



**UNIVERSIDAD NACIONAL DE CHIMBORAZO**

**FACULTAD DE INGENIERÍA**

**ESCUELA DE INGENIERÍA CIVIL**

**TRABAJO DE GRADO**

Previo a la obtención del Título de:

**INGENIERO CIVIL**

**MODALIDAD: PROYECTO DE INVESTIGACIÓN**

**TÍTULO:**

OPTIMIZACIÓN DE VIGAS – TRABE TIPO I ARMADAS MEDIANTE PLACAS  
DE ACERO, UTILIZANDO ALGORITMOS GENÉTICOS

**AUTOR:**

Hernán David Salazar Valverde

**DIRECTOR DEL PROYECTO**

Ing. Alexis Martínez.

2015

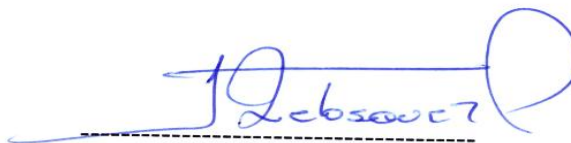
Los miembros del Tribunal de Graduación del proyecto de título: **“OPTIMIZACIÓN DE VIGAS – TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO, UTILIZANDO ALGORITMOS GENÉTICOS”**, presentando por: Hernán David Salazar Valverde y dirigida por: Ing. Alexis Martínez.

Una vez escuchada la defensa oral y revisado el informe final de investigación con fines de graduación escrito en la cual se ha constatado el cumplimiento de las observaciones realizadas, remite la presente para uso y custodia en la biblioteca de la Facultad de Ingeniería de la UNACH.

Para constancia de lo expuesto firman:

Ing. Víctor Velásquez

**PRESIDENTE DEL TRIBUNAL**



Firma

Ing. Alexis Martínez

**DIRECTOR DEL PROYECTO**



Firma

Ing. Daniel Ríos

**MIEMBRO DEL TRIBUNAL**



Firma

**AUTORÍA DE LA INVESTIGACIÓN**

“La responsabilidad del contenido de este Proyecto de Graduación, nos corresponde exclusivamente a: Hernán David Salazar Valverde y del Director del Proyecto: Ing. Alexis Martínez y el patrimonio intelectual a la Universidad Nacional de Chimborazo”



060359483-9

### **AGRADECIMIENTO.**

Quiero expresar mi más profundo agradecimiento primeramente a Dios por el camino recorrido, por darme salud, vida y la oportunidad de convertirme en un ser útil. Por su misericordia, bondad y fortaleza diaria.

De igual manera quiero agradecer a la “Universidad Nacional de Chimborazo” que me ha permitido seguir adelante en mi carrera profesional para llegar a cumplir mis aspiraciones; por brindarme la educación, tanto moral como académicamente, importante para mi formación espiritual. Gracias por su apoyo y ayuda durante este proceso.

Un reconocimiento especial a mis docentes y autoridades asesoras de mi trabajo quienes con demostrada capacidad, profesionalismo y desinterés han sabido conducirme en forma adecuada y oportuna en el desarrollo de este proceso.

*Hernán David Salazar Valverde.*

## **DEDICATORIA**

Quiero dedicar el presente trabajo a mi madre Elva Valverde por su esfuerzo, por ser la persona que me ha apoyado incondicionalmente, por ser la fuente de mi inspiración y motivación para superarme cada día más; y de esa manera cumplir mis sueños y metas planteadas. A mis hermanos, y mi Abuelita Rosita gracias por creer en mí y aportar para que pueda ser mejor persona, un ser útil para la sociedad. Por darme todo lo que soy como persona: valores, principios, buenos hábitos, costumbres; y, todo esto con mucho amor, paciencia y constancia.

A mi compañera de vida Sofy por su ayuda desinteresada la cual ha llegado a ser un gran apoyo en mi vida.

De manera especial a mis primos Fabián, Edíson y Jorge ya que con ellos he vivido las mejores e inolvidables experiencias con virtudes y defectos pero en los cuales he encontrado unión y apoyo incondicional en todo momento. A mis compañeros y amigos porque con ellos he compartido sueños y conocimientos en esta carrera universitaria los mismos que han enriquecido vida.

*Hernán David Salazar Valverde.*

## ÍNDICE GENERAL

ÍNDICE GENERAL .....	vi
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE TABLAS .....	xii
RESUMEN .....	xiii
CAPITULO 1.....	1
1. MARCO REFERENCIAL .....	2
1.1. INTRODUCCIÓN .....	2
1.2. PROBLEMATIZACIÓN .....	2
1.2.1. IDENTIFICACIÓN Y DESCRIPCIÓN DEL PROBLEMA.....	2
1.2.2. LIMITACIONES .....	3
1.2.3. FORMULACIÓN DEL PROBLEMA.....	4
1.3. OBJETIVOS.....	4
1.3.1. GENERAL.....	4
1.3.2. ESPECÍFICOS .....	4
1.4. JUSTIFICACIÓN.....	4
CAPITULO II.....	6
2. MARCO TEÓRICO .....	6
2.1.1. ANTECEDENTES .....	6
2.2. FUNDAMENTACION TEORICA.....	6
2.2.1. DISEÑO DE VIGAS ARMADAS (Plate Girders).....	6
2.2.1.1. ESTADOS LÍMITES POR FLEXIÓN.....	7
2.2.1.1.1. CEDENCIA DEL ALA EN TENSIÓN.....	7
2.2.1.1.2. PANDEO LOCAL DEL ALA.....	8
2.2.1.1.3. PANDEO LATERAL TORSIONAL .....	10
2.2.1.2. ESTADO LÍMITES POR CORTANTE.....	12
2.2.1.2.1. CEDENCIA DEL ALMA.....	13
2.2.1.2.2. PANDEO INELÁSTICO DEL ALMA .....	13
2.2.1.2.3. PANDEO ELÁSTICO DEL ALMA .....	13
2.3. ALGORITMOS GENÉTICOS (AG).....	14
2.3.1. INTRODUCCIÓN A LOS AG.....	14
2.3.2. DEFINICIÓN DE ALGORITMOS GENÉTICOS (AG) .....	15
2.3.3. ALGORITMOS GENÉTICOS EN LA OPTIMIZACIÓN.....	16

2.3.4.	VENTAJAS Y DESVENTAJAS DE LOS AG .....	17
2.3.4.1.	VENTAJAS .....	17
2.3.4.2.	DESVENTAJAS .....	17
2.3.5.	LIMITACIONES DE LOS AG .....	17
2.3.6.	CONDICIONES PARA LA APLICACIÓN DE UN ALGORITMO GENÉTICO .....	18
2.3.7.	MARCO DE DESARROLLO DE LOS AG .....	19
2.3.8.	COMPARACIÓN CON OTROS MÉTODOS DE OPTIMIZACIÓN .....	21
2.3.8.1.	ALGORITMOS GENÉTICOS Y MÉTODOS ENUMERATIVOS .....	21
2.3.8.2.	ALGORITMOS GENÉTICOS Y SISTEMAS EXPERTOS .....	21
2.3.8.3.	ALGORITMOS GENÉTICOS Y REDES NEURONALES.....	22
2.3.8.4.	EL ALGORITMO GENÉTICO SIMPLE .....	23
2.3.8.4.1.	CODIFICACIÓN.....	24
2.3.8.4.2.	POBLACIÓN .....	26
2.3.8.4.2.1.	TAMAÑO DE LA POBLACIÓN.....	26
2.3.8.4.2.2.	POBLACIÓN INICIAL .....	27
2.3.8.4.3.	FUNCIÓN OBJETIVO .....	27
2.3.8.4.4.	SELECCIÓN .....	29
2.3.8.4.5.	CRUCE.....	29
2.3.8.4.6.	MUTACIÓN.....	30
CAPITULO III.....		32
3.	MARCO METODOLÓGICO .....	32
3.1.	SISTEMATIZACIÓN DE LA METODOLOGÍA A EMPLEARSE .....	32
3.1.1.	ESTUDIO Y ANÁLISIS DEL MÉTODO DE DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO, CONDICIONES DE DISEÑO Y ECUACIONES.....	32
3.1.2.	INVESTIGACIÓN Y ANÁLISIS DE LA ESTRUCTURA DE ALGORITMO GENÉTICO.....	33
3.1.3.	DETERMINACIÓN DE VARIABLES DEL ALGORITMO GENÉTICO PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO.....	33
3.1.4.	DETERMINACIÓN DE LA FUNCIÓN OBJETIVO DEL ALGORITMO GENÉTICO DIRIGIDA AL DISEÑO VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO .....	33
3.1.5.	ESTABLECER LIMITACIÓN DEL ALGORITMO GENÉTICO PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO.	

3.1.6.	FORMULACIÓN DEL MODELO MATEMÁTICO QUE RIGE EL ALGORITMO GENÉTICO PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO.	34
3.1.7.	IMPLEMENTACIÓN DEL ALGORITMO GENÉTICO PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO.	40
3.1.8.	APLICACIÓN DEL ALGORITMO GENÉTICO A VIGAS-TRABE TIPO I.	40
3.1.9.	COMPARACIÓN DE RESULTADOS ENTRE EL DISEÑO TRADICIONAL Y UTILIZANDO EL ALGORITMO GENÉTICO.	40
3.2.	TIPO DE ESTUDIO.	40
3.3.	DISEÑO DE LA INVESTIGACIÓN	41
3.4.	OPERACIONALIZACIÓN DE VARIABLES	42
3.5.	HIPOTESIS	43
3.5.1.	IDENTIFICACIÓN DE VARIABLES.	43
3.5.1.1.	VARIABLE INDEPENDIENTE	43
3.5.1.2.	VARIABLE DEPENDIENTE.	43
3.6.	PROCEDIMIENTOS.	43
3.6.1.	PLANTEAMIENTO DE LOS EJERCICIOS DE DISEÑO VIGAS-TRABE TIPO I ARMADA MEDIANTE PLACAS DE ACERO.	44
3.7.	PROCESAMIENTO Y ANÁLISIS	46
3.7.1.	DISEÑO DE VIGAS-TRABE TIPO I ARMADA MEDIANTE PLACAS DE ACERO CON LA UTILIZACIÓN DE ALGORITMOS GENÉTICOS.	46
3.7.1.1.	EXPLICACIÓN GLOBAL DEL PROCESO QUE SIGUE EL ALGORITMO GENÉTICO APLICADO AL DISEÑO DE VIGAS-TRABE TIPO I ARMADA MEDIANTE PLACAS DE ACERO.	46
3.7.1.2.	EXPLICACIÓN PASO A PASO DEL PROCESO QUE SIGUE EL ALGORITMO GENÉTICO APLICADO AL DISEÑO DE VIGAS-TRABE TIPO I ARMADA MEDIANTE PLACAS DE ACERO.	47
3.7.1.2.1.	PROCESO 1	47
3.7.1.2.2.	PROCESO 2	48
3.7.1.2.2.1.	GENERAR LA POBLACIÓN INICIAL	49
3.7.1.2.2.2.	CODIFICACIÓN DE INDIVIDUOS	51
3.7.1.2.2.3.	GENERAR PAREJAS Y POSICIÓN DE CRUCE	52
3.7.1.2.2.4.	ALMACENAMIENTO DE INFORMACIÓN BINARIA	54
3.7.1.2.2.5.	REALIZAR CRUCE DE INDIVIDUOS.	54
3.7.1.2.2.6.	REALIZAR MUTACIÓN.	57
3.7.1.2.2.7.	TRANSFORMAR DE BINARIO A DECIMAL.	59
3.7.1.2.2.8.	EVALUAR LA FUNCIÓN DE APTITUD	61



3.7.1.2.2.9. REALIZAR SELECCIÓN DE INDIVIDUOS MEJOR ADAPTADOS .....	62
3.7.1.2.2.10. PASAR LA POBLACIÓN SELECCIONADA COMO NUEVA POBLACIÓN. ....	63
3.8. ANALISIS ESTADÍSTICO DE LOS RESULTADOS. ....	66
CAPITULO IV.....	67
4. RESULTADOS Y DISCUSIÓN .....	67
4.1. COMPARACIÓN MÉTODO CONVENCIONAL VS ALGORITMO GENÉTICO	71
4.2. COMPORTAMIENTO DEL ALGORITMO GENÉTICO .....	73
CAPITULO V.....	76
5. CONCLUSIONES Y RECOMENDACIONES. ....	76
5.1. CONCLUSIONES .....	76
5.2. RECOMENDACIONES .....	80
CAPITULO VI.....	81
6. PROPUESTA .....	81
6.1. TITULO DE LA PROPUESTA .....	81
6.2. TITULO DE LA PROPUESTA .....	81
6.3. OBJETIVO GENERAL .....	81
6.3.1. GENERAL.....	81
6.3.2. ESPECÍFICOS.....	81
6.4. DESCRIPCION DE LA PROPUESTA .....	82
6.4.1. PROGRAMA COMPUTACIONAL DEAC-TB PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO UTILIZANDO ALGORITMOS GENÉTICOS. ....	82
6.4.1.1. PROCESO LÓGICO DEL SOFTWARE DEAC-TB .....	83
6.4.2. ESTRUCTURACIÓN DEL SOFTWARE DEAC-TB.....	84
6.4.2.1. PAGINA ALGORITMO GENETICO. ....	85
6.5. MONITOREO Y EVALUACION DE LA PROPUESTA. ....	89
CAPITULO VII .....	90
7. BIBLIOGRAFIA.....	90
CAPITULO VIII.....	91
8. APENDICES Y ANEXOS. ....	91

## ÍNDICE DE FIGURAS

Figura 2.1. Compresión vertical en el alma debido a Fuerzas desbalanceadas en los patines...	8
Figura 2.2. Analogía de Plate-Girder rigirizado con cercha .....	12
Figura 2.3. Codificación de individuos en AG .....	19
Figura 2.4. Esquema de una red neuronal.....	23
Figura 2.5. Esquema de un algoritmo genético básico .....	24
Figura 2.6. Esquema de un cruce de un solo punto en un AG.....	25
Figura 2.7. Esquema de mutación de un individuo de AG .....	26
Figura 3.1 Esquema del diseño de investigación.....	41
Figura 3.2 Modelo Gráfico de Cruce .....	55
Figura 3.3 Esquema de Mutación de la posición 16 y 10 en la población X1.....	58
Figura 3.4 Esquema de Mutación de la posición 6 y 12 de la población X2.....	58
Figura 3.5 Esquema de Mutación de la posición 3 y 7 de la población X3.....	58
Figura 3.6 Esquema de Mutación de la posición 6 y 10 de la población X4.....	59
Figura 4.1 Eficacia Momento .....	67
Figura 4.2. Eficacia Cortante .....	67
Figura 4.3 Eficacia Momento .....	68
Figura 4.4. Cortante .....	68
Figura 4.5. Precio de viga convencional vs Algoritmo Genético .....	69
Figura 4.6. Precio de viga convencional vs Algoritmo Genético .....	70
Figura 4.7. Comparación Método Convencional Vs Algoritmo Genético .....	73
Figura 4.8. Comportamiento Algoritmo Genético.....	75
Figura 6.1. Esquema de un Algoritmo Genético Básico.....	83
Figura 6.2. Pantalla Principal del programa DEAC-TB .....	84
Figura 6.3. Grupo de Selección.....	84
Figura 6.4 Página Control de Datos.....	85
Figura 6.5. Ajuste del rango de generación de poblaciones. ....	85
Figura 6.6. Datos del diseño del perfil de acero .....	86
Figura 6.7. Esfuerzo, Fluencia y Material del perfil de acero.....	86
Figura 6.8. Datos de Cargas y Momentos.....	87
Figura 6.9 Número de Individuos .....	87
Figura 6.10 Porcentaje de Mutación .....	87
Figura 6.11 Datos de costo de materiales del perfil de acero .....	88

Figura 6.12 Botón Generar .....	88
Figura 6.13. Visualización de Resultados.....	88

## ÍNDICE DE TABLAS

Tabla 3.1 Cuadro de Operacionalización de Variables.....	42
Tabla 4.1. Resultados Comparación Método Convencional Vs Algoritmo Genético .....	72
Tabla 4.2. Resumen de Resultados .....	73
Tabla 4.3. Ejecuciones Método con Algoritmo Genético .....	74

## **RESUMEN**

El presente trabajo es una investigación de carácter teórico práctico, en la que se sistematiza el proceso de aplicación de un algoritmo genético en el diseño y optimización de vigas-trabe tipo I armadas mediante placas de acero, basado en la normativa del AISC-LRFD.

