



UNIVERSIDAD AUTÓNOMA DEL ESTADO  
DE HIDALGO

---

INSTITUTO DE CIENCIAS BÁSICAS E  
INGENIERÍA

MODELADO DE TASAS DE PRODUCCIÓN  
PARA LÍNEAS DE MANUFACTURA EN  
SERIE UTILIZANDO LA METODOLOGÍA  
DE SUPERFICIE DE RESPUESTA Y REDES  
NEURONALES ARTIFICIALES.

TESIS

QUE PARA OBTENER EL TÍTULO DE:  
**DOCTOR EN CIENCIAS EN  
INGENIERÍA INDUSTRIAL**

PRESENTA:

**Federico Núñez Piña.**

DIRECTOR DE TESIS:

Dr. Joselito Medina Marín

CODIRECTOR DE TESIS

Dr. Juan Carlos Seck Tuoh Mora





UNIVERSIDAD AUTÓNOMA DEL ESTADO DE HIDALGO  
 Instituto de Ciencias Básicas e Ingeniería

Mineral de la Reforma, Hgo., a 31 de octubre de 2018

**Número de control:** ICBI-D/851/2018  
**Asunto:** Autorización de impresión de tesis.

**M. EN C. JULIO CÉSAR LEINES MEDÉCIGO**  
**DIRECTOR DE ADMINISTRACIÓN ESCOLAR DE LA UAEH**

Por este conducto le comunico que el comité revisor asignado al C. Federico Núñez Piña, alumno del Doctorado en Ciencias en Ingeniería Industrial con número de cuenta 320378, autoriza la impresión del proyecto de tesis titulado "Modelado de tasas de producción para líneas de manufactura en serie utilizando la metodología de superficie de respuesta y redes neuronales artificiales" en virtud de que se han efectuado las revisiones y correcciones pertinentes.

A continuación se registran las firmas de conformidad de los integrantes del comité revisor.

PRESIDENTE	Dr. Norberto Hernández Romero
SECRETARIO	Dra. Eva Selene Hernández Gress
VOCAL	Dr. Joselito Medina Marín
SUPLENTE	Dr. Juan Carlos Seck Tuoh Mora

Sin otro particular reitero a Usted la seguridad de mi atenta consideración.

Atentamente  
 "Amor, Orden y Progreso"

Dr. Óscar Rodolfo Suárez Cas...  
 Director del ICBI

ORSC/MABC

UAEH  
 BIBLIOTECA



Ciudad del Conocimiento  
 Carretera Pachuca-Tulancingo km 4.5 Colonia  
 Carboneras, Mineral de la Reforma, Hidalgo.  
 México. C.P. 42184  
 Teléfono: +52 (771) 71 720 00 ext. 2231  
 Fax 2109  
 direccion\_icbi@uaeh.edu.mx

[www.uaeh.edu.mx](http://www.uaeh.edu.mx)

---

# Agradecimientos

Este trabajo no habría sido posible sin el apoyo y el estímulo de mi asesor, Doctor Joselito Medina Marín, bajo cuya supervisión escogí este tema y comencé la tesis. Al Doctor Juan Carlos Seck Tuoh Mora, mi consejero en las etapas finales del trabajo, y que también, ha sido generosamente servicial, y me ha ayudado de numerosos modos.

No puedo terminar sin agradecer a mi familia, en cuyo estímulo constante y amor he confiado a lo largo de mis años en la Universidad. Estoy agradecido también a los ejemplos de mis hermanos. Su coraje resuelto y convicción siempre me inspirarán. A mi compañera de vida Lucia y a mi hija Ximena. Espero seguir, con su apoyo y una gran gratitud, por brindarme un propósito. Es a ellos a quien dedico este trabajo.

---

# Resumen

El problema de asignación de búfers, es un problema de optimización, clasificado como un NP-hard combinatorio y su investigación es de gran importancia para el diseño de sistemas de manufactura. Este problema radica en asignar un tamaño óptimo para los búfers o almacenes intermedios entre cada máquina siendo de particular interés en los criterios monetarios, expresados en una forma de costos-beneficio, ya que si aumenta el tamaño de búfer es directamente proporcional al espacio ocupado en el área de trabajo, significando un costo, pero si es demasiado pequeño la estación de trabajo o máquina se quedaría sin abastecimiento generando paros en la producción, de igual manera generando costos e incumplimientos. Por otra parte, el cálculo del tamaño óptimo de búfer requiere de un gran tiempo computacional, afectando principalmente a los diseñadores de líneas de producción, particularmente en productos altamente cambiantes ya sea por sus ventas por temporadas, avance tecnológico, modas, etc.

El objetivo de este estudio fue el desarrollar una herramienta, en la cual los diseñadores de líneas de producción estarían interesados por sus rápidas soluciones al problema de asignación de búfers. especialmente cuando tengan que tomar decisiones a nivel operativo. Más específicamente se estructuró una red neuronal artificial óptima con un diseño de 1 a 4 capas con una combinación de 5, 8, 10, 12 y 15 neuronas para cada capa, generando 780 redes, para ello se utilizó el toolbox de Matlab NNtool, empleando el algoritmo de Levenberg-Marquardt, con una función de transferencia sigmoide, tomando 2 indicadores para medir el rendimiento de las redes neuronales artificiales, el error cuadrático medio (MSE) y la correlación (R).

Los resultados obtenidos para las líneas de producción exponenciales con una media  $\mu = 1$  y un número de estaciones  $\mathbf{K}=3$  a 20,  $\mathbf{N}=1$  a 20, obteniendo un error relativo de 0.34%, un error cuadrático medio de 1.150E-07 y una correlación de  $R=1$ . Se empleó la metodología de superficie de respuesta como una segunda herramienta que se acopla a los datos de entrada y salida, esta se desarrolló en Matlab, en donde se toma como el conjunto de máquinas  $k$  en el eje de las abscisas, en el eje de las ordenas el número de búfers  $N$  y en la tasa de producción de las líneas representa la superficie de la gráfica, de esta forma se generó un polinomio de

respuesta, el cual aproximó los valores con las siguientes métricas: error relativo 2.36 %, el error cuadrático medio de 1.22796 E-05 y una correlación de 0.999303176. Motivó a realizar una segunda prueba con 55 experimentos fuera del rango mencionado, teniendo como resultado un MSE= 3.57E-05 y una correlación de R=0.99875 con un intervalo máximo de **K=100** y **N=101** y para los valores de la metodología de superficie de respuesta no son tan alentadores con un error cuadrático medio de 66032.8992, la correlación de -.0225360 y el error relativo máximo fue de 241581.1947% concluyendo un mejor comportamiento de la red neuronal artificial.

---

# Abstract

The problem of allocation of buffers, is an optimization problem, classified as a combinatorial NP-hard and its research is of great importance for the design of manufacturing systems. This problem lies in assigning an optimal size for the buffers or intermediate stores between each machine, being of particular interest in the monetary criteria, expressed in a cost-benefit form, since if it increases the buffer size is directly proportional to the space occupied in the work area, meaning a cost, but if it is too small, the work station or machine would be without supply, generating stoppages in the production, in the same way generating costs and non-compliance. On the other hand, the calculation of the optimal buffer size requires a large computational time, affecting mainly the designers of production lines, particularly in highly changing products either due to seasonal sales, technological advances, fashions.

The objective of this study was to develop a tool, in which the production line designers would be interested in their quick solutions to the buffers assignment problem, especially when they have to make decisions at the operational level. More specifically, an optimal artificial neural network was structured with a design of 1 to 4 layers with a combination of 5, 8, 10, 12 and 15 neurons for each layer, generating 780 networks, for which the Matlab toolbox nntool was used, using the Levenberg-Marquardt algorithm, with a sigmoid transfer function, taking 2 indicators to measure the performance of artificial neural networks, the mean square error (MSE) and the correlation (R).

The results obtained for the exponential production lines with a mean  $\mu=1$  and a number of stations  $\mathbf{K}= 3$  to 20,  $\mathbf{N}= 1$  to 20, obtaining a relative error of 0.34 %, a mean square error of 1150E-07 and a correlation of  $\mathbf{R} = 1$ . The response surface methodology was used as a second tool that is coupled to the input and output data, this was developed in Matlab, where it is taken as the set of  $K$  machines on the abscissa axis, in the axis of the orders the number of buffers  $N$  and in the production rate of the lines represents the surface of the graph, in this way a response polynomial was generated, which approximated the values with the following metrics: error relative 2.36 %, the quadratic error of 1.22796 E-05 and a correlation

---

of 0.999303176. Motivated to perform a second test with 55 experiments outside the mentioned range, resulting in an  $MSE = 3.57E-05$  and a correlation of  $R = 0.99875$  with a maximum interval of  $K= 100$  and  $N= 101$  and for the values of the answer surface methodology are not as encouraging with mean square error of 66032.8992, the correlation of -0.0225360 and the maximum relative error was 241581.1947% with a better behavior of the neural network.

# Índice general

Agradecimientos	2
Resumen	3
Abstract	5
<b>1. Introducción</b>	<b>11</b>
1.1. Descripción del problema . . . . .	11
1.2. Justificación . . . . .	13
1.3. Objetivo General . . . . .	14
1.4. Objetivos Específicos . . . . .	14
1.5. Hipótesis . . . . .	15
<b>2. Estado del Arte</b>	<b>16</b>
2.1. Modelos de líneas de producción . . . . .	16
2.2. Optimización de líneas de producción . . . . .	17
2.3. Métodos por evaluación . . . . .	19
2.4. Métodos exactos . . . . .	20
<b>3. Marco Teórico</b>	<b>22</b>
3.1. Líneas de transferencia . . . . .	22
3.1.1. Características principales y clase de Lineas de Producción. . . . .	22
3.1.2. Resultados generales relacionados con la tasa de producción de la Línea	25
3.1.3. Cuestiones de búfer . . . . .	26
3.1.4. Características de una línea de producción. . . . .	32
3.1.5. La reversibilidad y la dualidad . . . . .	34
3.2. Redes neuronales artificiales . . . . .	35
3.2.1. Funciones de activación . . . . .	36
3.2.2. Estructura y Aprendizaje de la Red Backpropagation. . . . .	37

---

3.2.3.	Algoritmo Levenberg-Marquardt . . . . .	38
3.2.4.	Fase de la validación. . . . .	39
3.3.	Metodología de superficie de respuesta. . . . .	40
3.3.1.	Elección del diseño experimental. . . . .	41
<b>4.</b>	<b>Diseño de la red neuronal artificial y superficie de respuesta para líneas de producción confiables.</b>	<b>43</b>
4.1.	Diseño de las líneas de producción experimentales. . . . .	43
4.1.1.	Diseño de los experimentos. . . . .	43
4.2.	Diseño de la red neuronal. . . . .	46
4.2.1.	Configuración de la red neuronal artificial . . . . .	49
4.2.2.	Algoritmo de entrenamiento . . . . .	51
4.2.3.	Generación de la base datos de las redes neuronales artificiales . . . . .	52
4.2.4.	Algoritmo de búsqueda de la mejor red neuronal artificial . . . . .	53
4.2.5.	Simulación de la red neuronal artificial. . . . .	54
4.3.	Metodología de superficie de respuesta. . . . .	54
4.3.1.	Ajuste de la superficie. . . . .	55
<b>5.</b>	<b>Resultados y discusiones.</b>	<b>57</b>
5.1.	Primer experimento . . . . .	57
5.1.1.	Resultados de la red neuronal . . . . .	57
5.1.2.	Resultados de la metodología de superficie de respuesta. . . . .	62
5.2.	Segundo experimento. . . . .	65
<b>6.</b>	<b>Conclusiones</b>	<b>71</b>
6.1.	Trabajos futuros . . . . .	74
<b>A.</b>	<b>Experimentos usados para la red neuronal artificial</b>	<b>76</b>
<b>B.</b>	<b>Experimentos simulados por la red neuronal artificial</b>	<b>78</b>
	<b>Bibliografía</b>	<b>80</b>

---

# Índice de figuras

1.1. Línea de producción en serie con $K$ estaciones y $K-1$ almacenes (búfer) . . . . .	11
2.1. Producción en serie . . . . .	16
2.2. Métodos de solución. . . . .	18
3.1. Neurona artificial . . . . .	36
4.1. Representación de los 360 experimentos de líneas de producción . . . . .	46
4.2. Metodología para el diseño de la Red Neuronal Artificial . . . . .	48
4.3. 780 combinaciones de redes neuronales . . . . .	48
5.1. Red neuronal artificial óptima . . . . .	58
5.2. Comparación entre los 360 experimentos de [80] contra la salida de la red neuronal artificial. . . . .	60
5.3. Regresión para los 370 experimentos. . . . .	61
5.4. Grafifac de correlación de la MSR en el primer experimento . . . . .	64
5.5. Grafica de residuales de la MSR . . . . .	64
5.6. Correlación del segundo experimento . . . . .	66
5.7. Comparación de tasa de producción . . . . .	69
6.1. Gráficas de los experimentos de tasas de producción de [1] en contraste con las tasas generadas por la red neuronal artificial . . . . .	73

---

# Índice de tablas

3.1. Datos iniciales de la línea de producción . . . . .	28
3.2. Resumen de cálculos para $f_3(\rho_i, \rho_{i+1}) = \frac{1}{(\rho_i + \rho_{i+1})}$ . . . . .	29
3.3. Resumen de cálculos para $f_1(\rho_i, \rho_{i+1}) = \frac{1}{\min(\rho_i, \rho_{i+1})}$ . . . . .	29
3.4. Resumen de cálculos para $f_2(\rho_i, \rho_{i+1}) = \frac{1}{\rho_i} + \frac{1}{\rho_{i+1}}$ . . . . .	29
3.5. Simulación de los rendimientos de las tasa de producción para los dos casos. . . . .	31
3.6. Valores de rendimiento de las tasa de producción simuladas para los dos casos . . . . .	31
3.7. Resultados internos entre los dos casos . . . . .	31
4.1. Segunda prueba con 55 líneas [80, 81] de producción. . . . .	44
4.2. Indicadores del conjunto de neuronas de [81]. . . . .	47
4.3. Arreglo de 780 combinaciones de 4 capas para la red neuronal . . . . .	49
5.1. Tiempos computacionales . . . . .	57
5.2. Referencia de tiempos del primer experimento. . . . .	58
5.3. Resultados parciales de la simulación de los 360 experimentos. . . . .	59
5.4. Continuación de resultados parciales de la simulación de los 360 experimentos. . . . .	60
5.5. Resultados parciales de la MSR de los 360 experimentos. . . . .	63
5.6. Referencia de tiempos de [80] con la red neuronal artificial. . . . .	67
5.7. Errores de la red neuronal artificial desarrollados para los 55 datos. . . . .	68
5.8. Errores de la Metodología de superficie de respuesta desarrollados para el segundo experimento . . . . .	70
6.1. Comparación de métricas entre la red neuronal artificial y la metodología de superficie de respuesta . . . . .	72
A.1. 360 experimentos de líneas de producción . . . . .	77
B.1. 360 experimentos de líneas de producción generados por la red neuronal artificial . . . . .	79

---

# Capítulo 1

## Introducción

### 1.1. Descripción del problema

El problema de asignación de almacenes temporales (BAP) de sus siglas en ingles, Buffer Allocation Problem. Es una de las cuestiones más complejas para el diseño de sistemas de fabricación, en los cuales están involucrados varios tipos de variables, como: Las líneas de transferencia automáticas, la tasa de producción y las redes de comunicación [1].

El BAP se refiere a la asignación de una cantidad determinada a un almacén temporal (búfer),  $N$ , entre las  $K - 1$  ubicaciones de este almacén intermedio dentro de una línea de producción para lograr un objetivo o producto específico. Una de las situaciones, en una línea de producción, es la asignación del espacio del almacén temporal (búfer), ya que claramente, es un recurso caro, por lo que se desea modelos ideales, esto implica consideraciones de costos. Independientemente de los demás problemas de diseño que involucra una planta.

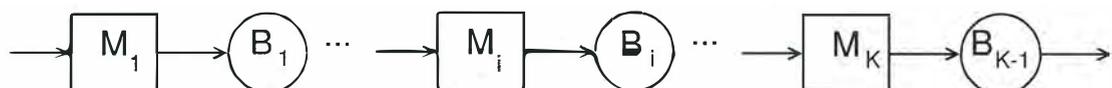


Figura 1.1: Línea de producción en serie con  $K$  estaciones y  $K-1$  almacenes (búfer)

El problema de asignación de un almacén intermedio puede expresarse principalmente en tres formas, dependiendo de la función objetivo[2]. Estas funciones objetivo pueden estar expresadas con la maximización de la tasa de rendimiento de la línea de producción, reduciendo al mínimo el total del tamaño de búfer en la línea y minimizar el inventario promedio de trabajo en proceso. Los modelos de estas formas del problema se proporcionan a continuación:

Problema BAP-1: Esta formulación del problema expresa, la maximización de la tasa de producción para una cantidad fija de búfer:

Encontrar  $B = (B_1, B_2, \dots, B_{K-1})$  de manera que

máx  $f(B)$

sujeto a

$$\sum_{i=1}^{K-1} B_i = N$$

$$B_i \geq 0$$

Donde:  $N$  es entero y no negativo fijo que denota el total del espacio del búfer disponible en el sistema, que tiene que ser repartida entre los  $K-1$  lugares de búfer para maximizar la tasa de rendimiento de la línea de producción de la máquina  $K$ . En esta formulación,  $B$  representa un vector de tamaño del búfer, y  $f(B)$  representa la tasa de rendimiento de la línea de producción como una función del tamaño del vector del búfer.

Problema BAP-2

La solución se acerca a este problema objetivo a lograr la tasa de rendimiento deseado con el tamaño de búfer total mínimo, como sigue:

$B = (B_1, B_2, \dots, B_{K-1})$

$$\text{mín} \sum_{i=1}^{K-1} B_i$$

sujeto a

$$f(B) \geq f^*$$

$$B_i \geq 0$$

Donde:  $K$  es el número de máquinas dentro de la línea,  $B$  es el tamaño del búfer,  $f^*$  es la tasa de producción deseada.

### Problema BAP-3

En este modelo de minimización, se reduce al mínimo la media del trabajo en proceso (WIP) en el inventario sujeto al tamaño de búfer total y la tasa de rendimiento deseada, como restricción:

Encontrar  $B = (B_1, B_2, \dots, B_{K-1})$  de manera que:

mín  $Q(B)$

sujeto a.

$$\sum_{i=1}^{K-1} B_i \leq N$$

$$f(B) \geq f^*$$

Donde  $Q(B)$  indica el promedio de inventario del trabajo en proceso, como una función del tamaño del vector del búfer y  $f^*$  es la tasa deseada del rendimiento.

## 1.2. Justificación

El estudio de las líneas de producción son empleadas para altos volúmenes de producción, particularmente en el sector automotriz. Las inversiones en este sector van de cientos de miles a millones de dólares[3]. México en el 2014 se convirtió en el séptimo productor de vehículos a nivel internacional y el primer lugar en América Latina, la producción y exportaciones de vehículos ligeros y pesados establecieron un nuevo récord histórico en el país. Es importante destacar el crecimiento que México ha tenido como productor de vehículos a nivel internacional, superando a Francia y España[4].

El desarrollo de nuevas herramientas para un rápido diseño de las líneas de producción, es de una alta importancia para productores de ciclos de vida cortos, desde un punto de productividad y ventaja competitiva dentro del mercado tan cambiante.

Una de las razones principales para tener un óptimo almacén temporal es que permite que las estaciones de trabajo, que están ordenadas una después de la otra, operen de manera independiente una de la otra. Esto reduce el tiempo de inactividad, por las diferentes variables

como: fallo de máquina, que se pare por falta de material o que no se tenga espacio para el producto. O por lo contrario si se tiene un exceso en el tamaño de los almacenes, se tendría un costo incurrido. Debido a esto la importancia de encontrar buenas u óptimas configuraciones de almacenamiento temporal(búfer), este es un importante problema de optimización que enfrenta la fabricación y los diseñadores de sistemas.

### 1.3. Objetivo General

Modelar líneas de producción en base al número de máquinas  $K$  y tamaños de búfers  $N$  por medio de un arreglo combinatorio para obtener la mejor red neuronal artificial comparada con la función de respuesta de la metodología de superficie de respuesta y desarrollar una solución al problema BAP-1.

### 1.4. Objetivos Específicos

- Definir el experimento numérico de las líneas de producción en base al número de máquinas  $K$  y los tamaños de búfers  $N$  basados en la literatura para la validación de las tasas de producción.
- Configurar los parámetros para la metodología de superficie de respuesta y la validación de los datos obtenidos contra las tasas de producción reales.
- Configurar los parámetros de la red neuronal con una estructura combinatoria de 5, 8, 10, 12 y 15 neuronas, para obtener una base de datos y validar las salida contra las tasas de producción real.
- Simular la red neuronal artificial para generar los experimentos y analizar sus comportamientos validando las estimaciones por medio del error cuadrático medio y la correlación entre los valores validados y la red neuronal.
- Simular los experimentos con la metodología de superficie de respuesta para generar la función y analizar su comportamiento validando las estimaciones por medio del error cuadrático medio y la correlación entre los valores validados y la red neuronal.

- Analizar y comparar los resultados de los dos métodos teniendo como eje el diseño numérico validado por medio del error cuadrático y la corrección entre valores validados y estimados.
- Diseñar un segundo experimento para demostrar la eficiencia de los métodos por medio de valores de máquinas  $K$  y búfers  $N$  fuera del grupo de entrenamiento de la red neuronal artificial y de la función obtenida de la superficie de respuesta.
- Analizar las tasas de producción arrojadas por los dos métodos, teniendo como eje los datos validados por la literatura por medio del error cuadrático y la correlación entre los valores validados y estimados.

## 1.5. Hipótesis

El diseño experimental de una comparación entre una red neuronal artificial y la metodología de superficie de respuesta generará una red neuronal artificial y una función a partir de la superficie de respuesta, auxiliará a la toma de decisiones en el diseño de líneas de producción fiables y exponenciales con una media de  $\mu = 1$ , prediciendo la mejor tasa de producción concerniente al problema de asignación de búfers (BAP-1) validando los parámetros de la red neuronal artificial y la función de respuesta, por medio del error cuadrático medio y la correlación entre las entradas contra las salidas simuladas.

---

# Capítulo 2

## Estado del Arte

### 2.1. Modelos de líneas de producción

Un sistema de manufactura, es un conjunto de máquinas, elementos de transporte, computadoras, inventarios y otros equipos e instalaciones, que son empleados para la elaboración de productos[5]. Un sistema de producción en línea o línea de transferencia, esta organizada con máquinas conectadas en serie y separadas por bufer (almacén intermedio) como se muestra en la Figura 2.1, se muestra cinco máquinas y cuatro búfers, donde los cuadros representan las máquinas  $M_i$  y los círculos los búfers  $B_i$ ,  $N_i$  representa el tamaño de búfer  $B_i$  y  $\bar{n}_i$  es el promedio del inventario del búfer  $B_i$ .

