

UNIVERSIDAD AUTÓNOMA DE SINALOA



FACULTAD DE INFORMÁTICA CULIACÁN FACULTAD DE CIENCIAS DE LA TIERRA Y EL ESPACIO POSGRADO EN CIENCIAS DE LA INFORMACIÓN

“DISEÑO DE UN MODELO BASADO EN AGENTES PARA LA SIMULACIÓN DEL
CRECIMIENTO URBANO INTEGRANDO AUTÓMATAS CELULARES Y AGENTES
COGNITIVOS”

Que como requisito para obtener el grado de
DOCTOR EN CIENCIAS DE LA INFORMACIÓN

PRESENTADO POR:

Álvaro Peraza Garzón

DIRECTOR DE TESIS:

Dr. René Rodríguez Zamora

Dr. Wenseslao Plata Rocha

Culiacán Sinaloa, Marzo 2020

*Para mi familia.
En especial, para los dos angelitos...*

Agradecimientos

El desarrollo de este trabajo ha significado un gran reto, el camino no fue fácil y sobre todo no hubiera sido posible transitarlo sin la ayuda y paciencia (sobre todo esto último) de un grupo al cual le tengo mucho cariño y de quienes estaré eternamente agradecido.

A mi incansable esposa e hija, son lo más importante en mi vida.

A mis hermanos y padres, gracias a su apoyo he alcanzado una meta más.

A mis directores de tesis, Dr. René Rodríguez Zamora y Dr. Wenseslao Plata Rocha, por su valioso apoyo, increíbles enseñanzas y sobre todo... asombrosa paciencia hacia mí.

A mis compañeros de posgrado, por su gran ayuda, los recuerdo con mucho cariño.

Al equipo de profesores de la Facultad de Informática Culiacán y Facultad de Ciencias de la Tierra y el Espacio, por su gran profesionalismo y calidad en sus materias.

Al CINVESTAV, por sus increíbles atenciones durante la estancia, en especial al Dr. Sergio Víctor Chapa Vergara.

A mis compañeros de trabajo porque cada día aprendo con ustedes y han sido un ejemplo de superación y amistad.

A la Universidad Autónoma de Sinaloa y al Instituto Tecnológico de Mazatlán, por la confianza otorgada al brindarme esta oportunidad de desarrollo profesional, y las facilidades para finalizar esta etapa.

ÍNDICE GENERAL

AGRADECIMIENTOS	II
RESUMEN	V
ABSTRACT	VI
ÍNDICE DE FIGURAS	VII
ÍNDICE DE TABLAS	IX
PARTE I INTRODUCCIÓN	10
CAPÍTULO 1: INTRODUCCIÓN	10
1.1 ANTECEDENTES	12
1.2 PLANTEAMIENTO DEL PROBLEMA	18
1.3 PREGUNTAS DE INVESTIGACIÓN	20
1.4 OBJETIVOS	20
1.4.1 Objetivo general	20
1.4.1 Objetivos específicos	20
1.5 RELEVANCIA DE LA INVESTIGACIÓN	21
PARTE II MARCO TEÓRICO	22
CAPÍTULO 2: TÉCNICAS DE MODELADO	22
2.1 MODELOS DE CRECIMIENTO URBANO	22
2.1.1 Simulación prospectiva	23
2.1.2 Escenarios	24
2.1.3 Factores de Crecimiento	25
2.1.4 Variables Territoriales	26
2.1.5 Variables Socioeconómicas	26
2.2 AUTÓMATAS CELULARES	27
2.2.1 Características de un AC	27
2.2.2 AC para Crecimiento Urbano	29
2.3 SISTEMAS MULTIAGENTE	31
2.3.1 Agente Cognitivo	32
2.3.2 Agente para Simular Cuestiones Sociales	33
2.3.3 Aspectos a Modelar	34
2.3.4 Metodologías para la construcción de Agentes	35
2.4 COSTO COMPUTACIONAL	37
2.4.1 OpenMP	38
2.4.2 El modelo fork-Join	38
CAPÍTULO 3: REDES NEURONALES	40
3.1 EL MODELO BIOLÓGICO	41
3.1.1 Aspectos Básicos de una Neurona Biológica	41
3.1.2 Funcionamiento de una Neurona Biológica	42
3.1.3 Analogía con las Redes Neuronales Artificiales	44
3.2 EL MODELO ARTIFICIAL	45
3.2.1 Aspectos Básicos de una Neurona Artificial	46

3.2.2 Estructura de una Red Neuronal Artificial	53
3.2.3 Mecanismo de Aprendizaje	55
3.3 FUNCIONAMIENTO DE UNA RED NEURONAL ARTIFICIAL	58
3.3.1 El Perceptrón.....	58
3.3.2 El Perceptrón Multicapa	61
3.3.4 Aplicaciones en Crecimiento Urbano	62
PARTE III METODOLOGÍA.....	64
CAPÍTULO 4: ANÁLISIS DE LA ZONA DE ESTUDIO.....	66
4.1 DELIMITACIÓN DE LA ZONA URBANA.....	67
4.2 DIGITALIZACIÓN.....	69
4.2.1 Categorías de uso urbano	70
4.2.2 Generación de Mapas Vectoriales	70
4.2.3 Rasterización.....	74
4.2.4 Herramientas de Análisis	75
CAPÍTULO 5: CARACTERIZACIÓN DEL AUTÓMATA CELULAR	78
5.1 CÁLCULO DEL POTENCIAL DE TRANSICIÓN	79
5.2 REGLA DE EVOLUCIÓN	79
5.2.1 Definición de la Vecindad.....	80
5.2.2 Matrices de Calibración	82
5.3 IMPLEMENTACIÓN EN OPENMP	83
CAPÍTULO 6: DISEÑO DE LA RED NEURONAL	86
6.1 ESTRUCTURA DE LA RNA	86
6.1.2 Algoritmo de Entrenamiento	87
6.2 ENTRENAMIENTO	89
CAPÍTULO 7: DISEÑO DEL AGENTE COGNITIVO	91
7.1 BDI.....	91
7.2 GAMA Y GAML.....	92
PARTE IV EXPERIMENTACIÓN.....	94
CAPÍTULO 8: RESULTADOS.....	94
8.1 ANÁLISIS DE CAMBIOS DE USO DE SUELO URBANO	94
8.1.1 Cuantitativo.....	95
8.1.2 Espacial	97
8.1.3 Selección de los factores de influencia	102
8.1.4 Variables de Demanda.....	104
8.1.5 Espacialización de Datos Socioeconómicos.....	104
8.2 SIMULACIONES CON EL AUTÓMATA CELULAR	106
8.2.1 Matrices de Calibración Resultantes.....	108
8.3 SIMULACIONES CON LA RED NEURONAL INTEGRADA.....	113
8.4 VALIDACIÓN DE SIMULACIONES	118
8.5 PROYECCIÓN A 2030	¡ERROR! MARCADOR NO DEFINIDO.
8.6 DEFINICIÓN DEL AGENTE COGNITIVO	119
CAPÍTULO: 9 CONCLUSIONES	121
9.1 DISCUSIÓN.....	124
9.2 TRABAJO FUTURO.....	125
BIBLIOGRAFÍA	127
ANEXOS.....	132

Resumen

Los distintos componentes que convergen en el crecimiento urbano, representan un reto computacional y logístico, debido al comportamiento complejo que surge con las interrelaciones de los elementos vecinos. El modelado del crecimiento urbano, en el contexto de los sistemas complejos, es un fenómeno que conviene estudiarse con el uso de distintas técnicas, entre ellas los Autómatas Celulares y Redes Neuronales.

El avance de los sistemas computacionales han traído como resultado el incremento de las capacidades de cálculo y desarrollo de nuevos paradigmas de programación, esto ha propiciado la reducción de tiempos y costo computacional, y ha dado pie a que cada vez sean más frecuentes los estudios en donde se combinen distintas técnicas de modelado.

El objetivo de la investigación es presentar un nuevo modelo de agente cognitivo que forme parte de un modelo basado en agentes para llevar a cabo simulaciones de crecimiento urbano, integrando Autómatas Celulares (AC) y Redes Neuronales Artificiales (RNA).

Para el desarrollo de los experimentos, se utilizaron imágenes satelitales para su digitalización y clasificación, se implementaron modelos de programación paralela (Fork-Join) para la agilización de cálculos, se utilizó un modelo clásico de AC delimitado para crecimiento urbano y el algoritmo de opera la RNA es el de retropropagación, para darle forma al modelo de agente se usó el modelo Belief–Desire–Intention (BDI).

Abstract

The different components that converge in urban planning represents a computational and logistical challenge, due to the complex behavior that arises with the interrelationships of its neighbor elements. The modeling of urban growth, in the context of complex systems, is a phenomenon that should be studied using different techniques, including Cellular Automata and Neural Networks.

The advances in computer systems has resulted in the increase in calculation capabilities and the development of new programming paradigms, this has led to the reduction of computational time and cost, and has led to the fact that studies in where different modeling techniques are combined.

The objective of the research is to present a new cognitive agent model that is part of an agent-based model to carry out urban growth simulations, integrating Cellular Automata (AC) and Artificial Neural Networks (ANN).

For the development of the experiments, satellite images were used for their digitization and classification, parallel programming models (Fork-Join) were implemented to streamline calculations, a classic model of a delimited CA for urban growth and the algorithm of which runs the ANN is the Backpropagation, to construct the agent model the Belief–Desire–Intention (BDI) model was used.

ÍNDICE DE FIGURAS

FIGURA 1. PROCESO DE ENTRADA SALIDA AL SMA.....	13
FIGURA 2. DIAGRAMA DE LA ESTRUCTURA	14
FIGURA 3. DINÁMICA DEL MODELO.....	15
FIGURA 4. ESTRUCTURA DE EJECUCIÓN DE LA SIMULACIÓN.....	16
FIGURA 5. INTERACCIÓN ENTRE LOS COMPONENTES DEL SISTEMA.....	17
FIGURA 6. CAPAS TEMÁTICAS [24].....	25
FIGURA 7. ESPACIO DE EVOLUCIONES DE UN AC	27
FIGURA 8. VECINDAD DE NEUMANN Y MOORE	29
FIGURA 9: VENTANA DE OBSERVACIÓN DE 3X3 PÍXELES.	30
FIGURA 10. SE DETERMINA EL MAYOR POTENCIAL.....	30
FIGURA 11. MODELO GENERAL DE AGENTE [5]	31
FIGURA 12. MODELO GENERAL DE AGENTES QUE APRENDEN [5]	33
FIGURA 13. PROCESO GENERAL DE FORK-JOIN	38
FIGURA 14. REGIÓN PARALELA ANIDADA (NESTED)	39
FIGURA 15. ESTRUCTURA DE UNA NEURONA BIOLÓGICA [42]	42
FIGURA 16. CONCENTRACIÓN DE IONES DE SODIO (NA+)	43
FIGURA 17. DIRECCIÓN DE LA PROPAGACIÓN DEL IMPULSO NERVIOSO A TRAVÉS DEL AXÓN [43]	44
FIGURA 18. NEURONA MCCULLOCH – PITTS	45
FIGURA 19. MODELO GENÉRICO DE NEURONA ARTIFICIAL [50].....	47
FIGURA 20. INTERACCIÓN ENTRE UNA NEURONA PRESINÁPTICA Y OTRA POSTSINÁPTICA [50]	47
FIGURA 21. FUNCIÓN DE TRANSFERENCIA ESCALÓN.....	49
FIGURA 22. FUNCIONES DE ACTIVACIÓN MIXTA	50
FIGURA 23. FUNCIÓN DE ACTIVACIÓN CONTINUA	50
FIGURA 24. ESTRUCTURA MULTINIVEL DE RED NEURONAL.....	54
FIGURA 25. RED MONOCAPA [59].....	54
FIGURA 26. UNIDAD U_j [8].....	57
FIGURA 27. PERCEPTRÓN APLICADO A LA FUNCIÓN OR [8].....	59
FIGURA 28. PERCEPTRÓN MULTICAPA [68].....	61
FIGURA 29. REGIONES DE DECISIÓN GENERADAS POR UN PERCEPTRÓN MULTICAPA [8]	61
FIGURA 30. MODELO DE SIMULACIÓN BASADO EN RNA-AC [70]	63
FIGURA 31. RUTA DEL PROCESO.....	66
FIGURA 32. ZONA URBANA DEL MUNICIPIO DE CULIACÁN SINALOA	67
FIGURA 33. MOSAICO DE ORTOFOTOS DE LA ZONA URBANA DE CULIACÁN 1997 Y EL POLÍGONO DE ESTUDIO (EN ROJO)	69
FIGURA 34. MOSAICO DE ORTOFOTOS DE LA ZONA URBANA DE CULIACÁN 2004 Y EL POLÍGONO DE ESTUDIO (EN ROJO)	69
FIGURA 35. PROCESO DE DIGITALIZACIÓN DE LA ZONA URBANA SOBRE LA ORTOFOTO DE 1997	71
FIGURA 36. DETALLE DE USOS URBANOS	72
FIGURA 37. MAPAS VECTORIALES DE LA ZONA URBANA	73
FIGURA 38. RED VIAL 1997.....	74
FIGURA 39. RED VIAL 2004.....	74
FIGURA 40. MAPAS RÁSTER DE LA ZONA URBANA	75
FIGURA 41. CATEGORÍA CON SUPERFICIE INICIAL Y FINAL, MODIFICADO DE [75]	77
FIGURA 42. VARIABLES DE ENTRADA AL AUTÓMATA.....	78
FIGURA 43. PLANTILLA DE VECINDAD DE 11X11 (MOORE)	81

FIGURA 44. DISTANCIA MANHATTAN A CELDAS VECINAS, DESDE LA CELDA CENTRAL	82
FIGURA 45. MATRIZ DE CALIBRACIÓN	83
FIGURA 46. CALCULO DEL POTENCIAL DE TRANSICIÓN (UNO POR NUCLEO) DE UNA CELDA HACIA DOS USOS POSIBLES	84
FIGURA 47. CALCULO DE POTENCIALES DE TRANSICIÓN PARA n USOS	84
FIGURA 48. ARQUITECTURA RNA PARA EXPERIMENTOS	87
FIGURA 49. ESTRUCTURA PERCEPTRÓN DE 3 CAPAS: ENTRADA = I, OCULTA = J, SALIDA = K.....	88
FIGURA 50. LÓGICA BDI	91
FIGURA 51. CONFIGURACIONES MÍNIMAS DE DEPENDENCIAS DE PROYECTOS.....	93
FIGURA 52. TABULACIÓN CRUZADA DEL PERIODO 1997 A 2004.....	95
FIGURA 53. TABULACIÓN CRUZADA DEL PERIODO 2004 A 2011	96
FIGURA 54. MAPA DE CAMBIOS DE 1997 AL 2004	97
FIGURA 55. CALCULO DE LA APTITUD COMERCIAL.....	99
FIGURA 56. MAPAS DE APTITUD PARA DISTINTOS USOS URBANOS	100
FIGURA 57. PRIORIDADES DE URBANIZACIÓN.....	101
FIGURA 58. MAPAS DE DISTANCIA A CARRETERAS, PROXIMIDAD A ZONAS DE INTERÉS Y ACCESIBILIDAD	102
FIGURA 59. ESPACIALIZACIÓN DE DATOS A PARTIR DE DATOS TABULADOS	105
FIGURA 60. VARIABLES SOCIOECONÓMICAS ESPACIALIZADAS	106
FIGURA 61. SIMULACIÓN EX POST DE 1997 A 2004 CON EL AC.....	107
FIGURA 62. PRIMERAS SIMULACIONES CON EL AC	110
FIGURA 63. VARIABLES SOCIOECONÓMICAS ESPACIALIZADAS COMO ENTRADA A LA RNA, Y LA SALIDA DE LA MISMA COMO ENTRADA AL MODELO	113
FIGURA 64. SIMULACIÓN EX POST DE 1997 A 2004, CON EL AC Y RNA INTEGRADA.....	114
FIGURA 65. DISEÑO FINAL DE LA RED NEURONAL ARTIFICIAL Y SUS INTERCONEXIONES.....	115
FIGURA 66. ESQUEMA DE ENTRENAMIENTO POR PIXEL.....	116
FIGURA 67. ESQUEMA DE CLASIFICACIÓN POR PIXEL.....	117
FIGURA 68. MAPA SIMULADO AL 2011 AC + RNA.....	118
FIGURA 69. DIAGRAMA DE ACTIVIDADES	120

ÍNDICE DE TABLAS

TABLA 1. MODELOS DE SIMULACIÓN PROSPECTIVA DE CRECIMIENTO URBANO [23].	23
TABLA 2. METODOLOGÍAS PARA EL DESARROLLO DE SISTEMAS MULTIAGENTE	36
TABLA 3. CLASIFICACIÓN DE USOS DE SUELO URBANO.	70
TABLA 4. MATRIZ DE TABULACIÓN CRUZADA PARA DOS MAPAS DE DIFERENTE FECHA [29]....	76
TABLA 5. VARIABLES SOCIOECONÓMICAS	90
TABLA 6. CAMBIOS EN EL PERIODO 1997 A 2004	95
TABLA 7. CAMBIOS EN EL PERIODO 2004 A 2011	96
TABLA 8. RESULTADOS ROC.....	104
TABLA 9. DATOS DE ENTRADA AL AC	108
TABLA 10. VALORES K PARA LAS MATRICES	108
TABLA 11. DEFINICIÓN DEL MBA CON AGENTE HABITANTE Y AGENTE PLANIFICADOR.....	119

PARTE I

INTRODUCCIÓN

CAPÍTULO 1: INTRODUCCIÓN

La evolución del uso de suelo en el territorio es un elemento fundamental en nuestra sociedad, en él se manifiestan distintas variables que afectan nuestra vida cotidiana, como la accesibilidad a distintos puntos de interés dentro de la ciudad, las pendientes del terreno que determinan zonas de riesgo, entre otros. Esta evolución ha cobrado interés, potenciada por los distintos problemas ambientales y especialmente aquellos en áreas urbanas y paisajes modificados por el hombre [1], que además, gracias a los avances en el campo de la informática y del desarrollo de importantes herramientas de análisis como los Sistemas de Información Geográfica (SIG) o los modelos de simulación, impulsan el estudio de los cambios que tienen lugar en las áreas metropolitanas . El análisis y desarrollo de nuevos instrumentos de planificación, ha causado que distintas disciplinas, específicamente la Inteligencia Artificial (IA), aborden desde un punto de vista informático y matemático, posibles soluciones alternativas a este problema en el ámbito de los sistemas complejos [2].

Un sistema complejo consta de distintos componentes que interactúan entre sí, dando como resultado un comportamiento emergente, es decir, el comportamiento del sistema no se puede inferir simplemente del comportamiento individual de sus componentes. La cantidad de información necesaria para describir el comportamiento de tal sistema es una medida de su complejidad. Las redes neuronales y el crecimiento urbano, son ejemplos de sistemas complejos,

en el primero, cada uno de sus elementos (neuronas) interactúan entre sí a través de sus conexiones (sinapsis), su actividad en conjunto es la de generar conocimiento; para el caso del crecimiento urbano, se involucran distintos factores (territoriales y espaciales), los cuales influyen para que el crecimiento se presente con determinadas características. Existen distintas técnicas que permiten modelar sistemas complejos, una de las más usadas, para el caso de las dinámicas urbanas, son los Autómatas Celulares [3].

Un autómata celular (AC), es un modelo matemático espacial y discreto, se caracteriza por su interacción a través de una malla n -dimensional, con sus celdas vecinas. Los modelos de crecimiento urbano, basados en AC, están orientados fundamentalmente a la representación de los atributos de una región geográfica específica sobre en una malla bidimensional, en la cual se establece un radio de vecindad y se aplica una regla de evolución para definir el comportamiento del AC [4].

Recientemente se ha explorado el uso de AC con otras técnicas de modelado, específicamente con modelos basados en agentes inteligentes, esto con el objetivo de abordar ciertas características que para un AC serían complicadas de modelar, tal es el caso de los componentes socioeconómicos del crecimiento urbano.

El término, agente, se ha popularizado dentro de los sistemas de información para describir a entidades computacionales. Estas entidades tienen la capacidad de interactuar con su entorno, perciben información de él y realizan acciones cuando es requerido, su capacidad para tomar decisiones les denota cierta inteligencia, en algunos casos también son llamados agentes inteligentes [5].

Los modelos basados en agentes (MBA) son capaces de interactuar entre ellos, formando sistemas multiagente, estos sistemas son usados mayormente como herramientas de simulación [6]. Los MBA están siendo cada vez más utilizados en la simulación de sistemas espaciales complejos, como es el caso del crecimiento urbano. Consiste en un modelo celular que considera el comportamiento de los agentes del sistema para realizar una simulación. Sus principales

componentes son los propios agentes, el entorno y las relaciones que existen entre ellos, plasmadas en el modelo a través de reglas de decisión [7].

Uno de los modelos que otorga a un agente de una mayor capacidad de decisión, son las Redes Neuronales Artificiales (RNA), estas toman los aspectos más importantes de las neuronas biológicas, emulando de cierta manera las funcionalidades del cerebro humano.

Desde sus inicios, la I.A ha tenido como modelo las funcionalidades inteligentes del ser humano, su motivación es el de construir máquinas que piensen como un humano o de emularlo en algunas de sus funciones. Una de las ramas nuevas y prometedoras de la I.A. corresponde a las redes de autoproceso, éstas son construidas a través de nodos procesadores de información, los cuales de su interacción depende el comportamiento del sistema. Bajo este tipo de redes se encuentran las RNA [8].

Las RNA están siendo utilizadas para una gran cantidad de aplicaciones de reconocimiento de patrones y análisis de información para su predicción y clasificación. Uno de los objetivos más ambiciosos de los científicos a lo largo de la historia ha sido el de crear máquinas que denoten inteligencia.

1.1 ANTECEDENTES

La integración de agentes y autómatas celulares como tecnologías que coexisten en la simulación de crecimientos urbanos se presentan en diversos escenarios[9] sin embargo, el poder de aprendizaje y clasificación de las RNA al parecer no se ha popularizado como herramienta para incrementar la capacidad de aprendizaje de un agente, esto en ambientes multiagentes y modelos de simulación.

El uso de estas tecnologías en modelos de crecimiento urbano, implica el análisis de los distintos factores que integran la dinámica urbana, desde naturales hasta artificiales. La cantidad de los factores incluidos en un modelo de simulación tiene un impacto directo en los costos

computacionales debido a la gran cantidad de cálculos que hay que realizar con cada uno de estos factores. Para reducir la carga computacional, se emplean técnicas como la regresión espacial, que ayudan a seleccionar aquellos factores más representativos con el objetivo de modelar una cantidad reducida de estos, pero obteniendo resultados similares a los observados.

A continuación se mencionan algunas investigaciones en donde se integran la mayoría de los modelos antes citados, en estos trabajos se pone de manifiesto la versatilidad que se alcanza al combinarlas para la solución de distintos planteamientos.

El trabajo “A Neural Network Multiagent Architecture Applied to Industrial Networks for Dynamic Allocation of Control Strategies Using Standard Function Blocks” [10], se presenta una estructura y la implementación de una red neuronal en un agente inteligente, esto con el objetivo de formar un sistema multiagente (SMA) (Figura 1) que sea capaz de detectar problemas en una línea de producción reduciendo al mínimo la intervención humana, se logra usando RNA, las cuales permiten que los agentes aprendan sobre los patrones que pueden ocasionar un problema y que este aprendizaje les permita manejarlo de manera correcta.

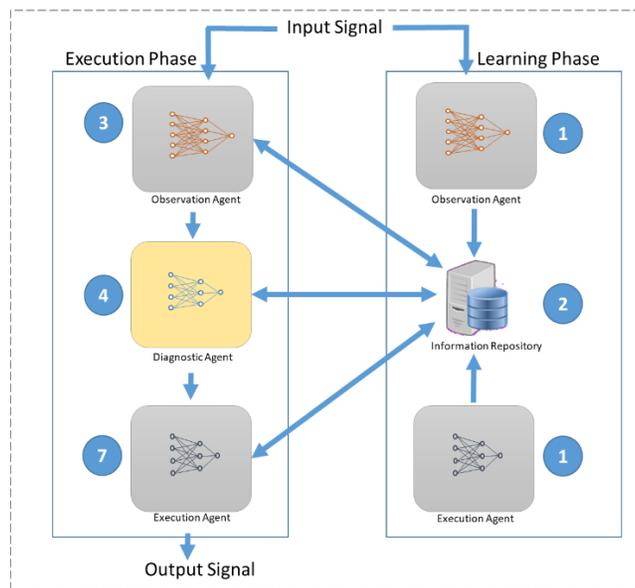


Figura 1. Proceso de Entrada Salida al SMA

El trabajo desarrollado para la comunidad de Cerrado en Brasil “A multi-agent model system for land-use change simulation” [11], no hace uso de RNA, la implementación del sistema multiagente sirvió como base para la presente investigación, y consiste en el movimiento de los agentes sobre un autómata celular, el cual sirve como entorno para el agente. Es decir, se tiene un autómata celular que simula el cambio en el uso de suelo, utiliza factores de entrada como: pendiente, tipo de vegetación, etc., y además se tiene un sistema multiagente que simula el factor “político” en donde se modelan las decisiones políticas, de esta manera, se tiene un esquema (Figura 2) de más de dos dimensiones en donde, el AC sirve de entorno para que el SMA interactúe con él. Las conclusiones de éste trabajo arrojan resultados más “realistas” con respecto a otros con los cuales fue comparado.

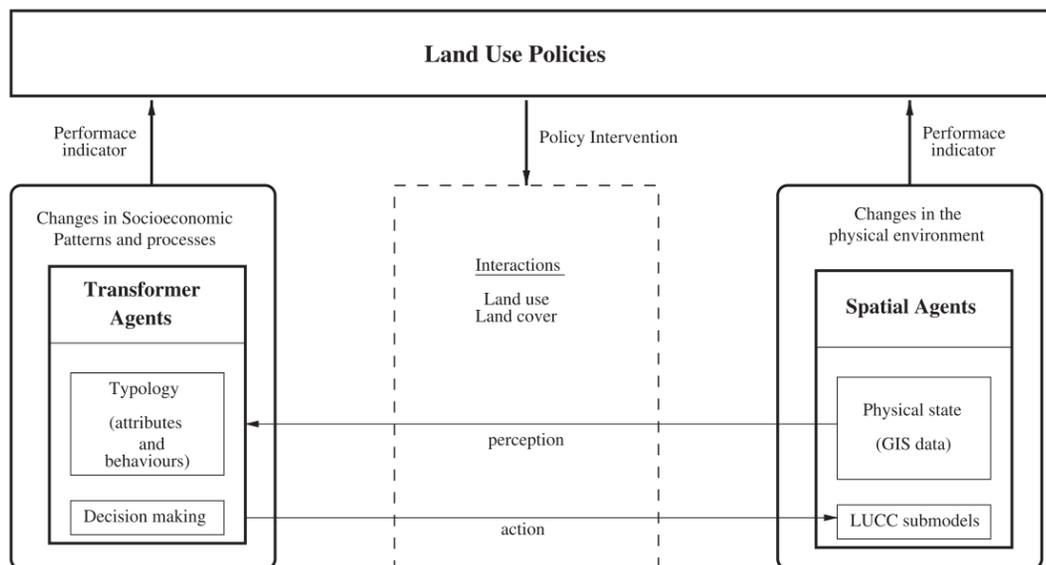


Figura 2. Diagrama de la estructura

De los trabajos antes mencionados, el primero utiliza métodos para la implementación de una RNA en un SMA y el segundo emplea un SMA en el que se representa el medio ambiente a través de un AC. Existen varios trabajos en donde agentes y autómatas celulares coexisten en diversas simulaciones, tal es el caso del trabajo "Simulación de sociedades artificiales de agentes y autómatas celulares en Ingeniería de Organización: aplicación a la gestión del agua" [12], en el que se aborda la elaboración de un modelo de simulación social basado en agentes para el estudio del balance recursos-demanda doméstica de agua en la región Metropolitana de

Barcelona (Figura 3). Este estudio se lleva a cabo mediante el análisis de escenarios de balance en el modelo y mediante el estudio del impacto de los procesos sociales en la demanda doméstica de agua. El simulador se centra en el consumo doméstico de agua y en la exploración de nuevas políticas para su evaluación.

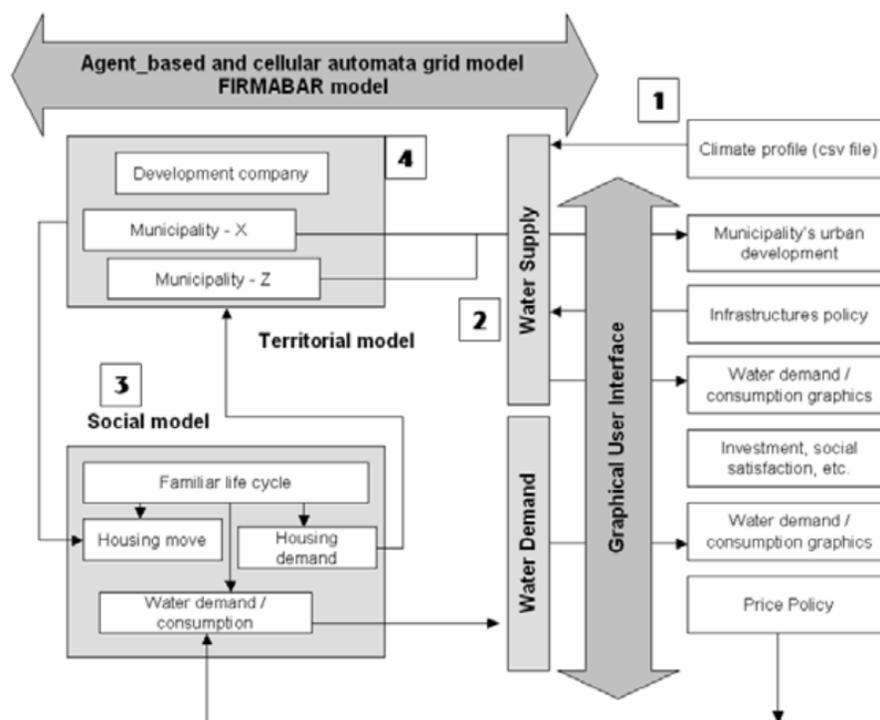


Figura 3. Dinámica del Modelo

Existe otro trabajo que combina ambas tecnologías “A Multi-Agent Cellular Automata System for Visualising Simulated Pedestrian Activity” [13], en el cual se plantean las primeras impresiones con respecto al desarrollo de un sistema multiagente el cual puede ser usado para simular actividad peatonal. El sistema está basado en autómatas celulares y agentes inteligentes, los agentes representan objetos o personas con su propio comportamiento, y estos se mueven a través de una red peatonal. Cada agente se encuentra en un espacio simulado sobre una cuadrícula representada por un AC. Cada iteración de la simulación se basa en una actualización paralela de los agentes siguiendo las normas locales.

Para el caso de agentes inteligentes con redes neuronales integradas existe el trabajo "Sistema Clasificador Multiagente Basado en una Red Neuronal", la contribución principal de esta investigación es un nuevo método de medición de la confianza, bajo ciertos indicadores en una negociación. Los resultados fueron comparados con un modelo de razonamiento de negociación llamado TNC (trust negotiation and communication) obteniendo resultados favorables [14].

El estudio "Modelado y simulación basada en agentes con SIG para la gestión de agua en espacios metropolitanos" [15], describe la arquitectura y las principales características de un modelo basado en agentes acoplado con un sistema de información geográfico para la estimación del consumo de agua doméstica, en la región metropolitana de Valladolid. El modelo (Figura 4) es una adaptación inicial del modelo FIRMABAR [16] basado en agentes y autómatas celulares. En el trabajo se repasan las principales técnicas de estimación de agua doméstica existentes y se argumentan las razones para complementarlas con la metodología basada en agentes acoplada con sistemas de información geográficos.

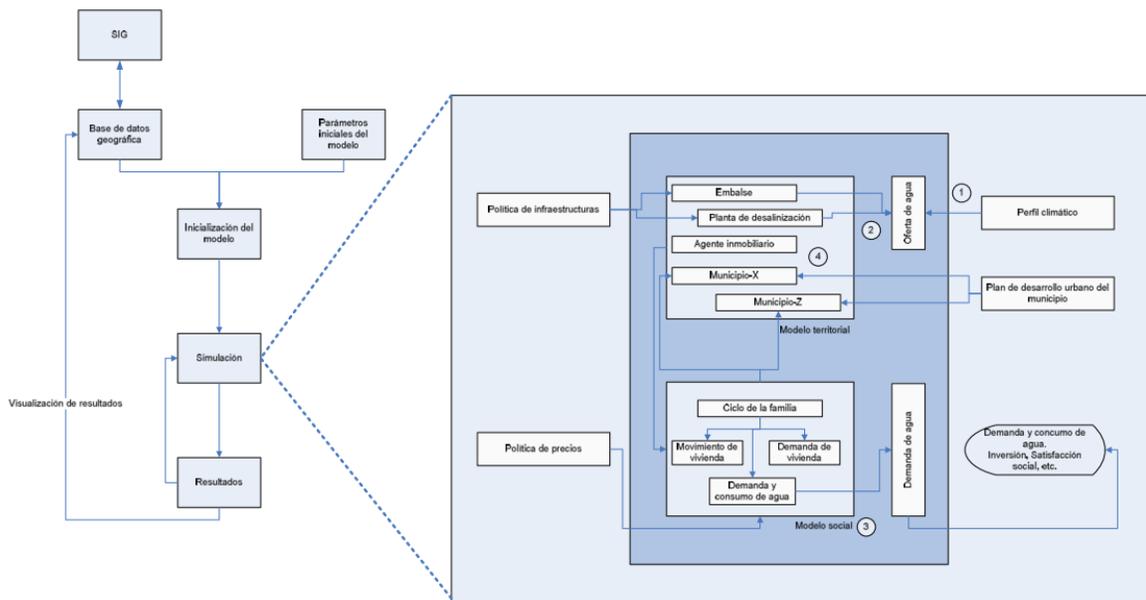


Figura 4. Estructura de ejecución de la simulación

El trabajo "PUMA (Predicting Urbanisation with Multi-Agents): MULTI-AGENT MODELLING OF URBAN SYSTEMS" [17], presenta la especificación de un modelo, su

calibración y su experimento llevado a cabo en la conurbación de Randstad situada en los Países Bajos. Se entiende cada vez más que los procesos de cambio en el uso del suelo son el resultado de las decisiones tomadas por actores individuales, como los propietarios de las tierras, autoridades, empresas etc. Con el fin de mejorar el uso del suelo se encuentra en fase de desarrollo el proyecto PUMA, el cual contiene varios módulos que representan el comportamiento de los actores, algunos otros módulos describen los hogares, el trabajo, carreras en relación con los eventos del ciclo de vida (matrimonio, nacimiento de un hijo, el envejecimiento, cambio de trabajo, etc.) y también sus patrones de actividad diaria. El módulo de empresas incluye la demografía de las empresas y su demanda relacionada con las instalaciones de producción.

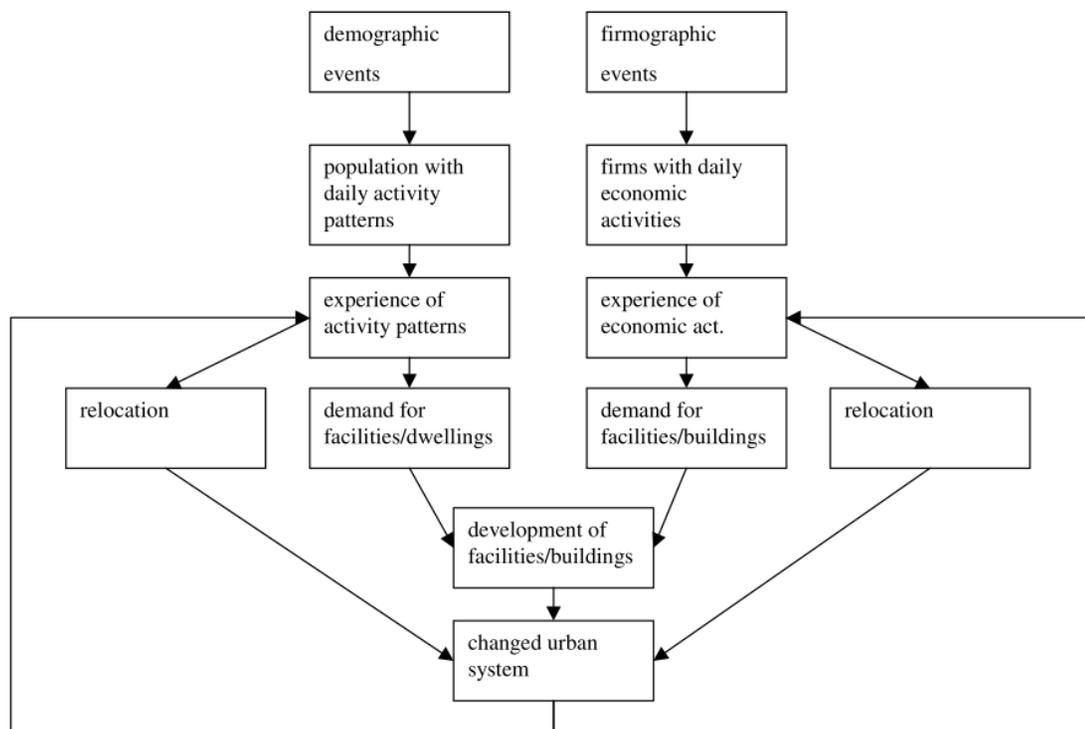


Figura 5. Interacción entre los componentes del sistema

Con base en las investigaciones analizadas, se pueden resaltar varios aspectos; la combinación de agentes, autómatas celulares y RNA permiten abordar una gran cantidad de temáticas (desde aplicaciones industriales hasta de carácter urbanístico), tales herramientas proporcionan un ambiente robusto en donde la selección de los factores adecuados como variables de entrada, producen resultados aceptables. Sin embargo, como se verá más adelante

en el capítulo 2.2 (Paralelización de Cálculos), queda manifiesto que, la combinación de estas técnicas de modelado junto con el aumento de las variables de entrada, conlleva a un incremento significativo del costo computacional. Esto es particularmente importante, debido a que cualquier planteamiento en donde se integren estos modelos de simulación, debe considerarse también, la cantidad de cálculos necesarios para producir resultados.

