

**DISEÑO DE UN SISTEMA MULTIAGENTE DE PROGRAMACIÓN DE TAREAS BASADO EN  
TECNICAS DE INTELIGENCIA ARTIFICIAL Y SIMULACIÓN DISCRETA PARA SISTEMAS DE  
SERVICIO TIPO RED DE COLAS MULTI-CLASE**

**GUSTAVO FLÓREZ BECERRA**

**Cód. 20101196007**

**Proyecto de Grado**

**Director**

**Ing. MSc Eduyn Ramiro López Santana**

**UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS**

**FACULTAD DE INGENIERÍA**

**MAESTRÍA INGENIERÍA INDUSTRIAL**

**BOGOTÁ**

**2016**

## Contenido

<b>CAPITULO I: INTRODUCCIÓN</b> .....	6
<b>1.1. Antecedentes</b> .....	6
<b>1.2. Planteamiento y delimitación del problema</b> .....	7
<b>1.3. Hipótesis</b> .....	7
<b>1.4. Objetivos de la investigación</b> .....	7
<b>1.4.1. Objetivo general</b> .....	7
<b>1.4.2. Objetivos específicos</b> .....	7
<b>1.5. Marco metodológico</b> .....	8
<b>1.6. Justificación</b> .....	9
<b>CAPITULO II: MARCO REFERENCIAL</b> .....	11
<b>2.1 Marco teórico</b> .....	11
<b>2.1.1. Redes de colas</b> .....	11
<b>2.1.2. Agentes inteligentes</b> .....	15
<b>2.1.3. Sistemas de inferencia difusos</b> .....	18
<b>2.2 Estado del arte</b> .....	22
<b>CAPITULO III: PROPUESTA</b> .....	41
<b>3.1 Notación general de redes de colas</b> .....	41
<b>3.2. Diseño metodológico</b> .....	42
<b>CAPITULO IV: CASO DE ESTUDIO</b> .....	48
<b>4.1. Descripción general del caso de estudio</b> .....	48
<b>4.2. Fase I: Modelación de datos de entrada</b> .....	52
<b>4.2.1. Distribución de probabilidad del tiempo entre llegadas</b> .....	52
<b>4.2.2. Distribución de probabilidad del tiempo de servicio</b> .....	66
<b>4.2.3. Modelación de la transición entre etapas de proceso</b> .....	74
<b>4.2.4. Construcción del modelo de simulación discreta</b> .....	83
<b>4.3. Fase II: Modelación de conjuntos difusos para variables de desempeño</b> .....	87
<b>4.4. Fase III: Definición de la arquitectura del sistema multi-agente</b> .....	92
<b>CAPITULO V: CONCLUSIONES Y RECOMENDACIONES</b> .....	101
<b>BIBLIOGRAFIA</b> .....	103
<b>ANEXOS</b> .....	108

## Lista de tablas

<b>Tabla 1</b> Procesos metodológicos y productos esperados para cada objetivo específico del proyecto de investigación.....	8
<b>Tabla 2</b> Ámbito de aplicación de las técnicas de análisis de redes de colas.....	13
<b>Tabla 3</b> Aplicabilidad de las técnicas de análisis de redes de colas según características del sistema clasificadas por notación de KENDALL .....	14
<b>Tabla 4</b> Caracterización de agentes según rol y condiciones de operación .....	16
<b>Tabla 5</b> Artículos de investigación seleccionados y relacionados con líneas de espera y programación de tareas .....	22
<b>Tabla 6</b> Artículos de investigación seleccionados y relacionados con sistemas multi-agente e inteligencia artificial .....	27
<b>Tabla 7</b> Comparación de características principales sistemas de inferencia difusos tipo Mamdani y TSK.....	45
<b>Tabla 8</b> Funciones de elementos incluidos en arquitectura del sistema multi-agente propuesto...	46
<b>Tabla 9</b> Esquema de codificación estados finales e intermedios de la solicitud .....	50
<b>Tabla 10</b> Pruebas estadísticas requeridas para la identificación de independencia y homogeneidad de los datos de entrada.....	54
<b>Tabla 11</b> Errores en el contraste de hipótesis estadísticas.....	55
<b>Tabla 12</b> Resumen características principales pruebas de bondad de ajuste seleccionadas .....	56
<b>Tabla 13</b> Estadísticos de prueba bondad de ajuste .....	57
<b>Tabla 14</b> Tabla de frecuencias observadas y esperadas acumuladas prueba de bondad de ajuste Kolmogorov-Smirnov.....	58
<b>Tabla 15</b> Tabla de frecuencias observadas y esperadas prueba de bondad de ajuste ji-cuadrado..	60
<b>Tabla 16</b> Calculo de distribuciones exponencial acumulada para $Y_i$ y $Y_{n+1-i}$ prueba de bondad de ajuste Anderson - Darling.....	61
<b>Tabla 17</b> Calculo de p-valores para ajuste a distribución exponencial del tiempo entre llegadas...	62
<b>Tabla 18</b> Secuencia del número de corridas arriba y abajo observadas en los tiempos entre llegadas sucesivas.....	64
<b>Tabla 19</b> Secuencia del número de corridas arriba y abajo de la media observadas en los tiempos entre llegadas sucesivas.....	65
<b>Tabla 20</b> Varianza de tiempos de servicio por etapa de proceso (estrato) .....	67
<b>Tabla 21</b> Valores calculados de error muestral y tamaño de muestra por etapa .....	67
<b>Tabla 22</b> Calculo de p-valores para ajuste a distribución exponencial del tiempo de servicio etapa "Pertinencia" .....	68
<b>Tabla 23</b> Comparación p-valores ajuste a distribuciones con propiedad de distribución de carencia de memoria .....	70
<b>Tabla 24</b> Resultado consolidado prueba de independencia corridas arriba y abajo para los tiempos de servicio por etapa.....	71
<b>Tabla 25</b> Resultado consolidado prueba de independencia corridas arriba y debajo de la media para los tiempos de servicio por etapa .....	71
<b>Tabla 26</b> Resultados aplicación de pruebas de bondad de ajuste para validar supuesto de normalidad de los tiempos de servicio .....	72

<b>Tabla 27</b> Resultados aplicación de la prueba de diferencia de medias de los tiempos de servicio por etapa de proceso y clase de servicio.....	73
<b>Tabla 28</b> Matriz de transición periodo 3.....	77
<b>Tabla 29</b> Probabilidades de la transición de estados en $P^{19}$ .....	77
<b>Tabla 30</b> Cálculo de probabilidades del vector de estado estable.....	79
<b>Tabla 31</b> Calculo de valores de recurrencia basados en probabilidades de estado estable de la matriz de transición.....	79
<b>Tabla 32</b> Probabilidades de la transición de estados en $P^{20}$ .....	80
<b>Tabla 33</b> Determinante de la matriz P .....	81
<b>Tabla 34</b> Eigenvalores del polinomio característico de la matriz de transición P.....	82
<b>Tabla 35</b> Bloques y funciones de la librería Simevents® utilizadas en el modelo de simulación .....	83
<b>Tabla 36</b> Tamaños de muestra obtenidos con las diez corridas iniciales del modelo de simulación.....	85
<b>Tabla 37</b> Tamaños de muestra calculados para confiabilidad estadística del 95% .....	85
<b>Tabla 38</b> Funciones de pertenencia seleccionadas para obtención de parámetros de modelación.....	87
<b>Tabla 39</b> Parámetros aproximados funciones de pertenencia etapa de servicio radicación .....	88
<b>Tabla 40</b> Parámetros aproximados funciones de pertenencia etapa de servicio pertinencia .....	88
<b>Tabla 41</b> Reglas difusas configuradas para controladores Mamdani (radicación y pertinencia) .....	90
<b>Tabla 42</b> Funciones implementadas en entorno de simulación por cada tipo de agente.....	92
<b>Tabla 43</b> Resultados prueba de diferencia de medias sobre el throughput promedio por etapa para los algoritmos de enrutamiento Round-Robin, Equiprobable y Sistema Multi-Agente (MAS).....	97
<b>Tabla 44</b> Resultados ANOVA sobre el efecto en el throughput promedio por etapa para los algoritmos de enrutamiento Round-Robin, Equiprobable y Sistema Multi-Agente (MAS).....	98
<b>Tabla 45</b> Medias muestrales del throughput generado por algoritmo de enrutamiento .....	99

## Tabla de figuras

<b>Figura 1</b> Estructura básica de una red de colas.....	11
<b>Figura 2</b> Estructura básica de un controlador difuso tipo Mamdani .....	20
<b>Figura 3</b> Estructura básica de un controlador difuso tipo Sugeno.....	20
<b>Figura 4</b> Esquema de la notación utilizada para la modelación de sistemas de redes de colas.....	41
<b>Figura 5</b> Esquema general de la metodología para la construcción del sistema multi-agente .....	42
<b>Figura 6</b> Arquitectura del sistema multi-agente propuesto .....	46
<b>Figura 7</b> Comportamiento mensual de solicitudes de servicios NO POS instaurados ante la EPS caso de estudio.....	49
<b>Figura 8</b> Protocolos de enrutamiento de transacciones del sistema de proceso de referencia .....	51
<b>Figura 9</b> Comportamiento del tiempo entre llegadas sucesivas.....	58
<b>Figura 10</b> Modelación del tiempo de servicio de la etapa pertinencia.....	70
<b>Figura 11</b> Esquema de configuración para la simulación de los recursos de cada etapa .....	84
<b>Figura 12</b> Funciones de pertenencia generadas mediante algoritmo fuzzy c-means .....	87
<b>Figura 13</b> Ajuste aproximado de funciones de pertenencia teóricas seleccionadas .....	89
<b>Figura 14</b> Funciones de pertenencia del controlador Mamdani etapa pertinencia .....	89
<b>Figura 15</b> Interfaz gráfica para observación de reglas difusas configuradas .....	90
<b>Figura 16</b> Superficie de respuesta controlador Mamdani etapa radicación.....	91
<b>Figura 17</b> Arquitectura implementada para el sistema multi-agente.....	92
<b>Figura 18</b> Comparación del throughput simulado en el escenario 1 .....	95
<b>Figura 19</b> Comparación del throughput simulado en el escenario 2 .....	96
<b>Figura 20</b> Comparación del throughput simulado en el escenario 3 .....	96

## CAPITULO I: INTRODUCCIÓN

### 1.1. Antecedentes

En sistemas de la vida real del tipo manufactura, servicios de salud, cómputo y comunicación, aplicaciones web con configuraciones distribuidas y calidad de servicio, el servicio al cliente no se completa hasta que este ha sido atendido de manera secuencial (o en serie) en diferentes etapas, configurando alrededor de cada una de ellas, líneas de espera (o colas) cuyos requerimientos son medidos en términos de tiempos de respuesta, throughput, “disponibilidad y seguridad” (Cruz, 2012). Una red de colas se define como un conjunto de recursos interrelacionados que en su operación se caracterizan por qué, “las entradas y salidas de tareas de los nodos no tienen que estar sincronizadas y pueden ser utilizados de manera simultánea” (Pazos, 2003).

Los efectos de la interacción entre diferentes etapas propias de procesos de manufactura y servicios como los citados anteriormente, inducen ciclos de realimentación del sistema como un todo y en consecuencia, hacen al sistema complejo por naturaleza, considerando adicionalmente que hay no-linealidad debido a la variabilidad inherente en los procesos (mezcla de productos, disciplinas de ruteo entre etapas, fallas aleatorias de equipos y tiempos de llegada-procesos estocásticos). Dichos sistemas pueden ser considerados como redes generales de colas y como tal “no existe un método analítico que este habilitado para describir eficientemente dicha dinámica no lineal” (Rabta, 2013).

Las métricas de desempeño de interés principal en dichos sistemas son el throughput y el tiempo de ciclo. El throughput se define como la tasa a la que los trabajos, partes o transacciones son procesados en el sistema, mientras que el tiempo de ciclo se refiere al tiempo que toma a la entidad atravesar el sistema. El decisor del sistema puede “hacer uso del throughput y tiempo de ciclo para comparar la eficiencia de diferentes configuraciones del sistema (en términos de número de equipos, operadores, áreas de almacenamiento-espera, etc.) y proyectar decisiones a largo plazo sobre expansión de capacidad” (Yang, 2010).

El problema a resolver consiste en definir una metodología para formular las políticas operacionales que permitan encontrar la secuencia en la cual las transacciones pasen a través de las diferentes etapas del proceso, “tal que se considere una programación factible y óptima” (Fonseca, 2002) con respecto a los criterios de desempeño antes descritos.

Tomando como referente un caso de estudio, la presente investigación busca responder la siguiente pregunta: ¿La formulación de políticas para el enrutamiento de transacciones y control de la tasa de servicio en sistemas generales de redes de colas brinda un método

para alcanzar el máximo throughput del sistema analizado, mientras que estabiliza el desempeño del sistema en un rango amplio de variables de operación?

## 1.2. Planteamiento y delimitación del problema

¿Cuáles deben ser los componentes, técnicas analíticas y computacionales de un sistema multi-agente cuyo propósito sea formular políticas factibles de enrutamiento y priorización de transacciones que respondan a los efectos de la interacción entre etapas de sistemas de redes de colas considerando las restricciones de operación y objetivos de desempeño requeridos?

## 1.3. Hipótesis

La integración de teoría de líneas de espera, simulación discreta y agentes inteligentes permite formular políticas factibles para el enrutamiento de transacciones en sistemas de redes de colas multi-clase, mejorando su desempeño en términos de throughput y tiempo de ciclo, dado que determina la configuración de recursos necesaria para garantizar la suficiencia en la capacidad del sistema en un rango amplio de variables de operación.

## 1.4. Objetivos de la investigación

### 1.4.1. Objetivo general

Diseñar un sistema multi-agente basado en la integración de técnicas analíticas, de simulación discreta y lógica difusa para la formulación de políticas de enrutamiento de transacciones y gestión de líneas de espera en sistemas generales de redes de colas multi-clase.

### 1.4.2. Objetivos específicos

- Caracterizar el objeto de estudio para un caso de aplicación (sistema de atención a usuarios en trámites de autorizaciones de servicios de salud), respecto a su configuración y medidas de desempeño.
- Establecer las condiciones y restricciones para la aplicación de las técnicas analíticas y computacionales requeridas para el diseño del sistema multi-agente.
- Diseñar el sistema multi-agente que permita la formulación de políticas de enrutamiento y servicio, aplicándolas al sistema de servicio de referencia.
- Validar escenarios de funcionamiento del sistema una vez identificadas las restricciones, reglas y variables de operación que afectan el desempeño y estabilidad del sistema.

## 1.5. Marco metodológico

Como base fundamental para el diseño del sistema multi-agente, se asume que el ámbito de aplicación de las técnicas a utilizar es equivalente en ambientes de manufactura y servicios, de características de configuración del sistema tipo redes generales de colas, cuyo objetivo es maximizar la tasa a la cual las entidades (trabajos o transacciones) son procesadas de manera secuencial. En la tabla 1, se asocian las actividades propuestas para el logro de cada uno de los objetivos específicos previstos para el desarrollo de la investigación.

**Tabla 1** Procesos metodológicos y productos esperados para cada objetivo específico del proyecto de investigación

Objetivo específico	Proceso metodológico	Producto esperado
<p>Caracterizar el objeto de estudio para un caso de aplicación (sistema de atención a usuarios en trámites de autorizaciones de servicios de salud), respecto a su configuración y medidas de desempeño.</p>	<ul style="list-style-type: none"> <li>• Recolección de datos históricos de mediciones de tasa de llegada-servicio, throughput y tiempo de ciclo respecto a objetivos de servicio.</li> <li>• Análisis de datos utilizando técnicas de estadística descriptiva e inferencial.</li> </ul>	<ul style="list-style-type: none"> <li>• Modelo conceptual del sistema tomado como caso de estudio.</li> <li>• Representación analítica del sistema de estudio usando teoría de redes de colas.</li> <li>• Modelación de transición entre estados-etapas de servicio empleando cadenas de Markov.</li> <li>• Modelo de simulación discreta del funcionamiento del sistema de referencia.</li> </ul>
<p>Establecer las condiciones y restricciones para la aplicación de las técnicas analíticas y computacionales requeridas para el diseño del sistema multi-agente.</p>	<ul style="list-style-type: none"> <li>• Revisión bibliográfica en bases de datos especializadas, seleccionando artículos con fecha de publicación superiores al año 2009.</li> <li>• Definición de objetivos, restricciones, reglas y supuestos de la metodología.</li> </ul>	<p>Selección de las técnicas de inteligencia artificial aplicables de acuerdo al ámbito de funcionamiento del sistema de referencia.</p>

Objetivo específico	Proceso metodológico	Producto esperado
Diseñar el sistema multi-agente que permita la formulación de políticas de enrutamiento y servicio, aplicándolas al sistema de servicio de referencia.	<ul style="list-style-type: none"> <li>• Diseño del sistema multi-agente de asignación de enrutamiento de transacciones</li> <li>• Implementación del modelo computacional a través de software especializado, para la programación del sistema diseñado</li> </ul>	Modelo computacional del sistema multi-agente
Validar escenarios de funcionamiento del sistema una vez identificadas las restricciones, reglas y variables de operación que afectan el desempeño y estabilidad del sistema.	Contraste estadístico del comportamiento del modelo computacional del algoritmo diseñado, respecto a políticas de gestión de líneas de espera (disciplinas de cola).	<ul style="list-style-type: none"> <li>• Tablas de resultados del análisis de varianza ANOVA</li> <li>• Conclusiones sobre el desempeño del modelo respecto a las políticas actuales de operación.</li> </ul>

Fuente: Elaboración propia

## 1.6. Justificación

La creciente necesidad de mejorar la respuesta ante cambios propios y externos debidos a la interacción entre etapas inherentes de los sistemas de manufactura y servicios, ha incentivado la búsqueda formal de técnicas que permitan representar la programación de tareas de sus procesos a través de estructuras simples e integradas con capacidad de adaptación frente a situaciones complejas en las que intervienen una o más variables que deben ser consideradas por el decisor del sistema.

El presente proyecto de investigación busca aportar a los desarrollos de técnicas analíticas, de simulación y agentes inteligentes integrados para la modelación del problema de programación de tareas (o “scheduling”) en ambientes de servicio, tomando como base los avances y resultados obtenidos con metodologías de similar enfoque y estructura, aplicadas en ambientes de manufactura.

La aplicación del aporte teórico y metodológico, se realizará tomando como base la información de un caso de estudio suministrado por una Entidad Promotora de Salud - EPS de Colombia, sobre un sistema de servicio administrativo cuyas características de funcionamiento guardan especial similitud con los sistemas de manufactura descritos en las más recientes investigaciones realizadas en lo que a programación de tareas se refiere, incluyendo dentro de su ámbito de aplicación, otros sistemas de servicio como lo son las redes de comunicaciones y en general, todos aquellos sistemas cuyo resultado hacia el

cliente, dependa de la interacción de una o varias fases sucesivas de servicio. Los resultados del proyecto de investigación serán compartidos con la Entidad que suministró la información, con el propósito de la posterior implementación del sistema propuesto, logrando con ello reducir los tiempos de proceso de las solicitudes recibidas por el sistema de servicio, frente a la satisfacción de los Usuarios y el cumplimiento de la normatividad vigente. El desarrollo del proyecto de investigación se enmarcara dentro de la línea de investigación del grupo Sistemas Expertos y Simulación - SES de la Universidad Distrital Francisco José de Caldas.

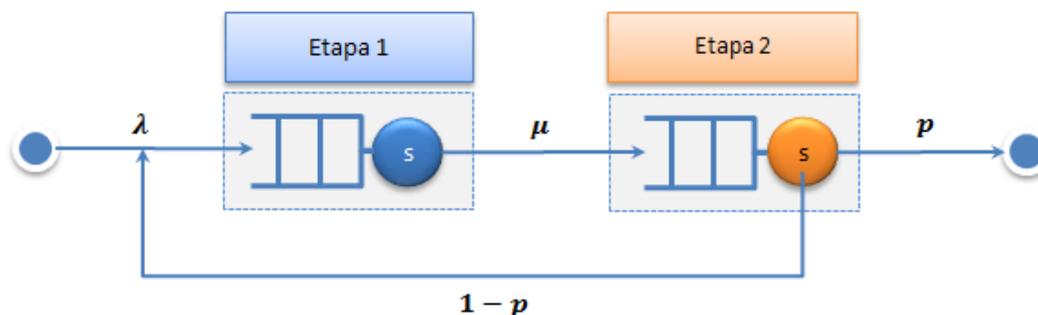
## CAPITULO II: MARCO REFERENCIAL

En el presente capítulo se realiza una descripción del conocimiento previamente construido y relacionado con teoría de líneas de espera y agentes inteligentes. En su orden y para cada uno de los dos (2) componentes del objeto de estudio, en la sección marco teórico se presentan los lineamientos teóricos fundamentales que sustentan los desarrollos prácticos aplicables posteriores y en la sección estado del arte se describe la información reciente obtenida mediante consulta directa en bases de datos especializadas.

### 2.1 Marco teórico

#### 2.1.1. Redes de colas

Una línea de espera se configura en el momento que un cliente llega a una instalación en búsqueda de un servicio y dadas sus características de capacidad se forma una cola o línea de espera. El recurso que brinda el servicio requerido (servidor) elige un cliente de la línea de espera para comenzar a prestar el servicio. Al culminar el servicio del cliente actual, se repite el proceso de elegir un nuevo cliente. Las redes de colas (o de líneas de espera) se configuran cuando el servicio al cliente se completa por su paso en diferentes etapas donde es atendido de manera secuencial (ver figura 1).



Variable	Descripción
$\lambda$	Tasa de llegada o tiempo entre llegadas por tipo de solicitud
$\mu$	Tasa de servicio
$s$	Cantidad de recursos por estación/etapa
$p$	Probabilidad de enrutamiento entre estaciones de trabajo

Fuente: Elaboración propia

**Figura 1** Estructura básica de una red de colas

En los modelos de líneas de espera incluidas las redes de colas, las llegadas y los tiempos de servicio se presentan en términos de distribuciones de probabilidad. Aunque dichas

distribuciones son los “factores principales para el análisis de líneas de espera” (Gupta, 2013), también se deben considerar otros factores como lo son:

- Características de las estaciones de servicio, su configuración y los protocolos de enrutamiento que determinan el flujo de transacciones entre estaciones.
- Número de servidores en cada etapa.
- Protocolo de servicio implementado en cada estación, que generalmente implican la prioridad dada a cada “clase” (o tipo) de transacción.
- Distribución del tiempo de servicio por clase de transacción, servidor y estación.
- Proceso de llegada especificando: los tiempos entre llegadas, número y destino de llegadas por instante de tiempo (época).
- Tamaño de la sala de espera por cada estación. Cuando el tamaño de la sala de espera es limitado, algunas de las transacciones evitan la congestión en las estaciones anteriores causando bloqueo en las estaciones siguientes.

En sentido general, una red de colas debe definirse en términos de:

- Tasas de llegada y de servicio
- Probabilidad de enrutamiento o proporción en que las clases de clientes son transferidos secuencialmente de una etapa de servicio a otra.

Dentro del análisis de un sistema de redes de colas debe considerarse que la probabilidad de enrutamiento induce ciclos de realimentación que aumentan la complejidad en la modelación y comprensión de este tipo de sistemas. Puesto que la red de colas es un sistema de nodos que interactúan, la operación de cada nodo y el “enrutamiento” puede depender de lo que esté ocurriendo a lo largo de la red, dada esta dependencia, puede ocurrir alguno o combinación de los siguientes fenómenos:

- Procesamiento paralelo o síncrono de transacciones en varios nodos.
- Alternancia en el enrutamiento de transacciones para evitar la congestión (o interferencia).
- Acelerar o desacelerar la tasa de procesamiento en los nodos siguientes que pueden estar ociosos o congestionados.
- Las transacciones se bloquean desde la entrada a una etapa específica de la red cuando dicha etapa no está en la capacidad de procesar más transacciones.

Una característica fundamental de las redes de colas es si estas son abiertas, cerradas o mixtas. Si la red permite la salida de clientes de la red luego de circular de forma predeterminada por los nodos requeridos para completar el servicio se considera una red abierta, en caso contrario se considera una red cerrada. La red puede recibir diferentes

clases de clientes (multi-clase) y a su vez puede ser abierta para ciertas clases de clientes y cerrada para otras, en cuyo caso puede considerarse como una red de colas mixta.

### 2.1.1.1. Técnicas de análisis de sistemas de redes de colas

Dependiendo del propósito de la modelación de la red de colas, la técnica de solución es seleccionada siguiendo como criterio la precisión del resultado esperado respecto a los supuestos de funcionamiento del sistema. Dado que la terminología utilizada para clasificar los métodos de análisis no es precisa, Baldwin et al. (2.003) propone su clasificación en dos (2) tipos a saber: Exactos y Aproximados. Dentro de las técnicas de análisis clasificadas como Exactas se identifican, las redes de Jackson y redes BCMP (Baskett et. al., 1.975). Por otro lado se identifican como técnicas Aproximadas: Mean Value Analysis (MVA) y Equilibrium Point Analysis (Baskett, 1.975). En la Tabla 2 se presenta el ámbito de aplicación de las técnicas de análisis anteriormente descritas, especificando los tipos de red y de clientes que pueden ser modelados con precisión (respecto a los supuestos de cada método) utilizando cada técnica. Se incluye la técnica denominada descomposición paramétrica de Kingman [12] considerando su aporte para el modelamiento de tiempos de flujo en sistemas de redes de colas.

**Tabla 2** Ámbito de aplicación de las técnicas de análisis de redes de colas

Técnica de análisis	Tipo de red	Tipo de cliente
Redes de Jackson	Abierta, Cerrada	Mono-clase
BCMP (Tipo I, II, III, IV)	Abierta, Cerrada, Mixta	Multi-clase
Descomposición paramétrica de Kingman	Abierta, Cerrada, Mixta	Multi-clase, Multi-clase con reintento
Mean Value Analysis	Abierta, Cerrada	Mono-clase
Equilibrium Point Analysis (EPA)	Abierta, Cerrada, Mixta	Multi-clase, Multi-clase con reintento

Fuente: Elaboración propia

En la Tabla 3, se relaciona en ámbito de aplicación de las técnicas de análisis descritas respecto a las características de los sistemas objeto de modelación con estos enfoques en términos de notación de Kendall, cuya codificación guarda la estructura de (1):

$$(1/2/3) (4/5/6) \quad (1)$$

Donde:

- 1, refiere al proceso de llegada que puede ser Poisson (M), determinista (D) o tasa de llegadas con distribución de probabilidad diferente a Poisson (G). Si el sistema

de redes de colas procesa diferentes clases de clientes (i) por convención, se asocia un subíndice que especifica la clase i (p. ej.:  $M_i$ , indica que el proceso de llegada es Poisson con diferentes clases de clientes).

- 2, representa el proceso de servicio que puede ser M, D o G
- 3, representa la cantidad de servidores por etapa de proceso en la red (1 a S).
- 4, representa la capacidad del sistema (finita o infinita).
- 5, refiere el orden de prioridad en la que los clientes en cola son atendidos (FCFS = First Come First Served, PS = Processor Sharing, SC = Service Center, LCFS = Last Come First Served. Entre otros.)
- 6, representa el tamaño de la población desde donde ingresan los clientes al sistema para ser atendidos

**Tabla 3** Aplicabilidad de las técnicas de análisis de redes de colas según características del sistema clasificadas por notación de KENDALL

Técnica de análisis	(1/2/3)	(4/5/6)
Redes de Jackson	(M/M/s)	( $\infty$ /FCFS/ $\infty$ )
BCMP (Tipo I)	( $M_i/M_i/s$ ) ( $M_i/G_i/s$ )	( $\infty$ /FCFS/ $\infty$ )
BCMP (Tipo II)	( $M_i/M_i/s$ ) ( $M_i/G_i/s$ )	( $\infty$ /PS/ $\infty$ )
BCMP (Tipo III)	( $M_i/M_i/s$ ) ( $M_i/G_i/s$ )	( $\infty$ /SC/ $\infty$ )
BCMP (Tipo IV)	( $M_i/M_i/1$ ) ( $M_i/G_i/1$ )	( $\infty$ /LCFS/ $\infty$ )
Descomposición paramétrica de Kingman	( $G_i/G_i/s$ )	( $\infty$ /FCFS/ $\infty$ )
Mean Value Analysis	(M/M/s)	( $\infty$ /FCFS/ $\infty$ )
Equilibrium Point Analysis (EPA)	( $M_i/M_i/s$ )	( $\infty$ /FCFS/ $\infty$ )

Fuente: Elaboración propia

La simulación discreta, puede considerarse uno de los enfoques más populares para el análisis de redes de colas, puesto que incluye en la modelación del sistema bajo estudio, detalles que no son considerados por los modelos analíticos o que son incorporados como supuestos, bajo la premisa que la precisión del modelo representado depende directamente de la calidad de la información de entrada. En el ámbito de manufactura el autor Jain et. al. (2.007) desarrolla un método iterativo usando MVA, para predicción de desempeño en sistemas flexibles de manufactura con múltiples dispositivos de manejo de materiales. Las dos (2) configuraciones de la red de colas modeladas con el algoritmo

iterativo propuesto, demuestran mejoras respecto al throughput, tiempo medio de servicio y tiempo medio de espera, respecto a la configuración previa de los dispositivos de manejo de materiales; utiliza un controlador neuro-difuso para comparar las medidas de desempeño obtenidas con MVA, demostrando la coherencia entre los resultados de ambas técnicas, dando la base para la automatización del sistema utilizando soft-computing. Cruz (2.009) examina el problema de maximizar el throughput en redes de colas con tiempo de servicio general, buscando la reducción del número total de espacios de espera y la tasa de servicio, a través de algoritmos genéticos multi-objetivo que permitan encontrar una solución factible a la necesidad de mejorar el servicio dado el conflicto "natural" entre el costo y el throughput. Yang et. al. (2.012) desarrolla un modelo de función de transferencia "híbrido" que combina análisis estadístico, simulación y análisis de líneas de espera, tomando como valores de entrada la tasa de trabajos al sistema, las variables de desempeño throughput y trabajo en proceso; el resultado final del modelo obtenido expone la falta de fidelidad de los modelos analíticos pre-establecidos para este tipo de sistemas. En el ámbito de servicios el autor Nassar et. al. (2.013) propone un sistema basado en Redes Neurales Difusas (Fuzzy Neural Networks) para resolver el problema de balanceo de carga en redes ad-hoc, consiguiendo un balanceo dinámico de carga que afronte los cambios continuos en la red.

El uso de redes neuronales artificiales Zadeh et. al. (2.011) ha demostrado optimizar la modelación y simulación de sistemas de espera, puesto que bajo este esquema pueden incluirse restricciones del sistema y objetivos de desempeño deseados, obteniendo flexibilidad y la capacidad de tratar la complejidad y no linealidad asociada con la modelización de sistemas de redes de colas. El uso del agentes inteligentes como un paradigma complementario a las técnicas de inteligencia artificial aplicadas a redes de colas, ha cobrado importancia en investigaciones como las realizadas por Araujo et. al. (2.004) en la cual, propone un enfoque de agentes autónomos e inteligentes orientados a metas y "scheduling de ganancia de referencia difusa" (Fuzzy Reference Gain-Scheduling - FRGS), que tiene la propiedad de adaptarse a sí mismo ante cambios en el ambiente, de modo que pueda emular el razonamiento humano. El agente utiliza información de entrada (o de referencia) como mecanismo para determinar en comportamiento (del sistema de control tomado como aplicación) de modo que se puedan lograr los objetivos deseados (meta).

### **2.1.2. Agentes inteligentes**

Las tecnologías basadas en agentes proveen una forma para contextualizar sistemas complejos y dinámicos, así como comprender la interacción social de entidades independientes actuando, aprendiendo y evolucionando de manera separada en respuesta a interacciones y estímulos de su ambiente local. Un agente inteligente percibe

el ambiente a través de sensores y actúa sobre el ambiente basado en un conjunto de reglas, a través de efectores o actuadores [18] lo cual le permite ser conceptualizado como un sistema adaptativo y auto-organizado frente a comportamientos complejos y estrategias específicas.

Los agentes inteligentes deben ser capaces de “actuar de forma autónoma y flexible en su entorno” Wooldridge et. al. (1.995), entendiéndose por flexibilidad:

- Autonomía y adaptabilidad, actúa y ajusta su comportamiento basándose en su experiencia.
- Reactividad, responde ante los cambios que percibe del entorno.
- Pro-actividad, es capaz de intentar el logro de sus propios objetivos y se anticipa a los requerimientos del decisor del sistema.
- Interacción social, se comunica con otros agentes y recursos del sistema a través de un lenguaje específico.

Los agentes se pueden caracterizar de acuerdo a los “atributos o propiedades que posean (y que definen su comportamiento) frente a la resolución de un determinado problema” (Botti, 2000). En la tabla 4 se presenta un esquema de caracterización de agentes definido por Kinsel (2005) de acuerdo a su función y propiedades; dicha clasificación no implica que las propiedades en un agente sean excluyentes.

**Tabla 4** Caracterización de agentes según rol y condiciones de operación

<b>Clasificación agente</b>	<b>Condiciones de operación</b>
Autónomo	No requiere intervención del Usuario
Adaptativo	Puede cambiar su comportamiento basado en experiencias pasadas
Basado en metas	No requiere que las posibles soluciones al problema sean suministradas
Reactivo	Requiere que las posibles soluciones al problema sean especificadas desde el principio No cambia/ajusta su comportamiento basado en la experiencia
Reunión de información	Recolecta, filtra y clasifica información
Interface	Requiere asistencia del Usuario

Fuente: Elaboración propia

Los agentes que son desarrollados de forma dependiente es decir, “interactúan como entidades que constituyen un sistema son denominados sistema multi-agente” (Huhns et. al., 1.998). La interacción entre los agentes puede ser diseñada en forma de “mensajes,

solicitudes, negociación o implementación de cambios en sus ambientes comunes” (Chen, 2.011). La interacción puede ser soportada habilitando a los agentes para coordinar sus actividades y comportamientos, cooperar para el logro de objetivos comunes o competir para mejorar el logro de sus objetivos individuales. En los sistemas multi-agente, un agente visto como una entidad autónoma, puede operar buscando maximizar su propio beneficio, lo cual redundaría en posibles fallas y conflictos con los objetivos individuales de los demás integrantes del sistema, generando fenómenos de competición. Frente a este fenómeno surge la definición de métodos de negociación, a través de los cuales un grupo de agentes comunican sus objetivos, y realizan acuerdos mutuos para la asignación de recursos limitados, mediante concesiones o búsqueda de alternativas aceptadas por todos los participantes. Dentro de los métodos generalmente utilizados se destacan: “Contract Net Protocol” (Smith, 1.979), “Audiciones” (Wooldridge, 2.002), “Teoría de juegos” (Sandholm, 2.002) y Argumentación (Jennings, 2.002).

El paradigma de agentes inteligentes dispone de ventajas potenciales para la programación de tareas:

- El paradigma de agentes utiliza computación paralela lo cual puede proveer sistemas de programación de tareas con alta eficiencia y robustez.
- Es posible integrar los procesos de planeación y programación de tareas considerando adicionalmente que de manera simultánea dichos procesos puedan ser optimizados.
- Los agentes de recursos pueden estar conectados directamente a los dispositivos físicos (sensores) que representan, de tal manera que puede lograrse “re-programación de tareas” en tiempo real, obteniendo con ello, “sistemas de manufactura con alta tolerancia a fallos” (Shen, 2.011).
- Para los recursos compartidos en los procesos puede lograrse una “compensación” del desempeño local hacia la mejora del desempeño global, ello sobre la base de la programación de tareas cooperativa.

La tecnología de agentes inteligentes puede ser combinada con otras técnicas de inteligencia artificial para obtener niveles de aprendizaje y toma de decisiones. Sharifian et. al. (2.011) desarrolla un algoritmo de balanceo de carga que clasifica, asigna y enruta peticiones en redes de computación, basándose en la predicción de la utilización y disponibilidad de los servidores utilizando redes neuronales artificiales y ANFIS. Madureira et. al (2.014) diseña un mecanismo de negociación de sistemas multi-agente para scheduling dinámico basado en inteligencia social y colectiva, soportado en técnicas computacionales que usan ideas de comportamientos sociales de insectos y otros sistemas biológicos (inteligencia de enjambre). En ambiente computacional, el mecanismo propuesto demuestra ser una alternativa factible para minimizar el “makespan” y

maximizar la ocupación de máquina. Metawei et. al. (2012) desarrolla un sistema distribuido multi-agente para lograr en balanceo de carga dinámico en redes de comunicaciones. El balanceo de carga lo logra identificando mediante una regresión lineal múltiple que garantiza el cumplimiento de los objetivos de migración de las transacciones entrantes hacia los nodos de la red que por su nivel de ocupación pueden asegurar un mayor flujo de transacciones procesadas.

