

Sistema Multi-robot para detección automática de “fallas” en un circuito utilizando redes bayesianas

Irvin Hussein López Nava,

Estudiante de la Maestría en Ciencias del
Instituto Nacional de Astrofísica, Óptica y Electrónica
hussein@ccc.inaoep.mx

Resumen. En el presente reporte se presentan los resultados obtenidos en el proyecto: Sistema multi-robot para detección automática de “fallas” en un circuito. Este sistema multi-robot está compuesto de tres robots móviles homogéneos y un servidor externo. El objetivo principal del proyecto es verificar ocho puntos de seguridad por al menos dos robots y verificar si existe una falla, el tercer robot solo inspeccionará el punto si los dictámenes de los otros robots son diferentes. Los valores sensados en cada punto de seguridad serán enviados al servidor para crear una red bayesiana y determinar la ruta de cada miembro del sistema multi-robot.

1 Introducción

En el área de la robótica existen problemas que pueden ser resueltos por un solo robot, sin embargo también existen tareas que necesitan de dos o más robots para cumplir un objetivo en común. Para ello es necesario crear sistemas multi-robot en los cuales cada miembro tiene bien definido su rol dentro del mismo.

La tarea a realizar por este sistema multi-robot es la inspección automática de “fallas” en un circuito, el cual tiene un conjunto de puntos de seguridad a inspeccionar. Los robots envían al servidor mensajes para informarle si hay o no falla, el servidor procesa la información, crea una red bayesiana para cada punto de seguridad con base en los valores crudos sensados y propaga las rutas de puntos a seguir por cada robot. La ruta recibida desde el servidor debe seguirse por cada robot para inspeccionar los puntos de seguridad indicados, y debe modificarse en función de los resultados obtenidos en la revisión de cada punto por los robots.

Los puntos de seguridad consisten en objetos capaces de ser detectados por los sensores de los robots. Se utilizaron diversos papeles en niveles de gris como puntos de seguridad para determinar si existe o no falla.

Los sensores utilizados por cada miembro del sistema multi-robot son dos fotoceldas (sensores de luz).

El circuito es un escenario preparado en el laboratorio de robótica de la Coordinación de Ciencias Computacionales en el Instituto Nacional de Astrofísica, Óptica y Electrónica. Este escenario se detalla en el Apéndice A.

El esquema a investigar es el siguiente:

- Circuito cerrado de un carril y un solo sentido de circulación.
- Cada punto de seguridad debe ser revisado al menos por dos robots.
- Cada robot es capaz de acceder a cada punto y de sensarlo.
- Cada punto de seguridad en discordia, i.e. con dos dictámenes diferentes, debe ser revisado por un tercer robot, quien confirmará uno u otro dictamen.

Cuando un robot se encuentre frente a cada punto de seguridad debe inspeccionar con su conjunto de sensores si existe o no “falla”. Debido a que las lecturas obtenidas de los sensores no son precisas, se estimarán las probabilidades calculadas por cada sensor y por cada robot y se creará una red bayesiana para su resolución. Debido a que cada fotocelda mide un rango de acción diferente, se caracterizará cada sensor en un ambiente controlado y se creará una tabla de datos con los valores obtenidos y con base en esta información se determinará el intervalo de falla para cada sensor.

2 Trabajo relacionado

Una revisión de los conceptos básicos de sistemas de robots móviles distribuidos así como de trabajos relacionados se encuentra descrita en Parker (2000). En Stroupe et Balch (2002) proponen la realización de tareas o misiones, donde un robot selecciona una tarea a ejecutar y debe evaluar la prioridad de todas las tareas existentes, además de las necesidades, capacidades y las tareas que los demás miembros del sistema han seleccionado. En Parker et al. (2002) proponen que inicialmente cada robot planifique de manera independiente una ruta a seguir. Las rutas son compartidas entre todos los miembros del sistema. De esta manera se tiene una visión global de las rutas que cada uno de los robots recorrerá para completar una tarea. En Méndez (2008) se propone un sistema multi-robot el cual realiza una tarea en común con robots homogéneos como los utilizados en este proyecto.

3 Metodología y desarrollo

Para el sistema multi-robot se utilizarán 3 robots compuestos por tarjetas HandyBoard®, pocket's PC HP iPAQ®, una laptop que participará como servidor, un router inalámbrico para crear una red local, un par de sensores (fotoceldas) para cada robot, cable serial/datos para comunicación entre la HandyBoard® y la iPAQ®, y efectores y actuadores para esta plataforma (figura 1).



