

**INSTITUTO TECNOLÓGICO Y DE ESTUDIOS
SUPERIORES DE MONTERREY**



**TECNOLOGICO
DE MONTERREY.®**

**Reconocimiento y Descubrimiento de
Patrones de Comportamiento en Sistemas
Multi-Agente Cooperantes**

Autor:

Huberto Ayanegui Santiago

**Sometido al Programa de Graduados en Informática
y Computación en cumplimiento parcial con los
requerimientos para obtener el grado de:**

Doctorado en Ciencias Computacionales

Asesor:

Dr. Fernando Ramos Quintana

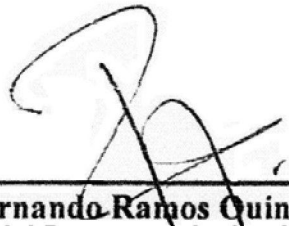
Cuernavaca, Morelos. Agosto del 2008

Reconocimiento y Descubrimiento de Patrones de Comportamiento en Sistemas Multi-Agente Cooperantes

Presentada por:

Huberto Ayanegui Santiago

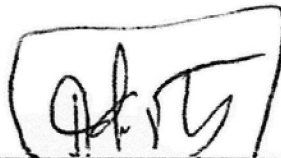
Aprobada por:



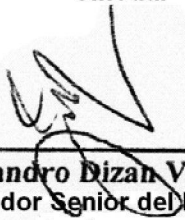
Dr. Fernando Ramos Quintana
Director del Programa de Graduados
en Informática, ITESM Campus Cuernavaca
Asesor



Dr. Enrique David Espinosa Carrillo
Profesor-Investigador ITESM
Campus Ciudad de México
Sinodal



Dr. Víctor Manuel de la Cueva Hernández
Director de la División de Ingeniería y
Arquitectura, ITESM Campus
Ciudad de México
Sinodal



Dr. Alejandro Dizan Vásquez Govea
Investigador Senior del Laboratorio de
Sistemas Autónomos del Instituto
Federal Suizo de Tecnología, Zurich.
Sinodal

Dedicatoria

*A mi amada esposa, Verónica,
quien ha sido mi ayuda idónea en estos años*

Agradecimientos

Deseo reconocer el papel fundamental de mi asesor, el Dr. Fernando Ramos, quien a lo largo de todo este proceso de conceptualización y desarrollo de la tesis estuvo siempre disponible y me brindó muchas horas de su tiempo para finalizar exitosamente este trabajo.

También agradezco a los sinodales: Dr. Enrique Espinosa, Dr. Víctor de la Cueva y el Dr. Alejandro Dizán Vásquez por sus valiosos comentarios y sugerencias.

Mi gratitud a la Universidad Autónoma de Tlaxcala, en particular al M.C. Antonio Durante y al M.C. Román Mendoza por su apoyo durante esta etapa de estudios y al PROMEP por su valioso soporte para recibir esta beca de estudios.

Al Tecnológico de Monterrey, a sus profesores y directivos por su guía y educación de excelencia así como por el apoyo económico que recibí para terminar los últimos semestres del programa.

Agradezco a mis papás, Huberto e Irma, por su confianza y ánimo para creer que no hay nada imposible de terminar.

A mi hermosa familia, mis hijos: Isaac, Valeria y Noemí, porque uds. son mi inspiración y fuerza para ser cada día mejor. Especialmente, a mi esposa amada, porque su amor es mi motor para seguir abriendo brechas.

Finalmente, pero en primer lugar, quiero agradecerle a Dios por ser el autor y consumidor de mi fe. Por ser el principio y el final de todo lo bueno que me ha sucedido, por haber moldeado mi carácter en estos años usando el "pretexto" de los estudios.....porque en Cristo no hay tiempo perdido ni trabajo en vano.

Resumen

Un Sistema Multi-Agente Cooperativo (SMAC) es un grupo de agentes que se comunica, coopera y utiliza mecanismos de coordinación con el fin de resolver algún problema que va más allá de las capacidades para un sólo agente. Existen tres áreas importantes de investigación que continúan abiertas en SMACs: La construcción de modelos relacionados con la comunicación entre agentes, las múltiples interacciones entre agentes y los comportamientos adoptados por los agentes en la ejecución de una tarea [Ferber, 1999]. El presente trabajo está relacionado con el descubrimiento del comportamiento en SMAC's, y específicamente en equipos de soccer robótico. Descubrir el comportamiento de equipos de soccer es un reto computacional debido a las múltiples interacciones que existen entre los jugadores. Esta tarea se vuelve más difícil cuando los agentes se encuentran en dominios dinámicos, competitivos y en tiempo real. En el soccer robótico, dos grupos de agentes se enfrentan adoptando comportamientos principalmente determinados por estrategias y tácticas que cada equipo lleva a cabo para vencer a su oponente. Uno de los retos más difíciles en el modelado de agentes es descubrir comportamientos de alto nivel como estrategias y tácticas, contando sólo con conocimiento de bajo nivel caracterizado principalmente por el movimiento de los agentes y del balón. Descubrir patrones de comportamiento en el soccer robótico, requiere de modelos de representación suficientemente expresivos que sean capaces de representar los aspectos más relevantes del equipo. En este trabajo de tesis se propone un modelo a varios niveles de abstracción en la representación: nivel individual (representación de información de cada agente); nivel relacional (relación entre un conjunto de agentes); y el nivel de formaciones (relación entre todos los agentes que determinan la estructura que soporta la formación del equipo).

El reconocimiento de formaciones es una tarea difícil debido a que los jugadores van cambiando de posición en cada momento del partido, convirtiéndose en un reto el seguimiento de la estructura que determinan los agentes del equipo. Esto dificulta también el descubrimiento de jugadas tácticas ya que éstas ocurren en una parte de la estructura que forma el equipo.

Con base en el modelo de representación a varios niveles de abstracción, se propone un modelo de seguimiento eficiente de formaciones basado en la construcción de estructuras topológicas capaces de reconocer formaciones y cambios de formación a pesar del dinamismo de los agentes y las múltiples interacciones entre estos. Además, estas estructuras topológicas soportan el descubrimiento de patrones de comportamiento táctico, los cuales revelan un nivel más detallado del comportamiento del equipo.

Las contribuciones principales de la tesis son:

1. Un modelo expresivo a varios niveles de representación.

-
2. Reconocimiento, descubrimiento e interpretación de patrones de comportamiento en diferentes niveles de representación.
 3. Un algoritmo robusto para el seguimiento de formaciones en ambientes multi-agentes basado en estructuras topológicas.

Para la validación del modelo propuesto, se llevaron a cabo pruebas que permitieron analizar a los equipos de soccer con el propósito de observar y extraer patrones de comportamiento estratégico basado en formaciones y descubrimiento de patrones de comportamientos táctico. Se presentan resultados de este trabajo tomando en cuenta diferentes equipos que participaron en la categoría de simulación desde el campeonato RoboCup 2000 y hasta los mejores equipos de Robocup 2007, extrayendo sus patrones de comportamiento ofensivos.

Índice general

Índice general	I
Índice de figuras	III
Índice de tablas	V
1. Introducción	1
1.1. Introducción	1
1.2. Planteamiento del problema	4
1.3. Hipótesis	4
1.4. Objetivos	4
1.5. Alcances	5
1.6. Contribuciones	5
2. Trabajos Relacionados	6
2.1. El Soccer Robótico	6
2.1.1. Introducción y categorías	6
2.1.2. Descripción del dominio de simulación de RoboCup	7
2.1.3. El coach	9
2.2. Modelado de Agentes	9
2.3. Modelado de Oponentes	10
2.3.1. Modelado externo	11
2.3.2. Modelado con oponentes ideales	11
2.3.3. Clasificación de oponentes	13
2.4. Reconocimiento de Planes	16
2.5. Reconocimiento de Formaciones y Estrategias	19
2.6. Descubrimiento de Tácticas	23
3. Modelo Propuesto	29
3.1. Introducción	29
3.2. Retos Computacionales	29
3.2.1. Características del ambiente	29
3.2.2. El dinamismo del soccer	29

3.2.3. Múltiples Relaciones	30
3.3. Modelo de Representación	31
3.3.1. Modelo multicapas propuesto	31
3.3.2. Nivel de formaciones	33
3.3.3. Nivel relacional	34
3.3.4. Nivel individual	40
3.4. Modelo de Reconocimiento de Patrones	42
3.4.1. Reconocimiento de formaciones	42
3.4.2. Descubrimiento de jugadas tácticas	47
3.5. Ventajas del Modelo	49
4. Experimentos y Análisis de Resultados	53
4.1. Reconocimiento de Patrones de Formaciones	54
4.1.1. Reconocimiento de Zonas	54
4.1.2. Reconocimiento de Formaciones	61
4.2. Descubrimiento de Patrones Tácticos	64
5. Conclusiones	70
5.1. Contribuciones	71
5.2. Trabajo Futuro	71
5.3. Aplicaciones en otros dominios	71
Referencias	73

Índice de figuras

1.1. Formación 5:3:2	3
2.1. El campo de juego del Soccer Server	8
2.2. Estructura del Soccer Server	9
2.3. Ejemplo de tiro a gol	12
2.4. Ejemplo de un RectGrid	14
2.5. Un escenario de combate aéreo simulado	17
2.6. Variables binarias de entrada a la red neuronal	21
2.7. Rejilla de zonas de marcación del oponente	23
2.8. Ejemplo de un trie	25
2.9. Sistema ISAAC	26
2.10. Cuatro frames mostrando una secuencia de ataque exitosa	26
2.11. Ejemplo de una jugada táctica	27
3.1. Conjunto de relaciones posibles entre jugadores de un equipo	31
3.2. Modelo de representación propuesto	32
3.3. Una formación 5:3:2	33
3.4. Relaciones entre jugadores	36
3.5. Construcción de un grafo planar	37
3.6. Cuatro agentes relacionados en un grafo topológico	39
3.7. Grafos planares para la formación 4:3:3	41
3.8. Configuración del espacio	41
3.9. Discretización de una trayectoria en código Freeman	42
3.10. Ejemplo del recorrido del defensa derecho y delantero izquierdo	43
3.11. Ejemplo del algoritmo de clustering	45
3.12. Proceso para reconocer patrones de formaciones	46
3.13. Rompimiento de la estructura	46
3.14. Dinamismo de una estructura de equipo	47
3.15. Proceso de descubrimiento táctico	47
3.16. Llegadas a la portería del oponente	48
3.17. Proceso de descubrimiento táctico	49
3.18. Grafo dual	50
3.19. Aplicación del grafo dual	50

4.1. Clasificación de Clustering para el equipo TsinghuAeolus.	56
4.2. Cambio del número de jugadores en defensa (TsinghuAeolus).	57
4.3. Jugadores defensas que abandonan temporalmente su zona.	57
4.4. Cambio del número de jugadores en la media (TsinghuAeolus).	58
4.5. Jugadores medios que abandonan temporalmente su zona media	58
4.6. Cambio del número de jugadores en la zona ofensiva (TsinghuAeolus)	58
4.7. Jugadores delanteros que abandonan temporalmente su zona.	58
4.8. Caso TsinghuAeolus. Reconocimiento de la formación general.	60
4.9. Caso SEU. Reconocimiento de la formación general.	60
4.10. Caso Brainstormers. Reconocimiento de la formación general.	60
4.11. Robustez del grafo topológico.	62
4.12. Clustering vs. Grafo topológico.	63
4.13. Comportamiento del equipo FCPortugal.	64
4.14. Clustering vs. Grafo topológico. Punto de vista del experto.	65
4.15. Acercamiento a la portería del oponente.	66
4.16. Secuencia de una jugada táctica.	67
4.17. Trayectorias generalizadas.	67
4.18. Llegadas a la portería del oponente.	69
4.19. Dos formas de trayectorias generalizadas.	69

Índice de tablas

3.1. Lista de formaciones	51
3.2. Reducción del número de relaciones	52
4.1. Lista de equipos a analizar	53
4.2. Agrupaciones encontradas por Clustering en TsinghuAeolus	55
4.3. Períodos de tiempo donde se reconocieron formaciones de equipo	61
4.4. Alineación de zonas en diferentes equipos	62
4.5. Ataques por el ala izquierda	68
4.6. Ataques por el ala derecha	68

Capítulo 1

Introducción

1.1. Introducción

Un Sistema Multi-Agente Cooperativo (SMAC) es un grupo de agentes que se comunica, coopera y utiliza mecanismos de coordinación con el fin de resolver algún problema que va más allá de las capacidades para un sólo agente [Sycara, 1998, Weiss, 1999]. El diseño de SMACs se complica cuando los agentes se encuentran inmersos en un ambiente donde compiten para obtener los mismos recursos.

Modelar sistemas multi-agente para descubrir las estrategias del equipo, es un proceso que permite reconocer la actividad estratégica de los agentes a partir de datos tales como las acciones observadas en el sistema multi-agente [Andraz Bezek and Bratko, 2006]. Una actividad estratégica es una actividad multi-agente que se lleva a cabo durante un tiempo determinado y que exhibe características importantes del equipo de agentes inmersos en algún dominio en particular. Por ejemplo, la manera en que los agentes se comportan bajo una situación defensiva.

Se ha demostrado que los SMACs pueden mejorar su eficiencia y desempeño si cuentan con conocimiento de alto nivel acerca del oponente, tal como estrategias del equipo contrario [Sycara, 1998, Carmel and Markovitch, 1996]. Este conocimiento les permite predecir acciones futuras de los otros agentes en diferentes situaciones. Sin embargo, por lo general, tal conocimiento no está disponible de antemano, sino que es necesario que los agentes sean capaces de modelar el comportamiento de los otros agentes a través de observar las acciones e interacciones que éstos llevan a cabo en diferentes situaciones dadas.

Una aplicación y reto dentro de los SMACs, son los robots que juegan fútbol soccer [Noda and Frank, 1998]. En este dominio, los agentes están inmersos en un universo complejo y dinámico, con un alto grado de competencia entre los participantes y con la necesidad de tomar decisiones en tiempo real. Además, entre los jugadores existen múltiples interacciones que no son fáciles de modelar debido a la naturaleza dinámica del juego. El equipo oponente tiene metas contrarias y compete para obtener los mismos recursos de tal forma que los agentes deben ser capaces de asimilar el

comportamiento del oponente y acomodar su propio comportamiento para enfrentarlo. Estas condiciones del dominio dificultan en gran medida la construcción de modelos capaces de descubrir comportamientos de equipos en partidos de soccer robótico [Lattner et al., 2005, Nair et al., 2004].

Durante los primeros años del soccer robótico, el desarrollo de equipos de soccer no contempló la tarea de modelar al equipo oponente. En su lugar, la interacción con el oponente estaba limitada a un nivel individual-reactivo, es decir, un agente con el balón solamente consideraba la influencia de los oponentes cuando éstos se acercaban demasiado a él y entonces existía un riesgo de perder el balón. Uno de los primeros trabajos en el soccer robótico que tuvo gran éxito al tomar en cuenta el comportamiento del equipo oponente, fue propuesto por Peter Stone [Stone et al., 2000b]. Stone propuso un modelo que consideraba al equipo contrario como un *oponente óptimo*. Aunque su modelo como tal no representaba en realidad el comportamiento del oponente sino un modelo genérico de las capacidades de los jugadores (basado en el método de Minimax [Shannon, 1950]), resultó ser un éxito rotundo en la copa mundial RoboCup 1999 [Stone et al., 2000a] consiguiendo el primer lugar del campeonato. En la Sección 2.3.2 se analiza esta propuesta con más detalles.

A partir de las observaciones repetidas de los agentes bajo diferentes situaciones de competencia, se pueden reconocer y descubrir patrones que reflejan el comportamiento de los agentes. La idea de modelar a otros agentes es una tarea compleja que enfrenta diferentes aspectos a ser analizados tales como el comportamiento de cada agente, las relaciones entre ellos y el comportamiento como grupo de agentes. Así, para comprender mejor el comportamiento de los agentes de una forma cabal, se deberían tomar en cuenta diferentes aspectos de comportamiento del sistema multi-agente [Tambe et al., 2000].

Por esta razón, existe la necesidad de contar con modelos de representación lo suficientemente expresivos que faciliten el reconocimiento y descubrimiento del comportamiento de los agentes tanto de manera individual como en equipo. Tales modelos facilitarían la tarea a los mecanismos de razonamiento para reconocer, descubrir e interpretar de una manera adecuada los patrones de comportamiento de los agentes.

Se pueden distinguir diferentes niveles de comportamiento de los jugadores durante un partido:

- Comportamiento individual de los jugadores.
- Comportamiento táctico, donde dos o más jugadores están involucrados.
- Comportamientos a nivel de formaciones que soportan estrategias defensivas y ofensivas.

Por un lado, un equipo que presume de jugar siguiendo ciertas estrategias debería jugar bajo el contexto de formaciones para asegurar orden, disciplina y organización durante un partido [Kuhlmann et al., 2004]. Por otro lado, las jugadas tácticas ocurren, por lo general, bajo el contexto de formaciones.

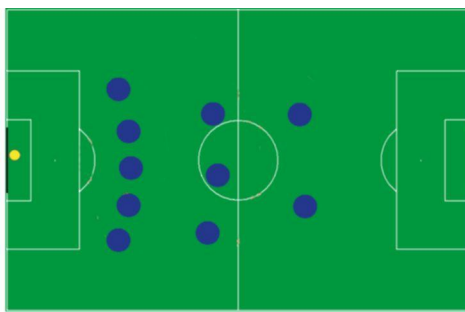


Figura 1.1: Formación 5:3:2

Una formación es una estructura basada en las posiciones y en un conjunto de relaciones establecidas entre los agentes de soccer. La formación representa la relación entre defensas, medios y delanteros de un equipo, por ejemplo, la formación 5:3:2 representa la organización de un equipo en cinco defensas, tres medios y 2 delanteros (ver Figura 1.1).

Un interés actual de investigación en el dominio de los agentes de soccer es el reconocimiento de formaciones y descubrimiento de tácticas de un equipo bajo observación [RoboCup, 2008a]. Esta información puede ser muy útil para el equipo oponente a fin de reducir el desempeño del equipo observado.

Sin embargo, debido a la naturaleza dinámica de un partido y a las múltiples interacciones entre jugadores, el reconocimiento de formaciones y el descubrimiento de jugadas tácticas, es una tarea muy difícil. En [Ayanegui and Ramos, 2007], se propuso un modelo basado en la construcción de estructuras topológicas capaz de seguir correctamente las formaciones y reconocer cambios reales a pesar de que las formaciones eran sometidas a las condiciones dinámicas y a las múltiples interacciones entre los jugadores. En [Ramos and Ayanegui, 2008a] se propuso el descubrimiento de jugadas tácticas bajo el contexto de formaciones.

Una de las contribuciones principales de este trabajo de tesis es el descubrimiento de patrones de comportamiento táctico soportados por el modelo de estructuras topológicas. La estructura topológica se representa por un grafo planar en nuestro estudio, el cual representa relaciones relevantes tales como: la relación entre agentes que pertenecen a la misma zona y la relación entre agentes que pertenecen a zonas vecinas.

Definición de un grafo topológico. Un grafo G es planar si puede representarse sobre la superficie de un plano de tal forma que los nodos representan diferentes puntos y dos ejes o aristas sólo se unen en sus terminaciones. La intersección de dos aristas fuera de sus terminaciones rompe la propiedad planar del grafo G . Este grafo G es también conocido como grafo planar topológico [Berge, 1983].

Por lo tanto, el reto consiste en encontrar dicho grafo planar, el cual sea capaz de

expresar las relaciones relevantes que soporten un adecuado seguimiento de estructuras que cubran aspectos estratégicos y tácticos.

Las jugadas tácticas a ser descubiertas están caracterizadas principalmente por la trayectoria de la pelota, la posición de los jugadores involucrados en la jugada y las zonas de la cancha donde ocurrió la jugada. Las estructuras topológicas de las formaciones en este caso, sirven para seguir a los jugadores involucrados en la jugada táctica en cada instante del juego.

El dominio de prueba de esta investigación es el soccer robótico simulado, específicamente, el Sistema Soccer Server [Emiel Corten, 1999], usado en la Robot World Cup Initiative [Kitano et al., 1997]. Se han descubierto jugadas tácticas dentro de las formaciones ofensivas. La validación del modelo está soportado por un trabajo experimental donde se analizaron 10 equipos que han destacado en varias competencias mundiales de RoboCup.

1.2. Planteamiento del problema

Las características del dominio del soccer robótico tales como: dinamismo, competitividad, tiempo real y múltiples interacciones entre los agentes, dificultan en gran medida la construcción de modelos que faciliten el reconocimiento y descubrimiento de patrones de comportamiento en equipos de soccer robótico. Además, las múltiples interacciones que existen entre los agentes del equipo tienden a complicar el manejo adecuado de tales relaciones en ambientes altamente dinámicos.

1.3. Hipótesis

Disponer de un modelo de representación expresivo, a varios niveles de abstracción, contribuye a facilitar el reconocimiento, descubrimiento e interpretación de patrones de comportamiento de un equipo de agentes.

Por el contrario, un modelo de representación pobre, limita en gran medida al mecanismo de inferencia para caracterizar el comportamiento de los agentes.

1.4. Objetivos

- Construir un modelo capaz de expresar diferentes niveles de representación de un equipo de agentes de soccer.
- Reconocer, descubrir e interpretar patrones de comportamiento en equipos de soccer robótico.

1.5. Alcances

1. Diseñar e implementar el modelo representacional.
2. Reconocer patrones de comportamiento en equipos de partidos reales de soccer robótico simulado.
3. Validar el modelo a través de comparar los patrones de comportamiento descubiertos por el modelo propuesto con las observaciones realizadas por humanos.

1.6. Contribuciones

- Un modelo expresivo a varios niveles de representación.
- Reconocimiento, descubrimiento e interpretación de patrones de comportamiento en diferentes niveles de representación.
- Un algoritmo robusto para el seguimiento de formaciones en ambientes multi-agentes basado en estructuras topológicas.

Capítulo 2

Trabajos Relacionados

2.1. El Soccer Robótico

2.1.1. Introducción y categorías

En esta sección se presenta una descripción y estado del arte del soccer robótico y en especial, de la categoría de simulación en RoboCup.

El soccer robótico, se ha convertido en uno de los problemas estándar de la inteligencia artificial [Matsubara et al., 1996, Kitano et al., 1997], así como alguna vez lo fue el ajedrez. El ajedrez es un juego por turnos, estático y centralizado, en el cual los jugadores tienen una visión completa y exacta del estado actual del juego. Por otro lado, el soccer robótico es un juego en tiempo real y dinámico, con información sensada incompleta y un control distribuido. Además, el diseño y modelación del comportamiento en sistemas multi-agentes competitivos es uno de los grandes retos de los SMACs.

La idea de usar el fútbol soccer para la investigación no es nueva. En 1993 Alan Mackworth propuso en un artículo titulado *On Seeing Robot* [Mackworth, 1993] la idea de que el fútbol podría ser un laboratorio muy interesante para realizar investigación en los campos de la robótica e inteligencia artificial. Un año más tarde, Sahota y Mackworth [Sahota and Mackworth, 1994] formalizaron esta propuesta planteando áreas de oportunidad para la investigación con sistemas multi-agentes cooperativos (SMACs).

En la actualidad, existen dos organizaciones principales que promueven el fútbol soccer robótico a nivel mundial a través de competencias anuales: RoboCup (Robot World Cup Initiative), formada en 1993 y FIRA, Federation of International Robot-soccer Association, formada en 1995. Anteriormente, FIRA tenía el nombre de MiRoSot (Mini-Robot soccer Tournament) debido a que sólo promovía la competencia en la categoría de robots físicos pequeños. RoboCup fue impulsada inicialmente por los japoneses, mientras que FIRA fue una iniciativa de investigadores coreanos. Tanto FIRA como RoboCup promueven el soccer robótico como un esfuerzo por aportar avances en el estado del arte de disciplinas como la robótica, control inteligente, comunicación, procesamiento de imágenes, mecatrónica, tecnología de cómputo, vida artificial y áreas

en ingeniería y tecnología.

RoboCup lanzó el siguiente reto a la comunidad científica:

Para el año 2050, desarrollar un equipo de robots humanoides completamente autónomos que puedan ganarle al equipo campeón de fútbol de humanos [RoboCup, 2008b].

Este gran desafío ha inspirado a cientos de investigadores en todo el mundo para participar en los campeonatos anuales que lleva a cabo esta organización. Aunque para muchas personas este reto representa un objetivo demasiado ambicioso para convertirse en realidad, lo que es realmente importante son todas aquellas contribuciones a la ciencia que se lograrán en la búsqueda de cumplir este gran reto.

RoboCup tiene diferentes ligas o categorías de competencia, con diferentes tamaños de robots y diferentes retos, entre las que se encuentran: simulación, robots físicos de tamaño pequeño, robots físicos de tamaño medio y humanoides. Mientras que las principales líneas de investigación a las que se enfocan las categorías de robots físicos buscan resolver aspectos de hardware como visión y mecanismos, la liga de simulación provee un ambiente favorable para las líneas de investigación en Sistemas Multi-Agente Cooperativos tales como: modelado de agentes, comunicación, coordinación, trabajo en equipo, entre otros. Debido a que el trabajo que aquí se presenta está basado en la liga de RoboCup dentro de la categoría de simulación, a continuación se exponen más detalles sobre este dominio en particular.

2.1.2. Descripción del dominio de simulación de RoboCup

La liga de simulación de RoboCup emplea un sistema de software llamado Soccer Server para simular el campo y los objetos (Figura 2.1). El Soccer Server es un programa que habilita un partido de soccer entre dos equipos desarrollados posiblemente en diferentes lenguajes de programación. Cada jugador del equipo es un agente que se conecta al servidor a través de una comunicación cliente-servidor. El Soccer Server provee un campo de soccer virtual y simula los movimientos de los jugadores y el balón. Un programa cliente provee el cerebro de un jugador al conectarse al Soccer Server a través de una red (usando un socket UDP/IP) y especificando las acciones que llevará a cabo como respuesta a la información sensada que recibe en cada ciclo de simulación.

Los jugadores reciben información visual y auditiva cada 150 mseg. sobre la red y pueden ejecutar acciones como patear, dar vuelta, girar la cabeza y hablar cada 100 mseg. El servidor procesa las acciones de los jugadores y genera nueva información visual. La información visual consiste básicamente de las distancias y ángulos de otros jugadores, el balón y marcas en la cancha. Los jugadores sólo pueden percibir objetos que están en su campo de visión y tanto la información visual sensada como la ejecución de las acciones contienen ruido. Además, la exactitud y cantidad de información sensada

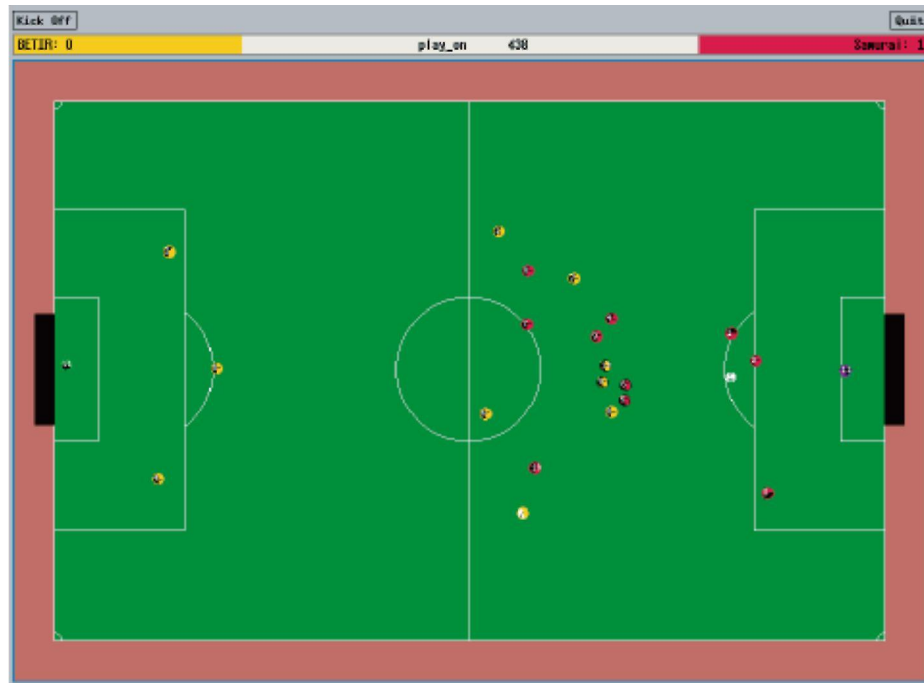


Figura 2.1: El campo de juego del Soccer Server

se reduce de acuerdo a la distancia de un objeto. La comunicación entre los clientes sólo es permitida si pasa por el servidor. El ancho de banda y el rango auditivo están limitados.