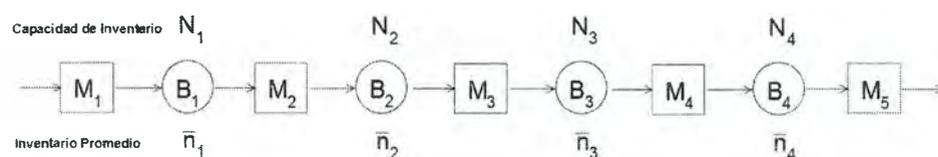


Figura 2.1: Producción en serie

La importancia económica de una línea de producción en serie, es muy importante ya que son usadas para la manufactura de altos volúmenes de productos entre ellos uno muy importante. La industria Automotriz.

Los capitales que se invierten van desde cientos miles a millones de dolares. En síntesis. Las líneas de producción representan un importante fenómeno dentro de la economía global y local.

El problema de asignación de búfer ha sido estudiado durante 50 anos y números artícu-

los se han publicado. El primer estudio, con respecto a este tema fue hecho por Koenigsberg en 1959[6], el cual da un análisis y revisión de los problemas básicos, asociados con un funcionamiento eficaz de los sistemas de producción.

Un análisis más detallado sobre los modelos matemáticos que describa el efecto de los almacenes temporales intermedios[7, 8] y algunos estudios más completos sobre este estudio lo pueden encontrar en[9, 10]. La clasificación del estudio del arte, se abordan los temas en optimización de líneas de producción, los métodos por simulación, métodos exactos y métodos aproximados.

Las soluciones aproximadas para resolver el BAP implica los métodos Generativos y de Evaluación por medio de iteraciones, estos métodos se combinan en una configuración cerrada como se observa en la Figura 2.2, en esta configuración un método de evaluación es usado para obtener el valor de la función objetivo, para un conjunto de entradas y para una búsqueda de la solución óptima, el valor de la función objetivo se comunica al método generativo.

El método de evaluación provee la predicción de distintas medidas de rendimiento como la tasa de producción y la longitud media de las líneas de espera, esto basado en métodos analíticos y de simulación. Los métodos analíticos pueden ser clasificados como exactos y aproximados. debido a que los métodos analíticos pueden ser aplicados solo para problemas con líneas clasificadas como pequeñas y los métodos aproximados usualmente emplean la evaluación para resolver el BAP, También existen varias técnicas de optimización, la enumeración completa es un simple método, sin embargo solo aplica para líneas relativamente pequeñas ya que el total de números de soluciones factibles crece exponencialmente, cuando el número total de máquinas y las capacidades del inventario intermedio incrementa, por lo que es imposible emplear la numeración completa para problemas de líneas de producción grandes. Por lo que varios métodos de búsqueda metaheurísticos son empleados y adaptados por los investigadores para tratar con eficiencia el BAP.

## 2.2. Optimización de líneas de producción

Los métodos generativos se centran en la búsqueda del tamaño óptimo de los almacenes temporales para mejorar el rendimiento del sistema. El algoritmo más simple, es aquel que emplea un método de enumeración completa de la línea, sin embargo, es aplicable solo para pequeños sistemas y que el número total de soluciones factibles crece exponencialmente.



Figura 2.2: Métodos de solución.

Cuando el número total de máquinas y el tamaño total de los almacenes temporales que se deberán asignarse para el sistema. Por lo tanto para sistemas grandes es imposible buscar a través del todo el espacio de solución. En los últimos años varios métodos de búsqueda y metaheurísticos son ampliamente adoptados por los investigadores para resolver el problema de naturaleza combinatoria del tamaño del almacén temporal.

Park en 1993, desarrolló un algoritmo heurístico, en dos fases para resolver el problema de minimización del total de almacenes temporales[11]. Sin embargo su método no siempre puede encontrar la solución óptima y tampoco converge en todos los casos. Seong en 1994 adopta el concepto de un pseudo gradiente y la proyección de este método, fue para resolver el problema de maximización de la tasa de producción para una línea de producción[12] y un año después emplea un método de gradiente para la maximización de la tasa de producción de un problema de producción exponencial[13], otros investigadores emplearon también la gradiente como Gershwin y Goldin[14] para resolver el problema de la capacidad de almacenes temporales, minimizando la capacidad total, basándose en la observación, en que si la tasa de producción se expandía en un primer orden, este se podría formular como un problema de programación lineal entera.

Los algoritmos de búsqueda, tienden a resolver la explosión exponencial del número de vectores de solución en este caso los algoritmos de búsqueda tienden a resolver la explosión exponencial del número de vectores de solución en este caso[5] presentan un algoritmo eficiente el cual aplica un enfoque primal-dual para minimizar el total de almacenes temporales, bajo una restricción de velocidad de producción, en su propuesta, el problema primordial es reducir el tamaño total de almacenes temporales sujeta a una tasa de producción, mientras que la formulación dual es maximizar la tasa de producción de la línea de producción, sujeta a una limitación del espacio total de almacenes. Los autores también estudiaron la maximización del beneficio de la línea a través de un método no lineal, que es rápido y preciso, pero no tienen

en cuanto la limitación de la tasa de producción. Vouros y Papadopoulos se basaron en un sistema llamado ASBA2, que emplea un sistema basado en el conocimiento que determina los planos óptimos de capacidad de almacenes, con el objetivo de maximizar el rendimiento de las líneas de producción este incorpora un algoritmo exacto[15]. Nahas emplean degradado de búsqueda local heurística[16]. [17–25] Desarrollaron problemas específicos y sus algoritmos de búsqueda para la solución de capacidades de almacenes. Hay principalmente dos desventajas de los métodos de búsqueda tradicionales. Una es que, la búsqueda tradicional, a veces no pueden saltar por encima de las soluciones óptimas locales en busca de los más óptimos globales. La otra desventaja es que con estos métodos aproximados es difícil para observar los pequeños cambios en los tamaños de búfer afectarían al sistema.

Los metaheurísticos son métodos de búsqueda que utilizan estrategias que guían el proceso de búsqueda y explorar el espacio de búsqueda a fin de encontrar soluciones óptimas o casi óptimas. Los algoritmos son aproximados y por lo general no deterministas. Los métodos de solución típicos en esta área incluyen Búsqueda Tabú[26–29], Simulado recocido[30, 31], Algoritmos Genéticos[32–39], y Optimización de colonias de hormigas[40, 41]. Para buscar mejor el espacio de la solución, la tendencia reciente es para hibridar el metaheurística con otros métodos tales como particiones anidadas[42], el método Branch y Bound[43], y la búsqueda local[38], La principal ventaja de los metaheurísticos frente a los métodos tradicionales de búsqueda es que pueden saltar por encima soluciones óptimas locales en busca de los óptimos globales. Su principal desventaja es que no son problemas en específico y por lo tanto, producen soluciones a una tipo de problema específico.

Por otra parte, la programación dinámica, un método de optimización ya conocido[44, 45], las redes neuronales artificiales[46–48], programación genética[49]y algoritmo sistemas inmunes[50] son empleados con éxito para la solución de BAP en líneas de producción. Por último, una serie de estudios incluyendo[48, 49, 51–53] emplean diversos diseños experimentales para la evaluación de las soluciones al problema de asignación de almacenes temporales.

### 2.3. Métodos por evaluación

Como se dijo antes, básicamente dos métodos se utilizan para la evaluación: Los métodos de análisis y simulación. El método analítico exacto muestra los resultados basados en los modelos de colas, estos son difíciles de obtener, y sólo están disponibles para las líneas de producción cortas. Por mucho las líneas de producción, métodos de evaluación general aproximadas están empleados. evaluativa aproximado que se utiliza con mayor frecuencia métodos para resolver el problema de asignación de memoria intermedia son la descomposición método, el método

de agregación, y la generalizada método de expansión.

Los métodos de descomposición es uno de los más empleados por la clasificación para resolver el BAP [5, 22, 40, 42, 54, 55]. La idea común de este método es descomponer el modelo original por medio de un análisis de un conjunto de subsistemas más pequeños que son más fáciles de resolver. Las ventajas principales del método de descomposición es su eficiencia computacional y la precisión para llegar a la solución. Sin embargo, la desventaja es que puede ser aplicable sólo bajo las suposiciones de que las tasas de producción son deterministas o siguen una distribución exponencial y las tasas de falló de máquina y reparación están geoméricamente o exponencialmente distribuidos al azar.

Otro método de aproximación, es el modelo de teorías de colas, este es un método de expansión generalizada. Que se utiliza bajo supuestos y se puede utilizar para tiempos de servicio en general, estas deben de estar distribuidas y las máquinas fiables y puede ser aplicable a dividir y combinar configuraciones así como configuraciones en serie [25, 31, 56–60].

Otro método de evaluación que se puede utilizar para resolver el BAP es el método de agregación[35, 43]. Se emplea con éxito el método para evaluar el rendimiento de la asignación de almacenes intermedios, en las líneas de producción no fiables. Lo básico de esta idea es en primer lugar una de las dos estaciones de un búfer-sublínea por una sola estación equivalente. A continuación, esta estación equivalente se combina con un búfer y de la estación de la línea original para formar una nueva de dos estaciones de un búfer-sublínea, el cual se agrega en una sola estación equivalente.

Este proceso se repite hasta que el último o primera estación es alcanzado, dependiendo de la dirección de la agregación. Si el objetivo es modelar de forma realista una línea larga y un sistema complejo, la simulación ofrece muchas ventajas en comparación a los métodos analíticos. Sin embargo, la principal desventaja de modelos de simulación es que es mucho tiempo. Los modelos de simulación son los más adecuado para abordar el diseño y problemas operacionales a nivel detallado, donde otras técnicas matemáticas no son lo suficientemente precisas para ser aplicado.

Los estudios de [19, 23, 38, 46–48, 51, 61–63] se puede dar como aplicaciones de simulación para la solución de BAP.

## 2.4. Métodos exactos

El estudio de las líneas de transferencias sin restricciones en las máquinas y almacenes intermedios temporales finitos, fueron realizados por Hillier y So en 1991 y bolling en el año

---

de 1993 [64, 65], con máquina de tiempo en proceso exponenciales, ambos trabajos emplean una evaluación completa de todos los almacenes temporales realizando la combinación total, minimizaban el espacio y maximizando la tasa de producción, los autores empleaban un método exacto para determinar la tasa de producción, la dificultad con este enfoque es que las soluciones exactas pueden ser solo para sistemas pequeños.

---

# Capítulo 3

## Marco Teórico

### 3.1. Líneas de transferencia

#### 3.1.1. Características principales y clase de Líneas de Producción.

Hay tres clases principales de fabricación de líneas de flujo. Estos son:

1. Los sistemas asíncronos.
2. Sistemas síncronos.
3. Sistemas continuos asíncronos y síncronos.

La única diferencia entre ellos es que todas las operaciones y las características de las máquinas a sí como los cambios de estado en la línea se producen simultáneamente, así como en los niveles, del almacén temporal en el sistema síncrono. En los sistemas asíncronos, las máquinas no están obligadas a iniciar o detener sus operaciones en el mismo instante. Aun cuando las máquinas se han fijado, igual que los tiempos de operación la presencia de los almacenes temporales entre ellos que les permita iniciar y detener de manera independiente, siempre que los búfers intermedios son no vacíos y no completos. En algunas aplicaciones, los tiempos de operación pueden ser aleatorios. Por último, la insuficiencia incierta y reparación incluso provocar falta de sincronización de los sistemas asíncronos en sistemas síncronos, todas las máquinas están obligadas a iniciar y detener sus operaciones en el mismo tiempo.

La característica que distingue a los sistemas continuos de los demás es que el material se trata como continuo en lugar de discretos. Es decir, en lugares de partes discretas que se desplazan, el almacén temporal a la máquina y viceversa en casos específicos, no es un líquido que es transferido continuamente. Los sistemas continuos son, naturalmente los sistemas de

producción en que el material procesado es un líquido en lugar de entidades discretas (por ejemplo tratamiento químico).

Una máquina se dice *bloqueada* si la parte procesada en ella no se puede poner el almacén temporal con escaso material o nulo lo inactiva y no hay ninguna parte para ser procesado en el almacén saturado. La función de un búfer es disociar máquinas es sujeto a una interrupción (un fallo o un tiempo de operación de largo, la máquina aguas arriba es sujeto a una interrupción (un fallo o un tiempo de operación de largo, la máquina aguas arriba puede todavía operar hasta que el búfer se vacía aguas abajo. Cuanto mayores sean los búfers de la línea, cuanto más tiempo antes de que el llenado o el vaciado de ocurrir, y cuanto mayor es la tasa de producción. Pares de máquinas que no tienen espacio de almacenamiento entre ellos tienen el mayor acoplamiento; y búfer infinitos, o áreas de almacenamiento que no están llenos, tienen la menor.

En la vida real, ya que todos los buffers tienen una capacidad finita, se puede producir el bloqueo. Existen dos tipos de bloqueo:

1. Bloqueo Después de servicio (BAS)
2. El bloqueo Antes de servicio (BBS)

BAS, también llamados tipo-1 bloqueo, de la producción de fabricación, bloqueo de transferencia o el bloqueo no inmediata, se produce en el instante de finalización de una parte de la máquina, si el búfer de aguas abajo está llena. En ese caso, la permanece en la máquina hasta que un espacio está disponible en el búfer de aguas abajo. Durante este tiempo, se evita que la máquina funcione y se dice que está bloqueada. Cuando un espacio se convierte disponible en la memoria intermedia de aguas abajo, la parte se transfiere inmediatamente a la del búfer de aguas abajo y la máquina puede empezar a procesar otra parte, si los hubiere. En BBS, también llamado tipo-2 bloqueo, bloqueo de la comunicación, el bloqueo de servicios o inmediato.

En BBS, también llamado tipo-2 bloqueo, bloqueo de la comunicación, el bloqueo de servicios o el bloqueo inmediato, la máquina puede empezar a procesar sólo si hay un espacio disponible en el búfer de aguas abajo. De lo contrario, tiene que esperar hasta que un espacio disponible. BBS se puede clasificar aún en función de si la posición (espacio) en la máquina puede estar ocupado o no mientras la máquina está bloqueada. Estos dos casos están bloqueados Antes Servicio con posición ocupada (BBS-PO) si el espacio en la máquina se utiliza

durante la obstrucción y bloqueo Antes de Servicio con posición no ocupada (BBS-PNO) si el espacio de la máquina no se utiliza durante la obstrucción.

En algunos sistemas, las máquinas son propensas a fallos. En la literatura, generalmente de dos, se consideran dos tipos de fallos los cuales son:

1. Operación Fallas dependientes(ODD)
2. Tiempo Fallas Dependientes(FDT)

Las probabilidades son las fallas que están relacionados con el procesamiento de las partes y por lo tanto sólo puede ocurrir cuando la máquina ésta funcionando. La máquina está trabajando medios que la máquina está hasta (operativos) y no está inactivo. Por otro lado, TDFs no están relacionados con el procesamiento de una parte y por lo tanto pueden ocurrir en cualquier momento, incluyendo el tiempo cuando una máquina está inactivo.

Cuando se produce un fallo, la máquina no puede procesar cualquier material. por lo que la corriente arriba de búfer no puede perder material y el búfer de aguas abajo no puede ganar material. Sistemas de qué máquinas pueden fallar se llaman líneas de flujo con máquinas fiables (FLRMs. En FLRMs, toda la aleatoriedad es debido a la variabilidad del proceso, mientras que, en Flums, aleatoriedad se debe tanto a los tiempos del proceso varían y fallas.

El material llega y parte de la línea de flujo en una variedad de maneras diferentes. Eso siempre es posible que la materia prima no esté disponible, o la eliminación de los productos terminados pueden retrasarse en la vida real. tales sistemas son sistemas no saturados. Por otro lado, en la literatura, casi siempre se supone que la primera máquina nunca se queda sin material y la última no ésta bloqueada. Tales sistemas se denominan sistemas saturados. Sin embargo, es posible para modelar un sitema no saturado con un sistema saturado mediante la adición de un paro de máquina por falta de materia prima inicial de la máquina como el proceso de llegada y una máquina de final que no está bloqueada (que significa que tiene infinita capacidad de búfer aguas abajo) ya que el proceso de salida a la línea. Por lo tanto, la segunda máquina del modelo correspondiente a la primera máquina del sistema real y la máquina del modelo corresponde a la última máquina del sistema real.

### 3.1.2. Resultados generales relacionados con la tasa de producción de la Línea

Varias medidas de desempeño son de interés en el análisis de las líneas de flujo. Lo más importante es la tasa de producción,  $P$ , que es el número promedio de piezas que sale del sistema por unidad de tiempo.

La tasa de producción de una línea está limitado de dos maneras. En primer lugar, el rendimiento puede ser no mayor que la de la máquina con la velocidad de producción aislado más pequeño. La máquina individual de una tasa de producción de una máquina es la tasa que operaría a si no fuera en un sistema con otras máquinas y búfer. Cuando las máquinas tienen diferentes aislados las tasas de producción, sus capacidades, excepto al más bajo se desperdician en gran medida. En segundo lugar, las interrupciones que causan falta de sincronización de búfers para estar llenos o vacíos también máquina de residuos de la capacidad. Los búfer se convierten en vacío o lleno porque las máquinas fallan o toman mucho tiempo para material de proceso en diferentes momentos. Si todas las máquinas podrían ser perfectamente, no se sincronizan sólo en la realización de las operaciones, sino también en su defecto y conseguir reparado, búfer haría no afectar el flujo. Es la falta de sincronización que hace que las máquinas que mueren de inanición o bloqueado, y por lo tanto perder la oportunidad de trabajar. Una relación fundamental de las líneas de flujo es la conservación de flujo que establece que todas las máquinas tienen las mismas tasas de producción, es decir:

$$P_1 = P_2 = \dots = P_N = P$$

Conservación de flujo se mantiene para las empresas y también para Flums siempre que existan hay desguance de piezas. Conservación de flujo puede ser establecido mediante el uso de trayecto de la muestra enfoque. El comportamiento trayecto de la muestra de cualquier línea de flujo puede ser descrita por medio de ecuaciones recursivas. Estas ecuaciones se definen como las *ecuaciones recursivas*. Estas ecuaciones se definen como las *ecuaciones de evolución* del flujo de línea.

### 3.1.3. Cuestiones de búfer

Los métodos de evaluación y optimización, se utilizan generalmente de forma iterativa para resolver el BAP. El primer método se utiliza para calcular el valor de la función objetivo, que suele ser la tasa de producción del sistema. El segundo método se aplica para encontrar los tamaños óptimos o casi óptimos de los tampones. Estos 2 métodos se comunican entre sí de manera iterativa dentro de una configuración de bucle cerrado.

Los métodos de evaluación, que proporcionan estimaciones de diversas medidas de rendimiento de la tasa de producción, se basan en procedimientos analíticos y simulación.<sup>3</sup> Los enfoques analíticos suelen ser aplicables bajo ciertas suposiciones, como las distribuciones de tiempo de procesamiento específicas o un pequeño número de estaciones. Se derivó una fórmula analítica exacta para calcular la tasa de producción para una línea de búfer de 2 máquinas-1 con tiempos de procesamiento determinísticos en el trabajo de Buzacott.<sup>4</sup> La mayoría de estos métodos se basaban en técnicas de descomposición o agregación. En el trabajo de Gershwin,<sup>9</sup> se presentó un enfoque de descomposición aproximado para evaluar la tasa de producción de una clase de sistemas de colas en tándem con búfers finitos. Dallery et al<sup>10</sup> desarrollaron las ecuaciones de descomposición para el modelo de material continuo y el algoritmo DDX (Dallery-David-Xie) para las líneas homogéneas para evaluar el rendimiento.

#### Procedimiento de para iniciar el cálculo de tamaño de búfer.

Para iniciar con el problema primero mencionaremos cual es el tamaño de la región factible dependiendo del tamaño de línea y de la capacidad de almacenamiento, la cual se calcula con la siguiente expresión.

$$\binom{N + C - 2}{C} = \frac{(N + C - 2)!}{(N - 2)!(C)!} \quad (3.1)$$

En donde  $C$  es el tamaño máximo de almacén y  $N$  es el número de espacios entre máquinas. antes de comprender la importancia del problema debemos conocer las condiciones iniciales del problema.

$$\lambda_i = \frac{1}{T_i} \quad (3.2)$$

$$\mu_i = \frac{1}{MTTF_i} \quad (3.3)$$

$$r_i = \frac{1}{MTTR_i} \quad (3.4)$$

$$e_i = \frac{MTTF_i}{MTTF_i + MTTR_i} = \frac{r_i}{(\mu_i + r_i)} \quad (3.5)$$

$$\rho_i = \lambda_i * e_i = \frac{\lambda_i * r_i}{(\mu_i + r_i)} \quad (3.6)$$

Donde:

$T_i$  = Tiempo promedio de trabajo de la máquina  $M_i$

$\lambda_i$  = Tasa de producción de la máquina  $M_i$

$MTTF_i$  = Tiempo promedio de falla de la máquina  $M_i$

$MTTR_i$  = Tiempo de promedio de reparación de la máquina  $M_i$

$\mu_i$  = Tasa promedio de falla de la máquina  $M_i$

$r_i$  = Tasa promedio de reparación de la máquina  $M_i$

$e_i$  = Eficiencia individual de la máquina  $M_i$

$\rho_i$  = Tasa de producción individual de la máquina  $M_i$

### Ejemplo numérico.

En un primer paso es definir los puntos críticos al definir la capacidad del búfer por lo que se tiene 3 funciones para calcular el número.

$$f_1(\rho_i, \rho_{i+1}) = \frac{1}{\min(\rho_i, \rho_{i+1})} \quad (3.7)$$

$$f_2(\rho_i, \rho_{i+1}) = \frac{1}{\rho_i} + \frac{1}{\rho_{i+1}} \quad (3.8)$$

$$f_3(\rho_i, \rho_{i+1}) = \frac{1}{(\rho_i + \rho_{i+1})} \quad (3.9)$$

Por lo que se debe de valorar de acuerdo con cada uno cual es la mejor y calcular en cada fase, para ello se nombra de esta forma a  $cr_i$  para iniar la primera iteración.

$$cr_i = f_1, f_2, f_3 \quad (3.10)$$

Para este caso numérico se realizará el ejemplo con  $f_3$  y se mostraran los resultados en las siguientes tablas.