Figura 1. Miembros del sistema Multi-robot

El sistema multi-robot se expresa como:

$$SMR = (G_R)$$

donde:

$$GR = \{R_1, R_2, \dots, R_n\} \text{ y } R_i = \{Ca, Co\}$$

y para los cuales:

R_i denota a cada uno de robots miembros el sistema.

Ca denota el conjunto de habilidades individuales para todo robot del sistema: avanzar, retroceder, parar, girar a la izquierda y a la derecha, detección del camino, recibir y enviar información al nivel superior.

Co denota el conjunto de comportamientos que un robot miembro del sistema puede ejecutar de manera individual o colectiva, aplicando elementos de **Ca** y otros **Co**: recibir y enviar información al nivel inferior, recibir y enviar información a los miembros del sistema multi-robot, codificar y decodificar mensajes, decidir acción.

El conjunto de habilidades **Ca** (bajo nivel) son programadas en lenguaje C sobre la tarjeta HandyBoard©, mientras que el conjunto **Co** (alto nivel) son programadas en lenguaje Java sobre la pocketPC HP iPAQ© misma que en la cual se crea un estructura de tipo árbol para usarse como red bayesiana y almacenar las probabilidades.

En la Figura 2 se presentan los módulos de alto y bajo nivel así como el flujo de comunicación entre estos módulos y los distintos dispositivos del robot.

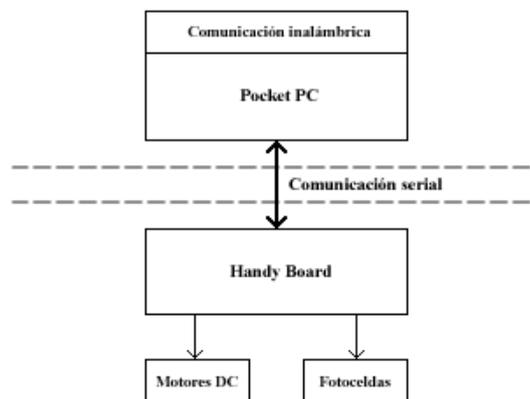


Figura 2. Organización interna de los robots

Como puntos de seguridad se utilizaron 8 marcas de papel en escala de grises, se asume que las tres primeras marcas no presentan falla, al sensar dichas marcas se tienen los valores más altos (figura 3).

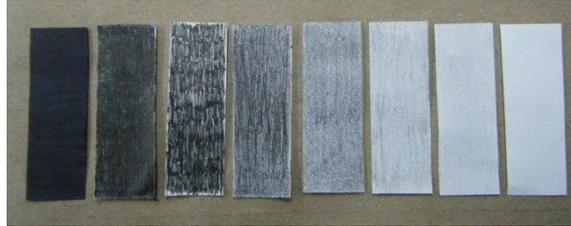


Figura 3. Puntos de seguridad

Representación (Red Bayesiana)

Para cada punto de seguridad se crea una red bayesiana (figura 4). Donde la probabilidad de cada punto de seguridad $\{PP1, PP2, \dots, PPn\}$ está determinada por la probabilidad de cada uno de los robots $\{PR1, PR2, \dots, PRn\}$ a su vez determinadas por la probabilidad de cada sensor $\{PS1, PS2, \dots, PSn\}$.

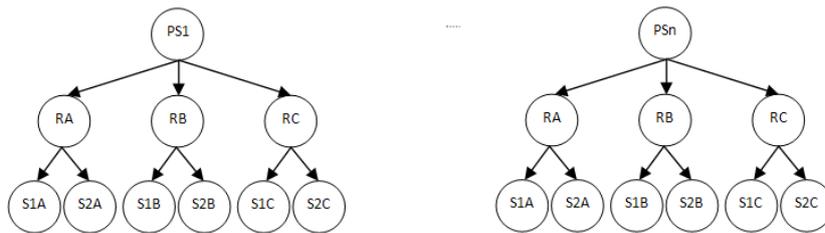


Figura 4. Red bayesiana para cada punto de seguridad

Para implementar el modelo probabilista para los experimentos se utilizó la herramienta Hugin© (figura 5), en dicho modelo se llenaron las tablas de probabilidad conjunta con base en las lecturas de 5 pruebas para determinar el intervalo de falla.

