

Planificación de caminos y trayectorias de una formación de robots autónomos

Motion planning for a formation of autonomous robots

Presentación: 6-7/10/2020

Doctorando:

Maira Estefanía Pereyra

Centro de Investigación en Informática para la Ingeniería (CIII), Facultad Regional Córdoba, Universidad Tecnológica Nacional – Argentina.
mepereyra@frc.utn.edu.ar

Director:

R. Gastón Araguás

Co-directores:

Gonzalo Pérez Paina, Miroslav Kulich

Resumen

En este trabajo se presentan los últimos resultados en investigación en el marco de la tesis *Planificación de caminos y trayectorias de una formación de robots autónomos*. La planificación de camino consiste en encontrar un camino continuo que lleve a un robot desde su posición inicial a su destino, evitando colisiones con su entorno. Basado en el método de exploración del Algoritmo Genético Simple (AGS) y del algoritmo de búsqueda clásico A*, se propone un algoritmo híbrido de planificación de caminos para sistemas multirobots. Este algoritmo encuentra los caminos óptimos (o casi-óptimos) libres de colisiones de una formación de robots desde una configuración inicial a una configuración final teniendo en cuenta la posibilidad de separación y su posterior reunión. Los resultados de las simulaciones demuestran que el algoritmo es apto para ser utilizado en entornos complejos con obstáculos y para un sistema multirobots.

Palabras clave: Algoritmo genético; algoritmo A* ; planificador de caminos; operadores evolutivos; robot móvil; diagrama de Voronoi; grafos de transición de estados; multirobots.

Abstract

In this work the latest research results are presented within the framework of the thesis *Motion planning for a formation of autonomous robots*. Path planning is the task of finding a continuous path that will drive the robot from the start to the goal location avoiding collisions with obstacles. Based on the Simple Genetic Algorithm (AGS) exploration method and the classic A* search algorithm, a hybrid path planning algorithm for multirobots systems is proposed. This algorithm finds the optimal (or near-optimal) collision-free paths of a robot formation from an initial configuration to a final configuration, taking into account the possibility of separation and their subsequent reunification (split&merge). The results of the simulations show that the algorithm is suitable for use in complex environments with obstacles and for a multirobots system.

Keywords: Genetic Algorithm; A* algorithm; path planning; evolutive operators; mobile robot; Voronoi diagram; transition states graph; multirobots.