### **2.1.3. Sistemas de inferencia difusos**

La lógica difusa es una rama de la inteligencia artificial que permite crear sistemas basados en el pensamiento humano que estén en capacidad de inferir ante situaciones con información imprecisa. La teoría de conjuntos difusos (fuzzy sets theory) definida por Zadeh (1965) puede ser vista como una generalización de la teoría clásica de conjuntos. El punto de partida de los conjuntos difusos consiste en considerar que la certeza de una proposición es cuestión de grado, es decir, difiere de la lógica clásica o booleana dado que esta última restringe la relación de un elemento frente a un conjunto de la manera “pertenece” o “no pertenece”. En lógica difusa el grado de relación de un elemento frente a un universo de discurso, se representa mediante funciones de pertenencia (membership functions). Las funciones de pertenencia permiten representar matemáticamente una noción subjetiva como temperatura de un lugar, estatura de las personas, desempeño de un proceso frente a otros, etc. Los sistemas difusos son una representación formal de descripciones lingüísticas en forma de reglas del tipo “si - entonces” (IF-THEN). Una variable lingüística es una variable cuyos argumentos son modelados mediante conjuntos difusos que disponen de fundamentos matemáticos rigurosos para representar los conjuntos difusos y sus relaciones.

El uso de lógica difusa para el diseño e implementación de sistemas de control, se ha visto motivado como una estrategia de control automático por su contribución a los siguientes aspectos:

- Mejora de la robustez de los métodos de control, respecto a los clásicos de tipo lineal.
- Diseño e implementación simplificada de control para modelos complejos.
- Autonomía y adaptabilidad, lo cual guarda coherencia con el objetivo de los sistemas multi-agente.

Un controlador difuso está compuesto por cuatro partes principales:

*1. Interfaz de Fusificación:* mapea a escala los valores nítidos de las variables de entrada transfiriéndolos al universo de discurso sobre las etiquetas lingüísticas correspondientes a los conjuntos difusos.

2. *Base de conocimientos*: consta de una base de reglas lingüísticas para controlar la variable de interés. Dicha base de reglas caracteriza las metas de control y las proposiciones definidas por los expertos como políticas para la operación del controlador. Los métodos más frecuentes para la definición de las reglas lingüísticas incluidas en la base de conocimientos contemplan el conocimiento de ingeniería de control, acciones de control de los operadores del sistema bajo análisis y el juicio de expertos.

3. *Lógica de decisiones*: Esta incluida dentro del controlador y representa el proceso de razonamiento y acción del tomador de decisiones, basándose en implicaciones lógicas.

4. *Interfaz de defusificación*. Transforma el mapeo a escala del universo de discurso difuso hacia el nítido de modo que se obtenga la acción de control nítida a partir de una acción de control difusa. La salida de control (valor nítido) puede obtenerse a través de alguno de los siguientes métodos:

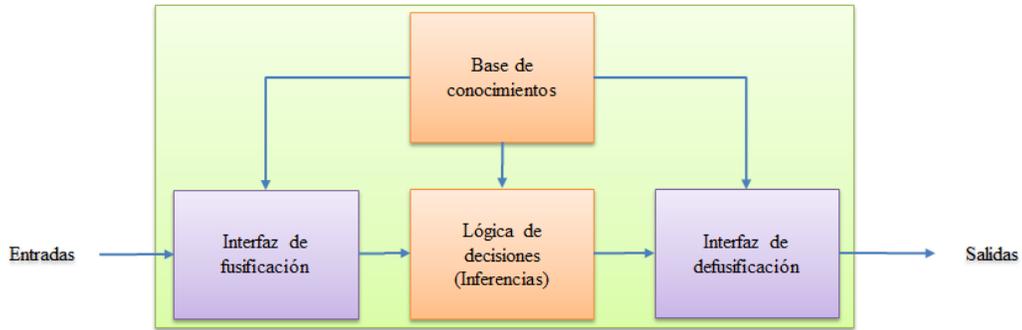
- 1) Centro de área o gravedad.
- 2) Centro máximo.
- 3) Izquierda máximo y Derecha máximo.

Cruz (2010) indica que los parámetros principales para el diseño de un controlador difuso deben considerar los siguientes aspectos:

- Estrategias de Fusificación e interpretación del operador correspondiente.
- Bases de datos: dominio discreto, normalización, partición del espacio y selección de las funciones de pertenencia.
- Base de reglas: elección de variables de entrada (antecedente) y de salida (consecuente), origen, derivación y cobertura de reglas lingüísticas.
- Lógica de decisiones: definición de la relación entre el antecedente y el consecuente en una regla lingüística (implicación difusa), interpretación de los operadores lógicos y mecanismo de inferencia.
- Estrategia e interpretación del método de defusificación.

#### **2.1.3.1. Diseño de controladores de reglas difusas tipo Mamdani**

El sistema de inferencia difusa (FIS) propuesto por Mamdani et al. (1975), fue el primero en trabajar con variables reales en el antecedente y consecuente. En un FIS tipo Mamdani, las reglas son del tipo “si - entonces”, si el FIS tiene más de una variable en el antecedente y una única en el consecuente, las reglas presentan la estructura: “Si  $X_1$  es  $A_1$  y... $X_n$  es  $A_n$ , entonces  $Y$  es  $B$ ”, donde X y Y representan las variables del antecedente y el consecuente respectivamente (ver Figura 2).



Fuente: Elaboración propia

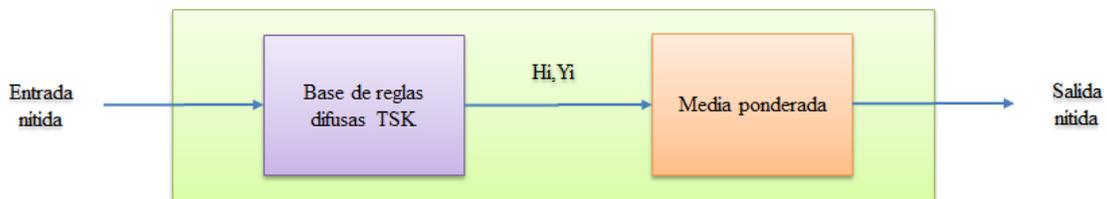
**Figura 2** Estructura básica de un controlador difuso tipo Mamdani

El diseño de un controlador difuso tipo Mamdani, considera a nivel procedimental la siguiente secuencia de actividades:

- 1) Selección de las funciones de pertenencia para el mapeo sobre el universo de discurso de las etiquetas lingüísticas (fusificación).
- 2) Definición de reglas a partir de proposiciones condicionales (“si - entonces”). El mecanismo de inferencia será una composición máx-mín.
- 3) Selección de las funciones de pertenencia para el mapeo sobre el universo de discurso nítido (defusificación), así como el método que se va a utilizar para encontrar el valor nítido resultante.

### 2.1.3.2. Diseño de controladores de reglas difusas tipo Sugeno

El sistema de inferencia difusa propuesto por Takagi y Sugeno (1.985), difiere del propuesto por Mamdani en que el consecuente está representado por una combinación lineal de las variables lingüísticas del antecedente en reglas difusas tipo TSK de la forma: “Si  $X_1$  es  $A_1$  y... $X_n$  es  $A_n$ , entonces  $Y = p_1 X_1 + \dots + p_n X_n + p_0$ ”, donde  $X$  y  $Y$  representan las variables del antecedente y el consecuente respectivamente, y los  $p_i$  representan parámetros reales. Dado que los FIS Sugeno no requieren interfaz de defusificación, la salida del FIS se obtiene como la media ponderada de las salidas individuales aportadas por cada regla (ver Figura 3).



Fuente: Elaboración propia

**Figura 3** Estructura básica de un controlador difuso tipo Sugeno

La principal desventaja del FIS tipo Sugeno consiste en que la variable del consecuente no se representa por conjuntos difusos, por esta razón no pueden constituirse como un marco de trabajo natural para representar el conocimiento del experto.

Kansal et al. (2.013) comparan la eficiencia del modelo FIS tipo Mamdani respecto al modelo Sugeno mediante su aplicación en el diseño de un control de flujo de agua para un molino de crudo característico de la industrial de cemento. Los resultados experimentales permitieron identificar que ambos tipos de FIS se desempeñan de manera similar, no obstante y a pesar de la desventaja en el no uso de funciones de pertenencia para la variable del consecuente en los FIS Sugeno, estos pueden ser integrados con redes neuronales, algoritmos genéticos y otras técnicas de optimización que permitan al controlador adaptarse al entorno. Conclusiones similares fueron obtenidas por Zaher et al (2.014), en este caso aplicado a la predicción de precios de fondos en Egipto. Los autores indican que la precisión del FIS Sugeno, puede mejorarse realizando combinaciones en las funciones de pertenencia utilizadas.

### ***2.1.3.3. Sistemas de control en lazo abierto***

En teoría de control un sistema de control de lazo abierto, se caracterizan porque la variable de salida (o controlada) no tiene efecto sobre la acción en la variable el control. Adicionalmente:

- La salida del sistema no se compara con un valor deseado o de referencia.
- A cada entrada de referencia le corresponde una condición o parámetro de operación fijo que no cambia en el tiempo.
- La exactitud de la salida depende de la calibración del controlador.
- Ante perturbaciones el sistema de control puede no cumplir adecuadamente con la función para la cual fue diseñado
- De manera general los sistemas de control en lazo abierto se utilizan en dispositivos de control secuencial donde las operaciones se realizan de una manera determinada por eventos (event driven) o por tiempo (time driven).

## 2.2 Estado del arte

En la tabla 5 se relacionan publicaciones científicas seleccionadas por su afinidad con el objeto de estudio y con el propósito de identificar los avances más recientes. Para lo anterior se tomaron como referencia documentos con fecha de publicación del año 2010 en adelante y disponibles en las bases de datos multidisciplinarias “ScienceDirect” e “IEEE Explore”. Los documentos relacionados a continuación son clasificados por el área temática principal líneas de espera, y programación de tareas, referenciando: año de publicación, título en idioma original y las técnicas utilizadas y los resultados alcanzados por los investigadores.

**Tabla 5** Artículos de investigación seleccionados y relacionados con líneas de espera y programación de tareas

Nro.	Título origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
1	Neural network metamodeling for cycle time-throughput profiles in manufacturing.  Yang (2010)	Propone un método de meta-modelación basado en redes neuronales artificiales para generar los perfiles de tiempo de ciclo (CT) y throughput (TH) en ambientes de producción mono y multi-producto.	[E1] Tasa de liberación de productos en el sistema (tiempo de ciclo). [E2] Tasa de llegada. [E3] Tasa de servicio. [E4] Mezcla de productos	<u>Líneas de espera</u> -Modelación de la capacidad aproximada del sistema e identificación de recursos cuello de botella. -Dividir la región factible de tiempo de ciclo (CT) y throughput (TH) en subregiones para ajustar y suavizar la superficie de respuesta. <u>Redes de Jackson</u> Calcular el tiempo de ciclo esperado <u>Redes neuronales artificiales</u> -Generar el ajuste de la superficie de respuesta CT-TH mediante el criterio least-square fitting, la cual fue obtenida inicialmente con la aplicación de líneas de espera.	Los resultados experimentales indican que el método propuesto disminuye el tiempo de ciclo y el throughput.
2	An integrated artificial	Proponer un esquema	[E1] Tasas de llegada	[T] Simulación discreta (C++)	Número óptimo de

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
	neural network-computer simulation for optimization of complex tandem queue systems  Zadeh et al (2011)	integrado de redes neuronales artificiales y simulación para optimización de sistemas de líneas de espera G/G/K, que sea capaz de mejorar su desempeño respecto a restricciones de producción, limitaciones del sistema y objetivos deseados.	y servicio [E2] Número máximo de servidores [E3] Tiempo permitido promedio de espera en la cola [E4] Queue policy: FIFO  [E1] Tasa de llegada [E2] Tasa de servicio [E3] Número máximo de servidores [E4] Tiempo permitido promedio de espera en la cola	Calculo del número óptimo de servidores con una única corrida para 30 casos de estudio o muestras de sistemas de espera de la vida real. El resultado de la simulación es transferido a MATLAB a través de una interfaz desarrollada en C++  <u>[T] Redes neuronales con retro-propagación</u> (MATLAB) Calcular el número óptimo de servidores en paralelo para el sistema G/G/K de referencia  <u>Arquitectura ANN</u> 4 [IL] - 45 [HL] - 4 [OL]; TF: Sigmoidea; (MAPE)	servidores en paralelo (que reemplazan los valores paramétricos del modelo de simulación)
3	Multi-objective scheduling of dynamic job shop using variable neighborhood search  Adibi et al. (2010)	Propone un método de análisis dinámico de scheduling en configuración job-shop que considera llegadas de trabajo y averías aleatorias de máquina.	[E1] Tiempo de finalización del trabajo [E2] Fecha de entrega de los trabajos. [E3] Tiempo total para procesar un grupo de trabajos. [E4] Tiempo promedio entre llegadas. [E5] Tiempo promedio entre fallas. [E6] Tiempo	<u>[T] Neighborhood search</u> Buscar la secuencia de scheduling optima local sobre los criterios makespan y tardiness. <u>[T] Red neuronal artificial</u> -Incorporar proceso de aprendizaje en el proceso de optimización considerando las variables: número de trabajos, costo promedio de la solución inicial, varianza en el tiempo de procesamiento y tiempo promedio de entrega.	El método propuesto demostró mayor eficiencia ante variaciones en las condiciones de planta, al ser comparado con otras reglas de enrutamiento definidas en la literatura.

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
			promedio para reparar.		
4	A hybrid EA for reactive flexible job-shop scheduling  Lin et al (2.012)	Propone un algoritmo evolutivo hibrido para problemas de scheduling reactivo que consiste en la combinación de un algoritmo genético y un modelo de optimización de enjambre de partículas.	[E1] Número total de máquinas. [E2] Número total de operaciones en trabajo.	<u>[T] Algoritmo hibrido evolutivo</u> -Implementa la regla de despacho SPT (Shortest Processing time), incorporando caídas y cambios en la operación de la máquina, así como modificaciones en las fechas de entrega. Durante la simulación, los parámetros relacionados con las perturbaciones (fallas) son representados como variables aleatorias.	Los experimentos conducidos con el algoritmo hibrido evolutivo demuestra versatilidad en comparación con resultados obtenidos por otros algoritmos en el contexto de problemas de gran escala.
5	Queueing model analysis and scheduling strategy for embedded multi-core SoC based on task priority  Qiu et al. (2013)	Propone un modelo de redes de colas y un método de análisis de desempeño de sistemas embebidos multi-nucleo SoC (System on Chip) basado en un mecanismo de manejo de tareas con prioridad.	[E1] Tasa de llegada de tareas. [E2] Tiempo de procesamiento de la tarea. [E3] Probabilidad de cambio de prioridad de la tarea.	<u>Modelación de redes de colas</u> Calcula el balance de flujo, de acuerdo a la tasa de llegadas, tiempo de procesamiento y prioridad de las tareas. <u>Algoritmo de scheduling adaptativo</u> -Identifica la prioridad de la tarea. -Determina si la tarea debe ser ubicada en una cola FIFO de alta prioridad. -Determina el segmento FIFO para ubicar la tarea. -Determina si por longitud del tiempo de espera, la tarea debe ser transferida. -Iterativamente, construye	Los resultados experimentales indican que el algoritmo diseñado mejora significativamente el balanceo en la asignación de tareas.

Nro.	Título origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
				redes de colas para garantizar el balance de flujo.	
6	Research on computational intelligence algorithms with adaptive learning approach for scheduling problems with batch processing machines  Noroozi et al. (2013)	Propone y desarrolla un algoritmo híbrido que integra algoritmos genéticos híbridos, simulación híbrida y optimización de enjambre de partículas, para resolver el problema de planeación de máquinas de procesamiento en lotes.	[E1] Número de trabajos. [E2] Número de máquinas. [E3] Número de lotes. [E4] Tiempo de procesamiento. [E5] Capacidad de la máquina. [E6] Costos de entrega temprana, tardía y makespan.	<u>Reglas heurísticas de despacho</u> -STPT, LTPT, SBMPT, LBMPT, EDD, MST. <u>[T] Algoritmo genético</u> -Genera soluciones óptimas locales del problema <u>[T] Hybrid simulated annealing</u> -Genera como solución inicial la mejor solución generada por el algoritmo genético. <u>[T] Particle Swarm Optimization</u> -Genera soluciones óptimas globales	Los experimentos de simulación realizados son comparados con algoritmos genéticos, optimización de enjambre de partículas y las reglas de enrutamiento STPT, LTBT, SBMPT, LBPMT, EDD, MST en el ambiente computacional del software LINGO. Los resultados evidencian mejor desempeño respecto a los algoritmos y reglas comparadas.
7	Application of queueing theory in health care: A literature review  Lakshmi et al. (2013)	Propone un sistema de clasificación de la teoría de líneas de espera aplicada a problemas prácticos de administración de servicios de salud.	[E1] Literatura de aplicaciones de líneas de espera aplicadas en administración de salud.	Categorización de técnicas de investigación de operaciones aplicables de acuerdo al objeto de aplicación (hospitales, farmacias, programación de salas de cirugía, admisión de pacientes en servicios de urgencia, etc.).	Resalta la versatilidad de la modelación vía simulación y líneas de espera, dado que ambas técnicas pueden incluir aleatoriedad y un nivel más amplio de detalle, en comparación con programación lineal y dinámica que omiten los efectos de la

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
					incertidumbre
8	An optimization based on simulation approach to the patient admission scheduling using a linear programming algorithm  Granja et al. (2014)	Presenta un enfoque de optimización basado en simulación y programación lineal, para el problema de administración de pacientes considerando su aplicación en un servicio de imagenología diagnóstica.	[E1] Actividad desarrollada por cada recurso. [E2] Tiempo promedio de duración de la tarea. [E2] Carga de trabajo de los recursos. [E4] Precedencia de las tareas.	<u>Simulación discreta</u> Modelación de la naturaleza estocástica de los tiempos de procesamiento y de la interdependencia entre recursos, para obtener la secuencia de atención de pacientes y los tiempos de inicio y finalización de la atención. <u>Simulated annealing</u> Minimizar el tiempo de espera de los pacientes según las secuencias de atención simuladas.	Los resultados obtenidos muestran una reducción promedio del 38% en el tiempo total de espera. La combinación de simulación discreta y técnicas de optimización demuestra su efectividad en la determinación de cuellos de botella.
9	Optimizing the strategic patient mix combining queueing theory and dynamic programming  Vanberkel et al. (2014)	Presenta la modelación de un hospital utilizando teoría de líneas de espera para evaluar el impacto sobre el servicio en condiciones de admisión de nuevos pacientes.	[E1] Tasa de llegada de solicitudes. [E2] Tiempo de procesamiento de solicitudes.	<u>Líneas de espera</u> Calcular el número de pacientes con tratamiento terminado y en curso. <u>Programación dinámica</u> Encontrar la secuencia más corta para agregar pacientes a los tratamientos de acuerdo a la capacidad de servicio de las etapas del proceso considerado.	Sugiere políticas públicas para la toma de decisiones sobre la prestación de servicios de salud.

Fuente: Elaboración propia

En la tabla 6 se relacionan publicaciones científicas seleccionadas para el área temática principal sistemas multi-agente e inteligencia artificial en diversos ámbitos de aplicación, identificando las técnicas utilizadas y los resultados alcanzados por los investigadores.

**Tabla 6** Artículos de investigación seleccionados y relacionados con sistemas multi-agente e inteligencia artificial

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
1	<p>A self-adaptive agent-based fuzzy-neural scheduling system for a wafer fabrication factory.</p> <p>Chen (2011)</p>	<p>Propone y desarrolla un Sistema multi-agente neuro-difuso para mejorar el rendimiento de trabajos planeados en una fábrica de productos alimenticios.</p>	<p>[E1] Información de la maquina (parámetros de capacidad máxima, método y precisión de procesamiento).</p> <p>[E2] Información del proceso (trabajos en proceso, material en inventario).</p>	<p><u>Agente de scheduling</u>                      -Informa las medidas de desempeño locales (estimación del tiempo de ciclo restante mediante mecanismo FCM-BPN).                      -Responde solicitudes de cambio en la política de scheduling enviadas por el agente coordinador.                      -Seleccionar el siguiente trabajo a ser procesado de acuerdo a una política difusa de dos (2) factores.</p> <p><u>Agente de monitorización</u>                      Recibe información de los agentes de scheduling y calcula el desempeño local de cada máquina y el global de toda la fábrica.</p> <p><u>Agente de coordinación</u>                      Decidir (por cada ruta de procesamiento), las solicitudes que deben se concedidas a cada agente de scheduling. Lo anterior tomando como referencia el logro de las metas globales.</p>	<p>Según los resultados experimentales en ambiente de simulación, el sistema multi-agente desarrollado mejora el desempeño del sistema respecto al tiempo promedio de ciclo.</p>

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
2	The dynamic transfer batch-size decision for thin film transistor-liquid crystal display array manufacturing by artificial neural network  Yang et al (2011)	Propone un modelo basado en redes neuronales artificiales como núcleo del mecanismo de toma de decisiones sobre el tamaño de lote de transferencia en ambientes de manufactura.	[E1] Tamaño de lote. [E2] WIP - Trabajo en proceso. [E3] Capacidad de las máquinas. [E4] Demanda.	<u>Simulación discreta</u> Modelación del sistema de producción y generación de escenarios de tamaño de lote de transferencia para obtener las muestras de datos para el entrenamiento y prueba del modelo de redes neuronales artificiales. <u>Redes neuronales artificiales</u> Mecanismo de decisión para determinación del tamaño de lote de transferencia en planta.	La comparación del modelo propuesto demuestra mejores resultados respecto a estrategias de lote de transferencia de amplio uso en la industria.
3	A predictive and probabilistic load-balancing algorithm for cluster-based web servers  Sharifian et al. (2011)	Propone un algoritmo de balanceo de carga que dinámicamente clasifica, asigna y enruta peticiones en función a la clase de transacción identificada.	[E1] Número de solicitudes en el sistema. [E2] Numero de recursos requeridos para atender la demanda de solicitudes.	<u>Modelo de línea de espera BCMP</u> Estimar la utilización y el tiempo promedio de respuesta de los servidores. <u>Radial Base Function Neural Network (RBFNN) y Adaptive Neuro-Fuzzy Inference System ANFIS</u> Ajustar los valores calculados con BCMP respecto a las mediciones "en-línea". <u>Probabilistic Preferred Short Job PPSJ</u> Brindar procesamiento preferente a las clases de solicitudes que tienen tiempo de servicio más corto y que al mismo tiempo tienen un alto número de peticiones en espera	El algoritmo propuesto, logra predecir la utilización y disponibilidad de los recursos, mediante un ciclo de control con realimentación.

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
4	Fuzzy neural network for dynamic load balancing of nodes for ad-hoc network using  Nassar et al. (2013)	Propone un Sistema basado en redes neuronales difusas para resolver el problema de balanceo de carga en redes de comunicaciones ad-hoc.	[E1] Longitud de la cola. [E2] Longitud de cola permitida. [E3] Throughput. [E4] Estado de la cola	<u>Fuzzy Neural Network 1 (FNN1)</u> Inferir el estado de la cola tomando como variables de entrada la longitud de la cola y la longitud de cola permitida. <u>Fuzzy Neural Network 2 (FNN 2)</u> Inferir la carga del nodo tomando como variables de entrada el throughput y el estado de la cola (esta última variable entregada por FNN1) <u>Simulación discreta (C++)</u> Evaluar la eficiencia del metodo propuesto para resolver el problema de balanceo de carga.	En ambiente de simulación, el algoritmo propuesto realiza un balanceo dinamico de carga que afronta cambios controlados en la red de comunicaciones.
5	Multi-agent approach to distributed ant colony optimization.  Ilie et al. (2013).	Presenta una arquitectura configurable distribuida de optimización de colonia de hormigas para el intercambio de mensajes asíncronos entre agentes.	[E1] Arquitectura del sistema multi-agente (nodos, cantidad de recursos, cantidad de agentes)	<u>Optimización de colonia de hormigas</u> Define la ubicación óptima de los agentes en los nodos de la red, la cual es almacenada por el agente maestro	Los resultados de la experimentación demostraron la factibilidad y escalabilidad del algoritmo propuesto en ambientes de redes de computadores.
6	IMAQSC: design and implementation of an intelligent multi-agent system for monitoring and controlling quality of cement production processes.  Mahdavi et al. (2013).	Propone un Sistema multi-agente para el control de la calidad en procesos de producción de cemento. El sistema multi-agente utiliza técnicas de inteligencia artificial (IMAQCS) basadas en reglas, que concentra la relación entre	[E1] Información de atributos y variables de calidad. [E2] Base de datos de conocimientos KBMA.	Arquitectura de agentes inteligentes: <u>Agente ejecutor de herramientas de control de calidad (QCTEA)</u> -Selecciona gráficos de control aplicables según los datos del área de la empresa en análisis. -Genera gráficos de control.	El prototipo de arquitectura implementado de IMAQCS, demostró ser de gran aporte para la monitorización, control y ajuste de la planta de cemento



Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
				<p>proceso de toma de decisión sobre scheduling incluyendo su coordinación</p> <p><u>Agente de tareas</u></p> <ul style="list-style-type: none"> <li>-Generar tiempos de procesamiento tempranos y tardíos.</li> <li>-Dividir las operaciones a través de los agentes de recursos</li> </ul> <p><u>Agente de recursos</u></p> <ul style="list-style-type: none"> <li>-Encontrar una solución óptima (ó cercana) local utilizando los algoritmos Swarm Intelligence.</li> <li>-Informar solución óptima al Agente Coordinador.</li> <li>-Utilizar SMSP (Single Machine Scheduling Problems) para la secuenciación de trabajos</li> </ul> <p><u>Agente coordinador</u></p> <ul style="list-style-type: none"> <li>-Monitorear la adaptación y recuperación de la operación ante fallas.</li> <li>-Identificar conflictos de restricciones en cada tarea</li> <li>-Integrar las soluciones individuales obtenidas por cada Agente de recursos para obtener una solución global factible.</li> </ul>	<p>inicial al problema de scheduling y entonces, aplica el mecanismo de reparación "self-healing agent" para adaptar las operaciones y generar una solución factible.</p> <p>2. El sistema aplica el sistema de negociación para las soluciones entregadas por los agentes, buscando alcanzar sus objetivos a través de la negociación</p>
			Scheduling plan	[T] Mecanismo de negociación Mejorar el plan de scheduling en términos de reducción de tiempos ociosos, tiempos de	Análisis y optimización de las soluciones locales generadas por los

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
				caída y tasa de ocupación de las maquinas.	Agentes de recursos
8	An agent-based intelligent algorithm for uniform machine scheduling to minimize total completion time.  Li et al. (2014).	Propone un algoritmo de scheduling inteligente basado en agentes llamado "ABISA", cuyo propósito consiste en robotizar el proceso de scheduling de manufactura por medio de inteligencia de máquinas. El algoritmo "ABISA" organiza los trabajos de acuerdo a la regla LPT (largest processing time-largest release date first).	[E1] Fecha de entrega del producto. [E2] Tiempo de procesamiento por máquina.	Arquitectura de agentes inteligentes: Define un agente por cada máquina encargados de pre-secuenciar, procesar, enviar y recibir los comandos de pre-secuenciación. El agente maestro coordina los agentes esclavos mediante el protocolo "token-ring", en el cual todos los equipos están conectados en una topología estrella que evita la colisión de datos.	Las soluciones obtenidas por el algoritmo "ABISA" presentan un mejor desempeño cercano al límite inferior de las heurísticas SPT-ECT y ERT-ECT.
9	Achieving dynamic load balancing through mobile agents in small world P2P networks.  Shen et al. (2014)	Presenta un método de balanceo de carga basado en agentes móviles y técnicas de agrupación de recursos para redes P2P (Peer-to-peer) no estructuradas.	[E1] Información de recursos disponibles. [E2] Capacidad de procesamiento de cada nodo. [E3] Número de total de transacciones en espera.	Propone una estrategia de agrupación de recursos en los nodos de la red balanceando la carga dentro de cada grupo, utilizando para ello tres (3) tipos de agentes: <u>Agente anfitrión (HA)</u> -Encargado de controlar los demás agentes. -Monitorea la red (grupos) en busca de nuevos recursos. <u>Agente de detección (DA)</u> Recorre cada nodo buscando congestión. <u>Agente de reconexión (RA)</u>	Los resultados de la simulación demuestran robustez, capacidad de adaptación y mejor desempeño comparada con productos de funciones similares disponibles en el mercado.

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
				Reconecta los enlaces congestionados a recursos con niveles de carga menores.	
10	Distributed key figure optimization approaches for global coordination in multi-agent systems for production control  Pantke et al (2014)	Presenta un enfoque para la coordinación global de agentes en sistemas multi-agente mediante el uso de indicadores clave de desempeño (KPI)	[E1] Recursos del proceso. [E2] Trabajos para procesar. [E3] Objetivos de desempeño del sistema	<u>Optimización de caja negra</u> -El agente coordinador recibe del agente asignado a cada máquina su identificación y las mediciones de los indicadores clave de desempeño representados en gráficas. -El agente coordinador utiliza las mediciones entregadas por los demás agentes y las utiliza como restricciones de la función objetivo global. -El agente coordinador mediante el criterio “min-max” pondera el resultado del grupo de máquinas respecto a los valores objetivo de los indicadores globales y sobre el resultado de esta comparación genera el plan de scheduling de nuevos trabajos.	Examina cinco (5) variantes del enfoque aplicado al caso de asignación de órdenes a diferentes máquinas.
11	Mathematical model and agent based solution approach for the simultaneous balancing and sequencing of mixed-model parallel two-sided	Presenta un modelo que integra el procedimiento de balanceo de línea en el problema de secuenciación utilizando agentes basados en optimización de colonia de	[E1] Tareas por línea de ensamble. [E2] Relación de precedencia de las tareas. [E3] Tiempo de procesamiento por tarea.	<u>Minimum Part Set</u> Calcula la demanda general (mayor común divisor de las demandas en la mezcla de productos). <u>Programación lineal</u> -Define como función objetivo	Los resultados del modelo propuesto en comparación con otras tres (3) heurísticas demostraron un desempeño superior

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
	<p>assembly lines.</p> <p>Kucukkoc et al. (2014).</p>	<p>hormigas.</p>	<p>[E4] Demanda por línea de ensamble.</p> <p>[E5] Estaciones de trabajo</p> <p>[E6] Periodo de planeación.</p>	<p>considerando las variables: número de estaciones de trabajo utilizadas, carga de trabajo suavizada y longitud de la línea.</p> <p>-Define como restricciones para: asignación de tareas, satisfacción de la demanda, relaciones de precedencia, capacidad regular y de estaciones multi-línea.</p> <p><u>Optimización de colonia de hormigas</u></p> <p>Identificación de una solución óptima factible para el balanceo de línea considerando la función objetivo y restricciones definidas.</p> <p><u>Sistema multi-agente</u></p> <p>Define la arquitectura de un sistema multi-agente con la estructura: Agente facilitador, Agente de planeación, Agente de secuenciación y Agente de balanceo. En conjunto los cuatro (4) tipos de agente interactúan intercambiando información y utilizando la optimización de colonia de hormigas para encontrar la</p>	<p>en términos de throughput y tiempo de ciclo.</p>

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
				mejor solución de balanceo para una secuencia de tareas dada.	
12	An intelligent simulation environment for manufacturing systems.  Ruiz et al. (2014).	Presenta un ambiente de simulación soportado en agentes para sistemas de manufactura inteligentes (SimIshop)	[E1] Características del sistema objeto de simulación	La arquitectura de SimIshop está compuesta por dos (2) capas, la capa superior representa el sistema de manufactura que es simulado por el sistema multi-agente (MSSMS), mientras que la capa inferior representa los agentes de soporte de simulación (MSSSP). Para la definición de MSSMS, SimIshop utiliza un meta-modelo basado en los estándares de arquitectura PROSA y ANEMONA que incluye características de la simulación relacionadas con la interacción interna y externa (de los agentes entre ambas capas), las estrategias de negociación, entre otras.	Comparado con entornos de simulación como arena y simio, SimIshop demuestra contribuciones considerables dado que provee al Usuario utilizar las ventajas del paradigma de agentes inteligentes por medio de interfaces de fácil usabilidad.
13	An adaptive process planning method based on features and intelligent agents for the manufacturing of large-scale parts.  Liu et al. (2015)	Propone un método adaptativo de planeación basado en características de máquinas y agentes inteligentes que permita acortar el proceso de planeación en entornos de manufactura.	[E1] Características de maquinas [E2] Características del proceso de manufactura	[T] Multi-Agent System (MAS) <b>Arquitectura del MAS</b> <u>Agente de planeación detallada</u> Decide las maquinas requeridas para el procesamiento. <u>Agente de selección de maquinas</u> -Búsqueda de las maquinas	El resultado del prototipo, demostró que la utilización de la tecnología basada en agentes acorta sustancialmente el tiempo de planeación dado que puede

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
				requeridas para el procesamiento -Completa el uso de la maquina dependiendo del grado de importancia. -Elabora un proceso de análisis para la correcta selección de la máquina. <u>Agente ontológico</u> Comunica los requerimientos a las maquinas <u>Agente de máquina - herramienta</u> Comunica la disponibilidad de las maquinas a los demás agentes.	adaptar en tiempo real los procesos a los cambios identificados en el ambiente.
14	Exploring the role of contracts to support of self-organized industrial symbiosis networks: an agent-based simulation study.  Albino et al. (2015).	Propone un esquema de contratos diseñado para garantizar que la simbiosis (en términos de sistemas adaptativos complejos que interactúan) en los procesos industriales es benéfica para todas las partes interesadas externas e internas (red industrial).	[E1] Requerimientos de manufactura. [E2] Demanda de productos finales entre organizaciones. [E3] Costo de compra de productos finales entre organizaciones. [E4] Longitud de tiempo de la relación entre las organizaciones que componen la red industrial.	Modela cada etapa de la red industrial como un agente y define una función de ajuste que representa las medidas de desempeño (o atributos) de interés de cada organización. Según el valor dado a cada medida, el agente se encarga de establecer o mantener el valor objetivo. <u>Programación no lineal</u> -Genera un problema no lineal donde la probabilidad de ruptura de la relación entre dos firmas de la red industrial, representa la función objetivo. Las restricciones están dadas	En ambiente de simulación demuestra la eficiencia de los mecanismos de contratación propuestos reflejado en la mejora de la simbiosis industrial, comparado con relaciones entre compañías sin el establecimiento de contratos.

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
				<p>por las posibles combinaciones de contratos entre agentes para garantizar relaciones (ganar-ganar).</p> <p><u>Simulación discreta</u></p> <p>-Modela la dinámica ambiental de la relación entre agentes en diferentes escenarios de tipos de contrato y con ello define una línea de base que representa el comportamiento de la red industrial en términos de incertidumbre.</p>	
15	<p>Evolvable Assembly Systems: A distributed architecture for intelligent manufacturing.</p> <p>Chaplin et al (2015).</p>	<p>Propone una nueva filosofía y arquitectura de software para sistemas de montaje evolutivos (evolvable assembly systems) basada en principios de descentralización, conocimiento de contexto y recursos inteligentes, la cual se implementa en una línea de producción farmacéutica utilizando tecnología de agentes inteligentes y servicios e distribución de datos.</p>	<p>[E1] Cambios en especificaciones del producto</p> <p>[E2] Configuración de operaciones y recursos en la línea de producción.</p> <p>[E3] Desempeño de la línea de producción.</p> <p>[E4] (PLC) Controles lógicos programables</p>	<p>Arquitectura de agentes inteligentes para control y adaptación de la configuración de la línea de producción (protocolo JADE):</p> <p><u>Agente de medio ambiente</u></p> <p>Distribución de datos relacionados con cambios en las especificaciones del producto.</p> <p><u>Agentes de adaptación y recursos</u></p> <p>-Implementar directrices de adaptación utilizando PLC.</p> <p>-Monitorear y registrar trazabilidad de partes</p> <p>-Monitorear y registrar trazabilidad del desempeño</p>	<p>Los resultados de la arquitectura implementada, demostraron la necesidad de utilizar algoritmos de aprendizaje evolutivo para un mejor gobierno, optimización de la reconfiguración (de las operaciones y equipo) así como de la flexibilidad de los sistemas de producción.</p>
16	An agent based intelligent distributed control paradigm for	Presenta el paradigma denominado Sistema de control distribuido	Información de la producción en tiempo real sobre tecnología	Arquitectura de agentes inteligentes para control de operación en ciclo cerrado	El resultado de la implementación demuestra que el

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
	manufacturing systems.  Zhang et al. (2016)	inteligente de manufactura junto con la utilización de componentes inteligentes que tienen embebido el conocimiento de los procesos de manufactura así como de su estado actual.	RFID: [E1] Características del producto. [E2] Conocimiento (tiempo de procesamiento y de transporte) [E3] Componentes físicos (máquinas y almacenamiento)	<u>Agente de maquina</u> Informa sobre el tiempo de operación y ubicación precisa de la(s) tarea(s) en ejecución. <u>Agente AGV</u> Predice el tiempo de procesamiento de las nuevas tareas y busca la combinación optima de recursos, para programar y asignar las tareas entrantes	modelo dispone de capacidad de aprendizaje, toma de decisiones y aplicación de métodos de procesamiento apropiados para afrontar perturbaciones como falla de equipos y procesamiento de órdenes de emergencia.
17	Designing a fuzzy Q-learning multi-agent quality control system for a continuous chemical production line - A case study.  Sahebjamnia et al. (2016)	Propone un sistema Fuzzy Q-learning multi-agente de control de calidad (FQL-MAQCS) para el control de una línea de producción continua de químicos.	[E1] Información de atributos y variables de calidad. [E2] Base de datos de conocimientos KRMA.	Arquitectura de agentes inteligentes: <u>Agente ejecutor de control de calidad (QCEA)</u> -Genera gráficos de control con la información recolectada a través de sensores. <u>Agente de análisis de datos de proceso (PDAA)</u> -Define reglas de control según posibles condiciones definidas en KBMA. -Evalúa reglas de control de datos. Si rechaza los datos ordena la extracción de datos nuevos. -Informa al agente de decisión departamental (DDMA) <u>Agente administración de</u>	Los resultados experimentales demuestran que FQL-MAQCS corrige oportunamente las fallas e interrupciones presentadas durante la aplicación del prototipo sobre el caso de estudio.