### Datos iniciales de la línea de ejemplo.

A continuación se dan los datos iniciales del problema en la Tabla 3.1, los cuales se toman

Tabla 3.1: Datos iniciales de la línea de producción

	Tasa de procesamiento( $\lambda_i$ )	Tasa de falla( $\mu_i$ )	Tasa de reparación( $r_i$ )	Eficiencia ( $e_i$ )	Tasa de producción individual $\rho_i$
Máquina 1	3.7	0.07	0.17	0.70833	2.62083
Máquina 2	1.5	0.11	0.37	0.77083	1.15625
Máquina 3	1.1	0.49	0.78	0.61417	0.67559
Máquina 4	3	0.19	0.5	0.72464	2.17391

de [66] para realizar los cálculos. basados en los datos se calculan los relativos críticos  $RC_i$  con las siguientes expresiones.

$$cr_i = f_3(\rho_i, \rho_{i+1}) \longrightarrow cr_i = \frac{1}{(\rho_i + \rho_{i+1})} \quad (3.11)$$

$$RC_i = \frac{cr_i}{cr_1 + cr_2 + \dots + cr_{N-1}} \quad (3.12)$$

Para calcular el tamaño de búfer inicial  $KI_i$  debemos multiplicar a  $C$  que es la cantidad total de elementos de cada búfer en la línea de producción con la siguiente expresión:

$$KI_i = RC_i * C \quad \text{para } i = 1, 2, \dots, (N - 1) \quad (3.13)$$

Evaluando en la expresión 3.11 tomando el punto crítico  $f_3(\rho_i, \rho_{i+1})$  para realizar la evaluación numérica para fines prácticos el restos de los puntos críticos se expresaran en las Tablas consecuentes.

$$cr_1 = \frac{1}{(2,62083 + 1,15825)} = 0,264755$$

$$cr_2 = \frac{1}{(1,15825 + 0,67559)} = 0,545899$$

$$cr_3 = \frac{1}{(0,67559 + 2,17391)} = 0,350938$$

y para calcular los relativos críticos de 3.12

$$RC_1 = \frac{0,264755}{1,161592} = 0,2279239$$

$$RC_2 = \frac{0,545899}{1,161592} = 0,4699577$$

$$RC_3 = \frac{0,350938}{1,161592} = 0,3021184$$

Tabla 3.2: Resumen de cálculos para  $f_3(\rho_i, \rho_{i+1}) = \frac{1}{(\rho_i + \rho_{i+1})}$

	$cr_i$	$KI_i$	Redondeo de $KI_i$	suma del resto a pieza	inicial búfers
búfer 1	0.264755	2.279239	2		2
búfer 2	0.545899	4.699577	4	1	5
búfer 3	0.350938	3.021184	3		3

Tabla 3.3: Resumen de cálculos para  $f_1(\rho_i, \rho_{i+1}) = \frac{1}{\min(\rho_i, \rho_{i+1})}$

	$cr_i$	$KI_i$	Redondeo de $KI_i$	suma del resto a pieza	inicial búfers
búfer 1	0.864865	2.260944	2		2
búfer 2	1.480186	3.869528	3	1	4
búfer 3	1.480186	3.869528	3	1	4

Para el tamaño inicial de  $KI_i$  se debe de multiplica a  $C = 10$  de 3.13

$$KI_1 = 0,2279239 * 10 = 2,279239$$

$$KI_2 = 0,4699577 * 10 = 4,699577$$

$$KI_3 = 0,3021184 * 10 = 3,021184$$

En la Tabla 3.2 se observa el resumen de los resultado así como el redondeo para las piezas. en el caso del búfer 2 se tiene 4.699577 por lo que se redondea a 5 para terminar con un búfer inicial de  $f_3(2 - 5 - 3)$ . Para los demás puntos se muestran en la Tabla 3.3 y 3.4 con un resultado de  $f_1(2 - 4 - 4)$  y  $f_2(2 - 4 - 4)$ .

Tabla 3.4: Resumen de cálculos para  $f_2(\rho_i, \rho_{i+1}) = \frac{1}{\rho_i} + \frac{1}{\rho_{i+1}}$

	$cr_i$	$KI_i$	Redondeo de $KI_i$	suma del resto a pieza	inicial búfers
búfer 1	1.246423	2.253253	2		2
búfer 2	2.345051	4.239326	4		4
búfer 3	1.940186	3.507421	3	1	4

### Cálculo de la tasa de producción.

Simulamos el sistema para 45000 piezas. Dado que queremos estimar el valor de rendimiento de estado estable para cualquier asignación de búfer, las observaciones durante el período de

simulación tendrán un efecto de aumento del sesgo, en el valor de rendimiento estimado. Por lo tanto, debemos decidir la duración del período de simulación y no debemos tener en cuenta las observaciones correspondientes a este período durante la estimación del rendimiento. Al predecir el período de simulación, deberíamos lograr dos objetivos en conflicto. Cuantos más datos descartemos pertenecientes al período de simulación, menor será el sesgo que tendremos en la estimación del rendimiento. Por otro lado, cuantos más datos descartemos del período de simulación, mayor será la variabilidad que tendremos en nuestro valor de rendimiento estimado. Por lo tanto, debemos comprometernos entre reducir el sesgo y aumentar la varianza para nuestra estimación de rendimiento.

Para el cálculo de la tasa de producción de una línea completa se tiene que realizar un procedimiento diferente y se evaluará de las siguientes expresiones:

$$X_{bar} = \sum_{i=1}^n \frac{X_{bar}}{n} \quad (3.14)$$

$$S^2 X = \sum_{i=1}^n \frac{(X_i - X_{bar})^2}{n - 1} \quad (3.15)$$

$$S^2(X_{bar}) = \frac{S^2(X)}{n} \quad (3.16)$$

Ya que las  $X_i$  están distribuidas normalmente se tiene que  $h$  es la mitad del ancho del intervalo con  $100(1-\alpha)\%$  para la tasa de producción se tiene:

$$h = t_{(n-1), 1-(\alpha/2)} * S(X_{bar}) \quad (3.17)$$

por consecuencia  $100(1-\alpha)\%$  para la salida de la tasa de producción, el intervalo es centrado por:

$$X_{bar} \pm t_{(n-1), 1-(\alpha/2)} * S(X_{bar}) \quad (3.18)$$

para este punto se decide el número de replicas por lo que se tiene que para un  $\alpha = 0,05$  y un estadístico  $t$ , que el número más razonable es de 10.

Comparamos los valores de rendimiento para cualquier asignación entre ellos para decidir si proceder o no a la mejor solución. En estos pasos, para lograr comparaciones más convenientes de asignaciones alternativas y resultados más precisos, utilizamos números aleatorios comunes. Esto significa que los números aleatorios generados para las mismas operaciones en

Tabla 3.5: Simulación de los rendimientos de las tasa de producción para los dos casos.

Número de réplicas	Rendimiento B(2-5-3)	Rendimiento B(2-4-4)	Diferencia de rendimientos
1	0.645809	0.642097	0.003712
2	0.650929	0.647573	0.003356
3	0.647366	0.644907	0.002459
4	0.651672	0.648316	0.003356
5	0.644689	0.640807	0.003882
6	0.65159	0.647887	0.003703
7	0.646329	0.643114	0.003215
8	0.651822	0.649662	0.00216
9	0.648823	0.64513	0.003693
10	0.651469	0.648163	0.003306
promedios	0.6490498	0.6457656	0.0032842

Tabla 3.6: Valores de rendimiento de las tasa de producción simuladas para los dos casos

Caso	Promedio ( $X_{bar}$ )	ancho medio(h)	(h / $X_{bar}$ )	$\%X_{bar} - h$	$\%X_{bar} + h$
(2,4,4)	0,6457656	0,002151472	0,33316606	0,643614128	0,64791707
(2,5,3)	0,6490498	0,001997834	0,30780903	0,647051966	0,65104763

la línea para cada solución serán los mismos. Más claramente, los tiempos consecutivos de procesamiento, reparación y falla de cualquier máquina serán iguales para cada asignación alternativa durante la simulación. En consecuencia, la diferencia entre las alternativas proviene solo de las capacidades asignadas a las ubicaciones de los búferes, lo que nos permite alcanzar comparaciones más saludables.

Para alcanzar resultados más saludables, también utilizamos la prueba de t para dos asignaciones alternativas. La siguiente Tabla 3.5 muestra los valores de rendimiento para dos alternativas y la Diferencias de rendimiento entre ellos para cada replica en la simulación:

Estos valores de rendimiento se muestran en la siguiente tabla con niveles de confianza del 95 %:

Con base en los valores dados en la Tabla 3.5, obtenemos el intervalo de confianza de t para las diferencias de rendimiento de la siguiente Tabla 3.7.

Tabla 3.7: Resultados internos entre los dos casos

Promedio ( $X_{bar}$ )	ancho medio(h)	(h / $X_{bar}$ )	$\%X_{bar} - h$	$\%X_{bar} + h$
0.0032842	0.000402048	12.24189878	0.002882	0.003686

La superioridad del tercer candidato también se puede ver desde el estadístico-t Intervalo de confianza para las diferencias de rendimiento para cada réplica. Todos estos los hallazgos respaldan nuestro diseño para el procedimiento de asignación inicial.

### 3.1.4. Características de una línea de producción.

En este documento, examinamos el BAP en una línea de producción en serie, como se muestra en la Figura 1.1, donde los rectángulos representan máquinas, y los círculos indican búferes, indicados como  $M_i, i = 1, \dots, k$ , y  $B_i, i = 1, \dots, N - 1$ , respectivamente. Los supuestos del **BAP** en una línea de producción en serie se enumeran a continuación:

Cada parte pasa por todas las máquinas y búferes en secuencia, desde la máquina  $M_1$  hasta la máquina  $M_k$ . La máquina  $M_i$  está arrancando en el momento  $t$  si  $M_{i-1}$  está trabajando y el búfer intermedio  $B_{i-1}$  está vacío; La máquina  $M_i$  está bloqueada en el momento  $t$  si  $M_{i+1}$  está abajo y la memoria intermedia  $B_i$  está llena. La máquina  $M_1$  tiene un suministro ilimitado de piezas, mientras que la máquina  $M_n$  posee un área de almacenamiento ilimitado. Por lo tanto, la máquina  $M_1$  nunca está muerta de hambre y la máquina  $M_n$  nunca está bloqueada. Los tiempos de procesamiento de todas las partes son constantes e iguales para todas las máquinas y se consideran como una unidad de tiempo. El tiempo de transporte es despreciable. Los fallos son dependientes de la operación y detectados instantáneamente. Una máquina no puede fallar cuando está hambrienta o bloqueada. Los tiempos de falla y los tiempos de reparación de las máquinas son independientes y se distribuyen de manera exponencial. Las máquinas pueden ser totalmente reparadas. El proceso de transición de estado tiene la propiedad de Markov (es decir, la distribución de probabilidad de estado de los estados futuros de una máquina depende solo del estado actual, en lugar de sus eventos precedentes).

La velocidad de producción aumenta monótonamente a medida que el aumento de las capacidades de amortiguamiento. Esto es propiedad de *monotonía*. Consideremos dos líneas de flujo, L1 y L2, que tienen máquinas idénticas, pero con diferente capacidad de búferes  $K_1$  y  $K_2$ . La capacidad de cada búfer en L2 es al menos tan grande como la memoria intermedia correspondiente en L1. Es decir,  $K_1 \leq K_2$ .

Entonces la tasa de producción de las líneas de flujo satisface

$$P(K_1) \leq P(K_2)$$

La tasa de producción de una línea de flujo en el que una memoria intermedia es infinita

puede obtenerse por la descomposición de la línea en dos sub-líneas de este búfer de capacidad infinita. Deje que sea la parte de la línea L que consta sólo de las primeras máquinas y los primeros  $(i - 1)$  búfer y de manera similar sea L la parte de la línea L que consta sólo de una última  $(N - i)$  máquinas y el último  $(N - i - 1)$  almacena temporalmente en el que el puesto de almacen intermedio  $B_i$  tiene la capacidad infinita deje a  $P^a$  y  $P^b$  ser las tasas de producción de líneas de  $L^a$  y  $L^b$  respectivamente. Entonces la tasa de producción,  $P_{inf}$ , de la líneas en el búfer es infinita

$$P_{inf} = \min(P^a, P^b).$$

Combinando el resultado anterior con la propiedad de monoticidad, se obtiene el siguiente límite superior para la tasa de producción de la línea donde no hay infinito capacidad de búfer:

$$P \leq \min(P^a, P^b)$$

Mediante la aplicación de esta descomposición varias veces, vemos que la tasa de producción de una línea de flujo está limitada por la velocidad de producción aislada de la máquina que tiene la menor tasa de producción aislada como se indica a continuación:

$$P \leq \min(p_i) \quad i = 1 \text{ to } N$$

Más escrito límite superior de la velocidad de producción de la línea original puede derivarse desde el enfoque de descomposición dado como:

$$P \leq \min(P^{i,i+1}) \quad i = 1 \text{ to } (N - 1)$$

Donde  $P^{i,i+1}$  es la tasa de producción de la línea de flujo de dos máquinas que consiste en  $M_i, B_i, M_{i+1}$ .

Este límite superior puede ser útil ya que las tasas de producción de las dos máquinas de las líneas de flujo se puede calcular exactamente en la mayoría de los casos. El también se pueden utilizar para obtener el siguiente límite inferior en la tasa de producción de la línea original:

$$P^0 < P$$

Donde  $P^0$  es la tasa de producción de línea de flujo sin almacenamiento intermedio.

### 3.1.5. La reversibilidad y la dualidad

Considere una línea de flujo  $L^r$ , que se obtiene de la línea de flujo  $L$  invirtiendo el flujo de partes. La primera máquina en  $L^r$  es la misma que la última máquina en  $L$ . Más generalmente  $M_i$  en la línea  $L^r$  es el mismo que el  $M_{N-i+1}$  en la línea  $L$ . También los búfers están invertidos en línea  $L^r$ . Entonces la tasa de producción de la línea invertida  $L^r$  es la misma que la tasa de producción de formación original  $L$  si ambas líneas su mecanismo de bloqueo es BAS. Esto es propiedad de reversibilidad.

La prueba de esta propiedad se basa en la comparación de las trayectorias de la muestra de los dos sistemas de nuevo utilizando las ecuaciones de evolución.

Consideremos ahora el caso de BBS. En ese caso, hay una mucho más fuerte equivalencia entre los dos sistemas ( $L^r$  y  $L$ ). Esta equivalencia se basa en el concepto de trabajo/agujero (o/agujero en parte) la dualidad. La idea es que en la línea  $L$ , siempre que una parte se mueve en una dirección, un orificio (espacio vacío) se mueve en la otra dirección. En el caso de BBS, es fácil comprobar que el mismo comportamiento de las piezas en el sistema invertido es el mismo como el comportamiento de agujeros en el sistema original y viceversa. Como resultado, la distribución en estado estacionario de las piezas en la línea invertida es distribución de los agujeros en la línea original. Esta equivalencia implica especialmente estos dos sistemas tienen la misma tasa de producción.

El concepto de la dualidad de trabajo/agujero todavía tienen sentido en el caso de BAS. Sin embargo, el comportamiento de las piezas en el sistema invertido ya no es el mismo que

en el comportamiento de agujeros en el sistema original.

En este capítulo damos el resumen de los antecedentes de las líneas de producción. Vale la pena para dar a este fondo ya que todas las investigaciones sobre las líneas de producción en el uso de literatura cualquiera de estas clases y características de las líneas de producción como el marco mientras que el modelado los introducen nuevas derivaciones utilizando básicamente los resultados generales relativos a las líneas de flujo. Por lo tanto, el contenido de este capítulo ayudará al lector a comprender el siguiente capítulo dónde vamos a introducir las investigaciones en las líneas de producción en la literatura.

### 3.2. Redes neuronales artificiales

Una RNA es un algoritmo de cálculo que se basa en una analogía del sistema nervioso. La idea general consiste en emular la capacidad de aprendizaje del sistema nervioso, de manera que la RNA aprenda a identificar un patrón de asociación entre los valores de un conjunto de variables predictoras (entradas) y los estados que se consideran dependientes de dichos valores (salidas). Desde un punto de vista técnico, la RNA consiste en un grupo de unidades de proceso (nodos) que se asemejan a las neuronas al estar interconectadas por medio de un entramado de relaciones (pesos) análogas al concepto de conexiones sinápticas en el sistema nervioso. A partir de los nodos de entrada, la señal progresa a través de la red hasta proporcionar una respuesta en forma de nivel de activación de los nodos de salida. Los valores de salida proporcionan una predicción del resultado en función de las variables de entrada. Desde el punto de vista de implementación práctica, los nodos son elementos computacionales simples que emulan la respuesta de una neurona a un determinado estímulo. Estos elementos, como las neuronas en el sistema nervioso, funcionan como interruptores: cuando la suma de señales de entrada es suficientemente alta (en el caso de una neurona diríamos que se acumula suficiente neurotransmisor), la neurona manda una señal a las neuronas con las que mantiene contacto (se genera un potencial de acción). Esta situación se modela matemáticamente como una suma de pesos de todas las señales de llegada al nodo que se compara con un umbral característico. Si el umbral se supera, entonces el nodo se dispara, mandando una señal a otros nodos, que a su vez procesarán esa información juntamente con la que reciben de nodos adyacentes como se observa en la Figura 3.2.

Evidentemente, la respuesta de cada nodo dependerá del valor de las interacciones con los nodos precedentes dentro de la estructura de la red. Como en el caso del sistema nervioso, el poder computacional de una RNA deriva no de la complejidad de cada unidad de proceso sino

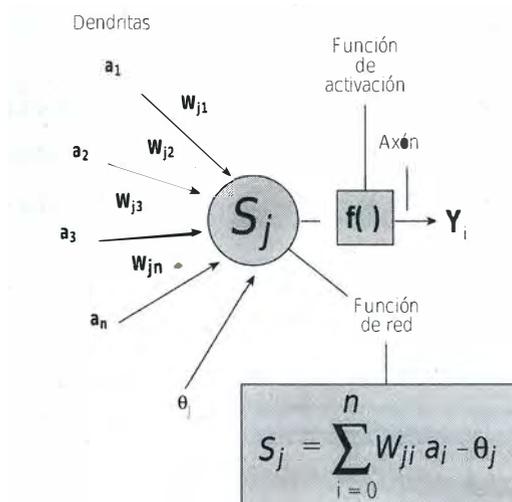


Figura 3.1: Neurona artificial

de la densidad y complejidad de sus interconexiones [67].

### 3.2.1. Funciones de activación

En este apartado vamos a hacer un inciso sobre las funciones de activación más típicas que se suelen utilizar. En concreto se trata de las funciones: escalón, lineal a trozos, sigmoide y tangente hiperbólica.

#### Función escalón.

La función escalón está definida por la expresión este tipo de neurona se refiere la literatura como neurona de McCulloch-Pitts [68]. Se emplea principalmente para clasificación, ya que la salida binaria es apropiada para la selección entre dos clases.

#### función lineal a trozos

El factor de amplificación (pendiente) en la región lineal de operación se asume que es la unidad. La forma de esta función de activación se aproxima a la respuesta de un amplificador no lineal. Se pueden dar los siguientes dos importantes casos particulares de la función lineal a trozos:

- La función lineal a trozos se convierte en un combinado lineal si la región lineal de operación se mantiene sin entrar en saturación.
- La función lineal a trozos se convierte en un escalón si el factor de amplificación en la región lineal tiende a infinito.

### Función sigmoide.

La sigmoide es una función de activación muy popular. Está definida por una función en forma de “s” y estrictamente creciente, mostrando un comportamiento suave. La ecuación que describe la sigmoide logística es la siguiente:

$$\varphi(v) = \frac{1}{1 + e^{-ax}}$$

donde  $a$  es el parámetro de pendiente. La importancia de esta función es que su derivada es siempre positiva y cercana a cero para los valores grandes positivos o negativos; además, toma su valor máximo cuando  $v$  es cero. Esto hace que se puedan utilizar las reglas de aprendizaje definidas para la función escalón, con la ventaja respecto a esta función, de que la derivada está definida para todo el intervalo. Posteriormente se verá que la derivabilidad de las funciones de activación es una característica fundamental para aplicar los algoritmos de aprendizaje

### 3.2.2. Estructura y Aprendizaje de la Red Backpropagation.

En una red Backpropagation existe una capa de entrada con  $n$  neuronas y una capa de salida con  $m$  neuronas y al menos una capa oculta de neuronas internas. Cada neurona de una capa (excepto las de entrada) recibe entradas de todas las neuronas de la capa anterior y envía su salida a todas las neuronas de la capa posterior (excepto las de salida). No hay conexiones hacia atrás feedback ni laterales entre las neuronas de la misma capa. La aplicación del algoritmo tiene dos fases, una hacia delante y otra hacia atrás. Durante la primera fase el patrón de entrada es presentado a la red y propagado a través de las capas hasta llegar a la capa de salida. Obtenidos los valores de salida de la red, se inicia la segunda fase, comparándose éstos valores con la salida esperada para así obtener el error. Se ajustan los pesos de la última capa proporcionalmente al error. Se pasa a la capa anterior con una retropropagación del error, ajustando los pesos y continuando con este proceso hasta llegar a la primera capa. De esta manera se han modificado los pesos de las conexiones de la red para cada patrón de aprendizaje del problema, del que conocíamos su valor de entrada y la salida deseada que debería generar la red ante dicho patrón. La técnica Backpropagation requiere el uso de neuronas cuya función de activación sea continua, y por lo tanto, diferenciable. Generalmente, la función utilizada será del tipo sigmoide [69].

