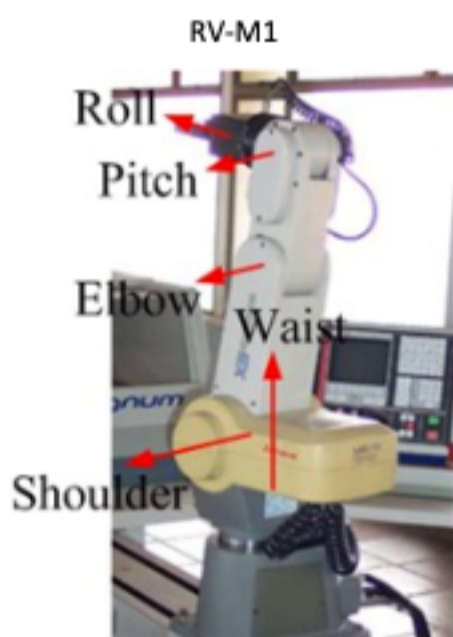
1. Introducción

Un sistema de teleoperación permite la interacción humana con ambientes inaccesibles a la presencia humana, bien sea debido a potenciales riesgos para los humanos o por la lejanía (Gómez, Ollero y García, 2006), extendiendo la habilidad de manipulación del humano a través de control coordinado, usualmente de manipuladores robóticos (Aracil, *et al.*, 2004), siendo el escenario típico el de un cliente (maestro) controlando al esclavo con un mando tipo palanca (joystick) mientras se visualiza la operación del robot a través de video. Las aplicaciones típicas de teleoperación de robots incluyen tareas cinemáticas industriales tales como ensamblado y soldadura (Ming, *et al.*, 1996; Codourey, Rodriguez & Pappas, 1997; Boboc, Moga & Talaba, 2012); la telerobótica espacial que

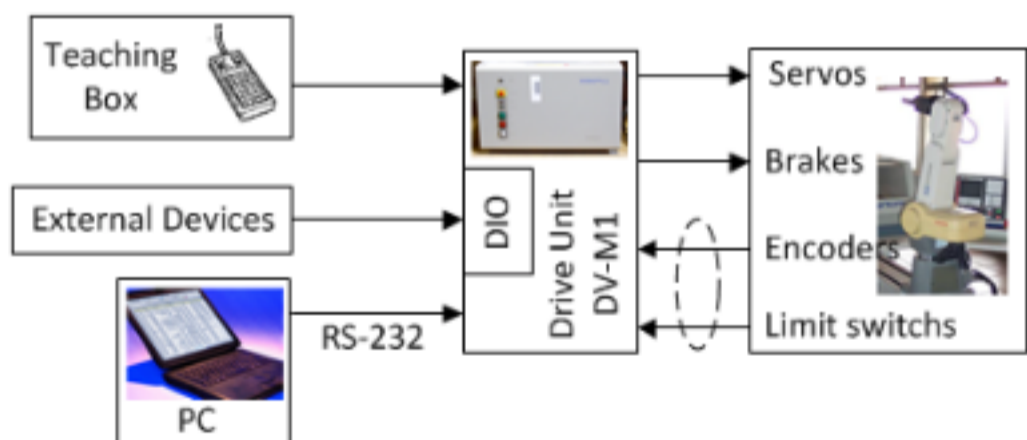
facilita la exploración humana remota en condiciones inhabitables en otros planetas (Volpe & Udomkesmalee, 2017); y la cirugía teleoperada, una de las áreas que más ha originado adelantos en teleoperación (Guthart & Salisbury, 2000; Preusche, Ortamaier & Hirzinger, 2002; Butner & Ghodoussi, 2003; Tavakoli, Patel & Moallem, 2005) utilizada especialmente en procedimientos mínimamente invasivos. Sin lugar a dudas que la teleoperación, en todas sus áreas de aplicación, recibió un gran impulso con la aparición de la World Wide Web o internet (Goldberg, *et al.*, 1995), reduciendo costos en la manera como se hacía la teleoperación hacía 50 años atrás de la fecha de (Goldberg, *et al.*, 1995), y aumentando la cobertura de la tecnología de teleoperación. En cuanto a su uso en manipuladores, los primeros trabajos con internet incluyen aplicaciones no solo de manipuladores industriales (Ahn, Stik & Sup, 2001; Baczynski & Baczynski, 2003; Kofman, *et al.*, 2005; Aschenbrenner, *et al.*, 2015) sino también de robots móviles (Schulz, *et al.*, 2000; Hu, *et al.*, 2001; Andreu, *et al.*, 2003). En la teleoperación de manipuladores en la Academia, se ha venido usando la internet para facilitar el uso de manipuladores robóticos por estudiantes que no tiene acceso a robots reales por medio de la teleoperación (Yang, *et al.*, 2004), conllevando a una gran cantidad de investigaciones que pretenden llevar al estudiante de manera virtual al espacio operacional del robot (Jiménez, *et al.*, 2005; Candelas y Sánchez, 2005; Álvarez, *et al.*, 2006) en las cuales el diseño de las interfaces gráficas de usuario facilita la interacción del usuario con el maestro (Candelas, Jara & Torres, 2006), llegando a remplazar cinemáticamente el joystick, sin descuidar el efecto del retardo en la conexión (Cortés, *et al.*, 2012; Muradore & Fiorini, 2016) para una manipulación segura y cercana al tiempo real (Mostefa, Kaddour-El-Boudadi & Vareille, 2018). Siendo el manipulador serial Mitsubishi Movemaster RVM1 uno de los más usados en la academia a pesar de su limitada funcionalidad y software de operación (Kumar, Kalra & Prakash, 2011), se justifica el desarrollo de plataformas de teleoperación (Dragan, Srinivasa & Lee, 2013) para la interacción entre laboratorios de instituciones educativas que faciliten la enseñanza y la investigación (Caicedo, *et al.*, 2009). Ya se han propuesto plataformas de comunicación para este manipulador (Świder, *et al.*, 2007.a; Świder, *et al.*, 2007.b) las cuales han permitido teleoperación del robot en plataformas comerciales con interfaces orientadas a la programación (Foit, 2008). El robot Mitsubishi Movemaster RV-M1 es un manipulador serial tipo vertical-articulado de cinco grados de libertad, como se muestra en la Figura 1.a. El manipulador imita un brazo antropomórfico por lo cual sus grados de libertad se asemejan a los movimientos del brazo humano. Los cinco motores que accionan sus elementos son tipo servo motores de corriente directa y posee una máxima capacidad de movimiento de carga de 1.2 kg (Mitsubishi Elec. Corp., 1992).

Figura 1

a. Cinco grados de libertad del Movemaster



b. Arquitectura original del robot Movemaster RV-M1



Fuente: elaboración propia

Este artículo presenta una herramienta diseñada para el control de movimiento articular del manipulador Mitsubishi RV-M1. Esta herramienta puede ser utilizada tanto en red comercial como en red académica RENATA y pretende facilitar el uso de este robot para fortalecer la enseñanza en forma remota de la robótica industrial a nivel académico y establecer alianzas de colaboración entre grupos de investigación para el desarrollo de aplicaciones industriales. La interfaz es desarrollada en Matlab, muy compacta e intuitiva que permite la operación remota del robot para operaciones sencillas, utilizando la caja controladora de limitada funcionalidad, Drive Unit DV-M1 mostrada en la Figura 1.b, facilitada por el fabricante la cual únicamente permite movimiento de punto a punto mediante control mono-articular. En cuanto a la disposición de la teleoperación, el controlador y el robot se encuentran en el servidor, como se muestra en la Figura 2, la típica arquitectura tipo batch

con algoritmo de control en el controlador (Al-Mouhamed, Nazeeruddin & Isla, 2010). La red RENATA es la principal "carretera" de información y las redes RUTA y UNIREN conectan el servidor y el cliente a la carretera, respectivamente.

2. Metodología

2.1. Cinemática del manipulador Movemaster RV-M1

2.1.1. Parámetros Cinemáticos

Para que un manipulador realice una tarea específica se debe establecer la posición y orientación de la herramienta y el posicionamiento relativo de cada eslabón. La metodología a utilizar para establecer los parámetros de referencia entre elementos del robot, es la teoría unificada de (Duffy & Rooney, 1975), la cual ha sido implementada con éxito en la cinemática de manipuladores seriales complejos, como el robot de seis eslabones (Duffy & Crane, 1980), e inclusive en aplicaciones espaciales (Crane, Duffy & Carnahan, 1991). Los elementos tridimensionales se definen usando seis parámetros cinemáticos (Crane & Duffy, 2007) y mostrados en la Figura 3.a son explicados a continuación.

