

Multi-Agent Communication Models for Cooperative Navigation in Complex Environments

Jonathan Rodríguez, Julio Godoy, Fernando Gutierrez

Abstract—Multi-agent navigation in restricted environments presents significant challenges, since agents need to move to their goals in an efficient manner while avoiding collisions with both static and dynamic obstacles. Previously, *C-Nav* was proposed as a method that can effectively coordinate groups of agents in very restricted environments. In this work, we propose and evaluate three alternative communications models for *C-Nav*, that intend to provide more flexibility to the original method. The results of our experiments show that each of our proposed methods can lead to significant improvements over *C-Nav* in specific types of restricted environments.

Index Terms—multi agent systems, navigation, multi agent coordination.

I. INTRODUCCIÓN

La navegación multi-agente consiste en mover un conjunto de agentes entre sus posiciones de inicio y destino, evitando colisionar con otros agentes y obstáculos en el ambiente. Este problema tiene aplicación en áreas como simulación de peatones, ingeniería de transporte y robótica de enjambre. Resolver este problema de forma centralizada ha demostrado ser intratable (P-SPACE hard [1]), por lo que muchos métodos han optado por una estrategia de control distribuida, donde cada agente decide cómo moverse de manera independiente [2], [3].

Múltiples métodos distribuidos han sido propuestos en las últimas décadas [2], [3], [4], [5]. Sin embargo, cuando se escala el número de agentes en ambientes que restringen su movimiento, los movimientos que estos métodos producen pueden ser poco eficientes y llevar a largas demoras en la navegación [6], pudiendo incluso producir bloqueos o *deadlocks* que impiden a los agentes llegar a sus objetivos.

Una alternativa para mejorar la eficiencia en la navegación multi-agente en ambientes restringidos es mediante la coordinación de movimientos en base a comunicación entre los agentes. Recientemente fue propuesto el método *C-Nav* [7], en el cual los agentes comparten su intención de movimiento usando un mecanismo de comunicación de *bajo costo* para moverse respetando las intenciones de los otros agentes. Esta cooperación entre agentes lleva a una disminución de los tiempos de viaje comparado con un método *estado-del-arte* para navegación multi-agente, *Optimal Reciprocal Collision Avoidance (ORCA)* [4].

En este trabajo, se proponen tres métodos que modifican *C-Nav* para incrementar la eficiencia en la navegación

multi-agente en ambientes restrictivos. Estos métodos ofrecen alternativas al modelo de comunicación original de *C-Nav*, donde cada uno tiene un enfoque particular sobre qué información debe ser comunicada, y su posterior utilización. Debido a estos enfoques, cada método ofrece mejoras en la navegación en distintos tipos de ambientes. El primer método, *liderazgo dinámico*, permite a los agentes identificar a líderes que puedan guiarlos hacia sus objetivos. Esto favorece la navegación en ambientes donde se puede identificar grupos de agentes que se dirigen a objetivos muy similares. El segundo método, *alternancia de velocidades*, intenta prevenir que los agentes se aglomeren mediante la alternancia de direcciones de navegación, evitando y resolviendo bloqueos que puedan producirse. Finalmente, el tercer método, *penalización basada en densidad*, intenta prevenir que agentes se agrupen en áreas de alta densidad. La baja densidad lleva a que los agentes no bloqueen la navegación de otros.

Los métodos propuestos fueron evaluados en seis escenarios simulados, que enfatizan restricción de movimiento en distintas formas. En general, los métodos propuestos ofrecen mejoras sobre *C-Nav*, con distintos niveles de eficiencia.

Este trabajo está organizado de la siguiente forma: en la sección II se describen trabajos de navegación y coordinación para sistemas multi-agente relacionados con este trabajo. En la sección III el problema de navegación multi-agente es formalizado, mientras que en la sección IV se describe *C-Nav*, el método base sobre el cual se desarrolló este trabajo. En la sección V se presentan los tres métodos propuestos. La sección VI describe los experimentos para evaluar lo propuesto, junto con su análisis. Finalmente, se presentan las conclusiones de este trabajo en la sección VII.

II. TRABAJOS RELACIONADOS

Desde la introducción de los *boids* [2], donde un conjunto de reglas simples puede producir comportamientos complejos descritos en la literatura como *flocking*, la navegación multi-agente ha sido objeto de muchas contribuciones. En concreto, varios modelos de navegación han sido propuestos, usando, por ejemplo, fuerzas sociales [3], modelos que consideran grupos de agentes [8], modelos basados en reglas cognitivas [9], basados en principios biomecánicos [10], factores psicológicos o sociológicos [11], [12], entre otros. Contribuciones a la navegación multi-agente descentralizada también han venido desde el área de control distribuido [13], [14], [15], y modelos de agentes markovianos [16], [17]. La mayoría de estos métodos son categorizados como *reactivos*, ya que actúan en base a observaciones actuales, sin tomar en cuenta el estado

Jonathan Rodríguez, Department of Computer Science, Universidad de Concepcion, Chile, jonarodriguez@udec.cl.

Julio Godoy, Department of Computer Science, Universidad de Concepcion, Chile, juliogodoy@udec.cl.

Fernando Gutierrez, AIDA SpA, Chile, fernando.a.gutierrez.gomez@gmail.com.

futuro del sistema de agentes [18]. En muchos problemas, el grado de incertidumbre respecto del ambiente hace que estos métodos reactivos sean la única opción disponible. Sin embargo, debido a que la mayoría de estos métodos no toman en consideración las posibles futuras posiciones de los agentes, tienden a tener comportamientos poco realistas, resultando, por ejemplo, en oscilaciones entre los agentes en situaciones de congestión. Considerando estas limitaciones, se han propuesto métodos basados en velocidades [19] que intentan anticipar las posiciones de los agentes, resultando en movimientos más naturales. En la última década este enfoque se ha popularizado, en especial el método *ORCA* [4], que ha sido utilizado para la simulación de peatones [10], [20], [21] y la navegación de robots [22] [23], [24], entre otras aplicaciones [25].

A pesar de que *ORCA* demuestra un buen desempeño en los dominios indicados previamente, al escalar el número de agentes en ambientes poblados con obstáculos, la eficiencia local de movimiento no es suficiente para lograr eficiencia de movimientos globales. En estas condiciones, el actuar miope de los agentes puede producir congestión, demorando a los agentes e incluso pudiendo llevar a punto muerto (*deadlocks*) la navegación cuando agentes se mueven, por ejemplo, en dirección opuesta en pasillos estrechos. En estos casos, la reciprocidad en la evasión de colisiones de *ORCA* no es suficiente para coordinar el movimiento de los agentes para resolver estas situaciones.

