

**Estudio de modelos para tráfico peatonal y su  
aplicación a pasillos en diferentes escenarios**

Tesis que presenta

**Jorge Daniel González Arostico**

Para obtener el grado de

**Maestro en Ciencias**

**(Matemáticas Aplicadas e Industriales)**

Directora de Tesis: Dra. María Luisa Sandoval Solís

Sinodales

Presidente: Dr. Joaquín Delgado Fernandez UAM-I

Secretaria: Dr. Patricia Saavedra Barrera UAM-I

Vocal: Dra. María Elena Lárraga Ramírez UNAM

División de Ciencias Básicas e Ingeniería

Departamento de Matemáticas

México, D.F., a 13 de diciembre del 2011.





UNIVERSIDAD AUTÓNOMA METROPOLITANA

---

---

DIVISIÓN DE CIENCIAS BÁSICAS E INGENIERÍA

Departamento de Matemáticas

**Estudio de modelos para tráfico peatonal y  
su aplicación a pasillos en diferentes  
escenarios**

T E S I S  
QUE PARA OBTENER EL GRADO DE:

**Maestro en Ciencias  
(Matemáticas Aplicadas e Industriales)**

P R E S E N T A:

**Jorge Daniel González Arostico**

Directora de Tesis: Dra. María Luisa Sandoval Solís

México, D.F., a 13 de diciembre del 2011.



# Agradecimientos

En primer lugar agradezco a la Dra. María Luisa Sandoval Solís el apoyo que me brindó como directora de tesis. Le agradezco su disponibilidad y calidad de sus observaciones y sugerencias. Asimismo agradezco al Dr. Joaquín Delgado, la Dra. Patricia Saavedra y la Dra. María E. Lárraga por haber revisado y proporcionado valiosos comentarios para la mejora de esta tesis.

También agradezco a mi novia Tania y mis amigos Christian y Luis Omar que siempre estuvieron conmigo y me apoyaron en todo momento.

Finalmente agradezco al proyecto “Modelos matemáticos para mejorar la operación de la Red del STC” de fondo Mixtos ICyTDF-CONACYT por el apoyo económico.

México, DF., Diciembre de 2011.  
Jorge Daniel González Arostico



# Dedicatoria

Esta tesis esta dedicada a mi preciosa madre.

*¿No es la mano gentil de tu madre la que corre las cortinas, y la dulce voz de tu madre la que te anima a levantarte y olvidar, bajo el sol brillante, los sueños horribles que tanto te asustaron cuando todo estaba en tinieblas?*

LEWIS CARROLL  
“Alicia en el Pais de las Maravillas”





# Índice general



## Introducción

Las zonas o entornos peatonales son espacios públicos o privados destinados al tránsito de personas a pie, tales como los pasillos del metro, centros comerciales, parques, aeropuertos, terminal de autobuses, centros de diversión, etc. En ocasiones estos lugares llegan a congestionarse debido a la gran demanda que tienen. Esto pone en riesgo la integridad física de las personas, ya sea en condiciones normales o en presencia de alguna emergencia donde sea necesario evacuar el inmueble.

El análisis del flujo peatonal no es algo nuevo, especialmente en el ámbito de la planificación de edificios y la protección contra incendios e incluso en el transporte, tradicionalmente ha sido enfocado mediante valores empíricos de carácter analítico como el nivel de capacidades o el diseño de estructuras.

El estudio del tráfico peatonal es importante para simular las posibles trayectoria que podría seguir los peatones ayudando así a evitar desastres durante situaciones de pánico y evacuaciones. Zainuddin et al. [?] y Sarmady et al. [?] realizaron un estudio en el Masjidal-Haram es el primer lugar santo de islam. En su centro se encuentra la Kaaba. Durante el Hajj y Umrah, los musulmanes han de circunvalar la Kaaba siete veces, en sentido antihorario. Este círculo se hace para demostrar la unidad de los creyentes en el culto del Dios Único. El objetivo de ambos trabajos es mejorar el problema del congestionamiento de personas en este lugar. El resultado del estudio sugiere varias medidas de mitigación para solucionar el problema de congestión.

Actualmente algunos países han incluido el análisis del tráfico peatonal en la planeación de construcciones obteniendo como resultado mejorar notablemente las condiciones de transporte y el movimiento de personas. Tal es el caso de China en los juegos olímpicos [?], donde registró 7-9 millones de espectadores y 10 mil atletas de 205 países. El metro adoptó una serie de medidas para asegurar la fluidez del tráfico, el cual incluyó un diseño específico de horarios de trenes y la ampliación de la operación. El resultado indica que la influencia sobre el volumen de personas que asistieron a los juegos olímpicos está significativamente asociado con el diseño del estadio olímpico.

Para mejorar el servicio y las condiciones en las zonas peatonales, primero se debe de entender el comportamiento de las personas dentro de estas instalaciones. Los estudios relacionados con el tráfico peatonal proporcionan una herramienta para la planeación y diseño de las mismas. De esta manera el rendimiento de las zonas peatonales aumentaría notablemente, en términos de eficacia, seguridad y comodidad. En general, los estudios se dividen en: recolección de datos y análisis de estos. En la recolección de datos, se recopila información acerca de las características individuales de la población en estudio, posteriormente esta información alimenta y valida la parte del análisis [?].



(a)

(b)

Figura 1.1: Metro de la Ciudad de México: (a) Estación de correspondencia Salto del agua y (b) Centro médico.

Las características del tráfico peatonal se dividen en: microscópicas y macroscópicas. El nivel microscópico involucra características individuales de los peatones, tales como sexo, edad, peso, velocidad<sup>1</sup>, etc. [?]. Por otro lado, la escala macroscópica toma en cuenta las características promedio de un conjunto de personas en una área fija: velocidad promedio, densidad promedio, etc. Existe una escala intermedia llamada mesoscópica, la cual se enfoca en las características de los grupos de personas.

Podríamos pensar que el movimiento de los peatones no tiene un orden pero sí un motivo y propósito, en este sentido se han realizado trabajos a nivel microscópico [?, ?] que muestran ciertos patrones en el flujo peatonal, ver figura ??.

En contraste a un peatón que camina solo en un pasillo, el aumento del número de personas en las instalaciones incrementa los problemas de interacción. Los estudios a nivel microscópico tratan a cada peatón individualmente y analizan el comportamiento interactivo entre las personas, en estos casos el peatón se puede modelar como una partícula. Para densidades grandes de personas los modelos macroscópicos tratan a los peatones como un fluido. De otra forma, el estudio no sería el adecuado para casos con bajas densidades de personas.

Por otra parte, el comportamiento de los peatones se describe en tres niveles: estratégico, táctico y operativo [?, ?].

1. **Estratégico:** las personas deciden una lista de actividades que desean realizar. Sin tener en cuenta su importancia, como por ejemplo, comprar boletos del metro y comprar el periódico.
2. **Táctico:** dadas las decisiones en el nivel estratégico, el nivel táctico se refiere a las decisiones a corto plazo. Los peatones determinan el orden de las actividades en la lista que desean ejecutar. El orden dependerá de la importancia de la actividad. Incluso pueden prescindir de actividades que no puedan ser completadas dentro del tiempo que disponen. Por último, se determina una ruta, con metas intermedias y destino.
3. **Operativo:** ejecuta las decisiones instantáneas acorde a lo planeado en el nivel táctico.

<sup>1</sup>Debido a que cada peatón tiene diferente velocidad al caminar esta se toma como una característica individual

Helbing [?] menciona que el comportamiento inicia con una *estimulación* seguida de un *proceso mental*, que finalmente culmina en una *reacción*. Estos estudios nos dicen que es posible comprender y por lo tanto modelar el comportamiento de los peatones. Hallar modelos que emulen el movimiento de los peatones ha sido tema de muchos investigadores [?, ?, ?, ?, ?, ?].

Una de las características que rige el comportamiento de los peatones es el efecto territorial que se manifiestan cuando las personas se mantienen alejadas unas de otras. Esta característica es el tema de investigación de los sociólogos y antropólogos desde hace mucho tiempo. En 1959, Edward Hall [?], introduce el término *proxémica*<sup>2</sup>, que describe los cuatro tipos de distancias interpersonales: distancia íntima, personal, social, pública. Helbing [?], menciona que los peatones sienten incomodidad al caminar cerca de otra persona. Esta incomodidad se incrementa de acuerdo a la distancia que existe entre los individuos. Claro está que los rangos de interacción dependen del tipo de región y/o cultura. Por ejemplo, se ha observado que los asiáticos requieren menos espacio y son más tolerantes a la invasión del espacio personal. Otro aspecto que han observado los investigadores es que los peatones con frecuencia revisan su entorno, casi siempre inconscientemente, a esto se le llama *escaneo*, de esta manera evitan obstrucciones a otras personas. Este escaneo es diferente en mujeres que en hombres, las mujeres pasan más tiempo escaneando el camino que los hombres. El escaneo se aplica para describir la interacción con otros peatones.

Existen dos tipos de momentos para que ocurra el encuentro entre dos personas.

**Emisión de signo crítico:** la comunicación entre peatones permite que cada persona dé a conocer el movimiento a realizar. Esta comunicación consiste de movimientos muy sutiles (movimiento de hombros, cabeza, mirada, etc.).

**Establecimiento del momento:** este punto describe el reconocimiento de ambas partes de las emisiones de signo crítico.

Hoogendorn et al. [?] introducen la teoría acerca del comportamiento de auto-organización de las personas dentro del tráfico peatonal. Esta teoría se refiere al "*principio del menor esfuerzo*", se basa en la suposición de que el objetivo de los peatones es minimizar su desutilidad<sup>3</sup> al caminar, ya sea acelerando, frenando, cambiando de dirección, etc.

El caso específico de flujo unidireccional donde todos los peatones tienen las mismas condiciones físicas (velocidad, talla, etc.), la distribución de estos sobre el pasillo tiende a ser homogénea, estas condiciones restringen a las personas a no rebasarse entre sí. Este resultado muestra un patrón donde los peatones permanecen detrás de otros formando capas que se superponen parcialmente, es decir, hileras escalonadas de personas. A este tipo de flujo se le llama flujo homogéneo.

Para el caso de un flujo heterogéneo (con respecto a la velocidad deseada) el cambio de velocidad o de dirección puede mejorar las condiciones individuales de los peatones. Estos pueden rebasar a otros, siempre y cuando, esto conlleve a una mejora substancial de su situación actual. Sin embargo, si tal beneficio no es suficiente entonces el peatón optará por disminuir su velocidad. La desutilidad producida por la disminución de la velocidad es compensada mediante la reducción de la probabilidad de colisionar con otras personas u obstáculos. Dependiendo de la población, para un flujo unidireccional heterogéneo, la teoría predice la formación de grupos con flujos homogéneos caminando con una misma velocidad.

En un flujo bidireccional, este tipo de fenómenos ocurre con mayor frecuencia, los peatones intensifican la búsqueda de espacios donde puedan moverse, tratando de minimizar su desutilidad. De acuerdo a esto

<sup>2</sup>El término *proxémica* describe las distancias medibles entre las personas mientras estas interactúan entre sí.

<sup>3</sup>Desutilidad concepto usado en microeconomía y teoría económica que significa utilidad negativa. Molestia, daño o satisfacción negativa que produce un cierto bien o servicio.

los peatones comienzan a integrarse en una misma línea caminando detrás de otra persona que lleva la misma dirección que él, este patrón es conocido como la formación de líneas dinámicas. La situación óptima del flujo bidireccional es cuando las líneas con misma dirección se agrupan en una corriente de flujo dividiendo así el pasillo en dos regiones. La fricción de estas corrientes se produce sólo en la frontera donde interactúan los peatones con dirección contraria, de esta manera el flujo dentro del pasillo es mayor, en comparación con las líneas dinámicas dispersas por el pasillo. Este tipo de patrones surgen en cruces de pasillos, formando estructuras homogéneas, presentándose como líneas diagonales. Estas formas plantean una situación en la cual el sistema tiene la menor desutilidad colectiva.

Es interesante observar que dependiendo del país, las corrientes de líneas dinámicas tienden a formarse del mismo lado donde los automóviles circulan, por ejemplo en México los automovilistas eligen el lado derecho en un vía de doble sentido igual que las personas.

Una vez que hemos llegado a comprender los aspectos más importantes del comportamiento peatonal podremos desarrollar modelos que reproduzcan el movimiento de los mismos. De esta manera analizaremos las estructuras antes de ser construidas o modificar las ya existentes. Este tipo de análisis reduce costos de remodelación y ayudan a aplicar políticas en instalaciones sin tener que recurrir al costoso método de prueba y error.

A continuación se enuncian los principales modelos para tráfico peatonal:

1. **Modelos discretos:** son aquellos que por lo regular son discretos tanto en el tiempo como en el espacio [?]. Modelan al peatón como una entidad que actualiza su posición mediante una función [?, ?, ?]:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \beta) \quad (1.1)$$

donde  $\mathbf{x}_t$  es el vector de estado al tiempo  $t$  y  $\beta$  son los parámetros del modelo. La principal ventaja de estos modelos es la rapidez con la que se realizan los cálculos de actualización de posición. Sin embargo, la desventaja radica en que al tratar con una cantidad grande de peatones (miles), éstos agotan rápidamente la memoria de la computadora.

2. **Modelos continuos:** se caracterizan por definir la posición mediante una función continua en el tiempo. La mayoría de estos modelos se basan en fuerzas y la posición se determina mediante ecuaciones diferenciales [?, ?, ?, ?, ?]:

$$\frac{d\mathbf{x}}{dt} = f(\mathbf{x}(t), \beta) \quad (1.2)$$

donde la trayectoria  $\mathbf{x}(t)$  es calculada usando métodos de integración numérica. Comúnmente requieren tomar valores relativamente pequeños para el paso de tiempo  $\Delta t$ , para asegurar estabilidad.

3. **Modelos basados en agentes:** se trata de entidades con un comportamiento propio. Desarrollados en el contexto de inteligencia artificial. La modelación basada en agentes tiene un amplio uso en la simulación del tráfico peatonal. Proporciona una gran flexibilidad, ya que el comportamiento de cada elemento en el sistema puede ser modelado independientemente y puede capturar interacciones complejas. Por ejemplo, en el tráfico peatonal cada persona es un agente y su comportamiento puede modelarse como una secuencia de elecciones específicas, tal como elegir un destino, un itinerario, una dirección general o donde colocar el siguiente paso [?].

4. **Modelos de colas** basados en eventos de entradas y salidas. Estos calculan el nivel de servicio en taquillas, torniquetes, puertas, subidas y ajadas de vehículos y otros lugares donde las personas esperan recibir algún tipo de servicio. Aunque estos modelos no son apropiados para capturar la dinámica peatonal comúnmente son utilizados para la simulación de evacuaciones puesto que no toman en cuenta la geometría de edificios y obstáculos [?, ?].
5. **Modelos híbridos:** combinan las ventajas de los modelos discretos y de los continuos [?, ?, ?, ?, ?] y en su mayoría utilizan un enfoque basado en agentes. El objetivo de estos modelos es realizar simulaciones de cantidades grandes de personas en extensas áreas.

En este trabajo se estudiarán dos tipos de modelos microscópicos: el modelo de fuerza social de tipo continuo y el modelo de autómata celular de tipo discreto.

Por otro lado, si observamos el movimiento de personas desde una gran altura, podríamos notar su semejanza con un flujo de un líquido que circula a través de un conducto a un nivel macroscópico. Por lo que el flujo de peatones puede describirse en términos de las variables siguientes: la densidad promedio  $\rho$  (número de personas por metro cuadrado), la velocidad  $v$  promedio y el *flujo*  $q$  (número de personas en una área definida). El diagrama fundamental es una curva que relaciona flujo  $q$  con la densidad  $\rho$ , ver figura ???. Observemos que en densidades pequeñas, el flujo se incrementa linealmente junto con la densidad. Por el contrario, conforme la densidad aumenta, es natural que las personas comiencen a estorbarse cada vez más, provocando que comience a disminuir el flujo de personas, la distancia entre los peatones y la velocidad. En el diagrama fundamental se usan tres tipo de representaciones: flujo-densidad, velocidad-densidad y velocidad-flujo. Estas relaciones representan la misma información por lo que a partir de una relación se puede deducir los otros dos. El flujo se obtiene multiplicando la velocidad por la densidad

$$q(\rho) = \rho v. \quad (1.3)$$

Recordemos que la velocidad es un vector  $\mathbf{v}$  con dirección, sentido y magnitud. Donde la rapidez es la magnitud del vector<sup>4</sup>,  $\|\mathbf{v}\| = v$ . Existen varios diagramas fundamentales, uno de los más citados es el dado por Weidmann [?] para un movimiento unidireccional de peatones sobre un pasillo. La expresión matemática para el flujo se obtuvo a partir de un ajuste de datos empíricos.

$$v(\rho) = v_{free} \left(1 - e^{-\gamma \left(\frac{1}{\rho} - \frac{1}{\rho_{max}}\right)}\right) \quad (1.4)$$

donde  $\gamma$  es un parámetro de ajuste,  $v_{free}$  velocidad sobre un flujo libre,  $\rho_{max}$  es la densidad máxima donde no hay flujo y  $\rho$  es la densidad actual. Estudios empíricos muestran que el mejor resultado se obtiene con  $\gamma = 1.913$ ,  $v_{free} = 1.34$  (metro por segundo) y  $\rho_{max} = 5.4$  (personas por metro cuadrado) para una población alemana. Cabe mencionar que estos valores son para flujos con densidades relativamente bajas en condiciones normales y no en casos de emergencia.

Algunos puntos especiales del diagrama [?], como se muestra en la figura ?? son:

- Flujo libre  $v_{free}$ . Es la velocidad promedio si  $q(\rho) = 0$  P/ms (P: personas) y  $\rho = 0$  P/m<sup>2</sup>, es decir, es igual a la pendiente de la función  $q(\rho)$  en el origen. Los peatones no se ven afectados por ninguna interrupción en su camino y pueden ir a su velocidad deseada. La velocidad de la pendiente de la recta que uno los puntos  $(p, q(\rho))$  y el origen.

---

<sup>4</sup>Denotaremos a los vectores en negritas

- Flujo crítico  $q_c$ . Es el flujo máximo. Debido a la relación entre densidad y velocidad, el flujo máximo no quiere decir que se alcanza en la velocidad máximas.
- Densidad crítica  $\rho_c$  es la densidad cuando  $q(\rho_c) = q_c$ .
- Densidad de embotellamiento  $\rho_j$ . Ocurre cuando la velocidad  $v = 0$  m/s y  $q(\rho_j) = 0$ .
- Región estable, para  $\rho < \rho_c$ , los peatones avanzan a una velocidad libre.
- Región inestable, se forma cuando la velocidad decrece con el incremento de la densidad, es decir, para todas las velocidades en  $\rho > \rho_c$ .
- Región de congestión, es cuando  $\rho > \rho_c$ .

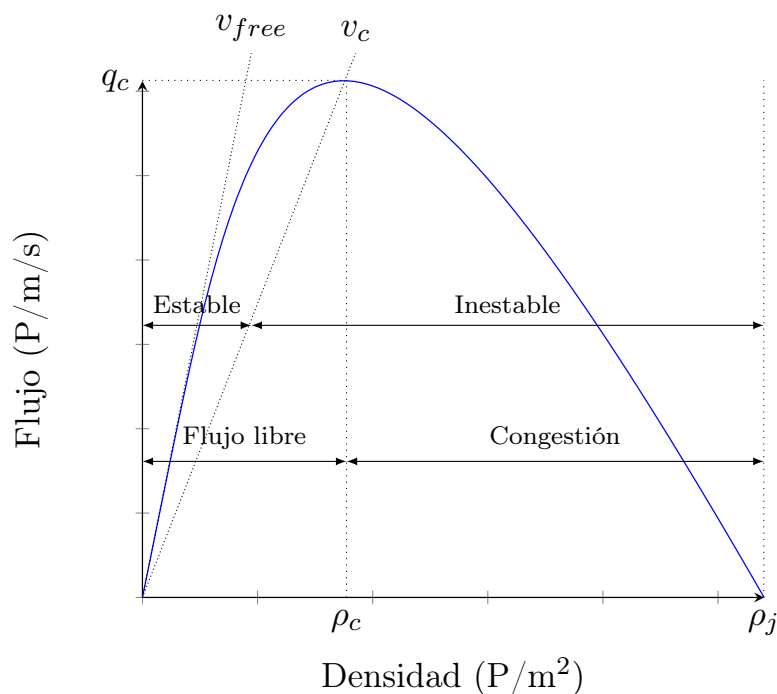


Figura 1.2: Diagrama fundamental sobre un pasillo de Weidmann: Flujo vs densidad

En particular, la calibración de los modelos de simulación del tráfico peatonal se realiza ajustando los parámetros de tal manera que los resultados obtenidos se acerquen a algún diagrama fundamental. Una vez que el modelo sea calibrado se puede predecir el nivel de servicio de alguna estructura en particular y en consecuencia tomar las medidas necesarias para optimizar el nivel de servicio.

Antes de emitir algún resultado del análisis completo de estudio, debemos comprender y reproducir los patrones que se han observado en el tráfico peatonal. En las últimas décadas muchos investigadores e ingenieros han enfocado su esfuerzo en desarrollar modelos que sean capaces de reproducir los patrones del comportamiento de las personas.

A pesar de que en la actualidad existen diversos modelos y paquetes aplicados a la simulación del tráfico peatonal, la mayoría de éstos han sido realizados en poblaciones europeas y asiáticas, ver Anexo B. México por su parte, debe planear y diseñar instalaciones utilizando como una herramienta el estudio del tráfico peatonal, de otra forma llegará a un punto en que no podrá controlar la afluencia de personas, lo cual podría resultar en desastres.



Esta investigación es motivada y financiada por el proyecto “Modelos matemáticos para mejorar la operación de la Red del STC” de fondos Mixtos ICyTDF-CONACYT.

Nuestro fin es introducirnos al campo de la modelación y simulación del tráfico peatonal con el objeto de proporcionar herramientas útiles para mejorar el tránsito de las personas en las estaciones de correspondencia y terminales de la Red del STC Metro. Para ello, nos centraremos en el estudio del movimiento de peatones sobre pasillos, cruce de corredores y cuellos de botella (escaleras, puertas, reducciones y ampliaciones de pasillos, esquinas, fuentes, etc.). Sin perder de vista los patrones colectivos observados en las últimas décadas: líneas dinámicas, líneas diagonales, oscilaciones en puertas, etc.

## 1.1. Objetivos

El propósito principal de la tesis es estudiar diversos modelos de tráfico peatonal con el fin de simular el comportamiento de las personas en pasillos con o sin cuellos de botella cuando los peatones no están en situación de pánico. Para ello, los objetivos particulares son:

1. Estudiar dos tipos de modelos microscópicos: los modelos de fuerza social y los de autómatas celulares (AC).
2. Reproducir los patrones colectivos que se han observado sobre pasillos (líneas dinámicas, diagonales, etc.)
3. Proponer y evaluar un modelo de AC llamado campo social capaz de reproducir las líneas dinámicas y representar la distancia social de las personas.

## 1.2. Estructura de la tesis

La estructura temática del texto es la siguiente. En el capítulo 2 se explican las características del modelo de fuerza social y una modificación al modelo introduciendo fuerzas granulares, este último con el fin de reproducir el diagrama fundamental de Weidmann. Este modelo contiene fuerzas granulares. Se presentan simulaciones en diferentes escenarios: pasillos con y sin obstáculos, ampliaciones y reducciones, cruces de corredores, puertas, etc.

En el capítulo 3 se recopilan los modelos de autómatas celulares para tráfico peatonal más relevantes hasta el momento, detallando las características de cada modelo. También mostramos diversas simulaciones en pasillos uni y bidireccionales utilizando los dos modelos de autómatas celulares implementados.

En el capítulo 4 presentamos nuestra propuesta, llamada modelo de campo social, basada en autómatas celulares. Este modelo se propuso con el fin de generar los patrones formados en el flujo peatonal y lograr una distancia social entre personas, debido a que en otros trabajos de AC estos dos factores no se han podido reproducir adecuadamente.

En el capítulo 5 daremos las conclusiones para los dos enfoques estudiados y proponemos algunos puntos para trabajo futuro.

El Anexo A proporciona los programas fuentes para un pasillo y una puerta usando el modelo de fuerza social y otros dos programas usando un modelo de autómatas celulares. En cada uno de ellos se describen las funciones que utilizan y el programa principal.

Finalmente, en el Anexo B se presenta una recopilación de paquetes de simulación de tráfico peatonal en condiciones normales y/o de emergencia.

# Modelo de fuerza social

## 2.1. Introducción

El modelo de fuerza social, desarrollado por Helbing et al.[?], describe el comportamiento de los peatones al caminar. Se apoya en la modelación de los cambios de conducta propuesta por Lewin [?]. De acuerdo a esta idea, los cambios del comportamiento son inducidos por los llamados “*campos sociales*” o “*fuerzas sociales*”. Sobre esta hipótesis se formula el modelo matemático donde las fuerzas (sociales) externas actúan sobre el peatón de tal manera que es motivado a moverse. El cambio de conducta estándar de los peatones es un proceso que generalmente inicia con una *estimulación*, que inducirá un *proceso mental*, lo que finalmente provocará una *reacción* al moverse.

1. **Estimulación.** Es la primera fase, depende de los objetivos personales que se desean alcanzar, de la situación y de la percepción del entorno que rodea al peatón.
2. **Proceso mental.** En esta fase se evalúan las alternativas y se toman las decisiones, incluyendo la secuencia en que serán ejecutadas. Estas decisiones se determinan básicamente con el fin de minimizar el tiempo, la distancia o aumentar la comodidad, incluso una combinación de ellas.
3. **Reacción.** Una vez dada la secuencia de decisiones en el proceso mental, el peatón reacciona moviéndose al ejecutar cada una de las decisiones. Así el movimiento (aceleración, desaceleración, dirección) del peatón es determinado, entre otras cosas, por el siguiente objetivo y la interacción con otras personas u objetos (paredes, obstáculos, etc).

### 2.1.1. Observaciones

Investigadores han estudiado video filmaciones del flujo peatonal y han descubierto patrones sobre la conducta individual que presentan los peatones en situaciones normales o de pánico.

#### Situaciones normales

Las situaciones normales (comportamiento cooperativo) con frecuencia se presentan cuando las personas realizan sus actividades planeadas con anticipación y actúan de manera natural sin ser guiadas por el pánico. Algunos patrones son:

1. Las personas caminan siempre con dirección deseada, hasta su objetivo. Rechazan la idea de moverse en dirección contraria a su dirección.
2. Con frecuencia los peatones eligen la ruta más rápida a su destino sin importar que sea la más corta ó las más cómoda. La trayectoria que forma su recorrido se aproxima a un polígono.
3. Siempre tratan de minimizar su esfuerzo para llegar a su destino.
4. La velocidad deseada individual corresponde a la velocidad más cómoda para caminar (que consuma la menor energía), de tal manera que el peatón alcance su objetivo a tiempo. Esta velocidad depende de la situación, sexo, edad, horario del día, propósito del viaje, etc.
5. Los peatones mantienen una distancia tanto a otras personas como a los bordes (calles, paredes y obstáculos). Esta distancia disminuye cuando la densidad de personas aumenta.
6. En lugares atractivos (pantallas electrónicas, ventanas, aparadores, etc.) con frecuencia hay mayor concentración de personas. Lo mismo ocurre cuando las personas se conocen entre sí, puesto que se van formando grupos.

La mayoría de las veces los turistas y niños perdidos no cuentan con una estrategia óptima de movimiento debido a que no están relacionados con el entorno que los rodea, lo que lleva a que oscilen de un lugar a otro o regresen sobre sus pasos.

### Situaciones de pánico

Las situaciones de pánico (comportamiento competitivo) se presentan cuando existe algún tipo de emergencias, tal como incendios, sismos u otra situación donde la integridad física se vea amenazada. En estas situaciones se observa un comportamiento colectivo guiado por el pánico.

1. Con frecuencia las personas se ponen nerviosas y algunas veces actúan de manera irracional, moviéndose lo más rápido como sea posible.
2. La interacción entre personas es de naturaleza física, empujándose unas a otras.
3. La presión que se registra en los atascos de personas alcanzan los 4500 Newtons por metro cuadrado, esto derrumbaría una pared o doblaría barras de acero.
4. En las salidas hay una disminución del flujo debido a las personas heridas o que están en el suelo.
5. Surge el efecto rebaño cuando las personas hacen lo mismo que aquellas que están enfrente. La mayoría de las veces este efecto provoca que las personas no tomen en cuenta las salidas de emergencia, concentrando su atención en las salidas principales.

Sin embargo, a pesar de lo anterior, todavía no se cuenta con modelos que describen exactamente el comportamiento de los peatones; primero porque los individuos pueden encontrarse con situaciones no estándares, es decir, cuando actúan compulsivamente. Segundo puede ser que aún no han aprendido una estrategia óptima, ya que no conocen otros caminos o se distraen con facilidad con el entorno que los rodea. Tercero, todas las conductas humanas muestran un grado de imperfección o irregularidad. Estas razones llevan a una desviación del comportamiento óptimo, la cuales se manejan como fluctuaciones de la conducta. Cabe mencionar que nuestro estudio, en su mayoría, se concentra en situaciones normales.

## Analogía con gases, fluidos y medios granulares

En densidades bajas, los peatones pueden moverse libremente y en multitudes se puede comparar con el comportamiento de los gases. En cambio, en densidades medianas, el movimiento de los peatones muestra sorprendentes analogías con el movimiento de fluidos y flujos granulares:

1. Las huellas de los peatones son similares a las líneas de corrientes en fluidos.
2. Entre flujos bidireccionales se observa un contacto un tanto “viscoso”.
3. La aparición de corrientes de flujo en multitudes de personas son análogas a los lechos de ríos.
4. Similar al fenómeno de segregación o estratificación en medios granulares, los peatones se organizan en líneas caminando en dirección uniforme.
5. En cuellos de botella (puertas, reducción de pasillos, escaleras) el paso del peatón es oscilatorio. Comparado con un reloj de arena.
6. La propagación ondas de choque en multitudes de personas empujándose hacia delante.
7. La obstrucción de salidas, en multitudes en pánico, es similar a la salida de medios granulares a través de pequeñas salidas.

En resumen, podríamos decir que las analogías con la dinámica de fluidos funcionan en situaciones normales, mientras que los aspectos granulares son importantes en situaciones de pánico.

### 2.1.2. Ideas básicas

#### Leyes de Newton

Debido a que el modelo de fuerza social considera las fuerzas exteriores ejercidas sobre un peatón. Se utilizan las principales leyes de la dinámica para describirlas. La primera ley de Newton se enuncia como:

“Un cuerpo en reposo o en movimiento rectilíneo uniforme permanecerá así a menos que se vea forzado al cambio debido a la aplicación de una fuerza externa.”

Es decir, ningún cuerpo por si solo puede modificar su estado por lo que requiere la manifestación de una fuerza que actué sobre él. Asimismo en el modelo de fuerza social, el peatón cambia su movimiento rectilíneo debido a las fuerzas exteriores, las cuales son ejercidas desde los objetos que se encuentran en su entorno, como por ejemplo los obstáculos, puertas, fuentes, plantas, etc. que están en su camino.

La segunda ley de Newton dice

“Cuando a un cuerpo se le aplica una fuerza constante, la aceleración resultante es proporcional a la fuerza neta que actúa sobre él e inversamente proporcional a su masa.”

La cual se refiere a los cambios en la velocidad que sufre un determinado cuerpo cuando recibe una fuerza, es decir, cuanto mayor se la magnitud de la fuerza aplicada, mayor será la aceleración. Lo mismo pasa con el peatón, las fuerzas externas ejercidas sobre él, ocasionan que esté reduzca o aumente su velocidad provocando una aceleración o desaceleración.

Finalmente, la tercera ley se expresa como

“A toda fuerza de acción se opone otra de igual magnitud en la misma dirección pero en sentido contrario.”

A diferencia de las otras leyes donde se hace referencia al comportamiento de un sólo cuerpo, la tercera ley implica dos cuerpos distintos. Sin embargo, esta ley no es válida en el modelo de fuerza social debido a que las fuerzas exteriores son ejercidas sobre el peatón en un sólo sentido y no existe otra fuerza que el peatón ejerza en sentido contrario, sobre todo de la misma magnitud.

## 2.2. Modelo

Tomando en cuenta lo mencionado anteriormente, el modelo de fuerza social describe cómo las fuerzas exteriores de su entorno, como paredes, obstáculos, personas, etc. influyen en el movimiento de un peatón. Además los objetivos personales definen una fuerza que dirige al peatón a objetivos intermedios hasta llegar a su destino final. El modelo es continuo en el tiempo y en el espacio, cuya ecuación de movimiento es una EDO para cada peatón usando la segunda ley de Newton

$$m_\alpha \frac{d\mathbf{v}_\alpha(t)}{dt} = \mathbf{F}_\alpha^T + \xi_\alpha, \quad (2.1)$$

donde  $\alpha = 1, 2, \dots$  indica el peatón;  $\xi_\alpha$  es un término de fluctuación que representa las variaciones aleatorias del comportamiento. Estas fluctuaciones se derivan de situaciones donde las personas se alejan de su comportamiento usual. El término  $\mathbf{F}_\alpha^T$  representa el efecto del entorno que influye sobre el comportamiento del peatón  $\alpha$  y está dado por

$$\mathbf{F}_\alpha^T = \mathbf{F}_\alpha + \sum_w \mathbf{F}_{\alpha w} + \sum_g \mathbf{F}_{\alpha g} + \sum_{\alpha \neq \beta} \mathbf{F}_{\alpha \beta}, \quad (2.2)$$

donde  $\mathbf{F}_\alpha$  denota la fuerza de movimiento en dirección al siguiente objetivo intermedio,  $\mathbf{F}_{\alpha \beta}$  representa la fuerza ejercida por el peatón  $\beta$  sobre el peatón  $\alpha$ ,  $\mathbf{F}_{\alpha w}$  define la fuerza ejercida por el objeto  $w$  (paredes, fuentes, plantas, etc) sobre el peatón  $\alpha$ . Finalmente  $\mathbf{F}_{\alpha g}$  representa una fuerza de atracción a personas u objetos. A  $\mathbf{F}_{\alpha \beta}$ ,  $\mathbf{F}_{\alpha w}$  y  $\mathbf{F}_{\alpha g}$  se les llaman fuerzas socio-psicológicas. Cada una de las fuerzas anteriores se describen a continuación.

### 2.2.1. Fuerza de movimiento

La fuerza de movimiento supone que el peatón  $\alpha$  con masa  $m_\alpha$  desea moverse en una dirección  $\mathbf{e}_\alpha$  con una velocidad  $\mathbf{v}_\alpha^d$  considerada como deseada, sin embargo ésta se adapta a la velocidad real  $\mathbf{v}_\alpha(t)$  al tiempo  $t$ , con un parámetro de relajación constante  $\tau_\alpha$ ,

$$\mathbf{F}_\alpha = m_\alpha \left( \frac{\mathbf{v}_\alpha^d - \mathbf{v}_\alpha(t)}{\tau_\alpha} \right). \quad (2.3)$$

Para describir la velocidad deseada debemos ver ciertos aspectos. Primero, suponemos que los objetivos intermedios son  $\{\mathbf{x}_\alpha^1, \mathbf{x}_\alpha^2, \dots, \mathbf{x}_\alpha^k, \mathbf{x}_\alpha^{k+1}, \dots, \mathbf{x}_\alpha^N\}$  tales como puertas, cruce de pasillos, cuellos de botella o lugares donde el peatón debe llegar para alcanzar su objetivo final  $\mathbf{x}_\alpha^N$ . La dirección que el peatón  $\alpha$  debe tomar para alcanzar el objetivo intermedio  $\mathbf{x}_\alpha^k$  esta dada por

$$\mathbf{e}_\alpha(t) = \frac{\mathbf{x}_\alpha^k - \mathbf{x}_\alpha(t)}{\|\mathbf{x}_\alpha^k - \mathbf{x}_\alpha(t)\|}. \quad (2.4)$$

Observemos que la trayectoria del peatón  $\alpha$  forma un polígono tal como se muestra en la figura ???. El objetivo  $\mathbf{x}_\alpha^k$  se deberá alcanzar en un tiempo  $T_\alpha$  dada su posición inicial  $\mathbf{x}_\alpha(t)$  en el tiempo  $t$ . Debido a que la distancia  $\|\mathbf{x}_\alpha^k - \mathbf{x}_\alpha(t)\|$  se deberá recorrer en el tiempo  $T_\alpha - t$ , entonces la rapidez deseada que debe llevar el peatón es

$$v_\alpha^d = \frac{\|\mathbf{x}_\alpha^k - \mathbf{x}_\alpha(t)\|}{T_\alpha - t}. \quad (2.5)$$

recordemos que la velocidad  $\mathbf{v}$  es una cantidad vectorial y la rapidez es la magnitud de la velocidad  $v = \|\mathbf{v}\|$ . Una vez dada la dirección y la rapidez obtenemos la velocidad deseada con la que se moverá el peatón

$$\mathbf{v}_\alpha^d(t) = \frac{\|\mathbf{x}_\alpha^k - \mathbf{x}_\alpha(t)\|}{T_\alpha - t} \mathbf{e}_\alpha(t) = v_\alpha^d \mathbf{e}_\alpha(t). \quad (2.6)$$

Sin embargo, la velocidad deseada está sujeta a las capacidades físicas de las personas, ya que es claro que una persona no podría recorrer una distancia muy grande en un tiempo muy corto. De modo que la rapidez está acotada como sigue:

$$v_\alpha^d(t) = \begin{cases} v_\alpha^{\min} & \text{si } v_\alpha^d(t) < v_\alpha^{\min} \\ v_\alpha^d(t) & \text{si } v_\alpha^{\min} < v_\alpha^d(t) < v_\alpha^{\max} \\ v_\alpha^{\max} & \text{si } v_\alpha^d(t) > v_\alpha^{\max} \end{cases}, \quad (2.7)$$

estos parámetros dependen de situaciones tales como la edad, sexo e incluso la cultura, y siguen una función de distribución que dependerá del tipo de evento a simular. Por ejemplo, la velocidad de los peatones que caminan sobre un pasillo se comporta como una función de distribución Gaussiana, la media y la varianza dependerá de la población, lugar, horario, etc.

