
Algoritmo de sincronización y control de sistemas de robots multiagente para misiones de búsqueda

Andrea Maybell Peña Echeverría



UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería




**Algoritmo de sincronización y control de sistemas de robots
multiagente para misiones de búsqueda**

Trabajo de graduación presentado por Andrea Maybell Peña Echeverría
para optar al grado académico de Licenciado en Ingeniería Mecatrónica

Guatemala,

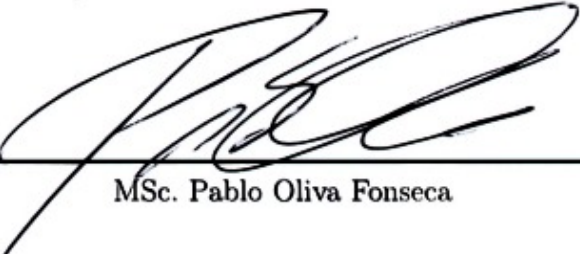
2019

Vo.Bo.:

(f) 
MSc. Miguel Zea Arenales

Tribunal Examinador:

(f) 
MSc. Miguel Zea Arenales

(f) 
MSc. Pablo Oliva Fonseca

(f) 
Ing. Kurt Kellner Juarez

Fecha de aprobación: Guatemala, 5 de diciembre de 2019. 2019

El interés por este trabajo surgió por las grandes posibilidades que presentan los sistemas multi-agente, posibilidades que continúan siendo exploradas. El tener la oportunidad de realizar un estudio en esta área que, incluso en países del primer mundo, sigue siendo una novedad, llena de interrogantes y problemas por resolver me parece increíble. A pesar de que el camino no fue fácil, disfruté cada paso de él. Espero esta investigación sea útil en un futuro para otras personas en este campo de la robótica.

Quisiera aprovechar este espacio para agradecer a quienes me acompañaron y apoyaron durante este proceso. De no ser por estas personas, este trabajo no habría sido posible.

- A mis papás, Evelyn y Javier, quienes me han acompañado en todas y cada una de las etapas de mi vida. Les agradezco su apoyo incondicional y por hacer todo lo posible por brindarme la oportunidad de estudiar esto que me apasiona, por motivarme a siempre seguir aprendiendo y a nunca rendirme.
- A mis hermanos, Jessica Scarlet y Pablo Javier, que a su manera siempre han sabido apoyarme. Por sacarme risas en los momentos más necesarios y por haber estado al pendiente de mi trabajo.
- A mi asesor, Miguel Zea, por guiarme en el proceso. Por compartirme sus conocimientos en la materia, por tener la disponibilidad de resolver mis dudas y siempre brindar alguna sugerencia.
- A mis amigos, que siempre estuvieron allí para apoyarme y motivarme. Por las palabras de ánimo, por los consejos y por estar siempre dispuestos a escuchar.
- A mis compañeros de carrera, con quienes compartimos el proceso aunque fuese en caminos diferentes. Gracias por las ideas, sugerencias, consejos, palabras de apoyo, por estar pendientes los unos de los otros y por hacer de este trabajo uno más ameno.

Prefacio	IV
Lista de figuras	XII
Lista de cuadros	XIII
Resumen	XIV
Abstract	XV
1. Introducción	1
2. Antecedentes	2
3. Justificación	6
4. Objetivos	8
4.1. Objetivo general	8
4.2. Objetivos específicos	8
5. Alcance	9
6. Marco teórico	10
6.1. Teoría de grafos	10
6.1.1. Conceptos básicos	11
6.1.2. Control de la formación	13
6.2. Teoría de control	15
6.2.1. Control de formación	16
6.3. Cinemática de robots diferenciales	17
6.4. Entorno de simulación Webots	18
6.4.1. Características del robot	18

7. Construcción del algoritmo de control	20
7.1. Grafos seleccionados	20
7.1.1. Matrices de adyacencia	21
7.2. Métricas de evaluación	23
7.2.1. Éxito de la formación	23
7.2.2. Cambio de formación ante obstáculos	24
7.3. Ecuaciones de consenso básicas	25
7.4. Ecuación de consenso modificada	29
7.4.1. Modelo 1	29
7.4.2. Modelo 2	33
7.4.3. Modelo 3	35
7.4.4. Modelo 4	40
7.4.5. Modelo 5	44
7.4.6. Modelo 6	48
7.5. Cambio de formación ante obstáculos	55
7.5.1. Control para evasión de obstáculos	55
7.5.2. Cambio de formación	56
8. Implementación en el entorno físico de pruebas	74
8.1. Simulaciones	74
8.1.1. Modificaciones adicionales	74
8.1.2. Éxito de la formación	75
8.1.3. Evasión de obstáculos y cambio de formación	85
9. Conclusiones	94
10.Recomendaciones	95
11.Bibliografía	96
12.Anexos	98
12.1. Repositorio	98

Lista de figuras

1.	Representación de un grafo	11
2.	Representación de un grafo dirigido	11
3.	Pasos de la inserción de Henneberg para un conjunto de 6 vértices	15
4.	Lazo canónico de control	15
5.	Modelo de E-Puck en Webots	19
6.	Grafo de prueba 1, formación triangular	21
7.	Grafo de prueba 2, formación hexagonal	21
8.	Trayectorias de los agentes en simulación de la ecuación de consensus sin modificar	26
9.	Velocidad de los agentes en simulación de la ecuación de consensus sin modificar	26
10.	Trayectorias de los agentes aplicando control de formación	27
11.	Velocidad de los agentes aplicando control de formación	27
12.	Trayectorias de los agentes aplicando control para evitar colisiones	28
13.	Velocidad de los agentes aplicando control para evitar colisiones	28
14.	Trayectorias de los agentes logrando una formación exitosa con el grafo de prueba 1, con una matriz totalmente rígida, utilizando el modelo aditivo	30
15.	Velocidad de los agentes logrando una formación exitosa con el grafo de prueba 1, con una matriz totalmente rígida, utilizando el modelo aditivo	30
16.	Trayectorias de los agentes al no tener éxito en la formación con el grafo de prueba 2, con una matriz mínimamente rígida, utilizando el modelo aditivo	31
17.	Velocidad de los agentes al no tener éxito en la formación con el grafo de prueba 2, con una matriz mínimamente rígida, utilizando el modelo aditivo	31
18.	Función de la tensión en las aristas para el modelo aditivo	32
19.	Trayectorias de los agentes donde se evidencia la acción de “rebote” logrando una formación exitosa con el grafo de prueba 2, con una matriz totalmente rígida, utilizando el modelo aditivo	32
20.	Velocidades de los agentes donde se evidencia la acción de “rebote” logrando una formación exitosa con el grafo de prueba 2, con una matriz totalmente rígida, utilizando el modelo aditivo	33
21.	Función de la tensión en las aristas al combinar el control contra colisiones, control de formación y mantenimiento de la conectividad	34

22.	Trayectorias de los agentes al combinar control de formación, control contra colisiones y mantenimiento de la conectividad	34
23.	Velocidad de los agentes al combinar control de formación, control contra colisiones y mantenimiento de la conectividad	35
24.	Trayectoria de los agentes al realizar una formación exitosa con el grafo de prueba 1 y una matriz totalmente rígida con el modelo 3	36
25.	Velocidad de los agentes al realizar una formación exitosa con el grafo de prueba 1 y una matriz totalmente rígida con el modelo 3	37
26.	Trayectoria de los agentes al realizar una formación exitosa con el grafo de prueba 2 y una matriz totalmente rígida con el modelo 3	37
27.	Velocidad de los agentes al realizar una formación exitosa con el grafo de prueba 2 y una matriz totalmente rígida con el modelo 3	38
28.	Trayectoria de los agentes al no lograr la formación del grafo de prueba 2 y una matriz mínimamente rígida con el modelo 3	38
29.	Velocidad de los agentes al no lograr la formación del grafo de prueba 2 y una matriz mínimamente rígida con el modelo 3	39
30.	Trayectoria de los agentes al no lograr la formación del grafo de prueba 2 y una matriz totalmente rígida con el modelo 3	39
31.	Velocidad de los agentes al no lograr la formación del grafo de prueba 2 y una matriz totalmente rígida con el modelo 3	40
32.	Trayectoria de los agentes al realizar una formación exitosa con el grafo de prueba 1 y una matriz totalmente rígida con el modelo 4	41
33.	Velocidad de los agentes al realizar una formación exitosa con el grafo de prueba 1 y una matriz totalmente rígida con el modelo 4	41
34.	Trayectoria de los agentes al realizar una formación exitosa con el grafo de prueba 2 y una matriz totalmente rígida con el modelo 4	42
35.	Velocidad de los agentes al realizar una formación exitosa con el grafo de prueba 2 y una matriz totalmente rígida con el modelo 4	42
36.	Trayectoria de los agentes al no lograr la formación del grafo de prueba 2 y una matriz mínimamente rígida con el modelo 4	43
37.	Velocidad de los agentes al no lograr la formación del grafo de prueba 2 y una matriz mínimamente rígida con el modelo 4	43
38.	Trayectoria de los agentes al no lograr la formación del grafo de prueba 2 y una matriz totalmente rígida con el modelo 4	44
39.	Velocidad de los agentes al no lograr la formación del grafo de prueba 2 y una matriz totalmente rígida con el modelo 4	44
40.	Función de la tensión para el control de formación entre agentes que no están conectados	45
41.	Trayectoria de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 5	46
42.	Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 5	46
43.	Trayectoria de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 5	47
44.	Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 5	47
45.	Trayectoria de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6	48

46.	Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6	49
47.	Trayectoria de los agentes, formación fallida con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6	49
48.	Velocidad de los agentes, formación fallida con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6	50
49.	Trayectoria de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6	50
50.	Velocidad de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6	51
51.	Trayectoria de los agentes, formación fallida con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6	51
52.	Velocidad de los agentes, formación fallida con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6	52
53.	Trayectorias de los agentes al evadir obstáculos	56
54.	Primer arreglo de obstáculos para pruebas	57
55.	Segundo arreglo de obstáculos para pruebas	57
56.	Tercer arreglo de obstáculos para pruebas	58
57.	Trayectorias de los agentes, formación inicial exitosa con grafos mínimamente rígidos y llegada a la meta exitosa en el primer arreglo de obstáculos con el tiempo de prueba de 3 segundos	59
58.	Velocidades de los agentes, formación inicial exitosa con grafos mínimamente rígidos y llegada a la meta exitosa en el primer arreglo de obstáculos con el tiempo de prueba de 3 segundos	60
59.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa con grafos mínimamente rígidos, sin cambios de formación en el primer arreglo de obstáculos con el tiempo de prueba de 3 segundos	60
60.	Trayectorias de los agentes, formación inicial exitosa que sufre deformación con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos	61
61.	Velocidades de los agentes, formación inicial exitosa que sufre deformación con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos	61
62.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa que sufre deformación con grafos mínimamente rígidos, sin cambios de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos	62
63.	Trayectorias de los agentes, formación inicial fallida con grafos totalmente rígidos y llegada a la meta fallida en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos	62
64.	Velocidades de los agentes, formación inicial fallida con grafos totalmente rígidos y llegada a la meta fallida en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos	63
65.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial fallida con grafos totalmente rígidos, sin cambios de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos	63

66.	Trayectorias de los agentes, formación inicial fallida con éxito final con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 5 segundos	64
67.	Velocidades de los agentes, formación inicial fallida con éxito final con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 5 segundos	64
68.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial fallida con éxito final usando grafos mínimamente rígidos, con dos cambios de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 5 segundos	65
69.	Trayectorias de los agentes, formación inicial exitosa con grafos totalmente rígidos y llegada a la meta fallida en el segundo arreglo de obstáculos con el tiempo de prueba de 5 segundos	65
70.	Velocidades de los agentes, formación inicial exitosa con grafos totalmente rígidos y llegada a la meta fallida en el segundo arreglo de obstáculos con el tiempo de prueba de 5 segundos	66
71.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa usando grafos totalmente rígidos, con un cambio de formación en el segundo arreglo de obstáculos con el tiempo de prueba de 5 segundos	66
72.	Trayectorias de los agentes, formación inicial exitosa y un agente fuera del radar, con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos	67
73.	Velocidades de los agentes, formación inicial exitosa y un agente fuera del radar, con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos	67
74.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa y un agente fuera del radar usando grafos mínimamente rígidos, sin cambios de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos	68
75.	Trayectorias de los agentes, formación inicial exitosa, con grafos totalmente rígidos y llegada a la meta fallida en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos	68
76.	Velocidades de los agentes, formación inicial exitosa, con grafos totalmente rígidos y llegada a la meta fallida en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos	69
77.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa usando grafos totalmente rígidos, con un cambio de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos	69
78.	Trayectoria de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots	75
79.	Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots	76
80.	Trayectoria de los agentes, formación fallida con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots	76

81.	Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots	77
82.	Trayectoria de los agentes, formación exitosa con grafo de prueba 1 y una matriz totalmente rígida con el modelo 6 adaptado a Webots	77
83.	Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz totalmente rígida con el modelo 6 adaptado a Webots	78
84.	Trayectoria de los agentes, formación fallida con grafo de prueba 1 y una matriz totalmente rígida con el modelo 6 adaptado a Webots	78
85.	Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz totalmente rígida con el modelo 6 adaptado a Webots	79
86.	Trayectoria de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots	79
87.	Velocidad de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots	80
88.	Trayectoria de los agentes, formación fallida con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots	80
89.	Velocidad de los agentes, formación fallida con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots	81
90.	Trayectoria de los agentes, formación exitosa con grafo de prueba 2 y una matriz totalmente rígida con el modelo 6 adaptado a Webots	81
91.	Velocidad de los agentes, formación exitosa con grafo de prueba 2 y una matriz totalmente rígida con el modelo 6 adaptado a Webots	82
92.	Trayectoria de los agentes, formación fallida con grafo de prueba 2 y una matriz totalmente rígida con el modelo 6 adaptado a Webots	82
93.	Velocidad de los agentes, formación fallida con grafo de prueba 2 y una matriz totalmente rígida con el modelo 6 adaptado a Webots	83
94.	Escenario 1 de la simulación en Webots	85
95.	Escenario 2 de la simulación en Webots	85
96.	Trayectorias de los agentes, formación inicial exitosa que sufre deformación, con tres cambios de formación y llegada a la meta exitosa en el primer escenario	86
97.	Velocidades de los agentes, formación inicial exitosa que sufre deformación, con tres cambios de formación y llegada a la meta exitosa en el primer escenario	86
98.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa que sufre deformación, con tres cambios de formación y llegada a la meta exitosa en el primer escenario . . .	87
99.	Trayectorias de los agentes, formación inicial exitosa, sin cambios de formación y llegada a la meta exitosa en el primer escenario	87
100.	Velocidades de los agentes, formación inicial exitosa que sufre deformación, sin cambios de formación y llegada a la meta exitosa en el primer escenario .	88
101.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa, sin cambios de formación y llegada a la meta exitosa en el primer escenario	88
102.	Trayectorias de los agentes, formación inicial fallida que llega a ser exitosa a medio camino y luego se deforma, y llegada a la meta exitosa en el segundo escenario	89
103.	Velocidades de los agentes, formación inicial fallida que llega a ser exitosa a medio camino y luego se deforma, y llegada a la meta exitosa en el segundo escenario	89

104.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial fallida que llega a ser exitosa a medio camino y luego se deforma, y llegada a la meta exitosa en el segundo escenario	90
105.	Trayectorias de los agentes, formación inicial exitosa con líder que se separa, sin cambios de formación y llegada a la meta exitosa en el segundo escenario .	90
106.	Velocidades de los agentes, formación inicial exitosa con líder que se separa, sin cambios de formación y llegada a la meta exitosa en el segundo escenario .	91
107.	Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa con líder que se separa, sin cambios de formación y llegada a la meta exitosa en el segundo escenario . . .	91

Lista de cuadros

1.	Descripción de los modelos desarrollados	29
2.	Error cuadrático medio de la formación	53
3.	Porcentaje de formación exitosa	54
4.	Energía del sistema	55
5.	Éxito de las formaciones iniciales y formaciones finales	71
6.	Llegadas exitosas a la meta y error relativo de formación final e inicial	72
7.	Cambios de formación	73
8.	Error cuadrático medio de la formación en la implementación	84
9.	Porcentaje de formación exitosa en la implementación	84
10.	Energía exigida por control en la implementación	84
11.	Éxito de las formaciones iniciales y formaciones finales en la implementación	92
12.	Llegadas exitosas a la meta y error relativo de formación final e inicial en la implementación	92
13.	Cambios de formación en la implementación	92

Este trabajo tuvo como objetivo el desarrollar e implementar un algoritmo de control para un sistema de robots multiagente orientado a misiones de búsqueda, basado en los métodos teóricos de grafos y control moderno. Se logró caracterizar el comportamiento de un enjambre de robots usando estas teorías. Específicamente el desarrollo se basó en la ecuación de consenso modificada y la implementación de grafos mínimamente rígidos.

Se realizó un proceso iterativo para la construcción y selección del subalgoritmo para lograr el éxito de la formación, posteriormente se incluyó el control para la evasión de obstáculos y cambio de formación. El paso entre un modelo y el siguiente fue determinado por las métricas de evaluación, donde el error cuadrático medio y la energía requerida por el sistema fueron las medidas más importantes en la evaluación del modelo de formación y el porcentaje de veces que el sistema llegaba a la meta fue el indicador de éxito del algoritmo completo. Una llegada a la meta implicaba que el sistema fue capaz de formarse y atravesar los obstáculos del entorno.

La implementación se llevó a cabo en el entorno de simulación Webots de Cyberbotics y se evaluaron las métricas establecidas para cuantificar el éxito de este. Estas métricas indicaron que el control únicamente de formación tuvo un éxito del 11 %, mientras que el algoritmo completo tuvo un éxito del 80 %. Es decir, que el enjambre es capaz de alcanzar la meta establecida, aunque no logre en su totalidad la formación inicial.

This work aimed to develop and implement a control algorithm for a multi-agent robot system oriented to searching missions, based on the theoretical methods of graphs and modern control. It was possible to characterize the behavior of a swarm of robots using these theories. The development was specifically based on the modified consensus equation and the implementation of minimally rigid graphs.

An iterative process was carried out for the construction and selection of the sub-algorithm to achieve the success of the formation, subsequently, the control for obstacle avoiding and formation switch was included. The step between one model and the next was determined by the evaluation metrics, where the mean square error and the energy required by the system were the most important measures in the evaluation of the formation model and the percentage of times the system reached the goal was the indicator of the success for the complete algorithm. Arrival at the goal implied that the system was able to form and break through the obstacles of the environment.

The implementation was carried out in the Cyberbotics Webots simulation environment and the metrics established to quantify its success were evaluated. These metrics indicated that the control for formation exclusively had a success of 11 %, while the complete algorithm had a success of 80 %. Which means, the swarm is able to reach the established goal, even if it does not fully achieve the initial configuration.

El estudio de sistemas multiagente es una rama de la robótica y sistemas de computación que surgió recientemente. Estos sistemas presentan un sinfín de posibles aplicaciones, dependiendo de los robots a utilizar, el algoritmo de control, la forma de comunicación, etc. Este trabajo se centra en una aplicación muy específica de la robótica de enjambre: la sincronización y control de formación.

Se desarrollará un algoritmo basado en la teoría de grafos y el control moderno. En este caso particular orientado a sistemas de búsqueda. Para lograrlo se llevará a cabo un proceso iterativo que permita la mejora de cada modelo nuevo con respecto al anterior. Como punto de partida se estudiarán los algoritmos y soluciones presentadas en trabajos previos relacionados. Con esta información puede procederse a la selección de los algoritmos que se adaptan al problema y se investigan los conceptos básicos de las dos teorías a combinar: teoría de grafos y teoría de control.

Teniendo el trasfondo teórico se continuará con la construcción del algoritmo. Es importante definir primero que nada las formaciones de prueba y las métricas de evaluación del algoritmo. Tanto las métricas como el proceso de construcción del algoritmo en sí se dividen en dos secciones importantes: éxito de la formación y cambio de formación ante obstáculos. Estos son los dos subalgoritmos que permiten alcanzar los objetivos. El primero se desarrollará de manera independiente, mediante el proceso iterativo y posteriormente se incluirá la parte del cambio de formación y el control para evasión de obstáculos al último modelo seleccionado.

Finalmente, el algoritmo teórico se implementará en un entorno de simulación controlado para evaluar su eficiencia tomando en consideración los parámetros físicos de los agentes. Se realizará una comparación entre las pruebas teóricas y las pruebas de la implementación y se concluye a partir de esta.

El control aplicado a la formación de robots ha sido un importante tema de estudio en los últimos años, debido a que son evidentes las ventajas que se tienen al implementar un grupo de robots en lugar de utilizar uno solo. Algunas de las principales razones para la utilización de estos sistemas son: la reducción de costos, la robustez, la eficiencia energética y el mejoramiento del desempeño. El control de estos grupos robóticos puede implementarse en distintas aplicaciones como: manipulación y transporte de objetos pesados, exploración, construcción de mapas en territorios desconocidos, operaciones de seguridad y vigilancia, misiones de búsqueda y rescate, etc.

Para llevar a cabo el control multiagente se han implementado diversas técnicas. Estas se diferencian en cuanto a los sensores y actuadores utilizados en los agentes, así como la forma en que los robots se comunican entre ellos y con su entorno. Independientemente de los medios utilizados para la movilización y comunicación de los robots, se pueden distinguir cuatro técnicas [1]:

- Seguimiento de líderes: Un robot líder guía el movimiento de todos los demás.
- Control basado en comportamientos: Se inspira en el comportamiento de los animales.
- Técnicas de estructura virtual: Se considera a la formación de robots como una unidad y los robots intentan alcanzar la estructura.
- Control basado en teorías de control: Se enfoca en la dinámica y estabilidad de la formación de estos sistemas.

Mientras que las estrategias de control se pueden clasificar en:

- Centralizada: Existe un monitoreo de control de todos los robots.
- Descentralizada: No existe un supervisor, se realimentan posiciones relativas de cada robot respecto a sus vecinos.

Es importante conocer el trabajo previo realizado en el área para contar con una base al momento de llevar a cabo una nueva aplicación del tema. A continuación, se describen ejemplos de estas implementaciones. Se destacan principalmente las técnicas utilizadas para el control y la comunicación.

1. En [2] se encuentra un sistema simple de comunicación mínima para el movimiento global de un grupo de robots. Se demostró que no es necesario utilizar un sistema de posicionamiento global para que los robots lleven a cabo un comportamiento global, coordinado y estable. Se utiliza la técnica de seguimiento al líder con dos tipos de formaciones: líder central y líder a la cabeza. El algoritmo propone que la posición de cada robot es relativa a su “robot amigo” y cada robot cuenta con un solo amigo.
2. La innovación en el trabajo realizado en [3] es la arquitectura de comunicación utilizada. Se utilizan dos procesadores: uno para la comunicación y otro para la navegación. También se aplica la técnica de seguimiento del líder, pero el grupo de robots se divide en subgrupos, por lo que no existe un único líder. Para que un usuario se comunique con el conjunto de robots debe hacerlo a través de su líder.
3. En el trabajo [1] se desarrolla un nuevo algoritmo de control de formación centralizado para robots móviles no-holonómicos, basado en la técnica de robots seguidores. En este caso el líder era el único con visión omnidireccional. El controlador logra que los seguidores alcancen la formación deseada mientras siguen al líder. Se realizó una prueba de estabilidad basada en la teoría de Lyapunov.
4. El artículo [4] presenta una nueva forma de controlar múltiples robots móviles en formación utilizando la estrategia de líder seguidor. Se logra evitar los problemas de singularidades y colisiones utilizando un control con retroalimentación. Los controladores de modo deslizante son desarrollados para el seguimiento virtual de los robots, su orientación y el control l-l.
5. En la tesis [5] se mencionan algunas técnicas interesantes para la asignación de tareas multi-robot, entre ellas las técnicas basadas en subastas y el modelado de formaciones. El método basado en subastas se utiliza para la reorganización de formaciones y la inicialización autónoma. Consiste en subastar cada posición de la configuración y asignarla al robot ganador. La oferta es una suma ponderada entre la distancia a la posición final y el número de vecinos actuales. En el caso del modelado la estrategia de control utilizada para cada robot se describe mediante un grafo de control de la formación. Las transiciones de una posición a otra pueden realizarse mediante una matriz de transición.
6. Una aplicación del método de subastas se encuentra en [6]. Aquí el método es utilizado principalmente para la inicialización de la formación de robots. Se explica que también es útil para la reconfiguración de formaciones.
7. La reconfiguración de formaciones se discute también en [7]. En este caso la reconfiguración es necesaria para la navegación entre obstáculos manteniendo cierta formación. Se utiliza la teoría de grafos y la teoría de control no lineal. Se define el control de matrices de transición para controlar el grafo de formación, usando a sus vez el control l-l y l- ψ .

8. Finalmente, en la aplicación [8] se concentran en la localización usando sensores de visión no calibrados. Esto ahorraría mucho trabajo en cuanto a la visión de computadora y permitiría tener un mayor error respecto al sensor de visión. Para ello se utiliza el filtro *Unscented Kalman*. Cabe resaltar que también se utiliza la técnica de seguimiento del líder.

De las implementaciones anteriores se observa que la técnica más utilizada es la de seguimiento de líderes. Sin embargo, las formas en las que se adapta en cada caso son muy distintas. Independientemente de los algoritmos planteados en cada uno de los trabajos anteriores, todos cumplen con ciertas características básicas que se deben tomar en cuenta para realizar una sincronización y control exitosos de los sistemas de robots multiagente: debe establecerse una distancia mínima entre los robots para evitar colisiones, cada robot debe contar con algún tipo de identificador para distinguirse del resto debido a que esto facilita la comunicación y estructuración, y es indispensable algún tipo de localización ya sea global o relativa a los robots vecinos.

Dejando a un lado los algoritmos de control y comunicación, existen diversas implementaciones físicas de la robótica de enjambre. Un mismo algoritmo puede ser utilizado para robots de diseños diversos, considerando su cinemática y sus limitaciones físicas. A continuación se mencionan algunas aplicaciones tangibles de esta tecnología:

1. La universidad de Georgia Tech inauguró su *Robotarium* en agosto del año 2017. El Robotarium es un proyecto que provee acceso remoto a una plataforma de investigación de robótica de enjambre gratuita. Este entorno de experimentación se construyó bajo la creencia de que cualquiera con una buena idea debería de ser capaz de probar sus algoritmos en robots reales en lugar de dejarlo solo en simulaciones. Su misión es: democratizar la robótica proveyendo acceso remoto a un centro de investigación de robots múltiples de última generación.[9][10]
2. Los Kilobots son sistemas robóticos de bajo costo y fáciles de usar para el desarrollo avanzado de enjambres de robots. Los investigadores del Instituto Wyss en la Universidad de Harvard se encuentran desarrollando sistemas robóticos y enfoques algorítmicos para hacer enjambres artificiales de robots que trabajen en colaboración para lograr un objetivo común. La tecnología Kilobot ha sido autorizada por K-Team Corporation para aplicaciones de investigación y educación como plataforma para algoritmos de enjambre colectivo en hardware.[11]
3. En Reino Unido se anunció la creación del primer Robotarium nacional. Este se basará en la innovadora investigación en robótica de Heriot-Watt, que ya está contribuyendo al impulso para revolucionar la fabricación, la asistencia sanitaria, la energía marina, la construcción y las ciencias marinas. El nuevo centro, se inaugurará en el 2021, es una parte integral del tema de la innovación impulsada por datos (DDI).[12]

Inspirados principalmente por el Robotarium en Georgia Tech [9], en la Universidad del Valle de Guatemala se comenzó con el desarrollo de su primera cama de pruebas y los primeros micro-robots para robótica de enjambre en el año 2018. Se tiene un algoritmo de visión de computadora capaz de detectar a cada micro-robot en la mesa de pruebas, con su

respectiva posición y ángulo respecto a la cámara. Este algoritmo integra el procesamiento de imágenes con la red de comunicación inalámbrica para permitir la experimentación en la cama de pruebas.[13][14] El siguiente paso consiste en implementar algoritmos de robótica de enjambre en esta plataforma.

La robótica orientada a entornos hostiles ha realizado importantes contribuciones a la humanidad. Mientras que la robótica multiagente presenta la posibilidad de realizar tareas complejas por un grupo de robots pequeños, sencillos y de bajo costo. Al realizar este trabajo se encontraría una forma en que los robots multiagente podrían ser utilizados para misiones de búsqueda en entornos hostiles. Esto involucra: misiones de rescate en zonas de desastres naturales, exploración de regiones desconocidas, búsqueda en entornos que presentan un riesgo para la vida humana, entre otras.

Trasladándose al contexto guatemalteco, este es un territorio muy propenso a sufrir desastres naturales. Un reporte del Fondo Mundial para la Reducción y Recuperación de Desastres (GFDRR), publicado en 2007, indica que el país se encuentra amenazado permanentemente de manera peligrosa por inundaciones fluviales, urbanas y costeras; terremotos, desprendimientos de tierra, tsunamis, erupciones volcánicas, ciclones e incendios forestales. [15] Si bien, las catástrofes naturales no se pueden evitar, se pueden tomar medidas para que los equipos de búsqueda y rescate se encuentren mejor preparados.

En lugar de arriesgar la vida de los voluntarios, se podría enviar un grupo de robots en una misión de búsqueda. Estos robots tendrían que atravesar obstáculos hasta llegar a su objetivo y podrían enviarle información al equipo de rescate. El uso de micro-robots específicamente, como es el caso de la robótica de enjambre, facilita la movilización en escenarios complicados como los que se presentan luego de un terremoto o un tsunami, arriesgándose a perder únicamente robots sencillos y de bajo costo fáciles de reemplazar.

Además, se contribuiría con la investigación en el campo de la robótica de enjambre. En los últimos años este tema ha sido bastante sonado y se han realizado estudios alrededor del mismo en países desarrollados. Con ello se ha podido observar la vastedad de posibilidades que esta tecnología presenta. Hay un sinnúmero de aplicaciones y algoritmos de implementación que aún no han sido explorados, así como otros que han quedado en la teoría y no han sido llevados a la práctica. Desde algoritmos de comunicación entre el enjambre como el caso de [2] y [3], así como algoritmos de movimiento y configuración [4] y [7].

Mientras, en países como Estados Unidos e Inglaterra, la robótica de enjambre no es un tema nuevo, en Guatemala prácticamente se desconocen las posibilidades de esta tecnología. Los estudios en este tipo de robótica en países desarrollados se iniciaron años atrás y las primeras aplicaciones funcionales ya han salido a la luz. Un ejemplo de la robótica multi-agente aplicada es el Robotarium de Georgia Tech que ahora permite un acceso remoto a sus robots para que cualquier persona que se encuentre trabajando en el campo pueda poner a prueba los algoritmos que desarrolle [9]. Y en Inglaterra la Universidad de Heriot-Watt a comunicado la apertura de su nuevo Robotarium para el año 2021 [12]. Con el reciente proyecto de *Robotat* [13][14] en la Universidad del Valle, es oportuno comenzar con el desarrollo e implementación de algoritmos en robótica multiagente en este país.

4.1. Objetivo general

Desarrollar e implementar un algoritmo de control para un sistema de robots multi-agente orientado a misiones de búsqueda, basado en los métodos teóricos de grafos y control moderno.

4.2. Objetivos específicos

- Caracterizar el comportamiento de un enjambre de robots usando teoría algebraica de grafos y sistemas de control, para desarrollar el algoritmo con base en estas teorías.
- Desarrollar simulaciones en software que permitan evaluar la efectividad del algoritmo propuesto, medida como la probabilidad de que el enjambre de robots alcance exitosamente su meta.
- Implementar el algoritmo propuesto en el entorno de simulación Webots de Cyberbotics que permite modelar los parámetros físicos de los agentes.

Como resultado de este trabajo se tiene un algoritmo funcional y eficiente para movilizar un enjambre de robots desde una posición aleatoria hasta una meta determinada, atravesando obstáculos. El algoritmo logra que el sistema multiagente mantenga una formación siempre que sea posible y reconfigurándose cuando mantener el patrón no le permite atravesar los obstáculos. Se utiliza un sistema de posicionamiento global, donde cada agente cuenta con un identificador propio que determina su posición dentro de la formación.

Se tienen simulaciones pertinentes con escenarios distintos. El algoritmo fue puesto a prueba en el simulador Webots de Cyberbotics, que permite modelar los parámetros físicos de los agentes, validando las simulaciones previas. Se determinó un área de pruebas plana de 2m x 4m con obstáculos de geometría toroidal.

El simulador presenta la ventaja de poder obtener las posiciones globales de todos los agentes, así como de los obstáculos. A diferencia de la plataforma física en la que se requeriría de una red de comunicación para obtener el posicionamiento local de cada robot con respecto a los demás y sensores omnidireccionales para reconocer los obstáculos del entorno. Además, el simulador permite utilizar robots comerciales de los que se ha comprobado su funcionalidad, lo cuál hace posible evaluar el éxito del algoritmo descartando fallas externas relacionadas al robot seleccionado para la implementación.

Las pruebas se realizaron con un enjambre de diez agentes, con posibilidad de reconfigurarse en dos formaciones distintas. Dichos agentes consistieron en robots del modelo E-Puck de GCTronic, robots diferenciales de geometría cilíndrica con dos ruedas.

