

Robots cooperativos, Quemes para la educación

Robots cooperative education Quemes

Phd. Enrique González Guerrero*

Msc. John Jairo Páez Rodríguez**

Ing. Fabián José Roldán***

Fecha de recepción: 14 de abril 2013

Fecha de aceptación: 5 de mayo de 2013

Resumen

Este trabajo nace de los resultados obtenidos en la implementación del proyecto *Quemes* en el año 2010 por el grupo SIDRe de la Pontificia Universidad Javeriana en convenio con Maloka (González, 2010). *Quemes* es una propuesta pedagógica fundamentada en la teoría de motivación para la creación, que utiliza como herramienta de mediación cognitiva robots para desarrollar competencias relacionadas con el trabajo cooperativo.

Los resultados obtenidos de la investigación del 2010 evidencian la necesidad de ampliar el potencial pedagógico, aumentando las actividades de cooperación entre los robots. Para lograrlo se plantea como objetivo general extender las actividades educativas de la plataforma *Quemes* con el diseño de nuevas tareas de cooperación entre los robots, a través de la incorporación de una pinza para sujetar objetos geométricos simples.

* Pontificia Universidad Javeriana. Director Maestría en Ingeniería de Sistemas y Computación. Bogotá, Colombia. Correo electrónico: egonzal@javeriana.edu.co

** Pontificia Universidad Javeriana. Universidad Distrital Francisco José de Caldas. Bogotá, Colombia. Correo electrónico: jjpaez2012@udistrital.edu.co

*** Pontificia Universidad Javeriana. Estudiante de maestría en Ingeniería de Sistemas y Computación. Bogotá, Colombia. Correo electrónico: foldan@javeriana.edu.co

Palabras claves: Robotica Móvil, Sistemas Multi Agente, Educación en Tecnología.

Abstract

This document describes the integrating a gripper on *Quemes* robots to increase their cooperation strategies, and thus broaden their educational potential. The integration of the clip begins with the design process and after mechanical built, the model is developed to link it to the control logic of each robot embedded software. The validation process can be said that the diversity of activities in *Quemes* favors the learning process of children and improving their attitudes toward teamwork.

Key words: Mobile Robotics, Multi Agent Systems, Technology Education

1. Antecedentes

A. Robótica y educación en tecnología

La educación en tecnología ha presentado una dinámica escolar muy marcada por las concepciones acerca de lo que significa la tecnología. Los trabajos de (Andrade, 1998), (Buch, 1999) y (Layton, 1993) presentan una mirada académica sobre la naturaleza del conocimiento tecnológico y sus implicaciones educativas. Se han propuesto diferentes objetivos, estrategias y ambientes de aprendizaje en su desarrollo. Es así como la robótica es considerada como una alternativa didáctica que facilita los procesos de aprendizaje (Eric, Feng, & Chun, 2010).

Dentro de su uso en la educación encontramos tres tendencias:

- Robot como proyecto de programación; en donde este se encuentra ensamblado o está dentro de un simulador y por medio de una interfaz gráfica se programa su comportamiento. (Mansi, Marwa, & Shikha, 2010), (Denis & Hubert, 2007), (Pavel & Richard, 2010).
- Robot como centro de aprendizaje; en donde se utilizan kits (Mitnik, Recabarren, Nussbaum, & Soto, Collaborative robotic instruction: A graph teaching experience, 2010) o se construyen modelos por completo (Ruiz, 1996). En los dos casos el proceso de aprendizaje gira en torno a la construcción del robot.
- Robots como agentes colaborativos en el aprendizaje; aquí el comportamiento de los robots se programa utilizando técnicas de inteligencia computacional como redes neuronales, lógica difusa, aprendizaje por refuerzo, árboles de decisión y redes bayesianas (Douglas, Deepak, James, & Lisa, 2007), con el objetivo que el robot aprenda comportamientos en función de la interacción con el niño (González, 2010). Las anteriores propuestas de sistemas inteligentes en robótica aplicados a la educación, (Orazio, Onofrio, Michela, & Stefano, 2007) nacen del área de los tutores inteligentes donde se abordan aspectos como: representación del cono-

cimiento, modelo pedagógico del tutor, modelo del estudiante y construcción del Tutor Inteligente. (Nkambou, Bourdeau, & Mizoguchi, 2010).

Las principales ventajas que tienen los ambientes de aprendizaje usando robots son las de permitir la integración de distintas áreas del conocimiento, operación con objetos manipulables para favorecer el paso de lo concreto a lo abstracto, operación y control de distintas variables al mismo tiempo, el desarrollo de un pensamiento sistémico y el aprendizaje autónomo (Ruiz, 1996). Para evidenciar lo anterior (Chil, Alonso, Ortiz, & Pastor, 2011) los autores proponen facilitar el aprendizaje de conceptos básicos de programación a través de la construcción de un robot programable; en Japón se presenta un robot que “susurra” palabras que sirven como activador cognitivo durante el proceso de aprendizaje en el área de matemáticas (Kayako, Masahiro, Kazuhiko, Reo, Hiroshi, & Norihiro, 2011), otra propuesta afirma que los profesores necesitan nuevas herramientas para atraer a los estudiantes en el estudio de aspectos técnicos, especialmente a las niñas, quienes son las que presentan mayor desinterés (Vollmer, Jeschke, Burr, Knipping, Scheurich, & Wilke, 2011).

Diversas experiencias de aprendizaje a través de la robótica demuestran que su uso genera un puente entre la motivación extrínseca e intrínseca que involucra al estudiante dentro del proceso; un aspecto importante es que las capacidades representativas del sujeto permiten estimar un número de posibles estados del robot, anticiparse a los resultados y seleccionar las acciones en donde se obtengan recompensas acordes a su motivación intrínseca. Por ejemplo el trabajo de Barak y Zadok, muestra cómo los estudiantes se motivan durante la construcción de diferentes robots para la solución de pro-

blemas complejos. Un ejemplo es la opción de estimular la motivación intrínseca a partir de la siguiente estrategia: los estudiantes deben programar el robot para que haga lo que ellos quieran y no las actividades que habitualmente plantean los docentes cuando desean enseñar un tema de programación. (Marian & Blaine, 2004).