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
				<u>conocimiento y reglas</u> Almacena, modifica y comparte las reglas almacenadas con los agentes (PDAA) y (CDMA). Implementa un algoritmo FQL.MAQCS para afrontar situaciones imprevisibles en la línea de producción.	
18	Agent and cyber-physical system based self-organizing and self-adaptive intelligent shopfloor  Zhang et al. (2016).	Presenta un modelo de planta inteligente para la ubicación oportuna de recursos que permite reducir perturbaciones respecto a los requerimientos de la operación	[E1] Información de manufactura en tiempo real [E2] Información de la maquina (parámetros de capacidad máxima, método y precisión de procesamiento). [E3] Información del proceso (trabajos en proceso, material en inventario). [E4] Costo de producción y logística. [E5] Reglas de negocio [E6] Base de datos de conocimiento	Arquitectura de agentes inteligentes: <u>Agente inteligente de máquina</u> -Use y envío de instrucciones y detalles de operación, obtenidas por sensores. -Reorganización de tareas en espera. -Reconfiguración de recursos basado en información en tiempo real. <u>Modelo auto-organizativo</u> -Descomposición de tareas a nivel de proceso. -Verifica cíclicamente tareas entrantes. Define configuración optima de máquinas (proactive Discovery modelling) basándose en costo de producción, costo de logística, capacidad y cantidad de máquinas disponibles. <u>Modelo auto-adaptivo</u> -Implementa redes de Petri para comparar eventos en	La arquitectura y modelos de función propuestos, aplicada en un caso de estudio permitió identificar una unión ajustada de modelos inteligentes de alto nivel con recursos de máquina de bajo nivel.

Nro.	Titulo origina/Autor/Año	Objetivo	Variables de entrada	Descripción del modelo	Resultados
				tiempo real respecto al estatus de la planeación. -Reorganiza tareas inconclusas para garantizar la disponibilidad de las maquinas ante entradas de nuevos trabajos.	

Fuente: Elaboración propia

La revisión del estado del arte permite identificar que las investigaciones recientes aplicadas sobre líneas de espera, programación de tareas y agentes inteligentes, están orientadas hacia el desarrollo de modelos híbridos que combinan técnicas analíticas con técnicas de inteligencia artificial en diferentes niveles, logrando con esto la veracidad en la representación realizada por los modelos ya que la aceptación de supuestos de funcionamiento de los sistemas de producción y operación reales, son omitidos al incorporar dentro de los modelos híbridos aspectos asociados a la incertidumbre más allá incluso de la representación inherente dada por los modelos estocásticos. Específicamente en el área de agentes, los esfuerzos están orientados hacia la integración del conocimiento dado por este tipo de tecnología, con dispositivos físicos que permitan implementar mediante protocolos embebidos y de manera directa los sistemas inteligentes propuestos por varios de los autores revisados quienes han logrado su implementación en ambientes de manufactura específicamente.

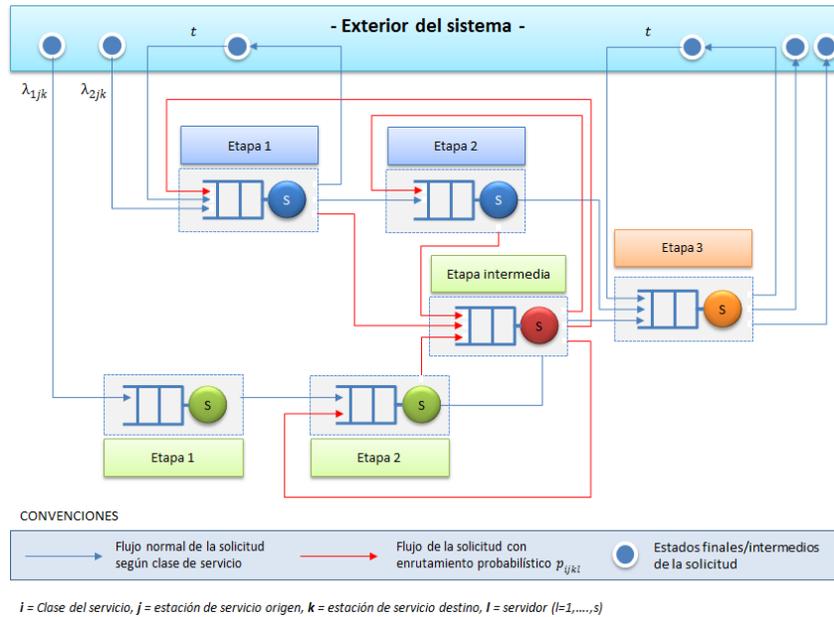
### CAPITULO III: PROPUESTA

En el presente capítulo se describen las técnicas estadísticas, de investigación de operaciones y de inteligencia artificial definidas para el desarrollo de las tres (3) fases del diseño metodológico del sistema multi-agente propuesto. Se realiza una descripción general de las funciones planteadas para cada uno de los agentes así como de las condiciones de operación y los flujos de información para el enrutamiento de transacciones y el balanceo en el uso de los recursos.

#### 3.1 Notación general de redes de colas

Teniendo en cuenta que una red de colas se configura cuando el servicio se completa en el paso por diferentes etapas donde se realizan actividades de transformación de manera secuencial, a continuación se define la notación que representara de forma unificada la configuración general de sistemas de redes de colas (ver figura 4) que se aplicara al caso de estudio de referencia así como los flujos de información. Las variables principales y secundarias a modelar dentro del diseño metodológico se identificaran bajo la estructura:

- $\lambda_{ijk}$ : tasa de llegadas de la solicitud tipo  $i$  a la etapa inicial  $j$  con destino al servidor  $l$  de la etapa  $k$ .
- $\mu_{ijl}$ : tasa de servicio de la solicitud tipo  $i$  en la etapa  $j$  con el servidor  $l$ .
- $p_{ijkl}$ : Probabilidad que la solicitud tipo  $i$  sea enrutada desde la etapa inicial  $j$  hacia el servidor  $l$  de la etapa  $k$ .



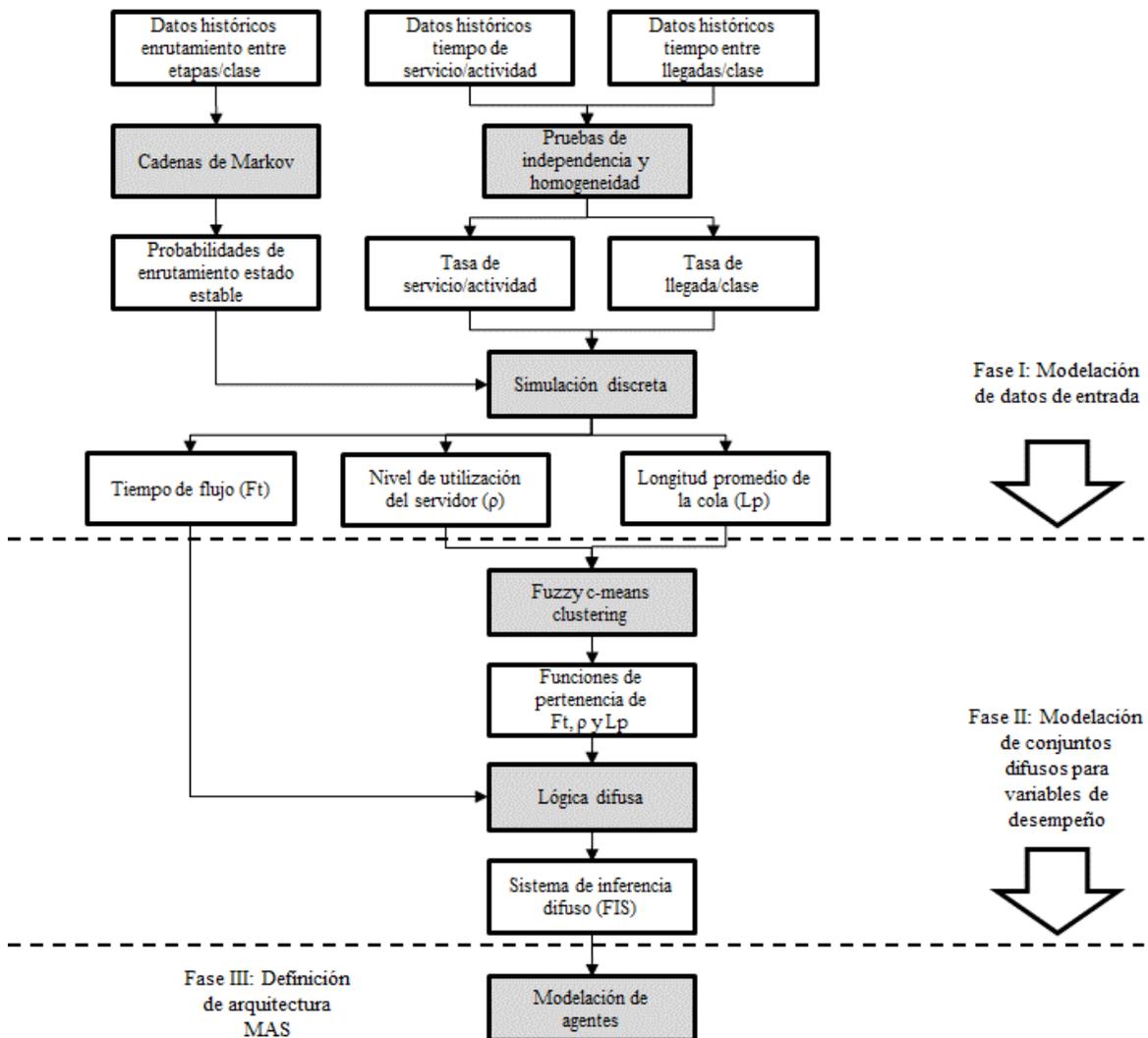
Fuente: Elaboración propia

**Figura 4** Esquema de la notación utilizada para la modelación de sistemas de redes de colas

### 3.2. Diseño metodológico

El diseño metodológico del sistema multi-agente busca responder a la siguiente pregunta de investigación: ¿Cuáles deben ser los componentes, técnicas analíticas y computacionales de un sistema multi-agente cuyo propósito sea formular políticas factibles de enrutamiento, balanceo de cargas y priorización de transacciones, que respondan a los efectos de la interacción entre etapas de sistemas de redes de colas, considerando las restricciones de operación y objetivos de desempeño requeridos?

La metodología para la construcción del sistema multi-agente que ofrezca una respuesta al problema de investigación planteado, considera la definición de tres (3) fases dependientes entre sí, como puede observarse en la figura 5.



Fuente: Elaboración propia

**Figura 5** Esquema general de la metodología para la construcción del sistema multi-agente

En la Fase I, se realiza el procesamiento de la información requerida para la modelación de las variables que explican el funcionamiento de un sistema de redes de colas. La modelación del tiempo entre llegadas sucesivas y la tasa de servicio para cada clase de servicio en términos de teoría de probabilidad, se realiza mediante la aplicación de pruebas de bondad de ajuste; en segunda instancia se utilizan pruebas de hipótesis para validar los supuestos de independencia y homogeneidad de los datos obtenidos mediante fuentes de información primarias. La modelación de la transición entre etapas de la red de colas como segundo elemento clave para la representación de este tipo de sistemas se realiza utilizando cadenas de Markov, cuya información es obtenida de reportes generados desde el sistema de información para gestión de casos dispuesto por la empresa tomada como caso de estudio. Ambos resultados de la modelación de las variables de la red de colas, es utilizado para simular el sistema y obtener los datos para la identificación de los conjuntos difusos que representen adecuadamente para cada servidor  $l$  de la etapa  $j$ , el comportamiento de las variables:

- Tiempo de flujo ( $Ft_{jl}$ ),
- Nivel de utilización ( $\rho_{jl}$ ) y,
- Longitud promedio de la cola ( $Lp_{jl}$ ).

En la Fase II se implementa el algoritmo de agrupación o clusterización difusa Fuzzy c-means clustering o FCM (Dunn, 1.973) para obtener las funciones de pertenencia a conjuntos difusos de las variables  $Ft_{lj}$ ,  $\rho_{lj}$  y  $Lp_{lj}$  cuyos datos son obtenidos mediante el modelo de simulación discreta desarrollado en la fase I del diseño metodológico. A diferencia de otros algoritmos para clasificación y reconocimiento de patrones como k-means (MacQueen, 1.967), FCM permite que un dato pertenezca a uno o más grupos (clusters) en este caso, de etiquetas lingüísticas (por ejemplo: “bajo”, “medio”, “alto”), dando con esto la posibilidad de modelar bajo un grado de incertidumbre metalingüístico y dentro de un universo de discurso definido, las variables consideradas para la construcción del sistema multi-agente propuesto. FCM se basa en la minimización de la función objetivo (2):

$$J_m = \sum_{i=1}^N \sum_{j=1}^C \mu_{ij}^m \|x_i - c_j\|^2 \quad , \quad 1 \leq m < \infty \quad (2)$$

Donde:

$m$ , es un número real mayor que 1 que indica la  $m$ -ésima potencia del grado de pertenencia del  $k$ -ésimo dato al clúster  $i$ , Cuando  $m$  tiene a 1 la partición óptima es cada vez más cercana a un clúster específico. En general los valores de  $m$  más utilizados se seleccionan del intervalo [1,30].

$\mu_{ij}$ , representa el grado de pertenencia de  $x_i$  al clúster  $j$ .

$c_j$ , es la dimensión (o valor) del centro del clúster.

Las particiones difusas son obtenidas mediante iteraciones que buscan alcanzar el logro de la función objetivo, mediante la actualización de la función de pertenencia  $\mu_{ij}$  y los centros del clúster, las cuales son calculados con las expresiones (3) y (4) respectivamente:

$$\mu_{ij} = \frac{1}{\sum_{k=1}^C \left( \frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad (3)$$

$$c_j = \frac{\sum_{i=1}^N \mu_{ij}^m \cdot x_i}{\sum_{i=1}^N \mu_{ij}^m} \quad (4)$$

El algoritmo requiere la construcción de la matriz denominada U con N filas correspondientes al total de datos observados y C columnas que representan al número de clusters seleccionados para cada caso particular. Los factores incluidos en la matriz U, corresponden a los valores de  $\mu_{ij}$  que indican el grado de pertenencia de cada dato a los C clusters definidos. Cabe resaltar que la sumatoria de los factores de cada columna debe ser estrictamente igual a 1.

En términos de pseudocódigo, el algoritmo FCM se define como:

1. Inicializar en  $U^{(0)}$  la matriz  $U = [\mu_{ij}]$ ,
2. En cada k-paso: calcular el vector de centros  $C^{(k)} = [c_j]$  con  $U^{(k)}$ , mediante:
$$c_j = \frac{\sum_{i=1}^N \mu_{ij}^m \cdot x_i}{\sum_{i=1}^N \mu_{ij}^m}$$
3. Actualizar  $U^{(k)}$  a  $U^{(k+1)}$  mediante:
$$\mu_{ij} = \frac{1}{\sum_{k=1}^C \left( \frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$
4. Si,  $\|U^{(k+1)} - U^{(k)}\| < \varepsilon$ , finalizan las iteraciones. En caso contrario vuelva al paso 2.

$\varepsilon$ , representa el criterio de parada y corresponde a un valor entre 0 y 1. El algoritmo converge a un mínimo local o punto de silla de la función objetivo  $J_m$ . Una vez obtenidos los clusters correspondientes a cada etiqueta lingüística para el rango de valores de cada

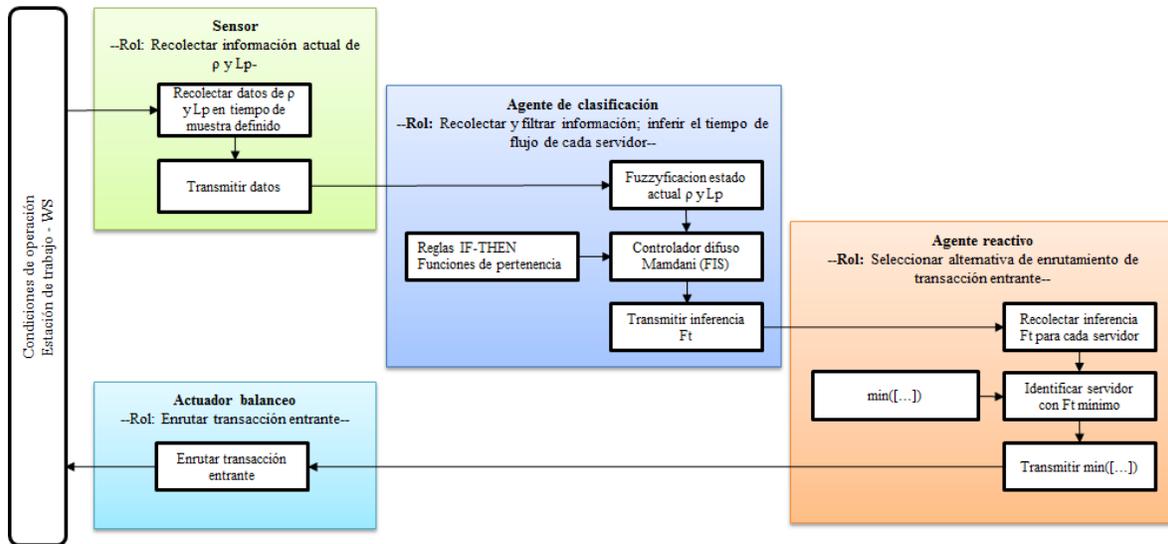
una de las tres (3) variables seleccionadas, mediante los criterios Desviación Absoluta Media (MAD, Mean Absolute Deviation) y Desviación Cuadrática Media (MSD, Mean Squared Deviation) se verificara el ajuste de las funciones de pertenencia obtenidas a través de FCM respecto a conjuntos difusos convencionales tipo sigmoidea, triangular, trapezoidal, gauss, etc., obteniendo con esto los parámetros aproximados que puedan representar adecuadamente las funciones de pertenencia iniciales para las cuales no se dispone de información inicial de sus parámetros. Dichos parámetros son utilizados en conjunto con reglas lingüísticas del tipo: *“si el nivel de utilización del servidor es bajo y la longitud promedio de la cola es baja, entonces el tiempo de flujo es bajo”*, que permitan en ambiente simulado inferir sobre la variable de salida (consecuente) generando a partir de esto un sistema de inferencia difusa (FIS, Fuzzy Inference System) que será implementado en el mismo ambiente de simulación como un controlador difuso para configurar un lazo de control abierto discreto. Dado el interés de representar el tiempo de flujo en términos de conjuntos difusos (al igual que las dos (2) variables incluidas en la entrada del sistema de inferencia) y no en términos de una función de la media ponderada de las variables de salida, se seleccionó para implementación un FIS tipo Mamdani el cual dispone de las ventajas expuestas en la tabla 7, respecto a los sistemas de inferencia difusa tipo Takagi, Sugeno y Kang (TSK).

**Tabla 7** Comparación de características principales sistemas de inferencia difusos tipo Mamdani y TSK

Característica	Mamdani	TSK
Sencillez en implementación de reglas difusas	Alta	Alta
Representación del nivel de conocimiento humano	Alto	Limitado
Continuidad en la superficie de salida (consecuente)	Media	Alta
Eficiencia computacional	Menor	Mayor
Requiere interfaz de defusificación	Basada en composición	No

Fuente: Elaboración propia

Finalmente en la Fase III, se define la arquitectura del sistema multi-agente, tomando como base el diseño de los roles y condiciones de interdependencia previstos para lograr en ambiente simulado, validar el impacto en el throughput global del sistema, en respuesta al mecanismo implementado de balanceo de cargas para los servidores (ver figura 6).



Fuente: Elaboración propia

**Figura 6** Arquitectura del sistema multi-agente propuesto

En la tabla 8, se relacionan las funciones de los sensores, agentes y actuadores requeridos para la implementación bajo el esquema de la presente propuesta de sistema multi-agente.

**Tabla 8** Funciones de elementos incluidos en arquitectura del sistema multi-agente propuesto

Clasificación agente	Condiciones de operación
Sensor	Recolecta de cada servidor la información disponible de las variables $\rho_{ji}$ y $L_{pji}$ en el tiempo de muestra configurado y la transmite al agente de clasificación
Agente de clasificación	Recibe del sensor la información de las variables $\rho_{ji}$ y $L_{pji}$ , la cual es cargada al FIS tipo Mamdani, quien genera el valor estimado de la variable de salida $F_{tji}$
Agente reactivo	Recopila los valores estimados de la variable $F_{tji}$ generados por cada uno de los agentes de clasificación dispuestos por cada servidor en cada etapa. Selecciona el mínimo valor de $F_{tji}$ y envía una señal que indica el enrutamiento de la transacción entrante al servidor con mínimo $F_{tji}$ .
Actuador de balanceo	Recibe la señal de enrutamiento y la ejecuta

Fuente: Elaboración propia

En este capítulo se describió la interacción de las técnicas estadísticas, de investigación de operaciones y de inteligencia artificial necesarias para el desarrollo de las fases previstas en el diseño metodológico para la construcción del sistema multi-agente propuesto. Específicamente para las técnicas de inteligencia artificial, se presentó la estructura del algoritmo de agrupamiento (fuzzy c-means) seleccionado para la obtención de las funciones de pertenencia y conjuntos difusos requeridos para la construcción del sistema de inferencia difuso que representa la función del agente de clasificación. Se seleccionó un controlador difuso tipo Mamdani, dado que este puede representar la variable de salida (tiempo de flujo) en términos funciones de pertenencia y etiquetas lingüísticas al igual que las variables de entrada (nivel de utilización del servidor y longitud promedio de la cola).

## CAPITULO IV: CASO DE ESTUDIO

En el presente capítulo en primera instancia se realiza una descripción detallada de las entradas, etapas de proceso y salidas de la operación de servicio tomada como caso de estudio para la aplicación del sistema multi-agente propuesto. En este capítulo se desarrollan las actividades correspondientes a las fases del diseño metodológico expuesto en el capítulo IV. Se desarrolla la aplicación de las pruebas estadísticas a las variables de entrada básicas para la modelación de un sistema de redes de colas (tasa de llegadas y tasa de servicio), realizando la validación exhaustiva de los supuestos que deben cumplirse para la aplicación de dichas técnicas. Con la misma información suministrada y de acuerdo a los posibles estados que puede presentar una solicitud, se realiza la modelación de la transición entre dichos estados utilizando para tal propósito, cadenas de markov. Seguido, se realiza la descripción de las actividades generales para la construcción del modelo de simulación discreta que será utilizado para la generación de los datos requeridos para la modelación de los conjuntos difusos incluida en la fase II del diseño metodológico. En la fase II se describen las actividades para la construcción del controlador difuso Mamdani cuya función de inferir el tiempo de ciclo en cada recurso está asociada al agente de clasificación. Para su configuración, el controlador recibe como entradas los parámetros de las funciones de pertenencia teóricas que por los criterios MAD y MSD fueron seleccionadas dada la similitud en su comportamiento respecto a las funciones de pertenencia iniciales generadas con la aplicación del algoritmo de agrupación difuso sobre los datos generados por el modelo de simulación discreta para tres (3) etiquetas lingüísticas en cada variable. Finalmente, en la fase III se describe la programación realizada para el procesamiento de los datos entregados por los controladores difusos de cada etapa, así como la configuración final de los tres (3) tipos de agente y se comparan los resultados obtenidos vía simulación respecto a los algoritmos de enrutamiento Round-Robin y Equiprobable aplicando un su orden, una prueba de diferencia de medias y un análisis de varianza ANOVA, para identificar la conveniencia del algoritmo propuesto.

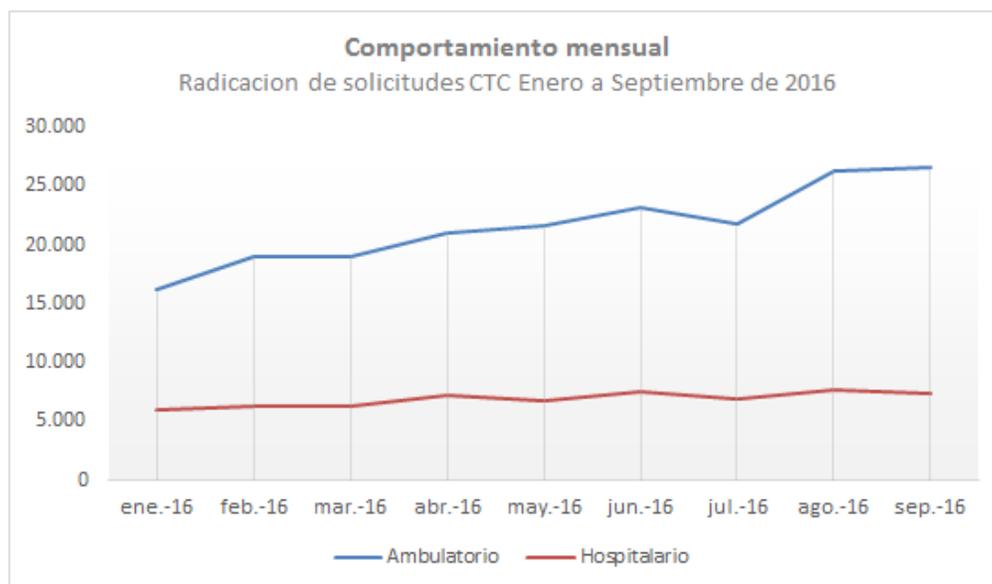
### 4.1. Descripción general del caso de estudio

Durante la prestación del servicio de salud a un Usuario, de acuerdo a las características del diagnóstico, el médico tratante puede solicitar el uso de tecnologías en salud (procedimientos, medicamentos y/o insumos) específicos no incluidos dentro del plan obligatorio de salud - POS. La solicitud debe ser instaurada por el médico o el Usuario ante la Entidad Promotora de Salud (de aquí en adelante EPS), para que a través de un comité tripartita conformado por un representante de los Usuarios, uno de las Instituciones Prestadoras de Servicios de Salud (de aquí en adelante IPS) y un tercer integrante en representación de la EPS, evalúen la pertinencia de la solicitud de servicio, dejando constancia de ello mediante un acta que posteriormente será requerida por la EPS para

reconocer a la IPS el cobro de los servicios prestados al Usuario, que como se mencionó, fueron previamente ordenados por el médico tratante. La EPS quien suministra la información para el caso de estudio desarrollado en el presente documento, tiene dispuestos los siguientes canales para la presentación de las solicitudes mencionadas dependiendo de las preferencias de servicio y el ámbito de prestación del servicio (ambulatorio u hospitalario):

- Oficinas de atención a Usuarios
- Directos a través de portal transaccional y correo electrónico.

Según los datos suministrados, para el periodo de Enero a Septiembre de 2016, las oficinas de atención al usuario son el canal más representativo para la presentación de solicitudes de servicios NO POS de ámbito ambulatorio en aproximadamente un 73% del total de solicitudes tramitadas directamente por los Usuarios, mientras que la presentación de solicitudes de ámbito hospitalario por portal transaccional y correo electrónico representan aproximadamente el 27% del total de solicitudes recibidas. En la figura 7, se presenta el comportamiento mensual en radicación por ámbito de emisión de la solicitud del servicio NO POS.



Fuente: Elaboración propia

**Figura 7** Comportamiento mensual de solicitudes de servicios NO POS instaurados ante la EPS caso de estudio

Para efecto de la modelación del sistema de redes de colas se asociara el ámbito de origen de la solicitud a una única clase de solicitud, bajo la siguiente notación:

- $\lambda_{1jk}$ : tasa de llegadas de la solicitud tipo 1 (ambulatorio) a la etapa inicial j con destino a la etapa k.
- $\lambda_{2jk}$ : tasa de llegadas de la solicitud tipo 2 (hospitalario) a la etapa inicial j con destino a la etapa k.

De manera general, las solicitudes de ámbito ambulatorio y hospitalario independiente al canal de presentación, son radicadas en el sistema de información de la EPS incluyendo además de la captura de la información básica de la solicitud, la digitalización y asociación de imágenes digitales de los soportes mínimos requeridos según la normatividad vigente. Únicamente en el caso de las solicitudes de ámbito hospitalario, la IPS envía las imágenes digitales de los soportes para la revisión y posterior aprobación o negación de la solicitud. Para efecto de clasificar a partir de la información entregada por la EPS, los posibles estados intermedios desde la radicación de la solicitud hasta su aprobación o negación, se definió arbitrariamente la codificación relacionada en la tabla 9:

**Tabla 9** Esquema de codificación estados finales e intermedios de la solicitud

Estado de la solicitud	Código
Sin gestión	1
Radicada	2
Legalizada	3
Devuelta	4
Aplazada	5
Aprobada	6
Negada	7
Otros procesos	8

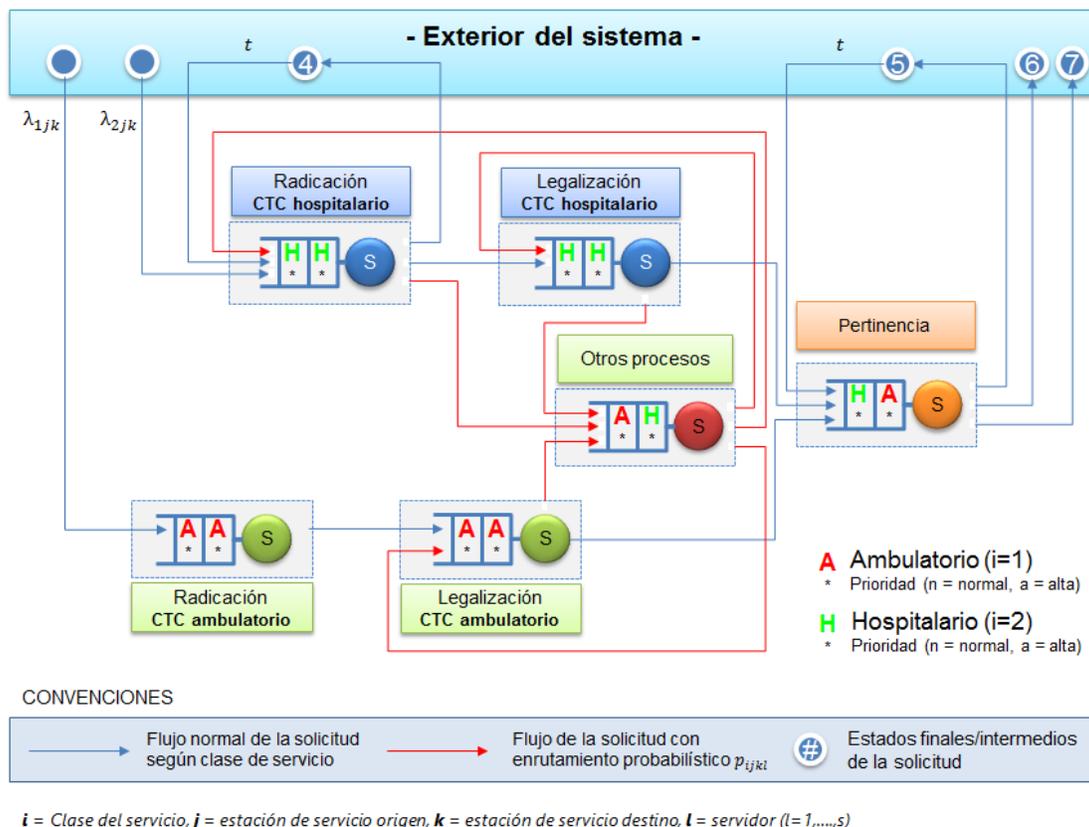
Fuente: Elaboración propia

Durante el proceso de radicación de la clase de solicitud  $\lambda_{2jk}$  (hospitalario), puede esta quedar en estado 4 (“Devuelta”) si se identifica que los soportes suministrados por la IPS no son los necesarios para radicar la solicitud o, en estado 8 (“Otros procesos”) si se identifica que el servicio solicitado requiere tramite de contratación con la IPS. En caso contrario de alguno o combinación de los dos (2) eventos anteriormente descritos, la solicitud puede transferirse a la etapa denominada “Legalización CTC Hospitalario”.

Durante el proceso de elaboración (o legalización) del acta del comité CTC y emisión del direccionamiento de la autorización para la prestación del servicio hacia una IPS en específico, la solicitud puede ser transferida a la etapa denominada pertinencia o quedar en estado 8 (“Otros procesos”) si se identifica que por el costo del servicio, la emisión de la autorización requiere aval del proceso administrativo de la EPS denominado “Alto

Costo". Dichas condiciones aplican de igual manera para la etapa encargada de legalizar las solicitudes de la clase de servicio denominada  $\lambda_{1jkl}$  (ambulatorio).

Como resultado de la etapa denominada "Pertinencia", las solicitudes pueden quedar en estado 6 ("Aprobada"), 7 ("Negada") o 5 ("Aplazada"). El estado 7 es alcanzable si el servicio solicitado corresponde a un servicio cubierto por el Plan Obligatorio de Salud (POS) en cuyo caso la solicitud debe ser emitida por el proceso administrativo de autorizaciones POS. Por otro lado, el estado 5 se alcanza si, la información suministrada por el medico solicitante no corresponde a la mínima requerida para identificar claramente la finalidad del servicio solicitado sobre el estado de salud actual del Usuario. En la figura 8 se esquematizan los protocolos de enrutamiento y los ciclos de realimentación que determinan el flujo de transacciones entre etapas dependiendo de las características propias de cada clase de solicitud (ambulatorio u hospitalario).



Fuente: Elaboración propia

**Figura 8** Protocolos de enrutamiento de transacciones del sistema de proceso de referencia

Dado que el proceso permite la salida temporal de solicitudes en estados intermedios luego de circular de forma predeterminada por las etapas requeridas para completar el trámite de la solicitud, el sistema será considerado como una red de colas multi-clase

abierta, cuyas características (en términos de la notación de Kendall & Lee) serán validadas mediante pruebas estadísticas de carácter inferencial cuyos fundamentos teóricos así como su resultado serán descritos en las siguientes secciones, considerando de manera adicional a lo anteriormente descrito, las siguientes premisas:

- La radicación de solicitudes de las clases  $\lambda_{1jkl}$  y  $\lambda_{2jkl}$  presentan picos dependiendo de la hora del día. Lo cual guarda coherencia con el comportamiento de las oficinas de atención a usuarios de la EPS.
- En un análisis preliminar sobre los datos históricos suministrados, la radicación de la clase de servicio denominada  $\lambda_{2jkl}$  (hospitalario) presenta un fenómeno de arribos masivos. Situación contraria con la radicación de la clase  $\lambda_{1jkl}$  donde se pudo observar una marcada tendencia al ajuste de los datos históricos a una distribución exponencial para el tiempo entre llegadas de dos (2) solicitudes sucesivas.
- Resultado de una toma de tiempos piloto, se identificó que (sin hacer distinción en la clase de solicitud) la tasa de servicio no ajusta a una distribución exponencial.
- Para todos los estados intermedios, se identifica un tiempo promedio de retorno de solicitud a la etapa origen de la transferencia de la solicitud.

#### **4.2. Fase I: Modelación de datos de entrada**

Para dar una perspectiva más exacta del sistema tomado como referencia para el desarrollo del sistema multi-agente propuesto, en esta sección se desarrollan las estimaciones sobre los datos recopilados y que corresponden a la denominada fase I, necesaria para la obtención del modelo. Se describe el tratamiento estadístico dado a las variables: 1) tiempo entre llegadas, 2) tiempo de servicio y 3) cantidad de transacciones enrutadas a las diferentes etapas del proceso.