### 2.2.2. Fuerza de repulsión a muros

La fuerza de repulsión  $\mathbf{F}_{\alpha w}(t)$  mantienen alejado al peatón de los bordes en las construcciones, tales como paredes, pilares, etc. Esto se debe a que las personas sienten inseguridad al caminar cerca de los bordes, ya que pueden lastimarse por estar distraídos o por alguna deformidad del borde. La fuerza de

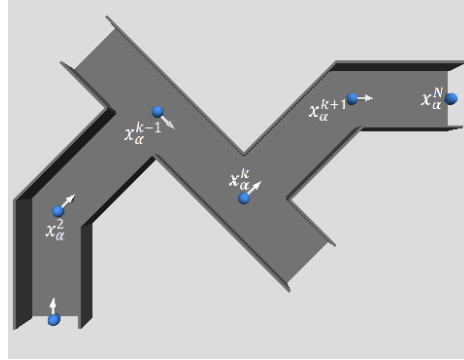


Figura 2.1: Trayectoria poligonal del peatón al alcanzar sus metas intermedias para llegar a su objetivo final.

repulsión se representa como menos el gradiente de un potencial y por lo tanto se considera como una fuerza conservativa:

$$\mathbf{F}_{\alpha w}(t) = -\nabla_{\mathbf{d}_{\alpha w}} U_{\alpha w}(\|\mathbf{d}_{\alpha w}\|) \quad (2.8)$$

con  $\mathbf{d}_{\alpha w} = \mathbf{x}_{\alpha} - \mathbf{x}_w$  el vector distancia entre la posición actual del peatón  $\mathbf{x}_{\alpha}$  y el punto  $\mathbf{x}_w$  de la pared (u obstáculo) más cercano al individuo  $\alpha$ .  $U_{\alpha w}(d)$  es un potencial de repulsión que debe ser monótono decreciente. Helbing[?] propone un potencial de la forma

$$U_{\alpha w}(\|\mathbf{d}_{\alpha w}\|) = m_{\alpha} A_w e^{-\|\mathbf{d}_{\alpha w}\|/B_w} \quad (2.9)$$

donde  $A_w$  es la intensidad de interacción entre el borde y al peatón, es decir determina el impacto de la “fuerza externa” sobre el peatón  $\alpha$  y  $B_w$  es el rango o distancia de interacción. En este caso la fuerza de repulsión a muros tiene la forma

$$\mathbf{F}_{\alpha w} = m_{\alpha} \frac{A_w}{B_w} e^{-\|\mathbf{d}_{\alpha w}\|/B_w} \left( \frac{\mathbf{d}_{\alpha w}}{\|\mathbf{d}_{\alpha w}\|} \right) \quad (2.10)$$

### 2.2.3. Fuerza de atracción a objetos

La fuerza  $\mathbf{F}_{\alpha g}$  representa la atracción que siente el peatón  $\alpha$  por el lugar  $g$  donde se encuentran otras personas (amigos, conocido, artistas, etc.) u objetos (ventanas, aparadores, locales, etc.). Sin embargo esta atracción pierde su efecto después de un determinado tiempo y él peatón continua su trayecto. Por lo que los efectos de atracción pueden ser modelados por un potencial  $W_{\alpha g}(\|\mathbf{d}_{\alpha g}\|)$  que sea monótono decreciente representado por

$$\mathbf{f}_{\alpha w}(\|\mathbf{d}_{\alpha g}\|) = -\nabla_{\mathbf{d}_{\alpha g}} W_{\alpha g}(\|\mathbf{d}_{\alpha g}\|) \quad (2.11)$$

con  $\mathbf{d}_{\alpha g} = \mathbf{x}_{\alpha} - \mathbf{x}_g$ . Los efectos de atracción son, por ejemplo, responsables de formar grupos de peatones [?]. Luego la fuerza de atracción esta dada por

$$\mathbf{F}_{\alpha g}(t) = m_{\alpha} W(\theta_{\alpha g}) \mathbf{f}_{\alpha g}(\|\mathbf{d}_{\alpha g}\|) \quad (2.12)$$



donde  $W(\theta_{\alpha g})$  es una función de peso que depende del ángulo de visión  $\theta_{\alpha g}$  del peatón  $\alpha$ . Tanto  $W$  como  $\mathbf{f}_{\alpha g}$  pueden depender de  $\alpha$  y  $g$ . La forma funcional de  $W$  se discute en el apartado siguiente.

#### 2.2.4. Fuerza de repulsión entre personas

El movimiento del peatón  $\alpha$  es influenciado por otras personas debido a que normalmente los peatones sienten incomodidad al caminar cerca de personas desconocidas. Por lo que el individuo  $\alpha$  reaccionará manteniéndose alejado de otra persona. Observemos que entre más cerca se encuentren los peatones, su incomodidad aumenta. De hecho, la distancia determina una esfera particular para cada peatón que será su zona de confort, lo cual puede interpretarse como *efecto territorial* [?]. De esta forma, la fuerza de repulsión  $\mathbf{F}_{\alpha\beta}$  ejercida por el peatón  $\beta$  al peatón  $\alpha$  se representa por

$$\mathbf{F}_{\alpha\beta}(t) = m_{\alpha}W(\theta_{\alpha\beta})\mathbf{g}(\mathbf{d}_{\alpha\beta}(t)) \quad (2.13)$$

con  $W(\theta_{\alpha\beta})$  una función de peso que depende del ángulo de visión del peatón  $\theta_{\alpha\beta}$ . La fuerza conservativa  $\mathbf{g}(\mathbf{d}_{\alpha\beta})$  representa los efectos de repulsión y está definida por

$$\mathbf{g}(\mathbf{d}_{\alpha\beta}) = -\nabla_{\mathbf{d}_{\alpha\beta}}V_{\alpha\beta}(b(\mathbf{d}_{\alpha\beta})) \quad (2.14)$$

donde  $V_{\alpha\beta}$  es monótona decreciente y depende de la distancia entre los peatones  $\mathbf{d}_{\alpha\beta}(t)$ . La forma funcional de  $W(\theta_{\alpha\beta})$  refleja el hecho que el movimiento del peatón  $\alpha$  se ven influenciado por las personas que percibe dentro de su rango de visión ( $2\theta_{\alpha\beta}$ ). Por lo tanto el peatón que lleva una dirección de movimiento  $\mathbf{e}_{\alpha}$  (vista siempre hacia el frente) evita chocar con las personas que logra ver; y aquellas que se encuentran fuera de su ángulo de visión sólo les da importancia  $c$ , con  $0 < c < 1$ :

$$W(\theta_{\alpha\beta}) = \begin{cases} 1 & , \text{ si } \mathbf{e}_{\alpha} \cdot \mathbf{g} \geq \|\mathbf{g}\| \cos(\theta_{\alpha\beta}) \\ c & , \text{ en otro caso.} \end{cases} \quad (2.15)$$

Johansson et al. [?] sugieren otra función de peso. Proponen que el ángulo formado entre los peatones  $\alpha$  y  $\beta$  se obtiene a partir del producto interior de  $\mathbf{e}_{\alpha}$  y el vector normalizado de la distancia  $\mathbf{d}_{\alpha\beta}$ , es decir,  $\cos(\theta_{\alpha\beta}) = \mathbf{e}_{\alpha} \cdot \frac{\mathbf{d}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta}\|}$ , ver figura ???. En este caso

$$W(\theta_{\alpha\beta}) = \lambda_{\alpha} + (1 - \lambda_{\alpha}) \frac{1 + \cos(\theta_{\alpha\beta})}{2}. \quad (2.16)$$

Luego la reacción del peatón a lo que pasa frente a él es más fuerte que lo que pasa a lado o detrás de él. El parámetro  $\lambda_{\alpha} \in [0, 1]$  refleja la fuerza de interacción entre los peatones que se encuentran a su alrededor. De modo que si  $\lambda_{\alpha} = 0$  entonces la influencia del movimiento del peatón  $\alpha$  dependerá sólo de los peatones que se encuentra dentro de un ángulo menor a  $90^{\circ}$ . Ahora bien, si  $\lambda_{\alpha} = 1$  el peatón  $\alpha$  se ve influenciado por todos los peatones a su alrededor. Por otro lado si  $\cos(\theta_{\alpha\beta}) = -1$  entonces el peso de la fuerza de repulsión es de  $\lambda_{\alpha}$ .

La parte más importante de la fuerza de repulsión entre peatones es  $\mathbf{g}(\mathbf{d}_{\alpha\beta})$  definida por el potencial de repulsión que mantiene separadas a las personas y evita que choquen entre ellas. Por la relevancia del potencial se presentan cuatro especificaciones en el apartado ???.

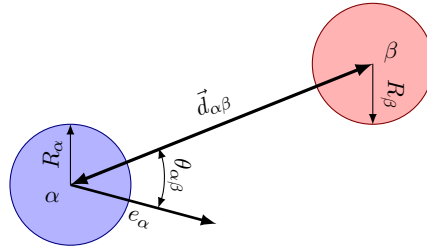


Figura 2.2: Ángulo  $\theta_{\alpha\beta}$  formado entre distancia del  $\alpha$  y  $\beta$  y la dirección de movimiento de  $\alpha$ .

### 2.2.5. Tipos de especificaciones

En este apartado se desarrollan cuatro formulaciones para el potencial de repulsión (ecuación (??)), las cuales serán de tipo exponencial

$$V_{\alpha\beta}(b_{\alpha\beta}) = A_{\alpha}B_{\alpha}e^{-b_{\alpha\beta}/B_{\alpha}}, \quad (2.17)$$

donde  $A_{\alpha}$  es la intensidad de interacción entre los peatones  $\alpha$  y  $\beta$ , y  $B_{\alpha}$  es el rango de interacción. Una diferencia importante entre las formulaciones radica en la forma de la variable  $b_{\alpha\beta}$  y los diferentes parámetros de la exponencial a partir de la geometría que se utilice. Y puesto que sólo cambia el parámetro  $b_{\alpha\beta}$  los efectos de repulsión para las diversas formulaciones se calculan de la misma manera

$$\mathbf{g}(\mathbf{d}_{\alpha\beta}(t)) = -\nabla_{\mathbf{d}_{\alpha\beta}} V_{\alpha\beta}(b(\mathbf{d}_{\alpha\beta})) = -\frac{d}{db_{\alpha\beta}} V_{\alpha\beta}(b_{\alpha\beta}) \nabla_{\mathbf{d}_{\alpha\beta}} b_{\alpha\beta}(\mathbf{d}_{\alpha\beta}). \quad (2.18)$$

#### Especificación circular

La primera formulación y la más sencilla, se llama *especificación circular*, tomando como parámetro  $b_{\alpha\beta}$  la distancia que hay entre dos peatones, definida por

$$b_{\alpha\beta} = \|\mathbf{d}_{\alpha\beta}\| - R_{\alpha} - R_{\beta}, \quad (2.19)$$

donde  $\|\mathbf{d}_{\alpha\beta}\|$  es la magnitud del vector distancia,  $R_{\alpha}$  y  $R_{\beta}$  son los radios de los peatones  $\alpha$  y  $\beta$  respectivamente, ver figura ???. Este parámetro resulta ser simétrico respecto a los cálculos, es decir, es el mismo valor tanto para el peatón  $\alpha$  como para el peatón  $\beta$ , por lo que computacionalmente resulta ser muy eficiente, ya que  $\mathbf{F}_{\alpha\beta}(t) = \mathbf{F}_{\beta\alpha}(t)$ . En este caso, los efectos de repulsión se escriben como

$$\mathbf{g}(\mathbf{d}_{\alpha\beta}(t)) = A_{\alpha}e^{-b_{\alpha\beta}/B_{\alpha}} \left( \frac{\mathbf{d}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta}\|} \right). \quad (2.20)$$

Note que esta formulación es un modelo isotrópico. Albert Steiner [?] menciona que la especificación circular se utiliza por defecto en el paquete SimWalk<sup>1</sup>. Por lo tanto la fuerza de repulsión entre personas tiene la siguiente forma:

$$F_{\alpha\beta}(t) = \left( \lambda_{\alpha} + (1 - \lambda_{\alpha}) \frac{1 + \cos(\theta_{\alpha\beta})}{2} \right) A_{\alpha}e^{(R_{\alpha}+R_{\beta}-d_{\alpha\beta})/B_{\alpha}} \frac{\vec{d}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta}\|} \quad (2.21)$$

<sup>1</sup>SimWalk es un paquete para simular tráfico peatonal con un enfoque microscópico

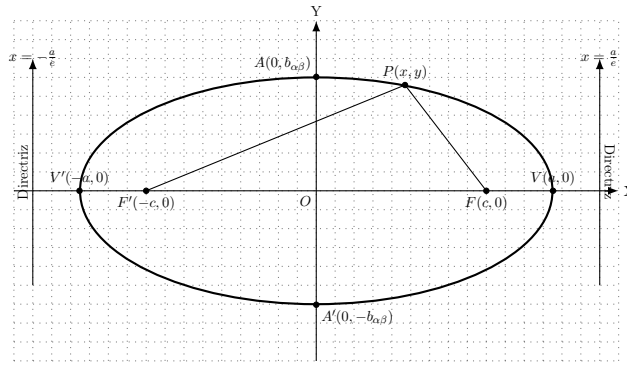


Figura 2.3: Elementos de la elipse donde se basa la especificación elíptica.

### Especificación elíptica I

Otra forma geométrica para  $b_{\alpha\beta}$ , llamada *especificación elíptica I* [?], es el eje semi menor de la elipse. En este caso el modelo es anisotrópico. La figura ?? muestra las componentes de una elipse. Donde  $F'$  y  $F$  son los focos, el segmento  $\overline{V'V'} = 2a$  se denomina eje mayor; el punto O es el centro de la elipse y  $\overline{AA'} = 2b_{\alpha\beta}$  es el eje menor. También se sabe que  $\overline{F'P} + \overline{FP} = 2a$ , si hacemos  $\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t \mathbf{e}_{\beta}\| = 2a$  y  $v_{\beta}\Delta t = 2c$ . Usando el teorema de Pitágoras se obtiene la relación

$$2b_{\alpha\beta} = \sqrt{(\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t \mathbf{e}_{\beta}\|)^2 - (v_{\beta}\Delta t)^2} \quad (2.22)$$

Esta especificación considera que  $v_{\beta}\Delta t$  es la longitud del paso del peatón  $\beta$ , donde  $v_{\beta} = \|\mathbf{v}_{\beta}\|$ . La razón de la misma se debe a que el peatón  $\alpha$  requiere espacio para su siguiente paso.

Con el fin de mostrar la sensibilidad de  $b_{\alpha\beta}$ , en la figura ?? observamos las diferentes elipses que se forman con los peatones  $\alpha$  y  $\beta$  con una distancia de  $\mathbf{d}_{\alpha\beta} = (2, 0)$ . Para las elipses en ?? mantenemos una longitud del paso  $v_{\beta}\Delta t = 1.95$ , variando las direcciones de movimiento  $\mathbf{e}_{\beta}$ . En la figura ?? mantenemos la dirección  $\mathbf{e}_{\beta} = (1, 0)$ , variando el tamaño de paso  $v_{\beta}\Delta t$ .

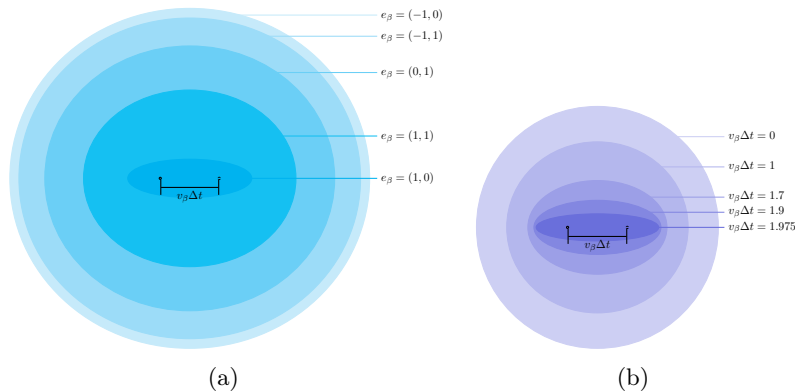


Figura 2.4: (a) Elipses resultante de la variación en la dirección de movimiento del peatón  $\beta$  y (b) de la variación en el tamaño de paso.

Antes de obtener la cantidad de repulsión para la especificación elíptica I, desarrollaremos algunos gradientes que nos ayudarán al cálculo de  $\mathbf{g}(d_{\alpha\beta})$ . Sean  $\mathbf{d}_{\alpha\beta} = (x_1, x_2)$ ,  $\|\mathbf{d}_{\alpha\beta}\| = \sqrt{x_1^2 + x_2^2}$ , y  $v_{\beta}\Delta t \mathbf{e}_{\beta} = (a_1, a_2)$ , tenemos que

$$\nabla_{\mathbf{d}_{\alpha\beta}} \|\mathbf{d}_{\alpha\beta}\| = \frac{\mathbf{d}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta}\|}. \quad (2.23)$$

De igual manera se tiene si  $\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta} = (x_1 - a_1, x_2 - a_2)$ , con  $\|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\| = \sqrt{(x_1 - a_1)^2 + (x_2 - a_2)^2}$ , entonces

$$\begin{aligned} \nabla_{\mathbf{d}_{\alpha\beta}} \|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\| &= \left( \frac{d}{dx_1} \left[ \sqrt{(x_1 - a_1)^2 + (x_2 - a_2)^2} \right], \frac{d}{dx_2} \left[ \sqrt{(x_1 - a_1)^2 + (x_2 - a_2)^2} \right] \right) \\ &= \left( \frac{x_1 - a_1}{\sqrt{(x_1 - a_1)^2 + (x_2 - a_2)^2}}, \frac{x_2 - a_2}{\sqrt{(x_1 - a_1)^2 + (x_2 - a_2)^2}} \right) \\ &= \frac{\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}}{\|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\|}. \end{aligned} \quad (2.24)$$

Puesto que el gradiente es lineal, se obtiene

$$\begin{aligned} \nabla_{\mathbf{d}_{\alpha\beta}} (\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\|) &= \nabla_{\mathbf{d}_{\alpha\beta}} (\|\mathbf{d}_{\alpha\beta}\|) + \nabla_{\mathbf{d}_{\alpha\beta}} (\|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\|) \\ &= \frac{\mathbf{d}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta}\|} + \frac{\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}}{\|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\|} \end{aligned}$$

Utilizando los cálculos anteriores y la regla de la cadena se desarrolla fácilmente  $\nabla_{\mathbf{d}_{\alpha\beta}} b_{\alpha\beta}(\mathbf{d}_{\alpha\beta})$ , luego

$$\begin{aligned} \nabla_{\mathbf{d}_{\alpha\beta}} b_{\alpha\beta}(\mathbf{d}_{\alpha\beta}) &= \frac{1}{2} \cdot \frac{2(\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\|)}{2\sqrt{(\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\|)^2 - (v_{\beta}\Delta t)^2}} \\ &\quad \cdot \nabla_{\mathbf{d}_{\alpha\beta}} (\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\|) \\ &= \frac{1}{2} \frac{(\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\|)}{2b_{\alpha\beta}} \cdot \left( \frac{\mathbf{d}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta}\|} + \frac{\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}}{\|\mathbf{d}_{\alpha\beta} - v_{\beta}\Delta t\mathbf{e}_{\beta}\|} \right) \end{aligned}$$

Finalmente, sustituyendo la ecuación anterior en la ecuación (??) obtenemos

$$\mathbf{g}(\mathbf{d}_{\alpha\beta}(t)) = A_{\alpha} e^{-b_{\alpha\beta}/B_{\alpha}} \cdot \frac{\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - \mathbf{y}_{\alpha\beta}\|}{4b_{\alpha\beta}} \cdot \left( \frac{\mathbf{d}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta}\|} + \frac{\mathbf{d}_{\alpha\beta} - \mathbf{y}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta} - \mathbf{y}_{\alpha\beta}\|} \right) \quad (2.25)$$

donde

$$\mathbf{y}_{\alpha\beta} = v_{\beta}\Delta t\mathbf{e}_{\beta}$$

Note que si  $\Delta t = 0$  en la ecuación (??), entonces los cálculos se reducen a la especificación circular. A pesar de que esta especificación podría ser más realista que la especificación circular (ecuación (??)), tiene la desventaja importante de no ser simétrica para lo peatones  $\alpha$  y  $\beta$ , lo cual aumenta la complejidad y afecta tiempo operacional.

## Especificación elíptica II

Otra formulación para el parámetro  $b_{\alpha\beta}$  es la *especificación elíptica II*, introducida por Shukla [?]. En esta especificación se consideran las velocidades  $\mathbf{v}_\alpha$  y  $\mathbf{v}_\beta$  de los peatones  $\alpha$  y el  $\beta$ . También incluye sus respectivas longitudes de paso  $v_\alpha\Delta t$  y  $v_\beta\Delta t$ . Sabemos que el peatón necesita espacio para dar el siguiente paso, por lo que podríamos decir que la especificación elíptica II toma en cuenta este tipo de necesidad. La gran ventaja de esta formulación es ser simétrica, definida por

$$2b_{\alpha\beta} = \sqrt{(\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - (\mathbf{v}_\beta - \mathbf{v}_\alpha)\Delta t\|)^2 - [(v_\beta - v_\alpha)\Delta t]^2}. \quad (2.26)$$

sustituyendo la siguiente igualdad para la derivación del potencial.

$$\mathbf{y}_{\alpha\beta} = v_\beta\Delta t\mathbf{e}_\beta - v_\alpha\Delta t\mathbf{e}_\alpha$$

## Comparación entre especificaciones

A continuación se presentan tres tipos de escenarios con el fin de hacer una comparación entre las especificaciones antes mencionadas [?].

- Escenario 1.** Consideremos dos peatones  $\alpha$  y  $\beta$  con una distancia  $\|\mathbf{d}_{\alpha\beta}\|$  entre ellos, si el peatón  $\alpha$  se mueve a una velocidad constante  $v_\alpha$  en dirección  $\mathbf{e}_\alpha$ , mientras que el peatón  $\beta$  permanece inmóvil, es decir, tendrá una velocidad de  $v_\beta = 0$ . Por lo tanto la ecuación (??) se reduce a

$$2b_{\alpha\beta} = \sqrt{(\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} + v_\alpha\Delta t\mathbf{e}_\alpha\|)^2 - [v_\alpha\Delta t]^2} \quad (2.27)$$

la ecuación es parecida a la ecuación (??), sin embargo esta toma en cuenta la velocidad del mismo peatón  $\alpha$ .

Caso 1. Un primer caso donde  $\mathbf{e}_\alpha$ , la dirección de movimiento del peatón  $\alpha$ , se dirige hacia el peatón  $\beta$ , como se muestra en la figura ???. Entonces  $\|\mathbf{d}_{\alpha\beta}\| > \|\mathbf{d}_{\alpha\beta} + v_\alpha\Delta t\mathbf{e}_\alpha\|$ , esto disminuye a  $b_{\alpha\beta}$  lo que indica que la fuerza de repulsión entre ellos crece.

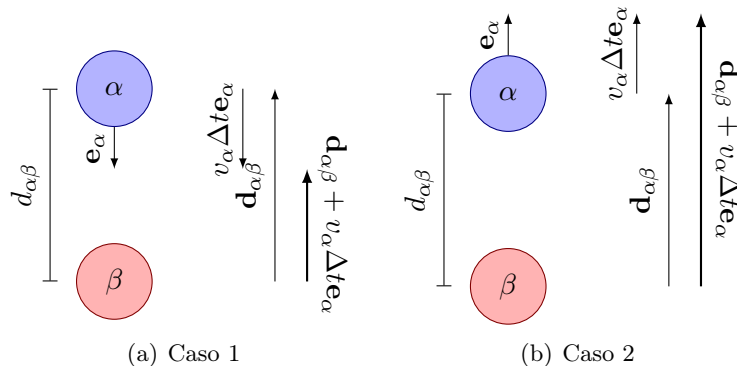


Figura 2.5: Vector resultante cuando  $\alpha$  y  $\beta$  (a) se acerca entre ellos y (b) cuando se aleja.

Caso 2. Contrario al primer caso, si el peatón  $\alpha$  se aleja en dirección contraria al peatón  $\beta$ , como se exhibe en la figura ??, por lo que  $\|\mathbf{d}_{\alpha\beta}\| < \|\mathbf{d}_{\alpha\beta} + v_\alpha \Delta t \mathbf{e}_\alpha\|$ . Entonces la magnitud del parámetro  $b_{\alpha\beta}$  crece y en consecuencia la fuerza de repulsión entre ellos disminuye.

Caso 3. Si la dirección del peatón  $\alpha$  es paralela al peatón  $\beta$ , como se observa en la figura ?. La magnitud  $\|\mathbf{d}_{\alpha\beta} + v_\alpha \Delta t \mathbf{e}_\alpha\|$  es mayor que en el primer caso, pero menor al caso anterior. Por lo tanto la fuerza de repulsión para esta distancia se encuentra entre las fuerzas de los dos casos anteriores.

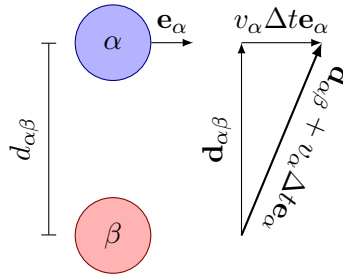


Figura 2.6: Vector resultante cuando el peatón  $\alpha$  lleva ortogonal a la distancia entre  $\alpha$  y  $\beta$ , mientras que  $\beta$  permanece inmóvil.

En contraste con la especificación elíptica I, si la velocidad del peatón  $\beta$  es igual a cero en la ecuación (??), lleva a que el parámetro  $b_{\alpha\beta}$  se reduzca a  $b_{\alpha\beta} = \|\mathbf{d}_{\alpha\beta}\|$ , similar a la especificación circular. Por lo tanto la especificación elíptica II es más realista para este escenario.

- **Escenario 2.** Si pensamos en la situación donde el peatón  $\alpha$  se dirige hacia un peatón inmóvil, con una velocidad constante ya sea de  $v_\alpha^1$  ó  $v_\alpha^2$  con  $v_\alpha^1 < v_\alpha^2$  como se muestra en la figura ?. La longitud del paso que toma el peatón son  $v_\alpha^1 \Delta t < v_\alpha^2 \Delta t$ . Esto lleva a pensar que el peatón  $\alpha$  con velocidad  $v_\alpha^2$  desacelere más rápido que si llevará la velocidad  $v_\alpha^1$ . Debido a que el parámetro  $b_{\alpha\beta}$  para la velocidad  $v_\alpha^2$  se reduce, las fuerzas de repulsión entre ellos se ven afectadas  $F_{\alpha\beta}^1 > F_{\alpha\beta}^2$ . Sin embargo con la especificación elíptica y circular son insensibles a la velocidad  $v_\alpha$  del peatón  $\alpha$ .

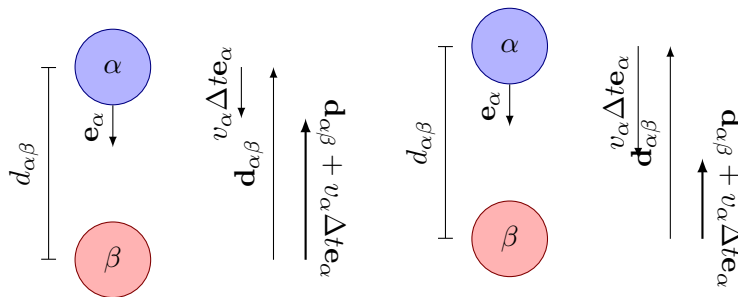


Figura 2.7: Ilustra como el peatón  $\alpha$  camina a diferentes velocidades en dirección a  $\beta$

- **Escenario 3.** Un caso más completo para el modelo elíptico II es cuando dos peatones caminan uno en dirección al otro, pero con velocidades  $v_\alpha > v_\beta$ , respectivamente, ver figura ?.

Mientras que en la especificación circular las velocidades no tienen dependencia sobre el comportamiento de los peatones, en las formulaciones elípticas lo hacen, lo cual ciertamente es más realista. El primer modelo elíptico supone una dependencia de las velocidades de los peatones, pero no es sensitiva con respecto a su propia velocidad. Por lo tanto se espera que la especificación elíptica II sea más realista.

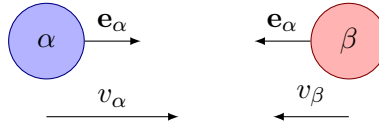


Figura 2.8: Dos peatones caminan uno hacia al otro con diferentes velocidades.

### Nueva especificación elíptica

Esta especificación, introducida por Shukla[?], toma en cuenta la velocidad absoluta, la cual tiene una influencia en los peatones

$$2b_{\alpha\beta} = \sqrt{\frac{(\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - (\mathbf{v}_\beta - \mathbf{v}_\alpha)\Delta t\|)^2 - [(v_\beta - v_\alpha)\Delta t]^2}{1 + v_\alpha\Delta t}} \quad (2.28)$$

por medio de la derivación del potencial obtenemos

$$\mathbf{g}(\mathbf{d}_{\alpha\beta}(t)) = \frac{A_\alpha e^{-b_{\alpha\beta}/B_\alpha}}{\sqrt{1 + v_\alpha\Delta t}} \cdot \frac{\|\mathbf{d}_{\alpha\beta}\| + \|\mathbf{d}_{\alpha\beta} - \mathbf{y}_{\alpha\beta}\|}{4b_{\alpha\beta}} \cdot \left( \frac{\mathbf{d}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta}\|} + \frac{\mathbf{d}_{\alpha\beta} - \mathbf{y}_{\alpha\beta}}{\|\mathbf{d}_{\alpha\beta} - \mathbf{y}_{\alpha\beta}\|} \right) \quad (2.29)$$

Hasta ahora hemos descrito las diferentes fuerzas que actúan sobre el comportamiento de los peatones, el siguiente paso es simular este comportamiento aplicando el modelo de fuerza social definido en las ecuaciones (??) y (??). Para ello hemos creado escenarios típicos donde los peatones puedan interactuar entre ellos. Estas ideas se presentan en el apartado ??.

## 2.3. Modificación del modelo

### 2.3.1. Fuerza granular

Cuando las multitudes de personas se enfrentan a una situación peligrosa se presenta un comportamiento colectivo que se convierte en una estampida, la cual es inducida por el pánico de las personas. El comportamiento individual con frecuencia genera bloqueos que pueden amenazar la vida de las personas involucradas [?]. En casos donde existe una densidad muy alta de peatones, el contacto entre ellos es mayor e incluso llegan a chocar, es aquí donde las fuerzas de empuje y fricción entran en el escenario [?]. Por lo tanto la fuerza de interacción física  $\mathbf{F}_{G\alpha}$ , entra en juego cuando los peatones están muy cerca entre ellos y existe un contacto físico ( $R_\alpha + R_\beta \geq d_{\alpha\beta}$ ). Por lo que se supone un “fuerza de cuerpo o de empuje”  $k_n(b_{\alpha\beta})\mathbf{e}_{\alpha\beta}^n$ , la cual contrarresta la compresión de cuerpo, y un desplazamiento de la fuerza de fricción  $k_t(b_{\alpha\beta})\Delta v_{\alpha\beta}\mathbf{e}_{\alpha\beta}^t$  que impide el movimiento tangencial “relativo”. Esta ecuación se inspira de las fórmulas de interacción granular [?]. Si  $b_{\alpha\beta} = (R_\alpha + R_\beta) - d_{\alpha\beta}$  y  $\mathbf{e}_{\alpha\beta}^n = \mathbf{d}_{\alpha\beta}/\|\mathbf{d}_{\alpha\beta}\| = (n_{\alpha\beta}^1, n_{\alpha\beta}^2)$  (vector normal), la fuerza granular toma la siguiente forma:

$$\mathbf{F}_{G\alpha}(t) = \sum_{i \neq j} k_n \Theta(b_{\alpha\beta}) \mathbf{e}_{\alpha\beta}^n + k_t \Theta(b_{\alpha\beta}) \Delta v_{\alpha\beta} \mathbf{e}_{\alpha\beta}^t \quad (2.30)$$

donde la función  $\Theta(z)$  es igual a  $z$ , si  $z \geq 0$ , y 0 en otro caso. Además  $\mathbf{e}_{\alpha\beta}^t = (-n_{\alpha\beta}^2, n_{\alpha\beta}^1)$  es el vector tangencial a la dirección de movimiento,  $\Delta v_{\alpha\beta}^t = (\mathbf{v}_\beta - \mathbf{v}_\alpha) \cdot \mathbf{e}_{\alpha\beta}^t$  es la diferencia de las velocidades proyectada sobre el vector tangencial, mientras que  $k_n$  y  $k_t$  son constantes muy grandes.

Por otro lado la interacción con las paredes se trata de forma análoga. Si  $d_{\alpha w}$  es la distancia a la pared  $w$ . Entonces  $\mathbf{e}_{\alpha w}^n$  denota la dirección perpendicular a la pared y  $\mathbf{e}_{\alpha w}^t$  la dirección tangencial a la misma, de modo que la fuerza de repulsión a paredes está dada por la ecuación

$$\mathbf{F}_{Gw}(t) = \sum_{i \neq j} k_n \Theta(R_\alpha - d_{\alpha w}) \mathbf{e}_{\alpha w}^n + k_t \Theta(R_\alpha - d_{\alpha w}) \Delta v_{\alpha w} \mathbf{e}_{\alpha w}^t \quad (2.31)$$

Resumiendo, el modelo de fuerza social modificado se escribe como

$$m_\alpha \frac{d\mathbf{v}_\alpha(t)}{dt} = \mathbf{F}_\alpha + \mathbf{F}_{\alpha\beta} + \mathbf{F}_{\alpha w} + \mathbf{F}_{G\alpha} + \mathbf{F}_{Gw} \quad (2.32)$$

### 2.3.2. Área de respeto

Incluir las fuerzas de interacción físicas  $\mathbf{F}_{G\alpha}$  y  $\mathbf{F}_{Gw}$  en el modelo de fuerza social no es suficiente para reproducir el diagrama fundamental de Weidmann. Parisi et al. [?] argumenta que el modelo de fuerza social es modelo competitivo y que en condiciones normales para peatones con velocidades bajas continua con este comportamiento competitivo, es decir, un peatón podría empujar a otro peatón que viene de frente. Sin embargo, en la realidad un peatón en estado normal puede pararse antes de hacer un contacto físico. Por lo que ellos consideran que cada persona tienen asociada un área de respeto. De esta manera el peatón disminuye su velocidad gradualmente siempre que otra persona viole esta área. Este mecanismo de auto detención evita que el peatón continúe empujando a otras personas cuando existen condiciones con densidades normales.

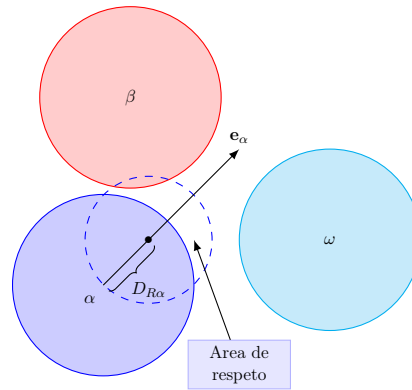


Figura 2.9: Geometría del área de respeto del peatón  $\alpha$ , donde el peatón  $\beta$  viola el área de respeto  $\alpha$ .

El área de respeto está delimitada por una circunferencia de radio  $D_{R\alpha} = R_F R_\alpha$ , donde  $R_\alpha$  es el radio del peatón y  $R_F$  el factor de respeto, cuyo centro se encuentra a una distancia  $D_{R\alpha}$  del centro del mismo en la dirección deseada ( $\mathbf{e}_\alpha$ ), ver figura ???. Si alguna persona toca esta área de respeto, la velocidad deseada  $v_\alpha^d$  del peatón  $\alpha$  se reduce a cero. Ésta velocidad se recupera cuando no hay otros individuos en el área de respeto. Observemos que la velocidad actual del peatón no es cero ( $v_\alpha \neq 0$ ) y es la que da el movimiento, hasta que la velocidad deseada se recupera. Con esto se produce una desaceleración continua en lugar de que el individuo pare repentinamente.



### 2.3.3. Región de interacción

Debido a que el costo computacional es muy grande si se toman en cuenta todas las fuerzas que interviene para cada uno de los peatones, definimos la región de interacción como un circunferencia alrededor de una persona que determina el territorio donde ésta siente con mayor intensidad las fuerzas que intervienen en el sistema, ver figura ???. Una idea similar se encuentra en Johansson et al. [?] para programación en paralelo. Notemos que la región de interacción no es lo mismo que la distancia social o zona de confort.

En nuestro caso consideramos que las personas mantienen una distancia pública y por tanto el radio de la región interacción es de 3 m [?].

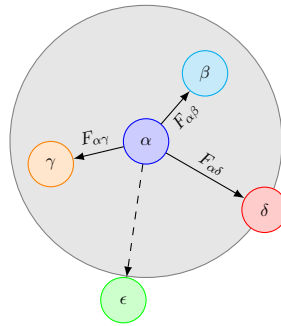


Figura 2.10: Región de interacción. El peatón  $\alpha$  solo mantiene una interacción con aquellos peatones que se encuentran cerca de él. No hay fuerza de repulsión entre  $\alpha$  y  $\epsilon$

## 2.4. Simulaciones numéricas

En este apartado se presentan principalmente las simulaciones numéricas de diversos escenarios donde los peatones interactúan entre ellos. El modelo que se utiliza es el de fuerza social definido en las ecuaciones (??) y (??). La programación se ha realizado en la plataforma llamada *Python*, ver Anexo A.

A pesar de que Keith Still [?] propone un ancho de hombros de 45.58 cm, consideramos en todas los ejemplos, 80 cm de ancho de hombros, con el fin de que se viera los peatones. Las velocidades deseadas  $v^d$  siguen una distribución Gaussiana con media  $\bar{v}_d = 1.34 \frac{m}{s}$  y una desviación estándar  $0.26 \frac{m}{s}$ , [?, ?, ?]. Por simplicidad no se consideran las fuerzas de atracción  $\mathbf{F}_{\alpha g}$  ni las fluctuaciones  $\xi(t)$  en las ecuaciones (??) y (??). Usamos el método de Euler para calcular la velocidad de cada peatón en el tiempo  $t + \Delta t$  como

$$\mathbf{v}_\alpha(t + \Delta t) = \mathbf{v}_\alpha(t) + \mathbf{F}_\alpha^T(t)\Delta t$$

donde  $\Delta t$  es el tamaño de paso. Después se calcula la posición del peatón con

$$\mathbf{x}_\alpha(t + \Delta t) = \mathbf{x}_\alpha(t) + \mathbf{v}_\alpha(t)\Delta t.$$

Las condiciones de frontera empleadas en este modelo son abiertas, de tal manera que las personas que salen del pasillo son eliminadas del sistema. El número de peatones que entran al sistema se calcula mediante una distribución uniforme entre 0 y 6, con diferentes intervalos de tiempo según el escenario.

Para todos los ejemplos numéricos se consideran los mismos valores de los parámetros que en Helbing [?]:  $A_w = 10 \frac{m^2}{s^2}$  y  $B_w = 0.2$  m. El tamaño de paso  $\Delta t$  es de  $0.2s$ , así como el parámetro de relajación  $\tau = 0.5s$ .

Con el fin de entender el comportamiento de cada una de las fuerzas de repulsión a muros y a personas, hemos realizado la simulación por separado de cada una de ellas.

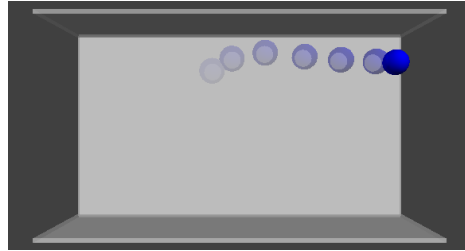


Figura 2.11: Ilustración de la fuerza de repulsión cuando un peatón se acerca a un muro.

Comenzamos con la simulación de la fuerza de repulsión a muros. Para ello se ha utilizado la fuerza de movimiento del peatón (ecuación (??)), la cual motiva a éste a moverse en la dirección  $\mathbf{e}_d$  junto con la fuerza de repulsión a muros, ecuación (??).

En la figura ?? observamos un peatón que se dirige a uno de los muros del pasillo. Cuando se aproxima lo suficiente al muro, el peatón siente una fuerza de repulsión de tal manera que se aleja de este. Una vez que esta fuerza pierde intensidad, el peatón continúa caminando en su dirección deseada.

Para simular la fuerza de repulsión entre peatones, tomamos solamente dos de ellos caminando uno en dirección del otro, como se muestra en la figura ?. Utilizamos la ecuación (??) con especificación circular. Con parámetros  $A_\alpha = 3.0 \frac{m^2}{s^2}$ , y  $B_\alpha = 0.5$  m.

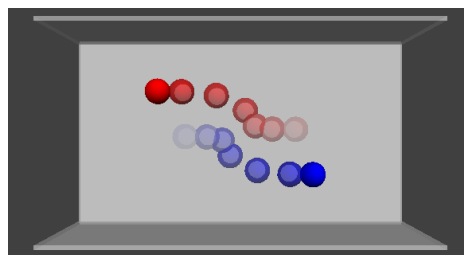


Figura 2.12: Ilustración de la repulsión de un peatón a otro.

Las medidas del pasillo son 18 m de largo y 10 m ancho, las esferas tienen un radio constante de 0.7 m. La distancia inicial entre ellos es de 4 m. Cuando están suficientemente cerca, a una distancia determinada por  $B_\alpha$ , ambos comienzan a sentir la fuerza de repulsión que un peatón ejerce sobre el otro. En el caso que ambos se encuentren completamente de frente (al mismo nivel), se quedan inmóviles hasta que otra fuerza perturbe este estado. Si uno de ellos lleva una velocidad mayor que el otro, el primero empujará al otro en dirección contraria.