1.2 PLANTEAMIENTO DEL PROBLEMA

En los últimos años han surgido numerosos instrumentos de modelización. Atendiendo a los propiamente espaciales, en el caso del crecimiento urbano son los modelos basados en AC los más ampliamente utilizados, tal es el caso de los trabajos [3], [4] aunque también se pueden encontrar modelos de regresión, redes neuronales, técnicas de EMC y todavía de manera incipiente modelos basados en agentes (MBA).

Los modelos basados en AC fundamentalmente están orientados hacia la representación de los atributos de una determinada región geográfica en una malla bidimensional en la que se define un radio de vecindad sobre el cual se aplica una determinada regla de evolución, con el uso de estos modelos ha sido posible generar escenarios territoriales de manera prospectiva.

Además de los modelos basados en AC, han surgido propuestas en las cuales se combina su uso con MBA, en las cuales se tiene por un lado, modelado el territorio a través del AC y por otro lado se modelan las entidades que actúan sobre ese territorio usando MBA. Sin embargo, las contribuciones en este tema disminuyen a medida que se buscan aplicaciones específicas en donde se combinen ambos modelos, particularmente lo que tiene que ver con el uso de MBA.

En este orden de ideas, se han planteado modelos que hacen uso de autómatas celulares, sistemas multiagentes y redes neuronales, tratando de complementar su aplicación en el estudio de sistemas complejos para diversos propósitos haciendo uso de la simulación como una herramienta de análisis prospectivo. Aunque estos modelos se han aplicado en disciplinas como

ecología, economía, demografía, etc., todavía está infra explorada su aplicación en la simulación sobre crecimiento urbano dentro del contexto de la generación de escenarios territoriales [12].

Aunque existen trabajos en donde se hace uso de Autómatas Celulares para simular el ordenamiento territorial, y otros donde se combinan con otras técnicas como los SMA [13], [17]–[21], existe una carencia de investigaciones en donde se explore la integración de un modelo formal de agente que implemente una RNA para dotar de las capacidades necesarias para la simulación de cuestiones sociales.

También es importante destacar que la generación y predicción de escenarios territoriales implica reducir una brecha existente entre el modelo de agentes inteligentes y el aprendizaje que debería incorporar a partir de un medio ambiente real. Esto sugiere el uso de una Red Neuronal para este propósito de acuerdo a lo que se plantean en los trabajos [5]–[8].

La aplicación de MBA para simulaciones está extendiéndose debido a su elevado potencial de modelización, lo que facilita el entendimiento de fenómenos complejos. Sin embargo, el uso de agentes inteligentes en la generación de escenarios territoriales está poco explotado.

Los resultados obtenidos en la presente investigación, contribuyen a potenciar el uso de agentes inteligentes en modelos de simulación que aborden la generación de escenarios territoriales. De esta manera se tiene la posibilidad de contribuir en el manejo sustentable de recursos, debido a que sería posible que, con la ayuda de simulaciones prospectivas del área de estudio, se tomen las decisiones pertinentes.

1.3 PREGUNTAS DE INVESTIGACIÓN

¿Cómo definir de manera formal un agente que modele el aprendizaje a través de una red neuronal para la predicción y generación de escenarios territoriales, tomando en consideración los distintos factores sociales que pueden afectar el ordenamiento del territorio?

¿Cómo construir un SMA en donde los agentes se integren y coexistan considerando en la definición de los mismos el uso de una RNA que modele el aprendizaje?

¿Qué características debe tener el AC que modele el ambiente el cual servirá de entorno al SMA?

1.4 OBJETIVOS

1.4.1 Objetivo general

Definir un SMA para crecimiento urbano que incluya el factor social mediante el uso de agentes cognitivos que interactúen en un medio ambiente representativo de una región geográfica modelada por un AC.

1.4.1 Objetivos específicos

- Caracterizar un AC que modele la zona urbana de Culiacán.
- Determinar la red neuronal más adecuada para el modelado del aprendizaje de un agente tomando en consideración factores sociales.
- Definir de manera formal un agente cognitivo que modele el aprendizaje.
- Construir un SMA con agentes cognitivos que interactúe con un AC geográfico.
- Desarrollar simulaciones prospectivas de la zona urbana de Culiacán.

La relevancia de esta investigación se puede dimensionar identificando sus impactos científico, social, y económico.

1.5 RELEVANCIA DE LA INVESTIGACIÓN

Impacto Científico

Evidentemente los resultados de todo proyecto de investigación deben tener un impacto positivo para el desarrollo del estado del arte. En este sentido, son varias las contribuciones que se obtienen como resultado de esta investigación.

La principal contribución de esta investigación que tiene un impacto científico es:

- Un nuevo modelo basado en agentes neuronales que integre AC, RNA, tomando en cuenta el factor social para para la simulación geoespacial de escenarios territoriales.

Impacto Social

El impacto social se desprende directamente de producir herramientas computacionales que de alguna manera ayuden en la planificación y ordenamiento de los recursos que integran una región geográfica beneficiando a una comunidad en específico.

Impacto Económico

El impacto económico derivado de este nuevo modelo de agentes, permitiría ayudar en la administración eficiente de los recursos a un bajo costo, puesto que los estudios, experimentos y proyecciones se realizarían por medio de simulaciones, es decir, en un entorno virtual. Con esto podemos disminuir considerablemente las limitaciones logísticas o financieras que pudieran existir antes de ser implementados.

PARTE II

MARCO TEÓRICO

CAPÍTULO 2: TÉCNICAS DE MODELADO

2.1 MODELOS DE CRECIMIENTO URBANO

Los modelos de simulación en lo que se ha denominado como simulación prospectiva, suelen emplearse de forma conjunta con el diseño de escenarios, entendidos estos como “imágenes hipotéticas del futuro, lógicas y plausibles, formuladas para mostrar posibles situaciones consecuencia de procesos causales o de toma de decisiones” [22].

Los futuros representados o imaginados a través de escenarios suelen corresponderse generalmente con situaciones probables, posibles o deseables, que dan lugar a diversos modos de pensar el futuro. Así, se puede pensar el futuro tratando de dar respuesta a preguntas del tipo ¿qué pasará?, a preguntas como ¿qué puede pasar? o ¿qué debería pasar? En relación con ellas, se pueden identificar tres formas o modos principales de pensamiento futuro [23]:

- El pensamiento predictivo intenta mostrar lo que pasará en un futuro, generalmente de carácter cercano, a través de la indagación de las posibilidades de evolución más probables.

- El pensamiento exploratorio está caracterizado por tratar de mostrar situaciones posibles de evolución futura basadas en la ocurrencia/ausencia de determinados eventos o decisiones tomadas.
- El pensamiento normativo trata por su parte de imaginar futuros “deseables” de acuerdo con un conjunto de diversos criterios. En este sentido, el énfasis reside en el establecimiento de parámetros deseables y en la forma de alcanzarlos.

Estos modos de pensamiento futuro, pueden constituir la base para el establecimiento de una clasificación de los diferentes escenarios.

2.1.1 Simulación prospectiva

Los modelos de mayor aplicación y desarrollo de los últimos años en el ámbito de la simulación del crecimiento urbano se detallan en la siguiente tabla:

Tabla 1. Modelos de simulación prospectiva de crecimiento urbano [23]

Escenarios	Modelo	Características
Exploratorios	Autómatas celulares	Definidos generalmente en una malla ráster, los modelos basados en AC permiten simular el crecimiento urbano en los diferentes píxeles a partir del efecto de atracción/repulsión ejercido por los píxeles en su vecindad. Son capaces de generar estructuras complejas y representar dinámicas urbanas.
	Modelos basados en agentes	Consideran el comportamiento de los agentes involucrados en los procesos de crecimiento urbano (a partir de un estado inicial dado), además de las relaciones entre los agentes, y entre ellos y su medio. La configuración espacial cambia de acuerdo a las interacciones entre los elementos del sistema de manera continua. Sostiene que el comportamiento a nivel individual genera un patrón espacial a nivel agregado, cuya suposición es válida para análisis de dinámica urbana.
Normativos	Evaluación Multicriterio	Utilizados habitualmente para determinar la capacidad de acogida del territorio para nuevos desarrollos urbanos. Permite combinar la aptitud o vocación del territorio para ese uso, así como el impacto que sobre éste pudiese ocasionar. Su implementación en un SIG es sencilla y flexible, permitiendo utilizar criterios de sostenibilidad dada la facilidad para combinar variables de tipo biofísico y socioeconómico.

Existen diversos trabajos en los cuales se analizan los resultados de varias simulaciones que sirven para la generación de escenarios [24], [25].

2.1.2 Escenarios

El modelado espacial comprende la construcción de modelos para producir resultados espaciales [26]. En muchos casos dichos modelos han sido desarrollados teóricamente en disciplinas como la geografía humana y luego implementados en un sistema de información geográfica (SIG). Un SIG provee una serie de funciones y operaciones de tratamiento de datos espaciales, las cuales han evolucionado notablemente desde la aparición del primer sistema de este tipo, sin embargo, muchas veces no ofrecen soluciones adecuadas para determinados problemas de modelado espacial. El modelado espacial se puede concebir como la unión entre métodos basados en modelos de simulación y las operaciones de los SIG para proveer herramientas adecuadas que sirven de apoyo a la planificación [24].

Dentro de los propósitos para los cuales se utiliza el modelado espacial esta la predicción y generación de escenarios, estos escenarios se pueden generar a partir de las capas temáticas que conforman una determinada área de estudio, como las que se observan en la Figura 6, se puede tomar una capa o generar un escenario a partir de una combinación de capas. Como resultado se pueden tener escenarios [23]:

- Los escenarios predictivos, los cuales se mueven en el campo del pensamiento futuro de lo probable, tienen como objetivo predecir lo que es probable que suceda en un plazo de tiempo generalmente corto.
- Los escenarios exploratorios, en este ámbito, muestran diversas evoluciones posibles de los crecimientos, generalmente en una escala temporal del medio plazo, en función del comportamiento futuro de un conjunto de factores.
- Los escenarios normativos por su parte, permiten discutir e imaginar las metas deseables para el crecimiento urbano, así como de los instrumentos y criterios necesarios para poder alcanzarlas.

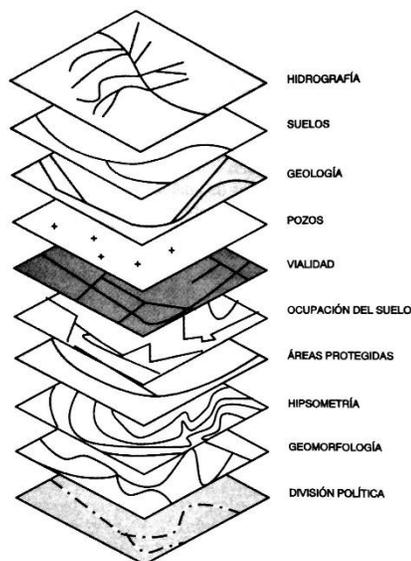


Figura 6. Capas temáticas [24]

2.1.3 Factores de Crecimiento

El planteamiento de cualquier modelo de simulación que se pretenda elaborar para estudiar los cambios que tienen lugar en un área metropolitana, los cuales son especialmente complejos requiere de un conocimiento previo del grado de influencia que diversos factores representan con respecto a ellos [27].

Para valorar el grado de influencia que presentan los factores seleccionados, se emplea un análisis de regresión espacial. Existen distintos tipos de análisis de regresión, como la regresión logística, que varían en función del número y la naturaleza de las variables involucradas, así como de la fórmula matemática capaz de describir la relación que las vincula [27], [28].

Una vez estimada la probabilidad de cambio en la ocupación urbana (variable a explicar) en función de los n factores involucrados (variables explicativas), a continuación se comparan las áreas con mayor probabilidad de cambio, con los cambios reales que han tenido lugar, empleando para ello el estadístico ROC (Relative Operating Characteristic). En función de la correspondencia que exista entre los cambios reales y la probabilidad de cambio estimada, se

obtiene un valor de ROC para cada variable, que valora el grado de correlación o de influencia de las distintas variables con la ocupación urbana [29].

2.1.4 Variables Territoriales

Es necesaria la representación del terreno, como una o varias capas temáticas. Podríamos definir a una variable territorial como una colección de elementos numéricos en n dimensiones, en donde cada elemento representa las condiciones codificadas del paisaje.

2.1.5 Variables Socioeconómicas

El territorio nacional mexicano está dividido en áreas con límites identificables en campo, denominadas áreas geoestadísticas, con tres niveles de desagregación: Estatal (AGEE), Municipal (AGEM) y Básica (AGEB), ésta puede ser urbana o rural. Estas áreas geoestadísticas cuentan con una clave única dentro del territorio nacional, esto permite relacionar los datos estadísticos a un lugar en específico [30].

Siendo un AGEB la unidad mínima de información poblacional, esta proporciona las variables socioeconómicas pertinentes para la presente investigación. Tal información se obtiene a través del portal de INEGI en distintos formatos: cvs, xls, etc. Es necesario realizar una adaptación a estos datos para espacializarlos.

2.2 AUTÓMATAS CELULARES

Los autómatas celulares (AC) son una clase de modelos matemáticos espaciales y discretos caracterizados por interactuar local e inherentemente, está representado por medio de una malla de celdas (Figura 7) [31].

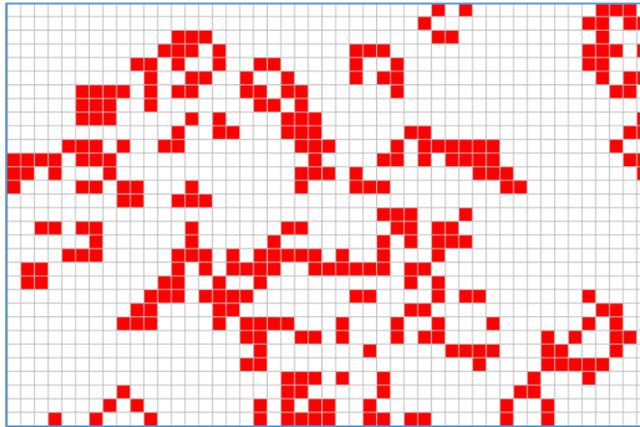


Figura 7. Espacio de evoluciones de un AC

Un AC se define por un conjunto de tripletas:

$$(I, S, R)$$

donde:

I representa el conjunto de entradas, S representa el conjunto de estados y R es la función que produce el estado a partir de una regla de evolución. Las entradas están definidas por n -tuplas de estados para un conjunto finito de celdas vecinas.

2.2.1 Características de un AC

Una de las características importantes de los AC es que se utilizan habitualmente para modelar sistemas físicos, químicos, biológicos o sociales, formados por muchos componentes que interactúan de forma local.

Se pueden definir como un sistema dinámico formado por un conjunto de elementos sencillos idénticos entre sí, pero que en conjunto son capaces de demostrar comportamientos complejos, este sistema dinámico está compuesto de los siguientes elementos [19], [32]:

1. El espacio en el que existe el autómata. En caso de que se trabaje en dos dimensiones, que será en la mayor parte de las ocasiones en el caso de la modelización de cambios urbanos, y al integrar los AC en un SIG; este espacio estará constituido por un plano bidimensional dividido en un número de subespacios homogéneos, conocidos como celdas. Cada celda contiene un autómata, que se encontrará rodeado de otros autómatas.
2. Un conjunto finito de estados en los que cada autómata (o lo que es lo mismo, cada celda) se puede encontrar. En el caso de su aplicación a fenómenos urbanos el conjunto finito de estados podía ser el conjunto de categorías de ocupación urbana que se distinguen, como por ejemplo residencial, industrial, equipamientos, viario, etc. En el caso de su aplicación a la modelización de incendios forestales, se podrían manejar por ejemplo tres categorías: quemado, ardiendo, y sin quemar.
3. Una definición de vecindad, es decir el conjunto de celdas que se establecen como vecindad de cada autómata. Ésta vecindad puede venir definida de dos formas estándar, la vecindad de Von Neumann y la vecindad de Moore. La vecindad de Moore comprende las 8 celdas adyacentes a la celda central más ella misma, mientras que la de Von Neumann incluye las 4 adyacentes más la central. A su vez, estas vecindades estándar pueden ser extendidas, como se observa en la Figura 8.
4. Un conjunto de reglas de transición que determinan los cambios de estado de cada celda en función de su estado inicial y de su vecindad. Las reglas de transición definen el estado del autómata (S_{t+1}) en el tiempo $t + 1$ como función del estado S_t en el tiempo t y una entrada I_t igualmente en el tiempo t . El estado resultante dependerá del estado inicial S_t así como del input I_t , que se derivará de la vecindad del autómata en el tiempo t .

$$T: (S_t, I_t) \rightarrow S_{t+1}$$

Regla de Transición

5. Finalmente es necesaria la existencia de una secuencia discreta de tiempos, en el que se aplican las reglas de evolución para cada celda, y su estado cambia o se mantiene, de acuerdo con la reglas de transición.

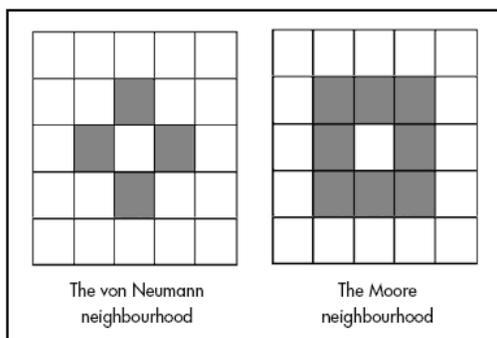


Figura 8. Vecindad de Neumann y Moore

En cualquier caso y de forma sintética, se puede afirmar que el éxito de los modelos de simulación urbana basados en AC se debe a que son capaces de simular dinámicas espaciales reproduciendo los patrones complejos mostrados en el crecimiento de una ciudad [27].

2.2.2 AC para Crecimiento Urbano

Por su capacidad para representar paisajes, en donde cada pixel del mapa puede ser representado por una celda en una malla bidimensional, los autómatas celulares son una de las técnicas más exploradas en el modelado territorial. Permiten simular el crecimiento urbano a partir del efecto de atracción/repulsión ejercido por los píxeles en su vecindad. Son capaces de generar estructuras complejas y representar dinámicas urbanas [9].

Uno de los modelos de simulación, basados en AC más usados es el propuesto por R. White, consta de un autómata celular delimitado. La idea general del modelo (Figura 9 y Figura 10), es el de evaluar los pesos y la distancia de los vecinos de cada celda del mapa, de acuerdo al uso de suelo que representan, como resultado de la evaluación, se tendrán los potenciales de transición de la celda para cada uso de suelo.

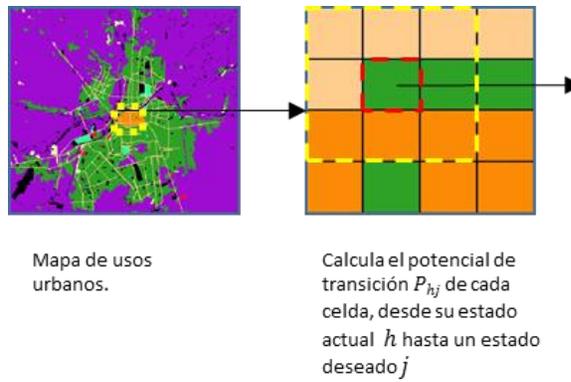


Figura 9: Ventana de observación de 3x3 pixeles.

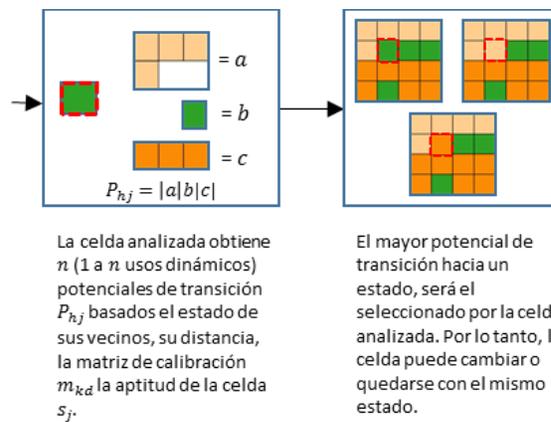


Figura 10. Se determina el mayor potencial

Se ha completado una época, cuando se calcula la última celda del mapa. Una simulación puede requerir de 1 o más épocas, todo en función de los periodos a simular.

2.3 SISTEMAS MULTIAGENTE

Un sistema multiagente permite simular las acciones individuales de un diverso grupo de agentes, donde al final se obtiene un resultado del comportamiento del sistema [33]. Un agente es cualquier entidad capaz de percibir su medio ambiente con la ayuda de sensores e interactuar con el entorno mediante el uso de actuadores. El término percepción dentro de este contexto indica que el agente puede recibir entradas (información) en cualquier instante, la secuencia de estas percepciones refleja el historial de lo que el agente ha recibido y es debido a ello que sus decisiones serán siempre en función de la secuencia completa de percepciones hasta ese momento [5].

El modelo general de un agente (Figura 11), muestra la interacción con el medio ambiente a través de sensores y actuadores, en donde la información recabada es percibida como la situación actual del sistema con quien interactúa, y con base a ello, analiza una posible respuesta.

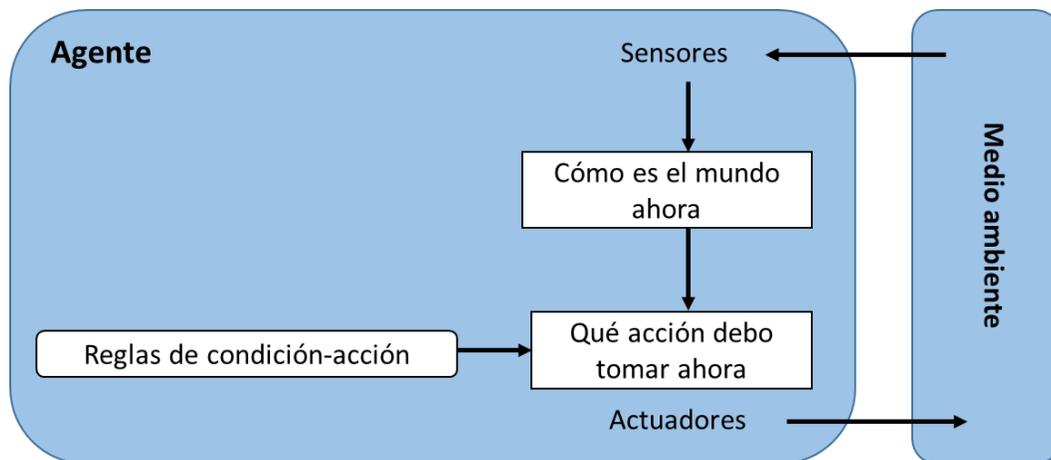


Figura 11. Modelo general de agente [5]

Una de las aplicaciones de los agentes es la simulación multiagente, donde son usados para desarrollar aplicaciones complejas compuestas de una multitud de subsistemas que interactúan entre sí obligando a distribuir la capacidad de decisión entre diversos agentes.

Un sistema multiagente permite la simulación de un sistema complejo coordinando los distintos subsistemas que lo componen e integrando los objetivos particulares de cada subsistema en un objetivo común [6].

Existen dos enfoques para construir sistemas multiagente [34]:

- Enfoque constructivista, aquel que ve un conjunto de agentes como un solo sistema, y es a este sistema al que le brinda de los mecanismos necesarios para que genere un comportamiento inteligente.
- Enfoque formal (clásico), consiste en proveer a cada agente de la mayor inteligencia posible utilizando descripciones formales del problema.

En ambos casos la inteligencia es definida utilizando un sistema formal para la descripción, raciocinio, inferencia de nuevo conocimiento y planificación de acciones a realizar en su entorno.

2.3.1 Agente Cognitivo

Hasta este momento se ha mencionado al agente como una entidad que puede emitir una decisión basada en un historial de percepciones y responder con base a dicho historial. Un agente que aprende se puede dividir en cuatro componentes conceptuales mostrados en la Figura 12, donde la distinción más importante entre el elemento de aprendizaje y el de actuación es que el primero está responsabilizado de hacer mejoras y el segundo de la selección de acciones externas.

El elemento aprendizaje se realimenta con las críticas (decisiones a mejorar) sobre la actuación del agente y determina cómo se debe de modificar el elemento actuación para

proporcionar mejores resultados en el futuro. El generador de problemas es el responsable de sugerir acciones que lo guiarán hacia experiencias nuevas e informativas. Es entre estos cuatro componentes, donde todos los agentes pueden mejorar su eficacia con la ayuda de mecanismos de aprendizaje [5].

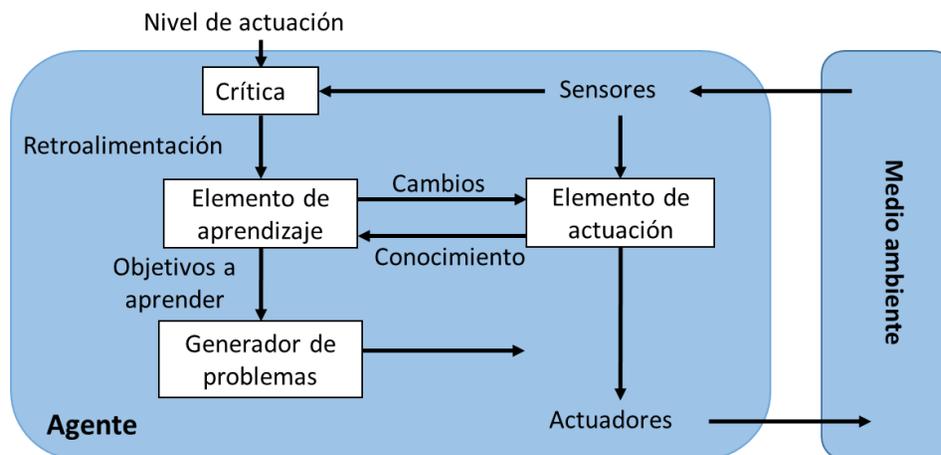


Figura 12. Modelo general de agentes que aprenden [5]

2.3.2 Agente para Simular Cuestiones Sociales

La programación de agentes está consolidada fuera del campo de la simulación social, los programas informáticos de agentes han estado muy influenciados por el trabajo en la IA, especialmente en un subcampo llamado Inteligencia Artificial Distribuida (IAD).

La IAD está dedicada a las propiedades y el diseño de redes de agentes que interactúan, por ejemplo, cómo se podría diseñar un grupo de agentes, cada uno con habilidad diferente, de forma que cooperen para solucionar un problema. Gran parte de este campo resulta relevante para la simulación social ya que un modelo que involucre a numerosos agentes (denominados modelos multiagente) puede servir para la simulación de sociedades.

Agentes y Agencialidad

El concepto de agencialidad en un individuo se usa para comunicar la naturaleza intencional de la actividad humana, está relacionado con conceptos como la intencionalidad, libre albedrío y el poder para alcanzar objetivos. En un agente este concepto de agencialidad tiene un alcance mucho más reducido, y se traduce en las siguientes propiedades:

- **Autonomía:** Un agente opera sin que otros tengan control directo de sus acciones y su estado interno.
- **Habilidad Social:** Un agente interactúa con otros mediante un determinado lenguaje.
- **Reactividad:** Los agentes perciben su entorno (puede ser el mundo físico, virtual, o uno producto de una simulación) y pueden responder al mismo.
- **Proactividad:** De la misma manera en que pueden responder a su entorno, los agentes también son capaces de tomar la iniciativa para lograr un objetivo propio.

2.3.3 Aspectos a Modelar

Para nuestro estudio, un agente puede estar construido para simular algunos aspectos (muy simplificados) de las intenciones humanas. Algunos de los atributos que se pueden modelar con agentes son:

- **Conocimiento y creencia:** El agente basa sus acciones en lo que saben de su entorno.
- **Inferencia:** Dada una base de conocimientos (creencias), el agente puede ser capaz de inferir más información de ellas.
- **Modelos Sociales:** Un agente puede aprender de las interrelaciones entre otros agentes.
- **Representación del conocimiento:** Un agente necesita representar de alguna forma el conocimiento que tiene, existen técnicas estudiadas en IA bajo la denominación de “representación del conocimiento”, esto mediante lógica de predicados o redes semánticas.
- **Objetivos:** El agente es autónomo e intencionado, cada acción obedece a la necesidad de satisfacer algún objetivo interno.

- **Planificación:** Un agente debe tener forma de determinar el comportamiento con mayor probabilidad de ocasionar el alcance de un objetivo.
- **Lenguaje:** Toda interacción entre agentes necesita ser modelizada especificando un “lenguaje”.
- **Emociones:** Existe poca investigación como la felicidad, tristeza, pena o cólera dentro de la IA, bajo este aspecto quedan aún preguntas básicas por resolver.

Aún con estas características de los agentes autónomos, es poco realista esperar que los modelos multiagente puedan simular la gran mayoría de los fenómenos humanos psicológicos y sociales a cualquier nivel de detalle, sin embargo, al igual que con otras formas de simulación planteadas [35], se debe intentar extraer aquellas características del objetivo que tienen mayor trascendencia teórica y concentrarse en modelarlas, haciendo caso omiso a las otras con menor trascendencia o que no son centrales para la materia en proceso de investigación.

Un modelo que logra dotar al agente de capacidades cognitivas, el cual potencia su aprendizaje y toma de decisiones , se encuentra aplicado en el trabajo “A Neural Network Multiagent Architecture Applied to Industrial Networks for Dynamic Allocation of Control Strategies Using Standard Function Blocks” [10] en el cual se presenta una estructura y la implementación de una red neuronal en un agente inteligente, con el objetivo de formar un sistema multiagente que sea capaz de detectar problemas en una línea de producción reduciendo al mínimo la intervención humana, esto lo logra usando RNA, las cuales permiten que los agentes aprendan sobre los patrones que pueden ocasionar un problema y que este aprendizaje les permita manejarlo de manera correcta.

2.3.4 Metodologías para la construcción de Agentes

Existen diversos factores que propician el uso de MBA como una solución apropiada a problemas que generalmente presentan métodos centralizados[36]. Uno de los factores es el de presentar un ambiente dinámico y complejo, para el caso de los AC, la malla bidimensional

usada para la representación del terreno, provee un ambiente de tales características para el cálculo de las interacciones de cada celda con su vecindad.

El diseño y análisis orientado a agentes, ha propiciado el desarrollo de distintas metodologías que sirven para formalizar la construcción de los MBA, a continuación se presenta (Tabla 2) un conjunto de ellas y una breve descripción:

Tabla 2. Metodologías para el desarrollo de sistemas multiagente

Metodología	Descripción	Comentarios
Vowel Engineering	Vowel o Vocal: A (por agentes), E (por entorno), I (por interacciones) y O (por organización). Consiste en tener en cuenta los elementos señalados por las vocales en un cierto orden. El orden se decide en función del tipo de sistema que queramos tener.	El proceso de desarrollo es el punto debil de esta metodología. Vowel Engineering solo proporciona las vocales como resumen de elementos a considerar en el desarrollo y un conjunto de tecnologías aplicadas.
MAS-CommonKADS	Plantea un desarrollo de SMA integrado con un ciclo de vida de software, concretamente el espiral dirigido por riesgos [Pressman 82]. Propone siete modelos para la definición del sistema: agente, tareas, experiencia, coordinación, comunicación, organización y diseño.	El proceso es costoso, las especificaciones son hechas en lenguaje natural.
BDI	Las arquitecturas BDI se inspiran en un modelo cognitivo del ser humano. Los agentes utilizan un modelo del mundo, una representación de cómo se les muestra el entorno. El agente recibe estímulos a través de sensores. Estos estímulos modifican el modelo del mundo que tiene el agente (representado por un conjunto de creencias). Para guiar sus acciones, el agente tiene Deseos. Un deseo es un estado que el agente quiere alcanzar a través de intenciones.	No tiene herramientas de soporte. Sin embargo, los modelos formales que se pueden encontrar tras esta metodología constituyen una referencia obligada para aquellos interesados en la integración de teorías de agentes y las prácticas de ingeniería.
MaSE	Parte del paradigma orientado a Objetos, se asume que un agente es sólo un objeto especializado.	Se dificulta expresar elementos característicos de la tecnología de agentes como lo es, el razonamiento de los agentes.

ZEUS	Propone un desarrollo en cuatro etapas [Collis et al. 99]: el análisis del dominio, el diseño de los agentes, la realización de los agentes y el soporte en tiempo de ejecución.	Conceptualmente, ZEUS es superior a MaSE, la primera está más orientada a la aplicación de tecnología de agentes (planificación, definición de ontologías, secuenciación de tareas), la segunda se orienta más a las prácticas de ingeniería convencional.
GAIA	Propone trabajar inicialmente con un análisis a alto nivel. En este análisis se usan dos modelos, el modelo de roles para identificar los roles clave en el sistema junto con sus propiedades definitivas y el modelo de interacciones que define las interacciones mediante una referencia a un modelo institucionalizado de intercambio de mensajes, como el FIPA-Request [FIPA 03].	Se queda a un nivel de abstracción demasiado alto. Según los autores, con ello se consigue desacoplar GAIA de las distintas soluciones de implementación de agentes. Dado el nivel de abstracción en que se queda es de esperar que el esfuerzo a invertir para pasar de una especificación GAIA hasta su implementación sea alto.
INGENIAS - MESSAGE	Define un conjunto de meta-modelos con los que hay que describir el sistema. Los meta-modelos indican qué hace falta para describir: agentes aislados, organizaciones de agentes, el entorno, interacciones entre agentes o roles, tareas y objetivos. Estos meta-modelos se construyen mediante un lenguaje de meta-modelado, el GOPRR (Graph, Object, Property, Relationship, and Role)	El proceso de desarrollo que propone INGENIAS es excesivo cuando se trata de desarrollos reducidos. A diferencia de MAS-CommonKADS que dota de procesos de desarrollo adaptados al tamaño del problema, INGENIAS da la impresión de dedicarse exclusivamente a desarrollos de gran tamaño.

2.4 COSTO COMPUTACIONAL

Si se toma en cuenta que los cálculos para los modelos anteriores, se deben hacer para cada pixel del mapa, nos encontramos con un problema de complejidad computacional muy alta $O(n^3)$, es decir, mientras mayor sea el tamaño de los mapas de entrada, el tiempo de ejecución de la simulación se incrementaría exponencialmente.

Existen bibliotecas computacionales que facilitan la implementación de estrategias de paralelización, esto con el objetivo de proveer funciones programables en distintos lenguajes, que ayuden a disminuir los tiempos de procesamiento de información.

2.4.1 OpenMP

OpenMP es una interfaz de programación de aplicaciones (API por sus siglas en inglés), principalmente diseñada para la ejecución de procesos en memoria compartida, es ampliamente usada en programación paralela [37].

En el ámbito de la simulación de sistemas complejos, es necesaria la optimización de los recursos de cómputo debido a la gran cantidad de cálculos a realizar. La simulación de crecimiento urbano, es un ejemplo en donde se requiere establecer estrategias de paralelización de procesos computacionales. Una de estas estrategias es la de fork-join [38].

2.4.2 El modelo fork-Join

OpenMP está basado en el modelo fork-join, consiste en dividir (fork) un proceso en una región paralela, en la cual se llevan a cabo tareas en cada procesador o núcleo disponible, al término de cada tarea, se vuelven a unir (join) y siguen el proceso principal, el esquema se muestra en la Figura 13.

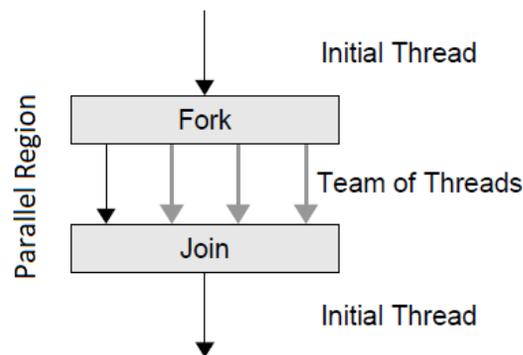


Figura 13. Proceso general de fork-join

Además, en caso de requerirse, OpenMP tiene la capacidad de crear una región paralela dentro de otra (anidada), por lo tanto, es posible dividir una tarea tanto como sea necesario y tanto como el hardware lo permita, como se ilustra en la Figura 14.

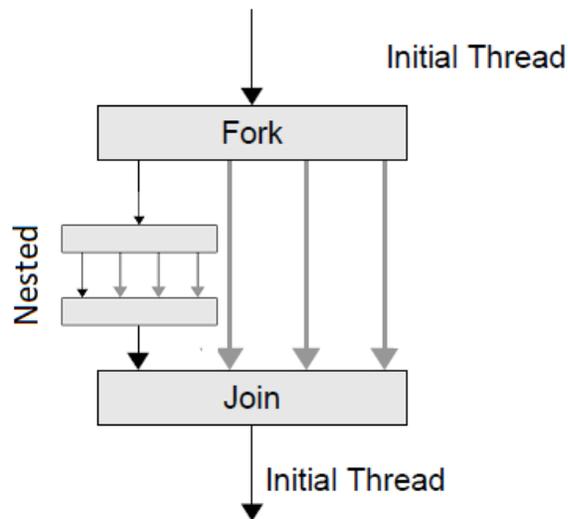


Figura 14. Región paralela anidada (nested)

La manipulación de los archivos para la implementación de la estrategia, fue llevada a cabo con la biblioteca GDAL (Geospatial Data Abstraction Library) , es una biblioteca de uso libre para la lectura y escritura de datos geospaciales, es uno de los elementos fundamentales en el programa, provee de funciones a bajo nivel que permiten la manipulación de archivos ráster.