##### ***4.2.1. Distribución de probabilidad del tiempo entre llegadas***

El proceso de entrada de clientes al sistema de líneas de espera, por lo regular se denomina, proceso de llegada. En general, los modelos analíticos existentes asumen que no ocurre más de una llegada en un instante dado; si hay más de una llegada en un instante de tiempo dado, se dice que el sistema recibe llegadas en masa. Si el número de clientes presentes no afecta al proceso de llegada, entonces este puede ser especificado mediante una distribución de probabilidad que rige el tiempo entre llegadas sucesivas. Definiendo a  $t_i$  como el tiempo en el cual llega el  $i$ -ésimo cliente. Para  $i \geq 1$ , se define como (5):

$$T_i = t_{i+1} - t_i \quad (5)$$

Como el  $i$ -ésimo tiempo entre llegadas. Al modelar el proceso de llegadas, se supondrá que las  $T_i$  son variables continuas, aleatorias e independientes descritas por la variable  $A$ . Suponiendo que  $A$  tiene una función de densidad de probabilidad  $a(t)$ , para  $\Delta t$  pequeñas (6):

$$P(t \leq A \leq t + \Delta t) \quad (6)$$

Esta alrededor de  $a(t) \Delta t$ . Como  $t$  no puede tomar valores negativos, se puede decir que (7):

$$P(A \leq c) = \int_0^c a(t)dt \quad (7)$$

Y (8),

$$P(A > c) = \int_c^{\infty} a(t)dt \quad (8)$$

La suposición que el tiempo entre llegadas está regido por la misma variable aleatoria, implica que la distribución de llegadas es independiente del momento del día o de la semana. Debido a fenómenos como horas pico, dicha suposición es con frecuencia irreal, pudiendo en este caso realizar una aproximación al descomponer la duración del día en segmentos.

Si se define a  $1/\lambda$  como el tiempo promedio entre llegadas (hora/llegada), se puede estimar  $1/\lambda$  (o tasa de llegadas) a partir de la función de densidad de probabilidad  $a(t)$  mediante (9):

$$\frac{1}{\lambda} = \int_0^{\infty} ta(t)dt \quad (9)$$

En los modelos analíticos de líneas de espera, un aspecto clave consiste en seleccionar  $A$  de tal manera que refleje la realidad. La elección más común de  $A$  es la distribución exponencial, dado que  $a(t)$  disminuye rápidamente para  $t$  pequeños, lo cual implica que son improbables los tiempos entre llegadas muy grandes.

Con el objetivo de validar el supuesto que los datos históricos observados son variables continuas, aleatorias e independientes, específicamente para los relacionados con:

- 1) tiempo entre llegadas por clase de transacción ( $\lambda_{ijk}$ ) y,
- 2) tiempo de servicio por actividad ( $\mu_i$ ),

Se realizó la aplicación de pruebas estadísticas específicas (ver tabla 10) para corroborar la hipótesis de independencia (las variables aleatorias no son correlacionadas) y determinación del tipo de distribución de probabilidad de ambos conjuntos de datos.

**Tabla 10** Pruebas estadísticas requeridas para la identificación de independencia y homogeneidad de los datos de entrada

Objetivo de la Prueba	Nombre de la prueba
Validar aleatoriedad de las variables continuas $\lambda_{ijkl}$ y $\mu_{il}$	Bondad de ajuste Kolmogov - Smirnov
	Bondad de ajuste ji-cuadrado ( $\chi^2$ )
	Bondad de ajuste Anderson - Darling
Validar independencia de las variables continuas $\lambda_{ijkl}$ y $\mu_{il}$	Corridas arriba y abajo
	Corridas arriba y debajo de la media
	Series

Fuente: Elaboración propia

Una prueba de bondad de ajuste es un procedimiento estadístico inferencial para contrastar la hipótesis que la función de distribución empírica de un conjunto de datos se ajusta a una función de distribución de probabilidad teórica.

Sea  $X_1, X_2, \dots, X_n$  una muestra aleatoria de una función de distribución de probabilidad  $F_X(x)$ , de manera general la hipótesis nula de la prueba de bondad de ajuste es de la forma (10):

$$H_0 : F_X(x) = F_0(x) \quad (10)$$

Con la correspondiente hipótesis alternativa (11):

$$H_1 : F_X(x) \neq F_0(x) \quad (11)$$

Donde  $F_0(x)$  denota una familia de distribuciones teóricas cuyos parámetros son conocidos. Las pruebas de bondad de ajuste existentes han sido obtenidas con base en alguno de los siguientes criterios.

- Función de distribución empírica.
- Función característica empírica.
- Momentos (poblacionales respecto a muestrales).
- Correlación y regresión.

Específicamente, las pruebas basadas en la función de distribución empírica (f.d.e), comparan la f.d.e con una función de distribución teórica. El estadístico de prueba es de la forma (12):

$$T_1 = d_1(F_0(x), F_n(x)) \quad (12)$$

Donde  $d_1(F_0(x), F_n(x))$  es una función de distancia.

La toma de decisión en el contraste de hipótesis puede tomarse observando el valor de una estadística cuya distribución de probabilidad, bajo la suposición de que  $H_0$  es cierta, se conozca. A partir de la información muestral se calcula un estadístico de prueba que se utiliza para determinar si se rechaza la hipótesis nula ( $H_0$ ) al comparar el valor calculado (del estadístico de prueba) respecto al valor crítico que delimita las regiones de la distribución de probabilidad de  $H_0$  a partir de las cuales es improbable que la hipótesis nula sea cierta. Por ejemplo, el cálculo del estadístico de prueba Kolmogorov-Smirnov consiste en identificar el valor máximo de la diferencia entre la probabilidad esperada (asumiendo que  $H_0$  es cierta) respecto a la probabilidad observada de la función de distribución empírica.

Una alternativa para la toma de decisión en el contraste de hipótesis consiste en emplear el p-valor asociado al estadístico de prueba  $D$  observado. El p-valor se define como (13):

$$pvalor = P(D_{esp} > D_{obs} | H_0 \text{ es cierta}) \quad (13)$$

Si el p-valor es grande en comparación con el nivel de significancia  $\alpha$ , se interpreta que siendo cierta la hipótesis nula, el valor observado del estadístico  $D$  era esperable. Por el contrario, si el p-valor es pequeño, podría concluirse que siendo cierta la hipótesis nula, era poco probable obtener en valor observado del estadístico  $D$ . La regla general de decisión al tomar como criterio el p-valor, puede expresarse como:

Si  $p\text{-valor} \geq \alpha \rightarrow$  Aceptar  $H_0$

Si  $p\text{-valor} < \alpha \rightarrow$  Rechazar  $H_0$

Al final de cualquier estudio de contraste de hipótesis y como parte de una validación sobre la precisión de la decisión, deben considerarse las siguientes situaciones:

- Rechazar  $H_0$  cuando es verdadera (o Error tipo I, denotado como  $\alpha$ ), o
- No rechazar  $H_0$  cuando esta es falsa (o Error tipo II, denotado como  $\beta$ ).

Por lo tanto y según lo anterior, en la tabla 11 se presentan los cuatro posibles sucesos que determinan si la decisión sobre  $H_0$  es correcta o incorrecta.

**Tabla 11** Errores en el contraste de hipótesis estadísticas

Alternativa	$H_0$ es cierta	$H_1$ es cierta
Se escogió $H_0$	No hay error (verdadero positivo)	Error tipo II ( $\beta$ o falso positivo)
Se escogió $H_1$	Error tipo I ( $\alpha$ o falso positivo)	No hay error (verdadero negativo)

Fuente: Elaboración propia

Considerando adicionalmente que:

- Los errores tipo I y II están relacionados de forma inversamente proporcional. Ello implica que el aumento en la probabilidad de uno tiene como resultado una disminución en la probabilidad del otro.
- El tamaño de la región de rechazo, y por tanto la probabilidad de cometer error tipo I, se puede reducir al ajustar el o los valores críticos.

El aumento del tamaño de muestra reducirá simultáneamente a  $\alpha$  y  $\beta$ .

Se aplicaron las pruebas de bondad de ajuste de Kolmogorov-Smirnov, Anderson Darling y Ji Cuadrado ( $\chi^2$ ) con el propósito de identificar la distribución de probabilidad de los datos empíricos correspondientes al tiempo entre llegadas sucesivas (por clase de servicio), basándose en la variabilidad y elementos de la teoría de probabilidad, bajo la siguiente hipótesis (ver tabla 12):

$H_0$ : el tiempo entre llegadas de dos (2) solicitudes sucesivas para la clase de servicio  $i$  se ajusta a una distribución exponencial de parámetro  $\lambda_i$ .

$H_1$ : el tiempo entre llegadas de dos (2) solicitudes sucesivas para la clase de servicio  $i$  no se ajusta a una distribución exponencial de parámetro  $\lambda_i$ .

**Tabla 12** Resumen características principales pruebas de bondad de ajuste seleccionadas

Característica	K-S*	$\chi^2$ **	A-D***
Prueba No Paramétrica	Si	Si	Si
Distribuciones de variables aleatorias continuas	Si	Si	Si
Distribuciones de variables aleatorias discretas	No	Si	No
Datos no agrupados	Si	No	Si
Distribuciones específicas para valores críticos	No	No	Si
Sensible al tamaño de intervalos de clase	-	Si	-
Sensible a valores centrales de la distribución	Si	-	No
Requiere tamaños de muestra representativos	-	Si	-

**K-S\***: Kolmogorov-Smirnov.  **$\chi^2$  \*\***: Ji-Cuadrado. **A-D\*\*\***: Anderson-Darling

Fuente: Elaboración propia

No se realizó la aplicación de la prueba de Shapiro-Wilk por tratarse de un procedimiento diseñado específicamente para comprobar normalidad, preferiblemente en muestras de datos con  $n < 30$ ).

En la tabla 13 se relacionan los estadísticos de prueba para cada uno de los tres (3) test considerados para el contraste de hipótesis.

**Tabla 13** Estadísticos de prueba bondad de ajuste

Prueba	Estadístico de prueba
Kolmogorov - Smirnov	$\text{Max} PEA_i - POA_i $
$\chi^2$ (Ji-Cuadrado)	$C = \sum_{i=1}^m \frac{(FE_i - FO_i)^2}{FE_i}$
Anderson - Darling	$A_n^2 = - \sum_{i=1}^N \frac{(2i - 1)}{n} [\ln F(Y_i) + \ln(1 - F(Y_{n+1-i}))] - n$

Fuente: Elaboración propia

A continuación se presentan los procedimientos correspondientes a cada una de las tres (3) pruebas presentadas en la tabla 5, calculadas a un nivel de confianza del 95% y cuyo propósito es probar el ajuste de los datos a la distribución de probabilidad teórica, tomando como referencia el tiempo entre llegadas sucesivas calculado sobre los datos históricos para la franja de 07:00:00 a 07:59:59 del día lunes.

#### Prueba de bondad de ajuste Kolmogorov-Smirnov

Los  $n = 38$  datos históricos correspondientes al tiempo entre llegadas fueron ubicados en una tabla de frecuencias de  $m = 7$  intervalos (ver tabla 11), con una amplitud de 0,59286 minutos obteniendo así, la frecuencia observada en cada intervalo  $i$  ( $FO_i$ ).

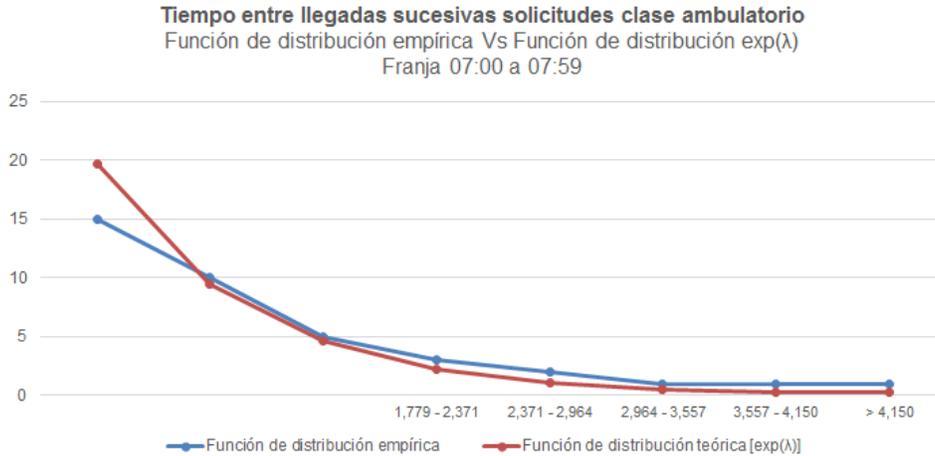
- 1) Con base en las frecuencias observadas para cada intervalo, se calculó la probabilidad observada ( $PO_i$ ) como (14):

$$PO_i = \frac{FO_i}{n} \quad (14)$$

- 2) Una vez calculada  $PO_i$  se calculó la probabilidad observada acumulada ( $POA_i$ ) como (15):

$$POA_i = PO_i + POA_{i-1} \quad (15)$$

- 3) Bajo el supuesto que el tiempo entre llegadas sucesivas de las solicitudes tipo 1 (ambulatorio) se ajusta a una distribución exponencial de parámetro lambda ( $\lambda$ ). En la figura 9 puede evidenciarse a priori una clara tendencia de la función de distribución empírica al ajuste a una distribución exponencial de parámetro lambda.



Fuente: Elaboración propia

**Figura 9** Comportamiento del tiempo entre llegadas sucesivas

Se calculó la probabilidad esperada ( $PEA_i$ ), mediante (16):

$$F(x) = 1 - e^{-LS/\lambda_i} \tag{16}$$

Donde  $LS$  y  $\lambda$  hacen referencia (ver tabla 14) al límite superior de la clase (de cada intervalo) y la media de la muestra respectivamente ( $\lambda_i = 0,82047$  solicitud/minuto).

**Tabla 14** Tabla de frecuencias observadas y esperadas acumuladas prueba de bondad de ajuste Kolmogorov-Smirnov

Intervalo		$FO_i$	$PE_i$	$FE_i$	$ FE_i - FO_i $	$ POA_i - PEA_i $	
LI	LS						
0,00000	-	0,59762	15	0,39474	0,39474	0,51731	0,12257
0,59762	-	1,19524	10	0,26316	0,65789	0,76701	0,10912
1,19524	-	1,79286	5	0,13158	0,78947	0,88754	0,09806
1,79286	-	2,39048	3	0,07895	0,86842	0,94572	0,07729
2,39048	-	2,98810	2	0,05263	0,92105	0,97380	0,05274
2,98810	-	3,58571	1	0,02632	0,94737	0,98735	0,03998
3,58571	-	4,18333	1	0,02632	0,97368	0,99390	0,02021
	>	4,15000	1	0,02632	1,00000	1,00000	0,00000
n			38				

Fuente: Elaboración propia

- 4) Se calculó el valor absoluto de la diferencia entre  $POA_i$  y  $PEA_i$  para cada intervalo y se identificó que la máxima diferencia entre ambas probabilidades es de 0,12257, es decir (17):

$$\text{Max}|PEA_i - POA_i| = |PEA_1 - POA_1| \quad (17)$$

Donde el subíndice 1 se refiere al intervalo 0,00000 - 0,59762.

- 5) Para un nivel de confianza  $\alpha$  del 5% con  $n = 38$  grados de libertad, de la tabla de valores críticos de Kolmogorov-Smirnov, se identifica un valor crítico  $D_{0,05;38} = 0,22062$ . Dado lo anterior se identifica que (18):

$$\text{Max}|PEA_i - POA_i| < D_{\alpha, n} \quad (18)$$

En cuyo caso, se puede concluir según los datos empíricos que no hay evidencia suficiente para rechazar la hipótesis nula donde se afirma que el tiempo entre llegadas de dos (2) solicitudes sucesivas para la clase de servicio 1 (ambulatorio) se ajusta a una distribución exponencial de parámetro  $\lambda_i$ .

### **Prueba de bondad de ajuste Ji Cuadrado ( $\chi^2$ )**

Al igual que en la prueba de bondad de ajuste de Kolmogorov-Smirnov, los  $n = 38$  datos históricos correspondientes al tiempo entre llegadas fueron ubicados en una tabla de frecuencias de  $m = 7$  intervalos, con una amplitud de 0,59286 minutos obteniendo así, la frecuencia observada en cada intervalo  $i$  ( $FO_i$ ). Conservando el supuesto que el tiempo entre llegadas sucesivas de las solicitudes tipo 1 (ambulatorio) se ajusta a una distribución exponencial de parámetro lambda ( $\lambda$ ), se calculó la probabilidad esperada ( $PO_i$ ), mediante la siguiente expresión (19):

$$F(x) = 1 - e^{-LS/\lambda_i} \quad (19)$$

Donde  $LS$  y  $\lambda$  hacen referencia al límite superior de la clase (de cada intervalo) y la media de la muestra respectivamente ( $\lambda = 0,82047$  solicitud/minuto). Luego de calcular  $PO_i$  para cada intervalo, se calculó la frecuencia esperada de cada clase ( $FE_i$ ) como (20):

$$FE_i = FO_i * PE_i \quad (20)$$

dada la probabilidad esperada para cada intervalo. En la tabla 15, se presenta el cálculo de las probabilidades y frecuencias esperadas para cada intervalo (i).

**Tabla 15** Tabla de frecuencias observadas y esperadas prueba de bondad de ajuste ji-cuadrado

Intervalo		FO <sub>i</sub>	PE <sub>i</sub>	FE <sub>i</sub>	FE <sub>i</sub> - FO <sub>i</sub>
LI	LS				
0,00000	- 0,59286	15	0,51731	19,65777	1,10363
0,59286	- 1,18571	10	0,24970	9,48861	0,02756
1,18571	- 1,77857	5	0,12053	4,58006	0,03850
1,77857	- 2,37143	3	0,05818	2,21075	0,28177
2,37143	- 2,96429	2	0,02808	1,06711	0,81556
2,96429	- 3,55714	1	0,01355	0,51508	0,45652
3,55714	- 4,15000	1	0,00654	0,24863	2,27074
	> 4,15000	1	0,00610	0,23199	2,54258
	n	38		C	4,99427

Fuente: Elaboración propia

Una vez obtenidas las frecuencias observadas y esperadas, se calculó para cada intervalo (i) el valor absoluto de la diferencia entre FE<sub>i</sub> y FO<sub>i</sub>, y posteriormente, el estadístico de prueba C según la expresión relacionada en la tabla 5, obteniendo así un valor C = 4,99427. Dado que el estadístico de prueba C es menor al valor crítico ( $\chi^2_{5,0,25} = 12,59000$ ) de las tablas de la distribución  $\chi^2$  con 5 grados de libertad y a un nivel de confianza del 95%, se puede concluir según los datos empíricos que no hay evidencia suficiente para rechazar la hipótesis nula donde se afirma que el tiempo entre llegadas de dos (2) solicitudes sucesivas para la clase de servicio 1 (ambulatorio) se ajusta a una distribución exponencial de parámetro  $\lambda_i$ .

### Prueba de bondad de ajuste Anderson - Darling

El cálculo del estadístico de prueba Anderson - Darling, se realizó sobre los datos empíricos sin agrupar, donde:

- i: Posición del dato en la fila i.
- Y<sub>i</sub>: Datos empíricos organizados de menor a mayor
- Y<sub>n+1-i</sub>: Datos empíricos organizados de mayor a menor

El propósito de los cálculos sucesivos realizados para aplicar la prueba de bondad de ajuste de Anderson - Darling, se relacionan a continuación y cuyos resultados pueden observarse en la tabla 16:

- F(Y<sub>i</sub>): Valor calculado de la distribución exponencial acumulada con media  $\lambda = 0,82047$  solicitud/minuto para el valor Y<sub>i</sub>.
- F(Y<sub>n+1-i</sub>): Valor calculado de la distribución exponencial acumulada con media  $\lambda = 0,82047$  solicitud/minuto para el valor Y<sub>n+1-i</sub>.
- Ln(F(Y<sub>i</sub>)): Logaritmo natural del valor calculado F(Y<sub>i</sub>).

- $\ln(1-F(Y_{n+1-i}))$ : Logaritmo natural de  $1-F(Y_{n+1-i})$ .
- $S_i$ : Corresponde al siguiente calculo, realizado por cada fila de la matriz de datos empíricos de tamaño  $n = 38$ , el cual se realiza según (21):

$$S_i = (2i - 1) [\ln F(Y_i) + \ln(1 - F(Y_{n+1-i}))] \quad (21)$$

**Tabla 16** Calculo de distribuciones exponencial acumulada para  $Y_i$  y  $Y_{n+1-i}$  prueba de bondad de ajuste Anderson - Darling

									$\lambda_i$	0,82048
$i$	$Y_i$	$Y_{n+1-i}$	$F(Y_i)$	$F(Y_{n+1-i})$	$1-F(Y_{n+1-i})$	$\ln(F(Y_i))$	$\ln(1-F(Y_{n+1-i}))$	$S_i$		
1	0,03333	4,18333	0,02698	0,96769	0,03231	-3,61271	-3,43233	-7,04504		
2	0,05000	4,06667	0,04019	0,96444	0,03556	-3,21404	-3,33661	-19,65195		
3	0,10000	3,46667	0,07877	0,94183	0,05817	-2,54120	-2,84432	-26,92759		
4	0,10000	3,06667	0,07877	0,91923	0,08077	-2,54120	-2,51613	-35,40129		
5	0,25000	3,01667	0,18545	0,91585	0,08415	-1,68497	-2,47511	-37,44069		
6	0,26667	2,90000	0,19651	0,90739	0,09261	-1,62703	-2,37938	-44,07053		
7	0,35000	1,95000	0,24961	0,79809	0,20191	-1,38784	-1,59993	-38,84103		
8	0,36667	1,91667	0,25980	0,79249	0,20751	-1,34782	-1,57258	-43,80607		
9	0,40000	1,91667	0,27977	0,79249	0,20751	-1,27377	-1,57258	-48,38799		
10	0,45000	1,75000	0,30872	0,76208	0,23792	-1,17531	-1,43583	-49,61177		
11	0,50000	1,66667	0,33651	0,74525	0,25475	-1,08913	-1,36746	-51,58849		
12	0,50000	1,63333	0,33651	0,73818	0,26182	-1,08913	-1,34011	-55,87265		
13	0,51667	1,43333	0,34552	0,69149	0,30851	-1,06271	-1,17602	-55,96810		
14	0,53333	1,41667	0,35441	0,68725	0,31275	-1,03731	-1,16234	-59,39051		
15	0,60000	1,16667	0,38877	0,61604	0,38396	-0,94476	-0,95722	-55,15753		
16	0,63333	1,03333	0,40526	0,57165	0,42835	-0,90322	-0,84783	-54,28243		
17	0,78333	0,98333	0,47413	0,55372	0,44628	-0,74627	-0,80680	-51,25130		
18	0,85000	0,93333	0,50212	0,53503	0,46497	-0,68891	-0,76578	-50,91399		
19	0,86667	0,88333	0,50889	0,51556	0,48444	-0,67553	-0,72475	-51,81054		
20	0,88333	0,86667	0,51556	0,50889	0,49111	-0,66251	-0,71108	-53,56994		
21	0,93333	0,85000	0,53503	0,50212	0,49788	-0,62544	-0,69741	-54,23649		
22	0,98333	0,78333	0,55372	0,47413	0,52587	-0,59110	-0,64271	-53,05375		
23	1,03333	0,63333	0,57165	0,40526	0,59474	-0,55922	-0,51964	-48,54848		
24	1,16667	0,60000	0,61604	0,38877	0,61123	-0,48444	-0,49229	-45,90611		
25	1,41667	0,53333	0,68725	0,35441	0,64559	-0,37506	-0,43759	-39,81979		
26	1,43333	0,51667	0,69149	0,34552	0,65448	-0,36890	-0,42391	-40,43344		
27	1,63333	0,50000	0,73818	0,33651	0,66349	-0,30356	-0,41024	-37,83145		
28	1,66667	0,50000	0,74525	0,33651	0,66349	-0,29404	-0,41024	-38,73528		
29	1,75000	0,45000	0,76208	0,30872	0,69128	-0,27170	-0,36921	-36,53210		
30	1,91667	0,40000	0,79249	0,27977	0,72023	-0,23257	-0,32819	-33,08513		
31	1,91667	0,36667	0,79249	0,25980	0,74020	-0,23257	-0,30084	-32,53835		
32	1,95000	0,35000	0,79809	0,24961	0,75039	-0,22553	-0,28717	-32,30020		
33	2,90000	0,26667	0,90739	0,19651	0,80349	-0,09718	-0,21879	-20,53833		
34	3,01667	0,25000	0,91585	0,18545	0,81455	-0,08791	-0,20512	-19,63277		
35	3,06667	0,10000	0,91923	0,07877	0,92123	-0,08422	-0,08205	-11,47252		
36	3,46667	0,10000	0,94183	0,07877	0,92123	-0,05993	-0,08205	-10,08074		
37	4,06667	0,05000	0,96444	0,04019	0,95981	-0,03621	-0,04102	-5,63771		
38	4,18333	0,03333	0,96769	0,02698	0,97302	-0,03285	-0,02735	-4,51458		

$A^2_{38}$	0,31281
$A^2_T$	1,32600

Fuente: Elaboración propia

Posterior a la realización de los cálculos descritos, se aplicó el estadístico de prueba relacionado en la tabla 10 obteniendo un valor de  $A_{38}^2 = 0,31281$  que, comparado con el valor crítico  $A_T^2 = 1,32600$  de la prueba de Anderson - Darling para una distribución exponencial al 95% de nivel de confianza, se puede concluir según los datos empíricos que no hay evidencia suficiente para rechazar la hipótesis nula donde se afirma (ver tabla 16) que el tiempo entre llegadas de dos (2) solicitudes sucesivas para la clase de servicio 1 (ambulatorio) se ajusta a una distribución exponencial de parámetro  $\lambda_i$ .

Utilizando el software Easyfit® se realizó computacionalmente el cálculo de los p-valores asociados a los estadísticos de prueba obtenidos en cada una de las tres (3) pruebas de bondad de ajuste aplicadas sobre los tiempos entre llegadas de solicitudes clase ambulatorio (ver tabla 17). A un nivel de significancia  $\alpha = 0,05$ , para la prueba de Kolmogorov-Smirnov y Ji-cuadrado, se identificaron p-valores grandes en comparación con el nivel de significancia  $\alpha$ , interpretando con ello que, siendo cierta la hipótesis nula de ajuste de los datos a una distribución exponencial de parámetro  $\lambda_i$ , los valores observados de los estadísticos de prueba correspondientes eran esperables.

**Tabla 17** Calculo de p-valores para ajuste a distribución exponencial del tiempo entre llegadas

Franja/prueba/parámetro			07:00:00 - 07:59:59
n			38
$\lambda$			0,82048
p - value (no aplica para A-D)	Kolmogorov-Smirnov		0,98358
	Anderson-Darling	Estadístico	2,02610
		Valor crítico	2,50180
	Ji-cuadrado		0,94281

Fuente: Elaboración propia

La selección del p-valor como criterio, permite evitar la posibilidad de rechazar la hipótesis nula  $H_0$  cuando esta es verdadera, situación conocida como Error tipo I (denotado como  $\alpha$ ).

Con la realización de las pruebas de bondad de ajuste se comprobó que la variable aleatoria que representa el tiempo entre llegadas de las solicitudes tipo ambulatorio se comporta como una distribución exponencial de parámetro ( $\lambda$ ) quedando validado el supuesto inicial que indicaba que la variable aleatoria es aleatoria de tipo continuo, requiriendo sobre esta base comprobar la hipótesis de no correlación de la variable aleatoria (independencia). En las siguientes secciones, se ejemplifica la aplicación de las pruebas de independencia seleccionadas (ver tabla 9) utilizando los datos históricos correspondientes a la franja de 07:00:00 a 07:59:59 de los días lunes, que fueron empleados para las pruebas de bondad de ajuste aplicadas. El resultado consolidado de

las pruebas de bondad de ajuste a la variable aleatoria tiempo entre llegadas puede verse en el anexo 1.

### **Pruebas de independencia de tiempos entre llegadas**

Las pruebas estadísticas aplicadas para comprobar la independencia de los tiempos entre llegadas sucesivas que desde el punto de vista de la función de distribución empírica representan a la variable aleatoria  $r_i$ , se formulan de manera general sobre la siguiente hipótesis:

$H_0$ : los números del conjunto  $r_i$  que representan el tiempo entre llegadas de dos (2) solicitudes sucesivas, son independientes.

$H_1$ : los números del conjunto  $r_i$  que representan el tiempo entre llegadas de dos (2) solicitudes sucesivas, no son independientes.

### **Prueba de corridas arriba y abajo**

El objetivo de esta prueba consiste en determinar una secuencia de números (S) que solo contiene unos y ceros, de acuerdo a una comparación entre  $r_i$  y  $r_{i-1}$ . Posteriormente se determina el número de corridas observadas  $C_o$  (una corrida se identifica como la cantidad de unos o ceros consecutivos). Luego se calcula el valor esperado (22):

$$\mu_{C_o} = \frac{2n - 1}{3} \quad (22)$$

La varianza del número de corridas (23) y,

$$\sigma_{C_o}^2 = \frac{16n - 29}{90} \quad (23)$$

El estadístico  $Z_0$  (24):

$$Z_0 = \frac{C_o - \mu_{C_o}}{\sqrt{\sigma_{C_o}^2}} \quad (24)$$

Si el estadístico  $Z_0$  es mayor que el valor crítico  $Z_{\alpha/2}$ , se concluye que los números del conjunto  $r_i$  no son independientes.

Sea  $r_i$  el conjunto de los 38 tiempos entre llegadas calculados para la franja de las 07:00:00 a las 07:59:59, en la tabla 15 se presenta la secuencia de números (S) que solo contiene

unos y ceros, de acuerdo a una comparación entre  $r_i$  y  $r_{i-1}$ . Se asignara el número cero (0) si el valor de  $r_i$  es menor o igual que el valor de  $r_{i-1}$ , (se asigna el valor 1 en caso contrario). Se considera una corrida a un grupo de unos o ceros sucesivos, y su longitud puede ser mayor o igual a 1. La cantidad de corridas observadas se denota como  $C_o$ . La cantidad de corridas (ver tabla 18) observadas para el conjunto  $r_i$  tiene un valor  $C_o = 29$ .

**Tabla 18** Secuencia del número de corridas arriba y abajo observadas en los tiempos entre llegadas sucesivas

Nro.	1/λ	Sentido	Secuencia	Nro.	1/λ	Sentido	Secuencia
1	0,533333	*		20	0,933333	-	0
2	1,666667	+	1	21	1,433333	+	1
3	1,750000	+	1	22	0,350000	-	0
4	0,250000	-	0	23	0,366667	+	1
5	2,900000	+	1	24	0,400000	+	1
6	1,950000	-	0	25	0,100000	-	0
7	3,466667	+	1	26	1,166667	+	1
8	3,066667	-	0	27	0,633333	-	0
9	4,183333	+	1	28	0,050000	-	0
10	1,633333	-	0	29	1,916667	+	1
11	0,866667	-	0	30	1,916667	-	0
12	0,500000	-	0	31	0,100000	-	0
13	4,066667	+	1	32	0,783333	+	1
14	0,450000	-	0	33	0,500000	-	0
15	3,016667	+	1	34	1,416667	+	1
16	1,033333	-	0	35	0,266667	-	0
17	0,850000	-	0	36	0,983333	+	1
18	0,883333	+	1	37	0,033333	-	0
19	0,516667	-	0	38	0,600000	+	1

Fuente: Elaboración propia

El valor esperado y varianza del número de corridas, se calculan utilizando (19) y (20) obteniendo  $\mu_{C_o} = 25$  y  $\sigma_{C_o}^2 = 6,433333$  respectivamente. Dado que al utilizar (21) se obtiene que el estadístico de prueba tiene un valor de  $Z_0 = 1,57704$  puede concluirse que comparado con el valor crítico  $Z_{0,025} = 1,96000$ , no hay evidencia suficiente para rechazar la hipótesis nula donde se afirma el conjunto  $r_i$  de los tiempos entre llegadas de dos (2) solicitudes sucesivas son independientes.

### ***Prueba de corridas arriba y abajo de la media***

El objetivo de esta prueba consiste en determinar una secuencia de números (S) que solo contiene unos y ceros, de acuerdo a una comparación entre los números aleatorios del conjunto  $r_i$  y su media muestral. Con esta información se calcula el número de corridas observadas  $C_o$  y los valores  $n_0$  y  $n_1$  que corresponden al número de ceros y numero de unos en secuencia respectivamente, cumpliéndose que (25):

$$n_0 + n_1 = n \quad (25)$$

Luego se calcula el valor esperado (26):

$$\mu_{C_o} = \frac{2n_0n_1}{n} + \frac{1}{2} \quad (26)$$

La varianza del número de corridas (27) y,

$$\sigma_{C_o}^2 = \frac{2n_0n_1(2n_0n_1 - n)}{n^2(n - 1)} \quad (27)$$

El estadístico  $Z_0$  (28)

$$Z_0 = \frac{C_0 - \mu_{C_o}}{\sqrt{\sigma_{C_o}^2}} \quad (28)$$

Si valor calculado del estadístico de prueba (ver tabla 19)  $Z_0$  está fuera del  $-z_{\alpha/2} \leq Z_0 \leq z_{\alpha/2}$ , se rechaza la hipótesis nula (los números del conjunto  $r_i$  son independientes).

**Tabla 19** Secuencia del número de corridas arriba y abajo de la media observadas en los tiempos entre llegadas sucesivas

Nro.	1/λ	Sentido	Secuencia	Nro.	1/λ	Sentido	Secuencia
1	0,533333	*		20	0,933333	-	0
2	1,666667	+	1	21	1,433333	+	1
3	1,750000	+	1	22	0,350000	-	0
4	0,250000	-	0	23	0,366667	-	0
5	2,900000	+	1	24	0,400000	-	0
6	1,950000	+	1	25	0,100000	-	0
7	3,466667	+	1	26	1,166667	-	0
8	3,066667	+	1	27	0,633333	-	0
9	4,183333	+	1	28	0,050000	-	0
10	1,633333	+	1	29	1,916667	+	1
11	0,866667	-	0	30	1,916667	+	1
12	0,500000	-	0	31	0,100000	-	0
13	4,066667	+	1	32	0,783333	-	0
14	0,450000	-	0	33	0,500000	-	0
15	3,016667	+	1	34	1,416667	+	1
16	1,033333	-	0	35	0,266667	-	0
17	0,850000	-	0	36	0,983333	-	0
18	0,883333	-	0	37	0,033333	-	0
19	0,516667	-	0	38	0,600000	-	0

Fuente: Elaboración propia

Al igual que en la prueba de corridas arriba y abajo, se considera a  $r_i$  el conjunto de los 38 tiempos entre llegadas calculados para la franja de las 07:00:00 a las 07:59:59, en la tabla 16 se presenta la secuencia de números (S) que solo contiene unos y ceros, de acuerdo a una comparación entre  $r_i$  y la media muestral. Se asigna el número uno (1) si el valor de  $r_i$  es mayor o igual que la media muestral, (se asigna el valor cero (0) en caso contrario). Se considera una corrida a un grupo de unos o ceros sucesivos, y su longitud puede ser mayor o igual a 1. La cantidad de corridas observadas se denota como  $C_o$ . La cantidad de corridas observadas para el conjunto  $r_i$  tiene un valor  $C_o = 14$  (ver tabla 19).

El valor esperado y varianza del número de corridas, se calculan utilizando (23) y (24) obteniendo  $\mu_{C_o} = 18,40541$  y  $\sigma_{C_o}^2 = 7,30448$  respectivamente. Dado que al utilizar (25) se obtiene que el estadístico de prueba tiene un valor de  $Z_o = -1,63001$  puede concluirse que comparado con el valor crítico  $Z_{0,025} = -1,96000$ , no hay evidencia suficiente para rechazar la hipótesis nula donde se afirma el conjunto  $r_i$  de los tiempos entre llegadas de dos (2) solicitudes sucesivas son independientes.

#### **4.2.2. Distribución de probabilidad del tiempo de servicio**

Asumiendo que los tiempos de servicio a los clientes son variables aleatorias, continuas e independientes, cuyo comportamiento se rige por la variable aleatoria  $S$ , cuya función de densidad de probabilidad es  $s(t)$ . Sea  $1/\mu$  el tiempo promedio de servicio de un cliente, se puede decir que (29):

$$\frac{1}{\mu} = \int_0^{\infty} ts(t)dt \quad (29)$$

Al igual que los tiempos entre llegadas, la mayoría de los modelos analíticos definidos por diversos autores, asumen que los tiempos de servicio son modelados como variables aleatorias exponenciales, teniendo con ello la ventaja de determinar la distribución de tiempo restante de servicio para un cliente sin la necesidad de rastrear el tiempo de servicio causado, dada la propiedad de falta de memoria de la distribución exponencial.