Otra razón es la mirada filosófica del animismo, especialmente en la cultura oriental, que plantea la tendencia que tienen los humanos para creer que los objetos inanimados están dotados de personalidad, alma y vida (Brown & Walker, 2008). Desde el ámbito de las ciencias sociales, se han desarrollado investigaciones en relación al animismo y el pensamiento infantil (Deval, 1975) afirmando que su origen está relacionado con la necesidad del hombre de explicar lo fortuito y la regularidad de las cosas. Pero no solo la apariencia de los robots es la base para la postura del animismo, ya que autores como (Scalzone & Tamburrini, 2012) plantean un estudio desde el psicoanálisis basados en las teorías de Freud sobre la necesidad de reflejar en los robots algún contenido mental que está reprimido o de utilizarlos como mecanismo para evadir la realidad.

Otro elemento de la robótica está relacionado con el Aprendizaje Cooperativo, elemento clave en la propuesta *Quemes*, considerado como una estrategia que promueve la participación colaborativa entre los estudiantes y cuyo propósito es conseguir que se apoyen mutuamente para alcanzar sus objetivos individuales y alcancen logros de aprendizaje de modo mutuo. “El aprendizaje cooperativo es el uso instructivo de grupos pequeños para que los estudiantes trabajen juntos y aprovechen al máximo el aprendizaje propio y el que se produce en la interrelación» (Johnson & Johnson, 1999).

El análisis en las investigaciones que utilizan robots junto con la metodología de aprendizaje cooperativo comienza con (Brigitte & Sylviane, 2001), quienes plantean la importancia de aprender a trabajar de forma colaborativa y desarrollan una estrategia de trabajo basada en el aprendizaje por proyectos, donde los estudiantes forman grupos entre 2 y 4 personas para la solución de un problema usando robots. También encontramos en el proyecto desarrollado por (Mitnik, Recabarren, Nussbaum, & Soto, 2010) una estrategia de aprendizaje con trabajo colaborativo para el desarrollo e interpretación de gráficas en el área de física con el uso de robots controlados por una PockedPC. En este trabajo los estudiantes desarrollan el aprendizaje de forma cooperativa cuando programan y evalúan de forma sincrónica los comportamientos cinemáticos del robot previamente programado.

En conclusión, la motivación y el trabajo en equipo son dos elementos que se destacan cuando se usan robots en la educación, ya que permiten el desarrollo de proyectos que requieren la integración de diferentes áreas del conocimiento en la solución de un problema, además toman los robots como un objeto de estudio que se articula con las demás asignaturas. En este sentido la propuesta de la plataforma *Quemes* aborda estos elementos cuando plantea actividades con robótica cooperativa que requieren la integración de distintos saberes y habilidades en los estudiantes para la solución de un problema débilmente estructurado.

B. Robots *Quemes*

La plataforma *Quemes*, inicialmente desarrollada por los grupos de investigación SIDRe y Takina de la Pontificia Universidad Javeriana, GIAP de la Universidad de los Andes y Maloka (Bustacara, Roldan, Buitrago, &

González, 2010). Es una herramienta pedagógica que permite a los usuarios ver reflejado el diseño y la programación orientada a eventos bajo el paradigma de computación de Sistemas Multi-Agentes de manera gráfica y muy intuitiva. Para la implementación de control de los robots se utilizó la plataforma BESA, que es una plataforma desarrollada en el Departamento de Ingeniería de Sistemas de la Pontificia Universidad Javeriana que facilita la implementación y ejecución de Sistemas Multi-Sgentes (SMA) (González, Avila, & Bustacara, 2003).

En una aproximación a los principios de *Quemes*, lo primero es el conocimiento de las capacidades físicas de los robots, compuestos de cuatro sensores infrarrojos y un dispositivo de comunicación que en conjunto permite navegar en un ambiente plano de color blanco, compuesto por una grilla elaborada en líneas negras. Cada robot cuenta con un chip que tiene implementado un sistema operativo en tiempo real para el control lógico de las diferentes tareas relacionadas con la identificación de cada robot, el movimiento de los motores en las dos ruedas, la lectura de los cuatro sensores infrarrojos para determinar los estados en el ambiente, la recepción y envío de los mensajes de comunicación que permiten que trabajen de forma cooperativa y, con este trabajo el control de la pinza.

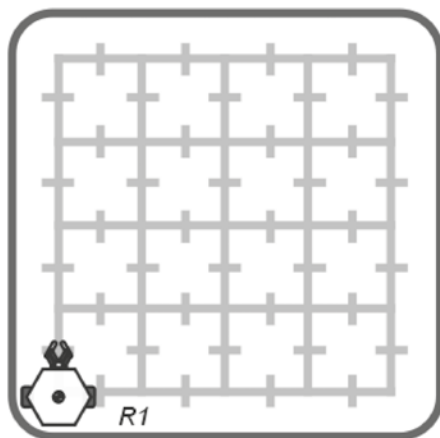
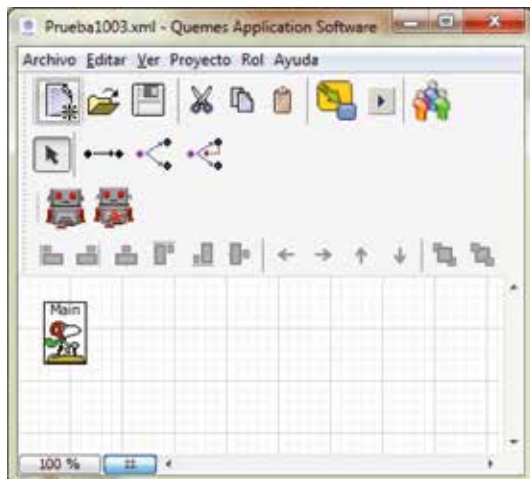
El segundo aspecto está relacionado el software llamado *Quemes*, que a través de una interfaz gráfica permite la programación de los robots bajo distintos aspectos relacionados con la cooperación. Es importante resaltar que entre el Software *Quemes* y los robots hay un medio de comunicación para establecer las condiciones de trabajo y solución del problema.