### 2.4.1. Comparación entre especificaciones

Consideramos un pasillo de 50 m de largo y 10 m de ancho. Con un flujo unidireccional, donde los peatones caminan en un mismo sentido de izquierda a derecha. Para estas simulaciones utilizamos los

	$A_\alpha$	$B_\alpha$	$\lambda$
Circular	2.1	0.3	0.85
Elíptica I	6	0.5	1
Elíptica II	4.3	1.07	1

Cuadro 2.1: Parámetros utilizados para las diferentes especificaciones.

valores de los parámetros descritos en el cuadro ???. El radio de los peatones es de un tercio de su velocidad actual. De las simulaciones de la figura ?? podemos observar que tanto la especificación circular como la elíptica II tienen un mejor comportamiento en comparación a la elíptica I. Las velocidades en la especificación elíptica I en ocasiones llegan a dispararse lo que provoca un movimiento irregular y mayores choques entre los peatones. Contrario a las otras especificaciones; en la circular las velocidades no se disparan.

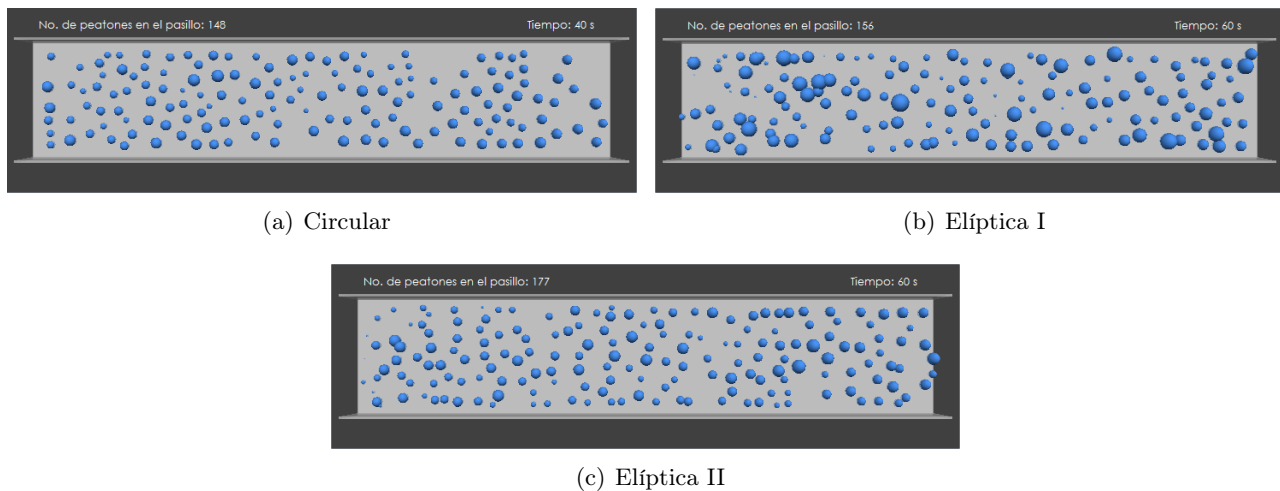


Figura 2.13: Comparación entre especificaciones para un flujo unidireccional. El radio de los peatones es del tercio de la velocidad. (a) Muestra una estabilidad en las velocidades a diferencia de (b) e incluso mejor que (c).

En un flujo bidireccional, la especificación elíptica I tiene un menor desempeño en comparación con las otras dos, la auto-organización entre los peatones no se presenta y los choques entre ellos son más frecuentes. Observamos por medio de los radios, que las velocidades de los peatones son muy diferentes. La especificación circular y elíptica II ofrece una mejor auto-organización, incluso la simulación de la figura ?? se aprecia que las líneas dinámicas, en la especificación circular, tienen a formar un flujo óptimo, formando dos corrientes de flujo.

Cabe mencionar que los valores de los parámetros del cuadro ??? para la circular y elíptica II los obtuvimos de [?, ?]. Sin embargo, los propuestos para la elíptica I no han tenido un buen desempeño, incluso hemos probado otros valores pero los resultados no han sido satisfactorios, los peatones chocan, se traslapan y no se observa una formación de líneas dinámicas. La ventaja de la especificación circular y la elíptica II es que son simétricas lo que aumenta la rapidez en los cálculos de las fuerzas. Y puesto que la especificación circular es mucho más sencilla y ofrece resultados parecidos a los de la elíptica II. En adelante utilizaremos la especificación circular.

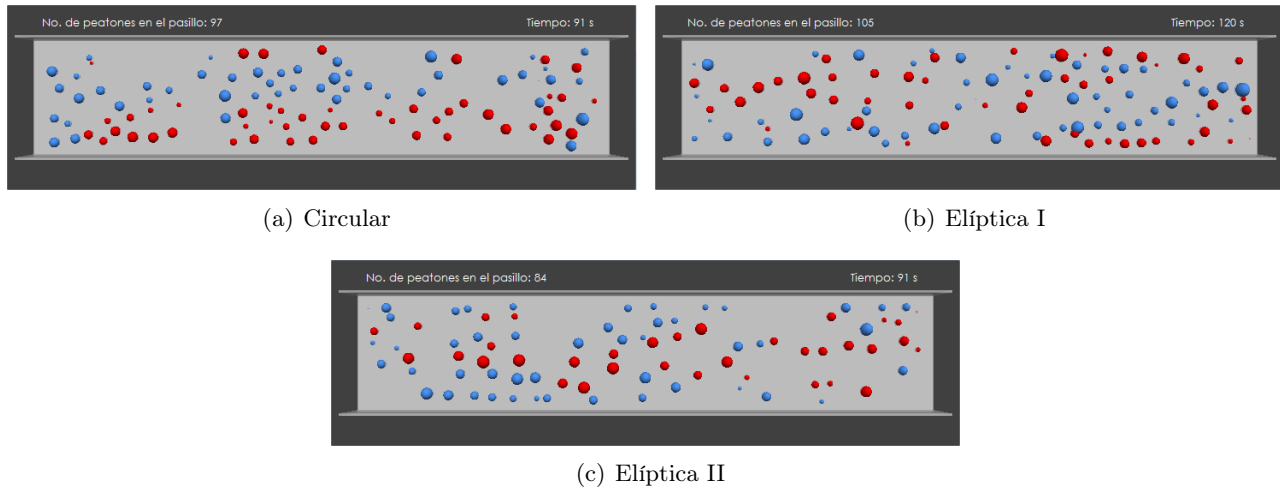


Figura 2.14: Comparación entre especificaciones para un flujo bidireccional. (a) Muestra un flujo ordenado a diferencia de (b) y (c).

## 2.4.2. Pasillo

### Flujo unidireccional

En la figura ?? se aprecian los resultados de la simulación ya con la especificación circular. Los peatones se distribuyen casi homogéneamente por todo el pasillo, manteniendo casi siempre la misma distancia entre ellos. Para el caso de un flujo heterogéneo, donde las personas que llevan diferentes velocidades, observamos que se forman grupos de personas, ver figura ??. Los peatones que llevan mayor velocidad desvían su camino con tal de mantener su velocidad deseada.

Si variamos el parámetro  $\tau < 1$ , con frecuencia los peatones se rebasan o empujan a otros, que llevan menor velocidad, con tal de llegar a su destino. Si  $\tau > 2$  entonces el comportamiento se vuelve irregular, oscilando de un lado a otro, ya que mantenerse alejado de los otros peatones es mucho mas importante que llegar a su destino.

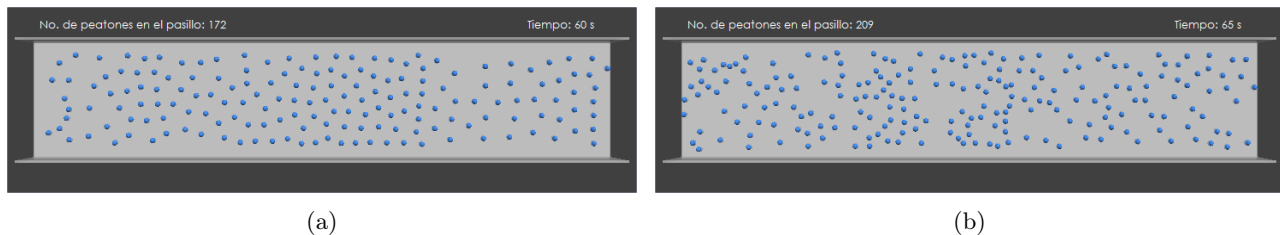


Figura 2.15: Ilustración del flujo uni-direccional. (a) Flujo homogéneo, los peatones mantienen una distancia entre ellos, mientras que (b) flujo heterogéneo, se forman cúmulos debido a la diferencia de velocidades.

Por otro lado, al variar el rango de repulsión  $B_\alpha$ , los peatones aumentan o disminuyen la distancia entre ellos. La intensidad de interacción  $A_\alpha$ , determina la magnitud de la fuerza de repulsión que siente el peatón  $\alpha$ . Por ejemplo, para  $B_\alpha < 0.2$  y  $A_\alpha > 10$  las personas se acercaran mucho hasta un punto en que reaccionan y se alejan de manera abrupta.

## Flujo bidireccional

De la figura ?? vemos dos tipos de peatones “rojos” que caminan de izquierda a derecha y “azules” de derecha a izquierda caminado dentro de un pasillo de 10 m de ancho y 50 m de largo.

En esta simulación se observa la formación de líneas dinámicas, como ya habíamos mencionado anteriormente, consiste en la formación de peatones en una hilera donde llevan una misma dirección de caminata uniforme en el mismo sentido [?], es decir un estado de mínima interacción [?] o una propiedad intrínseca de auto-organización [?]. El número de líneas formadas depende del ancho de pasillo o de la densidad de peatones. Es importante mencionar que en la simulación se utilizan objetivos fijos, es decir, cada peatón tiene como meta llegar al punto medio del extremo opuesto del corredor.

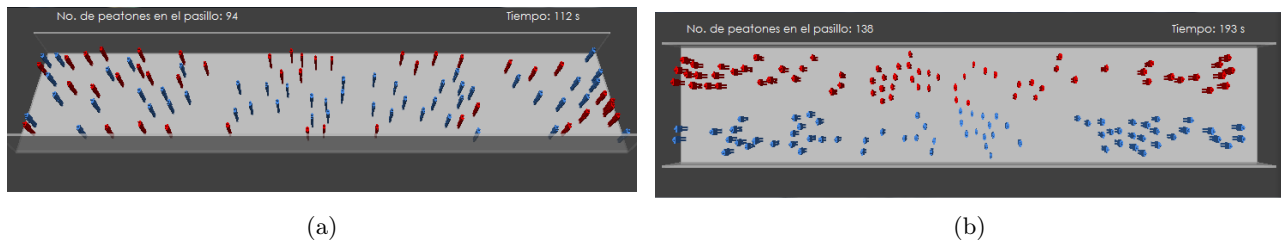


Figura 2.16: Formación de líneas dinámicas en flujo bi-direccional. (a) Líneas dinámicas, (b) agrupación de líneas dinámicas en dos corrientes de flujo.

Mientras la densidad de personas aumenta, las oportunidades de rebasar se reducen hasta el punto en que deja de existir, entonces el flujo heterogéneo se convierte en un flujo peatonal homogéneo. También podemos ver en la figura ?? como después de un tiempo, las líneas dinámicas se juntan dividiendo el pasillo en dos corrientes de flujo tal como pasa en el tráfico peatonal.

### 2.4.3. Cuellos de botella

#### Rotonda

Esta simulación se hace sobre un pasillo de tamaño 10 m ancho por 40 m de largo, con un obstáculo (circular) en medio del pasillo de 5 m de radio. La intensidad de interacción entre peatones es  $A_\alpha = 2.5$ , con un rango  $B_\alpha = 0.5$ , y  $\lambda = 0.8$ .

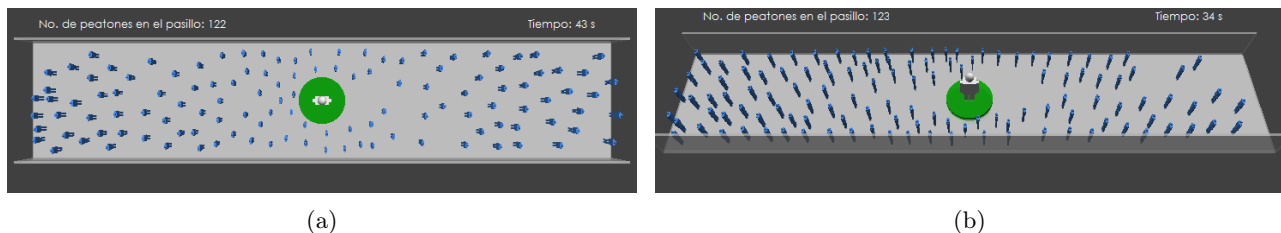


Figura 2.17: Pasillo con obstáculo. (a) Los peatones desvían su dirección después de acercarse al obstáculo. (b) Se incorpora otra fuerza de repulsión a muros desviando su movimiento mucho antes de llegar al obstáculo.

En la figura ?? observamos cómo los peatones desvían la dirección mucho antes de chocar con el obstáculo. Esto se logró introduciendo un potencial más a la fuerza de repulsión (a muros) que depende

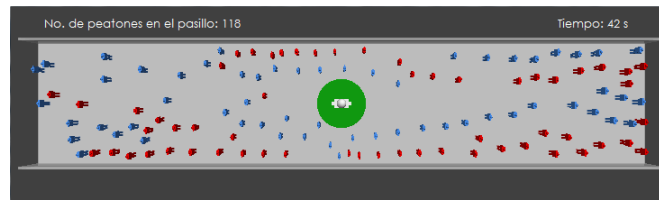


Figura 2.18: Flujo bidireccional de peatones sobre pasillo con obstáculo. Se presenta la formación de líneas dinámicas al rededor del obstáculo.

del ángulo entre la velocidad del peatón y la distancia al muro. De esta manera logran evadir el obstáculo sin tener que colocar una meta intermedia a los lados del obstáculo. Solo se tomaron los puntos medios en los extremo del corredor como objetivos. Si no se considerara esta fuerza el flujo chocaría contra el obstáculo distribuyéndose por los lados hasta lograr pasarlo ???. En un flujo bidireccional podemos notar la presencia de líneas dinámicas, ver figura ???.

### Puerta

El cruce de una puerta es un tipo de cuello de botella. El pasillo tiene 25 m de largo con un ancho de 10 m, la puerta tiene 2 m de ancho. Para este escenario, la intensidad de fuerza entre peatones es de  $A_\alpha = 3$  con rango de  $B_\alpha = 0.4$ , y  $\lambda = 0.85$ . En esta simulación se muestran un conjunto de personas atravesando una puerta. La distancia más cercana a la puerta determina la forma semicircular de las personas esperando su turno para pasar. Las personas al pasar muestran cambios oscilatorios en su movimiento. Para el caso bidireccional, aparece un intercambio de flujos que se da de la siguiente forma: si un peatón logra pasar a través la puerta, las personas detrás del él lo siguen. Esto ocurre porque las presiones de los dos lados son diferentes. El lado de menor presión permite el paso de personas del lado con mayor presión. Hasta llegar al punto en que las presiones se intercambian provocando con esto que el paso de las personas sea oscilatorio, ver figura ???. En este caso se utilizan tres metas: los puntos medios de los extremos y de la puerta.

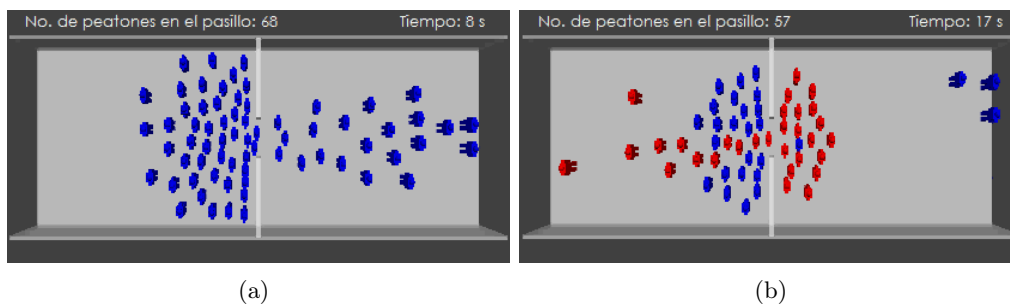


Figura 2.19: Cuello de botella tipo puerta, (a) y (b) muestran oscilaciones en el movimiento de los peatones al cruzar la puerta.

### Dos puertas

El pasillo mide 50 m de largo y 10 m ancho. Cada puerta tiene 2 m de ancho. Las simulaciones revelan que dos puertas adyacentemente separadas por una misma pared (figura ??), son más eficientes que sólo una puerta con el doble de ancho. La auto-organización se muestra cuando cada una de las puertas tiene

una sola dirección (sentido). La razón es porque las personas que pasan facilitan el camino para sus sucesores, similar a la formación de líneas dinámicas [?].

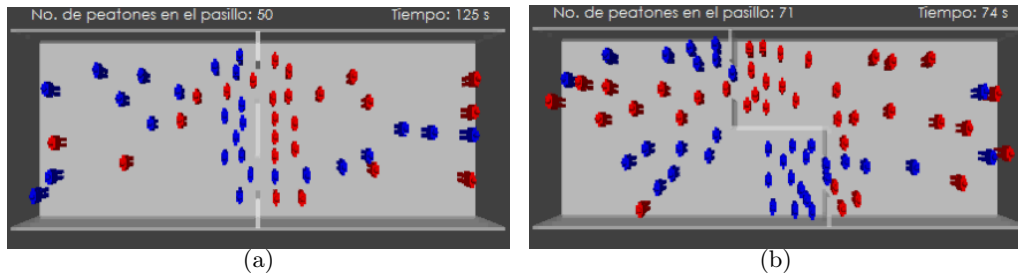


Figura 2.20: Flujo bidireccional con dos puertas. (a) Puertas sobre el mismo muro (en línea) y (b) en muros diferentes (a desnivel). Observamos que peatones se distribuyen mejor cuando las puertas se encuentra a desnivel.

Ahora, si en lugar de poner las puertas sobre la misma pared, las colocamos en desnivel. La simulación, en la figura ??, presenta una mejor auto-organización. Podemos ver que la mayoría de los peatones azules y rojos pasan por la puerta más lejana, formando dos corrientes.

### Reducción de pasillo

La ampliación y reducción de pasillos es la forma más común de cuello de botella. Estos escenarios aparecen con mayor frecuencia en lugares públicos. La estructura en esta simulación es un caso de un doble cuello de botella con un flujo uni y bidireccional. Los peatones tienen como meta llegar a entrada del pasillo angosto, salir y dirigirse a la parte media de la salida.

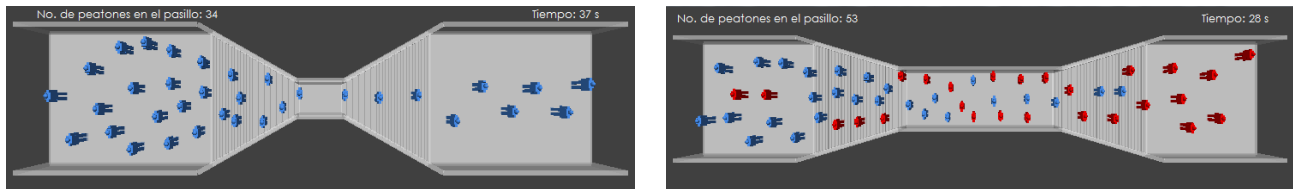


Figura 2.21: Reducción y ampliación de pasillo. La formación de líneas dinámicas y los cambios oscilatorios en el movimiento se hacen presentes en este tipo de estructuras.

La figura ?? muestra la formación de líneas dinámicas dentro de pasillo más angosto y un movimiento oscilatorio saliendo de este. El constante contacto que se genera entre peatones de dirección opuesta ocasiona una inestabilidad y un pequeño comportamiento oscilatorio. Si el flujo no es capaz de regularse entonces la presión se acumula y es capaz de generar situaciones de bloqueo que conllevaría al daño de muchas personas en situaciones de emergencia.

### Esquina

El pasillo tiene un ancho de 10 m. Para el caso sin esquina, el corte es de 4 m a un ángulo de  $45^\circ$ . Los parámetros son:  $\lambda = 1$ , con intensidad de interacción entre peatones  $A_\alpha = 2.5N$  y rango  $B_\alpha = 0.5$  m, ver figura ?. Recordemos que los peatones se mueven utilizando el menor esfuerzo al caminar, por lo tanto en este caso se dirigen al punto más cercano (a esquina) para dar vuelta. Por lo tanto, la menor manera de incrementar el flujo es redondear las esquinas, con esto se disminuye la distancia para llegar a la salida.

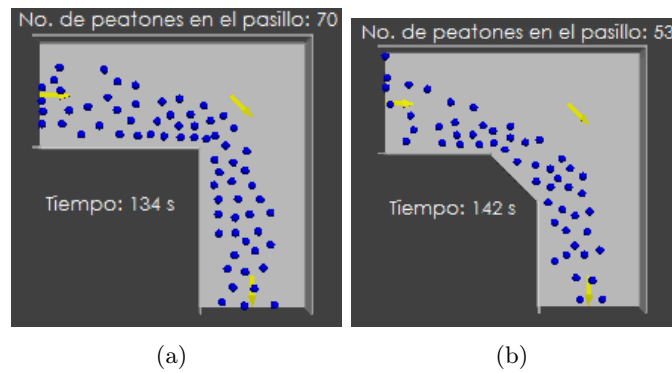


Figura 2.22: Flujo peatonal en una esquina: (a) esquina en punta (b) esquina recortada. Redondear las esquinas ofrece mayores rendimientos en cuanto al tiempo de recorrido que las esquinas punteadas

Para esta simulación se considera que la meta intermedia del peatón es un punto situado en medio del pasillo de salida, por lo tanto es importante definir bien las metas a fin de que el peatón de vuelta sin tropezar con paredes y lograr una simulación mucho más realista.

#### 2.4.4. Cruce de pasillos

##### Cruce ortogonal y diagonal

Se contrastan dos tipo de cruces de pasillos: cruce perpendicular y diagonal con inclinación de  $45^\circ$ , ver figura ???. Ambos pasillos tienen 40 m de largo y 10 m de ancho. Los peatones azules van de izquierda a derecha y los blancos de abajo hacia arriba. La simulación utiliza los siguientes parámetros: intensidad de fuerza entre peatones de  $A_\alpha = 3N$  con rango  $B_\alpha = 0.3$  m, y  $\lambda = 0.85$ .

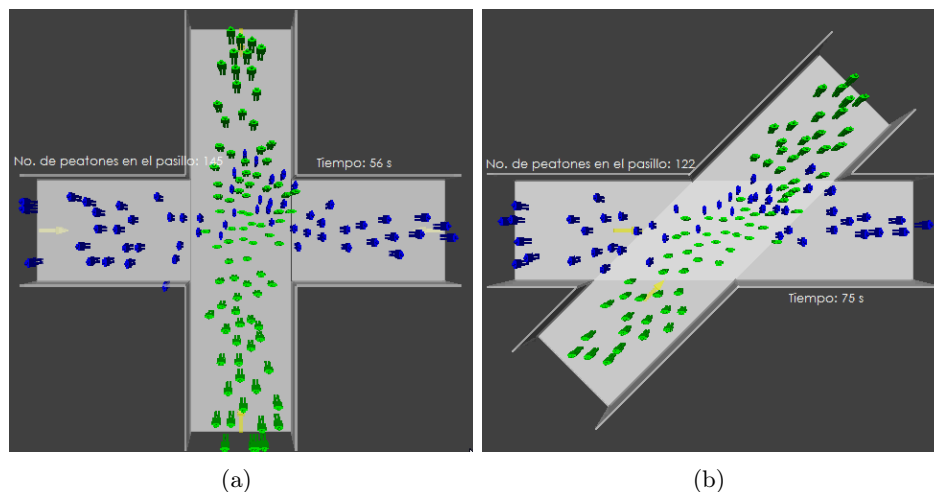


Figura 2.23: Cruce de pasillos con solo dos direcciones. (a) La corriente con mayor fuerza arrastra a los peatones que cruzan el pasillo provocando congestiones. (b) Las personas tiene mayor facilidad de incorporarse a su camino.

En las intersecciones las personas se enfrentan con diferentes patrones alternativos de movimiento, los cuales son inestables y tienen un corto periodo de vida. Se presenta la formación de grupos de personas



esperando continuar con su camino. La mayor parte del tiempo los grupos tienden a una estructura homogénea donde las velocidades individuales son parecidas entre ellos haciendo que las personas no rebasen aquellas que llevan misma dirección.

Los patrones homogéneos que emergen tienen forma en estos pasillos son pequeñas líneas diagonales. Las estructuras que se presentan son similares a las producidas en los flujos uni y bidireccionales.

En cruce ortogonal de pasillos, figura ??, las personas que tratan de cruzar son arrastrados por corriente de individuos que atraviesa el pasillo provocando que las personas busquen incorporarse a su camino propiciando congestionamientos. Para cruce diagonal de pasillos, figura ??, las personas arrastran por corrientes que cruzan el pasillo tiene mayor facilidad de incorporarse a su camino ya que a veces estas corrientes impulsan a los peatones a cruzar con mayor rapidez.

### 2.4.5. Cruce con y sin obstáculo

Por ejemplo, la fase durante la cual la intersección se cruza en dirección “vertical” u “horizontal” alterna con la fase temporal de un movimiento rotatorio (glorieta). Esta auto-organización en tráfico de rotonda produce en promedio un movimiento de personas más eficiente, pero en este caso es temporal. La eficiencia del tráfico peatonal se puede incrementar si ponemos un obstáculo en el centro de la intersección, ya que esto favorece un movimiento de rotonda, ver figura ??.

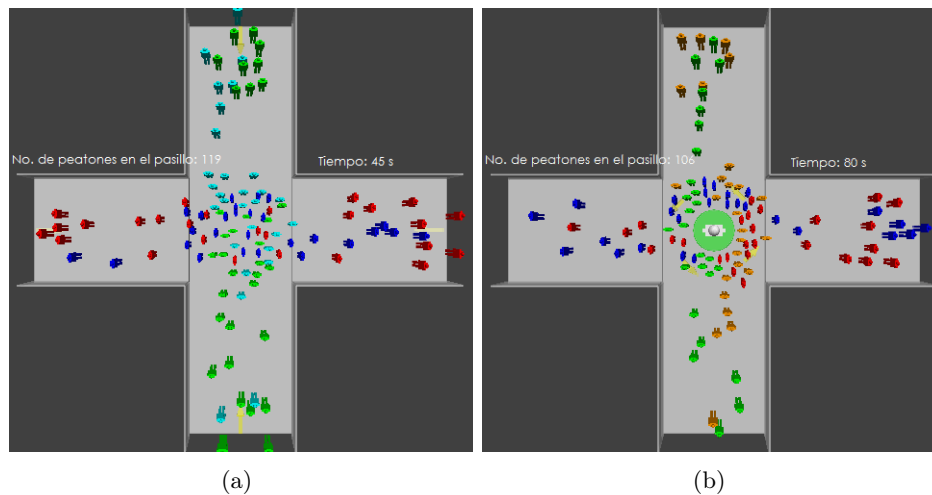


Figura 2.24: Cruce de pasillos con cuatro corrientes de flujo. (a) congestión de personas, (b) auto-organización mediante una rotonda.

## 2.5. Diagrama fundamental

Utilizando la ecuación (??), del modelo de fuerza social modificado, logramos reproducir el diagrama fundamental de Weidmann. Para ello consideramos los parámetros propuestos por Parisi en [?],  $A_\alpha = A_w = 2000$  N,  $B_\alpha = B_w = 0.08$  m,  $\Delta t = 0.01$  el paso de tiempo,  $\tau = 0.5$  s el tiempo de aceleración. Los parámetros  $k_n = 1.2 \times 10^5$  kg s<sup>2</sup> y  $k_t = 2.4 \times 10^5$  kg m<sup>-1</sup>s<sup>-1</sup>, además tomamos  $R_F = 0.85$ . La masa  $m$ , el ancho de los hombros  $d$  y la velocidad deseada  $v_\alpha^d$  son distribuidas uniformemente con los siguientes rangos:

$$\begin{aligned}
 m &\in [70 \text{ kg}, 90 \text{ kg}] \\
 d &\in [0.5 \text{ m}, 0.58 \text{ m}] \\
 v_{\alpha}^d &\in [0.9 \text{ m/s}, 1.5 \text{ m/s}]
 \end{aligned}$$

En la figura ?? se mide al flujo y la densidad en un área de  $21 \text{ m}^2$ . Se tomaron 500 segundos, de los cuales 60 de ellos se utilizaron para estabilizar la densidad en el área de estudio. En la figura ?? se muestra una comparación de las simulaciones modelo de fuerza social modificado contra Weidmann, se observa que el primer modelo reproduce cualitativamente el diagrama fundamental.

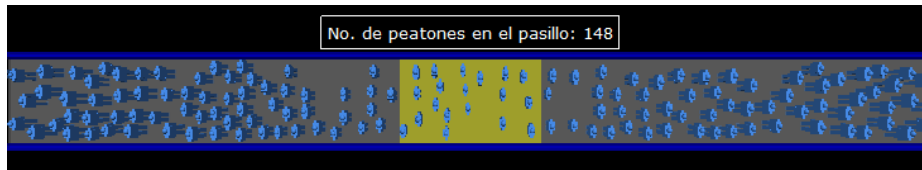


Figura 2.25: Simulación para la medición de flujo-densidad con flujo unidireccional.

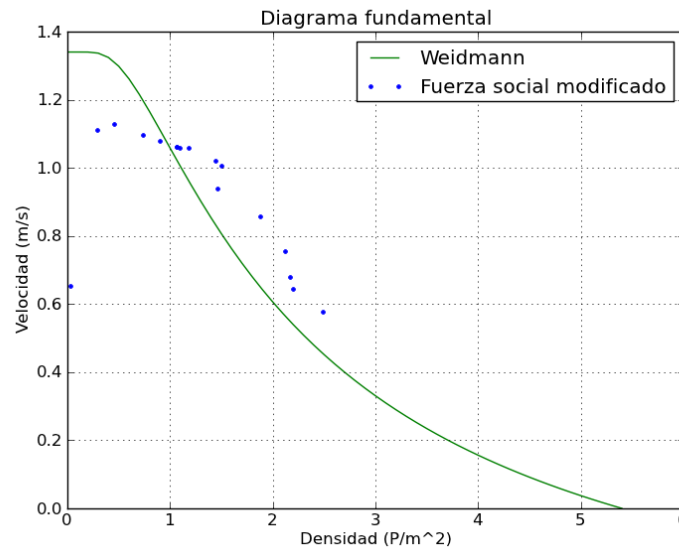


Figura 2.26: Comparación entre el diagrama fundamental de Weidmann y el modelo de fuerza social modificado.

Otras modificaciones al modelo de fuerza social son: [?, ?, ?, ?]

### 2.5.1. Estudio de carriles

El metro de la ciudad de México con el fin de mejorar el flujo de personas dentro de sus instalaciones ha implementado varios tipo de estructuras dentro de sus pasillos. Algunos de ellos consisten en la división de carriles con barandales o solamente usando señalizaciones, en otros casos han puesto barandales en zigzag. Sin embargo, ¿qué podemos decir de este tipo de estructuras?, ¿cuál es la mejor?



Figura 2.27: Metro de la ciudad de México. Estación de correspondencia Salto del agua.

Hicimos un pequeño estudio de un pasillo con dos y tres carriles utilizando solo líneas divisorias y flechas de señalización de flujo. El pasillo tiene 50 metros de largo y 10 de ancho. La medición de densidad y velocidades se hicieron sobre un área de  $30 \text{ m}^2$ . La simulaciones se ejecutaron por 500 s, con 60 s de ellos para estabilizar el flujo tal como se hizo anteriormente.

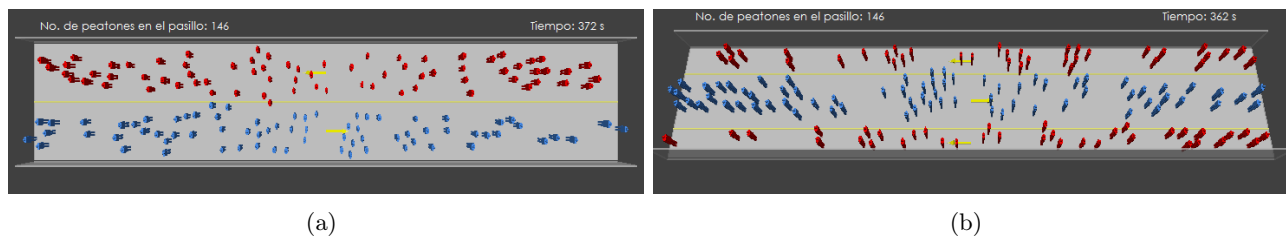


Figura 2.28: Simulación de flujo bidireccional para el caso de (a) dos carriles y (b) tres carriles.

Los peatones en azul van de izquierda a derecha y los peatones rojos en sentido contrario. El número de peatones azules y rojos es el mismo. Las condiciones de frontera son de tipo periódicas. Para inducir a los peatones a mantenerse dentro de su carril aplicamos una fuerza de repulsión en la línea de separación. La intensidad de esta fuerza es mucho menor que la de las paredes, con el fin de dejar invadir el otro carril cuando sea necesario. Los valores de la fuerza de repulsión de la línea son:  $A_l = 5N$  y  $B_l = 0.2m$ . Así también valores permiten que peatón pueda regresar a su carril cuando sea necesario.

Del diagrama fundamental ?? podemos observar que el pasillo con dos carriles tiene mayor flujo que el que tiene tres carriles. En la simulación notamos que dentro del pasillo con tres carriles había mayor inestabilidad ya que los peatones con direcciones opuestas con frecuencia invadían los carriles o se empujaban provocando pequeñas congestiones en sus carriles. A diferencia del pasillo con dos carriles la fricción entre los peatones con direcciones opuestas era solo era la frontera que dividía al pasillo haciendo que el flujo en sus carriles sea continuo y ordenado.

Para densidades mayores a las de flujo libre, surgen regiones donde los peatones bloquean su paso, comenzando pequeñas turbulencias sobre la línea de división de carril. Estos bloqueos con frecuencia surgen en pasillos con tres carriles, figura ??.

En ocasiones el tipo de estructura favorece la formación de tres carriles, por lo tanto aplicar una política forzando la formación de dos carriles ocasionaría mayores congestionamientos. Solo en estas circunstancias el flujo bidireccional dividido en tres carriles es la mejor opción, ver figura ??.

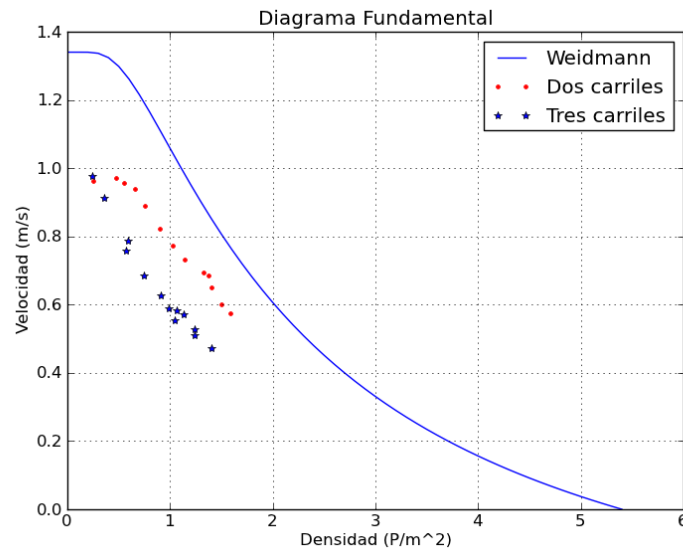


Figura 2.29: Diagrama de flujo para el análisis de pasillo con dos y tres carriles.

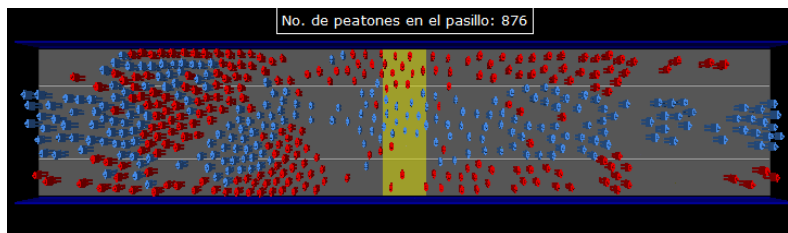


Figura 2.30: Congestión de peatones en pasillo con 3 carriles.



Figura 2.31: Estructura que provoca el formación de tres carriles.

## 2.6. Conclusiones

Se ha reproducido en diferentes simulaciones los diversos patrones colectivos como son:

- La formación de líneas dinámicas de peatones en pasillos con flujo bidireccional.
- Los cambios oscilatorios en la dirección de caminata que se producen en los cuellos de botella como

puertas o reducciones y ampliaciones de pasillos.

Hemos visto que estos patrones colectivos de movimiento se pueden interpretar como un fenómeno de auto-organización. Lo cual se puede aprovechar para que el tráfico peatonal sea más eficiente, por ejemplo incluir una fuente en el cruce de pasillos, redondear las esquinas, etc.

También hemos verificado que el modelo de fuerza social modificado abarca un espectro más amplio en condiciones normales y de pánico.

Por otro lado, hemos visto la importancia de estimar los parámetros  $A_\alpha$ ,  $B_\alpha$ ,  $\tau$ ,  $\lambda$  del modelo de fuerza social para cada uno de los escenarios. Esto nos ayudará significativamente a que las simulaciones sean más realistas.

Queda pendiente el estudio de cómo obtener los valores de estos parámetros para calibrar nuestro modelo en problemas reales.



# CAPÍTULO 3

## Modelos de autómatas celulares

### 3.1. Introducción

Lo primero que pensamos cuando escuchamos la palabra *autómata* es en una máquina con inteligencia artificial moviéndose o controlando alguna función de un mecanismo en específico. Para darnos una idea sobre los *autómatas celulares* (AC) pondremos como ejemplo las marquesinas: los anuncios de texto corredizos (rótulos electrónicos) son autómatas celulares ya que utilizan pequeños focos llamados leds (celdas) que tienen como finalidad mostrar texto o imágenes. Los focos encienden y apagan cada uno en un tiempo determinado, de tal manera que el mensaje sea claro para el que lo vea.



Figura 3.1: Rotulo eléctrico que utiliza leds (focos) para mostrar un mensaje.

La idea de autómata celular fue desarrollada por John von Neumann y Stanislaw Ulam en la década de los cuarenta, como una estructura matemática, la cual permitía investigar la organización existente en los procesos biológicos de la auto-reproducción celular, de tal manera, que se veía como un conjunto de células que crecían, se reproducían y morían a medida que pasaba el tiempo (en pasos discretos) [?].

Sin embargo, fue hasta la década de los sesenta cuando el matemático John Horton Conway desarrolló una aplicación práctica de los autómatas celulares con el famoso juego de la vida [?] y quizás el autómata celular más conocido, debido a la evolución de ciertos patrones, figura [?]. El tablero del juego es una malla formada por cuadros (células) que se extiende en todas direcciones. Cada célula tiene ocho células vecinas, próximas a ella. Las células tienen dos estados: *vivas* o *muertas*. El estado de la malla evoluciona en unidades de tiempo discretas. Las leyes que gobiernan este juego son:

1. **Reproducción y nacimiento:** una célula muerta (blanca) tiene exactamente tres células vecinas vivas (negras), se reproducen y dan vida a la primera para el siguiente paso de tiempo.
2. **Sobrevivencia:** una célula viva que tiene a lo más dos ó tres vecinas vivas, permanece viva.
3. **Sobrepoblación:** una célula viva que tiene más de tres vecinas vivas, muere.
4. **Aislamiento:** una célula viva con menos de dos vecinos vivos muere.