Se establecen los parámetros tanto de la técnica de algoritmos genéticos como el diseño de vigas-trabe tipo I, y de las dos en conjunto, obteniéndose como resultado un proceso de optimización.

Para la comprobación de la hipótesis, se analizan dos procesos; el primero en el cual se verifica la eficacia del diseño y en segundo lugar la eficiencia estructural y económica.

Se efectúa un análisis comparativo de los resultados generados por el software DEAC-TB creado como propuesta de esta investigación, entre el método de diseño de control tradicional con el diseño que aplica algoritmos genéticos, alcanzando un alto grado de eficiencia de esta técnica de optimización en los aspectos estructural y económico, consiguiendo como conclusión general que los algoritmos genéticos aplicados en forma adecuada al diseño AISC-LRFD de vigas-trabe, dan como resultado elementos óptimos de dimensiones económicas y que cumplen con la normativa establecida.



UNIVERSIDAD NACIONAL DE CHIMBORAZO  
FACULTAD DE INGENIERIA  
CENTRO DE IDIOMAS



Lic. Edison Salazar

02 de Junio de 2015

**SUMMARY**

This work is a theoretical and practical research which systematizes the process of applying a genetic algorithm in the design and optimization of plate-girder type I assembled with steel plates based on AISC-LRFD regulations.

The genetic algorithm technique and the plate-girder type I design parameters are established, and both at the same time, obtaining as a result an optimization process.

For the hypothesis testing, two processes were carried out; the first one in which the design efficiency is checked and the second one to check the structural and economic efficiency.

A comparative results analysis of the DEAC-TB, software created as a proposal, is carried out, between the method of traditional control design and the design that applies genetic algorithms, reaching a high efficiency degree with this optimization technique in economic and structural aspects, getting to the general conclusion that genetic algorithms applied properly to AISC-LRFD design of plate-girder result in optimal economic dimensions elements and meet with the established regulations.



## **CAPITULO 1**

### **1. MARCO REFERENCIAL**

#### **1.1. INTRODUCCIÓN**

La práctica del sistema de diseño estructural en la actualidad, en la mayor parte de calculistas, está basado por lo general y principalmente en el cumplimiento de las normas establecidas por los códigos correspondientes, lo cual no significa que no exista un cierto criterio de economía y ciertos aspectos que tienen influencia directa en la mayor proporción de la experiencia individual, y tampoco, el hecho de que se cumplan las normas significa que los elementos diseñados son los más convenientes económicamente hablando. Esto tiene sustento en varios estudios previos de otros autores, en los cuales se refleja que en nuestro medio, el cumplimiento de las normas no es una práctica común. El presente estudio pretende demostrar que cumpliendo los requisitos de diseño impuestos en los códigos y mediante la aplicación de técnicas no convencionales es posible obtener elementos eficientes en el aspecto estructural y económico, ya que el diseño estructural y en la mayoría de problemas de diseño se pueden obtener mediante la variación combinatoria de sus dimensiones una diversidad de soluciones validas en el aspecto estructural pero de todas ellas solo pocas o en ciertos casos solo una es la que genera la solución a menor costo. Esto es fundamental en un mundo competitivo y de creciente desarrollo como en el que nos desenvolvemos, en el que el ahorro de recursos cumple un papel importante y al explorar y experimentar nuevas técnicas y otras poco difundidas, se fomenta la innovación y el desarrollo, en este caso de la ingeniería, que es el campo desde donde se aportar a la sociedad.

#### **1.2. PROBLEMATIZACIÓN**

##### **1.2.1. IDENTIFICACIÓN Y DESCRIPCIÓN DEL PROBLEMA.**

En la actualidad en el Ecuador, la implementación de vigas-trabe para el diseño en edificaciones no se ha perfeccionado como una solución para edificaciones con cargas y

luces de importancia. El diseño de vigas-trabe se desarrolla a través de un método iterativo, en el cual se parte seleccionando las dimensiones básicas de la peralte, ala, alma; se seleccionan las propiedades del acero se selecciona el tamaño de la viga, se calcula las longitudes efectivas respecto a ambos ejes principales, a continuación se efectúa la revisión de cada caso de diseño y se analiza si está dentro del rango establecido, de lo contrario existen dos casos, el primero en el cual el elemento está sobredimensionado y el segundo en el cual esta sub dimensionado para lo cual se debe regresar al proceso inicial redimensionando de la sección aumentando o disminuyendo la sección de la misma según sea el caso, en este tipo de diseño tradicional no se considera el diseño óptimo. Los métodos clásicos de diseño estructural para vigas no permiten realizar un análisis multiobjetivo para entregar un elemento óptimo tanto en las secciones de los materiales utilizados como en el costo que conllevaría el uso de dicho dimensionamiento.

Con estos antecedentes se plantea la presente investigación titulada “Optimización de vigas-trabe tipo I armadas mediante placas de acero utilizando algoritmos genéticos”, la cual es factible realizar pues se dispone del tiempo y los recursos para su desarrollo.

### 1.2.2. LIMITACIONES

El proyecto de investigación se centrará en el diseño de vigas-trabe tipo I armadas mediante placas de acero de ejes simétricos sujetas a momentos y cortantes que resultaran de un diseño estructural previo.

Basado en la normativa impuesta por el Load and Resistance Factor Design for Estructural Steel Building (LRFD) de la AISC (American Institute of Steel Construction).

Dentro de las limitaciones marcadas por el LRFD para este tipo de vigas-trabe se presentan las siguientes:

- La condición más importante que debe cumplir un trabe para ser diseñado como tal es :

$$\frac{h}{tw} \geq \frac{970}{\sqrt{F_y}}$$

Caso contrario deberá ser diseñado como viga de acero.



Se empleará la técnica de optimización del algoritmo genético simple.

### **1.2.3. FORMULACIÓN DEL PROBLEMA**

¿Es posible diseñar vigas-trabe tipo I armadas mediante placas de acero utilizando algoritmos genéticos, en menor tiempo y más eficiencia, garantizando la optimización de la sección?

## **1.3. OBJETIVOS**

### **1.3.1. GENERAL**

Realizar un diseño de viga-trabe tipo I armada mediante placas de acero utilizando algoritmos genéticos.

### **1.3.2. ESPECÍFICOS**

- Comparar la efectividad del método propuesto que aplica algoritmos genéticos con el método tradicional LRFD.
- Estudiar los fundamentos conceptuales de los algoritmos genéticos y sus aplicaciones.
- Investigar los parámetros necesarios para la aplicación de un algoritmo genético en el diseño de vigas-trabe de acero.
- Desarrollar el proceso para la implementación del algoritmo genético en el diseño de vigas-trabe de acero.
- Comparar los resultados obtenidos mediante la aplicación del diseño por el método tradicional LRFD con el diseño que aplica algoritmos genéticos y su incidencia económica y estructural.

## **1.4. JUSTIFICACIÓN**

El avance tecnológico, el desarrollo de la ciencia y la globalización, entre otros eventos que crecen aceleradamente en la actualidad, nos obligan a ser cada vez más competitivos, el diseño estructural y en especial el diseño de vigas las cuales deben cumplir con los requisitos establecidos por las normas para garantizar su funcionalidad y economía. Según

el plan nacional del buen vivir, la inversión pública generara condiciones para fomentar la competitividad sistémica: Cualificación del talento humano, innovación de productos y procesos, ampliación y mejoramiento de la infraestructura de soporte (carreteras, conectividad, logística, entre otros).

Los contenidos de esta investigación desde los aspectos teóricos y prácticos servirán para optimizar recursos, ya que el software reducirá el tiempo de cálculo además de proporcionar secciones estructurales de mayor desempeño.

El presente proyecto pretende realizar una comparación entre el método propuesto con algoritmos genéticos y el método tradicional LRFD, para posteriormente estudiar los fundamentos conceptuales de los algoritmos genéticos y sus aplicaciones, Además de la indagación de los parámetros necesarios para la aplicación de un algoritmo genético en el diseño de vigas-trabe de acero y finalmente el desarrollo del proceso para la implementación del algoritmo genético en el diseño de vigas-trabe de acero, además de la comparación los resultados obtenidos mediante la aplicación del diseño por el método tradicional LRFD con el diseño que aplica algoritmos genéticos.

Los resultados de esta investigación serán de utilidad para profesionales de ingeniería civil, empresas, municipios, docentes, estudiantes, concejos provinciales, constructoras.

Con el desarrollo de esta investigación se pretende diseñar vigas trabe con mejor estructuración de sus elementos, mejores características geométricas y menor costo, en las cuales se podrá maximizar el desempeño estructural, el cumplimiento de la normativa y se minimizarán factores tales como el peso y el costo.

Esta investigación aplicará la teoría de los algoritmos genéticos al diseño de vigas-trabe armadas siguiendo la normativa impuesta por la AISC, método del LRFD.

Con la introducción en nuestro medio de este tipo de técnicas que aportarán al desarrollo del diseño estructural, se abrirá la puerta a futuras investigaciones en este campo, y no solo en el diseño estructural sino que además en diferentes ramas.

## CAPITULO II

### 2. MARCO TEÓRICO

#### 2.1.1. ANTECEDENTES

El diseño estructural en las últimas décadas se ha desarrollado aceleradamente, gracias al avance tecnológico así como de las normativas, la informática ha sido un aporte fundamental ya que mediante la misma se ha reducido el tiempo invertido en el diseño y se ha incrementado considerablemente productos finales de mayor calidad dado que:

- Las vigas principalmente trabajan a esfuerzos de flexión y el acero es un material cuya propiedad es la resistencia a la flexión.
- El acero proporciona mayor ductilidad a la estructura.<sup>1</sup>

La viga trabe tipo I al ser un elemento de patín grueso y alma delgada es capaz de soportar cargas considerablemente mayores que las vigas de acero convencionales para lo cual, los traveses son utilizados en casos especiales.

Los algoritmos genéticos son métodos adaptativos que puede usarse para resolver problemas de búsqueda y optimización los cuales están basados en el proceso genético de los organismos vivos, utilizan una analogía directa del fenómeno de evolución en la naturaleza. Trabajan con una población de individuos, cada uno representado una posible solución a un problema dado. A los más aptos se les da la oportunidad de reproducirse mediante cruzamientos con otros individuos de la población, produciendo descendiente con características de ambos padres. Los miembros menos adaptados poseen pocas posibilidades de que sean seleccionados para la reproducción y desaparecen. De esta manera el Algoritmo tenderá a optimizar los individuos de cada especie en las circunstancias actuales.

### 2.2. FUNDAMENTACION TEORICA

#### 2.2.1. DISEÑO DE VIGAS ARMADAS (Plate Girders)

---

<sup>1</sup> Borja, Esteban-VALLEJO, Vicente. Diseño de Columnas Mixtas. Escuela Politécnica Nacional 1986

Las vigas armadas o plate-girders, son secciones de gran peralte que se construyen mediante la unión de placas soldadas, remachadas, con el objetivo de obtener un elemento de mayor capacidad que las secciones estándares de uso común.

Principalmente son utilizadas en puentes de acero de considerable extensión en estructuras con fuertes cargas concentradas, o para vigas que necesiten cubrir luces muy grandes.<sup>2</sup>

El criterio para diferenciar entre vigas soldadas y plate-girders es el parámetro de esbeltez del alma  $\lambda r = \frac{970}{\sqrt{F_y}}$ . Secciones con un parámetro de esbeltez  $\lambda \leq \frac{970}{\sqrt{F_y}}$  son consideradas como vigas y se diseñan siguiendo los lineamientos para perfiles estándares, con la salvedad de que para secciones soldadas el esfuerzo residual que debe utilizarse es 16.5 ksi en lugar de 10 ksi.

En los “plate-girders” el parámetro de esbeltez del alma  $\lambda > \frac{970}{\sqrt{F_y}}$  hace necesario considerar los criterios especiales para el diseño de estos elementos.

Las vigas armadas están sometidas a los mismos estados límites que las secciones laminadas, solo que para evaluarlos hay que tener el cuidado de incluir las modificaciones pertinentes presentadas en el manual AISC-LRFD<sup>3</sup>.

### **2.2.1.1. ESTADOS LÍMITES POR FLEXIÓN.**

Los estados límites por flexión para una viga armada son: cedencia del ala en tensión, pandeo local del ala y pandeo lateral torsional. La resistencia a la flexión del elemento estará gobernada por el menor valor calculado para todos los estados límites de falla.

#### **2.2.1.1.1. CEDENCIA DEL ALA EN TENSIÓN**

Este estado límite se presenta cuando el ala en tensión desarrolla cedencia antes de que la viga presente algún tipo de inestabilidad por compresión.

---

<sup>2</sup> Theodore V. Galambos. **Diseño de estructuras de acero con LRFD. 1999**

<sup>3</sup> TAPIA, Édison. **Diseño de Estructuras en Acero. Universidad Nacional de Chimborazo 2011**

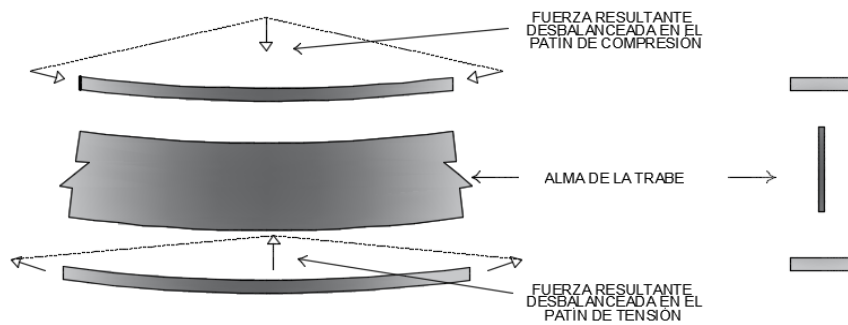


Figura 2.1. Compresión vertical en el alma debido a Fuerzas desbalanceadas en los patines

Su capacidad es igual a

$$M_{nt} = S_{xt} * R_e * F_{yft} \quad (2.2.1)$$

Donde:

$S_{xt}$  = Módulo Elástico a Tensión

$R_e$  = Coeficiente (2.2.2)

$F_{yft}$  = Esfuerzo de Fluencia del ala a tensión

$$R_e = \frac{12 + \frac{A_w}{A_{ft}}(3m - m^3)}{12 + 2 * \frac{A_w}{A_{ft}}} \leq 1.00 \quad (2.2.2)$$

Donde:

$m$  = Relación entre el esfuerzo de fluencia del alma y el esfuerzo de fluencia del patín.

$$m = \frac{F_{yw}}{F_{yt}} \quad (2.2.3)$$

$A_w$  = Área del alma

$A_{ft}$  = Área del patín a tensión.

### 2.2.1.1.2. PANDEO LOCAL DEL ALA.

Este tipo de inestabilidad se presenta por bifurcación longitudinal del ala en compresión trabajando como un elemento no atiesado, antes de que el ala en tensión pueda desarrollar cedencia. <sup>4</sup>El criterio que rige este estado límite es el parámetro de esbeltez del ala en compresión,  $\lambda = \frac{b_f}{2t_f}$ .

El pandeo local puede ser inelástico o elástico, dependiendo del parámetro de esbeltez y de los límites  $\lambda_p = \frac{65}{\sqrt{Fy_{fc}}}$  y  $\lambda_r = \frac{230}{\sqrt{\frac{Fy_{fc}}{K_c}}}$ .