CAPÍTULO 3: REDES NEURONALES

A lo largo de la historia el principal objetivo de los científicos ha sido desarrollar máquinas que denoten inteligencia al momento de realizar sus procesos, estos intentos han favorecido la aparición de líneas fundamentales para la obtención de máquinas inteligentes. En un inicio su desarrollo fue el resultado de las habilidades mecánicas de sus constructores, hoy en día con los avances informáticos la habilidad mecánica ha pasado a convertirse en habilidad informática, esto permite que existan diversas maneras de realizar procesos similares a los inteligentes y que se pueden encuadrar dentro de la denominada Inteligencia Artificial [8].

Inteligencia Artificial (IA) es una acepción acuñada en 1956, y su desarrollo se ha caracterizado por una sucesión de periodos alternativos de éxito y abandono de la misma. La IA siempre ha tenido como modelo natural las funcionalidades inteligentes del ser humano, su motivación ha sido construir máquinas que piensen como un ser humano, o al menos emularlo en alguna capacidad de tal modo que denotara cierta inteligencia [39].

Una de las ramas más prometedoras dentro de lo que se entiende por Inteligencia Artificial son las denominadas redes de autoproceso, estas redes son aquellas construidas por nodos en los que hay elementos procesadores de información de cuyas interacciones locales depende el comportamiento del sistema, bajo este tipo de redes se encuentran las llamadas “Redes Neuronales” [8].

Es llamada red neuronal debido a que contiene elementos interconectados. Estos elementos fueron creados después de estudios al sistema nervioso biológico, de otra manera, las redes neuronales son un intento de crear máquinas que funcionen de forma similar al cerebro humano, construyéndolas con componentes que se comporten como una neurona biológica [40].

3.1 EL MODELO BIOLÓGICO

El principal propósito de todos los sistemas neuronales es el control centralizado de varias funciones biológicas, algunas de ellas responsables del abastecimiento de energía, pues el sistema neuronal está conectado con el metabolismo, el control cardiovascular y la respiración [8]. Los sistemas neuronales están formados por neuronas, éstas son células vivas y, como tales, contienen los mismos elementos que forman parte de todas las células biológicas con una diferencia fundamental, su capacidad de comunicación.

La neurona es la unidad básica del cerebro humano, éste contiene alrededor de diez mil millones (10^{10}), en donde cada una de ellas está conectada a otras diez mil (10^4). El funcionamiento del cerebro es complicado y aún no ha sido comprendido del todo, sin embargo los aspectos básicos de cada neurona son relativamente claros [41].

3.1.1 Aspectos Básicos de una Neurona Biológica

Se puede apreciar la anatomía de una neurona biológica típica de un organismo vertebrado (Figura 15). Ésta célula tiene 4 principales áreas morfológicas definidas, el cuerpo o soma, las dendritas, el axón y las terminales sinápticas, cada una de las cuales realiza una contribución específica al procesamiento de señales [42].

El cuerpo o soma es el centro metabólico de la célula, contiene el núcleo el cual es el responsable de dos tipos de ramificaciones, el axón y las dendritas.

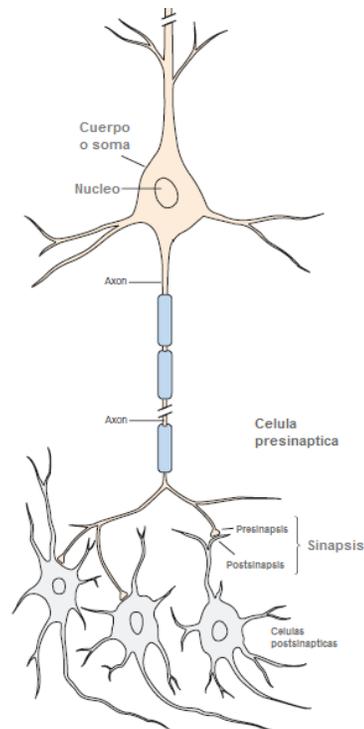


Figura 15. Estructura de una Neurona Biológica [42]

En términos generales, las dendritas reciben señales de entrada; el cuerpo celular las procesa y emite señales de salida; el axón transporta estas señales a las terminales presinápticas las cuales se encargan de distribuir la información (sinapsis) a un nuevo conjunto de neuronas. Una neurona recibe información de miles de otras neuronas y a su vez envía información a miles de neuronas más. Como se mencionó anteriormente se calcula que cada neurona está conectada a otras diez mil (10^4) teniendo un orden aproximado de conexiones totales en el cerebro de 10^{15} [8].

3.1.2 Funcionamiento de una Neurona Biológica

Las señales usadas por las neuronas son de dos tipos: eléctricas y químicas. La señal generada y transportada a lo largo del axón es un impulso eléctrico, mientras que la señal que se transmite entre los terminales axónicos de una neurona y las dendritas de la otra es de origen químico [39]; concretamente, esta “comunicación” entre neuronas se realiza mediante moléculas de sustancias transmisoras (neurotransmisores) que fluyen a través de unos contactos especiales, llamados sinapsis, que tienen la función de receptor y están localizados entre los terminales

axónicos y las dendritas de la neurona siguiente (espacio sináptico, entre 50 y 200 Angstroms)[8].

Potencial de Acción de la Neurona

La generación de señales eléctricas está relacionada con la composición de la membrana celular, la carga eléctrica del interior de la neurona es diferente a la de exterior debido a la distribución desigual de iones, creando una diferencia de potencial [43]. La diferencia se da en relación con la concentración de los iones sodio y potasio. El medio externo es 10 veces más rico en sodio que el interno, mientras que el interno es 10 veces más rico en potasio que el externo, esta diferencia de iones a cada lado de la membrana produce una diferencia de potencial de aproximadamente 70 milivoltios negativa en el interior de la célula. Es lo que se llama potencial de reposo de la célula nerviosa [8].

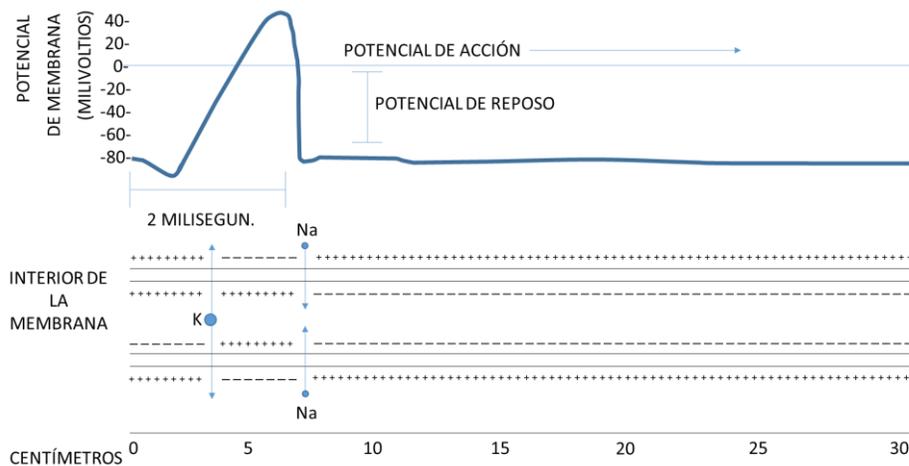


Figura 16. Concentración de iones de sodio (Na+) y potasio (K-) en la membrana del axón [8]

La llegada de señales procedentes de otras neuronas a través de las dendritas (recepción de neurotransmisores) actúa acumulativamente, bajando ligeramente el valor del potencial de reposo. Dicho potencial modifica la permeabilidad de la membrana, de manera que cuando llega

a cierto valor crítico comienza una entrada masiva de iones de sodio que invierten la polaridad de la membrana (Figura 16), esta inversión de voltaje es conocida como potencial de acción, se propaga (Figura 17) a lo largo del axón y, a su vez provoca la emisión de los neurotransmisores en las sinapsis de los terminales axónicos.

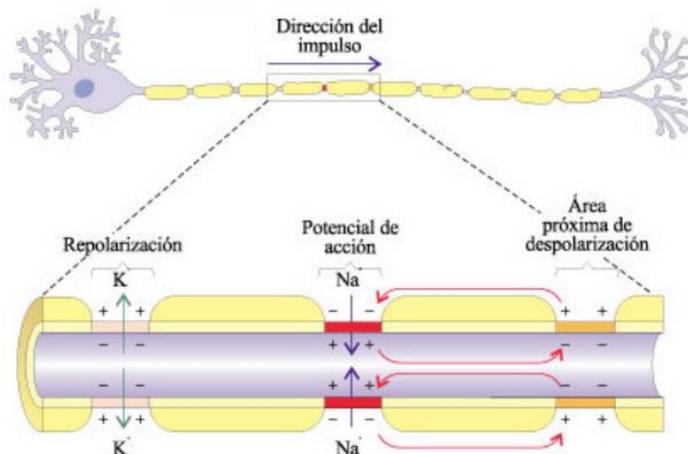


Figura 17. Dirección de la propagación del impulso nervioso a través del Axón [43]

Existen dos tipos de sinapsis según el efecto postsináptico; 1) Excitatorio, produce una despolarización de la membrana en la neurona postsináptica, si esta despolarización es suficientemente intensa, se producirá un potencial de acción. 2) Inhibitoria, produce una estabilización del potencial de la membrana, como resultado la neurona postsináptica estará inhibida, le será difícil alcanzar un potencial de acción [44].

3.1.3 Analogía con las Redes Neuronales Artificiales

La neurona artificial toma los aspectos más importantes del funcionamiento de las neuronas biológicas, y esto sirve como base para el modelado matemático de la neurona artificial. Por ejemplo, una neurona biológica tiene entrada de información a través de sus terminales (dendritas), el conjunto de entradas determinaran si la neurona se activa o permanece en reposo, si se activa se liberarán neurotransmisores a través de las terminales presinápticas, estos neurotransmisores serán la entrada a la siguiente neurona y el proceso se repetirá según el número de conexiones de la neurona. Para el caso de la neurona matemática McCulloch-Pitts

(Figura 18), tiene en x_0 a x_m el valor de las entradas, en w_{k0} a w_{km} el peso que tendrá cada una de las entradas, y en Y_k la suma de las entradas por el peso de cada una de ellas, lo anterior expresado por la ecuación:

$$y_k = \varphi \left[\sum_{j=0}^m w_{kj} x_j \right] \quad (1)$$

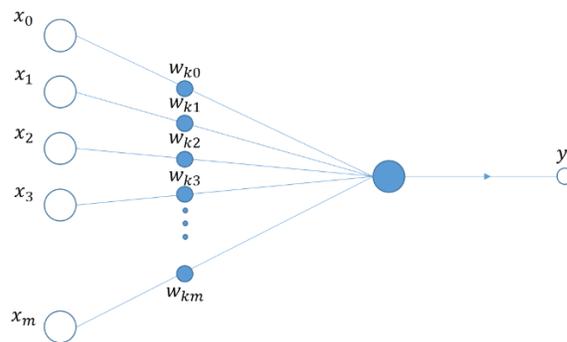


Figura 18. Neurona McCulloch – Pitts

Las señales que llegan a la sinapsis son las entradas a la neurona; éstas son ponderadas a través de un parámetro, denominado peso. Estas señales de entrada pueden excitar a la neurona (sinapsis con peso positivo) o inhibirla (peso negativo). El efecto es la suma de las entradas ponderadas (Y_k). Si la suma es igual o mayor que el umbral de la neurona, entonces la neurona se activa (da salida). La habilidad de ajustar las señales mediante el incremento o decremento del peso de cada entrada es un mecanismo de aprendizaje de una neurona artificial.

3.2 EL MODELO ARTIFICIAL

Desde Platón (427-347 a.C) hasta Alan Turing en 1936, se han formulado explicaciones teóricas sobre el cerebro humano y el pensamiento, siendo este último el primero en estudiar el cerebro como una forma de ver el mundo de la computación. Sin embargo, fueron Warren McCulloch

y Walter Pitts quienes concibieron los primeros fundamentos de la computación neuronal y, en 1943 lanzaron una teoría acerca de la forma de trabajar de las neuronas [45].

Debido a su constitución y a sus fundamentos, las Redes Neuronales Artificiales (RNA) presentan un gran número de características semejantes a las del cerebro, son capaces de aprender de la experiencia, de generalizar de casos anteriores a nuevos casos y abstraer características esenciales a partir de entradas que representan información irrelevante[8], entre otras.

Se han desarrollado aplicaciones usando RNA para fines tan variados como visión artificial, procesamiento de señales e imágenes, reconocimiento del habla y de caracteres, sistemas expertos, análisis de imágenes médicas, control remoto, control de robots, inspección industrial y exploración científica [46]. Se podrían clasificar todas estas aplicaciones en varios dominios o tipos de aplicaciones, indicando, además, el tipo de redes más utilizadas para cada tarea [47]:

- Asociación y clasificación: heteroasociadores y redes competitivas
- Regeneración / reconstrucción de patrones: redes de satisfacción de demanda
- Regresión y generalización: heteroasociadores
- Optimización: heteroasociadores

La habilidad de las RNA para reconocer patrones y aprender en base a ejemplos hace de ésta una metodología interesante para examinar y clasificar distintos tipos de datos [48].

3.2.1 Aspectos Básicos de una Neurona Artificial

Una neurona artificial es una función matemática concebida como un modelo crudo o una abstracción de una neurona biológica[49], como resultado, se obtiene un procesador elemental o neurona, el cual a partir de un vector de entrada procedente del exterior o de otras neuronas, proporciona una respuesta o salida[50]. Los elementos que constituyen a una neurona *i*-ésima se muestra en la Figura 19.

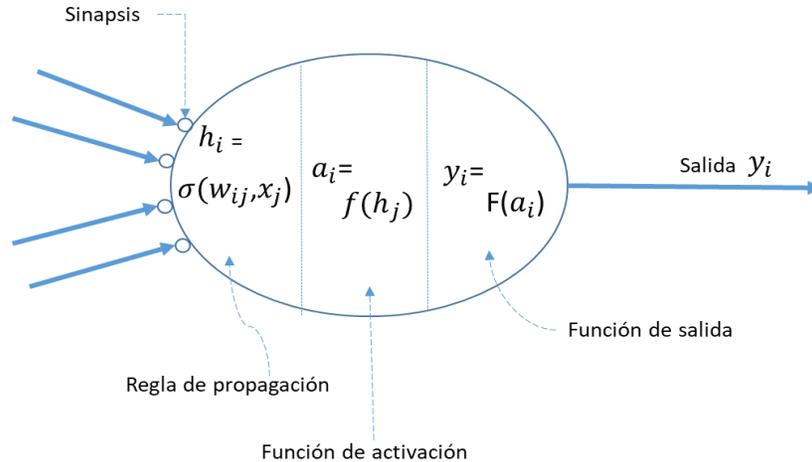


Figura 19. Modelo genérico de neurona artificial [50]

Conjunto de Entradas

Normalmente, una neurona tiene más de una entrada[51], ver Figura 20, la segunda entrada corresponde a la neurona presináptica (j), la salida de esa neurona (y_j) es a su vez la entrada (x_j) de la siguiente neurona (i), la salida de una neurona presináptica se convierte en la entrada de la siguiente neurona.

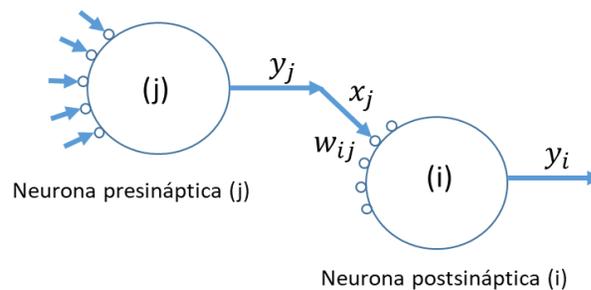


Figura 20. Interacción entre una neurona presináptica y otra postsináptica [50]

Pesos Sinápticos

Cada entrada ($x_a, x_b, x_c \dots x_j \dots x_z$) a la neurona postsináptica i , tiene un peso asociado (w)[51], para el caso de la entrada proveniente de la neurona presináptica j a la neurona postsináptica i (Figura 20), el peso correspondiente se indica como (w_{ij}).

Estado de Activación

Cualquier neurona del sistema necesita conservar sus estados en un tiempo t . Se especifica por un vector de N números $A(t)$, el cual representa el estado de activación del conjunto de unidades de procesamiento. La activación de una neurona U_i (U por unidad), en el tiempo t se designa por $a_i(t)$, el vector que contiene todas las unidades del sistema:

$$A(t) = (a_1(t), a_2(t), \dots, a_i(t), \dots, a_N(t)) \quad (2)$$

Función de Salida o Transferencia

Entre las unidades o neuronas que constituyen una red neuronal existe un conjunto de conexiones que unen unas a otras. Cada neurona transmite señales a las siguientes. Con cada unidad U_i se tiene una función de salida $f_i(a_i(t))$, que transforma el estado actual de activación $a_i(t)$ en una señal de salida $y_i(t)$, es decir:

$$y_i(t) = f_i(a_i(t)) \quad (3)$$

El vector que mantiene las salidas en un instante t de todas las neuronas es:

$$Y(t) = (f_1(a_1(t)), f_2(a_2(t)), \dots, f_i(a_i(t)), \dots, f_N(a_N(t))) \quad (4)$$

En ciertos modelos la salida de la neurona será igual a su activación, en tal caso la función f_i es la función identidad $f_i(a_i(t)) = a_i(t)$. En la mayoría de las implementaciones, f_i es de tipo sigmoïdal. Las funciones de transferencia típicas son: la función escalón, la función lineal y mixta, y la función sigmoïdal.

Función Escalón

La función escalón se usa por neuronas binarias, en éstas cuando la suma de las entradas es mayor o igual que el umbral de la neurona, la salida será 1; en caso contrario la salida será 0 o -1. Está definida por [52]:

$$f(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases} \quad f(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ -1 & \text{si } x < 0 \end{cases} \quad (5)$$

La Figura 21 representa gráficamente la función escalón. En ambos casos se toma como umbral cero; en caso que no lo fuera, el escalón quedaría desplazado.

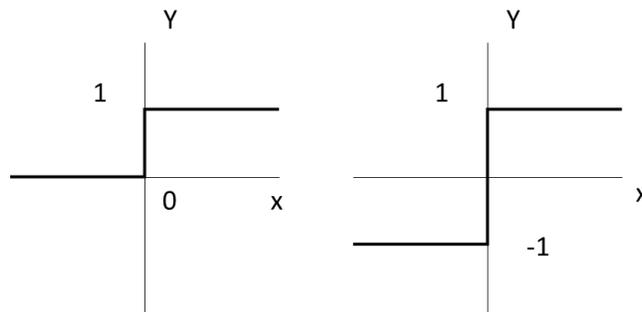


Figura 21. Función de transferencia escalón

Función Lineal y Mixta

La función lineal tiene la expresión $f(x) = x$. En las neuronas con función de transferencia mixta, se tiene que, si la suma de las señales de entrada es menor que un determinado límite inferior, la activación se define como 0 (o -1), por el contrario si la suma es mayor o igual que el límite superior, entonces la activación será 1. Si la suma no toca los límites superior o inferior, es decir, entre ambos límites, entonces la activación se definirá por la función línea. Está definida por:

$$f(x) = \begin{cases} 1 & \text{si } x < -c \\ 0 & \text{si } x > c \end{cases} \quad f(x) = \begin{cases} -1 & \text{si } x < -c \\ 1 & \text{si } x > c \\ a.x & \text{en otro caso} \end{cases} \quad (6)$$

Se representa como indica la Figura 22, se toma el límite superior de la suma de todas las entradas que afectan a la neurona durante el ciclo de operación (x) como c y el límite inferior como $-c$, y es la salida de activación de la neurona.

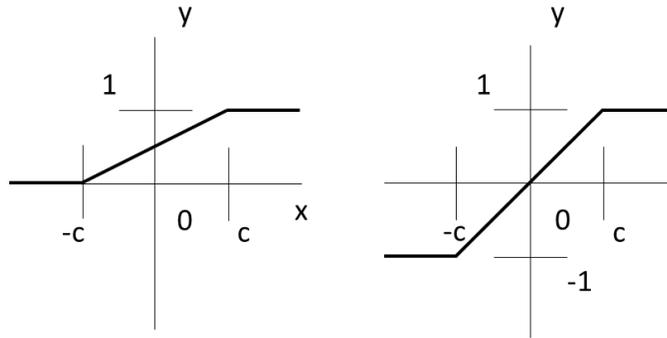


Figura 22. Funciones de activación mixta

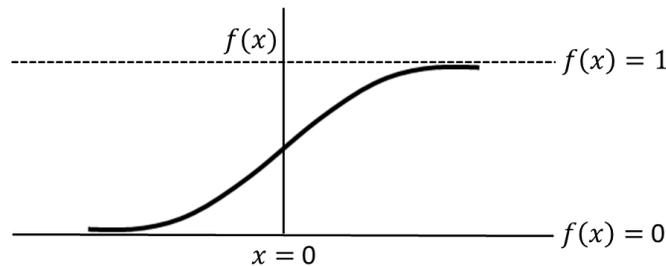
Función Sigmoidal

La función sigmoidal, es una función continua, también llamada curva sigmoidal o función logística[53]:

$$y = \frac{1}{1 + e^{-x}} \quad (7)$$

$$\text{y su derivada: } \frac{dy}{dx} = [1 - y(x)]y(x) = \frac{e^{-x}}{(1+e^{-x})^2} = \frac{e^x}{(1+e^x)^2} \quad (8)$$

Es la función más común en las implementaciones de RNA, la Figura 23, muestra la curva de decisión[54].



$$f(x) = 1/(1 + e^{-x})$$

Figura 23. Función de activación continua

Conexión Entre Neuronas

Las conexiones entre neuronas, tienen asociado un peso, éste es el que hace que la red adquiera conocimiento. Se considera y_i como el valor de salida de una unidad o neurona i en un instante determinado. Una neurona recibe las señales de las neuronas presinápticas, estas señales la dan la información del estado de activación de esas neuronas. Cada sinapsis (conexión) entre la neurona i y la neurona presináptica j esta ponderada por un peso W_{ij} . Como interpretación a las entradas y sus respectivos pesos, se realiza una suma net_j la cual contiene el potencial postsináptico (total de entradas), es decir:

$$net_j = \sum_i^N W_{ij} \cdot y_i \quad (9)$$

Esta ecuación muestra el procedimiento a seguir para combinar los valores de entrada de una neurona con los pesos correspondientes a estas entradas.

Regla de Activación

A partir de una neurona i y su entrada total (net_i), se produce un nuevo estado de activación combinando el estado actual (a_i) y (net_i).

Dado el estado de activación $a_i(t)$ de la unidad U_i y la entrada total que llega a ella net_i , el siguiente estado de activación $a_i(t+1)$, se obtiene aplicando la siguiente función o regla de activación:

$$a_i(t+1) = F(a_i(t), net_i) \quad (10)$$

Por lo general, F es la *función identidad*, esto significa que el estado de activación de una neurona en $t + 1$ coincidirá con el net de la misma en t . Por lo tanto, el parámetro que se le provee a la función de salida f de la neurona será directamente el net . La salida de una neurona i (y_i) será expresada de la siguiente manera:

$$y_i(t + 1) = f(net_i) = f\left(\sum_{j=1}^N W_{ij}y_j(t)\right) \quad (11)$$

Regla de Aprendizaje

El aprendizaje de una neurona se da en la modificación de sus pesos sinápticos. Biológicamente, se acepta que la información memorizada en el cerebro está más vinculada con la sinapsis que con la neurona en sí, es decir, el conocimiento se encuentra en la sinapsis. Para el caso de las neuronas artificiales, se puede considerar que el conocimiento se representa por el peso de las conexiones entre las neuronas. Una neurona “aprende” modificando los pesos de sus entradas, por lo tanto, el proceso de aprendizaje implicará la modificación de los pesos en las conexiones[55].

Un aspecto importante sobre el aprendizaje de una neurona es cómo se modificarán los pesos, es decir, qué criterios se seguirán para la modificación durante el aprendizaje de una neurona. Estos criterios determinarán lo que se conoce como *regla de aprendizaje*.

Vectores

En la mayoría de los modelos de neuronas y redes neuronales, se hace uso de la representación vectorial para las magnitudes[56]. Si se considera una red formada por varias capas de neuronas se considera la salida de una determinada capa con n unidades como un vector n -dimensional n $Y = (y_1, y_2, \dots, y_n)$, este vector de salida será posiblemente el conjunto de entradas a la siguiente capa m -dimensional, cada una de las unidades de esta capa tendrán n pesos asociados

a las conexiones de la capa anterior. Se tendrán m vectores de pesos n dimensionales asociados con la capa m . El vector con los pesos para la j -ésima unidad:

$$Y_j = (y_{j1}, y_{j2}, \dots, y_{jn}) \quad (12)$$

Se utilizaran vectores de entrada, vectores de salida y vectores de pesos para los diferentes cálculos realizados en la operación de una red neuronal.

3.2.2 Estructura de una Red Neuronal Artificial

Analizados los aspectos básicos de una neurona artificial, sus componentes y características, se abordará su constitución estructural de la cual se desprende el funcionamiento y los diferentes propósitos de una red neuronal[57].

La disposición de neuronas en una red se lleva a cabo formando capas o niveles con un número determinado de neuronas en cada una (Figura 24). A partir de la ubicación de la capa dentro de la red, se pueden clasificar en una de tres tipos:

- **Capa de entrada:** Es la que recibe la información proveniente de otras neuronas o fuentes externas a la red.
- **Capa oculta:** No tienen contacto con el exterior, el número de capas ocultas puede variar desde cero hasta el que sea necesario, generalmente es en este nivel en donde se lleva a cabo la mayor parte del procesamiento.
- **Capa de salida:** Es la encargada de transferir la información resultante hacia una fuente externa a la red.

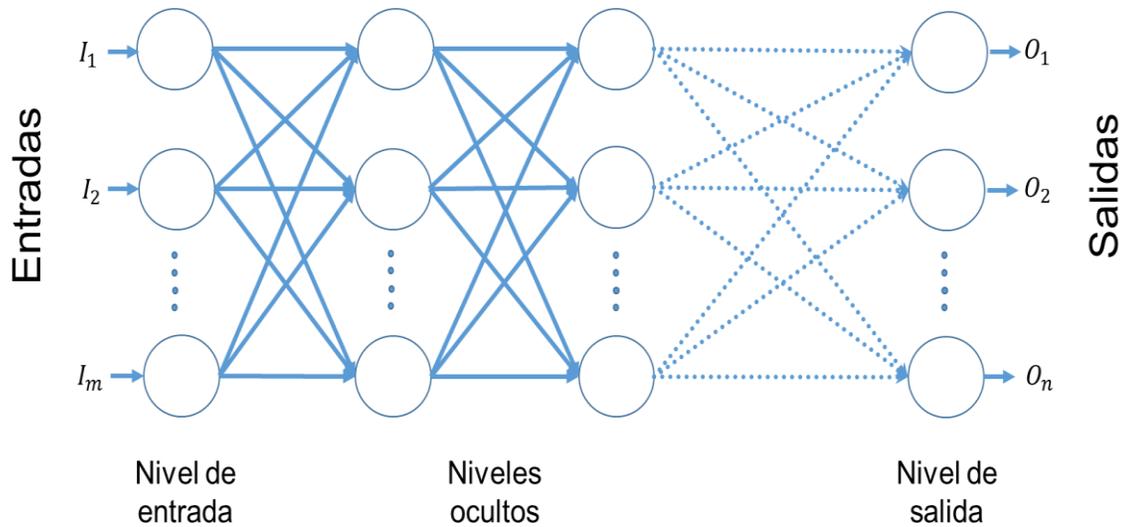


Figura 24. Estructura multinivel de Red Neuronal

Redes Monocapa

Es la red más simple, la cual consiste en una o más neuronas idénticas en paralelo[58], véase la Figura 25, en donde la capa en gris representa solo la entrada de datos y en la cual no se lleva ningún cálculo. En las redes monocapa como en la red de Hopfield, se establecen conexiones laterales entre las neuronas que pertenecen la única capa que forma la red.

Las redes monocapa se utilizan típicamente en tareas relacionadas con la llamada autoasociación, por ejemplo, para reconstruir información de entrada que se presente a la red incompleta o distorsionada.

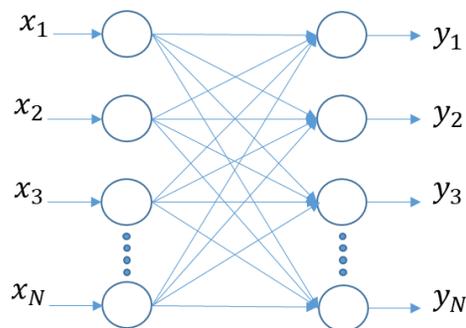


Figura 25. Red Monocapa [59]

Redes Multicapa

Las redes multicapa ofrecen una mayor complejidad tanto en su estructura como en el tipo o la cantidad de información que pueden clasificar, están conformadas por 2 o más capas de neuronas. Para identificar la capa a la cual pertenece una neurona, basta con fijar el origen de la entrada y el destino de su salida, normalmente, todas las neuronas de una capa tienen como señal de entrada a una capa anterior, y envían sus señales de salida a una capa posterior. A las conexiones de este tipo se les llama, *conexiones feedforward (hacia adelante)*.

En un gran número de redes también existe la opción de conectar la salida de las neuronas de una capa posterior a la entrada de una capa anterior, estas conexiones reciben el nombre de *conexiones feedback (hacia atrás)*. Dado estos tipos de redes, existe la posibilidad de clasificar dos tipos de redes con múltiples capas: redes *feedforward*, las cuales sus conexiones solo van hacia adelante, y redes *feedforward/feedback* en las cuales la comunicación va en ambos sentidos. En las redes *feedforward* siempre la señal va hacia adelante (a las capas posteriores), en ningún caso existen las conexiones hacia las capas anteriores, tampoco autorecurrentes (salida de una neurona aplicada a su propia entrada) o laterales (salida de una neurona aplicada a otra de la misma capa).

El PERCEPTRÓN, ADALINE, MADALINE y BACKPROPAGATION son las redes *feedforward* más conocidas, son especialmente útiles en aplicaciones de reconocimiento o clasificación de patrones, especialmente la última, Backpropagation [60].

3.2.3 Mecanismo de Aprendizaje

El aprendizaje, es el proceso al que una red neuronal es sometida y como respuesta se obtiene una modificación en sus pesos sinápticos, estos cambios, agudizan la capacidad de decisión de la red neuronal, cuanto más datos se le presenten y más cambios (adecuados) se realicen en sus pesos sinápticos, mejor y más confiable será la decisión de una red neuronal [61].

Se suelen considerar dos tipos de reglas de aprendizaje: las de aprendizaje supervisado y las de aprendizaje no supervisado. Debido a esto las redes neuronales se distinguen entre:

- Redes neuronales con aprendizaje supervisado
- Redes neuronales con aprendizaje no supervisado

La diferencia entre estos dos tipos de redes es, la existencia de un mecanismo supervisor el cual controla el proceso de aprendizaje de la red. En las redes con aprendizaje supervisado se establece la respuesta que la red deberá de generar a partir de una entrada determinada, el mecanismo supervisor evalúa la salida de la red contra la salida esperada y, si estas no coinciden se procederá a modificar el peso de las entradas, esperando que en el próximo cálculo la salida de la red se aproxime a la salida esperada [60].

Existen tres tipos de aprendizajes supervisados:

- Aprendizaje por corrección de error
- Aprendizaje por refuerzo
- Aprendizaje estocástico

El aprendizaje por corrección de error establece que, el peso de las conexiones de la red se ajustarán en función de la diferencia entre la salida esperada de la red contra la salida obtenida. La regla posible de aprendizaje para una neurona (Figura 26) se detalla en la siguiente fórmula[62]:

$$\Delta w_{ij} = \alpha y_i (d_j - y_j) \quad (13)$$

donde:

Δw_{ij} : Variación en el peso de la conexión entre i y j ($\Delta w_{ij} = w_{ji}^{actual} - w_{ji}^{anterior}$)

α : Factor de aprendizaje ($0 < \alpha \leq 1$)

y_i : Valor de salida de la neurona i .

d_j : Valor de salida deseado para la neurona j.

y_j : Valor de salida obtenido para la neurona j.

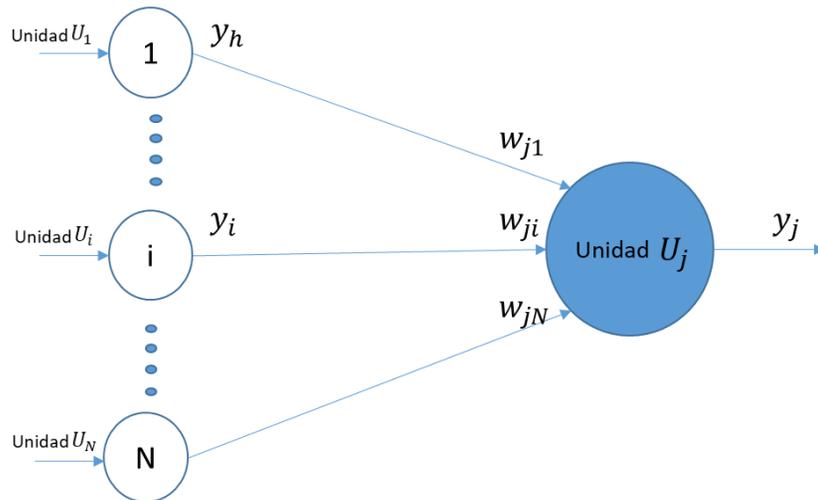


Figura 26. Unidad U_j [8]

Esta regla de aprendizaje es la utilizada por el perceptrón, desarrollado por Rosenblatt en 1958, sin embargo presenta una limitante muy importante, el no considerar el error global de la red durante el proceso de aprendizaje y solo se enfoca al error de una neurona. Este aspecto es mejorado por Widrow y Hoff en 1960, por la denominada, *regla delta o regla del mínimo error cuadrado*. Widrow y Hoff desarrollaron una función que acumulaba el error global de la red, expresada de la siguiente manera[63]:

$$Error_{global} = \frac{1}{2P} \sum_{k=1}^P \sum_{j=1}^N (y_j^{(k)} - d_j^{(k)})^2 \quad (14)$$

donde:

N: Neuronas de Salida

P: Cantidad de Información que debe aprender la red.

$\frac{1}{2P} \sum_{j=1}^n (y_j^k - d_j^k)^2$: Error cometido de la información k -ésima.

Lo que trata la regla delta es de encontrar los pesos adecuados para las conexiones que minimicen la función de error, el ajuste de los pesos se hace proporcionalmente a la variación del error que se obtiene al variar el peso, expresada en la siguiente fórmula:

$$\Delta w_{ij} = k \frac{\partial Error_{global}}{\partial w_{ij}} \quad (15)$$

Otro algoritmo de aprendizaje, es el llamado *regla delta generalizado* o algoritmo de retropropagación del error (backpropagation error), se le conoce también como la regla LMS (Least-Mean-Square Error). Este algoritmo se trata de una generalización a la regla delta para poder aplicarla a redes con conexiones hacia adelante (feedforward).

3.3 FUNCIONAMIENTO DE UNA RED NEURONAL ARTIFICIAL

Una red neuronal artificial puede analizar cualquier tipo de datos, sin ningún conocimiento previo de ellos, además, una red neuronal artificial puede ser representada tanto como una fórmula matemática, como en un modelo gráfico, son también métodos muy robustos al momento de analizar distintos fenómenos, pero requieren de un paradigma, el cual a través de mecanismos bien definidos explorarán y analizarán los datos que le sean presentados [64].

Las redes con conexiones hacia adelante, tienen la característica de poseer niveles y conexiones exclusivamente hacia adelante. Estas redes son excelentes clasificadores de patrones y utilizan el aprendizaje supervisado. Dentro de este grupo se encuentra el perceptrón multicapa con el algoritmo que lo opera llamado retropropagación, juntos forman la red con mayor aplicación de hoy en día [65].

3.3.1 El Perceptrón

Fue el primer modelo de neurona, desarrollado por Rosenblatt en 1958, ya se comentó que su regla de aprendizaje tiene una limitante muy importante al no considerar el error global, sin embargo, fue el modelo que despertó un gran interés en los años 60 por su capacidad de aprender a reconocer patrones sencillos [66].

El perceptrón separa dos regiones por un hiperplano, la ecuación que divide las regiones queda determinada por el peso de las entradas de la neurona y su umbral, dada la forma:

$$y_k = \sum_{j=0}^m w_{kj}x_j \quad (16)$$

donde:

$y_k = w_1x_1 + w_2x_2$: Expresión matemática de una recta.

w_0x_0 : Término independiente que permitirá desplazar la recta del origen (Umbral).

Su principal objetivo es la modificación sistemática de sus pesos (etapa de aprendizaje), hasta encontrar los pesos adecuados que permitan trazar una recta que divida el plano en dos espacios de las dos clases de valores de entrada.