Varios métodos han sido propuestos para coordinar las acciones de un conjunto de agentes, ya sea con mecanismos explícitos o implícitos [26], [27], [28]. En particular, para el problema de navegación, se han propuesto métodos de coordinación de agentes basados, por ejemplo, en *campos potenciales artificiales* [29], [30], aprendizaje [31], [32] o como resultado emergente de la interacción de agentes con reglas de comportamiento [33], [34]. Recientemente, el método de coordinación para navegación multi-agente *C-Nav*, basado en *ORCA*, introdujo el concepto de amabilidad para mejorar la eficiencia en tareas de navegación de múltiples robots diferenciales en ambientes restringidos [7].

III. FORMULACIÓN DEL PROBLEMA

En el problema considerado, tenemos un conjunto de n agentes independientes (es decir, un sistema totalmente distribuido), $\mathcal{A}_1, \dots, \mathcal{A}_n$, donde cada agente tiene una posición inicial y final individual. Los agentes se mueven en un ambiente 2D que puede incluir un conjunto de obstáculos O . Cada agente \mathcal{A}_i es modelado en forma de disco con un radio r_i . En el tiempo t , el agente \mathcal{A}_i tiene una posición \mathbf{p}_i , se mueve con una velocidad \mathbf{v}_i con un máximo de $v_i^{\text{máx}}$. Además, para desplazarse en el ambiente, cada agente \mathcal{A}_i puede escoger entre un conjunto de *acciones* o *velocidades preferidas* (Fig. 1), $\mathbf{v}_i^{\text{pref}}$, definidas de forma empírica, que abarcan todo el espacio de posibles direcciones de movimiento. La velocidad preferida escogida por el agente, en cada tiempo t , corresponde a su *intención de movimiento* en t . Sin perjuicio de lo anterior, \mathcal{A}_i también posee una velocidad preferida hacia el objetivo $\mathbf{v}_i^{\text{goal}}$, que apunta en todo momento al objetivo del agente. Idealmente, en la ausencia de otros agentes y obstáculos, la

intención de movimiento del agente es igual a su velocidad hacia el objetivo, $\mathbf{v}_i^{\text{goal}}$.

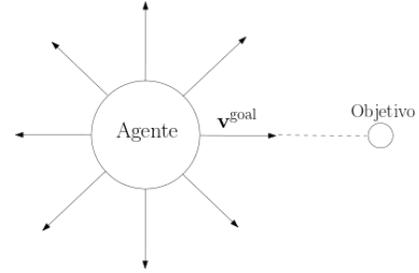


Fig. 1. Conjunto de velocidades preferidas o *acciones* disponibles para cada agente, con respecto a su posición objetivo. Cada velocidad preferida está separada por 45° de las adyacentes.

La tarea que motiva a este trabajo es mover a todos los agentes a su posición objetivo de forma segura (es decir, sin colisiones), y en el menor tiempo posible en ambientes restringidos. Como método base para este trabajo usamos *C-Nav* [7], el cual, a su vez, utiliza *ORCA* para convertir la intención de movimiento del agente en una velocidad que no involucre colisiones con obstáculos y otros agentes, \mathbf{v}^{new} .

IV. C-NAV

C-Nav es un método de coordinación para la navegación multi-agente que produce movimientos eficientes en ambientes de alta densidad y congestión. El Algoritmo 1 resume los pasos principales de *C-Nav* [7], en base a un ciclo percepción-planificación-actuación que se ejecuta de la siguiente manera:

Percepción. Cada agente observa las posiciones de los obstáculos, y las posiciones y velocidades (\mathbf{v}^{new}) de un subconjunto de agentes \mathcal{N} (de tamaño máximo $|\mathcal{N}|$), dentro del rango de percepción r_{perc} . *C-Nav* implementa un modelo de comunicación entre los agentes, en el que la intención de movimiento de cada uno (\mathbf{v}^{pref}), junto a su ID, es comunicada a los agentes más cercanos. (Alg. 1, línea 10).

Planificación. Para poder determinar su *velocidad preferida* o *acción* (Fig.1), cada agente en *C-Nav* utiliza la información de la *intención de movimiento* de los otros agentes para estimar el impacto de cada posible acción en su movimiento. En base a esto, cada agente determina la *restricción de movimiento* que impondría cada acción en los agentes más cercanos, que además estén más cerca de su posición objetivo que el propio agente (Alg. 1, línea 4). La *restricción de movimiento calculada* para un agente es la diferencia entre su *intención de movimiento* y su *velocidad observada*. Utilizamos la variable NVC_{ij} (*neighbor velocity change*) para determinar el efecto de cada acción del agente \mathcal{A}_i en las restricciones de movimiento de cada vecino j como sigue:

$$NVC_{ij} = v_i^{\text{máx}} - \|\mathbf{v}_j^{\text{pref}} - \mathbf{v}_j^{\text{new}}\| \quad (1)$$

A mayor valor de NVC_{ij} , más *amable* es la acción de \mathcal{A}_i evaluada con respecto a las restricciones del agente vecino \mathcal{A}_j . A partir de las restricciones de movimiento de los agentes cercanos, cada agente puede estimar cuál es la mejor siguiente acción. Para esto, las acciones presentadas en la Figura 1 son

simuladas por un breve horizonte temporal T (Alg. 1, línea 6) y evaluadas con la función de recompensa R_a (Alg. 1, línea 7), definida como sigue:

$$\mathcal{R}_a = (1 - \gamma) \cdot \mathcal{R}_a^g + \gamma \cdot \mathcal{R}_a^c \quad (2)$$

donde \mathcal{R}_a^g evalúa el progreso hacia el objetivo del agente en cada paso del horizonte de tiempo (T). Este componente favorece movimientos que acercan al agente a su objetivo. Más formalmente:

$$\mathcal{R}_a^g = \frac{\sum_{t=0}^{T-1} \left(\mathbf{v}_i^{\text{new}} \cdot \frac{\mathbf{g}_i - \mathbf{p}_i}{\|\mathbf{g}_i - \mathbf{p}_i\|} \right)}{T \cdot v_i^{\text{máx}}} \quad (3)$$

Por otro lado, \mathcal{R}_a^c evalúa el impacto de la acción a en los agentes cercanos, y se calcula mediante la suma de los valores de NVC , mediante la acción a , con un subconjunto de los agentes en \mathcal{N} ubicados entre el agente y su posición objetivo. Formalmente:

$$\mathcal{R}_a^c = \frac{\sum_{t=1}^{T-1} \sum_{j \in \mathcal{N}} NVC_{ij}}{(T-1) \cdot |\mathcal{N}| \cdot v_i^{\text{máx}}} \quad (4)$$

El *factor de coordinación* γ controla la influencia de cada uno de estos componentes en el valor final de la recompensa. La \mathbf{v}^{pref} a escoger corresponde a la acción que entregue la mayor recompensa (Alg. 1, línea 9).