Existiendo múltiples maneras de controlar un enjambre de robots, en este trabajo se presenta una aplicación que utiliza la teoría de grafos y de sistemas de control. En esta sección se brinda una introducción a los conceptos básicos de cada una de las teorías. Además, se describen los principales términos que serán utilizados en los capítulos posteriores.

6.1. Teoría de grafos

La teoría de grafos es una rama de las matemáticas que se dedica al estudio de los grafos. Conceptualmente un grafo se encuentra formado por vértices y por aristas que conectan dichos vértices. Definiéndolo formalmente se dice que un grafo es una pareja de conjuntos (V, E) , donde V es el conjunto de los vértices y E es el conjunto de las aristas, este último es un conjunto de pares de la forma (u, v) tal que $(u, v) \in V$. [16]

Por ejemplo, teniendo el grafo $G = (V, E)$, donde $V = \{0, 1, 2, 3, 4\}$ es el conjunto de vértices y $E = \{(0, 1), (0, 2), (0, 4), (1, 3), (2, 4)\}$ es el conjunto de aristas. Su representación gráfica se vería como la mostrada en la Figura 1.

Esta teoría es ampliamente utilizada en múltiples y diversas aplicaciones como: análisis de tráfico en redes de telecomunicaciones, búsqueda de la ruta más corta en mapas, bases de datos, relaciones en redes sociales, redes neuronales en inteligencia artificial, entre muchas otras. Estos ejemplos muestran la versatilidad de los grafos, independientemente de la aplicación, la teoría matemática y el manejo algebraico detrás es el mismo. En el presente estudio la teoría de grafos se aplicó en la definición de las formaciones y la red de comunicación entre los agentes.

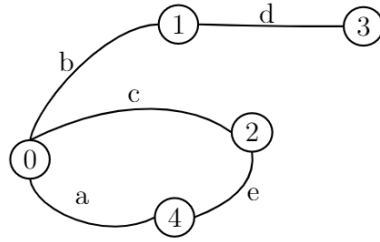


Figura 1: Representación de un grafo

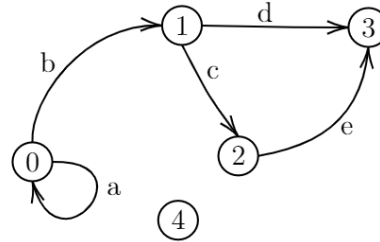


Figura 2: Representación de un grafo dirigido

6.1.1. Conceptos básicos

Una vez se ha definido el concepto de grafo, debe proseguirse con su descripción específica. Para ello se desarrollan conceptos de los tipos de grafos, su representación y sus propiedades, así como de las operaciones básicas que pueden realizarse con ellos. [16]

Clasificación de grafos

- Simple: Un grafo es simple si entre cualquier par de vértices existe como máximo una arista que los une.
- Conexo: En un grafo conexo para cualquier par de vértices (x, y) , existe al menos un camino posible entre x y y .
- Completo: Es aquel en el que todo par de vértices (a, b) tiene una arista que los une.
- Dirigido: También conocido como grafo orientado o digrafo, es aquel en el que las aristas tienen dirección. Un ejemplo en su representación gráfica se muestra en la Figura 2.
- Ponderado: Se le atribuye un número específico a cada arista, es decir, un peso. También se le conoce como grafo valuado o etiquetado.

Propiedades de los elementos de un grafo

- Vértices adyacentes: Son aquellos que conforman un lado de la arista. Cuando un vértice no tiene otro adyacente se dice que es un vértice aislado.

- Grado entrante de un vértice: Es el número de aristas que llegan a un vértice en un grafo dirigido.
- Grado saliente de un vértice: Es el número de aristas que salen del vértice en un grafo dirigido.
- Grado de un vértice: Número de aristas que lo tienen como extremo. En un grafo dirigido este es igual a la suma del grado saliente y del grado entrante. Se designa por $d(v)$ y corresponde al número de aristas incidentes sobre el vértice.

Eigenvalores

Entre las amplias aplicaciones del álgebra lineal, se encuentra la obtención de un conjunto especial de escalares asociados con una matriz.[17] Este conjunto especial de escalares se denominan eigenvalores o valores propios para los cuales se tiene también un conjunto de vectores asociados. Cuando se tiene un sistema lineal, al cual se le aplica algún tipo de transformación lineal, los valores propios resultan ser múltiplos escalares del conjunto de valores asociados. Por lo que, estos valores no cambian la dirección del conjunto de los vectores originales. Para que exista un valor propio en un sistema lineal con matriz A de $n \times n$ y v distintas de cero, debe satisfacerse que y se calcula: [18]

$$\begin{aligned} Av &= \lambda v \\ p(\lambda) &= \det(A - \lambda I) = 0 \end{aligned} \tag{1}$$

Representación matricial de un grafo

El trabajar con la representación gráfica de un grafo resultaría bastante impráctico y podría complicar el identificar sus propiedades. Por ello cuando se trabaja con grafos se opta por la representación matricial. Consiste en representar las propiedades de un grafo, como sus relaciones entre vértices y aristas mediante una matriz. Pueden seleccionarse distintos tipos de matrices según la manipulación o el análisis que se desee aplicar al grafo. A continuación se describen algunas de las matrices que pueden utilizarse.

- Matriz de incidencia: El grafo se representa por una matriz de v vértices por e aristas, donde el elemento a_{ve} contiene la información de la arista. Para grafos no dirigidos si la arista está conectada al vértice $a_{ve} = 1$, en caso contrario $a_{ve} = 0$. Cuando se tiene un grafo dirigido $a_{ve} = 1$ si la arista entra el vértice, $a_{ve} = -1$ si la arista sale del vértice y es $a_{ve} = 0$ si no se da ninguno de los casos anteriores. Se designa como I . Por ejemplo, la matriz de incidencia del grafo mostrado en la Figura 1 se vería de la siguiente manera:

$$I = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

- Matriz de adyacencia: Se puede representar el grafo con una matriz cuadrada de tamaño $v \times v$, donde v es la cantidad de vértices. Designada por A . Si hay una arista entre un vértice x y un vértice y , entonces el elemento $a_{xy} = 1$. Se muestra la matriz de adyacencia para el caso del grafo dirigido de la Figura 2:

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- Matriz de grado: Es una matriz diagonal de dimensiones $v \times v$ que muestra el grado de cada vértice. Se designa como D . Volviendo al grafo de la Figura 1 se tendría la siguiente matriz:

$$D = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

- Matriz laplaciana: Esta matriz se obtiene de operar otras matrices que representan al grafo. Se designa como L . Para el caso de grafos no dirigidos se tiene que la matriz $L = D - A$, mientras que para grafos dirigidos $L = II^T$. Por ende, para el grafo de la Figura 1 analizado anteriormente se tendría:

$$L = \begin{bmatrix} 3 & -1 & -1 & 0 & -1 \\ -1 & 2 & 0 & -1 & 0 \\ -1 & 0 & 2 & 0 & -1 \\ 0 & -1 & 0 & 1 & 0 \\ -1 & 0 & -1 & 0 & 2 \end{bmatrix}$$

6.1.2. Control de la formación

Para el problema de control de formación se tienen dos grafos asociados. El primero es el grafo de formación. Este es un grafo no dirigido que especifica la configuración deseada. Se usa un grafo ponderado donde los valores de las aristas representan las distancias entre los agentes. El segundo grafo asociado a este problema es el grafo que representa la red de comunicación. Este es un grafo dirigido donde las aristas del grafo se definen por la dinámica de lazo cerrado del sistema multi-robot. [19]

Grafo de formación

La definición de este grafo requiere introducir el concepto de rigidez. La rigidez se refiere a que todas las distancias entre los vértices de una estructura están definidas, de tal manera

que se logra una única estructura rígida. En otras palabras, para mantener una estructura rígida las distancias entre vértices deben mantenerse constantes.

A partir de esto surge el concepto de un grafo mínimamente rígido. Para que un grafo bidimensional sea definido como mínimamente rígido el grafo debe tener un estado tal que si se elimina cualquier arista el grafo ya no es rígido. Se puede revisar la rigidez de un grafo si se cumple que $e = 2v - 3$, donde e es el número de aristas y v es el número de vértices. [19] Si un grafo tiene una menor cantidad de aristas ya no se considera rígido, si tiene más el grafo sigue siendo rígido pero ya no es mínimamente rígido. Un grafo completo es un grafo totalmente rígido.

La complejidad y robustez del control está directamente relacionada con la cantidad de aristas del grafo. Por lo que utilizar un grafo completamente rígido resultaría en un diseño poco escalable ya que la cantidad de aristas necesarias si se tienen v vértices, es de $e = \frac{v^2-v}{2}$. Entonces, es beneficioso utilizar grafos mínimamente rígidos ya que no tienen restricciones innecesarias para mantener la formación.

Construcción de grafos mínimamente rígidos La operación conocida como inserción de Henneberg es un método utilizado para la construcción de grafos mínimamente rígidos. Este tipo de inserción conserva la rigidez porque cada vértice tiene dos grados de libertad, por lo que al conectar el vértice a la estructura previa, la posición está sujeta a dos restricciones eliminando los dos grados de libertad. [19] El procedimiento de esta operación es el siguiente, y se representa gráficamente en la Figura 3:

1. Numerar todos los vértices.
2. Añadir una arista entre el vértice 1 y el vértice 2.
3. El resto de vértices se agregan a la estructura mediante dos aristas.
4. Verificar que se ha cumplido con la condición $e = \frac{v^2-v}{2}$.

Red de comunicación

La forma en la que se comunican los agentes entre ellos puede ser representada por un grafo de comunicación. Este suele ser un grafo dirigido que indica qué agentes tienen comunicación entre ellos y la dirección de la misma. Se pueden tener tres diferentes tipos de redes, lo cual cambia su representación como grafo [20]:

- Red estática: Las aristas no varían en el tiempo.
- Red dinámica o dependiente del estado: Las aristas varían en el tiempo y pueden aparecer o desaparecer como funciones dependientes del estado de la red de agentes.
- Red aleatoria: Este es un tipo especial de red dinámica, donde la existencia de una arista en particular está dada por una distribución de probabilidad.

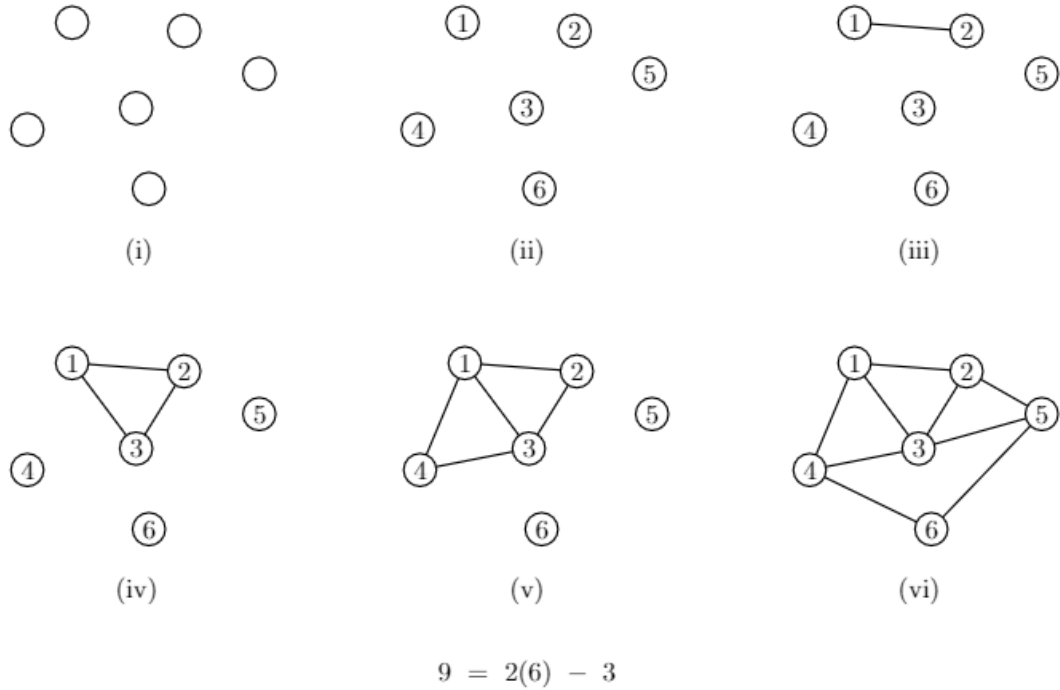


Figura 3: Pasos de la inserción de Henneberg para un conjunto de 6 vértices

6.2. Teoría de control

En el campo de la ingeniería y las matemáticas, la teoría de control lidia con el comportamiento de sistemas dinámicos. Un sistema de control consiste en la integración de varios subsistemas y procesos aplicados a sistemas cuya respuesta es inestable. Esta integración se realiza con el propósito de obtener un comportamiento determinado. La salida deseada del sistema se conoce como la referencia. La idea de estos sistemas es que el control manipula las entradas del sistema para obtener el efecto deseado en la salida. [21] El sistema de control comúnmente utilizado es el conocido como lazo canónico de control, que se muestra en la Figura 4.

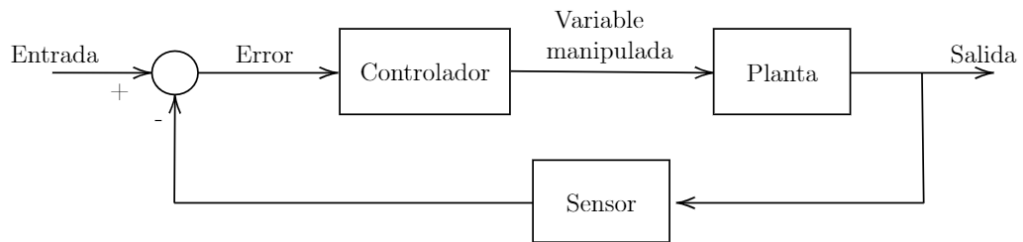


Figura 4: Lazo canónico de control

La modelación matemática de fenómenos físicos involucra un sistema de varias variables. A pesar que algunos sistemas pueden ser representados como sistemas lineales sencillos, surge la necesidad que la teoría de control abarque tanto sistemas univariados como multivariados.

De allí es que se separa la teoría de control en control clásico y control moderno, que se encargan respectivamente, de cada caso.

6.2.1. Control de formación

En cuanto a la formación, el control va enfocado a que los agentes se reúnan en una configuración determinada y que sean capaces de mantener dicha formación a pesar de las perturbaciones. Es decir, se desea controlar ya sea la posición global de cada agente o la posición relativa de los agentes respecto a sus vecinos; dependiendo de la red de comunicación. Para ello se modifica la velocidad de los agentes acorde al error que existe entre la posición relativa deseada y la posición actual. A continuación, se describe la teoría utilizada para poder determinar cómo debe ser modificada la velocidad de cada agente. [21].

Ecuación de consenso

Uno de los problemas fundamentales en el control de sistemas multiagentes es el de consenso, donde un grupo de agentes acuerda en llegar al mismo estado final. Este problema involucra n unidades dinámicas interconectadas mediante enlaces de intercambio de información. La tasa de cambio de cada unidad depende de la suma de su estado relativo a las otras unidades. Esto da como resultado la siguiente expresión

$$\dot{x}_i(t) = \sum_{j \in N(i)} (x_j(t) - x_i(t)), \quad i = 1, 2, \dots, n \quad (2)$$

donde, $N(i)$ es el conjunto de unidades adyacentes, o de vecinos, de la unidad i en la red de agentes. [20]

La ecuación 2 es la base para resolver la mayoría de problemas de sistemas multiagentes. El problema canónico, conocido como problema de rendezvous, busca que dado un grupo de agentes que solo conocen la distancia relativa entre sus vecinos todos coincidan en la misma posición. Mientras el grafo esté conectado, la ecuación de consenso lleva a todos los agentes al mismo estado.

Si se toma como base la ecuación de consenso y se le añaden pesos, es posible resolver problemas más allá del de rendezvous. Para añadir pesos, se debe definir la tensión de las aristas entre el grupo de agentes. De esta forma el problema se plantea en función de reducir lo máximo posible las tensiones entre agentes. Dada la tensión de una arista por

$$\varepsilon = \sum_{i=1}^N \sum_{j=1}^N a_{i,j} \varepsilon_{i,j} (\|x_i - x_j\|) \quad (3)$$

al derivar se obtiene

$$\frac{\partial \varepsilon_{i,j}}{\partial x_i} = \omega_{i,j} (\|x_i - x_j\|) (x_i - x_j) \quad (4)$$

de donde es posible despejar para el peso $\omega_{i,j}$. Este peso se define según el problema a resolver. A continuación, se muestran algunos ejemplos de cómo se puede modificar la ecuación de consenso para problemas específicos:

- Control de formación:

$$\varepsilon_{i,j} = \frac{1}{2}(\|x_i - x_j\| - d_{ij})^2 \quad (5)$$

$$\omega_{ij} = \frac{\|x_i - x_j\| - d_{ij}}{\|x_i - x_j\|} \quad (6)$$

$$\dot{x}_i = - \sum_{j \in N(i)} \frac{(\|x_i - x_j\| - d_{ij})(x_i - x_j)}{\|x_i - x_j\|} \quad (7)$$

- Mantenimiento de la conectividad:

$$\varepsilon_{ij} = \frac{\|x_i - x_j\|^2}{\Delta - \|x_i - x_j\|} \quad (8)$$

$$\omega_{ij} = \frac{2\Delta - \|x_i - x_j\|}{(\Delta - \|x_i - x_j\|)^2} \quad (9)$$

$$\dot{x}_i = - \sum_{j \in N(i)} \frac{(2\Delta - \|x_i - x_j\|)(x_i - x_j)}{(\Delta - \|x_i - x_j\|)^2} \quad (10)$$

6.3. Cinemática de robots diferenciales

Para transformar las velocidades lineales de un modelo de control al modelo cinemático de un robot de dos ruedas se hace uso de una matriz de difeomorfismo. La construcción de esta matriz parte de la idea que el punto que se desea controlar no se encuentra al centro del robot, sino un poco más adelante, es decir queremos determinar dónde se encontrará el robot al instante siguiente.[22]

Si el nuevo punto se determina de la siguiente forma:

$$\begin{cases} \tilde{x} = x + l \cos \phi \\ \tilde{y} = y + l \sin \phi \end{cases} \quad (11)$$

La velocidad de este nuevo punto estaría dado por las ecuaciones 12 y 13.

$$\dot{\tilde{x}} = \dot{x} - l\dot{\phi} \sin \phi = v \cos \phi - l\omega \sin \phi \quad (12)$$

$$\dot{\tilde{y}} = \dot{y} + l\dot{\phi} \cos \phi = v \sin \phi + l\omega \cos \phi \quad (13)$$

donde, l representa la distancia desde el centro del robot hasta el nuevo punto propuesto y ϕ es el ángulo de orientación del robot en el plano XY. Si se dice que

$$\dot{\tilde{x}} = u_1, \quad \dot{\tilde{y}} = u_2$$

entonces, se puede proponer la siguiente transformación:

$$\begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} v \\ l\omega \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (14)$$

donde,

$$R(\phi) = \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \quad (15)$$

y

$$\begin{bmatrix} v \\ \ell\omega \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \ell \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (16)$$

Así que para obtener la velocidad lineal y la velocidad angular del robot con dos ruedas puede utilizarse la transformación 17, donde u_1 y u_2 corresponden a los componentes en X y Y de la velocidad determinada por el modelo 6.

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\ell} \end{bmatrix} R(-\phi) \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (17)$$

Teniendo la velocidad lineal y angular del robot, puede realizarse una segunda transformación para obtener la velocidad rotacional de cada rueda del robot. Para obtener la velocidad de la rueda derecha y la rueda izquierda se utilizan las ecuaciones 18 y 19, respectivamente.

$$\Phi_R = \frac{v + \omega\ell}{r} \quad (18)$$

$$\Phi_L = \frac{v - \omega\ell}{r} \quad (19)$$

donde, r representa el radio de las ruedas.

6.4. Entorno de simulación Webots

Webots es un simulador de robots de código abierto que provee un entorno completo para modelar, programar y simular robots. Webots ha sido co-desarrollado por el Instituto Federal Suizo de Tecnología en Lausana, probado exhaustivamente, bien documentado y actualizado continuamente desde 1996. Es la solución más eficiente para obtener rápidamente resultados profesionales. [23]

Este software permite realizar simulaciones físicas precisas, esto gracias a sus modelos calibrados. Webots se basa en una versión extendida del motor de física ODE. Varios de los modelos de simulación provistos en Webots están calibrados contra sus contrapartes reales. Esto incluye objetos, sensores, actuadores y robots. El proceso de calibración consiste en ajustar los parámetros físicos de los objetos y dispositivos para que se comporten de la misma forma que el modelo físico real. Estos parámetros físicos incluyen: distribución de masa, coeficientes de fricción, propiedades de resorte, amortiguación, etc. Los usuarios pueden calibrar sus propios modelos para garantizar resultados de simulación realistas. [23]

6.4.1. Características del robot

Para una implementación de robótica de enjambre se considera que el robot E-Puck de GCTronics es una buena opción para las pruebas. Cuenta con una gran variedad de sensores y su programación es amigable por lo que es comúnmente utilizado en este tipo de aplicaciones. E-Puck es un robot móvil miniatura desarrollado originalmente por la Escuela Politécnica Federal de Lausana, con propósitos educativos por los diseñadores del robot Khepera.[24]

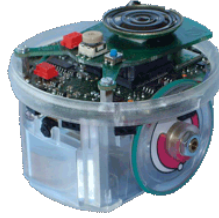


Figura 5: Modelo de E-Puck en Webots

Característica	Valor
Diámetro	71 mm
Altura	50 mm
Radio de la rueda	20.5 mm
Longitud del eje	52 mm
Peso	0.16 kg
Velocidad lineal máxima	0.25 m/s
Velocidad rotacional máxima	6.28 rad/s

Construcción del algoritmo de control

A continuación se describe el proceso de construcción del algoritmo de control. Esto implica el modelado matemático y las pruebas realizadas en software para probar dichos modelos. Se muestra el proceso iterativo que permitió la construcción final del algoritmo. Todas las simulaciones fueron realizadas en Matlab, aprovechando la facilidad de visualización que presenta este software.

7.1. Grafos seleccionados

La definición del modelo matemático final requirió de varias pruebas para hallar aquel que lograra el comportamiento ideal. Para ello se propusieron inicialmente dos grafos de pruebas mostrados en la Figura 6 y 7, ambos de 10 nodos. Los grafos se construyeron usando el método de inserción de Henneberg para lograr que fuesen mínimamente rígidos.

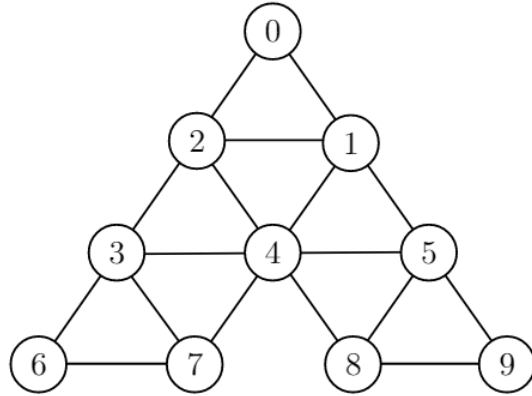


Figura 6: Grafo de prueba 1, formación triangular

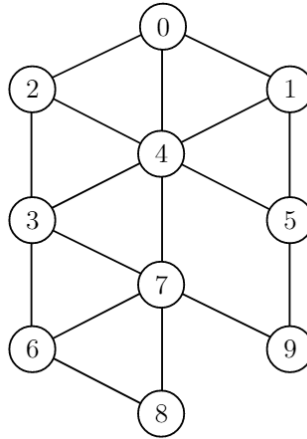


Figura 7: Grafo de prueba 2, formación hexagonal

Para seleccionar los grafos se buscó que las configuraciones fuesen lo suficientemente distintas en cuanto a la distribución de los agentes, las dimensiones de la formación y la cantidad de nodos adyacentes de cada vértice. Además, la asignación de los agentes se determinó de tal manera que las formaciones seleccionadas permitiesen una transición suave entre una y otra, es decir, que no implicasen cambios drásticos en la posición de los agentes.

7.1.1. Matrices de adyacencia

Para cada grafo se construyó su matriz de adyacencia mínimamente rígida y totalmente rígida. Esto con el fin de realizar una comparación sobre cómo la rigidez del grafo afecta el comportamiento del enjambre y el funcionamiento del algoritmo. Con matriz mínimamente rígida, se hace referencia a la matriz de adyacencia del grafo construido con el método de Henneberg; mientras que la matriz totalmente rígida es la matriz de adyacencia del grafo completo.

Matriz de adyacencia mínimamente rígida del grafo de prueba 1

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Matriz de adyacencia totalmente rígida del grafo de prueba 1

$$\begin{bmatrix} 0 & 1 & 1 & 2 & \sqrt{3} & 2 & 3 & \sqrt{7} & \sqrt{7} & 3 \\ 1 & 0 & 1 & \sqrt{3} & 1 & 1 & \sqrt{7} & 2 & \sqrt{3} & 2 \\ 1 & 1 & 0 & 1 & 1 & \sqrt{3} & 2 & \sqrt{3} & 2 & \sqrt{7} \\ 2 & \sqrt{3} & 1 & 0 & 1 & 2 & 1 & 1 & \sqrt{3} & \sqrt{7} \\ \sqrt{3} & 1 & 1 & 1 & 0 & 1 & \sqrt{3} & 1 & 1 & \sqrt{3} \\ 2 & 1 & \sqrt{3} & 2 & 1 & 0 & \sqrt{7} & \sqrt{3} & 1 & 1 \\ 3 & \sqrt{7} & 2 & 1 & \sqrt{3} & \sqrt{7} & 0 & 1 & 2 & 3 \\ \sqrt{7} & 2 & \sqrt{3} & 1 & 1 & \sqrt{3} & 1 & 0 & 1 & 2 \\ \sqrt{7} & \sqrt{3} & 2 & \sqrt{3} & 1 & 1 & 2 & 1 & 0 & 1 \\ 3 & 2 & \sqrt{7} & \sqrt{7} & \sqrt{3} & 1 & 3 & 2 & 1 & 0 \end{bmatrix}$$

Matriz de adyacencia mínimamente rígida del grafo de prueba 2

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Matriz de adyacencia totalmente rígida del grafo de prueba 2

$$\begin{bmatrix} 0 & 1 & 1 & \sqrt{3} & 1 & \sqrt{3} & \sqrt{7} & 2 & 3 & \sqrt{7} \\ 1 & 0 & \sqrt{3} & 2 & 1 & 1 & \sqrt{7} & \sqrt{3} & \sqrt{7} & 2 \\ 1 & \sqrt{3} & 0 & 1 & 1 & 2 & 2 & \sqrt{3} & \sqrt{7} & \sqrt{7} \\ \sqrt{3} & 2 & 1 & 0 & 1 & \sqrt{3} & 1 & 1 & \sqrt{3} & 2 \\ 1 & 1 & 1 & 1 & 0 & 1 & \sqrt{3} & 1 & 2 & \sqrt{3} \\ \sqrt{3} & 1 & 2 & \sqrt{3} & 1 & 0 & 2 & 1 & \sqrt{3} & 1 \\ \sqrt{7} & \sqrt{7} & 2 & 1 & \sqrt{3} & 2 & 0 & 1 & 1 & \sqrt{3} \\ 2 & \sqrt{3} & \sqrt{3} & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 3 & \sqrt{7} & \sqrt{7} & \sqrt{3} & 2 & \sqrt{3} & 1 & 1 & 0 & 1 \\ \sqrt{7} & 2 & \sqrt{7} & 2 & \sqrt{3} & 1 & \sqrt{3} & 1 & 1 & 0 \end{bmatrix}$$

A lo largo de la construcción del algoritmo se realizan comparaciones entre formaciones totalmente rígidas y mínimamente rígidas. Los modelos están enfocados en mejorar los resultados para formaciones mínimamente rígidas. Para comprender la razón de esta preferencia, se debe entender lo que representan las aristas en el grafo. Las aristas de cada grafo de formación a parte de indicar las distancias entre un agente y sus vecinos, determina la red de comunicación entre los robots del enjambre.

Expandiendo un poco la idea, esto implica que mientras más aristas se tengan en el grafo, más canales de comunicación deben existir. La utilización de un grafo totalmente rígido conlleva a que todos los agentes deban conocer su distancia con respecto a todos los demás agentes. En otras palabras, todos los robots deberían ser omniscientes. Mientras que al utilizar formaciones mínimamente rígidas, basta con que cada agente conozca la información de sus vecinos inmediatos únicamente. Esto disminuye la complejidad de la red de comunicación y permite utilizar un sistema de localización local en lugar de uno global.

7.2. Métricas de evaluación

En esta sección se definen los aspectos que serán evaluados en cada iteración para poder hacer un análisis cuantitativo de la efectividad del algoritmo. Primero se hace la aclaración que el algoritmo desarrollado puede dividirse en dos subalgoritmos principales: uno para lograr la formación a partir de un estado de posiciones aleatorias y otro para cambiar congruentemente de configuración cuando el enjambre se encuentre con obstáculos. Cada uno de estos subalgoritmos fue analizado de manera independiente.

7.2.1. Éxito de la formación

Para el subalgoritmo de formación se considerarán dos aspectos primordiales: el éxito de la formación y la energía utilizada por el sistema para alcanzar la configuración. El éxito de la formación será medido de dos maneras: de forma binaria y en forma porcentual. La medición binaria se refiere a que únicamente se evaluará si se logró la formación o no, para

esto se utilizará el error cuadrático medio:

$$ECM = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^n (x_{ij} - d_{ij})^2 \quad (20)$$

donde, x_{ij} representa la distancia actual entre el agente i y el agente j y d_{ij} es la distancia deseada entre dichos agentes, es decir, la que se encuentra en la posición ij de la matriz de formación seleccionada. Entonces, el lograr una formación implica que se tiene un error cuadrático medio menor a un umbral determinado. En los resultados mostrados en el Cuadro 2 se realizó la medición con 6 umbrales diferentes: 0.001, 0.005, 0.01, 0.02, 0.04 y 0.07. El primer umbral es para identificar las formaciones técnicamente perfectas y el último umbral es para incluir las configuraciones que aún se asemejan visualmente a la formación deseada.

La evaluación de manera porcentual indicará qué porcentaje de la formación fue exitosa, o sea qué porcentaje de los agentes llegaron a la posición deseada. Este es un modo de medir el error individual de cada robot. Para esto se calculará un porcentaje de error de la formación actual frente a la formación deseada:

$$\%error_{ij} = \frac{\sqrt{(x_{ij} - d_{ij})^2}}{|d_{ij}|} \times 100 \% \quad (21)$$

Los robots con distancias entre agentes i y j que presenten porcentajes de error mayores al límite establecido, en este caso 15 %, serán considerados como agentes fuera de la formación. Cualquier formación donde menos del 70 % de los agentes se encuentren en la configuración deseada será clasificada como una formación fallida. Por ello, en el Cuadro 3 se muestra la cantidad de simulaciones en las que el 100 %, 90 %, 80 % y 70 % de los agentes llegaron a la posición deseada.

Respecto a la medición de la energía del sistema, esta se hizo con base en las velocidades alcanzadas por cada robot durante la simulación. Se realizó una aproximación de la integral de la velocidad:

$$\sum_{i=1}^n \int_{t_i}^{t_f} \|\dot{x}_i\|^2 dt \quad (22)$$

La importancia de este análisis yace en que la energía de los robots reales es finita, es decir, se deben cumplir ciertos requerimientos de velocidades y aceleraciones máximas. Por lo que la energía debe ser optimizada. Los resultados de estas mediciones se presentan en el Cuadro 4. Se establecieron tres umbrales de energía para conocer cuántas simulaciones sobrepasaban estos límites: 60, 100 y 200. Además, se calculó la desviación estándar, el promedio y el coeficiente de variación (C.V.) de los resultados de todas las simulaciones. De esta manera es posible apreciar la variación existente entre los niveles de energía entre un evento y otro utilizando el mismo algoritmo.

7.2.2. Cambio de formación ante obstáculos

Luego, para el subalgoritmo de cambio de formación se vuelve a utilizar la métrica del error cuadrático medio para medir el éxito de la formación. Para este caso se mide el éxito de la formación inicial y el de la formación final, es decir la configuración con la que el enjambre de robots llegó a la meta. Esto indica si el algoritmo fue capaz de realizar el

cambio de configuración de forma exitosa. Además, se encuentra el error relativo entre la formación inicial y la formación final, esto indica qué tanto se deformó el enjambre de robots en su trayectoria hasta la meta. Dicho error relativo se calcula como:

$$ER = \frac{ECM_f}{ECM_i} \quad (23)$$

Un error relativo mayor a 1, indica que el error de la formación inicial es menor que el de la formación final. Es decir el enjambre de robots sufrió una deformación luego de atravesar el arreglo de obstáculos. Mientras que un error relativo menor a uno implica que la configuración final tuvo un menor error con respecto a la inicial. Estos resultados se muestran en los Cuadros 5 y 6.

Y finalmente se realiza un conteo de cuántas veces el grupo de robots inició su trayecto a través de los obstáculos, esto sucede únicamente si el enjambre logra establecer una formación inicial, y cuántas de estas veces fue capaz de llegar a la meta independientemente del error de formación. Adicionalmente se cuentan los cambios de configuración realizados por simulación, Cuadro 7.