Los tres módulos principales en el Soccer Server son:

1. Un módulo simulador del campo de juego(field simulator). Crea el mundo virtual básico del campo de soccer y calcula los movimientos de los objetos, verificando las colisiones entre estos.
2. Un módulo árbitro (referee). Este asegura que se cumplan las reglas del juego.
3. Un módulo para tablero de mensajes (message board). Este administra la comunicación entre los programas cliente.

La Figura 2.2 muestra la relación de estos módulos y cómo el Soccer Server se comunica con los clientes y el programa que visualiza los elementos de la cancha. Un cliente controla a un sólo jugador. El cliente recibe información sensada visual y auditiva del servidor y envía comandos de control al servidor. La información visual contiene información parcial del campo desde el punto de vista de los jugadores, tal que el programa del jugador es quien toma las decisiones con base en información incompleta. La comunicación entre los agentes es limitada a 128 bits de información y tiene como finalidad que los jugadores puedan comunicarse para tomar decisiones de las acciones que llevarán a cabo.

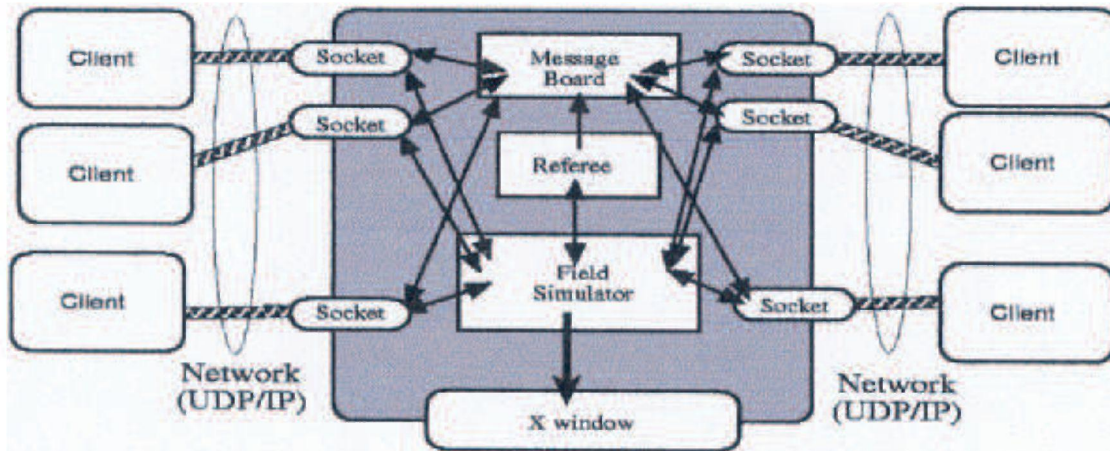


Figura 2.2: Estructura del Soccer Server

2.1.3. El coach

Además de los once jugadores, cada equipo puede conectar un cliente especial al servidor, llamado *coach*. Este agente coach es el equivalente al entrenador o director técnico en el fútbol soccer real y se caracteriza aquí por ser un agente que no puede ejecutar acciones en el campo. Sin embargo, a diferencia de los demás agentes, en cada ciclo del partido el coach recibe información visual sin ruido de cada uno de los jugadores así como de la pelota en la cancha. Es decir, en cada ciclo de simulación el coach recibe las posiciones y velocidades absolutas de todos los jugadores y del balón. También es permitido que el coach pueda comunicar mensajes arbitrarios a los jugadores mientras el juego es detenido y mensajes en un lenguaje estándar (CLang) en ciertos intervalos de tiempo. Esto hace del coach una herramienta valiosa para el análisis táctico y modelado de oponentes [Sean Buttinger, 2001].

La misma información que es enviada al coach durante el juego puede ser grabada en un archivo log, de tal forma que sea posible emplear algoritmos fuera de línea para analizar toda la información de cada juego.

El lenguaje estándar que el coach puede usar para comunicarse con sus jugadores es CLang. Este lenguaje es básicamente un sistema de reglas de producción que mapea situaciones a acciones. Las reglas son consejos que el coach envía a su equipo para actuar en cierta forma bajo condiciones específicas del juego.

2.2. Modelado de Agentes

El modelado de agentes aborda el problema de cómo construir modelos de comportamiento de un sistema multi-agente a partir de observaciones de las acciones que llevan los agentes. Como se demuestra en [Garrido-Luna et al., 1998], disponer de información acerca del comportamiento de agentes oponentes, permite obtener una clara

ventaja competitiva comparado a no contar con conocimiento alguno del oponente.

El modelado de agentes se ha convertido rápidamente en un tema de gran interés no sólo en el soccer robótico sino también en otros campos de investigación tales como: interfaces inteligentes de usuario [Lesh et al., 1999, Davison and Hirsh, 1998], ambientes virtuales para entrenamiento [Rickel and Johnson, 1999, Tambe and Rosenbloom, 1995c], entrenamiento automático de equipos [Patrick Riley, 2002], tutoriales inteligentes (TI) [Anderson et al., 1990], reconocimiento de planes [Kautz and Allen, 1986], reconocimiento automático de imágenes [Dufournaud et al., 2004], entre otros.

En un dominio multi-agente, dinámico y competitivo, un agente que persigue sus metas se encuentra con diversos eventos que impactan significativamente las acciones que lleva a cabo para alcanzar sus objetivos. Algunos de estos eventos pueden ser provocados por el agente mismo (ej. patear el balón en una dirección incorrecta) o por otros agentes con los que interactúa. Por esta razón, en muchos dominios los agentes deben modelar el comportamiento de otros agentes con los que tienen que están interactuando. Contar con modelos de otros agentes es valioso cuando se requiere predecir acciones futuras o descubrir estrategias a fin de sacar ventaja sobre el oponente.

Sin embargo, modelar un equipo de agentes es una de las tareas más complicadas en el campo de los sistemas multi-agentes, ya que implica descubrir comportamientos de alto nivel, tales como estrategias o tácticas, contando sólo con conocimiento de bajo nivel acerca del comportamiento básico de los agentes. Esta tarea es difícil ya que los agentes son entidades autónomas que buscan alcanzar sus metas definidas previamente y al mismo tiempo, ellos tienen que interactuar con compañeros de equipo, agentes adversarios y el medio ambiente que los rodea. Al contar únicamente con descripciones de bajo nivel del comportamientos básicos de los agentes (como correr, patear el balón, o dar vuelta), se vuelve una tarea difícil reconocer o descubrir comportamientos de alto nivel (como estrategias y tácticas). En las siguientes secciones se analizarán diferentes enfoques que se han empleado para modelar el comportamiento de equipos multi-agente.

2.3. Modelado de Oponentes

Si la tarea de reconocer el comportamiento de un equipo de agentes o jugadores se realiza sobre el equipo oponente en turno, se pueden obtener ventajas competitivas que impactan significativamente el resultado de una competencia [Carmel and Markovitch, 1996, Garrido-Luna et al., 1998].

El modelado de oponentes en SMAC (Sistemas Multi-Agentes Cooperativos) enfrenta problemas que no se encuentran en otros dominios. Por ejemplo, en algunos ambientes de SMAC se deben expresar objetivos mutuos y diferentes acciones de los agentes. En algunos casos es importante considerar a agentes reactivos más que deliberativos, e incluso, hay ocasiones donde se debe modelar al oponente al mismo tiempo que se está compitiendo contra él (modelado en línea). Investigaciones actuales se han enfocado al reconocimiento de planes dinámicos, ambientes en tiempo real y modelado del trabajo en equipos multi-agente [Sukthankar and Sycara, 2008].

En la actualidad, hay una gran interés en el modelado de oponentes dentro del dominio de soccer robótico [Visser and Weland, 2002, Steffens, 2004, Ledezma et al., 2004, Markovitch and Reger, 2005]. En esta sección se presentan diferentes enfoques que se han propuesto para obtener un modelo del oponente en el soccer, haciendo un énfasis especial en la liga RoboCup.

2.3.1. Modelado externo

Se refiere al hecho de que los humanos observen el comportamiento de equipos de agentes y sean ellos mismos quienes determinen las estrategias adecuadas para enfrentar al equipo oponente, ya sea ajustando parámetros o piezas de código. En los primeros años que se llevó a cabo el campeonato de RoboCup, ésta era la manera clásica en que se adaptaba el comportamiento de los jugadores de un equipo. Sin embargo, en la mayoría de las ocasiones, los diseñadores contaban con muy poco tiempo, entre juego y juego, para hacer un análisis atinado del oponente, determinar los ajustes necesarios en el equipo y finalmente implementar la contra estrategia correspondiente.

Además de las limitaciones de tiempo que el diseñador tiene para analizar a un equipo de soccer, también existe una falta de objetividad por parte de un humano para identificar importantes patrones de comportamiento que ocurren a diferentes niveles de abstracción, principalmente debido a los constantes cambios de posición de los jugadores del equipo y limitaciones en cuanto a la duración de los juegos. Por ejemplo, reconocer la formación del equipo, las jugadas tácticas que concluyeron en un gol, qué jugadores están provocando variantes en la formación y bajo qué circunstancias ocurren. Estos aspectos pueden no ser fácilmente observables por el hombre y demandan de algoritmos que reconozcan y descubran tales comportamientos en los equipos.

Por un lado, varios investigadores se han enfocado a realizar trabajos para crear herramientas que ayuden a los diseñadores de equipos de soccer robótico, a analizar de manera automática a equipos fuera de línea [Raines et al., 1999, Tambe et al., 2000, Lau and Reis, 2001]. Por otro lado, se han publicado también trabajos con el enfoque de modelado de oponentes en línea, es decir, mientras que el equipo analiza el comportamiento de su oponente, al mismo tiempo tiene que jugar contra él. Este último enfoque tiene la limitación de contar con muy poco tiempo para obtener un modelo adecuado del equipo. De hecho, Peter Stone, un reconocido y respetado autor en el área de soccer robótico, argumenta que es prácticamente *imposible* crear un modelo correcto del oponente considerando solamente los diez minutos de duración que tiene un partido de soccer robótico [Stone et al., 2000b].

2.3.2. Modelado con oponentes ideales

Dentro de los primeros trabajos para tomar en cuenta al oponente, Peter Stone y colegas propusieron una técnica llamada IMBBOP (Ideal Model Based Behavior Outcome Prediction) para modelar las acciones óptimas que llevarían a cabo los jugadores

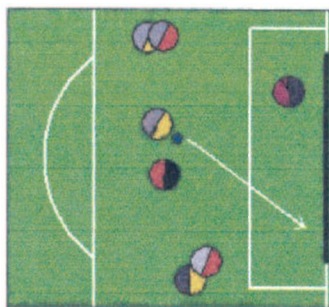


Figura 2.3: Ejemplo de tiro a gol.

oponentes dentro de un mundo ideal [Stone et al., 2000b]. La idea principal consiste en considerar al oponente como alguien que siempre ejecuta acciones óptimas. Similar al algoritmo Minimax [Shannon, 1950], el agente modelador enumera todas las acciones posibles que el otro agente puede tomar y asume que el agente siempre tomará aquella acción que es mejor para él y peor para el agente modelador. De esta forma, se puede considerar a un portero oponente óptimo como aquél que siempre reacciona en el momento en que es pateado el balón hacia su portería y que se mueve a la máxima velocidad permitida para atrapar el balón (aunque en la realidad no fuera así). Stone y colegas aplicaron ésta técnica con sus jugadores delanteros para hacer tiros a gol imparables, ya que los delanteros podían saber el momento clave para patear el balón hacia la portería y la dirección atinada para colocar el balón en una zona inalcanzable para el portero. En la Figura 2.3 se muestra un ejemplo de tiro a gol donde el jugador con la pelota prepara su disparo hacia una zona de la portería imposible de alcanzar por el portero.

Para decidir si un jugador dispara hacia la portería del oponente, el agente primero identifica la mejor posición de la portería a la que debe dirigir el balón. Generalmente toma en cuenta dos puntos que son las dos esquinas de la portería. Luego el agente traza líneas imaginarias desde la posición actual del balón hacia cada uno de estos puntos. El mejor punto para patear el balones es aquél que se encuentra mas alejado de la posición actual del portero. El agente entonces ejecuta el algoritmo IMBBOP para predecir si el disparo que hará a ese punto es posible que sea detenido por el portero, considerando la posición actual del balón y la máxima velocidad del portero para dirigirse hacia ese punto. En caso de que sea posible detener el disparo, el jugador con la pelota buscará dar un pase a un compañero con mejores posibilidades para tirar a gol.

A pesar de que se reportaron buenos resultados al aplicar ésta técnica (primer lugar en el campeonato de 1999), no se consigue realmente modelar el comportamiento del equipo. Existen oportunidades de gol que pueden ser aprovechadas si se toma en cuenta conocimiento del comportamiento real del portero. Como consecuencia, la tendencia de sobrestimar al oponente al considerarlo siempre como óptimo, pasa por alto sus debilidades reales y por lo tanto, buenas oportunidades de tiros a gol.

2.3.3. Clasificación de oponentes

Una de las ventajas competitivas en jugadores experimentados obedece al hecho de que tienen una idea clara de qué hacer en ciertas situaciones de juego debido a experiencias en juegos previos. Este conocimiento puede ser adquirido por jugar muchos partidos o por el constante entrenamiento. En ambos casos, un jugador experimentado puede comparar la situación actual con un rango amplio de experiencias anteriores; al reconocer una situación dada, el jugador sabrá qué acciones deberá ejecutar.

El área de clasificación del comportamiento del oponente se basa principalmente en contar con una biblioteca de clases de oponentes previamente definida y compararla con observaciones del oponente actual. A partir de ahí, se toma el modelo más parecido y éste puede emplearse como fuente de información para modificar el comportamiento del equipo [Riley and Veloso, 2002, Riley and Veloso, 2001, Steffens, 2002].

De manera similar, en el fútbol real el entrenador o los jugadores observan el comportamiento de los oponentes durante un juego y lo comparan con los oponentes que han tenido previamente. Con base en estas observaciones, identifican las estrategias del oponente actual y adaptan su comportamiento considerando como base una contra estrategia que les haya dado éxito anteriormente.

Muchos trabajos publicados sobre clasificación de oponentes en soccer robótico coinciden en los siguientes pasos:

1. Abstracter características de equipos en juegos previos.
2. Construir una biblioteca de clases de equipos.
3. Comparar las características de un equipo observado con las clases de equipos en la biblioteca.
4. Adaptación del comportamiento.

Abstracción de características

Decidir qué aspectos se deben considerar al modelar el comportamiento de un equipo y cómo reconocerlos, es la diferenciación que caracteriza a los diferentes trabajos relacionados en el área. Algunos aspectos básicos que pueden caracterizarse de un equipo de soccer robótico son:

- Pases (trayectorias y zonas de la cancha donde ocurren)
- Dribles
- posición del balón
- posición de los jugadores

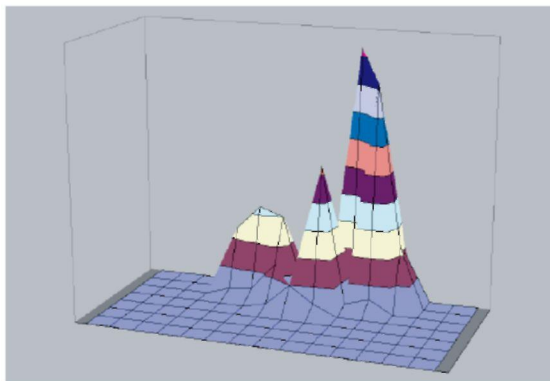


Figura 2.4: Ejemplo de un RectGrid.

Por otro lado, existen comportamientos de más alto nivel que también pueden caracterizarse, tales como estrategias de ataque y defensa, llegadas a gol, jugadas defensivas, etc. A continuación se discuten trabajos que analizan comportamientos básicos de los jugadores y más adelante se analizan trabajos que abordan el estudio de comportamientos de más alto nivel.

Riley y Veloso caracterizan comportamientos básicos como pases, posiciones del balón y posiciones de los jugadores en la cancha [Riley and Veloso, 2000]. Para ello, discretizan el campo de juego en pequeños cuadrados, los cuales juntos forman una estructura de datos que ellos llaman *RectGrid*. Para cada característica básica diferente, hay un RectGrid y cada evento observado es contabilizado en el cuadro correspondiente a ese evento (Figura 2.4). Esto les lleva a una distribución característica a través de los RectGrids. El trabajo de Riley y Veloso tiene la desventaja de que no utiliza información temporal y por lo tanto, después de que se registra un pase no es posible identificar hacia qué dirección ocurrió el pase, dado que el inicio y final son tratados igualmente.

Una desventaja con este enfoque es que no es posible diferenciar entre una táctica que usa pases desde el centro a los extremos con otra táctica que realiza pases desde las alas a la media cancha. Otra desventaja es que el modelo del oponente almacena solo información posicional o geométrica. No hay forma de representar explícitamente secuencias de pases o cualquier otra interacción que existe entre compañeros de equipo. La coordinación entre jugadores es una parte esencial de las tácticas y por lo tanto, se debería tomar en cuenta. Por esta razón, el modelo de representación propuesto en esta tesis, considera muy relevante el hecho de modelar las relaciones entre los jugadores del equipo de soccer.

Construcción de la biblioteca

Algunos autores como Peter Stone de la Universidad de Texas y Patrick Riley de Carnegie Mellon, sugieren que como la duración de un juego es demasiado corta para crear un modelo del oponente, uno debería contar con una biblioteca de clases de

oponentes de antemano y luego sólo compararlas con las observaciones del oponente actual. Por esta razón, al abstraer un conjunto de características en varios equipos (como el caso descrito anteriormente), se forma un conjunto de clases de oponentes. En algunos casos, hasta puede asociarse a la clase del oponente, la contra estrategia correspondiente que pueda traer el éxito sobre esa clase de oponente [Visser et al., 2000].

El reto aquí es contar con una biblioteca de clases de oponentes suficientemente completa que garantice, de alguna forma, que el oponente actual tiene un perfil correspondiente a algún modelo de la biblioteca previamente capturado. Esto puede traducirse en un verdadero problema pues no hay forma de asegurar lo anterior. En principio, uno esperaría que un conjunto reducido de modelos no sea suficiente para representar a una gran variedad de oponentes con estrategias y tácticas tan variadas.

Selección de las clases del oponente

La clase de oponente definida por Riley y Veloso [Riley and Veloso, 2000] es un conjunto de RectGrids, llamados configuraciones destino, de esta forma, un conjunto que contiene un RectGrid para las posiciones de pase del oponente podrían denotar un oponente que pasa desde el centro a las alas. En el trabajo de Riley las clases fueron construidas manualmente.

Una primera propuesta fue construir RectGrids para las configuraciones destino durante el juego y luego medir la similaridad con las correspondientes en la biblioteca [Riley, 1999]. Desafortunadamente, los experimentos revelaron que la identificación del modelo correcto fue sólo un poco mejor que hacerlo aleatoriamente. Se sugirió que esto fue debido al hecho que la similaridad entre los modelos diferentes era muy pequeña.

Otro enfoque para seleccionar a una clase de oponente es usar un clasificador bayesiano [Riley and Veloso, 2001]. Inicialmente hay una distribución probabilística definida manualmente sobre los modelos previamente determinados $\{M_1, M_2, \dots, M_m\}$ (cada modelo representa distribuciones de probabilidad de que un jugador esté en cierta posición de la cancha). Al principio, se utiliza el modelo que tiene la probabilidad más alta. Después de hacer algunas observaciones del oponente, se van actualizando las probabilidades en los modelos M_i . Luego, cada observación siguiente de los jugadores se usa para recalculan la distribución a través de la ecuación 2.1:

$$P[M_i|\omega] = \frac{P[\omega|M_i]P[M_i]}{P[\omega]} \quad (2.1)$$

En cada observación se toman en cuenta tres aspectos: Las posiciones iniciales de todos los jugadores (cada vez que es interrumpido el juego por alguna razón), el movimiento del balón y las posiciones finales de los jugadores. Esta información se usa para calcular una nueva distribución de probabilidad. A pesar de que es mejor usar este enfoque probabilístico que el uso de los RectGrids, sigue siendo un gran reto el contar con el número de modelos suficientemente descriptivos para clasificar al oponente actual. Además, también se mantiene el problema de cómo se construyen estos modelos.

En [Riley and Veloso, 2001] se definió manualmente cada modelo.

Adaptación del comportamiento

En los primeros trabajos de Riley y Veloso, no toman en cuenta el enfoque probabilístico descrito anteriormente [Riley, 1999, Riley and Veloso, 2000]. Sus investigaciones siguientes usaron los modelos del oponente para predecir probabilidades de posiciones de los jugadores [Riley and Veloso, 2002]. Sobre estas predicciones, se define un plan de posicionamiento para los jugadores. Así por ejemplo, en situaciones donde se interrumpe el juego y se va a realizar un tiro libre, se crea un plan de posicionamiento a fin de que la probabilidad de que un jugador oponente intercepte un pase sea minimizada.

A pesar de que hay resultados positivos usando el enfoque de clasificación del oponente [Riley, 1999, Riley and Veloso, 2002], no se han obtenido muy buenos resultados. El hecho demuestra que muchas veces el modelo del oponente con el que está razonando el equipo puede estar alejado de la realidad y por tal razón, descuida oportunidades valiosas del oponente actual.

2.4. Reconocimiento de Planes

En muchos casos, la ventaja de modelar el comportamiento de otros agentes ha permitido que puedan emplearse mecanismos de inferencia que permitan reconocer los planes o estrategias que llevan a cabo grupos de agentes. El reconocimiento de planes se ha desarrollado al mismo tiempo que la investigación en planeación, ya que en cierta forma, es el proceso inverso de la ejecución de un plan. Por un lado, la planeación tiene que ver con encontrar una secuencia de acciones para alcanzar un objetivo y la ejecución del plan es la realización de la secuencia planeada de acciones. Por otro lado, el reconocimiento de planes intenta inferir el plan que un agente ejecuta a partir de las acciones observadas. Por esta razón, el enfoque principal del reconocimiento de planes es sobre arquitecturas de agentes deliberativos (agentes que actúan con base en un plan).

El problema de reconocimiento de planes se ha definido como el problema de identificar un conjunto mínimo de acciones de alto nivel que son suficientes para explicar las acciones observadas [Kautz and Allen, 1986]. El trabajo de Kautz y Allen ha sido la base de muchos otros trabajos en reconocimiento de planes [Goldman et al., 1999, Bui, 2003]. En su representación, los planes son grafos con acciones de alto nivel como los nodos raíz, es decir, sus nodos representan metas y submetas. Otras acciones, descomposición o especializaciones dependen de nodos de las acciones de alto nivel. El reconocimiento de planes es entonces una forma de cubrir grafos.

Los agentes que intentan reconocer planes de otros agentes generalmente necesitan una biblioteca de planes, que contiene planes que el agente observado es probable que ejecute. Muchos trabajos usan planes que descomponen tareas a diferentes niveles [Goldman et al., 1999, Quilici et al., 1998, Rao, 1994].

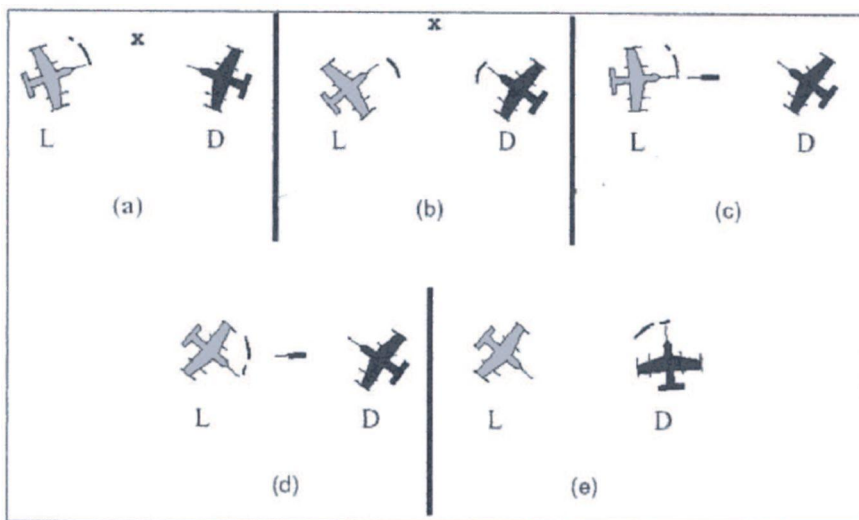


Figura 2.5: Un escenario de combate aéreo simulado.

Como se mencionó anteriormente, el reconocimiento de planes se enfoca en arquitecturas de agentes deliberativos. Así, tiene que ver mucho con el orden temporal o acciones y planes jerárquicos. Sin embargo, poca investigación se ha realizado en los comportamientos reactivos o planes que cambian rápidamente bajo dominios dinámicos como el soccer.

En dominios que son parcialmente observables, los agentes necesitan inferir acciones que no fueron observadas partiendo de aquellas que sí fueron visibles a fin de razonar exitosamente acerca de los planes y objetivos de los otros agentes. Tambe y colegas proporcionan un esquema para el seguimiento de eventos en un dominio de simulaciones militares donde es importante inferir acciones no observadas como el lanzamiento de un misil a partir de eventos que sí son observables como giros y movimientos del radar [Tambe and Rosenbloom, 1995a]. Por ejemplo, si no se observó el lanzamiento de un proyectil por parte de un avión enemigo, ésta acción puede ser deducida por el giro inesperado que hace el avión y que caracteriza a su lanzamiento (ver Figura 2.5). En la Figura 2.5 (a) y (b) se observa cómo el avión *L* comienza a alinearse para disparar, luego en la Figura 2.5(c) dispara y en la Figura 2.5(d) hace un giro para desviarse de la trayectoria indicando que ha disparado. Suponiendo que la acción de disparo no fue observada por el avión *D*, éste puede inferir tal acción a partir de observar que *L* cambia su trayectoria.