Para desarrollar actividades con los robots es necesario construir con cartulina blanca y

cinta negra una grilla que represente el ambiente de navegación de los robots, por ejemplo una ciudad. El tamaño del ambiente está relacionado con las características del problema planteado y con la cantidad de robots presentes durante la solución. La figura 1 presenta los tres conceptos generales abordados: Interfaz gráfica, Robots y Entorno de Navegación.

Figura 1. Componentes principales de Quemes.

a) Interfaz Gráfica b) Entorno de Navegación



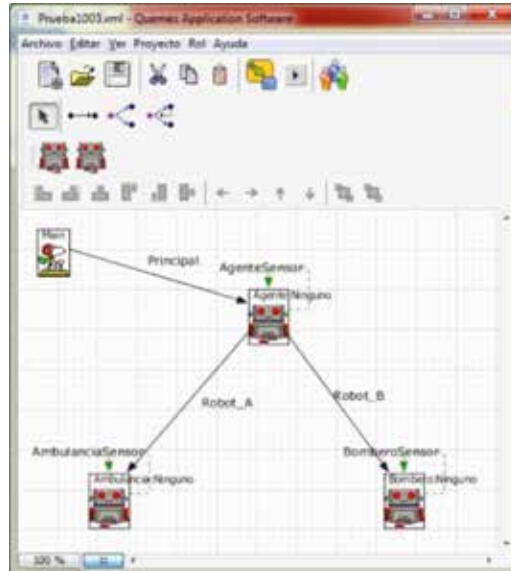
Fuente: elaboración propia

Para el funcionamiento de los robots es necesario diseñar la estrategia basada en Sistemas Multi-Agentes y después de identificar los elementos básicos de cooperación es necesario programar los robots mediante los siguientes pasos:

- Diseño del robot en la plataforma: para el diseño es necesario tener claridad en el rol que desempeña y las habilidades que él tiene. El comportamiento del robot se desarrolla mediante la interfaz gráfica de *Quemes* y la herramienta de programación llamada Percival, en donde se implementa el programa que indica las acciones en función de los eventos.
- La creación de los vínculos: los vínculos se establecen de acuerdo a las condiciones del problema y son: punto a punto, rol a multicast y rol a rol puntual.

A continuación se presenta una aproximación del diseño en la plataforma *Quemes*. La figura 2a presenta los vínculos entre roles y la figura 2b una aproximación a la herramienta desarrollada por Maloka, llamada Percival; en donde se definen los comportamientos de los agentes en función de los eventos. Finalmente el programa se compila y automáticamente se inicia la comunicación mediante un canal inalámbrico para que los robots desarrollen el trabajo.

Figura 2. Vínculos entre robot y diseño en la herramienta Percival



a)



b)

Fuente: elaboración propia

Después de analizar los antecedentes, se puede afirmar que la propuesta *Quemes* aborda de manera innovadora el paradigma educativo del constructivismo, la metodología de solución de problemas y robótica como herramienta de motivación durante el proceso de aprendizaje.

Desde la propuesta técnica, la construcción de los robots y el modelo de programación, se evidencia la aplicación de los conceptos más significativos dentro de la tendencia teórica para el diseño e implementación de la robótica cooperativa. Es así como el modelo de BESA, Contract Net, la programación gráfica de Percival, el generador de código y los robots, reflejan un desarrollo, que por

síntesis, no se encuentra en las demás propuestas consultadas. Sin embargo, es evidente que el proyecto se puede mejorar desde dos aspectos: ampliando las actividades de cooperación de los robots mediante la implementación de una pinza y diseñando nuevas actividades escolares para complementar el trabajo presente en las cartillas de "Robótica Cooperativa", propuestas en Maloka.

2. Descripción de los robots

Describir su estado inicial se limita a afirmar que se contaba con todas las partes y dispositivos electrónicos que permitían su ensamble. Sin embargo el software embebido de la versión inicial no era compatible con los nuevos requerimientos técnicos, lo que hizo necesario dedicar una gran parte del trabajo a ensamblar y hacer funcionar los robots y después sí integrar el control de la pinza. Para una comprensión del estado inicial es necesario mencionar que *Quemes* se resume en dos componentes: Robots y Control Inteligente.

Los robots tienen un diseño típico de mecanismo diferencial para realizar cinco movimientos (adelante, atrás, girar a la izquierda, derecha y sobre su propio eje) que se logran a través de dos motores DC con encoder, cuatro sensores infrarrojos ubicados en la parte posterior y una tarjeta electrónica que permite el control de velocidad a través del método de modulación de anchura de pulsos PWM.

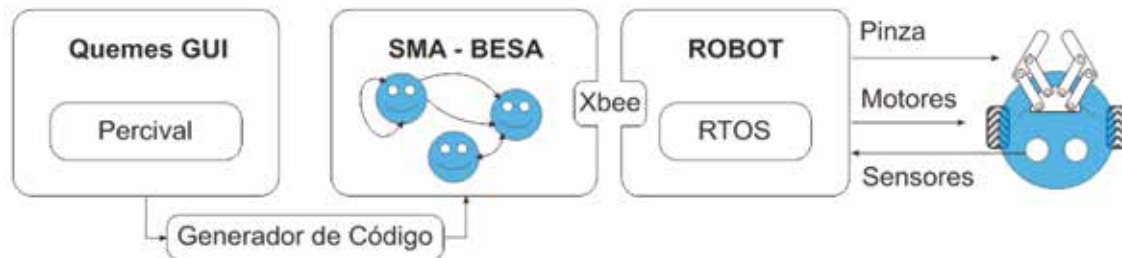
Los cuatro sensores infrarrojos obtienen información sobre el mundo para determinar

en qué estado se encuentran (cruce, siguiendo línea, girando) y así coordinar las acciones en función del desarrollo de la tarea cooperativa. Por medio del sistema embebido en RTOS se desarrolla la ejecución de las tareas de comunicación, lectura de sensores y movimiento de los motores en función de los eventos diseñados previamente en Percival.