En sentido particular, cuando los tiempos de servicio no siguen una distribución exponencial, con frecuencia se asume que estos se rigen por una distribución Erlang con parámetro de forma  $k$  y parámetro de proporcionalidad  $k\mu$ , implicando con ello que el tiempo de servicio de un cliente puede considerarse como un recorrido por  $k$  fases de servicio, en el cual el tiempo para terminar cada fase tiene la propiedad de carencia de memoria y un tiempo promedio de  $1/k\mu$ .

En situaciones muy particulares, los tiempos entre llegadas o de servicio pueden ser modelados como si tuvieran varianza cero, en cuyo caso, se consideran deterministas.

Para la obtención de los datos empíricos de los tiempos de servicio, se realizó un procedimiento de muestreo aleatorio estratificado definiendo una población de tamaño  $N$  dividida en  $k$  subpoblaciones (o estratos) de tamaños  $N_1, N_2, \dots, N_k$ . Dado que se requiere determinar la función de distribución de probabilidad del tiempo de servicio para cada una de las etapas en las cuales debe procesarse la solicitud mediante enrutamiento probabilístico (radicación, legalización y pertinencia), durante el estudio de toma de tiempos, se representó a cada etapa como una subpoblación (o estrato) de tamaño proporcional sobre la población de tamaño  $N$ , extrayendo de los tres (3) estratos muestras aleatorias de tamaño  $n_i$  para los tiempos de servicio, cumpliéndose que  $n_1 + n_2 + \dots + n_k = n$ . Para tal efecto, inicialmente se realizó un muestreo no probabilístico con el propósito de obtener los parámetros requeridos por (27) a modo de muestra piloto de los tiempos de servicio por cada etapa, y basado en la estimación obtenida calcular el tamaño de muestra (30) de cada etapa (estrato).

$$n = \frac{Z^2 \sigma^2}{e^2} \quad (30)$$

Se tomó una muestra piloto de 15 tiempos por cada etapa del proceso, obteniendo los parámetros de varianza muestral presentados en la tabla 20.

**Tabla 20** Varianza de tiempos de servicio por etapa de proceso (estrato)

Parámetro	Radicación	Legalización	Pertinencia
$\sigma^2$	2,46889	1,61114	1,18165

Fuente: Elaboración propia

Para calcular el parámetro correspondiente al error, se tomaron como referencia los percentiles 50 y 55 de los datos observados en la muestra piloto buscando con ello que los valores observados en la muestra no presenten una desviación promedio superior a 0,15 minutos (9,22 segundos) respecto al promedio de la población. De acuerdo al criterio anteriormente expuesto, en la tabla 21, se presentan los errores calculados para la selección de la muestra de tiempos de servicio de cada etapa del proceso, así como los tamaños de muestra seleccionados.

**Tabla 21** Valores calculados de error muestral y tamaño de muestra por etapa

Parámetro	Radicación	Legalización	Pertinencia
$e^2$	0,02647	0,02406	0,02051
$n_i$	358	221	257

Fuente: Elaboración propia

Para efecto de posteriores contrastes que permitan identificar si existe una diferencia estadísticamente representativa en el tiempo de servicio cuando la actividad es ejecutada por dos (2) servidores diferentes que procesan por separado solicitudes de la clase 1 o 2, la muestra de tiempos por cada actividad, se extrajo con dos (2) servidores para cada una de las tres (3) actividades, conservando la proporción de cada clase de solicitudes sobre la cantidad global de solicitudes recibidas.

Al igual que a los tiempos entre llegadas sucesivas, se aplicaron las pruebas de bondad de ajuste de Kolmogorov-Smirnov, Anderson Darling y Ji Cuadrado ( $\chi^2$ ) con el propósito de identificar la distribución de probabilidad de los datos empíricos obtenidos en las muestras de tiempo de servicio por etapa (y por clase de servicio), bajo las siguientes hipótesis:

$H_0$ : el tiempo de servicio por etapa para la clase de servicio  $i$  se ajusta a una distribución exponencial de parámetro  $\mu_i$ .

$H_1$ : el tiempo de servicio por etapa para la clase de servicio  $i$  se ajusta a una distribución exponencial de parámetro  $\mu_i$ .

A continuación se presentan los resultados de la aplicación de las tres (3) pruebas presentadas en la tabla 5, calculadas a un nivel de confianza del 95% y tomando como referencia el tiempo de servicio para la etapa denominada "Radicación" sin realizar distinción sobre la clase de solicitud (ambulatoria u hospitalaria), pero conservando la proporción promedio de cada clase de solicitud sobre la cantidad global de solicitudes entrantes al sistema, así: ambulatorio 71,38% y hospitalario 28,62%.

Utilizando el software Easyfit® se aplicaron las mismas tres (3) pruebas de bondad de ajuste aplicadas sobre los tiempos entre llegadas de solicitudes clase ambulatorio ((ver tabla 22).

**Tabla 22** Calculo de p-valores para ajuste a distribución exponencial del tiempo de servicio etapa "Pertinencia"

Franja/prueba/parámetro		Pertinencia	
n		163	
$\lambda$		0,21202	
p - value (no aplica para A-D)	Kolmogorov-Smirnov		0,00000
	Anderson-Darling	Estadístico	49,41700
		Valor crítico	2,50180
	Ji-cuadrado		0,00000

Fuente: Elaboración propia

A un nivel de significancia  $\alpha = 0,05$ , para la prueba de Kolmogorov-Smirnov y Ji-cuadrado, se identificaron p-valores cero, que en comparación con el nivel de significancia  $\alpha$ , permiten interpretar que hay evidencia suficiente para rechazar la hipótesis nula de ajuste de los datos de tiempo de servicio en la etapa “Pertinencia” a una distribución exponencial de parámetro  $\mu_i$ . (ver tabla 22).

Dados los resultados anteriores, se realizó la búsqueda de una función de distribución de probabilidad continua no negativa que permita modelar la función de distribución de probabilidad de los tiempos de servicio, considerando principalmente la importancia de la propiedad de carencia de memoria y la relación entre las siguientes funciones de distribución de probabilidad: exponencial, Weibull, Erlang y Gamma:

- La distribución de Weibull puede interpretarse como una generalización de la distribución exponencial, dado que la distribución exponencial es una distribución Weibull con parámetro de forma  $\alpha = 1$  (31).

$$exp(\beta) = weibull(1, \beta) \quad (31)$$

- La distribución Erlang con parámetro de forma k, implica que el tiempo de servicio se puede considerar como el paso por k etapas de servicio, en el cual el tiempo para determinar cada etapa tiene la propiedad de carencia de memoria (de la distribución exponencial) y una media (32):

$$\int_0^{\infty} ts(t)dt = \frac{1}{k\mu} \quad (32)$$

- La distribución Gamma puede interpretarse como una generalización de la distribución exponencial, dado que la distribución exponencial es una distribución Gamma con parámetro de forma  $\alpha = 1$  (33):

$$exp(\beta) = gamma(1, \beta) \quad (33)$$

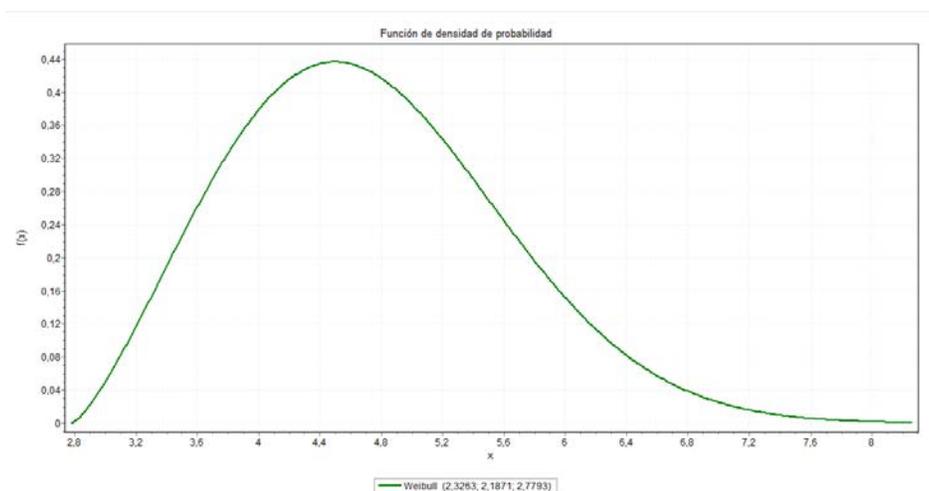
Según las relaciones anteriormente descritas entre la función de distribución de probabilidad exponencial y las funciones de distribución Weibull, Erlang y Gamma, y teniendo especial interés en la propiedad de carencia de memoria, puesto que preservar dicha propiedad permite determinar la distribución del tiempo restante de servicio para una solicitud sin tener que rastrear cuánto tiempo lleva en servicio, en la tabla 15 se recopilaron los p-valores de las tres (3) pruebas de bondad de ajuste empleadas, identificando p-valores altos para las tres (3) distribuciones. Se seleccionó la distribución Weibull dado que en comparación con las demás distribuciones presenta p-valores de 0,95294 y 0,64814 para las pruebas de Kolmogorov-Smirnov y Ji-cuadrado respectivamente (ver tabla 23).

**Tabla 23** Comparación p-valores ajuste a distribuciones con propiedad de distribución de carencia de memoria

Franja/prueba/parámetro		Exponencial	Weibull	Erlang	Gamma	
Parametros		$\lambda = 0,21202$	$\alpha = 2,3263$ $\beta = 2,1871$ $\gamma = 2,7793$	$m = 16$ $\beta = 0,22211$ $\gamma = 1,07330$	$\alpha = 16,403$ $\beta = 0,22211$ $\gamma = 1,07330$	
p - value (no aplica para A-D)	Kolmogorov-Smirnov	0,00000	0,95294	0,42155	0,91608	
	Anderson-Darling	Estadístico	49,41700	0,35456	1,35670	0,45356
		Valor crítico	2,50180	2,50180	2,50180	2,50180
	Ji-cuadrado	0,00000	0,64814	0,46269	0,59307	

Fuente: Elaboración propia

En la figura 10, se presenta la modelación del tiempo de servicio de la etapa pertinencia utilizando la distribución Weibull seleccionada.



Fuente: Elaboración propia

**Figura 10** Modelación del tiempo de servicio de la etapa pertinencia

### Pruebas de independencia de tiempos de servicio

Para comprobar la independencia de los tiempos de servicio obtenidos mediante muestreo, aplicaron las mismas pruebas empleadas sobre los tiempos entre llegadas sucesivas. Para ello, se formula sobre la variable aleatoria  $r_i$  la siguiente hipótesis:

$H_0$ : los números del conjunto  $r_i$  que representan el tiempo de servicio en la etapa  $j$ , son independientes.

$H_1$ : los números del conjunto  $r_i$  que representan el tiempo de servicio en la etapa  $j$ , no son independientes.

Las pruebas de independencia sobre los tiempos de servicio se aplicaron por separado para cada clase de solicitud (ambulatorio u hospitalario), etapa y servidor (3, 2 y 2

servidores para las etapas de “radicación”, “legalización” y “pertinencia”, respectivamente). En las tablas 24 y 25, se presentan los resultados de la aplicación de las pruebas de independencia de corridas arriba y abajo, y corridas arriba y abajo de la media, respectivamente. Los valores de  $Z_0$  indican que no hay evidencia suficiente para rechazar la hipótesis nula donde se afirma el conjunto  $r_i$  de los tiempos de servicio por cada etapa/servidor son independientes.

**Tabla 24** Resultado consolidado prueba de independencia corridas arriba y abajo para los tiempos de servicio por etapa

Clase solicitud	Etapa						
	Radicación			Legalización		Pertinencia	
	Ambulatorio	Ambulatorio	Hospitalario	Ambulatorio	Hospitalario	Ambulatorio	Hospitalario
Servidor	1	2	3	1	2	1	2
n	181	75	102	177	80	163	58
$\mu^2_{Co}$	120,33333	49,66667	67,66667	117,66667	53,00000	108,33333	38,33333
$\sigma^2_{Co}$	31,85556	13,01111	17,81111	31,14444	13,90000	28,65556	9,98889
$C_o$ (Q corridas)	119	48	66	120	50	109	33
$Z_0$ observado	-0,23624	-0,46205	-0,39491	0,41811	-0,80466	0,12454	-1,68749
$Z_{0,025}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96
Decisión	No rechaza $H_0$						

Fuente: Elaboración propia

**Tabla 25** Resultado consolidado prueba de independencia corridas arriba y debajo de la media para los tiempos de servicio por etapa

Clase solicitud	Etapa						
	Radicación			Legalización		Pertinencia	
	Ambulatorio	Ambulatorio	Hospitalario	Ambulatorio	Hospitalario	Ambulatorio	Hospitalario
Servidor	1	2	3	1	2	1	2
n	181	75	102	177	80	163	58
$\mu^2_{Co}$	84,60000	37,89189	49,71287	87,07955	39,73418	81,80247	29,07018
$\sigma^2_{Co}$	37,93783	17,41297	22,55850	41,13981	18,03560	39,31399	12,86689
$C_o$ (Q corridas)	91	38	50	89	36	83	24
$Z_0$ observado	1,03907	0,02591	0,06045	0,29941	-0,87928	0,19099	-1,41347
$Z_{0,025}$	1,96	1,96	1,96	1,96	1,96	1,96	1,96
Decisión	No rechaza $H_0$						

Fuente: Elaboración propia

### ***Diferencia de medias (tiempo de servicio)***

Con el propósito de modelar un único tiempo de servicio por cada etapa de la red de colas, y sabiendo que para efecto de identificar y contrastar estadísticamente diferencias significativas, la toma de tiempos se realizó de manera independiente y distinguiendo la clase de solicitud (ambulatorio u hospitalario) en diferentes servidores. Puesto que se desconoce la varianza poblacional, se aplicó una prueba de diferencia de medias que supone lo siguiente:

- Las poblaciones muestreadas se rigen bajo una distribución normal.

- Las dos (2) muestras son independientes.
- Las desviaciones estándar de ambas poblaciones son iguales.

Con la aplicación de esta prueba se desea probar a un nivel de significancia del 5% la hipótesis que (en promedio) el tiempo de servicio no tiene una diferencia representativa al compararlo de una clase de servicio a otra. Para ello, se formula sobre el tiempo promedio de servicio por etapa la siguiente hipótesis:

$H_0$ : No hay diferencia en el tiempo promedio de servicio para cada clase de servicio ( $\mu_1 - \mu_2 = 0$ ).

$H_1$ : Si hay diferencia en el tiempo promedio de servicio para cada clase de servicio ( $\mu_1 - \mu_2 \neq 0$ ).

Para garantizar la correcta aplicación de la prueba de diferencia de medias con varianzas conocidas, a continuación se resume la validación de los tres (3) supuestos anteriormente enunciados.

Con el propósito de validar el supuesto de normalidad en los datos muestreados se utilizó el software Easyfit® para aplicar sobre los tiempos de servicio (por clase de servicio y etapa), las tres (3) pruebas de bondad de ajuste presentadas en la tabla 26. Tomando como criterio los p-valores de las tres (3) pruebas, a un nivel de significancia  $\alpha = 0,05$ , se identificó que para los servidores número 1 y 3 de la etapa de radicación, hay evidencia suficiente para rechazar la hipótesis nula de ajuste de los datos de tiempo de servicio a una distribución normal de parámetros  $\mu$  y  $\sigma$ . Solo en el caso anterior, la normalidad será tomada como supuesto.

**Tabla 26** Resultados aplicación de pruebas de bondad de ajuste para validar supuesto de normalidad de los tiempos de servicio

Clase solicitud		Etapa						
		Radicación			Legalización		Pertinencia	
Servidor		Ambulatorio	Ambulatorio	Hospitalario	Ambulatorio	Hospitalario	Ambulatorio	Hospitalario
		1	2	3	1	2	1	2
n		181	75	102	177	80	163	58
$\mu$		3,74220	3,59500	3,67930	5,08480	5,19850	4,71660	5,02680
$\sigma$		1,34470	1,14320	1,27430	0,97195	1,10420	0,89231	1,18180
p - value (no aplica para A-D)	Kolmogorov-Smirnov	0,00080	0,43827	0,02828	0,06409	0,28227	0,42641	0,14515
	Anderson-Darling	6,79760	0,93971	2,91550	2,58840	1,29110	0,68521	1,06540
	Estadístico Valor crítico	2,50180	2,50180	2,50180	2,50180	2,50180	2,50180	2,50180
	Ji-cuadrado	0,00006	0,68834	0,00893	0,10872	0,56238	0,92564	0,07230
Decisión		Rechaza $H_0$	No rechaza $H_0$	Rechaza $H_0$	No rechaza $H_0$	No rechaza $H_0$	No rechaza $H_0$	No rechaza $H_0$

Fuente: Elaboración propia

Respecto al supuesto de independencia de los tiempos de servicio por clase de solicitud (ambulatorio u hospitalario), etapa y servidor, este quedo validado con el no rechazo de la hipótesis nula de independencia en los datos, una vez aplicadas las pruebas de corridas arriba y abajo, y corridas arriba y abajo de la media. Finalmente, para efectos prácticos en

la aplicación de la prueba de diferencia de medias, la igualdad en las desviaciones estándar de ambas poblaciones se conservara como un supuesto.

Sea X una población con media  $\mu_x$  y Y una población media  $\mu_y$ , ambas con varianzas poblacionales desconocidas pero iguales  $\sigma_x^2$  y  $\sigma_y^2$  respectivamente. El estadístico de prueba para contrastar la diferencia entre las dos medias de tiempos de servicio por etapa de proceso, clase de servicio y servidor, obtenidas a partir de muestras aleatorias de  $n_1$  observaciones de X y  $n_2$  observaciones de Y, se calcula según (34):

$$z = \frac{\bar{X}_1 - \bar{X}_2}{S_p \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}} \quad (34)$$

Donde el estimador de la varianza común para las dos poblaciones se calcula como (35):

$$S_p^2 = \frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 2} \quad (35)$$

Una vez calculado el estadístico de prueba es comparado con el valor crítico correspondientes de la distribución t-student, con  $v = n_1+n_2-2$  grados de libertad (número de observaciones de las muestras menos dos (2) dada la estimación de  $\mu_x$  y  $\mu_y$ ). La región de rechazo de  $H_0$  en un contraste bilateral se define como (36):

$$\{t < -t_{\frac{\alpha}{2}; n_1+n_2-2} \text{ o } t > t_{\frac{\alpha}{2}; n_1+n_2-2}\} \quad (36)$$

Como se indicó en el principio de este apartado, para efectos de la construcción del modelo de simulación discreta del sistema tomado como caso de estudio, interesa modelar un único tiempo de servicio por cada etapa de la red de colas, en la tabla 27 se presenta el resultado consolidado de la aplicación de la prueba de diferencias de medias.

**Tabla 27** Resultados aplicación de la prueba de diferencia de medias de los tiempos de servicio por etapa de proceso y clase de servicio

Hipotesis nula	Radicación			Legalización	Pertinencia
	$\mu_{1r-a} = \mu_{2r-a}$	$\mu_{1r-a} = \mu_{1r-h}$	$\mu_{2r-a} = \mu_{1r-h}$	$\mu_{1l-a} = \mu_{1l-h}$	$\mu_{1p-a} = \mu_{1p-h}$
n	254	281	175	255	219
$S_p^2$	1,28602	1,31942	1,21886	1,01292	0,96765
Z	0,94494	0,44227	-0,50175	-0,83896	-2,06217
$t_{0,025,n}$	2,24100	2,24100	2,26050	2,24100	2,24100
Decisión	No rechaza $H_0$				

Fuente: Elaboración propia

A partir de estos resultados, se puede concluir que no hay evidencia suficiente para rechazar la hipótesis nula que indica que no hay diferencia en el tiempo promedio de servicio para cada clase de servicio ( $\mu_1 - \mu_2 = 0$ ), por tal razón se tomara un único tiempo de servicio para cada etapa de proceso ya que se cuenta con el soporte estadístico para modelar bajo este criterio.

#### **4.2.3. Modelación de la transición entre etapas de proceso**

Los datos históricos relacionados con el enrutamiento de transacciones entre etapas del sistema son modelados utilizando cadenas de Markov de primer orden, al considerar que la decisión de enrutamiento de una transacción a través de los nodos de la red se toma de forma independiente. Izady et al. (2.011) modela los procesos de enrutamiento de pacientes en un departamento de atención a emergencias con cadenas de Markov, buscando que el 98% de los pacientes que ingresan a unidades de cuidado sean transferidos o admitidos dentro de un plazo no superior a cuatro (4) horas, según lo exigido por la normatividad aplicable en el país de origen tomado como caso de estudio. Chambers et. al (2.000) emplea cadenas de Markov para calcular el número esperado de visitas sobre los diferentes nodos de un proceso de manufactura utilizado como caso de aplicación del modelo optimización basado en redes neuronales artificiales.

#### **Modelación del enrutamiento de transacciones en el sistema de red de colas**

Sea  $X_t$  el valor de los posibles estados intermedios (definidos en la tabla 3) desde la radicación de la solicitud hasta su aprobación o negación, en el tiempo  $t$  (identificados con  $0, 1, 2, \dots$ ), y considerando que el estado intermedio  $X_t$  no se conoce con certeza, este puede ser considerado como una variable aleatoria. La anterior situación permite clasificar la relación entre las variables aleatorias  $X_0, X_1, X_2, \dots$  como un proceso estocástico discreto en el tiempo, al igual que los procesos de líneas de espera y la técnica denominada simulación discreta, que buscan modelar analíticamente el número de transacciones en el sistema o los posibles eventos y características simuladas en puntos discretos del tiempo.

Una cadena de Markov es un proceso estocástico discreto en el tiempo, el cual supone que en cualquier instante el sistema puede estar en un número finito de estados identificados con  $1, 2, \dots, s$ . Un proceso estocástico discreto es una cadena de Markov, si para  $t = 0, 1, 2, \dots$  se cumple que la distribución de probabilidad del estado en el tiempo  $t+1$  depende del estado en el tiempo  $t(i_t)$  y no depende de los estados por los que pasa la cadena en el camino  $i_t$  en el instante  $t$  (37).

$$\begin{aligned} P(X_{t+1} = i_{t+1} | X_t = i_t, X_{t-1} = i_{t-1}, \dots, X_0 = i_0) \\ = P(X_{t+1} = i_{t+1} | X_t = i_t) \end{aligned} \quad (37)$$

En las cadenas de Markov, se asume que para todo instante de tiempo  $t$  los estados  $i$  y  $j$  son independientes de  $t$  (38), donde  $p_{ij}$  representa la probabilidad que estando el sistema en el estado  $i$  en el instante de tiempo  $t$ , pase al estado  $j$  en el instante  $t+1$ .

$$P(X_{t+1} = j | X_t = i) = p_{ij} \quad (38)$$

El cambio de estado del sistema desde  $t$  hasta  $t+1$ , se denomina transición y el conjunto de las probabilidades  $p_{ij}$  se denominan probabilidades de transición de la cadena de Markov.

En general, las probabilidades de transición se organizan como una matriz de transición  $P$  de tamaño  $s \times s$  (39)

$$P = \begin{bmatrix} p_{11} & \cdots & p_{1s} \\ \vdots & \ddots & \vdots \\ p_{s1} & \cdots & p_{ss} \end{bmatrix} \quad (39)$$

Los elementos de la matriz  $P$  deben ser no negativos, y la suma de los elementos deben cumplir (40), es decir, la suma de las probabilidades en cada fila debe ser igual a 1.

$$\sum_{j=1}^{j=s} p_{ij} = 1 \quad (40)$$

Dados los estados  $i$  y  $j$ , una trayectoria de  $i$  a  $j$  es una secuencia de transiciones que inicia en  $i$  y termina en  $j$ , tal que cada transición en la secuencia tiene una probabilidad positiva de ocurrir.

**Probabilidades de estado estable**

Sea  $P$  la matriz de transición de una cadena ergódica de estado estable. Entonces existe un vector  $\pi = [\pi_1 \ \pi_2 \ \dots \ \pi_s]$  tal que (41):

$$\lim_{n \rightarrow \infty} P^n = \begin{bmatrix} \pi_1 & \cdots & \pi_s \\ \vdots & \ddots & \vdots \\ \pi_1 & \cdots & \pi_s \end{bmatrix} \quad (41)$$

Para  $n$  grande,  $P^n$  tiende a ser una matriz donde los elementos de las filas son idénticos. Lo anterior implica que después de un número determinable de periodos, la cadena de Markov se estabiliza con una probabilidad  $\pi_j$  en el estado  $j$  de manera independiente al estado inicial  $i$ .

El vector  $\pi = [\pi_1 \ \pi_2 \ \dots \ \pi_s]$  se denomina distribución de estado estable de la cadena de Markov. Para una cadena de Markov con una matriz de transición  $P$ , para  $n$  grande y toda  $i$  se cumple (42):

$$P_{ij}(n + 1) \cong P_{ij}(n) \cong \pi_j \quad (42)$$

Puesto que  $P_{ij}(n + 1)$  equivale a identificar el elemento de la matriz como “probabilidad de la fila  $i$  de  $P^n$  en la columna  $j$  de  $P^n$ ”, se puede escribir como (43):

$$P_{ij}(n + 1) = \sum_{k=1}^{k=s} P_{ik}(n) p_{kj} \quad (43)$$

Si  $n$  es grande, sustituyendo (42) en (43), se obtiene (44):

$$\pi_j = \sum_{k=1}^{k=s} \pi_k p_{kj} \quad (44)$$

En forma matricial, (44) puede escribirse como (45)

$$\pi = \pi P \quad (45)$$

Dado que se requiere obtener valores únicos de las probabilidades de estado estable y considerando que el sistema de ecuaciones definido por (44) tendrá un número infinito de soluciones, haciendo que  $n$  tienda a infinito, se obtiene (46).

$$\pi_1 + \pi_2 + \dots + \pi_s = 1 \quad (46)$$

Reemplazando (40) por cualquiera de las ecuaciones de (38), se puede resolver el sistema de ecuaciones para obtener las probabilidades de estado estable. Sea la matriz de transición definida en la tabla 28.

**Tabla 28** Matriz de transición periodo 3

		Estado futuro (j)							
		1	2	3	4	5	6	7	8
Estado actual (i)	1	0,00000	0,79835	0,08436	0,00412	0,00412	0,10905	0,00000	0,00000
	2	0,00000	0,47013	0,10649	0,02987	0,01429	0,36234	0,00000	0,01688
	3	0,00000	0,00253	0,00253	0,02525	0,02778	0,92677	0,00253	0,01263
	4	0,00000	0,00000	0,00000	0,08696	0,13043	0,60870	0,13043	0,04348
	5	0,00000	0,00000	0,00000	0,20000	0,00000	0,80000	0,00000	0,00000
	6	0,80000	0,10000	0,00000	0,00000	0,00000	0,10000	0,00000	0,00000
	7	0,00000	0,00000	0,00000	0,00000	0,00000	0,50000	0,50000	0,00000
	8	0,00000	0,92308	0,00000	0,00000	0,00000	0,07692	0,00000	0,00000

Fuente: Elaboración propia

Sobre la matriz de transición puede verificarse lo siguiente:

- 1) Todos los estados j son alcanzables desde los estados i, dado que como puede observarse en la matriz de transición, existe al menos una trayectoria que conduce de los estados i a cada estado j.
- 2) Todos los estados j son accesibles desde los estados i, la accesibilidad entre estados implica que  $p_{ij}^n > 0$ . Si la matriz de transición se multiplica 19 veces por sí misma ( $P^{19}$ ), se identifica que las posibles trayectorias desde los estados i hacia cada estado j, tienen la misma probabilidad. Lo anterior indica que en  $n = 19$  todos los estados son accesibles (ver tabla 29).

**Tabla 29** Probabilidades de la transición de estados en  $P^{19}$

		Estado futuro (j)							
		1	2	3	4	5	6	7	8
Estado actual (i)	1	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	2	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	3	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	4	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	5	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	6	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	7	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	8	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831

Fuente: Elaboración propia

- 3) Sobre la matriz  $P^{19}$  puede identificarse que los estados i y j se comunican dado que, j es alcanzable desde i y viceversa.
- 4) No presenta estados absorbentes o cerrados. es decir, la probabilidad de transición de ninguno de los estados cumple que  $p_{ii} = 1$ .



**Tabla 30** Cálculo de probabilidades del vector de estado estable

	Componente vector de estado estable*							
	$\pi_1$	$\pi_2$	$\pi_3$	$\pi_4$	$\pi_5$	$\pi_6$	$\pi_7$	$\pi_8$
Probabilidad	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831

\* El subíndice de  $\pi$  indica el posible estado de la solicitud, así: (1) Sin gestión, (2) Radicada, (3) Legalizado, (4) Devuelto, (5) Aplazado, (6) Aprobado, (7) Negado y (8) Otros procesos

Fuente: Elaboración propia

Un estado recurrente  $i$  se dice que es recurrente positivo si comenzando en  $i$ , el tiempo esperado hasta que la cadena vuelva al estado  $i$  es finito. Por el contrario, si el tiempo esperado hasta que la cadena vuelva al estado  $i$  es infinito, se dice que es recurrente nulo. Si el estado  $i$  es recurrente positivo y  $j$  se comunica con  $i$ , entonces  $j$  es recurrente positivo (ver tabla 31). Según puede observarse en la tabla 24 los valores de recurrencia ( $u_{ij} = 1/\pi_j$ ) son recurrentes dado que en todos los casos  $u_{ij} < \infty$ .

**Tabla 31** Cálculo de valores de recurrencia basados en probabilidades de estado estable de la matriz de transición

	Componente vector de estado estable*							
	$\pi_1$	$\pi_2$	$\pi_3$	$\pi_4$	$\pi_5$	$\pi_6$	$\pi_7$	$\pi_8$
Probabilidad	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
Valor de recurrencia	4,52799	2,50239	16,3018	55,2316	93,6046	3,62239	198,688	120,358

\* El subíndice de  $\pi$  indica el posible estado de la solicitud, así: (1) Sin gestión, (2) Radicada, (3) Legalizado, (4) Devuelto, (5) Aplazado, (6) Aprobado, (7) Negado y (8) Otros procesos

Fuente: Elaboración propia

- Un estado  $i$  es periódico con periodo  $k > 1$  si  $k$  es el número más pequeño tal que las trayectorias que conducen al estado  $i$  de regreso al estado  $i$  tienen una longitud que es múltiplo de  $k$ . Si un estado recurrente no es periódico, se denomina aperiódico. Al comparar las potencias 18, 19 y 20 (ver tabla 32) de la matriz de transición  $P$ , se puede observar que en las matrices de potencias pares e impares, las probabilidades en todas las filas permanecen en equilibrio y dado que la matriz se encuentra en estado estable a partir de la potencia 19, queda demostrado que la matriz es aperiódica.

Tabla 32 Probabilidades de la transición de estados en  $P^{20}$

		Estado futuro (j)							
		1	2	3	4	5	6	7	8
Estado actual (i)	1	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	2	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	3	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	4	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	5	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	6	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	7	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
	8	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831

Fuente: Elaboración propia

### **Polinomio característico, vectores y valores propios**

Un vector  $\vec{b} \in \mathbb{R}^n$ , tal que  $\vec{b} \neq \vec{0}$ , se dice que es un vector propio o eigenvector de la matriz P si y solo si se cumple (49):

$$P\vec{b} = \lambda\vec{b} \quad (49)$$

Se dice que el escalar  $\lambda$  es un valor propio o eigenvalor de la matriz P asociado al vector propio  $\vec{b}$  y de manera recíproca también se cumple. Aun cuando la matriz P sea real, sus valores propios asociados pueden ser números complejos. La ecuación (40) puede reescribirse como (50):

$$|P - \lambda I_n| \vec{b} = \vec{0} \quad (50)$$

Donde  $I_n$  es la matriz identidad de orden n. Puesto que por definición,  $\vec{b} \neq \vec{0}$ , (49) se satisface solo si  $|P - \lambda I_n|$  es una matriz singular y  $\vec{b}$  es un elemento del espacio nulo o núcleo de la matriz. En este sentido, su determinante debe ser cero tal como se nota en (51):

$$p(\lambda) = |P - \lambda I_n| = 0 \quad (51)$$

Expandiendo el determinante de (42), se obtiene una ecuación polinómica de n-ésimo orden en  $\lambda$ . Esta expresión se denomina ecuación o polinomio característico de la matriz P. Las raíces del polinomio característico son los valores propios o eigenvalores de la matriz P y los vectores que satisfacen (48) son los vectores propios o eigenvectores

asociados a cada eigenvalor. El cálculo del determinante de la matriz P, queda indicado de la siguiente manera:

		Estado futuro (j)							
		1	2	3	4	5	6	7	8
Estado actual (i)	1	0,00000	0,79835	0,08436	0,00412	0,00412	0,10905	0,00000	0,00000
	2	0,00000	0,47013	0,10649	0,02987	0,01429	0,36234	0,00000	0,01688
	3	0,00000	0,00253	0,00253	0,02525	0,02778	0,92677	0,00253	0,01263
	4	0,00000	0,00000	0,00000	0,08696	0,13043	0,60870	0,13043	0,04348
	5	0,00000	0,00000	0,00000	0,20000	0,00000	0,80000	0,00000	0,00000
	6	0,80000	0,10000	0,00000	0,00000	0,00000	0,10000	0,00000	0,00000
	7	0,00000	0,00000	0,00000	0,00000	0,00000	0,50000	0,50000	0,00000
	8	0,00000	0,92308	0,00000	0,00000	0,00000	0,07692	0,00000	0,00000

λ	0	0	0	0	0	0	0	0
0	λ	0	0	0	0	0	0	0
0	0	λ	0	0	0	0	0	0
0	0	0	λ	0	0	0	0	0
0	0	0	0	λ	0	0	0	0
0	0	0	0	0	λ	0	0	0
0	0	0	0	0	0	λ	0	0
0	0	0	0	0	0	0	λ	0
0	0	0	0	0	0	0	0	λ

Obteniendo el determinante mostrado en la tabla 33, el cual es la base para la obtención del polinomio característico de la matriz de transición P.

**Tabla 33** Determinante de la matriz P

		Estado futuro (j)							
		1	2	3	4	5	6	7	8
Estado actual (i)	1	λ	0,79835	0,08436	0,00412	0,00412	0,10905	0,00000	0,00000
	2	0,00000	0,470 - λ	0,10649	0,02987	0,01429	0,36234	0,00000	0,01688
	3	0,00000	0,00253	0,002 - λ	0,02525	0,02778	0,92677	0,00253	0,01263
	4	0,00000	0,00000	0,00000	0,086 - λ	0,13043	0,60870	0,13043	0,04348
	5	0,00000	0,00000	0,00000	0,20000	λ	0,80000	0,00000	0,00000
	6	0,80000	0,10000	0,00000	0,00000	0,00000	0,100 - λ	0,00000	0,00000
	7	0,00000	0,00000	0,00000	0,00000	0,00000	0,50000	0,500 - λ	0,00000
	8	0,00000	0,92308	0,00000	0,00000	0,00000	0,07692	0,00000	λ

Fuente: Elaboración propia

**Polinomio característico de la matriz de transición P**

Dado que la matriz de transición P tiene dimensión n x n, el polinomio característico de P es el polinomio de grado n con la estructura definida en (52).

$$p(P) = \det|P - \lambda I_n| = a_n \lambda^n + a_{n-1} \lambda^{n-1} + \dots + a_1 \lambda + a_0 \quad (52)$$

Siendo el polinomio característico de la matriz P:

$$p(\lambda) = \lambda^8 - 1,1596\lambda^7 + 0,2626\lambda^6 - 0,2147\lambda^5 + 0,0956\lambda^4 + 0,0193\lambda^3 - 0,0028\lambda^2 - 0,0004\lambda$$

En la tabla 34 se presentan las raíces o eigenvalores del polinomio característico. Como puede observarse, las raíces complejas o imaginarias (2 y 5) aparecen como pares conjugados (3 y 6).