Introducción

La planificación de caminos puede considerarse uno de los pilares fundamentales en la navegación autónoma de robots, junto con la percepción, localización y generación de trayectorias. En robótica la planificación de caminos consiste en encontrar un conjunto de caminos libre de colisión a través de un entorno con obstáculos, desde una ubicación inicial a un destino final deseado. Los caminos se obtienen optimizando algún parámetro específico (distancias a recorrer, energía, tiempo de operación, etc.) y aplicando luego restricciones respecto del tamaño y/o geometría de la formación. Dependiendo del conocimiento que tiene el robot sobre su entorno la planificación puede ser clasificada en dos categorías: planificación global, si el robot tiene un total conocimiento del entorno (Pereyra, Gonzalez Dondo & Araguás, 2018) o planificación local, si el robot conoce solo una parte del mismo (Pereyra, Gaydou & Araguás, 2017). Además ese entorno puede ser estático o dinámico. Los métodos de planificación global de caminos se dividen principalmente en dos categorías: métodos clásicos y métodos heurísticos. Los métodos clásicos son aquellos basados en Roadmap (RM), campos potenciales, descomposición en celdas entre otros. En estos métodos primero se modela el espacio de trabajo como una red o grafo de posibles caminos, uniendo los puntos inicial y final a dicho grafo, para después aplicar un algoritmo de búsqueda como A* (Hart & Nilsson, 1968), algoritmo de Dijkstra (Dijkstra, 1959) u otros. Este trabajo de tesis se enfoca en el problema de la planificación de caminos y trayectorias de múltiples robots operando en formación, con la capacidad de romper dicha formación y volver a agruparse posteriormente, lo que se conoce como split&merge. Dentro de esta temática se plantea investigar la extensión de los algoritmos existentes de planificación de caminos para un único robot tanto de tierra (tres grados de libertad) como aéreos (seis grados de libertad), a algoritmos aplicados a sistemas multi-robots, manteniendo sus criterios de completitud y optimalidad. Basado en métodos clásicos se proponen dos algoritmos de planificación global de caminos para sistemas multirobot en entornos estáticos. El primero de los algoritmos propuestos es una extensión del algoritmo de Dijkstra, el cual explora de forma exhaustiva un grafo de transición de estados y devuelve los caminos óptimos de cada robot en la formación desde su nodo de partida a cada posible nodo destino del grafo (Pereyra, Araguás & Kulich, 2017). El segundo es un algoritmo que, basado en la misma técnica que el anterior, determina cada uno de los caminos en forma secuencial (Pereyra, Araguás & Kulich, 2017b). Estos métodos clásicos son exactos, efectivos y determinísticos, aunque presentan ciertas debilidades en entornos muy complejos (Pereyra, Araguás & Kulich, 2017), (Raja, 2012). Debido a estas debilidades se plantea el uso de métodos heurísticos para resolver el problema de planificación de caminos. Entre los métodos heurísticos se incluyen algoritmos como optimización de colonia de hormigas (ACO, ant colony optimization) (Asl, Menhaj & Sajedin, 2014), algoritmos genéticos (AG) (Qu & Xing, 2013) y redes neuronales (NN, neural network) (Yang & Meng, 2000), entre otros. Si bien los métodos heurísticos son aproximados, proveen formas inteligentes de explorar el espacio de búsqueda, lo cual disminuye significativamente el tiempo de cómputo. Una de las técnicas que sobresalen sobre otras en la planificación de caminos son las basadas en algoritmos genéticos, debido a su capacidad de explorar el espacio de búsqueda preservando la mejor solución ya encontrada. En los últimos años muchos planificadores de caminos basados en AG han sido implementados (Hsu & Liu, 2014), (Alnasser & Bennaceur, 2016). Una de las limitaciones presente en estos trabajos es que todos comienzan con una solución inicial aleatoria, o adoptan un método aleatorio para generar una solución inicial válida. Este enfoque no es eficiente con respecto al tiempo y al costo computacional. Además utilizan operadores evolutivos comunes o convencionales, que aunque su rendimiento ha sido probado en numerosos problemas, la eficiencia del algoritmo evolutivo puede mejorarse considerablemente mediante el uso de operadores diseñados específicamente para el problema planteado y con un método de inicialización efectivo (Lee & Kim, 2016), (Lamini, Benhlima & Elbekri, 2018), (Nazarahari, Khanmirza, & Doostie, 2019). En base a lo expuesto se propone la implementación de un algoritmo híbrido de planificación de caminos para sistemas multirobots el cual utiliza como método de exploración el Algoritmo Genético Simple (AGS) en combinación con distintos algoritmos de búsqueda de caminos (Pereyra, Pucheta & Araguás, 2019), (Pereyra, Pérez Paina & Araguás, 2019).

Desarrollo

Planificador de caminos para múltiples robots en entornos estáticos

Un planificador de caminos para múltiples robots (MPP, multirobot path planning) debe encontrar el conjunto de caminos libre de colisiones, óptimo (o casi-óptimo) en términos de algún criterio en particular, desde una configuración inicial hasta una configuración deseada pasando por configuraciones intermedias en un entorno con obstáculos. La formación de robots puede separarse y reunirse en cada punto de bifurcación. Los robots pueden llegar a un punto de unión en diferentes tiempos. Para analizar el peor escenario aquí se considera que los robots llegan a un punto de unión en el mismo instante. Consecuentemente, el costo de la solución completa queda determinado por el costo del peor camino del conjunto de caminos individuales. La búsqueda de los caminos óptimos se realiza mediante un método de planificación de caminos híbrido, descrito en el Algoritmo 1, que combina técnicas de búsqueda clásicas y heurísticas basadas en

Algoritmo 1: MPP evolutivo

1. Modelado del entorno mediante grafo conectado $G(V,E)$
 2. Inicialización de la población H
 3. Subdivisión de población H_N
 4. Búsqueda del conjunto óptimo de caminos P_N en $G(V,E)$ desde el nodo $Q_{INI} \in V$ hasta el nodo $Q_{GOAL} \in V$ para cada robot $r_i \in R$ aplicando el algoritmo evolutivos sobre H_N
 5. Generación de la población

$$H_0 = \{P_1, P_2, \dots, P_{N-1}, P_N\}$$
 6. Búsqueda del conjunto óptimo de caminos P_0 en $G(V,E)$ desde en nodo $Q_{INI} \in V$ hasta el nodo $Q_{GOAL} \in V$ para cada robot $r_i \in R$ aplicando el algoritmo evolutivos sobre H_0
-