7.3. Ecuaciones de consenso básicas

Como punto de partida se simuló la ecuación de consensus y sus principales modificaciones. Estas simulaciones permitieron analizar el comportamiento de los agentes según la variación de parámetros como: distancia para evitar colisiones, distancia deseada por el control de formación, distancia máxima deseada entre agentes para que no desaparezcan del radar de sus vecinos, entre otras.

En la Figura 9 puede observarse que el consensus tiene un comportamiento suave, y logra que todos los agentes se reúnan en un mismo punto lo que lleva a que la velocidad sea cero. Esta velocidad de cero al final, es la que demuestra la convergencia. El problema canónico de rendezvous, no lleva a los agentes a cualquier punto común, los lleva al centro de masa de las posiciones iniciales [20], esto se hace evidente en la Figura 8.

Al aplicar las principales modificaciones al consensus, como lo son el control de formación y la evasión de obstáculos, se obtuvieron los resultados esperados. De la misma manera que el consensus, estas presentan un comportamiento suave y el centro de masa se mantiene, Figuras 10 a la 13.

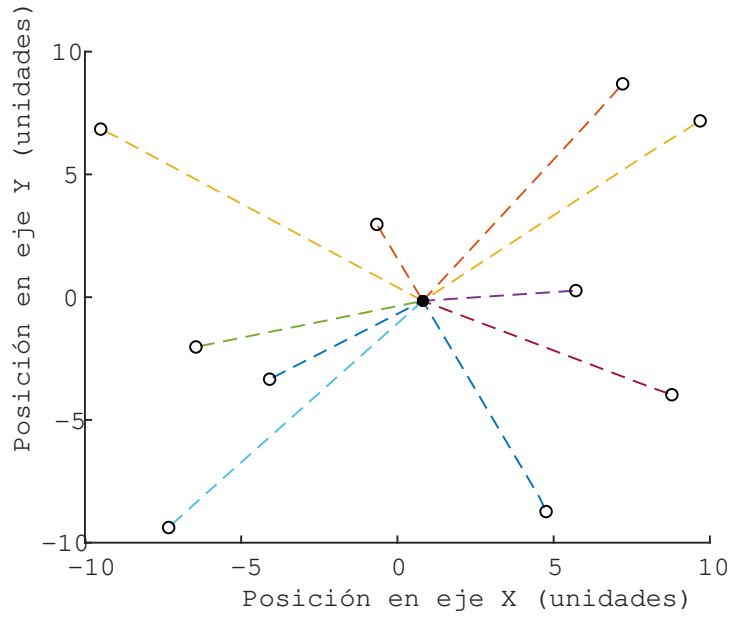


Figura 8: Trayectorias de los agentes en simulación de la ecuación de consenso sin modificar

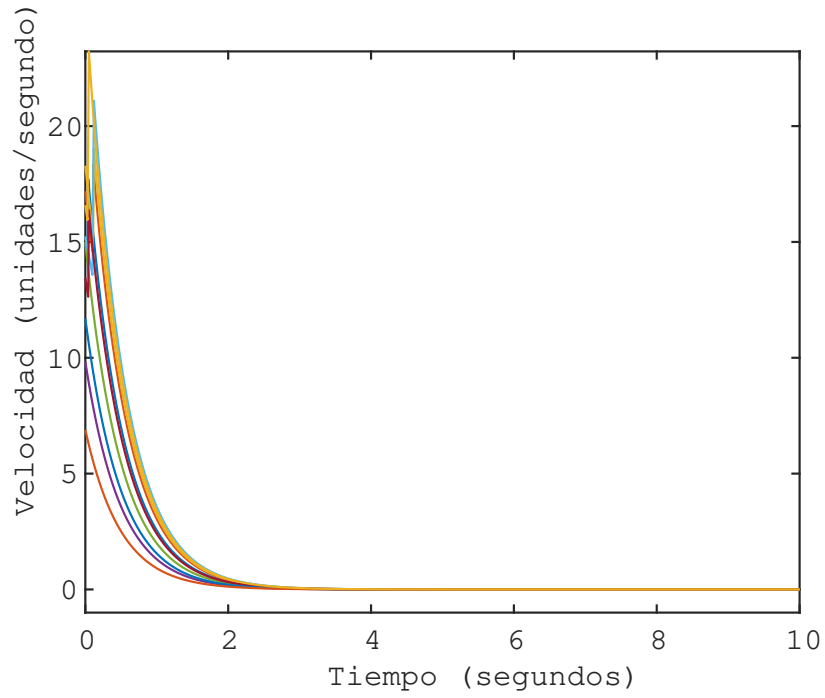


Figura 9: Velocidad de los agentes en simulación de la ecuación de consenso sin modificar

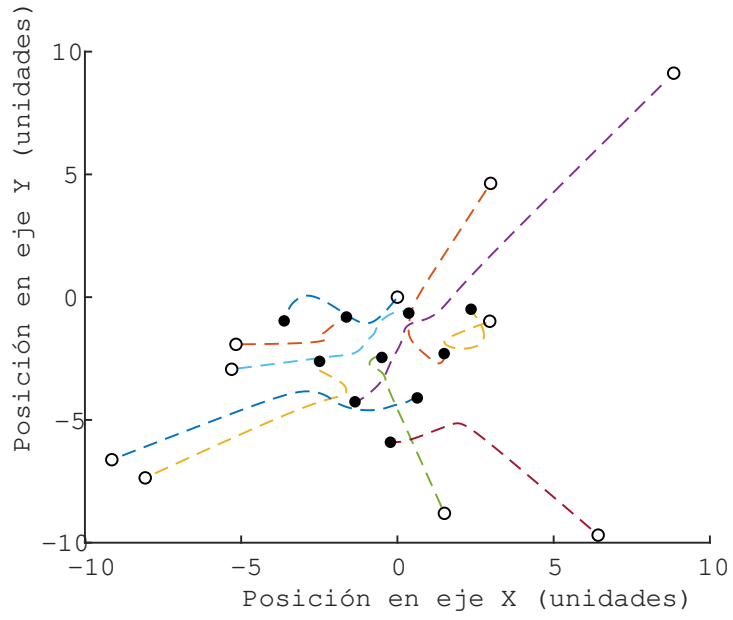


Figura 10: Trayectorias de los agentes aplicando control de formación

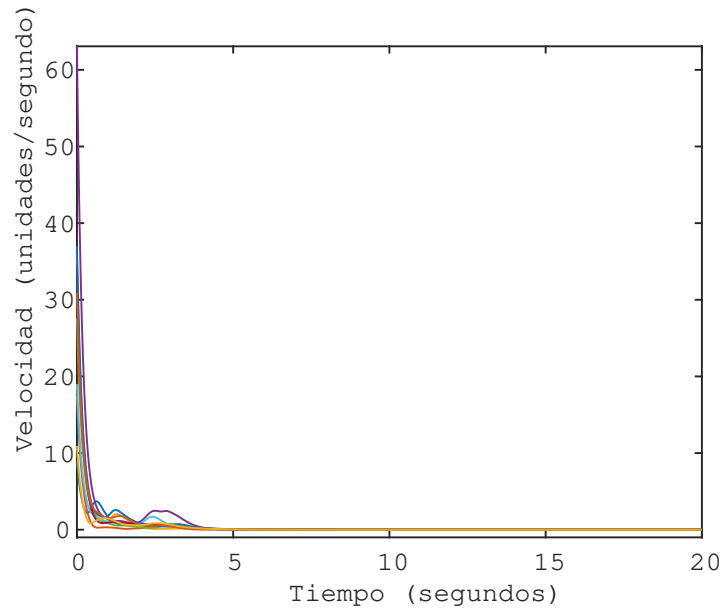


Figura 11: Velocidad de los agentes aplicando control de formación

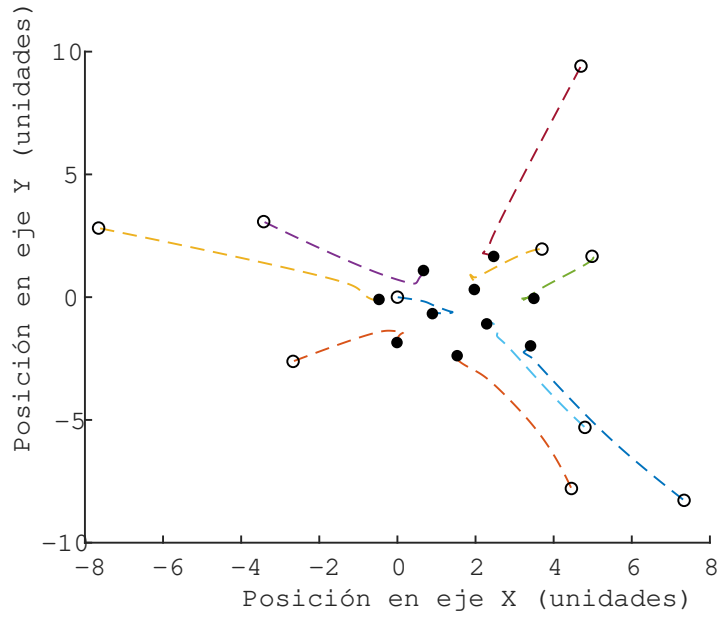


Figura 12: Trayectorias de los agentes aplicando control para evitar colisiones

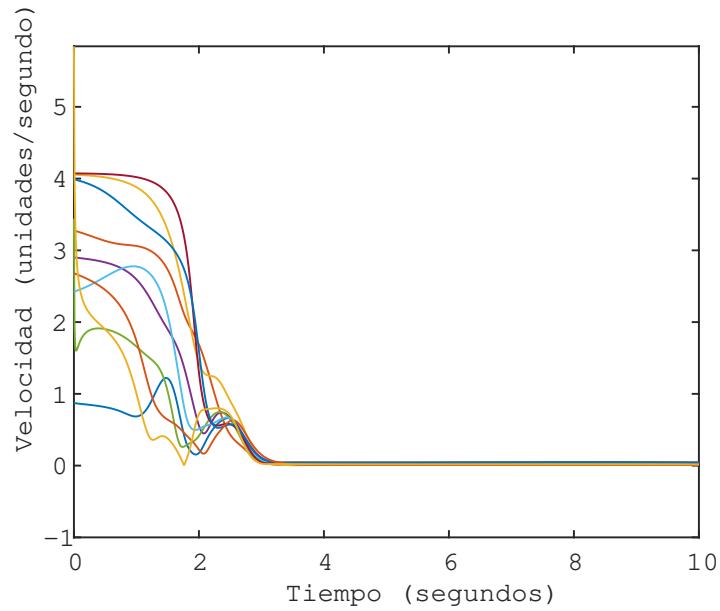


Figura 13: Velocidad de los agentes aplicando control para evitar colisiones

Luego de conocer el comportamiento individual de las modificaciones al consenso que se tomaron como referencia, se inició con las combinaciones de los modelos. Las simulaciones permitieron observar los detalles que tenían que mejorarse en el modelo que fue construido de forma iterativa.

7.4. Ecuación de consenso modificada

La modificación de la ecuación de consenso se realizó manipulando la tensión deseada entre las aristas del grafo formado por todos los agentes. Para hallar esta tensión se propusieron funciones racionales que tomaran en cuenta las distancias restringidas entre agentes y las distancias deseadas. Las restricciones están dirigidas a que los agentes no se choquen entre sí y las distancias deseadas dependen de la formación elegida. En el Cuadro 1 se presenta una descripción de los modelos desarrollados de manera iterativa.

Modelo	Descripción
0	Únicamente se aplica control de formación
1	Combinación aditiva del control de formación y evasión de obstáculos
2	Combinación de control de formación, evasión de obstáculos y mantenimiento de la conectividad
3	Combinación de control de formación y evasión de obstáculos
4	Modelo dinámico con control de formación y evasión de obstáculos
5	Modelo dinámico con control de formación usando coseno hiperbólico y evasión de obstáculos
6	Modelo dinámico con control de formación usando coseno hiperbólico y evasión de obstáculos, incluyendo límites de velocidad

Cuadro 1: Descripción de los modelos desarrollado.

7.4.1. Modelo 1

Por ello, la primera función propuesta toma en cuenta ambos aspectos: evitar colisiones y lograr la formación. Para llegar a este modelo se tomaron como referencia las ecuaciones (5) y (8). Se propuso un modelo en el que dichas ecuaciones se combinaban de forma aditiva, cada una con un peso diferente en la velocidad de los agentes:

$$\varepsilon_{ij} = \frac{1}{2}(\|x_i - x_j\| - d_{ij})^2 + \frac{\|x_i - x_j\|^2}{\|x_i - x_j\| - r} \quad (24)$$

Al utilizar matrices totalmente rígidas, el éxito de la formación total de este modelo fue del 58 % y se logró al menos el 70 % de la formación en 85 de 100 casos. Mientras que con las matrices mínimamente rígidas no se logra ninguna formación total y se dan menos de 12 eventos con al menos 7 agentes formados correctamente (Cuadro 3). Esto podría ser considerado como un éxito suficiente si se mantienen las formaciones rígidas.

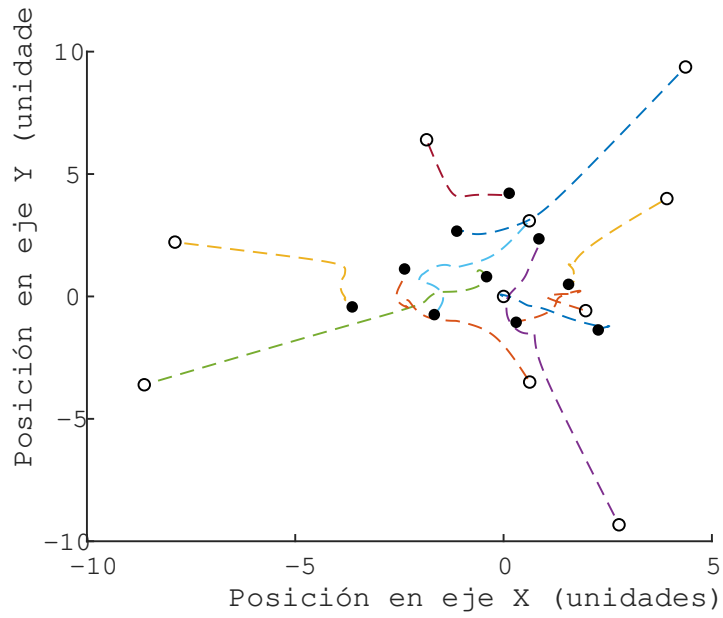


Figura 14: Trayectorias de los agentes logrando una formación exitosa con el grafo de prueba 1, con una matriz totalmente rígida, utilizando el modelo aditivo

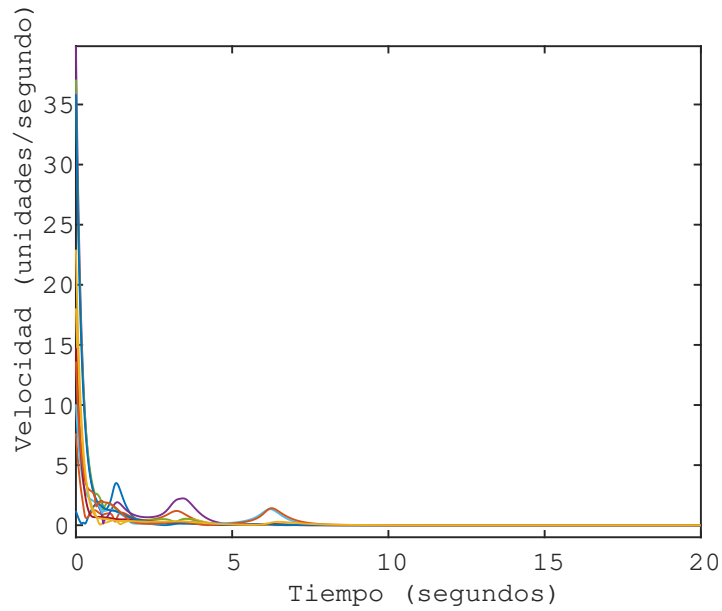


Figura 15: Velocidad de los agentes logrando una formación exitosa con el grafo de prueba 1, con una matriz totalmente rígida, utilizando el modelo aditivo

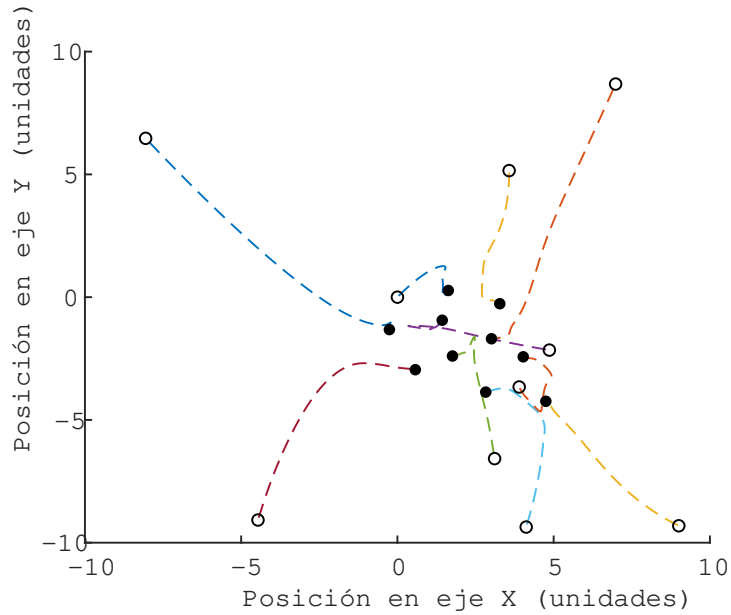


Figura 16: Trayectorias de los agentes al no tener éxito en la formación con el grafo de prueba 2, con una matriz mínimamente rígida, utilizando el modelo aditivo

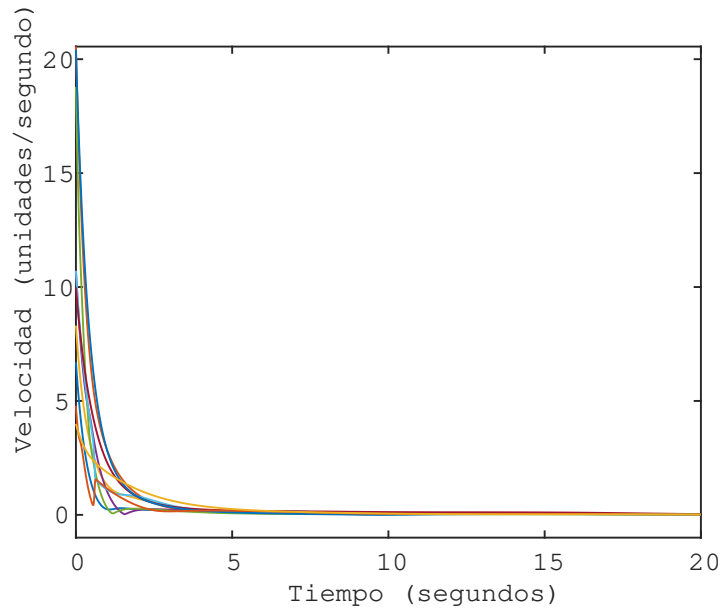


Figura 17: Velocidad de los agentes al no tener éxito en la formación con el grafo de prueba 2, con una matriz mínimamente rígida, utilizando el modelo aditivo

Sin embargo, al analizar la energía requerida por el sistema se tienen grandes magnitudes de energía, menos de 40 casos con energía menor a 60 unidades, y con coeficientes de variación muy variados entre sí. Estas mediciones corresponden a un problema visualizado cualitativamente, donde la formación era exitosa, Figura 19, pero la velocidad de los agentes tendía al infinito en ciertas ocasiones, Figura 20. Esto se debe a que al sumar los modelos que se deseaban combinar, la función final tiene ciertas discontinuidades, como se observa en

el gráfico de la función de tensión de la Figura 18. Esto hace parecer que los agentes rebotan a una gran velocidad, de una posición a otra. Cada modelo busca cumplir su objetivo de manera independiente, sin considerar las restricciones del resto.

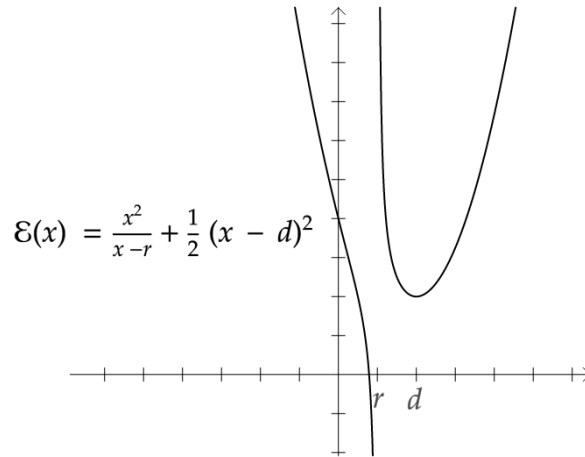


Figura 18: Función de la tensión en las aristas para el modelo aditivo

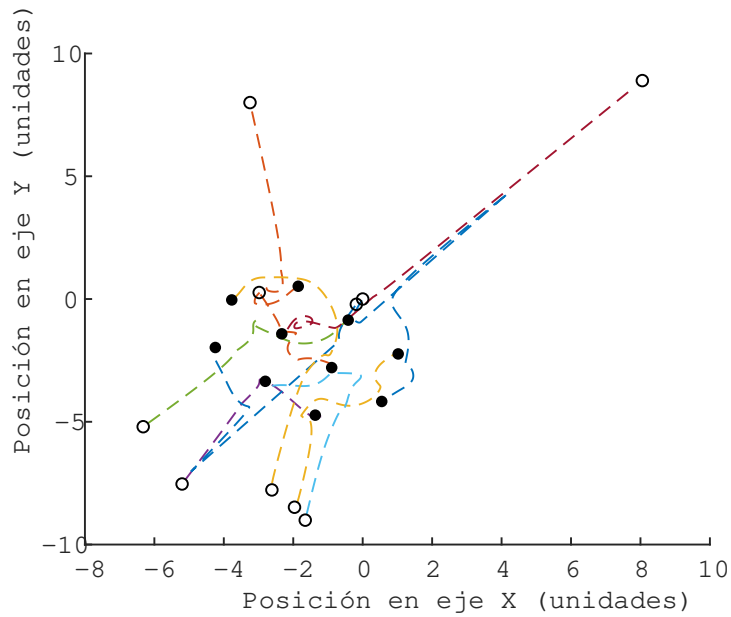


Figura 19: Trayectorias de los agentes donde se evidencia la acción de “rebote” logrando una formación exitosa con el grafo de prueba 2, con una matriz totalmente rígida, utilizando el modelo aditivo

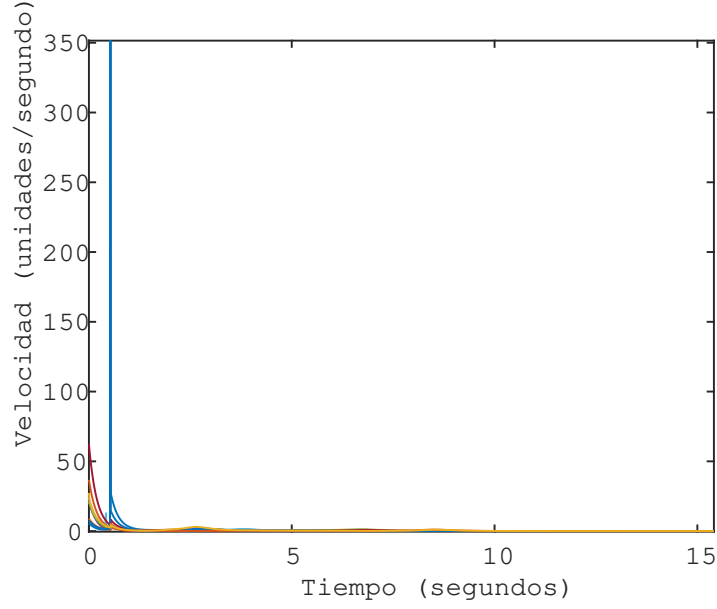


Figura 20: Velocidades de los agentes donde se evidencia la acción de “rebote” logrando una formación exitosa con el grafo de prueba 2, con una matriz totalmente rígida, utilizando el modelo aditivo

7.4.2. Modelo 2

Entonces, se desea tener una función continua que busque un mismo objetivo y que evite que los agentes se alejen indefinidamente. Contemplando esto, se propuso una nueva función, incluyendo un control para el mantenimiento de la conectividad y combinando las funciones referencia de manera directa:

$$\varepsilon_{ij} = \frac{(\|x_i - x_j\| - d_{ij})^2}{(\|x_i - x_j\| - r)(\|x_i - x_j\| - R)} \quad (25)$$

donde, d_{ij} es la distancia deseada entre los agentes i y j para mantener la formación, r es el radio de los agentes y R es la distancia máxima que se pueden alejar los agentes sin ser perdidos por el radar unos de otros. La gráfica de esta función se muestra en la Figura 21.

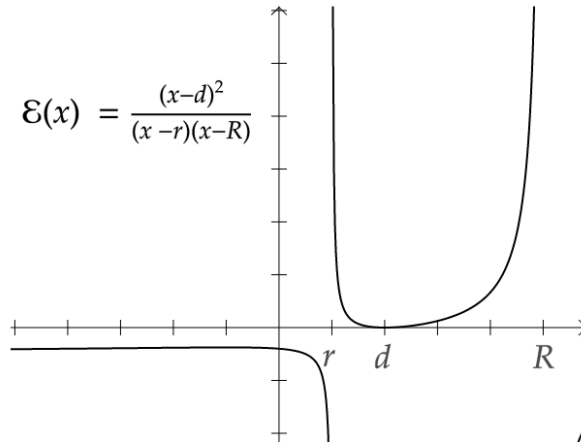


Figura 21: Función de la tensión en las aristas al combinar el control contra colisiones, control de formación y mantenimiento de la conectividad

Al incluir el control para mantenimiento de la conectividad, los agentes presentaron un comportamiento extraño muy alejado de lo deseado, figuras 22 y 23.

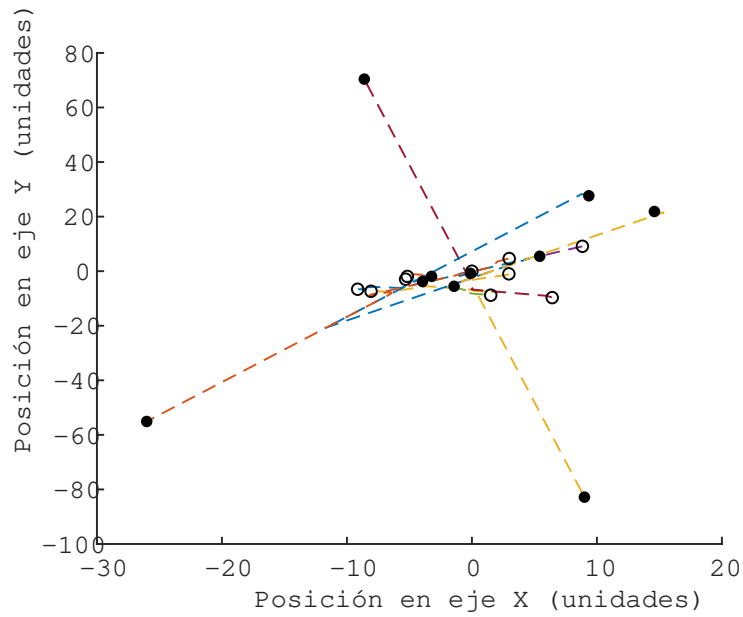


Figura 22: Trayectorias de los agentes al combinar control de formación, control contra colisiones y mantenimiento de la conectividad

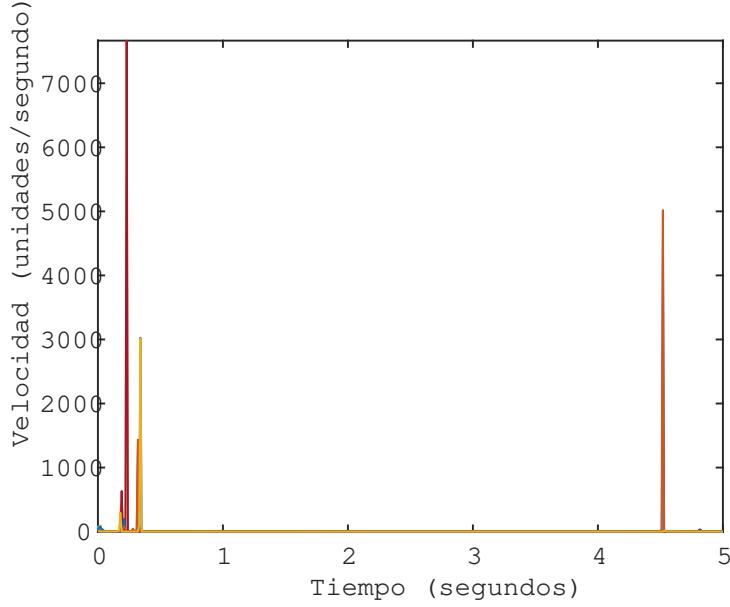


Figura 23: Velocidad de los agentes al combinar control de formación, control contra colisiones y mantenimiento de la conectividad

Este fue el modelo que presentó los porcentajes de éxito más bajos incluso utilizando matrices totalmente rígidas. También, se observa en el Cuadro 4 (modelo 2) que muestran los promedios de energía más altos de todas las simulaciones. El problema de esta función se encuentra en las asíntotas verticales. Ya que, si alguno de los agentes se encuentra fuera del alcance del radar, en este caso la distancia R , tendrá un comportamiento no deseado. El movimiento del agente estaría determinado por la parte negativa de la función que no cumple con los requerimientos.

7.4.3. Modelo 3

Descartando el control para mantenimiento de la conectividad, se mantuvo la combinación del control de formación y el control contra colisiones en una sola ecuación en lugar de sumarlas.

$$\varepsilon_{ij} = \frac{2(\|x_i - x_j\| - d_{ij})^2}{\|x_i - x_j\| - r} \quad (26)$$

Derivando y dividiendo entre la magnitud de la distancia entre agentes se tiene:

$$\omega_{ij} = \frac{4(\|x_i - x_j\| - d_{ij})(\|x_i - x_j\| - r) - 2(\|x_i - x_j\| - d_{ij})^2}{(\|x_i - x_j\| - r)^2(\|x_i - x_j\|)} \quad (27)$$

Finalmente, la velocidad de cada agente i está dada por:

$$\dot{x}_i = - \sum_{j \in N_i} \omega_{ij} (\|x_i - x_j\|) (x_i - x_j) \quad (28)$$

En las Figuras 24 a la 37 se muestran varios resultados obtenidos con esta simulación. Con esta función se obtuvieron mejores resultados, que en el modelo 2. Aunque se redujo

significativamente el éxito de formaciones totales entre este modelo y el modelo 1, aproximadamente la mitad, Cuadro 2. Los parámetros de energía no cambiaron significativamente entre los modelos, usando matrices totalmente rígidas. En el caso de las matrices mínimamente rígidas, este último modelo presentaba mejoras en la cantidad de energía necesaria. La energía promedio entre el modelo 1 y el modelo 3 se redujo en un 20 % y el coeficiente de variación en un 85 %. Sin embargo seguía siendo necesario utilizar matrices totalmente rígidas para lograr la formación, e incluso utilizando este tipo de matrices no se puede asegurar que en todas las ocasiones se llegará a la formación deseada.