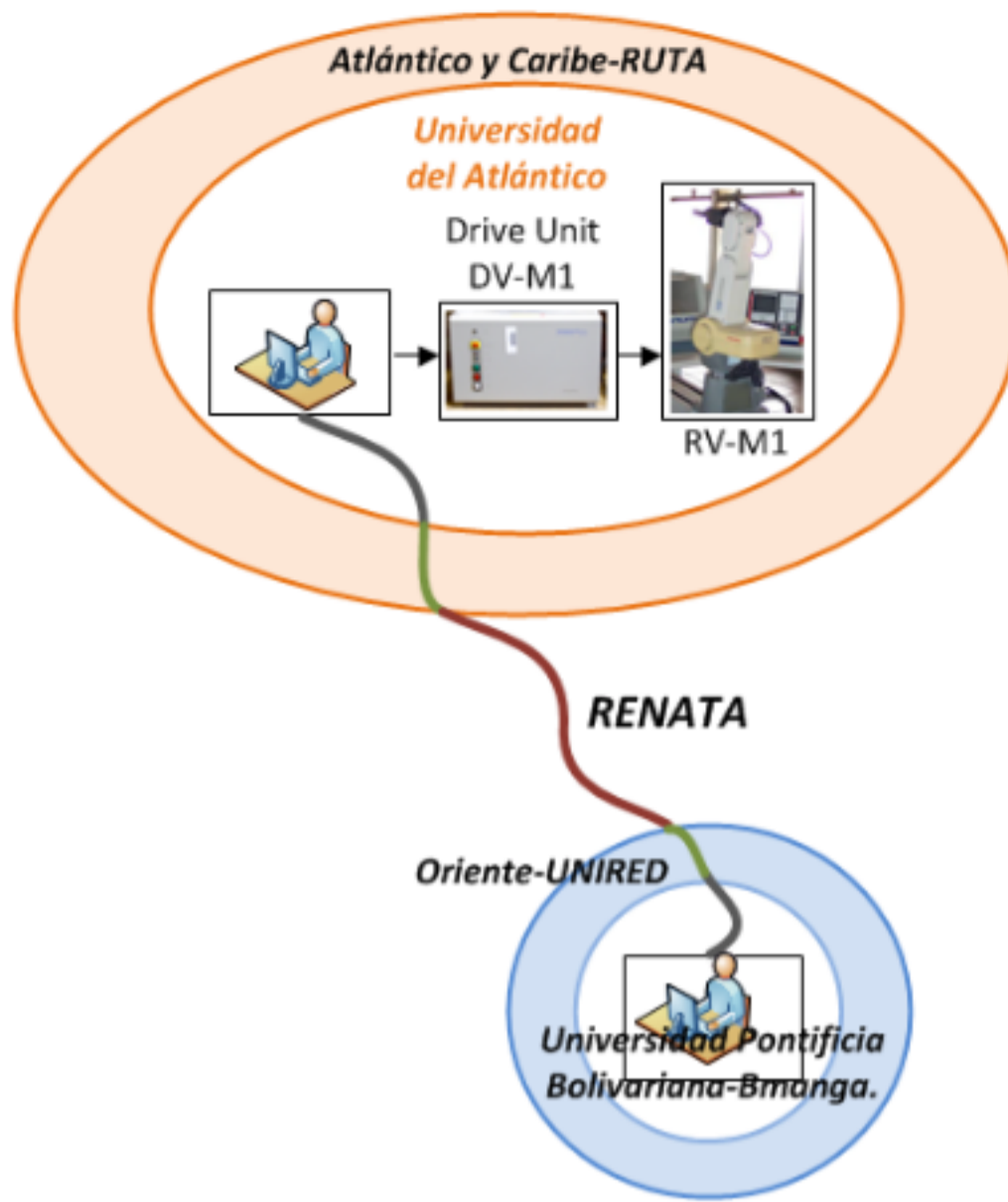
Vector de articulación S_i : para juntas de rotación es un vector unitario que define la rotación siguiendo la regla de la mano derecha. *Vector del elemento a_{ij}* : vector unitario perpendicular a ambos vectores S_i y S_j , sentido i-j. *Distancia de articulación S_i* : distancia entre los vectores a_{hi} y a_{ij} , medida a lo largo de S_i . *Distancia del elemento a_{ij}* : distancia entre S_i y S_j , medida a lo largo de a_{ij} . *Ángulo de articulación θ_i* : ángulo medido con la regla de la mano derecha utilizando como base el vector de articulación S_i , y barrido de a_{hi} a a_{ij} . *Ángulo del elemento α_{ij}* : ángulo medido mediante la regla de la mano derecha utilizando como base en vector del elemento a_{ij} , y barrido de S_i a S_j . En la Tabla 1 se muestran los valores de los parámetros cinemáticos tomados del manual del fabricante (Mitsubishi Elec. Corp., 1992), ilustrados en la Figura 3.b.

La posición y la orientación relativas entre dos elementos se expresan mediante una matriz de transformación (Jazar, 2007). La nomenclatura empleada genera una matriz de transformación recurrente que relaciona dos elementos consecutivos i e j, está dada por (1) (Crane & Duffy, 2007), donde $c_i = \cos(\theta_i)$, $s_i = \sin(\theta_i)$, $c_{ij} = \cos(\alpha_{ij})$, $s_{ij} = \sin(\alpha_{ij})$.

La matriz presentada en (1) permite plantear matricialmente la cinemática del manipulador para relacionar la posición, ${}^F P_H$, y orientación de la herramienta respecto al marco de referencia, Figura 3.c, por medio del producto matricial de (2), donde ${}^F T_1$ es la matriz de transformación del primer elemento en el marco de referencia, ${}^i T_j$ es la matriz de transformación del elemento j en el elemento i, y ${}^S T_T$ es la matriz de transformación de la herramienta en el último elemento del robot.

Figura 2

Esquema de comunicación en la teleoperación



$${}^i_jT = \begin{bmatrix} c_j & -s_j & 0 & a_{ij} \\ s_j c_{ij} & c_j c_{ij} & -s_{ij} & -s_{ij} s_j \\ s_j s_{ij} & c_j s_{ij} & c_{ij} & c_{ij} s_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$${}^F_T = {}^F_1T_1 {}^1_2T_2 {}^2_3T_3 {}^3_4T_4 {}^4_5T_5 {}^5_TT \quad (2)$$

La aplicación recurrente de (1) para hallar las matrices F_1T_1 , 1_2T_2 , 2_3T_3 , 3_4T_4 , 4_5T_5 y 5_TT , al ser reemplazadas en (2) generan la matriz de transformación del robot Mitsubishi RV-M1 en el marco fijo, dada en (3),

$${}^F_T = \begin{bmatrix} c_\phi c_{2+3+4} c_5 + s_\phi s_5 & -c_\phi c_{2+3+4} s_5 + s_\phi c_5 & c_\phi s_{2+3+4} & \delta_1 \\ s_\phi c_{2+3+4} c_5 - c_\phi s_5 & -s_\phi c_{2+3+4} s_5 - c_\phi c_5 & s_\phi s_{2+3+4} & \delta_2 \\ s_{2+3+4} c_5 & -s_{2+3+4} s_5 & -c_{2+3+4} & \delta_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

donde $c_\phi = \cos(\phi_1)$, $s_\phi = \sin(\phi_1)$, $c_{2+3+...} = \cos(\theta_2 + \theta_3 + \dots)$, $s_{2+3+...} = \sin(\theta_2 + \theta_3 + \dots)$, y

$$\delta_1 = c_\phi s_{2+3+4} p_z + c_\phi s_{2+3+4} s_5 + c_\phi c_{2+3} a_{34} + c_\phi c_2 a_{23} \quad (4)$$

$$\delta_2 = -s_\phi s_{2+3+4} p_z + s_\phi s_{2+3+4} s_5 + s_\phi c_{2+3} a_{34} + s_\phi c_2 a_{23} \quad (5)$$

$$\delta_3 = -c_{2+3+4} p_z - c_{2+3+4} s_5 + s_{2+3} a_{34} + s_2 a_{23}. \quad (6)$$

Tabla 1

Valores de los parámetros cinemáticos (Mitsubishi Elec. Corp., 1992)

Distancia de la articulación [mm]	Distancia del elemento [mm]	Ángulo del elemento [grados]	Ángulo de la articulación [grados]
--	$a_{12} = 0$	$\alpha_{12} = 90$	$\phi_1 = *$
$S_2 = 0$	$a_{23} = 250$	$\alpha_{23} = 0$	$\theta_2 = *$
$S_3 = 0$	$a_{34} = 160$	$\alpha_{34} = 0$	$\theta_3 = *$
$S_4 = 0$	$a_{45} = 0$	$\alpha_{45} = 90$	$\theta_4 = *$
$S_5 = 72$	$a_{56} = 0$	--	$\theta_5 = *$

*: Variable, a ser hallado en la cinemática inversa

2.1.2. Cinemática Directa

La cinemática directa, es decir, hallar la posición y orientación de la herramienta a partir de los ángulos de junta conocidos y las distancias de junta y ángulos, se resuelve de la siguiente manera.