Existen muchas variaciones de estas reglas las cuales producen diferentes patrones, los más conocidos son: patrones estáticos (el bloque y el barco), patrones recurrentes (el parpadeador y el sapo) y patrones que se trasladan por el tablero (planeador y la nave espacial), ver figura??

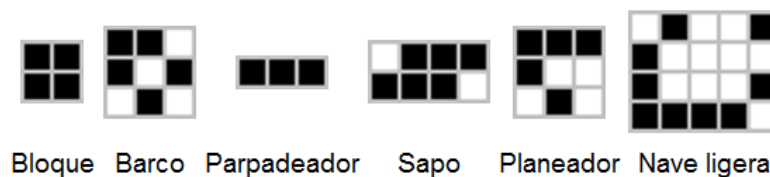


Figura 3.2: Patrones relacionados con el *juego de la vida*.

Entonces, un autómata celular está compuesto por una cuadrícula donde cada celda evoluciona y cambia de estado conforme pasa el tiempo. Además incluye una configuración inicial ( $t = 0$ ) y condiciones de frontera. La evolución del sistema dependerá de esta configuración inicial.

Los AC se aplican a simulación de evacuación de barcos y salas de cine, estudio de mercados y efectos de la publicidad, diversión, arte, investigación, simulación de procesos físicos, químicos y biológicos, etcétera. También hay teorías que cohesionan la mecánica newtoniana con la relativista y con la cuántica utilizando AC <sup>1</sup>. En este trabajo utilizaremos diferentes modelos de autómatas celulares para simular el tráfico peatonal bidireccional sobre un pasillo. En la sección 3.3 se presentan las ideas principales de varios modelos de autómatas celulares. Sin embargo, no todos los artículos presentan los algoritmos de los modelos. Hemos elegido el modelo de Klüpfel y el modelo de dos procesos para implementarlos por ser más claros en la descripción de sus reglas, los cuales se presentarán en los apartados 3.4 y 3.5.

## 3.2. Definición de autómata celular

Los autómatas celulares constituyen una clase particular de sistemas dinámicos discretos<sup>2</sup> debido a que no solamente el tiempo es discreto sino también el espacio. Un autómata celular se puede definir formalmente como una tupla  $(\mathcal{L}, \mathcal{S}, \mathcal{N}, f)$ , es decir un conjunto ordenado de objetos caracterizados por:

- $\mathcal{L}$  rejilla o cuadrícula (malla) de enteros. Cada cuadro es una celda.
- $\mathcal{S}$  conjunto finito de estados que puede tomar cada celda.

<sup>1</sup>Gerard't Hooft ganador del Premio Nobel de Física, desarrollo un modelo del universo en forma de autómata celular que permite que el entrelazamiento cuántico sea determinista. Hooft dice que: “El universo es un autómata celular en el que la realidad es simplemente la interpretación de una gigantesca y fantástica máquina compleja de computación”

<sup>2</sup>Un sistema dinámico es un sistema complejo que presenta el cambio o evolución de su estado a lo largo del tiempo; el comportamiento en dicho estado se puede caracterizar determinando los límites del sistema, los elementos y sus relaciones; se dice discreto si el tiempo se mide en pequeños lapsos



- $\mathcal{N}$  es la vecindad correspondiente a cada celda.
- $f$  función de transición (actualización) de estados, la cual se aplica a cada una de las celdas. Tomando como argumentos el estado de la celda y de sus vecinos en el tiempo  $t$ . Retorna el nuevo estado de la celda para el próximo paso de tiempo.

La función de configuración  $C_t : \mathcal{L} \rightarrow \mathcal{S}$  es una relación que asocia a cada celda de la malla un estado. La función de actualización (o transición)  $f$  cambia la configuración  $C_t$  a una nueva configuración  $C_{t+1}$ :

$$C_{t+1}(r) = f(\{C_t(i) | i \in \mathcal{N}(r)\}) \quad (3.1)$$

donde  $\mathcal{N}(r)$  es el conjunto de vecinos de la celda  $r$ ,  $\mathcal{N}(r) = \{i \in \mathcal{L} | r - i \in \mathcal{N}\}$ . La ecuación (??) describe la influencia local de la vecindad  $\mathcal{N}(r)$  sobre el conjunto de estados de la celda  $r$  en la evolución del tiempo, es decir, no hay interacciones de largo alcance.

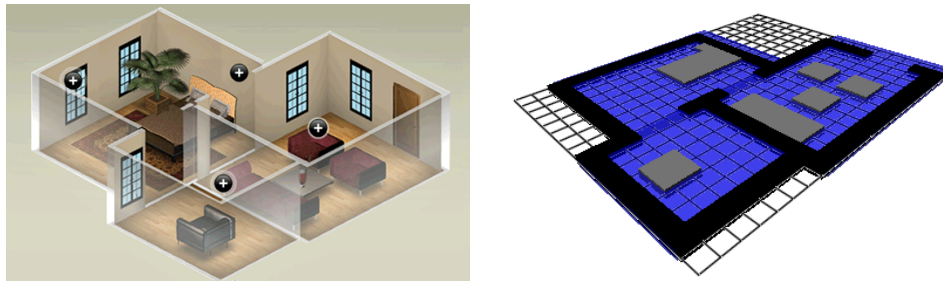


Figura 3.3: Transformación de una estructura en una cuadrícula de celdas. Las paredes u otros obstáculos fijos son celdas con un estado fijo.

Además los autómatas celulares están limitados por condiciones de frontera las cuales se refieren a qué ocurre en las celdas que tocan los bordes del espacio de estudio, ver figura ???. Estas condiciones de frontera se clasifican en:

- **Periódicas:** se considera que la cuadrícula se cierra tocando sus extremos; en 2D se forma un toro, ver figura ???. Si una celda está en la frontera, interaccionará con sus vecinos inmediatos y con las celdas que están en los extremos opuestos del autómatas, como si fueran círculos.
- **Abiertas:** se considera que las células fuera de la cuadrícula toman un valor fijo.
- **Reflectoras:** las celdas que están fuera del extremo de la cuadrícula toman los valores de aquellas que están dentro como si se tratara de un espejo que refleja los valores de las celdas.
- **Sin fronteras:** se puede asumir que cada vez que las células quedan fuera de la malla, la cuadrícula se expande para dar cabida a estas interacciones. Obviamente, para cada tipo de frontera existe un límite impuesto por la memoria disponible.

Por su parte, se distinguen dos tipos básicos de vecindades para cada celda:

1. *Vecindad de von Neumann.* Es un conjunto conformado por cuatro celdas, la de arriba, abajo, derecha e izquierda, alrededor de la celda  $r$ , ver figura ???. En este caso el radio es 1, ya que solamente se consideran los vecinos más cercanos. Esta vecindad fue utilizada por Von Neumann en su primer ejemplo en [?].

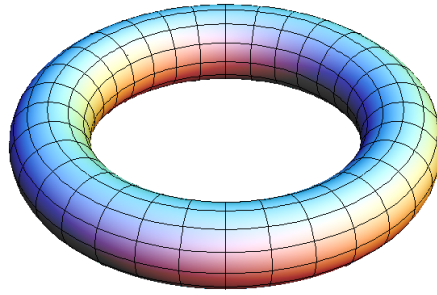


Figura 3.4: Condiciones de fronteras periódicas por los cuatro extremos de una cuadrícula.

2. *Vecindad de Moore*. Es una extensión de la vecindad de Von Neumann, la cual contiene celdas en la diagonal, ver figura ???. Esta es la vecindad que utiliza el juego de la vida.

De estas se derivan diferentes vecindades, extendiendo su radio o considerando sólo una parte de la vecindad. La figura ??? muestra una vecindad de Moore de radio 2.

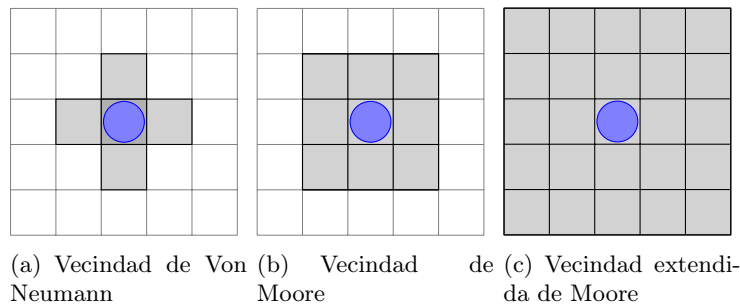


Figura 3.5: Tipos de vecindades

Cabe mencionar que actualmente se pueden formar un teselado del espacio de estudio con diferentes figuras, por ejemplo se generan celdas de forma hexagonal [?].

### 3.3. Flujo peatonal con autómatas celulares

En recientes décadas, el estudio de la dinámica peatonal ha tenido gran desarrollo, principalmente mediante modelos de autómatas celulares, debido a su eficiencia para modelar sistemas complejos. La rapidez de simulación y el número de personas con las que se pueden trabajar es mucho mayor que con otro tipo de modelos. En este apartado presentaremos algunos de los modelos con este tipo de enfoque.

#### 3.3.1. Modelo de parámetros dinámicos

Este modelo es introducido por Hao Yue et al. en [?]. El autor considera dos tipos de peatones: aquellos que se mueven hacia arriba y otros en dirección contraria. Cada celda puede ser ocupada solamente por un peatón. Los peatones tienen una velocidad máxima  $V_{max} = 1$ , es decir avanzan una celda por cada paso de tiempo. Las fronteras de arriba y abajo de la cuadrícula son de tipo periódicas e izquierda y derecha son cerradas. De esta manera se mantiene la misma densidad de personas dentro del sistema. Se toma la

vecindad de Moore con radio 1 como los posibles movimientos del peatón, ver la figura ???. Los peatones eligen esperar o moverse de acuerdo a una matriz de transición de pago  $P = p_{(ij)}$ , como se muestra en la figura ???. Cada elemento de la matriz corresponde al pago del movimiento hacia una de las nueve direcciones posibles.

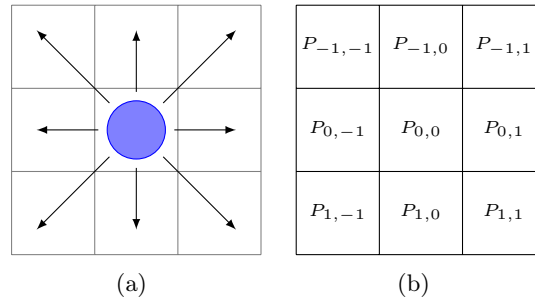


Figura 3.6: Los movimiento permitidos mediante la matriz de transición de pago.

Considerando el campo de movimientos, para cada celda se calcula cuatro parámetros dinámicos: parámetro de dirección, de vacío, de avance y de categoría. Observamos que para cada parámetro se construye una matriz de tamaño  $3 \times 3$ .

### Parámetro de dirección

El movimiento del peatón se caracteriza por un destino y una dirección. Los valores de esta matriz  $D$  representan una ganancia cuando se aproxima a su destino, una pérdida cuando se aleja de él y un valor nulo en movimientos laterales. Si la dirección del peatón va de izquierda a derecha como en la figura ?? , entonces.

-0.7	0	0.7
-1	● →	1
-0.7	0	0.7

$$D_{ij} = \begin{cases} 1, & \text{celda a la derecha;} \\ 0.7, & \text{celdas superior e inferior derecha;} \\ 0, & \text{celdas de los lados;} \\ -0.7, & \text{celdas superior e inferior izquierda;} \\ -1, & \text{celda a la izquierda.} \end{cases} \quad (3.2)$$

Observemos que si se mueve de frente a alguna de las dos celdas diagonales gana  $\sqrt{2}/2 \approx 0.7$  unidades de pago. Por conveniencia se le asigna este valor, de otro modo daría lugar a un movimiento irregular.

### Parámetro de vacío

Este parámetro refleja la influencia de las celdas vacías u ocupadas sobre los peatones, ya que éstos sentirán un mayor rechazo si quieren caminar hacia una celda ocupada. Si el peatón decide no moverse, entonces el valor del parámetro de vacío es cero.

$$E_{ij} = \begin{cases} 1, & \text{celdas vacías;} \\ 0, & \text{celda central;} \\ -1, & \text{celdas ocupadas.} \end{cases} \quad (3.3)$$

### Parámetro de avance

Este parámetro refleja la atracción de los peatones hacia la celda objetivo, mediante el número de celdas vacías dentro de un campo de visión [?]. Este campo de visión consiste de  $3 \times 5$  celdas como muestra la figura ?? . El parámetro de avance se describe como

$$F_{ij} = S_1 - \frac{S_2}{nc} \quad (3.4)$$

donde  $S_1$  es el número de celdas vacías,  $S_2$  número de celdas ocupadas y  $nc$  número de celdas dentro del campo de visión. En caso de que el peatón se encuentre cerca de alguna frontera el número de celdas dentro del campo se reduce.

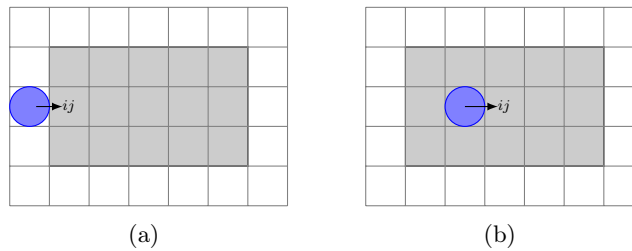


Figura 3.7: (a) Campo de visión para parámetro de avance. (b) campo para el parámetro de categoría, donde  $ij$  es la celda objetivo hacia donde el peatón desea avanzar.

### Parámetro de categoría

Este describe la proporción de peatones en la dirección deseada dentro del campo alrededor de la celda objetivo.

$$C_{ij} = S_1 - \frac{S_2}{nc} \quad (3.5)$$

donde  $S_1$  es el número de celdas vacías y peatones con misma dirección dentro del campo de visión,  $S_2$  número de peatones con diferente dirección y  $nc$  número de celdas dentro del campo de categoría, ver figura ??.

### Parámetro de lado derecho

Este parámetro fue introducido por Hao Yue et al. [?]. El parámetro refleja el hábito de las personas para caminar por el lado derecho.

$$R_{ij} = \begin{cases} r, & \text{celda objetivo de lado derecho del peatón;} \\ 0, & \text{celda de central;} \\ -r, & \text{celda objetivo de lado izquierdo del peatón.} \end{cases} \quad (3.6)$$

El rango de este parámetro se encuentra entre 0 y 0.3. Para valores mayores los peatones elegirán ciegamente el lado derecho.

Por lo tanto,  $P_{ij}$  determina la posible celda objetivo  $ij$  a ocupar, a partir del cálculo de los cinco parámetros dinámicos.

$$P_{ij} = D_{ij} + E_{ij} + F_{ij} + C_{ij} + R_{ij} \quad (3.7)$$

### Reglas de actualización

Las reglas de actualización son las siguientes:

1. En el modelo, los peatones sólo se pueden mover una celda en cada paso de tiempo en cualquiera de las 9 posibles direcciones, figura ??.
2. En cada paso de tiempo, para cada peatón se calcula el valor  $P_{ij}$  de las celdas vecinas y se elige aquella con el valor más grande en  $P_{ij}$ . Si existen dos valores máximos iguales se elige de manera aleatoria cualquiera de ellos con misma probabilidad.
3. En caso de dos o más peatones intentará moverse a una misma celda objetivo, entonces sólo uno de ellos tomará esta celda asignada de manera aleatoria con igual probabilidad.
4. Dentro de un paso de tiempo, en flujo bidireccional dos peatones se les permite conmutar sus posiciones si y sólo si ellos tienen como celda destino la que ocupa el otro.

#### 3.3.2. Modelo de patrones de colisión

Este modelo es desarrollado por Narimatsu et al. [?]. Se inspira en la idea de que los peatones evitan colisiones con otras personas. Por ello el modelo dice que las reglas locales que rigen el movimiento de los peatones se determinan automáticamente sobre el proceso de simulación. Por lo tanto introducen el concepto de “patrón de colisión” sobre un área de visión, y cada persona se enseña a no chocar con otras.

El peatón tiene permitido tres direcciones de movimiento, hacia delante, al frente a la izquierda y al frente a la derecha, tal como se muestra en la figura ??. A cada peatón se le asigna una de las 4 velocidades máximas, la cual mantiene en todo momento. El área de visión cambia dependiendo de la dirección del peatón. Se prohíbe los movimientos de superposición donde los peatones pretendan moverse a celdas que se encuentren ocupadas.

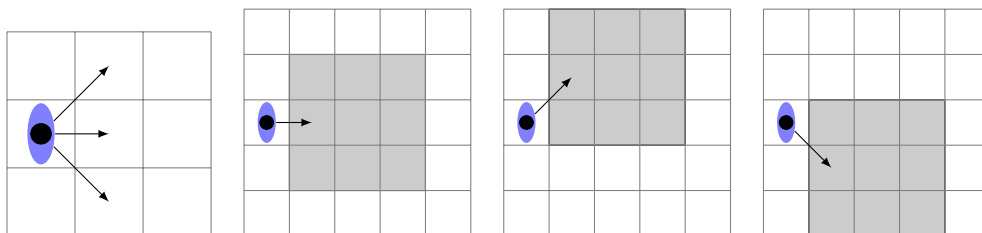


Figura 3.8: Ilustración del área de visión para cada movimiento.

### Adquisición de reglas locales

1. En un principio todos los peatones fluyen dentro del pasillo. Con frecuencia cada uno choca con otros peatones.

2. Cuando una persona choca con otra con dirección opuesta, se memoriza el patrón del área de visión, como un patrón de colisión. Este patrón se caracteriza de la siguiente manera:
  - a) Existencia de una pared (si: Y, no: N)
  - b) Número de personas con la misma dirección.
  - c) Número de personas con dirección opuesta.

Un ejemplo de este patrón se muestra en la figura ???. En este caso, el patrón de colisión se indica como “N0102”

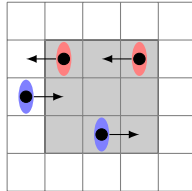


Figura 3.9: Ejemplo del patrón de colisión: N0102.

3. Este patrón no tiene información de la posición de los peatones, por lo tanto un patrón como en el ejemplo anterior, podría causar la colisión entre peatones. Es por esto que se determina una “probabilidad de colisión” la cual se estima del resultado de las simulaciones de 2000 tiempos.

$$\text{Probabilidad de Colisión} = \frac{B}{A}$$

A: Número de peatones tienen este patrón.

B: Número de colisiones con peatones que tienen este patrón.

La probabilidad se considera cero si  $A$  es menor a 100. Tanto el patrón de colisión como la probabilidad se le informa a todos los peatones simultáneamente.

4. El peatón pone atención sobre la área de visión y tiende a elegir el patrón cuya probabilidad de colisión es mínima. Si hay varios patrones con la misma probabilidad de colisión, las personas eligen cualquiera de ellos de manera aleatoria.
5. Cuando ocurre una colisión, se regresa a (2) y entonces el proceso continúa.

Los autores muestran resultados de sus simulaciones en diferentes tiempos, observando que a partir de los 6000 pasos de tiempo líneas dinámicas logran formarse, semejándose a personas formadas en una fila.

### 3.3.3. Modelo de campo de piso

Éste modelo fue desarrollado por Burstedde [?] quien introduce la idea del campo de piso. Las interacciones entre individuos para distancias cortas son de tipo repulsivas y en distancias grandes son de tipo atractivas. Así los peatones mantienen su distancia y tiene la ventaja de seguir a otras personas con la misma dirección.

El modelo maneja dos tipos de estados de ánimo: “feliz” o “infeliz”. El estado feliz son para los peatones que tratan de moverse en su dirección deseada, mientras que los peatones con estado infeliz se mueven aleatoriamente. De esta manera los peatones no bloquean el camino a otras personas reduciendo así embotellamientos. Este estado de ánimo se alcanza después de 4 iteraciones sin moverse.

$M_{-1,-1}$	$M_{-1,0}$	$M_{-1,1}$
$M_{0,-1}$	$M_{0,0}$	$M_{0,1}$
$M_{1,-1}$	$M_{1,0}$	$M_{1,1}$

Figura 3.10: Matriz de preferencias  $M = (M_{ij})$ .

### Modelo

Cada peatón cuenta con una matriz de preferencias la cual contiene las probabilidades de moverse a una de las 8 celdas a su alrededor [?]. La actualización de las posiciones se hace en paralelo para todos los peatones. Cuyo paso de tiempo es de 0.3 s.

### Campo de piso

Éste campo se puede pensar como una segunda cuadrícula, y puede ser discreto o continuo. Se distinguen de dos tipo de campo de piso: estáticos y dinámicos. El campo de piso estático  $S$  no cambia con el tiempo ni con la presencia de los peatones. Se utiliza para modelar la atracción del peatón a una zona de interés, por ejemplo puertas de emergencia o escaparates de tiendas.

Cada peatón cuenta con un campo de piso dinámico  $D$  el cual se modifica con la presencia de otras personas. Este campo se usa para modelar la atracción entre peatones por medio de una estela que dejan a su paso. La estela con el tiempo se desvanece. De esta manera los peatones pueden seguir los pasos de otros.

La probabilidad de transición  $p_{ij}$  para la dirección  $(i, j)$  está dada por

$$p_{ij} = NM_{ij}D_{ij}S_{ij}(1 - n_{ij}) \quad (3.8)$$

donde  $n_{ij}$  es 0 si la celda  $(i, j)$  está vacía y 1 si está ocupada,  $N$  es una factor que asegura que  $\sum_{(i,j)} p_{ij} = 1$  para las nueve celdas de dirección,  $S = (S_{ij})$  y  $D = (D_{ij})$  son las matrices asociadas al campo de piso estático y dinámico respectivamente.

El algoritmo de movimiento es el siguiente:

1. El campo dinámico  $D$  se modifica de acuerdo a las reglas que se verán en la siguiente sección.
2. Para cada peatón, calcula la probabilidad de moverse a una de las 8 celdas desocupadas a su alrededor, determinada por la ecuación (??).
3. El peatón elige una celda basado en la matriz de probabilidad de transición  $P = (p_{ij})$ .
4. Los conflictos cuando dos peatones o más intentan moverse a la misma celda, se resuelve eligiendo la mayor probabilidad relativa de cada peatón, como se muestra en la figura ??.
5. Los peatones tienen permitido avanzar solo una celda en cada paso de tiempo.
6. Los peatones alteran el campo dinámico de la celda que ocuparon después de moverse.

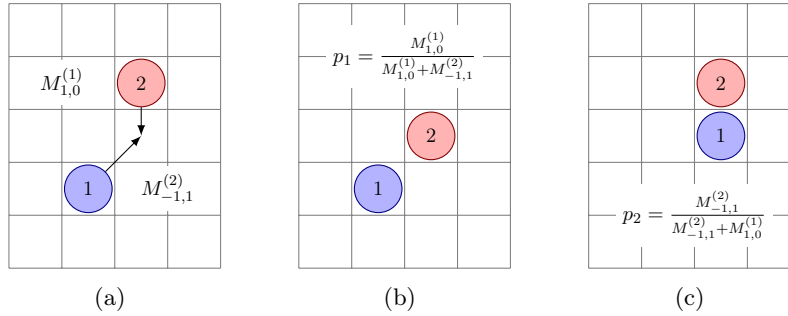


Figura 3.11: Solución de conflictos para el caso de dos personas con misma celda objetivo con matrices de preferencia  $M^{(1)}$  y  $M^{(2)}$  que corresponde al peatón 1 y 2 respectivamente.

A continuación veremos la forma explícita de la interacción entre peatones, modificando el campo dinámico, nos concentraremos en los campos discretos; en caso de que el lector desea saber de los campos continuos lo puede consultar en [?].

El campo discreto maneja dos tipos de partículas elementales: bosones y los fermiones<sup>3</sup>. Cada *fermión* esta asociado a un peatón. Los bosones impulsa a los fermiones a una dirección de preferencia e inducen la interacción entre ellos. Para cada fermión, la probabilidad de saltar a una celda aumenta de acuerdo al número de bosones en cada celda. Por lo tanto, el movimiento se determina por el gradiente del campo de piso, es decir, por la densidad de bosones.

A su vez los bosones se dividen en estáticos ( $s$ -bosones) y dinámicos ( $d$ -bosones). Al inicio de la simulación, cada celda se fija con un número específico de  $s$ -bosones  $\tau_s(x, y)$ . Siempre que un fermión salte de la celda  $(x, y)$  a otra celda, el número de  $d$ -bosones a su alrededor de la celda destino se incrementa a uno, formando una estela

$$\tau_d(x, y) \rightarrow \tau_d(x, y) + 1. \quad (3.9)$$

Después de que el fermión se mueva, el  $d$ -boson más viejo se elimina con una probabilidad  $p_e$ , siempre y cuando el tiempo de vida del bosón sea mayor a 1.

Para cada paso de tiempo los fermiones localizan las celdas vacías a su alrededor ( $n_{ij} = 0$ ). La probabilidad de saltar de la celda actual a una celda desocupada  $(i, j)$  esta dada por

$$p_{ij} = N e^{BJ_s \Delta_s(i,j)} e^{BJ_d \Delta_d(i,j)} (1 - n_{ij}) d_{ij} \quad (3.10)$$

con

$$\Delta_s(i, j) = \tau_s(i, j) - \tau_s(0, 0) \quad y \quad \Delta_d(i, j) = \tau_d(i, j) - \tau_d(0, 0) \quad (3.11)$$

donde  $d_i$  es el factor de correlación tomando en cuenta la dirección de la partícula en la celda 0 de donde proviene. Las variables  $J_s$  y  $J_d$  controla la intensidad de acoplamiento entre los fermiones y los bosones estáticos y dinámicos, respectivamente. Por su parte  $B$  juega el papel de una temperatura inversa.

El factor de correlación  $d_{ij}$  se introduce con el fin de que los fermiones no confundan su estela.

<sup>3</sup>Establece que no puede haber dos fermiones con todos sus números cuánticos idénticos. Los bosones pueden tener estados cuánticos simétricos por lo tanto no obedecen el principio de exclusión.



Sin embargo, esta formula no esta libre de colisiones. Por lo que, si  $m$  fermiones tratan de moverse a las misma celda, solo a uno de ellos realiza este movimiento con una probabilidad  $1/m$ . Ésta es otra forma ademas de la vista en la figura ??.

En las simulaciones mostradas en [?] se observa formación de líneas dinámicas en un pasillo. Para situaciones de evacuación se obtiene un semicirculo alrededor en una puerta de salida. Una de las ventajas de los campos de piso es que el peatón puede desplazarse a través de geometrías complejas.

A pesar de que los autores de [?] muestran simulaciones de un pasillo con líneas dinámicas, éstas no presentan un aspecto real. Las líneas son totalmente horizontales.

### 3.3.4. Modelo de distancias sociales

Este modelo lo presentan Jaroslaw et al. [?] donde el espacio esta representado por celdas de 0.25 cm. Los peatones se modelan por elipses cuyo centro coincide con el centro de la celda ocupada sobre una rejilla, por lo que toman en cuenta alguna geometría que limita a cada celda. Cada elipse tiene  $a = 0.225$  cm en el eje semimayor y  $b = 0.135$  cm de eje semimenor, estas medidas son tomadas como una aproximación al tamaño de las personas. Los peatones pueden transferirse a otra celda en una vecindad de Moore de radio 1. Una persona puede tomar cuatro diferentes posiciones dentro de la celda tal como se muestra en la figura ??, los cuales corresponden al giro de la elipse a:  $0, \pm 45, \pm 90$  y  $\pm 135$  grados respectivamente. Para cada paso de tiempo se determina la combinación de posiciones permitidas tomando en cuenta tanto la orientación de las elipses adyacentes y la intersección entre ellas. Las posiciones permitidas son aquellas donde la razón de intersección es menor a la tolerancia impuesta  $\epsilon_N \in [0, 1]$ .

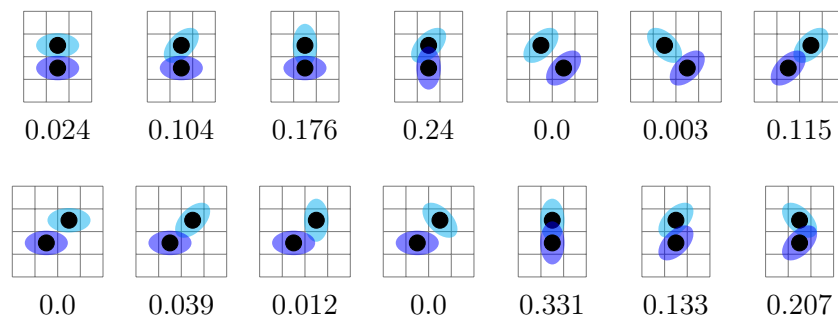


Figura 3.12: Combinación de 14 orientaciones de dos personas y cálculos de intersección de áreas.

### Representación de las distancias sociales

El área social del peatón se representa por medio de una elipse que emula el efecto territorial del peatón. El centro de la elipse puede o no corresponder al centro del peatón. Los autores proponen que el centro del área social se encuentre a una distancia  $t$  enfrente del centro de peatón, de manera que el área de visión sea mayor al frente del peatón, dando así mayor importancia a lo que ocurre delante de él.

En la figura ?? se presenta la manera de calcular la distancia social entre  $O$  el “observado” y los “intrusos” ( $A, B$  y  $D$ ). Aquí  $r_A$  es la distancia normalizada del centro del observado al centro del intruso ( $|OA|$ ), entre la distancia proyectada a la frontera del área social en dirección del intruso ( $|OA'|$ ). La distancia social entre peatones se describe como la “fuerza de la distancia social”, los autores proponen 4 formas de tal fuerza, como se presenta en la figura ??. La fuerza total que afecta al observador es la suma de las fuerzas de cada intruso que se encuentran dentro del área social. (por ejemplo:  $\mathbf{F}_s = \mathbf{F}_A + \mathbf{F}_B + \mathbf{F}_C$ ).

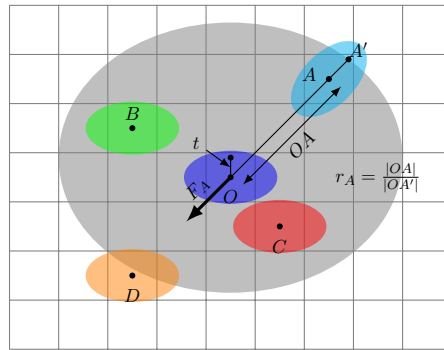


Figura 3.13: Elipse del área social: el eje semimayor equivale a  $4a$  y el eje semimenor a  $5b$ . Desplazamiento  $t$  a  $0.7b$

Cabe mencionar que esta fuerza no afecta a la orientación del peatón (elipse).

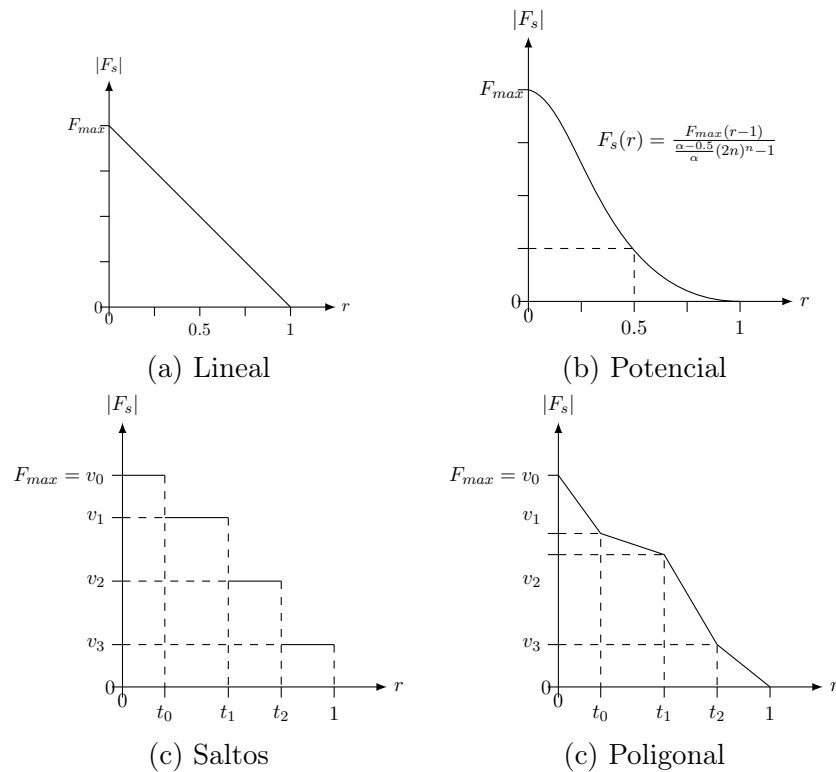


Figura 3.14: Ejemplos del tipo de fuerza social que puede aplicarse para mantener una distancia entre individuos.

### Algoritmo de movimiento

El algoritmo de movimiento consta de tres estados: *ir a*, *esperar en un objetivo intermedio* y *esperar*.

1. Verificar estado actual. Si el estado es *ir a* entonces ir a 2, si es *esperar en un objetivo intermedio* entonces ir a 4 ó si es *esperar* entonces ir a 5.

2. De la vecindad de Moore de radio 1, se elige la celda con menor potencial. En caso de que haya más de una celda con potencial mínimo, entonces elegir aquella que tenga el máximo grado de libertad.
3. Verificar si es objetivo intermedio. Si lo es, entonces ir a 4, si no entonces regresar a 1.
4. Si el objetivo es una salida, entonces detenerse. Si no, entonces verificar si existe un nuevo objetivo e ir a 1. Si no hay nuevo objetivo entonces ir a 5.
5. Verificar si se desea dejar el área de espera. Si se desea dejar entonces ir a 1. Si no ir a 6.
6. Verificar si se ha violado su área social, si si entonces determinar de una vecindad de radio 1 la celda más cómoda, si no entonces ir a 5.

La orientación del peatón cambia durante el movimiento a la nueva celda. El tipo de orientación que lleva el peatón siempre es en dirección del movimiento, en caso de tener una orientación no permitida entonces se elige una que si lo sea, de manera aleatoria. Cabe mencionar que las fuerzas sociales no afectan al cambio de orientación.

Los peatones poseen objetivos a los cuales se mueven a través de un potencial de campo (similar al campo de piso) donde se elige la celda con menor valor del campo. En caso de que existieran más de una celda a elegir, se escoge cualquiera de ellas de forma aleatoria.

El estado, donde las fuerzas sociales tiene influencia en los peatones es cuando están en estado de *espera*. Cada peatón en este estado es influenciado por lo demás peatones. Si las fuerza social exceden el umbral, entonces el peatón calcula una nueva celda objetivo, cambiando su estado a *ir*.

A continuación describiremos los modelos que se han implementado: Modelos de Klüpfel (apartado 3.4) y el Modelo de dos procesos (apartado 3.5).

### 3.4. Modelo de Klüpfel

Esta sección se enfoca al modelo de Klüpfel [?]. A diferencia de la mayoría de los modelos con autómata celular, éste trabaja con una velocidad máxima mayor a uno, por lo que cada peatón puede avanzar más de una celda en cada paso de tiempo.

Las suposiciones del modelo para el estudio del movimiento de personas son [?]:

1. La dinámica se puede representar por un autómata celular, es decir, por un modelo espacial y temporalmente discreto.
2. La orientación se basa en la señalización hacia un salida. Las desviaciones a la salida se pueden considerar como fluctuaciones de la dirección. La elección de la ruta está predeterminada.
3. Una conducta irracional se pueden incluir en desviaciones de movimiento como fluctuaciones de la velocidad.
4. Cada persona trata de optimizar el tiempo de recorrido de un punto a otro.
5. Las características individuales se pueden tomar como parámetros que determinan la conducta del peatón, principalmente la velocidad de caminata y la capacidad de orientación.
6. Las diferencias anteriores y por tanto los parámetros se comportan como distribuciones estadísticas.

Estas suposiciones se justifican por el hecho de que el proceso cognitivo<sup>4</sup> se basa en decisiones que están más allá de nuestro entendimiento [?]. Las llamadas “Heurísticas simples<sup>5</sup>” con frecuencia son más difíciles de modelar que la situaciones a detalle. Por lo tanto, esta etapa de desarrollo para adquirir habilidades cognitivas es aceptada y no se modelan. En su lugar, los resultados de estos procesos son directamente incorporados en la simulación, por ejemplo, la percepción y el proceso de señalización de salida, como se muestra en la figura ??, se transforma en información acerca de la mejor dirección de caminata incorporada en las celdas.

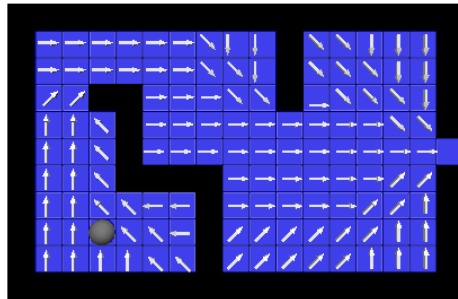


Figura 3.15: Señalización para guiar al peatón hacia la salida más próxima. Las flechas son vectores apuntando a la celda de salida.

### 3.4.1. Definiciones básicas

El sistema es una cuadrícula de dimensión  $x_{max} \times y_{max}$  celdas. El control de las celdas se maneja numerando de izquierda a derecha y de arriba hacia abajo. Por comodidad hemos cambiado esta última parte, numeramos de abajo hacia arriba. Así la vecindad  $\mathcal{N}$  para la celda  $r$  es  $\{r - v_{max}(1 + x_{max} + v_{max}i + j)\}$  con  $i, j = 0, 1, \dots, 2v_{max}$ , donde  $v_{max}$  es la velocidad máxima del peatón. Por lo tanto la vecindad de radio  $v_{max}$ , consta de  $(2v_{max} + 1)^2$  celdas.

En la figura ?? se observa la numeración de las celdas de una estructura junto con tres peatones de color azul. Debido a que la velocidad máxima de cada peatón es diferente, el tamaño de la vecindad alrededor de ellos cambia.

Las celdas son ocupadas por paredes, obstáculos o por peatones y cada una contiene una señalización hacia la salida más próxima. Para este caso los estados que se manejan son:

$$\mathcal{S} = (o, w, n, P(n), V) \quad (3.12)$$

donde  $w$  y  $o$  determinan si la celda es una pared o un obstáculo, respectivamente. Los valores pueden ser cero o uno, es decir, si  $w$  es uno entonces la celda es una pared. Cada celda solo puede ser ocupada por un peatón. El término  $n$  es el índice del peatón ocupando la celda,  $P(n)$  el conjunto de parámetros que describe al peatón  $n$ , y finalmente  $V$  es la dirección, la cual se puede interpretar como un gradiente. Para el caso de geometrías más complejas como en la figura ??, la orientación es básica para las simulaciones. Las direcciones se pueden representar mediante un campo escalar o vectorial. Existen diferentes técnicas para construirlas, ver [?, ?].

<sup>4</sup>Los procesos cognitivos, es decir aquellos procesos psicológicos relacionados con el atender, percibir, memorizar, recordar y pensar, constituyen una parte sustantiva de la producción superior del psiquismo humano.

<sup>5</sup>Las Heurísticas son reglas simples y eficientes que se han propuesto para explicar cómo la gente toma decisiones.