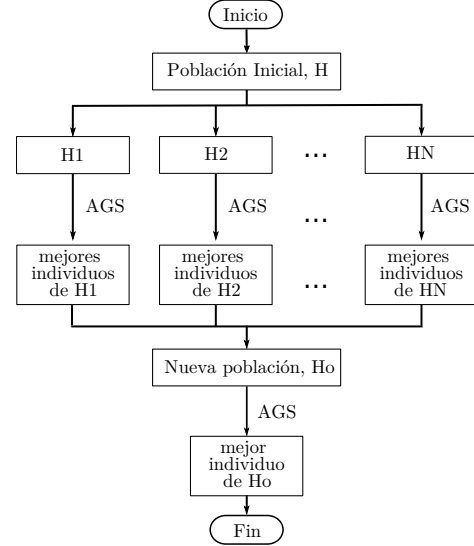


Figura 1: Estrategia evolutiva.

algoritmos genéticos. Dado un grafo representativo del entorno, un algoritmo secuencial similar al publicado en (Pereyra, Araguás & Kulich, 2017b) pero basado en A^* genera el conjunto de caminos entre las configuraciones inicial y final que conforman la población inicial H (líneas 1 y 2). Luego, esta población es subdividida en pequeñas poblaciones H_N sobre las que se realiza una búsqueda de caminos mediante la aplicación del algoritmo genético descrito en (Pereyra, Pérez Paina, Pucheta & Araguás) (líneas 3 y 4). De cada subpoblación se obtienen los mejores individuos. La unión de estos individuos forman luego la población inicial H_0 (línea 5) sobre la que se aplica el algoritmo genético configurado para la etapa final de búsqueda (línea 6). Al finalizar la evolución se obtienen los caminos óptimos (o casi-óptimos) para cada robot de la formación. Cabe mencionar que cada etapa evolutiva se configura de una forma específica. La primera etapa evolutiva aplicada a las subpoblaciones requiere una configuración de parámetros evolutivos distinta a la etapa evolutiva final (Figura 1).

Modelado del entorno y propiedades de los caminos

El modelado del entorno consiste en una representación mediante un grafo ponderado como se muestra en la Figura 2 (izquierda). Este grafo se obtiene a partir del diagrama de Voronoi de los obstáculos del entorno. Las aristas del grafo forman los caminos libres de colisiones por donde pueden circular los robots y cada una de ellas tiene asociado un valor que representa el costo (tiempo, energía, etc.) de ser atravesada por un robot. Para el caso de múltiples robots el costo de la arista puede ser variable, dependiendo de la cantidad de robots que la atraviesan. En este trabajo el costo representa el tiempo necesario para atravesar la arista considerando velocidades constantes de movimiento. El valor representativo de este tiempo puede estimarse mediante algún algoritmo de planificación de movimientos local (Pereyra, Gaydou & Araguás, 2017), o mediante una función de costo específica. Aquí se utiliza una función de costo que considera la longitud y ancho del corredor (arista), y la cantidad de robots que lo atraviesan, representado mediante un vector de costos $c = \{c_1, c_2, \dots, c_R\}$, donde c_r corresponde al costo de atravesar con r robots dicha arista. Sobre el grafo con costos $G(V,E)$ se realiza posteriormente la búsqueda de los R caminos mediante el planificador evolutivo. Cada uno de estos caminos es una secuencia de vértices de forma $p = \{v_0 = Q_{ini}, v_1, \dots, v_n = Q_{goal}\}$ que cumple con las propiedades y restricciones mencionadas a continuación.

Sea $G(V,E)$ el grafo conectado del entorno, con vértices $V = \{v_1, v_2, \dots, v_i\}$ y aristas $E = \{e_1, e_2, \dots, e_j\}$ y una flota de robots $R = \{r_1, r_2, \dots, r_R\}$, el conjunto de caminos que guiará a los robots por el mapa se denota $P = \{p_1, p_2, \dots, p_R\}$, con $p_i: Z \rightarrow V$. El camino de un robot r_i es una secuencia de vértices $p_i = \{v_{i1}, v_{i2}, \dots, v_{ik}\}$ tal que (v_{ij}, v_{ij+1}) es una arista del grafo.