El vector posición de la herramienta, \mathbf{P}_F , es extraído de los elementos (1,4), (2,4) y (3,4) de (3) para las componentes x, y y z, respectivamente, expresado mediante:

$$\mathbf{P}_F = \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix}. \quad (7)$$

La orientación de la herramienta es obtenida a partir de los nueve elementos (1,1) a (3,3) de (3), para expresar los ángulos del navegante conocidos como *Roll-Pitch-Yaw* (Guiñada-Cabeceo-Alabeo) (Jazar, 2007) a partir de (8) para el ángulo de guiñada α , (9) para el ángulo de cabeceo β , y (10) para el ángulo de alabeo, γ , los cuales son los ángulos que la herramienta ha rotado con respecto a los ejes del marco fijo x, y y z, respectivamente.

$$\alpha = \text{atan} \left(\frac{r_{32}}{r_{33}} \right) \quad (8)$$

$$\beta = -\text{asin}(r_{31}) \quad (9)$$

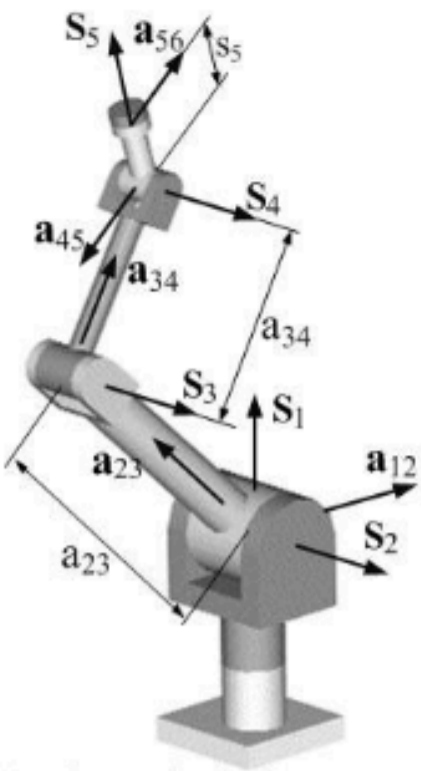
$$\gamma = \text{atan} \left(\frac{r_{21}}{r_{11}} \right) \quad (10)$$

2.1.3. Cinemática Inversa

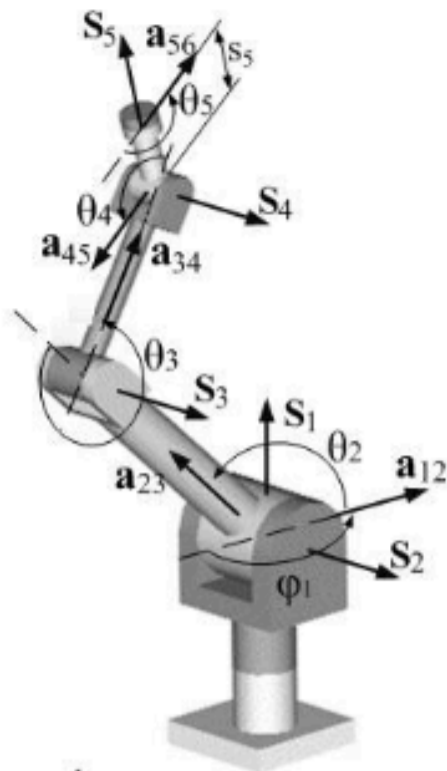
En la cinemática inversa se deben hallar los ángulos de posición de los elementos móviles del robot, a partir de los valores conocidos de la posición y orientación de la herramienta en el marco fijo. Es decir, se deben hallar los ángulos ϕ_1 , θ_2 , θ_3 , θ_4 y θ_5 , a partir de los elementos de la matriz en el lado derecho de (3). En este trabajo se adoptó del método matricial expuesto por (Roldán, *et al.*, 2012), el cual debe ser consultado para un detallado proceso de solución.

Figura 3

a. Parámetros cinemáticos unificados y ejes coordenados del robot RV-M1

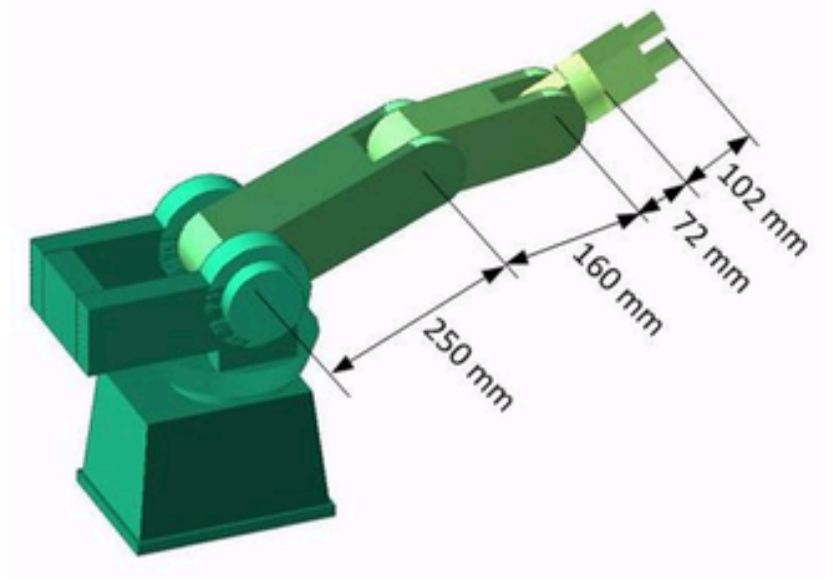


(a) Distancia de junta y de elemento

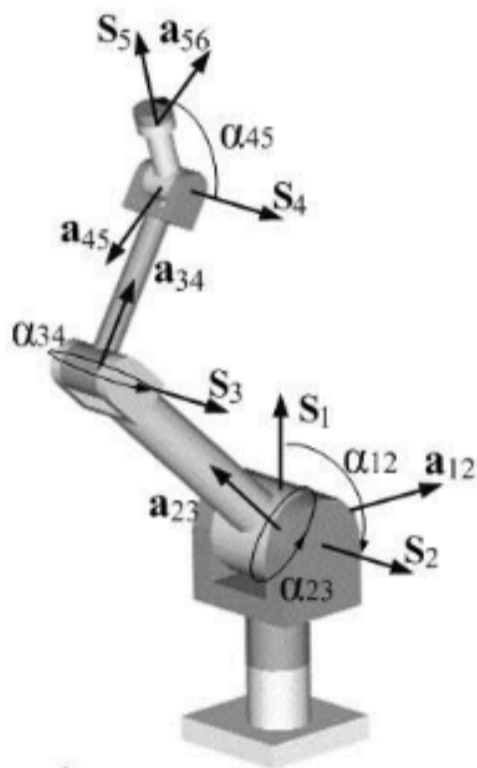


(b) Ángulos de junta

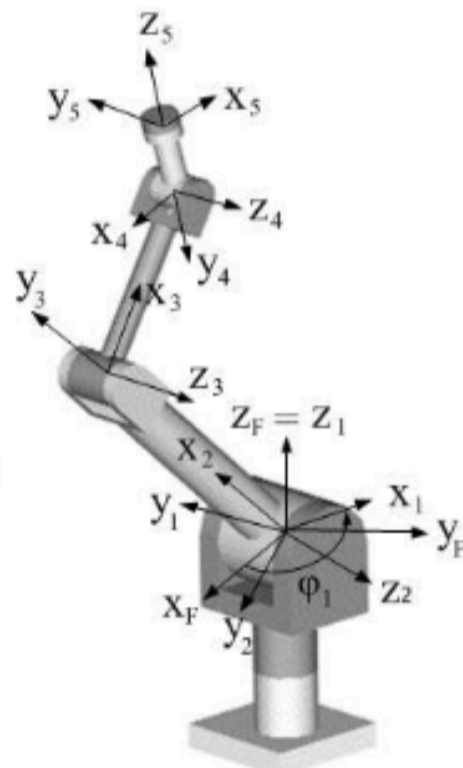
b. Dimensiones entre articulaciones del robot



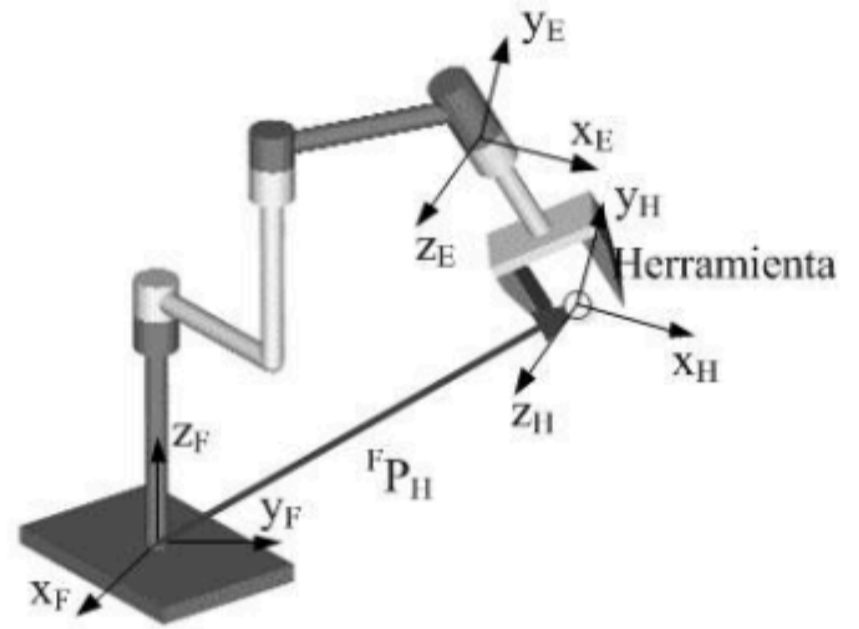
c. Sistemas coordenados fijo {F} y de la herramienta {H} en un manipulador serial



(c) Ángulos de elementos



(d) Ejes coordenados



Mediante el método de sucesiva premultiplicación de (3), en (Roldán, *et al.*, 2012) se demostraron las expresiones para los ángulos φ_1 , θ_2 , θ_3 , θ_4 y θ_5 , presentadas en (13) a (17), respectivamente.

$$\varphi_1 = \text{atan2}(r_{23}, r_{13}) \quad (13)$$

$$\theta_2 = \text{atan2}\left(\frac{Q}{\sqrt{P^2+Q^2}}, \frac{P}{\sqrt{P^2+Q^2}}\right) \pm \cos^{-1}\left(\frac{-R}{\sqrt{P^2+Q^2}}\right), \quad (14)$$

donde: $P = (S_5 + p_z)(r_{13}c_\varphi + r_{23}s_\varphi) - (r_{14}c_\varphi + r_{24}s_\varphi)$, $Q = (S_5 + p_z)r_{33} - r_{34}$,

$R = a_{23} + a_{34}c_3$.