**Tabla 34** Eigenvalores del polinomio característico de la matriz de transición P

$\lambda$	Eigenvalor
1	1
2	-0,1159 + 0,5050i
3	-0,1159 - 0,5050i
4	0,4881
5	-0,1395 + 0,0268i
6	-0,1395 - 0,0268i
7	0,0198
8	0,1626

Fuente: Elaboración propia

**Regularidad de la matriz de transición**

Por definición, si los estados de una cadena de Markov son recurrentes, aperiódicos y se comunican entre sí, se dice que la cadena es ergódica. Dado que la primera raíz del polinomio tiene el valor uno ( $\lambda_1 = 1$ ), se cumple lo descrito en (48). Necesariamente,  $\lambda_1 = 1$  debe ser un valor propio dominante de P ya que, si  $\lambda > 1$ , este tendera a infinito en todas las coordenadas, o si por el contrario  $\lambda < 1$  el limite tendería a cero para todas las coordenadas. Bajo ninguna de las dos últimas circunstancias se consideraría al vector de estado estable como una distribución de probabilidad. Considerando  $\lambda_1 = 1$  a continuación se desarrolla, una representación equivalente de (40) tomando la matriz traspuesta de P y el vector de estado estable traspuesto para demostrar que 1 es un valor propio dominante, es decir, se cumple la igualdad (53):

$$P^T \pi_i^T = \pi_i^T \tag{53}$$

Con este resultado puede evidenciarse que la cadena de Markov es regular

$P^T$	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>0,00000</td><td>0,00000</td><td>0,00000</td><td>0,00000</td><td>0,00000</td><td>0,80000</td><td>0,00000</td><td>0,00000</td></tr> <tr><td>0,79835</td><td>0,47013</td><td>0,00253</td><td>0,00000</td><td>0,00000</td><td>0,10000</td><td>0,00000</td><td>0,92308</td></tr> <tr><td>0,08436</td><td>0,10649</td><td>0,00253</td><td>0,00000</td><td>0,00000</td><td>0,00000</td><td>0,00000</td><td>0,00000</td></tr> <tr><td>0,00412</td><td>0,02987</td><td>0,02525</td><td>0,08696</td><td>0,20000</td><td>0,00000</td><td>0,00000</td><td>0,00000</td></tr> <tr><td>0,00412</td><td>0,01429</td><td>0,02778</td><td>0,13043</td><td>0,00000</td><td>0,00000</td><td>0,00000</td><td>0,00000</td></tr> <tr><td>0,10905</td><td>0,36234</td><td>0,92677</td><td>0,60870</td><td>0,80000</td><td>0,10000</td><td>0,50000</td><td>0,07692</td></tr> <tr><td>0,00000</td><td>0,00000</td><td>0,00253</td><td>0,13043</td><td>0,00000</td><td>0,00000</td><td>0,50000</td><td>0,00000</td></tr> <tr><td>0,00000</td><td>0,01688</td><td>0,01263</td><td>0,04348</td><td>0,00000</td><td>0,00000</td><td>0,00000</td><td>0,00000</td></tr> </table>	0,00000	0,00000	0,00000	0,00000	0,00000	0,80000	0,00000	0,00000	0,79835	0,47013	0,00253	0,00000	0,00000	0,10000	0,00000	0,92308	0,08436	0,10649	0,00253	0,00000	0,00000	0,00000	0,00000	0,00000	0,00412	0,02987	0,02525	0,08696	0,20000	0,00000	0,00000	0,00000	0,00412	0,01429	0,02778	0,13043	0,00000	0,00000	0,00000	0,00000	0,10905	0,36234	0,92677	0,60870	0,80000	0,10000	0,50000	0,07692	0,00000	0,00000	0,00253	0,13043	0,00000	0,00000	0,50000	0,00000	0,00000	0,01688	0,01263	0,04348	0,00000	0,00000	0,00000	0,00000	*	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>0,22085</td></tr> <tr><td>0,39962</td></tr> <tr><td>0,06134</td></tr> <tr><td>0,01811</td></tr> <tr><td>0,01068</td></tr> <tr><td>0,27606</td></tr> <tr><td>0,00503</td></tr> <tr><td>0,00831</td></tr> </table>	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831	=	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>0,22085</td></tr> <tr><td>0,39962</td></tr> <tr><td>0,06134</td></tr> <tr><td>0,01811</td></tr> <tr><td>0,01068</td></tr> <tr><td>0,27606</td></tr> <tr><td>0,00503</td></tr> <tr><td>0,00831</td></tr> </table>	0,22085	0,39962	0,06134	0,01811	0,01068	0,27606	0,00503	0,00831
0,00000	0,00000	0,00000	0,00000	0,00000	0,80000	0,00000	0,00000																																																																														
0,79835	0,47013	0,00253	0,00000	0,00000	0,10000	0,00000	0,92308																																																																														
0,08436	0,10649	0,00253	0,00000	0,00000	0,00000	0,00000	0,00000																																																																														
0,00412	0,02987	0,02525	0,08696	0,20000	0,00000	0,00000	0,00000																																																																														
0,00412	0,01429	0,02778	0,13043	0,00000	0,00000	0,00000	0,00000																																																																														
0,10905	0,36234	0,92677	0,60870	0,80000	0,10000	0,50000	0,07692																																																																														
0,00000	0,00000	0,00253	0,13043	0,00000	0,00000	0,50000	0,00000																																																																														
0,00000	0,01688	0,01263	0,04348	0,00000	0,00000	0,00000	0,00000																																																																														
0,22085																																																																																					
0,39962																																																																																					
0,06134																																																																																					
0,01811																																																																																					
0,01068																																																																																					
0,27606																																																																																					
0,00503																																																																																					
0,00831																																																																																					
0,22085																																																																																					
0,39962																																																																																					
0,06134																																																																																					
0,01811																																																																																					
0,01068																																																																																					
0,27606																																																																																					
0,00503																																																																																					
0,00831																																																																																					

Una vez verificado el supuesto de ergodicidad de la matriz de transición P, se puede concluir que el vector de estado estable calculado y mostrado en la tabla 22, representa adecuadamente el comportamiento a largo plazo de los cambios de estado de las solicitudes recibidas por el proceso.

#### 4.2.4. Construcción del modelo de simulación discreta

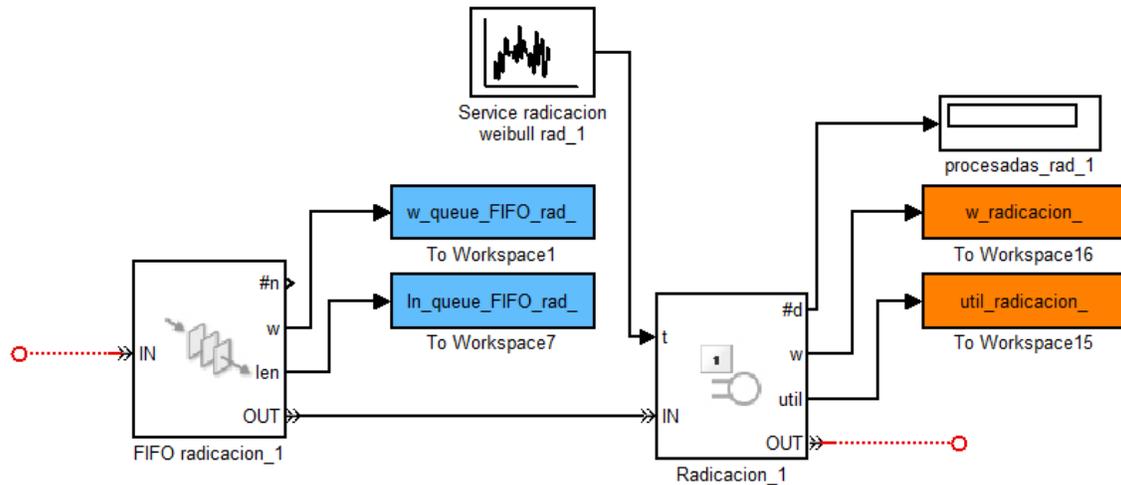
Con el resultado de la modelación del tiempo entre llegadas, tiempo de servicio y transición entre etapas, se realizó la construcción de un modelo de simulación discreta que represente adecuadamente la configuración actual del proceso tomado como caso de estudio y que permita extraer los datos necesarios para el análisis y modelación de las fases II y III de la metodología propuesta. Para tal propósito, se utilizó el entorno SimEvents® dada la posibilidad de generar modelos híbridos con las librerías de lógica difusa e inteligencia artificial disponibles en el software MatLab®. El modelo de simulación discreta incluyó las tres (3) etapas esquematizadas en la figura 6, utilizando los bloques y funciones relacionados en la tabla 35.

**Tabla 35** Bloques y funciones de la librería Simevents® utilizadas en el modelo de simulación

Bloque	Función/Uso
Time-based entity generator	Genera las entidades de acuerdo a la distribución de probabilidad configurada en el bloque Event-based random number
Event-based random number	Genera los tiempos entre llegadas de acuerdo al parámetro calculado de la distribución de probabilidad exponencial del tiempo entre llegadas obtenido en las pruebas de bondad de ajuste
FIFO queue	Representa la línea de espera que se configura frente a cada uno de los servidores. A partir de este bloque se extraen los datos correspondientes a las variables Longitud promedio de la cola ( $L_{ps_{ji}}$ ) y tiempo de espera promedio en espera.
Single server	Representa el recurso que procesa la solicitud. El tiempo de servicio lo recibe de un bloque tipo Event-based random number, cual tiene definido el parámetro de la distribución de probabilidad correspondiente (Weibull y Weibull 3P) para cada etapa. A partir de este bloque se extraen los datos correspondientes a las variables Nivel de utilización ( $\rho_{s_{ji}}$ ) y tiempo de espera promedio en el servidor.
Output switch	Enruta las entidades utilizando el algoritmo round robin, en cual selecciona las colas de destino de manera equitativa y en orden descendente. Enruta las entidades según las probabilidades de transición halladas mediante cadenas de Markov.
Path combiner	Agrupar las entidades enviadas desde las diferentes etapas del proceso y las envía a los output switch para su distribución

Fuente: Elaboración propia

De acuerdo a la información suministrada en el caso de estudio se modelaron cuatro (4), uno (1) y tres (3) servidores en las etapas de radicación, legalización y pertinencia respectivamente. Los servidores de cada una de las tres (3) etapas fueron modelados como recursos independientes con el propósito de extraer de forma separada (ver bloques color azul y naranja de la figura 11) los datos correspondientes a las variables Tiempo de flujo ( $Ft_{ij}$ ), Nivel de utilización ( $\rho_{ij}$ ) y Longitud promedio de la cola ( $Lp_{ij}$ ).



Fuente: Elaboración propia

**Figura 11** Esquema de configuración para la simulación de los recursos de cada etapa

Los datos para la variable tiempo de flujo se calculan como (52):

$$Tf_{ij} = w\_queue\_FIFO_{ij} + w_{ij} \quad (52)$$

Donde los subíndices  $i$  y  $j$  representan el servidor ( $i = 1, 2, 3, 4$ ) y la etapa de proceso ( $j = 1, 2, 3$ ) a la cual pertenece, respectivamente. La expresión (52) indica que el tiempo de flujo del servidor  $i$  en la etapa  $j$  es equivalente a la suma del tiempo de espera en cola más el tiempo de espera en el servidor, este último equivale al tiempo de servicio aleatorio según la distribución de probabilidad correspondiente.

La extracción de los datos hacia el entorno “workspace” permitió generar y posteriormente consolidar los datos de las diez (10) corridas iniciales e independientes generadas con diferentes semillas de la distribución exponencial que representa el tiempo entre llegadas de transacciones al sistema (ver tabla 33). El resultado de la prueba de diferencia de medias registrado en la tabla 26, permite considerar que no hay suficiente evidencia estadística para rechazar la hipótesis que indica que no hay diferencia en el tiempo promedio de servicio para cada clase de servicio (por etapa), dado esto, se asumió que los datos obtenidos sobre la totalidad de recursos por cada etapa pueden ser tratados

como replicas independientes de un mismo servidor para el cálculo mediante (30) del número de réplicas estadísticamente representativas presentado en la tabla 36.

**Tabla 36** Tamaños de muestra obtenidos con las diez corridas iniciales del modelo de simulación

Etapa	Muestra inicial			
	Cola		Servidor	
	w_queue_FIFO	len_queue_FIFO	w_radicacion	util_radicacion
<b>Radicación</b>	360	966	328	369
<b>Legalización</b>	64	210	63	137
<b>Pertinencia</b>	227	536	209	215
<b>Total muestras</b>	651	1.712	600	721

Fuente: Elaboración propia

Como puede observarse en la tabla 37, se generaron en total 6.087 muestras de datos correspondientes a las cuatro (4) variables consideradas para el análisis y modelación a desarrollar en la etapa II.

**Tabla 37** Tamaños de muestra calculados para confiabilidad estadística del 95%

Etapa	Muestra final			
	Cola		Servidor	
	w_queue_FIFO	len_queue_FIFO	w_radicacion	util_radicacion
<b>Radicación</b>	526	431	1.258	713
<b>Legalización</b>	86	137	186	131
<b>Pertinencia</b>	610	197	815	997
<b>Total muestras</b>	1.222	765	2.259	1.841

Fuente: Elaboración propia

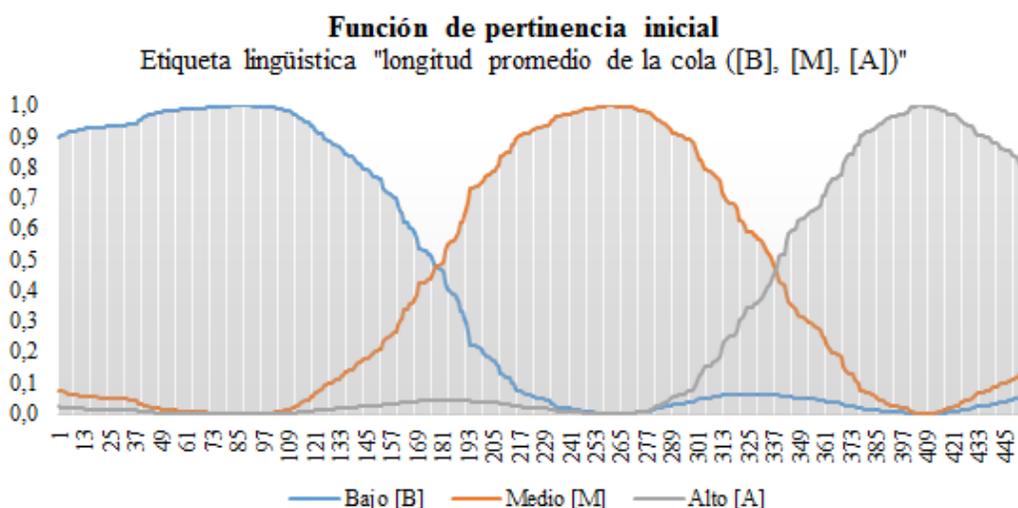
El resultado de la aplicación de las pruebas de bondad de ajuste para las variables de entrada tasa de llegada y tasa de servicio, permitió concluir que en términos de notación de Kendall & Lee, el sistema de redes de colas de referencia, se puede representar como un sistema (M/G/s), lo cual indica que la tasa de llegadas sigue un proceso tipo Poisson mientras que el tiempo de servicio sigue una distribución de probabilidad de tipo general, que para este caso pueden considerarse como distribuciones Erlang, Weibull o Gamma, las cuales fueron seleccionadas para representar el comportamiento de la variable tiempo de servicio dada su afinidad con la distribución exponencial en la característica de pérdida de memoria. Para efecto del modelo de simulación discreta la tasa de servicio se configuro como una distribución de probabilidad tipo Weibull (de uno y tres parámetros dependiendo de la etapa de servicio).

Los parámetros obtenidos de la modelación de los datos correspondientes a la tasa de llegada, tasa de servicio y probabilidades de transición entre etapas, así como las

características actuales de configuración de las estaciones de trabajo por etapa fueron incluidas en el modelo de simulación discreta del sistema de servicio tomado como caso de estudio según puede observarse en el anexo 10, y a partir de dicho modelo se generaron los datos para la definición de las funciones de pertenencia, cuyas actividades se desarrollan en la Fase II.

### 4.3. Fase II: Modelación de conjuntos difusos para variables de desempeño

Los datos obtenidos mediante simulación discreta para las variables Tiempo de flujo ( $F_{t_{ij}}$ ), Nivel de utilización ( $\rho_{ij}$ ), Longitud promedio de la cola ( $L_{p_{ij}}$ ), fueron procesados de manera independiente utilizando el algoritmo Fuzzy c-means, con el propósito de obtener funciones de pertenencia difusas que consideren las etiquetas lingüísticas “bajo(a)”, “medio(a)” y “alto(a)”, es decir agrupar los datos en tres (3) clusters difusos (ver figura 12 y anexos 2 al 7). El algoritmo F c-means fue implementado en el entorno de desarrollo Java Netbeans 8®.



Fuente: Elaboración propia

**Figura 12** Funciones de pertenencia generadas mediante algoritmo fuzzy c-means

Gráficamente se analizó por separado la forma de las funciones de pertenencia obtenidas a partir de la clusterización difusa y se asoció por inspección a cada una, funciones de pertenencia que por similitud en su forma brindarían una aproximación adecuada para posteriormente, obtener los parámetros para la modelación de las funciones de pertenencia obtenidas en el controlador difuso (ver tabla 38). Para ello se utilizaron los criterios: Desviación Absoluta Media (MAD, Mean Absolute Deviation) y Desviación Cuadrática Media (MSD, Mean Squared Deviation). En los anexos 8 y 9 se detalla el resultado obtenido en la aplicación de los dos (2) criterios anteriormente expuestos.

**Tabla 38** Funciones de pertenencia seleccionadas para obtención de parámetros de modelación

Etiqueta lingüística	Función de pertenencia aproximada
Bajo(a)	Z (zshaped), L (lshaped)
Medio(a)	Gauss, Trapezoidal, Triangular
Alto(a)	Sigmoidea, Gamma

Fuente: Elaboración propia

Los valores calculados de los criterios MAD y MSD, permitieron observar que las funciones z, trapezoidal y sigmoidea, en términos de error (o desviación) respecto a las funciones de pertenencia obtenidas mediante la aplicación del algoritmo fuzzy c-means, disponen de los parámetros (y forma) para una mejor aproximación a las funciones de pertenencia de las etiquetas lingüísticas bajo(a), medio(a) y alto(a) respectivamente. En las tablas 39 y 40 se relacionan los parámetros obtenidos para las funciones de pertenencia de las etiquetas lingüísticas asociadas a cada variable.

**Tabla 39** Parámetros aproximados funciones de pertenencia etapa de servicio radicación

Variable	Etiqueta lingüística (Bajo)	Etiqueta lingüística (Medio)	Etiqueta lingüística (Alto)
Nivel de utilización del servidor	Z (zshaped) a = 0,5814 b = 0,9503	Trapezoidal a = 0,5814, b = 0,8003 c = 0,8228, d = 0,9503	Sigmoidea a = 0,8284 b = 0,9503
Longitud promedio de la cola	Z (zshaped) a = 0,5695 b = 1,5989	Trapezoidal a = 0,7743, b = 1,5633 c = 1,8858, d = 3,2494	Sigmoidea a = 1,9299 b = 3,2280
Tiempo de flujo	Z (zshaped) a = 4,8781 b = 6,5492	Trapezoidal a = 4,9713, b = 6,3858 c = 7,1871, d = 9,5114	Sigmoidea a = 6,9788 b = 9,3665

Fuente: Elaboración propia

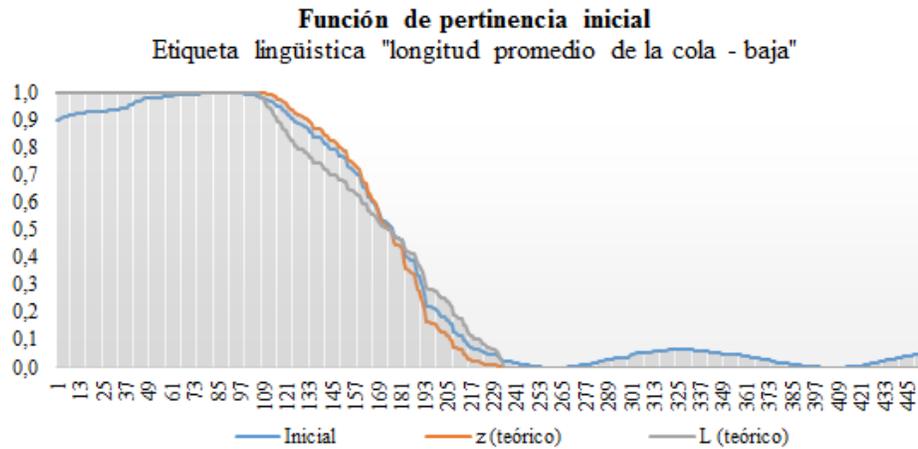
**Tabla 40** Parámetros aproximados funciones de pertenencia etapa de servicio pertinencia

Variable	Etiqueta lingüística (Bajo)	Etiqueta lingüística (Medio)	Etiqueta lingüística (Alto)
Nivel de utilización del servidor	Z (zshaped) a = 0,4387 b = 0,67734	Trapezoidal a = 0,3818, b = 0,6824 c = 0,7169, d = 0,8537	Sigmoidea a = 0,7020 b = 0,8537
Longitud promedio de la cola	Z (zshaped) a = 0,4386 b = 1,1836	Trapezoidal a = 0,4715, b = 1,2373 c = 1,3658, d = 2,3585	Sigmoidea a = 0,7020 b = 0,8537
Tiempo de flujo	Z (zshaped) a = 6,0928 b = 8,8443	Trapezoidal a = 6,2203, b = 8,8443 c = 9,4551, d = 12,667	Sigmoidea a = 9,3777 b = 12,667

Fuente: Elaboración propia

A modo de ejemplo, en la figura 13 se representa el ajuste aproximado de las funciones de pertenencia teóricas z y l, a la función de pertenencia obtenida mediante la clusterización

de los datos generados mediante la simulación del sistema en la etapa “pertinencia”, para la variable “longitud promedio de la cola, baja”.

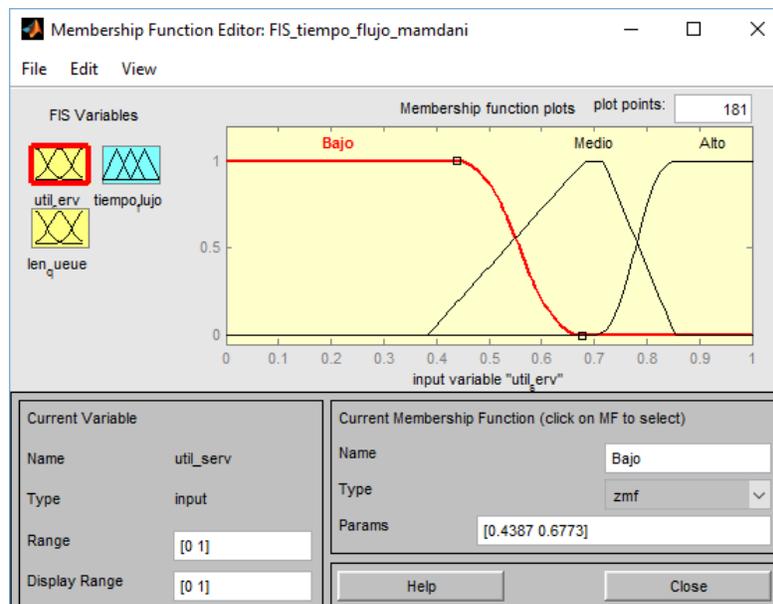


Fuente: Elaboración propia

**Figura 13** Ajuste aproximado de funciones de pertenencia teóricas seleccionadas

### **Configuración del controlador difuso Mamdani**

Una vez identificados los parámetros aproximados de las funciones de pertenencia correspondientes a las etiquetas lingüísticas de cada una de las tres (3) variables (ver anexos 11 y 12), se realizó la configuración del controlador difuso utilizando la librería “Fuzzy Logic Toolbox” de Matlab®. A modo de ejemplo en la Figura 14, puede observarse la representación de las funciones de pertenencia en la interfaz gráfica de Matlab®.



**Figura 14** Funciones de pertenencia del controlador Mamdani etapa pertinencia

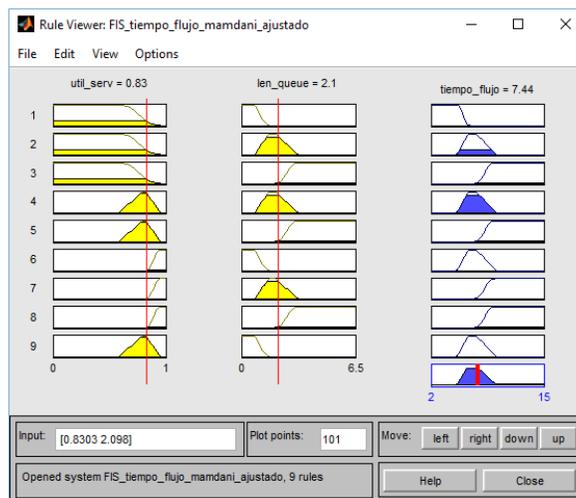
Cargados los parámetros de las funciones de pertenencia de los antecedentes (Nivel de utilización del servidor, Longitud promedio de la cola) y el consecuente (Tiempo de flujo), de definieron un total de ocho (8) reglas que representan las combinaciones más razonables para la inferencia del tiempo de flujo de un servidor (ver tabla 41). Durante la definición de las reglas del sistema de inferencia, se descartó la regla “*si nivel de utilización es bajo y longitud promedio de la cola es alta entonces tiempo de flujo es bajo, medio o alto*” dado que no es posible que dicha regla se presente durante el procesamiento de una transacción en la red de colas

**Tabla 41** Reglas difusas configuradas para controladores Mamdani (radicación y pertinencia)

Regla activada	Nivel de utilización del servidor	Longitud promedio de la cola	Variable de salida (tiempo de flujo)
1	Bajo	Baja	Baja
2	Bajo	Media	Medio
3	Medio	Baja	Medio
4	Medio	Media	Medio
5	Medio	Alta	Alto
6	Alto	Baja	Medio
7	Alto </td <td>Media</td> <td>Alto</td>	Media	Alto
8	Alto	Alta	Alto

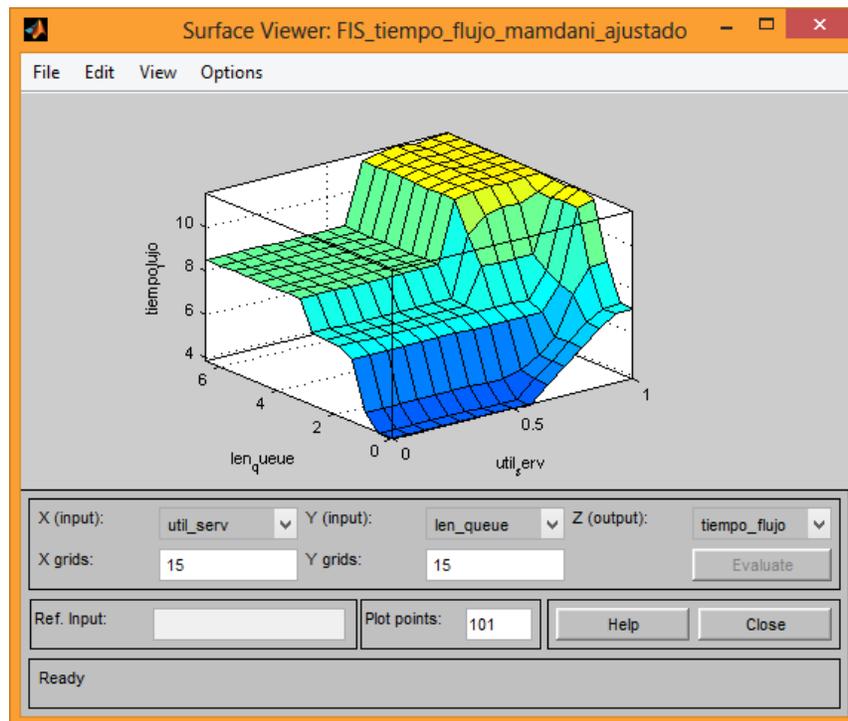
Fuente: Elaboración propia

En la figura 15, puede observarse (a modo de ejemplo), el resultado de la configuración de las reglas difusas en el control difuso Mamdani para la etapa de pertinencia. Como puede observarse la interfaz representa gráficamente los valores nítidos de las variables del antecedente y el consecuente.



**Figura 15** Interfaz gráfica para observación de reglas difusas configuradas

En la figura 16 (a modo de ejemplo), puede observarse la superficie de respuesta del controlador Mamdani para la etapa de pertinencia donde se permite examinar la influencia de las variables nivel de utilización ( $\rho_{ji}$ ) y longitud promedio de la cola ( $L_{pji}$ ) sobre la variable de salida tiempo de flujo ( $Ft_{ji}$ ). Puede analizarse a partir de esto que, si las variables nivel de utilización (del servidor) y longitud promedio de la cola toman valores nítidos entre el rango [0.6, 1] y [0, 1], el tiempo de flujo de una transacción se incrementara rápidamente de 7 a 11 minutos, de igual manera a valores inferiores en el nivel de utilización del servidor a los anteriormente descritos, el tiempo de flujo se incrementara de forma sostenida para valores de longitud promedio de las cola en los rangos [0, 1.5] y [2, 4] hasta llegar a un máximo aproximado de 12 minutos.



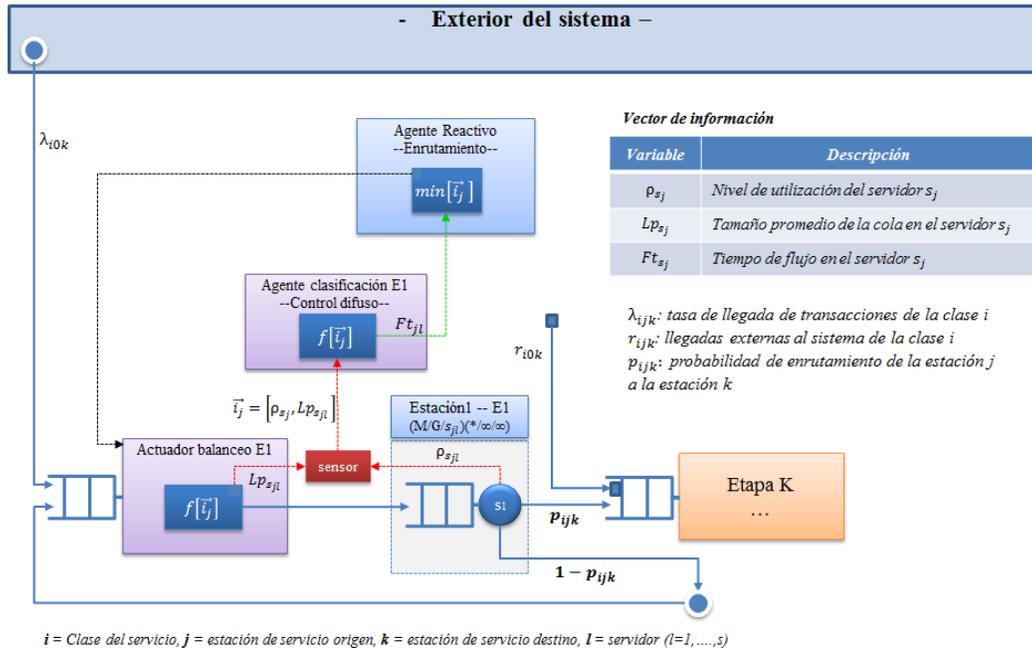
**Figura 16** Superficie de respuesta controlador Mamdani etapa radicación

Una vez configurados los controladores difusos, se construyó una función de uso exclusivo del agente reactivo para definir el puerto de salida (del switch para el enrutamiento) de acuerdo al criterio del mínimo valor de tiempo de flujo inferido por cada controlador difuso Mamdani.

Dada la necesidad de controlar cada recurso de manera independiente, se implementaron un total de cuatro (4) y tres (3) controladores difusos para cada servidor de las etapas de radicación y pertinencia respectivamente.

#### 4.4. Fase III: Definición de la arquitectura del sistema multi-agente

Una vez implementados los controladores difusos tipo Mamdani en el entorno de simulación discreta, se definió la arquitectura del sistema multi-agente como puede observarse en la figura 17.



Fuente: Elaboración propia

**Figura 17** Arquitectura implementada para el sistema multi-agente

En la tabla 42, se describen las funciones implementadas (en ambiente de simulación) para cada uno de los cuatro (4) tipos de agentes incluidos en el diseño.

**Tabla 42** Funciones implementadas en entorno de simulación por cada tipo de agente

Clasificación agente	Condiciones de operación
Sensor	Recolecta del servidor los datos de las variables $\rho_{sj}$ y $Lp_{sj}$ y los envía al agente de clasificación.
Agente de clasificación	Recibe los datos capturados por el sensor, y los carga al FIS tipo Mamdani, para inferir el valor estimado de la variable de salida $Ft_{sjl}$
Agente reactivo	Recopila los valores estimados de la variable $Ft_{sjl}$ generados por los agentes de clasificación de cada etapa. Solicita el enrutamiento de la transacción entrante al servidor con mínimo $Ft_{sjl}$ .
Actuador de balanceo	Recibe la señal de enrutamiento y la ejecuta.

Fuente: Elaboración propia

Como se mencionó en la fase II, se implementaron un total de siete controladores difusos (agentes de clasificación) encargados de recibir la información generada por el sensor configurado para cada variable y por cada recurso (servidor y línea de espera) durante la ejecución de la simulación discreta, para posteriormente transmitirla a la función (53), la cual corresponde a la etapa de radicación. Bajo las mismas condiciones se construyó el código de la función para la etapa de pertinencia (ver anexo 13).

```
function p = findmin(R1S1, R1S2, R2S1, R2S2,
R3S1, R3S2, R4S1, R4S2)

%#codegen
n1=0;
n2=0;
n3=0;
n4=0;
coder.extrinsic('readfis','evalfis');
a=readfis('nombre_sistema_inferencia');
n1=evalfis([R1S1 R1S2], a);
n2=evalfis([R2S1 R2S2], a);
n3=evalfis([R3S1 R3S2], a);
n4=evalfis([R4S1 R4S2], a);
if(n1<=n2 && n1<=n3 && n1<=n4)
    p=1;
elseif(n2<=n3 && n2<=n4)
    p=2;
elseif(n3<=n4)
    p=3;
else
    p=4;
end
```

(53)

La función mínimo relacionada (y correspondiente a la lógica del agente reactivo) mostrada en (53), recibe de los sensores ubicados en cada servidor la información de las variables consideradas en el diseño, posteriormente ejecuta en secuencia cada controlador difuso (evalfis [nivel\_utilizacion longitud\_de\_la\_cola], readfis), y a partir de los valores inferidos de tiempo de flujo evalúa el valor mínimo, asignando a la condición cumplida un valor discreto que representa el puerto de salida hacia el cual, el actuador de balanceo (output switch) debe enrutar la transacción entrante. En el anexo 14, puede observarse la secuencia de enrutamiento generada por cada uno de los switch ubicados a la entrada de cada etapa durante una corrida del modelo de simulación.

La implementación de la función mínimo (findmin), se realizó en el entorno de simulación discreta de SimEvents utilizando un bloque MATLAB function. Los demás agentes no

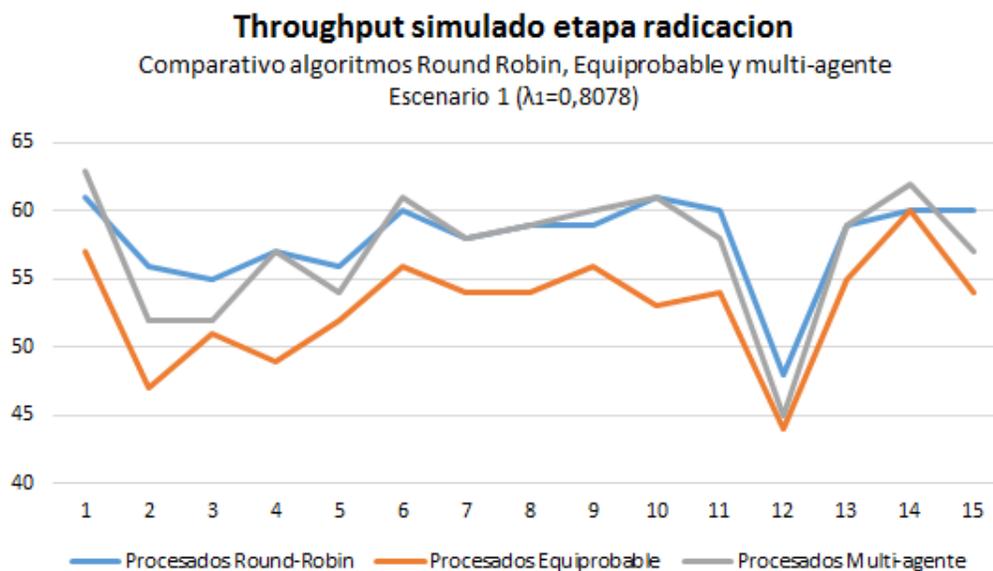
requirieron la construcción de código ya que la configuración se realizó desde las opciones por defecto definidas para cada bloque. En el anexo 10, se presenta el modelo de simulación construido, el cual incluye la implementación de los agentes descritos en las fases I, II y III para el diseño del sistema multi-agente propuesto.