Tambe usa un operador jerárquico que es implementado en la arquitectura Soar [Laird et al., 1987]. Este operador jerárquico es similar a los planes jerárquicos, pues descompone objetivos en un conjunto de acciones secuenciales. A través de comparar los eventos observados uno puede inferir los objetivos en los niveles más altos. En los dominios dinámicos como simulaciones militares se requiere la habilidad de modelar tanto comportamientos reactivos como deliberativos. En [Tambe and Rosenbloom, 1995c]

se propone una técnica para el seguimiento de agentes que consiste en que el agente asume que el oponente usa una jerarquía de operadores similar y solo ejecuta la jerarquía fuera de la vista del oponente. Esto puede también incluir modelar al agente fuera de la perspectiva del oponente. El seguimiento de agentes de Tambe supone que el modelador y el agente modelado usan los mismos operadores y espacios del problema. Podría ser el caso que existen maniobras estándares en escenarios militares las cuales también conoce el oponente. Pero para seguir a un oponente, el autor también hace la suposición que el modelador sabe el estado exacto del oponente. Sin embargo, por lo general, es imposible estimar o inferir el estado exacto de otro agente a partir de la observación, especialmente si algunas acciones de los otros son imposibles de detectar [Tambe and Rosenbloom, 1995b].

Intille y Bobick [Intille and Bobick, 1999] presentan un esquema para reconocer acciones en un sistema multi-agente a partir de evidencia visual y la evalúan por medio del análisis de cintas grabadas de fútbol americano. Definen relaciones temporales entre eventos, tales como *before* y *after*. Por ejemplo, la acción *run-with-ball* tiene el antecedente de la acción *catch* que ocurre antes de ella. Esta propuesta es inspirada en el reconocimiento de planes y funciona bien para el fútbol Americano el cual tiene muchas jugadas o tácticas predefinidas. Cada acción primitiva fue codificada en una red de creencias, capaz de representar los objetivos de los agentes así como los planes de tal forma que puedan aplicarse principios de reconocimiento de planes [Huber et al., 1994]. Si estas acciones son detectadas por las redes, entonces se prueban las relaciones temporales entre ellas. Las redes consisten de dos tipos de nodos: el primero es llamado *nodo de creencia no observable* que puede ser verdadero o falso y contiene estados internos de los agentes modelados que no pueden ser observados. Los otros son *nodos de evidencia observados* y representan las acciones que fueron observadas. Por ejemplo, no puede ser observado directamente, si un jugador está listo para atrapar el balón. Como tal, un evento no observable tiene que ser inferido de hechos observables como por ejemplo si el jugador está frente al balón o cerca de él. De esta manera, los estados internos no observables son una fuente de ambigüedad.

Los eventos observables son n-adas, así por ejemplo el resultado de un detector de distancia puede ser *inContact*, *nextTo*, *near*, *far* y así por el estilo. Los nodos de creencia pueden ser llenados con otras redes. Intille y Bobick argumentan que este enfoque trabaja bien debido a que la red total es solo una aproximación de los objetivos del agente.

Una extensión de las redes de creencia es el trabajo de Suryadi y Gmytrasiewicz [Suryadi and Gmytrasiewicz, 1999]. Mientras que la propuesta de Intille y Bobick se enfocó en comparar las observaciones con redes predefinidas, el trabajo de Suryadi y Gmytrasiewicz propone aprender estas redes. Puesto que el modelo correcto generalmente no es conocido con certeza, se mantiene una distribución de probabilidad sobre diferentes modelos. Si el comportamiento observado de un agente no es parecido a los modelos previamente definidos, entonces uno de estos modelos es modificado para mejorar la similaridad con las observaciones hechas.

Sharifi y colegas proponen un sistema multi-agente cuyo objetivo es crear a un comentarista automático de partidos de fútbol soccer [Mohsen Sharifi, 2003]. En algunos casos, el comentarista debe reportar eventos que ocurren al mismo tiempo que se observan en la cancha (tiempo real), por ejemplo, cuando un agente dribla y en otros casos, debe anticipar el resultado de las acciones de un agente, como un pase o tiro a la portería del oponente. Se subraya la importancia de un modelo que permita predecir el comportamiento de los agentes, especialmente cuando se llevan a cabo pases o intercepciones del balón. A través del uso de heurísticas y redes neuronales proponen un mecanismo de predicción para determinar qué agente recibirá el balón una vez que ha sido pateado por otro agente. Debido al alcance del trabajo, el análisis que realiza el sistema comentarista está más centrado en el movimiento del balón que en la estrategia del equipo. Es decir, el estudio se centra más en los jugadores que participan activamente cerca del balón que en la inferencia de un modelo de comportamiento de los equipos.

En [Bauer, 1999] se presenta un trabajo usando Clustering para encontrar grupos de secuencias similares los cuales son luego llevados a un algoritmo de adquisición de planes de supervisión especializados el cual determina los planes abstractos a partir de estos grupos o clusters. Este trabajo no ha sido aplicado en dominios continuos multi-variable: asume que las acciones son discretas y son ejecutadas por un solo agente. Además, el método de Bauer depende de la asignación manual de ciertos parámetros de Clustering.

2.5. Reconocimiento de Formaciones y Estrategias

En el soccer robótico, el reconocimiento de planes está basado principalmente en el análisis del comportamiento del equipo de jugadores. En [Bodey R. Baker, 2006] se menciona que en el soccer robótico, la forma de especificar estrategias es a través de formaciones de equipo. Así mismo, las formaciones determinan las posibles acciones cooperativas que pueden llevarse a cabo por parte de los jugadores. La formación es una descomposición del equipo en un conjunto de tareas o roles [Bodey R. Baker, 2006].

La formación es el elemento esencial que aporta información del comportamiento del equipo que los jugadores asumen en el juego, ya que ésta dicta las maneras en que los jugadores se relacionan con sus compañeros a través de definir el rol de cada jugador (defensa, medio o delantero).

La formación es uno de los conceptos principales en un equipo de soccer ya que es la forma en que puede definirse una buena distribución de los jugadores en la cancha. Cuando se usa una formación, los jugadores serán capaces de mantener bien cubiertas las partes principales de la cancha durante un juego y con ello evitar un *aglutinamiento* de los miembros del equipo en cierta área de la cancha.

Mientras que en el Capítulo 3 se discute más ampliamente el concepto de formación, aquí se discuten algunos de los trabajos más relevantes en reconocimiento de formaciones en soccer robótico.

Riley y Veloso intentan reconocer formaciones de equipo con el fin de imitar la formación del oponente. [Patrick Riley, 2002]. Ellos proponen un algoritmo que consiste en dos fases:

1. El objetivo de la primera fase es, para cada agente, encontrar un rectángulo con un área que depende de todos los puntos en donde más frecuentemente estuvo el agente durante los juegos pasados. Este proceso se realiza de manera independiente para cada agente.
2. La segunda fase busca capturar algunas correlaciones entre pares de agentes. La posición de los rectángulos se mueve para ajustarlos, pero su forma no cambia. Aquí se intentan conectar los centros de los rectángulos de cada par de agentes. La idea es poder agrupar estos rectángulos para definir a las tres zonas del equipo: defensas, medios y delanteros.

Riley usa el reconocimiento de la formación principalmente para imitar la formación del equipo oponente a fin de aplicar lo que él denomina *marcación basada en formación* [Patrick Riley, 2002]. A cada uno de los defensas se le asigna uno de los delanteros del oponente para marcarlo durante todo el juego.

Para el aprendizaje de reglas de pases, el sistema de Riley observa los pases de los equipos en juegos pasados para aprender reglas que capturen algunos de estos patrones de pases. Estas reglas pueden luego ser usadas ya sea para imitar un equipo o para predecir los pases que hará un oponente. Así, este aprendizaje usa el algoritmo de Clustering para crear regiones en el campo y C4.5 para generar reglas describiendo el comportamiento de pases de un equipo.

A diferencia del trabajo propuesto aquí, el trabajo de Riley y Veloso obtiene la formación del equipo pero sólo para fines de marcación personal imitando su formación [Patrick Riley, 2002]. Además, ellos no buscan construir un modelo de comportamiento del equipo de agentes para descubrir sus estrategias. En cuanto al aprendizaje de patrones de pases, lo que buscan es aprender a predecir los pases en juegos futuros. Sin embargo, no modelan explícitamente las relaciones entre los agentes.

Similar al trabajo de Riley, Gregory Kuhlmann y Peter Stone buscan reconocer formaciones con base en las posiciones promedio que recorre cada uno de los jugadores (*home positions*) [Kuhlmann et al., 2004]. A diferencia del trabajo de Riley, ellos no buscan relacionar las posiciones promedio de los jugadores pero introducen la idea de modelar un vector de atracción al balón para cada jugador. Es decir, ellos modelan la formación como la combinación de dos aspectos: a) la posición promedio (X, Y) de cada jugador en la cancha y b) el vector de atracción al balón (BX, BY) para todos los jugadores. En la tesis de Boer también se emplea este vector de atracción al balón [de Boer and Kok, 2002]. Sin embargo, Boer define además un conjunto de reglas para que un jugador se mueva de su posición promedio y vaya tras el balón, tomando en cuenta el rol que tiene ese jugador. Los valores X y Y se calculan como las coordenadas promedio x y y de cada jugador observado durante el juego. Los valores

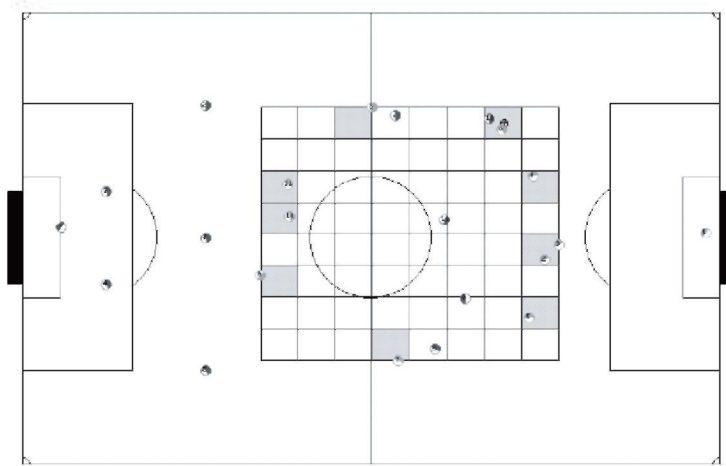


Figura 2.6: Entrada de la red neuronal con las posiciones de jugadores capturadas en la cuadrícula de 8x8

BX y BY se escogen cuidadosamente para cada posición a través de experimentación. La desventaja principal que presenta este enfoque es similar al de Riley, ya que no se modelan las relaciones entre los jugadores y por lo tanto una debilidad que ellos mismos comentan en su trabajo es que la atracción de la bola causó que los delanteros jugaran demasiado lejos hacia la portería del oponente, así que tuvieron que compensar esta situación ajustando manualmente las posiciones promedio para cada jugador delantero.

Visser y colegas hacen el reconocimiento de la formación de un equipo oponente de soccer usando redes neuronales [Visser et al., 2000]. La entrada de la red es un conjunto de 64 variables binarias que reciben el valor correspondiente de una cuadrícula de 8x8 celdas que se coloca alrededor de las posiciones de los jugadores oponentes en la cancha (ver Figura 2.6). Debe haber al menos un jugador dentro de una celda para que tal celda reciba el valor de 1, de otro modo deberá asignarse a 0. De esta manera, el total de 64 variables se introducen en la red neuronal y ésta las clasifica en una de 16 posibles formaciones previamente definidas.

Cada una de estas formaciones predefinidas tienen asociada una formación que contraresta a la formación del oponente. De esta manera, una vez que se clasifican las posiciones de los jugadores oponentes en turno en una formación conocida, los jugadores se alinean a la formación sugerida para contrarrestar la estrategia del oponente.

Una desventaja importante de este enfoque basado en redes neuronales se debe a que las redes son como cajas negras que clasifican las posiciones de los jugadores en una formación pero no modelan explícitamente las relaciones entre los jugadores. Debido al movimiento constante de los jugadores, la red neuronal podría estar obteniendo diferentes clasificaciones de formaciones cuando en realidad el equipo sigue jugando con la misma formación.

En el trabajo realizado por Anthony Yuen, se estudia de qué manera las redes neuronales multicapa pueden usarse para reconocer formaciones de equipo, intentando

contestar preguntas como cuál es el número óptimo de nodos ocultos de la red para tener una mayor efectividad en el reconocimiento de la formación [Yuen, 2001, Yuen, 2002]. Al igual que Visser, Yuen utiliza las posiciones de los jugadores oponentes para reconocer la formación, pero Yuen propone únicamente 5 formaciones predefinidas a ser reconocidas:

- Formaciones ofensivas: 4:3:3 y 4:2:4
- Formaciones defensivas: 4:4:2 y 5:3:2
- Formaciones balanceadas: 3:4:3

Yuen también utiliza una cuadrícula pero de tamaño 9x9 (81 celdas). La aportación principal del trabajo de Yuen es la demostración que hizo en [Yuen, 2002], donde explica la ventaja de utilizar una red neuronal multicapas en lugar de una red de una sola capa.

Es importante señalar que en el trabajo de Yuen se utilizan pocas formaciones a ser reconocidas. Sería interesante saber el resultado que obtiene al intentar reconocer un conjunto más completo de formaciones de equipo.

En el proyecto ORCA (Osnabrueck RoboCup Agents) [Andreas G.Nie, 2001], utilizan también una cuadrícula sobre toda la cancha de juego pero con el fin de hacer una marcación personal a los delanteros oponentes (ver Figura 2.7). El enfoque que ellos proponen es analizar juegos previos para obtener los porcentajes de ocupación geográfica de los jugadores, a fin de predecir en que celda estarán los jugadores en un futuro y poder marcarlos eficientemente. Ellos argumentan que no es buena idea reconocer la formación del oponente debido a que las posiciones que ocupan los jugadores depende en gran medida de la marcación, movimiento del balón y ruido. Sin embargo, como se discute en el siguiente capítulo, los equipos tienden a jugar con estrategias que dictan un comportamiento ordenado en el juego el cual permite soportar tácticas que los jugadores llevan a cabo de manera exitosa. En este trabajo, se considera que las formaciones son muy relevantes en el soccer robótico pues aportan información estratégica del equipo y a su vez, ofrecen el soporte adecuado para favorecer la implementación de tácticas acordes a la situación del juego.

Scerri e Ydrén [Scerri and Ydrén, 2000] implementaron un sistema que permitía a un experto introducir manualmente a un equipo de soccer robótico las formaciones y acciones que debería ejecutar el equipo, tal como se haría en un pizarrón antes del juego. Lo que hicieron fue diseñar un mecanismo para que, con base en una situación dada en el juego (como un tiro de esquina, defensa, o ataque), el experto especificaba la localización deseada de los jugadores así como la dirección en que los agentes debían driblar o dar pases si tenían el balón. Para ello, dotaron a cada agente con una arquitectura que usa un *árbol de activación*. Este árbol está ordenado para representar desde ideas de alto nivel como el estado actual del juego, hasta el comportamiento y la posición escogida para cada agente y finalmente acciones de más bajo nivel como patear, dar pases, driblar, etc. En cada nivel del árbol, se selecciona el nodo activado más alto (situación de juego). La selección del comportamiento altera cómo el nivel de activación de nodos de acción varían con el estado del mundo. En el tiempo de ejecución, se selecciona el

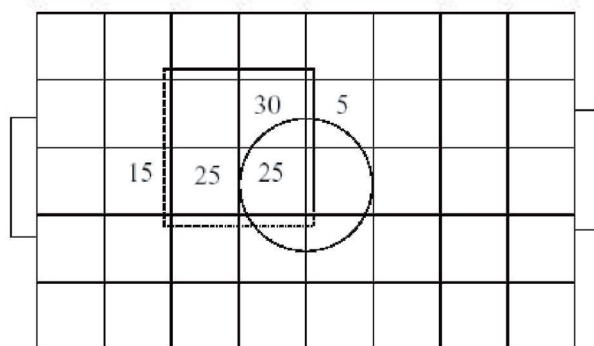


Figura 2.7: Los valores en las celdas representan el porcentaje de ocupación en esa zona de la cancha por parte del oponente

nodo más alto activado y se llevan a cabo las acciones determinadas en las hojas del sub-árbol correspondiente. El sistema toma ventaja de una gran cantidad de conocimiento disponible en el dominio del soccer, sin embargo no permite acciones cooperativas que puedan enlazarse una con otra para formar jugadas interesantes, está limitado a una sola acción dada una situación de juego. Otra limitación de este enfoque es que los jugadores sólo pueden cambiar de rol durante los cambios de estado.

2.6. Descubrimiento de Tácticas

En muchos casos, no es fácil comprender la diferencia entre *estrategias* y *tácticas*. Dado que en esta tesis se emplean los dos conceptos, a continuación se retoma la definición de estos conceptos en el ámbito de relaciones públicas y en el dominio militar. En el caso de las relaciones públicas, por un lado las estrategias abarcan la *fotografía completa* o el plan general que se llevará a cabo trazando objetivos y metas para una campaña. Por otro lado, las tácticas son actividades específicas creadas y seleccionadas para alcanzar los objetivos previamente trazados en las estrategias [Miyamoto, 2002].

En dominios militares, las tácticas (del griego *taktike*, el arte de organizar un ejército) son el nombre colectivo para los métodos que se emplean para vencer a un enemigo en la batalla. Las tácticas deben distinguirse de las estrategias militares en que, éstas últimas tienen más que ver con el plan general para lograr un resultado a largo plazo, es decir, el comportamiento global que tendrá el ejército durante la guerra [DOD, 2001]. Entre las tácticas y estrategias se encuentra el *arte operacional*, el cual es una metodología que tiene como objetivo, convertir una estrategia en una serie de misiones tácticas.

Con base en las definiciones anteriores de estrategia y táctica, se resume que las estrategias son planes que determinan los objetivos y comportamientos globales de las entidades participantes mientras que las tácticas son los métodos o acciones determinadas para alcanzar los objetivos previamente trazados en las estrategias.

En el caso específico del soccer robótico, se tiene:

- *Estrategia*: Es un plan que determina el comportamiento global de un equipo de agentes de soccer. La estrategia implica a todo el equipo de jugadores.

Por ejemplo, una estrategia ofensiva determina el comportamiento de los defensas, medios y delanteros cuando el equipo controla el balón y se dirige hacia la portería del oponente.

- *Táctica*: Es el conjunto de acciones encaminadas a lograr el plan estratégico. Una táctica involucra a un subconjunto de jugadores.

Por ejemplo, bajo una situación ofensiva, una táctica es expresada como una secuencia de pases que se realiza entre 3 o 4 jugadores para hacer una jugada de gol.

La sección anterior discute el tema de reconocimiento de planes o estrategias en equipos multi-agente; ahora se presentan algunos trabajos relacionados al análisis táctico de agentes.

Uno de los primeros trabajos para descubrir el comportamiento táctico es la propuesta de Kaminka y colegas [Kaminka et al., 2002] donde presentan una forma de aprender el comportamiento secuencial coordinado de equipos. La idea es que a partir de observaciones temporales multi-variable se descubran comportamientos coordinados del equipo de agentes. Los autores usan un conjunto de reconocedores de comportamiento que se enfocan solamente en reconocer comportamientos básicos de los agentes (por ejemplo pases o dribles). Así, mientras el equipo controle el balón, se ordenan las acciones que llevan a cabo asociándolas a los agentes participantes. Un ejemplo de una secuencia es:

$$Pase(Jugador1, Jugador2) \rightarrow Pase(Jugador2, Jugador3) \rightarrow Drible(Jugador3) \rightarrow \dots$$

Que indica que el Jugador1 le da un pase al Jugador2, luego, el Jugador2 le da un pase al Jugador3 y finalmente, el Jugador3 dribla.

Posteriormente, estas secuencias son representadas en un *trie* (una estructura de datos como un árbol) como se muestra un ejemplo en la Figura 2.8. En este ejemplo, se supone que se cuenta con dos secuencias: ABCA y BCAD, donde A, B, C y D son diferentes eventos. Suponiendo que se inicia con un trie vacío, se inserta la primera secuencia ABCA (Figura 2.8(a)). Luego se insertan los tres sufijos restantes de ABCA: {BCA, CA, A}, resultando en la Figura 2.8(b). Note que si ya existe un nodo igual al que se va a insertar, no se crea un nuevo nodo sino que sólo se incrementa el contador de frecuencia de ese nodo (el número entre paréntesis). Luego, se inserta la secuencia BCAD y sus sufijos {BCAD, CAD, AD, 0} dentro del trie resultando en la Figura 2.8(c). Se observa que la secuencia de un solo elemento más común es A, la secuencia más común de dos eventos es CA y la secuencia más común de tres elementos es BCA.

De esta manera, a partir del trie es posible observar aquellas secuencias o subsecuencias más comunes que pueden ser identificadas como comportamientos secuenciales del equipo.

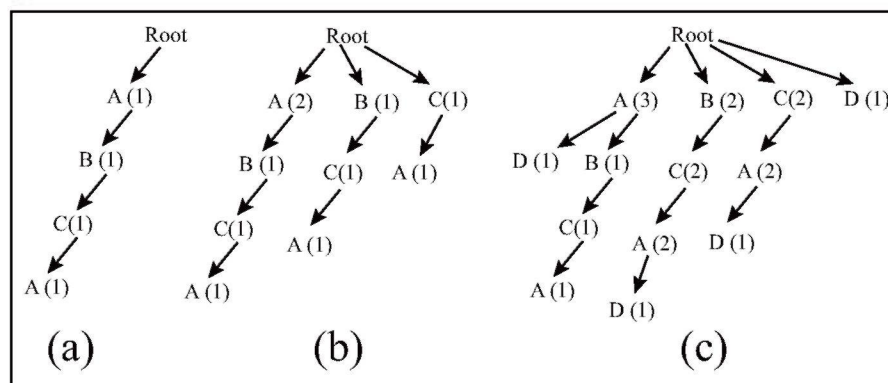


Figura 2.8: Ejemplo de un trie. Los números son la frecuencia de ocurrencias.

En el trabajo de Kaminka se analizan patrones de jugadas entre dos o tres jugadores sin tomar en cuenta el comportamiento estratégico del equipo en el cual ocurren las jugadas. No consideran las relaciones entre los jugadores. En el trabajo presentado en esta tesis, se consideran a las jugadas tácticas inmersas en una formación estratégica que no debe pasarse por alto ya que, por un lado, las estrategias enriquecen la interpretación de patrones tácticos al proveer un contexto más expresivo de información y por otro lado, las estrategias también proveen una plataforma o base sobre la cual surgen las posibles tácticas de ataque o defensa.

Raines y colegas en [Raines et al., 1999, Tambe et al., 2000] desarrollaron un sistema llamado ISAAC. Esta es una herramienta que está enfocada para que los diseñadores de equipos de soccer simulado puedan medir el desempeño de sus equipos y así detectar aquellas debilidades que le son difíciles de observar y analizar en tiempo real (ver Figura 2.9).

ISAAC analiza juegos de fútbol soccer simulado fuera de línea para sugerir consejos de cómo puede mejorar el desempeño del equipo. Sin embargo, estos consejos son sólo comprendidos por los diseñadores del equipo en lugar de los agentes mismos. ISAAC analiza un juego usando un algoritmo de árboles de decisión (C5.0) para generar reglas en las situaciones donde los jugadores individuales o en equipo han tenido o han podido tener éxito. Para ello, se enfoca en el análisis de situaciones que ellos llaman *eventos clave*. Los eventos claves son aquellas situaciones que pueden afectar directamente el resultado del juego y específicamente se analizan los tiros a la portería del oponente. Las reglas producidas por ISAAC se pueden usar para mejorar el desarrollo del equipo analizado mostrando qué tan exitoso es el equipo en ciertas situaciones y en cuáles puede mejorar. Sin embargo, este sistema no toma en cuenta un análisis que permita descubrir cómo es el comportamiento de los jugadores bajo ciertas situaciones del juego, en su lugar, se limita a crear modelos para medir el rendimiento y sugerir mejoras de desempeño del equipo. Por esta razón, este sistema tampoco puede cambiar el comportamiento del equipo automáticamente.

Andraz Bezek y colegas presentan un algoritmo que transforma una secuencia de

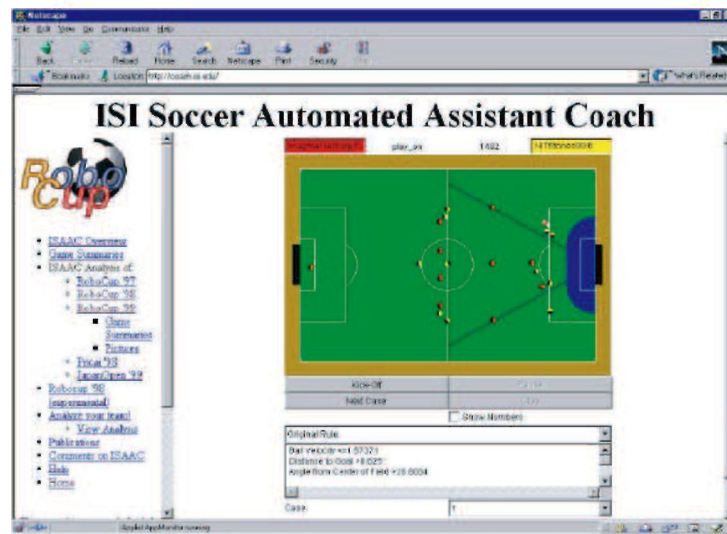


Figura 2.9: Sistema ISAAC. Analizador de equipos de soccer robótico

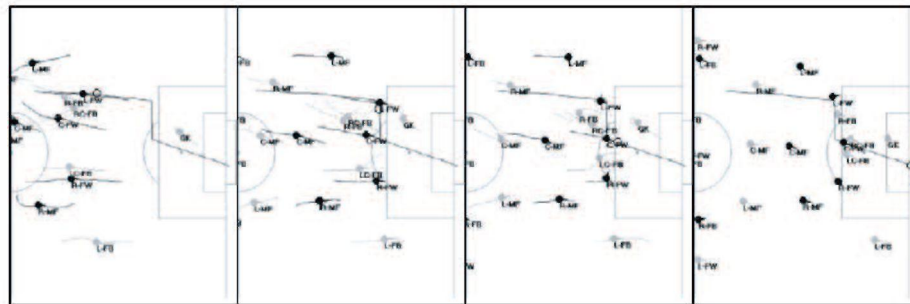


Figura 2.10: Cuatro frames mostrando una secuencia de ataque exitosa.

acciones de los jugadores de soccer en un conjunto de descripciones de acciones tácticas en forma gráfica [Andraz Bezek and Bratko, 2006]. Por ejemplo, en la Figura 2.10 se presenta un conjunto de 4 ciclos o frames que muestran una situación de ataque. Los jugadores son los círculos rellenos. El equipo atacante es negro mientras que el equipo que defiende es gris. El balón es el círculo blanco con borde negro; se muestra con una línea la trayectoria del balón a través de la secuencia. La secuencia de acciones en la Figura 2.10 podría ser descrita por un humano como: *El delantero izquierdo del equipo izquierdo dribla desde la media cancha hasta el área de penalti. Luego hace un pase al delantero central en el centro del área de penalti donde el receptor tira y anota un gol en la parte derecha de la portería.* Esta secuencia es representada por Bezek en la Figura 2.11 la cual consiste en los jugadores involucrados en la jugada, las acciones que llevan a cabo y posiciones de los jugadores implicados. El objetivo principal del trabajo de Bezek está dirigido a obtener una descripción de las jugadas de gol de un equipo en términos como la narración descrita arriba por un humano.