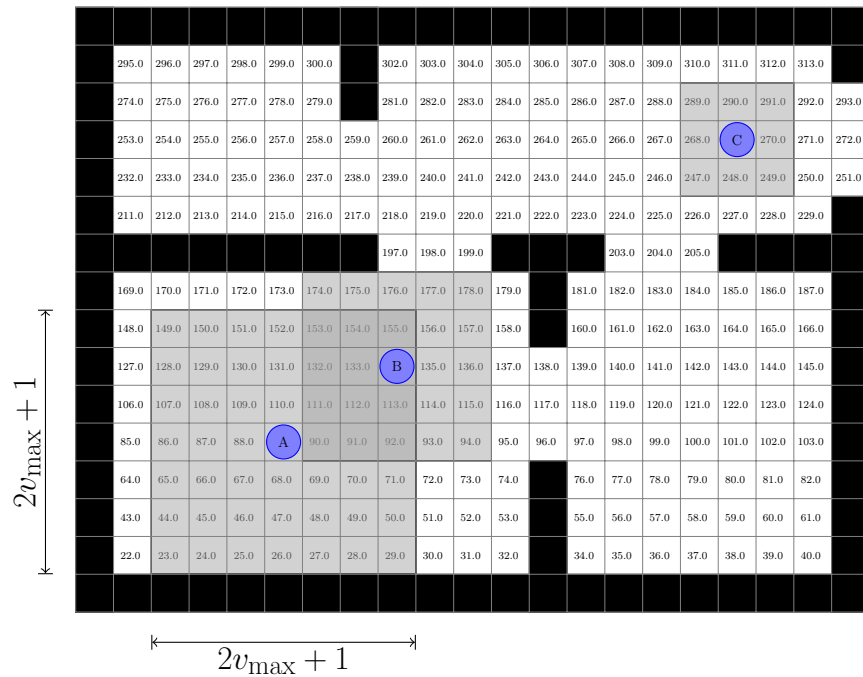


Figura 3.16: Los círculos azules representan las celdas ocupadas y sus vecindades respectivas en tonos de grises.

Cabe señalar que los tres parámetros de mayor importancia son: la longitud de la celda  $a$  (en metros/celda), la velocidad máxima  $v_{\max}$  (en celdas/paso de tiempo) y  $\Delta t$  (paso de tiempo). La longitud de las celdas se puede calibrar usando la densidad observada  $\rho_{\max}^{\text{emp}}$ : la relación entre estas es  $\rho_{\max}^{\text{emp}} \approx \frac{1}{a^2}$ . Algunos autores manejan esta densidad como 6 personas por metro cuadrado [?], luego la longitud de la celda es  $a = 0.4$  m. La velocidad máxima y la escala de tiempo (tiempo de reacción) están relacionados mediante

$$v_{\max}^{\text{emp}} = \frac{a \cdot v_{\max}}{\Delta t} \quad (3.13)$$

Si  $\Delta t$  se interpreta como el tiempo de reacción,  $v_{\max}$  se puede determinar midiendo este tiempo. Los resultados de pruebas psicológicas indican que el tiempo de reacción (el tiempo para tomar una decisión) es de 1 segundo [?]. Por consiguiente, la velocidad máxima está entre 2 y 5 celdas por  $\Delta t$  (de acuerdo a la población en estudio). Lo cual corresponde a una velocidad de 0.8 a 2 m/s.

Por su parte, las fluctuaciones que existen en la dirección  $V$  son gobernadas por los parámetros  $p_{\text{stop}}$  es la probabilidad de desacelerar (detenerse en un subpaso de tiempo) y  $p_{\text{sway}}$ , es la probabilidad de desviarse de una dirección dada. Los valores para  $p_{\text{stop}}$  están entre 0 y 0.1 y  $p_{\text{sway}}$  entre 0 y 0.03 (distribuidos uniformemente) [?]. Cuando  $p_{\text{stop}}$  y  $p_{\text{sway}}$  son cero (modelo determinista), los peatones no disminuyen su velocidad ni se detienen, por lo que siempre llevan una velocidad constante y la separación entre ellos es la misma todo el tiempo. Esto resulta imposible en situaciones reales, por ello se recomienda usar un modelo estocástico.

Ahora bien, habitualmente la regla de correspondencia de la función de transición (ecuación (??)) se define a través de un conjunto de reglas que cambian el conjunto de estados de una célula tomando en cuenta los estados de las celdas de su vecindad  $\mathcal{N}$ . Un peatón sólo puede moverse a una celda (vacía) que está en su vecindad  $\mathcal{N}$  y que no sea un obstáculo o pared. En caso de que el peatón se mueve, cambian los estados tanto de la celda que dejó como a la que se movió.

### 3.4.2. Algoritmo de movimiento

Supongamos que  $r$  es la celda ocupada por un peatón, el conjunto de reglas que rigen el movimiento de los peatones del modelo de Klüpfel está definido por la secuencia siguiente:

1. Tratar de acceder a la celda deseada  $d$  ( $d = r + V(r)$ ).
2. Si no es posible, tratar de ir a una celda vecina. Primero se busca a  $45^\circ$  de la celda deseada, ver figura ??.
3. Si tampoco es posible acceder, entonces tratar de moverse a una celda vecina a  $90^\circ$  respecto a la celda deseada.
4. Si no es posible acceder a ninguna de esas 5 posibles celdas, entonces detenerse.
5. El peatón se detiene con probabilidad  $p_{stop}$ .
6. La probabilidad  $p_{sway}$ , se desvía a  $45^\circ$  respecto a la dirección elegida en las etapas 1 a 3.
7. Finalmente el peatón avanza a la celda calculada en los pasos anteriores.

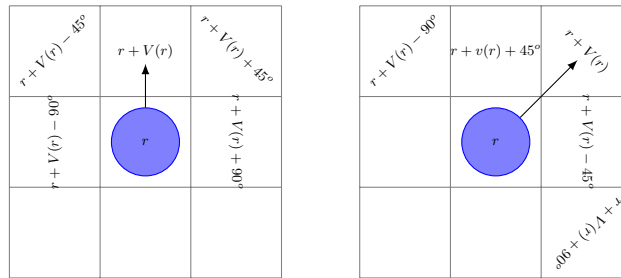


Figura 3.17: Ilustración de la notación usada en las direcciones del algoritmo de movimiento. La fecha corresponde a la dirección deseada del peatón.

La figura ?? muestra el diagrama de movimiento de los peatones. Note que el valor  $\sqrt{2} \approx 1.41$  se toma para los movimiento en diagonal y corresponde a una distancia mayor que la recorrida en los movimientos horizontales o verticales. Por ejemplo, para una velocidad  $v_{max} = 5$ , el peatón tiene permitido cinco pasos verticales u horizontales o cuatro pasos en diagonal. Este número de pasos es más claro cuando el tamaño de la celda es más pequeño, aproximándose a una vecindad de forma circular.

Dado que  $r$  es un escalar (número de celda) entonces  $V(r)$  también es escalar y denota l número de celda destino.  $V(r)$  puede manejarse como vector siempre que  $r$  sea la posición del peatón. Por conveniencia los desvíos en la figura se denotan como ángulos, más adelante mostraremos la forma de manejar estos ángulos como vectores o número de celda. Cuando  $d = r$  en peatón permanece en la misma celda, sin embargo el tiempo sigue corriendo por lo tanto  $t_{sub}$  aumenta en 1.

### 3.4.3. Trayectorias y distancia social

El algoritmo calcula la trayectoria total del peatón. Cada paso de tiempo  $\Delta t$  se divide en subpasos de tiempo donde todos los peatones calculan una parte de su trayectoria. Al completar  $\Delta t$  todos los peatones actualizan su posición, llamando a esto actualización en paralelo.

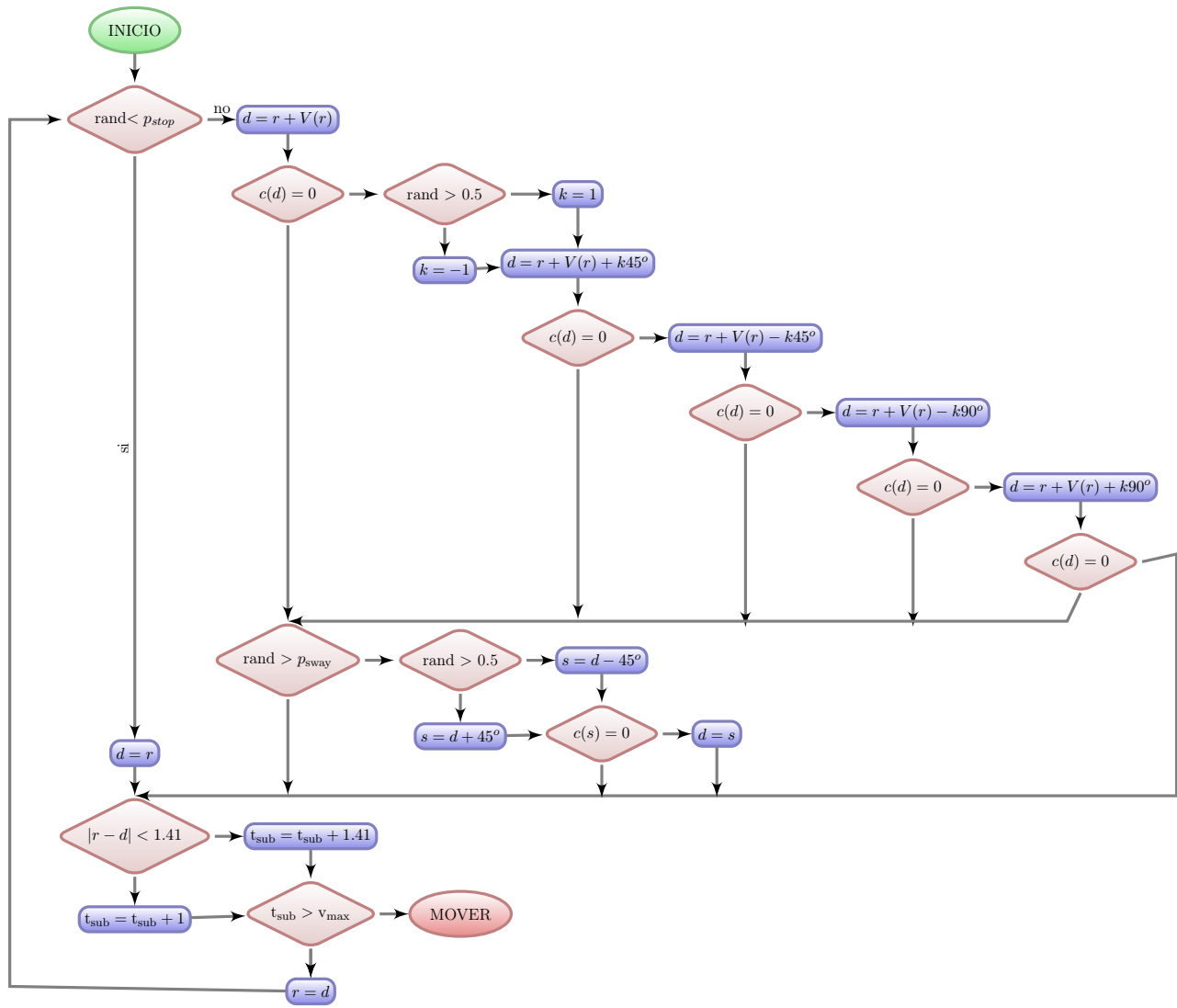


Figura 3.18: Diagrama de movimiento para el modelo de Klüpfel. El algoritmo se aplica hasta calcular la trayectoria del peatón. Donde  $r$  es la celda actual del peatón,  $V(r)$  la celda destino,  $c$  es la ocupación de la celda,  $t_{sub}$  la distancia recorrida en cada subpaso de tiempo y  $v_{max}$  es la velocidad máxima del peatón.

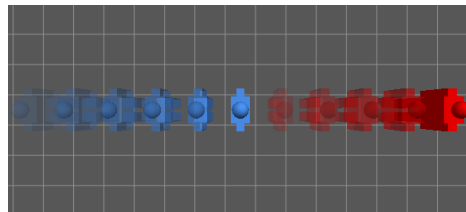


Figura 3.19: Ilustra la trayectoria de dos peatones caminando en misma dirección. La distancia entre ellos se debe a la velocidad del peatón rojo.

A diferencia del modelo de fuerza social [?], este modelo no mantiene una distancia entre los peatones, parecida al efecto territorial [?]. Sin embargo, si  $v_{max} > 1$  y el peatón bloquea todas las celdas que utiliza en su trayectoria, entonces automáticamente este mantiene una distancia entre el peatón que está detrás

de él, proporcional a la velocidad que lleva, ver figura ??.

Además, las velocidades máximas individuales  $v_{max}^i \neq v_{max}^j$  permiten a las personas avanzar un número máximo de subpasos:

$$v_{sub}^i = \begin{cases} 1, & \text{para } t = 1, 2, \dots, v_{max}^i, \\ 0, & \text{para } t = v_{max}^i + 1, \dots, v_{max}. \end{cases} \quad (3.14)$$

Las trayectorias y las posiciones en cada paso de tiempo se almacenan en un matriz  $T$  de tamaño  $N \times (v_{max} + 1)$ , donde  $N$  es el número total de peatones en el sistema y  $v_{max} = \max_i v_{max}^i$  es la velocidad máxima de la persona más rápida. Esto es  $T_{ij} \in \{1, 2, \dots, x_{max}y_{max}\}$ , donde  $r_{i1} = T_{i1}$  son la posiciones actuales de los peatones. Si cada peatón lleva una velocidad máxima menor a la velocidad máxima global  $v_{max}^i < v_{max}$ , las entradas posteriores a su trayectoria serán iguales, es decir, el peatón sólo tiene permitido avanzar  $v_{max}^i$  subpasos (celdas):

$$T = \begin{pmatrix} T_{10} & \cdots & T_{1v_{max}} \\ \vdots & \ddots & \vdots \\ T_{N0} & \cdots & T_{Nv_{max}} \end{pmatrix} \quad (3.15)$$

Una celda puede ser ocupada por a lo más un peatón, es decir, que  $T_{i,0} \neq T_{j,0} \forall i \neq j$ . Existen cuatro posibilidades concernientes a las restricciones de las trayectorias que se describen a continuación y se ejemplifican en la figura ??:

1. **Salta o para:** Si  $T_{iv_{max}+1} = T_{j1}$  para cualquier  $j \neq i$ . En este caso los peatones se mueven sólo si la celda destino está libre.
2. **Moverse lo más lejos posible:**  $T_{ir} \neq T_{jv_{max}+1} \quad \forall i \neq j, r$ . Esta versión permite cruce de trayectorias. El destino final es  $T_{iv_{max}} = T_{ir}$  con  $r = \max_{T_{is} \neq T_{jv_{max}}}(s)$ . A diferencia de la primera, el peatón avanza hasta encontrar otra persona en su camino.
3. **Subpasos:**  $T_{ir} \neq T_{jr} \quad \forall r, i \neq j$ . En este caso los peatones no tienen acceso a la misma celda en el mismo subpaso de tiempo. Para la actualización en paralelo esta versión puede requerir demasiado esfuerzo computacional, ya que se verifica la condición para todos los peatones en cada subpaso de tiempo.
4. **Sin cruce de caminos:**  $T_{ir} \neq T_{js} \forall i \neq j, r, s$ . Esta regla no permite el cruce de trayectorias.

Las cuatro reglas han sido implementadas por Klüpfel [?], ellos mencionan que la cuarta regla fue la mejor opción, debido a que se acerca más al diagrama fundamental empírico. Por ello, nos hemos decantado por esta regla. La implementación se realiza mediante el bloqueo completo de la trayectoria de cada peatón, posteriormente se actualizar la posición. Como ya hemos mencionado el cálculo de las trayectorias se realiza en subpasos de tiempo de  $\Delta t$ , calculando una parte de la trayectoria. Cada celda en la trayectoria es bloqueada cambiando su estado a ocupado, de esta manera se evita revisar toda la matriz de trayectorias para verificar que se cumpla la regla.



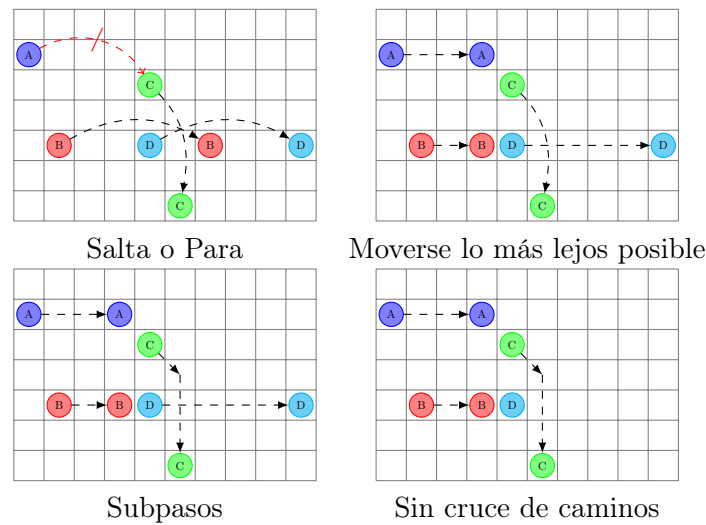


Figura 3.20: Las diferentes maneras de definir una trayectoria. Las líneas punteadas muestran la trayectoria del peatón. La línea roja significa que el peatón permaneció en su misma celda.

### 3.4.4. Modificación del modelo

#### Distancia social

De la experiencia que obtuvimos con el modelo de fuerza social observamos que el modelo de Klüpfel no considera la distancia social entre peatones. El problema se presenta justo cuando dos peatones se encuentran de frente, ya que al momento de cambiar de dirección ambos pueden desviarse por el mismo lado, permitiendo el cruce de trayectorias como se muestra en la figura ??.

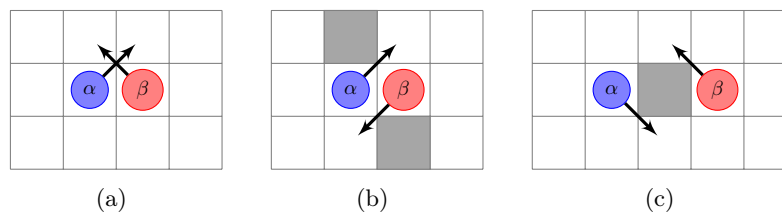


Figura 3.21: (a) muestra el caso donde ocurre cruce de trayectorias, (b) y (c) son soluciones al conflicto.

Por lo tanto sugerimos dos alternativas de solución a este problema, las cuales son:

1. Bloquear la celda a  $90^\circ$  en la dirección de desvío, el peatón que se actualice primero gana el lado bloqueando la celda. De este modo el otro peatón deberá elegir la dirección opuesta, como se muestra en la figura ??.
2. Bloquear una celda entre dos peatones que van en direcciones opuestas, de tal manera que el peatón que se desvíe primero obligará al otro a moverse en dirección contraria, ver figura ??.

Se eligió la segunda alternativa, ya que se mantiene una distancia social y solo bloquearíamos una celda en lugar de dos, lo cual daría mejor rendimiento en su ejecución. Para ello se aumento un estado más en las celdas donde 0 para las celdas vacías, 1 para las celdas ocupadas por un peatón y 0.5 para la celda

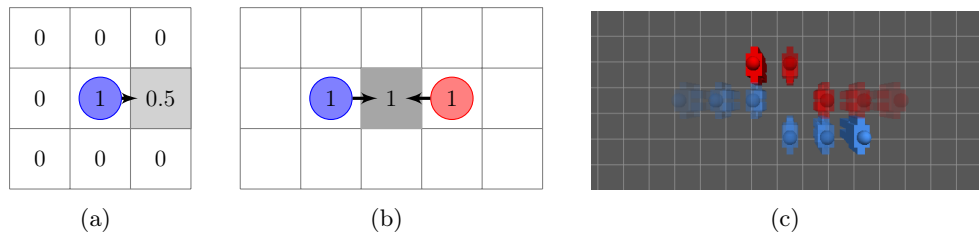


Figura 3.22: (a) muestra la configuración de las celdas donde 0 son las celdas vacías, 1 las ocupadas por un peatón y 0.5 por la presencia de un peatón, (b) conjuntando la celda de presencia y (c) desvió a partir de la celda como distancia social.

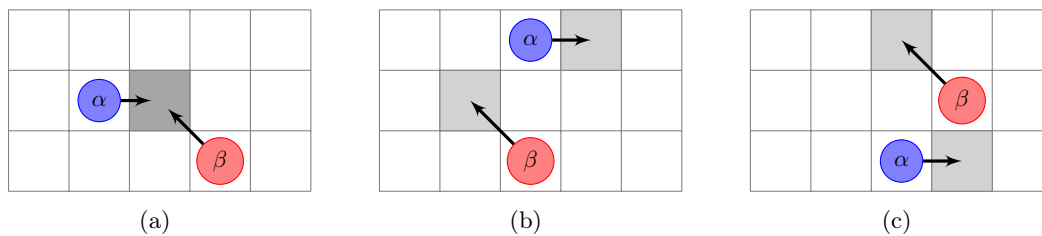


Figura 3.23: (a) Caso para cruce de direcciones, (b) y (c) son las posibles desviaciones. Cada uno siguen su dirección deseada llevando la celda de distancia el frente.

en dirección deseada, de esta manera los peatones sienten su presencia. El peatón puede avanzar ya que este valor no denota una celda ocupada, figura . En el momento en que dos peatones se encuentran de frente, la presencia de los dos se conjunta sumando estos valores dando como resultado una celda ocupada (artificialmente), figura ??, provocando el desvío.

En cruce de direcciones, la distancia social sigue permaneciendo haciendo que los peatones se desvíen en direcciones opuesta entre si. La figura

### 3.4.5. Simulaciones numéricas

En este apartado se presentan las simulaciones del tráfico peatonal uni y bidireccional sobre un pasillo utilizando el modelo de Klüpfel. Los programas se implementaron en la plataforma de *Python* y se presentan en el Anexo A. El pasillo esta formado de  $125 \times 25$  celdas ( $50 \times 10$  metros). Los peatones entran cada 2 segundos cuya posición inicial es aleatoria justo al borde del pasillo. Esta posición tiene una distribución normal centrada a la mitad del pasillo con 4 celdas de varianza. El número de peatones que entran también es aleatorio entre 0 a 6. Las velocidades que toman los peatones se distribuyen gaussianamente con media 1.34 m/s y varianza de 0.26 m/s.

Para estas pruebas numéricas no se consideraron fluctuaciones, por lo tanto el parámetro  $p_{sway} = 0$ .

Tuvimos que definir dos listas de direcciones para los movimientos permitidos de los peatones. Una contiene los vectores escalados de tal manera que el peatón pueda acceder a la celda deseada y la otra los números que puede tomar para acceder a las celdas vecinas. Es posible identificar a qué dirección nos referimos, ya que se numeran en sentido de las manecillas del reloj, ver figura ??

$$\text{Vec} = \{(0, a), (a, a), (a, 0), (a, -a), (0, -a), (-a, -a), (-a, 0), (-a, a)\}$$

$$\text{Num} = \{LC, LC + 1, 1, -LC + 1, -LC, -LC - 1, -1, LC - 1\}$$

donde  $LC$  es el número de celdas a lo largo del pasillo y  $a$  es el tamaño de la celda.

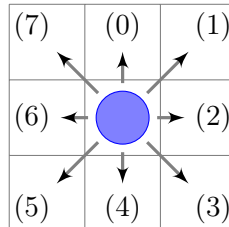


Figura 3.24: Las direcciones que puede tomar el peatón son numeradas en sentido de las manecillas del reloj.

### Modelo original

La primera simulación consiste en un pasillo con un flujo unidireccional de izquierda a derecha y se muestra en la figura ???. Las condiciones de frontera consideradas para esta simulación son abiertas por los extremos del pasillo.

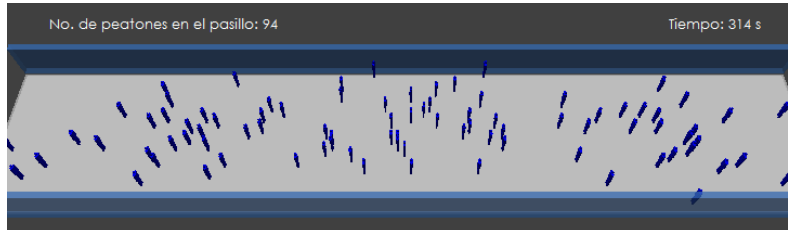


Figura 3.25: Simulación de un flujo unidireccional utilizando el modelo de Klüpfel en un pasillo, con condiciones de frontera abiertas.

Para la simulación bidireccional se expandió la cuadrícula de manera ficticia, ya que al introducir a los peatones y eliminarlos resultaba confuso, puesto que se eliminaban los que estaban en las fronteras del pasillo y al mismo tiempo los peatones que entraban. Por lo tanto, los peatones se introducen sobre una línea de celdas y se eliminan en otra, ver figura ???. Los peatones azules caminan de izquierda a derecha y los rojos en sentido contrario. Cabe mencionar que hemos podido simular cerca de 1000 personas dentro del pasillo. El principal problema del modelo original de Klüpfel es que no logra formar líneas dinámicas, los peatones no se auto-organizan. A partir de esto nos dimos cuenta que el modelo contenía otras problemáticas como no dejar una distancia entre peatones, ya que se desviaban casi al momento de chocar con otra persona. En estas desviaciones ocurrían cruces de trayectorias en movimientos diagonales. Por eso modificamos el modelo.

### Modelo modificado

En la figura ?? observamos la simulación del modelo modificado con la distancia social. Sin embargo con ésta modificación no fue suficiente para que el modelo sea capaz de reproducir las líneas dinámicas.

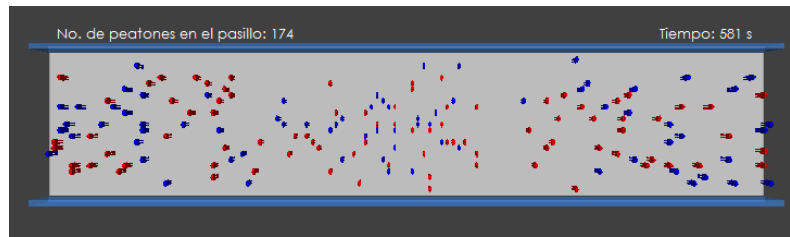


Figura 3.26: Simulación del flujo bidireccional en un pasillo con condiciones de frontera abierta. Los peatones azules avanzan de izquierda a derecha y rojos en sentido contrario. Los peatones no tienen una auto-organización.

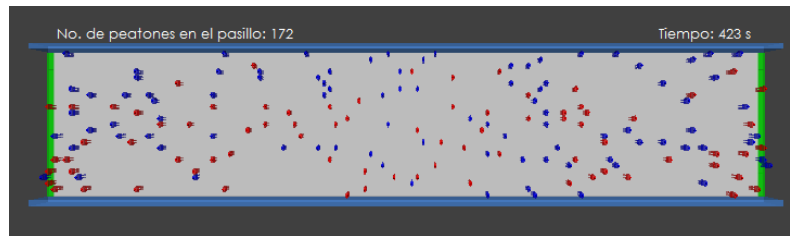


Figura 3.27: Modelo de Klüpfel modificado.

### Parámetro de lado derecho

Hao Yue et al. [?, ?] introducen un parámetro de lado derecho (modelo de parámetros dinámicos) para lograr que el flujo se divida en dos corrientes en una simulación bidireccional. Por lo que decidimos aumentar la probabilidad de desvío al lado derecho cuando el peatón no puede acceder a la celda objetivo. En la figura ?? se observa que a pesar de que los peatones eligen desviarse a la derecha en las simulaciones no es muy claro para densidades bajas, figura ???. Sin embargo, para densidades altas en pasillos con fronteras abiertas son un poco más claras las corrientes de flujo que forman los peatones al desviarse a su lado derecho, figura ??.

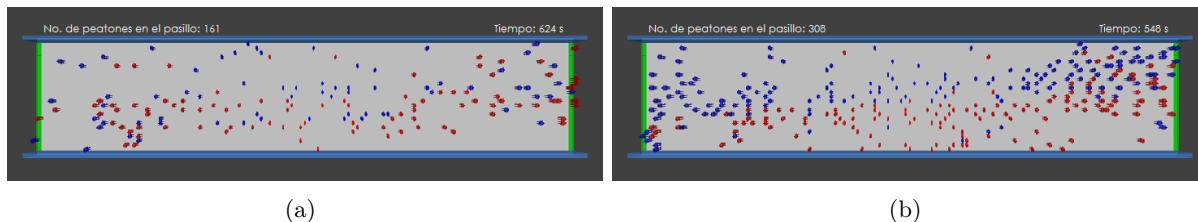


Figura 3.28: Flujo bidireccional con frontera abierta y parámetro de lado derecho.

Sin embargo, para el caso donde las fronteras son de tipo periódica la velocidad a la que se divide el flujo depende de la probabilidad que le asignemos a desviarse de lado derecho. Sin embargo, es posible que el flujo se divida sin tener que modificar esta probabilidad. Esto debido a que la información que obtiene los peatones de las celdas ocupadas y libres no se pierde una vez que estos hayan llegado a la frontera de esta manera los peatones llegan auto-organizarse, figura ??.

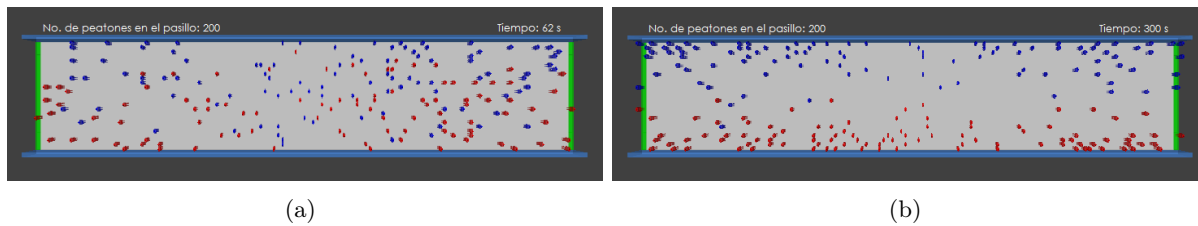


Figura 3.29: Flujo bidireccional con frontera periódica sin parámetro de lado derecho.

### Situaciones de bloqueo

En la figura ?? se presenta un bloqueo debido a la alta concentración de peatones dentro del pasillo. Esta situación se observa tanto para las fronteras abiertas como para las periódicas. Una vez que un pequeño grupo de personas se bloquean entre si es difícil que logran salir de esa situación provocando que se más personas se acumulen. Esta situación se debe a que no existen un parámetro que perturbe el comportamiento de los peatones y con ello se logre moverse.

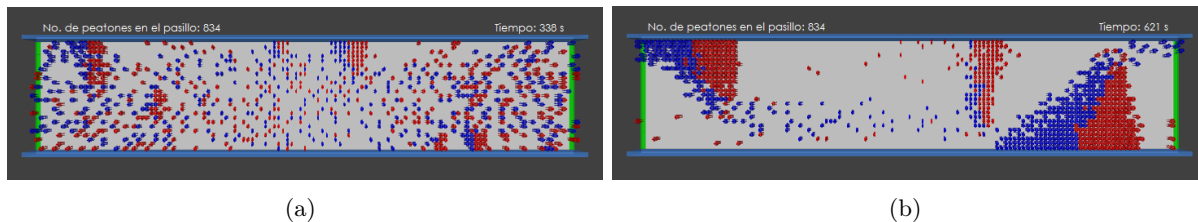


Figura 3.30: Situación de bloqueo. (a) comienza el bloqueo, (b) bloqueo casi terminal.

### 3.4.6. Resumen y observaciones

Este modelo considera una velocidad máxima mayor a 1 celda por paso de tiempo. Las direcciones de movimiento son: lateralmente, en diagonal y al frente. Es un modelo estocástico al tomar las probabilidades  $p_{stop}$  y  $p_{sway}$  que reduce la velocidad y desvía al peatón de una dirección dada respectivamente, si estas ambas probabilidades son 0 entonces el modelo es determinista.

En un inicio implementamos el modelo original de Klüpfel, uni y bidireccionalmente. De las simulaciones bidireccional observamos que no existían una auto-organización de los peatones. Notamos que el desvío de la dirección del peatón ocurre cuando desea esquivar a otra persona que se encuentra justo frente él. Dando una apariencia de que el peatón camina con la vista al piso, percatándose de lo que ocurre a unos centímetros delante de él. Además observamos que los peatones al momento de desviarse en un movimiento diagonal violaba el cruce de trayectorias.

A partir de estas problemáticas decidimos proporcionarle al peatón una distancia social de tal manera que se solucionara el cruce de trayectorias y pueda desviarse antes de chocar con otros peatones.

A pesar de que la distancia social logró resolver las problemáticas que teníamos con el modelo de Klüpfel, no se logro una auto-organización. Por ellos decidimos buscar e implementar otro modelo.

### 3.5. Modelo de dos procesos

Este modelo es introducido por Victor Blue y Jeffrey Adler en [?, ?], el cual tiene un parecido a los modelos para flujo vehicular con autómatas celulares [?]. El modelo supone que existen tres elementos fundamentales del movimiento peatonal bidireccional, *movimiento lateral*, *avance* y *mitigación de conflictos*. El movimiento lateral se refiere al cambio de línea (en flujo vehicular es cambiar de carril), ya sea para incrementar la velocidad o evitar conflictos. El avance se adapta a la velocidad deseada de cada peatón y a la posición de otros peatones frente a él. La mitigación de los conflictos es la manera que los peatones se aproximan entre sí en direcciones opuestas para evitar un punto muerto (bloqueo). Al igual que los modelos de parámetros dinámicos [?], el intercambio de posiciones es permitido, siempre y cuando dos peatones en dirección opuesta tengan  $gap = 0$  ó 1 celda de distancia con una probabilidad llamada  $p_{exchg}$  [?]. En el 2001 los autores aplican una pequeña modificación al modelo de tal manera que este sea capaz de simular el fenómeno de líneas dinámicas. Cabe mencionar que para los movimientos permitidos de los peatones se utiliza la vecindad de von Neumann como se muestra en la figura ??.

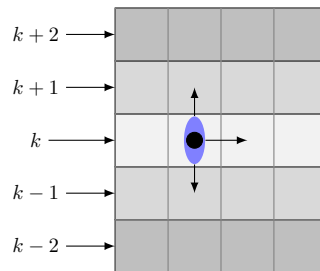


Figura 3.31: Ilustración de los movimientos permitidos del peatón en el modelo de dos procesos.

#### 3.5.1. Algoritmo de movimiento

El conjunto de reglas de movimiento se describe a continuación.

##### 1. Cambio de línea (Actualización en paralelo 1)

- a) Eliminación de conflictos: dos peatones que se encuentran lateralmente adyacentes no pueden cambiarse a la misma línea. Si existe una celda entre ellos entonces ésta se asignará con la misma probabilidad a uno de ellos.
- b) Identificación del gap: en la misma línea o líneas adyacentes se escoge la que tenga mayor avances hacia delante para alcanzar su velocidad máxima  $v_{max}$ , de acuerdo al subproceso para calcular el gap, seguido del paso de actualización de avance.
  - i Formación de líneas dinámicas:
    - Un paso fuera de la línea o carril de un peatón con dirección opuesta mediante la asignación  $gap = 0$ , si el peatón se encuentra dentro de 8 celdas.
    - Un paso hacia atrás de un peatón con la misma dirección cuando se evita a un peatón con dirección opuesta eligiendo cualquiera línea disponible con  $gap_{same,dir} = 0$  cuando  $gap = 0$
  - ii Las líneas con el mismo gap de longitud máxima se resuelve de la siguiente manera:
    - Para dos líneas adyacentes se escoge una misma probabilidad.

- Para dos líneas donde una de ellas sea la actual línea, se escoge seguir en la misma línea.
  - Para las tres líneas, se escoge seguir en la misma línea.
- c) Movimiento: cada peatón  $p_n$  se mueve a la línea elegida al completar los pasos  $i$  y  $ii$ .
2. Avance (Actualización en paralelo 2)
- a) Actualización de velocidad: Sea  $v(p_n) = gap$  donde el  $gap$  es calculado en 3.
  - b) Cambios: Si  $gap = 0$  o  $gap = 1$  y  $gap = gap_{opp}$  (es decir, para peatones en dirección opuesta). Entonces con probabilidad  $p_{exchg}$ ,  $v(p_n) = gap + 1$ , si no entonces  $v(p_n) = 0$ .
  - c) Movimiento: cada peatón  $p_n$  se mueve  $v(p_n)$  celdas hacia delante sobre la retícula.
3. Subproceso: Cálculo de  $gap$ , se toma 8 celdas adelante de visión como máximo ( $8 = 2 * V_{max}$ , donde  $V_{max}$  es la velocidad máxima más grande de todos los peatones).
- a) Peatones con misma dirección: Si se encuentra una celda ocupada con la misma dirección, entonces se asigna a  $gap_{same}$  el número de celdas entre los peatones, si no  $gap_{same} = 8$ .
  - b) Peatones con dirección opuesta: Si se encuentra un peatón con dirección opuesta, entonces  $gap_{opp} = \text{int}(0.5 * \text{número de celdas entre ellos})$ , si no  $gap_{opp} = 4$
  - c) Asignación del  $gap = \min(gap_{same}, gap_{opp}, v_{max})$

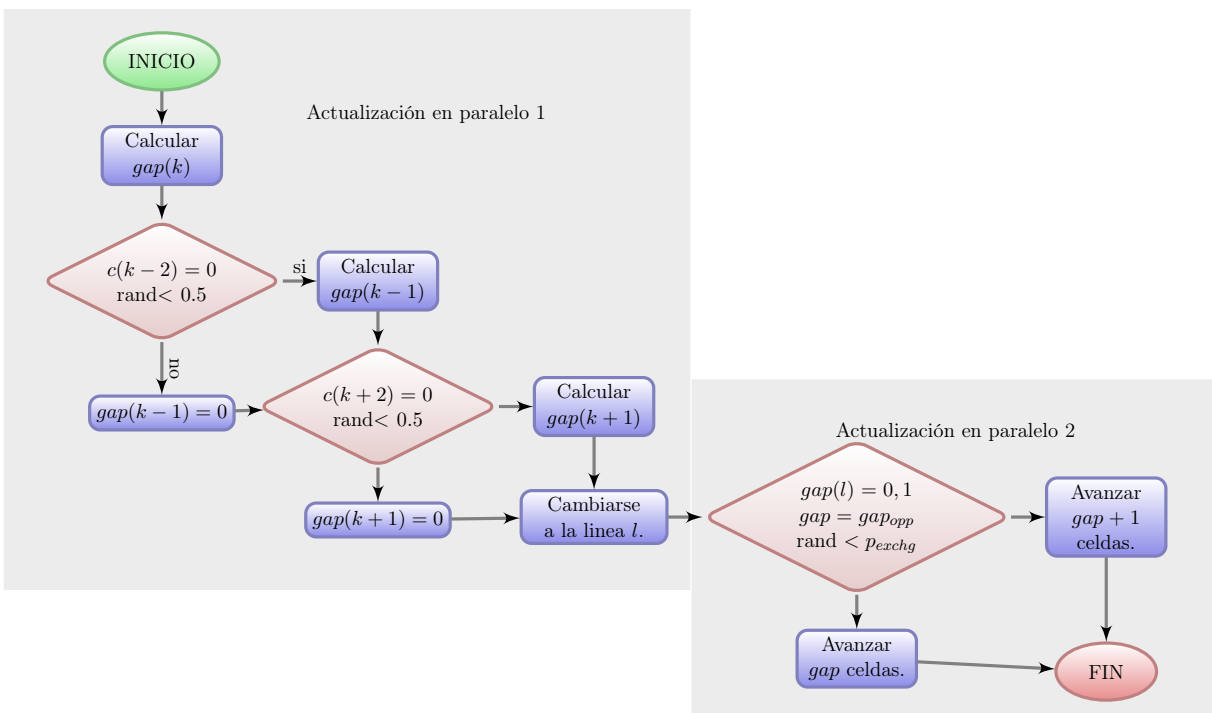


Figura 3.32: Diagrama de movimiento para el modelo de dos procesos. Donde  $c(k+2)$ ,  $c(k+1)$ ,  $c(k)$ ,  $c(k-1)$  y  $c(k-2)$  toma el valor 0 para las celdas vacías y diferente de 0 para las celdas ocupadas. La línea de mayor alcance se define como  $l = \{l \in \{k, k+1, k-1\} \mid gap(l) = \max[gap(k), gap(k+1), gap(k-1)]\}$ .