## Resultados

Con el propósito de disponer de un criterio valido para comparar el desempeño del método de inferencia, decisión y enrutamiento del sistema multi-agente respecto a otros dos (2) algoritmos de enrutamiento convencionales, se generaron tres (3) escenarios de simulación variando el parámetro tiempo entre llegadas tomando valores de  $\lambda_1 = 0,8078$ ,  $\lambda_2 = 1,2486$  y  $\lambda_3 = 1,0012$ , los cuales representan las franjas horarias comprendidas entre las 07:00 a 07:59, 10:00 a 10:59 y 14:00 a 14:59, respectivamente. Cada uno de los tres (3) escenarios anteriormente mencionados, consto de 15 réplicas independientes cada una con duración de 60 minutos. Los datos obtenidos de la simulación se utilizaron para contrastar el efecto sobre la variable de desempeño “throughput” (promedio por etapa) respecto a dos (2) de los algoritmos embebidos en el bloque “output switch” de SimEvents:

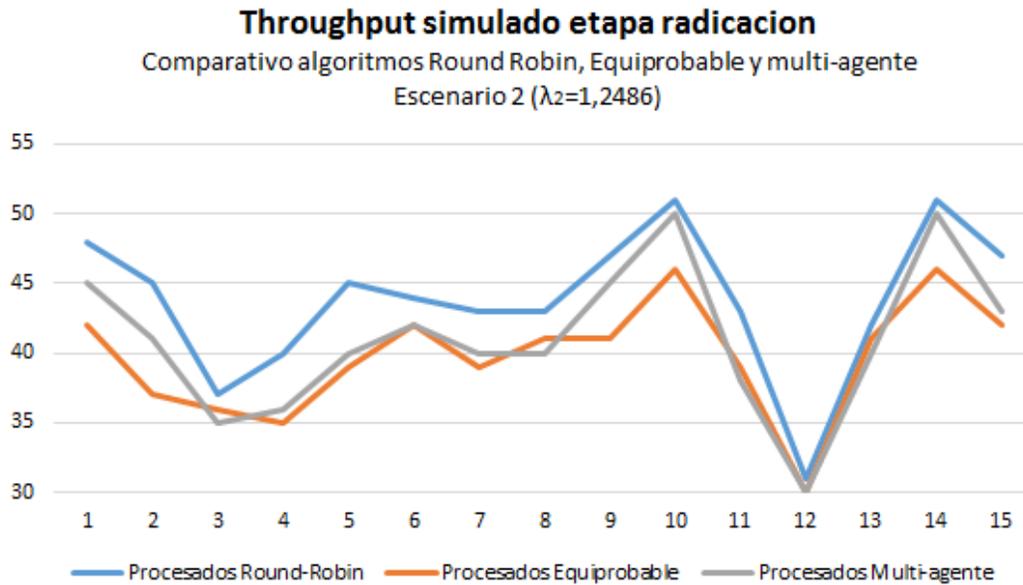
- Round - robin (selecciona el puerto de salida en orden secuencial descendente).
- Equiprobable (o de selección aleatoria del puerto de salida).

En las figuras 18, 19 y 20, se presenta gráficamente el comportamiento de los dos (2) algoritmos de referencia y el sistema multi-agente. Como se puede observar en los tres (3) escenarios, el throughput obtenido por los recursos de la etapa de radicación con la implementación del sistema multi.-agente correspondiente, es considerablemente superior al generado por el algoritmo de enrutamiento equiprobable. En comparación con el algoritmo Round-Robin, el sistema multi-agente presenta un desempeño inferior.



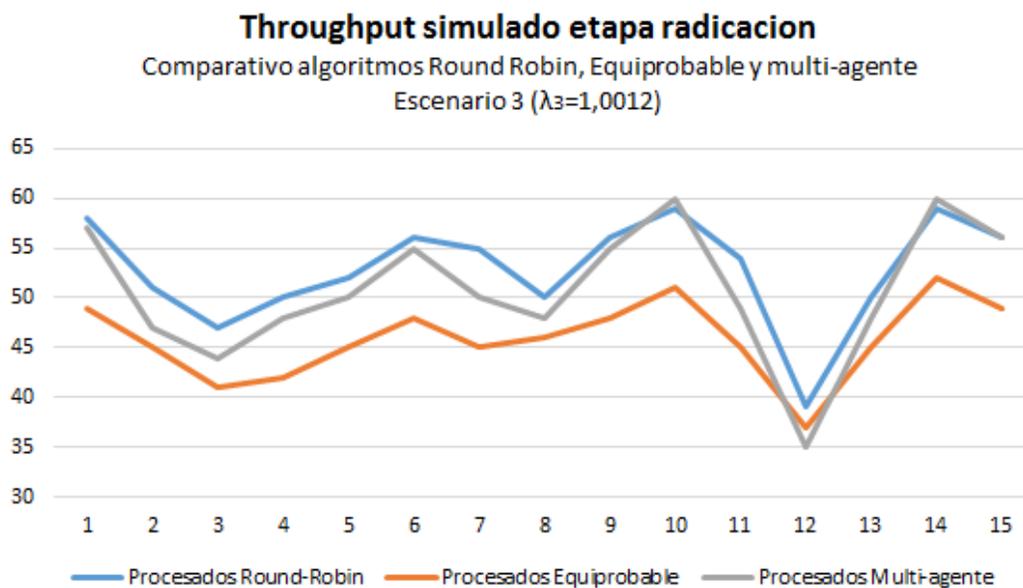
Fuente: Elaboración propia

**Figura 18** Comparación del throughput simulado en el escenario 1



Fuente: Elaboración propia

**Figura 19** Comparación del throughput simulado en el escenario 2



Fuente: Elaboración propia

**Figura 20** Comparación del throughput simulado en el escenario 3

Para efectos de la comparar estadísticamente los anteriores comportamientos, se condujeron las pruebas estadísticas diferencia de medias y análisis de varianza (ANOVA), sobre los datos de throughput obtenidos en cada una de las 15 réplicas de los tres

escenarios considerados de tiempo entre llegadas y cuyos resultados se describen a continuación.

**Prueba de diferencia de medias para comparar el throughput generado por cada etapa**

Con los datos obtenidos en modelos de simulación independientes (generados en los tres (3) escenarios) para cada algoritmo, se condujo una comparación de medias con el propósito de probar a un nivel de significancia del 5% la hipótesis que (en promedio) el throughput (transacciones procesadas) no tiene una diferencia representativa al compararlo de un algoritmo de enrutamiento a otro.

Para ello, se formula sobre el throughput por etapa de proceso la siguiente hipótesis:

H<sub>0</sub>: No hay diferencia en la cantidad de transacciones procesadas por tipo de algoritmo de enrutamiento ( $\mu_1 - \mu_2 = 0$ ).

H<sub>1</sub>: Si hay diferencia en la cantidad de transacciones procesadas por tipo de algoritmo de enrutamiento ( $\mu_1 - \mu_2 \neq 0$ ).

Los resultados consolidados de la prueba t de medias pareadas (ver tabla 43), permiten evidenciar que el throughput simulado para cada uno de los tres (3) algoritmos (incluido el implementado por el sistema multi-agente) son diferentes entre sí.

**Tabla 43** Resultados prueba de diferencia de medias sobre el throughput promedio por etapa para los algoritmos de enrutamiento Round-Robin, Equiprobable y Sistema Multi-Agente (MAS).

Hipotesis nula	Radicación		Pertinencia	
	$\mu_{\text{roundrobin}} = \mu_{\text{MAS}}$	$\mu_{\text{equiprobable}} = \mu_{\text{MAS}}$	$\mu_{\text{roundrobin}} = \mu_{\text{MAS}}$	$\mu_{\text{equiprobable}} = \mu_{\text{MAS}}$
n	15	15	15	15
F	-8,11055	5,16571	5,16571	-4,26546
Valor critico F	2,01537	2,01537	2,01537	2,01537
Decisión	Rechaza H <sub>0</sub>	Rechaza H <sub>0</sub>	Rechaza H <sub>0</sub>	Rechaza H <sub>0</sub>

Fuente: Elaboración propia

Según los resultados de la prueba t de diferencia de medias pareadas, se puede concluir a un nivel de significancia del 5% que el throughput promedio logrado con el algoritmo de enrutamiento implementado en el sistema multi-agente, tiene un efecto diferente sobre el volumen de unidades procesadas por cada etapa, al compararlo con los algoritmos Round-Robin y Equiprobable.

**Prueba ANOVA para contrastar el efecto de los algoritmos contrastados sobre el throughput generado por cada etapa**

Con el propósito de contrastar el efecto sobre el throughput (volumen de transacciones procesadas por etapa) de los tres (3) algoritmos simulados (tratamientos) en cada una de las tres (3) replicas (grupos), se condujo un análisis de varianza de dos factores en pares (Round-Robin respecto MAS y Equiprobable respecto a MAS), bajo la siguiente prueba de hipótesis:

H<sub>0</sub>: No hay diferencia en la cantidad media de transacciones procesadas por tipo de algoritmo de enrutamiento ( $\mu_1 = \mu_2$ ).

H<sub>1</sub>: Si hay diferencia en la cantidad media de transacciones procesadas por tipo de algoritmo de enrutamiento ( $\mu_1 \neq \mu_2$ ).

El resultado de la prueba ANOVA que contrasta cada pareja de tratamientos, se presenta en la tabla 44.

**Tabla 44** Resultados ANOVA sobre el efecto en el throughput promedio por etapa para los algoritmos de enrutamiento Round-Robin, Equiprobable y Sistema Multi-Agente (MAS).

Hipotesis nula	Radicación		Pertinencia	
	$\mu_{\text{roundrobin}} = \mu_{\text{MAS}}$	$\mu_{\text{equiprobable}} = \mu_{\text{MAS}}$	$\mu_{\text{roundrobin}} = \mu_{\text{MAS}}$	$\mu_{\text{equiprobable}} = \mu_{\text{MAS}}$
n	15	15	15	15
F (muestra)	65,97612	68,88517	0,75712	17,77845
Valor critico F	3,10516	3,10516	3,10516	3,10516
Decisión	Rechaza H <sub>0</sub>	Rechaza H <sub>0</sub>	No Rechaza H <sub>0</sub>	Rechaza H <sub>0</sub>

Fuente: Elaboración propia

Según los valores F (de la muestra) incluidos en los resultados de la prueba de varianza registrados en la tabla 42, se puede concluir a un nivel de nivel de significancia del 5% que el valor medio correspondiente, al efecto sobre el throughput promedio logrado con el algoritmo de enrutamiento diseñado e implementado en el modelo de simulación del sistema multi-agente para la etapa de pertinencia, respecto al resultado del algoritmo Round-Robin, no presenta una diferencia significativa en el throughput de la etapa de pertinencia que corresponde al throughput global del sistema. Por otro lado, el resultado de la prueba ANOVA permite observar que hay suficiente evidencia estadística para rechazar la hipótesis que indica que el algoritmo de enrutamiento generado por el sistema multi-agente, no presenta diferencias significativas sobre la cantidad promedio de transacciones procesadas por cada etapa del proceso al compararlo con Round-Robin (excepto en la etapa de pertinencia) y (en ambas etapas para el algoritmo de enrutamiento Equiprobable).

### **Valores promedio del throughput generado por algoritmo en cada etapa**

Tomando como base los resultados de la prueba t de diferencia de medias pareadas y la prueba de análisis de varianza ANOVA, donde en ambos casos se concluye que el efecto sobre el throughput generado es diferente entre los tres (3) algoritmos de enrutamiento simulados, se consolidaron los datos obtenidos en cada escenario para cada etapa, identificando sobre una muestra de 45 datos para los tres (3) algoritmos, que en promedio el sistema multi-agente registro un desempeño mayor respecto al volumen de transacciones procesadas (ver tabla 45), en comparación con el resultado (bajo las mismas condiciones de operación) del algoritmo equiprobable. Por ejemplo, en la etapa de radicación comparando dichos algoritmos, el sistema multi-agente logro contribuir en el procesamiento de cerca de tres (3) y dos (2) transacciones en las etapas de radicación y pertinencia respectivamente.

**Tabla 45** Medias muestrales del throughput generado por algoritmo de enrutamiento

Etapa	Throughput promedio/Etapa Algoritmo de enrutamiento		
	Round-Robin	Equiprobable	MAS
<b>Radicacion</b>	51,5111	46,2222	49,6667
<b>Pertinencia</b>	32,7778	29,6000	31,0889

Fuente: Elaboración propia

Según los datos anteriores, las etapas de radicación y pertinencia registran un mayor valor promedio de transacciones procesadas con la utilización del algoritmo Round-Robin. No obstante y según el resultado de la prueba de análisis de varianza, en términos generales el algoritmo Round-Robin y el implementado por el sistema multi-agente, representan valor medio poblacional estadísticamente equivalente en el throughput global del sistema, lo cual permite concluir que ambos algoritmos son alternativas viables para la mejora en el desempeño del proceso.

La rigurosidad de las técnicas estadísticas seleccionadas para la modelación de las variables básicas en la representación de un sistema de redes de colas, permitió incorporar la aleatoriedad inherente a dichas variables evitando con esto la incorporación de supuestos que puedan distorsionar los resultados obtenidos mediante el modelo construido de simulación discreta. La forma de modelación de las variables consideradas para la construcción del sistema de inferencia difuso, permiten entender en términos de razonamiento humano su comportamiento y adicionalmente, incorporan aspectos como la incertidumbre dado que las técnicas analíticas vigentes modelan las mismas variables en términos de promedios. El enfoque de sistema multi-agente representa una alternativa viable para el control en tiempo real del tipo de procesos incluidos en el alcance del presente documento, dado que estadísticamente se demostró con la simulación de diferentes escenarios de tiempo entre llegadas de transacciones del exterior del proceso

analizado, que dispone de un desempeño pertinente respecto a mediciones locales y globales de desempeño en comparación con otros algoritmos convencionales de enrutamiento de transacciones. Las fases definidas dentro del diseño metodológico demostraron suficiencia para la definición de la arquitectura de un sistema multi-agente, dado que las actividades y técnicas incluidas dentro de cada fase permitieron sistemáticamente recolectar la información necesaria para la obtención del modelo presentado en la fase III del diseño metodológico y anexos correspondientes de este documento. Por otro lado la utilización de tecnologías basadas en agentes, a diferencia de los algoritmos de enrutamiento convencionales, dispone de la versatilidad suficiente para incluir dentro del análisis y cuantificar el impacto de las variables que a criterio del tomador de decisiones deban considerarse como indispensables para medir el desempeño de los procesos en tiempo real.

## CAPITULO V: CONCLUSIONES Y RECOMENDACIONES

En el presente capítulo se compendian las conclusiones y recomendaciones obtenidas durante la aplicación del marco metodológico, respecto a la hipótesis de la investigación y los objetivos general y específicos

### **Conclusiones**

La integración de técnicas analíticas de investigación de operaciones, simulación discreta y tecnologías basadas en agentes inteligentes representa una alternativa versátil para el análisis, diseño y mejora de las operaciones, puesto dicha combinación permite incorporar en tiempo real el control de aspectos críticos para el desempeño del proceso, considerando de manera especial el impacto de los protocolos de enrutamiento que determinan el flujo de transacciones entre etapas de los procesos, garantizando así el balanceo en el uso de los recursos y la suficiencia en la capacidad del sistema en un rango amplio de variables de operación.

La caracterización del sistema de redes de colas tomado como caso de estudio, en términos de las técnicas estadísticas y de investigación de operaciones seleccionadas para la modelación de las variables básicas para su representación, permitió incorporar la aleatoriedad inherente a dichas variables evitando con esto la incorporación de supuestos que puedan distorsionar los resultados obtenidos mediante modelos de simulación discreta. El modelo de simulación construido permitió representar la configuración real del sistema incluyendo los flujos de información, derivando con ello la incorporación de las medidas de desempeño clave para este tipo de procesos, las cuales no son consideradas para el control del sistema real. A partir de esto, se demostraron a la empresa que suministro el caso de estudio, los beneficios en la mejora de la eficiencia del proceso al implementar las medidas de desempeño analizadas durante la presente investigación.

El nivel de desarrollo de los paquetes informáticos de última generación para este tipo de aplicaciones, facilita al tomador de decisiones las labores de diseño de sistemas de control que consideren la interacción de múltiples variables y en consecuencia la aplicación de diversas técnicas para la modelación, dado que en ambiente de simulación se pueden generar modelos híbridos con técnicas de investigación de operaciones e inteligencia artificial, considerando especialmente las ventajas demostradas en el uso de estas últimas para la definición de controles de proceso en tiempo real.

Las fases definidas dentro del diseño metodológico demostraron suficiencia para la definición de la arquitectura de un sistema multi-agente, dado que las actividades y técnicas incluidas dentro de cada fase permitieron sistemáticamente recolectar la información necesaria para la obtención del modelo presentado en la fase III del diseño metodológico y anexos correspondientes de este documento.

El enfoque de sistema multi-agente representa una alternativa viable para el control en tiempo real del tipo de procesos incluidos en el alcance del presente documento, dado que estadísticamente se demostró con la simulación de diferentes escenarios de tiempo entre llegadas de transacciones del exterior del proceso analizado, que en términos de throughput dispone de un desempeño pertinente respecto a mediciones locales y globales de desempeño en comparación con otros algoritmos convencionales de enrutamiento de transacciones.

### ***Recomendaciones***

La implementación del sistema multi-agente propuesto en ambiente real, puede realizarse siguiendo los parámetros de diseño identificados en la presente investigación. Como consideración adicional, durante la implementación puede considerarse la definición de una política de máxima longitud de tiempo de flujo y tamaño de transacciones en cola que permita al tomador de decisiones activar o desactivar recursos para asumir variaciones no previstas en la demanda de solicitudes recibidas.

En trabajos futuros puede incluirse dentro del sistema multi-agente desarrollado, el control de la interacción entre etapas, definiendo para ello condiciones de negociación y coordinación entre agentes, restringiendo así posibles conflictos en las metas individuales de cada agente que puedan impedir la mejora del throughput global del sistema. Para tal propósito podría utilizarse teoría de juegos como una alternativa, para identificar la política de “scheduling” más apropiada para el balanceo de recursos y el logro de las medidas de desempeño del sistema definidas por el decisor del sistema.

## BIBLIOGRAFIA

- F.R.B. Cruz; G. Kendall, L. White; A.R. Duarte; L.C. Brito (2012). "Throughput maximization of queueing networks with simultaneous minimization of service rates and buffers".
- J. Pazos, A. Suarez, R. Díaz (2003). Teoría de colas y simulación de eventos discretos. Prentice Hall. 137 - 152.
- B. Rabta, R. Schodl, G. Reiner; J. Fichtinger (2013). "A hybrid analysis method for multi-class queueing networks with multi-server nodes". *Decision Support Systems*. 54, 1541 - 1547.
- F. Yang (2010). Neural network metamodeling for cycle time-throughput profiles in manufacturing. *European Journal of Operational Research*. 205, 172 - 185.
- D.F. Fonseca, D. Navarrese (2002). Artificial neural networks for job shop simulation. *Advanced Engineering Informatics*. 16, 241 - 246.
- D. Gupta (2013). "Queueing models for healthcare operations". Handbook of Healthcare Operations Management: Methods and Applications. *International Series in Operations Research & Management Science*. 184. DOI 10.1007/978-1-46-5885-2\_2.
- R.O. Baldwin; N.J. Davis IV., S.F. Midkiff; J.E. Kobza (2003). "Queueing networks analysis: concepts, terminology, and methods". *Journal of systems and software*. 66, 99 - 117.
- F. Baskett, K. Chandy, R. Muntz, F. Palacios (1975). "Open, closed, and mixed networks of queues with different classes of customers". *Journal of Association for Computing Machinery* 22 (2), 248–260.
- W. Hopp, L. Spearman (2000). *Factory Physics*. 2 edn. McGraw-Hill Higher Education, New York.
- M. Jain, S. Maheshwari, K.P.S. Baguel (2007). "Queueing network modelling of flexible manufacturing system using mean value analysis". *Applied Mathematical Modelling*. 32, 700 - 711.
- F.R.B. Cruz (2009). "Optimizing the throughput, service rate and buffer allocation in the finite queueing networks". *Electronic notes in discrete mathematics*. 35, 163 - 168.
- F. Yang, J. Liu (2012). "Simulation-based transfer function modeling for transient analysis of general queueing systems". *European Journal of Operational Research*. 223, 150 - 166.
- K. Nassar, Z. AL-Musawi (2013). "Fuzzy neural network for dynamic load balancing of nodes for ad-hoc network using". *Journal of Basrah Researchers*.

- A. Zadeh, Z.S. Faiz, S.M. Asadzadeh, R. Tavakkoli-Moghamad (2011). "An integrated artificial neural network-computer simulation for optimization of complex tandem queue systems". *Mathematics and Computers in Simulation*. 82, 666 – 678.
- J. Araujo, K. Kienitz, S. Sandri, J. Da Silva (2004). "Fuzzy reference gain-scheduling approach as intelligent agents: FRGS agent".
- M. Wooldridge, N.R. Jennings (1995). "Intelligent agents: theory and practice. The knowledge engineering review", 10(2): 115 - 152.
- J. Botti (2000). Agentes inteligentes: el siguiente paso en la inteligencia artificial. NOVATICA 95 - 99. [www.ati.es/novatica/2000/145/vjulia-145.pdf](http://www.ati.es/novatica/2000/145/vjulia-145.pdf).
- S. Kinsel (2005). Agent based modeling (ABM). [web.ics.purdue.edu/~hwan/IE680/Final%20Presentation/Steve/ABM.ppt](http://web.ics.purdue.edu/~hwan/IE680/Final%20Presentation/Steve/ABM.ppt).
- M. Huhns, M.P. Singh (1998). "Readings in agents". Chapter 1, 1 - 24.
- T. Chen (2011). "A self adaptive agent-based fuzzy-neural scheduling system for a wafer fabrication factory". *Expert Systems with Applications*. 38, 7158 - 7168.
- R. Smith (1979). "The contract net protocol: High level communication and control in a distributed problem solver". Proceedings of the first international conference on distributed computing systems IEEE New York. 185 - 192.
- M. Wooldridge (2002). An introduction to multiagent systems. John Wiley and Sons Ltd.
- T. Sandholm (2002). eMediator: a next generation electronic commerce server. Agent technology for electronic commerce. 18(4) 656 - 676.
- N.R. Jennings (2002). An agent based approach for building complex software systems. ACM. 44(4) 35 – 41.
- S. Sharifian, S. Motamedi, M. Akbari (2011). "A predictive and probabilistic load-balancing algorithm for cluster-based web servers". *Applied soft computing*. 11, 970 - 981.
- A. Madureira, I. Pereira, P. Pereira, A. Abraham (2014). "Negotiation mechanism for self-organized scheduling system with collective intelligence". *Neurocomputing*. 132, 97 - 110.
- M.A. Metaweij, S.A. Ghoneim, S.M. Haggag, S. Nassar (2012). "Load balancing in distributed multi-agent computing systems". *Ain shams engineering journal*. 3, 237 - 249. May 23.
- A. Zadeh (2010). "Fuzzy sets. Information and Control". 8, 338 – 353. 1965.
- P. Cruz. "Inteligencia artificial con aplicaciones a la ingenieria". Alfaomega grupo editor.
- E.H. Mamdani, S. Assilian (1975). "An experiment in linguistic synthesis with a fuzzy logic controller" *International Journal of Man-Machine Studies*, Vol. 7, No. 1, pp. 1-13.

- M. Sugeno (1985). *Industrial applications of fuzzy control*, Elsevier Science Pub. Co.
- V. Kansal, A. Kaur (2013). "Comparison of Mamdani-type and Sugeno-type FIS for water flow rate control in a rawmill". *International journal of scientific & engineering research*. 4(6), 2580 - 2584. 2013.
- H. Zaher, A.E. Kandil, R. Fahmy (2014). "Comparison of Mamdani and Sugeno Fuzzy Inference systems for prediction (with application to prices of fund in Egypt)". *British journal of mathematics & computer science*. 4(21), 3014 - 3022.
- Dunn J.C. (1973). "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters", *Journal of Cybernetics* 3: 32-57.
- MacQueen J.B. (1967). "Some Methods for classification and Analysis of Multivariate Observations, *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*", Berkeley, University of California Press, 1:281-297.
- N. Izady, D. Worthington (2011). "Setting staffing requirements for time dependent queueing networks: The case of accident and emergency departments". *European Journal of Operational Research*. 219, 521 - 540.
- M. Chambers, C.A. Mount-Campbell (2000). "Process optimization via neural network metamodeling". *Int. J. Production Economics*. 79, 93 - 100.
- M.A. Adibi, M. Zandieh, M. Amiri (2010). "Multi-objective scheduling of dynamic job shop using variable neighborhood search". *Expert Systems with Applications*. 37, 282 - 287.
- L. Lin, M. Gen, Y. Liang, O. Katsuhisa (2012). "A hybrid EA for reactive flexible job-shop scheduling". *Procedia Computer Science*. 12, 110 - 115.
- T. Qiu, L. Feng, H. Jiang, S. Weifeng (2013). "Queueing model analysis and scheduling strategy for embedded multi-core SoC based on task priority". *Computers and Electrical Engineering*. 39, 24 - 33.
- A. Noroozi, H. Mokhtari, I.N.K. Abadi (2013). "Research on computational intelligence algorithms with adaptive learning approach for scheduling problems with batch processing machines". *Neurocomputing*. 4, 190 - 203.
- C. Lakshmi, A. Sivakumar (2013). "Application of queueing theory in health care: A literature review". *Operations research for health care*. 2, 25 - 39.
- C. Granja, B. Almada-Lobo, F. Janela, J. Seabra, A. Mendes (2014). "An optimization based on simulation approach to the patient admission scheduling using a linear programming algorithm". 52, 427 - 427.

- P. Vanberkel, R. Boucherie, E. Hans, J. Hurink (2014). Optimizing the strategic patient mix combining queueing theory and dynamic programming. *Computers & operations research*. 43, 271 - 279.
- T. Yang, Y. Shen (2011). "The dynamic transfer batch-size decision for thin film transistor-liquid crystal display array manufacturing by artificial neural network". *Computers & Industrial Engineering*. 60, 769 - 776.
- C. Liu, Y. Li, W. Shao, W. Shen (2015). "An adaptive process planning method based on features and intelligent agents for the manufacturing of large-scale parts". *IFAC*. 48(3), 2214 - 2219.
- H. Zhang, D. Tang, T. Huang, C. Xu (2016). "An agent based intelligent distributed control paradigm for manufacturing systems". *IFAC*. 49(12), 1549 --1554.
- J.C. Chaplin, O.J. Bakker, L. de Silva, D. Sanderson, E. Kelly, B. Logan, S.M. Ratchev (2015). "Evolvable Assembly Systems: A distributed architecture for intelligent manufacturing". *IFAC*. 48(3), 2065 - 2070.
- I. Mahdavi, B. Shirazi, N. Ghorbani, N. Sahebjamnia (2013). "IMAQSC: design and implementation of an intelligent multi-agent system for monitoring and controlling quality of cement production processes". *Computers in industry*. 64, 290 - 298.
- K. Li, J.Y.-T. Leung, B.-Y. Cheng (2014). "An agent-based intelligent algorithm for uniform machine scheduling to minimize total completion time". *Applied soft computing*. 25, 277 – 284
- X.J. Shen, L. Liu, Z.J. Zha, P.Y. Gu, Z.Q. Jiang, J.M. Chen, J. Panneerselvam (2014). "Achieving dynamic load balancing through mobile agents in small world P2P networks". *Computer networks*. 75, 134 - 148.
- N. Ruiz, A. Giret, V. Botti, V. Fera (2014). "An intelligent simulation environment for manufacturing systems". *Computers & Industrial Engineering*. 76, 148 - 168.
- N. Sahebjamnia, R. Tavakkoli-Moghaddam, N. Ghorbani (2016). "Designing a fuzzy Q-learning multi-agent quality control system for a continuous chemical production line - A case study". *Computers & Industrial Engineering*. 93, 215 - 226.
- Y. Zhang, C. Qian, J. Lv, Y. Liu (2016). "Agent and cyber-physical system based self-organizing and self-adaptive intelligent shopfloor". *IEEE transactions on industrial informatics*. (99), 1 - 10. DOI:[10.1109/TII.2016.2618892](https://doi.org/10.1109/TII.2016.2618892)
- F. Pantke, O. Herzog (2014). "Distributed key figure optimization approaches for global coordination in multi-agent systems for production control". *Procedia CIRP*. 19, 180 - 185.

V. Albino, L. Fraccascia, I. Giannoccaro (2015). "Exploring the role of contracts to support of self-organized industrial symbiosis networks: an agent-based simulation study". Journal of cleaner production. (\*), 1 - 14.

I. Kucukkoc, D. Zhang (2014). "Mathematical model and agent based solution approach for the simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines". Int. J. production economics. 158, 314 - 333.

S. Ilie, C. Badica (2013). "Multi-agent approach to distributed ant colony optimization". Science of computer programming. 78, 762 - 774.

## ANEXOS

Anexo 1: Resultado consolidado pruebas de bondad de ajuste Kolmogorov-Smirnov, Anderson-Darling y Ji-cuadrado para los tiempos entre llegadas de solicitudes al proceso caso de estudio

Anexo 2: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “nivel de utilización del servidor”, etapa “Radicación”.

Anexo 3: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “longitud promedio de la cola”, etapa “Radicación”.

Anexo 4: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “tiempo de flujo”, etapa “Radicación”.

Anexo 5: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “nivel de utilización del servidor”, etapa “Pertinencia”.

Anexo 6: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “longitud promedio de la cola”, etapa “Pertinencia”.

Anexo 7: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “tiempo de flujo”, etapa “Pertinencia”.

Anexo 8: Valores calculados criterios de desviación MAD y MSD para selección de parámetros aproximados de funciones de pertenencia etapa “Radicación”.

Anexo 9: Valores calculados criterios de desviación MAD y MSD para selección de parámetros aproximados de funciones de pertenencia etapa “Pertinencia”.

Anexo 10: Modelo de simulación discreta del sistema multi-agente propuesto

Anexo 11: Superficie de respuesta sistema de inferencia difuso etapa “Radicacion”.

Anexo 12: Superficie de respuesta sistema de inferencia difuso etapa “Pertinencia”.

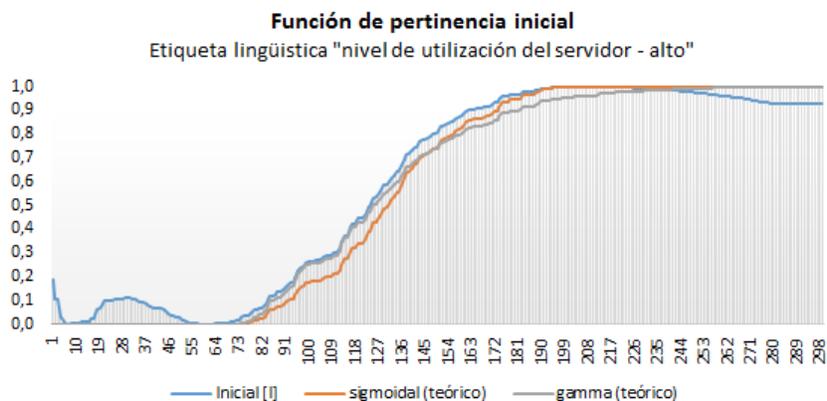
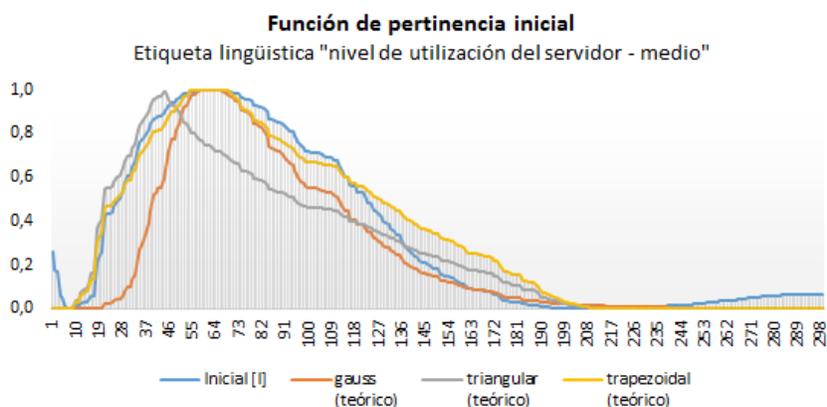
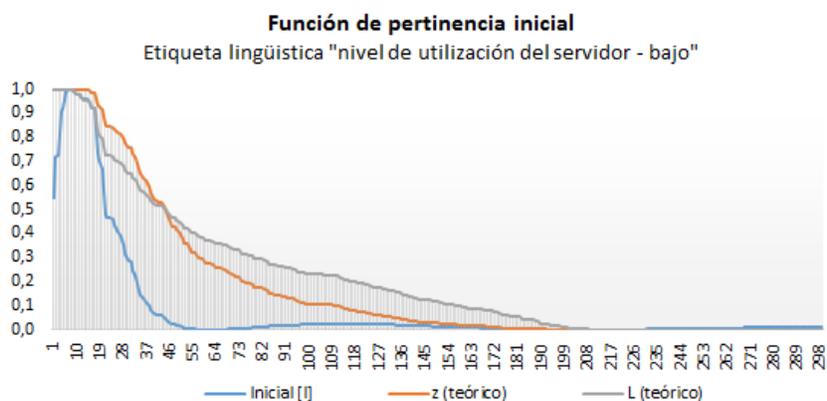
Anexo 13: Código de la función de selección del valor mínimo inferido de tiempo de flujo para los servidores de la etapa de pertinencia

Anexo 14: Simulación de la secuencia de asignación de transacciones realizada por el algoritmo propuesto a los recursos de las etapas de radicación y pertinencia a un tiempo entre llegadas de parámetro  $\lambda = 0,8078$

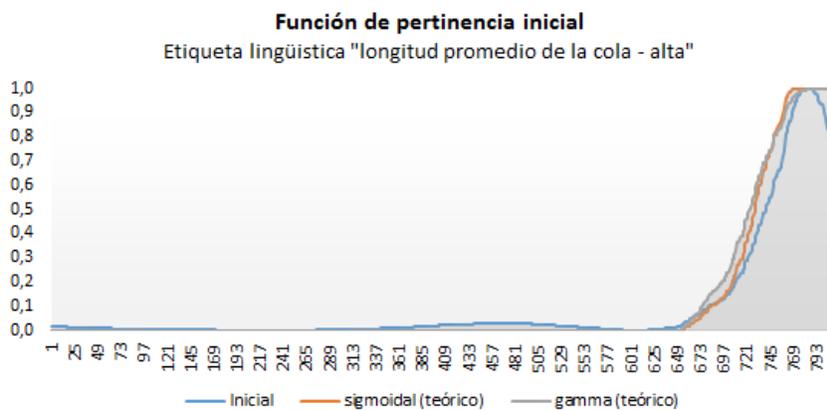
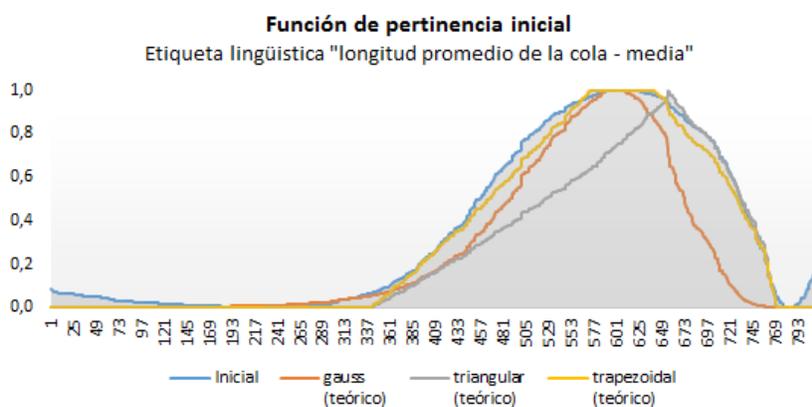
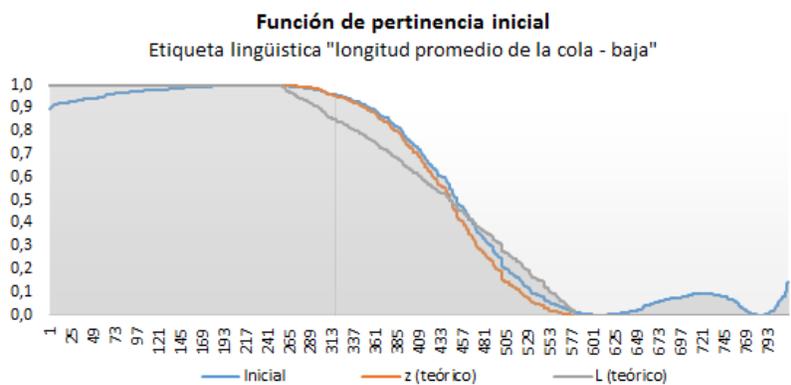
**Anexo 1: Resultado consolidado pruebas de bondad de ajuste Kolmogorov-Smirnov, Anderson-Darling y Ji-cuadrado para los tiempos entre llegadas de solicitudes al proceso caso de estudio**