El trabajo de Bezek es relevante en el área de descubrimiento de patrones de com-

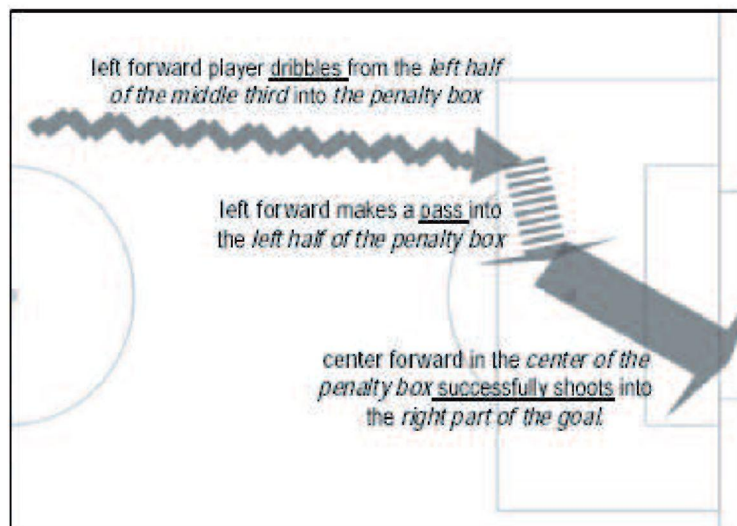


Figura 2.11: Ejemplo de una jugada táctica: ataque directo por el lado izquierdo.

portamiento a nivel táctico. Sin embargo, una de las limitaciones que se pueden citar de su trabajo es que sólo toma en cuenta a los agentes implicados en la jugada, es decir, aquellos jugadores que participaron directamente enviando y recibiendo un pase. Tomando en cuenta algunos otros jugadores que intervinieron en la jugada, aunque sea de manera indirecta pero que están muy relacionados con los primeros, la representación de los patrones puede ser mucho más rica. Por tal razón, en esta tesis se resalta la importancia de considerar el comportamiento táctico enmarcado dentro de un contexto estratégico que está determinado principalmente por la estructura general del equipo.

El artículo de Hirano y Tsumoto [Hirano and Tsumoto, 2004] presenta un método para encontrar patrones de pases de partidos de soccer previamente jugados. Toman en cuenta 2 características de la secuencia de pases: irregularidad temporal y requerimientos para la observación multiescala. En su trabajo, desarrollaron un método de comparar secuencias basado en comparación multiescala. El método es usado con Clustering jerárquicos para producir grupos de patrones de pases espaciales. Sus resultados experimentales demostraron que el método es capaz de producir algunos patrones de pases interesantes que están asociados con los goles. El trabajo de Hirano está más basado en buscar patrones de pases pero no toma en cuenta conocimiento del dominio ni la estrategia general del equipo.

Sukthankar y Sycara [Sukthankar and Sycara, 2005] presentan un trabajo de minimización de costos en el problema de reconocimiento del comportamiento humano en el dominio militar. Usando datos de movimientos de personas, el sistema es capaz de reconocer los comportamientos que un ser humano lleva a cabo. La clasificación del movimiento a bajo nivel es ejecutada usando máquinas de soporte vectorial (SVM) y un modelo oculto de Markov (HMM); la salida del clasificador es usada como una característica de entrada para el reconocedor del comportamiento. Dada la naturaleza

reactiva y dinámica del dominio, su sistema es capaz de manejar secuencias de comportamiento que son frecuentemente interrumpidas e intercaladas. Su propuesta identifica diferentes comportamientos usando una biblioteca de comportamientos previamente definidos. En esta tesis se propone un trabajo que descubre comportamientos tácticos sin contar con una biblioteca predefinida.

Capítulo 3

Modelo Propuesto

3.1. Introducción

3.2. Retos Computacionales

Las siguientes secciones describen los principales retos que se enfrentan para reconocer el comportamiento de un equipo de soccer robótico.

3.2.1. Características del ambiente

Las características del dominio del soccer robótico son:

- Toma de decisiones en tiempo real. Constantemente los agentes deben tomar decisiones como la de dar un pase o tirar a gol, atacar o defender, etc. donde se requiere de una rápida respuesta. Como consecuencia, el ambiente se torna altamente dinámico. En la siguiente sección se profundiza más este punto.
- Un ambiente competitivo. En este dominio, es necesario considerar que existe un equipo oponente con objetivos antagónicos.

Los sensores (visual y auditivo) de los jugadores contienen ruido, por tal razón no siempre perciben adecuadamente su ambiente ni tampoco se ejecutan siempre sus acciones de la misma forma.

La comunicación entre los jugadores no es completamente efectiva, debido al bajo ancho de banda con que cuentan y el ruido inmerso en el canal de comunicación.

3.2.2. El dinamismo del soccer

El ambiente del soccer es un mundo dinámico, es decir, tanto los jugadores como la pelota permanecen en constante movimiento, lo que impide reconocer fácilmente

estrategias y tácticas que podrían concebirse en universos estructurados. Por ejemplo, un juego de Soccer dura aproximadamente 10 minutos, los cuales equivalen a 6000 ciclos de simulación donde cada ciclo tiene una duración de 100 milisegundos. En cada ciclo, cada uno de los jugadores tiene oportunidad de decidir qué acción llevará a cabo entre un conjunto de acciones básicas como correr, dar vuelta o patear el balón.

A pesar del gran dinamismo que existe en el soccer robótico, la experiencia nos muestra que tanto en el soccer real de humanos como en el soccer robótico, la mayoría de los equipos juegan con estructuras definidas o formaciones, las cuales inducen un comportamiento organizado en los jugadores [Visser et al., 2000, Yuen, 2002]. El enfoque de este trabajo es sobre equipos que juegan respetando estructuras de equipo con base en una distribución de zonas llamadas: *Defensiva (D)*, *Media (M)* y de *Ataque (A)*, tal como el soccer real.

Los jugadores están en constante movimiento y temporalmente rompen su alineación como jugadores pertenecientes a su zona (por ejemplo, un defensa que corre a la zona central de la cancha y se despega de sus compañeros defensas). En la mayoría de los casos, éste es un comportamiento común que no debe traducirse como un cambio de formación necesariamente, sino como una variante de la formación de equipo. Por tal razón, se debe ser capaz de manejar los cambios constantes con una representación explícita de las relaciones entre los jugadores que pueda resultar en una forma eficiente de reconocer formaciones sometidas a un ambiente dinámico.

Si el fútbol soccer no fuera dinámico, entonces a partir de un sólo ciclo se podrían identificar qué agentes pertenecen a cada zona del equipo. Sin embargo, como éste no es el caso, existe un problema donde no es suficiente analizar un solo ciclo para determinar la formación que juega un equipo; en su lugar, es preciso considerar una secuencia de ciclos a fin de determinar patrones potenciales que expresen el comportamiento general del equipo.

3.2.3. Múltiples Relaciones

Otro reto para modelar el comportamiento de un equipo de soccer es la necesidad de contar con modelos de representación robustos capaces de manejar los cambios constantes de movimiento de los jugadores. Para ello, se debe considerar un manejo adecuado del conjunto de relaciones entre los jugadores. La Figura 3.1 muestra el conjunto de todas las relaciones posibles entre los jugadores de un equipo de once jugadores (el portero no aparece en la figura). Hay un total de 90 relaciones obtenidas por la expresión:

$$n(n - 1)$$

donde n representa el número de jugadores. Esta formula considera dos relaciones para cada par de jugadores. Por ejemplo, el jugador A está relacionado con el jugador B y el jugador B esta relacionado con el A . Como se observa en la figura, manejar un modelo de representación con tantas relaciones es una tarea complicada debido

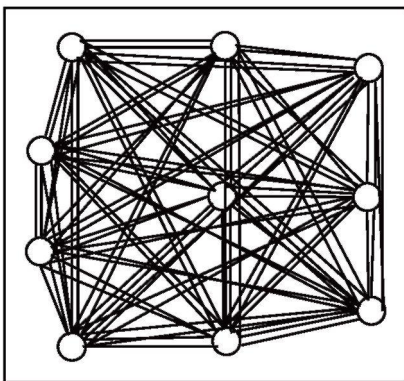


Figura 3.1: Conjunto de relaciones posibles entre jugadores de un equipo

al movimiento continuo que los jugadores llevan a cabo durante el juego, provocando cambios constantes en la estructura del equipo e impidiendo reconocer la formación con la cual el equipo está jugando. Una solución es representar solo aquellas relaciones que son relevantes debido a su alto índice de interacción entre los jugadores. Más adelante se presentan detalles de este enfoque.

3.3. Modelo de Representación

A continuación se presenta el modelo de representación propuesto en este trabajo.

3.3.1. Modelo multicapas propuesto

Para comprender mejor el comportamiento de un equipo de soccer, es necesario contar con un modelo de representación que tome en cuenta los siguientes aspectos:

Facilitar el reconocimiento de patrones de comportamiento a nivel de formaciones de equipo. Como se discutió anteriormente, la información que aporta la estructura o formación de un equipo, es un elemento relevante que contribuye a comprender el comportamiento del equipo [Ayanegui and Ramos, 2007]. Requiere no solo del reconocimiento de las formaciones sino también de la identificación de aquellos elementos principales que influyen en los cambios de formación del equipo.

- Permitir el descubrimiento de patrones de comportamiento a nivel táctico. Dado que el código de una formación ($D:M:A$) no es capaz de expresar la interacción que ocurre entre un subconjunto de jugadores, es necesario contar con un modelo que represente, a un nivel más detallado, las interacciones que ocurren entre los agentes que participan en una jugada, la cual, a su vez, se encuentra contextualizada por una formación de equipo [Ramos and Ayanegui, 2008b]. De esta manera, el descubrimiento de jugadas tácticas no deberá estar aislado de las formaciones

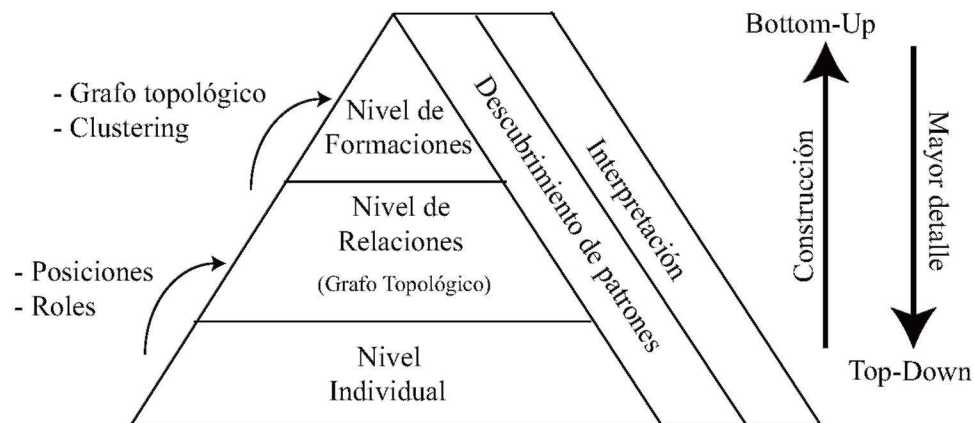


Figura 3.2: Modelo de representación propuesto

bajo las cuales ocurren tales jugadas, en su lugar, es necesario tomar en cuenta la formación del equipo para descubrir e interpretar adecuadamente los patrones de comportamiento tácticos.

Ser capaz de descubrir qué agente, de manera individual, es el detonador de una variante a nivel de formación o a nivel táctico.

El modelo de representación multicapas que se propone aquí contempla tres niveles de representación del comportamiento y en cada nivel se descubren e interpretan patrones como se observa en la Figura 3.2.

La flecha bidireccional indica la posibilidad de navegar en el modelo tanto en dirección ascendente (*bottom-up*) como descendente (*top-down*).

Las flechas que van de cada nivel inferior al superior indican que la construcción del modelo se realiza de abajo hacia arriba (*bottom-up*), ya que los datos individuales de los jugadores se utilizan para construir relaciones entre ellos, las cuales a su vez, forman estructuras que soportan formaciones de equipo.

La navegación *bottom-up* permite que, al contar con información individual de los jugadores, sea posible ascender al nivel de relaciones a fin de caracterizar el comportamiento individual de los jugadores e interpretar más ricamente su comportamiento dado el contexto de interacción con sus compañeros. A su vez, las relaciones que se representan en el segundo nivel permiten descubrir e interpretar jugadas tácticas incorporando conocimiento del dominio que se encuentra en el nivel de formaciones. Como se mencionó anteriormente, las jugadas tácticas siempre ocurren bajo una formación de equipo.

En las siguientes secciones, se explica más detalladamente en qué consiste cada uno de los niveles de representación y más adelante, se ilustra la forma en que se reconocen patrones de comportamiento estratégicos y tácticos.

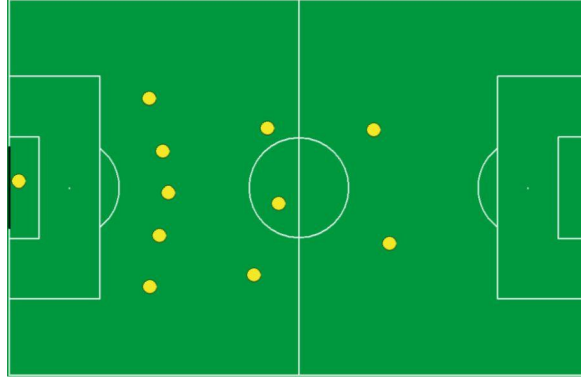


Figura 3.3: Una formación 5:3:2

3.3.2. Nivel de formaciones

Las formaciones son la manera en la cual un equipo de soccer alinea sus zonas defensiva, media y de ataque durante un partido. Cuando se habla de formaciones, primero se listan a los defensas, luego a los medios y por ultimo a los delanteros. Por ejemplo, un código 5:3:2 representa una formación compuesta de cinco defensas, tres medios y dos delanteros (ver Figura 3.3). Como en el soccer real, el portero no es considerado como parte de la formación.

La formación es una estructura que tienden a respetar el equipo y esta definida por la relación que existe entre las zonas defensiva, media y ofensiva de un equipo. Las formaciones del equipo revelan la estrategia general del equipo [Visser et al., 2000, L. P. Reis and Oliveira, 2000], es decir, a través de reconocer la formación del equipo se puede saber que tan defensivo u ofensivo es el equipo. Además, la formación aporta información relevante para descubrir patrones en las jugadas del equipo. Por esta razón, el reconocimiento de formaciones representa una fuente valiosa para comprender el comportamiento del equipo.

El número de formaciones posibles que se pueden tener en un equipo de soccer se puede calcular con la fórmula 3.1 para combinaciones con repetición:

$$CR_m^n = \left[\begin{matrix} m+n-1 \\ n \end{matrix} \right] = \frac{(m+n-1)!}{n!(m-1)!} \quad (3.1)$$

Donde n es el número de elementos a combinar y m es el número de grupos disponibles donde se clasificara. Para el caso de la liga de 11 vs. 11 jugadores, $n=10$ jugadores (el portero no cuenta) y $m=3$ zonas (defensiva, media y ofensiva). Por lo tanto, el número de formaciones posibles usando la fórmula 3.1 es 66:

$$CR_m^n = \left[\begin{matrix} 3+10-1 \\ 10 \end{matrix} \right] = \frac{12!}{10!2!} = 66 \quad (3.2)$$

La Tabla 3.1 muestra las 66 formaciones posibles.

Generalmente, un equipo mantiene una formación básica que busca respetar durante el partido y ejecuta variantes de esa formación que se encuentran en la vecindad de ésta. Por ejemplo, las formaciones 4:4:2 y 4:3:3 son formaciones vecinas de la formación 4:2:4 en la Tabla 3.1.

La Tabla 3.1 muestra el conjunto de formaciones que pueden generarse para equipos con once jugadores, donde D es la zona defensiva, M la zona media y A la zona ofensiva o de ataque.

Las formaciones defensivas son comunes en situaciones donde el equipo va ganando el partido por un gol y esta a punto de terminar el juego. Bajo estas condiciones, el equipo busca proteger a toda costa su ventaja en el marcador.

Las formaciones ofensiva-defensivas se caracterizan por jugar con una estrategia sin mucho riesgo.

Las formaciones de la 4:6:0 a la 4:1:5 pueden ser usadas como posibles variantes de formaciones clásicas 4's.

Formaciones clásicas: 4:4:2, 4:2:4 y 4:3:3. Sus vecinos pueden ser usados como variantes de estas formaciones.

Las formaciones ofensivas son usadas principalmente en los últimos minutos del partido cuando el equipo va perdiendo y hay desesperación en los jugadores para anotar un gol.

Las formaciones muy ofensivas son usadas en situación más desesperantes. Cuando el equipo busca anotar un gol a toda costa, arriesgándolo todo.

Por lo general, un código de formación del tipo $D:M:A$, representa únicamente el número de jugadores en las zonas defensiva, media y ofensiva del equipo. Aunque este código expresa la relación de pertenencia de los jugadores a las zonas del equipo, no es capaz de expresar información más detallada de las formaciones, tal como qué jugadores pertenecen a cada zona y la manera en que éstos se encuentran relacionados. Con el fin de proporcionar un contexto basado en la estructura del equipo al proceso de descubrimiento de comportamientos tácticos, en este trabajo de tesis se redefine el concepto de formación de equipo como se describe a continuación:

Una formación de equipo es una estructura que representa las relaciones entre los jugadores de la misma zona y entre jugadores de la(s) zona vecina(s).

La forma en que se construye la estructura de una formación se encuentra definida en el nivel relacional del modelo propuesto en la Figura 3.2. En la siguiente sección se explica con más detalle la forma de expresar las relaciones entre los jugadores.

3.3.3. Nivel relacional

En este nivel del modelo se encuentra la representación de las relaciones entre los jugadores del equipo.

Interacciones y Relaciones

Jaques Ferber define el concepto de interacción como aquella que ocurre cuando dos o más agentes se encuentran envueltos en una relación dinámica a través de la cual se llevan a cabo un conjunto de acciones recíprocas [Ferber, 1999]. Además, las interacciones no son sólo consecuencia de las acciones que se llevan a cabo por varios agentes, sino que al mismo tiempo, las interacciones son necesarias para establecer la estructura de la organización dentro de un SMAC [Ferber, 1999]. En otras palabras, es a través del intercambio, acuerdos y la influencia que los agentes ejercen sobre los demás, que los agentes se convierten en organizaciones con capacidad de realizar el trabajo en equipo.

Una organización o grupo de agentes es una *estructura* formada por un conjunto de agentes [Ferber et al., 2000]. Las interacciones que surgen entre dos o más agentes de esta organización, son el resultado de las relaciones que existen entre los agentes [Sycara, 1998, Wooldridge et al., 2000]. Por esta razón, en este trabajo de tesis resaltamos la importancia de modelar explícitamente las relaciones entre los jugadores a fin de facilitar el descubrimiento de patrones en las jugadas tácticas.

Por un lado, modelar las relaciones entre los jugadores de un equipo de soccer puede representar una tarea difícil debido a las condiciones dinámicas del juego, llevando a crear modelos complicados de manejar debido a las múltiples relaciones entre los jugadores. Por otro lado, si se representan las relaciones relevantes entre los jugadores, se pueden crear modelos capaces de representar las interacciones o jugadas principales que ocurren entre los jugadores. Para comprender la diferencia entre una relación relevante de otra que no lo es, Michel, Gouaich y Ferber discuten la diferencia entre una *interacción fuerte* y una *interacción débil* [Michel et al., 2003]:

- *Interacción fuerte*: Ocurre cuando la viabilidad de la acción de un agente depende de la acción de otro agente.

Interacción débil: Ocurre cuando la viabilidad de la acción de un agente no depende de la acción de otro agente.

Los autores argumentan que las interacciones fuertes necesitan un manejo explícito de la interacción misma, mientras que las interacciones débiles no la requieren.

De lo anterior se puede concluir que si dos o más de los agentes tienen una interacción fuerte, sus relaciones deben expresarse explícitamente. Por otro lado, para los agentes que tienen una interacción débil, sus relaciones pueden no estar representadas. En este trabajo, a las relaciones fuertes se les llama relaciones relevantes.

Relaciones Relevantes

En este trabajo de tesis, las relaciones relevantes son determinadas por la estructura que representa la distribución de los agentes de soccer en una formación de equipo. De esta manera, las relaciones construyen una estructura que puede representarse por un

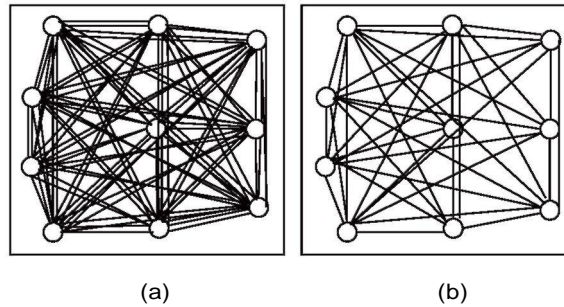


Figura 3.4: Relaciones entre jugadores. (a) 90 relaciones y (b) 45 relaciones

grafo, donde los nodos representan la posición de los agentes de soccer y los arcos las relaciones entre ellos.

La Figura 3.4(a) muestra las relaciones de los agentes de soccer para una formación 4:3:3 (cuatro defensas, tres medios y tres delanteros), donde por cada par de jugadores hay dos relaciones unidireccionales que indican una posible interacción entre ellos. Tomando en cuenta relaciones bidireccionales, se puede eliminar una relación por cada par de jugadores para obtener 45 relaciones. La Figura 3.4(b) ilustra estas 45 relaciones. A pesar de que ha sido reducido el número de relaciones, el modelo sigue siendo complicado para el seguimiento adecuado de las formaciones y jugadas tácticas.

Para facilitar más el modelo de representación de relaciones, en este trabajo de tesis se propone un modelo basado en grafos topológicos cuya estructura base es el triángulo, debido a su robustez a fin de representar las relaciones relevantes entre los jugadores [Ramos and Ayanegui, 2008a]. Una relación relevante es aquella que usa un agente de soccer en situaciones de intercambio de pases y posiciones en una forma estratégica. De esta manera, un agente estará relacionado con su vecino perteneciente a la misma zona (defensiva, media o de ataque) y con su vecino más cercano perteneciente a la zona vecina. El objetivo es construir una estructura simple pero robusta con base en relaciones relevantes capaces de manejar la naturaleza dinámica del juego en una forma topológica, con base en sub-grafos triangulares que son construidos como se indica a continuación:

- Paso 1. Relaciones entre agentes vecinos de la misma zona. Sea $Z_i = \{a_i < b_i < \mathbf{O}_i < \dots < n_i\}$ la zona i del equipo con n jugadores, donde el jugador a_i se localiza en lo alto (coordenada en y menor) de la zona Z_i y el jugador n_i en la parte baja (coordenada en y mayor) de la zona. El resto de los nodos están localizados en posiciones intermedias. Así, a_i está enlazado con b_i , b_i está enlazado con \mathbf{O}_i y así sucesivamente. Este proceso se realiza para las zonas defensiva (D), media (M) y de ataque (A) del equipo (ver Fig. 3.5(a)).

Paso 2. Relaciones entre agentes vecinos de zonas vecinas. Una vez que los nodos de las zonas han sido enlazados, los nodos de las zonas vecinas, con base en

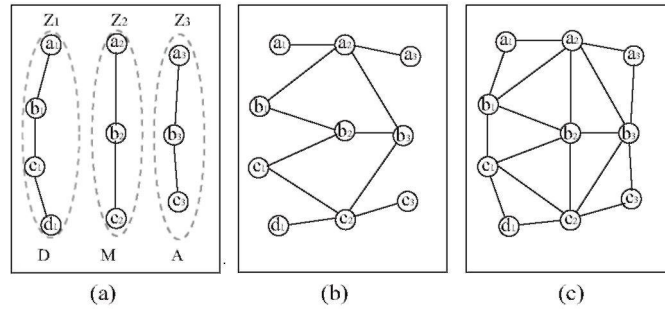


Figura 3.5: Construcción de un grafo planar: (a) Paso 1, (b) Paso 2, (c) Grafo planar obtenido

distancias mínimas, son enlazados hasta que se crea un grafo planar (ver Fig. 3.5(b)).

La Figura 3.5(c) muestra el grafo planar representado por subgrafos triangulares como resultado de aplicar los dos pasos anteriores.

El número total de relaciones de un grafo, el cual ha sido construido con base en el método descrito arriba, se calcula de la siguiente forma:

Primero, el total de relaciones entre los jugadores de una sola zona Z_i es:

$$|Z_i| - 1 \tag{3.3}$$

donde $|Z_i|$ representa el número de jugadores en la i -ésima zona del equipo (defensa, media o de ataque). Por ejemplo, para la zona defensiva (Z_1) en la Figura 3.5 hay 4 jugadores, por lo tanto, según la expresión 3.3 hay en total 3 relaciones entre los jugadores defensas.

De la expresión 3.3 se sabe que el total de relaciones en las tres zonas del equipo está dado por:

$$|Z_1| + |Z_2| + |Z_3| - 3 \tag{3.4}$$

donde $|Z_1|$ es el número de jugadores en la zona defensiva, $|Z_2|$ el número de jugadores en la zona media y $|Z_3|$ el número de jugadores en la zona de ataque.

Ahora se calcula el total de relaciones entre zonas vecinas. El número de relaciones entre las zonas defensiva y media es:

$$|Z_1| + |Z_2| - 1 \tag{3.5}$$

De igual forma, el número de relaciones entre las zonas media y de ataque son:

$$|Z_1| + |Z_2| - 1 \quad (3.6)$$

Por lo tanto, sumando las expresiones 3.5 y 3.6 se obtiene:

$$2|Z_2| + |Z_1| + |Z_3| - 2 \quad (3.7)$$

Luego, sumando las expresiones 3.4 y 3.7 se obtiene:

$$\begin{aligned} 2|Z_1| + 3|Z_2| + 2|Z_3| - 5 \\ = 2(|Z_1| + |Z_2| + |Z_3|) + |Z_2| - 5 \end{aligned} \quad (3.8)$$

Dado que el número de jugadores en los equipos de soccer que analizamos es 11:

$$|Z_1| + |Z_2| + |Z_3| = 10$$

De la expresión 3.8 se tiene:

$$\begin{aligned} &= 2(10) + |Z_2| - 5 \\ &= |Z_2| + 15 \end{aligned} \quad (3.9)$$

De esta forma, bajo el esquema presentado en esta tesis, el número total de relaciones relevantes en una formación del tipo $D:M:A$ para equipos de once jugadores esta dado por la expresión 3.9; donde $|Z_2|$ es el número de nodos de la zona media.

Por ejemplo, para el caso de los jugadores en la Figura 3.5(c), donde hay 3 jugadores de la zona media ($|Z_2| = 3$), el número total de relaciones relevantes dado por la expresión 3.9 es 18.

La Tabla 3.2 compara el número total de relaciones entre jugadores para el caso de formaciones clásicas. Se compara el modelo que representa a todas las relaciones bidireccionales contra el modelo de relaciones relevantes propuesto en este trabajo de tesis.

La Tabla 3.2 muestra que al modelar únicamente las relaciones relevantes propuestas aquí, en promedio se reduce en 60 % el número de relaciones entre jugadores. Como se demuestra en el siguiente capítulo, el modelo de relaciones es sencillo pero robusto para representar las relaciones relevantes y facilita el descubrimiento de patrones a nivel relacional.