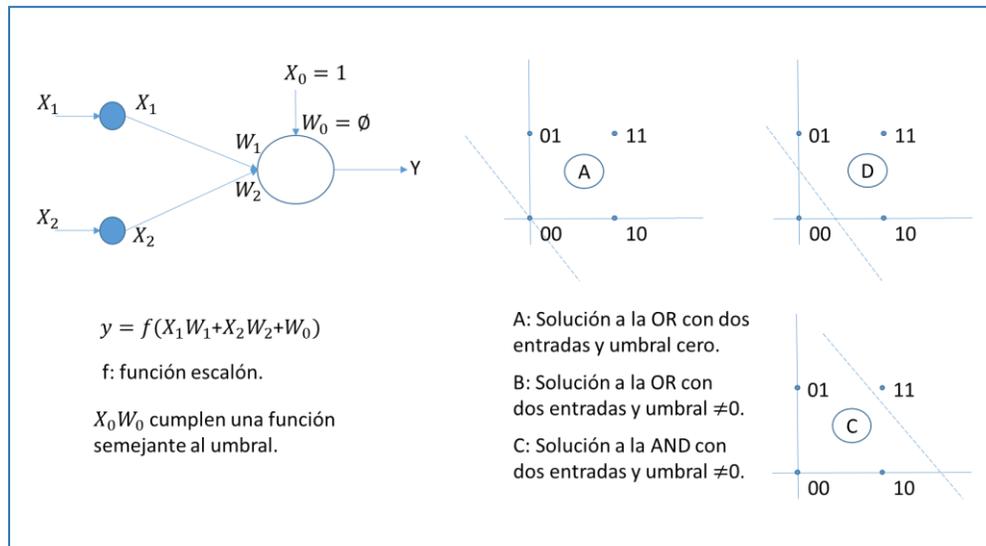


Figura 27. Perceptrón aplicado a la función OR [8]

Para la función OR se separan los valores 01, 10 y 11 del valor 00, si no se tuviese el umbral, la recta siempre pasaría por el origen, véase la Figura 27(A), agregando el umbral el número de soluciones se incrementan, además, si se quisiera separar la función AND en donde los valores 00, 01 y 10 dan diferente salida a 11, no podría llegarse a una solución sin tener el umbral, Figura 27 (C).

El algoritmo de aprendizaje de un perceptrón para clasificar la función OR es:

1. Inicialización de pesos y del umbral (w_i).
2. Se presenta el par de entrada (x_1, x_2) , en el caso del OR los pares de entrada serian: 00, 01, 10 y 00, un total de 4 pares y siempre de 1 para el caso de la entrada al umbral (x_0).
3. Se calcula la salida de la neurona:

$$y(t) = f \left(\sum_{i=0}^n w_i(t)x_i(t) \right) \quad (17)$$

donde: f es la función de transferencia escalón.

4. Si la salida deseada $d(t)$ es diferente a la obtenida $y(t)$ se modifican los pesos:

$$w_i(t + 1) = w_i(t) + \alpha [d(t) - y(t)] * x_i \quad (18)$$

5. Vuelve al paso 2.

El algoritmo termina hasta que los cuatro pares de entrada no presenten modificación en los pesos.

La separabilidad lineal usada para solucionar el problema de la función OR no es aplicable para problemas que se dividan en 3 o más regiones, ejemplo la OR – Exclusiva (XOR), en donde 00 y 11 devuelven 0 mientras que 01 y 10 devuelven 1, para este caso se ha de usar 2 neuronas para poder generar dos rectas, estas dos neuronas serian asociadas a una neurona de salida, obteniendo de esta manera una red de 3 neuronas conocida como perceptrón multicapa o multinivel [67].

3.3.2 El Perceptrón Multicapa

Esta red es de tipo feedforward, está compuesta por varias capas (Figura 28), esto le permite encontrar soluciones más complejas que la función OR. Las regiones de decisión (Figura 29) de un perceptrón multicapa varían en función del número de neuronas que componga la red.

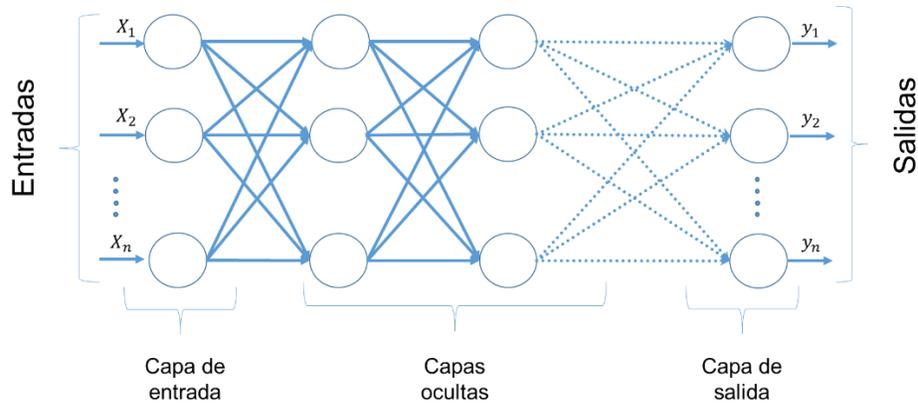


Figura 28. Perceptrón multicapa [68]

Un perceptrón básico de 2 capas (la del vector de entradas y la salida con función escalón), sólo puede identificar dos regiones y separarlas mediante una única frontera lineal.

Estructura	Regiones de decisión	Problema de la XOR	Clases con regiones mezcladas	Formas de regiones más generales
2 CAPAS 	MEDIO PLANO LIMITADO POR UN HIPERPLANO			
3 CAPAS 	REGIONES CERRADAS O CONVEXAS			
4 CAPAS 	ALBITRARIA COMPLEJIDAD LIMITADA POR EL NÚMERO DE NEURONAS			

Figura 29. Regiones de decisión generadas por un perceptrón multicapa [8]

El creciente número de neuronas en la red nos permite también aumentar la complejidad de la solución, esto nos lleva al problema de la selección del número de neuronas para la red, es decir, el tamaño de la red deberá ser el adecuado para que se forme una región lo suficientemente compleja acorde a la resolución del problema.

Un perceptrón con cuatro capas, puede formar regiones de decisión sumamente complejas, sin embargo estudios revelan que tampoco es conveniente que el tamaño de la red sea muy grande ya que puede generar ruido, dicho de otra manera, la solución que proponga la red puede no ser la adecuada para el problema.

Si bien un perceptrón multicapa ofrece una red de neuronas interconectadas, se necesita de un algoritmo que la opere, uno de los más usados actualmente debido a su versatilidad es el de retropropagación.

3.3.4 Aplicaciones en Crecimiento Urbano

El algoritmo de retropropagación usado en el perceptrón multicapa se ha aplicado a distintas clase de problemas, su versatilidad se debe al proceso de aprendizaje que implementa, ya que solo utiliza dos ecuaciones para propagar el error hacia atrás. Los campos de acción más representativos para este algoritmo son[69]:

- Traducción de texto escrito.
- Reconocimiento de lenguaje hablado.
- Reconocimiento óptico de caracteres (OCR).

En el ámbito de los sistemas de simulación para el crecimiento urbano, el algoritmo de retropropagación se ha aplicado en distintas fases. Uno de los usos más interesantes y que tienen especial relevancia para la presente investigación, es en la generación de mapas de probabilidades de cambios.

En el trabajo “Using neural networks and cellular automata for modelling intra- urban land- use dynamics” [70], se define un modelo (Figura 30) en donde una RNA calcula mapas de probabilidades previo a un análisis a través de un autómata celular.

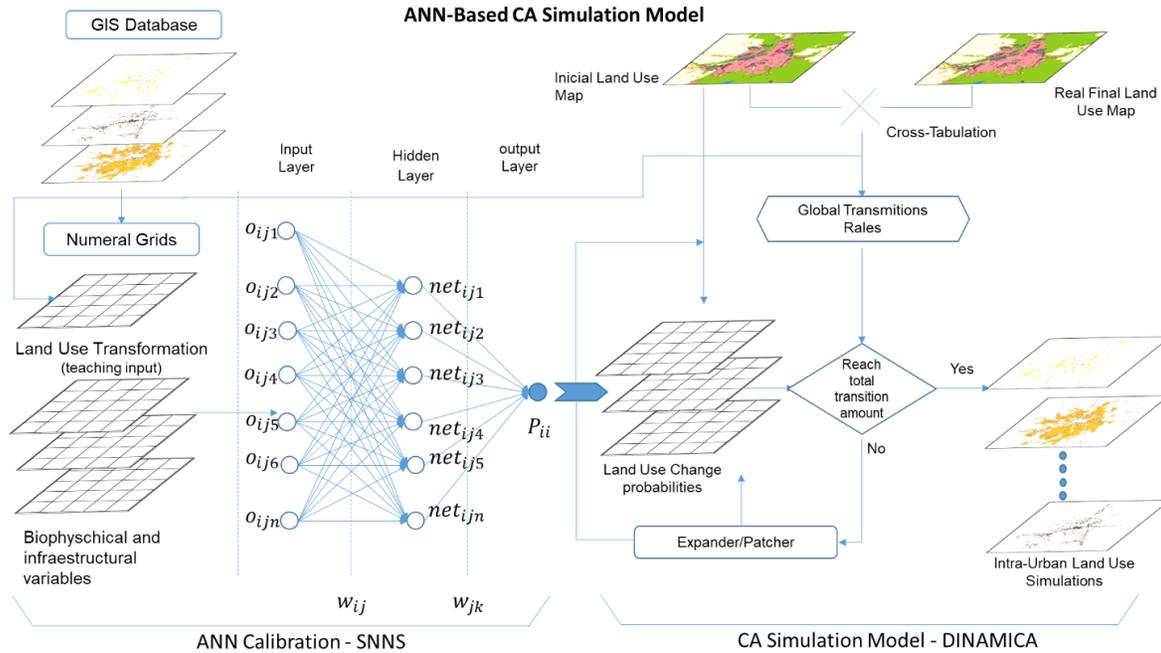
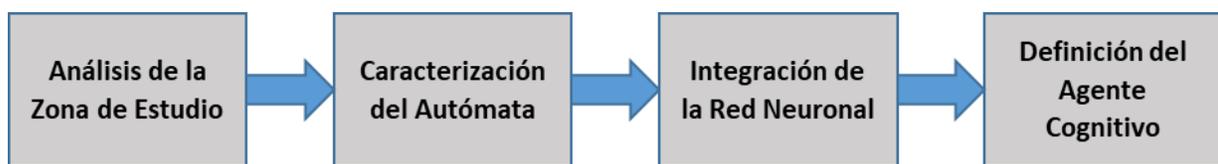


Figura 30. Modelo de simulación basado en RNA-AC [70]

La capacidad de clasificación de una RNA ha permitido su integración con otras técnicas como los sistemas multiagentes, en el trabajo “Cooperative Coevolution of Multi-Agent Systems” [71], se aborda el tema de sistemas multiagentes controlados por una red neuronal. Cada agente, con su propia red neuronal definida, es conducido para conseguir un objetivo.

En esta investigación, se hizo uso de una red neuronal para el control de un agente encargado de modelar las variables sociodemográficas dentro del sistema de simulación.

PARTE III METODOLOGÍA



Ruta del proceso

La metodología de cada una de las técnicas utilizadas en el desarrollo de los experimentos se presenta en 4 capítulos. El primer capítulo inicia con la generación de la cartografía base, en donde se digitaliza la zona de estudio para posteriormente generar los mapas ráster, los cuales servirán como entrada al sistema, además se ponderan los factores de entrada y se les asigna el peso correspondiente a cada uno de los factores de influencia con base a los resultados de la regresión logística. En el segundo capítulo se tiene la caracterización del AC, en él se lleva a cabo la definición de la vecindad y de las variables de entrada. También, se definen las matrices de calibración, las cuales afectarán a cada uno de los factores de entrada. El tercer capítulo trata sobre la definición de una red neuronal la cual es capaz de generar probabilidades de cambio para distintos usos, se explora su uso en conjunto con el AC y se define el mecanismo de cálculo. En el cuarto capítulo, se define un agente cognitivo (integrando la RNA con el agente) el cual pueda modelar factores sociales como, escolaridad, nivel de ingresos y edad. Además se establece la interacción del agente cognitivo con el autómata celular, con el objetivo de generar simulaciones prospectivas que aborden temáticas territoriales y socioeconómicas.

Para llevar a cabo los experimentos, fue necesario establecer las variables que tienen importancia en el crecimiento urbano, para ello fue necesario identificar los tipos de usos de suelo urbano y llevar a cabo la digitalización de la zona de estudio.

CAPÍTULO 4: ANÁLISIS DE LA ZONA DE ESTUDIO

En el presente capítulo se detallan los elementos de análisis del crecimiento urbano de Culiacán Sinaloa, con el objetivo de obtener un panorama general de los cambios a nivel de uso urbano que se han presentado a lo largo de 3 fechas distintas; 1997, 2004 y 2011. De inicio se tienen los mapas de la región, se parte con la digitalización de los mapas, posteriormente se generan los mapas ráster para poder realizar los cálculos pertinentes a través de land change modeler de Idrisi, estos cambios serán medibles a través de una regresión logística espacial, con la cual obtendremos el porcentaje de participación de cada uno de los usos en el desarrollo de la ciudad.

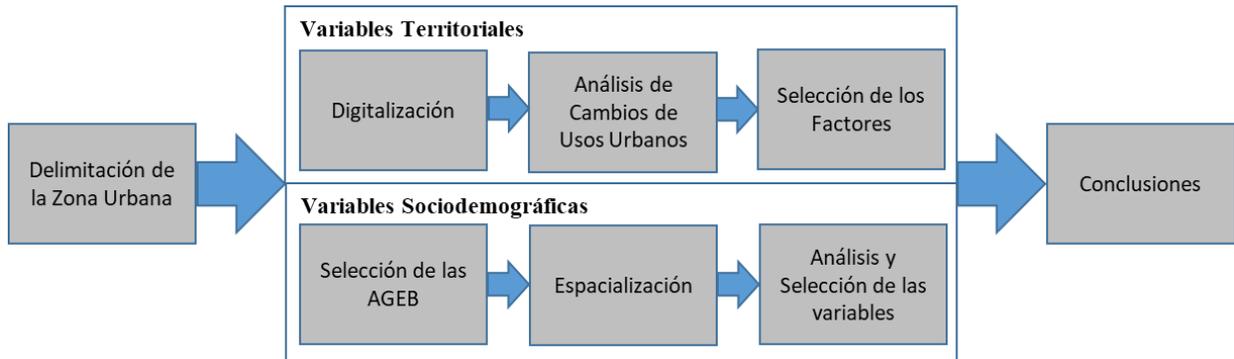


Figura 31. Ruta del proceso

El proceso (Figura 31) para obtener las variables necesarias, se lleva a cabo separando en 2 categorías la información espacial. Primero se necesitan los mapas del territorio, los cuales se digitalizan, se analizan cambios y se seleccionan los factores relevantes. Segundo, para la obtención de las variables socioeconómicas, se seleccionan las AGEB que se encuentran dentro del área de estudio, se espacializan, y se analiza su cambio a través de las fechas de estudio.

4.1 DELIMITACIÓN DE LA ZONA URBANA

La presente investigación se aplica al municipio de Culiacán, se ubica en la región centro del Estado de Sinaloa (Figura 32), formando parte del noroeste de México. Le corresponden las coordenadas: 24°48'15" N (latitud norte) y 107°25'52" O (longitud oeste), con una altitud de 54 metros sobre el nivel del mar. La ciudad de Culiacán concentra el 81% de la población del municipio que en los últimos 20 años ha registrado un crecimiento territorial y demográfico muy significativo, según el último censo del Instituto Nacional de Estadística y Geografía (INEGI), a la fecha la ciudad cuenta por el orden de 800,000 habitantes. El año de 1980 la ciudad registraba un área urbana de 5,163 hectáreas, para 1990 se incrementó a 7,377 hectáreas y para el año 2001 se tienen 9,800 hectáreas. Este crecimiento se dio de una manera desordenada, es decir, anárquica al amparo de liderazgos políticos dando como resultado que actualmente, la ciudad cuente con más de 275 colonias, las cuales en su mayoría se formaron en terrenos ejidales, zonas de reserva ecológica, lugares sin factibilidad de servicios por su composición topográfica, etc.[72] .

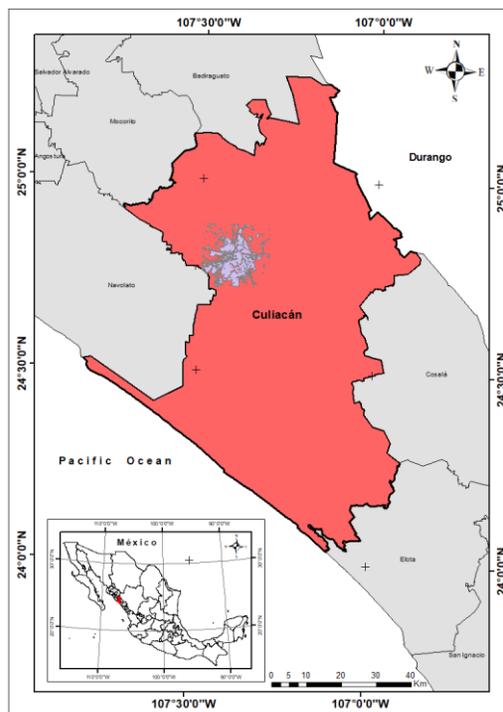


Figura 32. Zona urbana del municipio de Culiacán Sinaloa

Para regular el crecimiento desordenado presentado por la ciudad de Culiacán, se crea en diciembre de 2003 el Instituto Municipal de Planeación Urbana de Culiacán (IMPLAN), cuya misión es *“Ser el espacio técnico del Gobierno Municipal y la sociedad, responsable de todo el proceso de planeación: Planes, programas y proyectos que concentra y sistematiza información y gestiona bajo criterios estrictamente técnicos la planeación urbana de manera integral”*. En septiembre de 2008 bajo la gestión del IMPLAN se aprueba el Plan Director de Desarrollo Urbano de Culiacán, el cual servirá como un instrumento legal, técnico y normativo, que permitirá ordenar y regular el desarrollo urbano presente y futuro de la ciudad de Culiacán. La importancia de este plan radica en que (IMPLAN) [73]:

- Establece los usos de suelo permitidos y los prohibidos en la ciudad y área de futuro crecimiento.
- Señala la densidad de población y las alturas máximas de construcción permitidas.
- Induce a un reordenamiento de la estructura urbana de la ciudad, de carácter más compacto para evitar mayores desplazamientos y hacer más eficiente la inversión pública.
- Establece los límites hasta donde se llevarán servicios públicos y se permitirá urbanizar.
- Es congruente con la planeación del transporte y las vialidades.
- Incluye novedosos instrumentos normativos que hacen posible su aplicación.

Del plan director de desarrollo se desprenden algunas de las fuentes geográficas requeridas para la representación del territorio en un espacio de evolución, esto será de suma importancia para la caracterización del autómata.

Escalas de trabajo

Las fuentes necesarias para el estudio obedecen a escalas diferentes, en algunos casos se estuvo supeditado a la información disponible sobre el área y en otras por que la escala lo requería, esta multiescalaridad nos permitió abstraer el conocimiento adecuado del área de estudio.

Las ortofotos (geotif) de la zona urbana corresponden a las fechas de 1997 a una escala de 1:20000, de 2004 a una escala de 1:10000 y de 2011 a una escala de 1:10000 obtenidas de las oficinas de INEGI.

4.2 DIGITALIZACIÓN

Para identificar los distintos aspectos del crecimiento de la zona urbana en Culiacán y para delimitar su conjunto de AGEB, fue necesario realizar un proceso de digitalización al mosaico de ortofotos (Figura 33 y Figura 34). El primer paso consistió en la verificación de la georreferenciación de las imágenes (todas proyectadas en WGS 84 / UTM 13N) y de la delimitación del área de estudio.

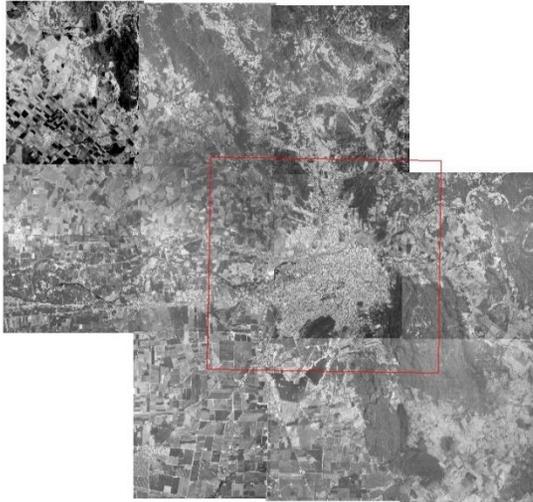


Figura 33. Mosaico de ortofotos de la zona urbana de Culiacán 1997 y el polígono de estudio (en rojo)

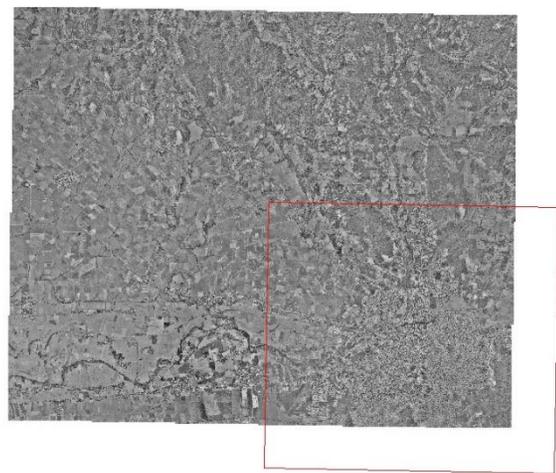


Figura 34. Mosaico de ortofotos de la zona urbana de Culiacán 2004 y el polígono de estudio (en rojo)

4.2.1 Categorías de uso urbano

La clasificación de los distintos usos urbanos (Tabla 3), se llevó a cabo tomando en cuenta a los lineamientos indicados por el Programa Municipal de Desarrollo Urbano de Culiacán del IMPLAN (Instituto Municipal de Planeación) [73].

Tabla 3. Clasificación de usos de suelo urbano.

índice	Usos Urbanos
1	Huecos Urbanos
2	Servicios
3	Zona Agrícola
4	Zona Comercial
5	Zona Forestal
6	Zona Industrial
7	Vivienda
8	Cuerpos de Agua
9	Equipamientos
10	No Urbanizable

4.2.2 Generación de Mapas Vectoriales

El proceso de digitalización se llevó a cabo sobre el mosaico de ortofotos (Figura 35), cabe señalar que no se partió de cero, existía un trabajo previo de digitalización para la zona urbana de Culiacán de 2011 [74], y sobre eso se realizó una “actualización” cartográfica para cada una de las escalas temporales.

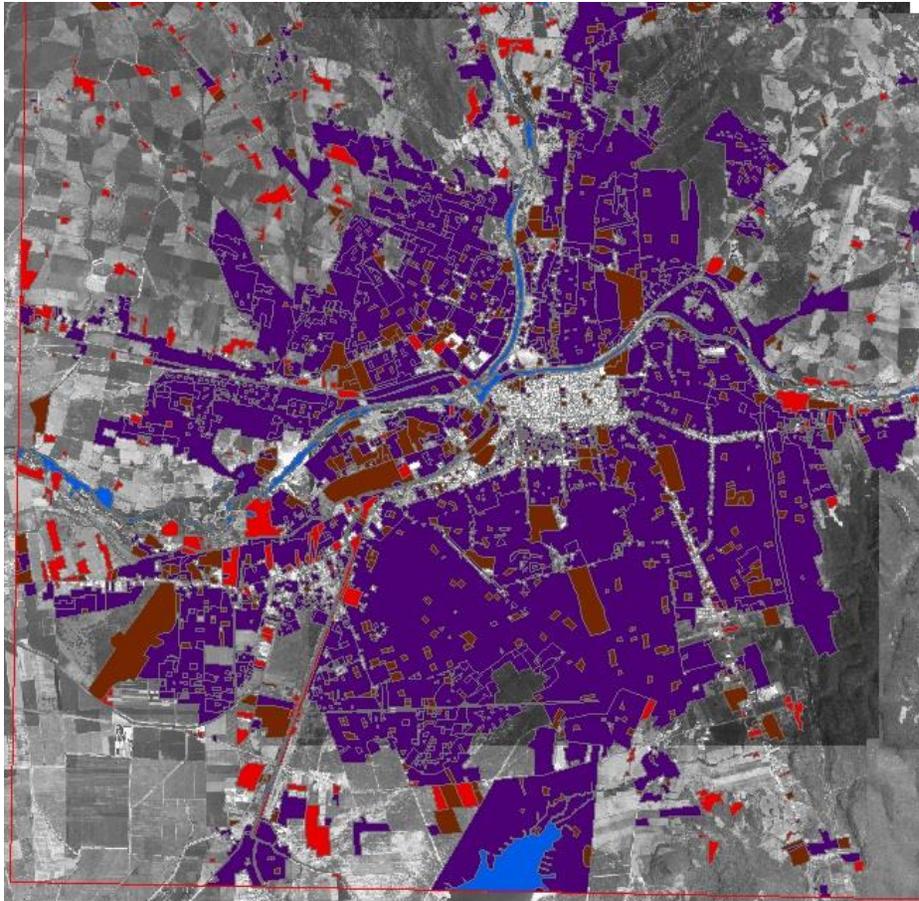


Figura 35. Proceso de digitalización de la zona urbana sobre la ortofoto de 1997

La digitalización y clasificación de los usos urbanos muchas veces tiende a ser subjetiva, ya que está sujeta a la apreciación de quien esté realizándola, bajo esta premisa se consultaron distintos medios electrónicos (fotos históricas en internet, google earth, etc.) para corroborar la clasificación de cada uno de los usos urbanos durante el proceso de digitalización, a continuación (Figura 36) se muestran una serie de imágenes donde se explican cada uno de los usos.

 <p>Huecos urbanos</p>	 <p>Servicios</p>	 <p>Zona agrícola</p>
 <p>Zona Comercial</p>	 <p>Zona forestal</p>	 <p>Zona industrial</p>
 <p>Cuerpos de agua</p>	 <p>Zona urbana</p>	 <p>Equipamientos</p>
 <p>No urbanizable</p>	<p>Huecos urbanos: áreas que se encuentra dentro de la zona urbana pero que no tienen ningún tipo de infraestructura y que no pueden considerarse como área verde por su descuido.</p> <p>Servicios: zonas establecidas que proporcionan cierta facilidad o ayuda para desempeñar tareas de la vida cotidiana.</p> <p>Zona agrícola: Son las zonas donde se tiene áreas de cultivo (vegetación, árboles frutales) bajo el cuidado del hombre.</p> <p>Zona comercial: Son las zonas donde se pueden comprar cualquier tipo de cosas, pero están establecidas y son de carácter notorio.</p> <p>Zona forestal: contienen grandes cantidades de vegetación y/o árboles que no están bajo el cuidado del hombre.</p> <p>Zona industrial: Son empresas, industrias y/o fábricas con el fin de transformar y/o manufacturar productos.</p> <p>Cuerpos de agua: zonas que contienen agua de manera natural y no son creadas por el hombre.</p> <p>Zona urbana: Son las áreas donde se tiene ya sea una o más casa habitación.</p> <p>Equipamientos: conformada por parques, escuelas e iglesias, tienen un crecimiento nulo o muy lento a lo largo del tiempo.</p> <p>No urbanizable: se considera no urbanizable o como parque debido a que no se pueden construir en esta zona ya que se conserva como área protegida.</p>	

Figura 36. Detalle de usos urbanos

Finalmente se han generado tres archivos shape los cuales representan la situación de la zona urbana de 1997, 2004 y 2011 (Figura 37). En este punto pueden ser fácilmente exportables al formato ráster para un posterior análisis de cambio de usos urbanos. También como productos, se tienen las vialidades de ambos periodos (Figuras 27 y 28),

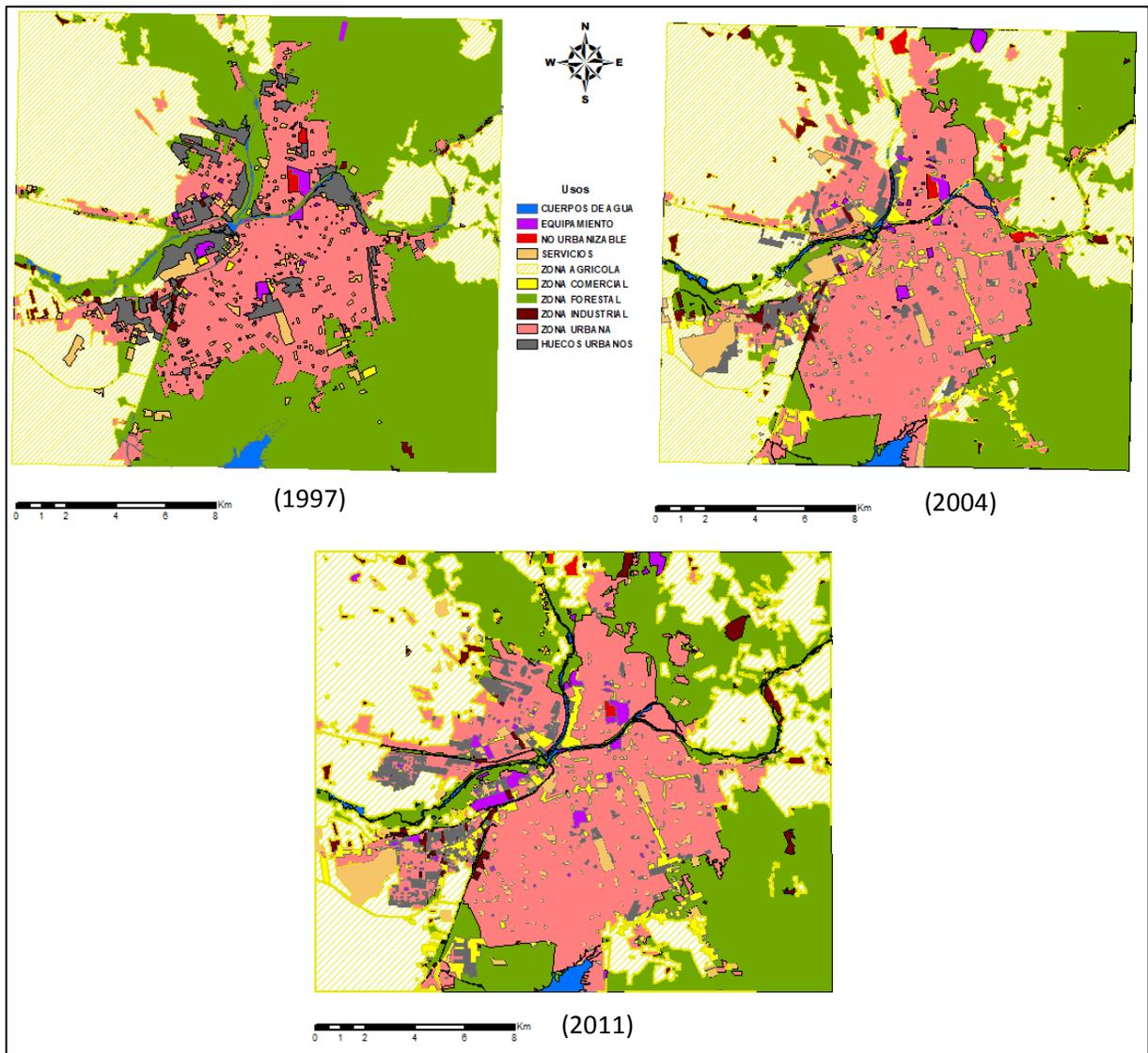
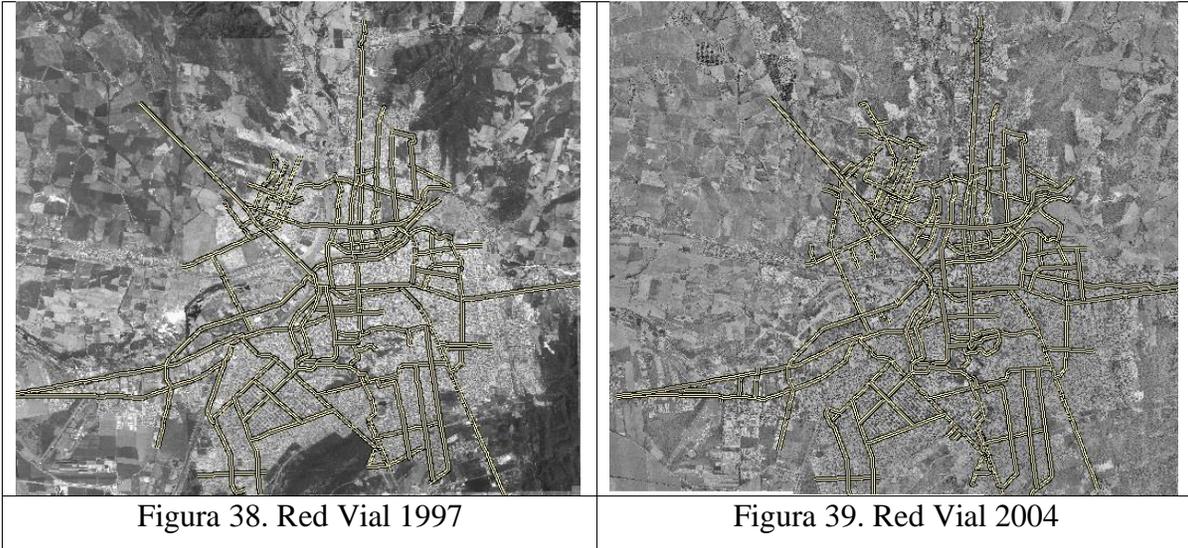


Figura 37. Mapas Vectoriales de la Zona Urbana

Además de los mapas de uso de suelo urbano, es necesario generar para su análisis, los mapas de proximidad a carreteras, accesibilidad y aptitud.



4.2.3 Rasterización

Derivado del proceso de digitalización y generación de mapas vectoriales, la conversión de los mapas a formato ráster se realizó mediante la opción “Polygon to Raster” de la caja de Herramientas de ArcGis. Debido a que las manzanas de la ciudad tienen una dimensión aproximada a 50 mts^2 , y siendo estas el principal y más elemental componente observable para los análisis de crecimiento urbano, se definió la UMC (unidad mínima Cartografiable), en 50 mts^2 para el tamaño del pixel. Desde Idrisi, con la herramienta Database Workshop y la opción “to Raster” se crearon los mapas. Los mapas rasterizados tienen una resolución de 397×366 píxeles, cubren una zona de $19,850 \text{ mts} \times 18,300 \text{ mts}$. Estos servirán como base para el análisis del crecimiento urbano de la zona de estudio, los mapas (Figura 40) muestran los usos definidos en el apartado 4.2.1.

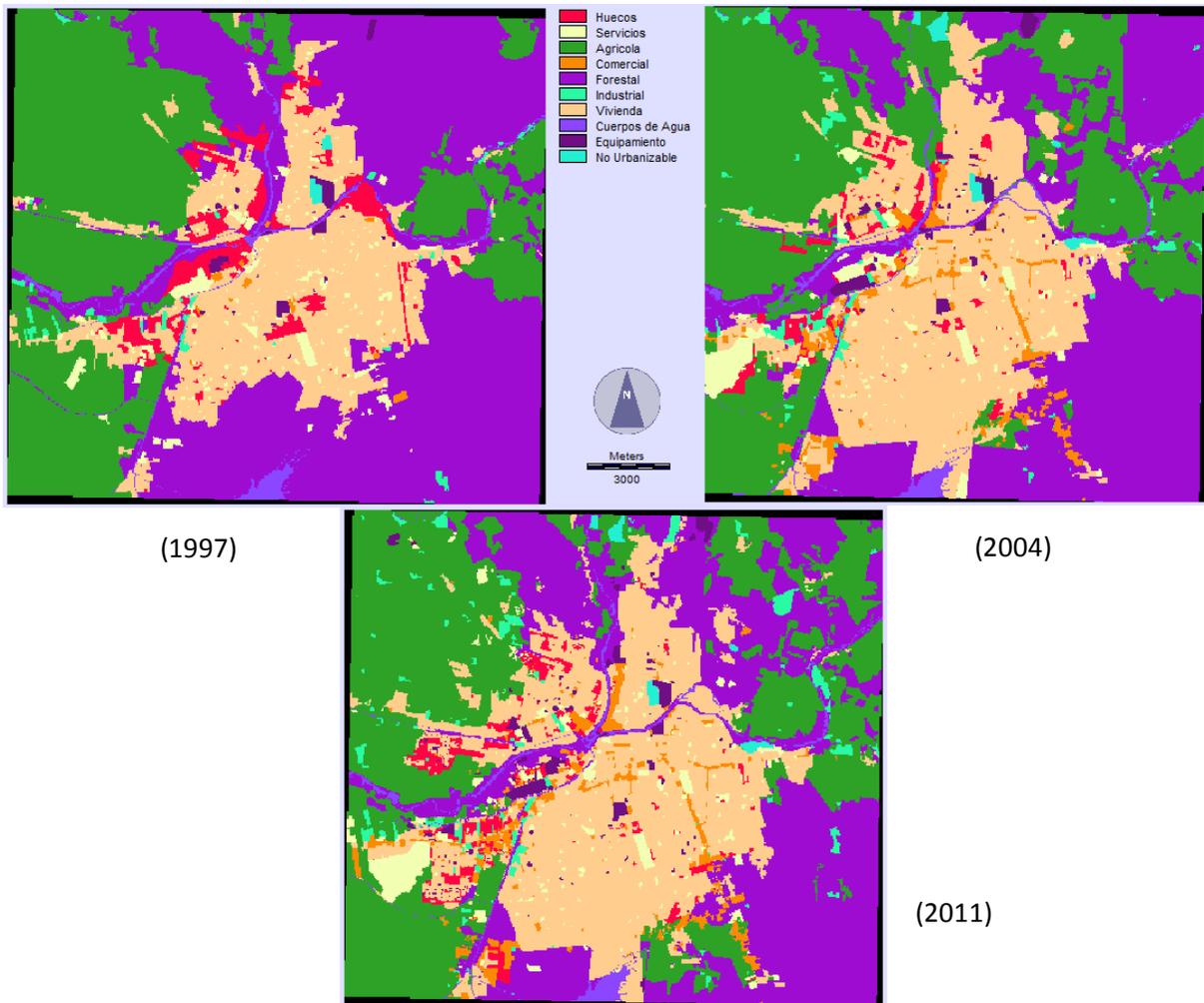


Figura 40. Mapas Ráster de la Zona Urbana

4.2.4 Herramientas de Análisis

Se ha usado el software Idrisi Selva® para el análisis de los elementos geográficos pertinentes en la investigación. Principalmente son 3 herramientas claves para el análisis de las variables ráster, a continuación se detallan:

Crosstab

Realiza un análisis mediante una tabulación cruzada (Tabla 4), donde se comparan dos mapas y sus respectivas variables categóricas. Esta herramienta, basada en el trabajo de Pontius[29], permite obtener las ganancias, pérdidas, persistencias, intercambios y cambios totales por cada categoría, entre dos imágenes que representan dos momentos temporales (véase Figura 41).