En el Control Inteligente, la programación de alto nivel (Percival, BESA, Herramienta de caracterización de SMAs y el Generador de Código) funcionaba de manera adecuada y no se realizó ningún aporte, solo se incluyó el código necesario para el funcionamiento de la pinza. En la figura 3 se presenta la relación de los componentes básicos para el funcionamiento del sistema y se da una breve explicación de los aspectos más relevantes.

Percival es un software desarrollado por Maloka que permite diseñar algoritmos de programación a través de diagramas de flujo en una interfaz gráfica. En *Quemes*, invoca los archivos XML que resumen los mecanismos de cooperación y comunicación durante la creación de los Sistemas Multi-Agentes para que el usuario establezca, de forma gráfica, las variables relacionadas con el estado del agente, las variables locales del evento, las asociadas al evento de entrada y de salida. Todo esto para definir las reacciones de los robots de acuerdo a los posibles eventos durante el desarrollo de las tareas de cooperación.

Figura 3. Control Inteligente en Qumes



Fuente: elaboración propia

BESA es un software desarrollado por el grupo SiDRe para el diseño y construcción de Sistemas Multi-Agentes, su arquitectura se compone de tres niveles: Agente, Social y Modelo de distribución. En *Qumes*, el nivel de agente sirve para caracterizar los robots por roles y como selector de eventos basado en guardas para determinar sus comportamientos. El nivel social permite agrupar los robots en diferentes niveles de abstracción de acuerdo a una organización jerárquica definida por el usuario. Entretanto el nivel sistema es una estructura abstracta en donde los robots están reunidos y desarrollando actividades de colaboración relacionados con la ejecución de eventos de forma asincrónica.

La herramienta de caracterización de software, mediante la aplicación JHotDraw, integra los elementos de Percival y BESA para el diseño de los comportamientos de los robots al trabajar de manera cooperativa. Esto permite crearlos, editar las variables asociadas a su creación, administrar los roles, establecer los vínculos de cooperación y comunicación. Todo esto sin necesidad de contar con conocimientos profundos en el área de programación.

El Generador de Código, después de verificar los archivos en Percival, construye el proyecto e inicia el proceso de comunicación entre *Qumes* y los robots a través de los módulos Xbee. La trama de comunicación contiene información del número del robot, elementos de sincronización, comandos para realizar la acción de movimiento como las de seguir línea, girar a la derecha, girar a la izquierda y parar (con el aporte de este trabajo se agregan las acciones de abrir y cerrar pinza y la información de los sensores para estas dos acciones) y los demás comandos necesarios para el desarrollo del trabajo cooperativo.

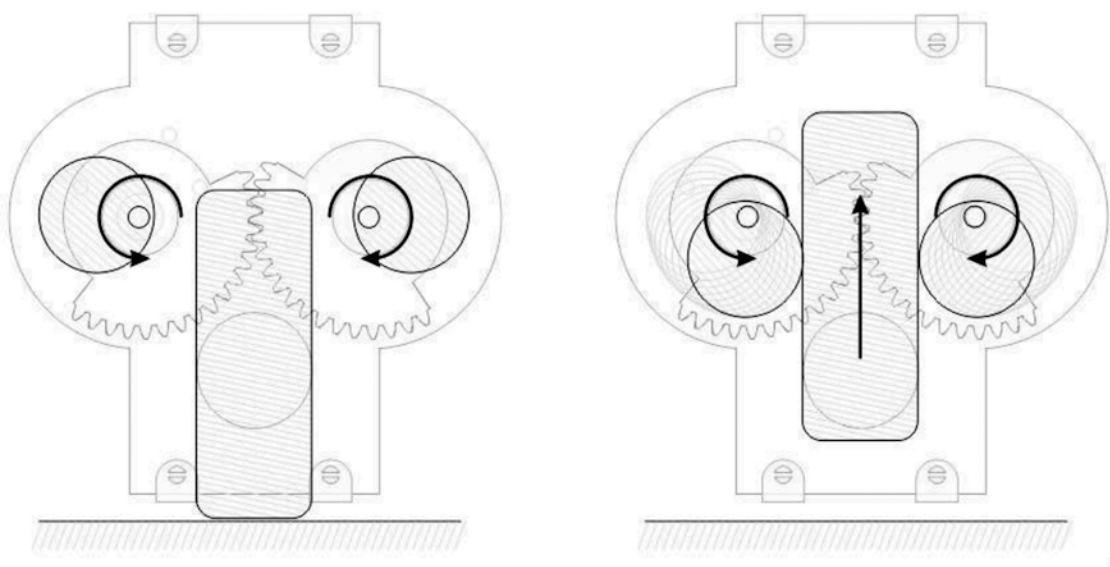
3. Diseño de la pinza

Como se menciona en la introducción los resultados obtenidos de la investigación del 2010 evidencian la necesidad de ampliar el potencial pedagógico por medio del aumento de las actividades de cooperación entre los robots. Para lograrlo se planteó como objetivo general extender las actividades educativas de la plataforma *Qumes* mediante el diseño de nuevas tareas de cooperación entre los robots a través de la incorporación de un sistema prensil de objetos geométricos simples. El proceso de diseño inicia con las siguientes condiciones:

- Generar por medio de un solo motor dos movimientos: uno horizontal para sujetar un objeto y otro vertical para levantarlo. El sistema debe ser liviano para minimizar los esfuerzos y el consumo de energía del robot durante su desplazamiento. Debe ser compacto y resistente bajo las condiciones de trabajo de la población infantil. El costo para su construcción debe ser mínimo. Considerar la posición más apropiada de la pinza en el robot.
- Para generar los dos movimientos, uno horizontal para sujetar un objeto y otro vertical para levantarlo, se utilizan los

principios de movimiento circular, fricción y elasticidad. En la figura 4 los cilindros giran en sentido contrario con una excentricidad D . A medida que esto sucede la distancia entre los dos disminuye hasta igualar el tamaño del objeto. A partir de esta condición, la fricción entre las superficies de los cilindros y el objeto va aumentando, los cilindros se deforman y como aún siguen girando provoca el movimiento vertical del objeto. El resultado de este comportamiento mecánico es la presión del objeto y un desplazamiento vertical en relación a su posición inicial que llega hasta 15 milímetros.