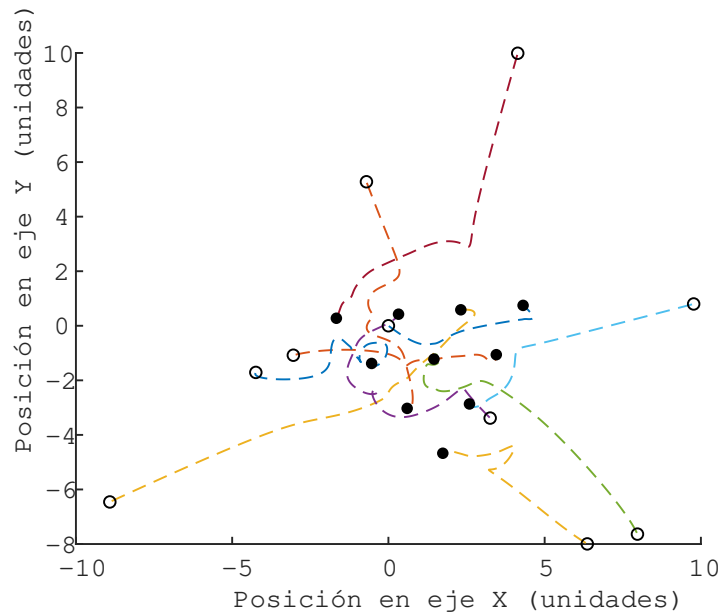


Figura 24: Trayectoria de los agentes al realizar una formación exitosa con el grafo de prueba 1 y una matriz totalmente rígida con el modelo 3

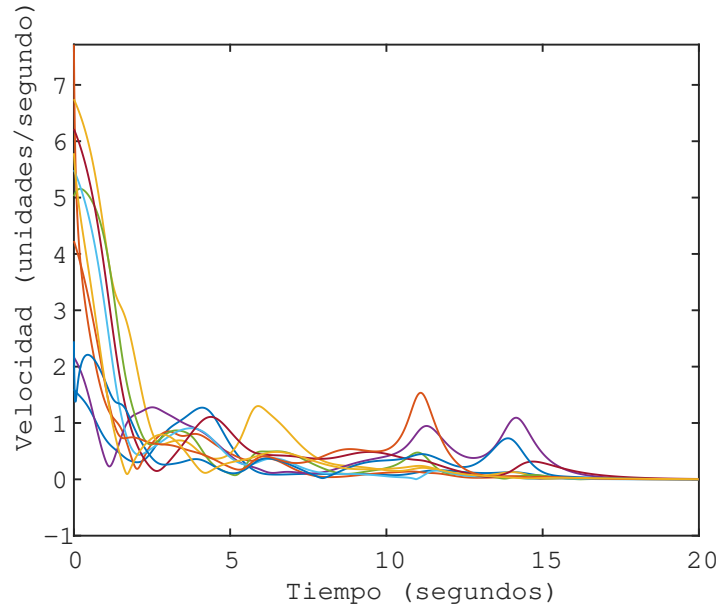


Figura 25: Velocidad de los agentes al realizar una formación exitosa con el grafo de prueba 1 y una matriz totalmente rígida con el modelo 3

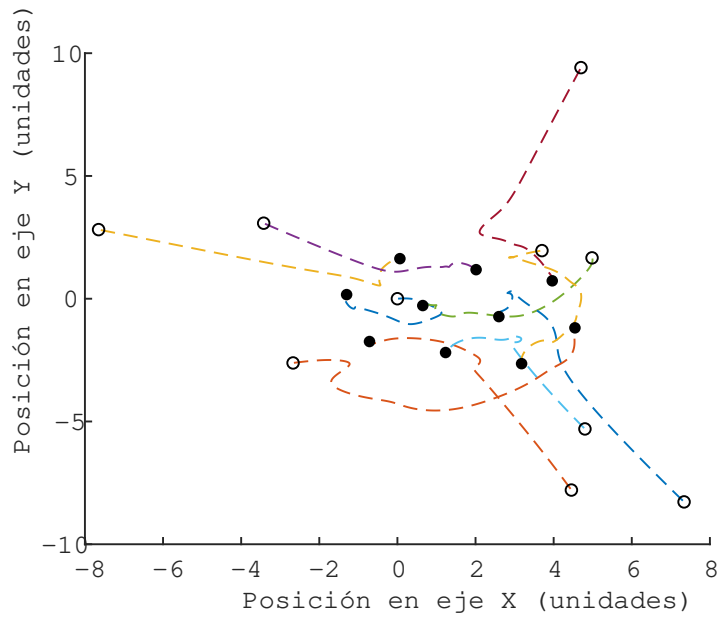


Figura 26: Trayectoria de los agentes al realizar una formación exitosa con el grafo de prueba 2 y una matriz totalmente rígida con el modelo 3

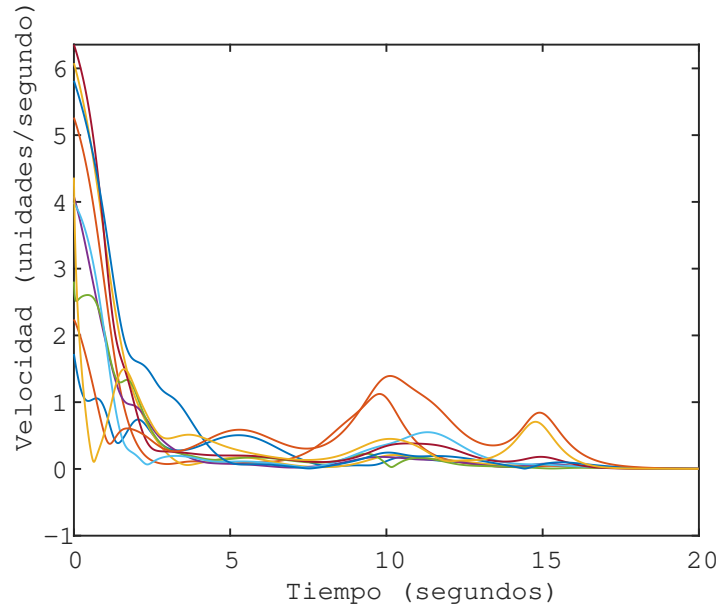


Figura 27: Velocidad de los agentes al realizar una formación exitosa con el grafo de prueba 2 y una matriz totalmente rígida con el modelo 3

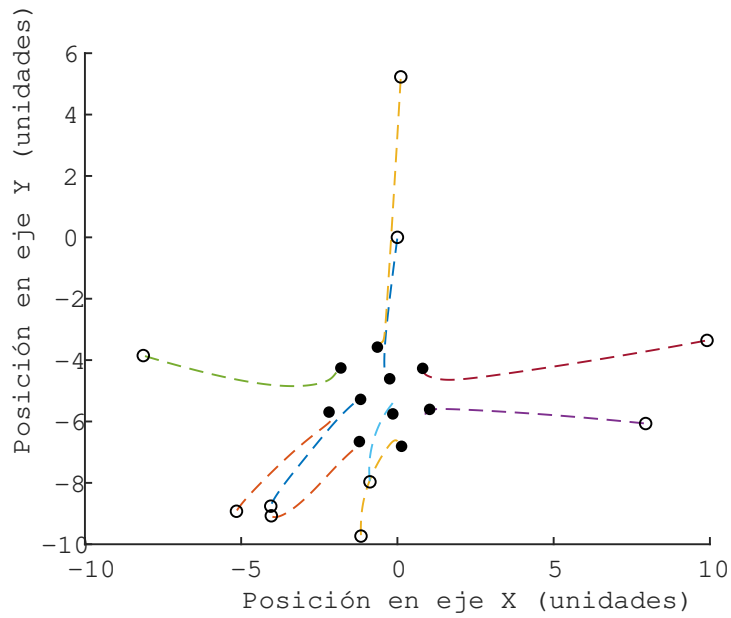


Figura 28: Trayectoria de los agentes al no lograr la formación del grafo de prueba 2 y una matriz mínimamente rígida con el modelo 3

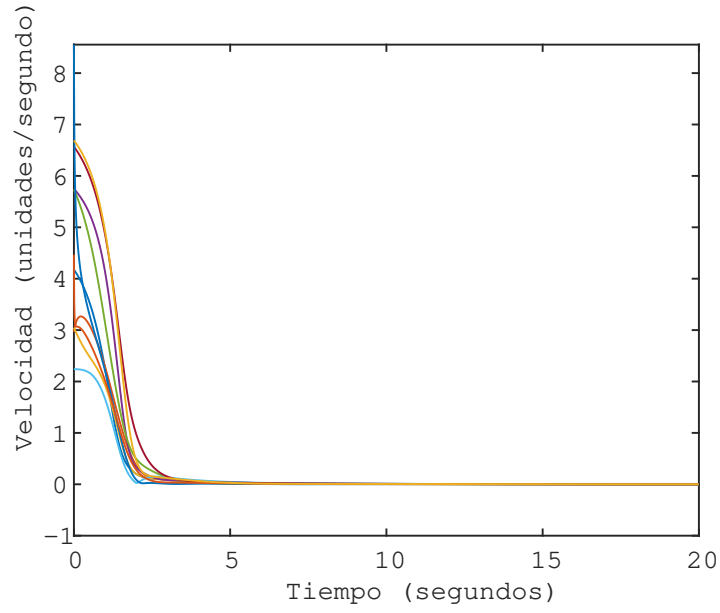


Figura 29: Velocidad de los agentes al no lograr la formación del grafo de prueba 2 y una matriz mínimamente rígida con el modelo 3

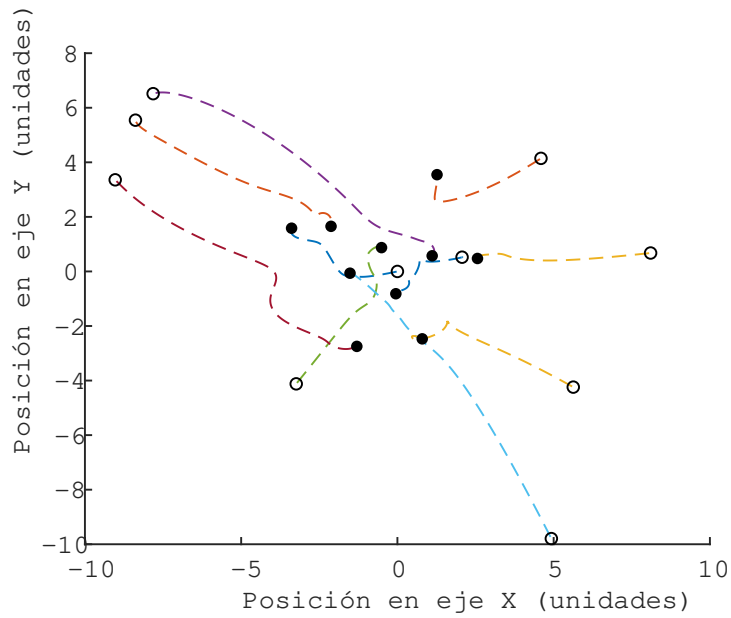


Figura 30: Trayectoria de los agentes al no lograr la formación del grafo de prueba 2 y una matriz totalmente rígida con el modelo 3

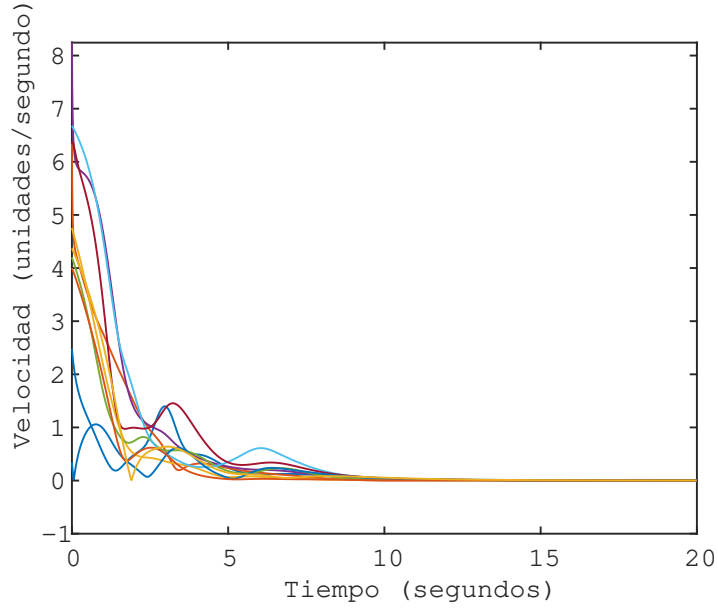


Figura 31: Velocidad de los agentes al no lograr la formación del grafo de prueba 2 y una matriz totalmente rígida con el modelo 3

7.4.4. Modelo 4

Tomando en cuenta los resultados de las funciones anteriores y sabiendo que mientras más restricciones se tienen respecto a las distancias entre los agentes, la formación como unidad se vuelve más difícil de manipular, se propone un modelo dinámico. Al inicio, cuando los agentes comienzan con posiciones aleatorias, se aplica únicamente el control para evitar colisiones mientras se logra el consenso. De esta forma se logra que todos los agentes dentro del radio del sensor de distancia se acerquen. Una vez los agentes se encuentren lo suficientemente cerca sin chocarse, se cambia al modelo que incluye el control de la formación, mostrado en (26). Esto no tuvo un efecto considerable en el porcentaje de éxito de las simulaciones pero el caso de agentes con velocidades que tendían al infinito se redujo. El hecho de que todos los agentes se acerquen unos con otros como primer paso asegura que todos estén bajo el radar de los demás, lo cuál facilita el lograr la formación. Sin embargo los cambios entre un control y otro implican un aumento brusco de la velocidad en los agentes y por lo tanto un aumento en la energía necesaria por el sistema.

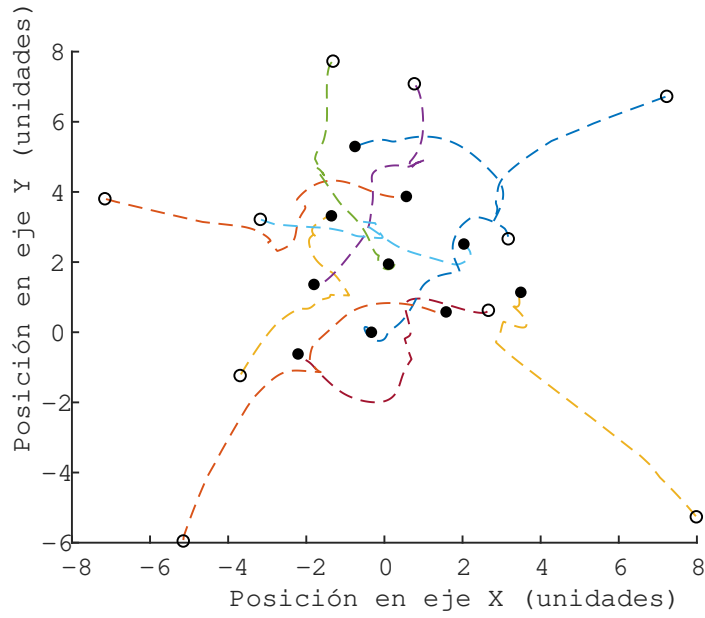


Figura 32: Trayectoria de los agentes al realizar una formación exitosa con el grafo de prueba 1 y una matriz totalmente rígida con el modelo 4

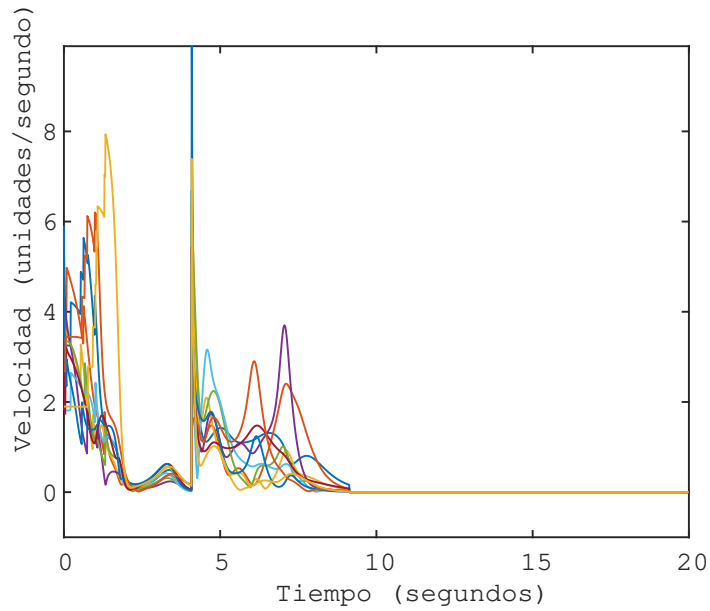


Figura 33: Velocidad de los agentes al realizar una formación exitosa con el grafo de prueba 1 y una matriz totalmente rígida con el modelo 4

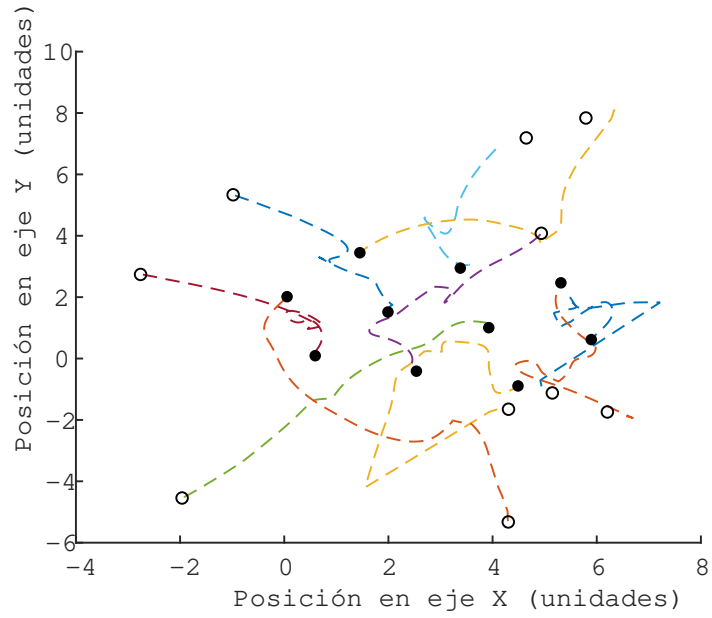


Figura 34: Trayectoria de los agentes al realizar una formación exitosa con el grafo de prueba 2 y una matriz totalmente rígida con el modelo 4

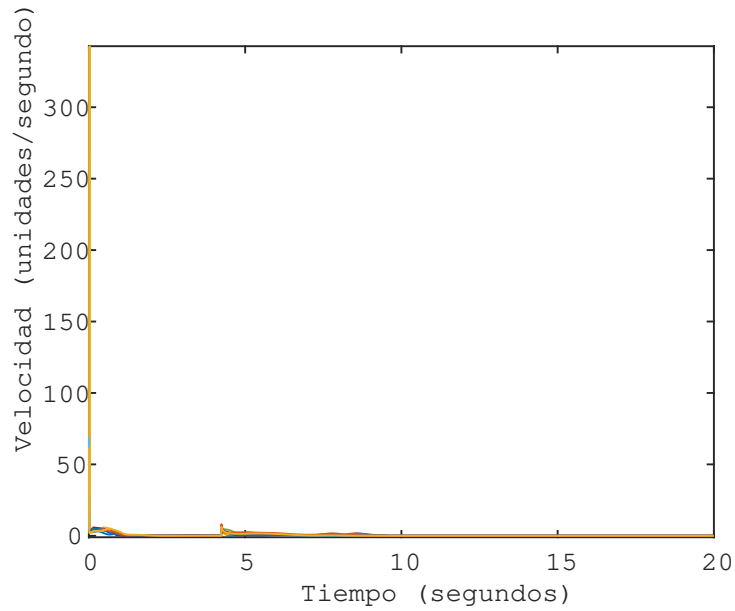


Figura 35: Velocidad de los agentes al realizar una formación exitosa con el grafo de prueba 2 y una matriz totalmente rígida con el modelo 4

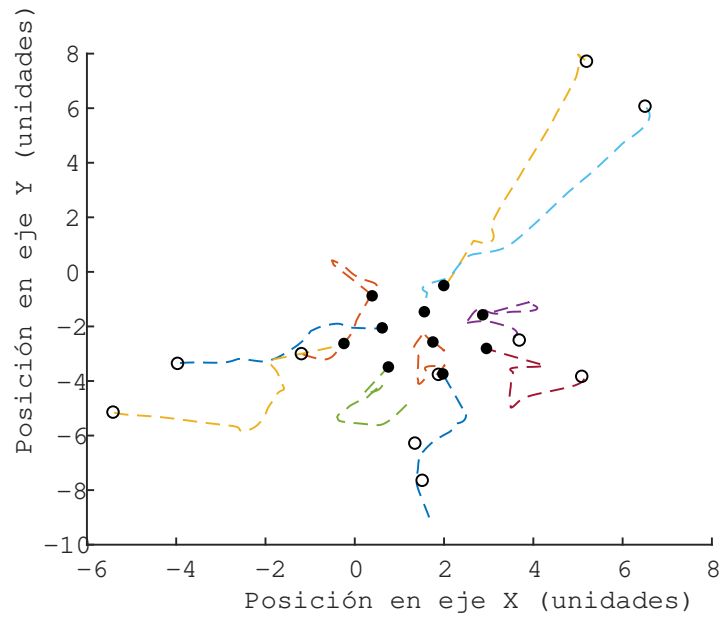


Figura 36: Trayectoria de los agentes al no lograr la formación del grafo de prueba 2 y una matriz mínimamente rígida con el modelo 4

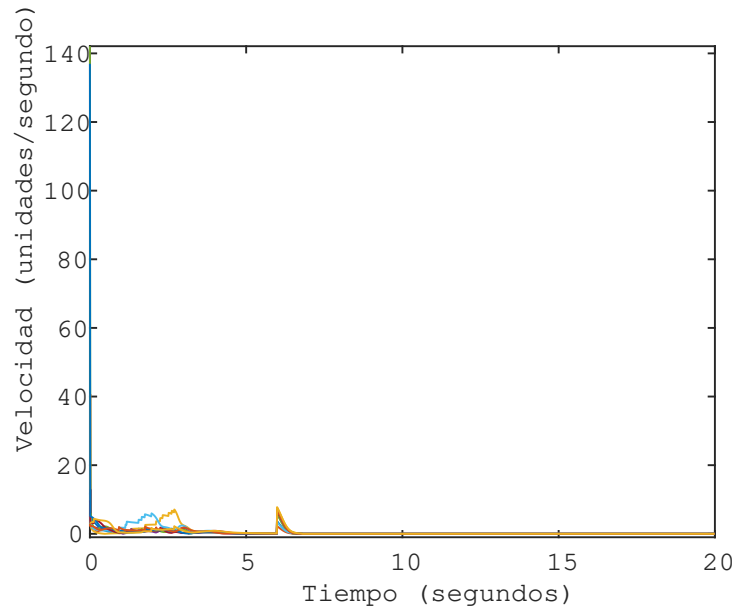


Figura 37: Velocidad de los agentes al no lograr la formación del grafo de prueba 2 y una matriz mínimamente rígida con el modelo 4

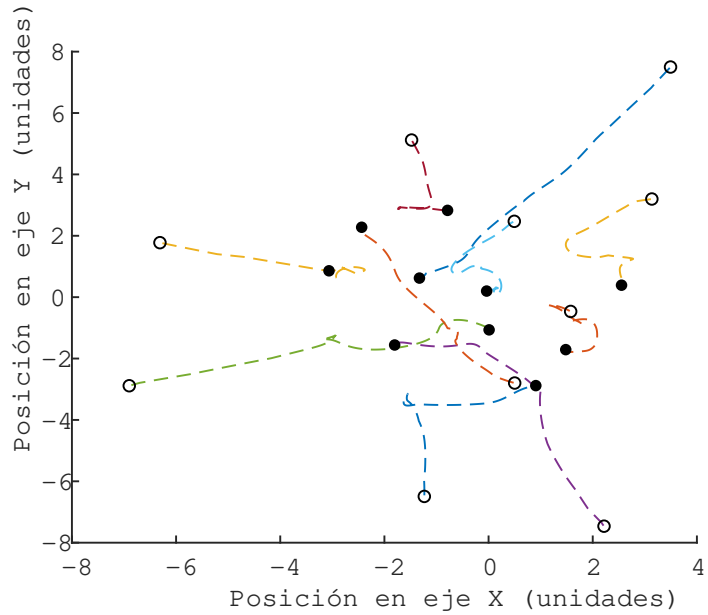


Figura 38: Trayectoria de los agentes al no lograr la formación del grafo de prueba 2 y una matriz totalmente rígida con el modelo 4

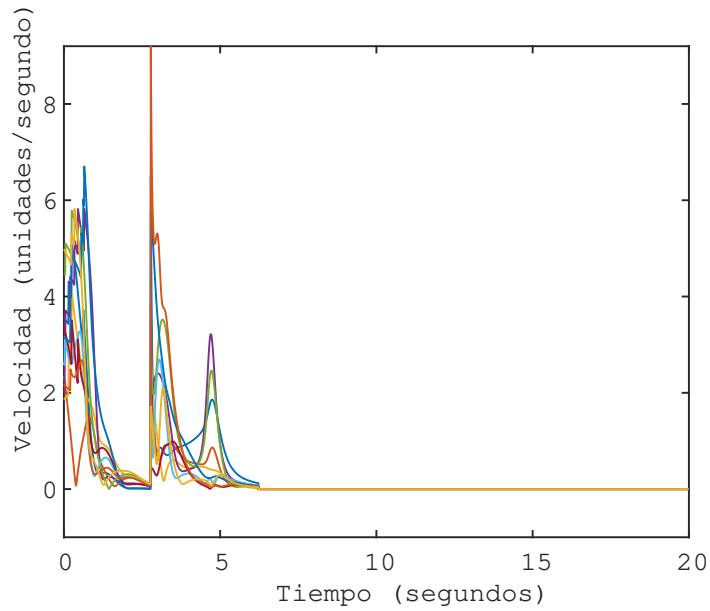


Figura 39: Velocidad de los agentes al no lograr la formación del grafo de prueba 2 y una matriz totalmente rígida con el modelo 4

7.4.5. Modelo 5

A pesar de haber obtenido mejoras con el modelo dinámico, se observó que todos los modelos anteriores presentaban complicaciones al utilizar grafos mínimamente rígidos. Dado que cada función de consenso modificada tiene un único mínimo bien definido, los agentes

buscan llegar a este mínimo. Esto implica que al aplicar el control de formación utilizando la matriz de adyacencia para definir las distancias entre los agentes, existirán parejas de nodos con un mínimo erróneo en una distancia de 0, porque no existe arista entre ellos. Ese mínimo en 0, forzará a la formación a disminuir la distancia entre agentes no conectados siendo únicamente el control para evitar colisiones el que lo impida. Entonces, se necesita tener una función “plana” que no fuerce a los agentes a un mínimo de cero, sino que permita que sea el control de formación el que decida dónde debe posicionarse.

El coseno hiperbólico cumple con ser esta función “plana”, ecuación (29), representada en la Figura 40. Así que se modificó el control de formación, de manera que cuando dos nodos no sean adyacentes, sea esta la función que influya en la velocidad del agente.

$$\varepsilon_{ij} = 0.01 \cosh(1.8 \|x_i - x_j\| - 8.4) \quad (29)$$

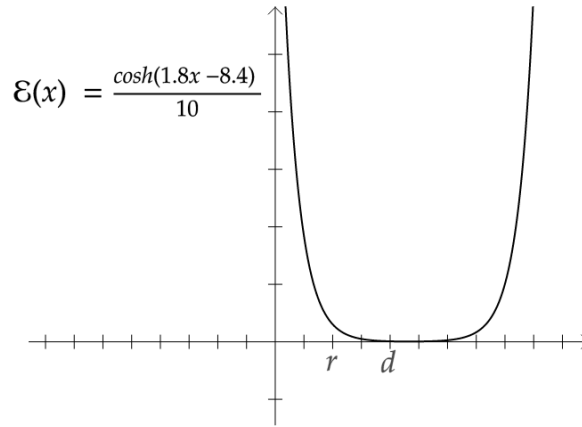


Figura 40: Función de la tensión para el control de formación entre agentes que no están conectados

Entonces, los pesos añadidos en la parte del modelo que incluye el control de formación están dados por:

$$\omega_{ij} = \begin{cases} d_{ij} = 0, & 0.018 \times \frac{\sinh(1.8\|x_i - x_j\| - 8.4)}{\|x_i - x_j\|} \\ d_{ij} \neq 0, & \frac{4(\|x_i - x_j\| - d_{ij})(\|x_i - x_j\| - r) - 2(\|x_i - x_j\| - d_{ij})^2}{(\|x_i - x_j\| - r)^2(\|x_i - x_j\|)} \end{cases} \quad (30)$$

Con este modelo se lograron entre 15% y 20% de formaciones exitosas utilizando grafos mínimamente rígidos. Al menos 20% de los casos logran el 70% de la formación. El éxito de las configuraciones con grafos totalmente rígidos no es tan diferente, entre el 20% y el 25%. A continuación, en las Figuras 41 a la 44 se muestran algunos de estos resultados.

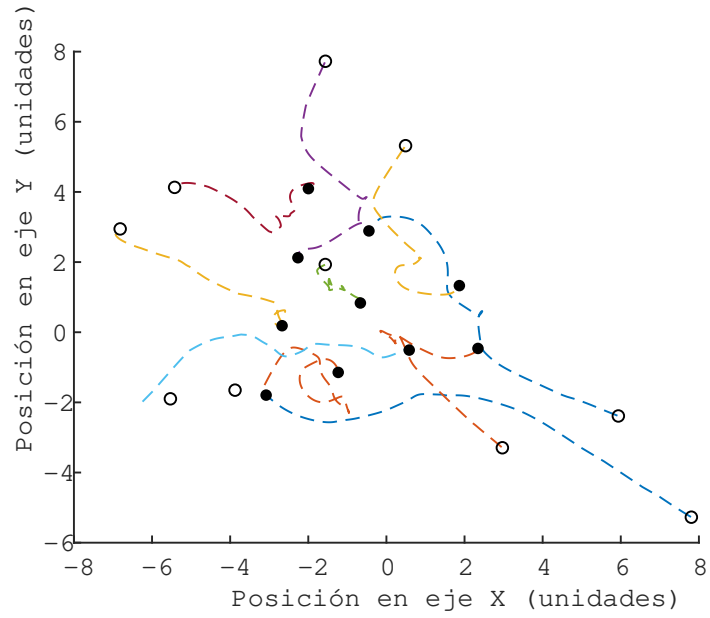


Figura 41: Trayectoria de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 5

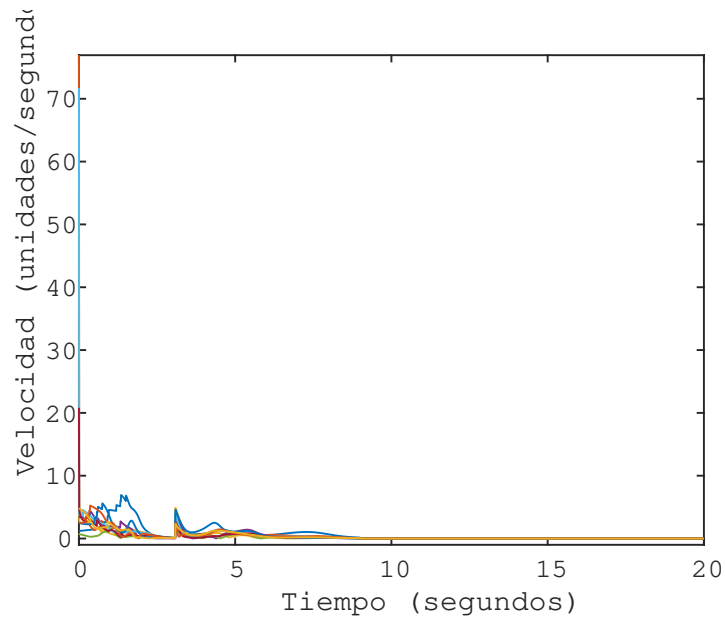


Figura 42: Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 5

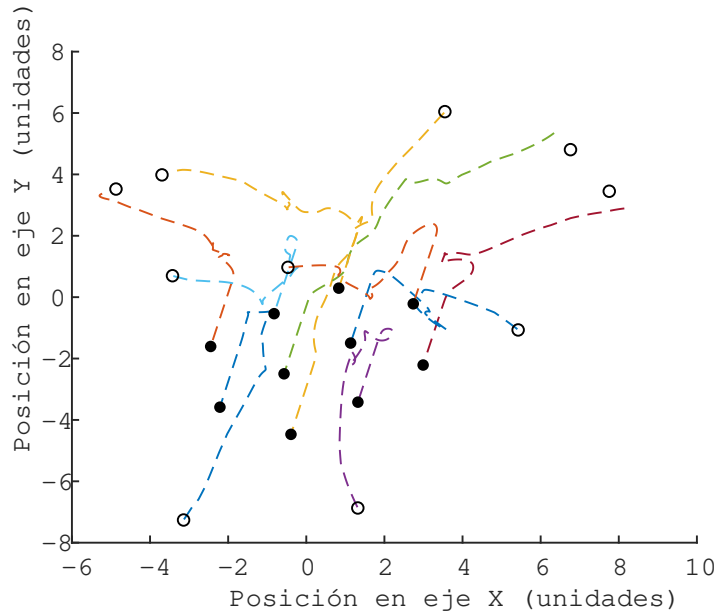


Figura 43: Trayectoria de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 5

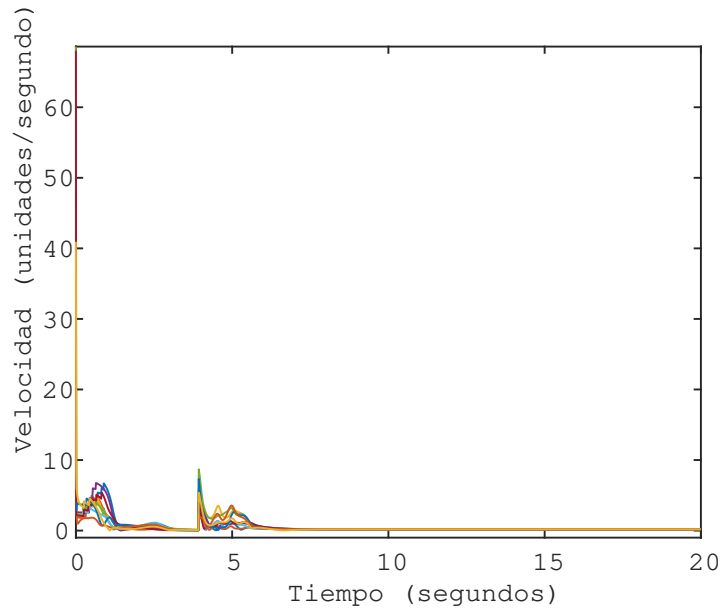


Figura 44: Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 5

En cuanto a la energía necesaria para que el sistema logre la formación, esta aumentó aproximadamente un 90% entre el modelo anterior y el presente. Esto implica que este modelo tal cuál no puede ser implementado en un entorno físico, ya que requeriría robots capaces de moverse a velocidades extremadamente altas. Por lo que se realizaron ajustes adicionales para reducir este parámetro de energía, pero a la vez lograr las formaciones con los grafos mínimamente rígidos.