$$\theta_3 = \text{atan2}\left(\frac{r_{34} + (S_5 + p_z)c_{2+3+4} - a_{23}s_2 - a_{34}s_2c_3}{a_{34}c_2}, \frac{A^2 + B^2 - a_{23}^2 - a_{34}^2}{2a_{23}a_{34}}\right) \quad (15)$$

$$\theta_4 = \text{atan2}(s_4, c_4), \quad (16)$$

donde: $S_4 = \frac{c_{2+3}(r_{14}c_\varphi + r_{24}s_\varphi) + r_{34}s_{2+3} - a_{23}c_3 - a_{34}}{S_5 + p_z}$, $C_4 = \frac{s_{2+3}(r_{14}c_\varphi + r_{24}s_\varphi) - r_{34}c_{2+3} - a_{23}s_3}{S_5 + p_z}$.

$$\theta_5 = \text{atan2}(r_{11}s_\varphi - r_{21}c_\varphi, r_{12}s_\varphi - r_{22}c_\varphi). \quad (17)$$

2.2. Comunicación con el robot y entre computadores

La teleoperación propuesta se concibe intuitivamente de la manera más sencilla posible, tratando de incluir únicamente las funciones básicas de cualquier manipulador, tales como: Recoger un objeto (cerrar la mano), mover el robot, y soltar el objeto (abrir la mano); es decir, la típica operación pick-and-place. Sin embargo, debido al protocolo de programación interna del controlador del robot es necesario incluir un mínimo de tres funciones adicionales: Localizar los ceros de las juntas, localizar el cero cinemático, y borrar los valores actuales del robot en un momento determinado cuando se incurre en un error de comunicación con el computador. Se debe entonces consultar en el catálogo del manipulador (Mitsubishi Elec. Corp., 1992) la sintaxis de los comandos que permiten la ejecución de las funciones necesarias, las cuales son resumidas en la Tabla 2. El comando MJ (Move Joint) requiere los valores de los ángulos de junta del manipulador, cinco en total, mostrados en la Figura 1, y cuyos valores límites son consultados en (Mitsubishi Elec. Corp., 1992) y (Goyal, & Sethi, 2010) para así evitar errores en la programación. Los valores límite son presentados en la Tabla 3.

Tabla 2
Comandos para operaciones básicas (Mitsubishi Elec. Corp., 1992)

Función	Comando	Sintaxis
Mover robot mediante las articulaciones	MJ	MJ #Waist, #Shoulder, #Elbow, #Pitch, #Roll
Cerrar la mano	GC	GC
Abrir la mano	GO	GO
Ir a la posición de origen mecánico	NT	NT
Ir al cero cinemático de juntas	OG	OG
Borrar valores	RS	RS

Tabla 3
Valores límite de los ángulos de junta (Mitsubishi Elec. Corp., 1992; Goyal & Sethi, 2010)

Junta	Ángulo	Valor mínimo [grados]	Valor máximo [grados]
1	<i>Waist</i> (Cintura)	-150	150
2	<i>Shoulder</i> (Hombro)	-30	100
3	<i>Elbow</i> (Codo)	-110	0
4	<i>Pitch</i> (Cabeceo)	-90	90
5	<i>Roll</i> (Alabeo)	-180	180

Antes de crear la estructura de las líneas de comando en Matlab que deben ser enviadas al manipulador, se debe establecer conexión con el robot. Se distinguen dos programas que deben ser creados y que deben ser ejecutados simultáneamente para la teleoperación, ellos son el programa del Cliente y el programa del Servidor. El servidor (o host) debe crear un puerto serial para comunicarse con el robot, típicamente conexión RS-232 o adaptador RS-232/USB, con la estructura "Str-Cat 1" en la Tabla 4, y crear un objeto tipo "jtcp" usando el Protocolo de Control de Transmisión (Transmission Control Protocol) con la concatenación "Str-Cat 2"; es posible utilizar software de Fuente Libre (Open Source) disponible en la Central de Intercambio de Archivos de Matlab (Matlab

Central File Exchange) (Bartlett, 2009), siempre y cuando se reconozcan los créditos del autor. A continuación el servidor deberá presionar Enter para aceptar la solicitud de conexión hecha desde el cliente con la estructura "Str-Cat 3", utilizando la dirección IP (Host_IP) y el puerto (Host_Port) que el servidor haya destinado para la comunicación. Establecido la comunicación, se proceden a crear las concatenaciones de caracteres para cada una de las funciones que se necesitan. Para la función que moverá las juntas del robot se tiene la concatenación "Str-Cat 4", la cual se envía al servidor mediante la línea "Str-Cat 5".

Tabla 4
Concatenación de caracteres para comunicación entre Servidor y Cliente

Str-Cat	Cadena de caracteres
1	s = serial('COM3','BaudRate',9600,'DataBits',8,'StopBits',1)
2	jTcpObj = jtcp('accept',1700,'timeout',20000)
3	jTcpObj = jtcp('request',Host_IP,Host_Port,'timeout',tout)
4	STR = strcat('MJ', num2str(WaistAngle),',',num2str(ShoulderAngle),',',num2str(ElbowAngle),',',num2str(PitchAngle),',',num2str(RollAngle))
5	jtcp('write',jTcpObj,cast(STR,'int8'))
6	hExp = actxcontrol('Shell.Explorer.2',[330 200 278 174],hfig)
7	Ip_can = strcat('http://',Host_IP,':1701/')
8	Navigate(hExp,Ip_can)

Un procedimiento similar se sigue para comunicar los comandos de las funciones restantes. La comunicación puede ser a través de entrada por teclado, o diseñando una interfaz de usuario. La imagen es visualizada mediante el enlace entre Matlab e Internet Explorer con la concatenación "Str-Cat 6", y se crea la concatenación de caracteres con la dirección IP del cliente y el puerto para video TCP, mostrada en "Str-Cat 7", lo cual ejecuta la navegación de la imagen en Internet con la cadena de caracteres en "Str-Cat 8". La Figura 4 muestra el diagrama de flujo para la programación de la rutina del servidor, mientras que la Figura 5 muestra lo mismo para el servidor. Se observa como para el cliente es un lazo que solo finaliza cuando se desea cerrar la interfaz, a diferencia del cliente cuyo programa finaliza si el tiempo de espera se alcanza.

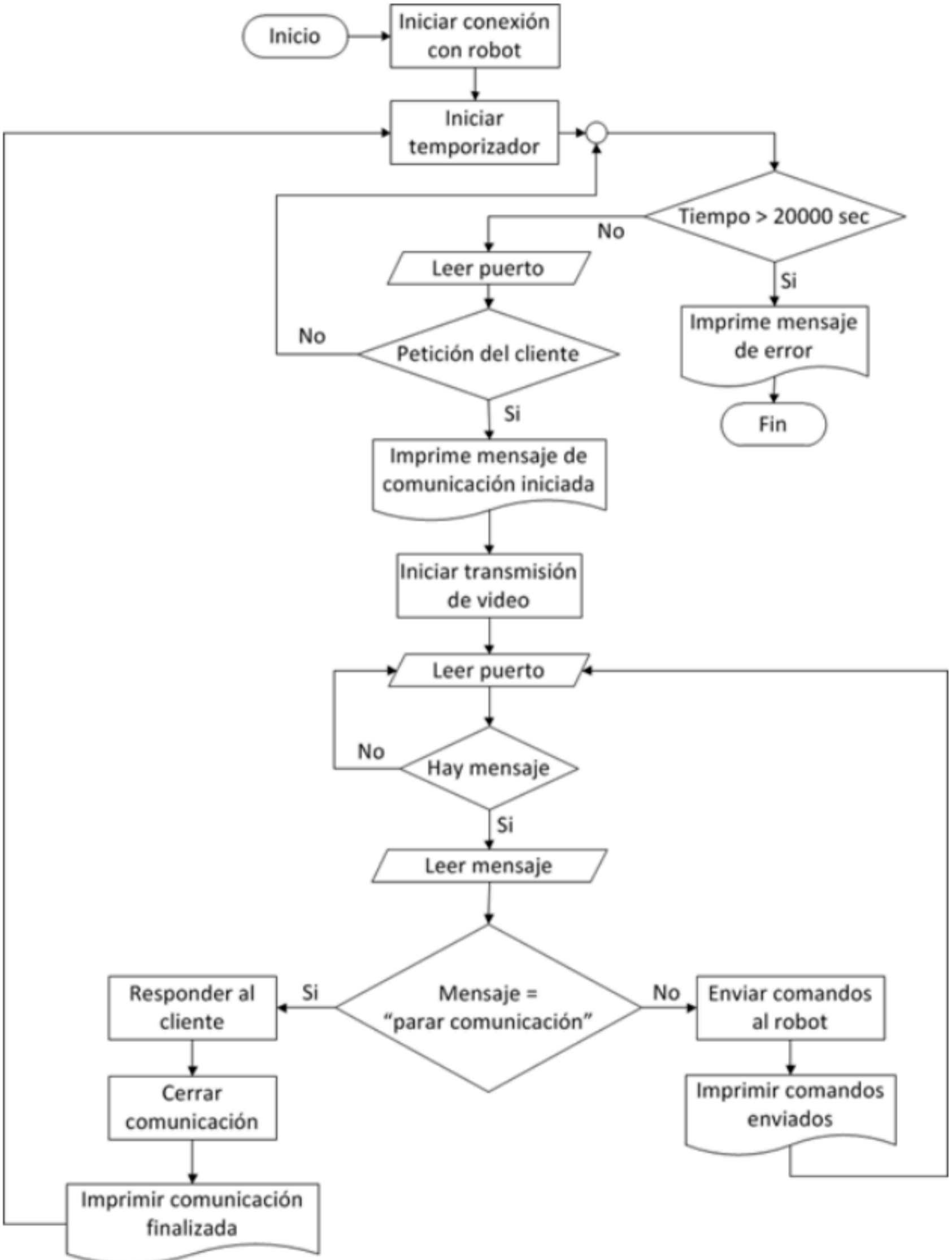
3. Resultados: interfaz gráfica para facilitar la teleoperación