Figura 4. Principio de funcionamiento cinemático de la pinza

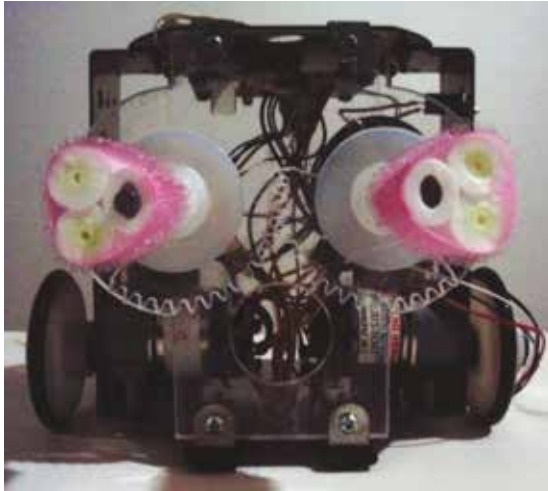


Fuente: elaboración propia

La pinza está construida en acrílico para la estructura y los piñones y nylon para los acoples entre los piñones y los ejes. Es importante aclarar que las piezas en acrílico son cortadas con láser y los acoples fueron cons-

truidos a través del proceso de torneado mecánico. La figura 5 muestra la manera en que el soporte de la pinza se utiliza para sujetar al robot y sostener todos sus elementos.

Figura 5. Implementación de la pinza en el robot



Fuente: elaboración propia

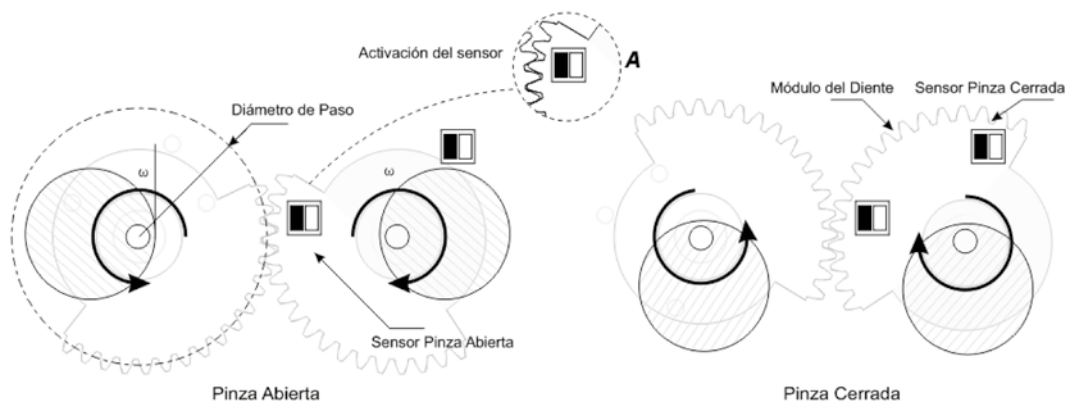
A. Control de movimiento de la pinza

Desde *Qumes GUI*, se han establecido dos comandos relacionados con los eventos de abrir y cerrar pinza, cuando a través del módulo de comunicación son enviados al robot el sistema de control activa el motor con la

velocidad y el sentido de giro correspondiente hasta que el sensor respectivo compruebe la condición de parada. Después de terminar la acción en la pinza, el software embebido permite que el robot continúe con la ejecución del programa. En la sección ampliada A de la figura 6, se observa como uno de los extremos del engranaje impulsor activa el sensor de pinza abierta y permite la condición de parada del motor; solo cambiará al estado de cierre cuando el mismo extremo del engranaje gire y coincida con el sensor de pinza cerrada. En el software embebido, el control de movimiento se logra a través de dos funciones llamadas: *abrirPinzaState* y *cerrarPinzaState*. Básicamente estas funciones asignan una velocidad y un sentido de giro al motor de la pinza de acuerdo a una condición de activación del sensor correspondiente.

La velocidad de giro está determinada por el diámetro de paso, la medida del ancho de los dientes de los engranajes y el número de dientes. Para el diseño de la pinza la relación de velocidad es 1:1 y depende del valor asignado al motor correspondiente en el software embebido a través de la función *abrirPinzaState*.

Figura 6. Estado abierto y cerrado de la pinza



Fuente: elaboración propia

B. Software embebido de los robots

Antes de abordar el rediseño del software embebido es necesario presentar una descripción de sus tres acciones básicas: Percepción, Comunicación y Movimiento.

- La percepción permite reconocer un plano de fondo blanco con vías negras que representan las calles de una ciudad. Esto se logra a través de cuatro sensores infrarrojos que están distribuidos para discriminar las vías, identificar un cruce o intersección de dos calles e identificar una esquina en el mundo sintético.
- La comunicación permite el envío de datos relacionados con su estado (ubicación en la grilla, posición de la pinza, dirección del movimiento y posibles eventos) y recepción de datos relacionados con las acciones de movimiento de los dos motores del robot y uno de la pinza.
- El movimiento está relacionado con el control de velocidad de los tres motores que conforman el robot. Dos para las ruedas y uno para la pinza.

En la nueva versión del RTOS se han dejado cinco tareas: *rmoc*, *QuemesControl*, *Receiver*, *Process* y *Transmitter*.

- *rmoc* (Robot Motion Control) envía a los motores las tres velocidades, dos para el movimiento y uno para la pinza. El valor de estas velocidades oscila entre -90 y 90 siendo 0 el valor en donde no hay movimiento en el motor.
- *QuemesControl* recibe la información con las instrucciones para movimiento del robot y la pinza (seguir línea, girar a la derecha, girar a la izquierda, parar, abrir pinza y cerrar pinza) y al finalizar envía un mensaje para indicar que ha terminado esa instrucción. La transición entre el inicio y final de los movimientos se regula a través de la información continua de los sensores infrarrojos. Por ejem-

plo, si la orden es seguir línea; el robot comienza a leer sensores y avanzar, hasta que la información de los sensores indique que ha llegado a un cruce o a una esquina, en donde envía un mensaje relacionado con la finalización de la tarea.