Las principales ventajas de usar el modelo de relaciones relevantes son:

1. Se reduce un 60 % del número de relaciones entre jugadores.

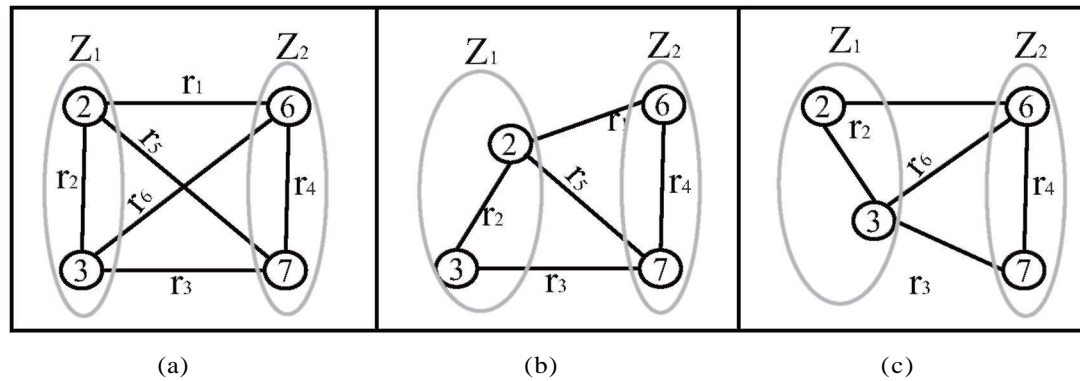


Figura 3.6: (a) Cuatro agentes relacionados, (b) Eliminando la relación r_5 , (c) Eliminando la relación r_6

2. Los sub-grafos triangulares son capaces de asumir un comportamiento topológico. Es decir, aún si una estructura está deformada debido a cambios posicionales de los nodos, la propiedad topológica de los grafos triangulares ayuda a preservar la estructura.

Grafo Topológicos

Un grafo planar topológico representa en este trabajo, las relaciones relevantes entre los jugadores de forma tal que no se intersecten tales relaciones, formando triángulos que soportan el dinamismo de los jugadores.

El criterio para *alambrar* a dos agentes que pertenecen a zonas vecinas es la distancia. Sin embargo, ya que la estructura que forman los agentes es dinámica, es común tener que *re-alambrar* la estructura con la misma formación sin que esto deba interpretarse como un cambio de formación. Por ejemplo, la Figura 3.6(a) muestra a cuatro agentes que están relacionados, donde los agentes 2 y 3 pertenecen a la zona 1 (Z_1) y los agentes 6 y 7 a la zona 2 (Z_2). Las relaciones r_2 y r_4 son relaciones entre agentes de la misma zona, mientras que las relaciones r_1 , r_3 , r_5 y r_6 son relaciones entre agentes de zonas vecinas. Aunque tanto la relación r_5 como r_6 son igualmente relevantes, para obtener un grafo planar debe *eliminarse* alguna de ellas. Es importante mencionar que la eliminación de alguna de estas relaciones no se debe a su falta de relevancia en la estructura sino más bien con el objeto de contar con un grafo planar. Por lo tanto, más que una eliminación de alguna de estas relaciones, es una selección por conveniencia. Dependiendo de la ubicación en la que se encuentren los agentes, se seleccionara ya sea la relación r_5 o la relación r_6 , dependiendo de qué relación entre los agentes sea la más cercana. La Figura 3.6(b) muestra la estructura que resulta al eliminar la relación r_5 y la Figura 3.6(c) muestra la estructura tras eliminar la relación r_6 . Ambas estructuras representan a la misma formación.

Con base en lo expuesto anteriormente, una formación, en este trabajo, se representa

como un grafo planar topológico en cualquiera de sus instancias, dependiendo de la ubicación de los nodos. La Figura 3.7 muestra todas las instancias de los grafos para la formación 4:3:3. De acuerdo a la situación, cualquiera de estos grafos puede representar a la formación 4:3:3 de igual forma, dependiendo de la ubicación de los agentes.

3.3.4. Nivel individual

El primer nivel del modelo propuesto representa información individual de los jugadores. La mayoría de esta información se puede obtener de manera directa a partir del sistema de simulación, el cual proporciona información de los jugadores en la cancha, tal como: posición del jugador en la cancha, número de la camiseta del jugador, ángulo de la cabeza del jugador y su velocidad. Sin embargo, hay información relevante de un jugador que no es proporcionada por el simulador, como el rol que éste desempeña en el equipo (defensa, medio o delantero). Esta información puede ser conocida a través de la capa superior del modelo o nivel de formaciones.

Los datos individuales de los agentes aportan información que puede ser empleada para construir modelos basados en el comportamiento individual de los agentes. Por ejemplo, usar un modelo para reconocer la zona de la cancha donde generalmente un agente se mueve con mayor frecuencia. De esta manera, varios trabajos han usado esta información del oponente para determinar un posicionamiento más estratégico de su equipo [Riley and Veloso, 2000, Riley and Veloso, 2001] aprovechando así las zonas de la cancha más descuidadas por el adversario.

En el nivel individual del modelo pueden descubrirse patrones de comportamiento para determinar la zona de la cancha que es recorrida por un agente comúnmente. Sin embargo, también pueden descubrirse patrones de aquellos agentes que rompen temporalmente su formación y su relación con otros agentes para actuar de una manera no convencional. Por ejemplo, el hecho de que un defensa con el balón cruce la media cancha y avance hacia la portería del oponente debido a que sus compañeros de la media y ofensiva se encuentran marcados y ésta es la única manera de avanzar.

El simulador de RoboCup maneja 360 grados para representar el ángulo entre dos objetos. Dado que este conjunto de valores es complicado de manejar, es necesario determinar el número de orientaciones posibles buscando un equilibrio para que sea, por un lado, lo suficiente para expresar la orientación de un objeto con respecto a otro y por el otro lado, no representar demasiados valores que impidan el manejo adecuado del espacio. Una buena caracterización emplea 8 orientaciones posibles (Figura 3.8(a)). Esta caracterización del espacio esta basado en el código de Freeman [Freeman, 1973].

Se puede hacer una mayor abstracción del espacio de orientaciones, por ejemplo, para un equipo que tiene su portería del lado izquierdo de la cancha, las direcciones 0,1 y 7 en la Figura 3.8(a) representan *derecha* y las direcciones 3,4 y 5 representan *izquierda*. Las direcciones 1,2 y 3 representan *arriba* y 5,6 y 7 *abajo*. Con esta nueva abstracción se tienen ahora 4 valores posibles: derecha, izquierda, arriba y abajo (ver Figura 3.8(b)).

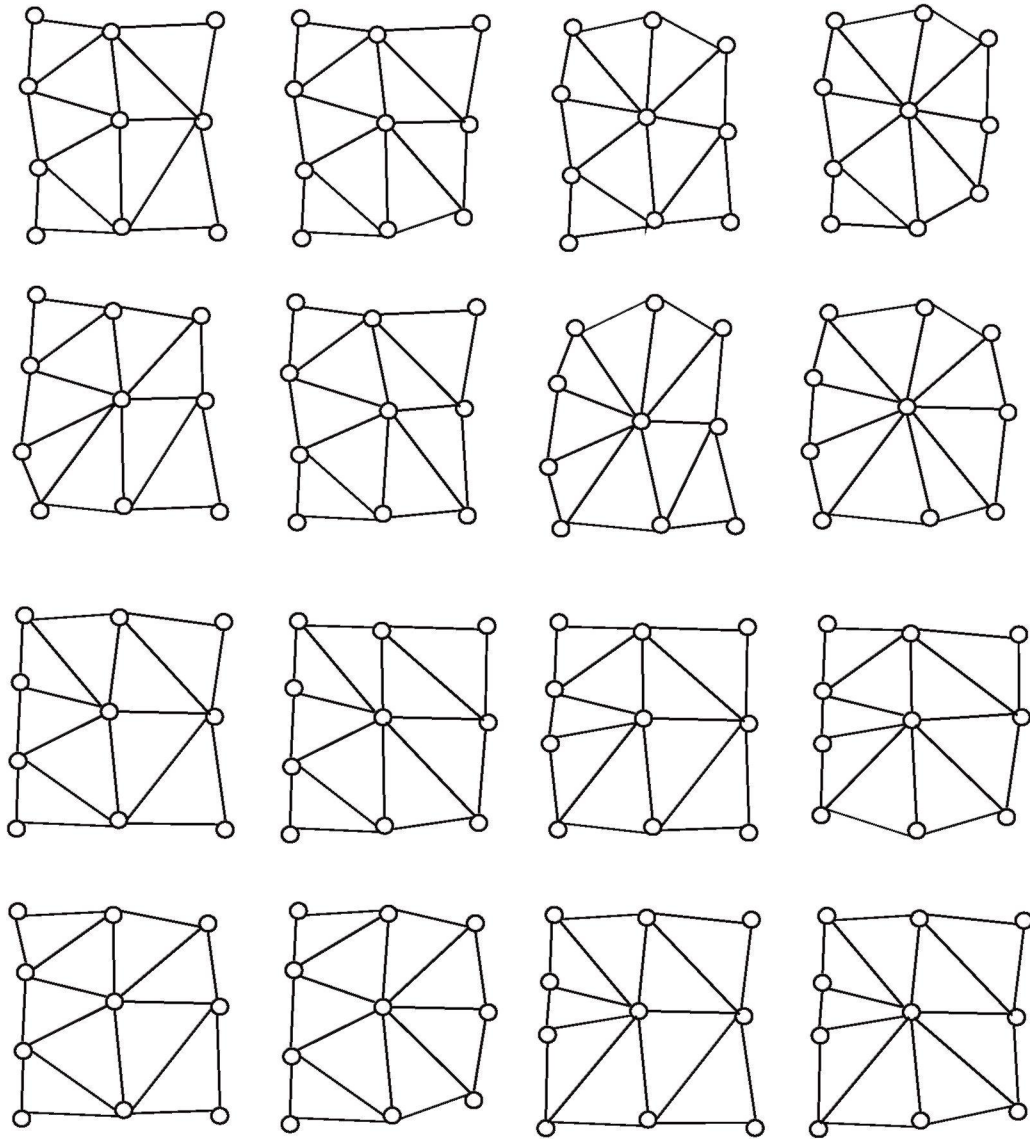


Figura 3.7: Dieciséis grafos planares posibles para la formación 4:3:3

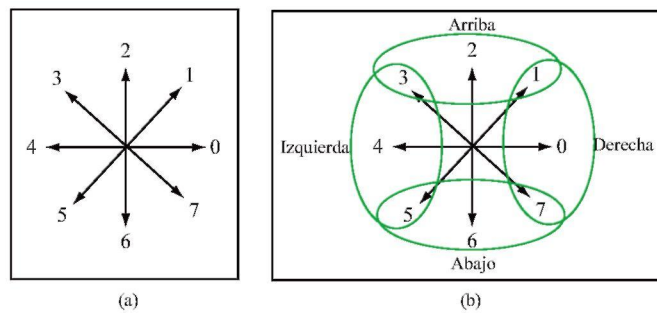


Figura 3.8: Configuración del espacio en a)8 valores y b)4 valores

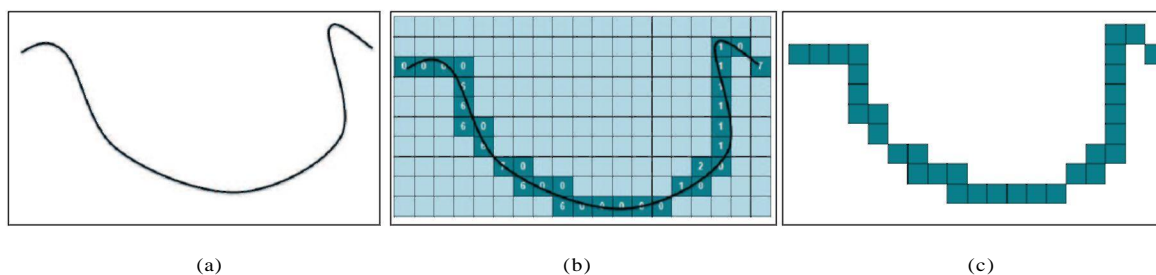


Figura 3.9: Discretización de una trayectoria en código Freeman

Las abstracciones de 8 y 4 valores son útiles en el nivel individual de representación para caracterizar el movimiento individual de un objeto, ya sea un agente o la pelota. El objetivo es realizar un seguimiento de la trayectoria recorrida en una situación dada. Por ejemplo, en la Figura 3.9(a) se muestra una trayectoria que puede ser el recorrido por un jugador o por el balón. En la Figura 3.9(b) se muestra que el código de Freeman para esa trayectoria correspondiente tiene una secuencia de 32 orientaciones: 00006660670600600000102011111107. En la Figura 3.9(c) se observa la trayectoria discretizada.

Entre mayor sea la resolución para discretizar una trayectoria, sería mayor la precisión y mayor la secuencia basada en el código de Freeman. Por otro lado, si es menor la resolución, la secuencia de Freeman será más corta pero la precisión será menor. Es necesario encontrar un equilibrio entre precisión y longitud de la secuencia del código de Freeman a fin de obtener un código sencillo pero a la vez suficiente para facilitar el descubrimiento de patrones de comportamiento. En los experimentos que se llevaron a cabo se obtuvieron buenos resultados usando una configuración de ocho valores posibles (0-7) (Ver Capítulo 4).

3.4. Modelo de Reconocimiento de Patrones

Esta sección presenta el modelo de reconocimiento de patrones de formaciones y el descubrimiento de tácticas de un equipo de soccer.

3.4.1. Reconocimiento de formaciones

Un patrón de comportamiento a nivel de formaciones es una estructura definida por un conjunto de relaciones entre jugadores que se repite en una secuencia de ciclos. En este trabajo, las formaciones representan la fuente principal para reconocer las estrategias de un equipo de soccer. Por lo tanto, un primer acercamiento para modelar el comportamiento de un equipo consiste en el reconocimiento de patrones de formaciones de equipo.

El objetivo es lograr el reconocimiento de patrones de equipo representado por diferentes formaciones jugadas por un equipo de soccer robótico durante un partido. Como

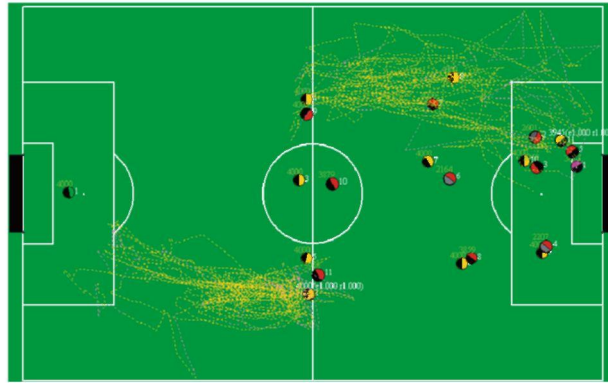


Figura 3.10: Ejemplo del recorrido del defensa derecho y delantero izquierdo

se ha señalado anteriormente, el reconocimiento de patrones de formación se dificulta debido a las condiciones dinámicas y en tiempo real del ambiente así como las múltiples interacciones entre los jugadores. Sin embargo, a pesar del alto grado de dinamismo en el movimiento de los jugadores, usualmente los equipos juegan en formas estratégica y organizada para respetar patrones predefinidos o formaciones [Kuhlmann et al., 2004, Visser et al., 2000, Kaminka et al., 2002, Wang Qinghua, 2000].

Debido a las condiciones dinámicas del juego, los jugadores están en constante movimiento y temporalmente rompen su alineación con sus compañeros que pertenecen a su misma zona. Manejar los cambios constantes sin una representación explícita de las relaciones entre los jugadores puede resultar en una forma ineficiente de reconocer formaciones sometidas a un ambiente dinámico. En la siguiente sección se explicarán cómo se reconocen las zonas y los jugadores pertenecientes a cada una de ellas.

Reconocimiento de zonas

Los equipos tienden a mantener una o varias formaciones durante el juego, es decir, cada jugador tiene una posición estratégica que define su rango de movimiento en el campo del soccer. El rol del jugador esta bastante relacionado con cierta área definida en la cancha dentro de la cual un jugador tiende a mantener sus recorridos. Por ejemplo, en la Figura 3.10 se observan las líneas punteadas que indican el recorrido en todo el juego del defensa derecho y el delantero izquierdo. Cualquiera de los comportamientos de un jugador depende de su rol actual. De acuerdo a la posición del jugador, los roles principales en el soccer robótico se pueden dividir en cuatro tipos:

Portero

Defensa

Medio

Delantero

Los diferentes roles se asocian con las diferentes posiciones y comportamientos que asumen los jugadores. Sin embargo, debido a las condiciones dinámicas cambiantes de un partido, un medio podría jugar el papel de delantero temporalmente conforme su equipo esté envuelto en un ataque. De manera similar, un delantero puede apoyar a la zona media o defensiva del equipo en una situación defensiva. De esta manera, aunque los jugadores tienen roles definidos claramente, en algunas situaciones de ataque o defensa, pueden mezclarse con compañeros de otra zona o rol para cumplir con algún objetivo. Sukthankar y Sycara llaman a esta situación *roles dinámicos* [Sukthankar and Sycara, 2006], pues los roles de un jugador cambian constantemente y por lo tanto un jugador puede tener comportamientos diferentes en un partido. Consecuentemente, el reconocimiento de patrones de formación se dificulta debido a las condiciones dinámicas y en tiempo real del ambiente. Sin embargo, es de suma importancia diferenciar entre un cambio temporal o permanente de rol, ya que sólo el segundo caso puede considerarse como un cambio real de formación. En otras palabras, tener la capacidad de diferenciar entre un cambio real y un cambio aparente de formación es crucial para comprender cómo opera un equipo, de lo contrario, el modelo de reconocimiento no será robusto al intentar caracterizar su comportamiento, ya sea a nivel estratégico o táctico.

Para reconocer el rol de un jugador, el primer paso que se lleva a cabo es reconocer la pertenencia de un jugador a una zona. Se puede asumir que las diferencias de posición dentro de un conjunto de jugadores de la misma zona (por ejemplo defensas), son menores que las diferencias entre zonas distintas [Andreas G.Nie, 2001]. Esto permite buscar grupos de jugadores a fin de caracterizar las zonas del equipo. En este trabajo, el algoritmo de clustering *K-meáns* [MacQueen, 1967] se aplica para reconocer esta primera etapa. *K-means* clasifica un conjunto de datos a través de cierto número de clusters (k clusters) definidos previamente. En este trabajo, $k=3$ tal que, las tres zonas serán definidas: defensiva, media y de ataque (D:M:A). A partir del archivo *.log*, los datos de un equipo se extraen y *K-means* es aplicado en cada ciclo de simulación del juego. Las posiciones de cada jugador, con respecto al eje x , se toman como la entrada del algoritmo de clustering y la salida del clustering es la clasificación, de acuerdo a la posición x , de todos los jugadores del equipo en los tres clusters. La Figura 3.11 muestra los 3 clusters reconocidos para un equipo de soccer que ataca hacia el lado derecho de la cancha. El primer cluster agrupa a 5 defensas, el segundo a 3 medios y 2 delanteros el último cluster. Debido al constante movimiento de los jugadores, no es factible concluir qué jugadores están en cada zona del equipo a partir de un sólo ciclo, se requiere de un período de tiempo para observar la tendencia que tienen los jugadores de agruparse en su zona correspondiente.

Si se aplica el algoritmo de clustering durante los primeros ciclos de simulación de un juego, el algoritmo es útil para determinar el número de jugadores que conforman a cada una de las tres zonas del equipo (D:M:A), sin embargo, el algoritmo de clustering no puede reconocer formaciones de equipo ya que, como se definió en la Sección 3.3.2, una formación está expresada por la estructura que representa la relación entre los jugadores del equipo, es decir, no sólo por cuántos jugadores conforman a cada zona

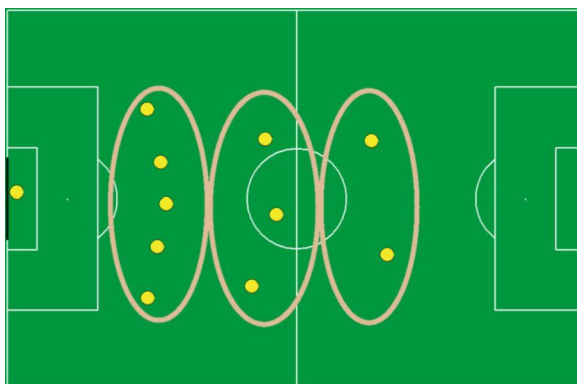


Figura 3.11: Ejemplo del algoritmo de clustering

del equipo.

El algoritmo de clustering es incapaz de representar la manera en que los jugadores se encuentran relacionados entre sí. Además, el algoritmo de clustering es muy sensible a cambios pequeños de movimiento que los jugadores realizan sobre la cancha. La razón de esto obedece a que los límites o fronteras entre dos clusters vecinos son demasiado *precisos* o *duros*, de forma tal que, un jugador con un pequeño movimiento puede ser agrupado en otra zona en un solo ciclo.

Dado que los patrones de formaciones están basados en relaciones determinadas por estructuras, se requiere de un modelo eficiente para el seguimiento de estructuras de formaciones. Tal modelo, debe ser suficientemente robusto para manejar los cambios de posición constantes de los jugadores. A continuación se describe como un conjunto de sub-grafos triangulares conectados, que usan como entrada el resultado del algoritmo K-means, puede ser usado como un modelo de representación adecuado, capaz de facilitar el reconocimiento de los patrones de formaciones.

Algoritmo de reconocimiento de formaciones

La Figura 3.12 muestra el proceso para reconocer patrones de formaciones y cambios de estructuras que soportan las formaciones. El primer modulo sirve para determinar las zonas usando un algoritmo de Clustering; el segundo modulo construye las relaciones múltiples las cuales son expresadas por un grafo topológico y finalmente en el tercer modulo se detecta el cambio de estructuras si la propiedad topológica de una estructura se rompe.

- *Modulo 1.* Reconocimiento de zonas de equipo. El algoritmo de Clustering se lleva a cabo durante los primeros ciclos del juego y se detiene hasta que el número de jugadores en cada grupo tiende a estabilizarse. De esta manera, las tres zonas de un equipo: defensiva, media y de ataque son reconocidas.
- *Modulo 2.* Construcción del grafo topológico. Con base en las tres zonas recono-

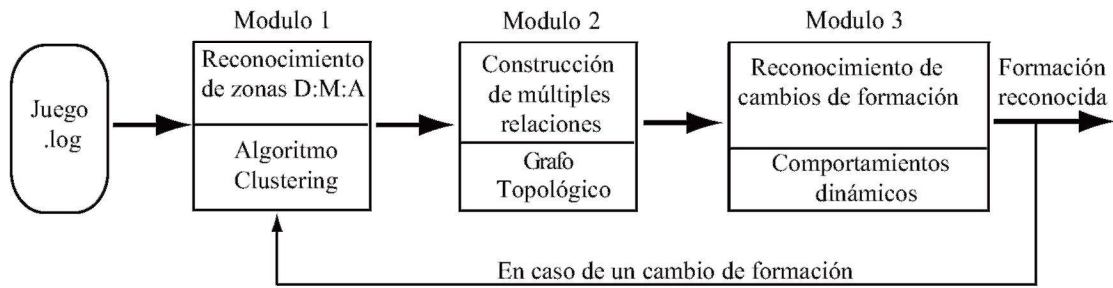


Figura 3.12: Proceso para reconocer patrones de formaciones

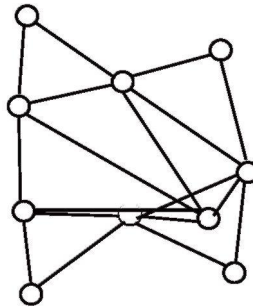


Figura 3.13: Rompimiento de la estructura

cidas por el algoritmo de Clustering, se construyen las múltiples relaciones relevantes y un grafo planar topológico.

- *Modulo 3.* Reconocimiento de Cambios de Estructuras que Soportan Equipos de Formaciones. El cambio de formación se detecta si la propiedad planar del grafo que corresponde a la estructura del equipo se rompe. Un grafo topológico es por definición un grafo planar. En un grafo planar cualquier par de nodos puede ser enlazado. Además, cualquier liga del grafo no debería intersectarse con otra. En caso de que la propiedad topológica de un grafo se rompa, entonces se debe construir otra estructura topológica correspondiente a la nueva formación del equipo. Un ejemplo de ligas intersectadas se muestra en la Figura 3.13. Las intersecciones ocurren cuando los jugadores cambian sus roles para construir una nueva formación o debido a un comportamiento reactivo en respuesta del oponente. Si las intersecciones de las ligas ocurren durante un período corto de tiempo, no debería de considerarse un cambio de formación. Sin embargo, si ocurren durante una secuencia considerable de ciclos, el algoritmo de Clustering debe redefinir las zonas con el fin de construir un nuevo grafo topológico.

El modelo topológico propuesto aquí, está basado en sub-grafos triangulares, capaces de manejar las deformaciones de estructuras debido a los cambios dinámicos y las múltiples relaciones. La Figura 3.14 muestra un equipo con la formación 4:3:3 y como

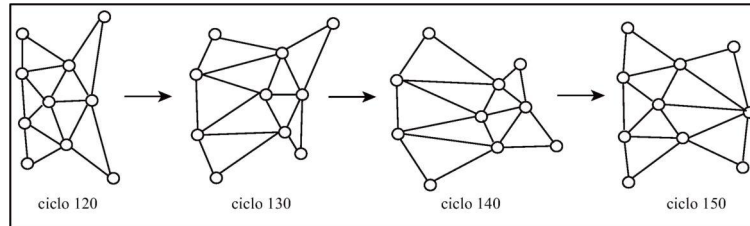


Figura 3.14: Cuatro ciclos que muestran la estructura de un grafo planar sometida al dinamismo de los jugadores del equipo

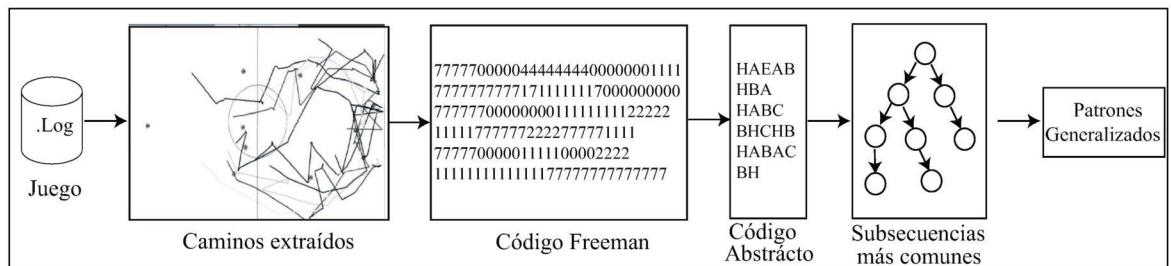


Figura 3.15: Proceso de descubrimiento táctico

los triángulos y las propiedades planares del grafo se mantienen a pesar del movimiento constante que tienen los jugadores.

3.4.2. Descubrimiento de jugadas tácticas

Las jugadas tácticas pueden ser representadas por una secuencia de grafos donde los agentes son representados por nodos y las relaciones por arcos entre ellos. Estos grafos son necesariamente sub-grafos del grafo que representa la formación bajo la cual ocurre la jugada táctica.

El modelo para descubrir patrones de comportamiento táctico propuesto en este trabajo se presenta en la Figura 3.15 y se explica a continuación como un conjunto de 6 pasos:

Paso 1. Lectura del archivo del juego (.log). Los datos que se leen del archivo de juego están relacionados principalmente con información de las posiciones de los jugadores y el balón.