Actuación. La acción previamente calculada corresponde a la nueva intención de movimiento, y es entregada como input a *ORCA* para obtener la velocidad libre de colisión respectiva (Alg. 1, línea 12). Después de esto, el proceso se repite nuevamente. Una descripción más detallada del método *C-Nav* se puede encontrar en [7].

Algorithm 1: El método *C-Nav* para el agente i

- 1: inicializar simulación
 - 2: **while** no en el objetivo **do**
 - 3: **if** EscogerAccion(t) **then**
 - 4: determinar agentes más restringidos
 - 5: **for all** $a \in \text{Actions}$ **do**
 - 6: simular acción a
 - 7: calcular recompensa R_a
 - 8: **end for**
 - 9: seleccionar acción a y obtener su $\mathbf{v}_i^{\text{pref}}$
 - 10: comunicar ID y $\mathbf{v}_i^{\text{pref}}$ a agentes cercanos
 - 11: **end if**
 - 12: calcular y ejecutar $\mathbf{v}_i^{\text{new}}$ para el siguiente paso
 - 13: actualizar posición \mathbf{p}_i del agente
 - 14: **end while**
-

A pesar de que *C-Nav* obtiene buenos resultados en un conjunto de escenarios, no se ha realizado un análisis de diferentes modelos de comunicación y su impacto en el desempeño del método, ni de qué modelos de comunicación podrían ser más adecuados para cada escenario. Por lo tanto, en este trabajo, se utiliza *C-Nav* como método base para proponer distintos modelos de comunicación, y distintas formas de

utilizar la información percibida mediante estos modelos de comunicación.

V. MÉTODOS PROPUESTOS

En esta sección, se presentan tres métodos que implementan diferentes modelos de comunicación, para mejorar la eficiencia en la navegación multi-agente. Estos métodos son variaciones de *C-Nav* que integran modificaciones en la información comunicada y su utilización.

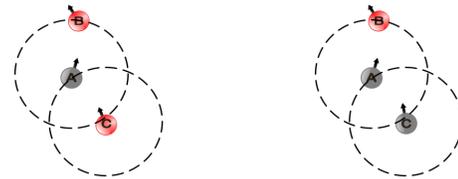
A. Método de Liderazgo Dinámico

En parte inspirado por los trabajos de Reynolds [2] y de He *et al.* [33], el primer método propuesto es *liderazgo dinámico (C-Nav-DL)*. Basado en el uso de \mathbf{v}^{goal} , este método busca identificar *líderes* que asistan al resto de los agentes a llegar a su objetivo.

Para identificar a un *líder adecuado*, cada agente debe primero determinar quiénes navegan en una dirección similar. Para esto, el agente \mathcal{A}_i debe determinar la similitud S_{ij} de su velocidad preferida al objetivo ($\mathbf{v}_i^{\text{goal}}$) respecto a la velocidad de cada agente vecino \mathcal{A}_j ($\mathbf{v}_j^{\text{goal}}$, $i \neq j$). La similitud S_{ij} se calcula mediante el producto punto entre ambas velocidades ($S_{ij} = \mathbf{v}_j^{\text{goal}} \cdot \mathbf{v}_i^{\text{goal}}$). Si S_{ij} es mayor al *umbral de similitud* (S_{thr}), el agente \mathcal{A}_j es etiquetado como un *aliado* del agente \mathcal{A}_i . A partir del conjunto de aliados de \mathcal{A}_i , el agente \mathcal{A}_j que se encuentre más cerca del objetivo de \mathcal{A}_i es el *líder* L_i . En el caso donde no se encuentre algún agente \mathcal{A}_j cuyo $S_{ij} > S_{thr}$, entonces el agente \mathcal{A}_i se convierte en su propio líder ($L_i = \mathcal{A}_i$).

Los agentes que son *líderes* utilizan *C-Nav* normal, por lo que su capacidad de llegar al objetivo es la misma que con dicho método. Por otro lado, los agentes que son *seguidores* pueden tener un comportamiento distinto a *C-Nav* dada la acción extra de seguimiento al líder. Sin embargo, a medida que los agentes *líderes* llegan a su objetivo, los agentes seguidores irán gradualmente cambiando su líder (ya que se convierten en *líderes*, o se mueven sin un líder), hasta que ya no haya agentes *seguidores*. En este momento, la tarea de navegación se completa usando el método *C-Nav*.

En este trabajo, la propiedad de *líder* es *transitiva* entre el conjunto de agentes conectados mediante dicha definición. Es decir, si un agente \mathcal{A}_i percibe como líder a un *vecino aliado* \mathcal{A}_j , el cual tiene a otro agente como líder ($L_j = \mathcal{A}_{k \neq j}$), entonces \mathcal{A}_i asume a L_j como el suyo ($L_i = L_j$). Esto lleva a que los agentes aliados que estén entre un agente y un líder hagan la labor de *repetidores* del líder en cuestión (Fig 2).



(a) $L_C = C$ (b) $L_C = B$
Fig. 2. C , al percibir al líder de A , decide seguir a B .

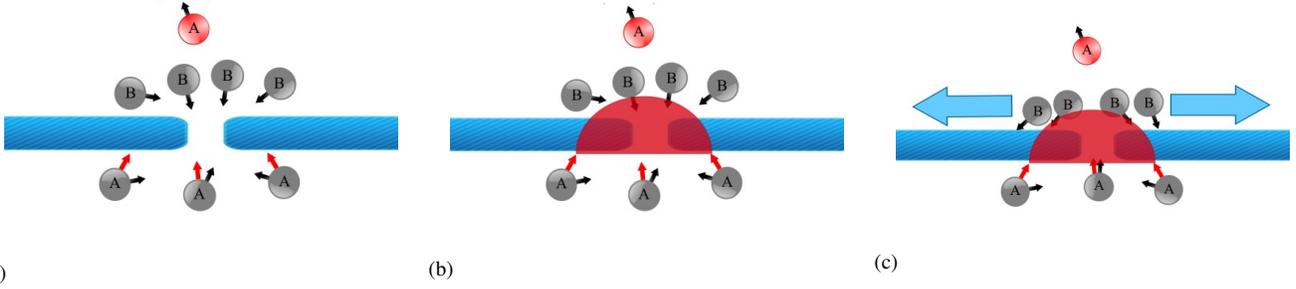


Fig. 3. Resolución de deadlock mediante alternancia de velocidades.