- *Receiver* es la tarea encargada de recibir y verificar la información que llega de *Quemes* (PC) o del mismo robot.
- *Process* es la tarea que interpreta la información y la distribuye en las variables correspondientes, lo que permite la activación y coordinación de todas las tareas que componen el software de control del robot.
- *Transmitter* es la tarea encargada de organizar y enviar la información hacia *Quemes* (PC).

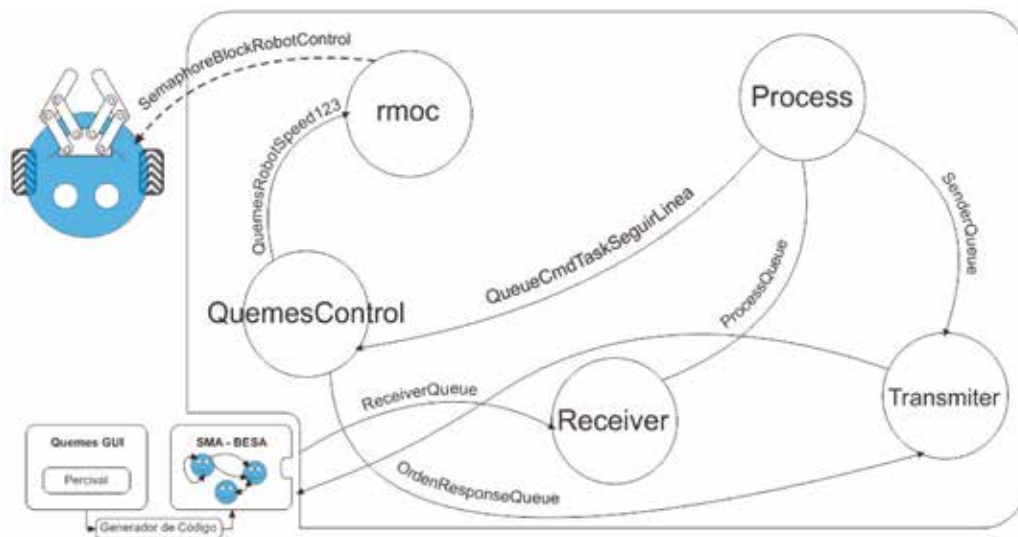
Las cinco tareas descritas permiten el funcionamiento del robot y se ejecutan de manera coordinada de acuerdo a la información presente en seis colas (queue). Estas son: *QueueRobotSpeed123*, *QueueCmdTaskSeguirLinea*, *receiverQueue*, *processQueue*, *senderQueue* y *orderResponseQueue*. *QueueRobotSpeed123* al recibir la información de las tres velocidades de los motores activa la tarea *rmoc* diseñada para enviar los valores de las velocidades al hardware de los motores. *QueueCmdTaskSeguirLinea* es el lugar donde se almacena el comando que llega de *Quemes*(PC) e indica el movimiento del robot o de la pinza. Si esto sucede activa la tarea *QuemesControl* que permite la navegación del robot, *receiverQueue* al recibir información sobre la presencia de información en el buffer del chip, activa la tarea *Receiver* que permite almacenar la trama completa que viene de *Quemes* (PC). *processQueue* al recibir la trama completa activa la tarea *Process* en donde se evalúa e identifica su contenido. *SenderQueue* al recibir la trama que ya ha sido validada en *Process*, permite la activación de la tarea *Transmitter*. *orderResponseQueue* es la cola

que recibe los mensajes asociados a la finalización de los diferentes estados del robot durante el proceso de navegación. Si se activa, permite que dichos mensajes sean enviados a Quemes (PC). Finalmente, está el semáforo SemaphoreBlockRobotControl diseñado para verificar la disponibilidad de enviar los datos de las velocidades hacia el hardware de los motores.

La figura 9 resume el funcionamiento del software embebido. Aquí, el trabajo de los robots inicia desde Quemes al enviar la trama con la información necesaria para el mo-

vimiento del robot. Esta información es recibida, validada y transmitida a la variable correspondiente dentro de la tarea QuemesControl, que tiene las funciones para la ejecución de los diferentes estados del robot (seguir línea, girar a la derecha, girar a la izquierda, parar, abrir pinza y cerrar pinza). Después de terminar algunos de estos estados el robot continúa recibiendo información hasta terminar la tarea. La necesidad del RTOS está en que el robot de manera paralela debe estar recibiendo información, ejecutando los movimientos y enviando la información de su estado a Quemes.

Figura 8. Principio de funcionamiento del programa RTOS de los robots



Fuente: elaboración propia

Al igual que la versión anterior, el software es desarrollado para un dsPIC 30f6010A, se programa en la plataforma MPLAB IDE v8.76 mediante el compilador MPLAB C30 y utiliza las librerías de FreeRTOS para el diseño general. El proyecto es llamado AgentCoop y contiene tres carpetas: Archivos_c compuesta de 11 archivos .c, Archivos_h compuesta de 6 archivos.h y FreeRTOS, en comparación

con la versión anterior con ocho carpetas, 28 archivos en C y ocho archivos .h. Se puede afirmar que la nueva versión está hecha a la medida. Es importante resaltar que el proceso de rediseño del software utiliza los fragmentos de código de las versiones anteriores que se consideraron útiles para la nueva versión de hardware de los robots, el código que principalmente se reutiliza es la programa-

ción del protocolo de comunicación entre el robot y *Quemes*, lo demás se ha modificado por completo.

C. Integración del Software de control a la pinza

Después de optimizar el programa se continúa con el plan de trabajo inicial para integrar la pinza a los robots. Los pasos necesarios consisten en diseñar en el software embebido dos estados y sus correspondientes rutinas para abrir y cerrar la pinza e integrar la pinza a los requerimientos de *Quemes*.