Donde:

$Fy_{fc}$  = Esfuerzo de fluencia del ala a compresión.

$$K_c = \frac{4}{\sqrt{\frac{h}{tw}}} \quad (0.35 \leq K_c \leq 0.763) \quad (2.2.4)$$

**Pandeo Local Inelástico** ( $\lambda_p < \lambda \leq \lambda_r$ ): Algunas fibras de la sección transversal de la viga han alcanzado el esfuerzo de cedencia al momento de la inestabilidad, mientras el resto se encuentra en el régimen elástico. En esta zona el esfuerzo de compresión en el ala está dado por:

$$Fcr_{FLB} = Fy_{fc} \left[ 1 - \frac{1}{2} \left( \frac{\lambda - \lambda_p}{\lambda_r - \lambda_p} \right) \right] \quad (2.2.5)$$

**Pandeo Local Elástico** ( $\lambda_r < \lambda$ ): Todas las fibras de la sección transversal de la viga están sometidas a un esfuerzo menor que el esfuerzo de cedencia al momento de la inestabilidad. En otras palabras, este pandeo se presenta completamente en el régimen elástico. En esta zona el esfuerzo de compresión del ala está dado por

$$Fcr_{FLB} = \frac{C_{PG}}{\lambda^2} \leq Fy_{fc} \quad , \quad C_{PG} = 26200K_c$$

---

<sup>4</sup> TAPIA, Édison. **Diseño de Estructuras en Acero. Universidad Nacional de Chimborazo 2011**

### 2.2.1.1.3. PANDEO LATERAL TORSIONAL

Este estado límite se presenta cuando el ala en compresión se bifurca lateralmente con rotación. Este tipo de inestabilidad es muy común en vigas de acero de sección abierta, ya que las mismas tienen poca capacidad en torsión.

El criterio que rige el pandeo lateral torsional en una viga armada es la longitud libre entre arriostramientos del ala en compresión,  $L_b$ . Arriostrar de forma correcta el ala en compresión de una viga evita el pandeo lateral torsional, y permite que la sección pueda desarrollar su capacidad en el ala en tensión.

Al igual que el pandeo local el pandeo lateral torsional puede ser inelástico o elástico, dependiendo de los parámetros  $\lambda = \frac{L_b}{r_T}$ ,  $\lambda_p = \frac{300}{\sqrt{F_y f_c}}$ ,  $\lambda_r = \frac{756}{\sqrt{F_y f_c}}$ .

En el cual  $r_T$  es el radio de giro del ala en compresión alrededor del eje y más un tercio de la parte del alma que actúa en compresión.<sup>5</sup>

**Pandeo Lateral Torsional Inelástico** ( $\lambda_p < \lambda \leq \lambda_r$ ): Al momento de la inestabilidad algunas fibras de la sección transversal de la viga han alcanzado el esfuerzo de cedencia, mientras el resto se encuentra en el régimen elástico. En esta zona el esfuerzo de compresión en la sección está dado por

$$F_c r_{FLB} = C_b F_y f_c \left[ 1 - \frac{1}{2} \left( \frac{\lambda - \lambda_p}{\lambda_r - \lambda_p} \right) \right] \quad (2.2.5)$$

El factor  $C_b$  toma en cuenta variaciones en el diagrama de momento. Las variaciones en el diagrama de momento ocasionan que los esfuerzos de compresión varíen también, por lo cual el código permite incrementar la resistencia por el factor  $C_b$  para vigas sometidas a pandeo lateral torsional inelástico o elástico. Este incremento tiene un límite, ya que la viga no puede

---

<sup>5</sup> Theodore V. Galambos. **Diseño de estructuras de acero con LRFD. 1999**

desarrollar un esfuerzo mayor que su esfuerzo de cedencia, es decir, la capacidad de la viga no puede exceder su momento plástico.

**Pandeo Lateral Torsional Elástico** ( $\lambda_r - \lambda$ ): Al momento de la inestabilidad todas las fibras de la sección transversal de la viga están sometidas a un esfuerzo menor que el esfuerzo de cedencia. En otras palabras, este pandeo se presenta completamente en el régimen elástico. En esta zona el esfuerzo de compresión del ala está dado por

$$F_{cr_{FLB}} = \frac{C_{PG}}{\lambda^2}, \quad C_{PG} = 286000 C_b$$

La capacidad en flexión se verifica a partir del estado límite que gobierna entre cedencia, pandeo local del ala y pandeo lateral torsional. Para los esfuerzos en compresión la capacidad en flexión se toma como

$$M_{nc} = S_{cx} * R_{PG} * R_e * F_{cr} \longrightarrow \text{Momento Nominal a compresión} \quad (2.2.6)$$

En donde

$$F_{cr} = \min(F_{cr_{FLB}}, F_{cr_{LTB}}) \quad (2.2.7)$$

$$R_{PG} = 1 - \left[ \left( \frac{a_r}{1200 + 300a_r} \right) \left( \frac{h_c}{t_w} - \frac{972}{\sqrt{F_{cr}}} \right) \right] \leq 1.0 \quad (2.2.8)$$

$$a_r = \frac{A_w}{A_f} \leq 10.0 \quad (2.2.9)$$

$$R_e = \frac{12 + a_r(3m - m^3)}{12 + 2a_r} \leq 1.0 \quad (2.2.2)$$

$$m = \frac{F_{yw}}{F_{cr}} \quad (2.2.3)$$

La capacidad final en flexión de la viga se toma como

$$M_n = \min(M_{nt}, M_{nc})$$



### 2.2.1.2. ESTADO LÍMITES POR CORTANTE.

Las vigas armadas son secciones delgadas de paredes abiertas, en las cuales los esfuerzos normales ocasionados por la flexión son resistidos principalmente por las alas, y los esfuerzos cortantes son resistidos por el alma. De esta manera, los estados límites a ser verificados dependerán de la capacidad del alma de la viga a la cedencia y al pandeo local inelástico o elástico.

Por otro lado, si la viga se encuentra rigidizada mediante placas regularmente espaciadas en toda su extensión su comportamiento al cortante será mejor que el de una viga sin rigidizadores. Esto se debe a que la viga rigidizada funcionará de manera similar a una cercha, donde los rigidizadores trabajarán como los elementos verticales en compresión mientras que el alma trabajará como las diagonales en tensión. Es importante señalar que esta consideración no se aplica para los paneles en los apoyos. La analogía de la cercha la podemos observar en la figura 2.2.

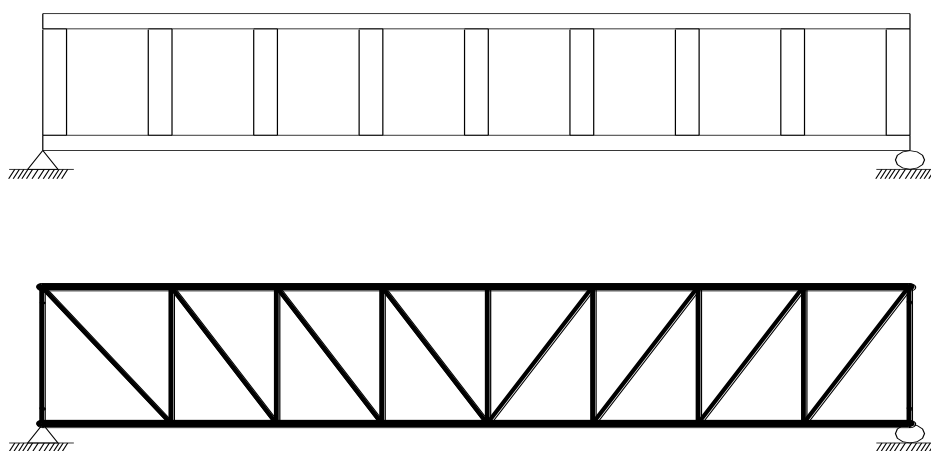


Figura 2.2. Analogía de Plate-Girder rigidizado con cercha

Los estados límites por cortante dependerán no sólo si se trata de una viga rigidizada o no, sino también de los parámetros de esbeltez del alma.

$$\lambda = \frac{h}{t_w} \quad , \quad \lambda_p = 187 \sqrt{\frac{k_v}{F_{yw}}} \quad y \quad \lambda_r = 234 \sqrt{\frac{k_v}{F_{yw}}} \quad (2.2.10)$$

### 2.2.1.2.1. CEDENCIA DEL ALMA

Este estado límite se presenta cuando el alma alcanza el esfuerzo de cedencia antes de que se presente algún tipo de inestabilidad por compresión.

Su capacidad es igual a  $V_n = 0.6 A_w Fy_w$

### 2.2.1.2.2. PANDEO INELÁSTICO DEL ALMA ( $\lambda_p < \lambda \leq \lambda_r$ ).

En este tipo de falla parte del alma cuando esta ha alcanzado el esfuerzo de cedencia, mientras que el resto se encuentra en el régimen elástico al momento del pandeo. En este caso el

parámetro para tomar en cuenta la inestabilidad  $C_v$ , se calcula como  $C_v = \frac{187 \sqrt{\frac{k_v}{Fy_w}}}{\frac{h}{t_w}}$

### 2.2.1.2.3. PANDEO ELÁSTICO DEL ALMA ( $\lambda > \lambda_r$ )

El alma se pandea en el régimen elástico, es decir, sin que ninguna fibra alcance el esfuerzo de cedencia. El parámetro  $C_v$  en este estado limite se calcula como  $C_v = \frac{44000 k_v}{\left(\frac{h}{t_w}\right)^2 Fy_w}$ .

Tanto para el pandeo inelástico como para el pandeo elástico la resistencia al cortante se calcula con la misma expresión. Sin embargo, existen fórmulas diferentes para vigas con rigidizadores y vigas sin rigidizadores. La capacidad al cortante para vigas con rigidizadores se calcula como:

$$V_n = 0.6 A_w Fy_w \left[ C_v + \frac{1-C_v}{1.15 \sqrt{1+\left(\frac{a}{h}\right)^2}} \right] \quad (2.2.11)$$

Para vigas sin rigidizadores la capacidad se calcula como

$$V_n = 0.6 A_w Fy_w C_v \quad (2.2.12)$$

Además, si se cumple que en un mismo punto  $0.75 \phi_b M_n \leq M_u \leq \phi_b M_n$  y  $0.6 \phi_v V_n \leq V_u \leq \phi_v V_n$ , se debe verificar la siguiente ecuación de interacción:

$$\frac{M_u}{\phi_b M_n} + 0.625 \frac{V_u}{\phi_v V_n} \leq 1.375$$

Para una viga simplemente apoyada con carga uniformemente distribuida, se puede demostrar que el punto donde ocurre el máximo valor de la ecuación de interacción es:

$$x = \frac{L}{2} - \left(\frac{5}{8}\right) \frac{\phi_b M_n}{\phi_v V_n} \quad (2.2.12)$$

Reemplazando este valor en la expresión anterior, obtenemos el valor máximo que puede alcanzar la ecuación de interacción en una viga simplemente apoyada con carga uniformemente distribuida.

$$f_{max} = \frac{M_u}{\phi_b M_n} + \left(\frac{25}{256}\right) \frac{\left(\frac{V_u}{\phi_v V_n}\right)^2}{\frac{M_u}{\phi_b M_n}} \quad (2.2.13)$$

## 2.3. ALGORITMOS GENÉTICOS (AG)<sup>6</sup>

### 2.3.1. INTRODUCCIÓN A LOS AG

La capacidad del ser humano para predecir el comportamiento de su entorno, se ha ido incrementando con el paso del tiempo. De igual modo, ha comprendido que, si bien era capaz de controlar muchos aspectos de su vida, y su interacción con lo que le rodeaba, no lo era para otros tantos.

La inteligencia artificial es responsable de muchos de esos logros. Los pioneros de esta ciencia estaban tan interesados en la electrónica, como en la biología, y por eso sus aplicaciones iban desde calcular trayectorias de misiles, a tratar de modelizar el cerebro, de imitar el proceso de aprendizaje humano, y de simular la evolución biológica.

---

<sup>6</sup> GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning

Los años ochenta marcan el florecimiento del interés de la comunidad científica por estos temas computacionales inspirados en la biología, que han visto como su desarrollo les llevaba a cotas inimaginables, primero en el campo de las Redes Neuronales, luego en el del Aprendizaje, y por último en lo que ahora se conoce como “Computación Evolutiva”, de la que los algoritmos genéticos constituyen su máximo exponente.

La primera mención del término, y la primera publicación sobre una aplicación del mismo, se deben a Bagley (1967), que diseñó algoritmos genéticos para buscar conjuntos de parámetros en funciones de evaluación de juegos, y los comparó con los algoritmos de correlación, procedimientos de aprendizaje modelizados después de los algoritmos de pesos variantes de ese periodo. Pero es otro científico el considerado creador de los Algoritmos Genéticos: John Holland, que los desarrolló, junto a sus alumnos y colegas, durante las décadas de 1960 y 1970. En contraste con las estrategias evolutivas y la programación evolutiva, el propósito original de Holland no era diseñar algoritmos para resolver problemas concretos, sino estudiar, de un modo formal, el fenómeno de la adaptación tal y como ocurre en la naturaleza, y desarrollar vías de extrapolar esos mecanismos de adaptación natural a los sistemas computacionales. La mayor innovación de Holland fue la de introducir un algoritmo basado en poblaciones con cruces, mutaciones e inversiones. Es más, Holland fue el primero en intentar colocar la computación evolutiva sobre una base teórica firme (Holland, 1975).

Hasta hace poco, esta base teórica, fundamentada en la noción de “esquemas”, fue la estructura sobre la que se edificaron la mayoría de los trabajos teóricos sobre algoritmos genéticos en las décadas siguientes.

En estos últimos años se ha generado una amplia interacción entre los investigadores de varios métodos de computación evolutiva, rompiéndose las fronteras entre algoritmos genéticos, estrategias evolutivas y programación evolutiva.<sup>7</sup>

Con estos antecedentes, es importante mencionar que su uso ha sido bastante difundido como herramienta de optimización en varios ámbitos como la economía, la medicina, la ingeniería y la Aeronáutica, entre los principales.

### **2.3.2. DEFINICIÓN DE ALGORITMOS GENÉTICOS (AG)**

Los AG son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización. Están basados en el proceso genético de los organismos vivos. A lo largo de las

---

<sup>7</sup> Piedad Tolmos Rodríguez-Piñero, Introducción a los algoritmos genéticos y sus aplicaciones.

generaciones, las poblaciones evolucionan en la naturaleza de acuerdo con los principios de la selección natural y la supervivencia de los adaptados, postulados por Darwin (1859). Por imitación de este proceso, los AG son capaces de ir creando soluciones para problemas del mundo real. La evolución de dichas soluciones hacia valores óptimos del problema depende en buena medida de una adecuada codificación de las mismas.

En la naturaleza los individuos de una población compiten entre sí en la búsqueda de recursos tales como comida, agua y refugio. Incluso los miembros de una misma especie compiten a menudo en la búsqueda de un compañero. Aquellos individuos que tiene más éxito en sobrevivir y en atraer compañeros tienen mayor probabilidad de generar un gran número de descendientes. Por el contrario individuos poco dotados producirán un menor número de descendientes. Esto significa que los genes de los individuos mejor adaptados se propagaran en sucesivas generaciones hacia un número de individuos creciente. La combinación de buenas características provenientes de diferentes ancestros, puede a veces producir súper-individuos, cuya adaptación es mucho mayor que la de cualquiera de sus ancestros. De esta manera, las especies evolucionaran logrando unas características cada vez mejor adaptadas al entorno en el que viven.

### **2.3.3. ALGORITMOS GENÉTICOS EN LA OPTIMIZACIÓN<sup>8</sup>**

La razón del creciente interés por los AG es que estos son un método global y robusto de búsqueda de las soluciones de problemas. La principal ventaja de estas características es el equilibrio alcanzado entre la eficiencia y eficacia para resolver diferentes y muy complejos problemas de grandes dimensiones.

Lo que aventaja a los AG frente a otros algoritmos tradicionales de búsqueda es que se diferencian de estos en los siguientes aspectos:

Trabajan con una codificación de un conjunto de parámetros, no con los parámetros mismos. Trabajan con un conjunto de puntos, no con un único punto y su entorno (su técnica de búsqueda es global). Utilizan un subconjunto del espacio total, para obtener información de universo de búsqueda, a través de las evaluaciones de la función a optimizar. Esas evaluaciones se emplean de forma eficiente para clasificar los subconjuntos de acuerdo con su idoneidad.