Es esta transitividad lo que permite que *C-Nav-DL* sea eficiente. Sea a^* la acción con mayor recompensa para un agente (según Eq. 2) entre las acciones disponibles para *C-Nav* (Fig. 1), y sea a^L la acción de seguimiento a su vecino A_j que es su líder, L_i en *C-Nav-DL*. La acción a^L será escogida por el agente siempre que $S_{ij} > S_{thr}$. Por lo tanto, este método aumenta las acciones de *C-Nav* en una dirección prometedora, ya que el líder L_i se encuentra entre el agente A_i y su objetivo. Esto permite al agente considerar acciones con un posible mejor valor de recompensa que las acciones disponibles para *C-Nav*. Por lo tanto, este método realiza una optimización local más refinada que *C-Nav*.

El modelo de comunicación de este método consiste en que cada agente A_i comparte, mediante broadcast: (1) su agente líder L_i y (2) su velocidad preferida hacia el objetivo v_i^{goal} (reemplazando la línea 10, Alg. 1). Con esta información compartida, cada agente puede determinar a sus aliados y escoger un líder que puede ser común a varios agentes. En caso de haber un líder en común, éste se encargará de *guiar* al resto de manera implícita, utilizando *C-Nav*, procesando de forma autónoma sus percepciones a medida que se mueve hacia su objetivo.

B. Método de Alternancia de Velocidades

El segundo método propuesto es la *alternancia de velocidades (C-Nav-VA)*, que busca evitar aglomeración a través del constante cambio de dirección (velocidades) de navegación de un agente. Con aspectos comunes con el método *Adaptive Learning for Multi-Agent Navigation (ALAN)* [35], *C-Nav-VA* buscar ser eficiente en ambientes que son restrictivos y dinámicos, en los cuales *C-Nav* puede llevar a un *deadlock*. Cuando la acción localmente óptima a^* (Eq. 2) para dos agentes es traducida a velocidades libre de colisión en sentido contrario, se puede producir un *deadlock*. Como *C-Nav* tiene una política *greedy* para escoger la acción localmente óptima, este *deadlock* puede potencialmente extenderse en el tiempo hasta que las condiciones cambien (Fig. 3a). Para evitar esto, *C-Nav-VA* cambia de estrategia de navegación para cada Δt , forzando a los agentes a distanciarse (Figs. 3b y 3c).

Para que un agente logre llegar a su objetivo, *C-Nav-VA* debe evitar que el cambio de velocidades entre el instante t y el $t + 1$ lleven a un agente a volver a su punto inicial ($v_i^{pref}(t) + v_i^{pref}(t + 1) = 0$). Considerando que en el instante t se utilice *C-Nav*, es necesario que para $t + 1$ se utilice una estrategia que sea complementaria a dicho método. Se eligió a

C-Nav-DL porque, como es mostrado en la sección A, provee posibilidades de acción fuera del *campo de visión* de *C-Nav*.

Si se utiliza *C-Nav-DL*, en el instante $t + 1$, un agente puede ser *líder* o *seguidor*. Cuando el agente es *líder*, la navegación se reduce a *C-Nav*, ya que aplica la misma estrategia que para el instante t . En el caso donde el agente es *seguidor*, su $v_i^{pref}(t + 1)$ es hacia L_i . Esto implica que $v_i^{pref}(t)$ y $v_i^{pref}(t + 1)$ pueden ser muy similares en dirección y opuestos en sentido, por lo que es posible que $v_i^{pref}(t) + v_i^{pref}(t + 1) \approx 0$. Sin embargo, esta situación de *oposición de velocidades* es difícil que se mantenga en el tiempo debido al dinamismo del medio. Por una parte, $v_i^{pref}(t)$ en *C-Nav* depende de las acciones que tome A_j , por lo que cualquier cambio en el entorno puede llevar a $v_i^{pref}(t) \neq v_i^{pref}(t + 2)$. Por otra parte, $v_i^{pref}(t)$ en *C-Nav-DL* depende de L_i , por lo que si $S_{ij} < S_{thr}$ (donde $L_i = A_j$) en $t + 2$ va a llevar a un nuevo L_i , lo que implica $v_i^{pref}(t) \neq v_i^{pref}(t + 2)$.

Esto permite evitar *deadlocks* producidos por acciones óptimas locales al forzar la navegación en consideración a un líder, mientras que dirige al agente hacia su objetivo. Por lo que para este trabajo, *C-Nav-VA* utilizará la acción sugerida por *C-Nav* para el instante t , mientras que utilizará la acción sugerida por *C-Nav-DL* para el instante $t + 1$.

Este método tiene un modelo de comunicación en el que cada agente A_i comparte: (1) su agente líder L_i , (2) la velocidad preferida v_i^{pref} y (3) la velocidad preferida hacia el objetivo v_i^{goal} (con lo que reemplaza la línea 10, Alg. 1).

C. Método de Penalización Basado en Densidad

La congestión generada por muchos agentes moviéndose en distintas direcciones, en espacios reducidos, dificulta su avance hacia sus objetivos individuales. El tercer método, *penalización por densidad (C-Nav-DP)*, implementa un modelo de comunicación basado en el concepto de densidad en el vecindario de cada agente, y les motiva a moverse a áreas de baja densidad. Este método guarda similitud con otros métodos propuestos para navegación social entre peatones [36], [37].

La densidad d_i de un agente A_i corresponde al número de agentes vecinos que se encuentren dentro de una circunferencia con centro en dicho agente, y con un *radio de detección de densidad* r_{dens} , donde $r_{dens} < r_{perc}$. Un ejemplo de la aplicación de este radio se puede apreciar en la Fig. 4a, donde el agente A tiene a otros 4 agentes dentro de su radio de

detección de densidad, por lo que, contándose a sí mismo, posee una densidad $d_A = 5$.

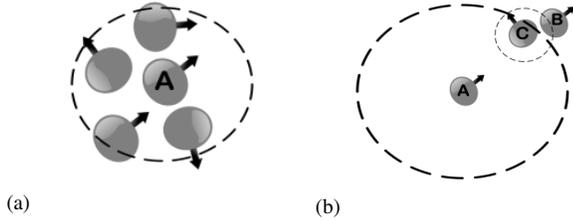


Fig. 4. (a) El agente A se encuentra en una zona de alta densidad, con $d_A = 5$. (b) El agente C le comunica su densidad $d_C = 2$ al agente A (que incluye a un agente, B , que A no puede percibir).