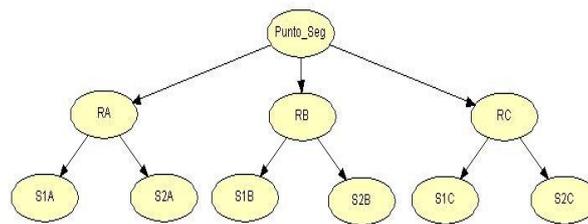


Figura 5. Red implementada por Hugin

El modelo anterior se implementó en el servidor de tal modo que cada robot al obtener sus lecturas le envían los datos al servidor quién se encarga de calcular las probabilidades de cada nodo de la red y de modificar la ruta de cada miembro del sistema, dichas rutas son enviadas a todos los robots.

Para obtener las probabilidades de las tablas conjuntas para cada variable se realizó un análisis de los datos obtenidos en las pruebas y al tener valores continuos se crearon intervalos para determinar los valores por cada sensor debido a que cada uno tiene un rango de acción diferente. Las tablas se muestran en la figura 6.

RA	Falla	No falla	RA	Falla	No falla
<30	1	0	<25	1	0
31-40	0.8	0.2	26-35	0.8	0.2
41-60	0.6	0.4	36-45	0.6	0.4
61-80	0.4	0.6	46-55	0.4	0.6
81-90	0.2	0.8	56-65	0.2	0.8
>91	0	1	>66	0	1

Lecturas del Robot A. Sensor 1 y Sensor2

RB	Falla	No falla	RB	Falla	No falla
<10	1	0	<15	1	0
11-15	0.8	0.2	15-20	0.8	0.2
16-20	0.6	0.4	21-25	0.6	0.4
21-25	0.4	0.6	26-35	0.4	0.6
26-30	0.2	0.8	36-45	0.2	0.8
>31	0	1	>46	0	1

Lecturas del Robot B. Sensor 1 y Sensor2

RC	Falla	No falla	RC	Falla	No falla
<40	1	0	<25	1	0
41-60	0.8	0.2	26-40	0.8	0.2
61-80	0.6	0.4	41-55	0.6	0.4
81-100	0.4	0.6	56-70	0.4	0.6
100-110	0.2	0.8	71-80	0.2	0.8
>111	0	1	>81	0	1

Lecturas del Robot C. Sensor 1 y Sensor2

Figura 6. Tablas de probabilidad conjunta de cada sensor según el robot.

Para las tablas de probabilidad de cada robot dado los puntos de seguridad se estableció una probabilidad de 0.5. Para la tabla del punto de seguridad también se estableció el valor 0.5.

4 Experimentos y resultados

Para la evaluación de las fallas se tomará como intervalo el valor mínimo de 0 y máximo de 0.53, ya completado el modelo probabilista se realizaron 3 experimentos para comprobar si es adecuado y los resultados obtenidos fueron buenos (figura 7). Los valores de esta tabla se expresan en los valores nominales SI y NO para las fallas.

Exp.	Robot	PP1	PP2	PP3	PP4	PP5	PP6	PP7	PP8								
1	Marca	2	1	5	7	4	6	8	3								
	RA	0.98	No	1.00	No	0.33	Si	0.04	Si	0.54	No	0.09	Si	0.02	Si	0.59	No
	RB	0.82	No	1.01	No	0.28	Si	0.08	Si	0.45	Si	0.18	Si	0.01	Si	0.53	Si
	RC									0.59	No					0.60	No
2	Marca	6	3	1	4	2	8	5	7								
	RA	0.08	Si	0.56	No	1.01	No	0.54	No	0.92	No	0.02	Si	0.28	Si	0.08	Si
	RB	0.12	Si	0.52	Si	0.98	No	0.51	Si	0.86	No	0.00	Si	0.25	Si	0.06	Si
	RC			0.56	No			0.52	No								
3	Marca	2	7	4	1	6	8	5	3								
	RA	0.94	No	0.10	Si	0.55	No	1.0	No	0.10	Si	0.03	Si	0.30	Si	0.57	No
	RB	0.85	No	0.08	Si	0.53	Si	0.97	No	0.11	Si	0.01	Si	0.27	Si	0.53	Si
	RC				0.49	Si										0.55	No

Figura 7. Resultados obtenidos en 3 experimentos

Para analizar esta última tabla es necesario recordar que las marcas 1, 2 y 3 no presentan falla. En cada uno de los 3 experimentos las marcas fueron cambiadas de lugar para observar el comportamiento del sistema multi-robot.

En la siguiente figura se ilustran las redes bayesianas creadas por el experimento número 1 (figura 8). Los datos utilizados están indicados en la figura 7. Cuando un punto de seguridad solo tiene las probabilidades de solo dos robots se observa que no fue necesario que el robot C verificara dicho punto de seguridad.