7.4.6. Modelo 6

Se optó por incluir un límite de velocidad al modelo anterior. De esta manera cuando la velocidad dada por el modelo 5, sea mayor al límite establecido solo se tomará en cuenta su dirección. Se determina el vector unitario de la velocidad y luego se le impone la magnitud de la velocidad máxima. Esta modificación no se realiza directamente en la ecuación de tensión, es un control posterior.

Esta pequeña modificación resultó en grandes cambios según las métricas de energía establecidas. La energía promedio entre el modelo 5 y el modelo 6 se redujo en casi un 96% y el coeficiente de variación se redujo más de un 90%, Cuadro 4. Se muestran algunos gráficos de velocidad al utilizar grafos mínimamente rígidos, tanto de formaciones exitosas como de formaciones fallidas, Figuras 45 a la 52

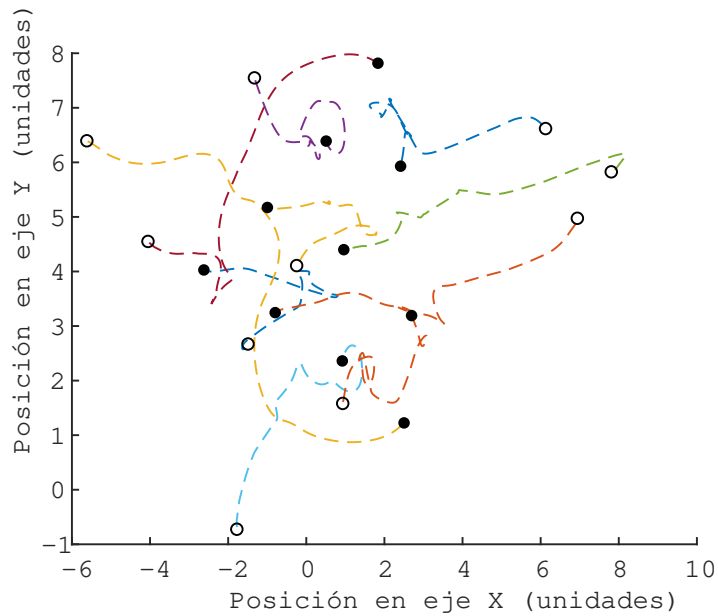


Figura 45: Trayectoria de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6

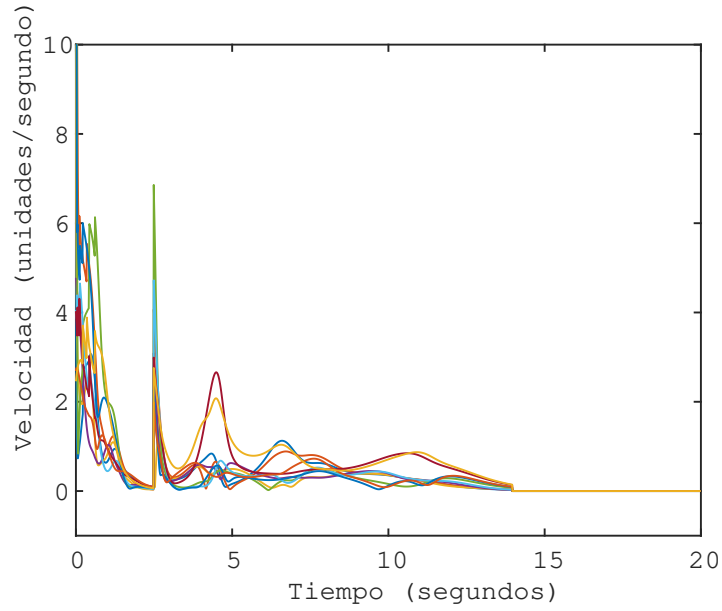


Figura 46: Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6

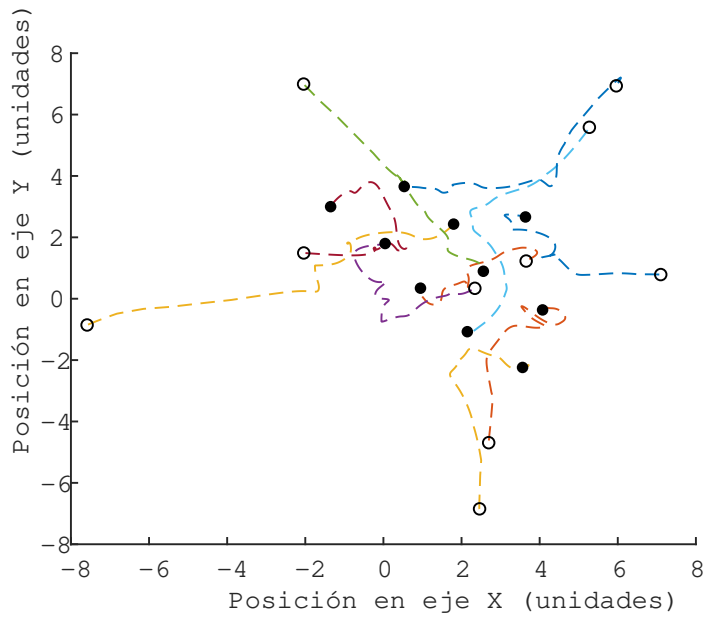


Figura 47: Trayectoria de los agentes, formación fallida con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6

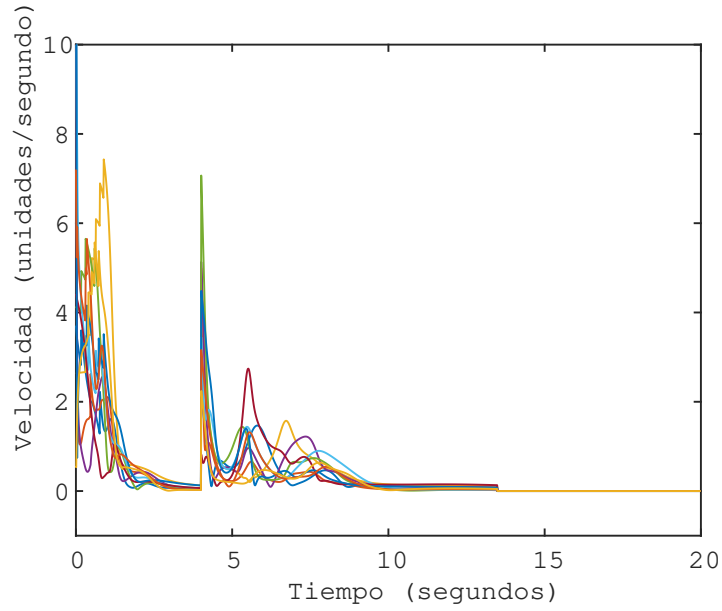


Figura 48: Velocidad de los agentes, formación fallida con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6

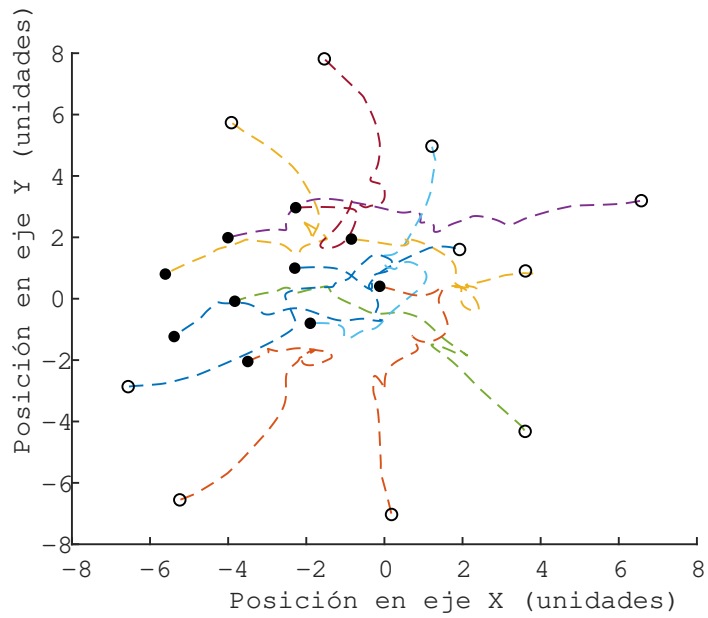


Figura 49: Trayectoria de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6

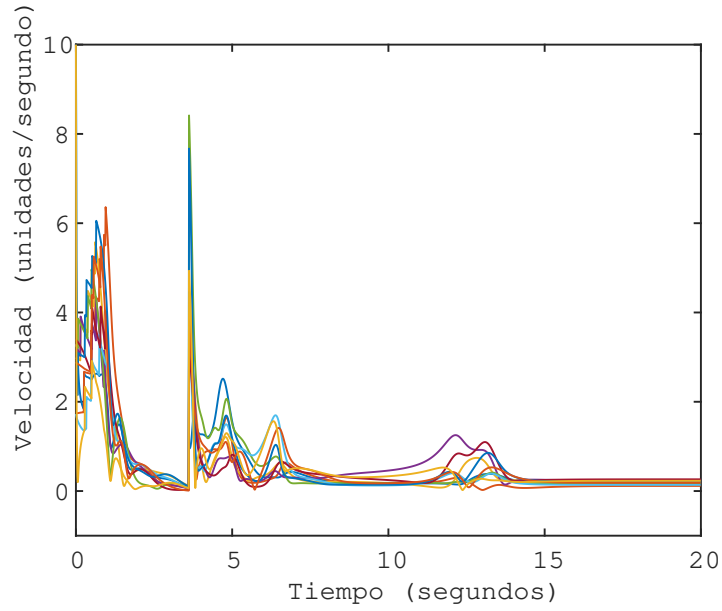


Figura 50: Velocidad de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6

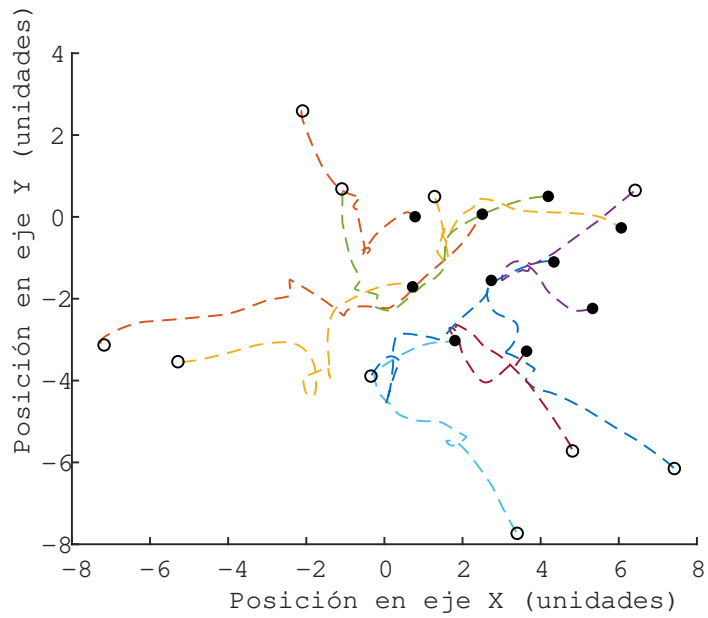


Figura 51: Trayectoria de los agentes, formación fallida con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6

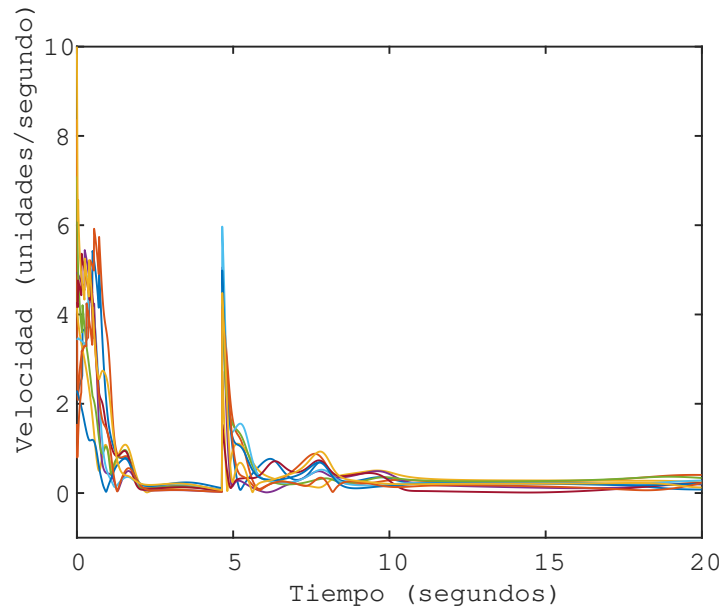


Figura 52: Velocidad de los agentes, formación fallida con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6

Modelo	Grafo	Rigidez	<0.001	<0.005	<0.01	<0.02	<0.04	<0.07
0	1	Mínima	0	0	0	0	0	1
0	1	Total	87	87	87	87	87	87
0	2	Mínima	0	0	0	0	0	0
0	2	Total	62	62	62	62	62	62
1	1	Mínima	0	1	1	2	4	4
1	1	Total	59	59	59	59	59	86
1	2	Mínima	0	0	0	0	0	0
1	2	Total	58	58	58	58	58	58
2	1	Total	0	0	0	0	1	1
2	2	Total	0	0	0	0	0	0
3	1	Mínima	0	0	0	0	0	0
3	1	Total	45	46	46	46	46	46
3	2	Mínima	0	0	0	0	0	0
3	2	Total	36	36	36	36	36	37
4	1	Mínima	0	0	0	0	0	0
4	1	Total	40	40	40	40	40	41
4	2	Mínima	0	0	0	0	0	0
4	2	Total	18	18	18	18	18	18
5	1	Mínima	0	0	4	12	12	13
5	1	Total	23	23	23	23	23	23
5	2	Mínima	0	11	11	11	11	20
5	2	Total	11	11	11	11	11	22
6	1	Mínima	0	0	3	10	10	10
6	1	Total	30	30	30	30	30	30
6	2	Mínima	0	13	13	13	13	25
6	2	Total	14	14	14	14	14	26

Cuadro 2: Error cuadrático medio de la formación. En este cuadro se muestra el porcentaje de simulaciones que tuvieron un error cuadrático medio menor al umbral indicado.

Modelo	Grafo	Rigidez	100 %	90 %	80 %	70 %	0 %
1	1	Mínima	0	2	4	6	88
1	1	Total	63	5	17	0	15
1	2	Mínima	0	1	3	0	96
1	2	Total	62	19	1	2	16
2	1	Total	2	0	1	1	96
2	2	Total	0	0	6	2	92
3	1	Mínima	0	0	0	0	100
3	1	Total	47	0	25	0	28
3	2	Mínima	0	0	0	0	100
3	2	Total	37	3	14	2	44
4	1	Mínima	0	0	0	0	100
4	1	Total	40	11	17	1	31
4	2	Mínima	0	0	0	0	100
4	2	Total	18	7	9	4	62
5	1	Mínima	13	0	0	0	87
5	1	Total	27	15	10	3	45
5	2	Mínima	11	3	7	1	78
5	2	Total	11	17	18	4	50
6	1	Mínima	16	13	3	1	67
6	1	Total	25	27	5	3	40
6	2	Mínima	13	1	6	1	79
6	2	Total	19	19	12	2	48

Cuadro 3: Porcentaje de formación exitosa. Este cuadro muestra el porcentaje de simulaciones en el que se logró el porcentaje de la formación indicado.

Modelo	Grafo	Rigidez	<60	<100	<200	Desviación	Promedio	C.V.
1	1	Mínima	38	95	98	2.00E+03	2.68E+02	7.445
1	1	Total	16	94	98	8.78E+01	8.40E+01	1.044
1	2	Mínima	20	98	98	7.24E+01	7.69E+01	0.942
1	2	Total	17	97	100	1.59E+01	7.18E+01	0.221
2	1	Total	0	6	21	1.13E+06	1.54E+05	7.344
2	2	Total	0	7	24	1.20E+07	1.31E+06	9.170
3	1	Mínima	55	100	100	8.76E+00	5.85E+01	0.150
3	1	Total	16	99	100	1.20E+01	7.12E+01	0.169
3	2	Mínima	58	100	100	8.72E+00	5.92E+01	0.147
3	2	Total	15	100	100	1.11E+01	7.20E+01	0.154
4	1	Mínima	20	83	96	9.07E+04	9.23E+03	9.825
4	1	Total	3	62	97	1.92E+02	1.22E+02	1.571
4	2	Mínima	15	87	95	5.49E+03	7.30E+02	7.526
4	2	Total	1	66	92	8.66E+03	1.39E+03	6.241
5	1	Mínima	2	52	96	1.04E+03	2.42E+02	4.309
5	1	Total	4	46	92	7.46E+02	2.27E+02	3.291
5	2	Mínima	0	16	91	1.79E+03	3.35E+02	5.351
5	2	Total	1	37	89	6.82E+02	2.66E+02	2.565
6	1	Mínima	7	67	100	1.64E+01	9.02E+01	0.181
6	1	Total	0	59	100	1.77E+01	9.68E+01	0.183
6	2	Mínima	1	25	100	1.45E+01	1.09E+02	0.133
6	2	Total	5	71	100	1.72E+01	9.01E+01	0.191

Cuadro 4: Energía del sistema. Este cuadro corresponde a las mediciones de energía de la simulaciones, se muestra el porcentaje de simulaciones con energía menor a los 3 parámetros establecidos. Además, se indica la desviación, el promedio y el coeficiente de variación de la energía en todas las simulaciones realizadas.

7.5. Cambio de formación ante obstáculos

La parte clave para el funcionamiento de este algoritmo de búsqueda está en que el enjambre de robots sea capaz de modificar su configuración al momento de encontrarse entre obstáculos que la deformen. Para ello se debe definir cómo se realizará la transición de una formación a otra y qué parámetros determinarán cuándo es necesario cambiar de configuración. También debe incluirse un control para evadir obstáculos como tal.

7.5.1. Control para evasión de obstáculos

Se comenzó por la propuesta del control para evadir obstáculos. A diferencia del control para evasión de colisiones utilizado para lograr la formación, este no está relacionado con la ecuación de consenso. Este es independiente, ya que no se busca arrastrar al enjambre hacia los obstáculos, únicamente se quiere que los evadan si se cruzan en su camino. Entonces, ahora la velocidad de cada agente se ve afectada por la distancia a la que se encuentre de

un obstáculo por la siguiente ecuación:

$$\omega = \frac{-0.01}{(d_{io} - 1)^2} \quad (31)$$

donde, d_{io} es la distancia entre el agente i y el obstáculo o .

Este control presentó los resultados deseados. Los agentes evaden el obstáculo exitosamente independientemente de la dirección desde la que se dirijan hacia él. En la Figura 53 se muestra un gráfico de las trayectorias de los agentes y se aprecia como rodean el obstáculo.

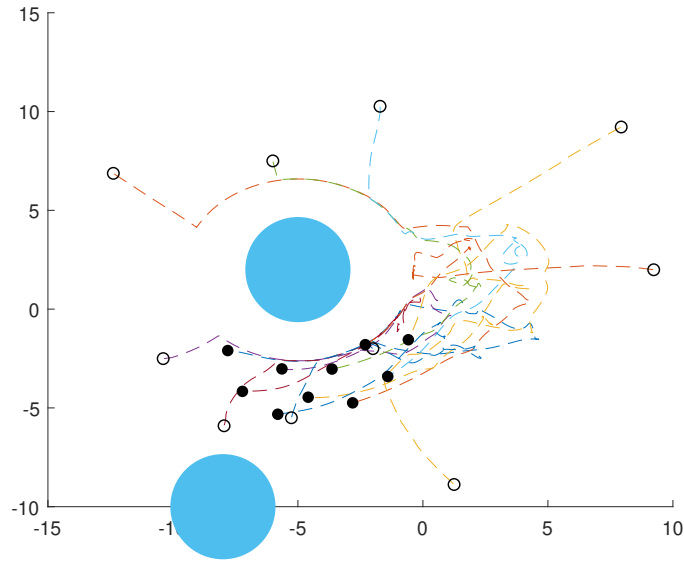


Figura 53: Trayectorias de los agentes al evadir obstáculos

7.5.2. Cambio de formación

Una vez definido el cómo se evadirán los obstáculos, se procede a la evaluación de si será necesario cambiar de formación. Esta decisión depende de qué tan alejada esté la configuración actual de la inicial. Así que, el parámetro que indicará esta diferencia será el error cuadrático entre las dos configuraciones a comparar. Al tener distintas formaciones posibles debe medirse el error entre la posición actual de los agentes y la posición de todos los grafos de las configuraciones disponibles. Aquel con el menor error, será el grafo elegido para que se reconfiguren los agentes y atraviesen los obstáculos. Se utilizaron tres arreglos de obstáculos diferentes para las pruebas, correspondientes a las Figuras 54, 55 y 56.

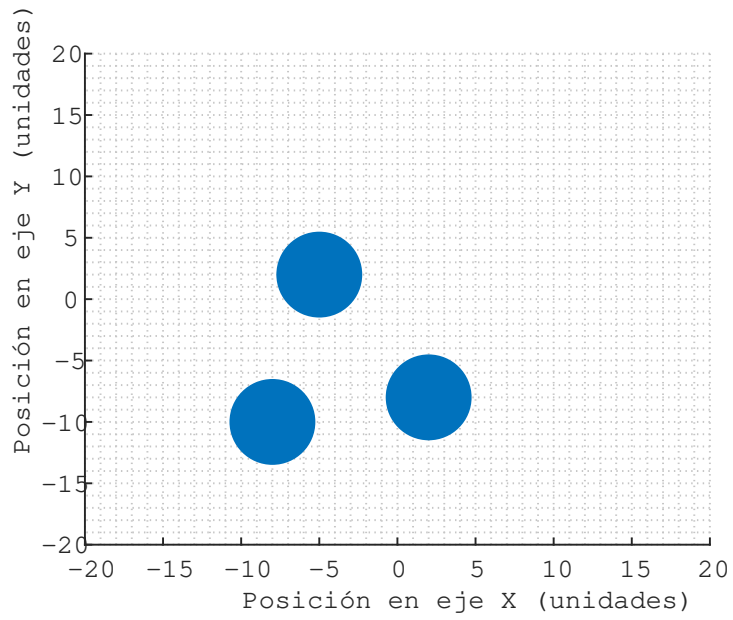


Figura 54: Primer arreglo de obstáculos para pruebas

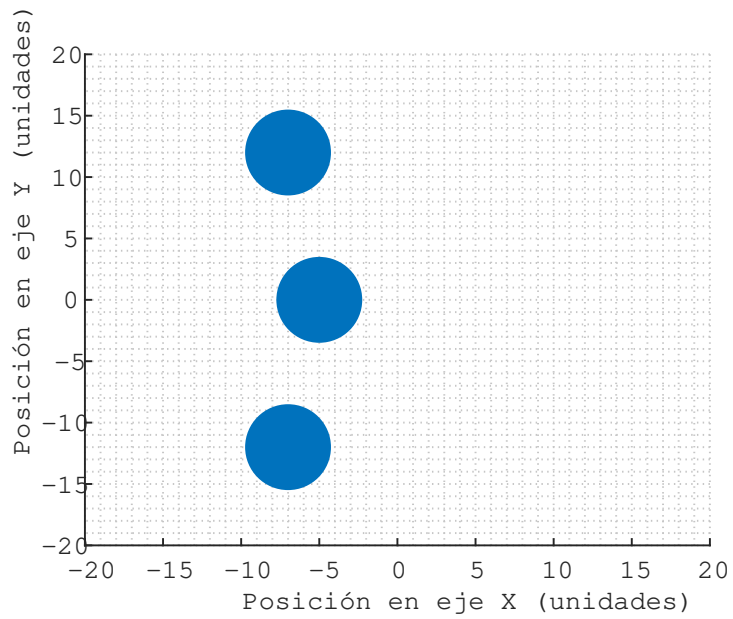


Figura 55: Segundo arreglo de obstáculos para pruebas

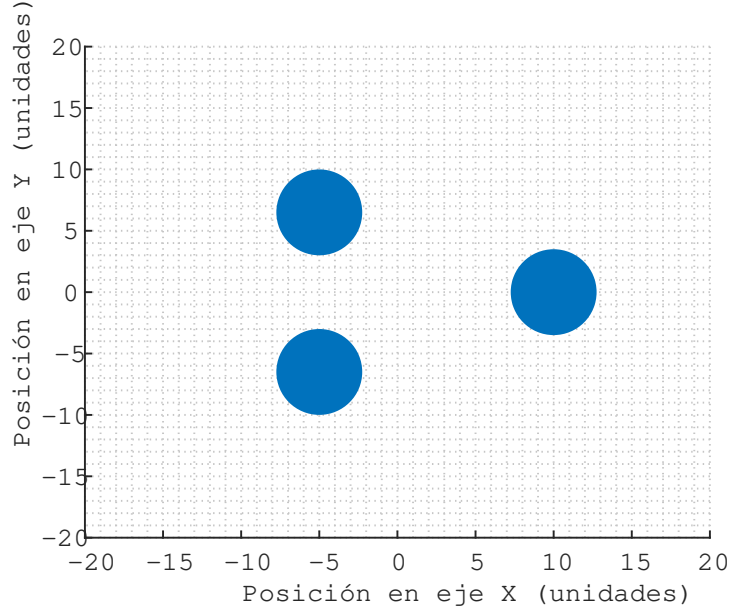


Figura 56: Tercer arreglo de obstáculos para pruebas

El identificar el grafo que represente el menor error no es suficiente para la selección de la formación. Puede que en el momento en el que los agentes se estén reconfigurando ese error varíe múltiples veces en un breve instante de tiempo. Esto ocasionaría que los agentes busquen lograr dos o más formaciones de manera intermitente. Así que el error de una nueva formación debe ser considerablemente menor o mantenerse por un largo espacio de tiempo como el “menor error” para indicarle al grupo que es momento de cambiar de configuración.

Se tienen las primeras pruebas haciendo variaciones con la cantidad de tiempo que los robots esperan que el error del grafo alternativo sea menor para decidir cambiar. En [20], se menciona una manera en la que podría calcularse el tiempo mínimo que debería esperarse antes de modificar la distribución de los agentes, esta expresión corresponde a la Ecuación 32. Haciendo uso de esta expresión se calcularon los tiempos correspondientes a cada grafo de prueba. Con ello se obtuvieron los resultados 33 y 34.

$$T_{conv} \geq \frac{1}{\lambda_2(L(G)) \ln\left(\frac{\kappa}{\varepsilon}\right)} \quad (32)$$

donde, $\lambda_2(L(G))$ representa el segundo eigenvalor mayor del Laplaciano del grafo de formación, κ es la máxima diferencia que puede existir entre el error de formación individual de cada agente contra el error global en el instante 0 y ε es la máxima diferencia entre el error individual y el error global en un instante muy cercano a la convergencia.

$$T_{conv1} = 0.509 \quad (33)$$

$$T_{conv2} = 0.437 \quad (34)$$

Sin embargo, ese caso contempla la aplicación del control de formación, definido en 7, sin ninguna modificación. Si bien esto presenta una idea de dónde comenzar, no es una fiel

referencia, ya que que la combinación de controles retrasa considerablemente el proceso de formación, especialmente la utilización de la función de coseno hiperbólica. Por ello se les aplicó un factor de 10 a los tiempos calculados en 33 y 34. Adicional a este factor, se optó por realizar pruebas con 3 tiempos distintos para observar la influencia de este factor en el éxito del algoritmo. Los tiempos seleccionados para las pruebas fueron: 3, 5 y 7 segundos. En las Figuras 57 a la 77, se muestran algunos de los resultados obtenidos con los tres tiempos de prueba utilizando grafos mínimamente rígidos y totalmente rígidos. Las líneas verticales discontinuas en los gráficos de error de formación indican el momento en el que se llevó a cabo el cambio de configuración en la simulación.

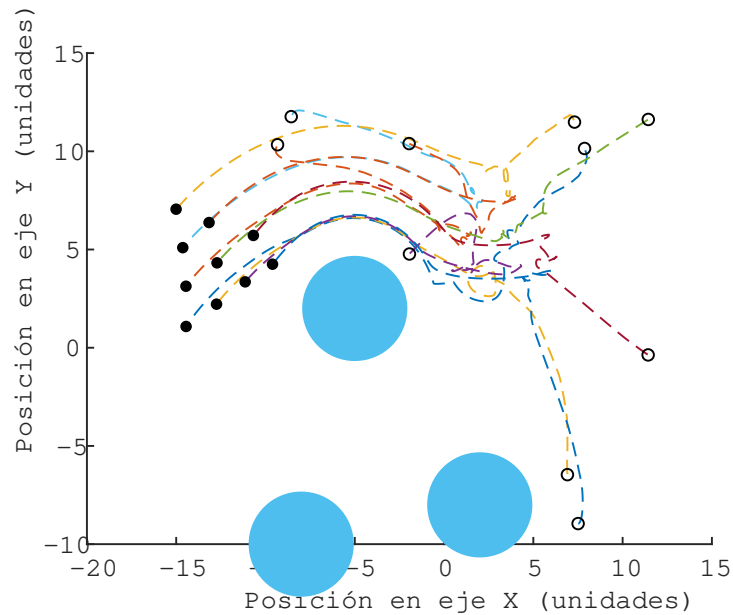


Figura 57: Trayectorias de los agentes, formación inicial exitosa con grafos mínimamente rígidos y llegada a la meta exitosa en el primer arreglo de obstáculos con el tiempo de prueba de 3 segundos

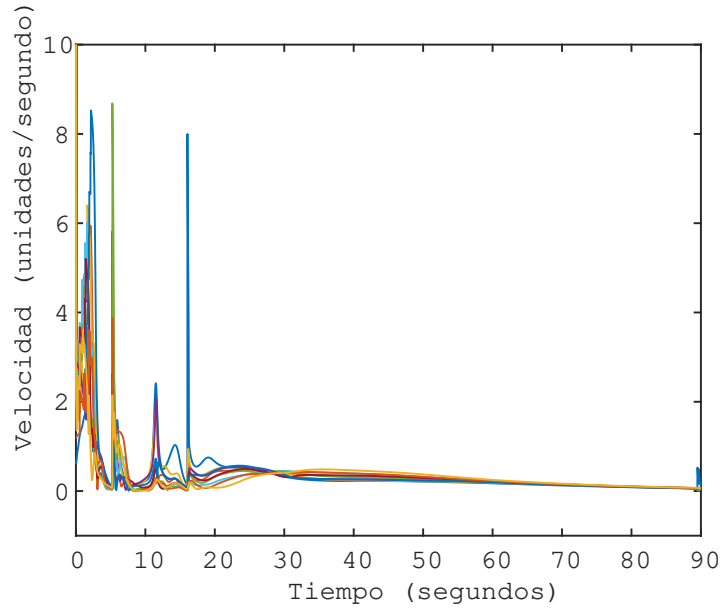


Figura 58: Velocidades de los agentes, formación inicial exitosa con grafos mínimamente rígidos y llegada a la meta exitosa en el primer arreglo de obstáculos con el tiempo de prueba de 3 segundos

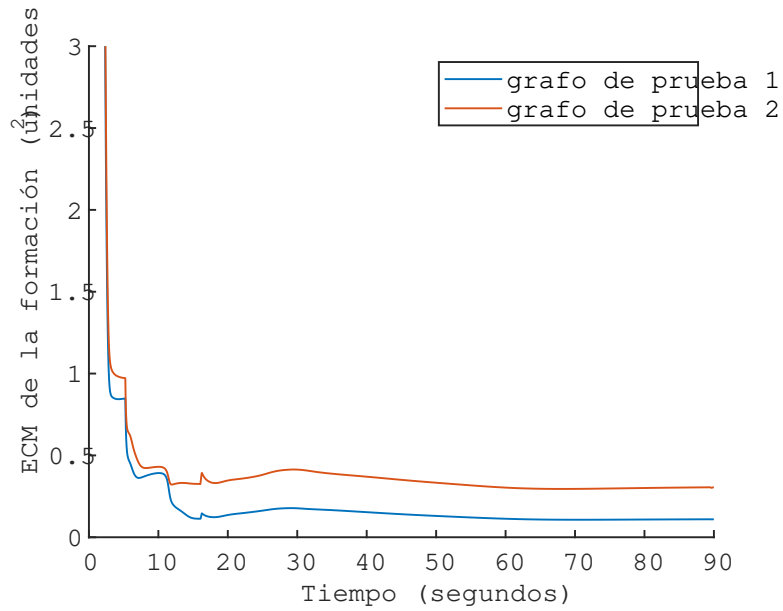


Figura 59: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa con grafos mínimamente rígidos, sin cambios de formación en el primer arreglo de obstáculos con el tiempo de prueba de 3 segundos