Tabla 4. Matriz de tabulación cruzada para dos mapas de diferente fecha [29]

		Tiempo 2						
		1	2	3	4	5	6	7
Tiempo 1	1		Clase 1	Clase 2	Clase n	Suma T_1	Pérdidas
	2	Clase 1	P_{11}	P_{12}	P_{1n}	P_{1+}	$P_{1+} - P_{jj}$
	3	Clase 2	P_{21}	P_{22}	P_{2n}	P_{2+}	$P_{2+} - P_{jj}$
	·	·	·	·	·	·	·	·
	·	·	·	·	·	·	·	·
	·	·	·	·	·	·	·	·
	·	·	·	·	·	·	·	·
	·	·	·	·	·	·	·	·
4								
5	Clase n	P_{n1}	P_{n2}	P_{nn}	P_{n+}	$P_{n+} - P_{jj}$	
6	Suma T_2	P_{+1}	P_{+2}	P_{+n}	P		
7	Ganancias	$P_{+1} - P_{jj}$	$P_{+2} - P_{jj}$	$P_{+n} - P_{jj}$			

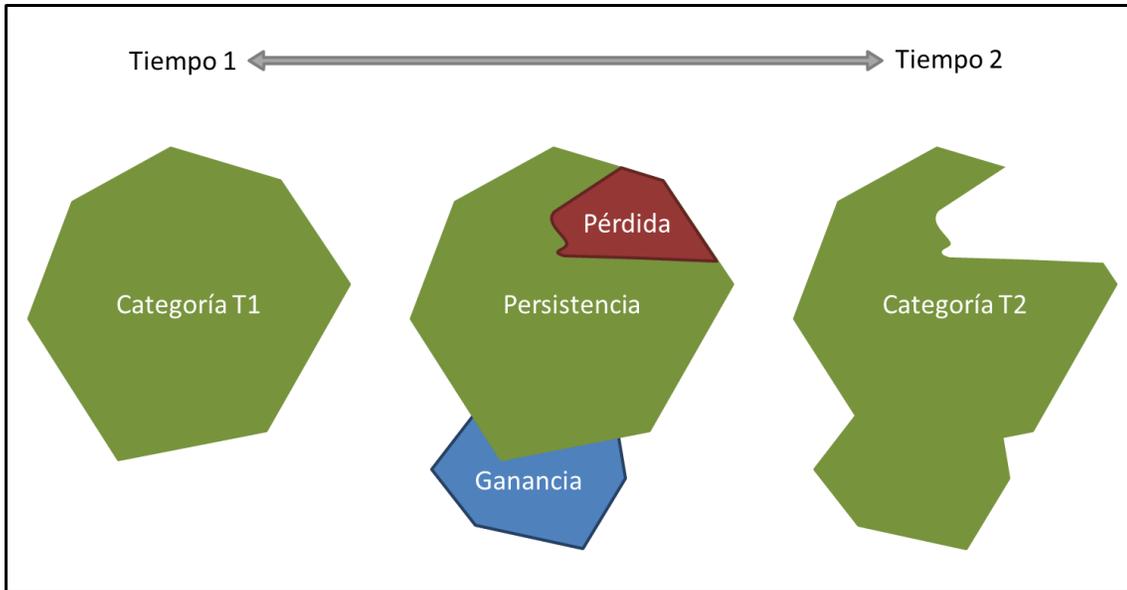


Figura 41. Categoría con superficie inicial y final, modificado de [75]

Logisticreg

Realiza una regresión logística binomial, en la que la variable dependiente de la entrada debe ser de naturaleza binaria, es decir, solo puede tener dos valores posibles (0 y 1). Dicho análisis de regresión se emplea generalmente para estimar un modelo que describe la relación entre una o más variables independientes continuas a la variable dependiente binaria. El supuesto básico es que la probabilidad de la variable dependiente toma el valor de 1 (respuesta positiva) siguiendo la curva logística, y su valor se puede estimar con la siguiente fórmula:

$$P(y = 1|X) = \frac{\exp(\sum BX)}{1 + \exp(\sum BX)} \quad (19)$$

donde:

P es la probabilidad de la variable dependiente.

X son las variables independientes, $X = (x_0, x_1, x_2, \dots, x_k), x_0 = 1$

B son los parámetros estimados, $B = (B_0, B_1, B_2, \dots, B_k)$

CAPÍTULO 5: CARACTERIZACIÓN DEL AUTÓMATA CELULAR

La caracterización de un autómata celular que permita modelar una zona urbana de estudio, tiene como requisito preparar ciertas variables para su funcionamiento. Las etapas para reunir lo necesario son principalmente:

- Generación de la cartografía base, necesaria como entrada al sistema. Esto permitirá evolucionar la situación territorial de un periodo determinado hacia un horizonte futuro, todo esto en un espacio de evolución bidimensional en un AC.
- Análisis de cambios, con ellos no solamente se tiene la información espacial, sino que se obtienen tendencias necesarias para las matrices de calibración, definición de reglas de transición del AC.
- Desarrollo de herramientas que nos permitan manipular cualquier aspecto de un AC, tal herramienta fue diseñada para admitir interfaces como, redes neuronales artificiales y agentes inteligentes.

Las variables de entrada al autómata (Figura 42), preparadas a través de un proceso de actualización cartográfica.

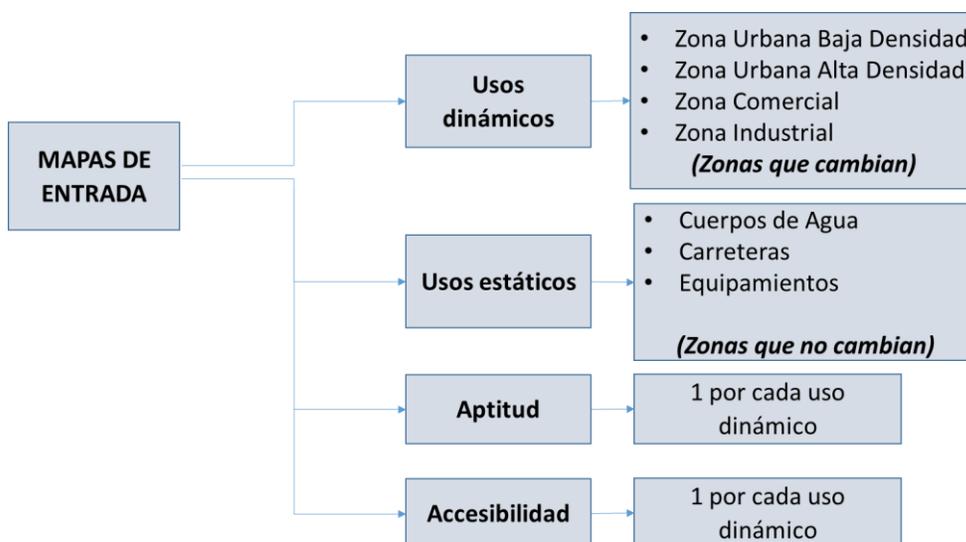


Figura 42. Variables de entrada al autómata

5.1 CÁLCULO DEL POTENCIAL DE TRANSICIÓN

El potencial de transición nos indica, el grado de idoneidad de un pixel, para convertirse en un determinado estado a otro, que para el caso de la presente investigación, tales estados representan usos urbanos presentes en el territorio modelado.

Formalmente un mapa ráster puede estar representado por una matriz de valores reales. Esta matriz se representa como $A = \{a_{ij}\}$ de orden $m \times n$ tal que $0 \leq i \leq m, 0 \leq j \leq n$ y donde cada elemento $A = [a_{ij}] \in \mathbb{R}$.

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

Un AC requiere una matriz de vecindad que sirva como filtro (ventana de observación) para analizar cada elemento $A = [a_{ij}]$, esta vecindad se representa formalmente como $B = \{b_{ii}\}$ de orden $n \times n$ tal que $0 \leq i \leq n$ y donde cada elemento $B = [b_{ii}] \in \mathbb{Z}$.

$$B = \begin{bmatrix} b_{i-1,j-1} & b_{i-1,j} & b_{i-1,j+1} \\ b_{i,j-1} & b_{ij} & b_{i,j+1} \\ b_{i+1,j-1} & b_{i+1,j} & b_{i+1,j+1} \end{bmatrix}_{3 \times 3 \text{ vecindad}}$$

Se usa el filtro de vecindad para calcular el potencial de transición de un estado h a otro j para cada elemento $A = [a_{ij}]$.

5.2 REGLA DE EVOLUCIÓN

La regla de evolución determina el comportamiento del AC. La metodología propuesta por R. White (mencionada en el apartado 2.1), en su modelo AC delimitado para la evolución de dinámicas urbanas, se especifica el cálculo del potencial de transición, se detalla a continuación [4]:

$$P_{hj} = v s_j a_j \left(1 + \sum_{k,i,d} m_{kd} I_{id} \right) + H_j \quad (20)$$

donde:

P_{hj} : es el potencial de transición del estado h al estado j .

v : termino de perturbación estocástica $v = 1 + [-\ln(\text{random})]^x$, ($0 < \text{random} < 1$), x permite ajustar el tamaño de la perturbación.

s_j : mapa ráster de aptitud para el estado j .

a_j : mapa ráster de accesibilidad para el estado j (distancias euclidianas a la carretera más cercana).

m_{kd} : matriz de calibración, contiene los valores de cada celda en función de su uso urbano k y la distancia a la celda analizada d .

Para cada elemento a_{ij} se calcula su P_{hj} si y solo si, su aptitud $s_j > 0$, es decir, para cada celda (pixel) en el mapa se calculará su potencial de transición, excepto para aquellas en las cuales su aptitud sea igual a cero. Para el cálculo de la vecindad, la matriz de calibración m_{kd} permite que cada celda vecina b_{ii} (dentro del filtro de vecindad) tenga una importancia (peso) con base a su estado k y distancia con respecto a la celda analizada a_{ij} . Las celdas vecinas cercanas, generalmente tendrán un peso mayor, se toman valores positivos para un efecto de atracción y negativos para repulsión, estos valores tienden a disminuir a medida que se incrementa la distancia entre la celda vecina y la analizada, esto es llamado efecto de disminución a distancia (Distance Decay Effect) [76]. Cuando se analizan las celdas vecinas, se requiere que estas sean analizadas según su estado, el componente I_{id} ayuda a filtrar (multiplicando por 1) las celdas que se ocupan analizar.

5.2.1 Definición de la Vecindad

La vecindad definida y más adecuada para los cálculos de crecimiento urbano[27], [77] consta de una malla bidimensional de 11x11 celdas, resultando en una ventana de observación de 121

celdas, esta actúa como una plantilla (Figura 43), y sirve como base para calcular los vecinos de una celda determinada.

(-5,-5)	(-5,-4)	(-5,-3)	(-5,-2)	(-5,-1)	(-5,0)	(-5,1)	(-5,2)	(-5,3)	(-5,4)	(-5,5)
(-4,-5)	(-4,-4)	(-4,-3)	(-4,-2)	(-4,-1)	(-4,0)	(-4,1)	(-4,2)	(-4,3)	(-4,4)	(-4,5)
(-3,-5)	(-3,-4)	(-3,-3)	(-3,-2)	(-3,-1)	(-3,0)	(-3,1)	(-3,2)	(-3,3)	(-3,4)	(-3,5)
(-2,-5)	(-2,-4)	(-2,-3)	(-2,-2)	(-2,-1)	(-2,0)	(-2,1)	(-2,2)	(-2,3)	(-2,4)	(-2,5)
(-1,-5)	(-1,-4)	(-1,-3)	(-1,-2)	(-1,-1)	(-1,0)	(-1,1)	(-1,2)	(-1,3)	(-1,4)	(-1,5)
(0,-5)	(0,-4)	(0,-3)	(0,-2)	(0,-1)	(0,0)	(0,1)	(0,2)	(0,3)	(0,4)	(0,5)
(1,-5)	(1,-4)	(1,-3)	(1,-2)	(1,-1)	(1,0)	(1,1)	(1,2)	(1,3)	(1,4)	(1,5)
(2,-5)	(2,-4)	(2,-3)	(2,-2)	(2,-1)	(2,0)	(2,1)	(2,2)	(2,3)	(2,4)	(2,5)
(3,-5)	(3,-4)	(3,-3)	(3,-2)	(3,-1)	(3,0)	(3,1)	(3,2)	(3,3)	(3,4)	(3,5)
(4,-5)	(4,-4)	(4,-3)	(4,-2)	(4,-1)	(4,0)	(4,1)	(4,2)	(4,3)	(4,4)	(4,5)
(5,-5)	(5,-4)	(5,-3)	(5,-2)	(5,-1)	(5,0)	(5,1)	(5,2)	(5,3)	(5,4)	(5,5)

Figura 43. Plantilla de vecindad de 11x11 (Moore)

Esta plantilla actúa como filtro para encontrar los vecinos válidos de la celda central (celda analizada), las coordenadas en cada una de las celdas vecinas indican la posición (renglón, columna) con respecto a la celda central, es decir, si se quiere calcular las vecindades para el pixel (200,200), se recorre toda la plantilla iniciando en la esquina superior izquierda con la celda (-5,-5), estos valores se suman al pixel por analizar obteniendo la primera posición de la celda vecina en el mapa (195,195), resultado de (200 - 5 ,200 - 5).

La distancia de la celda central a cualquiera de sus vecinas, está definida por la longitud Manhattan [78]. Observe (Figura 44) la cantidad de saltos necesarios para llegar a una de las celdas vecinas, partiendo desde la celda central, 3 para este caso.

10	9	8	7	6	5	6	7	8	9	10
9	8	7	6	5	4	5	6	7	8	9
8	7	6	5	4	3	4	5	6	7	8
7	6	5	4	3	2	3	4	5	6	7
6	5	4	3	2	1	2	3	4	5	6
5	4	3	2	1	0	1	2	3	4	5
6	5	4	3	2	1	2	3	4	5	6
7	6	5	4	3	2	3	4	5	6	7
8	7	6	5	4	3	4	5	6	7	8
9	8	7	6	5	4	5	6	7	8	9
10	9	8	7	6	5	6	7	8	9	10

Figura 44. Distancia Manhattan a celdas vecinas, desde la celda central
(Ventana de observación de 121 celdas)

La ubicación de las celdas que conforman la vecindad, es particularmente importante debido a que forman parte del parámetro necesario para las matrices de calibración.

5.2.2 Matrices de Calibración

Las matrices de calibración, ayudan a configurar la fuerza de atracción–repulsión que ejerce cada una de las celdas con sus vecinas. Si estas representan un uso urbano, entonces podemos modelar la atracción y repulsión entre los distintos usos de la zona de estudio. Tales fuerzas sufren además, del efecto de distancia (distance decay effect) [76], esta implica que, además de la situación de compatibilidad de usos contiguos, la distancia entre ellos puede alterar de manera positiva o negativa, la fortaleza de una atracción. Ejemplo de ello se observa con las zonas industriales y residenciales, a medida que estas últimas se alejan de las industriales, se motivan a florecer, sucede lo contrario al reducirse la distancia entre ellas.

La anatomía de una matriz de calibración (Figura 45), reglones corresponden a los usos (k) incluyendo los dinámicos y estáticos, las columnas corresponden a la distancia Manhattan (ver apartado 5.3).

	Distancias																	
K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1																		
2																		
3																		
4																		
5																		
6																		
7																		
8																		
9																		
10																		

Figura 45. Matriz de calibración

Formalmente, una matriz de calibración M_{kd} , donde k indica los usos totales (dinámicos y estáticos), y d indica distancia, está contemplada únicamente para el cálculo de los potenciales de transición de los usos dinámicos. Los usos estáticos no tienen matriz de calibración, debido a que son usos que en el modelo no se contemplan para cambio, sin embargo, si están contemplados dentro de la matriz de calibración, debido a que si tienen influencia sobre los usos dinámicos.

5.3 IMPLEMENTACIÓN EN OPENMP

La implementación de los cálculos de manera concurrente, en varios núcleos (Figura 46), se centra en dedicar un uso urbano a cada núcleo para que la generación de las simulaciones sea más ágil. La estrategia de paralelización con OpenMP consiste en compartir en memoria un único mapa para ejecutar el cálculo de cada uso urbano en un núcleo independiente en el procesador.

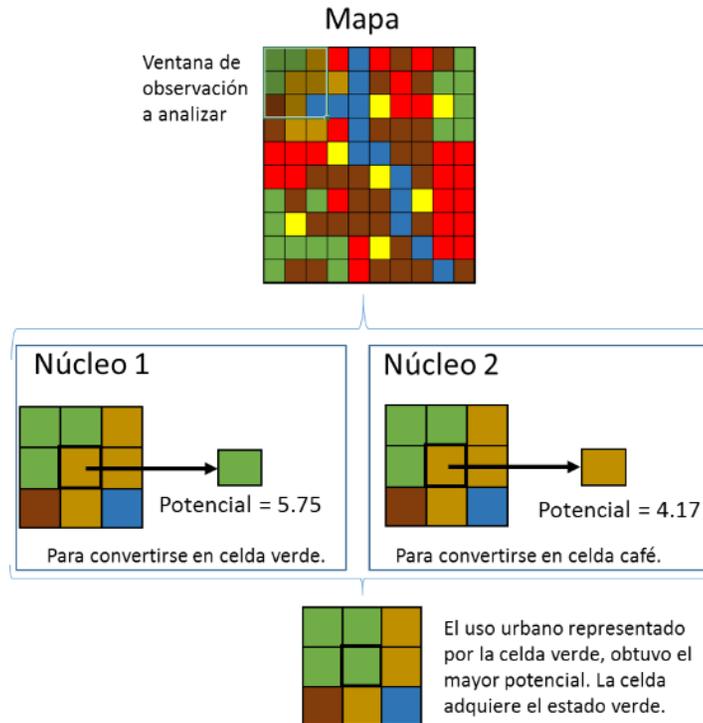


Figura 46. Calculo del potencial de transición (uno por nucleo) de una celda hacia dos usos posibles

Se identificó el cálculo del potencial de transición P_{hj} como la porción secuencial que podía paralelizarse para realizar las operaciones matemáticas pertinentes. Para n usos urbanos, la idea básica es que para cada celda A_{ij} se tiene que calcular su P_{hj} de su estado actual h al objetivo j , donde $j = n$.

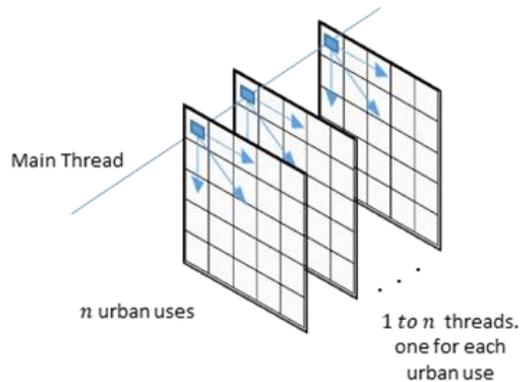


Figura 47. Calculo de potenciales de transición para n usos

El algoritmo usado para el cálculo de los potenciales de transición para cada uso dinámico, es el siguiente:

```
1: 0 to epoch;
2: (0,0) to mapSize (nxm);
   {
3: pixel = cell(nxm);
4: nCores = nDinamicUses;
5: omp_set_num_threads(nCores);
6: #pragma omp parallel
   {
7:     j = omp_get_thread_num();
8:     vectorP [j]=calculateP (pixel,j);
   }
9: newMap(nxm) = hPotential (vectorP);
   }
```

Se definen las épocas, se establece el ciclo para recorrer desde la primer celda hasta la última, Se obtiene el valor del pixel actual, se establece el número de núcleos del procesador a utilizar (tomando como base el número de usos dinámicos a calcular), se utiliza la directiva para establecer el número de hilos **omp_set_num_threads**, como parámetro se le envía el número de núcleos a ocupar, se inicializa la región paralela **#pragma omp parallel**, en ella se identifica el número de **hilo j = omp_get_thread_num()** y se le asigna al vector de potenciales el resultado del potencial de la celda analizada. Terminado el cálculo en todos los núcleos, se asigna el mayor potencial calculado a la celda analizada y se asigna a un nuevo mapa **nuevoMapa(nxm)= MayorPotencial(VectorPotenciales)**.

CAPÍTULO 6: DISEÑO DE LA RED NEURONAL

Siendo una RNA un potente clasificador y generalizador, el objetivo de su implementación al modelo de AC es para interpretar las variables socioeconómicas, y que pueda clasificarlas al asignarle un valor de clasificación hacia una determinada categoría. Con esto se robustece la simulación, aunque conlleva un serio incremento en los cálculos por pixel, la RNA viene a sustituir el parámetro de perturbación estocástica del modelo original de White (capítulo 2.1).

Sobre los elementos de la RNA, Giles [79], argumenta que el número de neuronas y el diseño total de una red neuronal dependerá de :

- El número de entradas y salidas de la red.
- El número de entrenamientos
- La cantidad de ruido en los datos.
- La complejidad de los datos a aprender.
- El tipo de función de activación de las neuronas.

Sin embargo, no existe un método unificado que permita determinar el mejor número de neuronas de una red [80], obligadamente se llevan a cabo experimentos para ver qué diseño es la que mejor generaliza y presenta un mínimo de error.

6.1 ESTRUCTURA DE LA RNA

El diseño final de la red es un perceptrón multicapa, el algoritmo que lo opera es el de retropropagación. La red esta constituida por 3 capas; $|S|$ neuronas para la capa de entrada, donde $S = \{x|x \text{ variables socioeconómicas}\}$, $|S| \times 2$ neuronas para la capa de oculta (siempre el doble que la capa de entrada), $|Y|$ neuronas para la capa oculta, donde $Y = \{x|x \text{ usos dinámicos}\}$, (

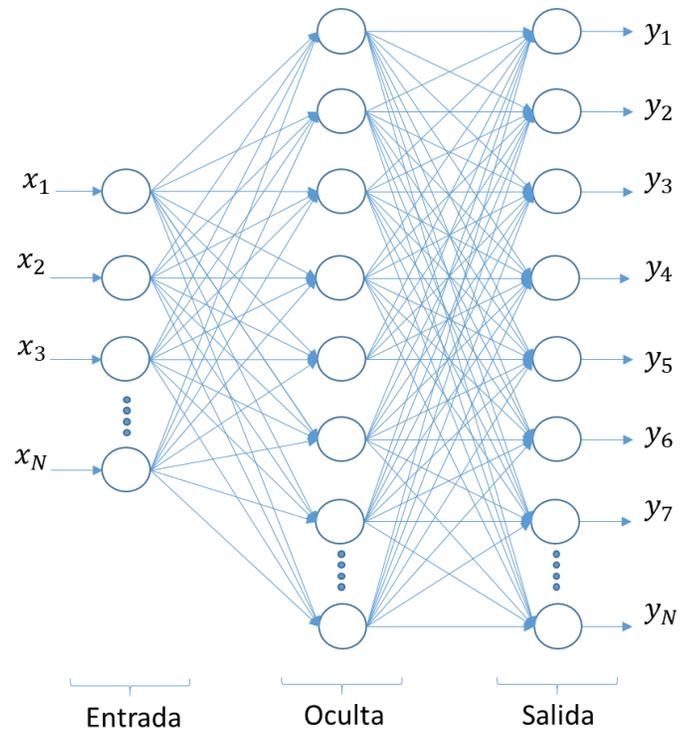


Figura 48

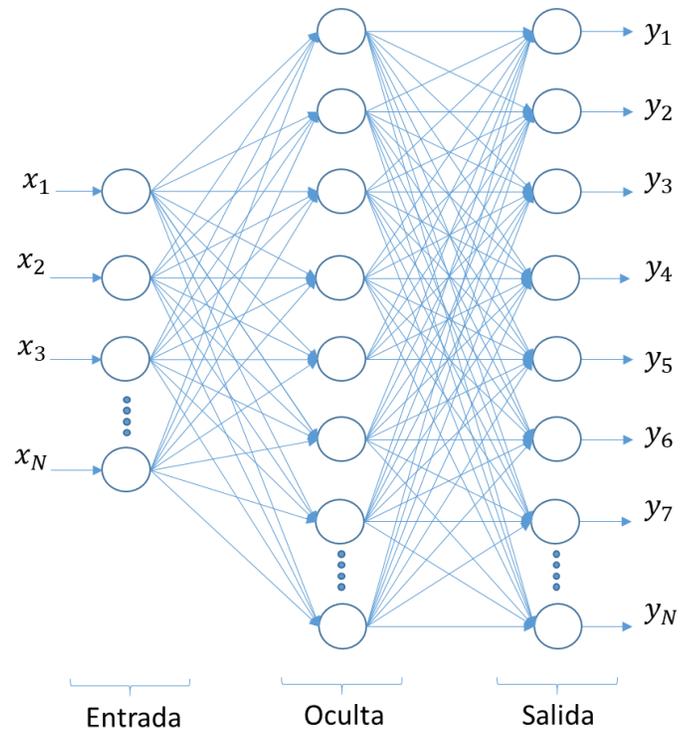


Figura 48).

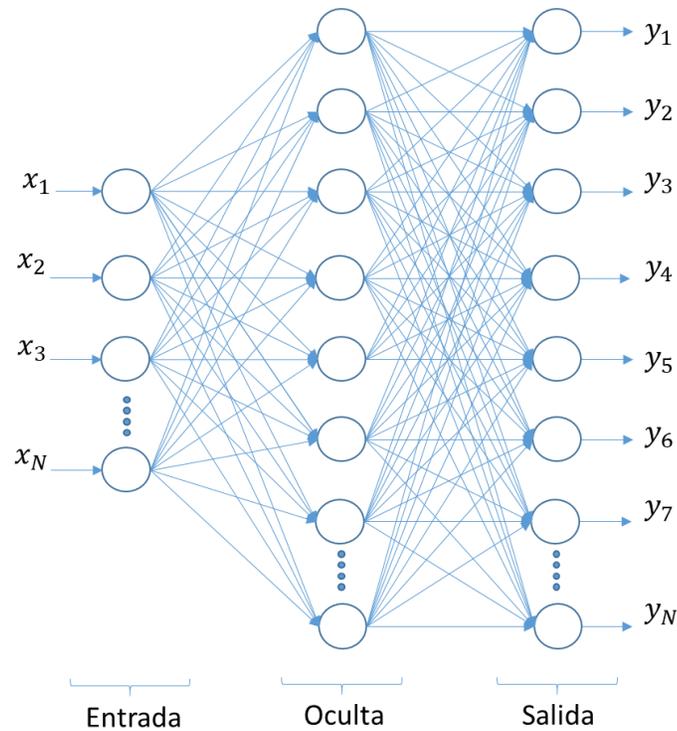


Figura 48. Arquitectura RNA para experimentos

Cabe señalar que no se optó por codificar la salida de la red a una forma binaria, es decir, usar menos neuronas para representar las respuestas (3 neuronas darían 2^3 respuestas), sino que la clasificación es 1 a 1 (una neurona de salida por cada uso dinámico).

6.1.2 Algoritmo de Entrenamiento

Para el entrenamiento de la red, se utilizó el algoritmo de retropropagación, el conjunto de entrenamiento consiste en de entradas-salidas proporcionados a la red como ejemplo. Un ciclo de propagación-adaptación de dos fases realiza lo siguiente:

Se procesa un patrón de entrada el cual afectará la primera capa de la red, esto se repetirá a lo largo de las siguientes capas hasta generar una salida, se compara el resultado obtenido contra el deseado y se calcula el error para cada neurona de la capa de salida.

Estos errores se transmiten hacia “atrás” partiendo desde la capa de salida hacia todas las capas de la red que hayan contribuido al cálculo de la salida. Estas neuronas recibirán el porcentaje del error según la participación de la neurona en el cálculo de la salida. Este proceso se repite capa por capa hasta que todas las neuronas hayan recibido su error correspondiente al error total. El error recibido servirá para reajustar el peso de las conexiones de la neurona, de esta manera, en el siguiente cálculo de los valores de entrada se obtendrá como salida un valor más aproximado a la salida deseada.

Teniendo una red como la mostrada en la Figura 49, el cálculo manual para la salida de un perceptrón multicapa usando el algoritmo de retropropagación queda como sigue[81]:

1.- Se asume que en todas las capas se hace uso de la función de activación sigmoidal:

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

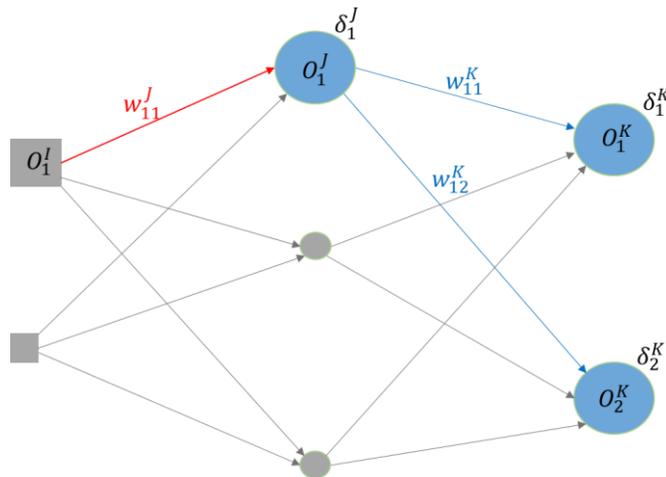


Figura 49. Estructura perceptrón de 3 capas: Entrada = I, Oculta = J, Salida = K.

2.- Se calculan las diferencias (δ 's) en la capa de salida (K).

Para las neuronas de salida δ_1^K y δ_2^K el cálculo es similar.

Suponga que la salida deseada está dada por t_1

Entonces, para δ_1^K : $\delta_1^K = O_1^K(1 - O_1^K)(O_1^K - t_1)$

$$\underbrace{\hspace{10em}}_{\text{Derivada de } \sigma}$$

3.- Se actualiza el peso de la conexión w_{11}^K :

$$\Delta w_{11}^K = -n\delta_1^K O_1^J$$

4.- Se procesa la diferencia de la neurona O_1^J :

$$\delta_1^J = \underbrace{O_1^J(1 - O_1^J)}_{\text{Derivada de } \sigma} \underbrace{(w_{11}^K \delta_1^K + w_{12}^K \delta_2^K)}_{\sum_{n=1,2} w_{1n}^K \delta_n^K}$$

5.- Finalmente se actualiza el peso de la conexión w_{11}^J :

$$\Delta w_{11}^J = -n\delta_1^J O_1^I$$

El algoritmo de retropropagación siempre busca encontrar un valor mínimo de error, mediante el ajuste a pasos de los pesos (gradiente descendente), esto lleva a realizar un descenso por la superficie del error hasta encontrar un valle, que podría ser en algunos casos un mínimo local. Por lo tanto, uno de los problemas de éste algoritmo de entrenamiento es que al minimizar el error se pueda caer en un mínimo local y no se salga de ahí, con lo cual no se llegaría a encontrar un mínimo global. Sin embargo, se tiene en cuenta que no para todos los casos es necesario alcanzar el mínimo global, sino que es más que suficiente alcanzar un mínimo de error preestablecido.

6.2 ENTRENAMIENTO

Los datos de entrada para el entrenamiento consisten en los mapas de las variables socioeconómicas espacializadas (Tabla 5), y definidas en el capítulo 6.3.5. El conjunto de los datos destinados al entrenamiento representó el 10% del mapa, y se seleccionaron de manera aleatoria.

Tabla 5. Variables socioeconómicas

Variable	Tipo
----------	------

Sexo	Discreta
Edad	Discreta
Escolaridad	Discreta
Nivel de Ingresos	Continua

CAPÍTULO 7: DISEÑO DEL AGENTE COGNITIVO

El diseño de un sistema multiagente aplicado a un caso específico, implica el entendimiento previo de los cambios e interacciones que suceden en el medio ambiente de estudio. Una vez identificada esta dinámica, es posible iniciar con el diseño de una solución que mimetice los aspectos relevantes del sistema a modelar.

7.1 BDI

La arquitectura BDI tiene sus raíces en el denominado Razonamiento Práctico, este tipo de razonamiento puede describirse como el proceso de decidir qué acción realizar para alcanzar las metas, e involucra dos importantes procesos: (a) Deliberación: esto es decidir qué metas se quieren alcanzar, y (b) Razonamiento de medios-fines (means-ends): decidir cómo se alcanzarán las metas [82].

Como se puede observar (Figura 50), la lógica BDI se basa en [83]:

- **CREENCIAS (BELIEFS):** A cada situación se le asocia un conjunto de “beliefs-accessible Worlds” (mundos que el agente cree posibles).
- **DESEOS (GOALS):** Cada situación dispone del conjunto de “goals-accessible worlds” para representar los objetivos del agente. Goals son deseos elegidos que sean consistentes.
- **INTENCIONES (INTENTIONS):** Representados por mundos que el agente se ha comprometido a intentar alcanzar.



Figura 50. Lógica BDI

7.2 GAMA Y GAML

GAMA es un entorno de desarrollo de modelado y simulación para la construcción y definición de simulaciones con agentes explícitamente espaciales. Parte de las bondades que este entorno proporciona son [84].

- Alta conectividad a los GIS actuales.
- Versatilidad en las simulaciones.
- Lenguaje integrado GAML

Desarrollado en Java, GAMA es accesible a través de una interfaz de programación de aplicaciones (API por sus siglas en inglés). Para poder usar la plataforma GAMA y su lenguaje de modelado GAML, es necesario instalar las siguientes bibliotecas:

- `msi.gama.core`: Encapsula el núcleo de las funciones de modelado y simulación que ofrece la plataforma: tiempo de ejecución, simulación, metamodelo, estructuras de datos, núcleo de simulación, programación, etc. Contiene 2 paquetes principales:
 - `msi.gama`
 - `msi.gaml`, define el lenguaje de modelado GAML: palabras clave, operadores, declaraciones, especies, habilidades.
- `msi.gama.application`: Describe la interfaz gráfica de usuario (paquete `msi.gama.gui`). Este proyecto también contiene el archivo `gama1.7.Eclipse3_8_2.product`. Contiene los siguientes subpaquetes :
 - `msi.gama.gui.displays`
 - `msi.gama.gui.navigator`
 - `msi.gama.gui.parameters`
 - `msi.gama.gui.swt`
 - `msi.gama.gui.views`
 - `msi.gama.gui.wizards`
 - `msi.gama.gui.viewers`

- `msi.gama.ext`: Reúne todas las bibliotecas externas en las que se basa GAMA.
 - `msi.gama.lang.gaml`: Contiene el archivo `gaml.xtext` que define la gramática GAML
 - `msi.gama.lang.gaml.ui`: Contiene el editor GAML.
- `msi.gama.processor`: Es responsable de procesar las anotaciones realizadas en el código fuente de Java y producir adiciones a GAML (Java, propiedades y archivos de documentación), que se agregan en un paquete fuente llamado "gaml.additions". Estas adiciones se cargan automáticamente cuando se lanza GAMA, lo que permite que las extensiones realizadas por los desarrolladores en otros complementos sean reconocidas cuando su complemento se agrega a la plataforma.
- `ummisco.gaml.editbox`: Proyecto utilizado para definir los cuadros de edición en la interfaz de usuario.

La dependencia (Figura 51) de las bibliotecas para el funcionamiento adecuado del entorno GAMA y del lenguaje de modelado GAML se observa a continuación.

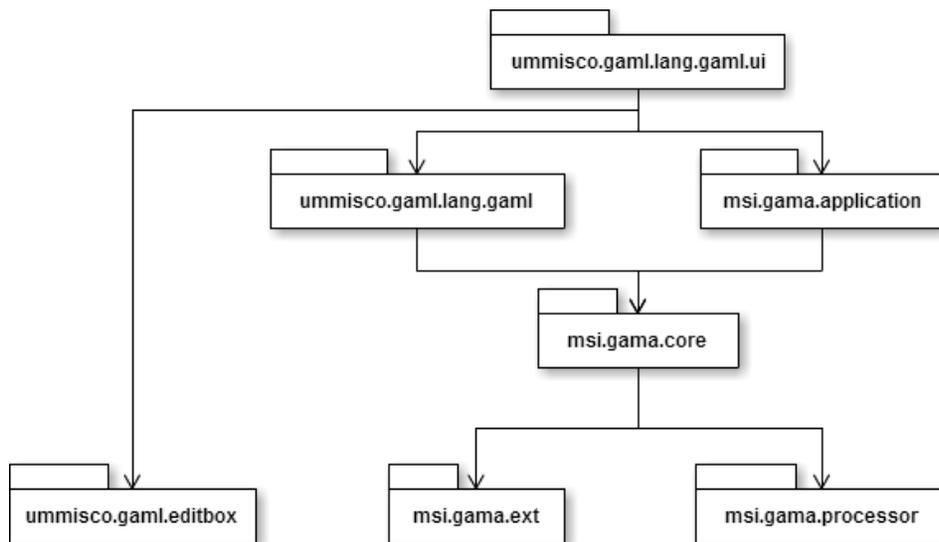


Figura 51. Configuraciones mínimas de dependencias de proyectos

PARTE IV EXPERIMENTACIÓN

CAPÍTULO 8: RESULTADOS

Los primeros resultados, corresponden a diversos mapas que reflejan las dinámicas urbanas de la zona de estudio en 2 momentos temporales (1997 y 2004). A través de estos mapas, fue posible la caracterización del autómata celular para representar el paisaje urbano, y con ello generar las simulaciones prospectivas que, corresponden al segundo grupo de resultados. La implementación de la red neuronal al autómata celular, generó el tercer grupo de resultados, son mapas que presentan un cierto grado de mejora con respecto al AC, pero que presentan una mayor complejidad de implementar. Por último, se presenta una propuesta de modelo de agentes cognitivos, que contempla los aspectos observados durante la presente investigación.