C -Nav-DP modifica el mecanismo de C -Nav para determinar la acción óptima local, a^* . En particular C -Nav-DP redefine la *amabilidad* de una acción (Ec. 4) como la característica que motiva al agente a moverse a zonas de baja densidad. Para un agente \mathcal{A}_i y un agente vecino cualquiera \mathcal{A}_j , NVC_{ij} (Ec. 1) se calcula de la siguiente manera:

$$NVC_{ij} = v_i^{\max} - \|\mathbf{v}_j^{\text{pref}} - \mathbf{v}_j^{\text{new}}\| \cdot fp \quad (5)$$

donde el *factor de penalización* (fp) penaliza o bonifica la diferencia entre la velocidad preferida \mathbf{v}^{pref} y la velocidad libre de colisión \mathbf{v}^{new} de cada agente vecino a \mathcal{A}_i , dependiendo de la relación entre la densidad de \mathcal{A}_i (d_i) y la de un agente vecino \mathcal{A}_j (d_j). Si $d_i < d_j$, entonces fp toma un valor alto (cercano a 2) con lo que incrementa artificialmente el efecto que la acción evaluada tiene sobre \mathcal{A}_j . Esto incrementa el valor de NVC_{ij} y hace que la recompensa de la acción evaluada disminuya. Por otro lado, si $d_i > d_j$, entonces fp toma un valor muy bajo (cercano a 0.01) para disminuir artificialmente el efecto que la acción evaluada tiene sobre \mathcal{A}_j . Esto disminuye el valor de NVC_{ij} y hace que la recompensa de la acción evaluada aumente. Finalmente, los valores de NVC son normalizados entre 0 y 1.

Esta nueva definición de comportamiento *amable* es análoga a la definición utilizada en C -Nav. Supongamos que el agente \mathcal{A}_i tiene como vecinos, entre otros, a los agentes \mathcal{A}_m y \mathcal{A}_k . Supongamos además que el vecino \mathcal{A}_m se encuentra en una zona de alta congestión, y posee una densidad mayor que la de \mathcal{A}_i , es decir, $d_i < d_m$, mientras que el vecino \mathcal{A}_k está en una zona de baja congestión, con una densidad menor que la de \mathcal{A}_i , es decir, $d_i > d_k$. En C -Nav, \mathcal{A}_i intenta moverse a su objetivo de forma tal que \mathcal{A}_m y \mathcal{A}_k , puedan moverse de acuerdo a su intención de movimiento. Sin embargo, \mathcal{A}_m , al encontrarse en una zona de alta congestión, deberá priorizar evitar colisiones antes que moverse hacia su objetivo, por lo que es muy probable que el efecto individual de la acción *amable* de \mathcal{A}_i sobre \mathcal{A}_m sea leve e incluso nulo. Por esta razón, y para no empeorar su situación, C -Nav-DP reduce en \mathcal{A}_i el valor de NVC_{im} y, por consecuencia, el valor de la acción que afecta (a través de su interacción) a dicho vecino. Por otro lado, la acción de \mathcal{A}_i tendrá más impacto en \mathcal{A}_k ya que, al estar en una zona de baja densidad, es más probable que posea un mayor control sobre sus movimientos, por lo que el comportamiento *amable* de C -Nav tendrá potencialmente un

mayor efecto sobre este vecino. Por esto, C -Nav-DP aumenta el valor de NVC_{ik} y, por consecuencia, el valor de las acciones que impliquen una interacción de \mathcal{A}_i con \mathcal{A}_k .

De esta forma, C -Nav-DP incentiva que los agentes se muevan hacia sus objetivos actuando de manera *amable* con sus vecinos en zonas de alta densidad, evitando acercarse a ellos. Esto tiene dos consecuencias: los agentes pueden acercarse más a sus objetivos al estar en zonas menos densas, ya que están menos restringidos y, fruto de esto, disminuyen la densidad para el resto de los agentes.

Por otro lado, al cambiar C -Nav-DP la forma en que se calcula la recompensa de cada acción con respecto a C -Nav, la acción escogida en ambos métodos puede ser diferente, pudiendo incluso llevar a un agente a alejarse de su objetivo si la densidad en otras direcciones es alta. Por lo tanto, este método llevará a los agentes a sus objetivos en las mismas condiciones de C -Nav siempre que: 1) el factor de coordinación (γ) en la ecuación 2 sea menor a 1, de tal manera que los agentes tengan tolerancia a un mínimo de densidad y sigan moviéndose hacia el objetivo (evitando acciones muy conservadoras) y 2) que existan, en el escenario de navegación, un subconjunto de agentes que puedan llegar a su objetivo sin interactuar en zonas de alta densidad.

En el modelo de comunicación implementado por este método, cada agente \mathcal{A}_i comparte: (1) su densidad d_i y (2) su intención de movimiento $\mathbf{v}_i^{\text{pref}}$, con sus agentes vecinos. En la Fig. 4b, el agente C le comunica a su agente vecino A su densidad, $d_C=2$, la cual A no podría calcular por sí mismo dado que el agente B - contabilizado por C - está fuera de su rango de detección.

VI. EXPERIMENTOS

En esta sección, presentamos detalles sobre la evaluación de los métodos propuestos, junto con sus resultados y discusión.

A. Escenarios de Evaluación

Para poder determinar la efectividad de los métodos propuestos en este trabajo, se han simulado diferentes escenarios de evaluación. La simulación fue implementada en $C++$ usando la biblioteca RVO2. La evaluación se enfoca en cómo los métodos afectan la navegación, por lo que consideramos las siguientes condiciones ideales: los agentes son holonómicos, la comunicación entre ellos es directa e instantánea (dentro de sus respectivos rangos de percepción), y los sensores perciben perfectamente las características del ambiente local en el momento de la medición. Estas condiciones pueden extenderse a robots no-holonómicos diferenciales, con sensores imperfectos, siguiendo los ejemplos de [23], [22], [7].

Finalmente, los parámetros críticos, tanto de los métodos propuestos como de C -Nav, fueron determinados a partir de múltiples ejecuciones de los distintos escenarios de evaluación. Se utilizaron como valores iniciales de referencia los presentados por Godoy *et al* [7], los cuales fueron ajustados a partir de 50 iteraciones de la ejecución de los métodos. Estos valores son: $S_{thres} = 2$, $T = 2$, $r_i = 0,5m$, $v_i^{\max} = 1,5m/s$, $|\mathcal{N}| = 10$, $r_{perc} = 10m$, $\Delta t = 0,1s$ y $r_{dens} = 2,5m$.