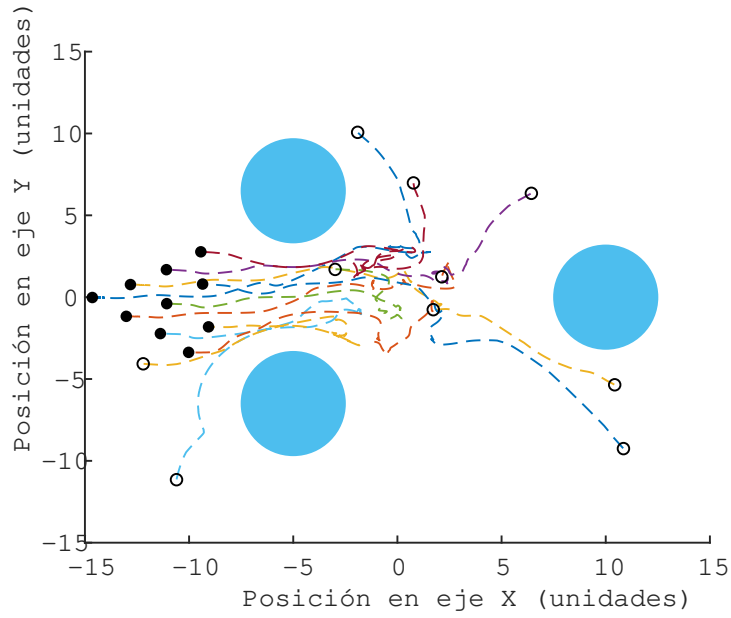


Figura 60: Trayectorias de los agentes, formación inicial exitosa que sufre deformación con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos

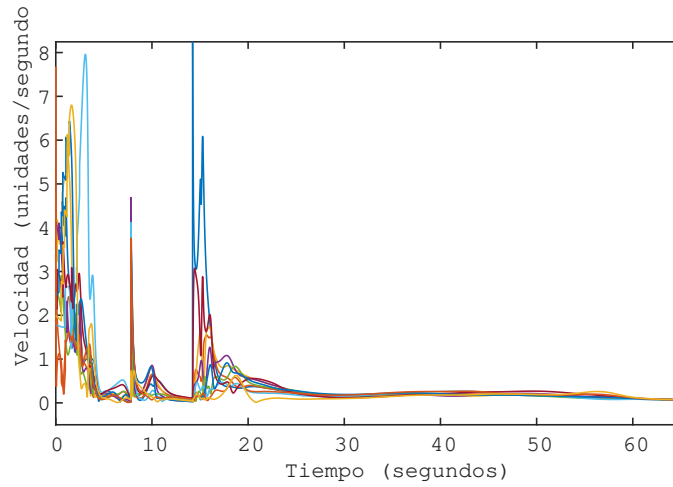


Figura 61: Velocidades de los agentes, formación inicial exitosa que sufre deformación con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos

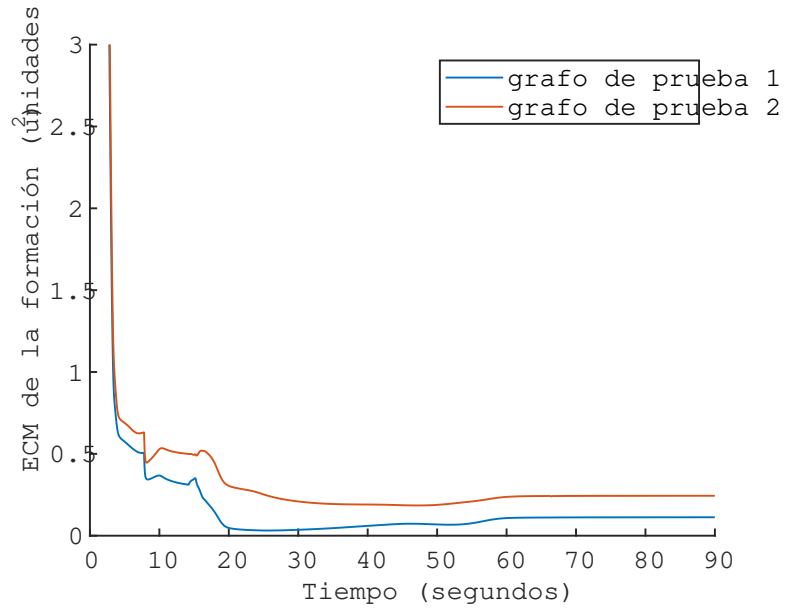


Figura 62: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa que sufre deformación con grafos mínimamente rígidos, sin cambios de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos

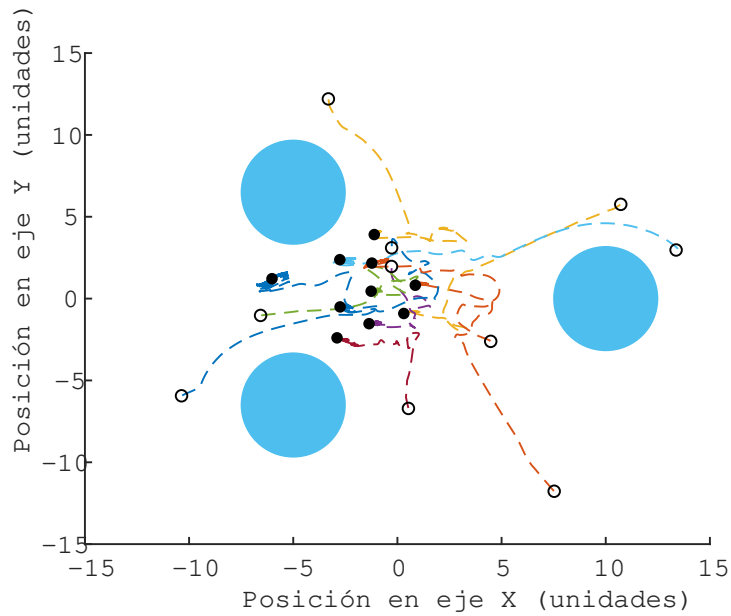


Figura 63: Trayectorias de los agentes, formación inicial fallida con grafos totalmente rígidos y llegada a la meta fallida en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos

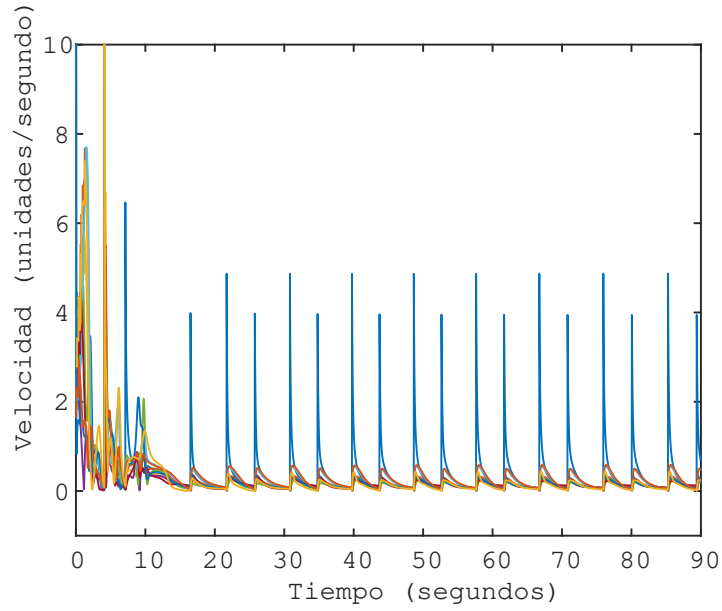


Figura 64: Velocidades de los agentes, formación inicial fallida con grafos totalmente rígidos y llegada a la meta fallida en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos

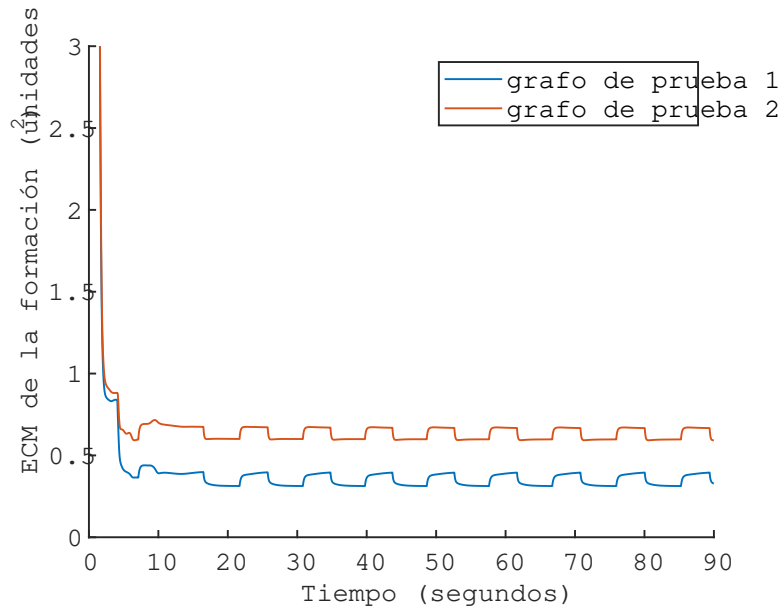


Figura 65: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial fallida con grafos totalmente rígidos, sin cambios de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 3 segundos

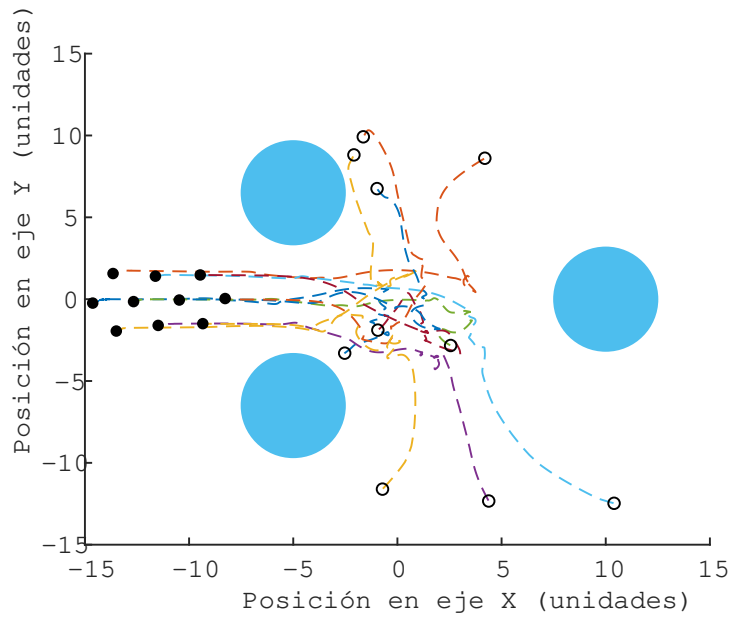


Figura 66: Trayectorias de los agentes, formación inicial fallida con éxito final con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 5 segundos

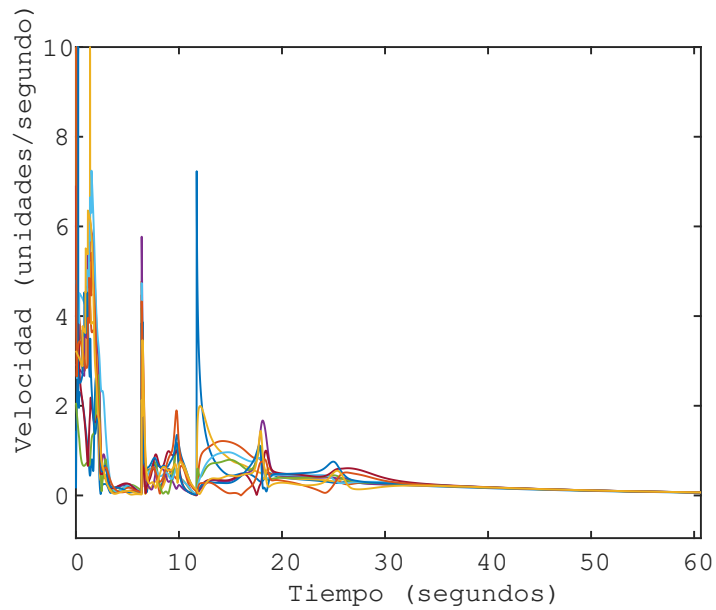


Figura 67: Velocidades de los agentes, formación inicial fallida con éxito final con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 5 segundos

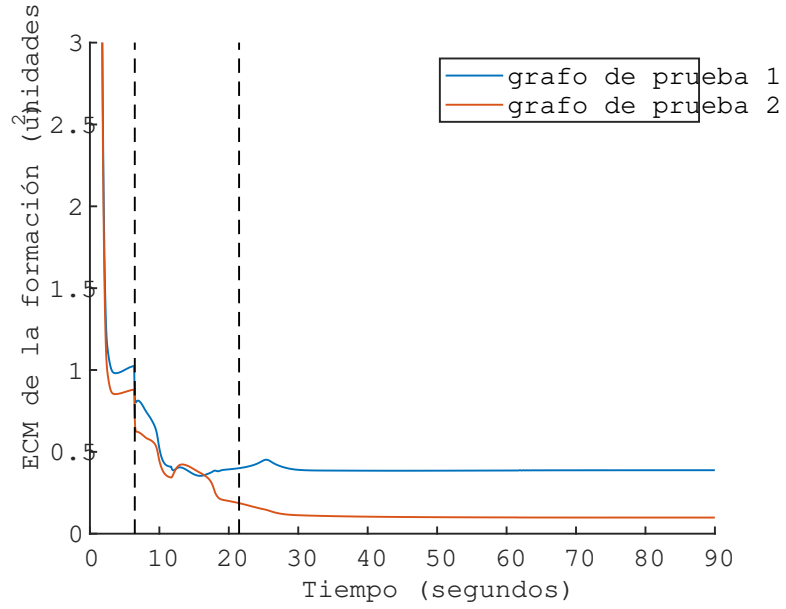


Figura 68: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial fallida con éxito final usando grafos mínimamente rígidos, con dos cambios de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 5 segundos

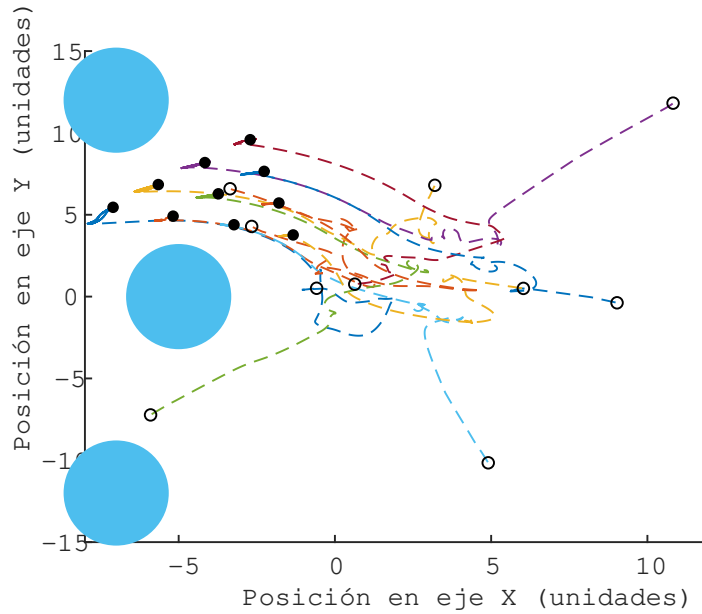


Figura 69: Trayectorias de los agentes, formación inicial exitosa con grafos totalmente rígidos y llegada a la meta fallida en el segundo arreglo de obstáculos con el tiempo de prueba de 5 segundos

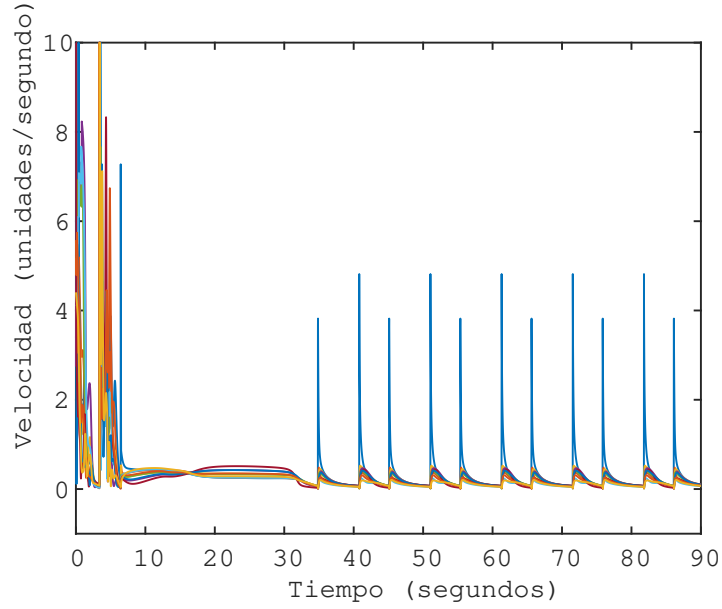


Figura 70: Velocidades de los agentes, formación inicial exitosa con grafos totalmente rígidos y llegada a la meta fallida en el segundo arreglo de obstáculos con el tiempo de prueba de 5 segundos

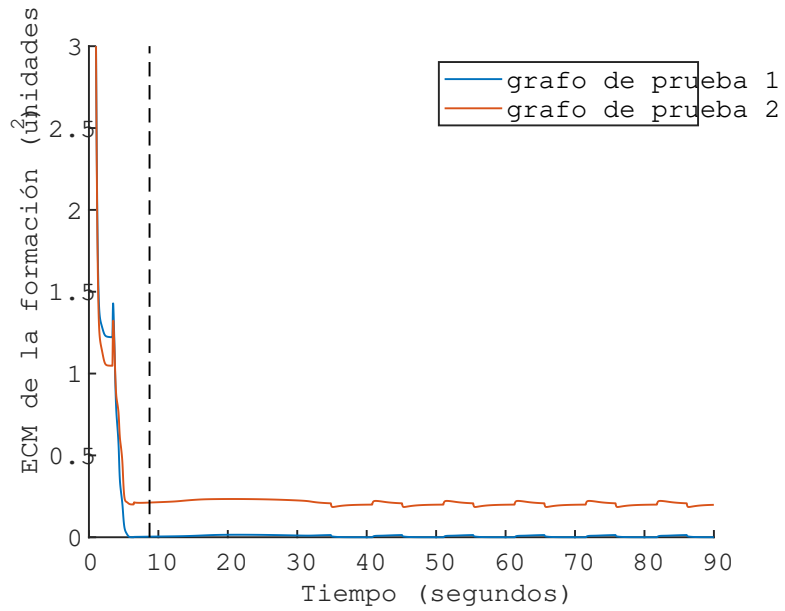


Figura 71: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa usando grafos totalmente rígidos, con un cambio de formación en el segundo arreglo de obstáculos con el tiempo de prueba de 5 segundos

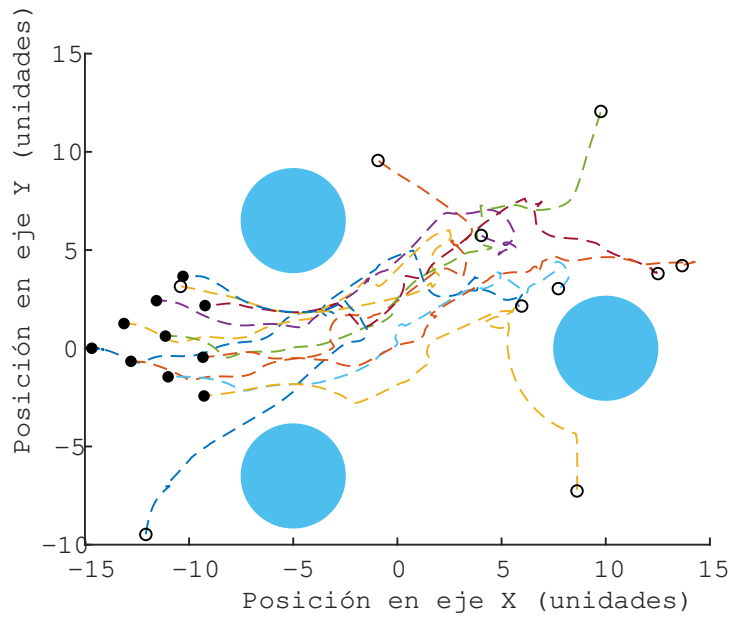


Figura 72: Trayectorias de los agentes, formación inicial exitosa y un agente fuera del radar, con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos

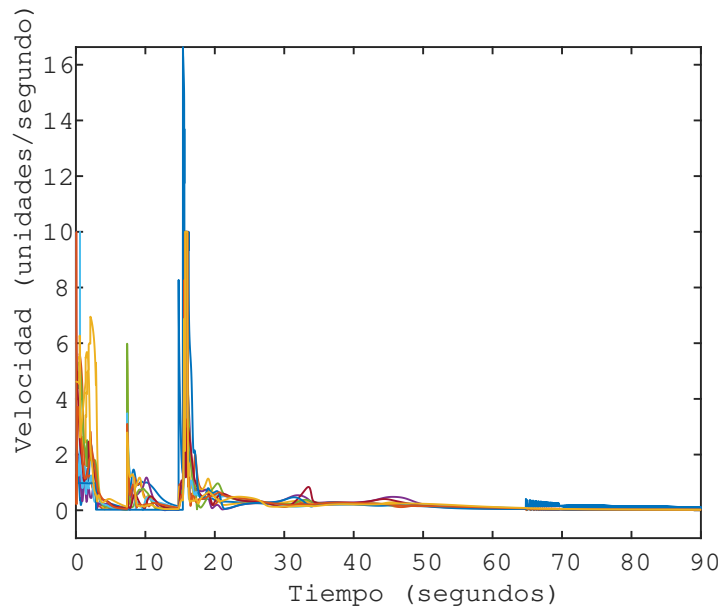


Figura 73: Velocidades de los agentes, formación inicial exitosa y un agente fuera del radar, con grafos mínimamente rígidos y llegada a la meta exitosa en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos

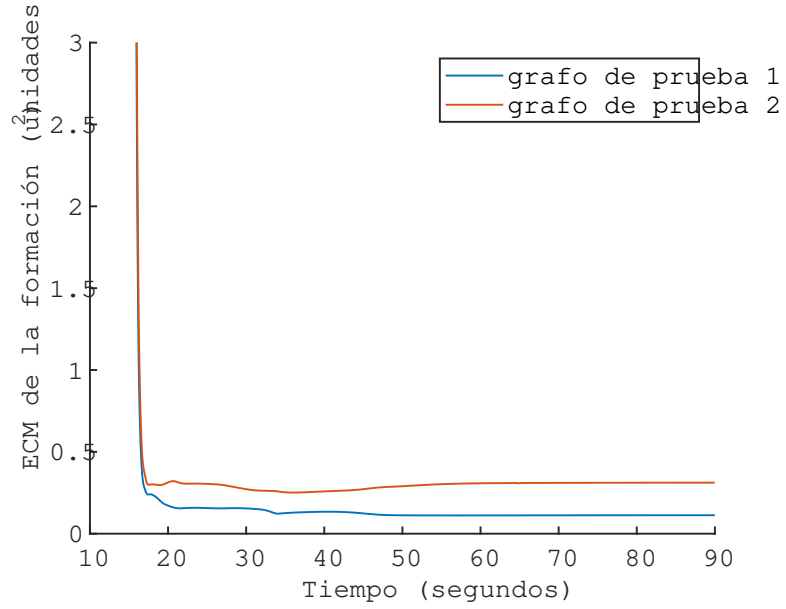


Figura 74: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa y un agente fuera del radar usando grafos mínimamente rígidos, sin cambios de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos

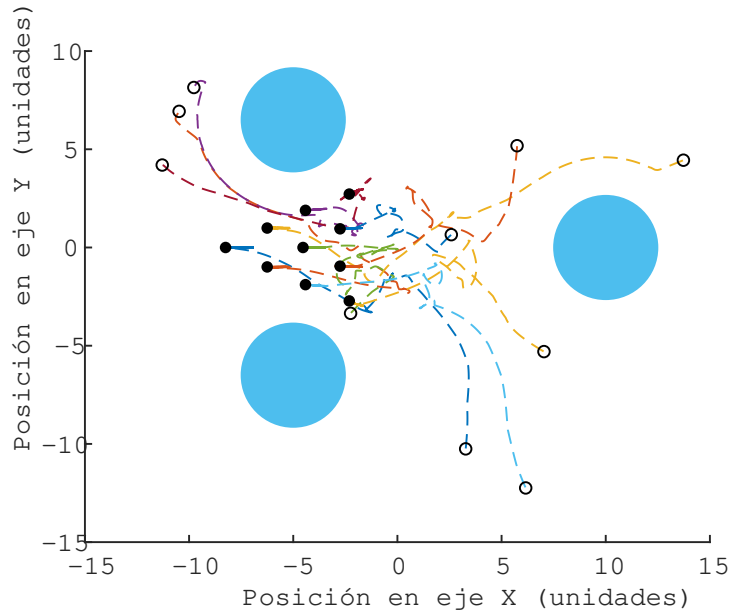


Figura 75: Trayectorias de los agentes, formación inicial exitosa, con grafos totalmente rígidos y llegada a la meta fallida en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos

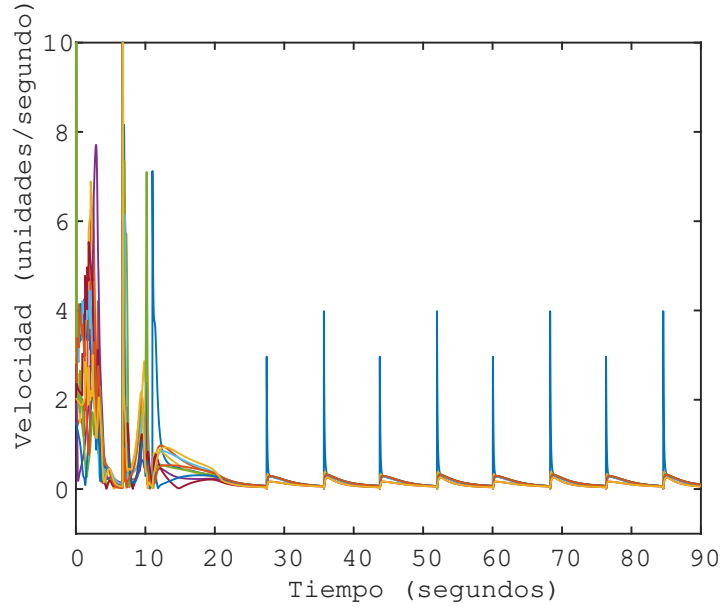


Figura 76: Velocidades de los agentes, formación inicial exitosa, con grafos totalmente rígidos y llegada a la meta fallida en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos

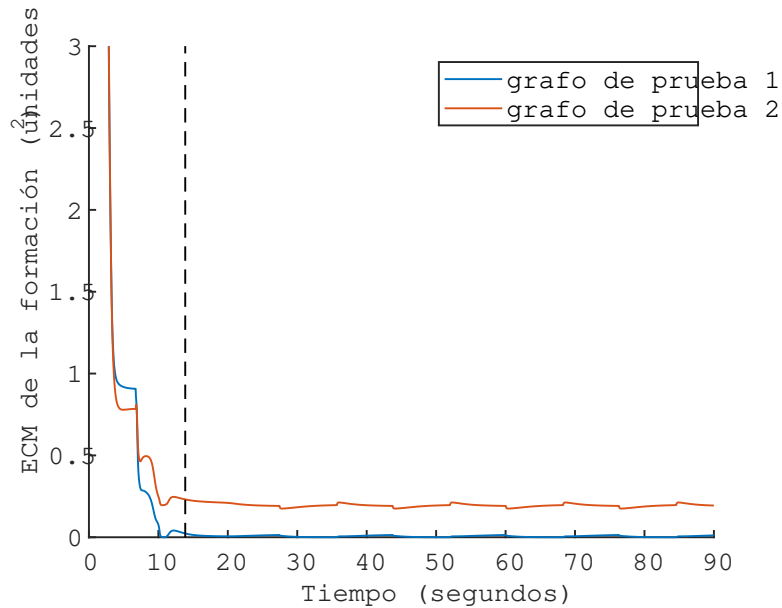


Figura 77: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa usando grafos totalmente rígidos, con un cambio de formación en el tercer arreglo de obstáculos con el tiempo de prueba de 7 segundos

Estos resultados permiten hacer varias observaciones de los diferentes casos que pueden presentarse al momento de aplicar el algoritmo de control. La primera observación es con respecto a la rigidez del grafo de formación. En la mayoría de los casos en los que se utilizaron grafos totalmente rígidos los agentes lograban la formación inicial, como en el caso de las Figuras 69 y 75, pero no tenían éxito en llegar a la meta. Esto se debe a que la velocidad

impuesta por el líder sobre la formación para movilizarse no era suficiente para atraer al resto de los agentes y lograr atravesar los obstáculos. El tener un control de formación tan rígido o con muchas restricciones, no le permitía al enjambre deformarse y por lo mismo no podía reconfigurarse. Este intento del líder por mover al resto del grupo se aprecia mejor en los gráficos de velocidad de las Figuras 64, 70 y 76, donde la línea azul representa la velocidad del líder que tiene cambios bruscos y luego regresa a 0. Es aquí donde cobra real importancia la utilización de grafos mínimamente rígidos, ya que estos permiten una ligera deformación en la configuración para dar paso a una nueva formación. Solo de esta manera es posible que los agentes atraviesen los obstáculos.

Ahora, particularmente con las formaciones mínimamente rígidas, se tiene el caso ideal en el que el grupo de robots logra una formación inicial exitosa y luego de una transformación finaliza con otra formación exitosa. Adicionalmente se tienen otros tres posibles resultados principales: una formación inicial exitosa sin necesidad de reconfiguración, Figuras 57 y 65; una formación inicial exitosa que se deforma al intentar reconfigurarse, Figuras 60 y 62; o una formación inicial fallida que se corrige en la reconfiguración, Figuras 66 y 68. La primera situación se suele presentar cuando el líder guía al enjambre por un camino holgado, donde puede avanzar libremente independientemente de su formación. El que la formación sea exitosa en un inicio y luego se deforme sin reconfigurarse sucede cuando el error de la deformación siguió siendo más pequeño que el error contra la formación alternativa. Y el último caso depende de qué tan alejada se encuentre la configuración inicial de una formación exitosa y qué tanto impacto tengan los obstáculos en su deformación.

En general, para los tres arreglos de obstáculos seleccionados los agentes no presentaron más de dos cambios de formación. En su mayoría sucede un único cambio de formación, ya sea al inicio de la simulación, Figura 71, o al momento de atravesar los obstáculos, Figura 77. En los casos de doble cambio de configuración, generalmente una transformación sucede al inicio de la simulación y otra al atravesar los obstáculos, Figura 68.

Los gráficos anteriores permitieron realizar observaciones principalmente cualitativas del algoritmo de evasión de obstáculos y cambio de formación. Sin embargo, esto no es suficiente para poder determinar cuál de los tres tiempos de pruebas es el indicado. Entonces se realiza un análisis cuantitativo utilizando las métricas descritas en la sección 7.2. Los resultados de este análisis se presentan en los Cuadros 5, 6 y 7.

Con dichas pruebas se determinó que el tiempo de espera previo a realizar el cambio de formación que brinda los mejores resultados es el de 7 segundos. Se llegó a esta conclusión tomando como aspecto más importante el porcentaje de simulaciones en el que el enjambre llegaba a la meta. Si se habla de misiones de búsqueda, lo más importante es que el grupo de robots llegue al punto establecido para encontrar el objetivo. El segundo factor al que se le dio prioridad fue al error relativo, se espera que a medida que el sistema multiagente avance este logre una mejor formación si es que en un inicio no fue exitosa. El tiempo de espera de 7 segundos presentó considerablemente menos fallos, es decir, casos en los que no se lograba converger a una formación inicial, comparado con las simulaciones con los otros dos tiempos. Además, la cantidad de formaciones finales exitosas indica en cuántos casos el sistema fue realmente capaz de reconfigurarse. Los resultados para estos tres aspectos se encuentran en los cuadros 5 y 6. El tiempo de espera de 7 segundos presentó considerablemente menos fallos, es decir, casos en los que no se lograba converger a una formación inicial, comparado con las simulaciones con los otros dos tiempos.

Tiempo (s)	Rígidez	Arreglo	Inicial	Final
3	Mínima	1	12	8
3	Mínima	2	14	12
3	Mínima	3	10	20
3	Total	1	28	42
3	Total	2	24	40
3	Total	3	24	40
5	Mínima	1	6	8
5	Mínima	2	10	12
5	Mínima	3	12	14
5	Total	1	32	52
5	Total	2	28	44
5	Total	3	32	46
7	Mínima	1	16	14
7	Mínima	2	20	6
7	Mínima	3	18	18
7	Total	1	38	48
7	Total	2	36	50
7	Total	3	0	46

Cuadro 5: Éxito de las formaciones iniciales y formaciones finales. En este cuadro se muestra el porcentaje de simulaciones que tuvieron éxito en lograr la formación inicial y las que tuvieron formaciones finales exitosas. Entendido como éxito el tener un error cuadrático medio menor a 0.07.

Además de la selección del tiempo de espera, lo evaluado en los cuadros vuelve a resaltar la importancia de utilizar grafos mínimamente rígidos. Puesto que con estas configuraciones la cantidad de simulaciones en las que se llega a la meta se reduce. Aunque el beneficio que presentaron estos grafos fue que en todos los casos se converge a una formación, no necesariamente exitosa, pero esto le permite al grupo de robots comenzar a moverse hacia la meta.