### Entrenamiento supervisado.

Este es el tipo más popular de entrenamiento y el que utilizaremos en esta Tesis. En este modo de entrenamiento, se puede considerar intuitivamente que existe un “profesor”

que “enseña” a la red, proporcionando información para llevarla a emular la función deseada. Supongamos que un conjunto de muestras de entrada se aplica a la entrada de la red. Entonces la respuesta a la salida se compara con la respuesta deseada que proporciona el “profesor”. Éste informa a la red, ya sea el resultado correcto o incorrecto, para actualizar los pesos en consecuencia. Para este tipo de aprendizaje hace falta conocer la salida deseada de las muestras de entrenamiento

### Entrenamiento no supervisado.

Las ANNs no supervisadas también reciben el nombre de redes auto-organizativas o, más frecuentemente, redes competitivas. Estas redes de aprendizaje competitivo no cuentan con la guía de un “profesor”. Su base de funcionamiento son las técnicas de clustering o agrupamiento. Su función es agrupar muestras con características similares en el mismo cluster. Algunas redes no supervisadas son los SOM, las LVQ, los clasificadores k-NN 3 y las RBFN (en su primera fase de entrenamiento). La idea inherente de todas estas redes es que la capa oculta de neuronas debe ser capaz de extraer las características estadísticas del conjunto de datos de entrada. En la mayoría de los casos las neuronas de la capa oculta compiten entre ellas para determinar cuál está más próxima al dato de entrada, y la neurona “ganadora” es la que se actualiza, lo que significa que se mueve más cerca del dato de entrada. Este tipo de red suele denominarse “winner-takes-all ”

### 3.2.3. Algoritmo Levenberg-Marquardt

El método de Levenberg-Marquardt (LM) [70, 71] es un tipo de algoritmo de entrenamiento similar a los denominados quasi-Newton. Su descripción original viene propuesta en [70] pero su aplicación al entrenamiento de ANNs aparece por primera vez en [72]. Para entender la funcionalidad del método LM es conveniente empezar por comprender el método de Newton. El método de Newton es un método de segundo orden que constituye una alternativa a los métodos de gradiente conjugado para optimización rápida. El paso básico del método de Newton durante el epoch  $n$  es

$$w(n+1) = w(n) - H(n)^{-1}g(n)$$

donde  $w$  es el vector de pesos,  $g(n)$  es el vector gradiente actual y  $H(n)$  es la matriz Hessiana (de derivadas segundas) de la función de error respecto a los pesos, evaluada en los valores actuales de pesos y umbrales. El método de Newton generalmente converge más rápido que los métodos de gradiente conjugado. Desafortunadamente, la matriz Hessiana es compleja y costosa de calcular para las redes feedforward. Los algoritmos denominados quasi-Newton (método

de la secante) calculan y actualizan una aproximación de la matriz Hessiana sin necesidad de resolver las derivadas de segundo orden en cada iteración. La actualización realizada es función del gradiente. Al igual que los métodos quasi-Newton, el algoritmo LM fue diseñado para acometer entrenamientos rápidos de segundo orden sin tener que calcular la matriz Hessiana. Cuando la función de error tiene la forma de una suma de cuadrados (el caso típico en las redes feedforward), entonces la matriz Hessiana puede aproximarse como

$$H = J^T J$$

donde  $J$  es la matriz Jacobiana que contiene las primeras derivadas de la función de error de la red respecto a los pesos y umbrales. El gradiente se obtiene como

$$g = J^T e$$

siendo  $e$  el vector de errores de la red. El cálculo de la matriz Jacobiana se reduce a un simple cálculo del método backpropagation [72] que es mucho más sencillo que construir la matriz Hessiana. El método LM actualiza los pesos de forma similar al método de Newton pero introduciendo los cambios anteriores: donde el parámetro se convierte en el método de Newton empleando la forma aproximada del cálculo de la matriz Hessiana: Cuando  $\mu$  es grande, se puede despreciar la aportación de la matriz Hessiana al cálculo y obtenemos el método de descenso de gradiente con un tamaño de paso ( $1/\mu$ ): El método de Newton es más eficaz cerca de un mínimo de error, así que el objetivo es migrar al método de Newton tan pronto como sea posible. Así pues, se reduce  $\mu$  tras cada paso exitoso (cada vez que el error se logra reducir) y sólo se aumenta cuando el error aumenta respecto a la iteración anterior. Con esta metodología, la función de error siempre disminuye tras cada iteración del algoritmo. El algoritmo LM parece ser el método más rápido para entrenar redes feedforward de tamaño moderado (hasta varios centenares de parámetros). Su principal desventaja es que requiere almacenar las matrices Jacobianas que, para ciertos conjuntos de datos, pueden ser muy grandes, lo que significa un gran uso de memoria.

### 3.2.4. Fase de la validación.

La validación cruzada  $k$ -fold consiste en particionar los datos disponibles en  $k$  subconjuntos con aproximadamente el mismo número de muestras. El proceso de validación cruzada supone  $k$  iteraciones, en las cuales el algoritmo de entrenamiento utiliza  $k - 1$  subconjuntos para entrenar, y deja fuera uno de ellos. El subconjunto que se deja fuera va rotando en cada iteración de manera que, al final del proceso, todos los datos han formado parte del entrena-

miento y del test en algún momento (en concreto  $k - 1$  veces para entrenar y una para testar). La medida del error cometido es, por lo tanto, la media aritmética de los errores cometidos en las diferentes iteraciones para  $k = 6$ . Este método llevado al extremo, para  $k = N$ , siendo  $N$  el número de muestras totales, produce el denominado método leave-one-out (LOO), que consiste en evaluar  $N$  veces el modelo, cada vez testando con una muestra distinta. También existe la posibilidad de utilizarla para optimizar parámetros durante el proceso de entrenamiento, o seleccionar modelos de entre varios posibles. La validación cruzada por LOO funciona bien para funciones de error continuas, como el MSE, pero puede encontrar dificultades (además de las evidentes en cuanto a tiempo de computación) al utilizarse para funciones discontinuas, como las tareas de clasificación. Tampoco es recomendable para selección de modelos, ya que la falta de continuidad hace que un pequeño cambio en los datos pueda dar lugar a un cambio radical en el modelo escogido [73]. Para la elección de subconjuntos de entradas en regresión lineal, Breiman y Spector [74] encontraron que una validación cruzada  $k$ -fold con  $k = 10$  o  $k = 5$  daba mejor resultado que LOO. Kohavi [75] también obtuvo mejores resultados para una validación cruzada 10-fold que para LOO para problemas de árboles de decisión.

### 3.3. Metodología de superficie de respuesta.

La metodología de la superficie de respuesta (MSR) fue desarrollada por Box y colaboradores en los 50 [76], [77]. Este término se originó a partir de la perspectiva gráfica generada después de la aptitud matemática del modelo, y su uso ha sido ampliamente adoptado en textos sobre quimioterapia métrica. La MSR consiste en un grupo de técnicas matemáticas y de estadística que se basan en el ajuste de modelos empíricos a los datos experimentales obtenidos en relación con el diseño experimental. Hacia este objetivo, las funciones polinómicas lineales o cuadradas son empleadas para describir el sistema estudiado y, en consecuencia, para explorar (modelar y desplazar) las condiciones experimentales hasta su optimización [78].

Algunas etapas en la aplicación de la (MSR) como una tecnología de optimización, son las siguientes:

1. La selección de variables independientes de los principales efectos en el sistema a través de estudios de detección y delimitación de la región experimental, de acuerdo con el objetivo del estudio y la experiencia del investigador;
2. La elección del diseño experimental y la realización de la experimentación de acuerdo con la matriz experimental seleccionada;

3. El tratamiento matemático-estadístico del experimental datos a través del ajuste de una función polinómica;
4. La evaluación de la forma física del modelo;
5. La verificación de la necesidad y posibilidad de realizar un desplazamiento en dirección a la región mal;
6. Obtener los valores óptimos para cada estudio variable.

### Selección de variables.

Las variables pueden afectar la respuesta del sistema, y es prácticamente imposible identificar y controlar las pequeñas contribuciones de cada una. Por lo tanto, es necesario seleccionar esas variables con efectos principales. Los diseños de detección deben ser comparadas y determinar cuál de las diversas variables experimentales y sus interacciones representan efectos más significativos. Completo o fraccionado. Se pueden usar diseños factoriales de dos niveles para este objetivo principalmente porque son eficientes y económicos [79].

#### 3.3.1. Elección del diseño experimental.

El modelo más simple que se puede usar en (MSR) se basa en una función lineal. Para su aplicación, es necesario que las respuestas obtenidas se ajustan bien a la siguiente ecuación:

$$y = \beta_0 + \sum_{i=1}^k \beta_i x_i + \varepsilon$$

Donde  $k$  es el número de variables,  $\beta_0$  es el término constante,  $\beta_i$  representa los coeficientes de los parámetros lineales,  $x_i$  representa las variables, y  $\varepsilon$  es el residuo asociado a los experimentos.

Por lo tanto, las respuestas no deben presentar ninguna curvatura. Para evaluar la curvatura, se debe usar un modelo de segundo orden. Los diseños factoriales de dos niveles se usan en la estimación de los efectos de primer orden, pero fallan cuando los efectos adicionales, como los efectos de segundo orden, son significativos. Por lo tanto, un punto central en diseños factoriales de dos niveles se puede usar para evaluar la curvatura. El siguiente nivel del modelo polinómico debe contener términos adicionales que describan la interacción entre las diferentes variables experimentales. De esta forma, un modelo para una interacción de segundo orden presenta los siguientes términos:

$$y = \beta_0 + \sum_{i=1}^k \beta_i x_i + \sum_{1 \leq i < j}^k \beta_{ij} x_i x_j + \varepsilon$$

Donde  $\beta_{ij}$  representa los coeficientes de los parámetros de interacción. Para determinar un punto crítico (máximo, mínimo), es necesario que la función polinómico contenga términos cuadráticos de acuerdo con la ecuación\* presentada a continuación.

$$y = \beta_0 + \sum_{i=1}^k \beta_i x_i + \sum_{i=1}^k \beta_{ii} x_i^2 + \sum_{1 \leq i < j}^k \beta_{ij} x_i x_j + \varepsilon$$

donde  $\beta_{ii}$  representa los coeficientes del parámetro cuadrático. Para estimar los parámetros en la ecuación anterior, el diseño experimental tiene que asegurar que todas las variables estudiadas se lleven a cabo en al menos tres niveles de factores. Por lo tanto, dos modelos, respuesta simétrica diseños de superficie están disponibles. Entre los diseños simétricos de segundo orden más conocidos se encuentran el diseño factorial de tres niveles, el diseño Box-Behnken, el diseño central compuesto y el diseño Doehlert. Estos diseños simétricos difieren entre sí con respecto a su selección de puntos experimentales, número de niveles para variables y número de carreras y bloques.

---

## Capítulo 4

# Diseño de la red neuronal artificial y superficie de respuesta para líneas de producción confiables.

### 4.1. Diseño de las líneas de producción experimentales.

El diseño de las líneas de producción tiene como fundamento, los siguientes parámetros en la estructura del sistema para alcanzar un objetivo y se limita principalmente en:

- Asignación de trabajo de cada estación.
- Determinar el número de máquinas en cada estación de trabajo.
- Especificación del número de los almacenes temporales o búfers.

Las líneas de producción están clasificadas en pequeñas, medianas y grandes en donde  $K$  son el número de máquinas y  $N$  son el número de búfers a asignar, para las líneas pequeñas se consideran las estaciones  $K=3,4,\dots,11$ , las líneas medianas con estaciones  $K=12,13,\dots,30$  y las líneas grandes con estaciones con  $K=40,50,\dots,100$ . En todas las líneas se toman como confiables exponenciales y equilibradas es decir con una tasa media de servicio de  $\mu = 1$ .

#### 4.1.1. Diseño de los experimentos.

Se tiene dos experimentos para comprobar la eficiencia de la red neuronal y metodología de superficie de respuesta para observar su comportamiento.

El primer experimento es tomado de [80] del cual se anexa un conjunto de líneas de producción con características citadas anteriormente, este conjunto de 360 datos representan las tasas de producción con un número de  $K$  maquina de 3 a 20 y 1 a 20  $N$ , este conjunto de datos sirvieron para entrenar las redes neuronales y emplear la metodología de superficie de respuesta mostrados en Figura 4.1 y en el Anexo A, este rango de líneas, están catalogadas en líneas pequeñas y medianas.

Para un segundo experimento se toma valores de [81] y de [80] para estos últimos se necesita estar fuera del rango de los valores del primer experimento para completar las 55 líneas, las cuales tiene las característica principal que son de medianas a grandes para tener un grado mayor de severidad de la red y de la metodología de superficie de respuesta. se muestra los valores en la Tabla 4.1, en donde refleja el tamaño de búfers y el número de máquinas.

Tabla 4.1: Segunda prueba con 55 líneas [80, 81] de producción.

ID	Tamaño de la línea	K	N	ID	Tamaño de la línea	K	N
1	Pequeña	11	22	29	Mediana	18	36
2	Mediana	21	10	30	Mediana	19	38
3	Mediana	22	11	31	Mediana	20	40
4	Mediana	23	11	32	Mediana	21	42
5	Mediana	24	12	33	Mediana	22	44
6	Mediana	25	12	34	Mediana	23	46
7	Mediana	26	13	35	Mediana	24	48
8	Mediana	27	13	36	Mediana	25	50
9	Mediana	28	14	37	Mediana	26	52
10	Mediana	29	14	38	Mediana	27	54
11	Mediana	30	15	39	Mediana	28	56
12	Mediana	20	21	40	Mediana	29	58
13	Mediana	21	22	41	Mediana	30	60
14	Mediana	22	23	42	Grande	40	41
15	Mediana	23	24	43	Grande	50	51
16	Mediana	24	25	44	Grande	60	61
17	Mediana	25	26	45	Grande	70	71
18	Mediana	26	27	46	Grande	80	81
19	Mediana	27	28	47	Grande	90	91
20	Mediana	28	29	48	Grande	100	101
21	Mediana	29	30	49	Grande	40	80
22	Mediana	30	31	50	Grande	50	100
23	Mediana	12	24	51	Grande	60	120
24	Mediana	13	26	52	Grande	70	140
25	Mediana	14	28	53	Grande	80	160
26	Mediana	15	30	54	Grande	90	180
27	Mediana	16	32	55	Grande	100	200
28	Mediana	17	34				

Para el diseño de la red neuronal y de superficie de respuesta se toma un conjunto de 360 líneas confiables con  $\mu = 1$  compuesta de  $\mathbf{K}=3$  hasta 20 y  $\mathbf{N}=1$  hasta 20, estos datos representaran las entradas a la red neuronal, se muestra gráficamente los experimentos en

la Figura 4.1. Los experimentos se tomaron de [80] y son ideales por la información que proporcionan a un diseñador de líneas de producción sobre todo para los dos primeros casos del problema BAP-1 soportando mejor sus decisiones. Los supuestos de estas líneas de producción son los siguientes:

1. Cada parte atraviesa todas las máquinas y búferes en secuencia, desde la máquina  $M_1$  hasta la máquina  $M_k$ .
2. La máquina  $M_i$  está sin material en el momento  $t$  si  $M_{i-1}$  está iniciando y el búfer intermedio  $B_{i-1}$  está vacío; la máquina  $M_i$  está bloqueada en el momento  $t$  si  $M_{i+1}$  está inactivo y el búfer  $B_i$  está lleno.
3. La máquina  $M_1$  tiene un suministro ilimitado de piezas, mientras que la máquina  $M_k$  posee un área de almacenamiento ilimitado. Por lo tanto, la máquina  $M_1$  nunca está sin material y la máquina  $M_k$  nunca está bloqueada.
4. Los tiempos de procesamiento de todas las partes son constantes e iguales para todas las máquinas y se consideran como una unidad de tiempo. El tiempo de transporte es despreciable.
5. Los fallos son dependientes de la operación y detectados instantáneamente. Una máquina no puede fallar cuando está sin material o bloqueada. Los tiempos de falla y los tiempos de reparación de las máquinas son independientes y se distribuyen de manera exponencial.
6. Las máquinas pueden ser totalmente reparadas.
7. El proceso de transición de estado tiene la propiedad de Markov (es decir, la distribución de probabilidad de estado de los estados futuros de una máquina depende solo del estado actual, en lugar de sus eventos precedentes).

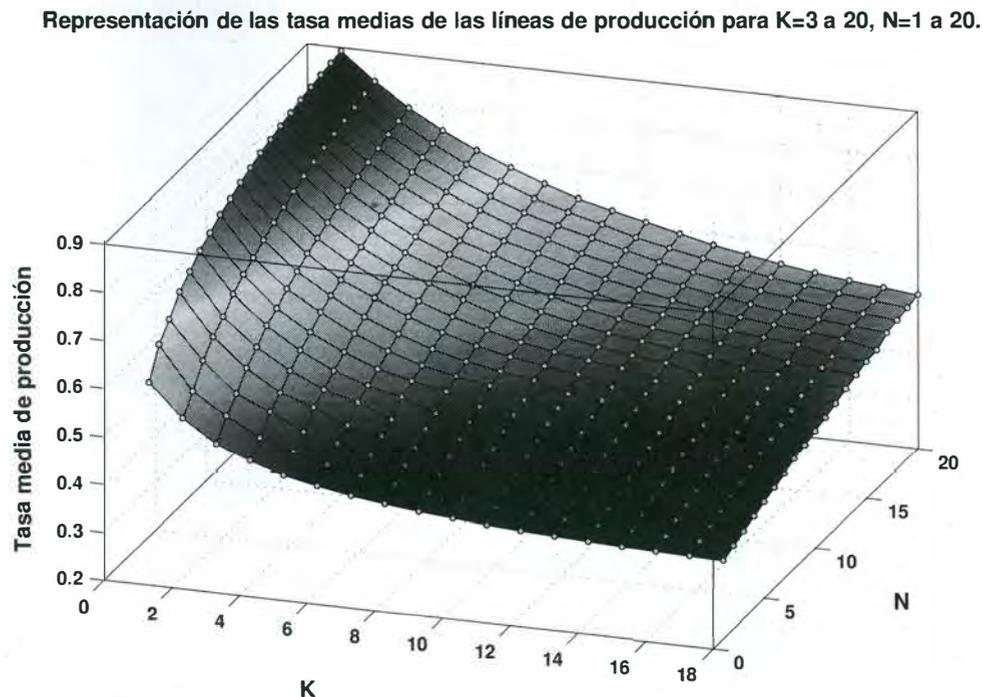


Figura 4.1: Representación de los 360 experimentos de líneas de producción

## 4.2. Diseño de la red neuronal.

Nuestro objetivo principal es diseñar y desarrollar una red neuronal artificial, que sea capaz de predecir la tasa de producción, ingresando los valores de  $K$  y  $N$ . La arquitectura de la red estará configurada de 1 a 4 capas, dos entradas  $K$  y  $N$  con valores enteros y una salida, que será la tasa de producción como se muestra en la siguiente Figura 4.3, donde las neuronas en cada capa será una combinación de  $n=5,8,10,12$  y  $15$  mostradas en la Tabla 4.2, que resultan un total de 780 redes neuronales, programando 50 iteraciones por red para encontrar la mejor. Los datos empleados para este estudio se encuentran en el Apéndice A, compuesto por 360 experimentos.

La justificación de este arreglo se debe al autor de [81] en donde probó 5 diferentes redes neuronales artificiales de 5, 8, 10, 12, y 15 neuronas ocultas analizando su rendimiento por medio de las métricas, del error medio cuadrático, teniendo un resultado bastante pequeño demostrando que este conjunto de datos es idóneo y se podrá ahorrar tiempo de computo, la Tabla 4.3 se observa el rendimiento que tuvo [81] notando que la mejor red es de 10 neuronas y comprobando lo que se ha observado por diferentes autores que aumentando el número de neuronas no significa que se encontrara un excelente resultado solo tardara más tiempo, en

base ha esto se toma la decisión de hacer una nueva configuración.

Tabla 4.2: Indicadores del conjunto de neuronas de [81].

Número de neuronas	Test A	
	MSE	R
5	1.50E-05	0.999681
8	4.11E-06	0.999912
10	3.42E-06	0.999926
12	3.63E-06	0.999916
15	2.04E-06	0.999955

Otro factor que motivo a utilizar una red neuronal fue por el trabajo de [80] en donde el foco de su trabajo fue comparar 5 algoritmos para encontrar solución al BAP-1, los 5 algoritmos son los siguientes: Algoritmos genéticos (GA), simulado recocido (SA), algoritmos myopic (MA), búsqueda Tabú (TS) y numeración completa (CE), en donde da pauta en una tabla de tiempos en donde abordaremos en el capítulo de resultados para un análisis de tiempo con respecto a la red neuronal generada, estos algoritmos son muy eficientes para líneas pequeñas, pero para líneas determinadas como grandes el tiempo de respuesta es demasiado grande por lo que se desarrolló la hipótesis que el diseño de una red neuronal podría ser más eficientes para líneas más grandes. Por lo que se genera una metodología en donde se explica a lo largo del capítulo y de forma general, en el diagrama mostrado en la Figura 4.2, se observa un algoritmo llamado REDBAP este generó las 780 redes neuronales. Una vez generados serán guardados en un archivo con extensión **.mat** para realizar una búsqueda de los mejores MSE y R para simular la red, una vez obtenida la mejor red, esta podrá calcular **n** número de redes sin importar la clasificación del tamaño en un tiempo más corto.