Los programas y funciones desarrollados basados en los códigos expuestos bien pueden ser ejecutados para enviar y recibir información a través de la ventana de comandos (Command Window) de Matlab. Sin embargo, es aconsejable el desarrollo de una interfaz gráfica que facilite el uso de la herramienta desarrollada y cree un ambiente amigable e interactivo que favorezca la academia (Piepmeier, Knowles & Bishop, 2003; Hamilton, 2007). La interfaz gráfica desarrollada es para el cliente y se compone de cuatro bloques básicos, presentados en la Figura 6, tal y como se observa cuando se ejecuta la aplicación desarrollada con Matlab-Guide desde Bucaramanga. No es obligatoria la sección de "Posición esperada" (pre-visualización) pero garantiza que la posición deseada del robot no incurrirá en una singularidad. A continuación se debe establecer la conexión con el servidor ubicado en Barranquilla, introduciendo la IP y el puerto de datos en el bloque de "Establish Communication", e inmediatamente se observa el robot en el laboratorio de la Universidad del Atlántico, ver Figura 7.

El stream de video se hace a través de www.webcamxp.com. La interfaz indica cuando la comunicación se ha establecido con el indicador de color verde. Se puede entonces escoger los valores de los ángulos que determinen la posición deseada. La secuencia de figuras mostradas en la Figura 8 ilustra el proceso de conexión descrito. Un video que demuestra el uso de la interfaz para la teleoperación está disponible bajo solicitud. La interfaz de cliente desarrollada incluye un total de 22

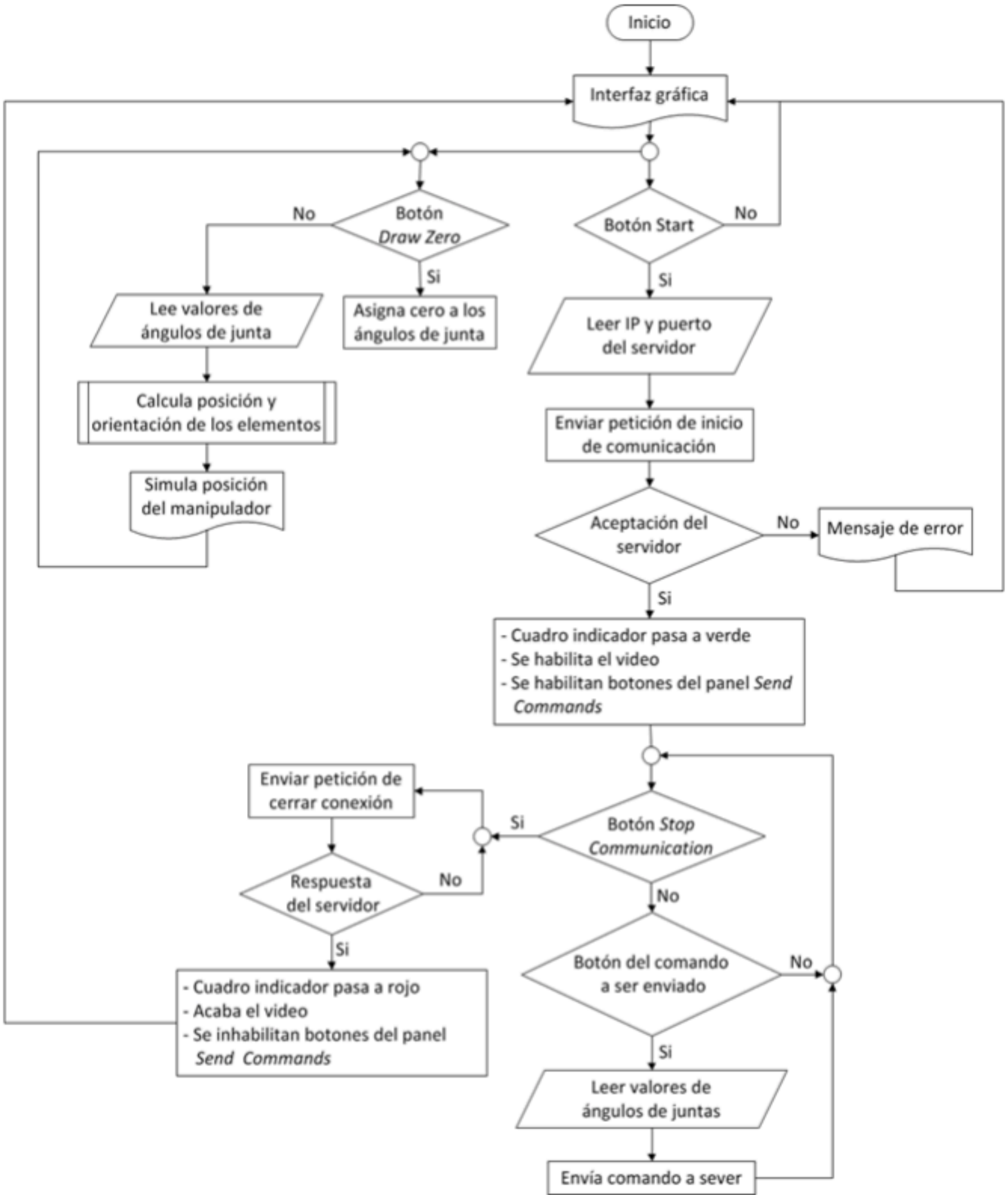
archivos para ejecución de conexión, gráfico de pre-visualización y envío de comandos. La interfaz está disponible para ser utilizada desde cualquier Universidad del país como cliente, que no cuente con un robot Movemaster RV-M1 y que dese incluirlo como prácticas remotas en sus cursos de robótica, de manera que se fortalezca la enseñanza de la robótica en el país por medio de la teleoperación.

Figura 4
Diagrama de flujo para la programación del servidor



Fuente: elaboración propia

Figura 5
Diagrama de flujo para la programación del cliente



Fuente: elaboración propia

Figura 6
Estructura de la GUI desarrollada



Fuente: elaboración propia

Figura 7

Comunicación establecida para la teleoperación



Fuente: elaboración propia

Figura 8

Secuencia de teleoperación

Paso 1: Inicio de programas



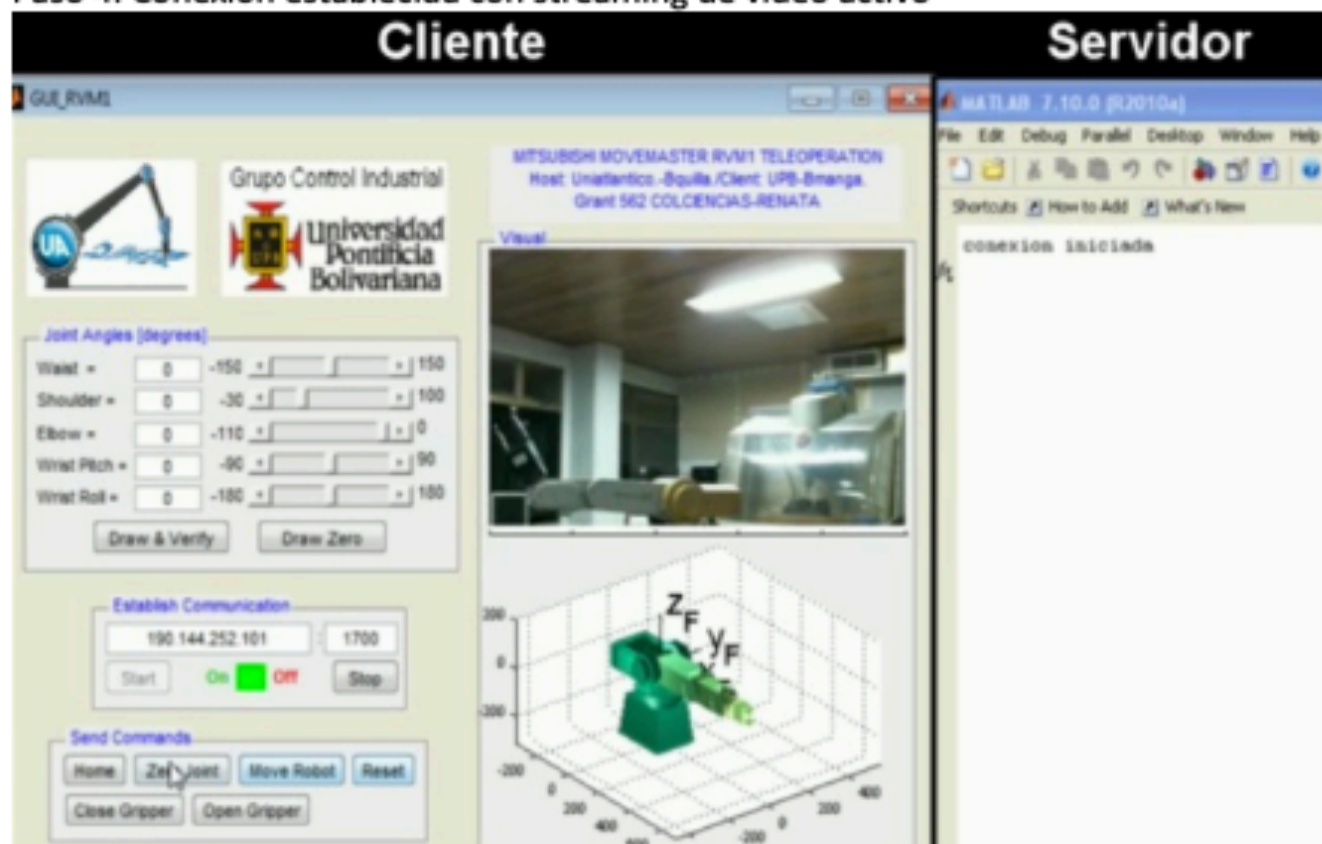
Paso 2: Ingreso de IP y puerto de acceso del host