Tiempo (s)	Rígidez	Arreglo	Fallo	No meta	Meta	ER < 1	ER > 1
3	Mínima	1	40	6	54	34	26
3	Mínima	2	46	12	42	32	22
3	Mínima	3	42	12	46	44	14
3	Total	1	0	62	38	44	56
3	Total	2	0	54	46	54	46
3	Total	3	0	40	60	44	56
5	Mínima	1	42	22	36	44	14
5	Mínima	2	38	18	44	44	18
5	Mínima	3	28	10	62	50	22
5	Total	1	0	50	50	56	44
5	Total	2	0	58	42	52	48
5	Total	3	0	46	54	40	60
7	Mínima	1	32	20	48	48	20
7	Mínima	2	18	10	72	42	40
7	Mínima	3	26	2	72	34	40
7	Total	1	0	66	34	50	50
7	Total	2	0	70	30	30	70
7	Total	3	0	40	60	46	54

Cuadro 6: Llegadas exitosas a la meta y error relativo de formación final e inicial. En este cuadro se muestra el porcentaje de simulaciones que tuvieron éxito en llegar a la meta (Meta), las que lograron converger a una formación pero no llegaron a la meta (No meta) y aquellas que no lograron converger a ninguna formación inicial (Fallo). Además se muestra el porcentaje de simulaciones con un error relativo menor a uno, lo cuál implica que el error de la formación final es menor que el error de la formación inicial; y el porcentaje de casos con error relativo mayor a uno que se refiere a la situación inversa.

Tiempo (s)	Rígidez	Arreglo	0	1	2
3	Mínima	1	26	66	8
3	Mínima	2	14	72	14
3	Mínima	3	16	72	10
3	Total	1	38	52	10
3	Total	2	26	66	8
3	Total	3	30	64	4
5	Mínima	1	10	82	8
5	Mínima	2	14	80	6
5	Mínima	3	16	82	2
5	Total	1	12	82	6
5	Total	2	26	72	2
5	Total	3	24	74	2
7	Mínima	1	20	74	6
7	Mínima	2	28	66	4
7	Mínima	3	36	60	4
7	Total	1	36	62	2
7	Total	2	32	64	2
7	Total	3	30	66	4

Cuadro 7: Cambios de formación. En este cuadro se presenta el porcentaje de simulaciones que realizaron 0, 1 y 2 cambios de formación.

Implementación en el entorno físico de pruebas

El algoritmo final desarrollado gracias al proceso iterativo, fue implementado en el entorno de simulación Webots de Cyberbotics. Este simulador permite modelar los parámetros físicos de los robots. Para estas simulaciones se decidió utilizar el robot E-puck de GCTronic, por ser uno de los robots incluidos en Webots con mayor documentación y que cumplían con ser lo suficientemente pequeños para la aplicación de robótica multiagente.

8.1. Simulaciones

Luego de incluir las transformaciones necesarias al modelo para que pudiese ser implementado por los robots del simulador, se procedió con las pruebas. Al igual que para el proceso de construcción del algoritmo, estas pruebas se dividieron en dos partes: éxito de la formación y cambio de formación. Se utilizaron las mismas métricas de evaluación para poder realizar una comparación entre el modelo teórico y el físico.

8.1.1. Modificaciones adicionales

Para que el algoritmo diseñado funcionase correctamente en la simulación de Webots, fue necesario modificar la ecuación de pesos que utiliza el coseno hiperbólico. Esto se debe a la transición de unidades utilizadas en Matlab a los metros utilizados por Webots. Con esta función en específico no puede únicamente hacerse un escalamiento ya que se debe modificar la región plana que abarca y el corrimiento horizontal. La función utilizada en este simulador se presenta en 35.

$$\varepsilon_{ij} = 0.01 \cosh(15 \|x_i - x_j\| - 8) \quad (35)$$

Además, en la implementación se decidió cuadruplicar el tiempo de espera entre los cambios de formación debido a que en este simulador tomaba más tiempo el converger a una formación. Esta diferencia en el tiempo también se ve influenciada por la modificación en la ecuación de coseno hiperbólico. Para determinar este factor de 4, se compararon los tiempos de finalización de algunas simulaciones teóricas con las simulaciones en Webots.

8.1.2. Éxito de la formación

Cómo primer punto se llevaron a cabo las simulaciones aplicando únicamente el modelo 6 adaptado a los robots E-Puck del simulador Webots. Estas pruebas permitieron realizar una comparación entre los resultados del modelo matemático ideal y la implementación física en un entorno controlado. Se muestran algunos gráficos de dichas simulaciones en las Figuras 78 a la 93 y en los Cuadros 8, 9 y 10 se presentan los resultados de evaluar las mismas métricas de las simulaciones en Matlab.

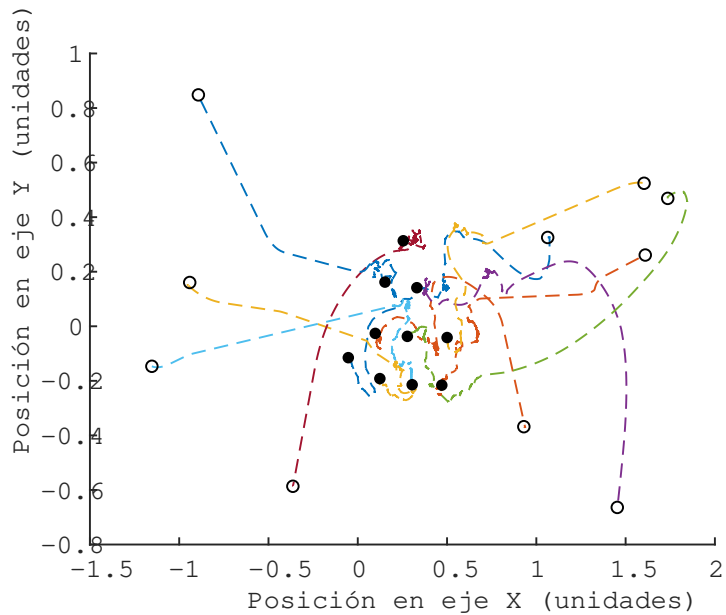


Figura 78: Trayectoria de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots

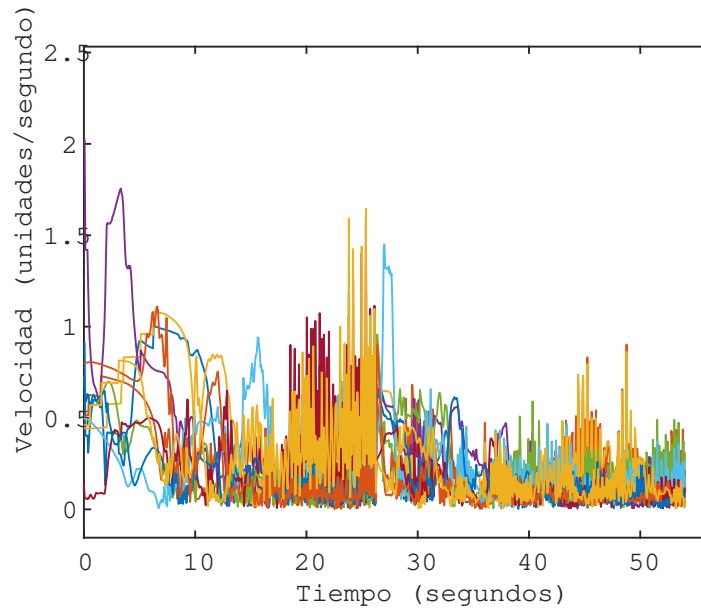


Figura 79: Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots

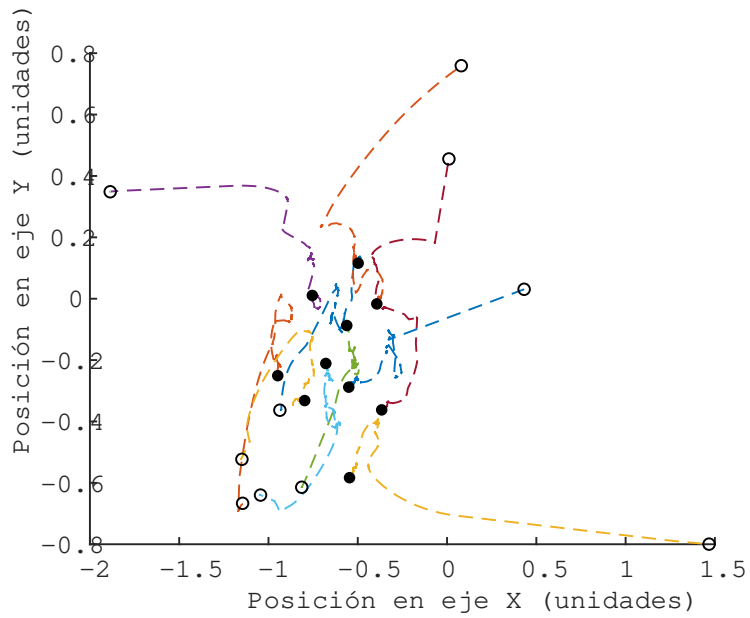


Figura 80: Trayectoria de los agentes, formación fallida con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots

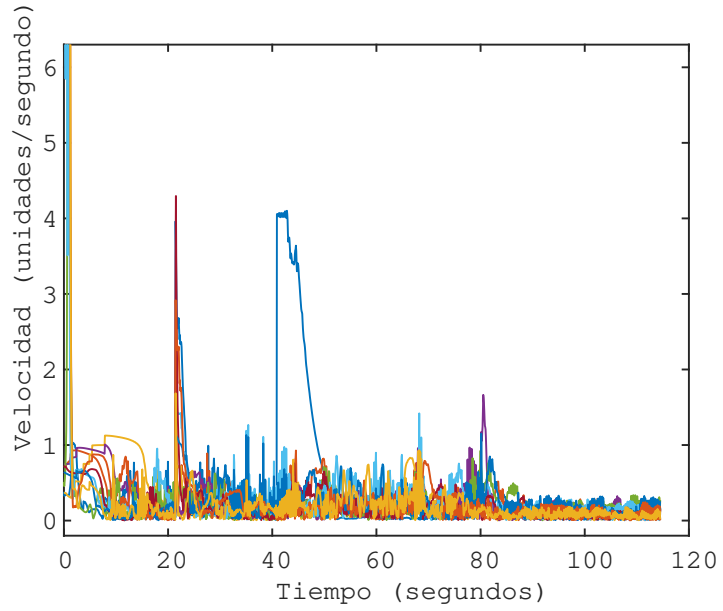


Figura 81: Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots

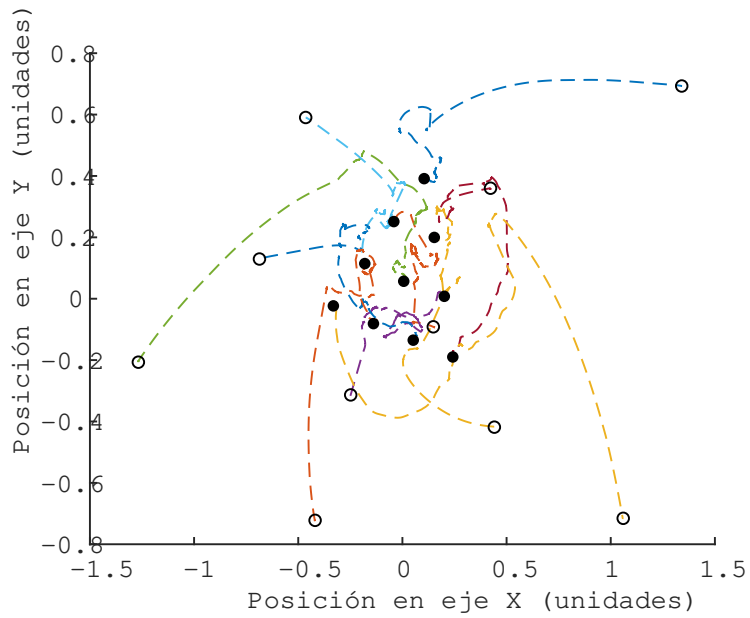


Figura 82: Trayectoria de los agentes, formación exitosa con grafo de prueba 1 y una matriz totalmente rígida con el modelo 6 adaptado a Webots

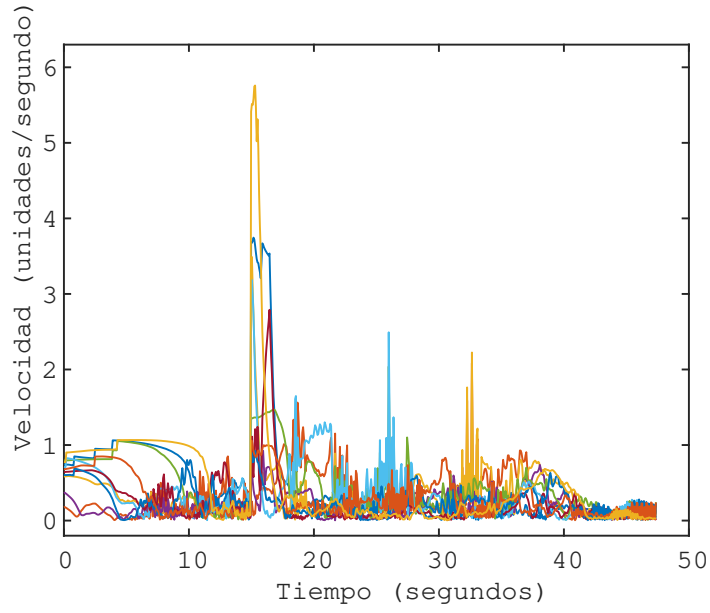


Figura 83: Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz totalmente rígida con el modelo 6 adaptado a Webots

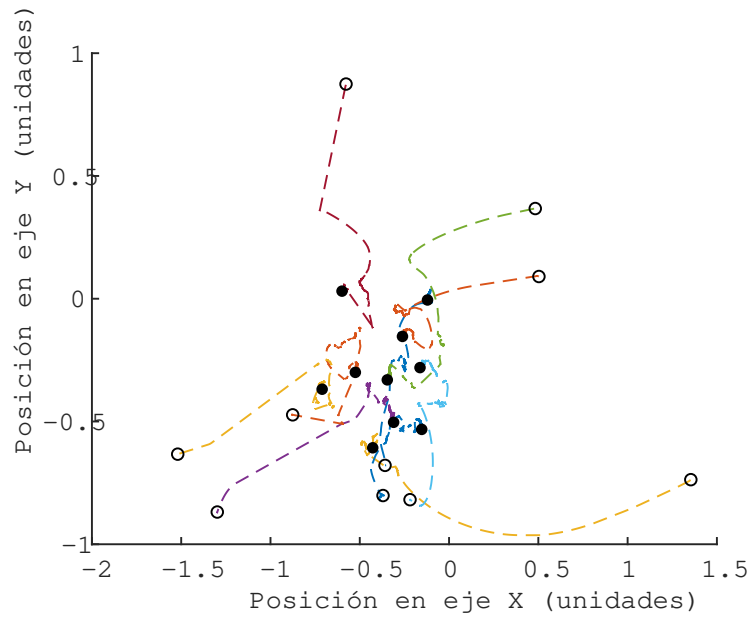


Figura 84: Trayectoria de los agentes, formación fallida con grafo de prueba 1 y una matriz totalmente rígida con el modelo 6 adaptado a Webots

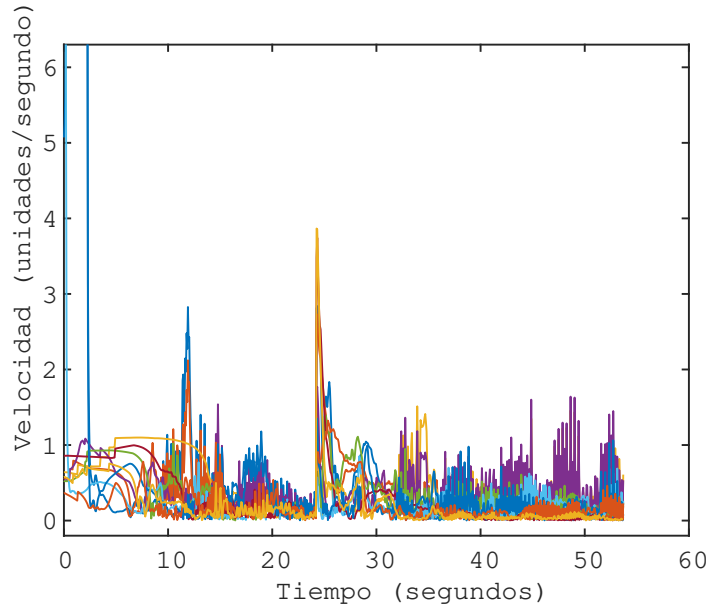


Figura 85: Velocidad de los agentes, formación exitosa con grafo de prueba 1 y una matriz totalmente rígida con el modelo 6 adaptado a Webots

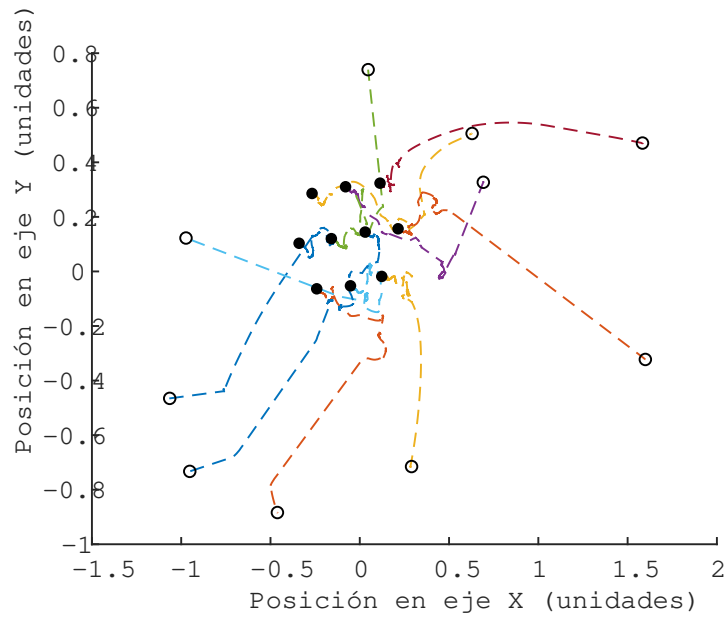


Figura 86: Trayectoria de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots

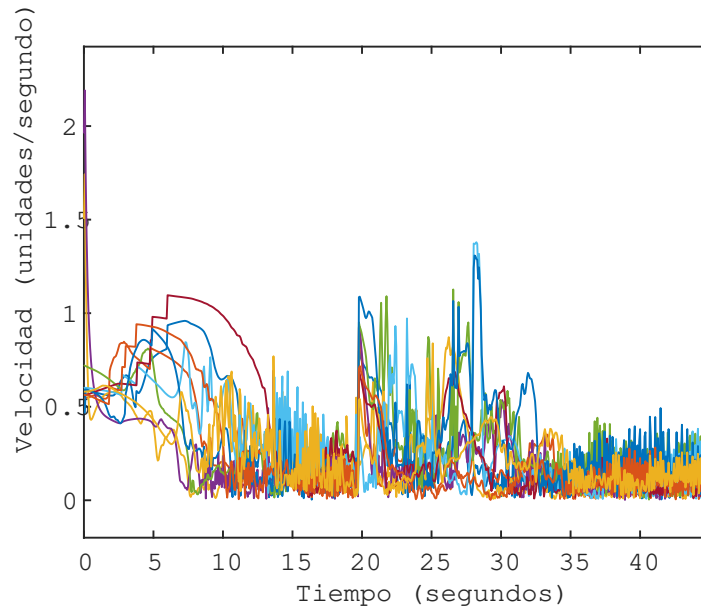


Figura 87: Velocidad de los agentes, formación exitosa con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots

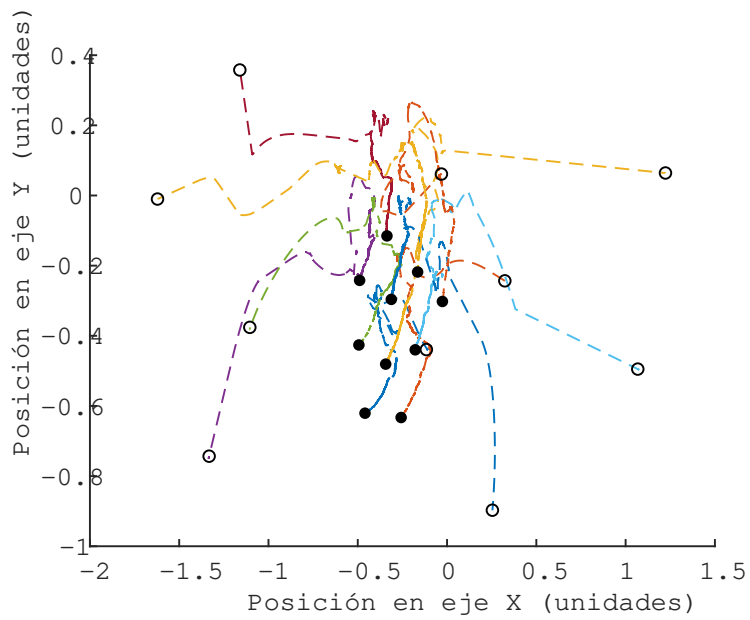


Figura 88: Trayectoria de los agentes, formación fallida con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots

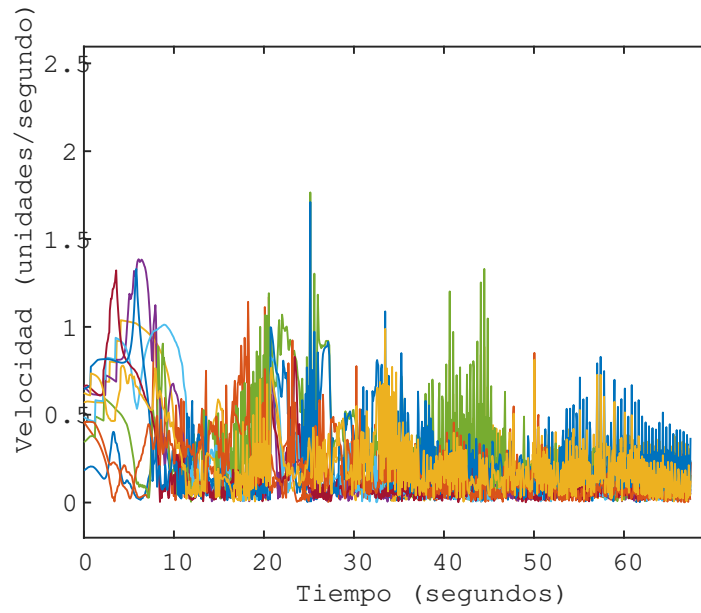


Figura 89: Velocidad de los agentes, formación fallida con grafo de prueba 2 y una matriz mínimamente rígida con el modelo 6 adaptado a Webots

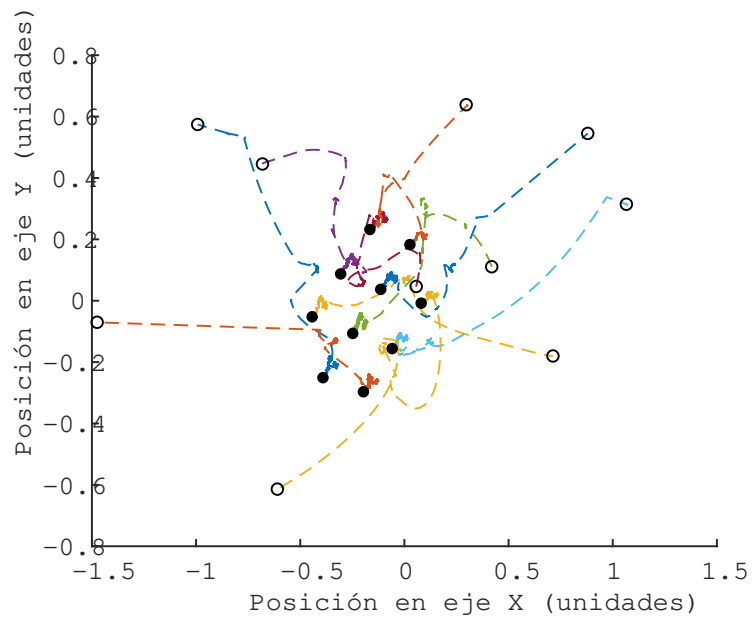


Figura 90: Trayectoria de los agentes, formación exitosa con grafo de prueba 2 y una matriz totalmente rígida con el modelo 6 adaptado a Webots

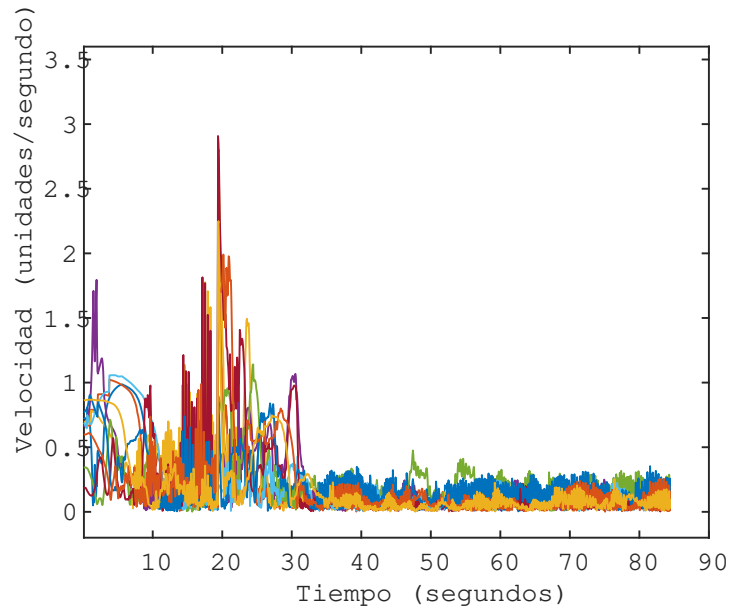


Figura 91: Velocidad de los agentes, formación exitosa con grafo de prueba 2 y una matriz totalmente rígida con el modelo 6 adaptado a Webots

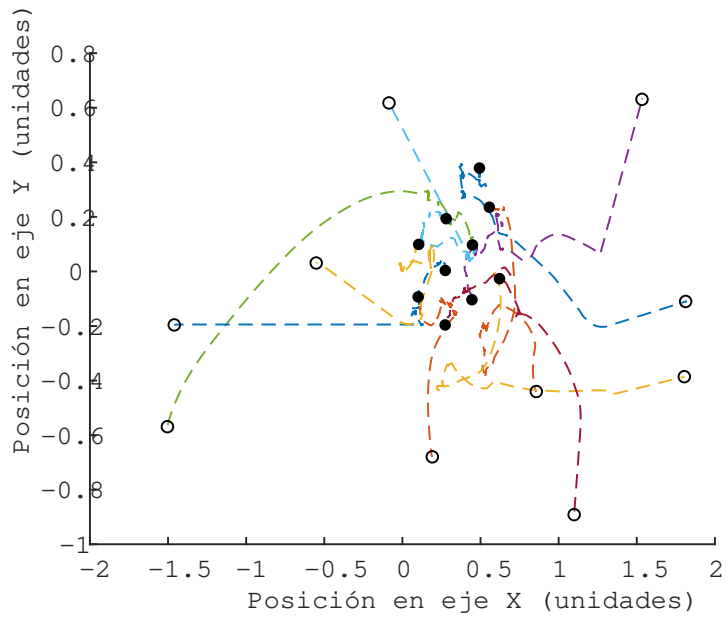


Figura 92: Trayectoria de los agentes, formación fallida con grafo de prueba 2 y una matriz totalmente rígida con el modelo 6 adaptado a Webots

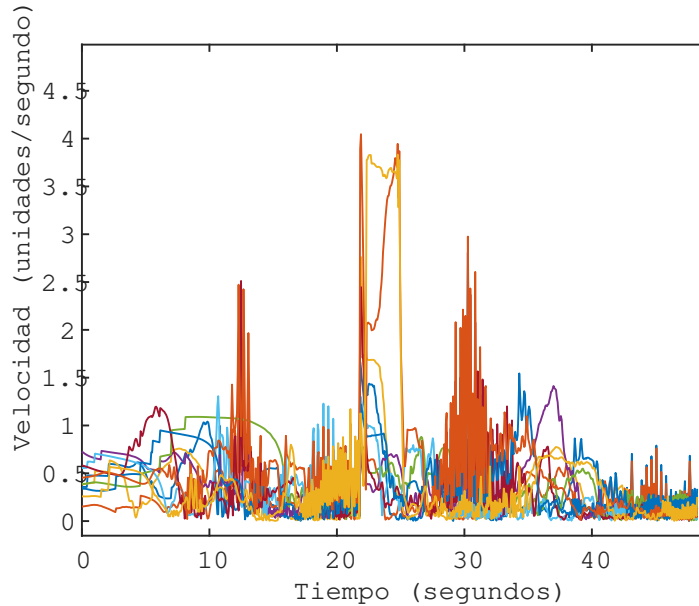


Figura 93: Velocidad de los agentes, formación fallida con grafo de prueba 2 y una matriz totalmente rígida con el modelo 6 adaptado a Webots

Los resultados anteriores tienen en común una gran cantidad de fluctuaciones en la velocidad de los agentes. Esta es la principal diferencia con los modelos teóricos, que tenían gráficos de velocidad más suaves y continuos. Esto puede deberse a varios factores, entre ellos: la comunicación entre el observador omnisciente y cada uno de los agentes que pudo haber introducido incluso pequeños retardos; así como las características de velocidad y aceleración máxima de los robots que implicaba truncar las velocidades propuestas por el algoritmo de control.

Sin embargo, a pesar de las fluctuaciones, puede apreciarse que la variación en magnitud tiende a ser pequeña, exceptuando algunos cambios bruscos que se dan principalmente en el cambio de control del modelo dinámico. Así que en cuanto a la forma, estos gráficos son bastante similares a los obtenidos en las simulaciones previas.

Se realizan algunas observaciones cualitativas a partir de los gráficos. Es posible lograr el éxito de la formación utilizando tanto matrices mínimamente rígidas como totalmente rígidas, al igual que en el modelo teórico. Los cambios más pronunciados en la velocidad de los agentes se presentan al inicio cuando los agentes inician en posiciones aleatorias y comienzan a acercarse y luego cuando realizan el cambio de control entre la evasión de colisiones y el control de formación combinado con evasión de colisiones. En la gran mayoría de casos al sistema multiagente le toma el cuádruple de tiempo lograr la convergencia si se compara con las simulaciones en Matlab.

Además del análisis cualitativo mediante gráficos, se realiza un análisis cuantitativo como el realizado para el caso teórico. Los resultados de este análisis se muestran en los Cuadros 8, 9 y 10. Para el caso de matrices totalmente rígidas se tiene alrededor de un 30 % de éxito en las formaciones. Mientras que para matrices mínimamente rígidas el éxito fue del 11 %. Porcentajes bastante similares a los obtenidos con el modelo 6 teórico, Cuadro 2. En cuanto al porcentaje de la formación alcanzada, no se tiene ningún caso en el que se haya

Grafo	Rigidez	< 0.001	< 0.005	< 0.01	< 0.02	< 0.04	< 0.07
1	Mínima	0	0	9	11	11	11
1	Total	22	28	28	28	29	29
2	Mínima	0	0	6	7	7	11
2	Total	9	16	18	18	18	33

Cuadro 8: Error cuadrático medio de la formación en la implementación. En este cuadro se muestra el porcentaje de simulaciones que tuvieron un error cuadrático medio menor al umbral indicado.

Grafo	Rigidez	100 %	90 %	80 %	70 %	0 %
1	Mínima	11	0	3	0	86
1	Total	29	2	0	0	69
2	Mínima	7	4	0	0	89
2	Total	18	7	0	0	75

Cuadro 9: Porcentaje de formación exitosa en la implementación. Este cuadro muestra el porcentaje de simulaciones en el que se logró el porcentaje de la formación indicado.

logrado el 70 % de la formación, y los casos del 90 % y 80 % de éxito son bastante escasos. Esto implica que se tiene un caso bastante binario, en donde se logra la formación con éxito total o simplemente no se logra. Esto presenta una gran diferencia con el caso teórico donde el porcentaje de formaciones logradas al 90 % era muy similar a las alcanzadas al 100 %, Cuadro 3.

Finalmente, en cuánto a la energía solicitada por el sistema, este fue el resultado con mayores diferencias entre el modelo teórico y la experimentación. La energía requerida por el sistema en la implementación es de hasta el 300 % de la requerida por el modelo teórico y el coeficiente de variación es casi 100 veces el coeficiente teórico, Cuadros 4 y 10. Esto coincide con el hecho de que las gráficas presentan muchas fluctuaciones, a más fluctuaciones se necesita más energía. No obstante, estos exagerados requerimientos de energía no impiden el alcanzar formaciones exitosas. Es decir, que el sistema cuenta con la energía suficiente para llevar a cabo el algoritmo de control.

Grafo	Rigidez	< 100	< 200	< 300	Desviación	Promedio	C. V.
1	Mínima	0	67	82	1.73E+03	6.00E+02	2.885
1	Total	0	54	76	2.13E+06	2.20E+05	9.725
2	Mínima	0	60	71	2.48E+04	4.32E+03	5.750
2	Total	1	48	73	1.53E+04	3.10E+03	4.944

Cuadro 10: Energía exigida por control en la implementación. Este cuadro corresponde a las mediciones de energía exigida por el control en las simulaciones en Webots, se muestra el porcentaje de simulaciones con energía menor a los 3 parámetros establecidos. Además, se indica la desviación, el promedio y el coeficiente de variación de la energía en todas las simulaciones realizadas.

8.1.3. Evasión de obstáculos y cambio de formación

Se realizaron las simulaciones en dos escenarios distintos utilizando únicamente el último modelo seleccionado por el proceso iterativo. Este consiste en el modelo dinámico que emplea la función de coseno hiperbólico para grafos mínimamente rígidos y que espera un tiempo de 28 segundos previo a realizar un cambio de configuración.