Franja/prueba/parámetro		6	7	8	9	10	11	12	13	14	15	16	17	18	
Conteo		11	40	64	75	73	68	67	62	59	66	63	63	18	
$\lambda$		-	e(0,8205)	e(1,1217)	e(1,2331)	e(1,2486)	e(1,1359)	e(1,118)	e(1,029)	e(1,0012)	e(1,0977)	e(1,0748)	e(1,0659)	e(0,3437)	
p - value (no aplica para A-D)	K-S	-	0,9835	0,6662	0,4755	0,6659	0,8426	0,6284	0,9901	0,9927	0,7169	0,8175	0,6121	0,5924	
	A-D	Estadístico	-	2,0261	0,8261	0,7385	2,3592	0,5220	0,4244	0,1846	1,9995	0,4218	2,2906	0,6232	0,5895
		Valor crítico	-	2,5018	2,5018	2,5018	2,5018	2,5018	2,5018	2,5018	2,5018	2,5018	2,5018	2,5018	2,5018
	Chi-Square	-	0,9428	0,4340	0,3179	0,7592	0,4389	0,5799	0,9835	0,8700	0,9101	0,6680	0,8925	0,6031	

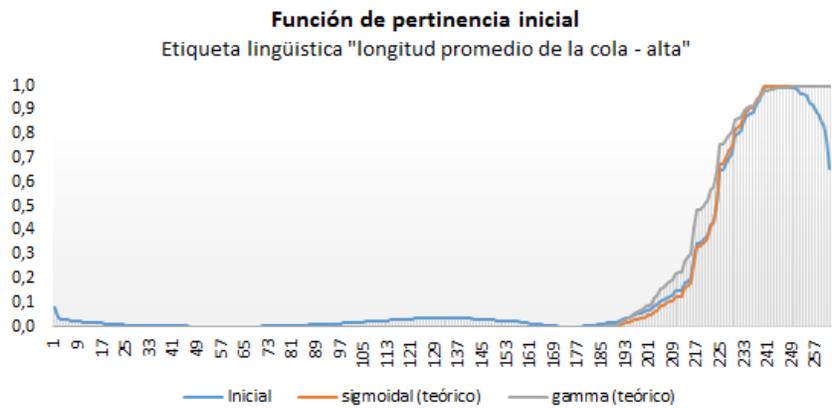
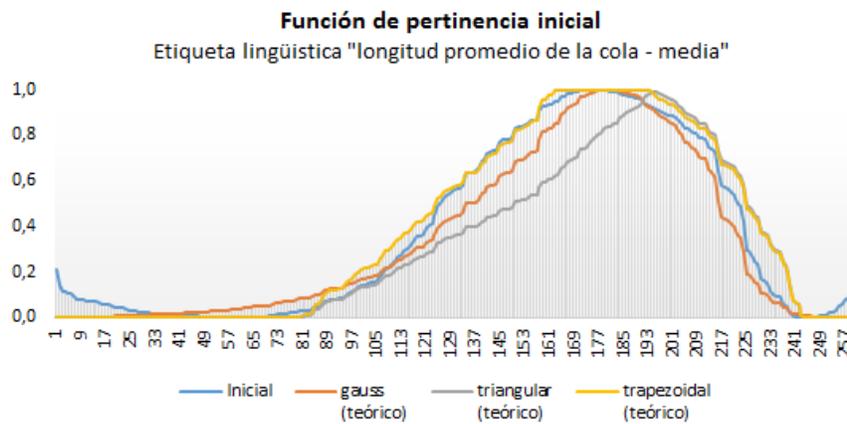
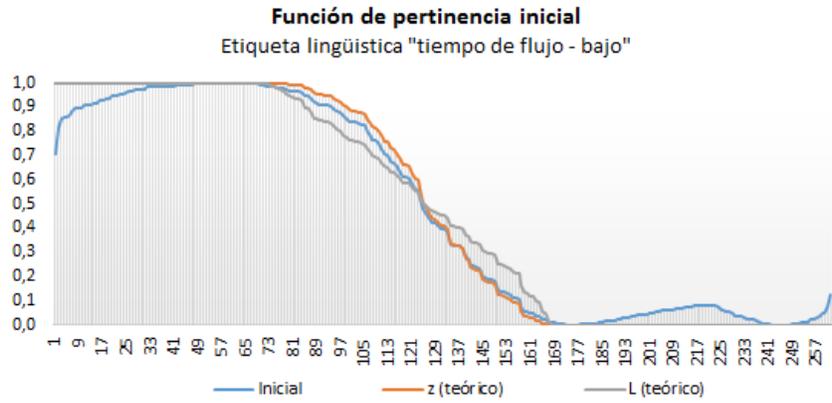
**Anexo 2: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “nivel de utilización del servidor”, etapa “Radicación”**



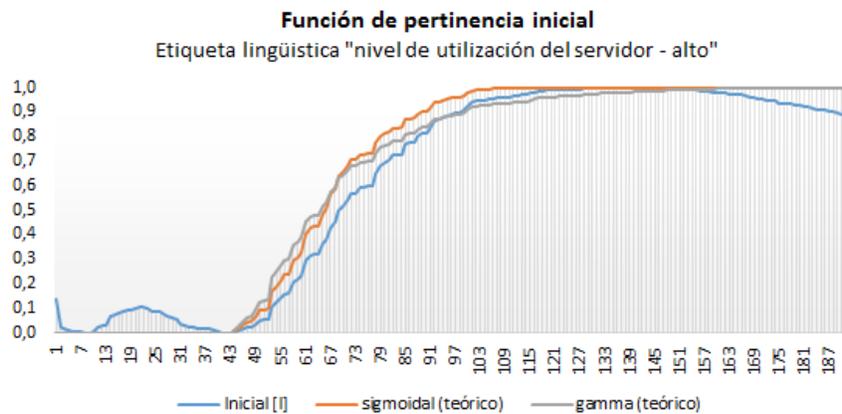
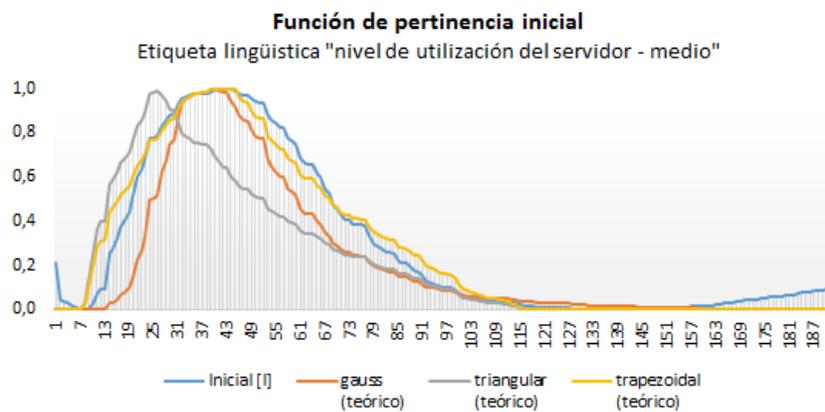
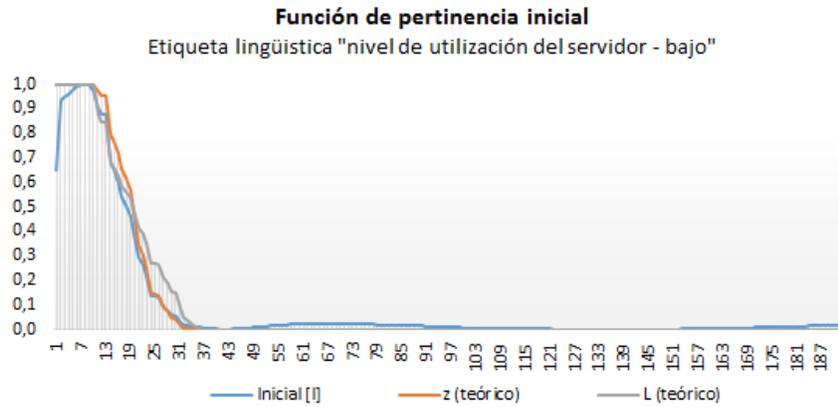
**Anexo 3: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “longitud promedio de la cola”, etapa “Radicación”**



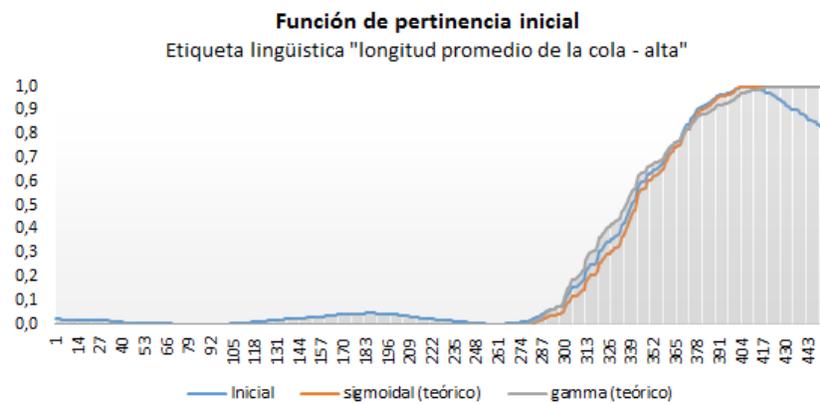
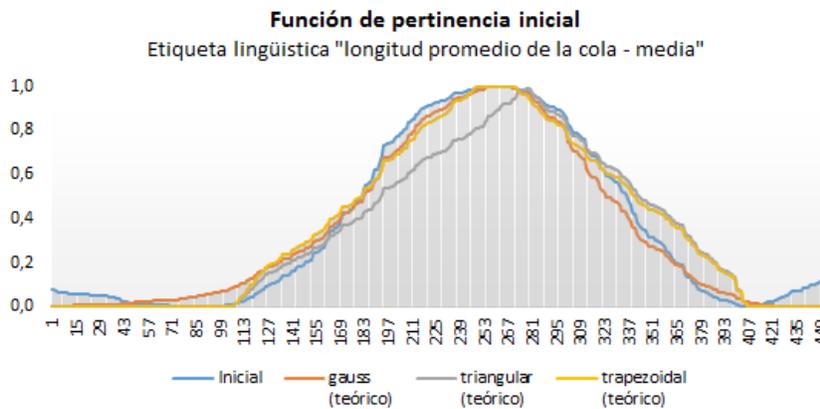
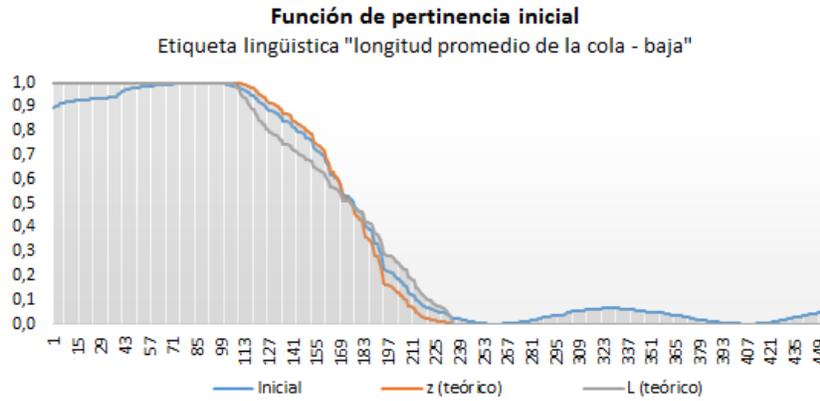
**Anexo 4: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “tiempo de flujo”, etapa “Radicación”**



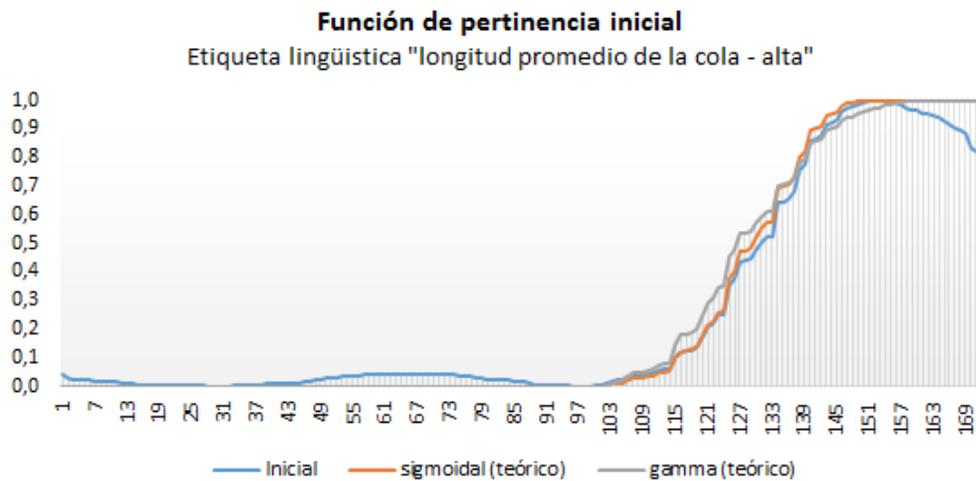
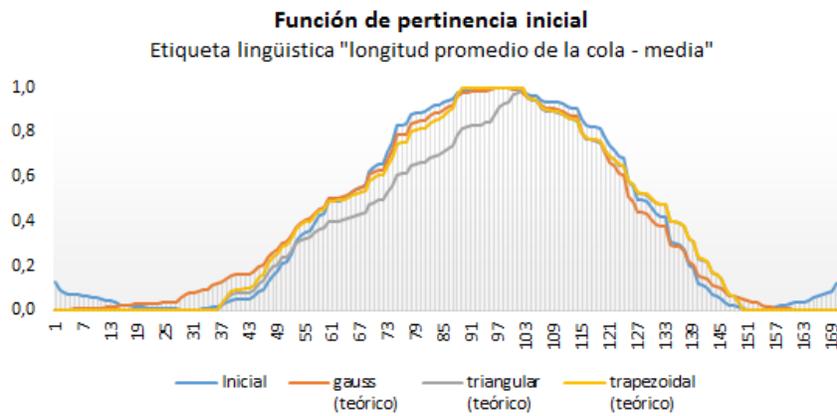
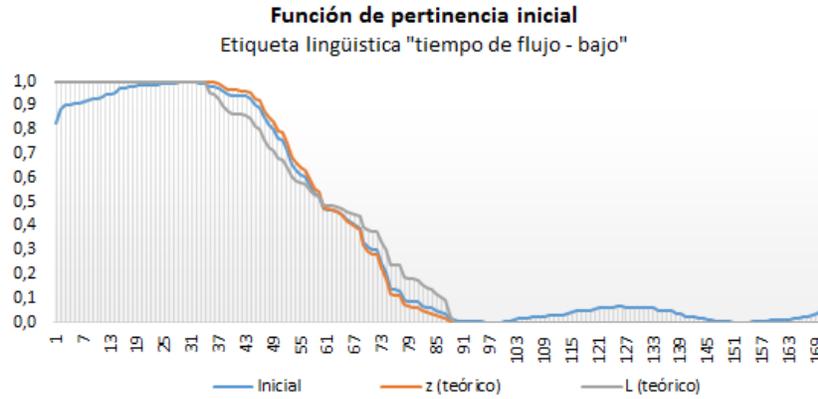
**Anexo 5: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “nivel de utilización del servidor”, etapa “Pertinencia”**



**Anexo 6: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “longitud promedio de la cola”, etapa “Pertinencia”**



**Anexo 7: Funciones de pertenencia obtenidas con la aplicación del algoritmo Fuzzy c-means para la variable “tiempo de flujo”, etapa “Pertinencia”**



**Anexo 8: Valores calculados criterios de desviación MAD y MSD para selección de parámetros aproximados de funciones de pertenencia etapa "Radicación".**

Variable: "nivel de utilización del servidor"

	Bajo		Medio			Alto		
	a	a	k	a	0,58140	a	a	a
Parámetros aproximados	0,58140	0,58140	220	b	0,80029	0,58140	0,82836	0,82836
	b	b		c	0,82279	b	b	k
	0,95026	0,95026		d	0,95026	0,95026	0,95026	220,00000
Criterio	z (teórico)	L (teórico)	gauss (teórico)	trapezoidal (teórico)	triangular (teórico)	sigmoidal (teórico)	gamma (teórico)	
MAD	28,73034	40,06793	26,09505	17,51761	30,36045	13,80578	12,37177	
MSD	9,10221	11,68732	6,44528	1,81800	6,01224	1,01786	0,78921	

Variable: "longitud promedio de la cola"

	Bajo		Medio			Alto		
	a	a	k	a	0,77432	a	a	a
Parámetros aproximados	0,56952	0,56952	3,5	b	1,56333	0,77432	1,929913	1,929913
	b	b		c	1,88578	b	b	k
	1,59889	1,59889		d	3,24939	3,24939	3,227970	2,00000
Criterio	z (teórico)	L (teórico)	gauss (teórico)	trapezoidal (teórico)	triangular (teórico)	sigmoidal (teórico)	gamma (teórico)	
MAD	26,41815	41,34021	81,64052	25,90958	77,24056	19,43478	22,78255	
MSD	1,52511	3,49986	23,11837	1,55575	18,08343	2,22824	2,87356	

Variable: "tiempo de flujo"

	Bajo		Medio			Alto		
	a	a	k	a	4,97134	a	a	a
Parámetros aproximados	4,87812	4,87812	0,7	b	6,38583	4,97134	6,97884	6,97884
	b	b		c	7,18716	b	b	k
	6,54921	6,54921		d	9,51138	9,51138	9,36655	0,70000
Criterio	z (teórico)	L (teórico)	gauss (teórico)	trapezoidal (teórico)	triangular (teórico)	sigmoidal (teórico)	gamma (teórico)	
MAD	8,74072	11,82981	15,14736	12,10518	26,74190	5,12070	7,28917	
MSD	0,60784	0,95285	1,46957	1,20431	5,54639	0,35214	0,64153	

**Anexo 9: Valores calculados criterios de desviación MAD y MSD para selección de parámetros aproximados de funciones de pertenencia etapa "Pertinencia".**

Variable: "nivel de utilización del servidor"

	Bajo		Medio			Alto		
	a	a	k	a	0,38182	a	a	a
Parámetros aproximados	0,43867	0,43867	120	b	0,68240	0,38182	0,70202	0,70202
	b	b		c	0,71691	b	b	k
	0,67734	0,67734		d	0,85377	0,85377	0,85377	130,00000
Criterio	z (teórico)	L (teórico)	gauss (teórico)	trapezoidal (teórico)	triangular (teórico)	sigmoidal (teórico)	gamma (teórico)	
MAD	3,18366	3,69202	15,03134	7,94631	22,06003	9,92963	9,59797	
MSD	0,26145	0,33242	2,77198	0,72237	6,04827	0,87334	0,85560	

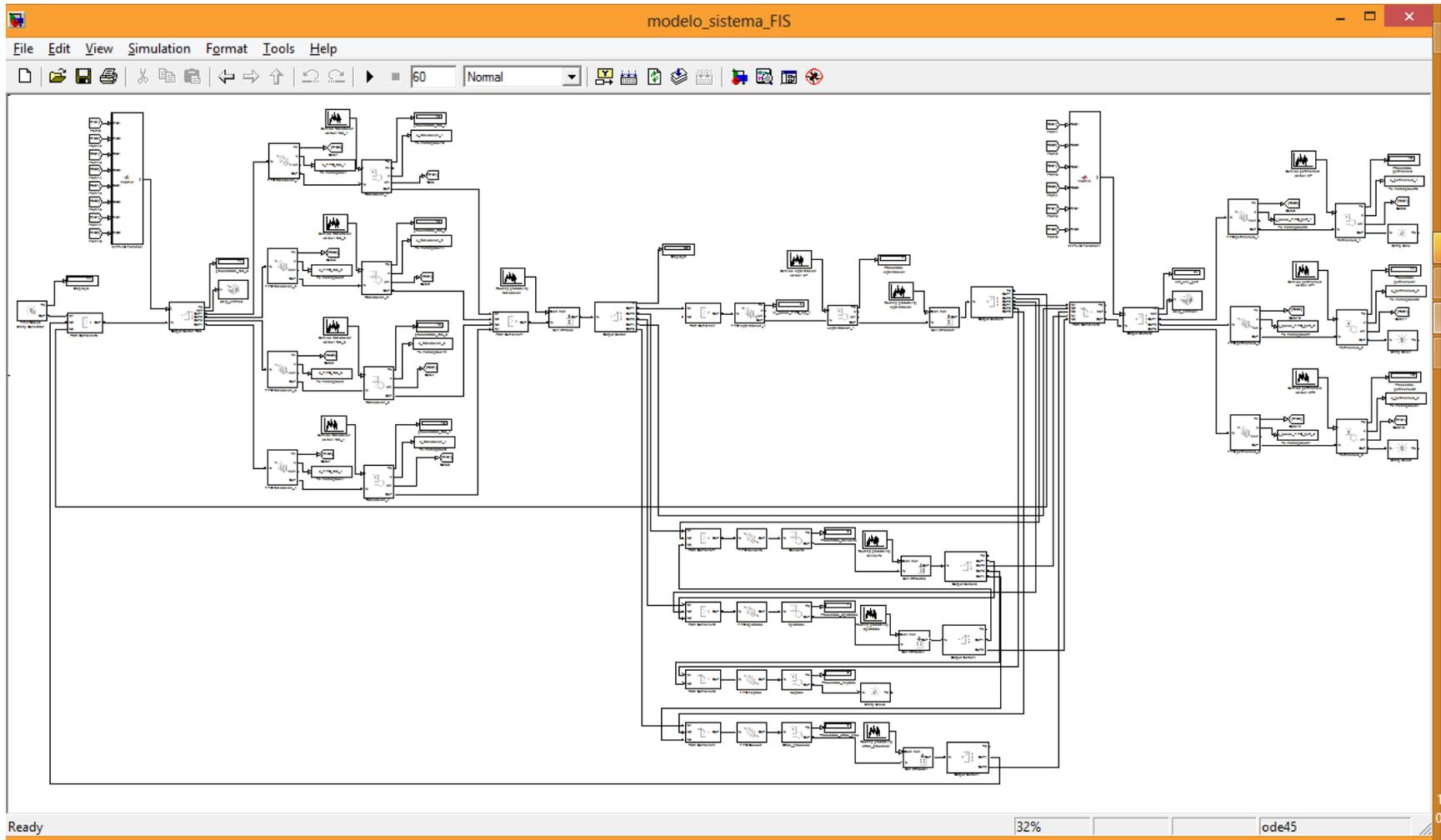
Variable: "longitud promedio de la cola"

	Bajo		Medio			Alto		
	a	a	k	a	0,47158	a	a	a
Parámetros aproximados	0,47086	0,47086	3,5	b	1,23733	0,47158	1,39897	1,39897
	b	b		c	1,36582	b	b	k
	1,18359	1,18359		d	2,35847	2,35847	2,33816	4,00000
Criterio	z (teórico)	L (teórico)	gauss (teórico)	trapezoidal (teórico)	triangular (teórico)	sigmoidal (teórico)	gamma (teórico)	
MAD	14,08348	16,56226	20,64675	25,75748	36,36168	12,09814	12,76550	
MSD	0,68005	0,97683	1,34237	2,42717	5,35180	0,80004	0,85542	

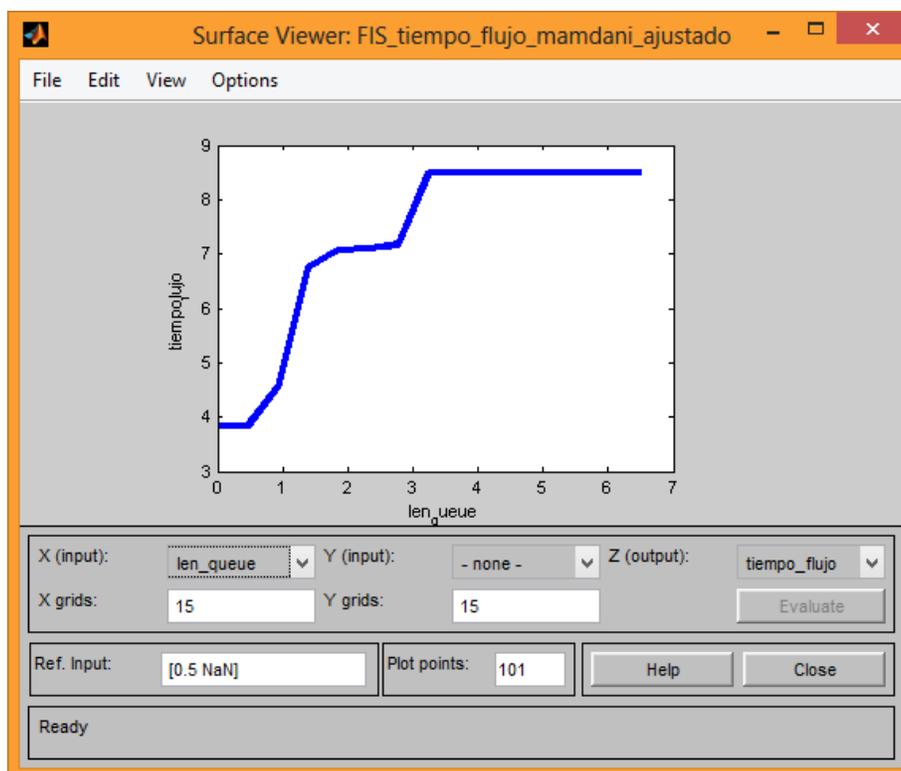
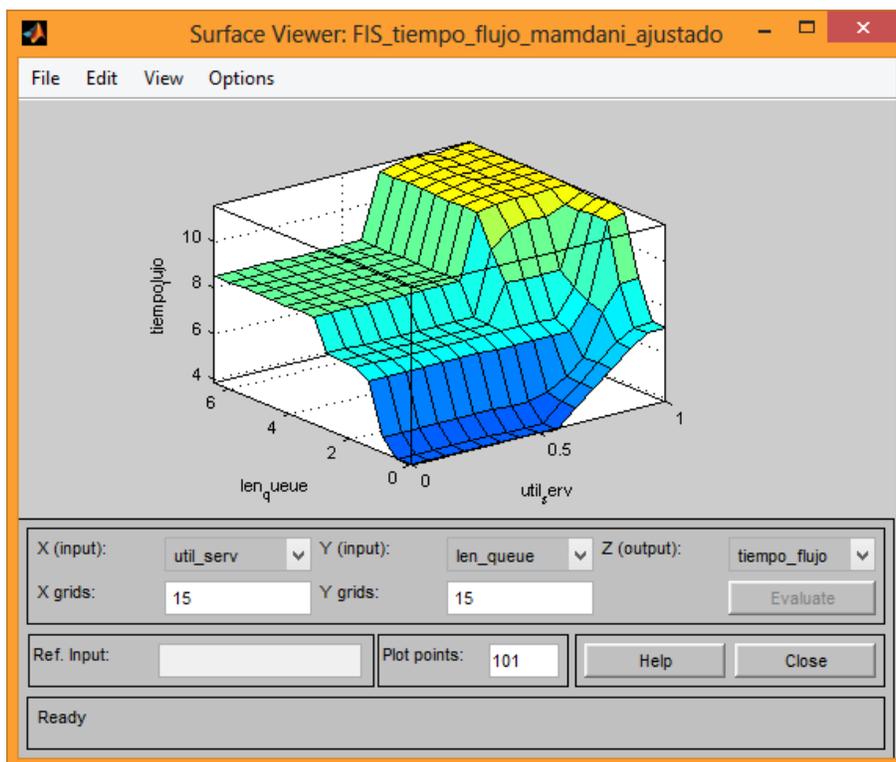
Variable: "tiempo de flujo"

	Bajo		Medio			Alto		
	a	a	k	a	6,22026	a	a	a
Parámetros aproximados	6,09284	6,09284	0,25	b	8,84431	6,22026	9,37766	9,37766
	b	b		c	9,45516	b	b	k
	8,84431	8,84431		d	12,66685	12,66685	12,66685	0,30000
Criterio	z (teórico)	L (teórico)	gauss (teórico)	trapezoidal (teórico)	triangular (teórico)	sigmoidal (teórico)	gamma (teórico)	
MAD	4,68895	7,01146	7,53080	7,23090	11,88675	4,08618	5,30502	
MSD	0,24316	0,47620	0,50729	0,48536	1,51539	0,21678	0,33721	

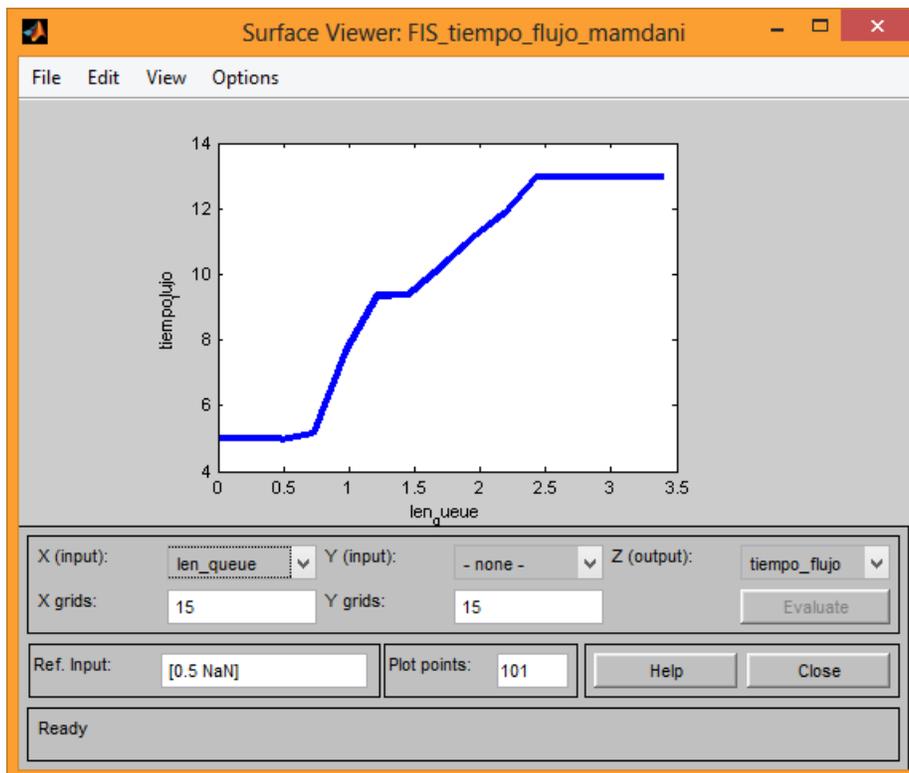
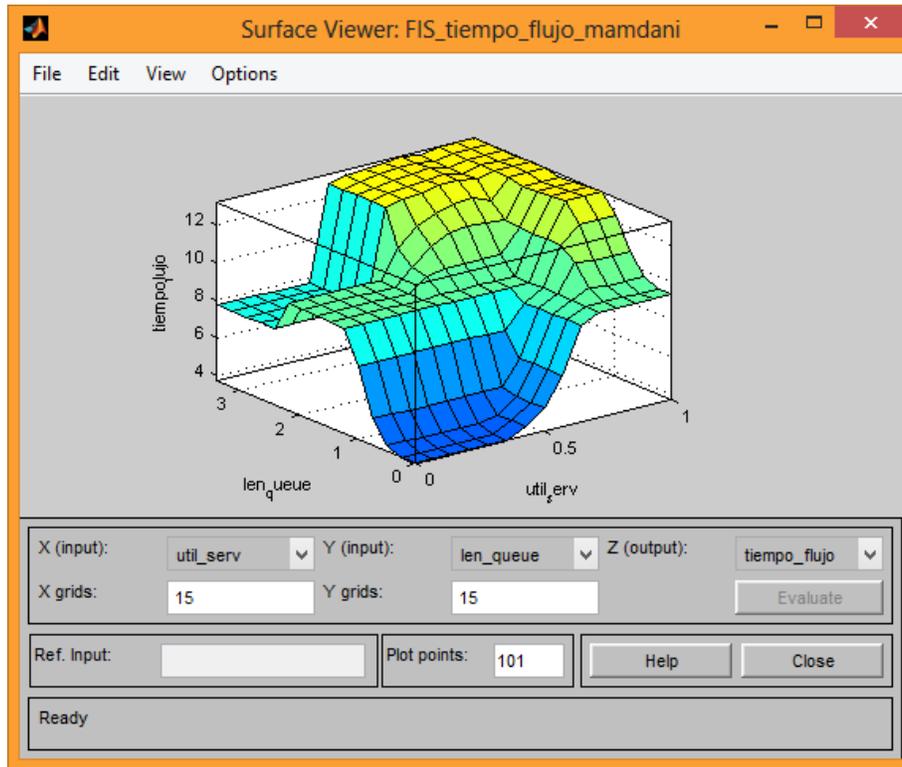
## Anexo 10: Modelo de simulación discreta del sistema multi-agente propuesto



**Anexo 11: Superficie de respuesta sistema de inferencia difuso etapa “Radicion”.**



**Anexo 12: Superficie de respuesta sistema de inferencia difuso etapa "Pertinencia".**



### Anexo 13: Código de la función de selección del valor mínimo inferido de tiempo de flujo para los servidores de la etapa de pertinencia

```
function p = findmin(R5S1, R5S2, R6S1, R6S2, R7S1, R7S2)

%#codegen

n1=0;

n2=0;

n3=0;

coder.extrinsic('readfis','evalfis');

a=readfis('nombre_sistema_inferencia');

n1=evalfis([R5S1 R5S2], a);

n2=evalfis([R6S1 R6S2], a);

n3=evalfis([R7S1 R7S2], a);

if(n1<=n2 && n1<=n3)

    p=1;

elseif(n2<=n3)

    p=2;

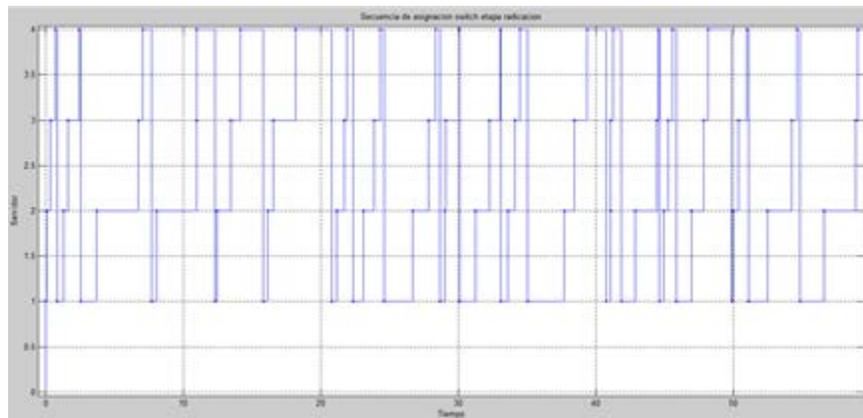
else

    p=3;

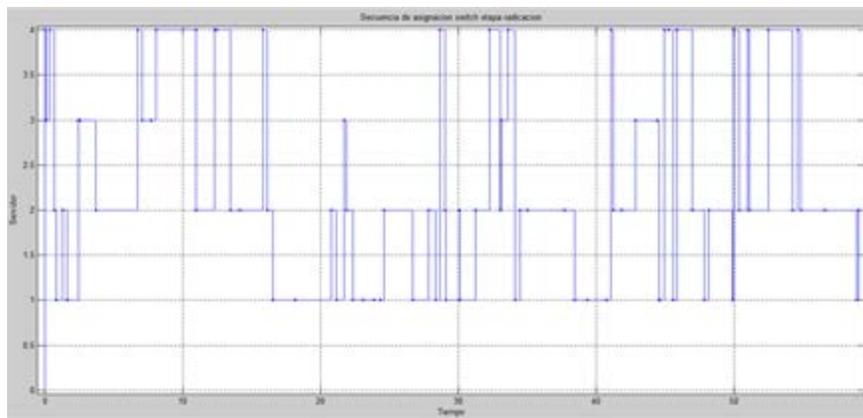
end
```

**Anexo 14: Simulación de la secuencia de asignación de transacciones realizada por el algoritmo propuesto a los recursos de las etapas de radicación y pertinencia a un tiempo entre llegadas de parámetro  $\lambda = 0,8078$**

**Secuencia de enrutamiento algoritmo Round-Robin**



**Secuencia de enrutamiento algoritmo equiprobable**



**Secuencia de enrutamiento sistema multi-agente**