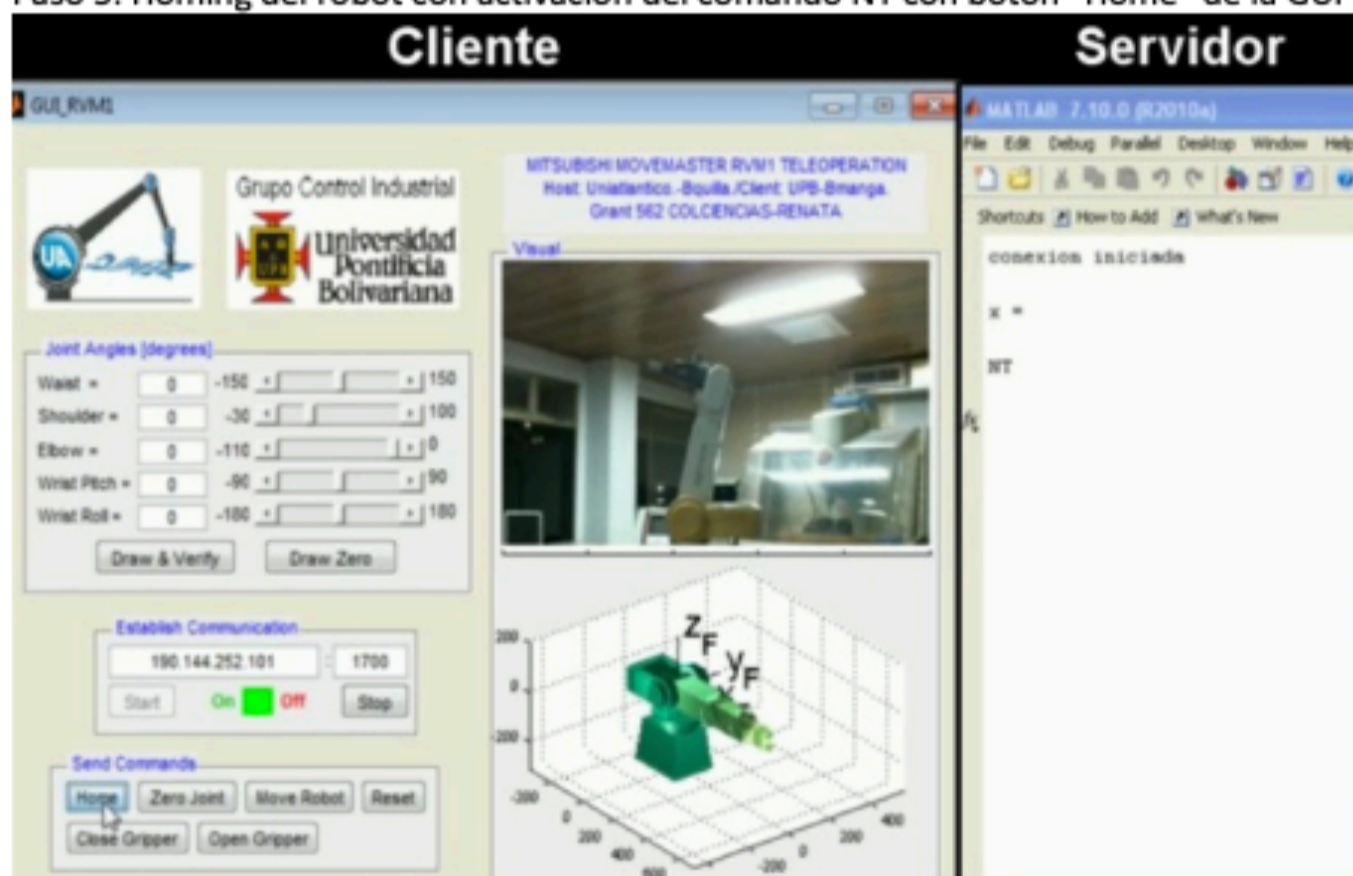
Paso 3: Reconocimiento preliminar a través de webcam.xp



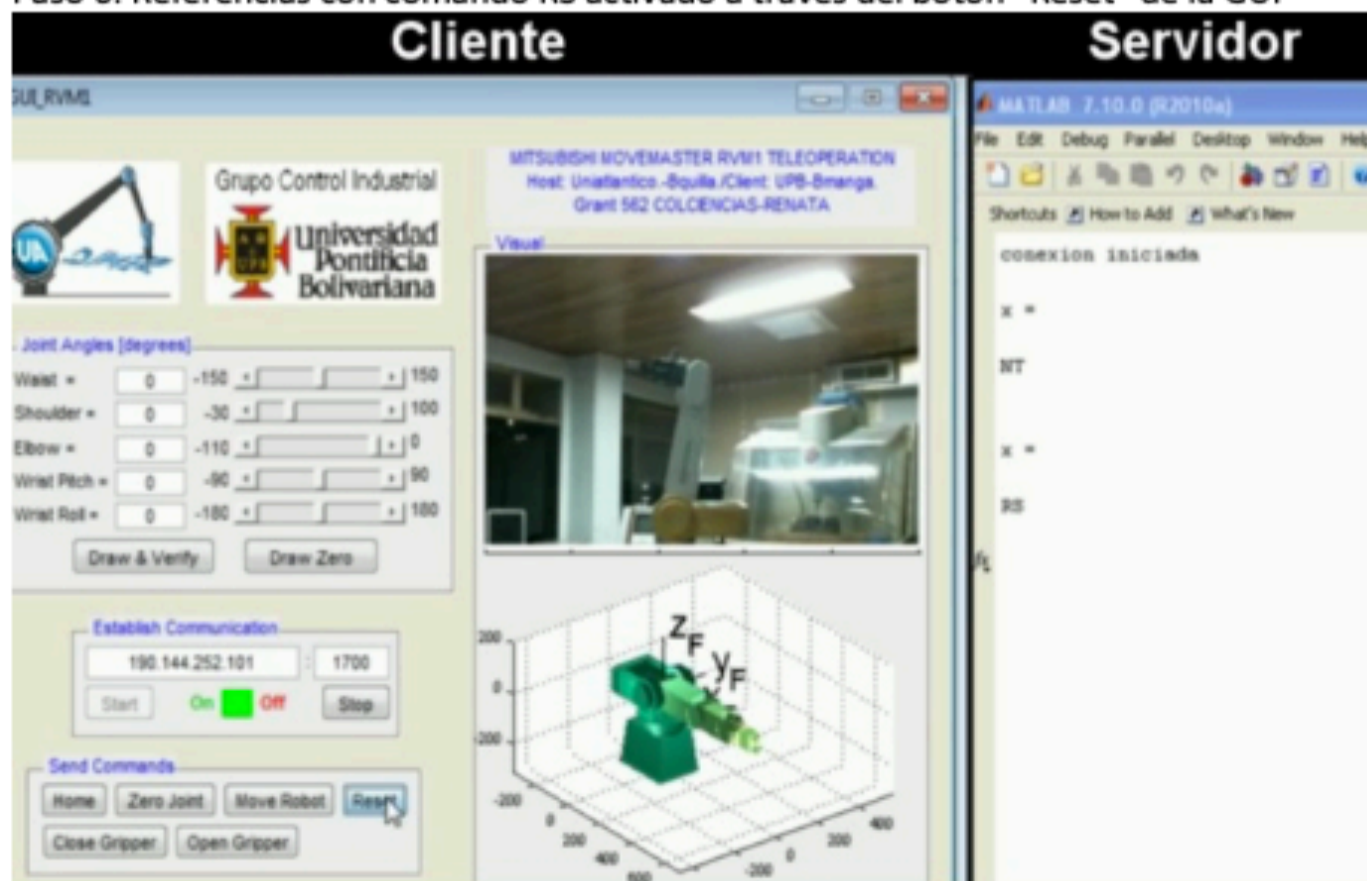
Paso 4: Conexión establecida con streaming de video activo