- En un inicio todos los robots de la formación se encuentran en la posición inicial, esto es $p_i(0) = Q_{ini}, \forall p_i \in P$.
- Para cada camino existe un estado $k_{min} \in Z$ tal que $p_i(k_{min}) = Q_{goal}$, donde el robot r_i alcanzó su objetivo utilizando el camino de menor costo.
- Dos caminos de P no colisionan (dados dos estados intermedios cualquiera $m, l \in \langle 0, k_{min} \rangle$ los caminos p_i, p_j se encuentran en colisión si $(p_i(m), p_i(m+1)) = (p_j(l+1), p_j(l))$).

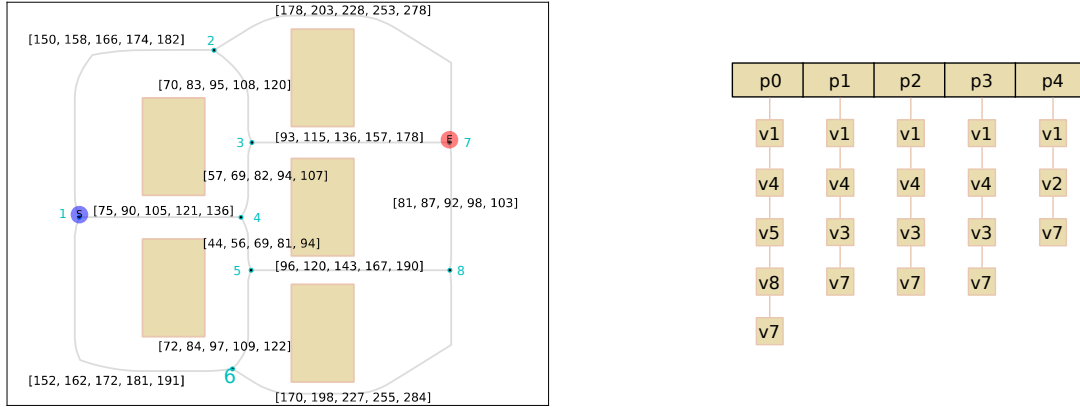


Figura 2: Representación del entorno e individuo.

- Sean dos caminos p_i, p_j y dos estados $m, l \in \langle 0, k_{\min} \rangle$ si $p_i(m) = p_j(l)$ entonces $m = l \leftarrow \max(m, l)$ lo cual significa que el primer robot en alcanzar un vértice compartido toma el costo del segundo, considerando el peor de los casos donde ambos robots llegan al vértice al mismo tiempo.

Algoritmo genético propuesto

El método de búsqueda de múltiples caminos utiliza como base el AGS con algunas modificaciones. La representación de los cromosomas, los operadores evolutivos de cruzamiento y mutación son modificados respecto a los del AGS teniendo presente la representación de los caminos detallada en la sección anterior. Los detalles son:

- Inicialización:** La población inicial a partir de la cual el algoritmo genético evoluciona depende de la etapa en la que este se aplica. En la primera etapa evolutiva, se generan subpoblaciones tomando de forma aleatoria un conjunto de posibles soluciones de una población inicial general previamente calculada aplicando (Pereyra, Araguás & Kulich, 2017). En la última etapa evolutiva, la población inicial se forma a partir de la unión de los mejores individuos de cada subpoblación luego de haber evolucionado durante un número dado de generaciones.
- Codificación del cromosoma:** La elección de la codificación del cromosoma es de gran importancia debido a la relación directa con la complejidad que puede presentar luego el diseño de los operadores evolutivos. Se propone un cromosoma que codifica los caminos individuales de cada robot de la formación. Esto es, cada cromosoma estará formado por R alelos, donde R es el número máximo de robots de la formación. A la vez, cada alelo contiene la lista de vértices que conforman el camino individual de cada robot desde el nodo inicial al nodo final, siendo su longitud variable con el número de vértices del camino. En la Figura 2 (derecha) se muestra un ejemplo de la representación de un individuo mediante un cromosoma de cinco alelos, $P = \{p_0, p_1, p_2, p_3, p_4\}$, con $p_0 = \{v_1, v_4, v_5, v_8, v_7\}$, $p_1 = \{v_1, v_4, v_3, v_7\}$, $p_2 = \{v_1, v_4, v_3, v_7\}$, $p_3 = \{v_1, v_4, v_3, v_7\}$, $p_4 = \{v_1, v_2, v_7\}$.