Paso 2. Extracción de trayectorias similares. Se extrae un conjunto de rutas o trayectorias del balón que ocurren bajo contextos similares. Por ejemplo, la Figura 3.16 muestra trayectorias que comienzan de la zona media de la cancha y luego son extendidos hacia las alas derecha o izquierda de la cancha, hasta que llegan a una zona cercana de la portería de gol.

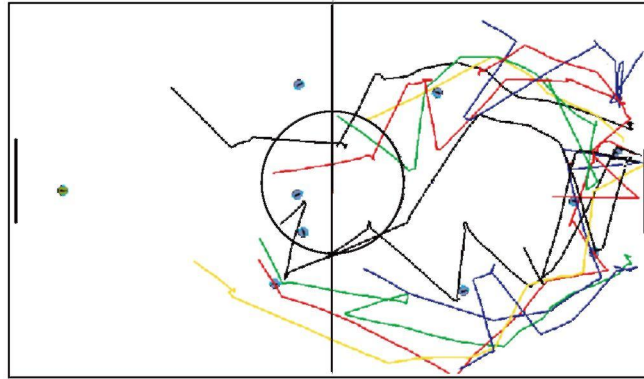


Figura 3.16: Caminos que indican llegadas del equipo izquierdo a la portería del opo-
nente (equipo derecho)

- *Paso 3.* Primera codificación en código Freeman. Se codifican el conjunto de trayectorias para representarse como una secuencia de dígitos usando el código Freeman visto anteriormente.
- *Paso 4.* Segunda codificación de Freeman. Las secuencias expresadas en el paso 3 son utilizadas para obtener un código más abstracto. Sea $\{A, B, C, D, E, F, G, H\}$ un conjunto de segmentos abstractos donde cada elemento de este conjunto representa una secuencia de códigos iguales, con base en la codificación de Freeman, con la misma orientación tal que, **A** representa una secuencia de 0's, **B** representa una secuencia de 1's y así sucesivamente. De esta manera, una trayectoria codificada en el paso 3 como: 1111111100002222222222, puede representarse por el código **BAC**. Esta nueva codificación representa la forma de las trayectorias sin tomar en cuenta la longitud de ellas, facilitando la comparación de diferentes jugadas tácticas potenciales.
- *Paso 5.* Identificación de las Sub-secuencias más comunes. En la búsqueda del descubrimiento de patrones generalizados, se usa un trie [Knuth, 1973]. Un trie es como una estructura de datos de un árbol el cual almacena eficientemente secuencias tal que las sub-secuencias duplicadas se almacenan sólo una vez, manteniendo el número de veces que éstas han aparecido. En el trie usado aquí, cada nodo representa una trayectoria en código abstracto descrito anteriormente y los hijos de un nodo representan las trayectorias que han aparecido siguiendo este camino. De esta forma, un camino que inicia desde la raíz del árbol representa una sub-secuencia que ha aparecido al menos una vez. Considere el siguiente ejemplo: Tomando dos caminos, **ABCA** y **BCAD**. Suponiendo que el trie está vacío, se inserta primero **ABCA** en el, resultando en el trie de la Figura 3.17(a). Luego, se insertan los tres sufijos de **ABCA**: $\{BCA, CA, A\}$, resultando en el trie de la Figura 3.17(b). Ahora, debe insertarse el segundo camino, **BCAD** y sus sufijos: $\{CAD, AD, D\}$ en el trie, resultando en el trie de la Figura 3.17(c). Note que

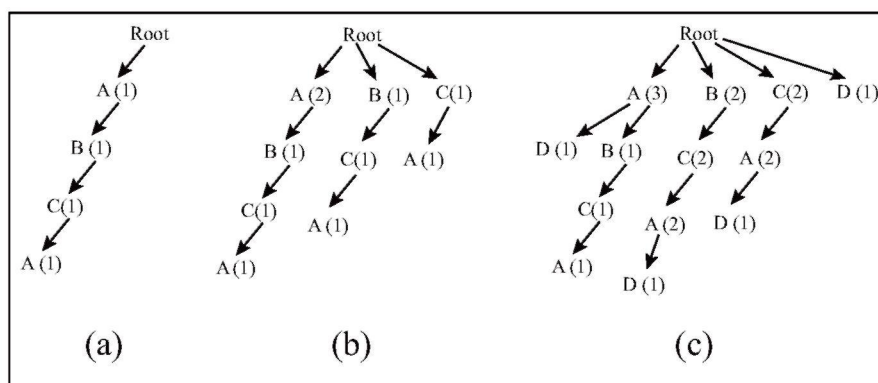


Figura 3.17: Proceso de descubrimiento táctico

en el trie final, la subsecuencia individual más común es **A**, la sub-secuencia de tamaño 2 más común es **CA** y la subsecuencia de tamaño 3 más común es **BCA**.

- *Paso 6. Asociación de jugadores y zonas.* Se refiere a la incorporación del conocimiento del dominio. Los grafos topológicos sirven no sólo para manejar la naturaleza dinámica del juego a través de un seguimiento eficiente, sino que también los grafos topológicos nos permiten conocer a cada instante del juego, qué jugadores están implicados en la jugada así como las relaciones entre ellos. Además, con base en las posiciones de los jugadores involucrados en la jugada, se incorporan las zonas de la cancha donde se llevó a cabo la jugada. Toda esta información, tanto los jugadores como las zonas de la cancha, es incorporada a las jugadas ofensivas como conocimiento del dominio a fin de descubrir e interpretar las jugadas tácticas.

3.5. Ventajas del Modelo

Debido a la naturaleza dinámica del soccer robótico, el cambio constante de posición de los jugadores tiende a complicar el análisis de sus formaciones, mostrando un aparente caos o desorden en su movimiento. Sin embargo, gracias a la estructura topológica que modela las relaciones relevantes entre los jugadores y soporta el dinamismo de éstos, es posible afirmar que un equipo de soccer tiende a conservar una estructura de equipo entre los jugadores que les permite desplegar las estrategias y tácticas en las diferentes situaciones de juego.

Como se menciona en el Capítulo 2, los trabajos relacionados con esta tesis han propuesto soluciones que reconocen formaciones a través de modelos más complejos que no son fácilmente extendibles para potencializar la interpretación del comportamiento de los jugadores, limitándose solamente a reconocer la formación de un equipo en un momento dado. En nuestro caso, la idea de utilizar un grafo topológico permite extender este trabajo para representar al grafo en un nivel de mayor abstracción con el

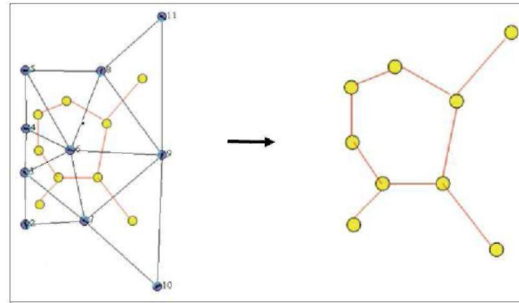


Figura 3.18: Grafo dual

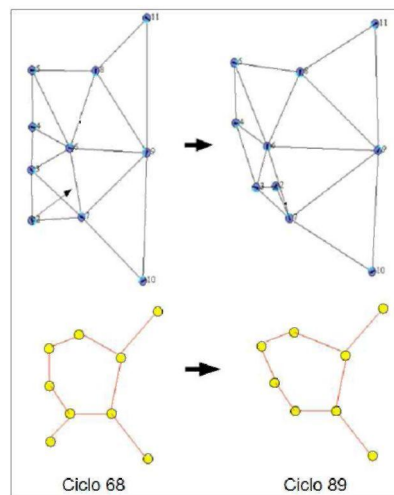


Figura 3.19: El rompimiento de la propiedad planar en el grafo topológico produce otro grafo dual

fin de analizar, desde otra perspectiva, la relación y comportamiento de los jugadores implicados directa o indirectamente en la jugada. En este trabajo, este nuevo nivel de representación está basado en grafo dual del grafo topológico. Dado que un grafo topológico está formado de relaciones relevantes que a su vez forman triángulos entre los jugadores, en su grafo dual, cada nodo representa a un triángulo del grafo topológico y cada arista representa la vecindad entre triángulos (ver Figura 3.18).

Cuando se rompe la propiedad planar en el grafo topológico, el grafo dual también es capaz de expresar que la vecindad de triángulos ha sufrido cambios. Por ejemplo, en la Figura 3.19 se muestra el caso donde el jugador 2 rompe la propiedad planar del grafo topológico y el grafo dual también es alterado. Por lo tanto, una aplicación importante del grafo dual es verificar, junto con el grafo topológico, un cambio de formación en el equipo de jugadores.

Estrategia	D	M	A	Estrategia	D	M	A
	10	0	0		3	2	5
	9	1	0		3	4	3
Formaciones	9	0	1	Formaciones	3	3	4
muy	8	2	0	ofensivas	2	8	0
defensivas	8	0	2		2	0	8
	8	1	1		2	7	1
	7	3	0		2	1	7
	7	0	3		2	6	2
	7	2	1		2	2	6
	7	1	2		2	5	3
Formaciones	6	4	0		2	3	5
defensivas	6	0	4		2	4	4
	6	3	1	Formaciones	1	9	0
	6	1	3	muy	1	0	9
	6	2	2	ofensivas	1	8	1
Formaciones	5	5	0		1	1	8
ofensiva-	5	0	5		1	7	2
defensiva	5	4	1		1	2	7
	5	1	4		1	6	3
	5	3	2		1	3	6
	5	2	3		1	5	4
Formaciones	4	6	0		1	4	5
posibles	4	0	6		0	0	10
variantes de 4's	4	5	1		0	10	0
	4	1	5		0	1	9
Formaciones	4	4	2		0	9	1
clasicas 4's	4	2	4		0	2	8
	4	3	3		0	8	2
Formaciones	3	7	0		0	3	7
ofensivas	3	0	7		0	7	3
	3	6	1		0	4	6
	3	1	6		0	6	4
	3	5	2		0	5	5

Tabla 3.1: Lista de formaciones

Formación	Rel. bidireccionales	Rel. Relevantes	% reducción
4:3:3	45	18	60
4:4:2	45	19	57.8
4:2:4	45	17	62.2
5:2:3	45	17	62.2
5:3:2	45	18	60

Tabla 3.2: Reducción del número de relaciones

Capítulo 4

Experimentos y Análisis de Resultados

En este capítulo se analiza un conjunto de diez equipos de soccer que han participado en varias competencias de RoboCup en la categoría de simulación. Estos equipos fueron escogidos para este trabajo de tesis debido a su participación destacada en los diferentes mundiales que se han llevado a cabo. La Tabla 4.1 muestra la lista de los diez equipos que se analizaron y el año en el que participaron.

No.	Equipo	Campeonato
1	AT-Humboldt	2006
2	Brainstormers	2007
3	FCPortugal	2002
4	Oxsy	2007
5	RoboSina	2005
6	SEU	2006
7	Brasil	2007
8	TsinghuAeolus	2002
9	BUGS	2002
10	WrightEagle	2007

Tabla 4.1: Lista de equipos a analizar

A continuación se presentan las pruebas del modelo propuesto en este trabajo a través del reconocimiento de patrones de formaciones y el descubrimiento de tácticas de equipo.

4.1. Reconocimiento de Patrones de Formaciones

El reconocimiento de formaciones consiste en identificar la formación o formaciones que un equipo utiliza durante el juego o en un período de tiempo de éste. Como se muestra en la Figura 3.12, la metodología propuesta en esta tesis para reconocer una formación consiste en tres pasos:

1. Reconocimiento de zonas
2. Construcción del grafo topológico
3. Seguimiento de la estructura dinámica

Las siguientes secciones muestran las pruebas que se llevaron a cabo para cada uno de los pasos anteriores en diferentes equipos de soccer.

4.1.1. Reconocimiento de Zonas

El objetivo del reconocimiento de zonas es determinar el rol de cada uno de los jugadores del equipo ya sea defensa, medio o delantero. Por esta razón, se definen tres zonas de un equipo:

1. Zona defensiva (D)
2. Zona media (M)
3. Zona ofensiva o de ataque (A)

Partiendo de la idea que los jugadores de una misma zona tienden a mantenerse alineados verticalmente (guardan una distancia relativamente corta en el eje x con respecto a jugadores de otra zona), entonces se intenta identificar a 3 grupos o clusters que señalen a las 3 zonas del equipo. A continuación se muestran los resultados de aplicar el algoritmo de Clustering K -Means (con $K=3$) a varios equipos de soccer.

En el primer caso, se analiza al equipo chino TsinghuAeolus quien ha participado en varios campeonatos mundiales y se ha destacado por los lugares que ha obtenido en las diferentes competencias. Se ha seleccionado este equipo porque, además de que fueron los mejores del mundo, tiene una tendencia de juego muy organizada, es decir, los jugadores tienden a respetar una formación del equipo durante un partido. Por ejemplo, la Tabla 4.2 muestra las diferentes agrupaciones que K -means (con $K=3$), obtuvo tras analizar al equipo TsinghuAeolus en el juego de la final 2002 al enfrentarse contra otro equipo chino llamado Everest.

La Tabla 4.2 presenta las agrupaciones identificadas según el orden de ocurrencia. Por lo tanto, la agrupación más frecuente es 4-3-3, la cual se mantuvo prácticamente durante todo el tiempo que duró el partido.

No.	Agrupación	Ciclos	%
1	4-3-3	4270	85.54
2	3-4-3	296	5.93
3	4-4-2	209	4.19
4	4-2-4	77	1.54
5	2-4-4	67	1.34
6	5-2-3	51	1.02
7	3-3-4	12	0.24
8	2-5-3	10	0.20

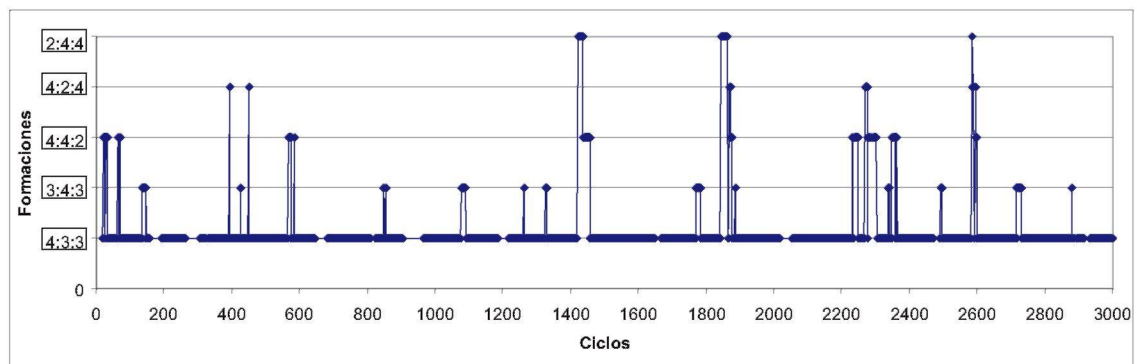
Tabla 4.2: Agrupaciones encontradas por Clustering en TsinghuAeolus

Nota: El número de ciclos total en la Tabla 4.2 es 4992 ya que de los 6000 ciclos de simulación que forman un juego completo, 1008 ciclos (el 10 % del juego total) fueron descartados debido a que durante esos tiempos ocurrieron situaciones fuera de juego, tales como saques de media cancha, saques de meta, fueras de lugar, entre otros.

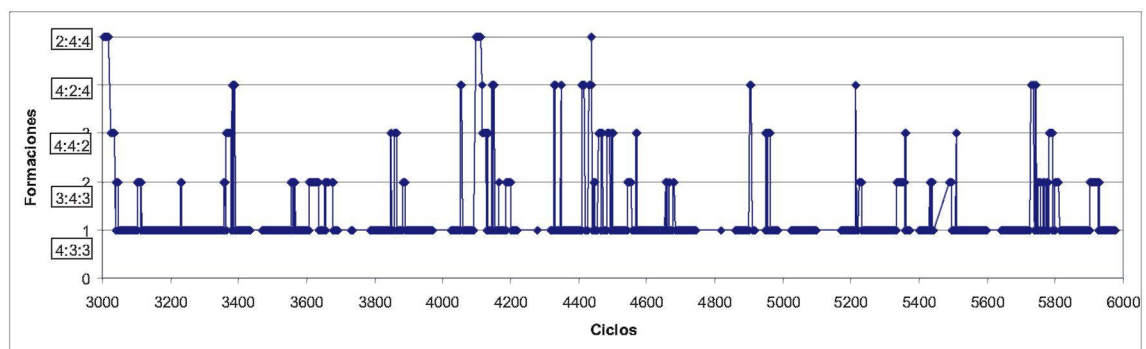
En el caso de este equipo, a pesar de que no se detectan varias agrupaciones principales, además de la 4-3-3, el algoritmo de Clustering clasificó todo el juego en ocho agrupaciones posibles. La Figura 4.1(a) muestra cómo detectó Clustering la secuencia de agrupaciones durante el primer tiempo y la Figura 4.1(b) muestra la secuencia del segundo tiempo.

La Figura 4.1 muestra que, excepto por la agrupación 2-4-4, los cambios de agrupamiento encontrados principalmente son variantes de la agrupación principal 4-3-3. La agrupación 2-4-4 obedece a la estructura que adopta el equipo después de haber anotado un gol, justo en el saque de la media cancha. Además, también se puede observar, sobretodo en el segundo tiempo, que el equipo continuamente cambió su agrupación de 4-3-3 a 3-4-3 debido al constante ataque que realizó sobre su oponente (la formación 3-4-3 es más ofensiva que la formación 4-3-3). Dicha agrupación es disparada generalmente cuando el equipo controla el balón y se encuentra en una situación de ataque.

Para comprender cómo se vio alterada la composición del número de jugadores en cada zona del equipo, la Figura 4.2 ilustra el caso para la zona defensiva, donde se muestra el número de jugadores que conformaron la zona defensiva del equipo durante todo el juego. Dado que la agrupación principal en el partido fue 4-3-3 y además, el equipo TsinghuAeolus dominó durante todo el juego, se observa que los defensas constantemente abandonaron la zona defensiva para apoyar a los medios, dejando principalmente 3 jugadores para defender en lugar de 4. En la búsqueda de encontrar qué jugador defensa era el que constantemente abandonaba su zona, la Figura 4.3 muestra que no fue un jugador en especial quien tendía a abandonar su zona, sino que los cuatro defensas incursionaron a la media, uno a la vez, en diferentes momentos del juego. También se observa en las figuras 4.2 y 4.3 que no hubo un sólo caso donde los jugadores defensas



(a)



(b)

Figura 4.1: Agrupaciones encontradas por Clustering en TsinghuAeolus. (a) Primer tiempo (ciclos 0-3000). (b) Segundo tiempo (ciclos 3001-6000)

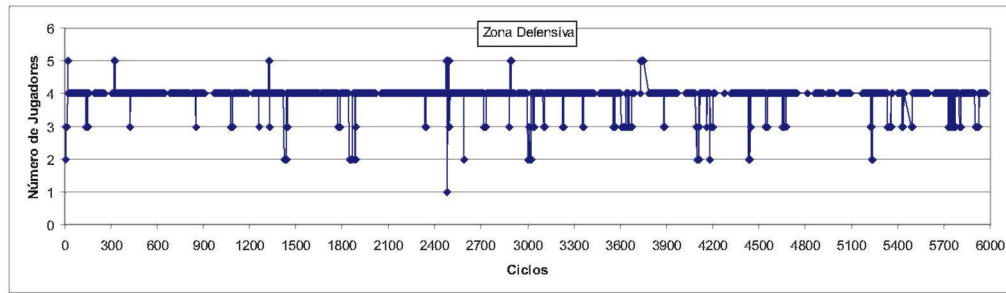


Figura 4.2: Cambio del número de jugadores en defensa (TsinghuAeolus)

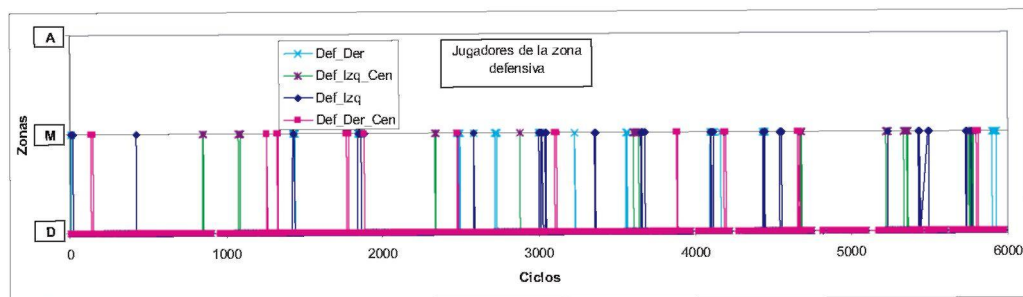


Figura 4.3: Jugadores defensas que abandonan temporalmente su zona

se incorporaran a la zona de ataque.

Un análisis similar se muestra en las Figuras 4.4 y 4.5 donde se muestra el caso para la zona media y los jugadores que originaron la variante del agrupamiento 4-3-3. En este caso, aunque la tendencia fue la de conservar 3 jugadores en la zona media, constantemente la zona media se ve apoyada por un cuarto jugador (el defensa que continuamente abandonaba su zona, como se mencionó arriba). Las incursiones del jugador medio a la zona ofensiva fueron llevadas a cabo específicamente por el jugador medio izquierdo, según la figura 4.5, en las situaciones de ataque.

Es importante señalar que el algoritmo de Clustering no reconoce formaciones debido a que no es capaz de representar las relaciones entre los jugadores, en su lugar, clustering es capaz de determinar la relación de pertenencia de los jugadores a las zonas del equipo.

Las Figuras 4.6 y 4.7 muestran el análisis de la zona ofensiva. En este caso, en la Figura 4.6 se muestra que en promedio el equipo conservó a sus 3 delanteros y en pocos casos se vio apoyado por un medio (el medio izquierdo) y a su vez, pocas veces un delantero abandonó su zona para apoyar a la zona media. La Figura 4.7 muestra que fue el delantero central quien principalmente apoyó a los medios.

Como se discute en la Sección 3.4.1 y con las pruebas que se han presentado aquí, se concluye que el algoritmo de Clustering es útil para identificar cuántos jugadores hay en cada una de las zonas del equipo: defensiva, media y ofensiva. Sin embargo, el algoritmo de clustering es muy inestable para identificar un cambio de formación debido a su gran

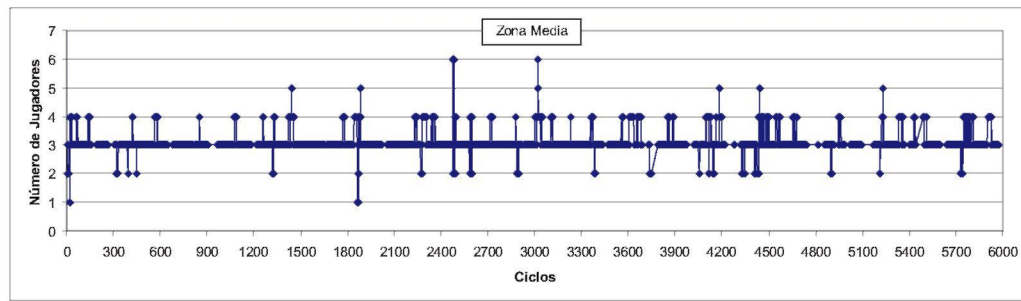


Figura 4.4: Cambio del número de jugadores en la media (TsinghuAeolus)

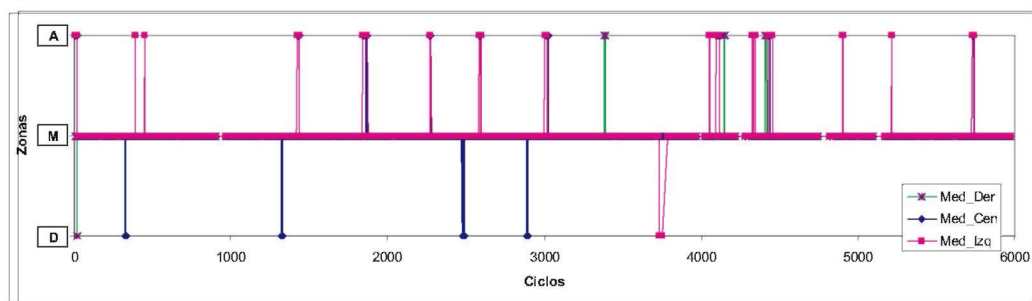


Figura 4.5: Jugadores medios que abandonan temporalmente su zona media

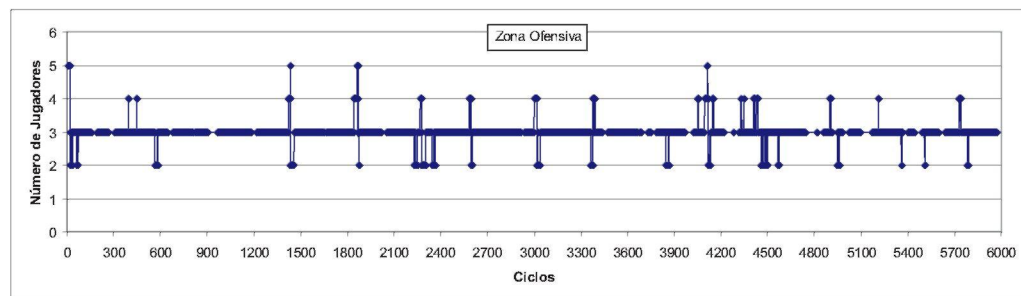


Figura 4.6: Cambio del número de jugadores en la zona ofensiva (TsinghuAeolus)

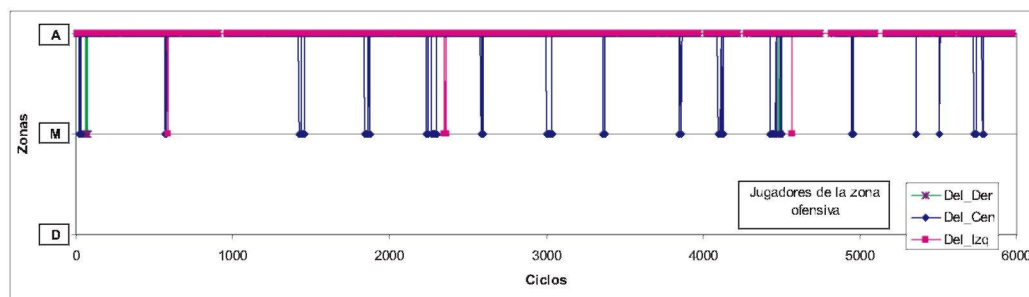


Figura 4.7: Jugadores delanteros que abandonan temporalmente su zona

sensibilidad al movimiento continuo de los jugadores. Por tal razón, en este trabajo la salida del algoritmo de Clustering se emplea para conocer el rol de los jugadores y con base en esto, construir la estructura que representa a las relaciones entre los jugadores dentro del nivel relacional del modelo de representación propuesto en esta tesis.

Para determinar el rol de los jugadores al inicio de un juego, se analizan los primeros ciclos del juego y se aplica el algoritmo de Clustering en cada uno de ellos. El objetivo es determinar el momento en el que el algoritmo de Clustering comienza a estabilizar su salida. En otras palabras, cuando el equipo termina de alinear a sus jugadores con la formación inicial acordada, el algoritmo de Clustering se estabiliza en su salida y entonces comienza a obtener la misma clasificación de jugadores para cada zona. Una vez conocido el rol de cada jugador, se puede identificar la *formación inicial* a través de la construcción del grafo topológico con las relaciones entre jugadores de la misma zona y jugadores de la(s) zona(s) vecina(s).