Para ambos escenarios se dispusieron tres obstáculos con geometría toroidal, posicionados de tal manera que fueren al grupo de robots a deformarse o reconfigurarse para llegar a la meta. La arena tiene dimensiones de 4m x 2m. Los obstáculos son los toroides morados y la meta está señalizada por una esfera amarilla flotante.

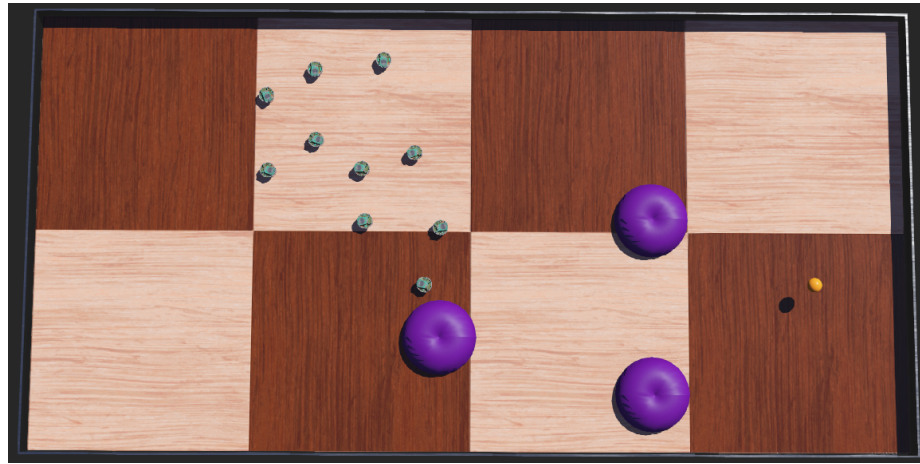


Figura 94: Escenario 1 de la simulación en Webots

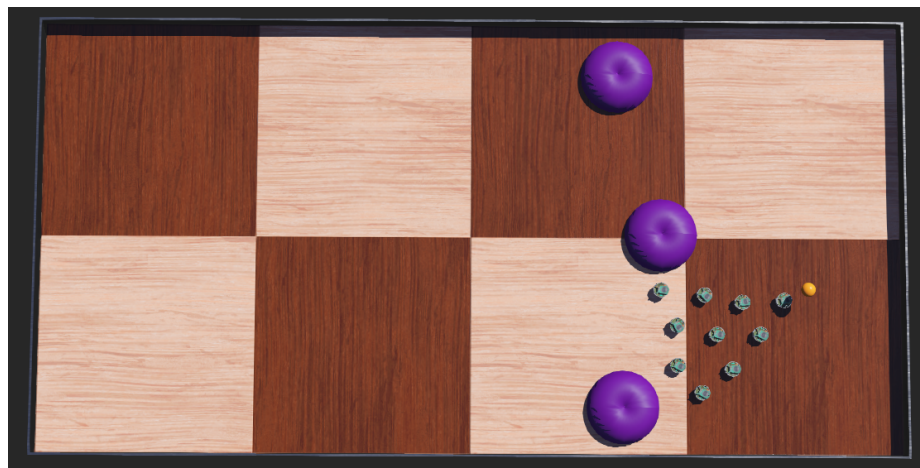


Figura 95: Escenario 2 de la simulación en Webots

A continuación, en las Figuras 96 a la 107, se presentan algunos de los resultados obtenidos con los dos diferentes arreglos de obstáculos. De la misma manera que con las simulaciones teóricas, se analiza la trayectoria y velocidad de cada agente así como el error

entre los dos grafos de formación. En cuanto al análisis cuantitativo se vuelven a utilizar las mismas métricas para evaluar el éxito del algoritmo.

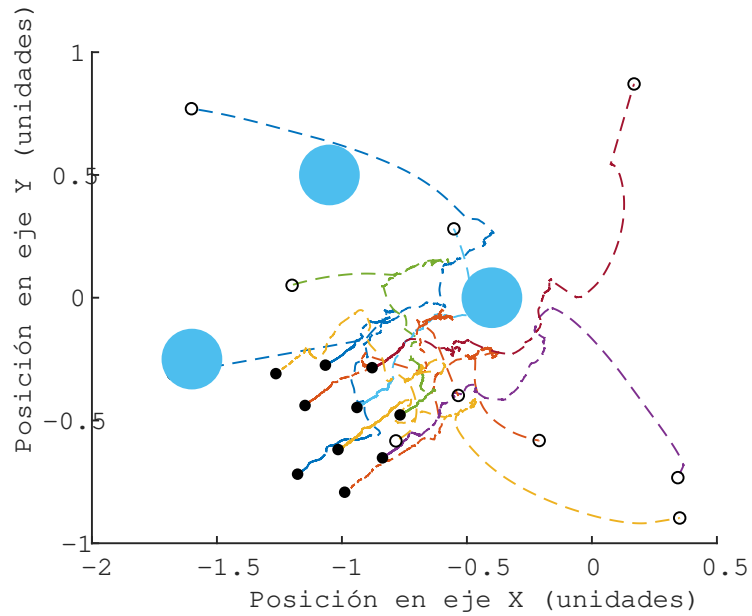


Figura 96: Trayectorias de los agentes, formación inicial exitosa que sufre deformación, con tres cambios de formación y llegada a la meta exitosa en el primer escenario

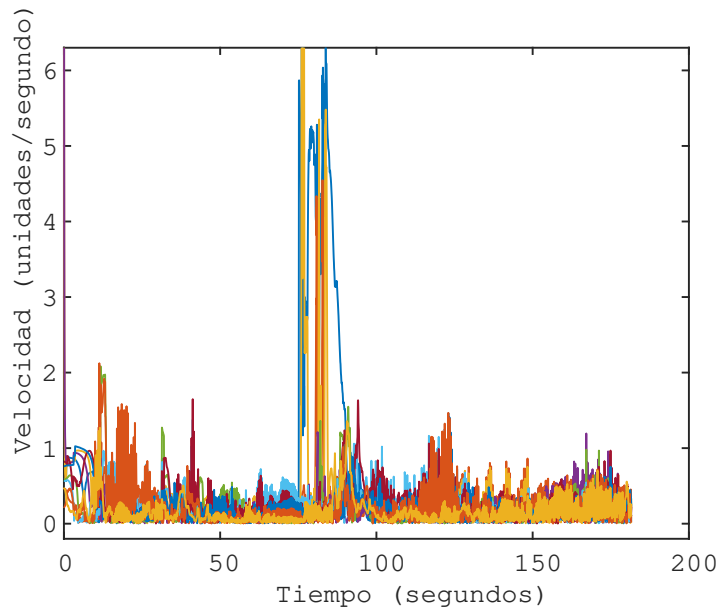


Figura 97: Velocidades de los agentes, formación inicial exitosa que sufre deformación, con tres cambios de formación y llegada a la meta exitosa en el primer escenario

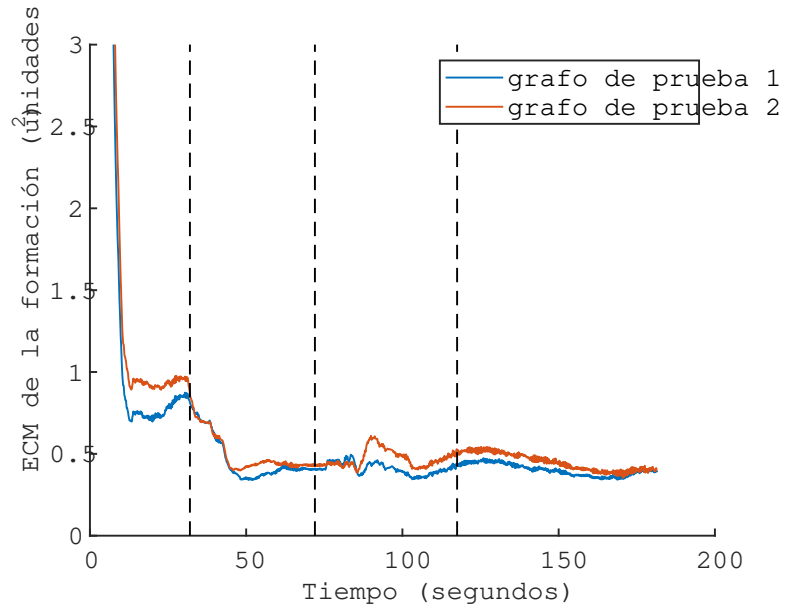


Figura 98: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa que sufre deformación, con tres cambios de formación y llegada a la meta exitosa en el primer escenario

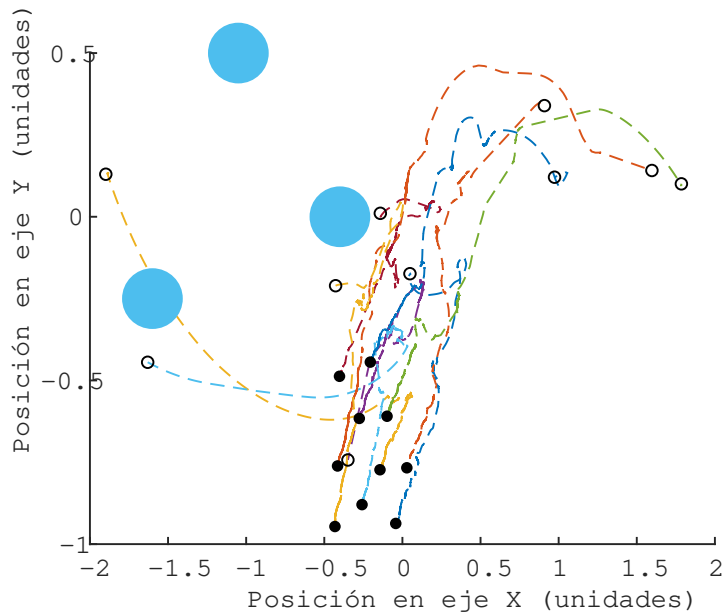


Figura 99: Trayectorias de los agentes, formación inicial exitosa, sin cambios de formación y llegada a la meta exitosa en el primer escenario

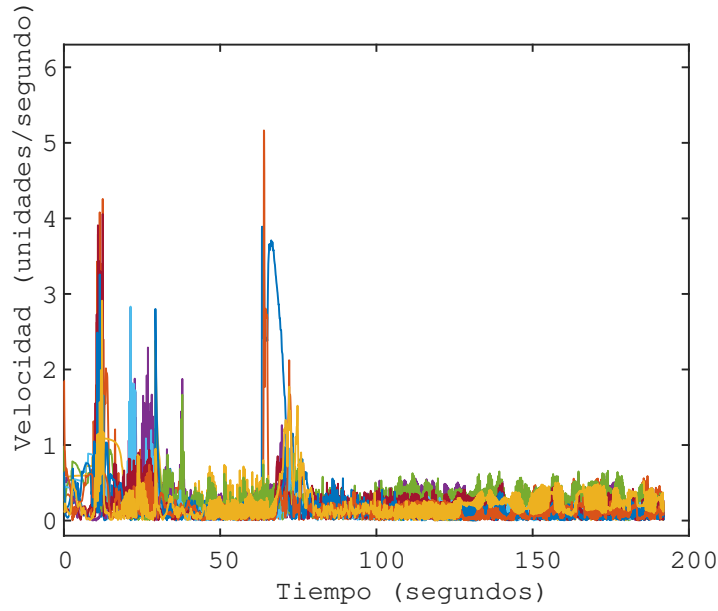


Figura 100: Velocidades de los agentes, formación inicial exitosa que sufre deformación, sin cambios de formación y llegada a la meta exitosa en el primer escenario

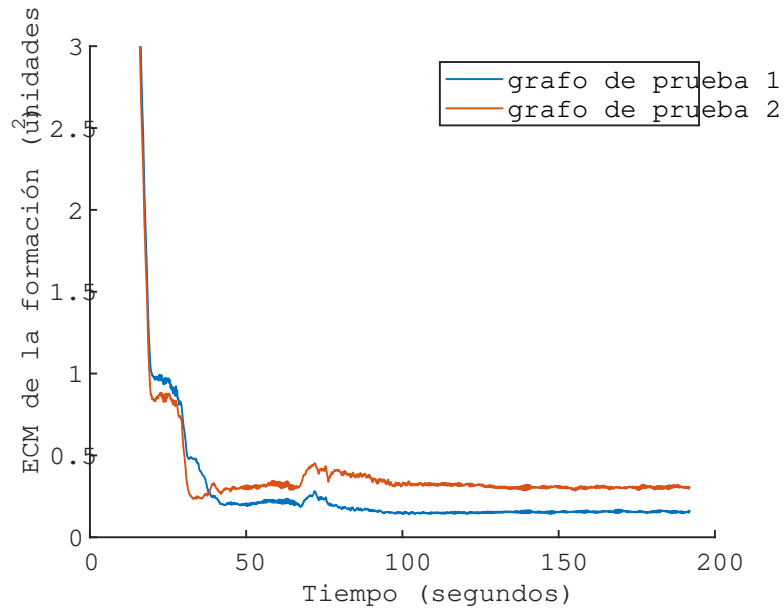


Figura 101: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa, sin cambios de formación y llegada a la meta exitosa en el primer escenario

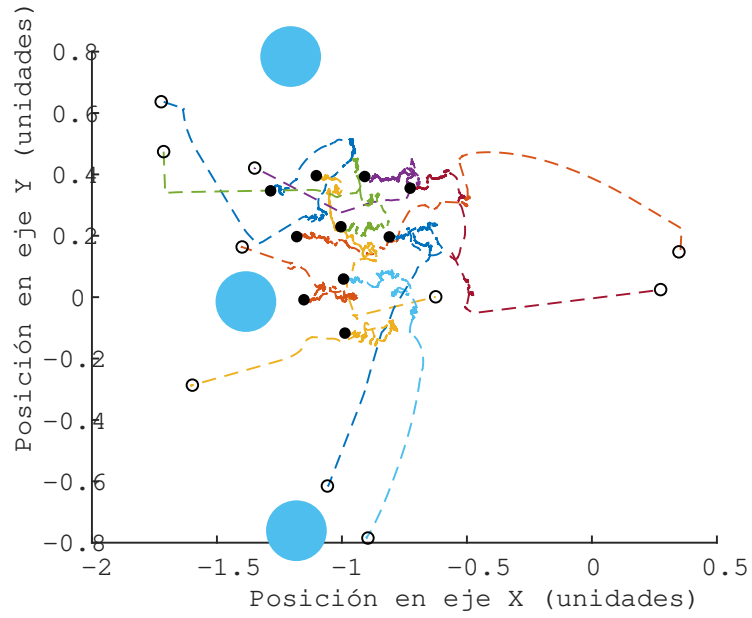


Figura 102: Trayectorias de los agentes, formación inicial fallida que llega a ser exitosa a medio camino y luego se deforma, y llegada a la meta exitosa en el segundo escenario

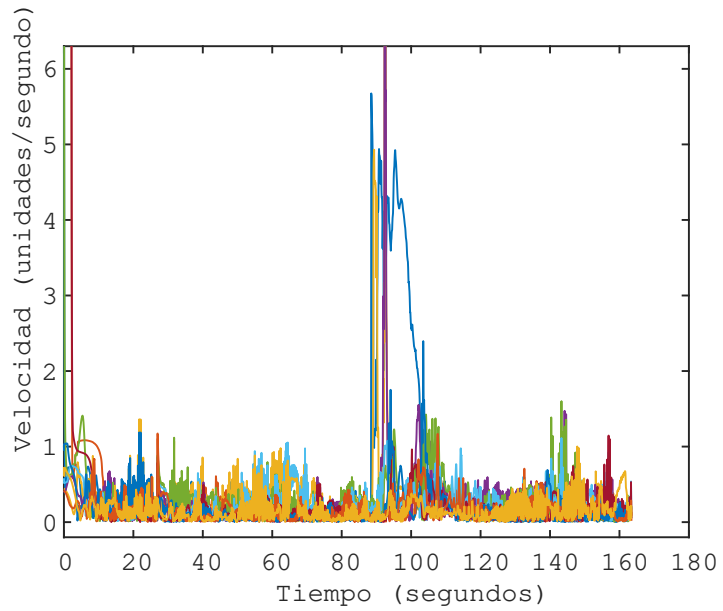


Figura 103: Velocidades de los agentes, formación inicial fallida que llega a ser exitosa a medio camino y luego se deforma, y llegada a la meta exitosa en el segundo escenario

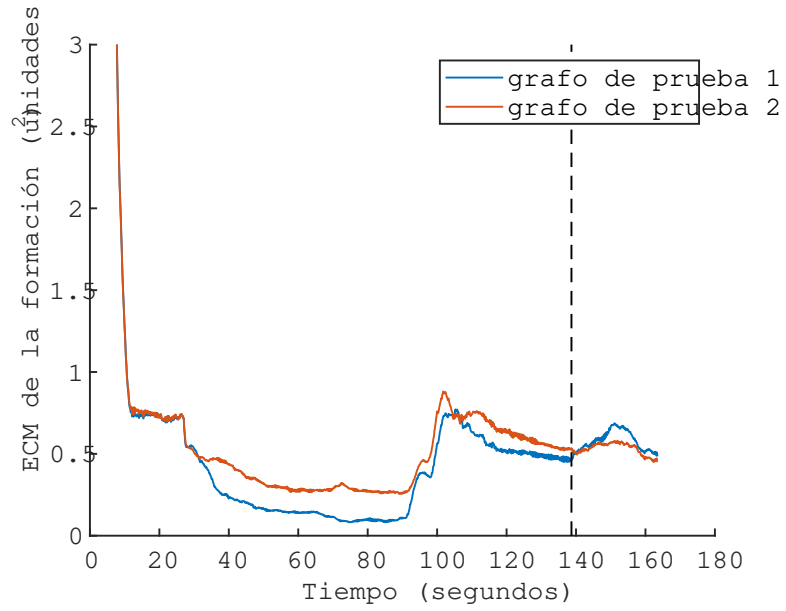


Figura 104: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial fallida que llega a ser exitosa a medio camino y luego se deforma, y llegada a la meta exitosa en el segundo escenario

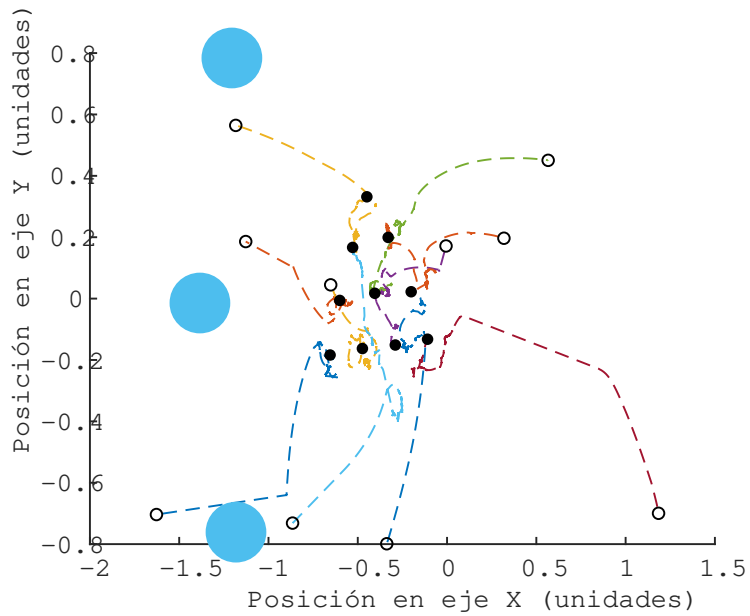


Figura 105: Trayectorias de los agentes, formación inicial exitosa con líder que se separa, sin cambios de formación y llegada a la meta exitosa en el segundo escenario

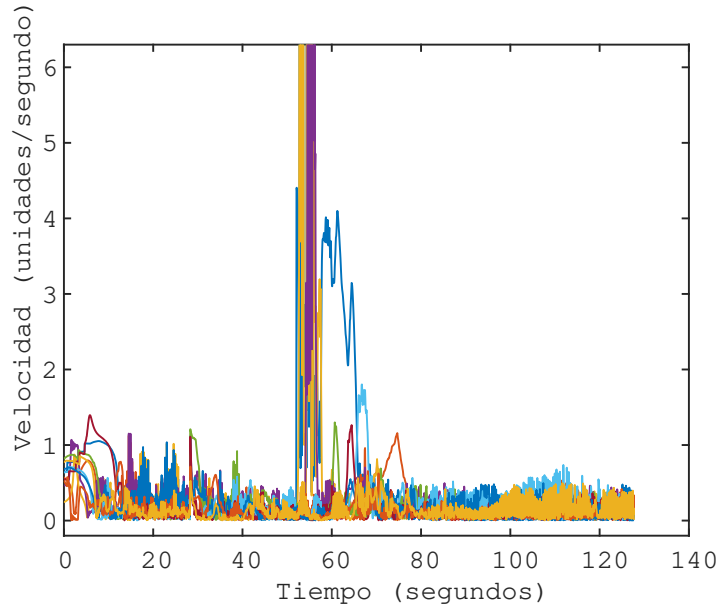


Figura 106: Velocidades de los agentes, formación inicial exitosa con líder que se separa, sin cambios de formación y llegada a la meta exitosa en el segundo escenario

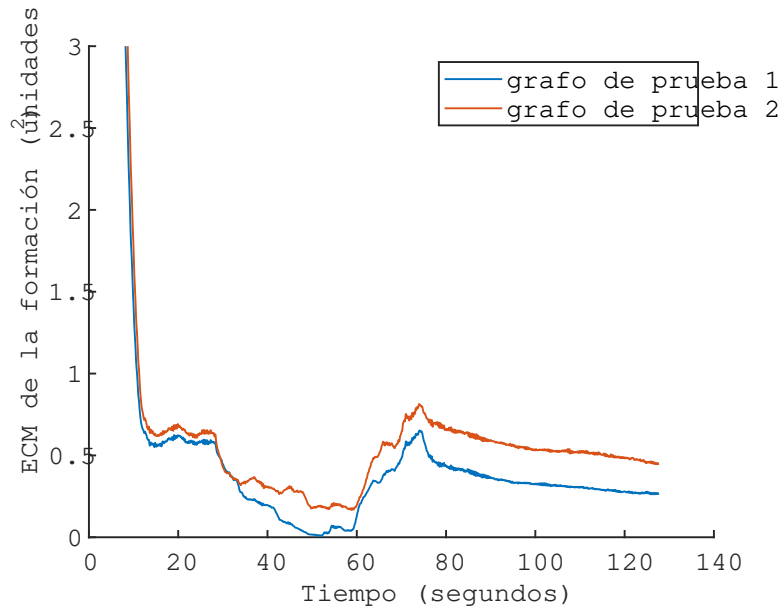


Figura 107: Error cuadrático medio de la configuración de los agentes comparado con los dos grafos de prueba, formación inicial exitosa con líder que se separa, sin cambios de formación y llegada a la meta exitosa en el segundo escenario

En los resultados presentados todos lograron llegar a la meta de forma exitosa. Se tienen tres casos en los que se logró la formación inicial, Figuras 96, 99 y 105. En el caso de la configuración inicial fallida, los agentes lograron reconfigurarse al final y terminaron con la formación deseada, Figura 102. De los casos iniciales exitosos, 2 se deforman previo a llegar a la meta, Figuras 96 y 102, donde el primer caso realiza tres cambios de formación 98.

Escenario	Inicial	Final
1	14	14
2	12	12

Cuadro 11: Éxito de las formaciones iniciales y formaciones finales en la implementación. En este cuadro se muestra el porcentaje de simulaciones que tuvieron éxito en lograr la formación inicial y las que tuvieron formaciones finales exitosas

Escenario	Fallo	Meta	No Meta	ER > 1	ER < 1
1	14	86	0	36	50
2	12	78	10	46	42

Cuadro 12: Llegadas exitosas a la meta y error relativo de formación final e inicial en la implementación. En este cuadro se muestra el porcentaje de simulaciones que tuvieron éxito en llegar a la meta (Meta), las que lograron converger a una formación pero no llegaron a la meta (No meta) y aquellas que no lograron converger a ninguna formación inicial (Fallo). Además se muestra el porcentaje de simulaciones con un error relativo menor a uno, lo cuál implica que el error de la formación final es menor que el error de la formación inicial; y el porcentaje de casos con error relativo mayor a uno que se refiere a la situación inversa.

La variedad de posibles resultados sigue siendo la misma que en las simulaciones previas, lo único que cambia es la frecuencia con la que se da cada caso. Para examinar esto se realiza el análisis cuantitativo, presentado en los Cuadros 11, 12 y 13.

Examinando estos resultados cuantitativos se nota que el éxito de formaciones iniciales y finales disminuyó considerablemente comparado con las simulaciones teóricas. Sin embargo, los resultados de error relativo fueron bastante similares. Los cantidad de casos en los que la formación final mejora con respecto a la formación inicial es prácticamente la misma que en el caso inverso. El porcentaje de experimentos realizados en la implementación en los que el enjambre llegó a la meta con éxito es de aproximadamente el 80 % para ambos escenarios considerados. Este resultado se encuentra bastante cercano al obtenido en las simulaciones previas, Cuadro 6. Tanto la cantidad de ocasiones en las que no se llegó a la convergencia de la formación como la cantidad de ocasiones en las que se converge a alguna formación pero no se llegó a la meta disminuyeron respecto a lo sucedido en el modelo teórico.

Otro aspecto interesante de resaltar es que los cambios por simulación se redujeron en la implementación. Fueron sumamente pocas las oportunidades en las que se necesitaron 1 o 2 cambios de formación, en general el enjambre continuaba intentando mantener la misma formación inicial a pesar de la deformación causada por los obstáculos.

Escenario	0	1	2
1	82	12	6
2	88	6	6

Cuadro 13: Cambios de formación en la implementación. En este cuadro se presenta el porcentaje de simulaciones que realizaron 0, 1 y 2 cambios de formación.

En conclusión, la implementación presentó resultados satisfactorios en cuanto al éxito total del algoritmo, se llegó a la meta en el 80 % de los casos. El resto de parámetros y métricas muestran distintas variaciones con respecto a la simulación, pero los resultados no se alejan de lo esperado y deseado. Si se ahonda un poco más en los aspectos evaluados pueden encontrarse las oportunidades de mejora para este algoritmo, algunas de estas se abarcan en el capítulo 10.

1. La combinación del control de formación y el control contra colisiones en una sola función racional permite que los agentes alcancen formaciones con un mínimo error cuadrático medio al usar grafos totalmente rígidos. Sin embargo, el utilizar grafos totalmente rígidos reduce el porcentaje de éxito del sistema en llegar a la meta establecida ya que el exceso de restricciones dificulta el atravesar los obstáculos. De allí surge la importancia de la función coseno hiperbólico permite que los agentes se ajusten únicamente según el mínimo establecido por el control de formación, usando un grafo mínimamente rígido.
2. El utilizar un modelo de control dinámico reduce la cantidad de casos en los que los agentes se alejan demasiado del radar de los demás pero aumenta la energía requerida por el sistema en los puntos de cambio de control. Así que el imponer un límite de velocidad, reduce la magnitud de dicha energía.
3. El tiempo de espera para los cambios de formación es un factor que debe ser ajustado según la implementación. A menor tiempo de espera entre el cambio de configuraciones del sistema, se presenta una mayor dificultad para lograr una formación inicial exitosa.
4. La evaluación de las métricas establecidas en el algoritmo implementado en el entorno de simulación Webots, cumplió con avalar la efectividad del algoritmo desarrollado. El subalgoritmo propuesto para llegar a la formación deseada alcanza hasta un 25 % de éxito en simulación y un 11 % en la implementación, utilizando grafos mínimamente rígidos. En al menos el 90 % de los experimentos los agentes convergieron a una formación en la implementación aunque no necesariamente fuese exitosa. El algoritmo completo cumple con llegar a la meta hasta en un 80 % de las ocasiones en su implementación.

1. Evitar asíntotas verticales como restricciones en las funciones de tensión entre aristas.
2. Aplicar un control de aceleración en los agentes para evitar cambios bruscos en la velocidad y así disminuir la energía requerida por el sistema.
3. Realizar pruebas con más variedad de grafos mínimamente rígidos y con un distinto número de agentes.
4. Para su implementación física considerar un sistema de posicionamiento global que se comunique de manera rápida y eficiente con todos los agentes.
5. Mantener la utilización de grafos mínimamente rígidos si se emplea un sistema de localización local para simplificar la comunicación.
6. Considerar incluir un sistema de asignación que determine la posición que debe tomar cada agente dependiendo del error de formación inicial.

-
- [1] F. Roberti, J. M. Toibero, R. F. Vassallo y R. Carelli, “Control Estable de Formación Basado en Visión Omnidireccional para Robots Móviles No Holonómicos,” *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 8, n.º 1, págs. 29-37, ene. de 2011. DOI: 10.1016/s1697-7912(11)70005-x.
 - [2] J. Fredslund y M. J. Mataric, “A general algorithm for robot formations using local sensing and minimal communication,” *IEEE Transactions on Robotics and Automation*, vol. 18, n.º 5, págs. 837-846, oct. de 2002, ISSN: 1042-296X. DOI: 10.1109/TRA.2002.803458.
 - [3] D. J. Bruemmer, D. D. Dudenhoefter, M. D. McKay y M. O. Anderson, “A Robotic Swarm for Spill Finding and Perimeter Formation,” 2008.
 - [4] Q. Ha y G. Dissanayake, “Robotic Formation Control using Variable Structure Systems Approach,” vol. 2006, jul. de 2006, págs. 37-42, ISBN: 0-7695-2589-X. DOI: 10.1109/DIS.2006.64.
 - [5] A.-T. Gheorghita, “Sistema de Control de un Convoy Automático con Base Experimental,” Tesis de mtría., Universidad Complutense de Madrid, 2013.
 - [6] R. L. Long, R. Mead y J. B. Weinberg, “Distributed Auction-Based Initialization of Mobile Robot Formations,” en *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*, 2010. dirección: <http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/view/1673>.
 - [7] J. Desai, J. Ostrowski y V. Kumar, “Modeling and control of formations of nonholonomic mobile robots,” *IEEE Transactions on Robotics and Automation*, vol. 17, n.º 6, págs. 905-908, 2001. DOI: 10.1109/70.976023.
 - [8] G. L. Mariottini, F. Morbidi, D. Prattichizzo, G. J. Pappas y K. Daniilidis, “Leader-Follower Formations: Uncalibrated Vision-Based Localization and Control,” en *Proceedings 2007 IEEE International Conference on Robotics and Automation*, IEEE, abr. de 2007. DOI: 10.1109/robot.2007.363679.
 - [9] *Robotarium*, 2018. dirección: <https://www.robotarium.gatech.edu/>.

- [10] A. Solomon, *Georgia Tech's Robotarium is .a shining beacon of robotic awesomeness*", jul. de 2018. dirección: <https://www.atlantamagazine.com/news-culture-articles/georgia-tech-robotarium-shining-beacon-robotic-awesomeness/>.
- [11] *Programmable Robot Swarms*, mayo de 2019. dirección: <https://wyss.harvard.edu/technology/programmable-robot-swarms/>.
- [12] H.-W. University, *Green light for National Robotarium*, dic. de 2018. dirección: <https://www.hw.ac.uk/news/articles/2018/green-light-for-national-robotarium.htm>.
- [13] A. J. R. Hernández, "Desarrollo e implementación de algoritmo de visión por computador en una mesa de pruebas para la experimentación con micro-robots móviles en robótica de enjambre," Tesis de mtría., Universidad del Valle de Guatemala, 2019.
- [14] M. J. C. Martínez, "Diseñar e implementar una red de comunicación inalámbrica para la experimentación en robótica de enjambre," Tesis de mtría., Universidad del Valle de Guatemala, 2019.
- [15] A. García, *Guatemala está expuesta a toda clase de desastres, excepto a tormentas de nieve (por el momento)*, jun. de 2018. dirección: <https://www.plazapublica.com.gt/content/guatemala-esta-expuesta-toda-clase-de-desastres-excepto-tormentas-de-nieve-por-el-momento>.
- [16] K. Ruohonen, *Graph theory*, ép. Lecture notes. 2013. dirección: <http://gen.lib.rus.ec/book/index.php?md5=4c908924a3ae2a7170811eea4201729f>.
- [17] K. Hoffman, *Linear algebra*. Prentice-Hall, 1971.
- [18] S. Grossman, *Elementery linear algebra*. Saunders College, 1994.
- [19] L. Krick, "Application to Graph Rigidity in Formation Control of Multi-Robot Networks," Tesis de mtría., University of Toronto, 2007.
- [20] M. Mesbahi y M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 6 de ago. de 2010, 424 págs., ISBN: 0691140618. dirección: https://www.ebook.de/de/product/10047947/mehran_mesbahi_graph_theoretic_methods_in_multiagent_networks.html.
- [21] N. S. Nise, *Control system engineering*. Benhamin/Cummings, 1995.
- [22] D. M. Egerstedt, *Control of Mobile Robots - Module 7*.
- [23] C. Ltd., *Plans & Pricing*. dirección: <https://cyberbotics.com/>.
- [24] *GCTronic' e-puck*. dirección: <https://cyberbotics.com/doc/guide/epuck#e-puck-model>.

12.1. Repositorio

Enlace al repositorio con las simulaciones de los modelos teóricos y la implementación:
<https://github.com/AMaybellP/Tesis2019-AMPE>