- Función de fitness:** La siguiente se propone como función de fitness,

$$F(x) = C - c_{\max}$$

donde, C es el costo que corresponde a llevar la totalidad de los robots por un mismo camino y c_{\max} el costo del camino de mayor costo del conjunto, $c_{\max} = \max(c_{p1}, c_{p2}, \dots, c_{pR})$.

- Operadores evolutivos:** Los operadores evolutivos encargados de alterar la composición de los individuos de la población a través de las generaciones son similares a los propuestos en AGS, con las siguientes modificaciones,

I. Operadores de cruzamiento:

- Cruzamiento a nivel de caminos o cromosomas: Dados dos individuos de la población se genera un hijo considerando los R caminos de menor costo y otro hijo con los R caminos restantes. Ver Figura 3 (a) y Figura 3 (b).
- Cruzamiento a nivel de aristas o alelos: Dados dos individuos de la población se toman de a uno los caminos del primer individuo y se combinan con uno de los caminos del otro individuo, siempre que sea posible. Dos caminos son combinables si contienen una arista intermedia en común. Ver Figura 3 (c) y Figura 3 (d).

Notar que luego de los cruzamientos los costos de cada camino deben recalcularse debido a que los cambios de configuración pueden modificar los costos de las aristas individuales.

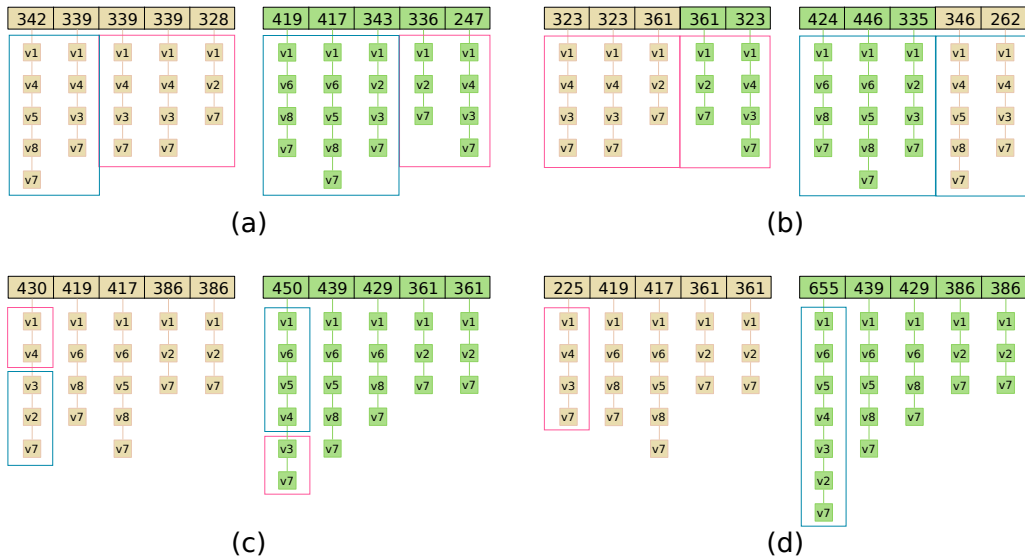


Figura 3: Cruzamiento a nivel de caminos (a) y (b). Cruzamiento a nivel de aristas (c) y (d).

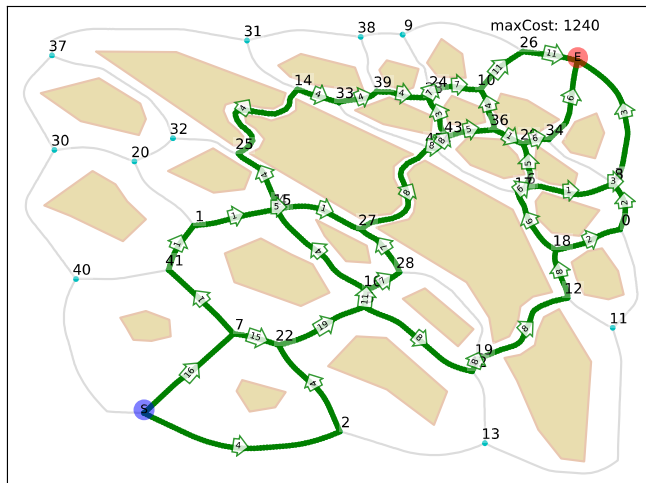
II. Operadores de mutación:

- I. Mutación a nivel de aristas o alelos: por cada camino de una solución se elimina de forma aleatoria una de sus aristas del grafo y se recalcula el camino mediante el algoritmo de Dijkstra para llegar al nodo final desde el vértice correspondiente a la arista eliminada.
- II. Mutación a nivel de caminos: dado un individuo de R caminos se toman aleatoriamente uno de ellos y se copian $R-1$ veces, reemplazando $R-1$ caminos del individuo, seleccionados aleatoriamente.

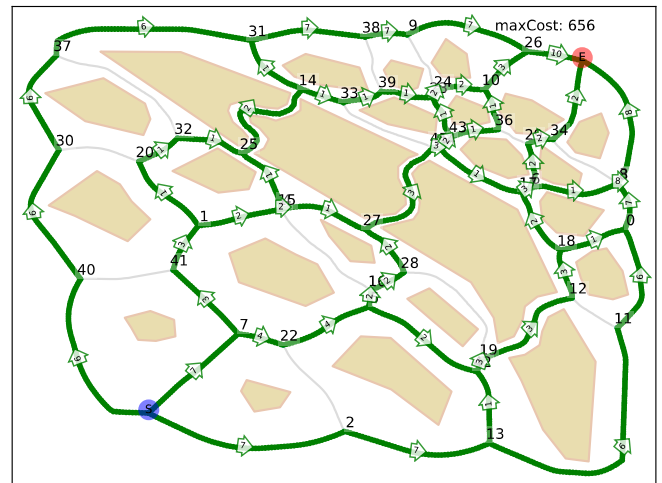
- V. *Parámetros de control y condición de parada:* La elección de los parámetros de control del algoritmo evolutivo, tales como el tamaño de la población, la probabilidad de cruzamiento y mutación, son de fundamental importancia ya que afectan de forma directa el desempeño del algoritmo. En este sentido se configuraron los parámetros de control dependiendo de la etapa en la que se realiza el proceso evolutivo según la estrategia evolutiva implementada (Figura 1) y en base a la experimentación del método y bibliografía consultada (Pereyra, Pérez Paina, Pucheta & Araguás).

Resultados

Los experimentos fueron realizados sobre una serie de mapas de diferentes complejidades y números de obstáculos (nodos) que contienen. Estos mapas son algunos de los publicados en <http://imr.ciirc.cvut.cz/planning/maps.xml> (específicamente se utilizaron los mapas llamados gaps, dense, staggered brick wall, potholes, var density, var density2 y var density3). Los caminos óptimos sobre los que se realiza la evaluación del algoritmo propuesto se obtienen a partir de un método de búsqueda exhaustivo (Pereyra, Araguás & Kulich, 2017). Este algoritmo devuelve el camino óptimo para cada configuración (mapa, Q_{ini} , Q_{goal} , R), pero es de una alta complejidad computacional, y se incrementa exponencialmente con el número de robots y el número de vértices del grafo. Cada evaluación se realiza corriendo el algoritmo de búsqueda con optimización de tipo secuencial (Pereyra, Araguás & Kulich, 2017b), luego corriendo el algoritmo de búsqueda propuesto con optimización basada en algoritmos genéticos (Pereyra, Pérez Paina & Araguás, 2019) y comparando ambos resultados con el obtenido mediante el algoritmo de búsqueda exhaustivo (Pereyra, Araguás & Kulich, 2017). De la evaluación del algoritmo realizado con un número máximo de tres robots sobre un total de aproximadamente 2000 caminos, se obtuvo que el 92 % de los caminos encontrados por el algoritmo secuencial resultaron en caminos óptimos mientras que utilizando el algoritmo genético los caminos óptimos ascendieron al 97 %, dentro de las 20 generaciones en promedio, y con tiempos de cómputo de $\approx 0,2$ segundos por generación. Para hacer extensivo el análisis a más de tres robots y en mapas con mayor cantidad de nodos se realiza una comparación entre los algoritmos secuencial e híbrido (ya que el uso del algoritmo exhaustivo como referencia se hace inviable para esta complejidad). En la Figura 4 se muestra un ejemplo de MPP para una formación de 20 robots en un mapa de 42 nodos aplicando el algoritmo secuencial (a) y el algoritmo híbrido (b). Los resultados de las simulaciones demuestran que el algoritmo es apto para ser utilizado con formaciones de decenas de robots, devolviendo caminos óptimos en un alto porcentaje con tiempos de cómputo mucho menores al método exhaustivo. Como trabajo futuro se espera poder investigar nuevos tipos de operadores evolutivos tendiendo a disminuir los tiempos de cómputo con el fin de poder aplicar el algoritmo en tiempo real y considerar entornos dinámicos.