Los siguientes 2 cuestionamientos son relevantes en el proceso de identificar la formación inicial de un equipo de soccer robótico:

1. ¿A partir de qué momento se puede apreciar la formación inicial que el equipo utiliza?
2. ¿Cuántos ciclos de simulación se requieren para determinar la formación inicial?

Obviamente, la respuesta a las preguntas anteriores están en función del equipo que se analiza y éste a su vez, de las estrategias iniciales determinadas por los diseñadores o programadores del equipo. Sin embargo, al analizar diez equipos que han participado en diferentes campeonatos, se pueden responder a ambas preguntas.

En el primer caso, a la gran mayoría de los equipos se les puede reconocer su formación inicial en los primeros 250 ciclos de simulación. No puede reconocerse antes de este tiempo debido a varias consideraciones, tales como: ciclos *muertos* que pasan antes de la patada inicial del juego, los equipos arrancan el juego con un agrupamiento muy diferente a la formación que emplearán y el tiempo que se requiere para que los jugadores alcancen las posiciones correspondientes en la cancha. Todos estos eventos ocurren en los primeros momentos del partido, provocando cambios de posición en los jugadores que no pueden asociarse con una interpretación adecuada de tal comportamiento.

La Tabla 4.3 presenta la lista de equipos de soccer que se analizaron para identificar sus formaciones. Se observan los ciclos de simulación donde se reconoció la formación inicial (F_g) y la formación inicial reconocida.

Para el caso de los equipos TsinghuAeolus, SEU y Brainstormers, la Figuras 4.8, 4.9 y 4.10 muestra gráficamente las diferentes clasificaciones de zonas que obtuvo el algoritmo de Clustering en los primeros 500 ciclos del juego y se señala la etapa del juego donde se reconoció la formación general del equipo. Se puede apreciar que en todos ellos hay mucha variación en la clasificación en los primeros ciclos del juego hasta que los jugadores adoptan su formación general de equipo.

Para responder a la segunda pregunta planteada anteriormente, no existe un número de ciclos que pueda fijarse para determinar la formación del equipo, pues como se

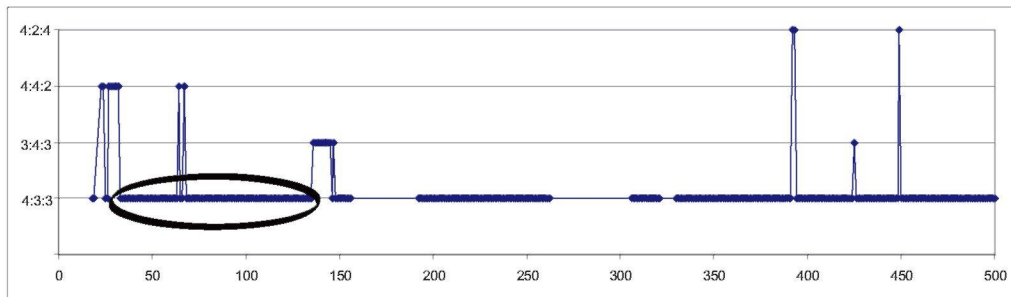


Figura 4.8: Reconocimiento de la formación general de TsinghuAeolus

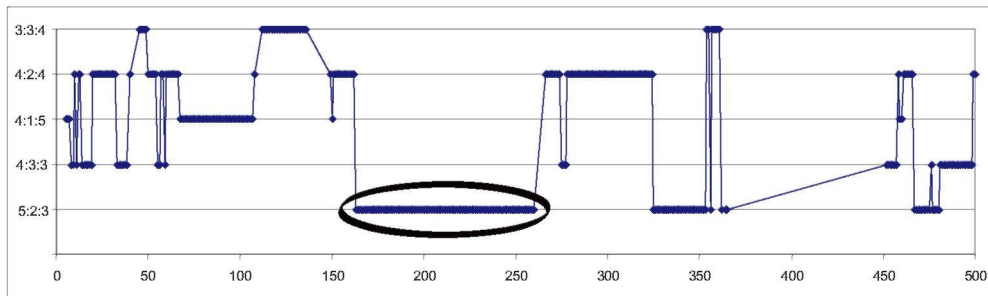


Figura 4.9: Reconocimiento de la formación general de SEU

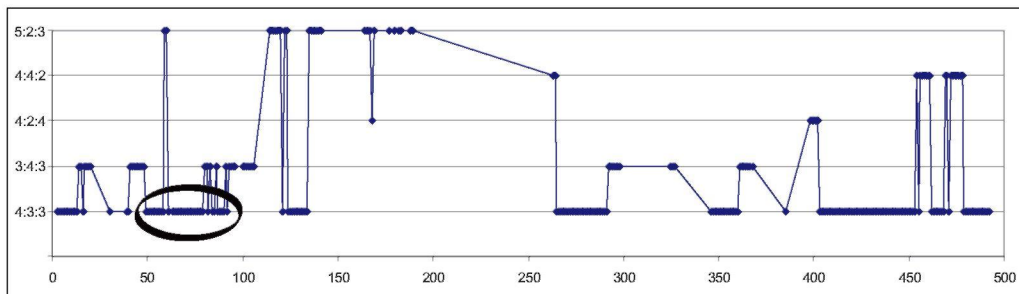


Figura 4.10: Reconocimiento de la formación general de Brainstormers

Equipo	Ciclos donde se reconoció la F_g	F_g
AT-Humboldt	[96-133]	4:3:3
Brainstormers	[49-92]	4:3:3
FCPortugal	[22-53]	4:3:3
Oxsy	[2-32]	5:2:3
RoboSina	[81-165]	4:2:4
SEU	[163-260]	5:2:3
Brasil	[106-185]	4:3:3
TsinghuAeolus	[34-135]	4:3:3
BUGS	[329-409]	5:3:2
WrightEagle	[85-172]	4:3:3

Tabla 4.3: Períodos de tiempo donde se reconocieron formaciones F_g de equipo

discute arriba, depende de cada equipo el dinamismo que sus jugadores muestran en la cancha. Sin embargo, a través de experimentación se puede obtener que una formación tiende a permanecer un tiempo considerable que en la práctica oscila entre 30 y 50 ciclos. Menos que esto son agrupaciones temporales que los jugadores llevan a cabo pero que no impactan de manera significativa ya que no es un comportamiento inducido o planeado. De la misma manera, para validar que un equipo ha cambiado de formación, los jugadores deberían permanecer al menos de 30 a 50 ciclos con la nueva estructura del equipo.

4.1.2. Reconocimiento de Formaciones

En la Sección 3.3.3 se discute la importancia de utilizar un modelo robusto para administrar el constante movimiento de los jugadores a fin de diferenciar entre un cambio de formación real de uno que no lo es. Para esto, en la sección pasada el algoritmo de Clustering mostró debilidades en casos donde los jugadores se mueven causando nuevas configuraciones en los grupos de cada zona del equipo, cuando en realidad no hay cambios reales de formación.

Con base en el agrupamiento de la formación inicial que Clustering detectó, ahora se cuenta con la conformación inicial de las zonas del equipo. Es decir, al conocer qué jugadores inicialmente tienen los roles de defensa, medio y delantero, ahora se determina las relaciones entre los jugadores del equipo de acuerdo a lo discutido en la Sección 3.3.3, a fin de crear un grafo topológico planar.

Para el caso de los equipos de la Tabla 4.3, el algoritmo de Clustering reconoció las pertenencias de cada jugador a una zona del equipo y con esa información se pudo crear la estructura que relaciona a los jugadores del equipo. Cada formación conformó sus zonas por los jugadores que se ilustran en la Tabla 4.4.

Equipo	Fg	D	M	A
AT-Humboldt	4:3:3	2,3,4,5	6,7,8	9,10,11
Brainstormers	4:3:3	2,3,4,5	6,7,8	9,10,11
FCPortugal	4:3:3	2,3,4,5	6,7,8	9,10,11
Oxsy	5:2:3	2,3,4,8,5	6,7	9,10,11
RoboSina	4:2:4	2,3,4,5	6,7	9,10,11,8
SEU	5:2:3	2,3,4,10,1	6,7	8,9,5
Brasil	4:3:3	2,3,4,11	6,7,10	9,5,8
TsinghuAeolus	4:3:3	2,3,4,5	6,7,8	9,10,11
BUGS	5:3:2	2,3,4,5,6	7,8,9	10,11
WrightEagle	4:3:3	2,3,4,5	6,7,8	9,10,11

Tabla 4.4: Alineación de zonas en diferentes equipos

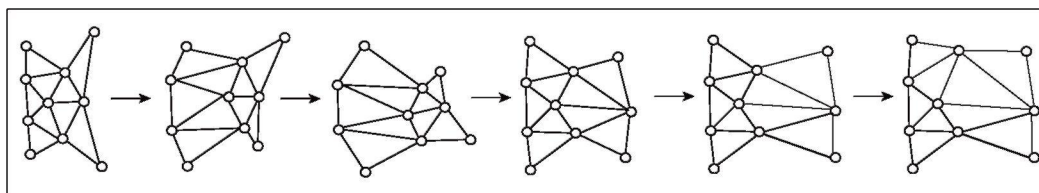


Figura 4.11: Secuencia de ciclos que ilustra la robustez del grafo topológico

Con base en esta información, se pueden establecer las relaciones relevantes de acuerdo a la Sección 3.3.3. Se expresan las relaciones entre compañeros de zona y entre compañeros de zona vecina aplicando la idea de vecinos cercanos y grafos planares. En la figura 4.11 se observa el caso del equipo TsinghuAeolus, donde se muestra una secuencia de ciclos de simulación y se aprecia que la formación 4:3:3 se conserva (no hay intersecciones) a pesar del movimiento dinámico de los jugadores.

En la figura 4.12 se observa el caso del equipo de Portugal en el 2002, comparando los resultados de Clustering contra los del modelo del grafo topológico. La figura ilustra la gran variación reconocida por Clustering contra una estabilidad más pronunciada en la triangulación. El grafo topológico se mantuvo en la misma formación donde Clustering varió mucho.

Ahora se analiza un equipo que sí realizó cambios de formación. Es el caso del equipo FCPortugal al jugar contra el equipo alemán Brainstormers en el 2006. FCPortugal fue desarrollado por la Universidad de Aveiro y la Univesidad de Porto. Es relevante analizar a este equipo en este trabajo debido a que ha obtenido los primeros lugares en varias copas del mundo. En la Figura 4.13 se muestra el resultado de aplicar Clustering en los 6000 ciclos. La figura también muestra el resultado del punto de vista de un experto quien observó el juego varias veces para identificar la formación y el cambio de formación. El experto encontró 15 cambios de formación, entre ellas: 5:2:3, 4:3:3 y

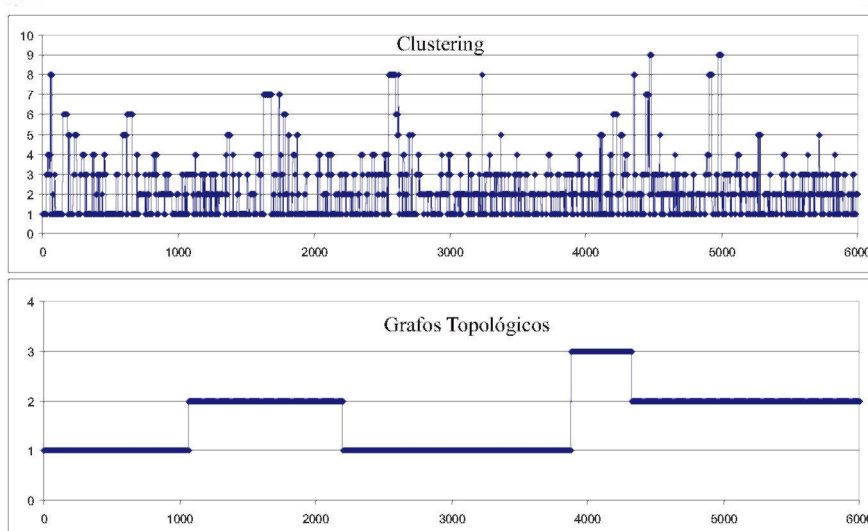


Figura 4.12: Comparación entre Clustering y grafos topológicos (Portugal)

4:2:4. La formación 4:3:3 fue la formación más frecuente identificada por el experto.

Para el algoritmo de Clustering, la Figura 4.13 muestra el resultado de ejecutar este algoritmo en cada uno de los 6000 ciclos. Como se observa, el algoritmo de Clustering reconoció las tres zonas del equipo en 13 clasificaciones diferentes (eje de las y); muchos de los casos fueron clasificaciones con un muy corto período de tiempo. Como el punto de vista del experto, el arreglo 4:3:3 es la estructura más frecuente reconocida por Clustering. Un arreglo 5:2:3 es la segunda estructura más frecuente reconocida. Notar que el alto grado de cambios detectados por Clustering complica el reconocimiento de cambios reales de formaciones. Por esta razón, la exactitud es 42% con respecto al punto de vista del experto.

El resultado del grafo planar topológico en la Figura 4.13 muestra un comportamiento más estable comparado al algoritmo de Clustering y similar al análisis del experto. De hecho, el modelo del grafo topológico reconoció 9 cambios de formación, 7 de ellos correspondieron con los cambios observados por el experto. Los casos donde el experto observó cambios de formación y el modelo de grafos planares no lo hizo, se debe al hecho de que la propiedad de grafos planares no fue violada. Un trabajo futuro puede ser introducir una distribución de probabilidad entre formaciones para determinar el cambio de formación aunque la propiedad de los grafos planares no sea violada. Mientras que el algoritmo de Clustering indica un cambio de roles en los jugadores (clasificaciones diferentes), el modelo de grafos topológicos planares es lo suficientemente robusto para administrar los cambios de las posiciones de los jugadores.

La Figura 4.14 muestra el resultado de aplicar el análisis descrito anteriormente, donde se compara el resultado de *Clustering* y de grafos planares topológicos con las observaciones del experto:

- Juego 1: FCPortugal vs. Brainstormers

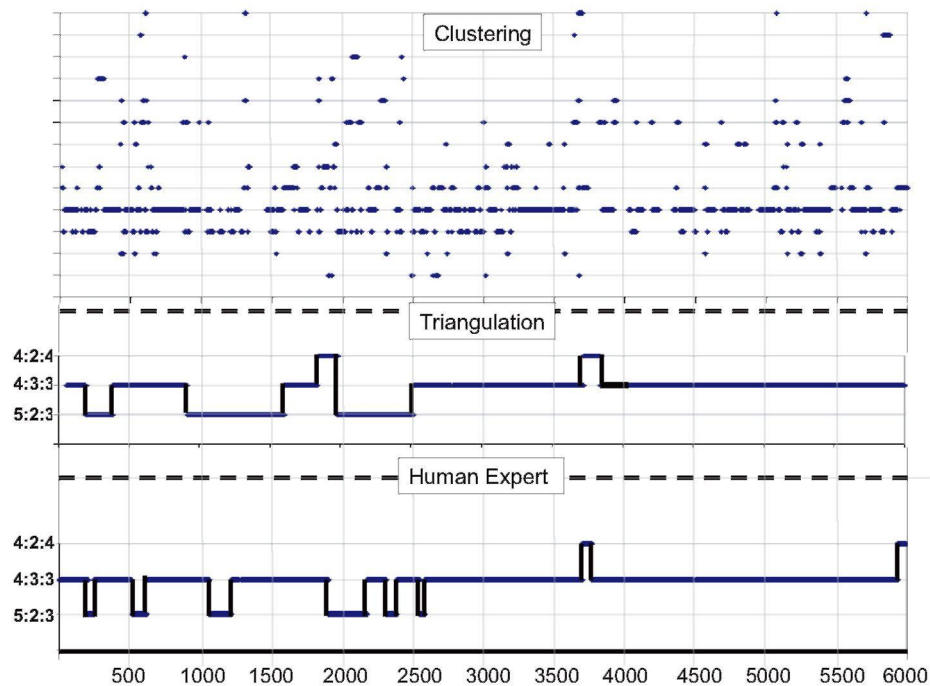


Figura 4.13: Comportamiento del equipo FCPortugal. Resultados del algoritmo de Clustering, grafo planar topológico y punto de vista del experto.

- Juego 2: SEU vs. FCPortugal
- Juego 3: AT-Humboldt vs. Brainstormers
- Juego 4: Brainstormers vs. SEU
- Juego 5: SEU vs. AT-Humboldt

En todos los juegos analizados, el equipo que se tomó como caso de estudio fue el primer equipo que aparece en la lista de arriba. Notar que SEU obtuvo diferentes resultados en los dos juegos (juego 2 y juego 5) ya que un equipo no siempre juega con el mismo comportamiento sino que acomoda sus estrategias de acuerdo al oponente en turno que enfrenta.

En general, el resultado con el modelo de grafos topológicos planares fue del 75% con respecto al punto de vista el experto, mientras que el algoritmo de Clustering tuvo un 39% de exactitud.

4.2. Descubrimiento de Patrones Tácticos

En un primer paso, las trayectorias del balón fueron extraídas para ser analizadas y codificadas en código Freeman. Para analizar el ataque de un equipo, se obtuvo el

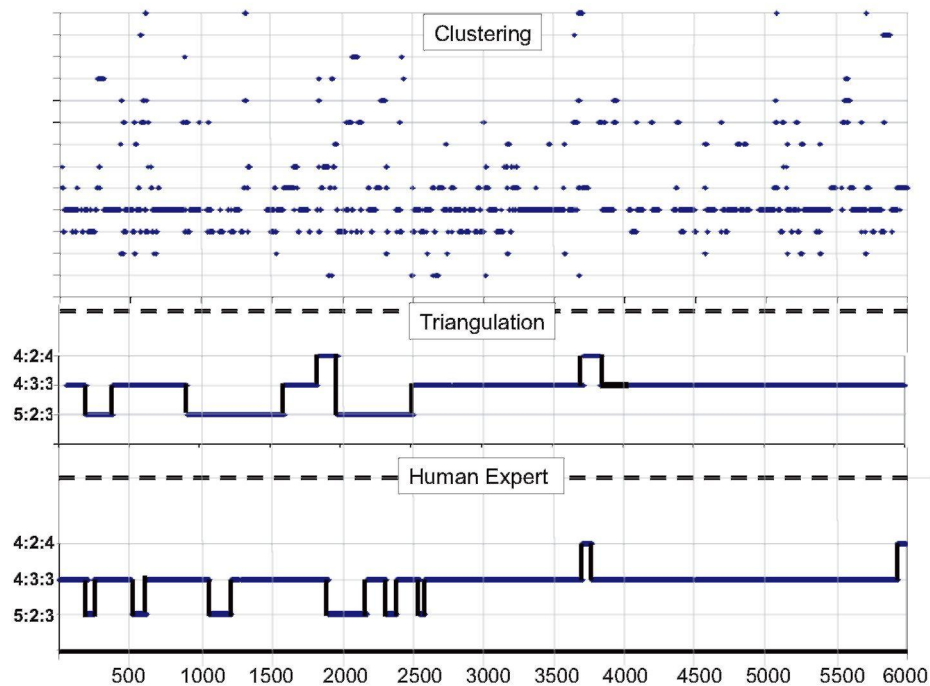


Figura 4.13: Comportamiento del equipo FCPortugal. Resultados del algoritmo de Clustering, grafo planar topológico y punto de vista del experto.

- Juego 2: SEU vs. FCPortugal
- Juego 3: AT-Humboldt vs. Brainstormers
- Juego 4: Brainstormers vs. SEU
- Juego 5: SEU vs. AT-Humboldt

En todos los juegos analizados, el equipo que se tomó como caso de estudio fue el primer equipo que aparece en la lista de arriba. Notar que SEU obtuvo diferentes resultados en los dos juegos (juego 2 y juego 5) ya que un equipo no siempre juega con el mismo comportamiento sino que acomoda sus estrategias de acuerdo al oponente en turno que enfrenta.

En general, el resultado con el modelo de grafos topológicos planares fue del 75% con respecto al punto de vista el experto, mientras que el algoritmo de Clustering tuvo un 39% de exactitud.

4.2. Descubrimiento de Patrones Tácticos

En un primer paso, las trayectorias del balón fueron extraídas para ser analizadas y codificadas en código Freeman. Para analizar el ataque de un equipo, se obtuvo el

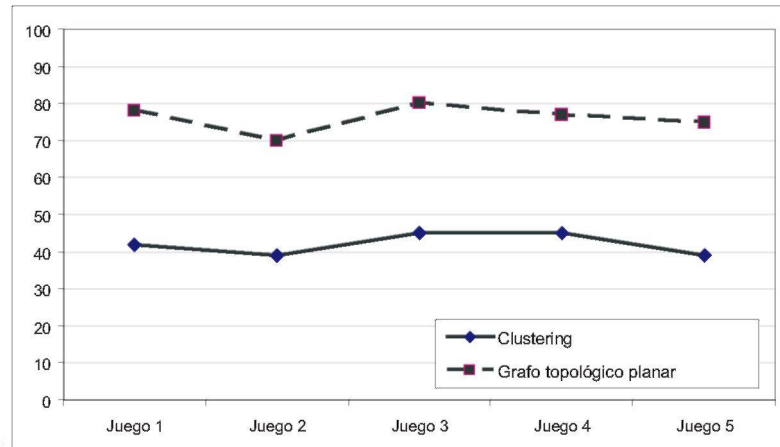


Figura 4.14: Resultado del modelo de grafos topológicos planares vs. Clustering con respecto al punto de vista del experto.

conjunto de trayectorias del balón que más se acercaban a la portería del oponente. Para el caso del equipo TsinghuAeolus, la Figura 4.15 muestra las trayectorias extraídas y sus correspondientes códigos Freeman. De esta forma, el conjunto de trayectorias puede ser comparado numéricamente a través de medir la similitud entre ellos. Otra ventaja de esta codificación es que podemos tener una idea acerca de qué tan largos son las trayectorias, pero desde el punto de vista del comportamiento, la forma adoptada por la trayectoria y obviamente las propiedades asociadas con la intención o propósito de este, en este caso, acercarse a una posición de disparo a la portería.

Debido a estas razones, en este trabajo se propone una representación más abstracta. Luego, las trayectorias codificadas en código Freeman se recodifican para obtener un código más abstracto, como se mostró en la Sección 3.3.4. Las trayectorias representadas por los códigos abstractos han facilitado la aplicación del modelo para descubrir patrones de comportamiento relacionados con jugadas tácticas. Es importante señalar que las trayectorias similares no son necesariamente aquellas que terminan en un gol, sino aquellos que asumen un comportamiento similar desde el inicio de la trayectoria hasta el objetivo final.

Para simplificar el análisis de las trayectorias del balón, se puede observar en la Figura 4.15, que el equipo atacó principalmente de dos maneras: por el ala izquierda y por el ala derecha de la cancha. Por tal razón, es conveniente dividir el análisis buscando patrones de llegadas tanto en el ala izquierda como en el ala derecha de la cancha.

Como se discutió en la Sección 3.4.2, este trabajo de tesis resalta el hecho de que las jugadas tácticas ocurren siempre bajo el contexto formaciones modeladas por grafos topológicos. Estos grafos sirven no sólo para manejar la naturaleza dinámica del juego, sino que además, los grafos topológicos nos permiten conocer a cada instante del juego, qué jugadores están implicados en la jugada así como las relaciones entre ellos.

Por ejemplo, para el caso del equipo TsinghuAeolus, en la Figura 4.16(a) se observa

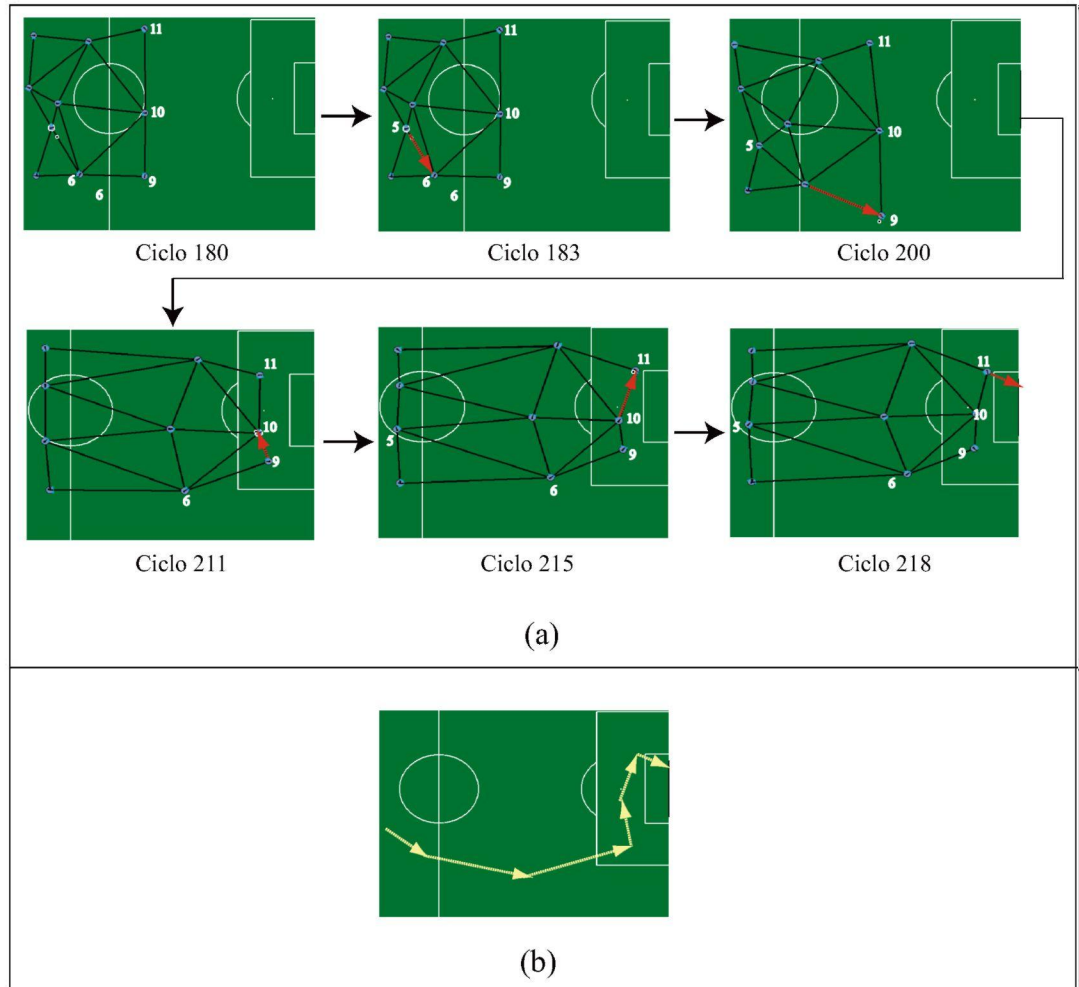


Figura 4.16: Secuencia de ciclos en una jugada táctica soportada por la formación 4:3:3. a) Secuencia de ciclos que ilustran los pases entre los jugadores 5,6,9,10 y 11. b) Resumen de la trayectoria del balón desde el ciclo 180 hasta el ciclo 218

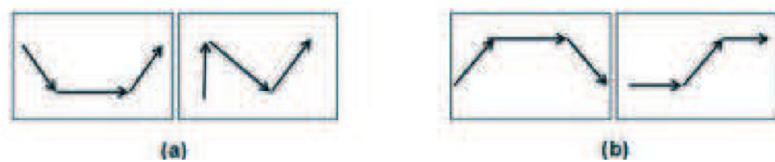


Figura 4.17: Trayectorias generalizadas de comportamientos tácticos: a) ataques por el ala derecha y b) ataques por el ala izquierda.