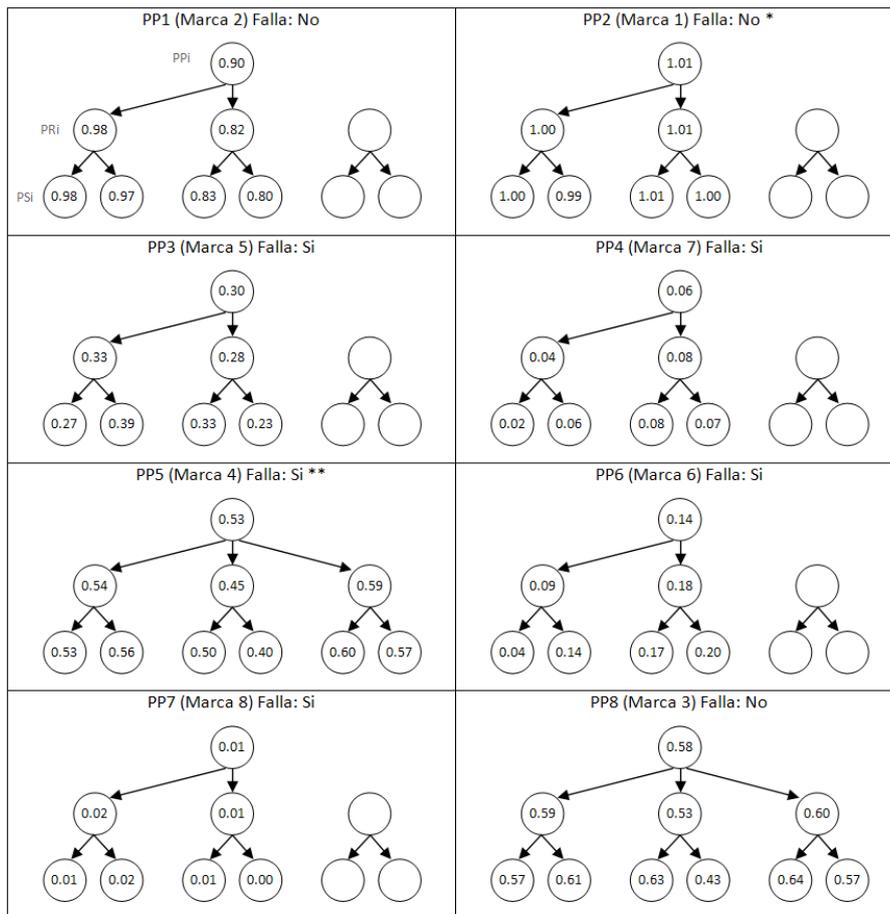


Figura 8. Redes bayesianas creadas en el experimento 1

* En este punto de seguridad observamos que la probabilidad es más que uno, esto es debido a que la lectura de los sensores estuvieron por encima del umbral estimado en las pruebas. Esto ocurrió en los experimentos 1 y 2.

** En este punto de seguridad se tiene un valor falso negativo, es decir, en realidad la marca es de falla, sin embargo el robot C que tendría que confirmar que si la había, no lo hizo. Esto solo ocurrió en el experimento 1. Pese a este dictamen, se determina que hay falla por la probabilidad calculada con los valores de los 3 robots.

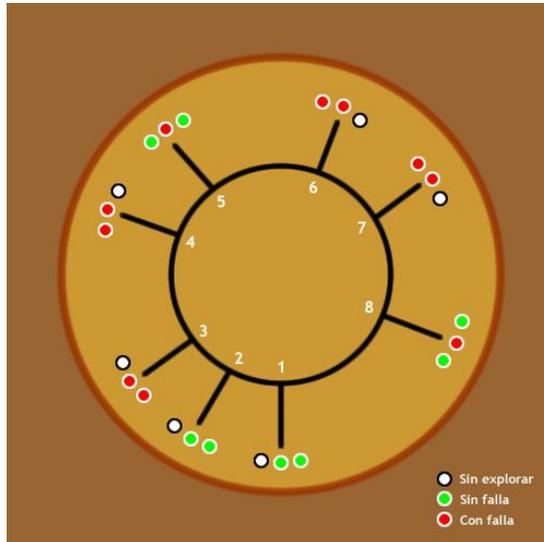


Figura 9. Representación gráfica del experimento 1

Al enviar los datos de cada robot hacia el servidor, éste crea una red bayesiana y actualiza el estado de cada punto de seguridad, para el experimento número 1 se ilustra el estado de todos los puntos al finalizar la inspección (Figura 9). Los puntos 1 y 2 no presentan falla. Los puntos 3 y el 4 si la detectan. En el punto 5 el robot A no detecta falla, el B si, entonces el robot C realiza su inspección y determina que no hay falla, sin embargo la red probabilística para ese punto determina que si hay falla, lo que es correcto. Los puntos 6 y 7 si presentan falla. Por último para el punto 8, el robot A determina que no hay falla, el robot C que si, entonces el robot C realiza su inspección y determina que no hay falla, lo que es correcto.