(a)



(b)

Figura 4: MPP en formación de 20 robots. Aplicando algoritmo secuencial (a) y algoritmo híbrido (b).

Referencias

- R. B. Hart P. E., Nilsson N. J., (1968). "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107.
- E. W. Dijkstra, (1959). "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271.
- S. Yang, M. Meng, (2000). "An efficient neural network approach to dynamic robot motion planning," *Neural Networks*, vol. 13, no. 2, pp. 143–148.
- S. P. Raja, (2012) "Optimal path planning of mobile robots," *International Journal of Physical Sciences*, vol. 7, no. 1, pp. 1314–1320.
- A. T. Qu H., Xing K., (2013). "An improved genetic algorithm with co-evolutionary strategy for global path planning of multiple mobile robots," *Neurocomputing*, vol. 120, no. 1, pp. 509–517.
- A. Noormohammadi Asl, M. B. Menhaj, A. Sajedin, (2014). "Control of leader-follower formation and path planning of mobile robots using asexual reproduction optimization (aro)," *Appl. Soft Comput.*, vol. 14, pp. 563–576.
- C. Hsu and Y. Liu, (2014). "Path planning for robot navigation based on cooperative genetic optimization," in *Proceedings of the 11th IEEE International Conference on Networking, Sensing and Control*, April 2014, pp. 316–321.
- J. Lee and D.-W. Kim, (2016). "An effective initialization method for genetic algorithm-based robot path planning using a directed acyclic graph," *Information Sciences*, vol. 332, pp. 1 – 18. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0020025515007847>
- S. Alnasser and H. Bannaceur, (2016). "An efficient genetic algorithm for the global robot path planning problem," in *Sixth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, pp. 97–102.
- M. E. Pereyra, R. G. Araguás, M. Kulich, (2017). "Sequential path planning for a formation of mobile robots with split and merge," in *Modelling and Simulation for Autonomous Systems*, vol. 10756. Springer International Publishing, pp. 59–71.
- M. E. Pereyra, R. G. Araguás, M. Kulich, (2017b). "Sequential path planning for a formation of mobile robots with split and merge," in *4th IEEE Latin American Conference on Computational Intelligence, LA-CCI 2017*. IEEE, Nov. 2017.
- M. E. Pereyra, D. Gaydou, R. G. Araguás (2017). "Planificador jerárquico de trayectorias para sistemas multi-robots", *Jornadas Argentinas de Robótica, JAR2017*.
- C. Lamini, S. Benhlima, and A. Elbekri, (2018). "Genetic algorithm based approach for autonomous mobile robot path planning," *Procedia Computer Science*, vol. 127, pp. 180 – 189, *ICDS2017*.
- M. E. Pereyra, D. Gonzalez Dondo, R. G. Araguás (2018). "Multirobot path planning using distributed target tracking in a camera network", in *International Symposium of Experimental Robotics - Outreach Workshop -, ISER2018*.
- M. Nazarahari, E. Khanmirza, and S. Doostie, (2019). "Multiobjective multi-robot path planning in continuous environment using an enhanced genetic algorithm," *Expert Systems with Applications*, vol. 115, pp. 106–120.
- M. E. Pereyra, G. Pérez Paina, M. Pucheta, R. G. Araguás (2019). "Planificación de caminos para múltiples robots basado en algoritmo genético", *XXIV Congreso sobre Métodos Numéricos y sus Aplicaciones, ENIEF2019*.
- M. E. Pereyra, G. Pérez Paina, R. G. Araguás (2019). "Planificación de caminos para múltiples robots mediante algoritmo híbrido de búsqueda", *Jornadas Argentinas de Robótica, JAR2019*.