No necesitan conocimientos sobre el problema a resolver; es decir, no están sujetos a restricciones. Por ejemplo, se pueden aplicar a funciones no continuas, lo cual abre un amplio campo de aplicaciones que no podrían ser tratadas por los métodos tradicionales.

---

<sup>8</sup> GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning

Utilizan operadores probabilísticos, en vez de los típicos operadores determinísticos de las técnicas tradicionales.

Resulta sumamente fácil ejecutarlos en las modernas arquitecturas masivas en paralelo.

Cuando se usan para problemas de optimización, resultan menos afectados por los máximos locales de las técnicas tradicionales.

#### **2.3.4. VENTAJAS Y DESVENTAJAS DE LOS AG**

##### **2.3.4.1. VENTAJAS**

- No necesitan conocimientos específicos sobre el problema que intentan resolver.
- Operan de forma simultánea con varias soluciones, en vez de trabajar de forma secuencial como las técnicas tradicionales.
- Cuando se usan para problemas de optimización, los resultados con una función objetivo tienden a presentar menor número de falsas soluciones que con las técnicas tradicionales.
- Usan operadores probabilísticos, en vez de los típicos operadores determinísticos de las otras técnicas.

##### **2.3.4.2. DESVENTAJAS**

- Pueden tardar mucho en converger, o no converger en absoluto, dependiendo en cierta medida de los parámetros que se utilicen tamaño de la población, número de generaciones, etc.
- Pueden converger prematuramente debido a una serie de problemas de diversa índole.

#### **2.3.5. LIMITACIONES DE LOS AG<sup>9</sup>**

El poder de los Algoritmos Genéticos provienen del hecho de que se trata de un técnica robusta, y pueden tratar con éxito una gran variedad de problemas provenientes de diferentes áreas, incluyendo aquellos en los que otros métodos encuentran dificultades.

---

<sup>9</sup> GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning

Si bien no se garantiza que el Algoritmo Genético encuentre la solución óptima, del problema, existe evidencia empírica de que se encuentran soluciones de un nivel aceptable, en un tiempo competitivo con el resto de algoritmos de optimización combinatoria. En el caso de que existan técnicas especializadas para resolver un determinado problema, lo más probable es que superen al Algoritmo Genético, tanto en rapidez como en eficacia. El gran campo de aplicación de los Algoritmos Genéticos se relaciona con aquellos problemas para los cuales no existen técnicas especializadas. Incluso en el caso en que dichas técnicas existan, y funcionen bien, pueden efectuarse mejoras de las mismas hibridándolas con los Algoritmos Genéticos.

### **2.3.6. CONDICIONES PARA LA APLICACIÓN DE UN ALGORITMO GENÉTICO<sup>10</sup>.**

La aplicación más común de los Algoritmos Genéticos ha sido la solución de problemas de optimización, en donde han demostrado ser muy eficientes y confiables. Sin embargo no todos los problemas pudieran ser más apropiados para la técnica, y se recomienda en general tomar en cuenta las siguientes características del mismo antes de intentar usarla:

- Su espacio de búsqueda debe estar delimitado dentro de un cierto rango.
- Debe poderse definir una función de aptitud que indique que tan buena o mala es una cierta respuesta.
- Las soluciones debe codificarse de una forma que resulte relativamente fácil de implementar en la computadora.

El primer punto es más importante, y lo más recomendable es intentar resolver problemas que tengan espacios de búsqueda discretos aunque éstos sean muy grandes. Sin embargo, también podrá intentarse usar la técnica con espacios de búsqueda continuos, pero preferentemente cuando exista un rango de soluciones relativamente pequeño.

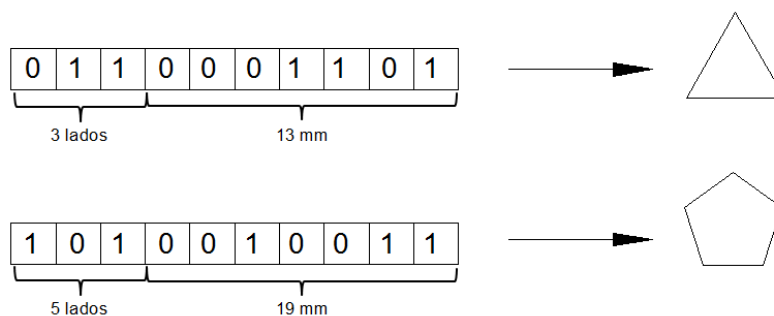
La función de aptitud no es más que la función objetivo de nuestro problema de optimización. El algoritmo genético únicamente maximiza, pero la minimización puede realizarse fácilmente

---

<sup>10</sup> Diseño a flexión de vigas rectangulares de hormigón armado basado en la normativa ecuatoriana y mediante la aplicación de algoritmos genéticos , 2011, Felix Armando Ashqui Laguna

utilizando el recíproco de la función maximizante (debe cuidarse, por supuesto, que el recíproco de la función no genere una división por cero). Una característica que debe tener esta función es que tiene que ser capaz de penalizar a las soluciones, y de gratificar a las buenas, de forma que sean estas últimas las que se propaguen con mayor rapidez.

La codificación más común de las soluciones es a través de cadenas binarias, aunque se han utilizado también números reales y letras. El primero de estos esquemas ha gozado de mucha popularidad debido a que es el que propuso originalmente Holland, y además porque resulta muy sencillo de implementar.



*Figura 2.3. Codificación de individuos en AG*  
*Fuente: "Genetic Algorithms-in Search, Optimization and machine learning".*  
*GOLDBERG, David E. (1989) pag. 85*

### 2.3.7. MARCO DE DESARROLLO DE LOS AG<sup>11</sup>

Antes de continuar ahondando en la técnica de los Algoritmos Genéticos sería interesante dejarla situada dentro de un marco más amplio. Esto hace referencia a la rama de la inteligencia Artificial que se ha denominado computación evolutiva.

El término computación evolutiva se refiere al estudio de los fundamentos y aplicaciones de ciertas técnicas heurísticas (basadas en la experiencia) de búsqueda, fundamentadas en los principios naturales de la evolución. Una gran variedad de algoritmos evolutivos han sido propuestos pero principalmente pueden clasificarse en: Algoritmos Genéticos, Programación Evolutiva, Estrategias Evolutivas, Sistemas Clasificadores y Programación Genética. Esta clasificación se basa sobre todo en detalles de desarrollo histórico más que en el hecho de un funcionamiento realmente diferente, de hecho las bases biológicas en las que se apoyan son esencialmente las mismas. Las diferencias entre ellos se centran en los operadores que se usan

<sup>11</sup> GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning



en cada caso y en general en la forma de implementar la selección, reproducción y sustitución de individuos en una población.

Aunque los detalles de la evolución no han sido completamente comprendidos, incluso hoy en día, existen algunos puntos en los que fundamentan:

- La evolución es un proceso que opera a nivel de cromosomas, y no a nivel de individuos. Cada individuo es codificado como un conjunto de cromosomas.
- La selección natural es el mecanismo mediante el cual los individuos mejor adaptados son los que tienen mayores probabilidades de reproducirse.
- El proceso evolutivo tiene lugar en la etapa de la reproducción. Es en esta etapa donde se reproduce la mutación, que es la causante de que los cromosomas de los hijos puedan ser diferentes a los de los padres, y el cruce, que combina los cromosomas de los padres para que los hijos tengan cromosomas diferentes.

De forma breve, pasamos a comentar cada una de los algoritmos mencionados anteriormente, para que el lector pueda tener una idea de las similitudes y diferencias entre ellas.

Los Algoritmos Genéticos.- Resuelven los problemas generando poblaciones sucesivas a las que se aplican los operadores de mutación y cruce. Cada individuo representa una solución al problema, y se trata de encontrar al individuo que represente la mejor solución.

La Programación Genética.- Funciona igual que la técnica anterior pero se centra en el estudio de problemas cuya solución es un programa. De manera que los individuos de la población son programas que se acercan más o menos a realizar una tarea que es la solución.

La Programación Evolutiva.- Es otro enfoque de los algoritmos genéticos que imiten lo mejor posible a la naturaleza, en cada caso, más que la relación de los padres son su descendencia. En este caso no se utiliza el operador de cruce, tomando la máxima importancia el operador de mutación.

Estrategias Evolutivas.- Se centran en el estudio de problemas de optimización e incluyen una visión del aprendizaje en dos niveles: a nivel de genotipo, y a nivel de fenotipo.

Sistemas Clasificadores.- Engloban el estudio de problemas en los que la solución buscada se corresponde con toda una población.

## **2.3.8. COMPARACIÓN CON OTROS MÉTODOS DE OPTIMIZACIÓN<sup>12</sup>.**

### **2.3.8.1. ALGORITMOS GENÉTICOS Y MÉTODOS ENUMERATIVOS.**

Existe la posibilidad teórica de encontrar soluciones a problemas a optimización enumerando todas las soluciones posibles para todos los casos y posteriormente buscando la misma en la base de datos resultante. Los problemas se limitan entonces a un sistema de búsqueda eficiente de un caso en concreto. Por ejemplo los libros con tablas de logaritmos tradicionales constan de una larga serie de cálculos para todos los valores usuales. La solución consiste simplemente en buscar en la lista el número decimal y retornar el logaritmo dado.

La memorización de las tablas de multiplicar que se enseñan a los niños es otro ejemplo usual. Se espera que ante la pregunta ¿Cuánto es siete por cinco? Los niños respondan instantáneamente treinta y cinco sin tener que estar calculando mentalmente la multiplicación. Este método es factible siempre que el número de valores sea manejable. De otra manera el simple cálculo de los mismos se vuelve imposible.

La memorización de una serie de datos no es otra cosa que la construcción en la memoria del equivalente a una base de datos en donde se busca la pregunta y se encuentra automáticamente la respuesta.

Los Algoritmos genéticos usan técnicas heurísticas para la resolución del problema, lo cual limita drásticamente en número de datos a utilizar.

### **2.3.8.2. ALGORITMOS GENÉTICOS Y SISTEMAS EXPERTOS**

Un Sistema Experto es un programa de computadora que encuentra soluciones a problemas de tipo condicional con la estructura.

Los Sistemas Expertos tuvieron su apogeo en la década de los ochenta, aproximadamente de 1979 a 1985. En esa época se los llegó a considerar verdaderas panaceas que resolverían muchos de los problemas cotidianos del hombre. Incluso se formaron en ese entonces varias compañías con el objetivo específico de realizarlos y comercializarlos. Algunos fueron exitosos y funcionaron bien, pero las dificultades planteadas anteriormente no tardaron en aparecer. En particular:

---

<sup>12</sup> GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning

- Existen temas en los cuales el conocimiento no es estático, sino que la aparición de nueva información altera las pautas o reglas de inferencia de los resultados. La necesidad permanente de reevaluar las reglas por medio de expertos humanos lleva al sistema a una operatoria lenta y burocrática.
- Cada conocimiento nuevo implica reentrenar manualmente el sistema. Los Sistemas Expertos demostraron no ser útiles en este campo.
- Existen temas en los cuales la interrelación de ciertas variables no es conocida. Si la información disponible de cierto asunto es limitada, y no se conoce el comportamiento de alguna de sus variables, el sistema experto tendrá grandes dificultades de programarse ya que sus reglas serán impresas.

### **2.3.8.3. ALGORITMOS GENÉTICOS Y REDES NEURONALES**

Una Red Neuronal es el intento de poder realizar una simulación computacional del comportamiento de partes del cerebro humano mediante la réplica en pequeña escala de los patrones que éste desempeña para la formación de resultados a partir de los sucesos percibidos. El cerebro consta de unidades llamadas neuronas, las cuales están conectadas entre sí formando una red (de ahí la denominación red neuronal).

Concretamente, se trata de poder analizar y reproducir el mecanismo de aprendizaje de sucesos que poseen los animales más evolucionados.

La red simula grupos de neuronas, llamados capas las cuales están relacionadas unas con otras. Los datos se introducen en la primera capa, llamada capa de entradas, cada capa transfiere la información a sus vecinas, teniendo un peso o ponderación para los valores, lo que va modificando los mismos en su paso a través de la red.

Cuando los datos llegan a la última de las capas, llamada capa de salida el valor resultante es tomado como el resultado de la red. La red puede ser entrenada para diversos usos, entre ellos como mecanismo de optimización. En este sentido, se puede expresar que serían un modelo alternativo competitivo con los Algoritmos genéticos, si se las programa para este fin. En rigor de verdades, la literatura sugiere que se podrían hacer modelos mixtos o híbridos en donde se combinen las ventajas de las redes neuronales y los Algoritmos genéticos, aunque hay muy poco material disponible en este campo. Tal vez esto se deba al hecho que los AG y el estudio de las redes forman dos ramas o escuelas separadas dentro de la inteligencia artificial, por lo

que existe una preferencia en los investigadores en perfeccionar alguno de los dos modelos antes que tratar de unirlos.

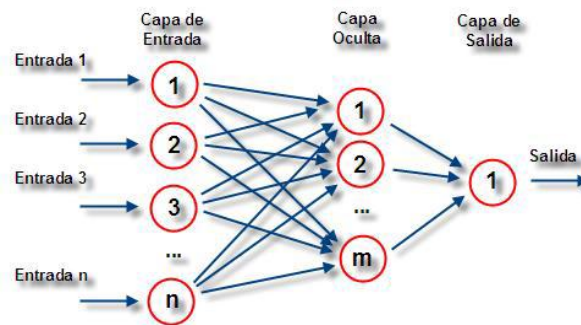


Figura 2.4. Esquema de una red neuronal  
Fuente: "<http://jast1992.blogspot.com/>

#### 2.3.8.4. EL ALGORITMO GENÉTICO SIMPLE<sup>13</sup>

El Algoritmo Genético Simple, también denominado Canónico se representa en la figura 2.5, como se verá a continuación, se necesita una codificación o representación del problema, que resulte adecuada para el mismo. Además se requiere una función de ajuste o adaptación al problema, la cual asigna un número real a cada posible solución codificada. Durante la ejecución del algoritmo los padres deben ser seleccionados para la reproducción, a continuación dichos padres seleccionados se cruzarán generando dos hijos, sobre cada uno de los cuales actuará un operador de mutación. El resultado de la combinación de las anteriores funciones será un conjunto de individuos (Posibles soluciones al problema), los cuales en la evolución del Algoritmo Genético formarán parte de la siguiente población.

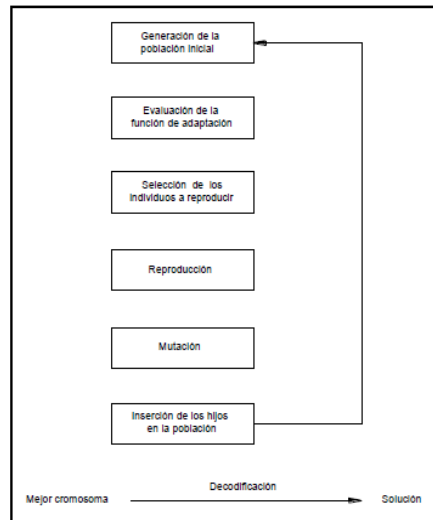
Ahora bien; un esquema del funcionamiento general de un algoritmo genético podría ser el siguiente:

Algoritmo Genético

1. Genera una población inicial.
2. Iterar hasta un criterio de parada.
3. Evaluar cada individuo de la población.

<sup>13</sup> GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning

4. Seleccionar los progenitores.
5. Aplicar el operador de cruce y mutación a estos progenitores.
6. Incluir la nueva descendencia para formar una nueva generación.