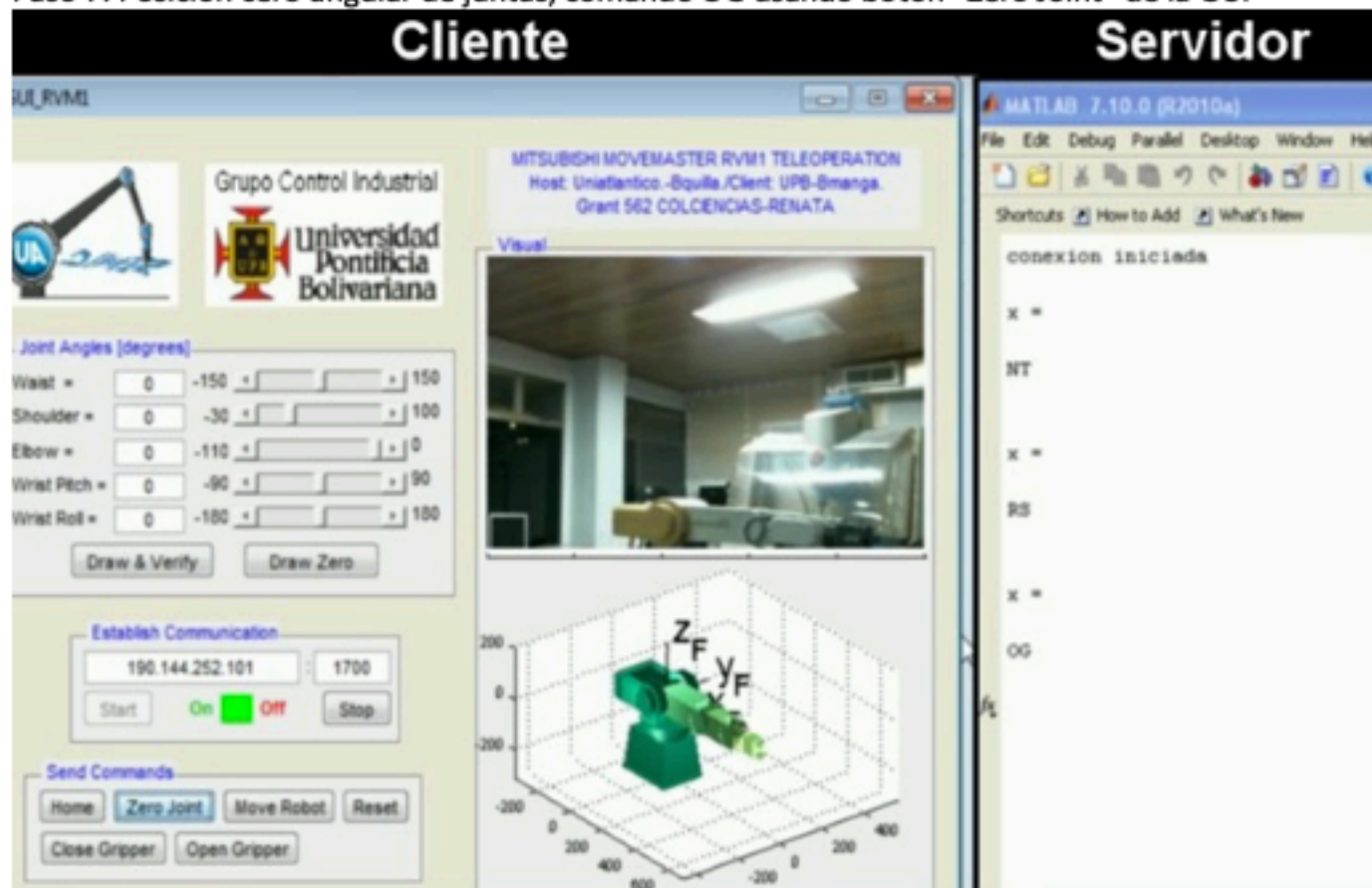


Paso 5: Homing del robot con activación del comando NT con botón "Home" de la GUI

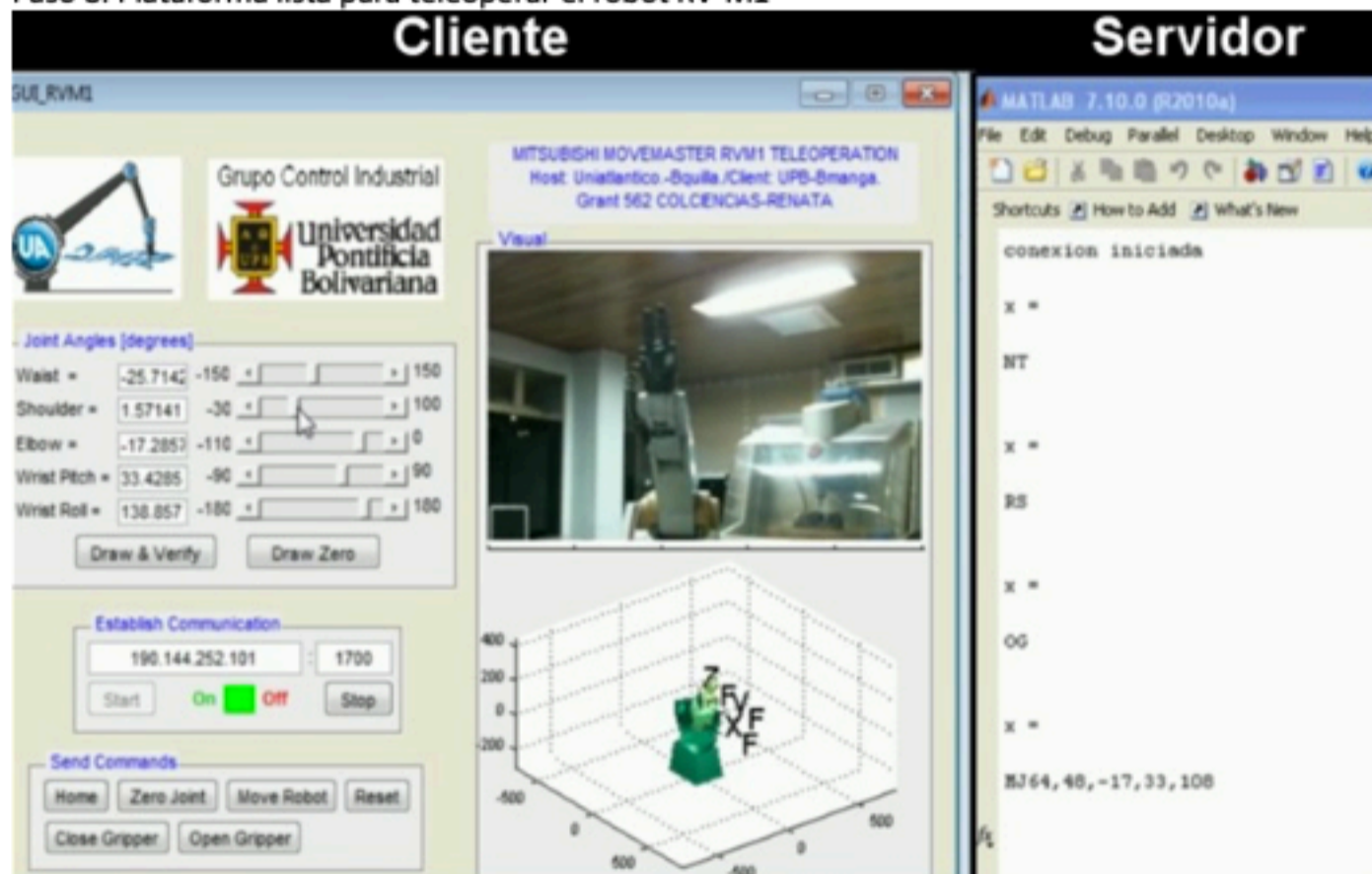


Paso 6: Referencias con comando RS activado a través del botón "Reset" de la GUI





Paso 8: Plataforma lista para teleoperar el robot RV-M1



(Fuente: elaboración propia)

4. Conclusiones

Se presentó el resumen de los aspectos de programación más importantes en el diseño de los programas (cliente y servidor) que permitan la teleoperación de un robot con amplio uso en la academia, sin necesidad de modificar la limitante arquitectura del controlador suministrado por el fabricante. Se encontró que, aunque únicamente se desee desempeñar una básica operación de pick-and-place que típicamente necesita de tres movimientos básicos, se necesitan habilitar un total de seis funciones del robot. Se presentaron los comandos de Matlab más relevantes en los cuales se basan los programas que generan la interfaz de teleoperación más sencilla posible para el Movemaster RV-M1. La existencia del Open Source Community facilita el protocolo de comunicación, ahorrando tiempo en la programación. El enfoque de este reporte es puramente académico, dirigido a los laboratorios de Universidades que tienen (servidores) o no (clientes) el manipulador RV-M1 y está enmarcado en la creación de un laboratorio de teleoperación por visión.

Agradecimientos

Los autores agradecen el apoyo brindado por RENATA-COLCIENCIAS Convocatoria 487, y a la