Pases entre jugadores	Zonas de la cancha	¿GOL?
4,7,10,11	zona defensiva - area chica	
8,11	media - area chica	
4,10,8,11,10	media - area grande	
5, 10,11,10,11	media - area grande	
8,11,10	area grande - area chica	Gol
7,11,10,11	media - area grande	
8,11,10	media - area chica	Gol
2,10,11,9,11	media - area chica	

Tabla 4.5: Ataques por el ala izquierda

Pases entre jugadores	Zonas de la cancha	¿GOL?
5,6,9,10,11	media-area chica	Gol
5,6,9,11	media-area chica	Gol
11,10,11,10	media-area grande	
2,7,6,10,9,11	media-area chica	Gol
5,6,9,10,11	defensiva-area chica	Gol
6,9,10,9	media-area grande	
6,10,11	area grande-area chica	Gol
7,6,9	media - area chica	
6,9	media-area grande	

Tabla 4.6: Ataques por el ala derecha

trayectorias representando las jugadas tácticas, no dependen del equipo analizado. Las estructuras topológicas usadas para seguir las formaciones han sido un soporte muy importante para aportar el contexto de la jugada a fin de descubrir los patrones tácticos del equipo. Este contexto aporta tanto qué jugadores participan en la jugada así como las zonas de la cancha en las cuales se desarrolló tal jugada.

Por un lado, contar con un modelo de representación expresivo, facilita el reconocimiento y descubrimiento de patrones de comportamiento. El modelo de representación propuesto en este trabajo proporciona un contexto rico en expresividad construido a nivel de formaciones a fin de permitir el descubrimiento de tácticas. Por otro lado, la falta de información dificulta en gran manera el descubrimiento de patrones. Por ejemplo, si sólo se contara con información individual de los jugadores, sería muy complicado descubrir jugadas tácticas que impliquen la interacción entre varios jugadores.

Las fortalezas del modelo presentado en esta tesis son:

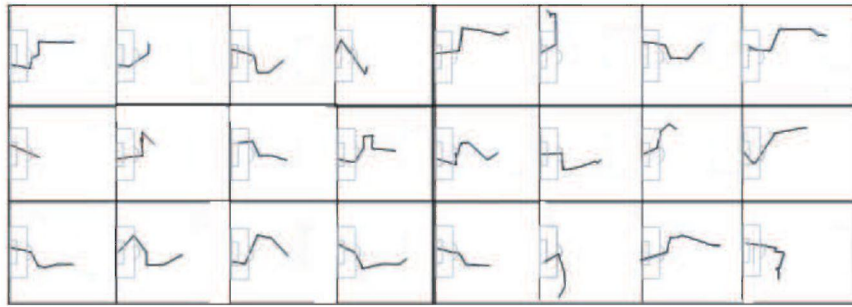


Figura 4.18: Trayectorias extraídas de llegadas cerca de la portería del oponente. El equipo está atacando de derecha a izquierda.

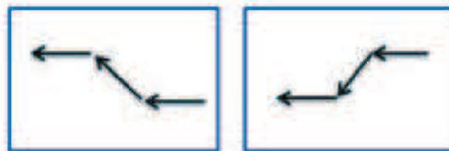


Figura 4.19: Dos formas de trayectorias generalizadas de comportamientos tácticos.

1. Gracias al grafo topológico, se administran mejor los seguimientos de las formaciones.
2. Gracias al grafo topológico, se facilita el descubrimiento de tácticas al proveer un contexto rico.
3. El modelo de representación permite una navegación *top-down* y *bottom-up*.

La principal debilidad del modelo es que no contempla casos de equipos que juegan de una manera no estructurada, es decir, sin una formación definida.

Capítulo 5

Conclusiones

El descubrimiento de jugadas tácticas y las formaciones que soportan las estrategias del equipo, representan información relevante para implementar contra estrategias o tácticas para sacar ventaja del oponente y vencerlo.

Este trabajo aborda el problema de reconocer patrones de formaciones y descubrir patrones tácticos en dominios complejos en tiempo real y altamente dinámicos, tal como el soccer robótico. Estas características provocan múltiples interacciones entre los agentes y dificultan la tarea de identificar patrones de comportamiento. En este trabajo se presentó un modelo robusto para reconocer patrones formaciones así como identificar los cambios de formación bajo ciertas circunstancias del juego. El modelo está basado en una representación expresiva que toma en cuenta las relaciones relevantes entre los jugadores, específicamente, relaciones de vecindad entre jugadores de la misma zona y las relaciones de vecindad entre jugadores de zonas vecinas. Para enfrentar este problema, esta tesis contribuye con un modelado de relaciones a varios niveles de abstracción que facilita el reconocimiento de formaciones y estrategias de un equipo. Se ha demostrado que el modelo, a nivel de formaciones, facilita el reconocimiento de formaciones y cambios de formación.

En esta tesis, los grafos topológicos triangulares han sido los elementos clave para un reconocimiento correcto de patrones de formaciones debido a sus características topológicas las cuales toleran los movimientos dinámicos de los jugadores sin cambios de relación y de formación entre ellos. El triángulo es una figura básica muy robusta con un comportamiento noble que no pierde expresividad; a partir de un conjunto de triángulos se pueden formar muchas figuras o estructuras. La topología del grafo dicta la relación de las partes e indica cuando hay un cambio de estructura al romperse las propiedades topológicas de la estructura.

Con respecto al descubrimiento a nivel relacional, en este trabajo se resalta la importancia del descubrimiento de patrones tácticos dentro del contexto de las formaciones. Las jugadas tácticas fueron caracterizadas no sólo por una secuencia indicando la trayectoria del balón, sino que se remarca la importancia de incluir a los jugadores y las zonas de la cancha a través de las cuales ocurrieron los pases en la jugada.

El modelo de grafos topológicos que soportó el descubrimiento de jugadas tácticas ha sido determinante para extraer la secuencia de sub-grafos que contienen a los jugadores involucrados en la jugada táctica.

El comportamiento de las trayectorias representando jugadas tácticas están más relacionadas con la forma de la trayectoria misma que con la longitud de ésta. Para enfrentar este problema, las trayectorias originales fueron codificadas dos veces, usando el código de Freeman, donde la segunda codificación es más abstracta que la primera. La diferencia entre ellas es que la primera contiene la longitud de la trayectoria y la segunda expresa más su forma. Con este modelo se descubrieron jugadas tácticas ofensivas relevantes.

5.1. Contribuciones

- Un modelo expresivo a varios niveles de representación.
- Reconocimiento, descubrimiento e interpretación de patrones de comportamiento en los diferentes niveles de representación del modelo propuesto.
- Un algoritmo robusto para el seguimiento de formaciones en ambientes multi-agentes basado en estructuras topológicas.

5.2. Trabajo Futuro

- Descubrimiento de variantes que aporten información clave a las estrategias y tácticas del equipo. Por ejemplo, las variantes de formación que puede emplear el equipo alrededor de una formación base cuando hay una situación defensiva u ofensiva. Las variantes pueden llegar a ser la clave estratégica de los equipos en condiciones de anotar un gol.
- Descubrimiento de tácticas defensivas a fin de contar con un mayor conocimiento del comportamiento de un equipo de soccer robótico.
- Reconocer y descubrir patrones de comportamiento en modo *on-line* con el objeto de integrar dicho modelo a un equipo de soccer robótico.

5.3. Aplicaciones en otros dominios

Uno de los mayores retos de la biología moderna consiste en entender cómo y cuando se expresa la información codificada en el ADN, es decir, el control temporal y espacial de la expresión de genes específicos, ya sea ante determinados eventos intracelulares o ante estímulos externos. Muchas de las respuestas, en un momento dado, requerirán

de cambios en el estado de expresión del genoma: genes que estaban prendidos necesitarán ser apagados y genes latentes necesitarán ser activados. En los organismos, dicha regulación de la expresión génica se lleva a cabo primordialmente al inicio de la transcripción y está mediada comúnmente por proteínas reguladoras, ya sea para inhibirla o favorecerla, de acuerdo a las necesidades de la célula. La expresión génica se lleva a cabo por sistemas de regulación a través de una red de interacciones entre el ADN, el ARN y las Proteínas. Esto conlleva a la necesidad de entender el comportamiento estructural dinámico de estos sistemas, lo cual se vuelve más complejo a medida que se involucran más y más componentes interconectados a través de retroalimentaciones negativas y positivas.

Debido a que las funciones de una célula no residen en las moléculas por sí mismas sino en sus interacciones, el desarrollo de modelos formales representa una contribución importante para entender el comportamiento de sistemas bioquímicos complejos como las redes regulatorias. La mera intuición no puede abarcar los efectos de las múltiples interacciones regulatorias simultáneas que ocurren dentro de las células. Un enfoque basado en grafos topológicos puede ayudarnos a entender el funcionamiento y comportamiento de la expresión génica, además de ser capaces de monitorear el comportamiento de las proteínas reguladoras en la transcripción.

Referencias

- [Anderson et al., 1990] Anderson, J. R., Boyle, C. F., Corbett, A. T., and Lewis, M. W. (1990). Cognitive modeling and intelligent tutoring. *Artif. Intell.*, 42(1):7-49.
- [Andraz Bezek and Bratko, 2006] Andraz Bezek, M. G. and Bratko, I. (2006). Multi-agent strategic modeling in a robotic soccer domain. In *AAMAS '06: Autonomous Agents and Multi-Agent Systems*, pages 15-22, New York, NY, USA. ACM Press.
- [Andreas G.Nie, 2001] Andreas G.Nie, Collin Rogowski, P. H. (2001). The osnabrueck robocup agents project. In *Project*.
- [Ayanegui and Ramos, 2007] Ayanegui, H. and Ramos, F. (2007). Recognizing patterns of dynamic behaviors based on multiple relations in soccer robotics domain. In Ghosh, A., De, R. K., and Pal, S. K., editors, *PREMI*, volume 4815 of *Lecture Notes in Computer Science*, pages 33-40. Springer.
- [Bauer, 1999] Bauer, M. (1999). From interaction data to plan libraries: A clustering approach. In Dean, T., editor, *IJCAI*, pages 962-967. Morgan Kaufmann.
- [Berge, 1983] Berge, C. (1983). *Graphes*. Guathier-Villars.
- [Bodey R. Baker, 2006] Bodey R. Baker, Mark Reynolds, W. L. (2006). Strategy specification for teamwork in robot soccer. In Wei Liu, M.-A. W., editor, *Proceedings of the 2006 international symposium on Practical cognitive agents and robots*, volume 213, pages 129-140, New York, NY, USA. ACM.
- [Bui, 2003] Bui, H. H. (2003). A general model for online probabilistic plan recognition. In Gottlob, G. and Walsh, T., editors, *IJCAI*, pages 1309-1318. Morgan Kaufmann.
- [Carmel and Markovitch, 1996] Carmel, D. and Markovitch, S. (1996). Incorporating opponent models into adversary search. In *AAAI/IAAI, Vol. 1*, pages 120-125.
- [Davison and Hirsh, 1998] Davison, B. and Hirsh, H. (1998). Probabilistic online action prediction.
- [de Boer and Kok, 2002] de Boer, R. and Kok, J. (2002). The incremental development of a synthetic multi-agent system: The uva trilearn 2001 robotic soccer simulation team.

- [DOD, 2001] DOD (2001). Department of defense dictionary of military and associated terms. *Joint Publication 1-02*.
- [Dufournaud et al., 2004] Dufournaud, Y., Schmid, C., and Horaud, R. (2004). Image matching with scale adjustment. *Computer Vision and Image Understanding*, 93(2):175-194.
- [Emiel Corten, 1999] Emiel Corten, Peter Stone, T. Y. (1999). Soccerserver manual ver. 5 rev. 00 beta (for soccerserver ver.5.00 and later).
- [Ferber, 1999] Ferber, J. (1999). *Multi-Agent System: An Introduction to Distributed Artificial Intelligence*. Addison Wesley Longman.
- [Ferber et al., 2000] Ferber, J., Gutknecht, O., Jonker, C. M., Treur, J., and Muller, J.-P. (2000). Organization models and behavioral requirements specification for multi-agent systems. In *ICMAS*, pages 387-388. IEEE Computer Society.
- [Freeman, 1973] Freeman, H. (1973). On the encoding of arbitrary geometric configurations. In *IRE Transactions, editor*.
- [Garrido-Luna et al., 1998] Garrido-Luna, L., Brena, R., and Sycara, K. P. (1998). Towards modeling other agents: A simulation-based study. In Sichman, J. S., Conte, R., and Gilbert, N., editors, *MABS*, volume 1534 of *Lecture Notes in Computer Science*, pages 210-225. Springer.
- [Goldman et al., 1999] Goldman, R. P., Geib, C. W., and Miller, C. A. (1999). A new model of plan recognition. In Laskey, K. B. and Prade, H., editors, *UAI*, pages 245-254. Morgan Kaufmann.
- [Hirano and Tsumoto, 2004] Hirano, S. and Tsumoto, S. (2004). Finding interesting pass patterns from soccer game records. In Boulicaut, J.-F., Esposito, F., Giannotti, F., and Pedreschi, D., editors, *PKDD*, volume 3202 of *Lecture Notes in Computer Science*, pages 209-218. Springer.
- [Huber et al., 1994] Huber, M. J., Durfee, E. H., and Wellman, M. P. (1994). The automated mapping of plans for plan recognition. In de Mantaras, R. L. and Poole, D., editors, *UAI*, pages 344-351. Morgan Kaufmann.
- [Intille and Bobick, 1999] Intille, S. S. and Bobick, A. F. (1999). A framework for recognizing multi-agent action from visual evidence. In *AAAI/IAAI*, pages 518-525.
- [Kaminka et al., 2002] Kaminka, G. A., Fidanboyly, M., Chang, A., and Veloso, M. M. (2002). Learning the sequential coordinated behavior of teams from observations. In Kaminka, G. A., Lima, P. U., and Rojas, R., editors, *RoboCup*, volume 2752 of *Lecture Notes in Computer Science*, pages 111-125. Springer.

- [Kautz and Allen, 1986] Kautz, H. A. and Allen, J. F. (1986). Generalized plan recognition. In *AAAI*, pages 32-37.
- [Kitano et al., 1997] Kitano, H., Tambe, M., Stone, P., Veloso, M. M., Coradeschi, S., Osawa, E., Matsubara, H., Noda, I., and Asada, M. (1997). The robocup synthetic agent challenge 97. In Kitano, H., editor, *RoboCup*, volume 1395 of *Lecture Notes in Computer Science*, pages 62-73. Springer.
- [Knuth, 1973] Knuth, D. E. (1973). *The Art of Computer Programming*. Addison-Wesley.
- [Kuhlmann et al., 2004] Kuhlmann, G., Stone, P., and Lallinger, J. (2004). The ut austin villa 2003 champion simulator coach: A machine learning approach. In Nardi, D., Riedmiller, M., Sammut, C., and Santos-Victor, J., editors, *RobuCup*, volume 3276 of *Lecture Notes in Computer Science*, pages 636-644. Springer.
- [L. P. Reis and Oliveira, 2000] L. P. Reis, N. L. and Oliveira, E. (2000). Situation based strategic positioning for coordinating a team of homogeneous agents. In M. Hannebauer, J. W. and Pagello, E., editors, *Balancing Reactivity and Social Deliberation in Multi-Agent Systems*, volume 2103 of *Lecture Notes in Computer Science*, pages 175-197. Springer.
- [Laird et al., 1987] Laird, J. E., Newell, A., and Rosenbloom, P. S. (1987). SOAR: An architecture for general intelligence. *Artificial Intelligence*, 33:1-64.
- [Lattner et al., 2005] Lattner, A. D., Miene, A., Visser, U., and Herzog, O. (2005). Sequential pattern mining for situation and behavior prediction in simulated robotic soccer. In Bredenfled, A., Jacoff, A., Noda, I., and Takahashi, Y., editors, *RoboCup*, volume 4020 of *Lecture Notes in Computer Science*, pages 118-129. Springer.
- [Lau and Reis, 2001] Lau, N. and Reis, L. P. (2001). Fc portugal 2001 team description: Flexible teamwork and configurable strategy. In 0002, A. B., Coradeschi, S., and Tadokoro, S., editors, *RoboCup*, volume 2377 of *Lecture Notes in Computer Science*, pages 515-518. Springer.
- [Ledezma et al., 2004] Ledezma, A., Aler, R., Sanchús, A., and Borrajo, D. (2004). Predicting opponent actions by observation. In Nardi, D., Riedmiller, M., Sammut, C., and Santos-Victor, J., editors, *RobuCup*, volume 3276 of *Lecture Notes in Computer Science*, pages 286-296. Springer.
- [Lesh et al., 1999] Lesh, N., Rich, C., and Sidner, C. (1999). Using plan recognition in human-computer collaboration.
- [Mackworth, 1993] Mackworth, A. K. (1993). On seeing robots. Technical Report "TR-93-05", Carnegie Mellon University".

- [MacQueen, 1967] MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In Cam, L. M. L. and Neyman, J., editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281-297. University of California Press.
- [Markovitch and Reger, 2005] Markovitch, S. and Reger, R. (2005). Learning and exploiting relative weaknesses of opponent agents. *Autonomous Agents and Multi-Agent Systems*, 10(2):103-130.
- [Matsubara et al., 1996] Matsubara, H., Noda, I., and Hiraki, K. (1996). Learning of cooperative actions in multiagent systems: A case study of pass play in soccer. In Sen, S., editor, *Working Notes for the AAAI Symposium on Adaptation, Co-evolution and Learning in Multiagent Systems*, pages 63-67, Stanford University, CA.
- [Michel et al., 2003] Michel, F., Gouaich, A., and Ferber, J. (2003). Weak interaction and strong interaction in agent based simulations. In Hales, D., Edmonds, B., Norling, E., and Rouchier, J., editors, *MABS*, volume 2927 of *Lecture Notes in Computer Science*, pages 43-56. Springer.
- [Miyamoto, 2002] Miyamoto, C. (2002). Strategy vs tactics. *Public Relations Strategies*, First Quarter.
- [Mohsen Sharifi, 2003] Mohsen Sharifi, Hamed Mousavian, A. A. (2003). Predicting the future state of the robocup simulation environment: Heuristic and neural networks approaches. In *IEEE International Conference on Systems, Man and Cybernetics*. IEEE.
- [Nair et al., 2004] Nair, R., Tambe, M., Marsella, S., and Raines, T. (2004). Automated assistants for analyzing team behaviors. *Autonomous Agents and Multi-Agent Systems*, 8(1):69-111.
- [Noda and Frank, 1998] Noda, I. and Frank, I. (1998). Investigating the complex with virtual soccer. In Heudin, J.-C., editor, *Virtual Worlds*, volume 1434 of *Lecture Notes in Computer Science*, pages 241-253. Springer.
- [Patrick Riley, 2002] Patrick Riley, Manuela Veloso, G. K. (2002). An empirical study of coaching. In H. Asama, T. Arai, T. F. and Hasegawa, T., editors, *Distributed Autonomous Robotic Systems*, volume 5, pages 215-224. Springer.
- [Quilici et al., 1998] Quilici, A., Yang, Q., and Woods, S. (1998). Applying plan recognition algorithms to program understanding. *Autom. Softw. Eng.*, 5(3):347-372.
- [Raines et al., 1999] Raines, T., Tambe, M., and Marsella, S. (1999). Automated assistants to aid humans in understanding team behaviors. In Veloso, M. M., Pagello, E., and Kitano, H., editors, *RoboCup*, volume 1856 of *Lecture Notes in Computer Science*, pages 85-102. Springer.

- [Ramos and Ayanegui, 2008a] Ramos, F. and Ayanegui, H. (2008a). Discovering behavior patterns in multi-agent teams. In Nguyen, N. T., Jo, G., Howlett, R. J., and Jain, L. C., editors, *KES-AMSTA*, Lecture Notes in Computer Science, pages 391-400. Springer.
- [Ramos and Ayanegui, 2008b] Ramos, F. and Ayanegui, H. (2008b). Discovering tactical behavior patterns supported by topological structures in soccer-agent domains. In Padgham, Parkes, M. and Parsons, editors, *AAMAS 2008*, volume 4953, pages 1421-1424. International Foundation for Autonomous Agents and Multiagent Systems.
- [Rao, 1994] Rao, A. S. (1994). Means-end plan recognition - towards a theory of reactive recognition. In *KR*, pages 497-508.
- [Rickel and Johnson, 1999] Rickel, J. and Johnson, W. L. (1999). Animated agents for procedural training in virtual reality: Perception, cognition and motor control. *Applied Artificial Intelligence*, 13(4-5):343-382.
- [Riley, 1999] Riley, P. (1999). Classifying adversarial behaviors in a dynamic, inaccessible, multi-agent environment. Technical Report CMU-CS-99-175, Carnegie Mellon University.
- [Riley and Veloso, 2000] Riley, P. and Veloso, M. M. (2000). On behavior classification in adversarial environments. In Parker, L. E., Bekey, G. A., and Barhen, J., editors, *DARS*, pages 371-380. Springer.
- [Riley and Veloso, 2001] Riley, P. and Veloso, M. M. (2001). Recognizing probabilistic opponent movement models. In 0002, A. B., Coradeschi, S., and Tadokoro, S., editors, *RoboCup*, volume 2377 of *Lecture Notes in Computer Science*, pages 453-458. Springer.
- [Riley and Veloso, 2002] Riley, P. and Veloso, M. M. (2002). Planning for distributed execution through use of probabilistic opponent models. In Ghallab, M., Hertzberg, J., and Traverso, P., editors, *AIPS*, pages 72-82. AAAI.
- [RoboCup, 2008a] RoboCup (2008a). *RoboCup Challenge: Opponent Modeling* (<http://www.robocup.org/research/5114.html>). Fecha de acceso: Junio, 2008.
- [RoboCup, 2008b] RoboCup (2008b). *Sitio oficial de RoboCup*. (<http://www.robocup.org>). Fecha de acceso: Junio, 2008.
- [Sahota and Mackworth, 1994] Sahota, M. and Mackworth, A. (1994). Can situated robots play soccer? In *Proceedings of Canadian AI-94*, pages 249-254. Laboratory for Computational Intelligence, Department of Computer Science.

- [Scerri and Ydrén, 2000] Scerri, P. and Ydrén, J. (2000). End user specification of robocup teams. In Veloso, M. M., Pagello, E., and Kitano, H., editors, *RoboCup*, volume 1856 of *Lecture Notes in Computer Science*, pages 450-459. Springer.
- [Sean Buttinger, 2001] Sean Buttinger, Marco Diedrich, L. H. (2001). The dirty dozen team and coach description. In 0002, A. B., Coradeschi, S., and Tadokoro, S., editors, *RoboCup*, volume 2377 of *Lecture Notes in Computer Science*, pages 543-546. Springer.
- [Shannon, 1950] Shannon, C. E. (1950). Programming a computer for playing chess. *Philosophical Magazine*, 41:256-275.
- [Steffens, 2002] Steffens, T. (2002). Feature-based declarative opponent-modelling in multi-agent systems.
- [Steffens, 2004] Steffens, T. (2004). Adapting similarity measures to agent types in opponent modelling.
- [Stone et al., 2000a] Stone, P., Riley, P., and Veloso, M. M. (2000a). The cmunited-99 champion simulator team. In *AI Magazine*, volume 21, pages 33-40.
- [Stone et al., 2000b] Stone, P., Riley, P., and Veloso, M. M. (2000b). Defining and using ideal teammate and opponent agent models. In *AAAI*, pages 1040-1045. AAAI Press / The MIT Press.
- [Sukthankar and Sycara, 2006] Sukthankar, G. and Sycara, K. (2006). Simultaneous team assignment and behavior recognition from spatio-temporal agent traces. In *AAAI*. AAAI Press.
- [Sukthankar and Sycara, 2008] Sukthankar, G. and Sycara, K. (2008). Robust and efficient plan recognition for dynamic multi-agent teams. In *Proc. of 7th International Conference on Autonomous Agents and Multiagent Systems AAMAS 2008*, pages 1383-1386. International Foundation for Autonomous Agents and Multiagent Systems.
- [Sukthankar and Sycara, 2005] Sukthankar, G. and Sycara, K. P. (2005). A cost minimization approach to human behavior recognition. In Dignum, F., Dignum, V., Koenig, S., Kraus, S., Singh, M. P., and Wooldridge, M., editors, *AAMAS*, pages 1067-1074. ACM.
- [Suryadi and Gmytrasiewicz, 1999] Suryadi, D. and Gmytrasiewicz, P. J. (1999). Learning models of other agents using influence diagrams.
- [Sycara, 1998] Sycara, K. (1998). Multiagent systems. *AI Magazine*, 10(2):79-93.
- [Tambe et al., 2000] Tambe, M., Raines, T., and Marsella, S. (2000). Agent assistants for team analysis. *AI Magazine*, 21(3):27-31.

- [Tambe and Rosenbloom, 1995a] Tambe, M. and Rosenbloom, P. S. (1995a). Architectures for agents that track other agents in multi-agent worlds. In Wooldridge, M., Muller, J. P., and Tambe, M., editors, *ATAL*, volume 1037 of *Lecture Notes in Computer Science*, pages 156-170. Springer.
- [Tambe and Rosenbloom, 1995b] Tambe, M. and Rosenbloom, P. S. (1995b). Event tracking in a dynamic multi-agent environment. *Computational Intelligence*, 12(3).
- [Tambe and Rosenbloom, 1995c] Tambe, M. and Rosenbloom, P. S. (1995c). Resc: An approach for real-time, dynamic agent tracking. In *International Joint Conference on Artificial Intelligence*, pages 103-111.
- [Visser et al., 2000] Visser, U., Drucker, C., Hiibner, S., Schmidt, E., and Weland, H.-G. (2000). Recognizing formations in opponent teams. In Stone, P., Balch, T. R., and Kraetzschmar, G. K., editors, *RoboCup*, volume 2019 of *Lecture Notes in Computer Science*, pages 391-396. Springer.
- [Visser and Weland, 2002] Visser, U. and Weland, H.-G. (2002). Using online learning to analyze the opponents behavior. In Kaminka, G. A., Lima, P. U., and Rojas, R., editors, *RoboCup*, volume 2752 of *Lecture Notes in Computer Science*, pages 78-93. Springer.
- [Wang Qinghua, 2000] Wang Qinghua, Y. B. (2000). Agent brigade in dynamic formation of robotic soccer. In of the 3rd World of Intelligent Control, P. and Automation, editors, *Intelligent Control and Automation*, volume 1, pages 174-178. IEEE.
- [Weiss, 1999] Weiss, G., editor (1999). *Multiagent systems: a modern approach to distributed artificial intelligence*. MIT Press, Cambridge, MA, USA.
- [Wooldridge et al., 2000] Wooldridge, M., Jennings, N. R., and Kinny, D. (2000). The gaia methodology for agent-oriented analysis and design. *Autonomous Agents and Multi-Agent Systems*, 3(3):285-312.
- [Yuen, 2001] Yuen, A. (2001). Lazarus team description. In Andreas Birk, Silvia Coradeschi, S. T., editor, *RoboCup*, volume 2377 of *Lecture Notes in Computer Science*, pages 522-525. Springer.
- [Yuen, 2002] Yuen, A. (2002). Evaluation of robocup team formation recognition using neural networks.