*Figura 2.5. Esquema de un algoritmo genético básico*

*Fuente: "Genetic Algorithms – in Search, Optimization and Machine Learning." De GOLDBERG, David E. (1989) pág. 118*

#### **2.3.8.4.1. CODIFICACIÓN**

Se supone que los individuos (posibles soluciones del problema), pueden representarse como un conjunto de parámetros (que denominaremos genes), los cuales agrupados forman una ristra de valores (a menudo referida como cromosoma). Si bien el alfabeto utilizado para representar los individuos no debe necesariamente estar constituido por el (0,1), buena parte de la teoría en la que se fundamentan los Algoritmos Genéticos utilizan dicho alfabeto. En términos biológicos, el conjunto de parámetros representando un cromosoma particular denomina genotipo. El genotipo contiene la información requerida para construir un organismo, el cual se refiere como fenotipo. Los mismos términos se utilizan en el campo de los Algoritmos Genéticos. La adaptación al problema de un individuo depende de la evaluación del fenotipo. Esta última puede inferirse a partir del genotipo, es decir puede ser computada a partir del cromosoma, usando la función de evaluación. La función de adaptación debe ser diseñada para cada problema de manera específica. Dado un cromosoma particular, la función de adaptación

le asigna un número real que se supone refleja el nivel de adaptación al problema del individuo representado por el cromosoma.

Durante la fase reproductiva se seleccionan los individuos de la población para cruzarse y producir descendientes, que constituirán, una vez mutados, la siguiente generación de individuos. La selección de padres se efectúa al azar usando un procedimiento que favorezca a los individuos mejor adaptados, ya que a cada individuo se le asigna una probabilidad de ser seleccionado que es proporcional a su función de adaptación. Este procedimiento se dice que está basado en la ruleta sesgada. Según dicho esquema, los individuos bien adaptados se escogerán probablemente varias veces por generación, mientras que, los pobremente adaptados al problema, no se escogerán más que de vez en cuando.

Una vez seleccionados dos padres, sus cromosomas se combinan, utilizando habitualmente los operadores de cruce y mutación. Las formas básicas de dichos operadores se describen a continuación.

El operador de cruce.- coge dos padres seleccionados y corta sus ristas de cromosomas en una posición escogida al azar, para producir dos subristras iniciales y dos subristras finales. Después se intercambian las subristras finales, produciéndose dos nuevos cromosomas completos (véase la Figura 2.6). Ambos descendientes heredan genes de cada uno de los padres. Este operador se conoce como operador de cruce basado en un punto.

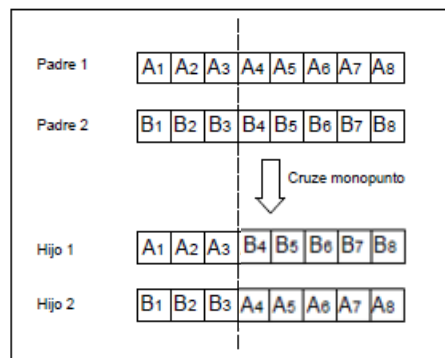


Figura 2.6. Esquema de un cruce de un solo punto en un AG

Fuente: “Genetic Algorithms – in Search, Optimization and Machine Learning.”  
DeGOLDBERG, David E. (1989) pág. 85

Habitualmente el operador de cruce no se aplica a todos los pares de individuos que han sido seleccionados para emparejarse, sino que se aplica de manera aleatoria, normalmente con una

probabilidad comprendida entre 0.5 y 1.0. En el caso en que el operador de cruce no se aplique, la descendencia se obtiene simplemente duplicando los padres.

El operador de mutación.- se aplica a cada hijo de manera individual, y consiste en la alteración aleatoria (normalmente con probabilidad pequeña) de cada gen componente del cromosoma. La Figura 2.7 muestra la mutación del octavo gen del cromosoma.

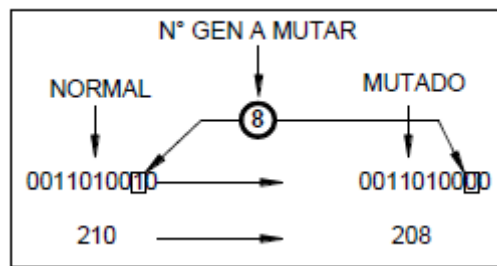


Figura 2.7. Esquema de mutación de un individuo de AG

Fuente. "Diseño de columnas a flexocompresión biaxial utilizando algoritmos genéticos".  
Andrés Donoso-Stalin Parreño

Sí bien puede en principio pensarse que el operador de cruce es más importante que el operador de mutación, ya que proporciona una exploración rápida del espacio de búsqueda, éste último asegura que ningún punto del espacio de búsqueda tenga probabilidad cero de ser examinado, y es de capital importancia para asegurar la convergencia de los Algoritmos Genéticos.

Si el Algoritmo Genético ha sido correctamente implementado, la población evolucionará a lo largo de las generaciones sucesivas de tal manera que la adaptación media extendida a todos los individuos de la población, así como la adaptación del mejor individuo se irán incrementando hacia el óptimo global. El concepto de convergencia está relacionado con la progresión hacia la uniformidad, un gen ha convergido cuando al menos el 95 % de los individuos de la población comparten el mismo valor para dicho gen. Se dice que la población converge cuando todos los genes han convergido. Se puede generalizar dicha definición al caso en que al menos un poco de los individuos de la población hayan convergido. A medida que el número de generaciones aumenta, es más probable que la adaptación media se aproxime a la del mejor individuo.

#### 2.3.8.4.2. POBLACIÓN

##### 2.3.8.4.2.1. TAMAÑO DE LA POBLACIÓN

Una cuestión que uno puede plantearse es lo relacionado con el tamaño idóneo de la población. Parece intuitivo que las poblaciones pequeñas corren el riesgo de no cubrir adecuadamente el espacio de búsqueda, mientras que el trabajar con poblaciones de gran tamaño puede acarrear problemas relacionados con el excesivo costo computacional.

Goldberg efectuó un estudio teórico, obteniendo como conclusión que el tamaño óptimo de la población para ristas de longitud  $I$ , con codificación binaria, crece exponencialmente con el tamaño de la ristra<sup>14</sup>.

Este resultado traería como consecuencia que la aplicabilidad de los Algoritmos Genéticos en problemas reales sería muy limitada, ya que resultarían no competitivos con otros métodos de optimización combinatoria. Alander, basándose en evidencia empírica sugiere que un tamaño de población comprendida entre 1 y 21 es suficiente para atacar con éxito los problemas por él, considerados.

#### **2.3.8.4.2.2. POBLACIÓN INICIAL**

Habitualmente la población inicial se escoge generando ristas al azar, pudiendo contener cada gen uno de los posibles valores del alfabeto con probabilidad uniforme. Se podría preguntar qué es lo que sucedería si los individuos de la población inicial se obtuviesen como resultado de alguna técnica heurística o de optimización local. En los pocos trabajos que existen sobre este aspecto, se constata que esta inicialización no aleatoria de la población inicial, puede acelerar la convergencia del algoritmo genético. Sin embargo en algunos casos la desventaja resulta ser la prematura convergencia del algoritmo, queriendo indicar con esto la convergencia hacia óptimos locales.

#### **2.3.8.4.3. FUNCIÓN OBJETIVO**

De acuerdo con el problema que se desea desarrollar, se debe definir la forma matemática de la función que debemos mejorar (ya sea maximizar o minimizar), y es el mecanismo de evolución de las generaciones<sup>15</sup>.

Dos aspectos que resultan cruciales en el comportamiento de los algoritmos genéticos son la determinación de una adecuada función de adaptación o función objetivo, así como la codificación utilizada.

Idealmente interesaría construir funciones objetivo con ciertas regularidades, es decir funciones objetivo que verifiquen que para dos individuos que se encuentren cercanos en el

---

<sup>14</sup> GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning

<sup>15</sup> Estudio de los Algoritmos genéticos para el desarrollo de sistemas de control, Patricia Gómez Otero.



espacio de búsqueda, sus respectivos valores en las funciones objetivo sean similares. Por otra parte una dificultad en el comportamiento del algoritmo genético puede ser la existencia de gran cantidad de óptimos locales, así como el hecho de que el óptimo global se encuentre muy aislado.

La regla, general para construir una buena función objetivo es que ésta debe reflejar el valor del individuo de una manera real, pero en muchos problemas de optimización combinatoria, donde existe gran cantidad de restricciones, buena parte de los puntos del espacio de búsqueda representan individuos no válidos.

Para este planteamiento en el que los individuos están sometidos a restricciones, se han propuesto varias soluciones. La primera sería la que podríamos denominar absolutista, en la que aquellos individuos que no verifican las restricciones, no son considerados como tales, y se siguen efectuando cruces y mutaciones hasta obtener individuos válidos, o bien, a dichos individuos se les asigna una función objetivo igual a cero.

Otra posibilidad consiste en reconstruir aquellos individuos que no verifican las restricciones. Dicha reconstrucción suele llevarse a cabo por medio de un nuevo operador que se acostumbra a denominar reparador.

Otro enfoque está basado en la penalización de la función objetivo. La idea general consiste en dividir la función objetivo del individuo por una cantidad (la penalización) que guarda relación con las restricciones que dicho individuo viola.

Dicha cantidad puede simplemente tener en cuenta el número de restricciones violadas o bien el denominado costo esperado de reconstrucción, es decir el coste asociado a la conversión de dicho individuo en otro que no viole ninguna restricción.

Otra técnica que se ha venido utilizando en el caso en que la computación de la función objetivo sea muy compleja, se la denomina evaluación aproximada de la función objetivo. En algunos casos la obtención de  $n$  funciones objetivo aproximadas puede resultar mejor que la evaluación exacta de una única función objetivo (supuesto el caso de que la evaluación aproximada resulta como mínimo  $n$  veces más rápida que la evaluación exacta).

Un problema habitual en las ejecuciones de los Algoritmos Genéticos surge debido a la velocidad con la que el algoritmo converge. En algunos casos la convergencia es muy rápida, lo que suele denominarse convergencia prematura, en la cual el algoritmo converge hacia óptimos locales, mientras que en otros casos el problema es justo lo contrario, es decir se produce una convergencia lenta del algoritmo. Una posible solución a estos problemas pasa por efectuar transformaciones en la función objetivo. El problema de la convergencia

prematura, surge a menudo cuando la selección de individuos se realiza de manera proporcional a su función objetivo. En tal caso, pueden existir individuos con una adaptación al problema muy superior al resto, que a medida que avanza el algoritmo dominan a la población. Por medio de una transformación de la función objetivo, en este caso una comprensión del rango de variación de la función objetivo, se pretende que dichos súper-individuos no lleguen a dominar a la población.

El problema de la lenta convergencia del algoritmo, se resolvería de manera análoga, pero en este caso efectuando una expansión del rango de la función objetivo.

La idea de especies de organismos, ha sido imitada en el diseño de los algoritmos genéticos en un método propuesto por Goldberg y Richardson, utilizando una modificación de la función objetivo de cada individuo, de tal manera que individuos que estén muy cercanos entre sí devalúen su función objetivo, con objeto de que la población gane en diversidad<sup>16</sup>.

#### **2.3.8.4.4. SELECCIÓN**

La función de selección de padres más utilizada, es la denominada función de selección proporcional a la función objetivo, en la cual cada individuo tiene una, probabilidad de ser seleccionado como padre que es proporcional al valor de su función objetivo.

Una de las maneras de superar el problema relacionado con la rápida convergencia proveniente de los súper-individuos, que surge al aplicar la anterior función de selección, es el efectuar la selección proporcional al rango del individuo, con lo cual se produce una repartición más uniforme de la probabilidad de selección.

#### **2.3.8.4.5. CRUCE**

El Algoritmo Genético Canónico descrito anteriormente utiliza el cruce basado en un punto, en el cual los dos individuos seleccionados para jugar el papel de padres, son recombinados por medio de la selección de un punto de corte, para posteriormente intercambiar las secciones que se encuentran a la derecha de dicho punto.

Se han investigado otros operadores de cruce, habitualmente teniendo en cuenta más de un punto de cruce. De Jong investigó el comportamiento del operador de cruce basado en múltiples puntos, concluyendo que el cruce basado en dos puntos, representaba una mejora mientras que

---

<sup>16</sup> Diseño a flexión de vigas rectangulares de hormigón armado basado en la normativa ecuatoriana y mediante la aplicación de algoritmos genéticos, 2011, Felix Armando Ashqui Laguna.

añadir más puntos de cruce no beneficiaba el comportamiento del algoritmo. La ventaja de tener más de un punto de cruce radica en que el espacio de búsqueda puede ser explorado más fácilmente, siendo la principal desventaja el hecho de aumentar la probabilidad de ruptura de buenos esquemas. En el operador de cruce basado en dos puntos, los cromosomas (individuos) pueden contemplarse como un circuito en el cual se efectúa la selección aleatoria de dos puntos. Desde este punto de vista, el cruce basado en un punto, puede verse como un caso particular del cruce basado en dos puntos, en el cual uno de los puntos de corte se encuentra fijo al comienzo de la ristra que representa al individuo.

En el denominado operador de cruce uniforme (Syswerda) cada gen, en la descendencia se crea copiando el correspondiente gen de uno de los dos padres, escogido de acuerdo a una máscara de cruce generada aleatoriamente. Cuando existe un 1 en la máscara de cruce, el gen es copiado del primer padre, mientras que cuando exista un 0 en la máscara de cruce, el gen se copia del segundo padre.

En la literatura, el término operador de cruce uniforme se relaciona con la obtención de la máscara de cruce uniforme, en el sentido de que cualquiera de los elementos del alfabeto tenga asociada la misma probabilidad. Hablando en términos de la teoría de la probabilidad, la máscara de cruce está compuesta por una muestra aleatoria de tamaño  $A$  extraída de una distribución de probabilidad de Bernoulli de parámetro  $1/2$ .

Si tuviésemos en cuenta el valor de la función de adaptación de cada padre en el momento de generar la máscara de cruce, de tal manera que cuanto mayor sea la función de adaptación de un individuo, más probable sea heredar sus características, podríamos definir, un operador de cruce basado en la función objetivo, en el cual la máscara de cruce se interpreta como una muestra aleatoria de tamaño 1 proveniente de una distribución de Bernoulli de parámetro.

#### **2.3.8.4.6. MUTACIÓN<sup>17</sup>**

La mutación se considera un operador básico, que proporciona un pequeño elemento de aleatoriedad en la vecindad (entorno) de los individuos de la población. Si bien se admite que el operador de cruce es el responsable de efectuar la búsqueda a lo largo del espacio de posibles soluciones, también parece desprenderse de los experimentos efectuados por varios investigadores que el operador de mutación va ganando en importancia a medida que la población de individuos va convergiendo (Davis).

---

<sup>17</sup> GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning

Schaffer y col. encuentran que el efecto del cruce en la búsqueda es inferior al que previamente se esperaba. Utilizan la denominada evolución primitiva, en la cual, el proceso evolutivo consta tan sólo de selección y mutación. Encuentran que dicha evolución primitiva supera con creces a una evolución basada exclusivamente en la selección y el cruce. Otra conclusión de su trabajo es que la determinación del valor óptimo de la probabilidad de mutación es mucho más crucial que el relativo a la probabilidad de cruce.

La búsqueda del valor óptimo para la probabilidad de mutación, es una cuestión que ha sido motivo de varios trabajos. Se recomienda la utilización de una probabilidad de mutación del bit de  $(1/L)$ , siendo  $L$  la longitud de la cadena.

Si bien en la mayoría de las implementaciones de Algoritmos Genéticos se asume que tanto la probabilidad de cruce como la de mutación permanecen constantes, algunos autores han obtenido mejores resultados experimentales modificando la probabilidad de mutación a medida que aumenta el número de iteraciones<sup>18</sup>.

La mutación se encarga de modificar en forma aleatoria uno o más genes del cromosoma de un descendiente. De esta manera, la tasa de mutación indica el porcentaje de bits que se deben mutar (cambiar de 0 a 1 o viceversa en el caso binario). El valor típico de la tasa de mutación es de 0.05<sup>19</sup>.