Para la integración de la pinza se agregan dos variables dentro de la tarea Process para identificar si el mensaje en la trama enviado desde *Quemes* contiene información para la apertura o cierre de la pinza; allí las variables están nombradas como ID_OPEN_GRIPPER y ID_CLOSE_GRIPPER. Si en la trama de llegada hay información para el control de la pinza, se coloca en las variables INDEX_ABRIR_PINZA e INDEX_CERRAR_PINZA según sea el caso y además, se envía a la cola (queue) QueueCmdTaskSeguirLinea para que se active la tarea QuemesControl en donde se encuentran las funciones ABRIR_PINZA_STATE y CERRAR_PINZA_STATE y se activarán según sea el caso.

Activar alguna de estas funciones significa asignar una velocidad al motor de la pinza y enviar esta información a la cola QueueRobotSpeed123 para activar la tarea rmoc que permite el movimiento en la pinza hasta que el sensor correspondiente de apertura o cierre de la pinza se active, al finalizar el movimiento se envía un mensaje a *Quemes* por medio de la tarea Process que lo recibe y lo integra a la trama de salida hacia *Quemes*, el cual está relacionado con la terminación de la acción de apertura o cierre de la pinza. Posteriormente *Quemes* utiliza esta informa-

ción para continuar con la coordinación de la tarea cooperativa de los robots.

D. Oportunidades de cooperación de los robots a través de la pinza

Se ha propuesto la ampliación de las actividades de cooperación por medio de la implementación en cada robot de un sistema prensil. Ahora hay más oportunidades para desarrollar actividades cooperativas en los robots, lo que incide de manera significativa en el desarrollo de actividades pedagógicas. El sistema prensil amplía las siguientes funciones:

- **Especialización:** a diferencia de la propuesta anterior de *Quemes*, esta nueva versión permite clasificar los dos grados de especialización: con o sin pinza. Este nuevo nivel permite el diseño de tareas más complejas y amplía las posibilidades de distribución de roles de acuerdo a las capacidades de los robots.
- **Comunicación:** a diferencia de la propuesta anterior de *Quemes*, al existir agentes con capacidades diferentes se establecen nuevos mensajes, dado que estos pueden estar en función de las capacidades de los robots.
- **Coordinación:** a diferencia de la propuesta anterior de *Quemes*, los estados de coordinación aumentan en tanto la nueva diversidad de roles permite establecer jerarquías de trabajo durante la solución de un problema.
- **Solución de conflictos:** en la versión anterior los conflictos se presentaban principalmente durante el uso de las vías. Ahora, con la integración de la pinza estos también se pueden presentar por el requerimiento de un objeto que se desea sujetar y trasladar.
- **Interacción:** La heterogeneidad de los robot con o sin pinza permite diseñar problemas de cooperación que requieran

diversos grados de participación de robots con diferentes roles durante la solución de una tarea.

4. Conclusiones

Se diseñó e implementó una pinza para los robots de la plataforma *Quemes*. De acuerdo a los antecedentes consultados el diseño de la pinza además de ser innovador, no altera estructuralmente el diseño de los robots, cumple con las condiciones de bajo costo y bajo consumo de energía en tanto solo utiliza un motor para su funcionamiento.

La integración de la pinza al sistema de control permite un funcionamiento eficaz para la sujeción de objetos geométricos simples, de tamaño proporcional a la pinza y con pesos menores de 100 gramos. El método de control se integró al sistema embebido propuesto dentro de la lógica de diseño del sistema original, aunque se reitera que se realizó un rediseño completo porque la versión inicial del RTOS estaba diseñada para una arquitectura electrónica de robots que ya no existían.

Se ampliaron las actividades educativas en una cartilla que mantiene los parámetros de diseño gráfico, pedagógico y didáctico establecidos por Maloka en la primera versión del proyecto. Las actividades están fundamentadas en la teoría constructivista, donde el aprendizaje es más significativo en un ambiente marcado por la manipulación y control de objetos, en este caso los robots y el entorno gráfico.

5. Referencias

- [1] Andrade, E. (1998). Ambientes de aprendizaje para la educación en tecnología. *Educación en Tecnología*, 72-91.
- [2] Bandura, & Gruce J.E, M. F. (1967). Some Social Determinats of Self-Monitoring Reinforcement Systems. *Journal of Personality and Social Psychology*.
- [3] Barak, M., & Zadok, Y. (2009). Robotics projects and learning concepts in science, technology and problem solving. *International Journal of Technology and Design Education*, 289'307.
- [4] Blank, D., Kummar, D., Marshall, J., & Meeden., L. (2007). *Advanced Robotics Projects for Undergraduate Students*. Computer Science Program Bryn Mawr College.
- [5] Botelho, S., & Alami, R. (1999). M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement. *Conference Robotic and Automation*.
- [6] Brigitte, D., & Sylviane, H. (2001). Collaborative learning in an educational robotics environment. *Computers in Human Behavior* 17, 465-480.
- [7] Brown, L. A., & Walker, W. H. (2008). Prologue: Archaeology, Animism and Non-Human Agents. *Journal of Archaeological Method and Theory*, 297-299.
- [8] Brush, T. (1997). The effects on Student Achievement and Attitudes when using integrated learning system with cooperative pairs. Pp 51-64 ISSN 1042-1629.
- [9] Buch, T. (1999). *Sistemas Tecnológicos, una mirada a la teoría general de la artificialidad*. Buenos Aires: Aique.
- [10] Bustacara, C., Roldan, F., Buitrago, S., & González, E. (2010). *QUEMES: Technology Education based on Cooperative Robots*. Pontificia Universidad Javeriana. -Bogotá, Colombia. Manuel Avellaneda, Laura Fajardo, Maloka-Bogotá.
- [11] Cao, Y. U., Fukunaga, A. S., & Kahng, A. B. (1997). *Cooperative Mobile Robotics: Antecedents and Directions*.
- [12] Chil, R., Alonso, D., Ortiz, F., & Pastor, J. (2011). Using Robotics as a Motivational Tool: An Ada Interface to a Pioneer