La primera etapa de actualización en paralelo, es un conjunto de reglas que se aplican a cada peatón para determinar el cambio de línea (carril), basado en las condiciones actuales. La línea que se escogerá es

aquella tenga mayor avance hacia delante ya sea izquierda, derecha o la misma línea en que se encuentra. Después de lo anterior todos los peatones actualizan su posición moviéndose a la línea elegida. La segunda etapa actualización en paralelo, consiste en un conjunto de reglas de movimientos hacia delante que se aplican a cada peatón. El movimiento permitido (y por lo tanto la velocidad) de cada peatón se basa en la velocidad deseada de cada uno de ellos, restringido al espacio disponible (*gap*) para avanzar. Finalmente todos los peatones “saltan” de la posición actual a la calculada.

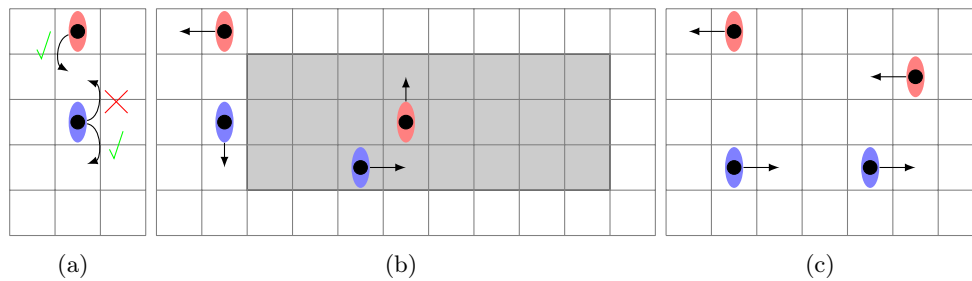


Figura 3.33: Ejemplo de la primera actualización en paralelo. (a) Muestra la eliminación de conflictos, (b) asignación de línea, (c) cambio de línea.

En la figura ?? se muestra un ejemplo de la primera actualización en paralelo. La figura ?? exhibe un ejemplo de cómo el peatón azul pierde la asignación de la celda de su izquierda debido a que la probabilidad favoreció al peatón rojo. La celda a su derecha está disponible por lo tanto se calcula que línea proporciona el mayor avance hacia delante. Finalmente en la figura ?? actualiza las posiciones; aquellos que no cambiaron de línea es porque en ella alcanzan su velocidad máxima.

### 3.5.2. Modificación del modelo

El modelo permite intercambio de posiciones cuando estos van en dirección contraria y se encuentran de frente o a una celda de distancia, ver figura ??.

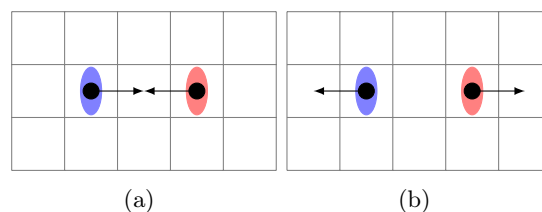


Figura 3.34: (a) Ilustra el caso cuando dos peatones se encuentran de frente (b) la probabilidad  $p_{exchg}$  favorece cambiar de posiciones, continuando así con camino.

Nosotros consideramos que las personas no realizan este tipo de cruce, por lo que descartamos la idea de intercambiar las posiciones o traslapes entre cuerpos. Por lo tanto, eliminamos esta parte junto con la probabilidad de intercambio  $p_{exchg}$ .

Por otro lado, el modelo no toma en cuenta algún tipo de distancia social, excepto cuando  $gap = 1$  para direcciones opuestas. Por ello decidimos aumentar la distancia social restando una celda de la siguiente manera

$$gap_{opp} = \text{int} (0.5 * \text{número de celdas entre ellos}) - 1$$



No debe haber problemas incluso cuando  $gap_{opp} = -1$  ya que siempre es una distancia positiva mayor a cero al calcular  $gap$ . Además consideramos 10 celdas de visión en lugar de 8. Así aseguramos la existencia de una distancia social entre ellos al calcular el  $gap$ , descartando la posibilidad de encontrarse cara a cara. Sin embargo hasta ahora no hemos podido introducir de otra forma una distancia social para considerar los lados del peatón.

### 3.5.3. Simulaciones numéricas

Se construyó un modelo a escala de un humano usando medidas estándar de la antropometría del cuerpo humano [?], con el fin de que las simulaciones sean mucho más realistas, ver figura ??.

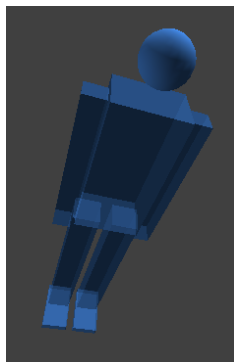


Figura 3.35: Figura de un humano realizada en *Python*

Para la simulación consideramos un pasillo de dimensiones 10 m de ancho y 50 m de largo, con celdas de tamaño 0.4 m, el paso de tiempo  $\Delta t = 1$  s, y la velocidad máxima de 5 celdas por segundo (2 m/s).

#### Modelo original

Comenzamos la simulación con el modelo original. En la figura ??, las franjas azules representan las paredes (fronteras cerradas) y las verdes con las entradas al pasillo (fronteras abiertas). Los peatones entran aleatoriamente de 0 a 10 personas por cada 6 segundos en los extremos. La densidad dentro del pasillo se mantiene cerca de las 190 personas. En la simulación notamos la formación de líneas dinámicas y el comportamiento se lleva a un estado óptimo. A pesar de estos resultados, creemos que el intercambio de posiciones (bajo  $p_{exchg}$ ) es algo que no debería de permitirse.

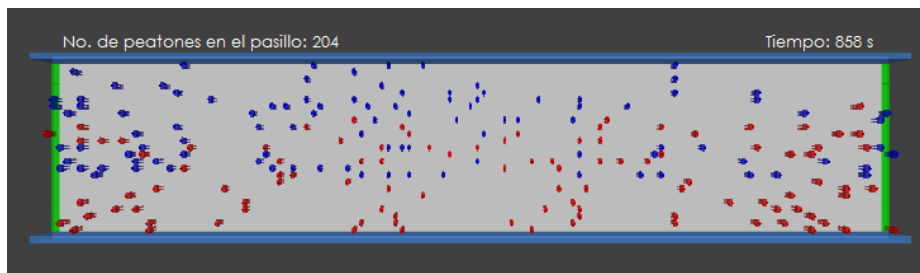


Figura 3.36: Simulación del flujo bidireccional mediante el modelo de dos procesos modelo original.

### Modelo modificado

En este caso utilizamos el modelo original sin el parámetro  $p_{exchg}$ . Con este modelo no se logró obtener la formación de líneas dinámicas. En ocasiones se formaban algunas líneas dinámicas. Sin embargo, después algunos pasos de tiempo se desvanecían por completo. Esto se debe a que los peatones al elegir la línea con mayor avance hacia delante, no les importa romper la formación con tal de alcanzar su destino, ver figura ???. Observamos que al quitar este parámetro poco realista se pierde la auto-organización.

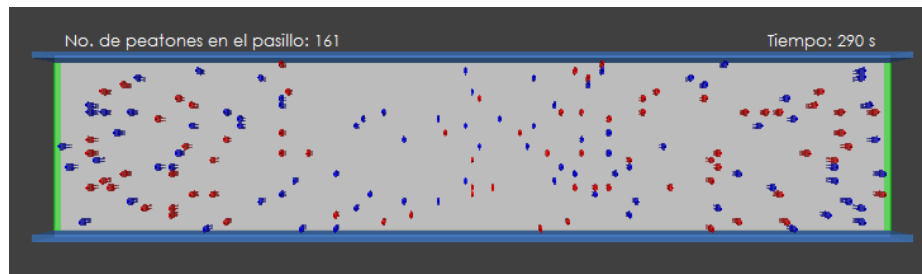


Figura 3.37: Simulación del flujo bidireccional mediante el modelo de dos procesos modificado con frontera abierta.

Decidimos utilizar fronteras periódicas con el modelo modificado, con el fin de ver el comportamiento y si se presentaba la auto-organización. La figura ?? muestra la simulación con una densidad de  $0.4 P/m^2$ . Sin embargo, no es capaz de llegar a un estado óptimo, es decir, dividir el flujo en dos corrientes. Ni siquiera ha sido capaz de reproducir líneas dinámicas aun contando con la información de las fronteras. Es claro que el modelo no tiene los elementos necesarios que propicie la reproducción de estos patrones.

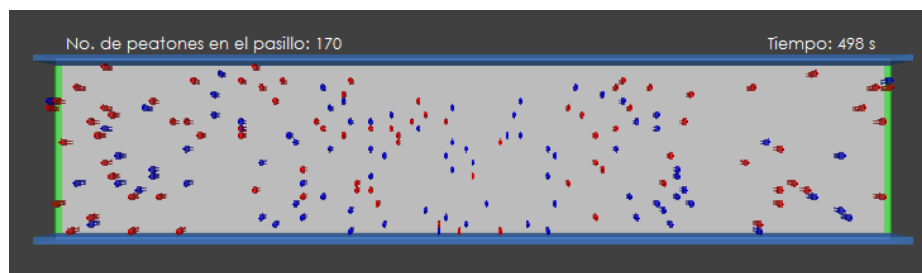


Figura 3.38: Flujo bidireccional mediante el modelo de dos procesos modificado con fronteras periódicas.

### Situaciones de bloqueo

A diferencia del modelo de Klüpfel, el modelo de dos procesos no presenta grandes bloqueos ya sea para fronteras abiertas o periódicas. Incluso se puede observar de la figura ?? que los peatones forman pequeñas hileras. El modelo mismo mantiene una perturbación en los peatones de tal manera que ambos flujos continúan su movimiento.

#### 3.5.4. Resumen y observaciones

En este modelo el movimiento de los peatones tiene dos posibles direcciones: lateralmente y al frente. Antes de realizar los movimientos laterales se evitan los conflictos en el cruce de trayectorias diagonales. Es importante observar que si se realiza un movimiento lateral solamente se contabilizan las celdas que

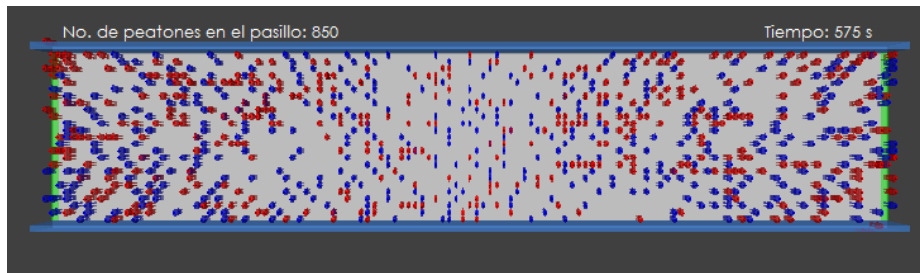


Figura 3.39: Situación de bloqueo. La densidad de peatones es de  $1.7 \text{ P/m}^2$

el peatón avanza hacia el frente. El modelo acepta el intercambio de posiciones lo cual permite la auto-organización. Una vez que se completó el cálculo de la trayectoria, los peatones salta de su posición inicial a la posición calculada. Este salto refleja un comportamiento poco realista en la simulación. Además el modelo original no considera una distancia social entre las personas, es decir, los peatones caminan muy cerca entre ellos e incluso se pueden atravesar.

### 3.6. Conclusiones

Hemos implementado dos modelos de autómatas celulares, el modelo de Klüpfel y el modelo de dos procesos. Hasta el momento, los modelos con AC que hemos estudiado no logran proporcionar simulaciones realistas. Debido a que el comportamiento competitivo de los peatones no alcanzan un estado de auto-organización de modo que los patrones del tráfico peatonal no se presentan. Además estos modelos no se manejan con la teoría del mínimo esfuerzo, ya que no toman cuenta a las personas que llevan la misma dirección. Eso provoca que en un flujo bidireccional no se formen líneas multidinámicas.

Ambos algoritmos han sido modificados con la idea de incluir la distancia social y la formación de líneas dinámicas. Pero no tuvimos éxito.

La capacidad máxima que hemos alcanzado en las simulaciones dentro del pasillo es de 1000 peatones, lo que resulta en un menor costo computacional comparado con el modelo de fuerza social.

Por lo anterior decidimos proporcionarle al peatón un campo de visión y un parámetro dinámico apropiado tal que le ayuden a tomar en cuenta a los peatones que llevan la misma dirección que él, evaluando con ello la posibilidad de seguirlos. Estas ideas se desarrollan en el capítulo 4.



## Modelo de campo social

En el capítulo anterior hemos visto algunos modelos basados en autómatas celulares aplicadas al tráfico peatonal. La mayoría de estos modelos tienen la finalidad de cambiar la línea de camino, mediante diferentes tipos de probabilidades o parámetros: prefijados o dinámicos. Estos últimos se calculan de acuerdo a las condiciones que existen alrededor de los peatones. Un factor importante para determinar el parámetro dinámico es definir apropiadamente la forma del área que influye en el comportamiento de los peatones.

Pocos modelos de autómatas toman en cuenta la distancia social entre peatones [?, ?]. En este capítulo presentaremos un modelo llamado campo social donde introducimos la idea de distancia social por medio de un campo social, simular al campo de piso dinámico (sección ??), el cual semeja el efecto territorio [?].

Nuestro propósito es combinar un parámetro dinámico junto con la superposición de diferentes campos de influencia, con el fin de obtener simulaciones numéricas que generen líneas dinámicas y evitar el contacto físico. Además el modelo de campo social puede reproducir el diagrama fundamental de Weidmann [?]. En el resto del capítulo se desarrollan estas ideas.

### 4.1. Descripción del Modelo

Nuestro modelo microscópico se basa en autómatas celulares e intenta simular el tráfico bidireccional sobre un pasillo. El AC se define sobre un malla de celdas (rectangular) de tamaño  $I_1 \times I_2$  contenida en  $\mathbb{R}^2$ . Cada celda es cuadrada de  $0.40 \times 0.40$  m<sup>2</sup>. Los peatones entran y salen por las fronteras laterales (izquierda y derecha). Se imponen condiciones de frontera cerradas para los bordes de arriba y abajo (no cruce de paredes). Además se emplean dos tipos de condiciones de frontera para los bordes laterales: abiertas y periódicas.

Existen tres elementos fundamentales del movimiento de peatones que un modelo microscópico debe tomar en cuenta: mitigar conflictos, movimientos laterales (intercambio de línea de caminata) y hacia adelante [?]. Por ello, los movimientos permitidos del peatón son los mostrados en la figura ??.

Recordemos que la formación de líneas dinámicas surge porque los peatones, en su mayoría, eligen gastar menos energía y caminar en líneas con la misma dirección. Por lo que el peatón sólo se preocupa por mantener la distancia con la persona que está de frente. En el caso de que no se formaran las líneas, los peatones se encontrarían de frente y continuamente tendrían que desviar su ruta evitando el contacto

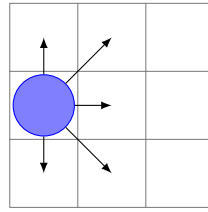


Figura 4.1: Movimientos permitidos en el modelo.

físico, lo que ocasionaría un mayor desgaste para el peatón. Un factor que permite la auto-organización de este patrón colectivo es la distancia social [?, ?, ?].

#### 4.1.1. Campo social

El efecto territorial incide en los cambios del comportamiento de los peatones, los cuales son guiados por campos sociales que motivan a los peatones a moverse. Helbing [?] toma esta idea para modelar el comportamiento de los peatones mediante fuerzas de repulsión de tal manera que se alejan de objetos o personas para no chocar y herirse. El campo social semeja un espacio individual donde cada persona se siente cómoda. Entonces el campo social actúa de manera similar a la fuerza de repulsión del modelo de fuerza social.

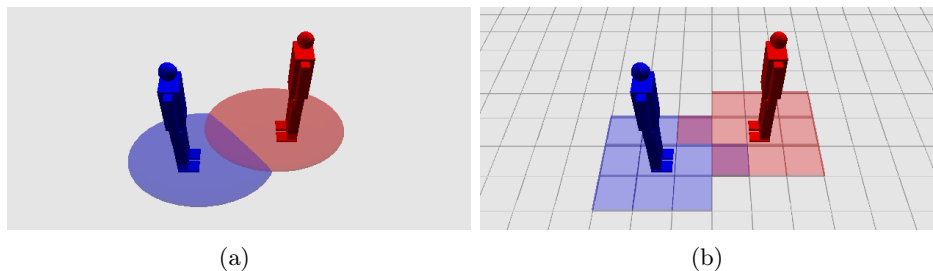


Figura 4.2: Ejemplos de campos sociales: (a) continuo y (b) discreto.

Cuando otra persona invade el espacio social del peatón, este se mueve en la dirección donde disminuye esta incomodidad, actuando como una fuerza de repulsión a otras personas, de esta manera se logra mantener la distancia social entre ellos. Además este campo aumenta la presencia del individuo de tal manera que los otros logren darse cuenta.

La figura ?? muestra dos tipos de campos sociales: continuo y discreto. La forma de los campos discretos es similar a los campos continuos, debido a que estos últimos son elipses.

Cada celda del campo social (discreto) se le asigna un valor, como se muestra en la figura ?. En caso de que los campos se intersecten, los valores de las celdas traslapadas se suman, ver figura ?. Luego, la intensidad o presencia del peatón aumenta. Asignamos el valor 2 al núcleo del campo donde se encuentra el peatón así la intensidad del campo es más fuerte en este punto. Las dos celdas que contienen el valor 0.5 asegura que el peatón no se confundirá con su propio campo. Los valores 1 a su alrededor proporciona una distancia para figura ??

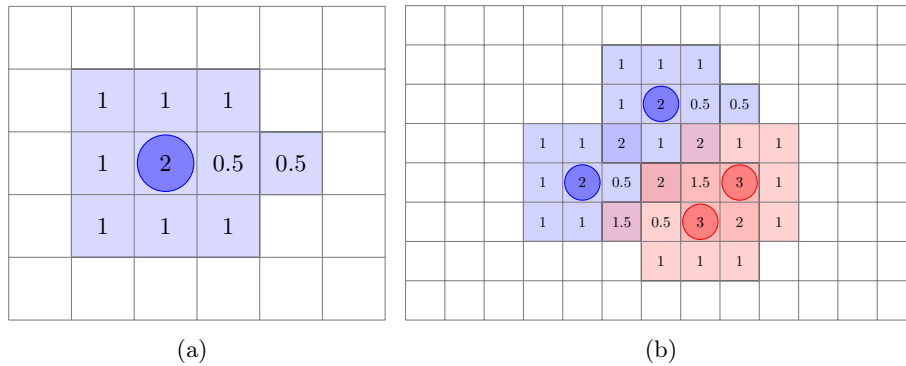


Figura 4.3: (a) Configuración del campo social y (b) intersección de campos sociales.

### 4.1.2. Campo de visión

La mayoría de los peatones caminan viendo todo lo que sucede enfrente de ellos, de esta manera pueden evitar chocar contra obstáculos o personas. El campo de visión refleja la atención que pone el peatón a las cosas que pasan en frente, dándole más importancia a éstas que a las que suceden a su lado o detrás de él. Por otro lado, el alcance de visión del peatón comienza a ser importante a cierta distancia, cuando observa que puede cambiar de dirección para evitar chocar con los obstáculos. Esta característica permite al peatón tomar una decisión, continuar caminando en línea recta o desviarse.

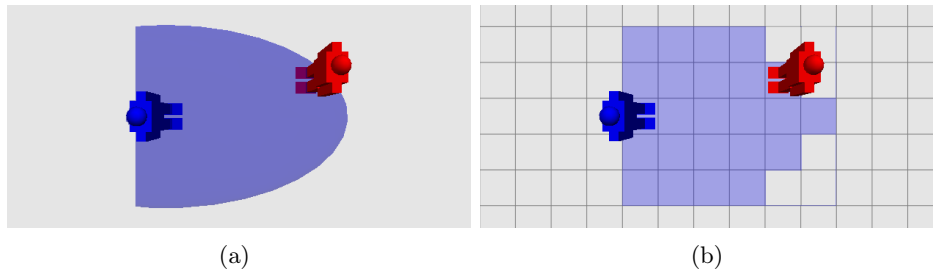


Figura 4.4: Campos de visión: (a) continuo y (b) discreto.

La figura ?? exhibe dos tipos de campos de visión: continuo y discreto. El alcance de visión de un peatón será igual a su velocidad individual  $v^i$ , así tendrá una visión del recorrido en un paso de tiempo. El peatón de la figura ?? tiene una velocidad de 5 celdas por paso de tiempo, mientras que el peatón de la figura ?? de 3 celdas por paso de tiempo. El peatón con velocidad alta debe de tener una visión más profunda para evitar chocar con obstáculos o personas. Si la velocidad es menor la visión se reduce ya que no es necesario tener una gran profundidad de visión para evitar obstáculos en su camino.

### 4.1.3. Parámetro social ponderado

Al igual que los parámetros dinámicos que introduce Hao Yue et al. [?] para controlar el movimiento de los peatones, nosotros proponemos el llamado parámetro social ponderado. La idea de este parámetro es ayudar en la decisión del peatón a elegir la línea con mayor concentración de peatones que llevan la misma dirección, de esta manera es más probable que se de una formación de líneas dinámicas. Para ello, se definen dos parámetros intermedios: el peso y el valor social por línea. El peso, denotado por  $w_k^i$ , se define por la relación siguiente:

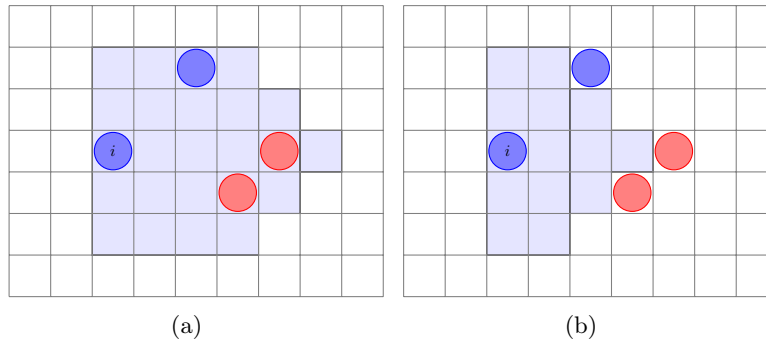


Figura 4.5: Diferentes alcances del campo visual: el peatón en (a) lleva una velocidad  $v^i = 5$ , mientras que el peatón en (b) su velocidad es  $v^i = 3$ .

$$w_k^i = 1 + \sum_{j=1}^{np_k} \frac{d_j}{l_k \times v^i} \quad (4.1)$$

donde  $np_k$  es el número de personas que hay en la  $k$  línea de caminata,  $v^i$  es el número de celdas de alcance del campo de visión,  $l$  toma el valor de 1 para las líneas de caminata  $k - 1, k$  y  $k + 1$  y el valor de 2 para las líneas  $k - 2, k + 2$ ;  $d_j$  es igual a 1 si el peatón  $j$  camina con dirección opuesta al peatón  $i$ , y  $-1$  para aquellos que tienen la misma dirección. La composición de las líneas de caminata se presenta en la figura ???. Las celdas con igual tonalidad pertenecen a una misma línea de caminata.

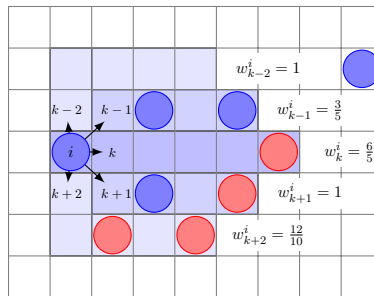


Figura 4.6: Cálculo de los pesos y los valores sociales por línea: (a) para la línea  $k$ , (b) para las cinco líneas.

El valor social  $s_k$  de la línea  $k$  se obtiene sumando el campo social de cada celda que forma esa línea de caminata, ver figura ???. El parámetro social ponderado  $w_k^i$  se calcula multiplicando el peso de cada línea por su respectivo valor social, ver el ejemplo de la figura ??.

#### 4.1.4. No cruce de trayectorias

La restricción de “no cruce de trayectorias” sirve para modelos que utilizan velocidades máximas mayores a 1 por celda en cada paso de tiempo, figura ???. El peatón avanza siempre y cuando otra persona no haya bloqueado su trayectoria. Consideramos esta restricción por funcionar adecuadamente en el modelo de Klüpfel [?], la forma en que se calcula la trayectoria es la que propone el autor, ver sección 3.4.3. Con esta condición el modelo elimina el conflicto cuando dos peatones desean moverse a una misma celda, ya que el primero que se actualiza es el que tiene la ventaja de tomar la celda.



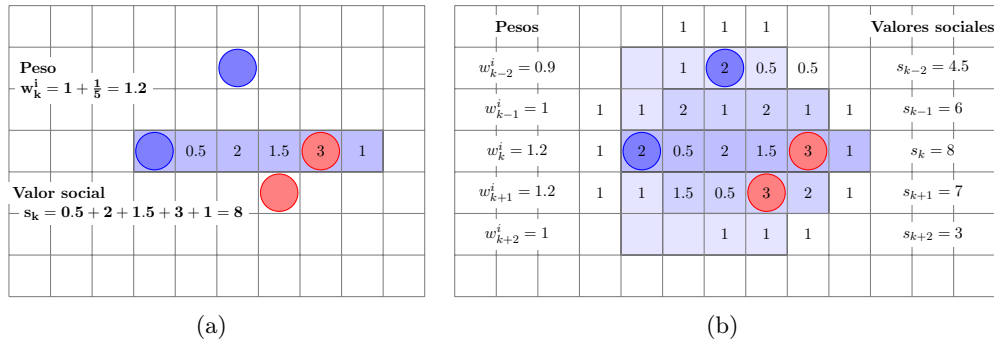


Figura 4.7: Cálculo de los pesos y los valores sociales para la línea  $k$ , (b) para todas la líneas.

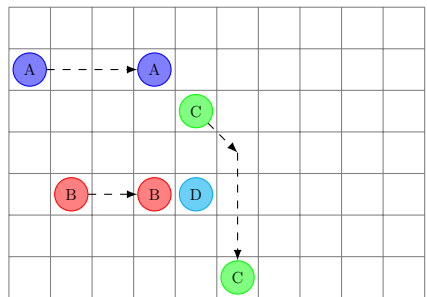


Figura 4.8: El modelo no permite el cruce de trayectorias entre peatones.

#### 4.1.5. Algoritmo de movimiento

El movimiento se lleva a cabo en dos partes. Primero se calcula la trayectoria completa de cada peatón y posteriormente se actualizan sus posiciones a la nueva celda. Cada paso de tiempo esta seccionado en subpasos de tiempo. El algoritmo de movimiento se aplica a cada peatón en cada subpaso de tiempo hasta completar el paso de tiempo total o la velocidad individual. La actualización de las posiciones de los peatones se hace en paralelo, es decir, en cada paso de tiempo, el peatón se mueven a la última celda de la trayectoria calculada. La figura ?? muestra el diagrama del flujo de movimiento del peatón cuya secuencia de pasos se presenta a continuación.

##### I. Cálculo de trayectoria por peatón

1. Calcular el alcance de visión del peatón.
2. Calcular el parámetro social ponderado sobre la  $k$  línea de caminata del peatón.
3. Si el valor social ponderado es menor a uno, entonces verificar si la celda se encuentra desocupada e ir a paso 6. Sino ir al paso 4.
4. Calcular el valor de cada parámetro social ponderado, para las líneas de caminata adyacentes  $k - 1, k + 1, k + 2, k - 2$ , tal como se muestra en la figura ??.
5. Comparar y elegir la celda desocupada con menor valor social ponderado. Es decir,  $z = \{z \in [k, k - 1, k + 1, k + 2, k - 2] \mid vp_z = \min[vp_k, vp_{k-1}, vp_{k+1}, vp_{k+2}, vp_{k-2}], Cell(z) = 0\}$
6. Mover y actualizar el campo social del peatón, ir al paso 7.
7. En caso de no encontrar una celda desocupada entonces el peatón permanece en la misma celda.
8. Termina

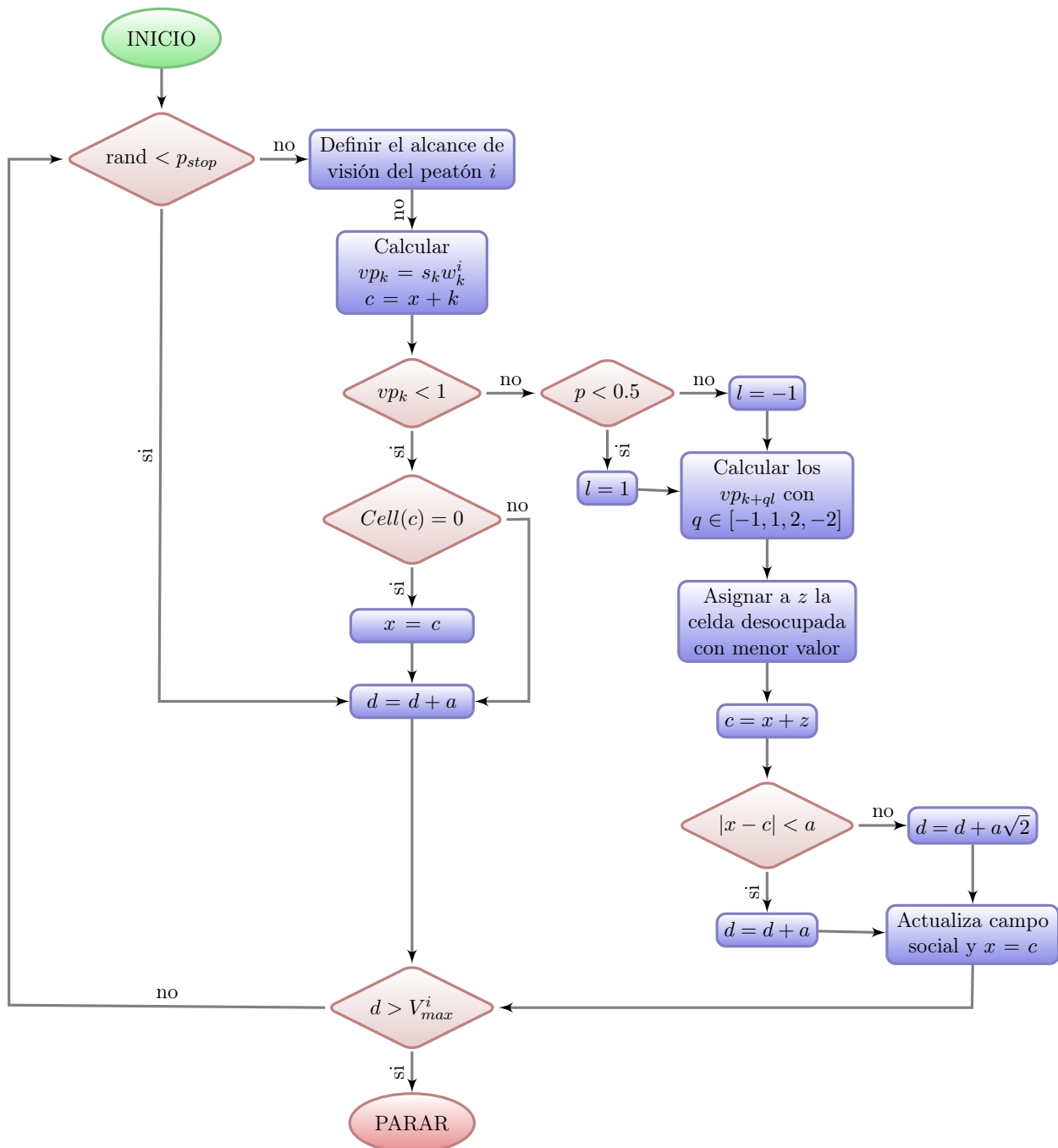


Figura 4.9: Diagrama de flujo para el modelo de campo social. Donde  $x$  representa la celda actual del peatón y  $vp$  el valor social ponderado,  $a$  es el tamaño de la celda,  $k$  es la línea (carril) actual del peatón,  $d$  mide la distancia recorrida por el peatón, en el caso cuando  $x = c$   $d$  aumenta  $a$  denotando el transcurso de tiempo.

## 4.2. Simulaciones numéricas

Consideramos dos tipos de fronteras en un pasillo de tamaño  $10 \text{ m} \times 50 \text{ m}$  para flujo bidireccional: (a) abiertas y (b) periódica. Los peatones de azul se mueven de izquierda a derecha y los de color rojo

en sentido opuesto. Los parámetros utilizados en las simulaciones son: el tamaño de celda  $a = 0.4$  m y el paso de tiempo  $\Delta t = 0.2$  s.

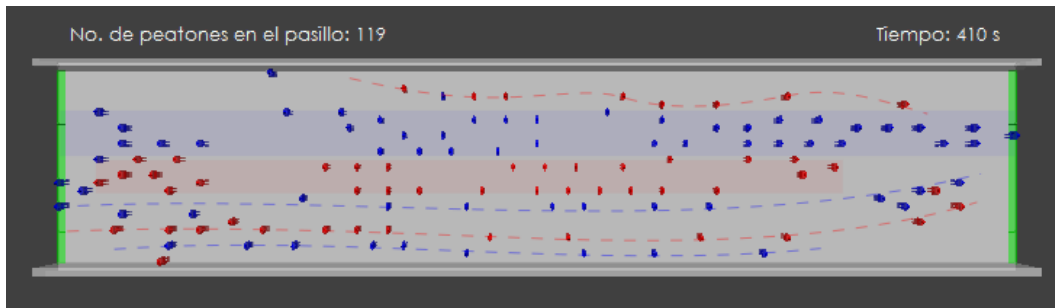


Figura 4.10: Simulación numérica de un flujo bidireccional en un pasillo con condiciones de frontera abiertas. Los peatones azules caminan de izquierda a derecha y rojos en sentido contrario. Se observa la formación de líneas dinámicas.

Entre 0 y 5 peatones se introducen aleatoriamente por cada frontera lateral a cada 2 pasos de tiempo. Cada velocidad individual se distribuye normalmente con media 1.34 m/s y varianza 0.26 m/s.

La figura ?? muestra la simulación numérica con condiciones de frontera abiertas. Observamos que la formación de líneas dinámicas se hace presente, lo que nos indica que el modelo es capaz de reproducir este tipo de patrón.

Para el caso de fronteras periódicas, figura ??, observamos que la mayoría de las simulaciones presentan una auto-organización mediante la formación de cuatro corrientes de flujo, dos de ellas más grandes que las otras. Una vez que estas corrientes se han formado prevalecen todo el tiempo sin tener una gran variación en su forma.

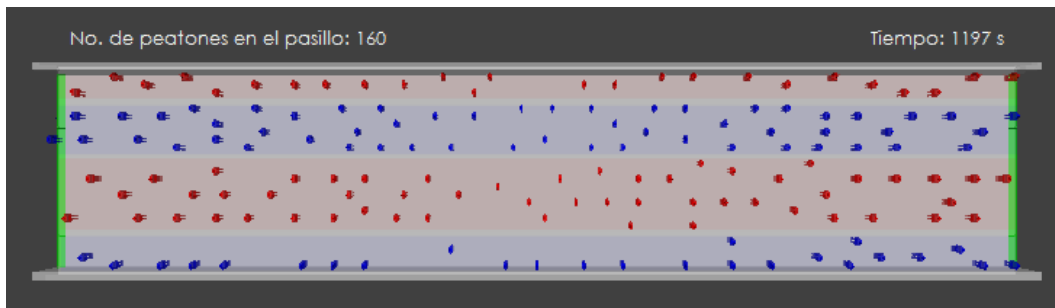


Figura 4.11: Simulación numérica con condiciones de frontera cerrada. Las líneas dinámicas se agrupan formando 4 corrientes de flujo.

#### 4.2.1. Diagrama fundamental

Ahora consideramos un pasillo de 3.6 m de ancho y 40 m de largo con fronteras de tipo periódica. La simulación es un flujo unidireccional, donde los peatones caminan de izquierda a derecha. Al inicio se distribuyeron las personas sobre todo el pasillo de manera uniforme. El número de personas se mantiene fijo.

Basándonos en los trabajos de Parisi [37], para cada densidad la simulación se realiza durante 500 segundos, con 60 segundos para estabilizar el flujo, una vez que el flujo se estabilizó se comenzó a registrar

los datos de velocidad y densidad sobre una región de tamaño 3.6 m por 5.6 m.

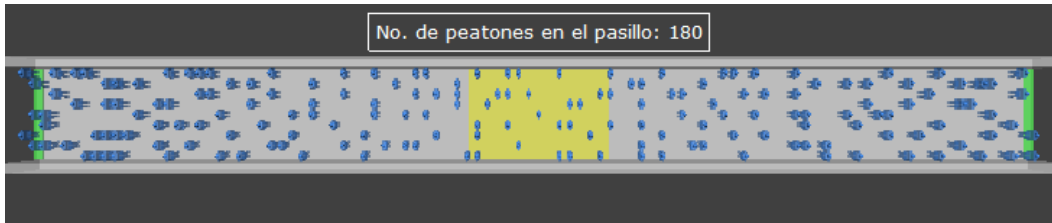


Figura 4.12: Simulación para el diagrama fundamental. La densidad de personas es de  $0.32 \text{ P/m}^2$

La figura ?? muestra: (a) la curva empírica de Weidmann (continua), (b) los datos calculados con el modelo de campo social (puntos rojos) y (c) los datos calculados con el modelo de fuerza social modificado (asteriscos azules) [?]. Cada punto en la gráfica es el promedio de 440 datos registrados en la simulación. Claramente, podemos ver que los datos del modelo de campo social son cercanos a la curva de Weidmann inclusive para densidades altas (mayores que  $4.8 \text{ P/m}^2$ ) con un bajo costo computacional.

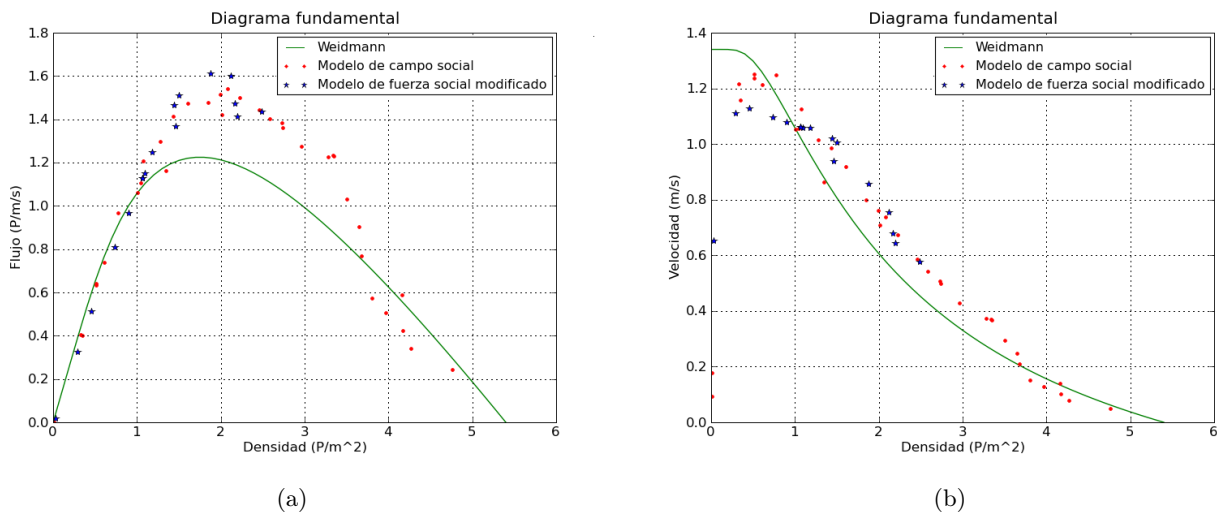


Figura 4.13: Diagrama: (a) flujo-densidad y (b) velocidad-densidad.