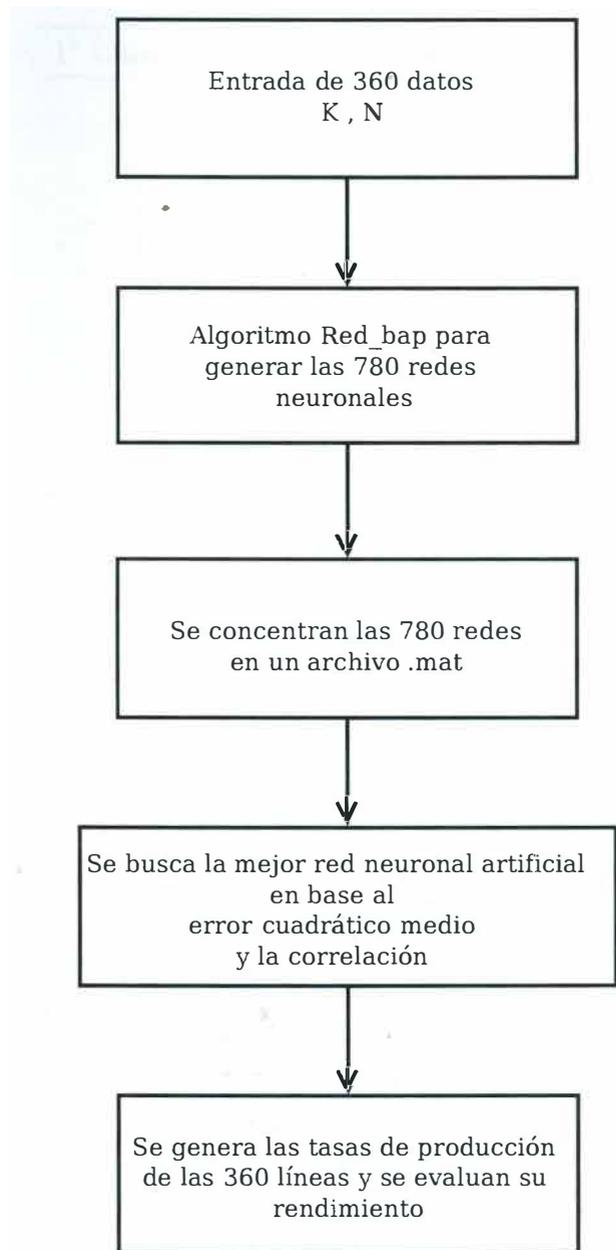


Figura 4.2: Metodología para el diseño de la Red Neuronal Artificial

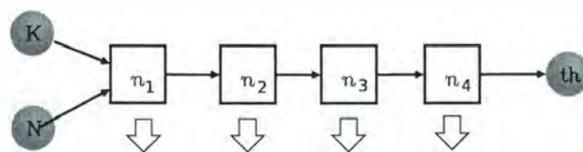


Figura 4.3: 780 combinaciones de redes neuronales

Tabla 4.3: Arreglo de 780 combinaciones de 4 capas para la red neuronal

1° Capa	2° Capa	3° Capa	4° Capa
5	5 - 5	5- 5 - 5	5 -5 -5 - 5
8	5 - 8	5- 5 - 8	5 -5 -5 - 8
10	5 - 10	5- 5 -10	5 -5 -5 -10
12	5 -12	5- 5 -12	5 -5 -5 -12
15	5 -15	5- 5 -15	5 -5 -5 -15
	8 - 5	5- 8 - 5	5 -5 -8 - 5
	8 - 8	5- 8 - 8	5 -5 -8 - 8
	8 -10	5- 8 -10	5 -5 -8 -10
	8 -12	5- 8 -12	5 -5 -8 -12
	8 -15	5- 8 -15	5 -5 -8 -15
	10- 5	5- 10- 5	5 -5 -10- 5
	10- 8	5- 10- 8	5 -5 -10- 8
	10-10	5- 10-10	5 -5 -10-10
	10-12	5- 10-12	5 -5 -10-12
	10-15	5- 10-15	5 -5 -10-15
	12- 5	5- 12- 5	5 -5 -12- 5
	12- 8	5- 12- 8	5 -5 -12- 8
	12-10	5- 12-10	5 -5 -12-10
	12-12	5- 12-12	5 -5 -12-12
	12-15	5- 12-15	5 -5 -12-15
	15- 5	5- 15- 5	5 -5 -15- 5
	15- 8	5- 15- 8	5 -5 -15- 8
	15-10	5- 15-10	5 -5 -15-10
	15-12	5- 15-12	5 -5 -15-12
	15-15	5- 15-15	5 -5 -15-15
		⋮ - ⋮ - ⋮	⋮ - ⋮ - ⋮ - ⋮
		15-15-15	15-15-15-15
5	25	125	625 =780

#### 4.2.1. Configuración de la red neuronal artificial

La configuración de la red es de una hasta cuatro capas con una combinación de 5, 8, 10 y 15 neuronas que dio como resultado 780 redes para obtener ésta configuración se realizó las siguientes instrucciones.

```
configuración de las capas con 5 8 10 12 y 15 neuronas
function conf = generaConf(nhl)
nros = [5 8 10 12 15];
tam = size(nros,2);
switch(nhl)
    case 1, conf = nros';
    case 2,
        conf = zeros(tam^2,nhl);
        indice = 1;
```

Estas líneas son llamadas en el código principal de tal forma que realiza las 780 redes.

```
nmaxHL = 4;
nmaxNeurons = 15;
ind_arch = 0;

for capa=1:nmaxHL
    conf = ones(1,capa);
    topeNeurons = ones(1,capa) * nmaxNeurons;
    disp(['Capas: ' num2str(capa)]);
    for indice=1:(nmaxNeurons^capa)
        disp(conf);
        suma = 0;
        % Create a Fitting Network
        hiddenLayerSize = conf;

        for k=1:capa
            if mod(conf(k),nmaxNeurons) == 0
                conf(k) = 1;
            else
                conf(k) = conf(k) + 1;
                break;
            end
        end
    end
end
end
```

Para esta red multicapa, la práctica general es dividir los datos en 3 subconjuntos.

1. El primer de ellos es el de entrenamiento que se utiliza para el cálculo del gradiente y actualizar la red de pesos y sesgos en nuestro diseño toma el 70 % del total de los datos, estos se toman aleatoriamente.
2. El segundo subconjunto es la validación tomando el error como punto de referencia, éste normalmente disminuye durante la fase inicial del entrenamiento, sin embargo cuando la red comienza a tratar los datos el error en la validación se incrementa, para contrarrestar estos guardan los pesos y sesgos mínimos para evitar un desbordamiento en la red, se toman para la validación un 15 % del total de los datos.
3. La prueba del error del sistema no se utiliza durante el entrenamiento, pero se utiliza para comparar los diferentes modelos si la prueba alcanza un mínimo en el número de iteraciones que la validación, esto podría indicar una mala división del conjunto de los datos para este subconjunto se toma también un 15 %.

```
% Setup Division of Data for Training, Validation, Testing
```

```
net.divideParam.trainRatio = 70/100;
```

```
net.divideParam.valRatio = 15/100;
```

```
net.divideParam.testRatio = 15/100;
```

### 4.2.2. Algoritmo de entrenamiento

Para el proceso de entrenamiento de la red neuronal se emplea el algoritmo Levenberg-Marquardt backpropagation con el comando **Trainlm** que es una función de entrenamiento en red, actualiza los valores de peso y sesgo según la optimización de Levenberg-Marquardt.

**Trainlm** es a menudo el algoritmo más rápido backpropagation en toolbox de Matlab, y es altamente recomendable como un algoritmo supervisado de primera elección, aunque requiere más memoria que otros algoritmos. Para cada capa se emplea la función de transferencia sigmoide.

$$f(x) = \frac{1}{1 + e^{ax}} \quad (4.1)$$

Donde  $a$  es un parámetro de la pendiente.

El entrenamiento se realiza de acuerdo con los parámetros como se muestra en el código.

```
% Train the Network
[net,tr] = train(net,inputs,targets);
```

En la primer fase se declaran las entradas en donde **Target** representa los 360 experimentos de las líneas de producción, las entradas **K** y **N**.

Los indicadores para medir el rendimiento de la red neuronal, son los siguientes:

1. El error cuadrático medio (MSE) [42]:

$$MSE = \frac{1}{n} \sum_{i=1}^n (Target_i - Output_i)^2 \quad (4.2)$$

Donde  $n$  es el número de diferentes datos de entrada de la red Neuronal,  $Target_i$  es el valor de los 360 experimentos para la red neuronal artificial y  $Output_i$  es la salida real de la red neuronal artificial.

2. Coeficiente de correlación. [42] Este mide la correlación entre los 360 experimentos de las líneas de producción ( $Target$ ) y los valores de salida de la red neuronal artificial ( $output$ ) definiéndose como:

$$R = \frac{cov(Target, Output)}{\sigma_{Target} \sigma_{Output}} \quad (4.3)$$
$$-1 \leq R \leq 1$$

Donde  $cov(Target, Output)$  son los valores de la red neuronal y  $\sigma_{Target}$ ,  $\sigma_{Output}$  son los valores de las desviaciones estándar de la  $Target$  y  $Output$ . El valor máximo de  $R = 1$ , esto indica que la relación lineal es positiva por lo contrario cuando  $R = 0$ , no existe una correlación lineal entre la producción y los valores objetivos.

### 4.2.3. Generación de la base datos de las redes neuronales artificiales

Una vez ejecutado el algoritmo guardara las redes neuronales generadas con extensión **.mat** la cual servirá para la simulación el procedimiento se da en las siguientes líneas de comando:

```
Prom = [Prom suma/iter];
ind_arch = ind_arch + 1;
nombre = ['/home/fede/Documentos/RNA_TR/R_TR/redOptimaT_']
num2str(ind_arch) '.mat'];
```

```
save(nombre, 'netOpt', 'p', 'trOpt', 'errorsOpt',  
'hiddenLayerSizeOpt', 'Prom');
```

Para evitar que se guarden redes que no son óptimas se ejecutan líneas en las cuales se discrimina estas redes y solo guarda la mejor para tener una red de las 50 iteraciones que realiza.

```
newRendimiento = performance;  
if newRendimiento < rendimiento  
    rendimiento = newRendimiento;  
    p = performance;  
    netOpt = net;  
    trOpt = tr;  
    errorsOpt = errors;  
    hiddenLayerSizeOpt = hiddenLayerSize;  
end
```

Una vez creada la base de datos con 780 redes, se realiza otro algoritmo de búsqueda.

#### 4.2.4. Algoritmo de búsqueda de la mejor red neuronal artificial

La secuencia de pasos de la búsqueda consiste en la comparación de todas las redes neuronales, para esta investigación se tomaron dos aspectos importantes que el error cuadrático medio (MSE) y el coeficiente de correlación R se compara todas contra todas realizando un barrido total, tomando los mejores indicadores para R el más cercano a 1 y para MSE tendiendo a 0.

```
function [arch, r2Max, rsmeMax, R2_RNA] = buscaRNA  
tic
```

```
for i=1:tam  
    load(Arch(i).nombre);  
    [r, sme] = validaRNA(netOpt);  
    R_RNA = [R_RNA; i, r, me];
```

```
    if r > rMax
        rMax = r;
        arch = Arch(i).nombre;
        rsmeMax = rsme;
        i
        r2
        view(netOpt);
    end
end
toc
end
```

#### 4.2.5. Simulación de la red neuronal artificial.

La simulación de la red neuronal artificial se realiza con el comando **sim(net,[K;N])** en donde **net** es la red neuronal artificial óptima guardada con extensión **.mat**, **K** el número de maquinas y **N** el tamaño de búfers, con este formato se llama la red con los mejores rendimientos entre las 780 redes.

```
sim(net,[ 3 ; 1 ])
```

Por medio de este comando se obtiene el valor **K=3, N=1**, de esta forma se generan todos los 360 experimentos y se observa el comportamiento de la red cuando se introduzcan valores fuera de estos experimentos.

### 4.3. Metodología de superficie de respuesta.

El objetivo es desarrollar la Metodología de Superficie de Respuesta, así como todas las técnicas que intervienen en el proceso para comparar los resultados de la red neuronal y determinar cual técnica se acopla al comportamiento de los datos. Para ello se han ilustrado dos problemas en los cuales se utiliza dicha metodología con el fin último de conseguir el comportamiento de la variable respuesta, es decir, la variable que recibe influencia de diversos factores de  $K$  máquinas y  $N$  búfers. Para conseguir dicha respuesta, para el diseño del ajuste de la superficie se tiene dos factores:  $K$  y  $N$  designados como  $x$  y  $y$ , respectivamente se empleara Matlab toolbox para el desarrollo numérico, en donde analizaremos una regresión múltiple de

la siguiente forma:

$$\begin{aligned} Th(x, y) = & \theta_0 + \theta_1x + \theta_2y + \theta_3x^2 + \theta_4xy + \theta_5y^2 \\ & + \theta_6x^3 + \theta_7x^2y + \theta_8xy^2 + \theta_9y^3 + \theta_{10}x^4 \\ & + \theta_{11}x^3y + \theta_{12}x^2y^2 + \theta_{13}xy^3 \end{aligned} \quad (4.4)$$

Donde:  $Th$  es la variable de respuesta;  $\theta_0$  es ;  $\theta_1, \dots, \theta_{13}$  are the linear coefficients an  $x, y$  are the coded independent variables.

Se han empleado un total de 360 experimentos de líneas de producción en este trabajo para evaluar los efectos de los dos parámetros independientes principales sobre la eficiencia de la línea de producción.

La ejecución del proceso se evaluó analizando la respuesta  $Th$ , que depende de los factores de entrada  $N, K$  y la relación entre la respuesta y los parámetros del proceso de entrada se describe mediante 4.4.

### 4.3.1. Ajuste de la superficie.

El enfoque de función de ajuste es una técnica para la determinación simultánea de ajustes óptimos de variables de entrada que pueden determinar niveles óptimos de rendimiento para una o más respuestas.

El procedimiento de ajuste de superficie implica los pasos para Matlab:

(1) determinar las variables  $x_1, x_2$  que para este caso en específico serán  $K$  y  $B$  respectivamente, estas variables se cargaran como vectores:

$K = [3:20]$

$B = [1:20]$

estos valores tendrán que ser discretos.

(2) **Función de respuesta.** la función de respuesta se puede determinar con una función polinomial. el éxito de la experimento consta, en que la superficie de respuesta se pueda ajustar al polinomio de orden  $n$ .

Para los valores de la superficie, se toman las 360 salidas de las líneas de producción y que para fines de Matlab se tomara como el eje z.

```
Th=[0.598771 0.657964 0.690055 0.725474 0.747229 0.770778 0.786484 0.803267  
0.815134 0.827698 0.836978 0.846735 0.854191 0.861987 0.868105 0.874479  
0.879591 0.884899 0.889234 0.893721 0.533761 0.567763 0.610984 0.641064  
0.662088 0.687469 0.70797 0.722243 0.7389 0.753749 0.764591 0.775829  
0.787071 0.795604 0.803621 0.812423 0.819311 0.825486 0.832367 0.838036  
0.486654 0.526182 0.550301 0.583023 0.606704 0.631371 0.646434 0.664948  
0.681167 0.697935 0.709031 0.720369 0.731851 0.743976 0.752476 0.761146  
0.768638 0.777764 0.784476 0.791315 0.459349 0.487276 0.521246 0.539542  
0.564783 0.586611 0.604203 0.6251 0.638171 0.650346 0.665323 0.677436  
0.69146 0.701212 0.709759 0.71922 0.727641 0.737682 0.745217 0.751711  
0.520847]
```

Una vez cargados los datos el software realizara el ajuste polinomial arrojando los errores y la función de respuesta, siendo esta el foco de la metodología de respuesta.

---

# Capítulo 5

## Resultados y discusiones.

### 5.1. Primer experimento

#### 5.1.1. Resultados de la red neuronal

Basados en la estructura de la red neuronal artificial, se realizó la simulación de los 360 experimentos. Las etapas del diseño se muestran en la siguiente tabla 5.1, en donde se observa su tiempo de procesamiento, todos los experimentos se ejecutaron en una computadora con las siguientes especificaciones: Intel(R) Core(TM) i5-2450M CPU @ 2.50 GHz, 4.00 GB RAM.

Tabla 5.1: Tiempos computacionales

Algoritmos	Tiempo en segundos
<b>redbap</b> : para generar las 780 redes neuronales artificiales	66993.040
<b>buscar_red_opt</b> : para encontrar el mejor MSE y R	13635.300
Simulación de los 360 experimentos de las líneas de producción	41.315
Total	80669.655 22 hrs. con 41 min.

En comparación con los tiempos obtenidos con [80] no se puede comparar puntualmente ya que el equipo en el que se realizaron sus experimentos es una máquina con un procesador Pentium Dual Core CPU E5300 @ 2.6 Ghz sin embargo, nos puede dar un referente la tabla 5.2 de resultados, en donde:

Para realizar los 360 experimentos con un algoritmo de simulado recocido y ocupa 31958

Tabla 5.2: Referencia de tiempos del primer experimento.

algoritmo	K	N'	Número de experimentos	Tiempo total de CPU
Simulado recocido	3 a 20	1 a 10	$18 * 20 = 360$	9 h (31958 seg)
ann	3 a 20	1 a 10	$18 * 20 = 360$	22 h con 41 min (80669.655 seg)
msr	3 a 20	1 a 10	$18 * 20 = 360$	2.123 seg

segundos aproximadamente 9 horas con referente a las 22 horas con 41 minutos aproximados de la red sin embargo una ventaja que tiene la ANN, una vez generada la red, los tiempos para calcular la tasa de producción de una línea de manufactura, solo es de 0.116 segundos y el algoritmo (SA), el tiempo de calculo depende de lo complejo del problema, es decir entre más grande sea la línea de producción más tiempo tardará en dar una respuesta.

El tiempo total para obtener la red es de 22 horas con 41 minutos aproximadamente. La red neuronal artificial óptima encontrada se compone de un arreglo de 4 capas, la primer capa se integra con 8 neuronas, la segunda con 8 neuronas, en su tercer capa con 10 neuronas y finalmente en la cuarta capa con 10 neuronas. Esta red se representa en la Figura 5.1.

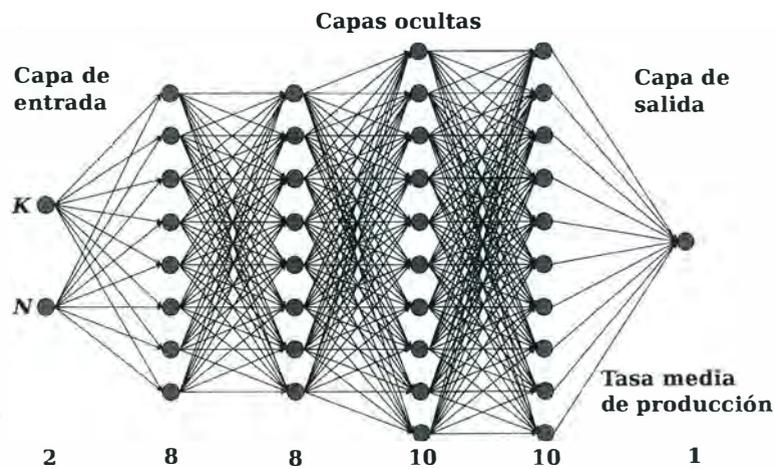


Figura 5.1: Red neuronal artificial óptima

Esta red neuronal artificial fue empleada para la simulación de los 360 experimentos. En la tabla 5.2 se observan algunos resultados, comparados con la tasa de producción de [80], calculando su error absoluto y el error relativo, los resultados completos se pueden ver en el Apéndice B.

Tabla 5.3: Resultados parciales de la simulación de los 360 experimentos.

ID	K	N	Tasa de producción de los experimentos [80]	Tasa de producción de la RNA	Error absoluto	Error relativo (%)
1	3	1	0.598771	0.59878482	1.38197E-05	0.002308015
2	3	2	0.657964	0.657972081	8.08117E-06	0.001228209
3	3	4	0.725474	0.725445945	2.80551E-05	0.003867145
4	3	6	0.770778	0.770290869	0.000487131	0.063199923
5	3	8	0.803267	0.802427272	0.000839728	0.104539147
6	3	10	0.827698	0.827419157	0.000278843	0.033689003
7	3	12	0.846735	0.846516781	0.000218219	0.025771832
8	3	14	0.861987	0.861723751	0.000263249	0.03053983
9	3	16	0.874479	0.874146662	0.000332338	0.038004135
10	3	18	0.884899	0.884603111	0.000295889	0.03343762
11	7	1	0.437107	0.437514721	0.000407721	0.093277072
12	7	2	0.465856	0.465797107	5.88926E-05	0.012641795
13	7	4	0.51778	0.516707804	0.001072196	0.207075628
14	7	6	0.552118	0.552068326	4.96739E-05	0.008996979
15	7	8	0.588898	0.588773563	0.000124437	0.021130557
16	7	10	0.620724	0.620768787	4.47874E-05	0.007215345
17	7	12	0.642505	0.642842497	0.000337497	0.052528275
18	7	14	0.665562	0.664687128	0.000874872	0.131448564
19	7	16	0.68696	0.686645603	0.000314397	0.045766462
20	7	18	0.703324	0.702991821	0.000332179	0.047229873
21	11	1	0.392788	0.393039778	0.000251778	0.06410027
22	11	2	0.410961	0.411512069	0.000551069	0.134092686
23	11	4	0.445664	0.445133688	0.000530312	0.118993682
24	11	6	0.478039	0.477766156	0.000272844	0.057075662
25	11	8	0.510407	0.510439498	3.24977E-05	0.00636702
26	11	10	0.530022	0.529755714	0.000266286	0.050240601
27	11	12	0.550636	0.550814476	0.000178476	0.032412618
28	11	14	0.572147	0.572217978	7.09781E-05	0.012405575
29	11	16	0.593234	0.593261571	2.75707E-05	0.004647528
30	11	18	0.611552	0.611526068	2.59318E-05	0.004240324

Tabla 5.4: Continuación de resultados parciales de la simulación de los 360 experimentos.

ID	K	N	Tasa de producción de los experimentos [80]	Tasa de producción de la RNA	Error absoluto	Error relativo (%)
31	11	20	0.625781	0.625796483	1.54826E-05	0.002474122
32	15	1	0.373106	0.373221754	0.000115754	0.031024444
33	15	2	0.387197	0.386906302	0.000290698	0.075077566
34	15	4	0.412171	0.412824504	0.000653504	0.158551653
35	15	6	0.436688	0.436995946	0.000307946	0.070518605
36	15	8	0.460443	0.460421618	2.13819E-05	0.004643776
37	15	10	0.482753	0.48274955	3.44956E-06	0.000714561
38	15	12	0.507087	0.507046462	4.05384E-05	0.007994359
39	15	14	0.521423	0.521415805	7.1952E-06	0.001379915
40	15	16	0.535417	0.535389245	2.77552E-05	0.005183848
41	15	18	0.550121	0.550477153	0.000356153	0.064740842
42	15	20	0.565731	0.565518818	0.000212182	0.037505742
43	19	4	0.393936	0.394082372	0.000146372	0.037156296
44	19	8	0.432267	0.43223376	3.32399E-05	0.00768966
45	19	12	0.468468	0.468490775	2.27752E-05	0.004861637
46	19	16	0.505224	0.50517011	5.38902E-05	0.010666603
47	19	19	0.522036	0.522063077	2.70771E-05	0.005186832
48	20	5	0.399858	0.399885131	2.71306E-05	0.006785057
49	20	10	0.444622	0.444093684	0.000528316	0.118823681
50	20	20	0.520847	0.520838909	8.09087E-06	0.001553406

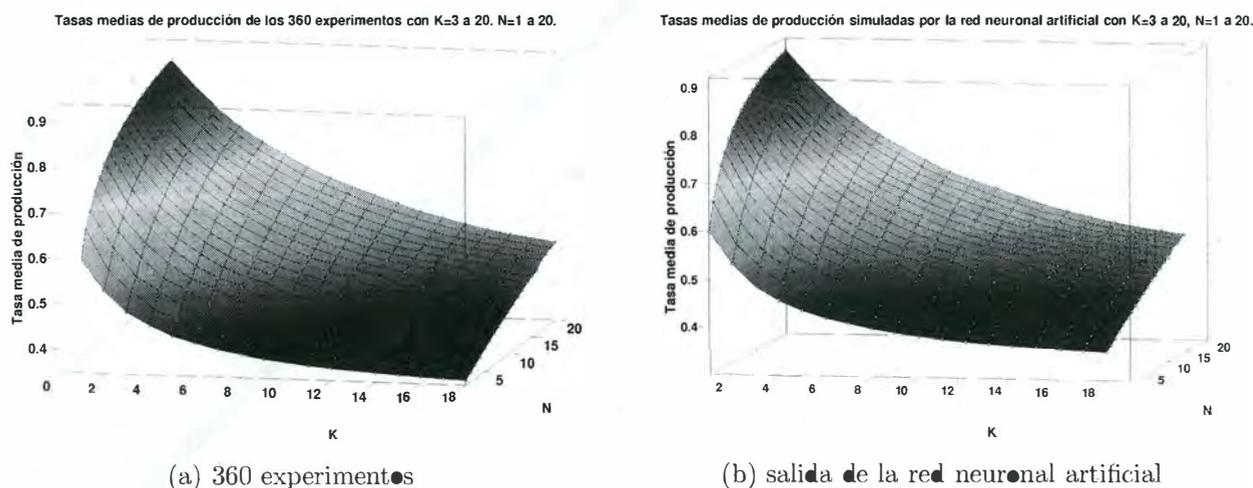


Figura 5.2: Comparación entre los 360 experimentos de [80] contra la salida de la red neuronal artificial.