8.1 ANÁLISIS DE CAMBIOS DE USO DE SUELO URBANO

Partiendo de los mapas ráster generados para la zona de estudio (Figura 40), se llevó a cabo el análisis para identificar los patrones de cambio, con base a ellos se determinaron las matrices de transición del modelo, además se identificaron los usos urbanos con mayor relevancia y tendencias de cambio. Con la herramienta CrossTab de Idrisi, se obtuvieron las matrices de tabulación cruzada, con esto se analizaron las pérdidas y ganancias de cada uso urbano. Posteriormente con el módulo Land Change Modeler (LCM) de Idrisi, se observaron los mapas de pérdidas y ganancias para cada una de las categorías. Además, a través de una regresión

logística espacial, se analizó el grado de influencia que tienen ciertos factores con respecto al crecimiento.

8.1.1 Cuantitativo

Los valores representan el número de pixeles, que para nuestro caso, cada uno equivale a 50 *mts*². Es necesario manejar las pérdidas y ganancias de esta manera, debido a que el píxel es la unidad mínima de información para el diseño del sistema. El primer periodo de análisis corresponde a los años 1997 y 2004 (Figura 52), el segundo periodo del 2004 a 2011(Figura 53), ambos muestran un rango de 7 años de evolución de la zona urbana, la tabulación cruzada detalla las pérdidas y ganancias entre ambos periodos.

		2004										Total
		Huecos	Servicios	Agrícola	Comercial	Forestal	Industrial	Cuerpos de Agua	Vivienda	Equipamiento	No Urbanizable	Pérdidas
		1	2	3	4	5	6	7	8	9	10	
1997	Huecos	1104	413	342	295	566	179	73	2209	28	0	5209
	Servicios	39	1132	67	462	110	67	5	596	362	0	2840
	Agrícola	585	1028	37925	226	2155	646	61	613	0	268	43507
	Comercial	6	1	0	179	12	1	0	82	0	0	281
	Forestal	417	299	10392	949	38678	201	290	7404	159	167	58956
	Industrial	49	28	41	25	244	347	4	70	0	0	808
	Cuerpos de Agua	1	0	41	0	103	7	1628	29	9	1	1819
	Vivienda	199	680	79	1259	27	229	31	23294	119	7	25924
	Equipamiento	4	87	0	8	5	0	0	28	555	11	698
	No Urbanizable	61	0	0	0	0	0	0	13	0	137	211
Total Ganancias		2465	3668	48887	3403	41900	1677	2092	34338	1232	591	140253
Resumen de Pérdidas y Ganancias para el periodo 1997 - 2004												
Ganancias		1361	2536	10962	3224	3222	1330	464	11044	677	591	35411
Persistencias		1104	1132	37925	179	38678	347	1628	23294	555	137	104979
Pérdidas		4105	1708	5582	102	20278	461	191	2630	143	74	35274

Figura 52. Tabulación cruzada del periodo 1997 a 2004

De la matriz de tabulación podemos recoger lo siguiente:

Tabla 6. Cambios en el periodo 1997 a 2004

Categoría	Ganancia	Pérdida	Persistencia	Cambios Netos	Cambios Totales
Servicios	2536	1708	1132	828	4244
Agrícola	10962	5582	37925	5380	16544

Comercial	3224	102	179	3122	3326
Forestal	3222	20278	38678	17056	23500
Industrial	1330	461	347	869	1791
Vivienda	11044	2630	23294	8414	13674
Totales	32318	30761	101555	35669	63079

		2011										Total Pérdidas
		Huecos 1	Servicios 2	Agrícola 3	Comercial 4	Forestal 5	Industrial 6	Cuerpos de Agua 7	Vivienda 8	Equipamiento 9	No Urbanizable 10	
2004	Huecos	1235	37	218	139	31	64	3	586	152	0	2465
	Servicios	244	2710	9	82	31	19	1	286	283	3	3668
	Agrícola	845	322	36677	147	6874	566	70	3120	265	1	48887
	Comercial	60	11	15	3102	37	24	0	149	5	0	3403
	Forestal	377	223	6440	145	33186	405	54	994	76	0	41900
	Industrial	98	42	43	35	39	1374	1	38	7	0	1677
	Cuerpos de Agua	0	6	60	0	277	7	1638	100	1	3	2092
	Vivienda	325	168	726	241	696	78	21	31973	108	2	34338
	Equipamiento	1	142	0	3	63	1	6	64	952	0	1232
	No Urbanizable	0	0	65	17	4	0	2	10	8	485	591
Total Ganancias		3185	3661	44253	3911	41238	2538	1796	37320	1857	494	140253
Resumen de Pérdidas y Ganancias para el periodo 2004 - 2011												
Ganancias		1950	951	7576	809	8052	1164	158	5347	905	494	27406
Persistencias		1235	2710	36677	3102	33186	1374	1638	31973	952	485	113332
Pérdidas		1230	958	12210	301	8714	303	454	2365	280	106	26921

Figura 53. Tabulación cruzada del periodo 2004 a 2011

De la matriz de tabulación resultante, podemos recoger lo siguiente:

Tabla 7. Cambios en el periodo 2004 a 2011

Categoría	Ganancia	Pérdida	Persistencia	Cambios Netos	Cambios Totales
Servicios	951	958	2710	7	1909
Agrícola	7576	12210	36677	4634	19786
Comercial	809	301	3102	508	1110
Forestal	8052	8714	33186	662	16766
Industrial	1164	303	1374	861	1467
Vivienda	5347	2365	31973	2982	7712
Totales	23899	24851	109022	9654	48750

8.1.2 Espacial

Se hace uso de la opción land change modeler de Idrisi para generar los mapas de pérdidas y ganancias para cada uno de los usos dinámicos. Es importante destacar las persistencias, debido al componente de inercial del modelo de simulación, además de las tendencias de cambio, para modelar y delimitar el desplazamiento espacial del agente cognitivo.

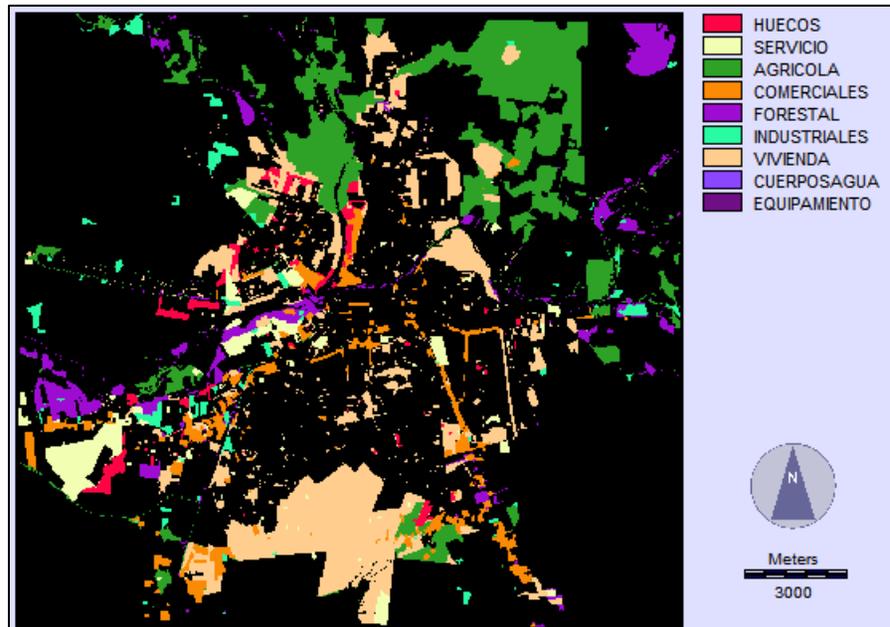


Figura 54. Mapa de cambios de 1997 al 2004

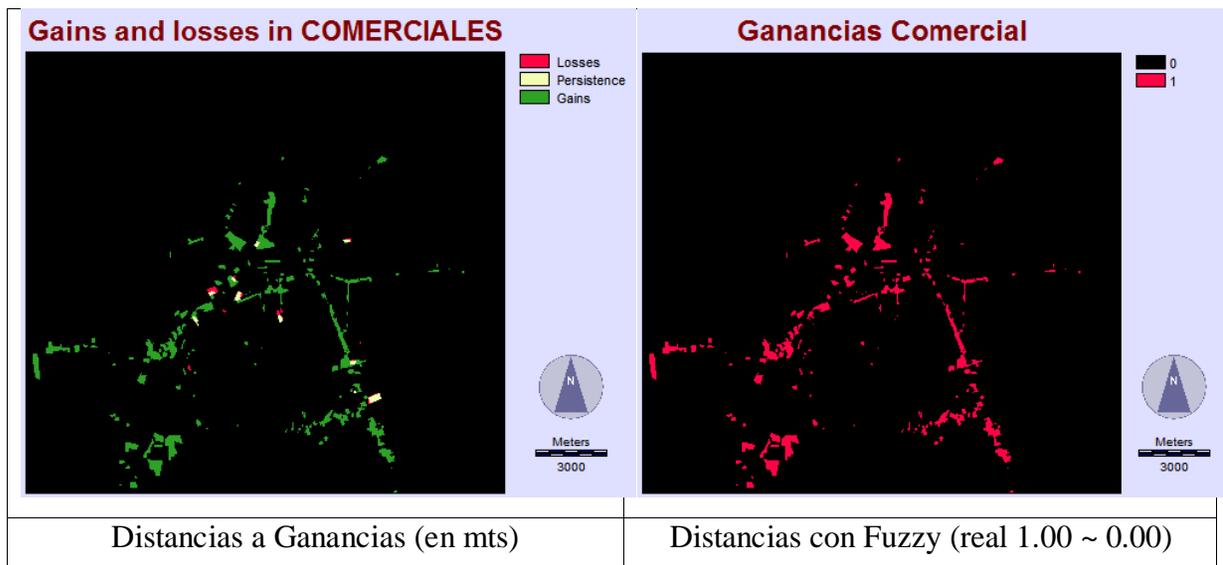
Mapas de Aptitud, Proximidad y Accesibilidad

La aptitud se entiende como el grado de idoneidad que presenta un territorio, para albergar una actividad o un fenómeno. Se utilizan distintas características del terreno para generar mapas que permitan identificar las zonas clave donde desarrollar las actividades requeridas.

Para la elaboración de los mapas de aptitud, se combinaron los mapas de pendientes (en grados) y la distancia a las ganancias por cada uso. Se usó el comando Fuzzy para las distancias, con los parámetros:

- Lineal.
- Real (0.00 a 1.00).
- Decreciente.
- Puntos de control, valores mínimos y máximos en distancias.

Después se usó el comando Overlay para multiplicar el mapa resultante del comando Fuzzy con el mapa de pendientes. A continuación (Figura 55) se muestra la operación para el uso comercial.



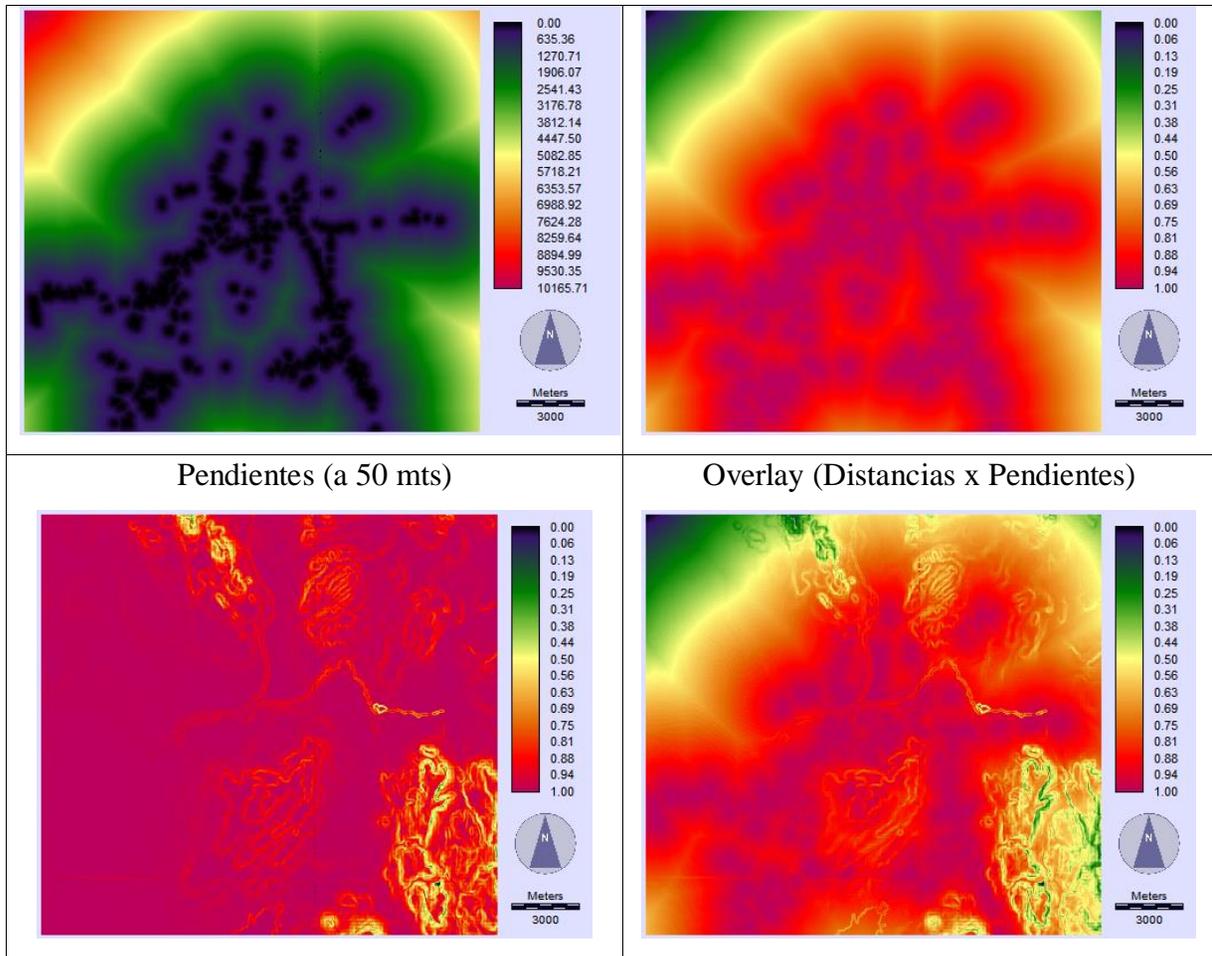


Figura 55. Calculo de la aptitud Comercial

A continuación se muestran los mapas de aptitud restantes:

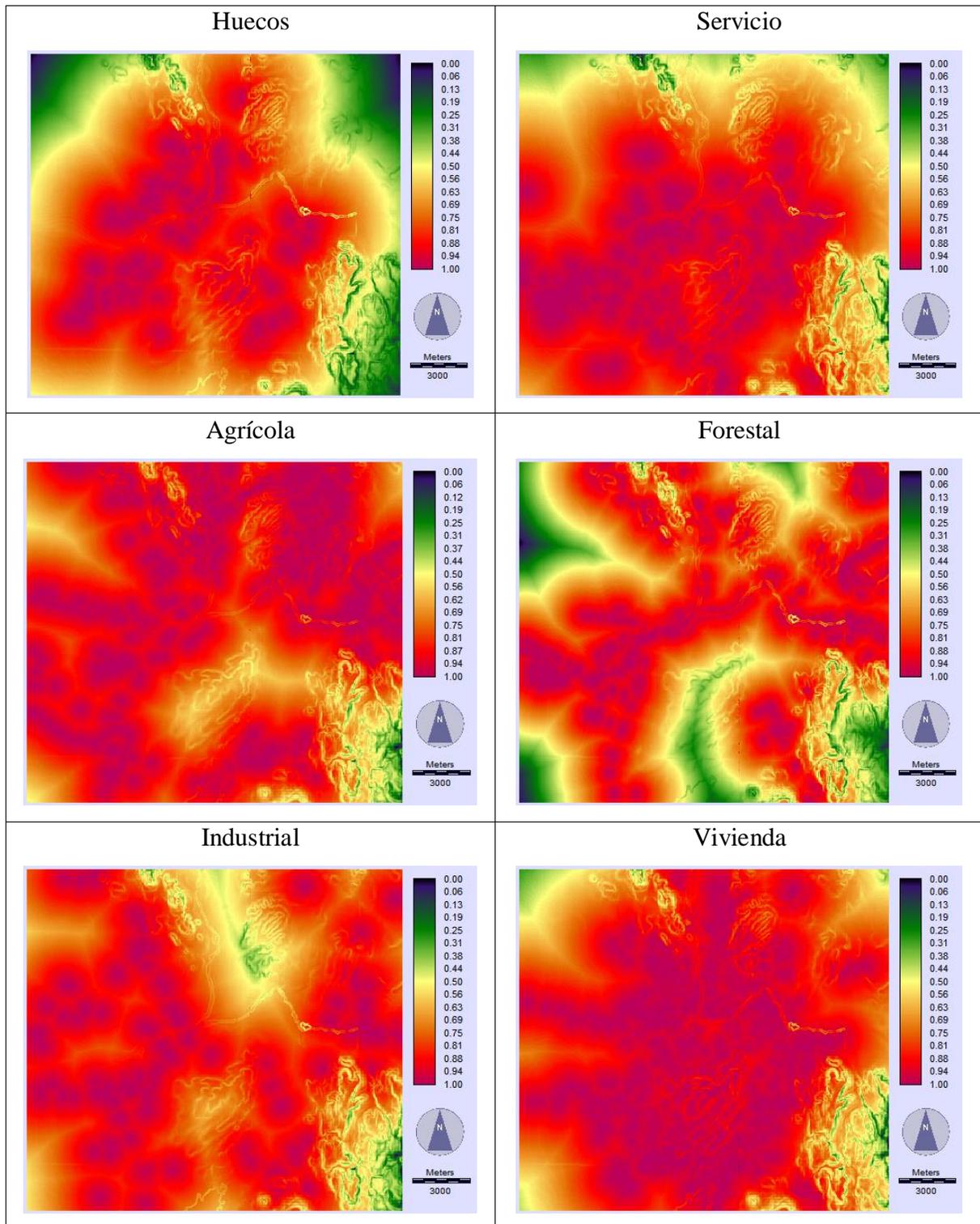


Figura 56. Mapas de aptitud para distintos usos urbanos

Cabe señalar que para el mapa de aptitud para las zonas residenciales se obtuvo combinando 3 mapas; pendientes del terreno, las ganancias de la vivienda, y como caso especial, el de prioridades de urbanización.

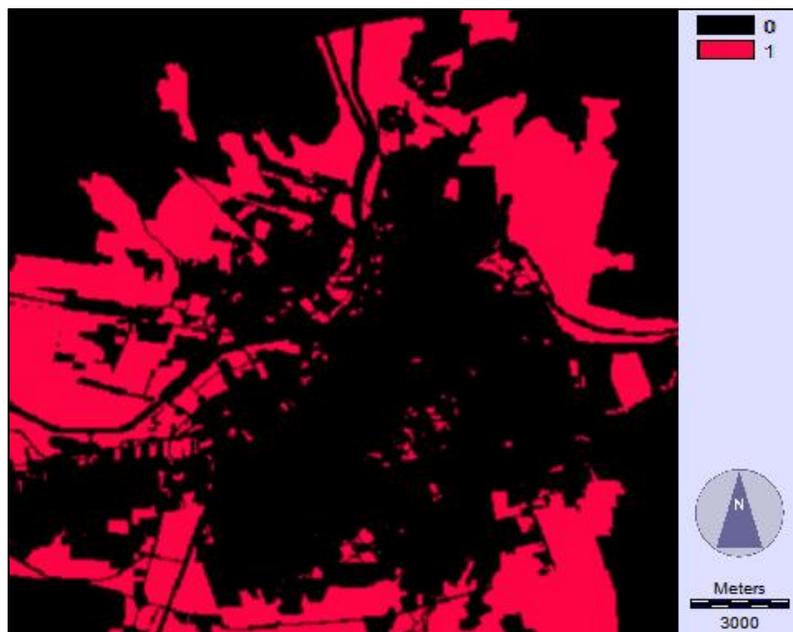


Figura 57. Prioridades de urbanización

Proximidad a Carreteras y Accesibilidad a Núcleos Urbanos

La distancia a carreteras (Figura 58) muestra la distancia euclidiana, categorizada de manera lineal a un rango de 0.0 ~ 1.0 de un determinado pixel, a la carretera más cercana. El mapa de accesibilidad, muestra una combinación de la distancia a carreteras con la proximidad a zonas urbanas de interés (para este caso usamos equipamientos). El objetivo de estos mapas es la de proveer mayor información durante los análisis de cambios, se ha comprobado que tienen relevancia como factores de crecimiento urbano [85],[27].

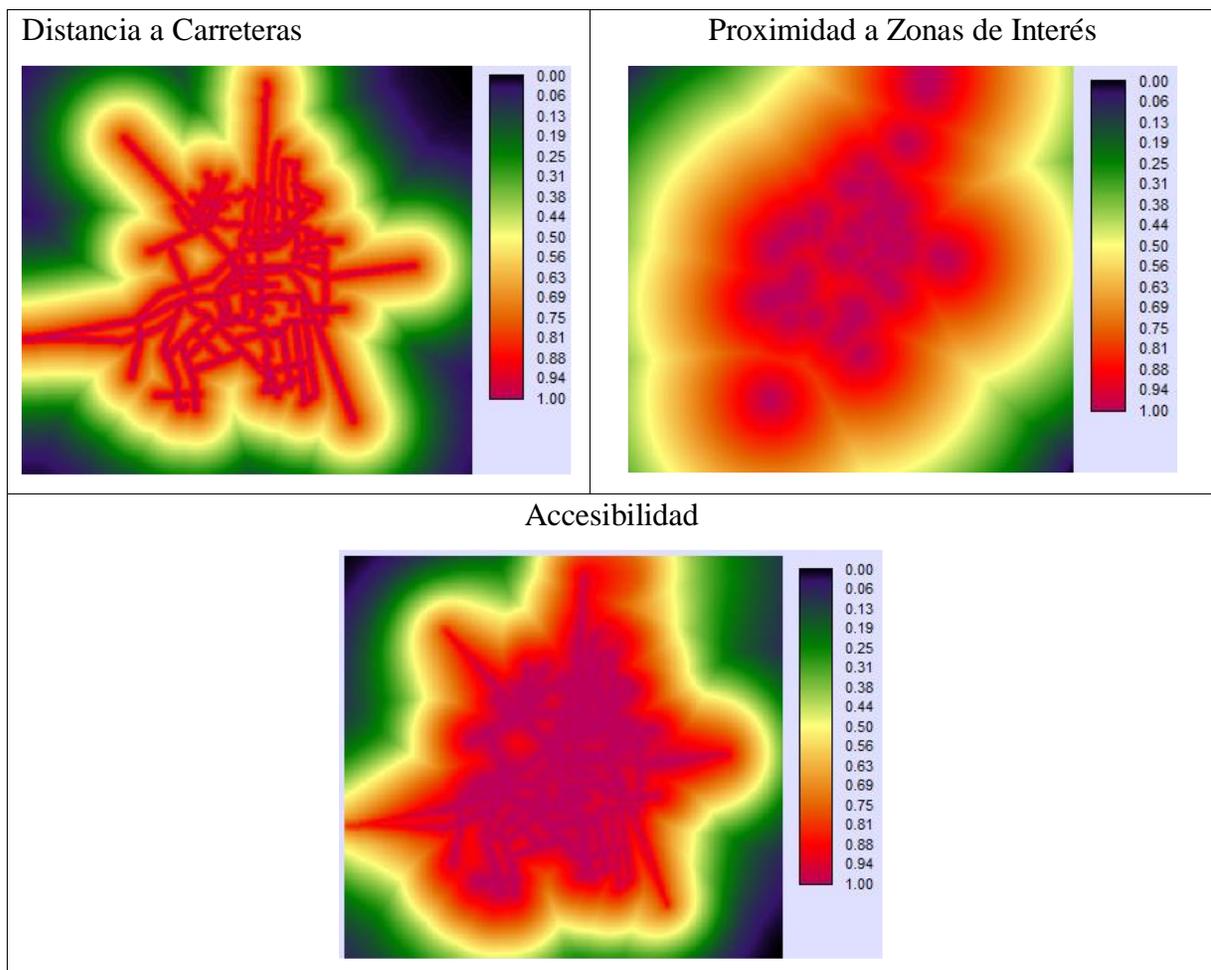


Figura 58. Mapas de distancia a carreteras, proximidad a zonas de interés y accesibilidad

8.1.3 Selección de los factores de influencia

Para definir un modelo de simulación es necesario identificar el grado de influencia que presentan ciertos factores con respecto al crecimiento urbano [76]. Quedaron fuera de los alcances de esta investigación, el proceso de exploración de todos los factores posibles que podrían explicar el crecimiento urbano, se definió a través de otras investigaciones

[27],[86],[87], la selección de un conjunto de variables sencillas, fácilmente cartografiables. De esta forma, estos análisis pueden ser fácilmente reproducibles y aplicables a otras áreas [27].

Los factores seleccionados:

Proximidad: Representa la distancia a las zonas urbanas de interés, proyecta una mayor probabilidad de desarrollo alrededor de áreas urbanizadas ya existentes. ha sido ampliamente usada en simulaciones de ocupación urbana [88][27].

Distancia a carreteras: Las calles y avenidas importantes, determinan y detonan el desarrollo urbano en cualquier categoría, es por ello que la cercanía a estas marcan el paso en el crecimiento de la ciudad.

Accesibilidad: El mapa de accesibilidad modela los tiempos aproximados de traslado a las zonas urbanas, su estudio para esta investigación se limita a la combinación de la distancia a carreteras con la proximidad a las zonas urbanas de interés.

Las pendientes y orientaciones como factores ambientales pueden ser entendidas factores condicionantes del crecimiento, ya que pueden limitar el desarrollo de áreas urbanas en algunas de las zonas del ámbito de estudio.

Prioridades de Urbanización mencionadas en el plan director de desarrollo, condicionan las reglas de crecimiento para las zonas urbanas. Se generó un mapa de prioridades combinado con el de aptitud para su experimentación y evaluación.

Para comprobar que los factores seleccionados, sean pertinentes para la presente investigación, se utilizó una herramienta de regresión logística espacial, LogisticReg (ver apartado 6.2.4).

El grado de influencia de las variables seleccionadas con respecto al crecimiento urbano de 1997 a 2004 es el siguiente:

Tabla 8. Resultados ROC

Factor	Valor
Proximidad	0.797
Distancia a carreteras	0.712
Accesibilidad	0.697
Prioridades	0.628
Pendientes	0.576

Se muestra el grado de correlación (Tabla 8) de las variables con respecto al crecimiento real, entre las fechas de análisis. Estos valores permiten determinar, qué factores podemos usar como punto de partida en las simulaciones.

8.1.4 Variables de Demanda

Es necesario tener en consideración las ganancias de superficie anual de la zona de estudio por cada uso de suelo urbano, esto permite que el modelo ajuste en cada iteración la cantidad a asignar de un uso de suelo determinado. Dado que el sistema implementado no realiza un cálculo automático de demanda, es necesario introducirlo de manera manual. Estos valores son calculables a través de un análisis de cambio entre dos fechas para calcular la demanda de superficie para cada uso urbano.

En el estudio cuantitativo (ver sección 8.1.1) podemos ver las matrices de cambios, sólo resta señalar que el análisis presenta cambios a lo largo de 7 años, con esto podemos calcular las demandas anuales e introducirlas de manera manual en el modelo.

8.1.5 Espacialización de Datos Socioeconómicos

Las variables socioeconómicas más representativas, se seleccionaron con base a los trabajos [89], [90]. En donde se detallan que las variables socioeconómicas que aportan más información sobre tendencias de desarrollo urbano son: edad, escolaridad, sexo y nivel de ingresos. La

entidad que provee tal información para el territorio nacional mexicano es el área geoestadística básica (AGEB).

La información de cada AGEB fue recolectada de INEGI, sin embargo, la información que proveen es un archivo tabular, por lo tanto se tuvo que realizar la espacialización de los datos (Figura 59). Usando la opción de Joins and Relates (en ArcGis), se establece una relación entre el archivo con los datos tabulados y el shape que contiene la información de las manzanas de la zona urbana.

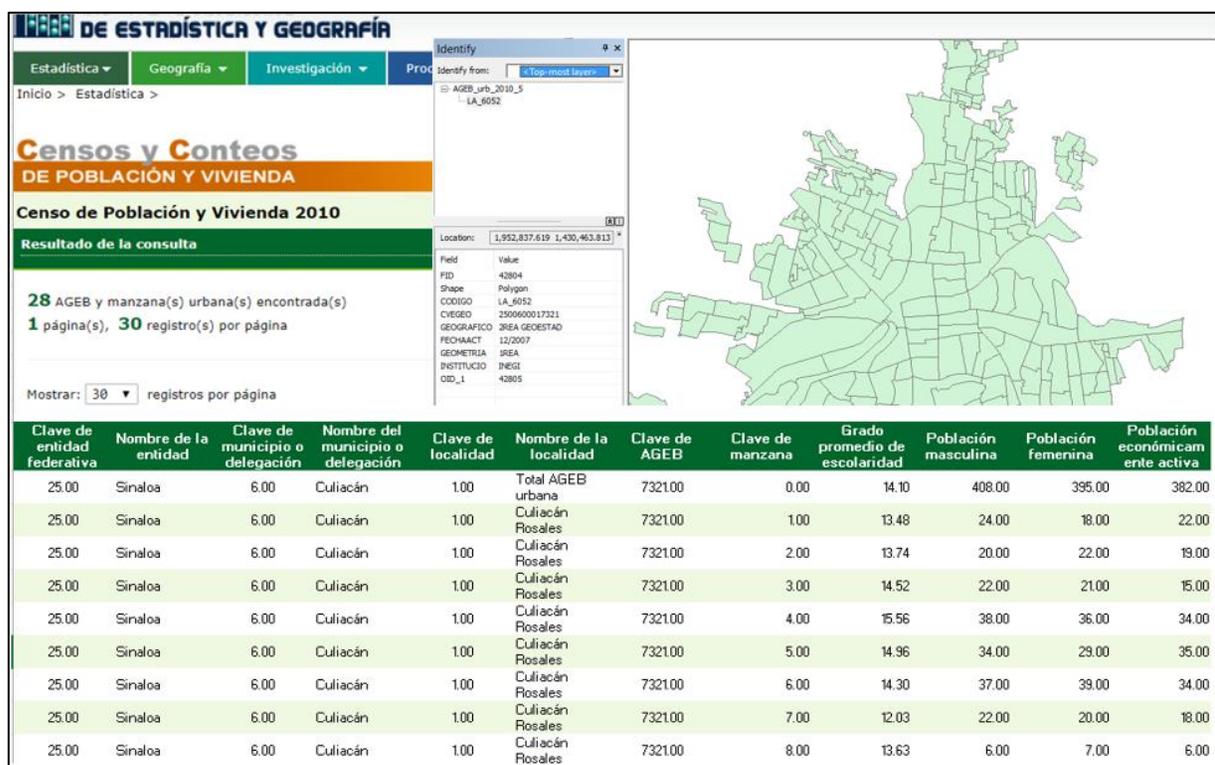


Figura 59. Espacialización de datos a partir de datos tabulados

Las variables socioeconómicas espacializadas (

Hombres	Mujeres
---------	---------

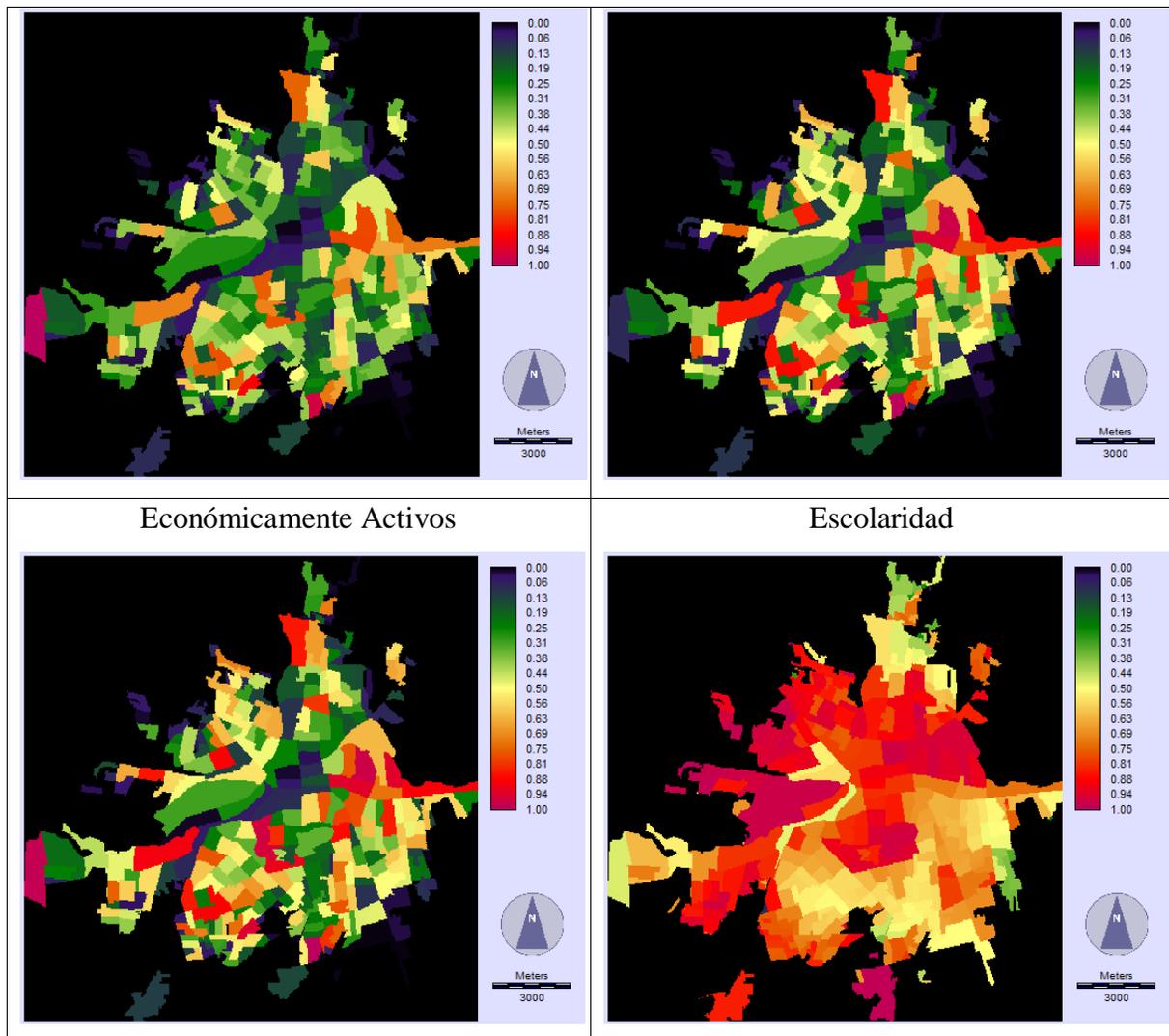


Figura 60), están normalizadas con el comando Fuzzy; configuración: sigmoideal, real (rango 0.0 ~ 1.0), ascendente. Los valores cercanos a 1, son aquellos con la mayor representación en su categoría, por ejemplo, las zonas cercanas a 1 en el mapa de escolaridad, significan que poseen un mayor grado de escolarización con respecto a otras zonas; cercanas a 0, viceversa.