### 4.3. Conclusiones

Hemos propuesto un modelo de AC que incorpora tres características principales: un campo social, un campo de visión y un parámetro social ponderado. Este conjunto de características hace que cada peatón sea capaz de darse cuenta de las personas que se encuentran delante del el, distinguiendo de aquellas que van en misma dirección y las que no. Mostrando una tendencia a seguir a personas con misma dirección formando líneas dinámicas. Por lo tanto el modelo es capaz de reproducir este patrón tal como se observa en los datos empíricos publicados en otros lados. Nuestro modelo de campo social ajusta muy bien la curva empírica de Weidmann para densidades altas tales como  $4.8 \text{ P/m}^2$ .

Las ventajas de modelar con un autómata celular son su simplicidad y su velocidad computacional, debido a que en lugar de resolver ecuaciones diferenciales como lo hace el modelo de fuerza social, solo siguen reglas de movimiento fáciles de ejecutar. Esto es conveniente para las simulaciones a tiempo real.

## Conclusiones y perspectivas

En la presente investigación hemos estudiado dos tipos de modelos microscópicos: fuerza social y autómatas celulares. Nos enfocamos en situaciones normales y no de pánico.

El modelo de fuerza social se representa por un sistema ecuaciones diferenciales ordinarias acopladas, una ecuación por cada peatón en el sistema. Estudiamos cada una de las fuerzas que componen el modelo, así como las modificaciones que han hecho algunos autores. Construimos diferentes escenarios con el fin de estudiar el comportamiento de los peatones y ver si el modelo es capaz de reproducir patrones que se han observado en el tráfico peatonal. En la mayoría de estos escenarios la auto-organización dentro de pasillos se presenta mediante la formación de líneas dinámicas. En el caso de cuellos de botella se reprodujeron las oscilaciones que son características en este tipo de estructuras: puertas, escaleras, ampliación y reducción de pasillos, etc. En cruce de pasillos, se presentaron la formación de líneas diagonales.

Por otra parte, los modelos con autómatas celulares se caracterizan por ser discretos en tiempo y espacio. Cada peatón es modelado como un entidad que ocupa una celda en la cuadrícula. La dinámica de los peatones se rigen por reglas de movimiento que actualizan la posición del peatón en casa paso de tiempo. Hemos estudiamos diversos modelos con autómatas celular, de los cuales sólo dos implementamos para su simulación: el modelo de Klüpfel y el modelo de dos procesos. Sin embargo ninguno de ellos logró reproducir los patrones del tráfico peatonal. Debido a la experiencia que se obtuvo al estudiar el modelo de fuerza social y los modelos de autómatas celular, propusimos un modelo capaz de reproducir dichos patrones, principalmente la formación de líneas dinámicas respetando la distancia social de los transeúntes.

El modelo de fuerza social tiene la gran desventaja de utilizar demasiado tiempo de CPU cuando en la región de estudio intervienen cientos de personas, ya que se debe encontrar la solución de un sistema de cientos de ecuaciones por cada paso de tiempo. Por otro lado, los modelos de autómatas celulares son más rápidos y hasta cierto punto prácticos a costa del almacenamiento en memoria, debido a que cada celda puede requerir de varios parámetros y en escenarios de gran tamaño se necesitan miles de celdas.

El trabajo futuro, para el modelo de fuerza social es hacer la programación en paralelo para aumentar la rapidez en los cálculos, reducir grados de libertad para controlar los movimientos, hallar un tipo de especificación que tome en cuenta las velocidades de los peatones con menores operaciones. Debemos incluir un algoritmo para la búsqueda de trayectoria. Manejar otra forma de área de respeto para la interacción entre peatones. Finalmente hallar una forma de reducir operaciones en el cálculo de la dirección deseada y fuerza de repulsión.

Para el modelo de campo social, el trabajo futuro es hacer un análisis de los valores del campo, aunque los valores que se han manejado hasta ahora han mostrado buenos resultados, creemos que es necesario hacer el análisis correspondiente. También debemos ampliar el radio del campo para que los peatones se percaten con mayor anticipación de la presencia de otras personas. El tamaño de la celda que hemos manejado hasta ahora ha sido de 40 cm, medida promedio que utilizan varios autores. Sin embargo, es necesario realizar una discretización del espacio con celdas más pequeñas, con ello habrá mayor diversidad de peatones en el sistema. Además el campo social tendrá mejor forma aproximándose a una elipse, de esta manera las interacciones serán mucho más suaves, donde los peatones se alejan gradualmente sin tener un salto brusco de una a otra celda. Para implementar el modelo en estructuras más complejas es necesario aplicar un algoritmo adecuado para el cálculo de dirección de cada peatón. Simular el modelo en otras estructuras más grandes. Introducir un parámetro que perturbe el comportamiento en situaciones de bloqueo. Hacer un análisis para obtener tiempos de estabilidad del flujo, donde las velocidades no tienen variaciones drásticas y se mantienen oscilando en un cierto valor. Estudiar el comportamiento de los peatones con altas densidades. Finalmente, darle un enfoque al modelo para situaciones de emergencia, observando aquellos escenarios relacionados con multitudes en pánico.

# Anexos





# Programas

En este Anexo se incluyen cuatro programas. Para el modelo de fuerza social se presentan los programas de flujo bi-direccional y un cruce de puerta. Asimismo, para el modelo de autómatas celulares se presentan los de flujo uni-direccional y bi-direccional.

Todos los códigos fuente se programaron sobre la plataforma de *Python*, el cual es un programa orientado a objetos. De esta manera nos permitirá tratar a los peatones con mayor facilidad manipulando características (atributos) y métodos (funciones) de cada peatón. *Visual Python* es el paquete principal para la visualización en 3D. Además se incluyen paquetes tales como *Numpy* que nos ayudan a operar con matrices y *random* que genera números aleatorios de diferentes distribuciones.

## A.1. Modelo de fuerza social

Comenzamos por describir las funciones que componen el programa. La función *Construye(x)*, toma como argumento un vector  $x$ , esta función construye el escenario, es decir, las paredes y el piso que conforman el sistema donde los peatones estarán. El argumento  $x$  es un vector con puntos, los cuales son extremos de paredes. Cabe mencionar que los puntos son tomados en el primer cuadrante (vectores positivos) y posteriormente se reflejarán a los otros tres cuadrantes. Por lo que sólo tenemos que introducir las coordenadas de una parte del escenario y éste se completa después.

```
#-----Construye las paredes-----# def
Construye(x):
    W=[box(pos=(0,0,-0.6),size=(2*x[0,0],2*abs(x[0,1]),0.2),color=mycolor, opacity=opacidad)]
    for i in range(x.shape[0]-1):
        a = x[i,1];l=-1
        if x[i+1,1] > x[i,1]:
            a = x[i+1,1];l=1
        if abs(x[i+1,0]-x[i,0])!=0 and abs(x[i+1,1]-x[i,1])!=0:
            W.append(box(pos=(x[i,0]+(abs(x[i+1,0]-x[i,0]))/2.,a-(abs(x[i+1,1]-x[i,1]))/2.,1.25),
                size=(sqrt(pow(x[i+1,0]-x[i,0],2)+ pow(x[i+1,1]-x[i,1],2)),0.2,2.5),
                axis=(abs(x[i+1,0]-x[i,0])*l,abs(x[i+1,1]-x[i,1]),0),
                color=mycolor,
                opacity=opacidad))

            W.append(box(pos=(-x[i,0] - (abs(x[i+1,0] - x[i,0]))/2., a - (abs(x[i+1,1]-x[i,1]))/2., 1.25),
                size=(sqrt(pow(x[i+1,0]-x[i,0],2) + pow(x[i+1,1]-x[i,1],2)),0.2,2.5),
                axis=(abs(x[i+1,0]-x[i,0])*l,-abs(x[i+1,1]-x[i,1]),0),
                color=mycolor,
```

```

opacity=opacidad))

W.append(box(pos=(x[i,0] + (abs(x[i+1,0] - x[i,0])))/2. , -a +(abs(x[i+1,1]-x[i,1]))/2. , 1.25),
size=(sqrt(pow(x[i+1,0]-x[i,0],2) + pow(x[i+1,1]-x[i,1],2)),0.2,2.5),
axis=(-abs(x[i+1,0]-x[i,0])*1,abs(x[i+1,1]-x[i,1]),0),
color=mycolor,
opacity=opacidad))

W.append(box(pos=(-x[i,0] - (abs(x[i+1,0] - x[i,0])))/2. , -a +(abs(x[i+1,1]-x[i,1]))/2. , 1.25),
size=(sqrt(pow(x[i+1,0]-x[i,0],2) + pow(x[i+1,1]-x[i,1],2)),0.2,2.5),
axis=(-abs(x[i+1,0]-x[i,0])*1,-abs(x[i+1,1]-x[i,1]),0),
color=mycolor,
opacity=opacidad))

k=0
for j in linspace(x[i,0],x[i+1,0],30):
    W.append(box(pos=(j,0,0),
size=(abs(x[i+1,0]-x[i,0])/30.,2*x[i,1] + 1*k*2*abs(x[i+1,1]-x[i,1])/30.,0.2),
color=mycolor,
opacity=opacidad))

    W.append(box(pos=(-j,0,0),
size=(abs(x[i+1,0]-x[i,0])/30.,2*x[i,1] + 1*k*2*abs(x[i+1,1]-x[i,1])/30.,0.2),
color=mycolor,
opacity=opacidad))
    k+=1
else:
    W.append(box(pos=(x[i,0] + (abs(x[i+1,0] - x[i,0])))/2. , a - (abs(x[i+1,1]-x[i,1]))/2. , 1.25),
size=(abs(x[i,0]-x[i+1,0]) + abs(x[i,1]-x[i+1,1]),0.2,2.5),
axis=(abs(x[i+1,0]-x[i,0]),abs(x[i+1,1]-x[i,1]),0),
color=color.blue,
opacity=opacidad))

    W.append(box(pos=(-x[i,0] - (abs(x[i+1,0] - x[i,0])))/2. , a - (abs(x[i+1,1]-x[i,1]))/2. , 1.25),
size=(abs(x[i,0]-x[i+1,0]) + abs(x[i,1]-x[i+1,1]),0.2,2.5),
axis=(abs(x[i+1,0]-x[i,0]),-abs(x[i+1,1]-x[i,1]),0),
color=color.blue,
opacity=opacidad))

    W.append(box(pos=(x[i,0] + (abs(x[i+1,0] - x[i,0])))/2. , -a+(abs(x[i+1,1]-x[i,1]))/2. , 1.25),
size=(abs(x[i,0]-x[i+1,0]) + abs(x[i,1]-x[i+1,1]),0.2,2.5),
axis=(-abs(x[i+1,0]-x[i,0]),abs(x[i+1,1]-x[i,1]),0),
color=color.blue,
opacity=opacidad))

    W.append(box(pos=(-x[i,0] - (abs(x[i+1,0] - x[i,0])))/2. , -a+(abs(x[i+1,1]-x[i,1]))/2. , 1.25),
size=(abs(x[i,0]-x[i+1,0]) + abs(x[i,1]-x[i+1,1]),0.2,2.5),
axis=(-abs(x[i+1,0]-x[i,0]),-abs(x[i+1,1]-x[i,1]),0),
color=color.blue,
opacity=opacidad))

    W.append(box(pos=(x[i,0] + (abs(x[i+1,0] - x[i,0])))/2. ,0,0),
size=(abs(x[i,0]-x[i+1,0]),2*abs(x[i+1,1]),0.2),
axis=(-abs(x[i+1,0]-x[i,0]),-abs(x[i+1,1]-x[i,1]),0),
color=mycolor,
opacity=opacidad))

    W.append(box(pos=(-x[i,0] - (abs(x[i+1,0] - x[i,0])))/2. ,0,0),
size=(abs(x[i,0]-x[i+1,0]),2*abs(x[i+1,1]),0.2),
axis=(-abs(x[i+1,0]-x[i,0]),-abs(x[i+1,1]-x[i,1]),0),
color=mycolor,
opacity=opacidad))

return W

```

La función *Figura* construye la representación de un humano a escala, esta función toma como argumento *Color*, *Escala* y *Trans* que es el color, el tamaño y el nivel de transparencia que llevará el humano.

```

def Figura(Color,Escala,Trans):
    Humano=frame()
    #Zapatos
    #box(frame=Humano, pos=(-0.28409*Escala,0.41*Escala,0.133/2*Escala),
    size=(1.13636*Escala,0.6363*Escala,0.2*Escala),
    color=Color,
    opacity=Trans)

    #box(frame=Humano, pos=(-0.28409*Escala,-0.41*Escala,0.133/2*Escala),
    size=(1.13636*Escala,0.6363*Escala,0.2*Escala),
    color=Color,
    opacity=Trans)

    #Piernas
    box(frame=Humano, pos=(0,0.4*Escala,2*Escala),
    size=(0.55*Escala,0.55*Escala,4*Escala),
    color=Color,
    opacity=Trans)

    box(frame=Humano, pos=(0,-0.4*Escala,2*Escala),
    size=(0.55*Escala,0.55*Escala,4*Escala),
    color=Color,
    opacity=Trans)

    #Torso
    box(frame=Humano, pos=(0,0,5.37*Escala),
    size=(0.9*Escala,1.25*Escala,2.7*Escala),
    color=Color,
    opacity=Trans)

    #Brazos
    box(frame=Humano, pos=(0,0.83*Escala,5.1*Escala),
    size=(0.4*Escala,0.4*Escala,3*Escala),
    color=Color,
    opacity=Trans)

    box(frame=Humano, pos=(0,-0.83*Escala,5.1*Escala),
    size=(0.4*Escala,0.4*Escala,3*Escala),
    color=Color,
    opacity=Trans)

    #Cabeza
    sphere(frame=Humano,pos=(0,0,7.3*Escala),
    radius=0.4*Escala,
    color=Color)

    return Humano

```

Seguimos con la función *Persona*, en esta mandamos a llamar la función *Figura*, a la cual se le asignan las características individuales que utilizaremos durante la simulación. Para comenzar a familiarizarse con el tipo programación pondremos como ejemplo, la expresión  $Ind[i].v$  es  $v$  la velocidad de  $Ind$  el individuo  $i$ , el cual es un objeto que manda a llamar su atributo  $v$  de velocidad. La función *Persona* toma como argumentos *entrada*, *sale* y  $n$  es el lado por donde entrará, la posición será calculada dentro de la misma función, *sale* es por donde saldrá del sistema y finalmente  $n$  el número de peatones que construirá la función.

```

def Persona(entra,sale,n):
    Ind=[] # Lista de individuos
    col=[(0.3,0.6,1),color.red,color.white,color.cyan] # Lista de colores
    for i in range(n):
        Ind.append(Figura(col[entra],radio,1)) # Agregar un individuo
        Ind[i].radius=radio # Radio de la persona
        Ind[i].di=array([0,0]) # Dirección
        Ind[i].v=array([.000001,.000001]) # Velocidad

```

```

Ind[i].r=mag(Ind[i].v)          # Rapidez
Ind[i].a=array([0,0])         # Aceleración
Ind[i].rd=gauss(1.34,0.26)    # Rapidez deseada
Ind[i].rmax=1.3*Ind[i].rd     # Rapidez maxima
while 1:      # Asigna la posición inicial
    y = gauss(0,Ancho[entra]/3)
    if abs(y) < Ancho[entra]/2.:
        if EyS[entra,1] == 0:
            Ind[i].pos=vector(EyS[entra] + y*array([0,1]))
            Ind[i].Pos=array(EyS[entra] + y*array([0,1]))
        else:
            Ind[i].pos=vector(EyS[entra] + y*array([1,0]))
            Ind[i].Pos=array(EyS[entra] + y*array([1,0]))

        break
    if entra > 2:
        Ind[i].axis=(0,1,0)

    Ind[i].PosK = [Metas[entra], 100*(EyS[sale]/mag(EyS[sale]))]
    i+=1
return Ind

```

Todas las funciones anteriores se incluyen dentro de un archivo llamado *MisFunciones.py* y son llamadas en el programa principal con la instrucción *from MisFunciones import \**. Además estas funciones son utilizadas tanto para el flujo bidireccional y cruce de una puerta.

### A.1.1. Flujo bidireccional

La primera parte del programa se compone de los datos iniciales tanto para la construcción del escenario como de los parámetros del modelo de fuerza social. Además de los arreglos que se utilizan para el cálculo de fuerzas. La simulación se inicia introduciendo un número aleatorio de personas, entre 0 y 6 de ellas a cada 2 segundos. En adelante se utiliza el modelo de fuerza social, comenzando por el *Cálculo de fuerzas de repulsión entre peatones*. La fuerza de repulsión que toma en cuenta el peatón *i* son todas aquellas que se generan entre él y los peatones que se encuentre a 3 metros de distancia de él. Cabe mencionar que el término  $(\text{lamda1} + (1-\text{lamda1}) * (1 + \text{dot}(P[i].v, -dij/b) )/2))$  es el peso de *f* la fuerza de repulsión y da como resultado una nueva fuerza.

```

#-----Paqueteria-----# from visual import * from numpy
import fmod from MisFunciones import * from random import choice,
gauss, randint

#Estas son las propiedades de la pantalla
myscene=display(title='Simulación de flujo
peatonal',x=0,y=0,width=950,height=300,auntocenter=1,autoscale=1)

#-----Piso paredes y etiquetas-----# opacidad=0.5
# Transparencia de las paredes mycolor=color.gray(0.5) # Color
de las paredes EyS=array([[-25,0],[25,0]]) # Entradas y salidas de
los peatones Metas =[[25,0],[-25,0]] # Metas intermedias
Ancho=[10,10] # Ancho de entradas y salidas

x=array([[0,5],[25,5]]) # Vector para la construcción de paredes

largo=51 # Largo del pasillo ancho=10
# Ancho del pasillo tamX=x.size/2 # Numero de vectores
en x W=Construye(x) # Función constructora del
escenario #-----Datos iniciales-----# P=[]
# Inicia lista de Peatones lamda1=0.9 # Parámetro Vo=3.
# Fuerza de Iteración entre peatones desv=0.5 # Rango
de Iteración entre peatones Uo=10. # Fuerza de

```

```

Iteración entre muros R=0.2          # Rango de Iteración
entre muros ta=0.5                  # Parámetro de relajación dt=0.2
# Intervalo de tiempo Fim = array([0,0]) # Vector de fuerzas a
muros a=zeros((2,300))             # Vector de aceleración k=0
# Contador de iteraciones p=0        # Contador de
peatones radio=0.2                  # Radio de los peatones Puertas=[0,1]
# Número de puertas de salida y entrada al sistema
#-----#
l2=label(pos=(0,5.5,7),text='No. de peatones en el pasillo:
'+str(p))

#####-----COMIENZA LA SIMULACIÓN-----#####
while 1:
#-----Entrada de peatones-----#
if not fmod(k,2/dt):
for Entra in [0,1]:
n=randint(0,6)                      # Número aleatorio de personas
Salida=Entra
while Salida == Entra:
Salida=choice(Puertas)
P=append(P,Persona(Entra, Salida, n)) # Agregando n personas a la lista
p+=n
NP=range(p)
l2.text='No. de peatones en el pasillo: '+str(p)
Fij=zeros((2,p))

#-----Calculo de fuerzas de repulsion entre peatones-----#
for i in NP:
for j in range(i+1,p):
dij=P[i].Pos - P[j].Pos
b = mag(dij)
if b < 3: #Se consideran aquellas personas dentro de un radio de 3 unidades que afectan al peatón i
f = (Vo)*(exp((2*radio - b)/desv))*(dij/b)
Fij[:,i] = Fij[:,i] + (lamda1 + (1-lamda1)*(1 + dot(P[i].v,-dij/b))/2))*f
Fij[:,j] = Fij[:,j] - (lamda1 + (1-lamda1)*(1 + dot(P[j].v,-dij/b))/2))*f

#-----Calculo de dirección-----#
P[i].di = (P[i].PosK[0] - P[i].Pos)/mag(P[i].PosK[0] - P[i].Pos)

#-----Fuerza de movimiento-----#
Fi = (P[i].rd*P[i].di - P[i].v)/ta

#-----Fuerza de repulsion a muros-----#
Fim = (Uo/R)*exp(-(5. - abs(P[i].pos.y) - radio)/R)*array([0,-copysign(1,P[i].pos.y)])

#-----Calculo de la aceleracion-----#
a[:,i]=Fi + Fim + Fij[:,i]
#-----Actualización de posiciones y eliminación de peatones-----#
i=0
while i<p:
P[i].v = P[i].v + a[:,i]*dt
P[i].r = mag(P[i].v)
if P[i].r > P[i].rmax:
P[i].v = (P[i].v/P[i].r)*P[i].rmax
P[i].r = P[i].rmax

P[i].pos = P[i].Pos + P[i].v*dt
P[i].Pos = array([P[i].pos.x, P[i].pos.y])

if mag(P[i].Pos - P[i].PosK[0]) < 2:
P[i].PosK = delete(P[i].PosK,0,axis=0)

if abs(P[i].pos.x) >= largo/2 or abs(P[i].pos.y) >= largo/2:
P[i].visible = 0 #Borra de pantalla al peatón
P=delete(P,i) #Elimina del conjunto de peatones
del NP[p-1] #Elimina el ultimo termino del arreglo
p-=1 #Elimina del contador
i-=1

```

```

        12.text='No. de peatones en el pasillo: '+str(p)
        i+=1
        k+=1
#####-----FIN LA SIMULACION-----#####

```

Una vez que el programa obtiene todas las fuerzas de repulsión de cada peatón, comienza el cálculo de la dirección deseada, donde  $P[i].PosK[0]$  es el siguiente objetivo del peatón  $i$  y conforme el peatón alcanza sus objetivos se van eliminando del arreglo  $PosK$ , es por eso que siempre se pide la primera entrada del arreglo. Ahora bien, en el cálculo de repulsión a muros utiliza la función *copysign*; como su nombre lo indica copia el signo, en este caso copia el signo de la posición en  $y$  del peatón  $i$ , esto con el fin de saber a que pared esta más cerca el peatón y debido a que el escenario esta centrado en el origen es más fácil distinguir esto. Una vez que se terminó el cálculo de todas la fuerzas, son sumadas y almacenadas en un arreglo llamado  $a$  que representa la aceleración de cada peatón. Posteriormente actualizamos las velocidad, rapidez y posición de cada peatón. Si el peatón completó su recorrido y llegó a su salida, este se elimina del sistema.

### A.1.2. Cruce de una puerta

La diferencia entre este programa y el anterior son la metas intermedias y la fuerza de repulsión a muros. La meta intermedia para todos los peatones se localiza en el centro de la puerta, por lo tanto la primera meta de los peatones es llegar a la puerta, después a su salida. En esta simulación no se implemento un flujo continuo de personas debido a que se reprodujeron varios escenarios de algunos artículos de Helbing. Por tanto contamos con 70 personas en el sistema que irán saliendo conforme lleguen a su destino. Por otro lado el cálculo de la fuerza a muros considera dos puntos, el punto más cercano a la pared más proxima del pasillo y el punto de la pared de la puerta más cercano al peatón. Este último término se calcula siempre y cuando este a la altura de la pared de la puerta (cuando no tenga de frente la puerta).

```

#-----Paqueteria-----# from visual import * from numpy
import fmod from MisFunciones import * from random import choice,
gauss, randint

#Estas son las propiedades de la pantalla
myscene=display(title='Simulación de flujo
peatonal',x=0,y=0,width=950,height=300,auntocenter=1,autoscale=1)

#-----Piso paredes y etiquetas-----# opacidad=0.5
# Transparencia de las paredes mycolor=color.gray(0.5) #
Color de las paredes EyS=array([[ -12.5,0],[12.5,0]]) # Entradas y
salidas de los peatones Metas=[[0,0],[0,0]] # Metas
intermedias Ancho=[10,10] # Ancho de entradas y
salidas

x=array([[0,5],[12.5,5]]) # Vector para la construcción de
paredes

largo=2*max(x[:,0]) # Largo del pasillo
ancho=2*min(x[:,1]) # Ancho del pasillo tamX=x.size/2
# Número de vectores en x

W=Construye(x) # Funcion de construcción del
escenario #-----Construccion paredes de la puerta-----#
W.append(box(pos=(0,3,2.5-0.6), size=(0.3,4,5),
color=color.gray(0.5), opacity=opacidad))
W.append(box(pos=(0,-3,2.5-0.6), size=(0.3,4,5),
color=color.gray(0.5), opacity=opacidad))

#-----Datos iniciales-----# P=[]

```

```

# Inicia lista de peatones lamda1=0.8          # Parámetro para el
# peso de la fuerza de repulsion entre peatones lamda2=1          #
# Parámetro para el peso de la fuerza de repulsion a muros Vo=2.5
# Fuerza de Iteración entre peatones desv=0.5          # Rango de
Iteración entre peatones Uo=10.          # Fuerza de Iteración
entre los muros R=0.2          # Rango de Iteración a muros
ta=0.5          # Relajación dt=0.2          # Intervalo de
Tiempo a=zeros((2,200))          # Matriz de aceleración
Fij=zeros((2,200))          # Matriz de fuerzas de repulsion entre peatones
Fim = array([0,0])          # Matriz de fuerza de repulsion a muros k=0
# Contador de iteraciones p=70          # Número de peatones
en el sistema radio=0.4          # Radio de los peatones
NP=range(p)          # Arreglo del número de peatones
P=append(P,Persona(0,1,35))          # Lista inicial de la creación del 35
con entrada por la izquierda P=append(P,Persona(1,0,35))          # Lista
inicial de la creación del 35 con entrada por la derecha
Puertas=[0,1]          # Número de puertas de salida y entrada al
sistema #-----#
l2=label(pos=(0,6,7),text='No. de peatones en el pasillo: '+str(p))

#####-----COMIENZA LA SIMULACIÓN-----#####
while 1:
    Fij=zeros((2,p))
    #-----Calculo de fuerzas entre peatones-----#
    for i in NP:
        for j in range(i+1,p):
            dij=P[i].Pos - P[j].Pos
            b = mag(dij)
            if b < 3:
                f = (Vo)*(exp((P[i].radius + P[j].radius - b)/desv))*(dij/b)
                Fij[:,i] = Fij[:,i] + (lamda1 + (1-lamda1)*(1 + dot(P[i].di,-dij/b))/2))*f
                Fij[:,j] = Fij[:,j] - (lamda1 + (1-lamda1)*(1 + dot(P[j].di,-dij/b))/2))*f
            #-----Calculo de direccion-----#
            P[i].di = (P[i].PosK[0] - P[i].Pos)/mag(P[i].PosK[0] - P[i].Pos)
            #-----Fuerza de movimiento-----#
            Fi = (P[i].rd*P[i].di - P[i].v)/ta
            #-----Fuerza de repulsion a muros-----#
            Fim = [0,0]

            if abs(P[i].pos.y)<1-0.4: # Calculo de la fuerza de la pared del pasillo, siempre y cuando tenga de frente la puerta
                Fim = (Uo/R)*exp(-(ancho/2 - abs(P[i].pos.y))/R)*array([0,-(P[i].pos.y+1)/abs(P[i].pos.y+1)])
            else:
                Fim = (Uo/R)*exp(-(ancho/2 - abs(P[i].pos.y))/R)*array([0,-(P[i].pos.y+1)/abs(P[i].pos.y+1)])\
                    + (Uo/R)*exp(-(abs(P[i].pos.x)-0.2)/R)*array([copysign(1,P[i].pos.x), 0])
            #-----Calculo de la aceleracion-----#
            a[:,i]=Fi + Fim + Fij[:,i]
            Fij[:,i]=[0,0]

        #-----Actualizacion de posiciones y eliminacion de peatones-----#
        i=0
    while i<p:
        P[i].v = P[i].v + a[:,i]*dt
        P[i].r = mag(P[i].v)
        if P[i].r > P[i].rmax:
            P[i].v = (P[i].v/P[i].r)*P[i].rmax
            P[i].r = P[i].rmax

        P[i].pos = P[i].Pos + P[i].v*dt
        P[i].Pos = array([P[i].pos.x, P[i].pos.y])

        if mag(P[i].Pos - P[i].PosK[0]) < 0.7:
            P[i].PosK = delete(P[i].PosK,0,axis=0)

        if abs(P[i].pos.x) >= largo/2 or abs(P[i].pos.y) >= largo/2:
            P[i].visible = 0 #Borra de pantalla
            P=delete(P,i)          #Elimina del conjunto de peatones
            del NP[p-1]          #Elimina el ultimo del arreglo para el for
            p-=1          #Elimina del contador
            i-=1

```

```

        l2.text='No. de peatones en el pasillo: '+str(p)
    i+=1
    k+=1
#####-----FIN LA SIMULACION-----#####

```

## A.2. Modelo de autómatas celulares

Los paquetes que se incluyen son: *visual*, *time*, *numpy*, *visual controls*, *random* y *MisFunciones*. Del paquete *time* importamos una función *sleep* que sirve para detener la ejecución por unos segundos. El paquete *visual controls* nos proporciona objetos tales como botones, barra de desplazamiento y switches, estas herramientas la utilizamos para controlar la velocidad de la simulación. Los paquetes como *visual*, *numpy* y *MisFunciones* ya los hemos mencionado anteriormente.

La función *Persona* para autómatas celulares, crea a las personas llamando la función *Figura* que hemos mencionado anteriormente; a esta figura le asignamos características como, dirección, velocidad, trayectoria, distancia recorrida y posición inicial.

```

def Persona(n,a,trans,entra,Vmax,AC,LC,EyS):
    Aux = [1,LC-2]      # Arreglo auxiliar para la posicion inicial en la trayectoria
    I = []             # Lista de individuos
    Color = [(0.3,0.6,1), color.red, color.green,color.magenta,
             color.orange, color.white, color.yellow,color.black,color.blue]

    i=0
    Entra=zeros(n)
    while i<n:         # Calcula la celda para la posición inicial
        y=int(gauss(12,4))
        aux=0
        for index in range(i):
            if y == Entra[index]:
                aux=1
                break
        if aux == 0 and 0<y<25:
            Entra[i]=y
            i+=1

    for i in range(n):
        I.append(Figura(Color[entra],a,trans))      # Lista de individuos
        I[i].Vmax = gauss(1.34,0.26)              # Velocidad maxima
        I[i].pos = (EyS[entra],(Entra[i]+1)*a,0)  # Posición Inicial del peaton
        I[i].T = zeros(Vmax+1)                   # Arreglos de Trayectoria del individuo en un paso de tiempo
        I[i].T[0] = (Entra[i]+1)*LC + Aux[entra] # Primera posición en la trayectoria
        I[i].d = entra+1                          # Dirección
        I[i].t = 0                                # Distancia de la trayectoria recorrida
    return I

```

Al comienzo se introducen los datos y parámetros iniciales, se construye el escenario y se asignan a las celdas tanto las direcciones como sus estados. Los peatones se incorporan al pasillo cada 3 segundos. Después se utiliza el algoritmo de Klüpfel para flujo peatonal. Cabe mencionar que primero se calcula la trayectoria de cada peatón, sin que se crucen ninguna de estas, para después actualizar la posición. El algoritmo comienza preguntando si en ese paso de tiempo el peatón se detendrá, con una probabilidad *PStop* de que se detenga, así mismo pregunta si el peatón ha llegado a su velocidad máxima. En caso de alguna de estas sea afirmativa el peatón actualiza su trayectoria a la misma posición. En caso contrario el peatón trata de avanzar hacia adelante ( $V(r)$ ), si no puede pregunta por las otras opciones de desviarse ya sea a 45 o 90 grados de la celda objetivo. Una vez que haya encontrado la primera celda disponible, este introduce una nueva trayectoria en su recorrido, asigna el estado de ocupado a la celda y suma la distancia recorrida para el paso de tiempo actual.



```

#-----Paqueteria-----# from visual import * from time
import sleep from numpy import mod from MisFunciones import * from
visual.controls import slider, controls from random import random,
randint

Display=display(title='Simulacion con Autómatas
Celular',x=0,y=0,width=950,height=300,center=(24.8, 5,
1.15),autoscale=1) #####-----DATOS INICIALES Ancho=10      #Ancho
del pasillo en unidades Largo=50      #Largo del pasillo en
unidades a=0.4      #Tamaño de la celda AC=Ancho/a      #Ancho
del pasillo en celdas LC=Largo/a      #Largo del pasillo en celdas
Vmax=5      #Maxima Velocidad de los peatones PStop=0.1
#Probabilidad de detenerse Psway=0.03      #Probabilidad de
desviarse EyS=[0.4,49.2] #####-----INICIALIZA Y DEFINE LAS CELDAS
OCUPADAS S=zeros(((AC + 2)*LC + 2,3))      #Inicializa matriz de
Estados (States) S[0:LC,0] = 2      #Asigna las paredes
correspondientes S[(AC + 2)*LC - LC:(AC + 2)*LC + 2,0] = 2
#Introduce las direcciones que llevan cada celda S[:,1] = 2 S[:,2] =
6 #####-----INICIALIZACIÓN DE PARÁMETROS P=[]
#Lista de personas NP=[]      #Orden de las personas en la
lista k=0      #Contador p=0      #Para una
densidad constante Tsub1 = range(1,Vmax+1) Tsub2 = range(Vmax+1)
#####-----DEFINE LA LISTA CON LOS VECTORES DE DIRECCIÓN Y LOS
NUMEROS
Vec=array([[0,a],[a,a],[a,0],[a,-a],[0,-a],[-a,-a],[-a,0],[-a,a]])
Num=array([LC, LC+1, 1, -LC+1, -LC, -LC-1, -1, LC-1, LC])

#####-----DISEÑO GRÁFICO DEL ENTORNO
piso=box(pos=(24.8,5.2,0),size=(50,10,0.2), color=color.gray(0.5),
opacity=0.5) w1=box(pos=(24.8,0,1.25),
size=(50,a,2.5),color=color.blue, opacity=0.5)
w2=box(pos=(24.8,10.4,1.25), size=(50,a,2.5),color=color.blue,
opacity=0.5)

# Cuadrícula el pasillo for i in range(int(AC+3)):
    curve(pos=[(0-a/2,i*a-a/2,-0.2),(50-a/2,i*a-a/2,-0.2)])
for i in range(int(LC+1)):
    curve(pos=[(i*a-a/2,-a/2,-0.2),(i*a-a/2,10.6,-0.2)])
# Imprime en la pantalla de simulación el número de peatones dentro
del pasillo. l2=label(pos=(25,11.8,0),text='No. de peatones en el
pasillo: '+str(p),font='century gothic')

# Se visualiza el control de velocidad de la simulación
Control=controls(x=1000, y=0, width=270, height=100,
title='Velocidad de Animación',range=60) s1=slider(pos=(-50,0),
width=10, length=100, axis=(1,0,0))

#-----COMIENZO DE LA SIMULACIÓN-----# while 1:
    if not mod(k,3):      # Entrada de peatones al pasillo cada 3 segundos
        for entra in [0]:
            n = randint(0,10)      # Número aleatoria de personas que entran
            P = append(P,Persona(n,a/2.,1,entra,Vmax,AC,LC,EyS))
            p+=n
            NP=range(p)
            l2.text='No. de peatones en el pasillo: '+str(p)
        auxT=[]
        Control.interact()      # Momento en que el programa interactuar con el control de velocidad de simulación
        sleep(s1.value/200.)      # Velocidad de simulación
        for tsub in Tsub1:      # Subpasos
            for i in NP:      # Peadones
                C = P[i].T[tsub-1]
                if P[i].t < P[i].Vmax and random()>PStop and 0 < (C%LC)*a < Largo - a/2.:
                    #Elegir lado para desviarse en caso de hacerlo.
                    sign = 1
                    if random() < 0.5:
                        sign = -1

            #-----V(r), Avanza hacia la celda objetivo-----#

```

```

if S[C + Num[S[C, P[i].d]],0] <= 1:          # Verifica si la celda esta disponible
    P[i].T[tsub] = C + Num[S[C, P[i].d]]    # Calcula la nueva celda d su trayectoria
    S[P[i].T[tsub],0] = 2                  # Asigna el estado de ocupado a la nueva celda
    P[i].t += a                            # Suma la distancia de recorrido

#-----V(r)+45°, Se desvía a la derecha de la celda objetivo-----#
elif S[C + Num[S[C, P[i].d] + sign*1],0] <= 1:
    P[i].T[tsub] = C + Num[S[C, P[i].d] + sign*1]
    S[P[i].T[tsub],0] = 2
    P[i].t += a*1.4142

#-----V(r)-45°, Se desvía a la izquierda de la celda objetivo-----#
elif S[C + Num[S[C, P[i].d] - sign*1],0] == 0:
    P[i].T[tsub] = C + Num[S[C, P[i].d] - sign*1]
    S[P[i].T[tsub],0] = 2
    P[i].t += a*1.4142

#-----V(r)-90°, Se desvía 90 grados a la derecha de la celda objetivo-----#
elif S[C + Num[S[C, P[i].d] - sign*2],0] == 0:
    P[i].T[tsub] = C + Num[S[C, P[i].d] - sign*2]
    S[P[i].T[tsub],0] = 2
    P[i].t += a

#-----V(r)+90°, Se desvía 90 grados a la izquierda de la celda objetivo-----#
elif S[C + Num[S[C, P[i].d] + sign*2],0] == 0:
    P[i].T[tsub] = C + Num[S[C, P[i].d] + sign*2]
    S[P[i].T[tsub],0] = 2
    P[i].t += a

else:
    P[i].T[tsub] = P[i].T[tsub-1]
    P[i].t += a

#-----Mantiene la Distancia Social entre peatones-----#
aux = P[i].T[tsub] + Num[S[P[i].T[tsub], P[i].d]]
if S[aux, 0] <= 1: #Asegurando de no desbloquear celdas por error
    S[aux, 0] += 1
    if S[aux,0] == 2:
        auxT.append(aux)

#elif random() > PSway:
#    Aplicar el desvio a 45°

else:
    P[i].T[tsub] = P[i].T[tsub-1]
    P[i].t += 1

#-----Asigna el estado libre a las celdas ocupadas por la trayectorias anteriores-----#
for i in NP:
    for tsub in Tsub2:
        S[P[i].T[tsub],0] = 0
S[auxT,0] = 0
i = 0
#-----Actualiza las posiciones-----#
while i < p:
    C = P[i].T[Vmax] #Guarda posicion actual

    if 0 < (C/LC)*a < Largo - a/2.:          # Pregunta si aun esta dentro del pasillo
        P[i].pos = ((C/LC)*a, int(C/LC)*a, 0) # Calcula la coordenada de la posición actual
        P[i].T[0] = C                        # Asigna la celda a la trayectoria
        S[C,0] = 2                           # Asigna el estado ocupado a la celda actual
        P[i].t = 0                            # Se reinicia el contador de distancia recorrida
        i += 1
    else:                                     #Borra peatones
        S[C,0] = 0
        P[i].visible = 0
        P = delete(P,i)
        del NP[p-1]
        p -= 1

```

```
k+=1
```

```
#####
```

Una vez que se hayan actualizado las posiciones, el programa pregunta si el peatón ha llegado a su destino, en caso de que lo haya hecho éste sale del sistema.