Los parámetros para la validación de la red se dan por la correlación entre las entradas contra las salidas y el error cuadrático medio, para esta red neuronal artificial la correlación es de  $R=1$  y el  $MSE=1.150E-07$  por lo que son buenos resultados. En la Figura 5.2 se observa la gráfica.

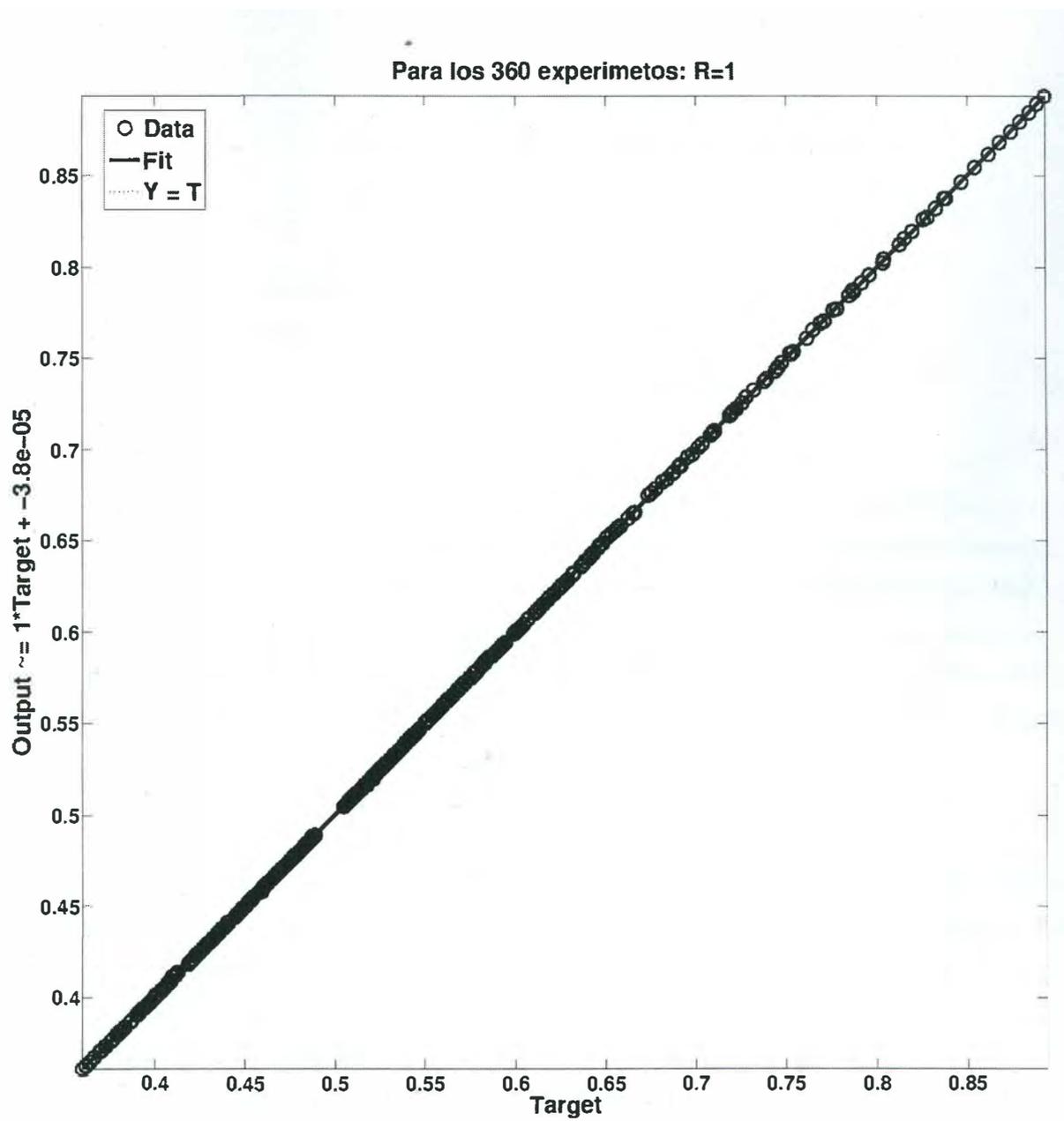


Figura 5.3: Regresión para los 370 experimentos.

### 5.1.2. Resultados de la metodología de superficie de respuesta.

Los resultados del modelo de la metodología de superficie de respuesta se da en la siguiente ecuación:

$$\begin{aligned}
 Th(x, y) = & 0,8661 - 0,1342x + 0,04927y \\
 & + 0,01331x^2 - 0,001985xy - 0,002671y^2 \\
 & - 0,0005775x^3 - 8,568E - 05x^2y + 0,0002081xy^2 \\
 & + 4,734E - 05y^3 + 9,157E - 06x^4 + 4,064E - 06x^3y \\
 & - 2,961E - 06x^2y^2 - 3,058E - 06xy^3
 \end{aligned} \tag{5.1}$$

Donde  $Th$  es la tasa de producción,  $x$  es K máquinas,  $y$  es N búfer. La gráfica de la superficie se muestra en la Figura 5.4, la predicción del modelo tiene una correlación entre las tasa de producción de [80] se muestra en la Tabla 5.5 en donde se calcula el error relativo y absoluto. Se tiene un error máximo absoluto de 0.017151311 con respecto a los valores de [80] y un mínimo de 3.14314095972712E-07 para el error máximo relativo es de 2.36415240242931 y el mínimo es de 0.00061246999287, por lo que se puede observar que la función 5.1 representa de una buena manera la superficie de tasas de producción.

La predicción del modelo ha sido validado por medio de una correlación entre los valores generados por [80] y los generado por el polinomio (5.1) mostrado en la Figura 5.4. Estos parámetros de validación de la MSR se dan por la correlación entre las entradas contra las salidas que se obtuvo así como el error medio cuadrático dando como resultados los siguientes: la correlación es de 0.999603176045516 esta se observa en la Figura 5.4 y un MSE= 1.22796058122841E-05 por lo que se puede observar que su comportamiento es aceptable.

En cuanto a la gráfica de residuales en donde se puede observar las zonas en donde la aproximación del polinomio (5.1) se muestra en la Figura 5.5 en donde se observa que la principal zona de errores es al inicio y final de la gráfica 5.5 en la parte Izquierda y deerecha es decir en los extremos.

Tabla 5.5: Resultados parciales de la MSR de los 360 experimentos.

ID	K	N	Tasa de producción de [80]	Tasa de producción de MSR	Error absoluto	Error relativo (%)
1	3	1	0.598771	0.609057642	0.010286642	1.71795928660537
2	3	2	0.657964	0.645758365	0.012205635	1.85506121915484
3	3	4	0.725474	0.708322689	0.017151311	2.36415240242928
4	3	6	0.770778	0.757964157	0.012813843	1.66245572655161
5	3	8	0.803267	0.796514737	0.006752263	0.840600074445979
6	3	10	0.827698	0.825806397	0.001891603	0.228537824182252
7	3	12	0.846735	0.847671105	0.000936105	0.110554659958556
8	3	14	0.861987	0.863940829	0.001953829	0.226665715376226
9	3	16	0.874479	0.876447537	0.001968537	0.225109693886302
10	3	18	0.884899	0.887023197	0.002124197	0.240049655384399
11	7	1	0.437107	0.414981492	0.022125508	5.06180591937444
12	7	2	0.465856	0.44154176	0.02431424	5.21926088748455
13	7	4	0.51778	0.488448864	0.029331136	5.66478736142763
14	7	6	0.552118	0.527986432	0.024131568	4.37072654758583
15	7	8	0.588898	0.561252512	0.027645488	4.69444419916522
16	7	10	0.620724	0.589345152	0.031378848	5.05520134552557
17	7	12	0.642505	0.6133624	0.0291426	4.53577793168923
18	7	14	0.665562	0.634402304	0.031159696	4.68171199677866
19	7	16	0.68696	0.653562912	0.033397088	4.86157680214275
20	7	18	0.703324	0.671942272	0.031381728	4.46191627187471
21	11	1	0.392788	0.387575562	0.005212438	1.32703595832866
22	11	2	0.410961	0.407927837	0.003033163	0.738065899197249
23	11	4	0.445664	0.444766785	0.000897215	0.201320950312332
24	11	6	0.478039	0.476999677	0.001039323	0.217413851171146
25	11	8	0.510407	0.505284209	0.005122791	1.00366785721985
26	11	10	0.530022	0.530278077	0.000256077	0.04831440959056
27	11	12	0.550636	0.552638977	0.002002977	0.363757000995226
28	11	14	0.572147	0.573024605	0.000877605	0.153388027901921
29	11	16	0.593234	0.592092657	0.001141343	0.192393389455091
30	11	18	0.611552	0.610500829	0.001051171	0.17188579221392
31	11	20	0.625781	0.628906817	0.003125817	0.49950653663183
32	15	1	0.373106	0.37607937	0.00297337	0.796923662444446
33	15	2	0.387197	0.389375485	0.002178485	0.562629617481542
34	15	4	0.412171	0.414735105	0.002564105	0.622097381911886
35	15	6	0.436688	0.438510045	0.001822045	0.417241829406806
36	15	8	0.460443	0.460770865	0.000327865	0.071206425116677
37	15	10	0.482753	0.481588125	0.001164875	0.241298345116449
38	15	12	0.507087	0.501032385	0.006054615	1.19399925456579
39	15	14	0.521423	0.519174205	0.002248795	0.431280361625797
40	15	16	0.535417	0.536084145	0.000667145	0.124602879624663
41	15	18	0.550121	0.551832765	0.001711765	0.311161544460204
42	15	20	0.565731	0.566490625	0.000759625	0.134273179302518
43	19	4	0.393936	0.390219777	0.003716222	0.94335704276812
44	19	8	0.432267	0.429667377	0.002599623	0.601392889117142
45	19	12	0.468468	0.467697089	0.000770911	0.164560012636933
46	19	16	0.505224	0.500176305	0.005047695	0.999100399030929
47	19	19	0.522036	0.518407182	0.003628818	0.695127922212268
48	20	5	0.399858	0.3962075	0.0036505	0.91294909692941
49	20	10	0.444622	0.44616	0.001538	0.34591180823262
50	20	20	0.520847	0.5195	0.001347	0.258617213884305

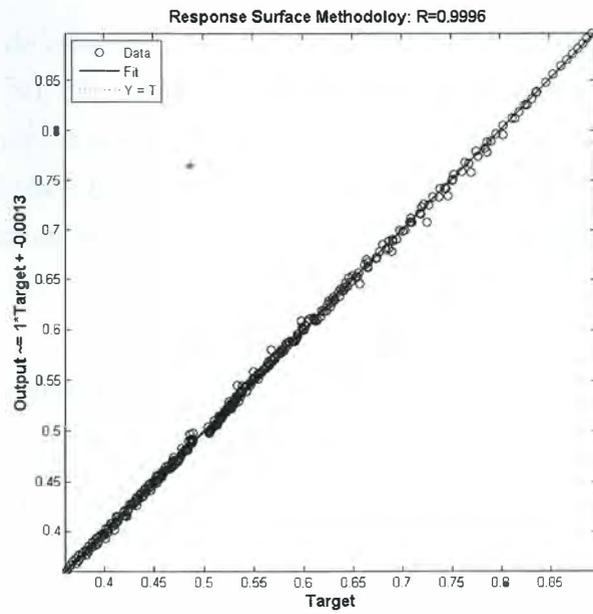


Figura 5.4: Gráfico de correlación de la MSR en el primer experimento

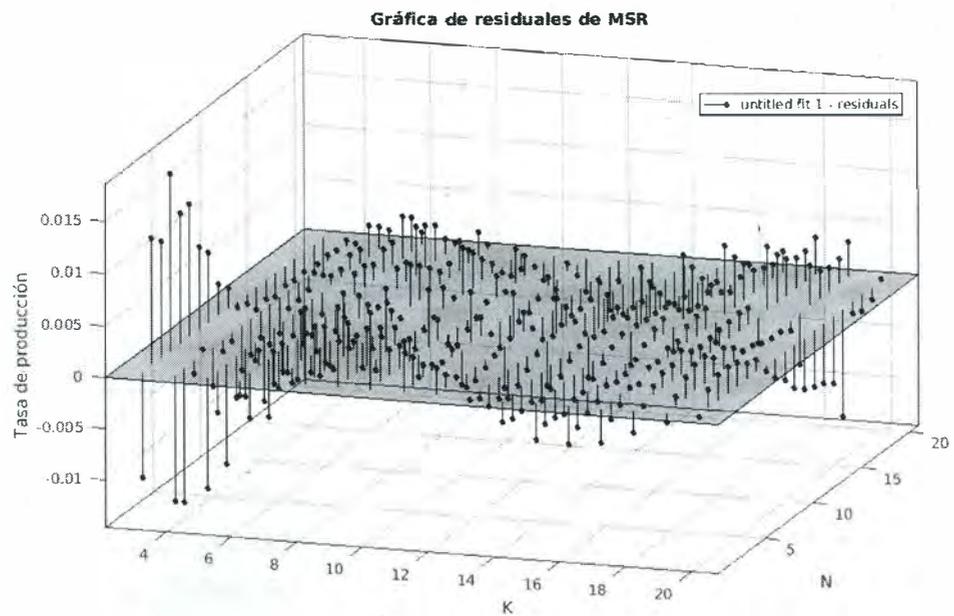


Figura 5.5: Gráfica de residuales de la MSR

## 5.2. Segundo experimento.

Dados los resultados de esta red neuronal artificial, se decidió realizar una segunda prueba con 55 experimentos de [80, 81], líneas de producción que no se encuentran dentro de los primeros 360, estos experimentos se clasificaron de acuerdo al tamaño de las líneas de producción como se observa en la tabla 5.6. Se tomó una línea pequeña, ya que la mayoría de las líneas de la primera prueba entran en este rango, para las líneas medianas se tienen 40 y para las líneas grandes se tienen 14.

Para el segundo experimento, se utilizó la misma red neuronal artificial con la configuración de 8:8:10:10 neuronas en 4 capas, de igual manera la misma función generada por la Metodología de superficie de respuesta declarada en la ec 5.1 y se toman los tiempos de referencia de la Tabla 5.6, como se observa, las líneas a partir de la tercera a la 16, los tiempo del algoritmo de SA son muy buenos, pero la ventaja de la red es que una vez que se obtiene la red ANN, el tiempo de respuesta es de 0.116 segundos, sin embargo no se esta tomando el tiempo de generación y búsqueda de 66993.04 y 13635.3 respectivamente que en suma son 80628.34 segundos, por lo que se tiene que la red neuronal artificial tiene desventajas de tiempo para líneas pequeñas y algunas medianas, pero tiene una gran ventaja para líneas consideradas como grandes y más complejas ya que se observa los tiempo de las líneas 1 y 2 del algoritmo de SA y en suma son 96119 segundos un tiempo considerable, por lo que la red neuronal artificial una vez generada calcularía la tasa de producción con muy buena precisión sin importar el tamaño de la línea de producción.

Para esta segunda prueba los resultados son alentadores con una correlación de  $R=0.99875$  y un error cuadrático medio de  $MSE=3.57E-05$  en la tabla 5.3 se calcula el error absoluto y el error relativo, siendo el máximo de 0.013927 y 2.76, respectivamente, con  $K=100$  y  $N=101$  para la red neuronal artificial.

Estos resultados demuestran que la red neuronal artificial tiene la capacidad de predecir valores fuera de rango, de los primeros 360 experimentos con  $K=3$  a 20 y  $N=1$  a 20, en un tiempo más corto con 6.416 segundos y una correlación bastante aceptable  $R=0.99875$  esto se puede observar en la Figura 5.3.

En la comparación gráfica de los 55 experimentos contra los que arroja la red neuronal artificial se tiene una similitud aceptable para las primeras 23 líneas de producción clasificadas como medianas, esto se puede observar en la Figura 5.4, de tal forma que a partir de los experimentos 48 al 52 con  $K=100$ , 40, 50, 60, 70 y  $N=101$ , 80, 100, 120, 140, con sus

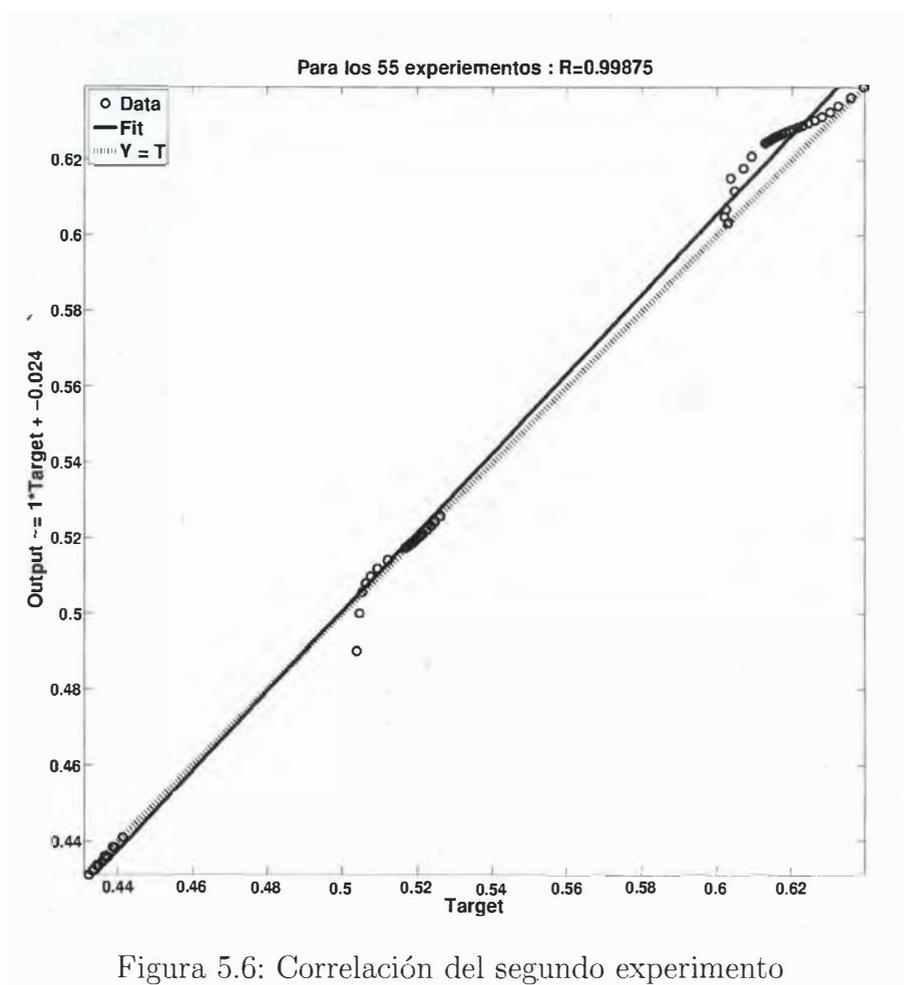


Figura 5.6: Correlación del segundo experimento

Tabla 5.6: Referencia de tiempos de [80] con la red neuronal artificial.

No.	K	N	Throughput		CPU Tiempo (en segundos)	
			SA	ANN	SA	ANN
1	100	101	0.504117	0.49019	36002	0.116
2	100	200	0.60302	0.60321	60117	0.116
3	10	11	0.557569	0.557449	9	0.116
4	10	20	0.643031	0.64311	25	0.116
5	10	5	0.472661	0.475901	10	0.116
6	11	12	0.550636	0.550814	23	0.116
7	11	22	0.638915	0.639236	39	0.116
8	11	5	0.461879	0.462024	14	0.116
9	12	13	0.545856	0.545527	21	0.116
10	12	24	0.635557	0.636344	39	0.116
11	12	6	0.465294	0.465218	27	0.116
12	13	14	0.541295	0.541408	42	0.116
13	13	26	0.632201	0.634189	99	0.116
14	13	6	0.45413	0.45418	27	0.116
15	14	15	0.538093	0.53812	131	0.116
16	14	28	0.630049	0.632574	75	0.116
			Suma de tiempo		96700	1.856
			Tiempo	Generación de ANN	-	66993.04
			Tiempo	Búsqueda de la mejor ANN	-	13635.3
			Tiempo	Total	96700	80630

correspondientes, el error es significativo, pero nos brinda un resultado aceptable para tomar una decisión con un fundamento aceptable. Sin embargo, la tendencia del error de estos datos va decreciendo, se observa que en los últimos 3 experimentos, el error disminuye tomando en cuenta el tamaño de las líneas de producción con  $K=80, 90$  y  $100$ ,  $N=160, 180$  y  $200$ , respectivamente, siendo el experimento 55 con un error relativo del  $0.03\%$  y un error absoluto de  $0.000199$ .

Tabla 5.7: Errores de la red neuronal artificial desarrollados para los 55 datos.