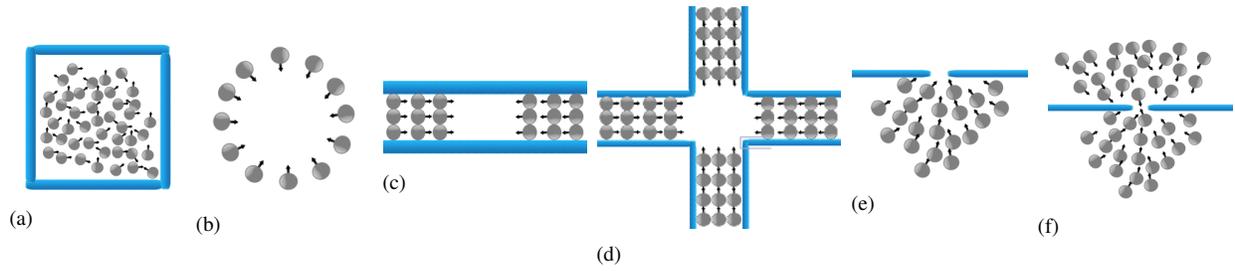


Fig. 5. Descripción de escenarios de evaluación. Con la excepción del último caso, estos escenarios fueron seleccionados del conjunto de evaluación propuesto por Godoy *et al* [7].

B. Escenarios

Para la evaluación, se consideraron cinco escenarios de navegación, presentados por Godoy *et al* [7], además de un sexto escenario que es una extensión natural de los previos cinco (Figura 5). Los escenarios son descritos a continuación:

1. CROWD: 300 agentes, con ubicación inicial y objetivo aleatoria, se mueven en un cuadrilátero delimitado por paredes (Figura 5a).
2. CIRCLE: 64 agentes deben moverse a sus posiciones antípodas en un círculo (Figura 5b).
3. BIDIRECTIONAL: dos grupos de 9 agentes se mueven en dirección opuesta, dentro de un área delimitada por dos paredes laterales (Figura 5c).
4. CROSSING: 80 agentes son divididos en cuatro grupos que son ubicados en las extremidades del escenario. Los agentes son orientados, y su objetivo, hacia el grupo opuesto (Figura 5d).
5. CONGESTED: 32 agentes son ubicados en un área que tiene una apertura en el medio, y deben salir a través de dicha apertura que sólo permite el tránsito de un agente a la vez (Figura 5e).
6. BIDIRECTIONAL CONGESTED: El área es idéntica que en el escenario anterior (Congested). La diferencia consiste en que los agentes (30 en total) son divididos en dos grupos, cada grupo ubicado en una de las dos áreas, y con objetivos en el área contigua (Figura 5f).

C. Métrica de Evaluación

Al igual que en [7], se usa como métrica de evaluación el *tiempo de interacción (TI)*, propuesto por Godoy *et al.* [35]. Esta métrica permite evaluar el tiempo que los agentes ocupan en interacciones, tanto con otros agentes como evadiendo obstáculos, independiente de otras características de cada escenario. El *TI* de un conjunto de agentes \mathcal{A} , se define de la siguiente forma:

$$TI(\mathcal{A}) = TTime(\mathcal{A}) - MinTTime(\mathcal{A}) \quad (6)$$

donde *TTime* mide el tiempo de viaje del conjunto de agentes \mathcal{A} en un escenario y *MinTTime* es la cota máxima del tiempo mínimo (teórico) en el que los agentes pueden llegar a sus objetivos en cada escenario, cuando estos no interactúan entre ellos y viajan a máxima velocidad. Ambos, *TTime* y *MinTTime* consideran tanto el tiempo promedio de viaje μ

del conjunto de agentes como su dispersión σ , y se calculan de la siguiente forma:

$$TTime(\mathcal{A}) = \mu(GT(\mathcal{A})) + 3\sigma(GT(\mathcal{A})) \quad (7)$$

$$MinTTime(\mathcal{A}) = \mu(MinGT(\mathcal{A})) + 3\sigma(MinGT(\mathcal{A})) \quad (8)$$

donde *GT* y *MinGT* son los conjuntos de tiempos de navegación real y mínimo, respectivamente, para los agentes en \mathcal{A} . Al incorporar el tiempo mínimo necesario para navegar un escenario, el tiempo de interacción *TI* permite evaluar de forma más adecuada el aporte de cada método evaluado.

D. Resultados y Discusión

En la Tabla I se presentan los resultados de la evaluación realizada sobre los tres modelos implementados en este trabajo, en cada uno de los escenarios indicados en la Figura 5. Para cada método se presenta el promedio y la desviación estándar de su tiempo de interacción (*TI*).

El método *C-Nav-VA*, es más efectivo en los escenarios *Crowd*, *Crossing* y *Bidirectional Congested*. En estos escenarios, ocurre una alta congestión lo que puede llevar a *C-Nav* hacia un atascamiento entre agentes que van en direcciones contrarias, sin poder elegir una acción que le permita romper la simetría que impide a los agentes moverse hacia una posición diferente de la actual durante un periodo de tiempo. *C-Nav-VA*, por otra parte, alterna entre la acción recomendada por *C-Nav* y *C-Nav-DL*, rompiendo la simetría de acciones, forzando a los otros agentes a efectuar movimientos evasivos que no serían considerados si sólo utilizaran *C-Nav*. Lo anterior tiende a disminuir el tiempo de duración de situaciones de congestión, reduciendo así el *TI* con respecto a *C-Nav*. Sin embargo, *C-Nav-VA* tiene un peor desempeño en el escenario *Circle*, que se caracteriza por tener periodos de libre circulación de los agentes (sin congestión), en los cuales la alternancia de velocidades se traduce en un progreso más lento hacia sus objetivos. Se observa que *C-Nav-VA* tiene un mejor desempeño en aquellos escenarios donde, de forma inevitable, los agentes se ven obligados a enfrentar una alta congestión en algún punto de su trayecto, que involucre agentes que vayan en una misma dirección, así como también agentes en la dirección contraria; por otro lado, su desempeño se reduce en aquellos escenarios donde los agentes tengan largos trayectos libres de congestión.

El método *C-Nav-DP* muestra un mejor rendimiento en escenarios donde los agentes tienen mayor libertad para alejarse de las masas más densas de agentes. En *Congested*, *C-Nav* aglomera a todos los agentes frente al único espacio para

TABELA I

PARA CADA ESCENARIO, PRESENTAMOS EL *tiempo de interacción (TI)* PROMEDIO, MÁS LA DESVIACIÓN ESTÁNDAR DE CADA MÉTODO, EN SEGUNDOS. EL MEJOR TIEMPO PARA CADA CASO ES PRESENTADO EN NEGRITA, Y EL SEGUNDO MEJOR TIEMPO CON EL SÍMBOLO *.