- Robot. Ada-Europe: Springer Berlin / Heidelberg.
- [13] Csicszentmihalyi, M. (1998). *Creatividad: El flujo de la psicología del pensamiento y la invención*. Paidós.
- [14] Denis, B., & Hubert, S. (2007). *Collaborative learning in an educational robotics environment*. Centre de Recherche sur l'Instrumentation en Formation.
- [15] Deval, J. (1975). *El animismo y el pensamiento infantil*. Madrid: Siglo Veintiuno Editores.
- [16] Dias, B., & Anthony, S. (2001). *A Market Approach to Multirobot Coordination*. Robotics Institute, Carnegie Mellon University., 1-26.
- [17] Dias, B., Zlot, R., Zinck, M., González, J. P., & Anthony, S. (2004). *A Versatile Implementation of the TraderBots Approach for Multirobot Coordination*.
- [18] Douglas, B., Deepak, K., James, M., & Lisa, M. (2007). *Advanced Robotics Projects for Undergraduate Students*. Computer Science Program Bryn Mawr College.
- [19] Dudek, G., Jenkin, M. R., Milios, E., & Wilkes, D. (1997). *A Taxonomy for Multi-Agent Robotics*. Kluwer Academic Publishers, Boston. *Manufactured in The Netherlands.*, 5-29.
- [20] Eric, Z., Feng, L., & Chun, H. L. (2010). *Student satisfaction and self-efficacy in a cooperative robotics course*. Tamkang University, Taipei, Taiwan, ROC.
- [21] Farinell, A., Iocchi, L., & Nardi, D. (2007). *Multi-Robot Systems: A classification focused on coordination*. ICRA 2007: 3850-3855.
- [22] Ferber, J. (1995). *Multi-Agent Systems, an introduction to distributed artificial intelligence*. Paris: Addison Wesley.
- [23] Gerkey, B. P., & Mataric, M. J. (2004). *A formal analysis and taxonomy of task allocation in multi-robot systems*.
- [24] Goddard, R. (2000). *Collective teacher efficacy: Its meaning measure, and impact on student achievement*. University of Michigan. Wayne K. Hoy and Woolfolk Hoy. The Ohio State University.
- [25] Golfarelli, M., Maio, D., & Rizzi, S. (1997). *A Task-Swap Negotiation Protocol Based on the Contract Net Paradigm*.
- [26] González, E. (2010). *QUEMES: Technology Education based on Cooperative Robots*. Pontificia Universidad Javeriana.
- [27] González, E., Avila, J., & Bustacara, C. (2003). *BESA Behavior oriented Event driven and Social based Agent Framework*.
- [28] Johnson, & Johnson. (1990). *Using cooperative learning in mathematics. Cooperative learning in mathematics: A handbook for teachers*. New York: Addison-Wesley., Pág. 103-125.
- [29] Johnson, D., & Johnson, R. (1999). *Aprender juntos y solos*. Buenos Aires: Aique.
- [30] Kayako, N., Masahiro, S., Kazuhiko, S., Reo, M., Hiroshi, I., & Norihiro, H. (2011). *Effect of Robot's Whispering Behavior on People's Motivation*. Kyoto, Japan: Springer.
- [31] Layton, D. (1993). *Technology's Challenge to Science Education*. Open University Press, Buckingham, UK.
- [32] Liu, E. Z., & Lin, C. H. (2010). *Student satisfaction and self-efficacy in a cooperative robotics course*. National Central University, Chung-Li, Taiwan, ROC, Chiung Sui Chang, Tamkang University, Taipei, Taiwan, ROC.
- [33] Mansi, G., Marwa, M., & Shikha, P. (2010). *Robots Byte In: An exploration of computer Science education in Middle Schools*.
- [34] Marian, P., & Blaine, P. (2004). *Using Robotics to Motivate 'Back Door' Learning*. *Education and Information Technologies*, 147-158.

- [35] Michael, B., Brett, B., & Manuela, V. (s.f.). Plays as Effective Multiagent Plans Enabling Opponent-Adaptive Play Section.
- [36] Mitnik, R., Recabarren, M., Nussbaum, M., & Soto, A. (2010). Department of Computer Science, School of Engineering.
- [37] Mitnik, R., Recabarren, M., Nussbaum, M., & Soto, A. (2010). Collaborative robotic instruction: A graph teaching experience. Department of Computer Science, School of Engineering, Pontificia Universidad Católica de Chile.
- [38] Mumford, L. (1994). Técnica y Civilización. Madrid: Alianza Editorial.
- [39] Nkambou, R., Bourdeau, J., & Mizoguchi, R. (2010). Advances in Intelligent Tutoring Systems. Springer-Verlag Berlin Heidelberg, SCI 308, pp. 1-12.
- [40] Orazio, M., Onofrio, G., Michela, P., & Stefano, N. (2007). Breedbot: An Edutainment Robotics System to Link Digital and Real World. Springer-Verlag Berlin H, 74-81.
- [41] Parker, L. E. (1997). Task-Oriented Multi-Robot Learning in Behavior-Based Systems. 305-322.
- [42] Parker, L. E. (2008). ALLIANCE: An Architecture for Fault Tolerant, Cooperative Control of Heterogeneous Mobile Robots. Center for Engineering Systems Advanced Research Oak Ridge National Laboratory.
- [43] Parker, L. E. (2003). Current research in multirobots systems.
- [44] Pavel, P., & Richard, B. (2010). Educational Robotics Initiatives in Slovakia. Department of Applied Informatics, Faculty of Mathematics, Physics and Informatics, Comenius University.
- [45] Roncone, K. (2002). Toy story: Materials engineering at play. JOM Journal of the Minerals, Metals and Materials Society, 16-19.
- [46] Ruiz, V. S. (1996). Ciencia y tecnología a través de la robótica cognoscitiva. Perfiles educativos - Universidad Nacional Autónoma de México, Abril - Junio, número 72.
- [47] Ryan, R., & Edward, D. (2000). La Teoría de la Autodeterminación y la Facilitación de la Motivación Intrínseca, el Desarrollo Social, y el Bienestar. American Psychologist, 68-78.
- [48] Scalzone, F., & Tamburrini, G. (2012). Human-robot interaction and psychoanalysis. AI & Society.