---

<sup>18</sup> GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning

<sup>19</sup> Estudio de los Algoritmos genéticos para el desarrollo de sistemas de control, Patricia Gómez Otero

## CAPITULO III

### 3. MARCO METODOLÓGICO

#### 3.1. SISTEMATIZACIÓN DE LA METODOLOGÍA A EMPLEARSE

Para el siguiente estudio se realizó un proceso sistemático, que se detalla a continuación:

1. Estudio y análisis del método de diseño de vigas-trabe tipo I armadas, condiciones de diseño y ecuaciones.
2. Investigación y análisis de la estructura de algoritmo genético.
3. Determinación de variables del algoritmo genético para el diseño vigas-trabe tipo I armadas mediante placas de acero.
4. Determinación de la función objetivo del algoritmo genético dirigida al diseño de vigas-trabe tipo I armadas mediante placas de acero.
5. Establecer limitación del algoritmo genético para el diseño de vigas-trabe tipo I armadas.
6. Formulación del modelo matemático que rige el algoritmo genético para el diseño de vigas-trabe tipo I armadas.
7. Implementación del algoritmo genético en el diseño de vigas-trabe tipo I armadas.
8. Aplicación del algoritmo genético a vigas-trabe tipo I armadas, reales.
9. Comparación de resultados entre el método tradicional de cálculo y el algoritmo genético.

##### **3.1.1. ESTUDIO Y ANÁLISIS DEL MÉTODO DE DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO, CONDICIONES DE DISEÑO Y ECUACIONES.**

Esta es la primera etapa del estudio en la cual se analizara a fondo los fundamentos e hipótesis en las cuales se basara el diseño de vigas-trabe tipo I armadas mediante placas de acero.

En esta etapa se hizo imprescindible realizar de una forma detallada la deducción de ecuaciones que rigen el diseño de vigas-trabe tipo I armadas mediante placas de acero, las cuales fueron de gran importancia en la determinación del modelo matemático que va a regir el algoritmo genético y componentes tales como función objetivo y limitaciones.

### **3.1.2. INVESTIGACIÓN Y ANÁLISIS DE LA ESTRUCTURA DE ALGORITMO GENÉTICO.**

En esta fase se estudió la estructuración de un algoritmo genético y se considera como fundamental para el desarrollo de este proyecto. Posteriormente se programó un algoritmo genético mismo que después de haberlo sometido a suficientes pruebas se lo puede reconocer como método de optimización, lo cual es nuestro objetivo.

### **3.1.3. DETERMINACIÓN DE VARIABLES DEL ALGORITMO GENÉTICO PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO.**

Posteriormente luego de haber analizado en forma teórica y puesto en práctica el algoritmo genético básico se procede a la determinación de variables tanto del diseño de vigas-trabe tipo I como las necesarias para el algoritmo genético que en nuestro caso fueron: Base (b), espesor del patín (bf), espesor del alma (tw), altura (h) del alma.

### **3.1.4. DETERMINACIÓN DE LA FUNCIÓN OBJETIVO DEL ALGORITMO GENÉTICO DIRIGIDA AL DISEÑO VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO.**

En esta etapa se determinara un sistema de evaluación que medirá el comportamiento de las variables que encontrara el algoritmo genético, y es responsable de la minimización de las funciones que se vaya a optimizar, esta es la encargada de regular los factores que forman parte de las ecuaciones de búsqueda.

### **3.1.5. ESTABLECER LIMITACIÓN DEL ALGORITMO GENÉTICO PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO.**

Se procedió a determinar las condiciones y restricciones que debían cumplir las variables que conforman la población obtenida en cada individuo y cada generación las misma que después de un proceso de selección de los individuos más aptos o los que mejor minimizan la función objetivo, pasan a la siguiente generación, concediéndoles el derecho de reproducirse, con lo cual garantizamos que el material genético o información se reproduzca en las siguientes generaciones, obviamente después de la combinación del material genético de otros individuos que igualmente fueron seleccionados por el algoritmo genético. Así sucesivamente los individuos por n generaciones van evolucionando hasta que los individuos llegan a su límite de convergencia, es decir hasta que no exista otra solución que produzca mejores resultados al aplicarlas en conjunto con todas las demás variables, mediante las restricciones se ha logrado

controlar que las soluciones cumplan con la normativa para diseño de vigas-trabe que propone el AISC-LRFD.

### **3.1.6. FORMULACIÓN DEL MODELO MATEMÁTICO QUE RIGE EL ALGORITMO GENÉTICO PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO.**

Luego de haber determinado variables, restricciones y función objetivo para poder aplicar el algoritmo genético, se posee todas las herramientas necesarias para poder formular nuestro modelo matemático, que servirá como la trayectoria a seguir para utilizar el algoritmo genético y su programación, este modelo matemático es el encargado de determinar la idoneidad o no de los individuos de cada población de acuerdo a su función de selección, la ecuación que servirá para evaluación de vigas-trabe será:

$$Z = \left[ \left[ (X2 - (2 * X4)) * X3 \right] + \left[ 2 * (X1 * X4) \right] \right] * \frac{1}{100^3} * 7850 * P1 \quad (2.2.13)$$

Donde:

Z= costo por metro lineal del Trabe diseñado por el algoritmo genético. (2.2.13)

X1= Base del ala del perfil de acero o BF.

X2= Altura del perfil de acero o H.

X3= Espesor del alma del perfil de acero o TW.

X4= Espesor del ala del perfil de acero o TF.

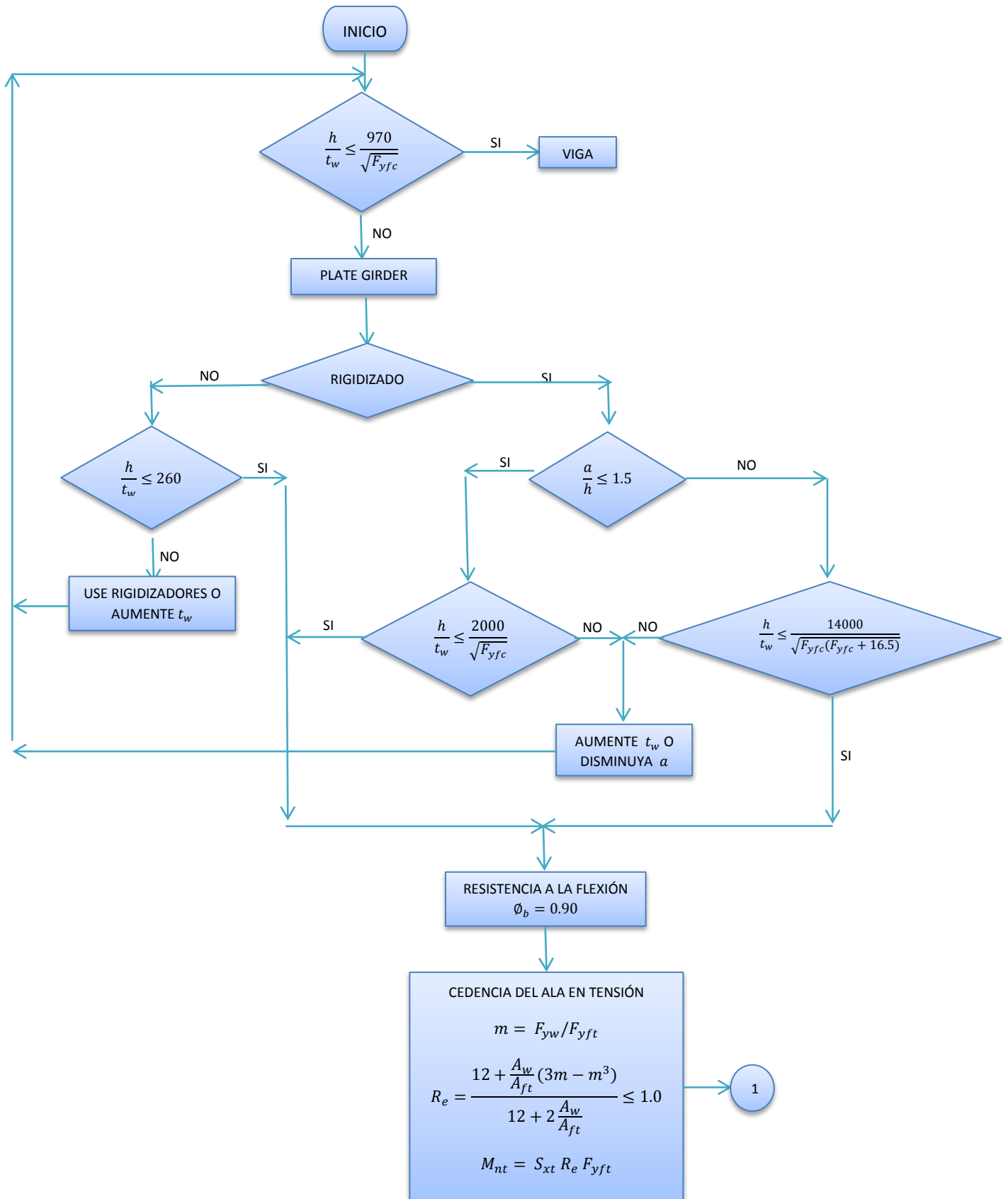
P1= Costo de un Kg de acero estructura en dólares.

El factor  $\frac{1}{100^3}$  transforma el área del perfil de  $mm^2$  a  $m^2$ , y el factor 7850 Kg/m<sup>3</sup> es el peso específico del acero.

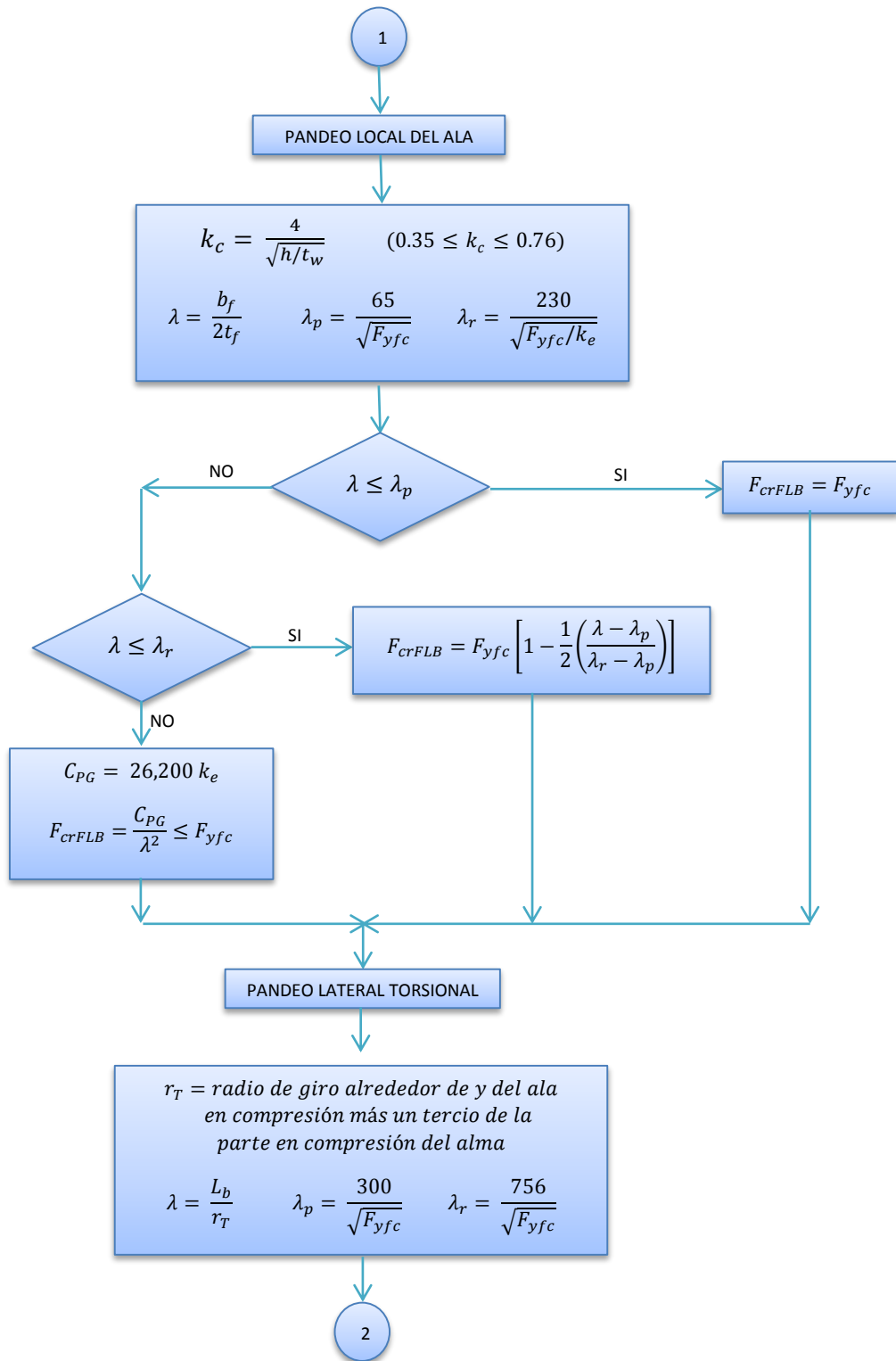
Uno de los principales objetivos al implementar un algoritmo genético al diseño de vigas-trabe tipo I armadas mediante placas de acero es el de encontrar la viga más económica y esto se refleja en la ecuación de la función objetivo que es minimizar el costo total que implica la combinación eficiente de materiales.

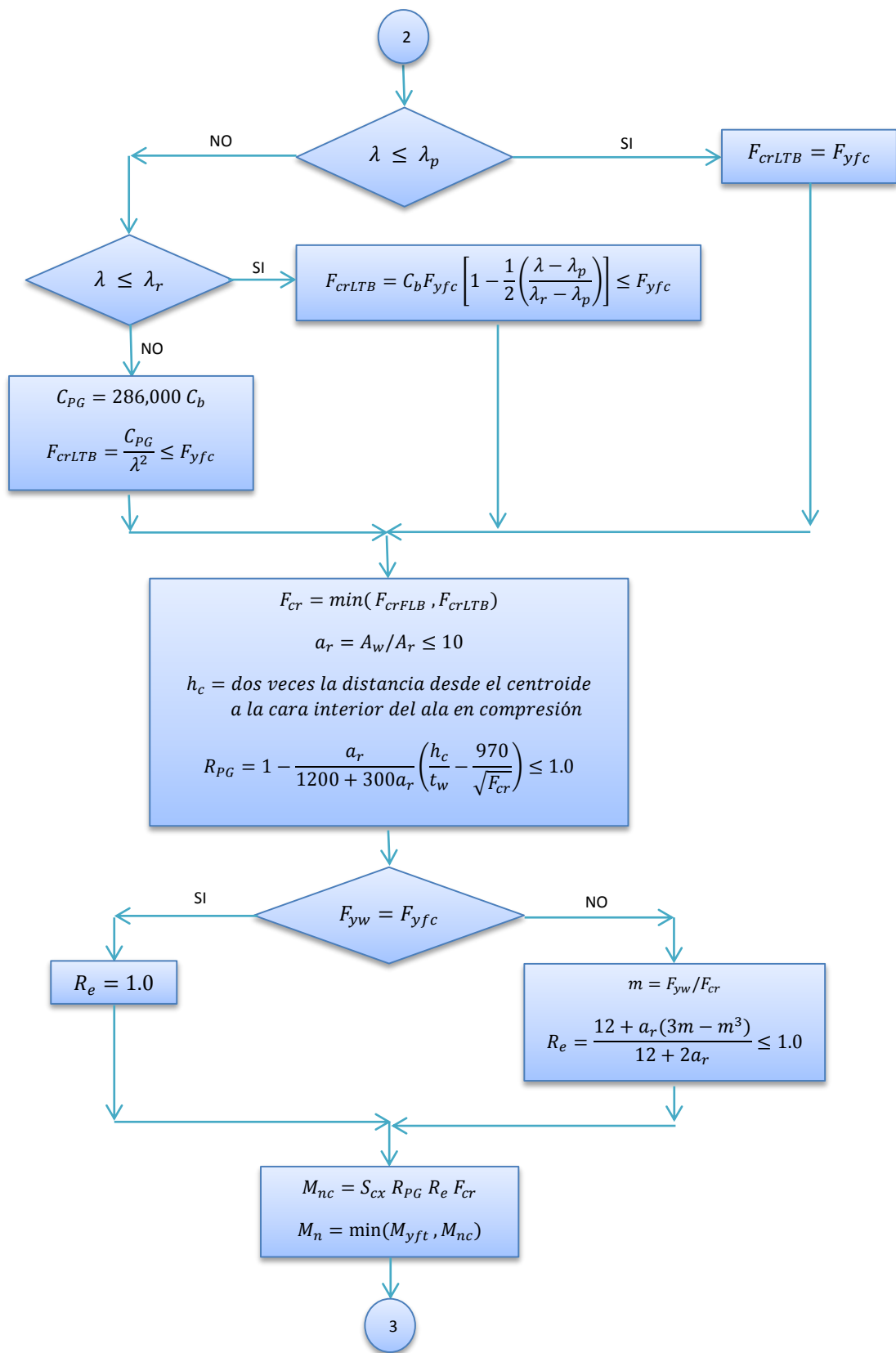
Estas funciones objetivo deben estar sujetas a restricciones es decir ecuaciones dadas por el LRFD para el diseño de vigas-trabe, el algoritmo ira evaluando cada generación para determinar las óptimas que brinden una minimización de costo, y que además cumplan con la normativa que a continuación mostramos en el siguiente diagrama de flujo:

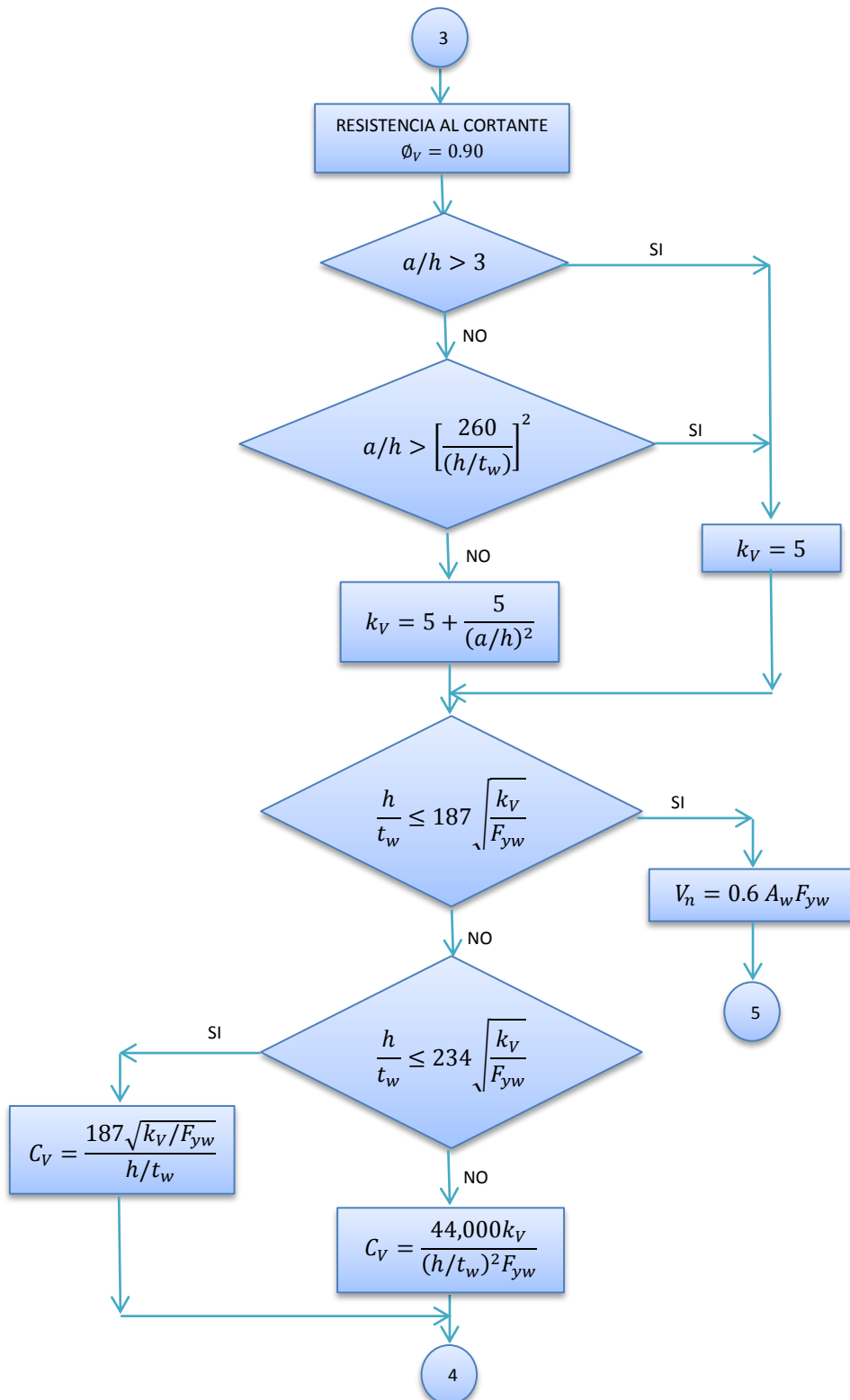
**Grafico 3.1. Diseño LRFD para Vigas-Trabe tipo I armadas mediante placas de acero.**  
**Elaborado Por: Hernán David Salazar Valverde**

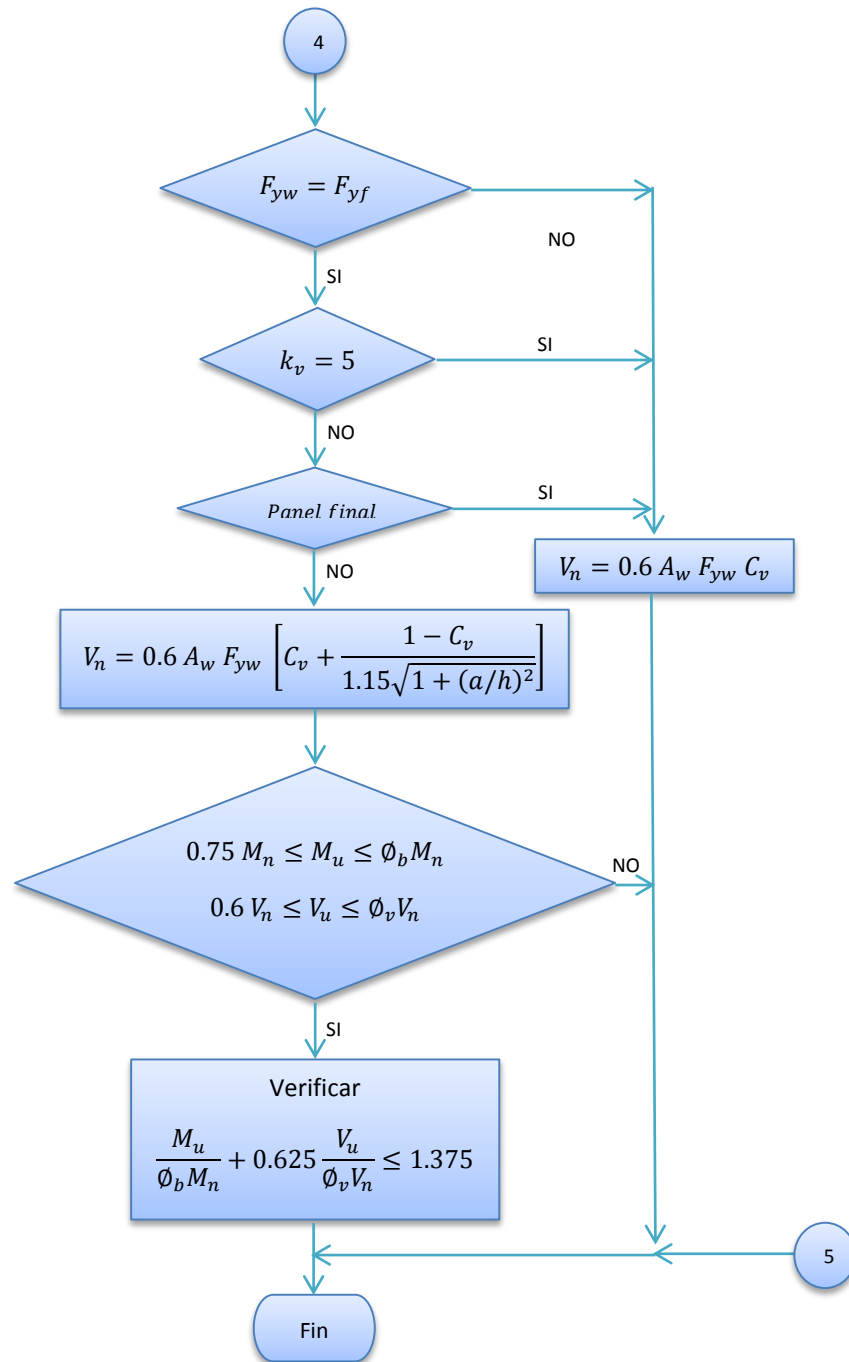












### **3.1.7. IMPLEMENTACIÓN DEL ALGORITMO GENÉTICO PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO.**

Al llegar a esta etapa ya hemos determinado nuestras funciones objetivas y nuestras ecuaciones de diseño, por lo cual empezamos a implementar el Algoritmo Genético que es totalmente programable utilizando el lenguaje de programación web a través del software Net Beans versión 8.0.

Empezamos con un modelo básico del algoritmo genético, para chequear vigas previamente diseñadas a mano, se realizaron varios cambios al Algoritmo básico ya que en primera instancia se usó una sola variable hasta llegar a programarla con las necesarias para nuestro caso, y por otro lado los cambios se hicieron ya que inicialmente obteníamos resultados completamente fuera de la realidad sobre todo en lo que se refiere a dimensiones.

### **3.1.8. APLICACIÓN DEL ALGORITMO GENÉTICO A VIGAS-TRABE TIPO I.**

Una vez programado el algoritmo genético para el diseño vigas-trabe tipo I armadas, en forma obligatoria se pasa por una serie de pruebas que determinen la idoneidad del mismo, con lo cual se inicia el proceso de pruebas en ejemplos planteados de elementos reales, en este proceso se encontraron una serie de falencias que se fueron salvando al ir evolucionando el diseño de algoritmo entre los cuales resaltamos:

- Lógica del proceso del planteamiento del algoritmo.
- Pérdida del óptimo por degeneración.

### **3.1.9. COMPARACIÓN DE RESULTADOS ENTRE EL DISEÑO TRADICIONAL Y UTILIZANDO EL ALGORITMO GENÉTICO.**

En esta etapa analizamos los resultados de los diseños obtenidos mediante el método tradicional y mediante la aplicación del programa computarizado DEAC-TB desarrollado como propuesta en la presente investigación, con la finalidad de obtener resultados que nos ayuden a determinar similitudes y diferencias para así emitir juicios de valor con el objetivo de demostrar la idoneidad del Software realizado.

## **3.2. TIPO DE ESTUDIO.**

Para el siguiente estudio fue necesaria la aplicación de varios tipos de investigación los cuales detallamos a continuación:

En primera instancia es una investigación de tipo descriptiva, ya que la aplicación del algoritmo genético es un campo poco difundido por lo cual se necesita una descripción del funcionamiento y las bases científicas para que el lector asimile de una forma total el estudio. Al analizar el proyecto también se hace necesario una investigación experimental, pues cumple con las tres condiciones esenciales de este tipo, las mismas que son:

- Se puede manipular las variables independientes.
- Se puede medir el efecto que produce las variables independientes sobre las dependientes.
- Se tiene un control interno de la situación experimental.

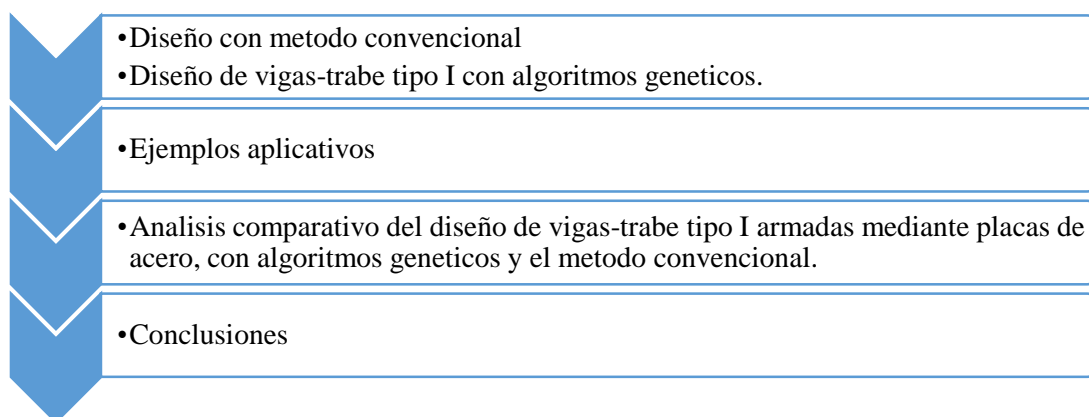
Cabe recalcar que sí, en forma física no se desarrolla la experimentación en un laboratorio convencional, los datos son procesados en un laboratorio virtual, pues al desarrollar un algoritmo genético se manipulan directamente variables para al final llegar a obtener conclusiones coherentes.

A así mismo, según la naturaleza del estudio, utilizamos un investigación cuantitativa, pues se maneja valores numéricos directos, lo cual se relaciona con la investigación de tipo experimental.

Al encontrarse este proyecto dentro de un campo innovador, el nivel de la investigación es aplicativo pues la innovación es una de las principales características de la misma.

### 3.3. DISEÑO DE LA INVESTIGACIÓN

Para alcanzar los objetivos planteados se desplegó el siguiente plan investigativo en el cual se detalla la secuencia a seguir:



*Figura 3.1 Esquema del diseño de investigación*

### 3.4. OPERACIONALIZACION DE VARIABLES

VARIABLE	TIPO DE VARIABLE	DEFINICION	OPERACIONALIZACION	DIMENSION	INDICADOR	INDICE
Algoritmos Genéticos	Independiente	Teoría que expone la optimización aplicada al diseño de vigas-trabe.	Forma en que opera el diseño con Algoritmos genéticos y su efectividad en la obtención de elementos de menor costo y que cumplan con la especificación AISC-LRFD	Desempeño de los Algoritmos genéticos en el diseño de vigas	Porcentaje de resultados coherentes	Índice de efectividad del algoritmo genético
Viga-trabe tipo I armada mediante placas de acero.	Dependiente	Las vigas trabe son elementos formados de placas de acero estructural, conectados entre sí para resistir conjuntamente las cargas por las solicitaciones.	Base altura espesor de la sección de acero estructural y costo por metro lineal del trabe.	Cantidad de vigas-trabe obtenidas de mínimo costo y que cumplan con la especificación AISC-LRFD	Porcentaje de vigas-trabe en las cuales se puede aplicar algoritmos genéticos	Índice de vigas-trabe que cumplan con la especificación AISC-LRFD y que tengan el menor costo diseñados con algoritmos genéticos
Diseño vigas trabe (Plate Girders)	Dependiente	Proceso para dimensionar un elemento sometido a carga.	Forma en que se dimensionan los elementos de una viga-trabe	Obtención de dimensiones de los elementos de una viga-trabe diseñada a	Porcentaje de viga-trabe diseñadas con costos mínimos y que cumplan con la especificación AISC-LRFD	Índice de viga-trabe que cumpla un diseño aceptable.

Tabla 3.1 Cuadro de Operacionalización de Variables

### **3.5. HIPOTESIS.**

“Al aplicar Algoritmos Genéticos al diseño de vigas-trabe tipo I armadas mediante placas de acero, se obtendrán resultados eficaces, eficientes en el aspecto estructural y económicos, cumpliendo la normativa establecida por la AISC-LRFD”.