	<i>C-Nav</i>	<i>C-Nav-DL</i>	<i>C-Nav-VA</i>	<i>C-Nav-DP</i>
Crowd	33,06 ± 0,17	52,66 ± 6,80	29,96 ± 0,27	* 30,51 ± 0,66
Circle	11,76 ± 0,05	37,67 ± 0,88	28,30 ± 0,06	* 13,09 ± 0,19
Bidirectional	9,79 ± 0,47	2,4 ± 0,15	* 7,72 ± 0,33	18,72 ± 1,47
Crossing	* 62,82 ± 0,57	79,54 ± 0,77	56,28 ± 0,78	69,88 ± 2,07
Congested	134,4 ± 1,48	* 112,89 ± 1,19	130,85 ± 1,44	99,57 ± 0,97
Bidirectional Congested	* 125,24 ± 0,84	387,73 ± 46,76	112,61 ± 0,88	133,61 ± 1,79

avanzar hacia el objetivo. *C-Nav-DP*, en cambio, mantiene a algunos agentes alejándose del espacio de mayor densidad, lo que disminuye la presión sobre los agentes ubicados frente a la salida, permitiéndoles salir del escenario y reduciendo la cantidad de agentes (y por ende la congestión) restantes. En el escenario *Crowd* es posible observar una lógica similar, con algunos agentes tratando de alejarse de la congestión, lo que permite al resto resolver el problema de navegación con menos agentes de lo que lo haría *C-Nav* por naturaleza. Por otro lado, este método no resulta particularmente efectivo en escenarios como *Circle*, *Crossing*, *Bidirectional* y *Bidirectional Congested* debido al poco margen de movimiento que poseen los agentes en estos escenarios. que no les permite evitar zonas densas. Por lo tanto, *C-Nav-DP* muestra mejor desempeño en aquellos escenarios donde el agente tiene el espacio suficiente para evitar las zonas congestionadas, no siendo muy efectivo en lugares estrechos que limiten mucho el movimiento, donde la amabilidad de *C-Nav* es más efectiva para resolver las congestiones que se desarrollan.

Finalmente, *C-Nav-DL* muestra una mejora significativa en los escenarios *Bidirectional* y *Congested*, con respecto a *C-Nav*. En *Bidirectional*, los agentes escogen como líder al agente más adelantado en cada grupo, formando una fila detrás de él, evitando ser arrastrados por el grupo en dirección contraria. En *Congested*, *C-Nav* aglomera a los agentes frente a la única salida, generando una disputa entre los agentes, mientras que *C-Nav-DL* genera una jerarquía entre los agentes que permite que un líder avance a la vez, reduciendo el *TI*. Este método ve su rendimiento reducido en escenarios donde la congestión impide que los seguidores puedan reunirse con su líder, como *Crowd*, donde las posiciones iniciales y la diversidad de objetivos de los agentes no suelen favorecer la labor de seguimiento. También es posible observar un peor rendimiento que *C-Nav* en escenarios donde los grupos ya conformados se topan con otros grupos de distintas direcciones, con un espacio demasiado reducido para evitarlos, como ocurre en *Crossing*, *Bidirectional Congested* y *Circle*. En resumen, *C-Nav-DL* tiene mejor desempeño en escenarios con pocos obstáculos que impidan a los líderes evitar a otros líderes y sus respectivos grupos, de lo contrario podría formarse una congestión que tardaría más tiempo en resolverse que en *C-Nav*. Este método también tiene mayor efectividad que *C-Nav* cuando no hay agentes en dirección contraria, permitiendo una coordinación pacífica, sin disputas de posición.

VII. CONCLUSIONES

En el presente trabajo, se han propuesto tres métodos que implementan modelos de comunicación, variaciones de *C-Nav*, que buscan mejorar la navegación de agentes en distintas circunstancias. Estos modelos ayudan a reducir y evitar congestiones en situaciones específicas. Los experimentos muestran que el método *C-Nav-DL* es más efectivo cuando existen grupos naturalmente formados (varios agentes viajando en una misma dirección). El método *C-Nav-VA* es más efectivo cuando los agentes poseen objetivos en distintas direcciones y se bloquean mutuamente, ayudando a resolver la congestión. Finalmente, *C-Nav-DP* muestra mejor desempeño cuando los agentes pueden moverse a zonas de baja densidad, sin perjudicar significativamente su tiempo de viaje al objetivo.

A partir de los resultados obtenidos en este trabajo, se planifica incorporar estos métodos con planificadores globales, que pueden complementar la tarea de navegación de los agentes en ambientes donde los obstáculos no sólo limitan los movimientos de los agentes, sino les impiden acercarse de manera directa a sus objetivos.

AGRADECIMIENTOS

La investigación presentada en este artículo fue apoyada por el Fondo Nacional De Ciencia Y Tecnología (FONDECYT) de ANID, Chile, mediante los proyectos de Postdoctorado No. 3170971 y de Iniciación No. 11191197.

REFERENCIAS

- [1] J. Yu and S. M. LaValle, "Structure and intractability of optimal multi-robot path planning on graphs," in *Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.
- [2] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, 1987, pp. 25–34.
- [3] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.
- [4] J. van den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-Body Collision Avoidance," in *Robotics Research*, C. Pradalier, R. Siegwart, and G. Hirzinger, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 3–19.
- [5] I. Karamouzas, B. Skinner, and S. J. Guy, "Universal power law governing pedestrian interactions," *Physical review letters*, vol. 113, no. 23, p. 238701, 2014.
- [6] A. Garcimartín, J. M. Pastor, C. Martín-Gómez, D. Parisi, and I. Zurriguel, "Pedestrian collective motion in competitive room evacuation," *Scientific reports*, vol. 7, no. 1, p. 10792, 2017.
- [7] J. Godoy, S. J. Guy, M. Gini, and I. Karamouzas, "C-nav: Distributed coordination in crowded multi-agent navigation," *Robotics and Autonomous Systems*, p. 103631, 2020.