ID	Tamaño de la línea	K	N	Tasa de producción de los 55 experimentos [80, 81]	Tasa de producción de la red neuronal artificial	Error absoluto	Error relativo en %
1	Pequeña	11	22	0.63892	0.639236	0.000316	0.05
2	Mediana	21	10	0.43869	0.438564	0.000126	0.03
3	Mediana	22	11	0.44135	0.4412	0.00015	0.03
4	Mediana	23	11	0.43656	0.436088	0.000472	0.11
5	Mediana	24	12	0.43927	0.438463	0.000807	0.18
6	Mediana	25	12	0.43494	0.433669	0.001271	0.29
7	Mediana	26	13	0.43739	0.435921	0.001469	0.34
8	Mediana	27	13	0.43345	0.432226	0.001224	0.28
9	Mediana	28	14	0.43586	0.434505	0.001355	0.31
10	Mediana	29	14	0.43234	0.431178	0.001162	0.27
11	Mediana	30	15	0.43447	0.433623	0.000847	0.19
12	Mediana	20	21	0.52608	0.5259	0.00018	0.03
13	Mediana	21	22	0.52465	0.524538	0.000112	0.02
14	Mediana	22	23	0.52361	0.523318	0.000292	0.06
15	Mediana	23	24	0.52247	0.522231	0.000239	0.05
16	Mediana	24	25	0.52146	0.521265	0.000195	0.04
17	Mediana	25	26	0.52051	0.52041	0.0001	0.02
18	Mediana	26	27	0.51962	0.519653	0.000033	0.01
19	Mediana	27	28	0.51892	0.518983	0.000063	0.01
20	Mediana	28	29	0.51815	0.518388	0.000238	0.05
21	Mediana	29	30	0.51747	0.517859	0.000389	0.08
22	Mediana	30	31	0.5168	0.517385	0.000585	0.11
23	Mediana	12	24	0.63556	0.636344	0.000784	0.12
24	Mediana	13	26	0.6322	0.634189	0.001989	0.31
25	Mediana	14	28	0.63005	0.632574	0.002524	0.4
26	Mediana	15	30	0.62789	0.631348	0.003458	0.55
27	Mediana	16	32	0.626	0.630397	0.004397	0.7
28	Mediana	17	34	0.62428	0.629639	0.005359	0.86
29	Mediana	18	36	0.62277	0.629014	0.006244	1
30	Mediana	19	38	0.62159	0.628481	0.006891	1.11
31	Mediana	20	40	0.62033	0.628011	0.007681	1.24
32	Mediana	21	42	0.61926	0.627583	0.008323	1.34
33	Mediana	22	44	0.6183	0.627184	0.008884	1.44
34	Mediana	23	46	0.61743	0.626804	0.009374	1.52
35	Mediana	24	48	0.61666	0.626437	0.009777	1.59
36	Mediana	25	50	0.61583	0.626078	0.010248	1.66
37	Mediana	26	52	0.61517	0.625725	0.010555	1.72
38	Mediana	27	54	0.61458	0.625376	0.010796	1.76
39	Mediana	28	56	0.61399	0.625029	0.011039	1.8
40	Mediana	29	58	0.61345	0.624684	0.011234	1.83
41	Mediana	30	60	0.61288	0.624339	0.011459	1.87
42	Grande	40	41	0.51223	0.51429	0.00206	0.4
43	Grande	50	51	0.50949	0.512086	0.002596	0.51
44	Grande	60	61	0.50768	0.510035	0.002355	0.46
45	Grande	70	71	0.50639	0.508186	0.001796	0.35
46	Grande	80	81	0.50544	0.505767	0.000327	0.06
47	Grande	90	91	0.5047	0.500121	0.004579	0.91
48	Grande	100	101	0.50412	0.490193	0.013927	2.76
49	Grande	40	80	0.60928	0.620941	0.011661	1.91
50	Grande	50	100	0.60714	0.617804	0.010664	1.76
51	Grande	60	120	0.6037	0.615156	0.011456	1.9
52	Grande	70	140	0.60475	0.611799	0.007049	1.17
53	Grande	80	160	0.60244	0.606839	0.004399	0.73
54	Grande	90	180	0.60205	0.604855	0.002805	0.47
55	Grande	100	200	0.60302	0.603219	0.000199	0.03

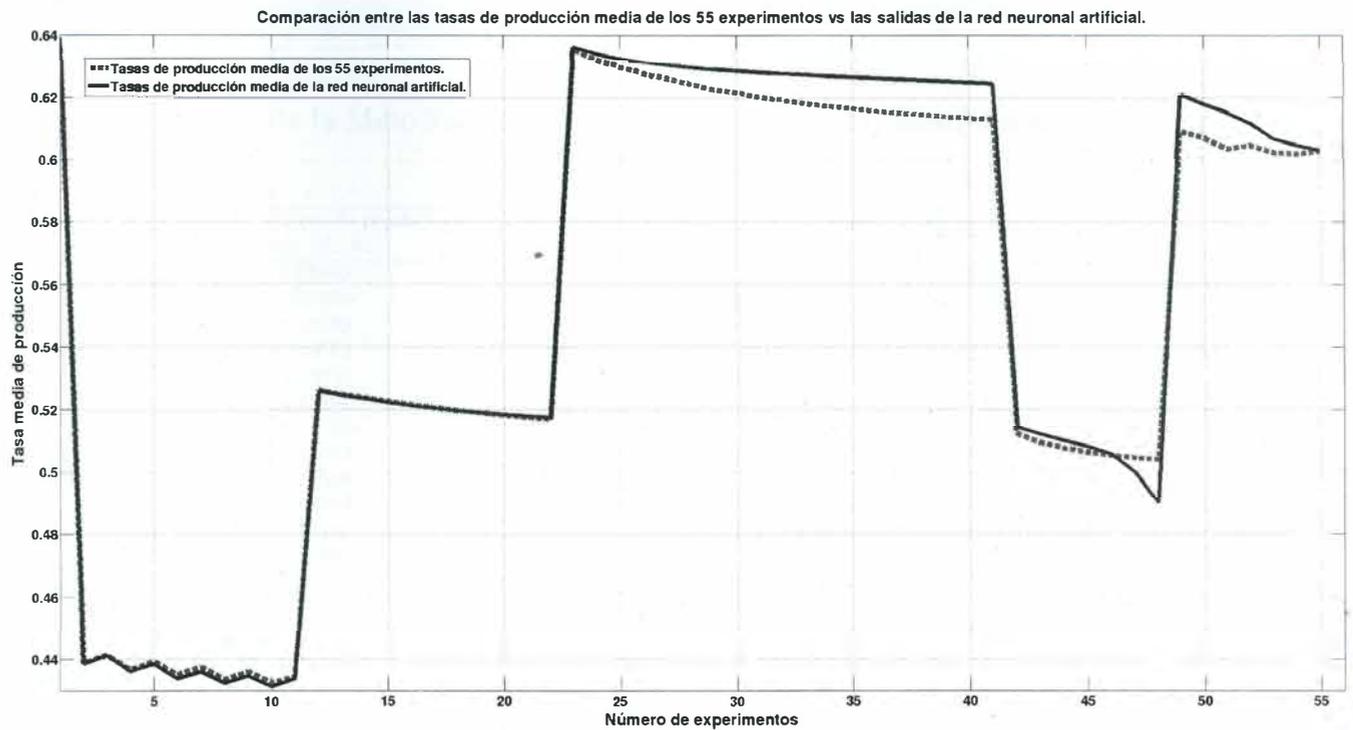


Figura 5.7: Comparación de tasa de producción

Para los resultados con la Metodología de superficie de respuesta no son muy alentadores se tiene una correlación de  $-0.22536096452846$  y un error cuadrático medio de  $66032.8992607818$  los resultados se muestran en la Tabla 5.8 en donde se observa los errores absolutos y relativos en donde se muestra que la MSR no predice valores fuera con los que se generó el polinomio 5.1.

Tabla 5.8: Errores de la Metodología de superficie de respuesta desarrollados para el segundo experimento

ID	Tasa de producción de [80, 81]	Tasa de producción de RSM	Error relativo	Error absoluto
1	0.63892	0.64797	1.41619	0.00905
2	0.43869	0.44707	1.90913	0.00838
3	0.44135	0.46388	5.10444	0.02253
4	0.43656	0.47747	9.37187	0.04091
5	0.43927	0.51204	16.56523	0.07277
6	0.43494	0.54625	25.59193	0.11131
7	0.43739	0.60809	39.02729	0.17070
8	0.43345	0.67281	55.22257	0.23936
9	0.43586	0.77337	77.43512	0.33751
10	0.43234	0.88045	103.64699	0.44811
11	0.43447	1.03306	137.77418	0.59859
12	0.52608	0.52245	0.68933	0.00363
13	0.52465	0.52449	0.02956	0.00016
14	0.52361	0.53013	1.24466	0.00652
15	0.52247	0.54055	3.46125	0.01808
16	0.52146	0.55715	6.84458	0.03569
17	0.52051	0.58147	11.71155	0.06096
18	0.51962	0.61523	18.40003	0.09561
19	0.51892	0.66033	27.25046	0.14141
20	0.51815	0.71883	38.73032	0.20068
21	0.51747	0.79298	53.24177	0.27551
22	0.51680	0.88519	71.28264	0.36839
23	0.63556	0.64897	2.10969	0.01341
24	0.63220	0.64765	2.44437	0.01545
25	0.63005	0.64109	1.75222	0.01104
26	0.62789	0.62589	0.31883	0.00200
27	0.62600	0.59820	4.44063	0.02780
28	0.62428	0.55373	11.30144	0.07055
29	0.62277	0.48771	21.68764	0.13506
30	0.62159	0.39492	36.46601	0.22667
31	0.62033	0.26970	56.52314	0.35063
32	0.61926	0.10591	82.89672	0.51335
33	0.61830	-0.10302	116.66237	0.72132
34	0.61743	-0.36415	158.97898	0.98158
35	0.61666	-0.68498	211.07849	1.30164
36	0.61583	-1.07345	274.30896	1.68928
37	0.61517	-1.53798	350.00840	2.15315
38	0.61458	-2.08743	439.65221	2.70201
39	0.61399	-2.73115	544.81923	3.34514
40	0.61345	-3.47889	667.10267	4.09234
41	0.61288	-4.34091	808.28058	4.95379
42	0.51223	3.45511	574.52293	2.94288
43	0.50949	11.76926	2210.00843	11.25977
44	0.50768	31.09318	6024.56194	30.58550
45	0.50639	68.42086	13411.49493	67.91447
46	0.50544	132.47480	26109.79727	131.96936
47	0.50470	233.70596	46205.91726	233.20126
48	0.50412	384.29380	76130.62020	383.78968
49	0.60928	-21.82494	3682.08705	22.43422
50	0.60714	-65.38815	10869.86362	65.99529
51	0.60370	-152.80182	25410.88620	153.40552
52	0.60475	-306.40275	50766.01902	307.00750
53	0.60244	-553.09326	91908.85399	553.69570
54	0.60205	-924.34119	153632.29632	924.94324
55	0.60302	-1456.17990	241581.19465	1456.78292

---

## Capítulo 6

### Conclusiones

El problema de asignación de búfers tiene características multiobjetivas que implican criterios monetarios, expresados en una forma de costo-beneficio, por otra parte se requiere de una gran cantidad de tiempo computacional, sobre todo para las líneas de producción medianas y grandes, estas últimas en particular aumenta considerablemente, cuando las estaciones de trabajo  $\mathbf{K}$  y el tamaño de búfer  $\mathbf{N}$  se incrementa. Sin embargo la situación ideal para un diseñador sería contar con una herramienta eficiente en donde se ingrese las entradas  $\mathbf{K}$  y  $\mathbf{N}$ , obteniendo el resultado, de la tasa promedio de producción. Por ello en este trabajo se propuso una estructura para la generación de una red neuronal artificial óptima, con la cual se obtenga los resultados eficientemente en un tiempo relativamente rápido.

Se desarrolló la Metodología de superficie de respuesta con las 360 tasas de producción de [80], en donde se tuvo un rendimiento de  $MSE= 1.22796E-05$  y una correlación de 0.999603176 en comparación a la red neuronal artificial, el polinomio (5.1) tiene un buen rendimiento en cuanto a precisión y tiempo de ejecución fue de 2.123 segundos con respecto a las 22 horas con 41 minutos de la red neuronal artificial, sin embargo en lo que respecta para el experimento 2 el polinomio no cumple con las expectativas arrojando resultados erróneos. estos resultados se muestran en la Tabla 6.1.

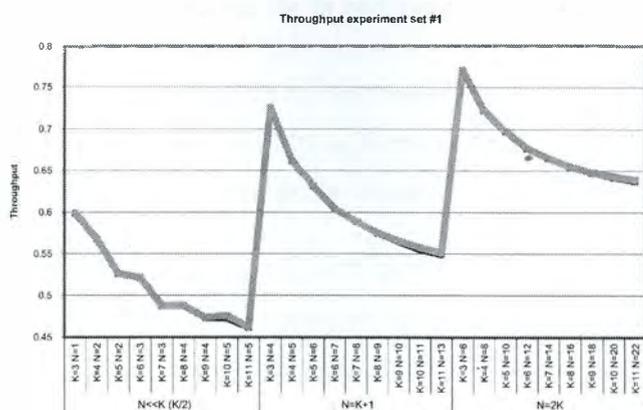
Tabla 6.1: Comparación de métricas entre la red neuronal artificial y la metodología de superficie de respuesta

Métricas	RNA	MSR
Para el primer experimento		
Error cuadrático medio	1.15E-07	1.23E-05
R	1	0.999603176
Error máximo	0.34 %	2.36 %
Para el segundo experimento		
Error cuadrático medio	3.57E-05	66032.89926
R	0.99875	-0.225360965
Error máximo	2.70 %	241581.19 %

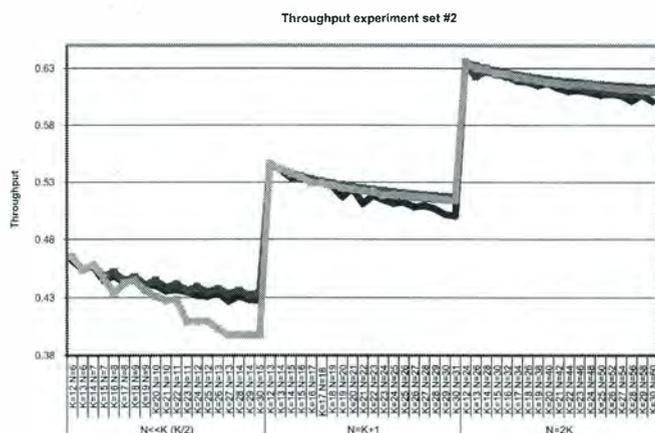
La red neuronal artificial óptima se desarrolló a partir de las combinaciones de 5, 8, 10, 12 y 15 neuronas, generando 780 redes, siendo la red con un arreglo de 4 capas ocultas y su correspondiente en cada capa de 8, 8, 10, 10 neuronas para la red neuronal artificial, ofreciendo soluciones para las líneas de producción exponenciales con una media  $\mu = 1$  y un número de estaciones  $\mathbf{K}=3$  a 20,  $\mathbf{N}=1$  a 20, obteniendo un error relativo de 0.34 %, un error medio cuadrático de 1.150E-07 y una correlación de  $R=1$ .

Un fundamento principal, de los buenos resultados de la red neuronal artificial se debe a la forma peculiar de escalón y al conjunto de datos para el entrenamiento, esto se puede observar en la Figura 6.1. Una de las características principales es al inicio y al final de las gráficas, en donde se observa una gran variabilidad y en comparación de la gráfica (c) la cual es la generada por la red neuronal para el segundo experimento, en donde se observa estos elementos, se repiten en bloque y al tener esta forma, la red neuronal se acopla con una gran precisión, lo que es más complicado para un algoritmo de recocido simulado. El acierto de la red neuronal artificial, es el entrenamiento y los patrones que siguen los datos, ya que se refleja en las primeras gráficas (a, b, y c) la tendencia a ser repetitivos en bloques, por otro lado la metodología que se empleó, dio una particular eficiencia al discriminar otras redes que tenían, niveles más bajos en sus métricas de validación, lo que da como resultado una eficiente red neuronal artificial que calcula una tasa de producción en 0.116 segundos, dando una ventaja para las líneas consideradas como medianas y grandes.

La contribución de la red neuronal artificial propuesta, se basa en que puede predecir con bastante precisión, en una forma rápida, las líneas de producción con estaciones  $\mathbf{K}=3$  a 100,  $\mathbf{N}=1$  a 200, con un tiempo promedio de 0.116 segundos por línea de producción.



(a)



proporcionando una flexibilidad mayor a un diseñador de líneas de producción. Se resaltan las siguientes ventajas de la red neuronal artificial:

- En referencia a los tiempos de la red neuronal, no se recomienda para líneas de producción pequeñas, debido al tiempo de la metodología empleada, sin embargo si se trata de un conjunto grande de líneas de producción se podría considerar.
- La red neuronal artificial es una opción ideal para el cálculo de las líneas medianas a grandes por el tiempo de cómputo que requieren otros algoritmos y el conjunto de líneas que se desea calcular, y otro valor agregado es que una vez generada la red neuronal solo se introduce el número de máquinas y búfers y dará la respuesta 0.116 segundos sin importar el tamaño de la línea de producción.
- La metodología para generar la red neuronal artificial nos da pauta para desechar todas las redes neuronales que no den certeza en el resultado y asegurar que la red que se elija tiene los mejores resultados en los indicadores, por lo que se recomienda esta metodología para otros problemas con las mismas características.
- La metodología de superficie de respuesta no es la herramienta adecuada para el resolver el BAP-1, se podría analizar el polinomio y describir el comportamiento del problema.
- El resultado para obtener una tasa media de producción de una línea en un formato estándar es en mili segundos.
- El diseñador no tiene que entender el diseño propio de la red para obtener los resultados óptimos, solo debe de introducir los valores de  $\mathbf{K}$  y  $\mathbf{N}$ .
- Un análisis de sensibilidad es más rápido por la generación de las soluciones y el amplio margen del número de estaciones  $\mathbf{K}$  y los tamaños de búfers  $\mathbf{N}$ .

## 6.1. Trabajos futuros

Basado en las observaciones de la tesis, para las investigaciones futuras será interesante el comportamiento de la red neuronal artificial, con la incorporación de más datos para el entrenamiento y seguir la misma metodología en cuanto a la generación de las redes neuronales artificiales, con las mismas combinaciones, y la discriminación de las redes menos efectivas y por consiguiente generar los valores de las tasas de rendimiento comparando con los datos existentes en la literatura y observar los errores máximos relativos y absolutos, la correlación

y el error cuadrático medio, para comprobar mediante la gráfica de los datos, si mantiene el mismo comportamiento de escalón positivo, para su análisis de predicción y determinar más características propias del problemas.

La segunda propuesta será desarrollar una interfaz en donde el arreglo de la red neuronal artificial no dependiera de Matlab y sea independiente aumentando su flexibilidad y rapidez de predicción para el usuario, se recomienda colocar un diagrama que represente la líneas de producción así como los supuestos que se manejan para estas líneas.

La tercer propuesta es la solución del BAP-2 como complemento al tener las tasas de producción eficientes lo que resta es buscar mediante un algoritmo que imite el método de la búsqueda de raíces como: el método de Bisección, Secante o Newton. En donde se presente un vector el cual contenga el número de piezas en un búfer y que la suma de este vector sea la capacidad de almacenamiento en una línea de producción y la eficiencia se busque en comparativa con la que genera la red neuronal y sea más ágil encontrar este vector.

## Apéndice A

### Experimentos usados para la red neuronal artificial

Tabla A.1: 360 experimentos de líneas de producción

N=1	K=3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	0.598771	0.533761	0.486654	0.459349	0.437107	0.422491	0.409803	0.400865	0.392788	0.386833	0.381302	0.377094	0.373106	0.369999	0.367006	0.364635	0.362321	0.36046
3	0.657904	0.567763	0.526182	0.487276	0.465856	0.446635	0.432007	0.421123	0.410961	0.403891	0.397361	0.391592	0.387197	0.382881	0.379377	0.376299	0.373249	0.370968
4	0.690055	0.610984	0.550301	0.521246	0.487657	0.470347	0.452806	0.439219	0.428596	0.420265	0.411639	0.405847	0.400134	0.395295	0.39081	0.387422	0.38378	0.380794
5	0.725474	0.641064	0.583023	0.539542	0.51778	0.487968	0.473621	0.458169	0.445664	0.435022	0.426602	0.419675	0.412171	0.407211	0.40237	0.397987	0.393936	0.39035
6	0.747229	0.662088	0.606704	0.564783	0.533155	0.515214	0.488175	0.476101	0.461879	0.4503	0.440121	0.432187	0.425007	0.419246	0.412523	0.408211	0.403877	0.399858
7	0.770778	0.687469	0.631371	0.586611	0.552118	0.528878	0.513241	0.488359	0.478039	0.465294	0.45413	0.445153	0.436688	0.429924	0.423952	0.418919	0.412825	0.408975
8	0.786484	0.70797	0.646434	0.604203	0.570608	0.542927	0.525219	0.511675	0.488487	0.479587	0.467761	0.457728	0.44869	0.440722	0.434133	0.428213	0.423031	0.418661
9	0.803267	0.722243	0.664948	0.6251	0.588898	0.560084	0.537223	0.522589	0.510407	0.488604	0.480845	0.470115	0.460443	0.451931	0.444837	0.437865	0.432267	0.427015
10	0.815134	0.7389	0.681167	0.638171	0.602596	0.575121	0.551107	0.532957	0.520221	0.509358	0.488687	0.481885	0.471865	0.462811	0.454694	0.447782	0.441071	0.435614
11	0.827698	0.753749	0.697935	0.650346	0.620724	0.590484	0.565352	0.544793	0.530022	0.518443	0.508479	0.488763	0.482753	0.47358	0.46509	0.4575	0.450546	0.444622
12	0.836978	0.764591	0.709031	0.665323	0.631759	0.601497	0.578068	0.557569	0.539477	0.52731	0.516787	0.488821	0.483485	0.474879	0.46686	0.45954	0.452856	0.446174
13	0.846735	0.775829	0.720369	0.677436	0.642505	0.617502	0.591646	0.56906	0.550636	0.536089	0.524914	0.515506	0.507087	0.488875	0.484107	0.476179	0.468468	0.461474
14	0.854191	0.787071	0.731851	0.69146	0.652972	0.627448	0.600715	0.580587	0.561886	0.545856	0.532844	0.52313	0.514285	0.506528	0.488916	0.48464	0.47718	0.470031
15	0.861987	0.795604	0.743976	0.701212	0.665562	0.636831	0.615045	0.592533	0.572147	0.556005	0.541295	0.530653	0.521423	0.513319	0.506039	0.488955	0.485097	0.478194
16	0.868105	0.803621	0.752476	0.709759	0.675012	0.645494	0.623798	0.60014	0.582358	0.565595	0.550314	0.538093	0.52841	0.519901	0.512382	0.505608	0.488984	0.485493
17	0.874479	0.812423	0.761146	0.71922	0.68696	0.654984	0.631983	0.61311	0.593234	0.574776	0.559852	0.546543	0.535417	0.526538	0.5187	0.511629	0.505224	0.48901
18	0.879591	0.819311	0.768638	0.727641	0.695223	0.66573	0.640204	0.621164	0.599708	0.583958	0.568224	0.554938	0.542587	0.532781	0.524741	0.517526	0.510889	0.504883
19	0.884899	0.825486	0.777764	0.737682	0.703324	0.673332	0.647947	0.628715	0.611552	0.593799	0.576764	0.56293	0.550121	0.539455	0.531054	0.523422	0.51648	0.510285
20	0.889234	0.832367	0.784476	0.745217	0.71022	0.683664	0.656313	0.636059	0.618821	0.599376	0.585133	0.570594	0.558244	0.547023	0.537102	0.529206	0.522036	0.515617
20	0.893721	0.838036	0.791315	0.751711	0.71842	0.691169	0.665853	0.643031	0.625781	0.610277	0.594265	0.578389	0.565731	0.554167	0.543618	0.534973	0.527575	0.520847

---

## Apéndice B

### Experimentos simulados por la red neuronal artificial

Tabla B.1: 360 experimentos de producción generados por la red neuronal artificial