Referencias bibliográficas

- Ahn, S., Stik, B., & Sup, J. (June 2001). *A teleoperation position control for 2-dof manipulators with control input saturation*. Proceedings of the IEEE International Symposium on Industrial Electronics- ISIE, Pusan, Korea, pp. 1520-1525.
- Al-Mouhamed, M., Nazeeruddin, M., & Isla, S. (2010). **Experimental evaluation of feedback modalities for five teleoperation tasks**. *IEEE Transactions on Instrumentation and Measurement*, 59(2), pp. 361-371.
- Álvarez, B., Ortiz, F., Pastor, J., Sánchez, P., Losilla, F., y Ortega, N. (2006). **Arquitectura para control de robots de servicio teleoperados**. *Revista Iberoamericana de Automática e Informática Industrial*, 3(2), pp. 79-89.
- Andreu, D., Fraise, P., Roqueta, V., & Zapata, R. (December 2003). *Internet enhanced teleoperation towards a remote supervised delay regulator*. Proceedings of the IEEE International Conference on Industrial Technology-ICIT, Maribor, Slovenia, pp. 663-668.
- Aracil, R., Barrio, J., Ferre, M., y Bogado, J. (2004). *Identificación y modelado de un sistema maestro-esclavo para teleoperación*. Memorias de XXV Jornada de Automática, Ciudad Real, España, Septiembre 8-10.
- Aschenbrenner, D., Fritscher, M., Sittner, F., Krauf, M., & Schilling, K. (2015). **Teleoperation of an industrial robot in an active production line**. *International Federation of Automatic Control*, 48(10), pp. 159-165.
- Bartlett, K. (2009). *TCP/IP communications in Matlab (Matlab Central File Exchange)*. Recuperado de: <http://www.mathworks.com/matlabcentral/fileexchange/24524-tcpip-communications-in-matlab>
- Baczynski, M. & Baczynski, J. (December 2003). *Palm IIIc as advanced, low cost master part of teleoperation system*. Proceedings of the IEEE International Conference on Industrial Technology- ICIT, Maribor, Slovenia, pp. 714-717.
- Boboc, R., Moga, H., & Talaba, D. (2012). **A review of current applications in teleoperation of mobile robots**. *Bulletin of the Transilvania University of Braşov Series I: Engineering Sciences*, 5(54)2, pp. 9-16.
- Butner, S., & Ghodoussi, M. (2003). **Transforming a surgical robot for human telesurgery**. *IEEE Transactions on Robotics*, 19(5), pp. 818-824.
- Caicedo, E., Bacca, E., Calvache, B., Cardona, J., y Buitrago, J. (2009.a). **Laboratorio distribuido con acceso remoto a través de RENATA para la experimentación en robótica**. *Revista Educación en Ingeniería*, 4(7), pp. 51-61.
- Candelas, F., y Sánchez, J. (2005). **Recursos didácticos basados en internet para el apoyo a la enseñanza de materias del área de ingeniería de sistemas y automática**. *Revista Iberoamericana de Automática e Informática Industrial*, 2(2), pp. 93-101.
- Candelas, F., Jara, C., & Torres, F. (2006). **Flexible virtual and remote laboratory for teaching Robotics**. *Current Developments in Technology-Assisted Education*, 3, pp. 1959-1963.
- Codourey, A., Rodriguez, M., & Pappas, I. (July 1997). *A task-oriented teleoperation system for assembly in the microworld*. Proceedings of the IEEE International Conference on Advanced Robotics-ICAR, Monterey, CA, USA, pp. 235-240.
- Cortés, O., Sánchez, G., Roldán-Mckinley, J., y Yime-Rodríguez, E. (2012). **RSM en la minimización del retardo en la red durante teleoperación de robots**. *e-colabora: Revista de Ciencia, Educación, Innovación y Cultura Apoyadas por Redes de Tecnología Avanzada*, 2(4), pp. 118-140.
- Crane, C., Duffy, J., & Carnahan, T. (1991). **A kinematic analysis of the space station remote manipulator system**. *Journal of Field Robotic Systems*, 8(5), pp. 637-658.
- Crane, C., Duffy, J., & Carnahan, T. (1992). **A kinematic analysis of the modified flight telerobotic servicer manipulator system**. *Journal of Robotic Systems*, 9(3), pp. 461-480.
- Crane, C. & Duffy, J. (2007). *Kinematic Analysis of Robot Manipulators*. New York: Cambridge University Press.
- Dragan, A., Srinivasa, S., & Lee, K. (2013). **Teleoperation with intelligent and customizable interfaces**. *Journal of Human-Robot Interaction*, 2(2), pp. 33-57.

- Duffy, J., & Rooney, J. (1975). **A foundation for a unified theory of analysis of spatial mechanisms.** *Journal of Engineering for Industry, Trans. ASME*, 97(4)B, pp. 1159-1164.
- Duffy, J., & Crane, C. (1980). **A displacement analysis of the general spatial seven link, 7R mechanism.** *Mechanism and Machine Theory*, 15, pp. 153-169.
- Foit, K. (2008). **Remote programming of the Mitsubishi Movemaster robot by using the web-based interface.** *Journal of Achievements in Materials and Manufacturing Engineering*, 31(2), pp. 639-645.
- Gómez, J., Ollero, A., y García, A. (2006). *Teleoperación y Telerobótica*. Madrid: Pearson-Prentice Hall.
- Goldberg, K., Mascha, M., Gentner, S., Rothenberg, N., Sutter, C., & Wiegley, J. (May 1995). *Desktop teleoperation via the World Wide Web*. Proceedings of the IEEE International Conference on Advanced Robotics-ICAR, Nagoya, Japan, pp. 654-659.
- Goyal, K., & Sethi, D. (2010). **An analytical method to find the workspace of a robotic manipulator.** *Journal of Mechanical Engineering, Transactions of the Mechanical Engineering Division-The Institution of Engineers-Bangladesh*, 41(1), pp. 25-30.
- Guthart, G., & Salisbury, J. (April 2000). *The intuitive TM telesurgery system: Overview and application*. Proceedings of the IEEE International Conference on Advanced Robotics-ICAR, San Francisco, CA, USA, pp. 618-621.
- Hamilton, C. (2007). **Using MATLAB to advance the robotics laboratory.** *Journal of Computer Applications in Engineering Education*, 15(3), pp. 205-213.
- Hu, H., Yu, L., Wo, P. & Zhou, Q. (2001). **Internet-based robotic systems for teleoperation.** *International Journal of Assembly Automation*, 21(2), pp. 1-10.
- Jazar, R. (2007). *Theory of Applied Robotics: Kinematics, Dynamics and Control*. New York: Springer.
- Jiménez, L., Puerto, R., Reinoso, O., Fernández, C., y Ñeco, R. (2005). **Recolab: Laboratorio remoto de control utilizando Matlab y Simulink.** *Revista Iberoamericana de Automática e Informática Industrial*, 2(2), pp. 64-72.
- Kofman, J., Wu, X., Luu, T., & Verma, S. (2005). **Teleoperation of a robot manipulator using a vision-based human-robot interface.** *IEEE Transactions on Industrial Electronics*, 52(5), pp. 1206-1219.
- Kumar, R., Kalra, P., & Prakash, N. (2011). **A virtual RV-M1 robot system.** *Journal of Robotics and Computer-Integrated Manufacturing*, 27(6), pp. 994-1000.
- Ming, H., Huat, Y., Lin, W., & Hui, Z. (April 1996). *On teleoperation of an arc welding robotic system*. Proceedings of the IEEE International Conference on Advanced Robotics-ICAR, Minneapolis, MN, USA, pp. 1275-1280.
- Mitsubishi Electric Corporation. (1992). *Industrial Micro-Robot System User Manual, ModelRV-M1*. Japan: Mitsubishi Corporation.
- Mostefa, M., Kaddour-El-Boudadi, L., & Vareille, J. (2018). **Safe and efficient mobile robot teleoperation via a network with communication delay.** *International Journal on Interactive Design and Manufacturing*, 12(1), pp. 37-47.
- Muradore, R., & Fiorini, P. (2016). **A review of bilateral teleoperation algorithms,** *Acta Polytechnica Hungarica*, 13(1), pp. 191-218.
- Piepmeyer, J., Knowles, K., & Bishop, B. (2003). **The use of Matlab for robotic control in an undergraduate robotics laboratory.** *Journal of Computers in Education*, 13(3), pp. 68-75.
- Preusche, C., Ortamaier, T., & Hirzinger, G. (2002). **Teleoperation concepts in minimal invasive surgery.** *Control Engineering Practice*, 10(11), pp. 1245-1250.
- Roldán, J., Sosa, J., Yime, E., y Díaz, J. (2012). **Cinemática inversa matricial del manipulador 5R Mitsubishi Movemaster RV-M1.** *Revista Épsilon*, 19, pp. 33-56.
- Schulz, D., Burgard, W., Fox, D., Thrun, S., & Cremers, A. (2000). **Web interface for mobile robots in public places.** *IEEE Robotics and Automation Magazine*, pp. 48-56.
- Świder, J., Foit K., Wszolek, G., & Mastrowski, D. (2007.a). **The off-line programming and simulation software for the Mitsubishi Movemaster RV-M1 robot.** *Journal of Achievements in Materials and Manufacturing Engineering*, 20(1-2), pp. 499-502.
- Świder, J., Foit, K., Wszolek, G., & Mastrowski, D. (2007.b). **The system for simulation and off-**

line, remote programming of the Mitsubishi Movemaster RV-M1 robot. *Journal of Achievements in Materials and Manufacturing Engineering*, 25(1), pp. 7-14.

Tavakoli, M., Patel, R., & Moallem, M. (2005). **Haptic interaction in robot-assisted endoscopic surgery: a sensorized end-effector.** *The International Journal of Medical Robotics and Computer Assisted Surgery*, 1(2), pp. 53-63.

Volpe, R., & Udomkesmalee, G. (2017). *NASA Mobility and robotics systems-Section 347*. Recuperado de: <http://www-robotics.jpl.nasa.gov/>

Yang, X., Chen, Q., Petriu, D., & Petriu, E. (October 2004). *Internet-based teleoperation of a robot manipulator for education*. Proceedings of The 3rd IEEE International Workshop on Haptic, Audio and Visual Environments and Their Applications-HAVE, Ottawa, Canada, pp. 7-11.

1. Grupo de Investigación DIMER. Universidad del Atlántico. Programa de Ingeniería Mecánica. Profesor Asociado. javierroldan@mail.uniatlantico.edu.co

2. Grupo de Investigación DIMER. Universidad del Atlántico. Programa de Ingeniería Mecánica. Profesor Asociado. eugenioyime@mail.uniatlantico.edu.c

3. Ingeniero Mecánico. Universidad del Atlántico. jlsosarodriguez@hotmail.com

Revista ESPACIOS. ISSN 0798 1015
Vol. 40 (Nº 2) Year 2019

[Index]

[In case you find any errors on this site, please send e-mail to [webmaster](#)]