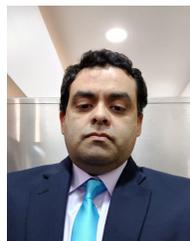
- [8] O. B. Bayazit, J.-M. Lien, and N. M. Amato, "Better group behaviors in complex environments using global," *Artificial life*, vol. 8, no. 8, p. 362, 2003.
- [9] J. Funge, X. Tu, and D. Terzopoulos, "Cognitive modeling: knowledge, reasoning and planning for intelligent characters," in *26th Annual Conference on Computer Graphics and Interactive Techniques*, 1999, pp. 29–38.
- [10] S. J. Guy, J. Chhugani, S. Curtis, P. Dubey, M. C. Lin, and D. Manocha, "Pedestrians: A least-effort approach to crowd simulation," in *Symposium on computer animation*, 2010, pp. 119–128.
- [11] S. J. Guy, S. Kim, M. C. Lin, and D. Manocha, "Simulating heterogeneous crowd behaviors using personality trait theory," in *Proceedings of the 2011 ACM SIGGRAPH/Eurographics symposium on computer animation*, 2011, pp. 43–52.
- [12] M. Popelová, M. Bída, C. Brom, J. Gemrot, and J. Tomek, "When a couple goes together: walk along steering," in *International Conference on Motion in Games*. Springer, 2011, pp. 278–289.
- [13] M. Egerstedt and X. Hu, "A hybrid control approach to action coordination for mobile robots," *Automatica*, vol. 38, no. 1, pp. 125–130, 2002.
- [14] A. Filotheou, A. Nikou, and D. V. Dimarogonas, "Decentralized control of uncertain multi-agent systems with connectivity maintenance and collision avoidance," in *2018 European Control Conference (ECC)*. IEEE, 2018, pp. 8–13.
- [15] R. Olfati-Saber and R. M. Murray, "Distributed cooperative control of multiple vehicle formations using structural potential functions," in *IFAC world congress*, vol. 15, no. 1. Barcelona, Spain, 2002, pp. 242–248.
- [16] D. Cerotti, M. Gribaudo, and A. Bobbio, "Presenting dynamic markovian agents with a road tunnel application," in *2009 IEEE International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems*. IEEE, 2009, pp. 1–4.
- [17] A. Bobbio, D. Cerotti, M. Gribaudo, M. Iacono, and D. Manini, "Markovian agent models: a dynamic population of interdependent markovian agents," in *Seminal Contributions to Modelling and Simulation*. Springer, 2016, pp. 185–203.
- [18] J. H. Reif and H. Wang, "Social potential fields: A distributed behavioral control for autonomous robots," *Robotics and Autonomous Systems*, vol. 27, no. 3, pp. 171–194, 1999.
- [19] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.
- [20] S. Juniastuti, M. Fachri, S. M. S. Nugroho, and M. Hariadi, "Crowd navigation using leader-follower algorithm based reciprocal velocity obstacles," in *2016 International Symposium on Electronics and Smart Devices (ISESD)*. IEEE, 2016, pp. 148–152.
- [21] F. Muhammad, S. Juniastuti, S. M. S. Nugroho, and M. Hariadi, "Crowds evacuation simulation on heterogeneous agent using agent-based reciprocal velocity obstacle," in *2018 International Seminar on Intelligent Technology and Its Applications (ISITIA)*. IEEE, 2018, pp. 275–280.
- [22] D. Hennes, D. Claes, W. Meeussen, and K. Tuyls, "Multi-robot collision avoidance with localization uncertainty," 2012, pp. 147–154.
- [23] J. Alonso-Mora, A. Breitenmoser, M. Rufli, P. Beardsley, and R. Siegwart, "Optimal reciprocal collision avoidance for multiple non-holonomic robots," in *Distributed Autonomous Robotic Systems*. Springer, 2013, pp. 203–216.
- [24] A. Bojeri and G. Iacca, "Evolutionary optimization of drone trajectories based on optimal reciprocal collision avoidance," in *2020 27th Conference of Open Innovations Association (FRUCT)*. IEEE, 2020, pp. 18–26.
- [25] S. Curtis, S. Guy, B. Zafar, and D. Manocha, "Virtual Tawaf: A case study in simulating the behavior of dense, heterogeneous crowds," in *2011 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2011*, Dec. 2011, pp. 128–135.
- [26] R. Ostos, F. F. Ramos, B. Castillo, and V. G. Félix, "Selection of coordination mechanisms in intelligent environments," *IEEE Latin America Transactions*, vol. 13, no. 9, pp. 3120–3126, 2015.
- [27] C. Yu, M. Zhang, F. Ren, and X. Luo, "Emergence of social norms through collective learning in networked agent societies," in *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, 2013, pp. 475–482.
- [28] C. I. Mavrogiannis, W. B. Thomason, and R. A. Knepper, "Social momentum: A framework for legible navigation in dynamic multi-agent environments," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 361–369.
- [29] C. W. Warren, "Multiple robot path coordination using artificial potential fields," in *Proceedings., IEEE International Conference on Robotics and Automation*. IEEE, 1990, pp. 500–505.
- [30] R. Gayle, W. Moss, M. C. Lin, and D. Manocha, "Multi-robot coordination using generalized social potential fields," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 106–113.
- [31] S. Iqbal and F. Sha, "Actor-attention-critic for multi-agent reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2019, pp. 2961–2970.
- [32] Y. Jin, Y. Zhang, J. Yuan, and X. Zhang, "Efficient multi-agent cooperative navigation in unknown environments with interlaced deep reinforcement learning," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 2897–2901.
- [33] L. He, J. Pan, W. Wang, and D. Manocha, "Proxemic group behaviors using reciprocal multi-agent navigation," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 292–297.
- [34] F. Ducatelle, G. A. Di Caro, A. Förster, M. Bonani, M. Dorigo, S. Magnenat, F. Mondada, R. O'Grady, C. Pinciroli, P. Régnard *et al.*, "Cooperative navigation in robotic swarms," *Swarm Intelligence*, vol. 8, no. 1, pp. 1–33, 2014.
- [35] J. Godoy, T. Chen, S. J. Guy, I. Karamouzias, and M. Gini, "ALAN: adaptive learning for multi-agent navigation," *Autonomous Robots*, vol. 42, no. 8, pp. 1543–1562, 2018.
- [36] A. Vega, L. J. Manso, D. G. Macharet, P. Bustos, and P. Núñez, "Socially aware robot navigation system in human-populated and interactive environments based on an adaptive spatial density function and space affordances," *Pattern Recognition Letters*, vol. 118, pp. 72–84, 2019.
- [37] P. Henry, C. Vollmer, B. Ferris, and D. Fox, "Learning to navigate through crowded environments," in *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2010, pp. 981–986.



Jonathan Rodríguez Rebolledo Obtuvo el título de Ingeniero civil informático en la Universidad de Concepción. Sus áreas de interés son la ingeniería de software y el desarrollo de proyectos en general.



Julio Godoy Obtuvo su doctorado en Computación en la Universidad de Minnesota, USA. Actualmente es profesor asistente en la Universidad de Concepción, Chile. Sus intereses de investigación están en la intersección de inteligencia artificial distribuida y robótica.



Fernando Gutierrez Obtuvo su doctorado en Computación en la Universidad de Oregon, USA. A través de su cargo en AIDA, trabaja en la integración de técnicas de Inteligencia Artificial a procesos industriales. Sus áreas de interés son integración de representación de conocimiento en sistemas multi-agentes y Web Semántica.