Hombres	Mujeres
---------	---------

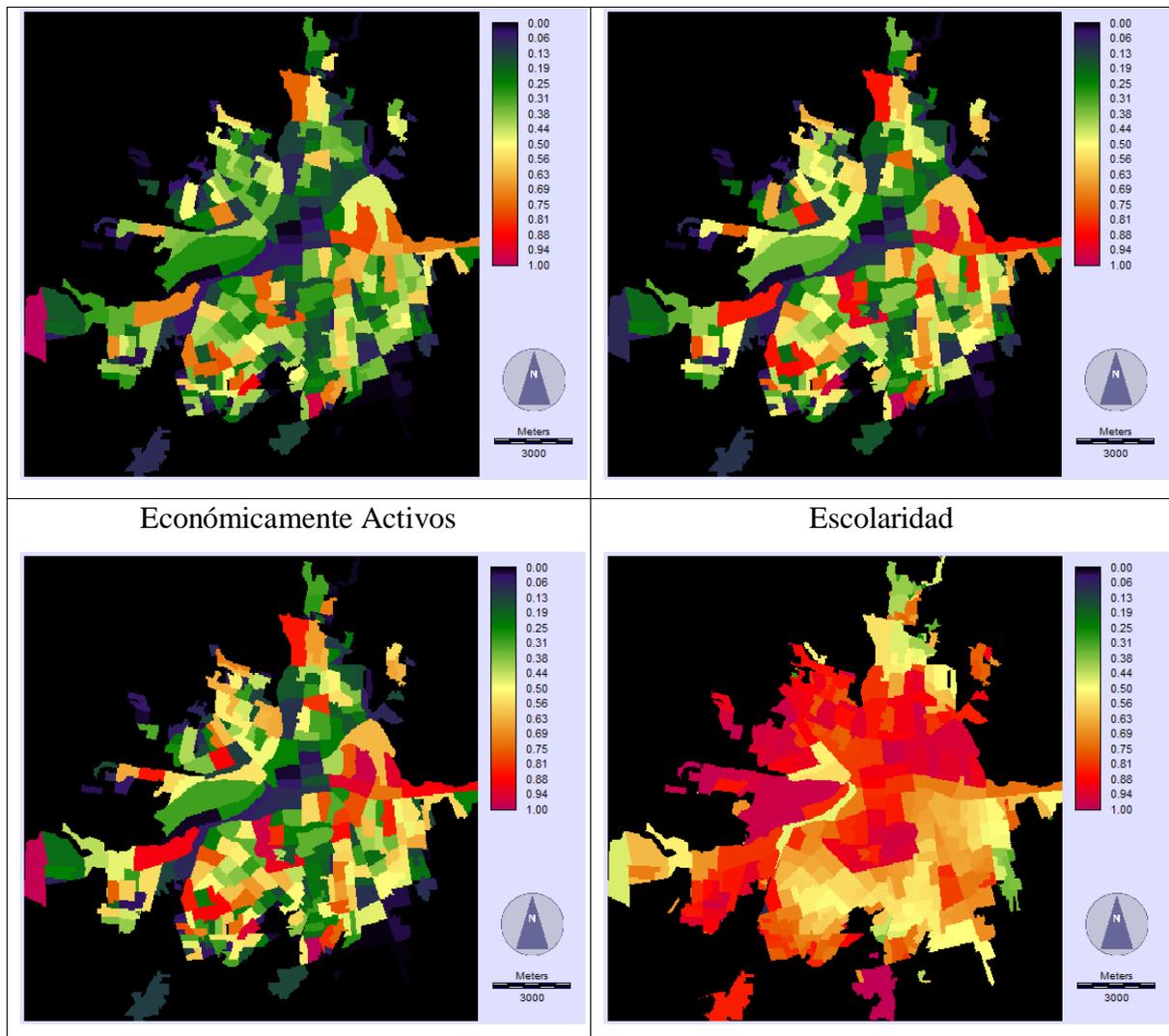


Figura 60. Variables socioeconómicas espacializadas

8.2 SIMULACIONES CON EL AUTÓMATA CELULAR

La primera simulación ex post (entre 2 fechas históricas) de 1997 a 2004, tiene como espacio inicial de evolución, a la malla bidimensional representada por el mapa ráster de 1997 a 7 (1 época por año, de 1997 a 2004). En este espacio, se realizaron los cálculos de los

de transición mencionados (ver apartado 7.1). Los resultados de esta primera simulación se observan en la

Figura 61, además se incluyen las matrices de calibración correspondientes. El resultado de la comparación del mapa real del 2004 versus el simulado al 2004, muestran una coincidencia de moderada a baja según los índices estadísticos Cramer's y Kappa.

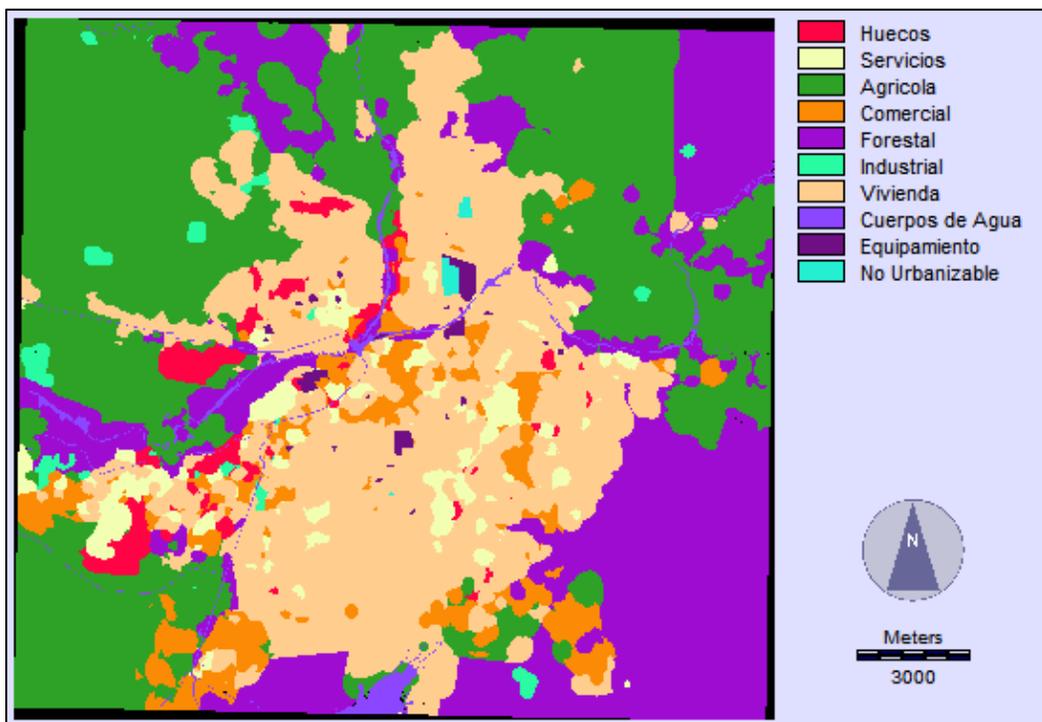


Figura 61. Simulación ex post de 1997 a 2004 con el AC

Cramer's V = 0.5622 Kappa = 0.6589

Las variables de entrada al sistema (ver Tabla 9) se dividen por **categoría, tipo y aplicación**. La primera columna son los distintos mapas de entrada, la segunda describe si es una variable que cambia o no (dinámica | estática) y la última columna, se refiere al cálculo del potencial en el modelo de AC (ver apartado 7.2).

Tabla 9. Datos de entrada al AC

Categoría	Tipo	Aplicación
Huecos	dinámico	vecindad
Servicios	dinámico	vecindad
Agrícola	dinámico	vecindad
Comercial	dinámico	vecindad
Forestal	dinámico	vecindad
Industrial	dinámico	vecindad
Vivienda	dinámico	vecindad
Cuerpos de agua	estático	vecindad
Equipamiento	estático	vecindad
No Urbanizable	estático	vecindad
Aptitud	n/a	ponderación
accesibilidad	n/a	ponderación

8.2.1 Matrices de Calibración Obtenidas

Las matrices fueron ajustadas de manera manual, existen distintos trabajos calibrarlas de manera automática [91], [92]. Sin embargo, con el propósito de examinar a detalle las interacciones que presentaba cada uso, se optó por realizar la calibración de manera manual. A continuación se muestran las matrices con los valores finales. Los usos referidos por K se observan en la Tabla 10. Valores K para las matrices, además se menciona de nuevo que, las distancias corresponden a la cantidad de saltos necesarios para llegar de la celda central a su vecina (ver apartado 7.4).

Tabla 10. Valores K para las matrices

Valor K	Categoría
1	Huecos
2	Servicios
3	Agrícola
4	Comercial
5	Forestal
6	Industrial
7	Vivienda

8	Cuerpos de agua
9	Equipamiento
10	No Urbanizable

Los análisis de pérdidas y ganancias, son el punto de partida para el ajuste de las matrices de calibración, en el análisis de 1997 – 2004 (Figura 52), puede observarse que las viviendas son las que más aprovecharon los huecos urbanos, sin embargo, aunque se tenga la certeza de esto, la calibración no siempre es un proceso lineal. Son tantas las interacciones que existen, que pueden suscitarse comportamientos emergentes, es decir, que los estados del autómatas pudieran tomar un camino no contemplado. Pero precisamente de eso se trata, de evitar al máximo ese comportamiento emergente que caracteriza a los AC, y se puede lograr, mientras se reúnan los elementos para afinar y hasta cierto punto condicionar, la regla de decisión del modelo.

En ese orden de ideas, y con respecto a la siguiente matriz, pensar en asignar un valor muy elevado a $k = 7$ “vivienda” (según lo indica la Tabla 10), podría ser lo primero y más lógico.

Distancias																		
K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
2	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
3	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
4	10	5	0	-5	-10	-15	-20	-25	-30	-35	-40	-45	-50	-55	-60	-65	-70	-75
5	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
6	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
7	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
8	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
9	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
10	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3

Sin embargo, en las primeras simulaciones (Figura 62), vivienda no solo poblaba todos los huecos, sino que se propagaba por todo el mapa. Se puede observar, como la vivienda no deja un hueco disponible, ni da oportunidad a los demás usos.

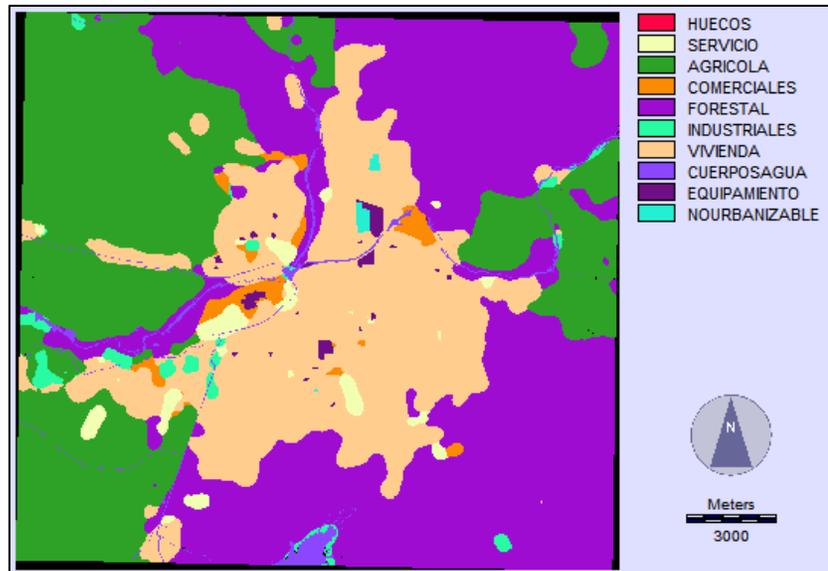


Figura 62. Primeras simulaciones con el AC

El caso anterior es similar a los demás usos, se llevó a cabo la calibración manual en función de los resultados obtenidos en distintas simulaciones, a continuación se muestran las matrices de calibración resultantes.

Servicios

	Distancias																	
K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7
2	100	98	96	94	92	90	88	86	84	82	80	78	76	74	72	70	68	66
3	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7
4	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
5	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7
6	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
7	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43
8	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
9	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
10	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3

Agrícola

	Distancias																	
K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	40	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	-50	-45	-40	-35	-30	-25	-20	-15	-10	-5	0	5	10	15	20	25	30	35
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Comercial

	Distancias																	
K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13
2	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13
3	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13
4	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43
5	-30	-25	-20	-15	-10	-5	0	5	10	15	20	25	30	35	40	45	50	55
6	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13
7	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13
8	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7
9	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13
10	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13

Forestal

	Distancias																	
K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	10	9	8	7	6	5	4	3	2	1	1	1	1	1	1	1	1	1
2	10	9	8	7	6	5	4	3	2	1	1	1	1	1	1	1	1	1
3	10	9	8	7	6	5	4	3	2	1	1	1	1	1	1	1	1	1
4	10	9	8	7	6	5	4	3	2	1	1	1	1	1	1	1	1	1
5	60	9	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43
6	10	9	8	7	6	5	4	3	2	1	1	1	1	1	1	1	1	1
7	10	9	8	7	6	5	4	3	2	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Industrial

	Distancias																	
K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7
2	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7
3	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7
4	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7
5	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7
6	70	69	68	67	66	65	64	63	62	61	60	59	58	57	56	55	54	53
7	10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Urbana

	Distancias																	
K	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
2	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
3	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
4	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13
5	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
6	-50	-47	-44	-41	-38	-35	-32	-29	-26	-23	-20	-17	-14	-11	-8	-5	-2	1
7	100	98	96	94	92	90	88	86	84	82	80	78	76	74	72	70	68	66
8	-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3
9	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
10	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3

8.3 SIMULACIONES CON LA RED NEURONAL INTEGRADA

Hasta este punto, se había respetado el modelo de R. White con el fin de replicar sus resultados. Sin embargo, para tener una mayor precisión del modelo, se propone la sustitución del coeficiente estocástico ν , por uno que denominaremos c_j , dónde este último término, significa el valor de clasificación c que asigna la RNA, hacia un estado objetivo j . Entonces, la variación al modelo, es resultado de la integración de la RNA al AC, se muestra la salida de la RNA sustituyendo el parámetro estocástico del modelo original, por uno de clasificación (Figura 63).

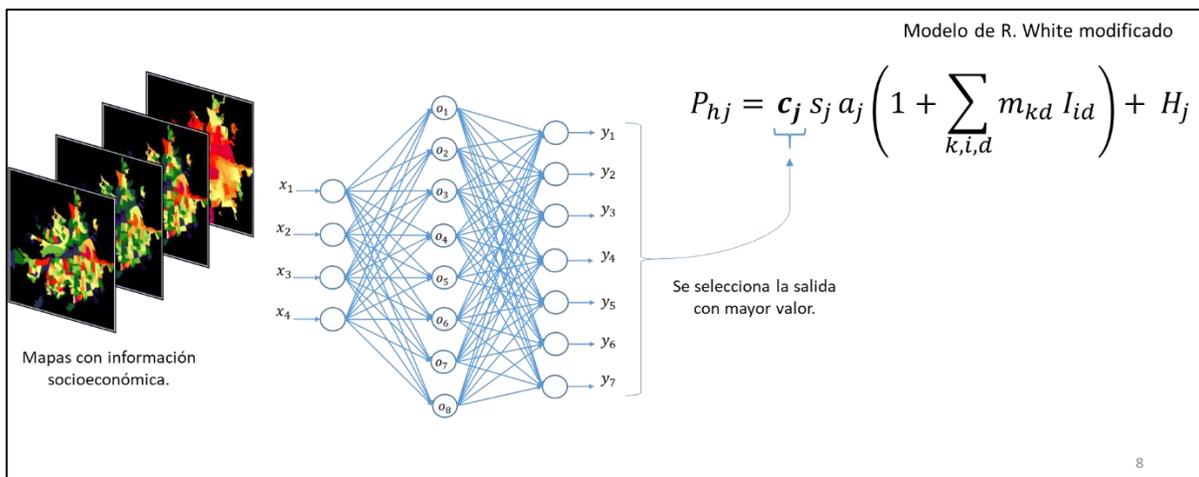


Figura 63. Variables socioeconómicas espacializadas como entrada a la RNA, y la salida de la misma como entrada al modelo

El conjunto de entrenamiento de la red (Figura 66), consistió en 500 quintuplas (las 4 variables socioeconómicas, y la salida deseada), usando el algoritmo de retropropagación. Una vez alcanzado el mínimo de error (0.1), se dio por terminado el entrenamiento, en ese momento la RNA se encuentra lista para procesar información.

Conservando las configuraciones del apartado (10.1), se ejecutó una simulación ex post, de nuevo para el rango de 1997-2004, pero ahora incluyendo la RNA entrenada y el AC. El resultado de esta nueva simulación (Figura 64), manifestó una ligera mejoría con respecto a los

obtenidos únicamente con el AC (Figura 61). De manera visual no es muy evidente tal mejoría, eso se ve reflejado en el índice Kappa (pixel a pixel), sin embargo, los patrones radiales parecen estar desarrollarse de una manera más controlada. Tiene sentido, la categorización que ofrece el aporte de la RNA al potencial de transición, con el valor de predicción del pixel (y no uno aleatorio), ha mejorado ligeramente la salida producida.

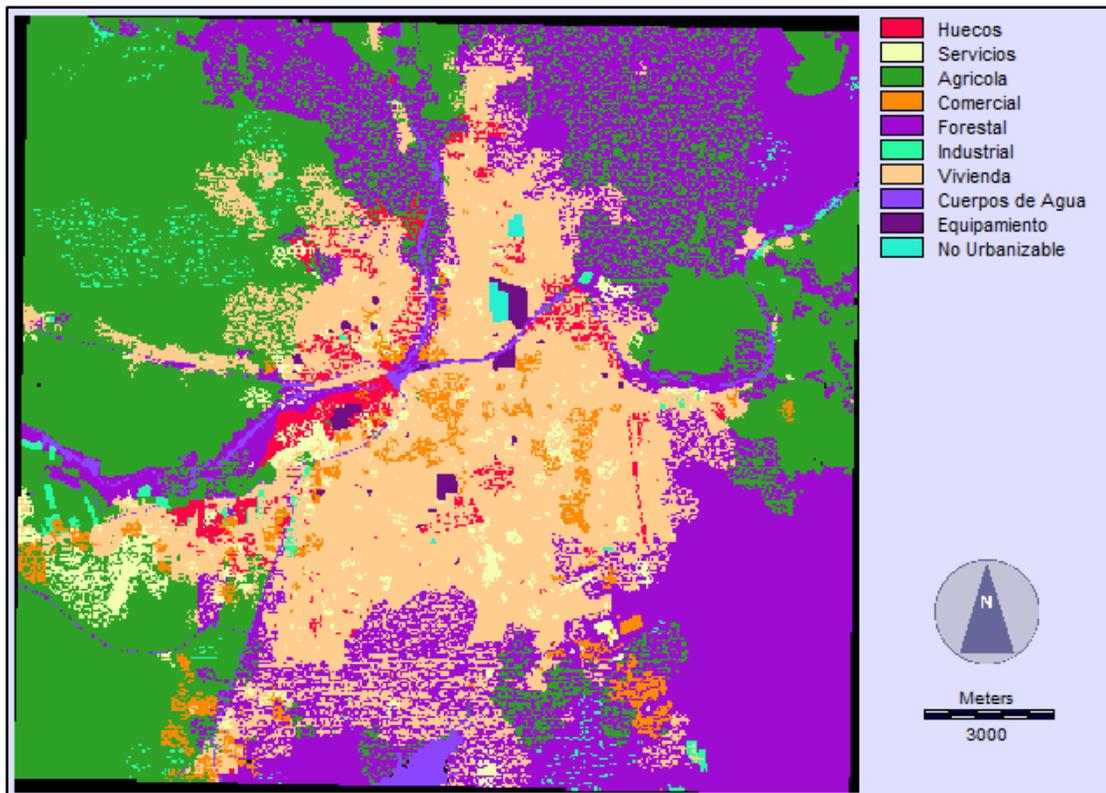


Figura 64. Simulación ex post de 1997 a 2004, con el AC y RNA integrada

Cramer's V = 0.6016 Kappa = 0.6595

Sobre la dispersión de los píxeles en el mapa, esto se debe a que el AC genera estas nuevas ganancias de uso de suelo con base a las características de sus vecinos, esto influye en que su aparición no sea uniforme.

El diseño final de la red neuronal utilizada (Figura 65), consta de 4 neuronas en la capa de entrada, esto para procesar los valores provenientes de las 4 variables socioeconómicas (escolaridad, económicamente activos y sexo [hombre y mujer]). En la capa oculta se tienen 8 neuronas, y en la capa de salida se tienen 7 neuronas con función de transferencia sigmoideal, se busca que la red emita valores entre 0.0 y 1.0; cercano a 1.0 para valores de clasificación altos, y cercano a 0.0 para valores bajos.

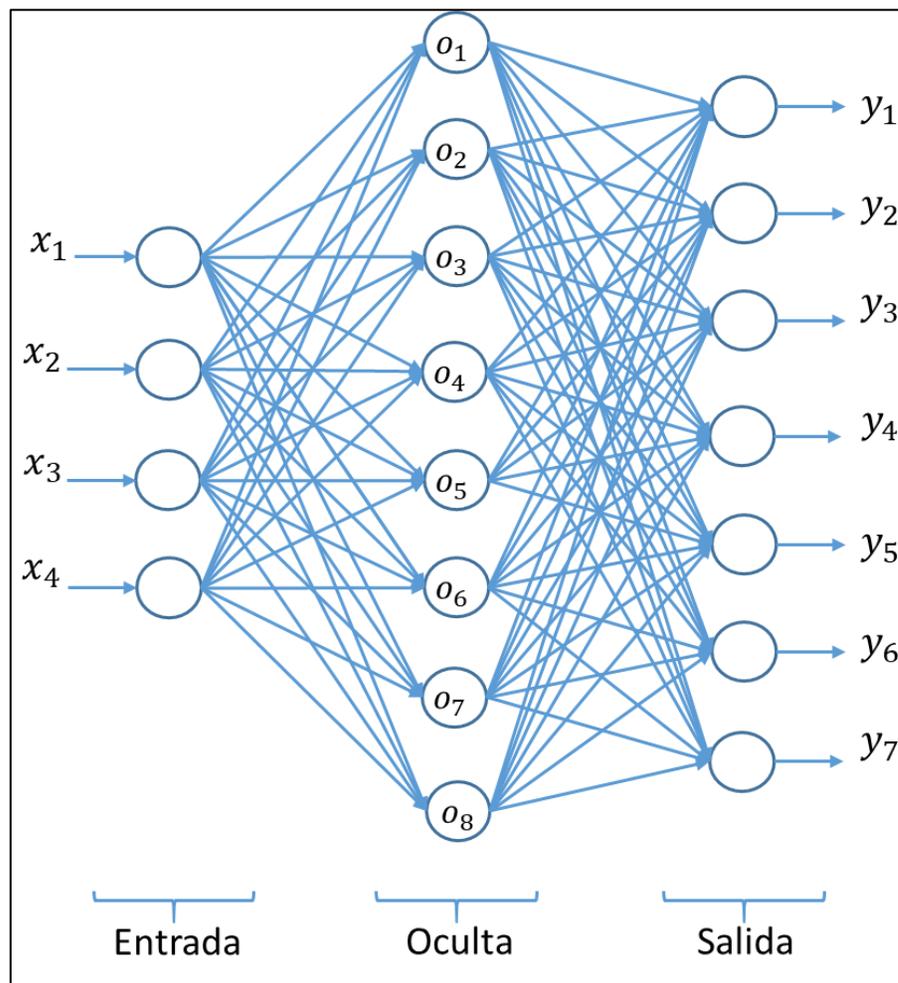


Figura 65. Diseño final de la red neuronal artificial y sus interconexiones

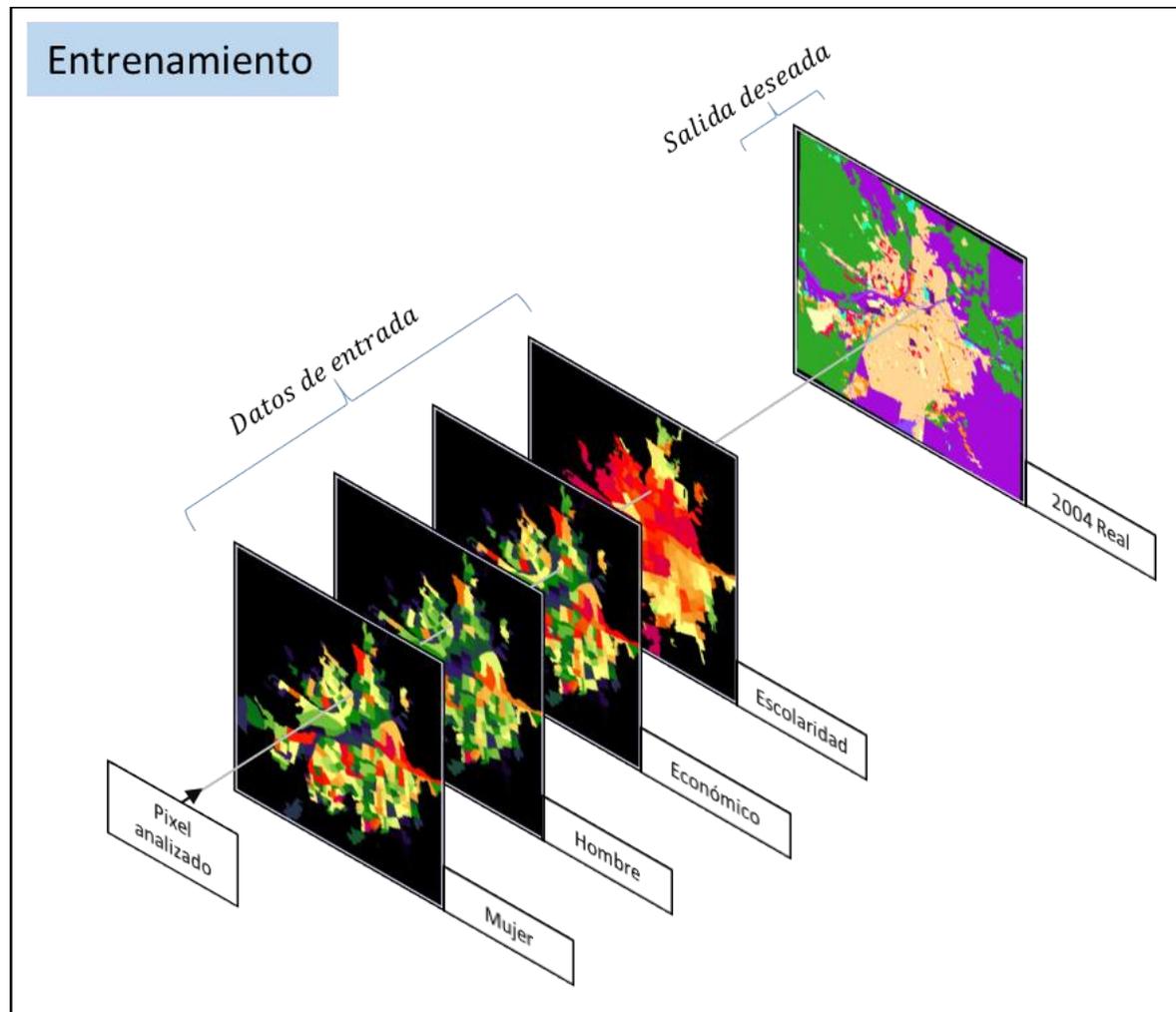


Figura 66. Esquema de entrenamiento por pixel

En general, ya sea por entrenamiento o por obtener un valor de clasificación durante la simulación, la RNA procesa la entrada en la misma coordenada, en 4 mapas distintos (para este caso), como lo muestra la Figura 66. Para el entrenamiento, se utiliza el mapa del año objetivo a simular, con ello se tiene la salida deseada (la ganancia de cada uso urbano) para supervisar y ajustar los pesos de la red, hasta alcanzar el mínimo de error.

Durante la ejecución de la simulación y clasificación (Figura 67), es decir, después del entrenamiento, ya no se ocupa la salida deseada, en este punto la RNA se limita a entregar el resultado de la clasificación al AC, en función de los valores socioeconómicos procesados.

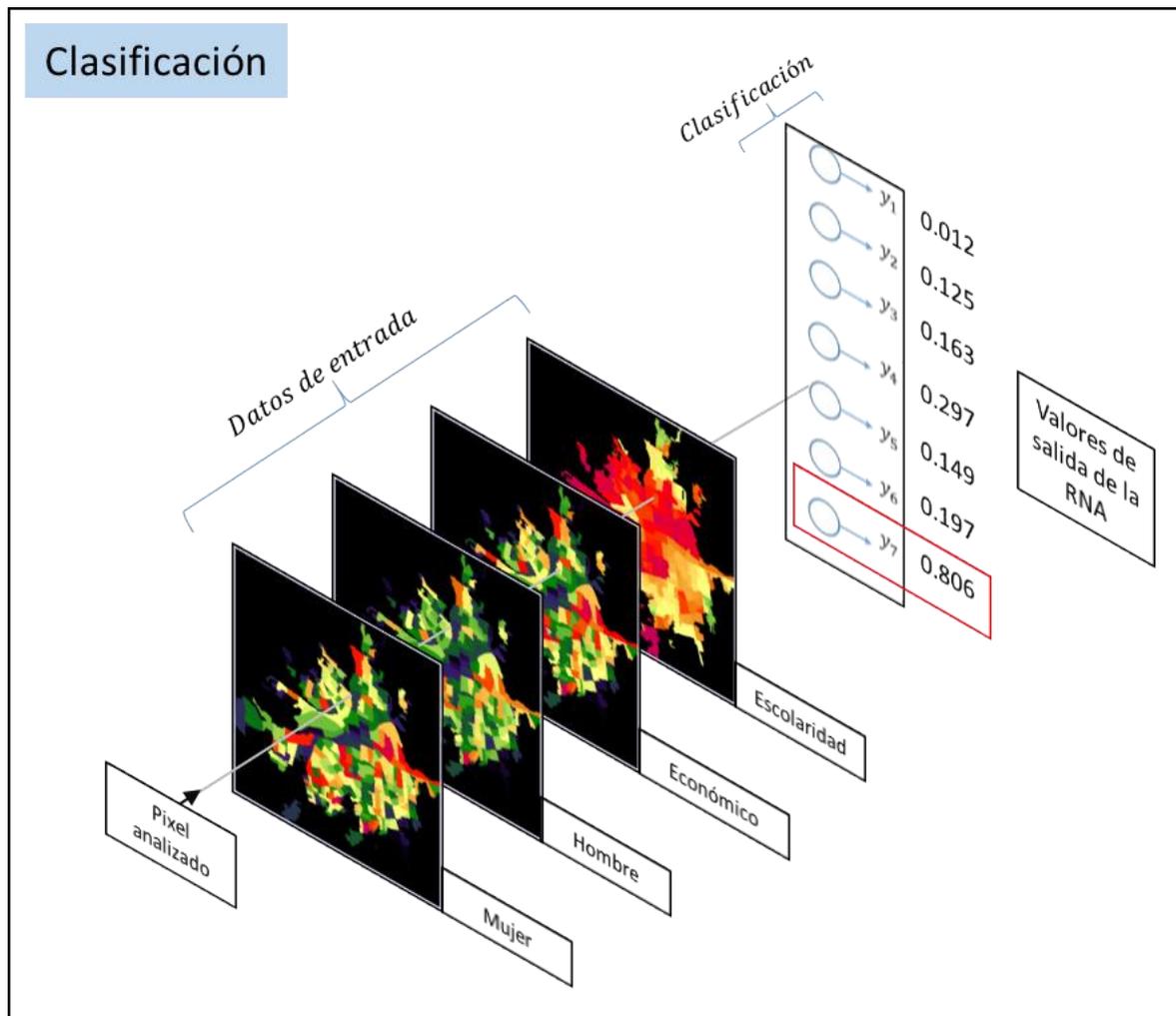


Figura 67. Esquema de clasificación por pixel

Observe la salida de la RNA, el vector de resultados contiene el valor de clasificación de cada neurona, para cada uso dinámico. Con referencia a la clasificación de usos de suelo urbano (Tabla 3), según los datos de entrada, al calcular los potenciales de transición de un determinado pixel en el mapa, el uso #7 (vivienda) obtendría una ponderación de 0.806 en los cálculos, el #2 (servicios) una ponderación del 0.125, así sucesivamente. Estas ponderaciones llegan al AC a través del parámetro c_j (Figura 63).

8.4 VALIDACIÓN DE SIMULACIONES

El proceso de validación, consiste en generar una segunda simulación a otra fecha anterior, esto con el fin de analizar el grado de similitud que alcanza el modelo al generar pronósticos fuera del rango original de configuración y ajustes. Para este propósito, el mapa inicial (de 1994) se simuló a 14 épocas, la validación se llevó a cabo con el mapa del 2011 (Figura 40).

El resultado (Figura 68), muestra el comportamiento del sistema de simulación al generar un mapa fuera de los alcances iniciales. Las configuraciones para esta prueba fueron las mismas que se usaron para generar el mapa del 2004.

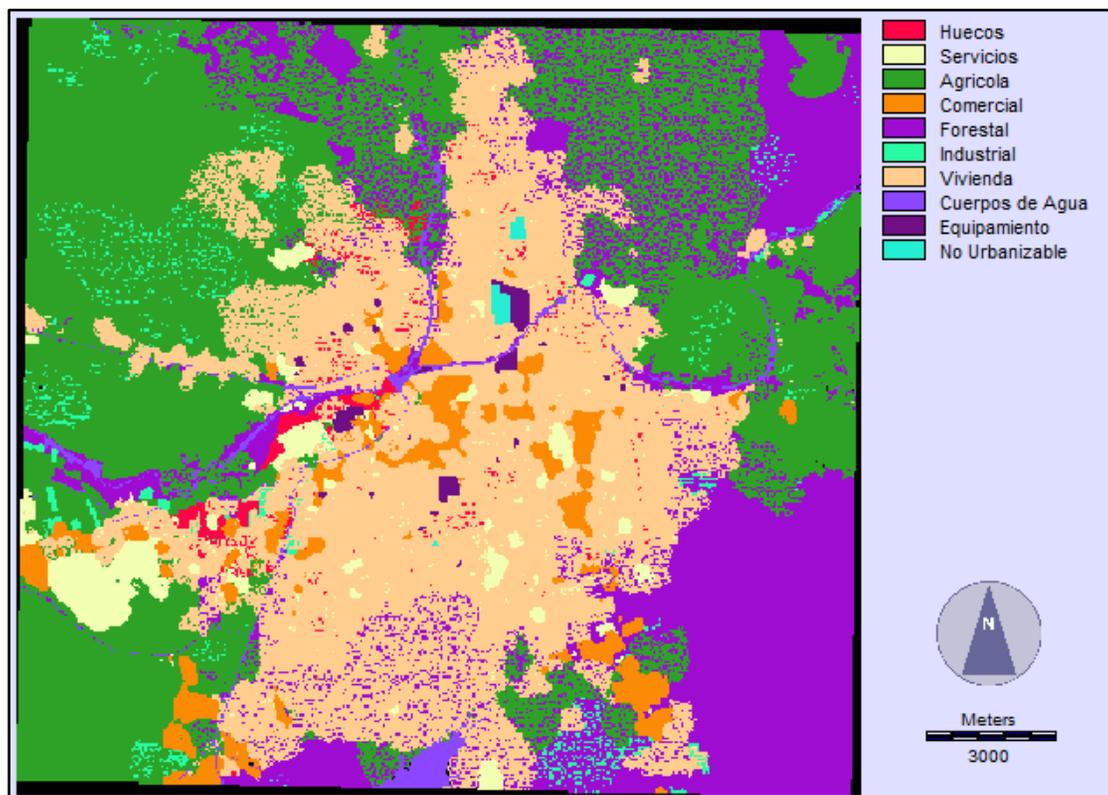


Figura 68. Mapa simulado al 2011 AC + RNA

Los índices **Cramer's V** = 0.5626 y **Kappa** = 0.5726, disminuyeron (con respecto a la simulación de 7 épocas versus el mapa del 2004) al aumentar el número de épocas. Es de esperarse que la precisión de la simulación sea menor mientras se incrementa el horizonte de la simulación.

Con los resultados obtenidos hasta este punto, se tiene la metodología necesaria para integrarla en el diseño de un agente cognitivo, que permita modelar las características que aporta un AC y una RNA en el ámbito del crecimiento urbano.

8.6 DEFINICIÓN DEL AGENTE COGNITIVO

Con base a los resultados, el diseño del agente cognitivo contempla las interacciones y necesidades vistas durante los experimentos. Se definen dos tipos de agente, agente habitante y agente planificador (Tabla 11); se utiliza BDI (ver apartado 9.1) como paradigma para definirlos.

Tabla 11. Definición del MBA con agente habitante y agente planificador

Tipos de Agente	Características	Instancia	Características Específicas
Habitante	Creencias: Pesos de la Red Neuronal Deseos: Búsqueda del mejor potencial de transición Intenciones: Selección del mejor pixel	Agente Residencial	Creencias: Red Neuronal entrenada para la clasificación de Zonas Residenciales. Deseos: Clasificación de pixel a Zona Residencial. Intenciones: Aplicación de algoritmo neuronal para la clasificación.
		Agente Comercial	Creencias: Red Neuronal entrenada para la clasificación de Zonas Comerciales. Deseos: Clasificación de pixel a Zona Comercial. Intenciones: Aplicación de algoritmo neuronal para la clasificación.
		Agente Industrial	Creencias: Red Neuronal entrenada para la clasificación de Zonas Industriales. Deseos: Clasificación de pixel a Zona Industrial. Intenciones: Aplicación de algoritmo neuronal para la clasificación.

	Creencias: Mapa de demanda		Creencias: Mapa de demanda para un uso urbano específico.
Planificador	Deseos: Distribución espacial del crecimiento	Agente Planificador	Deseos: Distribución espacial del crecimiento con base a su mapa de demanda.
	Intenciones: Selección de la zona		Intenciones: Selección de la zona, en función de la distribución espacial.

El diagrama mostrado en la Figura 69, muestra la secuencia de actividades para cada agente siguiendo el modelo BDI, independientemente del tipo de agente.