N=1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
0.59878482	0.533754697	0.486653436	0.457969421	0.437514721	0.422604853	0.411234656	0.401256872	0.390339778	0.386576248	0.381426397	0.377040275	0.373221754	0.370165347	0.367168922	0.364591494	0.362337664	0.360465491	0.358906599	0.357592325	0.356400063
0.657972081	0.567804106	0.520285741	0.488425237	0.461077482	0.440461804	0.417128116	0.420879597	0.411512069	0.403586848	0.39768826	0.391648888	0.386906302	0.382806505	0.379461894	0.376114625	0.373268235	0.3709376	0.3690829826	0.367366872	0.365766672
0.690127228	0.610772482	0.550138576	0.52002475	0.487709669	0.470461788	0.453036813	0.439098011	0.428804557	0.419850665	0.412130473	0.406673962	0.400171891	0.395230376	0.390945381	0.387346838	0.383766672	0.380829826	0.378466672	0.376230376	0.374098011
0.725445945	0.641877539	0.583071317	0.539756015	0.516707804	0.487790117	0.473579895	0.458065134	0.445133688	0.435323043	0.426661786	0.419145319	0.412845504	0.407346137	0.402290642	0.397866021	0.394082372	0.390931969	0.38829826	0.385806672	0.383376672
0.747432374	0.66199613	0.607078908	0.564686907	0.532213957	0.514571198	0.488198481	0.47590156	0.462024962	0.45001382	0.440694408	0.432832528	0.424981224	0.418844904	0.413437256	0.408339959	0.403870616	0.399885131	0.396398672	0.393098672	0.390004406
0.770290669	0.687269758	0.631204378	0.586663386	0.552068326	0.528653338	0.512951446	0.488489188	0.477766156	0.465218427	0.454180073	0.4457169045	0.436995046	0.429223686	0.423075587	0.418648607	0.413534461	0.40904406	0.40494406	0.40104406	0.39734406
0.787380784	0.707853801	0.647407928	0.604105523	0.570854033	0.543123924	0.52522474	0.511606079	0.488606413	0.479328292	0.467840965	0.457575308	0.448919189	0.441021629	0.434178399	0.428339471	0.423138827	0.418002404	0.413002404	0.408002404	0.403002404
0.802427272	0.722211966	0.664806076	0.6295101364	0.60202017	0.58292017	0.56382017	0.54472017	0.52562017	0.50652017	0.48742017	0.46832017	0.44922017	0.43012017	0.41102017	0.39192017	0.37282017	0.35372017	0.33462017	0.31552017	0.29642017
0.815801916	0.73854747	0.68202017	0.64806076	0.625101364	0.602141961	0.579182558	0.556223155	0.533263752	0.510304349	0.487344946	0.464385543	0.441426140	0.418466737	0.395507334	0.372547931	0.349588528	0.326629125	0.303669722	0.280710319	0.257750916
0.827410157	0.753400442	0.6972924188	0.6672924188	0.6442924188	0.6212924188	0.5982924188	0.5752924188	0.5522924188	0.5292924188	0.5062924188	0.4832924188	0.4602924188	0.4372924188	0.4142924188	0.3912924188	0.3682924188	0.3452924188	0.3222924188	0.2992924188	0.2762924188
0.837555191	0.765465018	0.709466669	0.68202017	0.654573666	0.627127163	0.599680660	0.572234157	0.544787654	0.517341151	0.489894648	0.462448145	0.435001642	0.407555139	0.380108636	0.352662133	0.325215630	0.297769127	0.270322624	0.242876121	0.215429618
0.846516781	0.776304668	0.720500398	0.692924188	0.665348475	0.637772562	0.610196649	0.582620736	0.555044823	0.527468910	0.500000000	0.472531087	0.445062174	0.417593261	0.390124348	0.362655435	0.335186522	0.307717609	0.280248696	0.252779783	0.225310870
0.85452377	0.795751611	0.743000282	0.715900282	0.688800282	0.661700282	0.634600282	0.607500282	0.580400282	0.553300282	0.526200282	0.499100282	0.472000282	0.444900282	0.417800282	0.390700282	0.363600282	0.336500282	0.309400282	0.282300282	0.255200282
0.861723751	0.795751611	0.743000282	0.715900282	0.688800282	0.661700282	0.634600282	0.607500282	0.580400282	0.553300282	0.526200282	0.499100282	0.472000282	0.444900282	0.417800282	0.390700282	0.363600282	0.336500282	0.309400282	0.282300282	0.255200282
0.868230007	0.804432123	0.752000282	0.724900282	0.697800282	0.670700282	0.643600282	0.616500282	0.589400282	0.562300282	0.535200282	0.508100282	0.481000282	0.453900282	0.426800282	0.399700282	0.372600282	0.345500282	0.318400282	0.291300282	0.264200282
0.874146662	0.812440499	0.760000282	0.732900282	0.705800282	0.678700282	0.651600282	0.624500282	0.597400282	0.570300282	0.543200282	0.516100282	0.489000282	0.461900282	0.434800282	0.407700282	0.380600282	0.353500282	0.326400282	0.299300282	0.272200282
0.8795727	0.819747261	0.767300282	0.740200282	0.713100282	0.686000282	0.658900282	0.631800282	0.604700282	0.577600282	0.550500282	0.523400282	0.496300282	0.469200282	0.442100282	0.415000282	0.387900282	0.360800282	0.333700282	0.306600282	0.279500282
0.884600311	0.826347941	0.773900282	0.746800282	0.719700282	0.692600282	0.665500282	0.638400282	0.611300282	0.584200282	0.557100282	0.530000282	0.502900282	0.475800282	0.448700282	0.421600282	0.394500282	0.367400282	0.340300282	0.313200282	0.286100282
0.889323373	0.832282008	0.780453439	0.753353439	0.726253439	0.699153439	0.672053439	0.644953439	0.617853439	0.590753439	0.563653439	0.536553439	0.509453439	0.482353439	0.455253439	0.428153439	0.401053439	0.373953439	0.346853439	0.319753439	0.292653439
0.8933806916	0.837626947	0.7859245247	0.7588245247	0.7317245247	0.7046245247	0.6775245247	0.6504245247	0.6233245247	0.5962245247	0.5691245247	0.5420245247	0.5149245247	0.4878245247	0.4607245247	0.4336245247	0.4065245247	0.3794245247	0.3523245247	0.3252245247	0.2981245247

---

# Bibliografía

- [1] J. MacGregor Smith and F. Cruz, “The buffer allocation problem for general finite buffer queueing networks,” *IIE Transactions*, vol. 37, no. 4, pp. 343–365, 2005.
- [2] L. Demir, S. Tunali, and D. T. Eliiyi, “The state of the art on buffer allocation problem: a comprehensive survey,” *Journal of Intelligent Manufacturing*, vol. 25, no. 3, pp. 371–392, 2014.
- [3] C. Shi and S. B. Gershwin, “An efficient buffer design algorithm for production line profit maximization,” *International Journal of Production Economics*, vol. 122, no. 2, pp. 725–740, 2009.
- [4] I. Automotriz, “Monografía, 2014, dirección general de industrias pesadas y de alta tecnología, secretaría de economía,” 2014.
- [5] S. B. Gershwin and J. E. Schor, “Efficient algorithms for buffer space allocation,” *Annals of Operations Research*, vol. 93, no. 1-4, pp. 117–144, 2000.
- [6] E. Koenigsberg, “Production lines and internal storage—a review,” *Management Science*, vol. 5, no. 4, pp. 410–433, 1959.
- [7] J. A. Buzacott and J. G. Shanthikumar, *Stochastic models of manufacturing systems*, vol. 4. Prentice Hall Englewood Cliffs, NJ, 1993.
- [8] H. Papadopoulos, J. Browne, and C. Heavey, *Queueing theory in manufacturing systems analysis and design*. Chapman & Hall, 1993.
- [9] Y. Dallery and S. B. Gershwin, “Manufacturing flow line systems: a review of models and analytical results,” *Queueing systems*, vol. 12, no. 1-2, pp. 3–94, 1992.
- [10] H. Papadopoulos and C. Heavey, “Queueing theory in manufacturing systems analysis and design: A classification of models for production and transfer lines,” *European journal of operational Research*, vol. 92, no. 1, pp. 1–27, 1996.

- 
- [11] T. Park, “A two-phase heuristic algorithm for determining buffer sizes of production lines,” *THE INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH*, vol. 31, no. 3, pp. 613–631, 1993.
- [12] D. Seong, S. Chang, and Y. Hong, “An algorithm for buffer allocation with linear resource constraints in a continuous flow production line,” *Department of Industrial Engineering, POSTECH, Korea, Technical Report IE-TR-94-05*, 1994.
- [13] D. Seong, S. Chang, and Y. Hong, “Heuristic algorithms for buffer allocation in a production line with unreliable machines,” *International Journal of Production Research*, vol. 33, no. 7, pp. 1989–2005, 1995.
- [14] S. B. Gershwin and Y. Goldis, *Efficient algorithms for transfer line design*. Massachusetts Institute of Technology, 1995.
- [15] G. A. Vouros and H. Papadopoulos, “Buffer allocation in unreliable production lines using a knowledge based system,” *Computers & operations research*, vol. 25, no. 12, pp. 1055–1067, 1998.
- [16] N. Nahas, D. Ait-Kadi, and M. Nourelfath, “A new approach for buffer allocation in unreliable production lines,” *International journal of production economics*, vol. 103, no. 2, pp. 873–881, 2006.
- [17] L. FUXMAN, “Optimal buffer allocation in asynchronous cyclic mixed-model assembly lines,” *Production and Operations Management*, vol. 7, no. 3, pp. 294–311, 1998.
- [18] J. H. Harris and S. G. Powell, “An algorithm for optimal buffer placement in reliable serial lines,” *IIE transactions*, vol. 31, no. 4, pp. 287–302, 1999.
- [19] K.-C. Jeong and Y.-D. Kim, “Heuristics for selecting machines and determining buffer capacities in assembly systems,” *Computers & industrial engineering*, vol. 38, no. 3, pp. 341–360, 2000.
- [20] H. Papadopoulos and M. Vidalis, “A heuristic algorithm for the buffer allocation in unreliable unbalanced production lines,” *Computers & Industrial Engineering*, vol. 41, no. 3, pp. 261–277, 2001.
- [21] N. Hemachandra and S. K. Eedupuganti, “Performance analysis and buffer allocations in some open assembly systems,” *Computers & Operations Research*, vol. 30, no. 5, pp. 695–704, 2003.

- [22] H. Tempelmeier, “Practical considerations in the optimization of flow production systems,” *International Journal of Production Research*, vol. 41, no. 1, pp. 149–170, 2003.
- [23] I. Sabuncuoglu, E. Erel, and Y. Gocgun, “Analysis of serial production lines: characterisation study and a new heuristic procedure for optimal buffer allocation,” *International Journal of Production Research*, vol. 44, no. 13, pp. 2499–2523, 2006.
- [24] R. I. Zequeira, J. E. Valdes, and C. Berenguer, “Optimal buffer inventory and opportunistic preventive maintenance under random production capacity availability,” *International Journal of Production Economics*, vol. 111, no. 2, pp. 686–696, 2008.
- [25] H. K. Aksoy and S. M. Gupta, “Near optimal buffer allocation in remanufacturing systems with n-policy,” *Computers & Industrial Engineering*, vol. 59, no. 4, pp. 496–508, 2010.
- [26] C. M. Lutz, K. R. Davis, and M. Sun, “Determining buffer location and size in production lines using tabu search,” *European Journal of Operational Research*, vol. 106, no. 2, pp. 301–316, 1998.
- [27] L. Demir, S. Tunali, and D. T. Eliiyi, “An adaptive tabu search approach for buffer allocation problem in unreliable production lines,” in *24th Mini EURO conference on continuous optimization and information-based technologies in the financial sector, Selected Papers*, pp. 207–212, 2010.
- [28] L. Demir, S. Tunali, and A. Løkketangen, “A tabu search approach for buffer allocation in production lines with unreliable machines,” *Engineering Optimization*, vol. 43, no. 2, pp. 213–231, 2011.
- [29] L. Demir, S. Tunali, and D. T. Eliiyi, “An adaptive tabu search approach for buffer allocation problem in unreliable non-homogenous production lines,” *Computers & Operations Research*, vol. 39, no. 7, pp. 1477–1486, 2012.
- [30] D. D. Spinellis and C. T. Papadopoulos, “A simulated annealing approach for buffer allocation in reliable production lines,” *Annals of Operations Research*, vol. 93, no. 1-4, pp. 373–384, 2000.
- [31] D. Spinellis, C. Papadopoulos, and J. M. Smith, “Large production line optimization using simulated annealing,” *International Journal of Production Research*, vol. 38, no. 3, pp. 509–541, 2000.
- [32] D. D. Spinellis and C. T. Papadopoulos, “Stochastic algorithms for buffer allocation in reliable production lines,” *Mathematical Problems in Engineering*, vol. 5, no. 6, pp. 441–458, 2000.

- [33] A. Dolgui, A. Eremeev, A. Kolokolov, and V. Sigaev, "A genetic algorithm for the allocation of buffer storage capacities in a production line with unreliable machines," *Journal of Mathematical Modelling and Algorithms*, vol. 1, no. 2, pp. 89–104, 2002.
- [34] J. A. Qudeiri, H. Yamamoto, R. Ramli, and K. R. Al-Momani, "Development of production simulator for buffer size decisions in complex production systems using genetic algorithm," *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, vol. 1, no. 3, pp. 418–429, 2007.
- [35] J. A. Qudeiri, H. Yamamoto, R. Ramli, and A. Jamali, "Genetic algorithm for buffer size and work station capacity in serial-parallel production lines," *Artificial Life and Robotics*, vol. 12, no. 1-2, pp. 102–106, 2008.
- [36] F. Cruz, T. Van Woensel, and J. M. Smith, "Buffer and throughput trade-offs in m/g/1/k queueing networks: A bi-criteria approach," *International Journal of Production Economics*, vol. 125, no. 2, pp. 224–234, 2010.
- [37] S. Kose, *Capacity improvement in a real manufacturing system using a hybrid simulation/genetic algorithm approach*. PhD thesis, MS Thesis, Dokuz Eylul University, Graduate School of Natural and Applied Sciences, 2010.
- [38] M. Amiri and A. Mohtashami, "Buffer allocation in unreliable production lines based on design of experiments, simulation, and genetic algorithm," *The International Journal of Advanced Manufacturing Technology*, vol. 62, no. 1, pp. 371–383, 2012.
- [39] C. Srinivas, B. Satyanarayana, K. Ramji, and N. Ravela, "Determination of buffer size in single and multi row flexible manufacturing systems through simulation," *International Journal of Engineering Science and Technology*, vol. 3, no. 5, 2011.
- [40] A. Artiba, F. Riane, M. Nourelfath, N. Nahas, and D. Ait-Kadi, "Optimal design of series production lines with unreliable machines and finite buffers," *Journal of Quality in Maintenance Engineering*, vol. 11, no. 2, pp. 121–138, 2005.
- [41] N. Nahas, M. Nourelfath, and D. Ait-Kadi, "Selecting machines and buffers in unreliable series-parallel production lines," *International Journal of Production Research*, vol. 47, no. 14, pp. 3741–3774, 2009.
- [42] L. Shi and S. Men, "Optimal buffer allocation in production lines," *IIE Transactions*, vol. 35, no. 1, pp. 1–10, 2003.

- [43] A. Dolgui, A. V. Ereemeev, and V. S. Sigaev, “Hbba: hybrid algorithm for buffer allocation in tandem production lines,” *Journal of Intelligent Manufacturing*, vol. 18, no. 3, pp. 411–420, 2007.
- [44] H. Yamashita and T. Altiok, “Buffer capacity allocation for a desired throughput in production lines,” *IIE transactions*, vol. 30, no. 10, pp. 883–892, 1998.
- [45] A. Diamantidis and C. Papadopoulos, “A dynamic programming algorithm for the buffer allocation problem in homogeneous asymptotically reliable serial production lines,” *Mathematical problems in Engineering*, vol. 2004, no. 3, pp. 209–223, 2004.
- [46] A. Bulgak, “Analysis and design of split and merge unpaced assembly systems by meta-modelling and stochastic search,” *International Journal of Production Research*, vol. 44, no. 18-19, pp. 4067–4080, 2006.
- [47] F. Altıparmak, B. Dengiz, and A. A. Bulgak, “Buffer allocation and performance modeling in asynchronous assembly system operations: An artificial neural network metamodeling approach,” *Applied Soft Computing*, vol. 7, no. 3, pp. 946–956, 2007.
- [48] B. Can and C. Heavey, “A comparison of genetic programming and artificial neural networks in metamodeling of discrete-event simulation models,” *Computers & Operations Research*, vol. 39, no. 2, pp. 424–436, 2012.
- [49] B. Can and C. Heavey, “Comparison of experimental designs for simulation-based symbolic regression of manufacturing systems,” *Computers and Industrial Engineering*, vol. 61, no. 3, pp. 447–462, 2011.
- [50] Y. Massim, F. Yalaoui, L. Amodeo, É. Châtelet, and A. Zeblah, “Efficient combined immune-decomposition algorithm for optimal buffer allocation in production lines for throughput and profit maximization,” *Computers and Operations Research*, vol. 37, no. 4, pp. 611–620, 2010.
- [51] I. Sabuncuoglu, E. Erel, and A. GURHAN KOK, “Analysis of assembly systems for interdeparture time variability and throughput,” *IIE Transactions*, vol. 34, no. 1, pp. 23–40, 2002.
- [52] N. A. Raman and E. K. R. Jamaludin, “Implementation of toyota production system (tps) in the production line of a local automotive parts manufacturer,” *Proceedings of International Conference on Mechanical, Manufacturing Engineering*, 2008.

- [53] T. McNamara, S. Shaaban, and S. Hudson, “Unpaced production lines with three simultaneous imbalance sources,” *Industrial Management & Data Systems*, vol. 111, no. 9, pp. 1356–1380, 2011.
- [54] D. Seong, S. Y. Chang, and Y. Hong, “An algorithm for buffer allocation with linear resource constraints in a continuous-flow unreliable production line,” *Asia-Pacific Journal of Operational Research*, vol. 17, no. 2, p. 169, 2000.
- [55] S. Helber, “Cash-flow-oriented buffer allocation in stochastic flow lines,” *International Journal of Production Research*, vol. 39, no. 14, pp. 3061–3083, 2001.
- [56] S. Daskalaki and J. M. Smith, “Combining routing and buffer allocation problems in series-parallel queueing networks,” *Annals of Operations Research*, vol. 125, no. 1-4, pp. 47–68, 2004.
- [57] J. Smith and F. Cruz, “The buffer allocation problem for general finite buffer queueing networks,” *Unpublished.[Online]. Available: <http://www.ecs.umass.edu/mie/faculty/smith>*, 2000.
- [58] F. R. Cruz, A. R. Duarte, and T. Van Woensel, “Buffer allocation in general single-server queueing networks,” *Computers & Operations Research*, vol. 35, no. 11, pp. 3581–3598, 2008.
- [59] J. M. Smith, F. R. Cruz, and T. Van Woensel, “Topological network design of general, finite, multi-server queueing networks,” *European Journal of Operational Research*, vol. 201, no. 2, pp. 427–441, 2010.
- [60] F. R. B. d. Cruz, G. Kendall, L. While, A. R. Duarte, and N. L. C. Brito, “Throughput maximization of queueing networks with simultaneous minimization of service rates and buffers,” *Mathematical Problems in Engineering*, vol. 2012, 2012.
- [61] G. Gürkan, “Simulation optimization of buffer allocations in production lines with unreliable machines,” *Annals of Operations Research*, vol. 93, no. 1-4, pp. 177–216, 2000.
- [62] H. Yamamoto, J. A. Qudciri, and E. Marui, “Definition of flt with bypass lines and its simulator for buffer size decision,” *International Journal of Production Economics*, vol. 112, no. 1, pp. 18–25, 2008.
- [63] D. Battini, A. Persona, and A. Regattieri, “Buffer size design linked to reliability performance: A simulative study,” *Computers & Industrial Engineering*, vol. 56, no. 4, pp. 1633–1641, 2009.

- [64] F. S. Hillier and K. C. So, “The effect of machine breakdowns and interstage storage on the performance of production line systems,” *THE INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH*, vol. 29, no. 10, pp. 2043–2055, 1991.
- [65] F. S. Hillier, K. C. So, and R. W. Boling, “Notes: Toward characterizing the optimal allocation of storage space in production line systems with variable processing times,” *Management Science*, vol. 39, no. 1, pp. 126–133, 1993.
- [66] Ö. Selvi, *The line balancing algorithm for optimal buffer allocation in production lines*. PhD thesis, Department of Industrial Engineering and the Institute of Engineering and Science of Bilkent Univ., 2002.
- [67] J. Trujillano, J. March, and A. Sorribas, “Aproximación metodológica al uso de redes neuronales artificiales para la predicción de resultados en medicina,” *Med Clin (Barc)*, vol. 122, no. s1, 2004.
- [68] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943.
- [69] M. A. Valencia Reyes, “Algoritmo backpropagation para redes neuronales: conceptos y aplicaciones,” tech. rep., Instituto Politécnico Nacional. Centro de Investigación en Computación, 2007.
- [70] K. Levenberg, “A method for the solution of certain non-linear problems in least squares,” *Quarterly of applied mathematics*, vol. 2, no. 2, pp. 164–168, 1944.
- [71] D. W. Marquardt, “An algorithm for least-squares estimation of nonlinear parameters,” *Journal of the society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.
- [72] M. T. Hagan and M. B. Menhaj, “Training feedforward networks with the marquardt algorithm,” *IEEE transactions on Neural Networks*, vol. 5, no. 6, pp. 989–993, 1994.
- [73] L. Breiman *et al.*, “Heuristics of instability and stabilization in model selection,” *The annals of statistics*, vol. 24, no. 6, pp. 2350–2383, 1996.
- [74] L. Breiman and P. Spector, “Submodel selection and evaluation in regression. the x-random case,” *International statistical review/revue internationale de Statistique*, pp. 291–319, 1992.
- [75] R. Kohavi *et al.*, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in *Ijcai*, vol. 14, pp. 1137–1145, Stanford, CA, 1995.

- 
- [76] S. G. Gilmour, “Response surface designs for experiments in bioprocessing,” *Biometrics*, vol. 62, no. 2, pp. 323–331, 2006.
- [77] B. B. Neto, I. Scarminio, and R. Bruns, “Statistical design–chemometrics,” 2006.
- [78] R. F. Teófilo, M. Ferreira, *et al.*, “Quimiometria ii: planilhas eletrônicas para cálculos de planejamentos experimentais, um tutorial,” *Química Nova*, 2006.
- [79] T. Lundstedt, E. Seifert, L. Abramo, B. Thelin, Å. Nyström, J. Pettersen, and R. Bergman, “Experimental design and optimization,” *Chemometrics and intelligent laboratory systems*, vol. 42, no. 1-2, pp. 3–40, 1998.
- [80] C. Papadopoulos, M. O’Kelly, and A. Tsadiras, “A dss for the buffer allocation of production lines based on a comparative evaluation of a set of search algorithms,” 2013.
- [81] A. Tsadiras, C. Papadopoulos, and M. O’Kelly, “An artificial neural network based decision support system for solving the buffer allocation problem in reliable production lines,” *Computers & Industrial Engineering*, vol. 66, no. 4, pp. 1150 – 1162, 2013.