### A.2.1. Flujo bidireccional

La diferencia de este programa con el anterior es que cada celda maneja dos direcciones: 2 y 6, los que van a la derecha y los que van a la izquierda, respectivamente. Esta dirección se asigna al peatón con la instrucción  $P[i].d$ . Exceptuando esta diferencia, todo lo demás es igual al programa anterior.

```
from visual import * from time import sleep from numpy import mod
from Humano import * from visual.controls import slider, controls
from random import random, randint

Display=display(title='Simulacion con Autómatas
Celular',x=0,y=0,width=950,height=300,center=(24.8, 5,
1.15),autoscale=1) #####-----DATOS Ancho=10 # Ancho del
pasillo en unidades Largo=50 # Largo del pasillo en unidades
a=0.4 # Tamaño de la celda AC=Ancho/a # Ancho del
pasillo en celdas LC=Largo/a # Largo del pasillo en celdas
Vmax=5 # Maxima Velocidad de los peatones PStop=0.1 #
Probabilidad de detenerse Psway=0.03 # Probabilidad de
desviarse EyS=[0.4,49.2] #####-----INICIALIZA Y DEFINE LAS CELDAS
OCUPADAS S=zeros(((AC + 2)*LC + 2,3)) #Inicializa matriz de
Estados (States) # Asigna las paredes correspondientes S[0:LC,0] = 2
S[(AC + 2)*LC - LC:(AC + 2)*LC + 2,0] = 2 # Introduce las
direcciones que llevan cada celda S[:,1] = 2 S[:,2] = 6
#####-----INICIALIZACIÓN DE PARÁMETROS P=[] #Lista
de personas NP=[] #Orden de las personas en la lista
k=0 #Contador p=0 #Para una densidad
constante Tsub1 = range(1,Vmax+1) Tsub2 = range(Vmax+1)
#####-----DEFINA LA LISTA CON LOS VECTORES DE DIRECCION Y LOS
NUMEROS
Vec=array([[0,a],[a,a],[a,0],[a,-a],[0,-a],[-a,-a],[-a,0],[-a,a]])
Num=array([LC, LC+1, 1, -LC+1, -LC, -LC-1, -1, LC-1, LC])

#####-----DISEÑO GRAFICO DEL ENTORNO
piso=box(pos=(24.8,5.2,0),size=(50,10,0.2), color=color.gray(0.5),
opacity=0.5) w1=box(pos=(24.8,0,1.25),
size=(50,a,2.5),color=color.blue, opacity=0.5)
w2=box(pos=(24.8,10.4,1.25), size=(50,a,2.5),color=color.blue,
opacity=0.5)

# Cuadrícula el pasillo for i in range(int(AC+3)):
curve(pos=[(0-a/2,i*a-a/2,-0.2),(50-a/2,i*a-a/2,-0.2)])
for i in range(int(LC+1)):
curve(pos=[(i*a-a/2,-a/2,-0.2),(i*a-a/2,10.6,-0.2)])

# Imprime en pantalla el numero de peatones dentro del pasillo
l2=label(pos=(25,11.8,0),text='No. de peatones en el pasillo:
'+str(p),font='century gothic')

# Se visualiza el control de velocidad de la simulación
Control=controls(x=1000, y=0, width=270, height=100,
title='Velocidad de Animación',range=60) s1=slider(pos=(-50,0),
width=10, length=100, axis=(1,0,0)) #####-----COMIENZO DEL
ALGORITMO-----##### while 1:
if not mod(k,6): # Entrada de peatones al pasillo cada 6 segundos
for entra in [0,1]:
n = randint(0,10)
P = append(P,Persona(n,a/2.,1,entra,Vmax,AC,LC,EyS))
```

```

p+=n
NP=range(p)
l2.text='No. de peatones en el pasillo: '+str(p)
auxT=[]
Control.interact()      # Momento en que el programa interactuar con el control de velocidad de simulación
sleep(s1.value/200.)   # Velocidad de simulación
for tsub in Tsub1:     # Subpasos
    for i in NP:       # Peatones
        C = P[i].T[tsub-1]
        if P[i].t < P[i].Vmax and random()>PStop and 0 < (C%LC)*a < Largo - a/2.:
            #Elegir lado para desviarse en caso de hacerlo.
            sign = 1
            if random() < 0.5:
                sign = -1

            #-----V(r), Avanza hacia la celda objetivo-----#
            if S[C + Num[S[C, P[i].d]],0] <= 1:      # Verifica si la celda esta disponible
                P[i].T[tsub] = C + Num[S[C, P[i].d]] # Calcula la nueva celda d su trayectoria
                S[P[i].T[tsub],0] = 2                # Asigna el estado de ocupado a la nueva celda
                P[i].t += a                           # Suma la distancia de recorrido

            #-----V(r)+45°, Se desvía a la derecha de la celda objetivo-----#
            elif S[C + Num[S[C, P[i].d] + sign*1],0] <= 1:
                P[i].T[tsub] = C + Num[S[C, P[i].d] + sign*1]
                S[P[i].T[tsub],0] = 2
                P[i].t += a*1.4142

            #-----V(r)-45°, Se desvía a la izquierda de la celda objetivo-----#
            elif S[C + Num[S[C, P[i].d] - sign*1],0] == 0:
                P[i].T[tsub] = C + Num[S[C, P[i].d] - sign*1]
                S[P[i].T[tsub],0] = 2
                P[i].t += a*1.4142

            #-----V(r)-90°, Se desvía 90 grados a la derecha de la celda objetivo-----#
            elif S[C + Num[S[C, P[i].d] - sign*2],0] == 0:
                P[i].T[tsub] = C + Num[S[C, P[i].d] - sign*2]
                S[P[i].T[tsub],0] = 2
                P[i].t += a

            #-----V(r)+90°, Se desvía 90 grados a la izquierda de la celda objetivo-----#
            elif S[C + Num[S[C, P[i].d] + sign*2],0] == 0:
                P[i].T[tsub] = C + Num[S[C, P[i].d] + sign*2]
                S[P[i].T[tsub],0] = 2
                P[i].t += a

            else:
                P[i].T[tsub] = P[i].T[tsub-1]
                P[i].t += a

            #-----Mantiene la Distancia Social entre peatones-----#
            aux = P[i].T[tsub] + Num[S[P[i].T[tsub], P[i].d]]
            if S[aux, 0] <= 1: #Asegurando de no desbloquear celdas por error
                S[aux, 0]+=1
                if S[aux,0]==2:
                    auxT.append(aux)

            #elif random()>PSway:
            #Aplicar el desvio a 45°

        else:
            P[i].T[tsub] = P[i].T[tsub-1]
            P[i].t += 1
#-----Asigna el estado libre a las celdas ocupadas por la trayectorias anteriores-----#
for i in NP:
    for tsub in Tsub2:
        S[P[i].T[tsub],0] = 0
S[auxT,0]=0
i=0

```

```
#-----Actualiza las posiciones-----#
while i < p:
    C = P[i].T[Vmax] #Guarda posicion actual

    if 0 < (C/LC)*a < Largo - a/2.:          # Pregunta si aun esta dentro del pasillo
        P[i].pos = ((C/LC)*a,int(C/LC)*a,0) # Calcula la coordenada de la posición actual
        P[i].T[0] = C                       # Asigna la celda a la trayectoria
        S[C,0] = 2                           # Asigna el estado ocupado a la celda actual
        P[i].t = 0                           # Se reinicia el contador de distancia recorrida
        i+=1
    else:                                     #Borra peatones
        S[C,0] = 0
        P[i].visible = 0
        P = delete(P,i)
        del NP[p-1]
        p -= 1
    k+=1
#####
```



## Paquetes de simulación

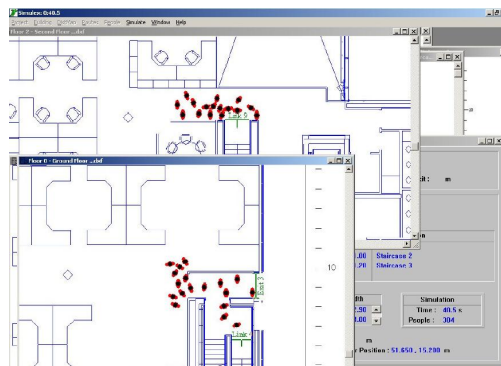
Los paquetes existentes se clasifican en dos grupos: simulación de evacuaciones y simulación de situaciones normales. Los son usados para calcular el tiempo que toman los peatones para salir de un edificio o una área en situaciones de emergencia, por ejemplo gases tóxicos, incendios, congestión de salidas, etc. El paquete de simulación de multitudes para situaciones normales es usado para reproducir el movimiento de miles de personas en condiciones habituales. Estos paquetes comenzaron a usarse para proporcionar un entorno más cómodo y desplazar rápidamente a las multitudes.

A continuación enlistaremos algunos paquetes que hasta el momento se conocen.

### B.1. Simulex

Simulex es una paquete desarrollado en Inglaterra por Dr. Peter Thompson [?, ?]. Simulex puede simular evacuaciones de grandes multitudes en construcciones complejas. El paquete ha sido validado en tres casos de estudio conducidos en Nueva Zelanda [?]. La prueba involucra evacuaciones de tres edificios. La evacuaciones se ha llevado acabo en la vida real y también usando el paquete de simulación.

Se usa para diseñar y editar la geometría del entorno usando un programa de edición visual el cual viene con el paquete. El programa también soporta archivos de AutoCAD. Los agentes pueden tener diferente tamaño, velocidad y movimiento de retraso. Este programa un algoritmo distancia mapeable (The Distance maps algorithm) para mover a los peatones hacia la salida más cercana. El paquete proporciona detalles del flujo y el conteo total de los peatones pasando una salida en especifica. También despliega la información



de la simulación formato de texto.

## B.2. PedGo

PedGo fue desarrollado en Alemania por TraffGo HT GmbH, como resultado de un proyecto de investigación llamado BYPASS por Tim Meyer-König y Dr. Hubert Klüpfel [?, ?]. Este paquete proporciona simulaciones en tiempo real con multitudes mayores a 10,000 personas capaces de reproducirse en un computadora con procesador *Pentium 3* con 500MHz. PedGo deber ser capaz de simular evacuaciones de grandes multitudes usando una computadora más poderosa. Fue usado para simular el peor caso de prueba de un escenario de evacuación con 100,000 peregrinos en el Puente de Jamarat [?].

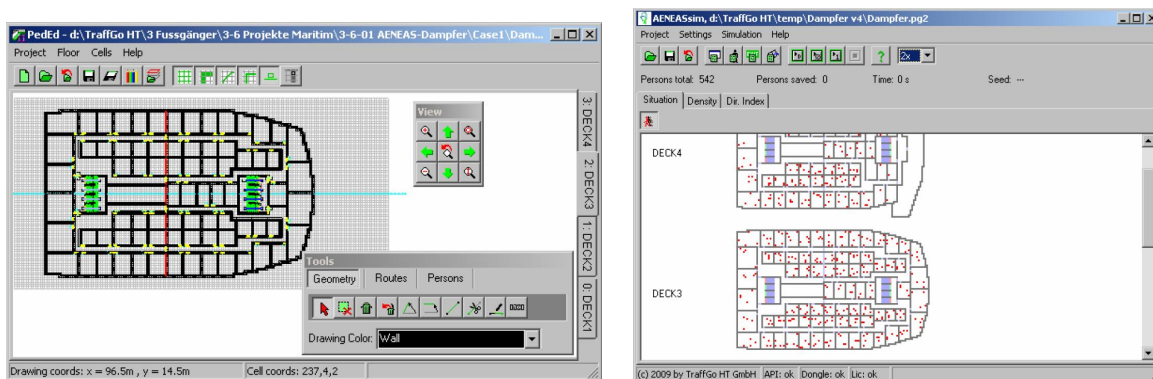


Figura B.1: Diseño de un entorno y simulación en PedGo. Imágenes tomadas del manual de PedGo versión 2.5

PedGo es capaz de editar y modificar los diseños de entornos. Los peatones en el paquete son asignados con diferentes propiedades (velocidad máxima, nivel de paciencia, cambios de dirección, reacción, inercia, probabilidad de reducción de velocidad y agrupación) usando una distribución. El paquete usa un modelo de multi-agentes para simular los movimientos peatonales. La salida visual, tiempo de evacuación, gráfica de densidad, tiempo de viaje y distancia recorrida para cada peatón son la cantidad de salida para el paquete.

## B.3. GridFlow

GridFlow es un producto de David Purser de BRE Ltd, Inglaterra [?, ?, ?, ?]. Este paquete ha sido validado en diferentes tipos de construcciones y son diseñados para simular solamente evacuación de multitudes.

Puede importar diseños en AutoCAD mediante un paquete adicional llamado JOSEFINE. También puede especificar movimientos previos de retraso y distribuciones de velocidad para los peatones. El programa es capaz de simular efectos de smoke, incendios y gases tóxicos en un entorno. GridFlow se basa en un modelo microscópico de autómatas celulares. Es posible exportar datos a Excel, el número de personas en cada sección en cada momento, tiempos de salida y la pantalla de visualización (usando JOSEFINE) son cantidades de salidas para el paquete.



## B.4. ASERI

ASERI fue desarrollado por el Dr. Volker Shneider del Integrierte Sicherheits-Technik GmbH, Alemania. Este paquete puede simular una gran cantidad de agentes en una geometría de entornos muy complejos, [?, ?]. ASERI también simula situaciones de emergencia y evacuaciones.

Este paquete tiene la capacidad de crear importar diseños de AutoCAD y crear sus propios diseños. Los agentes pueden tener características como, talla, velocidad, sexo, edad, aptitud, incapacidad, tolerancia al humo y objetivos individuales. Algunos de los resultados que proporciona el paquete son el promedio de los tiempos de salida de los peatones, embotellamientos y congestiones, trayectorias de cada uno de los individuos y captura de video.

## B.5. Legion

Legion es un paquete desarrollado por Legion Internacional Ltd., UK [?]. Este paquete puede simular miles de peatones en una simple PC y puede usado para evacuaciones y situaciones normales sin emergencias.

Legion proporciona facilidades para importar y modificar diseños de entorno. Es posible especificar entradas, salidas y rutas de movimiento. En Legion, los peatones deciden en base a los objetivos, poseen diferentes niveles de conocimiento, experiencias y objetivos. Ellos también tienen percepción del entorno a su alrededor. Este modelo usa un algoritmo continuo de “ruta de menor esfuerzo” para la simulación de los movimientos. Tiene diferentes tipos de plantillas para los peatones como viajeros y turistas.

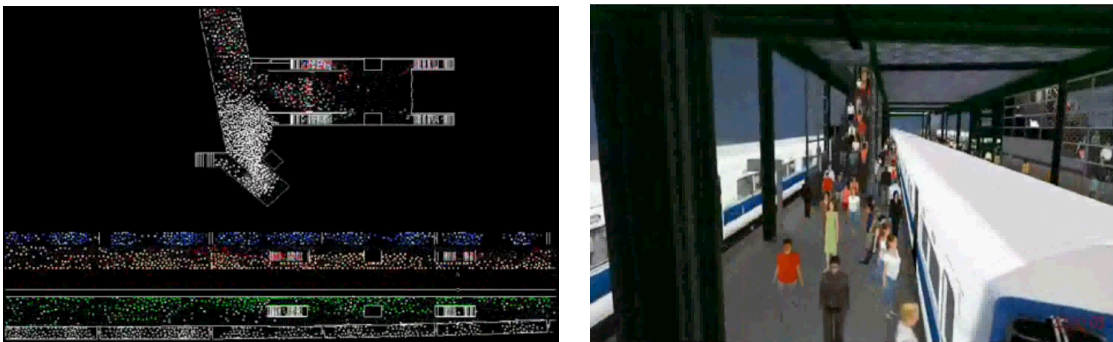


Figura B.2: Simulación y visualización del paquete Legion. Imágenes tomadas de la presentación de Legion.

Este paquete es capaz de proporcionar resultados como, niveles de densidades locales, áreas de congestión, tiempo que toman los individuos para llegar a sus destinos, captura de vídeo e imágenes.

## B.6. EXODUS

EXODUS es un paquete desarrollado por FSEG (Fire Safety Engineering Group) y La Universidad de Greenwich, Inglaterra. El paquete está escrito sobre la plataforma C++ el cual usa una técnica orientada a objetos. Es posible importar archivos en formato “dxf”. Este paquete es utilizado para simular situaciones normales y de evacuación. EXODUS es capaz de simular el comportamiento de una gran número de personas en entornos complejos. Por ejemplo es capaz de simular 1,000 personas en una área de 2,500 m<sup>2</sup> con un procesador de 1.9GHz y una memoria RAM de 1Gb. Y 125,000 personas en una ciudad.



Figura B.3: Simulación de una evacuación en una estación del metro usando buildingEXODUS. Imagen tomada de la página de EXODUS

El programa es capaz de distinguir salidas de emergencia y salidas comunes, también puede asignar salidas de acuerdo al uso

Es posible descargar la versión de prueba [?]. Se puede observar algunos ejemplo de simulaciones en [?].

## B.7. STEPs

STEPs es un paquete de simulación desarrollado por Matt MacDonald de Inglaterra [?]. Este paquete ha sido validado contra el NFPA (Standard for Fixed Guideway Transit and Passenger Rail Systems) en dos casos de estaciones de tren [?]. Para ambos casos STEPs ha sido capaz de reproducir tiempos de evacuación comparados con NFPA. STEPs tiene amplias características para la simulación de evacuaciones y también estudia movimientos de peatones en situaciones normales es decir en situaciones sin emergencia. Este paquete es capaz de simular decenas de miles de peatones en una computadora poderosa. Además, STEPs es capaz de simular efectos de grupo (como familiares). También puede importar archivos de AutoCAD , STEPs tiene su propio paquete de diseño.

STEPs simula el comportamiento y movimientos de peatones individualmente. Es posible determinar detalles de velocidad, talla, niveles de paciencia, trayectoria de los agentes individualmente. La representación gráfica de las densidades locales, niveles de uso, uso de las salidas, densidades en regiones deseadas, tiempo total de evacuaciones, número de peatones en cada region, discapacitados y peatones lentos, registros de trayectorias, movimientos de datos y captura de video en una simulación son los salidas disponibles. Además es posible ver la simulación en 2D y 3D.

## B.8. SimWalk

SimWalk es un paquete de simulación desarrollado por Savannah Simulations AG, Suiza [?]. Este paquete puede ser usado para simular situaciones normales y de evacuación de multitudes. Requiere definir áreas de inicio, puntos intermedios y puntos de salida. SimWalk modela relativamente cantidades grandes de personas y es usado para en diferentes ingenierías e Institutos de investigación.

SimWalk proporciona el paquete SimDraw con el cual se puede importar diseños de AutoCAD y crear o modificar nuevos diseños de entorno. El paquete proporciona pequeñas diferencias entre agentes (velocidad

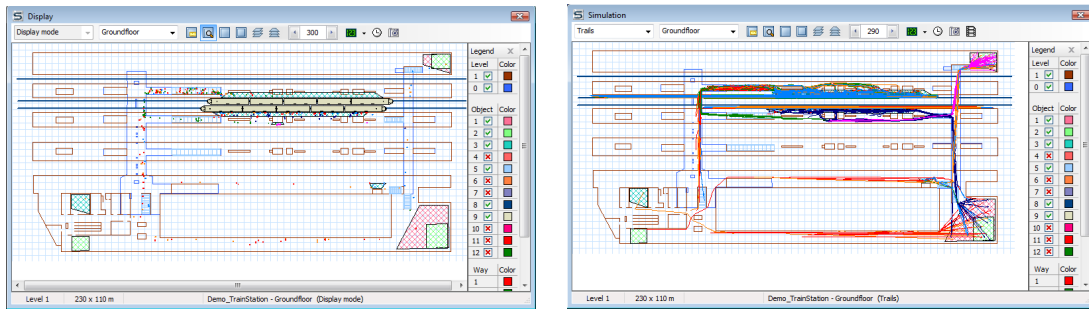


Figura B.4: Simulación de una estación de tren en situación normal. Imagen tomada de la simulación demo que aporta SimWalk

y movimientos de retrasos).

SimWalk se basa en el modelo de Fuerza Social. Este paquete detalla resultados como cargas en cada área, líneas de trayectorias, contadores, información de flujo, nivel de servicio, velocidades y tiempos que toman los peatones a llegar a su destino, grabación y reproducción.

## B.9. NOMAD/SimPed

NOMAD y SimPed son paquetes desarrollados por la Universidad Tecnológica de Delft. NOMAD es un modelo microscópico para simular el comportamiento peatonal. Este consiste de dos modelos mutuamente dependientes. EL modelo “top-level” que predice la actividad de los peatones y la elección de áreas, así como la elección de rutas en lugares públicos. El “down-level” es un modelo usado para describir el comportamiento al caminar. SimPed es un modelo de simulación macroscópico creado para describir y visualizar los patrones de camino de los peatones dentro de las estaciones de transferencia, incluyendo la interacción en los vehículos de transporte públicos. Otra motivación para desarrollar este modelo es para estimar la media y la varianza en los tiempos de transferencias de los pasajeros. SimPed contribuye al análisis del impacto en el diseño de estaciones y horarios de varios sistemas de transportes públicos en flujo peatonal a través de estaciones de transferencia [?].

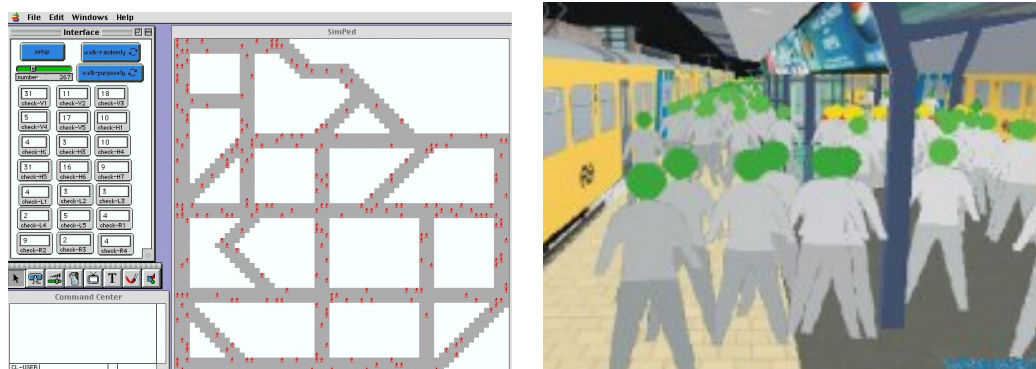


Figura B.5: Entorno de simulación con SimPed. Imagen tomada de la página principal de la Universidad de Delft.



# Bibliografía

- [1] Zarita Zainuddin, Kumatha Thinakaran y M Abu-Sulyman Ibtesam. Simulating the circumambulation of the kaaba using simwalk. *European Journal of Scientific Research*, 38(3):454–464, 2009.
- [2] Siamak Sarmady, Fazila Haron y A. Z. H. Talib. Evaluation of existing software for simulating the crowd at masjid al-haram. *Jurnal Pengurusan JWZH*, 1(1):137–145, 2007.
- [3] Yukun Jiang. Analysis on beijing subway flows during the 29th olympics. *Journal of Transportation Systems Engineering and Information Technology*, 8(6):46–51, 2008. <http://www.sciencedirect.com/science/article/B8H0W-4VDRVVO-4/2/771665c37a984c27140ae1cad9ea4e75>.
- [4] Kardi Teknomo. *Microscopic Pedestrian Flow Characteristics: Development of an Image Processing Data Collection and Simulation Model*. Tesis, Tohoku University, Japón, marzo 2002.
- [5] Winnie Daamen y Serge P. Hoogendoorn. Experimental research on pedestrian walking behavior. *Transportation research record*, 1828:20–30, 2003.
- [6] Dirk Helbing, Illés J. Farkas, Péter Molnár y Tamás Vicsek. *Pedestrian and evacuation dynamics*, capítulo Simulation of pedestrian crowds in normal and evacuation situations, pp 21–58. Springer Verlag, Berlín, 2002.
- [7] Albert Steiner, Michel Philipp y Alex Schmid. Parameter estimation for a pedestrian simulation model. Ascona, Suiza, 12-14 septiembre 2007. 7th Swiss Transport Research Conference.
- [8] Dirk Helbing y Péter Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51:4282–4286, 1995.
- [9] Victor J. Blue y Jeffrey L. Adler. Cellular automata microsimulation for modeling bi-directional pedestrian walkways. *Transportation Research Part B*, 35(3):293–312, marzo 2001.
- [10] Serge P. Hoogendoorn y Piet H. L. Bovy. Simulation of pedestrian flows by optimal control and differential games. *Optimal Control Applications and Methods*, 24(3):153–172, 2003.
- [11] Humbert Klüpfel. *A cellular automaton model for crowd movement and egress simulation*. Tesis doctoral, University of Duisburg-Essen, Alemania, 2003.
- [12] Edward T. Hall. *The Silent Language*. Garden City, Nueva York, 1959.
- [13] Serge P. Hoogendoorn y Winnie Daamen. *Traffic and Granular Flow '03*, volume 4, capítulo Self-Organization in Pedestrian flow, pp 373–382. Springer Berlin Heidelberg, 1st edition, 2005.

- [14] Peter G. Gipps y B. Marksjo. A micro-simulation model for pedestrian flows. *Mathematics and Computers in Simulation*, 27:95–105, 1985.
- [15] Carsten Burstedde, Kai Klauck, Andreas Schadschneider y Johannes Zittartz. Simulation of pedestrian dynamics using a two-dimensional cellular automaton. *Physica A*, 295:507–525, 2001.
- [16] Hao Yue, Hongzhi Guan, Juan Zhang y Chunfu Shao. Study on bi-direction pedestrian flow using cellular automata simulation. *Physica A*, 389:527–539, 2010.
- [17] Shigeyuki Okazaki y Matsushita Satoshi. A study of simulation model for pedestrian movement with evacuation and queuing. *Proceeding of the International Conference on Engineering for Crowd Safety*, (1):271–280, 1993.
- [18] W. J. Yu, R. Chen, L. Y. Dong y S. Q. Dai. Centrifugal force model for pedestrian dynamics. *Physica Review E*, 72(2):026112–026117, agosto 2005.
- [19] Mohcine Chraïbi, Armin Seyfried, Andreas Schadschneider y Wolfgang Mackens. Quantitative description of pedestrian dynamics with a force-based model. In *WI-IAT '09 Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, volume 3, Washington, D.C. , Estados Unidos, 2009.
- [20] Luis R. Izquierdo, José M. Galán, José I. Santos y Ricardo del Olmo. Modelado de sistemas complejos mediante simulación basada en agentes y mediante dinámica de sistemas. *Revista de Metodología de Ciencias Sociales*, 16:85–112, 2008.
- [21] G. G. Lovas. Modelling and simulation of pedestrian traffic flow. *Transportation research B*, 28:429–443, 1994.
- [22] Christian Gloor, Laurent Maunon y Kai Nagel. A pedestrian simulation for hiking in the alps. Monte Verità, Ascona, Suiza, marzo 2003. Swiss Transport Reseach Conference.
- [23] Christian Gloor, Duncan Cavens, Eckart Lange, Kai Nagel y Schmid Willy. *A Pedestrian Simulation for Very Large Scale Applications*, pp 167–188. Number 23. Institute for Computational Science/ Institute for Spatial and Landscape Planning, diciembre 2003.
- [24] Christian Gloor, Pascal Stucki y Kai Nagel. Hybrid techniques for pedestrian simulations. In Slot Peter M. A., Chopard Bastien y Hoekstra Alfons G., editors, *Cellular Automata*, volume 3305, pp 581–590, Amsterdam, The Netherlads, octubre 2004. 6th Automata Cellular for Research and Industry (ACRI 2004), Springer Berlín.
- [25] Adrien Treuille, Seth Cooper y Zoran Popovi. Continuum crowds. *ACM Transactions on Graphics (TOG)*, 25(3), 2006.
- [26] John M. Usher y Lesley Strawderman. Simulating operational behaviors of pedestrian navigation. *Computers and Industrial Engineering*, 59(4):736–747, 2010.
- [27] Ulrich Weidmann. *Transporttechnik der Fußgänger*. Schriftenreihe des IVT 90, Zürich, Alemania, 1992.
- [28] Lewin. *Field Theory in Social Science*. Harper and Brothers, Nueva York, 1951.
- [29] Dirk Helbing. A fluid-dynamic model for the movement of pedestrians. *Complex Systems*, 6:391–415, 1992.

- [30] A. E. Schefflen y N. Ashcraft. *Human Territories: How We Behave in Space-Time*. Prentice Hall, Englewood Cliffs, New Jersey, 1976.
- [31] Helbing Dirk Johansson, Anders y Pradyumn Kumar Shukla. Specification of a microscopic pedestrian model by evolutionary adjustment to video tracking data. *Advances in Complex Systems*, 10:271–288, 2007.
- [32] Pradyumn Kumar Shukla. Modeling and simulation of pedestrians. Tesis de maestría, 2005.
- [33] Pradyumn Kumar Shukla. On modeling and evolutionary optimization of nonlinearly coupled pedestrian interactions. In Cecilia Di Chio, Stefano Cagnoni, Carlos Cotta, Marc Ebner, Anikó Ekárt, Anna Esparcia-Alcazar, Chi-Keong Goh, Juan Merelo, Ferrante Neri, Mike Preuß, Julian Togelius y Georgios Yannakakis, editors, *Applications of Evolutionary Computation*, volume 6024 of *Lecture Notes in Computer Science*. Springer Berlin.
- [34] Dirk Helbing, Illés J. Farkas y Tamás Vicsek. Simulating dynamical features of escape panic. *Nature*, 407:487–490, 2000.
- [35] Taras I. Lakoba, David J. Kaup y Neal M. Finkelstein. Modifications of the helbing-monár-farkas-vicsek social force model for pedestrian evolution. *Simulation*, 81(5):339–352, 2005.
- [36] Gerald H. Ristow y Hans J. Hermann. Density patterns in two-dimensional hoppers. *Physica Review E*, 50:R5–R8, 1994.
- [37] Daniel R. Parisi, Marcelo Gilman y Herman Moldovan. A modification of the social force model can reproduce experimental data of pedestrian flows in normal conditions. *Physica A*, 388:3600–3608, 2009.
- [38] Jaroslaw Was, Bartłomiej Gudowski y Pawel J. Matuszyk. Social distances model of pedestrian dynamics. In Yacoubi Chopard B. y Bandini S., editors, *Lectures Notes in Computer Science: Proceeding of 6th International Conference on Cellular Automata in Research and Industry (ACRI)*, volume 4173, pp 492–501, Berlín, 2006. Springer Verlag.
- [39] Still G. Keith. *Crowd dynamics*. Tesis doctoral, University of Warwick, Reino Unido, agosto 2000.
- [40] Le Roy F. Henderson. The statistics of crow fluids. *Nature*, (229):381–383, 1971.
- [41] F. P. D. Navin y R. J. Wheeler. Pedestrian flow characteristics. *Traffic Engineering*, 39:30–33, 1969.
- [42] Dirk Helbing y Vicsek T. Optimal self-organization. *New Journal of Physics*, 1:13.1–13.17, 1999.
- [43] Armin Seyfried, Bernhard Steffen, Wolfram Klingsch, Thomas Lippert y Maik Boltes. The fundamental diagram of pedestrian movement revisited-empirical results and modelling. In Andreas Schadschneider, Thorsten Pöschel, Reinhart Kühne, Michael Schreckenberg y Dietrich E. Wolf, editors, *Traffic and Granular Flow'05*, pp 305–314. Springer Berlin Heidelberg, 2007.
- [44] Armin Seyfried, Steffen y Thomas Lippert. Basic of modeling the pedestrian flow. *Physica A*, 368:232–238, 2006.
- [45] J. Von Neumann. *Theory of self-reproducing automata*. University of Illinois Press, Londres, 1966.
- [46] Martin Gardner. Mathematical games: The fantastic combinations of john conway's new solitaire game of life. *Scientific American*, 223(4):120–123, 1970.

- [47] Suchindran Maniccam. Traffic jamming on hexagonal lattice. *Physica A*, (321):653–664, agosto 2003.
- [48] Hao Yue, Herui Hao, Xiaoming Chen y Chunfu Shao. Simulation of pedestrian flow on square lattice based on cellular automata model. *Physica A*, 384:567–588, 2007.
- [49] Katsutoshi Narimatsu, Toshihiko Shiraishi y Shin Morishita. Acquisiting of local neighbor rules in the simulation of pedestrian flow by cellular automata. In Sloot P. M. A, Chopard B. y Hoekstra A. G., editors, *Lectures Notes in Computer Science: Proceeding of 6th International Conference on Cellular Automata in Research and Industry (ACRI)*, volume 3305, pp 211–219, Berlín, 2004. Springer Verlag.
- [50] Gerd Gigerenzer, Peter M. Todd y ABC research group. *Simple Heuristics that make us Smart*. Oxford University Press, Nueva York, 1999.
- [51] Pascal Stucki. Obstacles in pedestrian simulations. Master’s thesis, ETH Zúrich, Zúrich, 2003.
- [52] National Fire Protection Association. *SFPE Handbook of Fire Protection Engineering*, 2nd edition edition, 1995.
- [53] Victor J. Blue y Jeffrey L. Adler. Emergent fundamental pedestrian flows from cellular automata microsimulation. *Transportation Research Board Record*, 1644:29–36, 1998.
- [54] Marcus Rickert, Kai Nagel, Michael Schreckenberg y Andreas Latour. Two-lane traffic simulations using cellular automata. *Physica A*, 231:534–550, 1995.
- [55] Victor J. Blue y Jeffrey L. Adler. Cellular automata microsimulation of bi-directional pedestrian flows. *Journal Transportation Research Board*, 1678:135–141, 1999a. <http://www.ulster.net/~vjblue/CA-TRB99.pdf>.
- [56] Ernest J. McCormick. *Ergonomia. Factores Humanos en Ingeniería y Diseño*, capítulo Antropometría aplicada y espacio de trabajo. Gustavo Gili, Barcelona, 1980.
- [57] Peter Thompson, Hakan Lindström, Per-Anders Ohlsson y Stephen Thompson. Simulex: Analysis and changes for imo compliance. In Edwin R. Galea, editor, *Proceedings of 2nd International Conference: Pedestrian and Evacuation Dynamics*, pp 173–184, Londres, UK, agosto 2003. CMS Press.
- [58] Simulex: software para simulación del tráfico peatonal, última consulta: 15 de junio del 2011. <http://www.iesve.com/Software/VE-Pro/Simulex>.
- [59] P. A. Olsson y M. A. Regan. A comparison between actual and predicted evacuation times. *Safety Science*, 38(2):139–145, Julio 2001.
- [60] Pedgo: software para simulación del tráfico peatonal, última consulta: 15 de junio del 2011. <http://www.traffgo-ht.com/en/pedestrians/products/pedgo/index.html>.
- [61] Hubert Klüpfel. The simulation of crowds at very large events. In Andreas Schadschneider, Reinhart Kühne, Thorsten Pöschel, Michael Schreckenberg y Dietrich E. Wolf, editors, *Traffic and Granular Flow '05*. Springer, 2006.
- [62] Erica Dawn Kuligowski. The evaluation of a a performance based design process for hotel building: The comparation fo two egress models. Tesis de maestría, University of Maryland, 2003.
- [63] Guanquan Chu, Jinhua Sun, Qingsong Wang y Sining Chen. Simulation study on the effect of pre-evacuation time and exit width on evacuation. *Chinese Science Bulletin*, 51:1381–1388, 2006.



- [64] M. Bensilum y David A. Purser. Grid flow: An object-oriented building evacuation model combining pre-movement and movement behaviours for performance-based design. *Fire Safety Science*, 7:941–952, 2003.
- [65] Gridflow: software para simulación del tráfico peatonal, última consulta: 15 de junio del 2011. <http://www.bre.co.uk/page.jsp?id=269>.
- [66] V. Schneider y R. Konnecke, editors. *Simulating Evacuation Processes with ASERI*, 2001.
- [67] Aseri: software para simulación del tráfico peatonal, última consulta: 15 de junio del 2011. <http://www.ist-net.de/>.
- [68] Legion: software de simulación, última consulta: 13 de diciembre de 2009. <http://www.legion.com>.
- [69] Exodus: software para simulación del tráfico peatonal, última consulta: 15 de junio del 2011. [http://fseg.gre.ac.uk/exodus/exodus\\_products.html](http://fseg.gre.ac.uk/exodus/exodus_products.html).
- [70] Animaciones del software para simulación del tráfico peatonal exodus, última consulta: 15 de junio del 2011. [http://fseg.gre.ac.uk/fire/EXODUS\\_animations.asp](http://fseg.gre.ac.uk/fire/EXODUS_animations.asp).
- [71] Steps: software para simulación del tráfico peatonal, última consulta: 15 de junio del 2011. <http://www.mottmac.com/skillsandservices/software/stepsoftware/>.
- [72] *NFPA 130: Standard for Fixed Guideway Transit and Passenger Rail Systems*. National fire protection association, Quincy Massachusetts, 2000 edition, 2000.
- [73] SimWalk: Software de simulación, última consulta: 13 de diciembre de 2009. <http://www.simwalk.com>.
- [74] Simped: software para simulación del tráfico peatonal, última consulta: 15 de junio del 2011. <http://www.tudelft.nl/live/pagina.jsp?id=d95d3a14-3d74-4eb0-b9f6-66f321e98aac&lang=en>.



Casa abierta al tiempo

**UNIVERSIDAD AUTÓNOMA METROPOLITANA**

Fecha : 09/12/2011

Página : 1/1

**CONSTANCIA DE PRESENTACION DE EXAMEN DE GRADO**

La Universidad Autónoma Metropolitana extiende la presente CONSTANCIA DE PRESENTACION DE EXAMEN DE GRADO de MAESTRO EN CIENCIAS (MATEMÁTICAS APLICADAS E INDUSTRIALES) del alumno JORGE DANIEL GONZALEZ AROSTICO, matrícula 208381818, quien cumplió con los 147 créditos correspondientes a las unidades de enseñanza aprendizaje del plan de estudio. Con fecha trece de diciembre del 2011 presentó la DEFENSA de su EXAMEN DE GRADO cuya denominación es:

ESTUDIOS DE MODELOS PARA TRAFICO PEATONAL Y SU APLICACION A PASILLOS EN DIFERENTES ESCENARIOS

Cabe mencionar que la aprobación tiene un valor de 60 créditos y el programa consta de 207 créditos.

El jurado del examen ha tenido a bien otorgarle la calificación de:

Aprobar

**JURADO**

Presidente

DR. JOAQUIN DELGADO FERNANDEZ

Secretaria

DRA. PATRICIA SAAVEDRA BARRERA

Vocal

DRA. MARIA ELENA LARRAGA RAMIREZ

**UNIDAD IZTAPALAPA**

**Coordinación de Sistemas Escolares**

Av. San Rafael Atlixco 186, Col. Vicentina, México, DF, CP 09340 Apdo. Postal 555-320-9000, Tels. 5804-4880 y 5804-4883 Fax: 5804-4876





Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

# ACTA DE EXAMEN DE GRADO

No. 00068

Matricula: 208381818

ESTUDIOS DE MODELOS PARA TRAFICO PEATONAL Y SU APLICACION A PASILLOS EN DIFERENTES ESCENARIOS

En México, D.F., se presentaron a las 10:30 horas del día 13 del mes de diciembre del año 2011 en la Unidad Iztapalapa de la Universidad Autónoma Metropolitana, los suscritos miembros del jurado:

DR. JOAQUIN DELGADO FERNANDEZ  
DRA. MARIA ELENA LARRAGA RAMIREZ  
DRA. PATRICIA SAAVEDRA BARRERA

Bajo la Presidencia del primero y con carácter de Secretaria la última, se reunieron para proceder al Examen de Grado cuya denominación aparece al margen, para la obtención del grado de:

MAESTRO EN CIENCIAS (MATEMÁTICAS APLICADAS E INDUSTRIALES)

DE: JORGE DANIEL GONZALEZ AROSTICO



JORGE DANIEL GONZALEZ AROSTICO  
ALUMNO

y de acuerdo con el artículo 78 fracción III del Reglamento de Estudios Superiores de la Universidad Autónoma Metropolitana, los miembros del jurado resolvieron:

*Aprobar*

Acto continuo, el presidente del jurado comunicó al interesado el resultado de la evaluación y, en caso aprobatorio, le fue tomada la protesta.

REVISÓ

LIC. JULIO CESAR DE LARA ISASSI  
DIRECTOR DE SISTEMAS ESCOLARES

DIRECTOR DE LA DIVISION DE CBI

DR. JOSE ANTONIO DE LOS REYES  
HEREDIA

PRESIDENTE

DR. JOAQUIN DELGADO FERNANDEZ

VOCAL

DRA. MARIA ELENA LARRAGA RAMIREZ

SECRETARIA

DRA. PATRICIA SAAVEDRA BARRERA