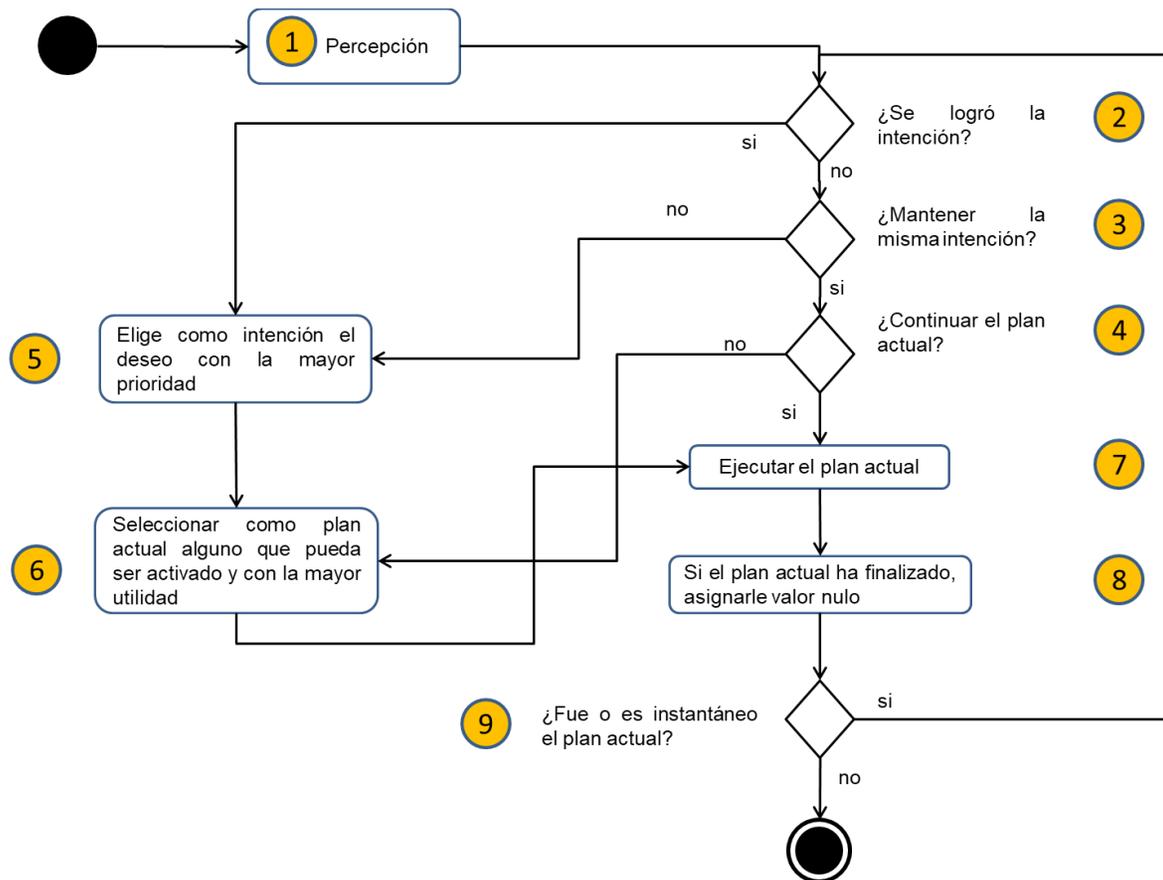


Figura 69. Diagrama de actividades

CÁPITULO: 9 CONCLUSIONES

El planteamiento inicial de la investigación, explora el incremento en cuanto al uso de distintas técnicas de modelado para el estudio de sistemas complejos. Se hace mención que, los autómatas celulares son una de las técnicas más utilizadas para el modelado del crecimiento urbano, sin embargo, resulta reducida la cantidad de modelos en los que se integran técnicas como Redes Neuronales y Agentes Inteligentes. En este orden de ideas, a continuación se examinan los resultados obtenidos, y el cómo estos pueden dar respuesta al planteamiento inicial. Finalmente se discuten los resultados desde distintas perspectivas.

Una de las preguntas de investigación que surgieron a raíz del planteamiento propuesto fue *¿Qué características debe tener el AC que modele el ambiente el cual servirá de entorno al MBA?*, para contestar la pregunta, es necesario dividir la respuesta en 3 partes:

- Primero, se deben obtener las variables territoriales del sistema: para crear las variables de entrada al AC, fue necesario delimitar la zona de estudio (para la delimitación del espacio de evolución), crear las capas correspondientes a cada uso urbano (para definir los estados del autómata), ejecutar un análisis de pérdidas y ganancias (para definir las demandas de suelo), y posteriormente observar las dinámicas urbanas resultantes (para calibrar las matrices). Además de los mapas de aptitud y accesibilidad.
- Segundo, la configuración del AC: según lo analizado con las variables territoriales, llamado también, caracterización. Consistió principalmente en la calibración de las matrices de vecindad del autómata, este procedimiento, es el resultado directo de la información proporcionada por las variables territoriales.
- Tercero, el entorno adecuado para el MBA: el espacio de evolución del AC para la simulación de crecimiento urbano, consiste en una malla bidimensional de $m \times n$, la cual contiene los estados actuales del territorio. Esta representación bidimensional, sirve como entorno territorial (y de percepción) para los agentes del modelo, a través de él se cataliza la participación de los componentes del MBA.

Para dar una respuesta integral a la pregunta planteada, podemos resumir los 3 apartados de la siguiente manera: Las características -iniciales- que debe tener un AC, serán aquellas que emanen de las dinámicas urbanas observadas en el área de estudio, además de la afinación del modelo a través de las matrices de calibración, sin olvidar que el espacio de evolución debe ser bidimensional para que sirva como entorno de percepción para un MBA.

Una vez caracterizado el AC, se aborda la siguiente pregunta de investigación *¿Cómo definir de manera formal un agente que modele el aprendizaje a través de una red neuronal para la predicción y generación de escenarios territoriales, tomando en consideración los distintos factores sociales que pueden afectar el ordenamiento del territorio?* De nuevo, para contestar la pregunta es necesario responder en 3 partes:

- Primero, fue necesaria la integración de una RNA con un AC, con los resultados obtenidos, además de comprobar la mejoría que se logra al robustecer un AC con otra técnica, una RNA para este caso, también se determinó y comprobó a través de las simulaciones, que las variables socioeconómicas aportan información pertinente al desarrollo del territorio urbano.
- Segundo, modificación a la regla de evolución de un AC: el modelo original propuesto por R. White, para el cálculo del potencial de transición P_{hj} , no contempla el uso de una RNA. Fue necesario modificar el modelo (ver apartado 8.1) para sustituir el coeficiente de perturbación estocástica v por la salida de la red c_j .
- Tercero, BDI para definir el modelo: Usando el paradigma BDI se formalizan los niveles de abstracción requeridos (Creencias, Deseos, Intenciones), tomando como punto de partida, la interacción de la RNA con el AC.

La respuesta integral a la segunda pregunta, se resume en: La definición formal de un agente, viene dada por los mecanismos de interacción que existen entre los 3 componentes del sistema; el medio ambiente, el AC y la RNA. La generalización de estas interacciones se toman como

base y se formalizan haciendo uso del paradigma BDI, en donde se definen las creencias, los deseos e intenciones, según el tipo de agente.

Por último, y debido a que involucra toda la investigación realizada, la pregunta final representaba el mayor reto, *¿Cómo definir un MBA en donde los agentes se integren y coexistan considerando en la definición de los mismos el uso de una RNA que modele el aprendizaje?* para responder, es necesario retomar las respuestas a las 2 preguntas anteriores:

- El resultado de la primera, nos proporcionó un AC territorial, caracterizado por las dinámicas del área de estudio. Esto nos llevó a la comprensión de que partir de las interacciones existentes:
 - Las variables socioeconómicas aportan una mejora.
 - La demanda de suelo anual y su distribución espacial son características importantes para la definición del MBA.
- El resultado de la segunda, nos entregó ese mismo AC pero con una RNA integrada, y además, como resultado del acercamiento de las variables socioeconómicas a la RNA, se sentaron las bases para la definición del agente cognitivo (uno para cada uso), el cual pronto se convertiría como el principal elemento del MBA.

En ese orden de ideas, y considerando los aspectos antes mencionados, la definición de cada tipo de agente (Tabla 11), tuvo que contemplar características como:

- Manejo de mapas de demanda.
- Distribución equitativa del crecimiento.
- Ejecución de la red neuronal.
- Valoración del píxel según la base de conocimiento.

Tales características forman parte de distintos agentes, estos coexisten en el modelo desarrollando sus actividades en el mismo entorno. Sus interacciones, quedan plasmadas en el diagrama de actividades (Figura 69), y cada agente queda definido formalmente con el

paradigma BDI. Parte del trabajo futuro (Anexo), es la materialización de estas definiciones, es decir, construir el Sistema Multiagente a partir de las actividades y definiciones mencionadas, haciendo uso del lenguaje de modelado GAML y su ejecución en GAMA.

9.1 DISCUSIÓN

Para discutir los resultados, es necesario abordar los detalles desde dos puntos de vista, el territorial y el computacional.

Dentro del aspecto territorial, cuando se habla de características iniciales (caracterización del autómeta), es necesario ser cuidadosos en la selección de las variables representativas del cambio, una mala selección podría aportar datos innecesarios al modelo y llevar al incremento en los cálculos sin ningún tipo de beneficio. La correcta generación de la cartografía base, es un factor determinante en la calidad de las variables y de su selección, es por ello que se recomienda, que quienes produzcan tal información sea personal especializado y con conocimiento del área de estudio, esto con el fin de disminuir los errores de fotointerpretación.

Desde el punto de vista computacional, se experimentaron algunas dificultades, a continuación se mencionan algunas:

- Utilizar un SIG o generar herramientas propias: Sin duda, los SIG actualmente ofrecen una gran cantidad de funciones que fueron útiles para la parte territorial, y algunos como IDRISI ofrecen inclusive módulos programables para implementar un AC y una RNA, sin embargo estos módulos son “cajas negras”, es decir, no existe manera de acceder al código interno para realizar las adecuaciones que se tenían contempladas, por ello se eligió generar las herramientas necesarias.
- La selección del lenguaje: para el AC, se desarrollaron distintas herramientas en Microsoft C# y ANSI C, siendo este último el que se utilizó para los cálculos de mayor costo computacional. En algunas situaciones se hizo uso de programación paralela para agilizar los tiempos de procesamiento.

Finalmente, al explorar la efectividad de un AC con una RNA integrada, se descubrió una mejora en las simulaciones con respecto a las producidas únicamente con el AC, sin embargo la cantidad de cálculos necesarios y el tiempo de procesamiento se eleva considerablemente. Lo anterior podría verse como un problema, pero en realidad no lo es; independientemente del incremento en el costo computacional, las herramientas y técnicas desarrolladas cumplen con el objetivo de mejorar los resultados.

Sin embargo, es necesario acercar este tipo de herramientas a investigadores que no son especialistas en computación para operarlas, esto se puede lograr con el desarrollo de herramientas intuitivas con una interfaz hombre-máquina que permita fácilmente operar los módulos desarrollados.

9.2 TRABAJO FUTURO

Con base a lo mencionado en los últimos capítulos, existen algunas actividades y aristas de investigación posibles, que pueden continuar con aspectos en los que se decidió “pasar a lo siguiente” en pro de seguir avanzando hacia los objetivos principales. Cada una de estas aristas presenta distintos grados de complejidad que, a futuro, se contemplan abordar a través de proyectos institucionales e integradores, donde se impulse la formación de recurso humano. A continuación se mencionan:

- Desarrollar e implementar el SMA en GAMA y GAML a partir del MBA propuesto. Se toma como primera actividad futura a completar, la puesta en marcha de un SMA que sirva como nicho de experimentos y convergencia de ideas.
- Análisis de sensibilidad. Es uno de los módulos próximos a desarrollar, con esto podrá medirse la variación que sufren los resultados a partir de la variación aleatoria de los datos de entrada.
- Cálculo automático de la demanda. Implementar un módulo integrado a los demás desarrollados, para que el cálculo de la demanda se haga de manera automática.

- Calibración automática de las matrices. Explorar algoritmos evolutivos que se han aplicado (los genéticos por ejemplo), en la búsqueda y optimización de valores para matrices de calibración.
- Integración de los distintos programas elaborados, en una herramienta con interfaz intuitiva. Además de la construcción del SMA, los programas desarrollados durante la investigación para los experimentos, tienen el potencial para ser utilizados por otros investigadores tanto en el área de crecimiento urbano, como en otras áreas afines.
- Aplicar desde otro enfoque la RNA al modelo de crecimiento. Las variables socioeconómicas procesadas, se podrían convertir en mapas de clasificación, con esto se tendría oportunidad de aplicarlos a otras simulaciones o modelos de crecimiento, sin necesidad de tener una RNA integrada.
- Generación de cartografía base, permanente y en constante mejora de lo que se tiene, apoyados con un servidor de mapas. Durante la investigación, se trabajó en la instalación de un servidor de mapas con el objetivo de mantener centralizado el acervo cartográfico, y fomentar el trabajo colaborativo con respecto al mantenimiento y mejoras a los mapas. Sin embargo, el servidor quedó en pausa temporal, por distintos motivos técnicos.

BIBLIOGRAFÍA

- [1] Eea, *The pan-European environment: glimpses into an uncertain future*, vol. 4, núm. 4. 2007.
- [2] Á. Peraza Garzón, R. Rodríguez Zamora, W. Plata Rocha, y U. A. J. de Jesus, “Sistemas Multiagente para la Simulación de Dinámicas Urbanas”, *Geografía y Sistemas de Información Geográfica (GEOSIG)*, pp. 56–65, 2016.
- [3] Y. Bar-Yam, *Dynamics of Complex Systems*. 1997.
- [4] R. White, G. Engelen, y I. Uljee, “The use of constrained cellular automata for high-resolution modelling of urban land-use dynamics”, *Environ. Plan. B Plan. Des.*, vol. 24, núm. 3, pp. 323–343, 1997.
- [5] S. J. Rusell y P. Norvig, *Inteligencia Artificial. Un Enfoque Moderno*. 2004.
- [6] A. Mas, *Agentes Software y Sistemas Multi-Agente. Conceptos Arquitecturas y Aplicaciones*. 2005.
- [7] M. Wooldridge, *An Introduction to MultiAgent Systems*, Second. Wiley, 2009.
- [8] R. H. José y J. M. Victor, *Redes Neuronales Artificiales. Fundamentos, modelos y aplicaciones*. 2000.
- [9] F. Aguilera Benavente, “Predicción del crecimiento urbano mediante sistemas de información geográfica y modelos basados en autómatas celulares”, *Geofocus Rev. Int. Cienc. y Tecnol. la Inf. Geográfica*, pp. 81–112, 2006.
- [10] V. Machado, a. Neto, y J. D. De Melo, “A Neural Network Multiagent Architecture Applied to Industrial Networks for Dynamic Allocation of Control Strategies Using Standard Function Blocks”, *IEEE Trans. Ind. Electron.*, vol. 57, núm. 5, pp. 1823–1834, 2010.
- [11] C. G. Ralha, C. G. Abreu, C. G. C. Coelho, A. Zagherro, B. Macchiavello, y R. B. Machado, “A multi-agent model system for land-use change simulation”, *Environ. Model. Softw.*, vol. 42, pp. 30–46, 2013.
- [12] M. Galán, A. López, y R. Martínez, “Simulación de sociedades artificiales de agentes y autómatas celulares en Ingeniería de Organización : aplicación a la gestión del agua”, pp. 1033–1042, 2004.
- [13] J. Dijkstra, H. J. P. Timmermans, y A. J. Jessurun, “A Multi-Agent Cellular Automata System for Visualising Simulated Pedestrian Activity”, 2000.
- [14] a Saltelli, S. Tarantola, y K. Chan, “A Role for Sensitivity Analysis in Presenting the Results from MCDA Studies to Decision Makers”, *J. Multi-Criteria Decis. Anal.*, vol. 145, núm. May, pp. 139–145, 1999.
- [15] M. Galán, A. López, y R. Martínez, “Modelado y simulación basada en agentes con SIG para la gestión de agua en espacios metropolitanos”, 2006.
- [16] A. López-Paredes, D. Saurí, y J. M. Galán, “Urban water management with artificial societies of agents: The FIRMABAR simulator”, vol. 81 No. 3, pp. 189–199, 2005.
- [17] E. Dick, K. de Jong, T. Harry, y A. Bakema, “PUMA: MULTI-AGENT MODELLING OF URBAN SYSTEMS”, pp. 1–25, 2005.
- [18] S. Pengyuan, “A More Realistic Simulation of Pedestrian based on Cellular Automata”, 2009.
- [19] Almeida *et al.*, “Stochastic cellular automata modeling of urban land use dynamics: Empirical development and estimation”, *Comput. Environ. Urban Syst.*, vol. 27, pp. 481–509, 2003.
- [20] A. D. Syphard, K. C. Clarke, y J. Franklin, “Using a cellular automaton model to forecast the effects of urban growth on habitat pattern in southern California”, *Ecol. Complex.*, vol. 2, pp. 185–203, 2005.
- [21] A. Spicher, N. Fats, y O. Simonin, “Translating Discrete Multi-Agents Systems into Cellular Automata : Application to Diffusion-Limited Aggregation”, pp. 270–282, 2010.

- [22] F. Aguilera Benavente, W. Plata Rocha, y J. Bosque Sendra, “Diseño y simulación de escenarios de demanda de suelo urbano en ámbitos metropolitanos”, *Rev. Int. sostenibilidad, Tecnol. y humanismo*, pp. 57–80, 2009.
- [23] F. . Aguilera, M. . Gómez Delgado, y C. C. Cantergiani, “Instrumentos de simulación prospectiva del crecimiento urbano”, *Ciudad y Territ. Estud. Territ.*, vol. XLII, núm. 1, pp. 481–496, 2010.
- [24] M. Gomez Delgado y J. I. Barredo Cano, *Sistemas de Información Geográfica y Evaluación Multicriterio en la Ordenación del Territorio. 2ª Ed*, 2005a ed. 2005.
- [25] C. Cantergiani, M. G. Delgado, y C. Vergara, “Desarrollo de un Modelo Basado en Agentes para la simulación del crecimiento urbano : Submodelo de asignación de nuevo suelo urbanizable”, 2014.
- [26] J. I. Barredo, L. Demicheli, C. Lavalle, M. Kasanko, y N. McCormick, “Modelling future urban scenarios in developing countries: An application case study in Lagos, Nigeria”, *Environ. Plan. B Plan. Des.*, vol. 31, pp. 65–84, 2004.
- [27] F. A. Benavente, “Análisis Espacial para la Ordenación Eco-Paisajística de la Aglomeración Urbana de Granada”, Universidad de Granada, 2008.
- [28] J. P. Rigol Sánchez, M. Chica Olmo, E. Pardo Igúzquiza, V. Rodríguez Galiano, y M. Chica Rivas, “Análisis e integración de datos espaciales en investigación de recursos geológicos mediante Sistemas de Información Geográfica”, *Bol. la Soc. Geol. Mex.*, vol. 63, pp. 61–70, 2011.
- [29] R. G. Pontius, E. Shusas, y M. McEachern, “Detecting important categorical land changes while accounting for persistence”, *Agric. Ecosyst. Environ.*, vol. 101, pp. 251–268, 2004.
- [30] INEGI, “Marco Geoestadístico Nacional”, *Marco Geoestadístico Nacional*, 2017. [En línea]. Disponible en: <http://www.inegi.org.mx/geo/contenidos/geoestadistica/>.
- [31] M. Füllsack y S. Viczek, “Introduction to Systems Scientific Modeling and Simulation”, 2012.
- [32] E. W. Weisstein, “Cellular Automaton”, 2015. .
- [33] A. T. Crooks y S. Wise, “Multi-Agent Systems for Urban Planning”, pp. 29–56, 2014.
- [34] EcuRed, “Sistemas Multiagentes”, 2014. [En línea]. Disponible en: http://www.ecured.cu/index.php/Sistemas_Multiagentes. [Consultado: 01-ene-2014].
- [35] F. Aguilera Benavente, L. M. Valenzuela Montes, J. A. Soria Lara, M. Gómez Delgado, y W. Plata Rocha, “Escenarios Y Modelos De Simulación Como Instrumento En La Planificación Territorial Y Metropolitana”, *Ser. Geográfica*, vol. 17, pp. 11–28, 2011.
- [36] J. Xie y C. Liu, “Multi-agent Systems and their Applications”, *J. Int. Counc. Electr. Eng.*, vol. 8972, pp. 1–10, 2017.
- [37] R. Chandra, *Parallel Programming in OpenMP*. 2001.
- [38] A. Ortega *et al.*, “Modelos de computación inspirados en biología”. 2003.
- [39] P. Ponce, *Inteligencia Artificial. Con aplicaciones a la ingeniería*. 2010.
- [40] P. Phil, *Neural Networks*. 2000.
- [41] R. Beale y T. Jackson, *Neural Computing. An Introduction*. 1990.
- [42] T. M. J. Erick. R. Kandel, James H. Schwartz, “Nerve cells and behavior”, en *Pinciples of Neural Science*, 1991.
- [43] C. Francisco, “Fundamentos Biológicos del Aprendizaje y la Memoria”, 2005. [En línea]. Disponible en: [http://www.uhu.es/francisco.cordoba/asignaturas/FBAM/TEMAS PDF/3-LA NEURONA.pdf](http://www.uhu.es/francisco.cordoba/asignaturas/FBAM/TEMAS%20PDF/3-LA%20NEURONA.pdf). [Consultado: 01-ene-2012].
- [44] A. Barboza, “Sinapsis y Tipos de Sinapsis”, *Mayo 17*, 2012. [En línea]. Disponible en: <http://www.slideshare.net/alejandrabza/sinapsis-y-tipos-de-sinapsis>. [Consultado: 01-ene-2012].

- [45] “Historia de las Redes Neuronales”, 2012. [En línea]. Disponible en: <http://gruporedesneuronales.wordpress.com/historia-de-las-redes-neuronales-2/>. [Consultado: 01-ene-2012].
- [46] D. Andina de la fuente, “Redes Neuronales Artificiales”. [En línea]. Disponible en: <http://www.gc.ssr.upm.es/inves/neural/ann2/concepts/app.htm>. [Consultado: 01-ene-2012].
- [47] T. Gustavo y I. Gustavo, “Organizaciones Inteligentes - Pronósticos y Predicciones”, 2013. [En línea]. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/21427/Documento_completo.pdf?sequence=1. [Consultado: 01-ene-2013].
- [48] “Arbib, M.A. [1987], Brains, Machines and Mathematics, 2 nd Edition, New York, Springer-Verlag.”
- [49] “Artificial Neuron”, 2012. [En línea]. Disponible en: http://en.wikipedia.org/wiki/Artificial_neuron. [Consultado: 01-ene-2012].
- [50] B. Martín del Brio y A. Sanz, *Redes Neuronales y Sistemas Borrosos*, 2007a ed. Alfaomega RA-MA, 2007.
- [51] “Neuron Model and Network Architectures”. .
- [52] J. Glyn y B. David, *Matemáticas avanzadas para ingeniería*. 2002.
- [53] W. MathWorld, “Sigmoid Function”. [En línea]. Disponible en: <http://mathworld.wolfram.com/SigmoidFunction.html>.
- [54] “http://www.cds.caltech.edu/~murray/books/AM05/pdf/am06-xferfens_16Sep06.pdf”.
- [55] “Brooks, V. [1986], The Neural Basis for Motor Control, Oxford University Press, New York.”
- [56] “Cristianini, N. and Shawe-Taylor, J. [2000], An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods, Cambridge, U.K, Cambridge Univ, Press.”
- [57] “Dayhoff, J., [1990], Neural Network Architectures - An Introduction, New York, Van Nostrand Reinhold.”
- [58] “Single-Layer Perceptron Neural Networks”, 2012. [En línea]. Disponible en: <http://lcn.epfl.ch/tutorial/english/perceptron/html/intro.html>. [Consultado: 01-ene-2012].
- [59] K. Kiyoshi, “Single-Layer Network”. [En línea]. Disponible en: <http://www.ece.utep.edu/research/webfuzzy/docs/kk-thesis/kk-thesis-html/node15.html>. [Consultado: 01-ene-2011].
- [60] “Gupta, M.M. and Rao, D.H., (Editors), [1994i], Neuro-Control Systems: Theory and Applications, A Vol. of Selected Reprints, IEEE Neural Networks Council, IEEE-Press, New York, 607 pages.”
- [61] “Gupta, M.M. and Sanchez, E., (Editors), [1982], Approximate Reasoning in Decision Analysis, North-Holland, New York, Amsterdam, Oxford, 480 pages”.
- [62] “Gupta, M.M. and Sanchez, E., (Editors), [1983], Fuzzy Information and Decision Processes, North-Holland, New York, Amsterdam, Oxford, January, 451 pages”.
- [63] “Hassoun, M.H. [1995], Fundamentals of Artificial Neural Networks, MIT Press, Cambridge, MA.”
- [64] K. Priddy y P. Keller, *Artificial Neural Network (Introduction)*. 2005.
- [65] “Haykin, S. [1999], Neural Networks: A Comprehensive Foundation, 2 nd Edition, Prentice-Hall, NJ.”
- [66] “Hecht-Nielsen, R. [1990], Neurocomputing, Addison-Wesley Publishing Company.”
- [67] “Kandel, E.R. and Schwartz, J.H. [1985], Principles of Neural Science, New York, North-Holland.”
- [68] “http://ftp.nluug.nl/metalab/docs/LuCaS/Presentaciones/200304curso-glisha/redes_neuronales/curso-glisha-redes_neuronales-html/x105.html”, 2012. [En línea]. Disponible en:

- http://ftp.nluug.nl/metalab/docs/LuCaS/Presentaciones/200304curso-glisa/redes_neuronales/curso-glisa-redes_neuronales-html/x105.html. [Consultado: 01-ene-2012].
- [69] “Neural Networks”. [En línea]. Disponible en: http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html. [Consultado: 01-ene-2012].
- [70] C. M. Almeida, J. M. Gleriani, E. F. Castejon, y B. S. Soares-Filho, “Using neural networks and cellular automata for modelling intra- urban land- use dynamics”, *Int. J. Geogr. Inf. Sci.*, vol. 22, núm. 788848368, pp. 943–963, 2008.
- [71] C. H. Yong y R. Miikkulainen, “Cooperative Coevolution of Multi-Agent Systems”.
- [72] J. A. Inzunza, “La Planeación Urbana en el Municipio Mexicano: Culiacán, Un Caso de Estudio”, 2003.
- [73] “IMPLAN”. [En línea]. Disponible en: <http://www.implanln.gob.mx/www/>. [Consultado: 01-ene-2015].
- [74] K. Alvarez, “Elaboración de Cartografía de Uso de Suelo a partir de Imágenes Geo-eye del Área Metropolitana de Culiacán de Rosales, Sinaloa”, 2015.
- [75] W. Plata-Rocha, M. Gómez Delgado, y J. Bosque Sendra, “Cambios de uso del suelo y expansión urbana en la comunidad de Madrid (1990-2000)”, *Scr. Nov.*, vol. 13, 2009.
- [76] J. Cheng y I. Masser, “Modelling urban growth patterns: A multiscale perspective”, *Environ. Plan. A*, vol. 35, pp. 679–704, 2003.
- [77] V. Kocabas y S. Dragicevic, “Assessing cellular automata model behaviour using a sensitivity analysis approach”, *Comput. Environ. Urban Syst.*, vol. 30, pp. 921–953, 2006.
- [78] M. Batty, “Agents, Cells and Cities: New Representational Models for Simulating Multi-Scale Urban Dynamics”, *UCL Cent. Adv. Spat. Anal.*, vol. 44, núm. 0, pp. 0–18, 2008.
- [79] “Lawrence, S., Giles, C.L., and Tsoi, A.C. (1996), ‘What size neural network gives optimal generalization? Convergence properties of backpropagation’. Technical Report UMIACS-TR-96-22 and CS-TR-3617, Institute for Advanced Computer Studies, University of M”.
- [80] “Elisseeff, A., and Paugam-Moisy, H. (1997), ‘Size of multilayer networks for exact learning: analytic approach’. Advances in Neural Information Processing Systems 9, Cambridge, MA: The MIT Press, pp.162-168.”
- [81] “Backpropagation Explanation”. [En línea]. Disponible en: http://dl.dropboxusercontent.com/u/7412214/Explanation_1.pdf. [Consultado: 01-ene-2013].
- [82] D. Singh, L. Padgham, y B. Logan, “Integrating BDI Agents with Agent-Based Simulation Platforms”, *Auton. Agent. Multi. Agent. Syst.*, vol. 30, núm. 6, pp. 1050–1071, 2016.
- [83] P. Taillandier, M. Bourgais, P. Caillou, C. Adam, y B. Gaudou, “A BDI Agent Architecture for the GAMA Modeling and Simulation Platform”, vol. 1, pp. 3–23, 2017.
- [84] ummisco, “GAMA-Platform”, 2020. .
- [85] J. Pozueta, “MOVILIDAD Y PLANEAMIENTO SOSTENIBLE : Hacia una consideración inteligente del transporte y la movilidad en el planeamiento y en el diseño urbano”, pp. 1–97, 2000.
- [86] A. Aguilera, J. Carrilo, y G. Grageda, “Autómatas celulares para modelar el uso del suelo urbano: Una revisión del modelo de White”, *Vetas*, vol. 3, pp. 85–106, 2001.
- [87] D. O’Sullivan y P. M. Torrens, “Cellular models of urban systems”, *CASA Work. Pap.*, vol. 22, p. 13, 2000.
- [88] C. Matesanz, G. Lazaro, y S. Preciado, “ESPACIAL DE VARIABLES SOCIODEMOGRÁFICAS CON SISTEMAS DE INFORMACIÓN GEOGRÁFICA Y TELEDETECCIÓN”, pp. 723–736.
- [89] N. Gilbert y T. Klaus G., *Simulación para las Ciencias Sociales*. 2005.
- [90] L. Poelmans y A. Van Rompaey, “Complexity and performance of urban expansion models”, *Comput.*

Environ. Urban Syst., vol. 34, núm. 1, pp. 17–27, 2010.

- [91] K. C. Clarke, S. Hoppen, y L. J. Gaydos, “Methods And Techniques for Rigorous Calibration of a Cellular Automaton Model of Urban Growth”, pp. 1–11.
- [92] F. Wu, “Calibration of stochastic cellular automata: the application to rural-urban land conversions”, vol. 16, núm. 8, pp. 795–818, 2002.

ANEXOS

Proyecto en los que se participó:



Título del Proyecto	Caracterización de un autómata celular para la simulación prospectiva de áreas urbanas en Sinaloa					
Clave	5767.16-P					
Vigencia	1 de octubre al 31 de septiembre de 2017					
Director(a) Responsable	Rodríguez Zamora René					
Colaboradores	Peraza Garzón Álvaro, Osuna Peraza Sergio Saúl, Cañedo García Francisco, Barrón Hernández Miguel Ángel					
Entrega de Informe Final	1 al 31 de octubre del 2017					
Asignación Presupuestal	Mes	Abril	Mayo	Junio	Septiembre	Total
	21701	-	-	-	\$35,300.00	\$35,300.00
	31903	-	-	-	-	-

Director(a) Responsable	Rodríguez Zamora René					
Colaboradores	Peraza Garzón Álvaro, Osuna Peraza Sergio Saúl, Cañedo García Francisco, Barrón Hernández Miguel Ángel					
Entrega de Informe Final	1 al 31 de octubre del 2017					
Asignación Presupuestal	Mes	Abril	Mayo	Junio	Septiembre	Total
	21701	-	-	-	\$35,300.00	\$35,300.00
	31903	-	-	-	-	-

Título del Proyecto	Objetos de aprendizaje en b-learning como apoyo en el diagnóstico de la dislexia evolutiva en niños de primaria					
Clave	5768.16-P					
Vigencia	1 de octubre al 31 de septiembre de 2017					
Director(a) Responsable	Ríos Vélez José Daniel					
Colaboradores	Martell Neira Nina, Amezcua Magallón Verónica, Triana Hernández José Luis Guadalupe					
Entrega de Informe Final	1 al 31 de octubre del 2017					
Asignación Presupuestal	Mes	Abril	Mayo	Junio	Septiembre	Total
	21701	-	-	-	\$30,000.00	\$30,000.00
	31903	-	-	-	-	-

Arcos de Belén Núm. 79, PH, Col. Centro, Del. Cuauhtémoc, C.P. 06010, Ciudad de México, Tels. (55) 3601-7500, Ext. 65048, 65066, e-mail: d_posgrado@tecnm.mx

Artículos publicados.



GESIG *Geografía y Sistemas de Información Geográfica (GEOSIG)*. Revista digital del Grupo de Estudios sobre Geografía y Análisis Espacial con Sistemas de Información Geográfica (GESIG). Programa de Docencia e Investigación en Sistemas de Información Geográfica (PRODISIG). Universidad Nacional de Luján, Argentina. <http://www.gesig-proeg.com.ar> (ISSN 1852-8031)

Luján, Año 8, Número 8, 2016, Sección II: Metodología. pp. 56-65

SISTEMAS MULTIAGENTE PARA LA SIMULACIÓN DE DINÁMICAS URBANAS

Álvaro Peraza Garzón¹, René Rodríguez Zamora²,
Wenseslao Plata Rocha³ y José de Jesús Uriarte Adrián⁴

¹ Universidad Autónoma de Sinaloa, México, (aperaza@uas.edu.mx)

² Universidad Autónoma de Sinaloa, México, (rene.rodriguez@info.uas.edu.mx)

³ Universidad Autónoma de Sinaloa, México, (wenses@uas.edu.mx)

⁴ Universidad Autónoma de Sinaloa, México, (jesus.uriarte@info.uas.edu.mx)

RESUMEN

Se propone el desarrollo de un Sistema Multiagente (SMA) para simular el crecimiento urbano tomando en consideración los distintos factores que lo componen. Este SMA se define a través de dos capas, la primera se utiliza para representar la región geográfica de interés para el análisis y se caracteriza por medio de un Automata Celular (AC), mismo que en la definición de sus estados representa adecuadamente las propiedades del paisaje (suelo urbanizable, pendientes, accesibilidad, etc.) Para esto se hace uso de una latiz bidimensional en la que se determina una geometría que especifica el radio de vecindad espacial para todas y cada una de las celdas o células evaluadas sincrónicamente con una correspondiente función de transiciones locales. Esta capa constituye el medio ambiente para una segunda capa compuesta por agentes inteligentes con la capacidad de desplazarse sobre la región espacial que representa el AC. Las interacciones entre estos agentes y los atributos que los definen (edad, ingresos, escolaridad, etc.) permiten modelar ciertos aspectos que tienen que ver con las características, interacciones o procesos de toma de decisión que surgen en grupos sociales que habitan una determinada región geográfica. Así pues, se define de manera formal un SMA que incluye elementos como: paisaje, escala y aspectos sociales para posteriormente realizar simulaciones cuyos resultados puedan ser utilizados en la construcción de escenarios territoriales en lo referente al crecimiento urbano. Finalmente, estos escenarios se proyectan en mapas haciendo uso de interfaces que permiten la interconectividad con un SIG de código abierto como gvSIG o GRASS.

Palabras Clave: *Sistemas Multiagente, Modelos de Simulación, Crecimiento Urbano*

OpenMP Implementation in the Characterization of an Urban Growth Model Cellular Automaton

Alvaro Peraza Garzón
Instituto Tecnológico de Mazatlán
Universidad Autónoma de Sinaloa
Sinaloa, México

René Rodríguez Zamora
Universidad Autónoma de Sinaloa
Instituto Tecnológico de Mazatlán
Sinaloa, México

Wenseslao Plata Rocha
Facultad de Ciencias de la Tierra y el
Espacio
Universidad Autónoma de Sinaloa
Sinaloa, México

Abstract—This paper presents the implementation of a parallelization strategy using the OpenMP library, while developing a simulation tool based on a cellular automaton (CA) to run urban growth simulations. The characterization of an urban growth model CA is shown and it consists of a digitization process of the land use in order to get all the necessary elements for the CA to work. During the first simulation tests we noticed high processing times due to large quantity of calculations needed to perform one single simulation, in order to minimize this we implemented a parallelization strategy using the fork-join model in order to optimize the use of available hardware. The results obtained show a significant improvement in execution times in function of the number of available cores and map sizes, as a future work, it is planned to implement artificial neural networks in order to generate more complex urban growth scenarios.

Keywords—Cellular automata; parallel programming; simulation models; OpenMP; urban growth

I. INTRODUCTION

The evolution in the land use of the territory is a fundamental element in our society, since it manifests different variables that affect our daily life, for example, accessibility to different points of interest within the city, slopes of the land, etc. This evolution has gained interest mainly fueled by the different environmental problems especially those in urban areas [1]. Thanks to the advances in the computing field and the development of important analytical tools such as Geographic Information Systems (GIS) or simulation models, the study of the changes taking place in metropolitan areas has been promoted [2]. The analysis of the environmental alterations that result from these changes and the development of new planning instruments, has caused that different disciplines, specifically the Artificial Intelligence (AI), approaches from a computer and mathematical point of view to give alternative solutions to this problem [3].

Numerous modeling tools have emerged in recent years. In the case of urban growth, the models based on cellular automata (CA) are the most widely used [4]. Regression models, artificial neural networks (ANNs), multi-criteria evaluation techniques (MCE), and still incipient, agent-based models (ABM) can also be found.

The CA based models are oriented fundamentally towards the representation of the attributes of a given geographic region

in a two-dimensional lattice, in which a neighborhood radius is defined and a certain rule of evolution is applied in order to define the behavior of the CA. With the use of these models it has been possible to generate territorial scenarios prospectively [5]. To generate these scenarios, a characterization of a CA is needed, this has different components, such as the size of the study area, maps of urban uses, map scales, neighborhood radius, evolution rules, slopes, and others geographical factors [6].

The developing of a CA based simulation tool to generate territorial scenarios prospectively in order to implement future simulation techniques, bring us to address some challenges. One of them was, the huge amount of mathematical operations needed in one single simulation, because the complexity of the algorithm to do such operations results to be exponential.

One key calculus in the whole simulation process is, the transition potentials (TPs) of each cell in the map, these TPs show the probability of a cell to change from one state to another. The amount of these TPs have a direct impact on the computation cost needed to perform the mathematical calculations.

To optimize these calculations, we enhanced sequential algorithms with parallelization strategies in order to maximize computational hardware. The library OpenMP (Open Multi-Processing), widely used in parallel programming, helps to implement a parallel strategy called fork-join. This allows to take advantage of hardware resources for the execution of processes in shared memory [7].

The present work aims to implement the fork-join strategy to speed up the necessary TPs calculations and to compare the results against the first sequential algorithm used in the simulation.

The base maps for the experiments were generated from the study area of Culiacan, México. Being the faster growing city in the State of Sinaloa, we plan to use the simulation tool to understand the dynamics of the urban changes and to forecast for planning urban development as a future work.

The remainder of this paper is structured as follows. All material and methods such as, the study area, digitation process, CA model and OpenMP are defined in Section II. Calculus of transition potentials for each pixel using the fork-join model are explained in Section III. Also proposed