5 Conclusiones y trabajo a futuro

La resolución colectiva de problemas es un área que falta por explotar, me parece interesante debido a que la mayoría de los robots trabajan en ambientes controlados. Si el problema requiere lo contrario entonces nos encontramos con un factor extra a considerar, el ambiente, el cual altera de forma importante los valores de los sensores y afecta la actuación del robot. Para ello es necesario utilizar en conjunto un sistema multi-robot y modelos probabilistas.

El siguiente esquema a investigar es:

- Circuito cerrado de dos carriles y un solo sentido de circulación.
- Cada punto de seguridad debe ser revisado al menos por dos robots.
- Cada robot es capaz de acceder a cada punto, rebasando por el carril "interno" a otros robots de ser necesario.
- Cada punto de seguridad en discordia, i.e. con dos dictámenes diferentes, debe ser revisado por un tercer robot, quien confirmará uno u otro dictamen.

5 Referencias

1. Parker, L. E. Current state of the art in distributed autonomous mobile robotics. En *Distributed Autonomous Robotic Systems*, volume 4, pages 3–12. Springer, Tokio (2000)
2. Stroupe A. et Balch T. Mission-relevant collaborative observation and localization. En *Multi-Robot Systems: From Swarms to Intelligent Automata, Proceedings from the 2002 NRL Workshop on Multi-Robot Systems*, pp. 31 – 40. Kluwer Academic Publishers (2002)
3. Parker L. E., Fregene K., Guo Y., et Madhavan R. Distributed heterogeneous sensing for outdoor multi-robot localization, mapping, and path planning. En *Proceedings of the 2002 NRL Workshop on Multi-Robot Systems*, pp. 21–30. Kluwer Academic Publishers (2002)
4. José Alberto Méndez-Polanco, Angélica Muñoz-Meléndez. *Collaborative Robots for Indoor Environment Exploration*. En *Proceedings of the 10th International Conference on Control, Automation, Robotics and Vision, ICARCV 2008*.

Apéndice A: Circuito de Navegación

El ambiente preparado consiste en un espacio de 2 x 2 metros por lado en el piso forrado con papel café, dentro de este espacio se utilizó una pared de 25 cm. de alto para formar un círculo con 90 cm. de diámetro. Para la navegación de los robots se adhirió una línea negra al circuito para formar el carril en forma circular con un diámetro de 60 cm. Los puntos de seguridad están ubicados a distancias no equidistantes ubicadas al exterior de la línea de navegación y al interior de la pared de modo que para la inspección los robots móviles deben de salir del carril (figura4). Los experimentos se realizarán bajo un ambiente controlado, en un salón cerrado, de noche y con luz blanca (Figura 10).

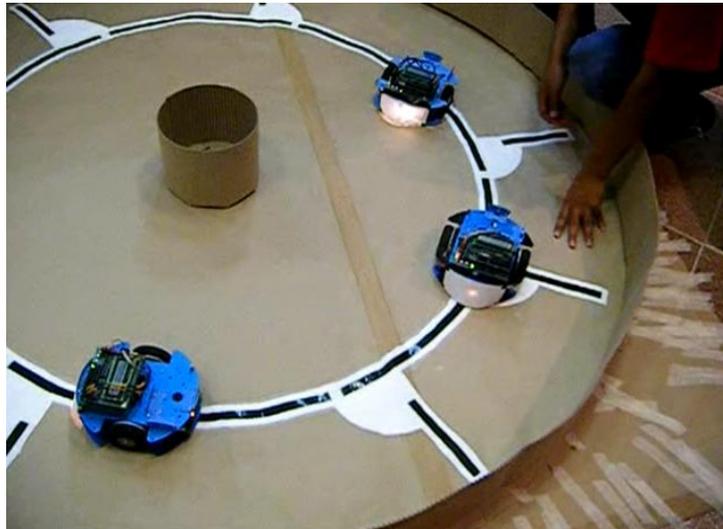


Figura 10. Circuito implementado