#### **3.5.1. IDENTIFICACIÓN DE VARIABLES**

##### **3.5.1.1. VARIABLE INDEPENDIENTE**

- Algoritmos Genéticos

##### **3.5.1.2. VARIABLE DEPENDIENTE**

- Viga-trabe tipo I armada mediante placas de acero.
- Diseño vigas-trabe (Plate Girders)

### **3.6. PROCEDIMIENTOS.**

La recopilación de información de la presente investigación está centrada básicamente en la puesta en marcha de la aplicación del algoritmo genético aplicado al diseño a Viga-trabe tipo I armada mediante placas de acero, para sustentar la aplicabilidad del diseño de vigas-trabe se ejecutó la resolución de ejercicios de este tipo de vigas obtenidos de libros de diseño de estructuras y tesis de grado de diferentes autores aclarando que los mismos se encuentran dentro de los límites fijados al inicio de esta investigación, los cuales son para Viga-trabe tipo I armada mediante placas de acero con límites de diseño normados por el AISC-LRFD procediendo a comparando resultados, para en lo posterior obtener datos con la aplicación del algoritmo genético en un caso práctico y consiguiendo conclusiones generales para cada caso. No hubo necesidad de personal de apoyo para la recopilación de datos, ya que se la realizó en forma directa entre el diseñador y los datos arrojados mediante el software, al ser una investigación de tipo experimental y de nivel aplicativo, no fue necesaria la aplicación de técnicas de recolección de datos, tales como encuestas , entrevistas o alguna otra herramienta por tratarse de una aplicación comprobable por medios matemáticos, por lo cual no se necesita recolectar información a más de la bibliográfica y sugerencias que han procedido de apreciaciones y experiencias de personas que están relacionadas con este tema.



Se debe indicar en cuanto a la ubicación cronológica de la recopilación y procesamiento de información, que esta actividad se la aplicó una vez que se dio de alta el proceso y funcionamiento del algoritmo genético analizado a través de la fase de prueba, con la cual se certifica la idoneidad de la aplicación para el diseño de Vigas-trabe tipo I armada mediante placas de acero.

### 3.6.1. PLANTEAMIENTO DE LOS EJERCICIOS DE DISEÑO VIGAS-TRABE TIPO I ARMADA MEDIANTE PLACAS DE ACERO.

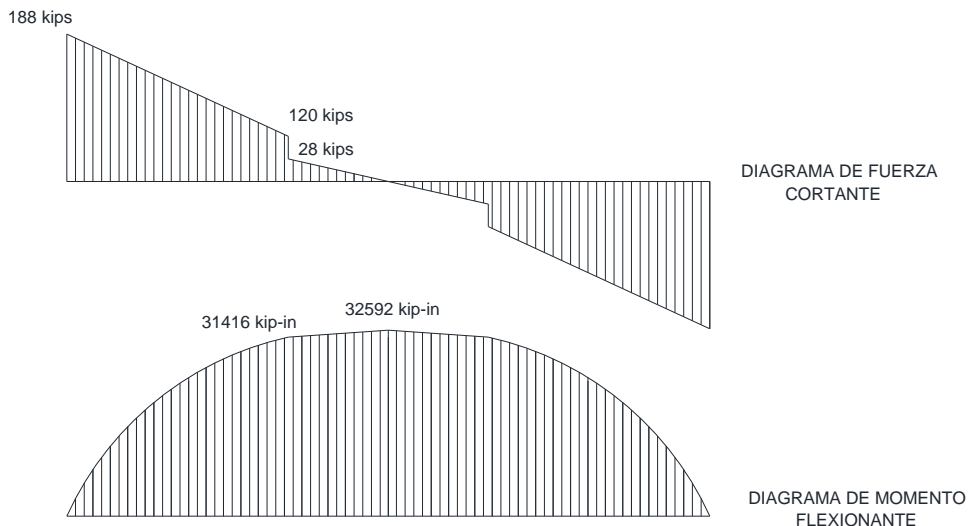
Los ejercicios que se detallan a continuación servirán para comprobar la idoneidad del diseño mediante la aplicación de Algoritmos Genéticos, los cuales son extraídos de libros de estructuras en acero los cuales se detallarán en cada ejercicio.

#### EJERCICIO N° 1<sup>20</sup>

Diseñe una trabe armada para las cargas mostradas.

El material es acero A36:

$$F_y = 36 \text{ ksi} \quad E = 29000 \text{ ksi}$$



#### Características de la estructura:

<sup>20</sup> Theodore V. Galambos. **Diseño de estructuras de acero con LRFD. 1999**

$L = 48 \text{ ft} ; 14.63 \text{ m}$

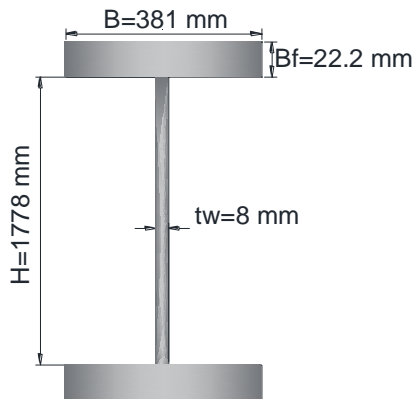
$L_b = 17 \text{ ft} ; 5.18 \text{ m}$

**Condiciones de cargas:**

$M_{\max} = 32592 \text{ kip-in} ; 376.29 \text{ ton-m}$

$V_{\max} = 188 \text{ kips} ; 85.45 \text{ ton}$

**Propiedades Geométricas de la sección en mm:**



**Resultados principales:**

$M_{\max} = 35530 \text{ kip-in} ; 410.21 \text{ ton-m}$

$V_{\max} = 279.1 \text{ kips} ; 126.86 \text{ ton}$

**EJERCICIO N° 2<sup>21</sup>**

Diseñe una trabe armada por flexión y cortante usando unidades SI, una vez realizado el análisis estructural las cargas por solicitaciones son las siguientes.

**Condiciones de cargas:**

$M_{\max} = 8715.6 \text{ kN-m} ; 880.28 \text{ ton-m}$

$V_{\max} = 2421 \text{ kN} ; 244.52 \text{ ton}$

**Características de la estructura:**

$F_y = 345 \text{ MPa} ; 50 \text{ ksi}$

$E = 200000 \text{ Mpa} ; 29000 \text{ ksi}$

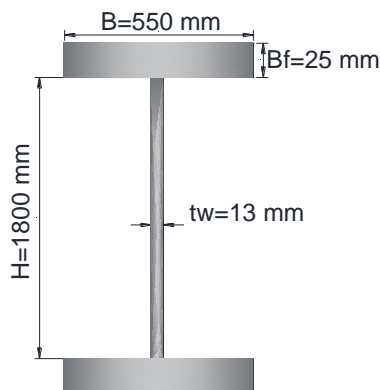
$L = 18 \text{ m}$

El Trabe esta soportado lateralmente en toda la cuerda superior  $L_b = 0$

---

<sup>21</sup> Theodore V. Galambos. **Diseño de estructuras de acero con LRFD. 1999**

## Propiedades Geométricas de la sección en mm:



### Resultados principales:

$M_{max} = 8956.46 \text{ kN-m}$  ;  $904.60 \text{ ton-m}$

$V_{max} = 2496.72 \text{ kN}$  ;  $252.10 \text{ ton}$

### 3.7. PROCESAMIENTO Y ANÁLISIS

A continuación se presentan los datos generados mediante el método de optimización del diseño de Vigas-trabe tipo I armada mediante placas de acero a través de Algoritmos Genéticos.

#### 3.7.1. DISEÑO DE VIGAS-TRABE TIPO I ARMADA MEDIANTE PLACAS DE ACERO CON LA UTILIZACION DE ALGORITMOS GENETICOS.

Esta sección es el punto céntrico de la presente investigación, en la cual se desarrollará paso a paso la aplicación de un algoritmo genético para el diseño de Vigas-trabe tipo I armada mediante placas de acero.

##### 3.7.1.1. EXPLICACIÓN GLOBAL DEL PROCESO QUE SIGUE EL ALGORITMO GENÉTICO APLICADO AL DISEÑO DE VIGAS-TRABE TIPO I ARMADA MEDIANTE PLACAS DE ACERO<sup>22</sup>

Del marco teórico traemos a memoria que un algoritmo genético es un imitador del proceso evolutivo de los seres vivos, mencionado en la teoría de la evolución de Charles Darwin y aplicando procesos genéticos, en la cual a partir de una población inicial, en nuestro caso de ecuaciones formadas por las dimensiones, materiales, factores de reducción , variables de costo y geometría se van reproduciendo, imponiendo un porcentaje de mutación y bajo un modelo

---

<sup>22</sup> Ashqui Armando. "DISEÑO A FLEXION DE VIGAS RECTANGULARES DE HORMIGÓN ARMADO BASADO EN LA NORMATIVA ECUATORINA Y MEDIANTE LA APLICACIÓN DE ALGORITMOS GENETICOS".

de cruce, con el propósito de que al llegar a la función de selección exista una población lo suficientemente variada para que la misma seleccione a los mejores individuos y pasen a la siguiente generación, simulando a la selección natural, sobreviviendo y reproduciéndose los individuos cuyas características los han convertido en los más aptos, los individuos son variables que actúan en ecuaciones de evolución, las nuevas variables, que volverán a realizar el mismo proceso en generaciones, hasta cuando la población converja, esto quiere decir que es homogénea, o en otras palabras ha llegado al punto en donde la combinación de variables da como resultado el valor mínimo al evaluar la función objetivo y este valor empieza a repetirse indefinidamente, tanto en la función de evaluación como en la población de las últimas generaciones de cada ambiente.

### **3.7.1.2. EXPLICACIÓN PASO A PASO DEL PROCESO QUE SIGUE EL ALGORITMO GENÉTICO APLICADO AL DISEÑO DE VIGAS-TRABE TIPO I ARMADA MEDIANTE PLACAS DE ACERO**

Lo mencionado en el artículo anterior se describirá paso a paso en este punto, debido a la gran cantidad de información que opera el algoritmo genético es conveniente que el manejo se lo realice aplicando paquetes informáticos, lo cual se lo ha realizado en esta investigación.

#### **3.7.1.2.1. PROCESO 1**

La lógica del proceso comienza con la lectura de las variables de entrada, las cuales se las ha agrupado según su tipo, y según la sección de la viga, dichos grupos y variables son:

#### **Grupo1: datos de materiales y características de la viga:**

Esfuerzo fluencia del acero=  $f_{yc}$

En caso de ser dos o más materiales:

Esfuerzo fluencia del acero en el patin=  $f_{yt}$

Esfuerzo fluencia del acero en el alma=  $f_{yw}$

Módulo de elasticidad del acero=  $E$

Factor de flexión=  $\phi$

Factor de Cortante=  $\phi_v$

#### **Grupo2: Cargas y Momentos**

Momento último =  $M_U$

Cortante último =VU

Estos datos se los obtiene realizando previamente un análisis estructural de la edificación a la que pertenece la viga.

### **Grupo3: Rangos y relaciones**

Longitud no soportada: Lb

Calculo del gradiente del momento=Cb

### **Grupo4: Costo de los materiales**

Costo del acero estructural por Kg =P1

#### **3.7.1.2.2. PROCESO 2**

Cabe aclarar que en el proceso anterior no se solicitó dimensiones de la sección, ya que éstas junto con otras adicionales son las que el algoritmo genético deberá encontrar.

Siguiendo la lógica del proceso se procedió a la inicialización de algoritmo genético, el mismo que está configurado de la siguiente manera:

- Generar la población inicial
- Codificación de individuos
- Generar parejas y posición de cruce
- Almacenamiento de información binaria.
- Realizar cruce de individuos
- Realizar mutación.
- Transformar de binario a decimal
- Evaluar la función de aptitud
- Realizar selección de individuos mejor adaptados
- Pasar la población seleccionada como nueva población inicial.

Este proceso trabaja para cada ambiente de la población, los cuales son:

- Población de búsqueda de X1, que es la base del patín de la trabe (bf)

- Población de búsqueda de X2, que es la altura del alma de la trabe (h)
- Población de búsqueda de X3, que es el espesor del alma de la trabe (tw)
- Población de búsqueda de X4, que es el espesor del patín del trabe (tf)

Estas son las dimensiones del perfil de acero estructural para producir una resistencia mayor o igual a las cargas últimas requeridas.

### 3.7.1.2.2.1. GENERAR LA POBLACIÓN INICIAL

En este paso se generan aleatoriamente 10 individuos para cada uno de los ambientes descritos anteriormente, de la siguiente manera didáctica para una viga tipo trabe.

#### Población inicial para x1

No	X1
	Población
	inicial decimal
1	211
2	484
3	457
4	270
5	345
6	294
7	363
8	355
9	294
10	423

#### Población inicial para x2

No	X2
	Población
	inicial decimal
1	1774
2	1156
3	1844
4	1378
5	1287
6	1963
7	1835
8	1140
9	1616
10	1521

### Población inicial para x3

No	X3
	Población
	inicial
	decimal
1	18
2	20
3	11
4	13
5	18
6	4
7	12
8	3
9	11
10	9

### Población inicial para x4

No	X4
	Población
	inicial
	decimal
1	10
2	81
3	99
4	95
5	17
6	35
7	52
8	35
9	69
10	47

La generación de la población inicial es aleatoria, esto se lo realiza con una función que genere al azar un número dentro de los rangos establecidos de: espesores, base-altura y secciones, y que para cada ambiente son las siguientes.

X1:  $X1_{\min}=200$ ,  $X1_{\max}=500$

X2:  $X2_{\min}=500$ ,  $X2_{\max}=2500$

X3:  $X3_{\min}=1$ ,  $X1_{\max}=20$

X4:  $X3_{\min}=10$ ,  $X1_{\max}=100$

### 3.7.1.2.2.2. CODIFICACIÓN DE INDIVIDUOS

Este es el paso en el cual se procede a la transformación del número decimal almacenado en cada uno de los ambientes a número binario ya que a través de este proceso se realiza el cruce de información conocida como información genética y mutación. Al individuo en número binario se le conoce como genotipo, mientras que al individuo en número decimal se le conoce con el nombre de fenotipo, a cada componente del genotipo los cuales son valores únicamente de ceros y unos se les denomina alelos y con el nombre de gen a cada valor dentro del alelo. La transformación de la población inicial es la siguiente:

**Para x1**

**Población inicial transformada en binario**

No	X1	X1
	Población	Población
	inicial	inicial
	decimal	binario
1	211	011010011
2	484	111100100
3	457	111001001
4	270	100001110
5	345	101011001
6	294	100100110
7	363	101101011
8	355	101100011
9	294	100100110
10	423	110100111

**Para x2**

**Población inicial transformada en binario**

No	X2	X2
	Población	Población
	inicial	inicial
	decimal	binario
1	1774	011011101110
2	1156	010010000100
3	1844	011100110100
4	1378	010101100010
5	1287	010100000111
6	1963	011110101011
7	1835	011100101011
8	1140	010001110100
9	1616	011001010000
10	1521	010111110001



### Para x3

#### Población inicial transformada en binario

No	X3	X3
	Población	Población
	inicial	inicial
	decimal	binario
1	18	10010
2	20	10100
3	11	01011
4	13	01101
5	16	10000
6	4	00100
7	12	01100
8	3	00011
9	11	01011
10	9	01001

### Para x4

#### Población inicial transformada en binario

No	X4	X4
	Población	Población
	inicial	inicial
	decimal	binario
1	10	0001010
2	81	1010001
3	99	1100011
4	95	1011111
5	17	0010001
6	35	0100011
7	52	0110100
8	35	0100011
9	69	1000101
10	47	0101111

#### 3.7.1.2.2.3. GENERAR PAREJAS Y POSICIÓN DE CRUCE

La pareja de cruce se la genera aleatoriamente con valores que van de acuerdo a la cantidad total de individuos que se hayan generado, en este caso 10, cada valor indica la pareja con la cual cada individuo combinará su información genética, para la posición de cruce se genera una lista teniendo su valor máximo la cantidad total de alelos de cada ambiente. De la siguiente manera:

## Para x1

### Genotipo pareja y posición de cruce

No	X1		Pareja	Posición
	Población	Población		
	inicial	inicial	cruce	cruce
	decimal	binario		
1	211	011010011	1	2
2	484	111100100	1	3
3	457	111001001	1	4
4	270	100001110	10	6
5	345	101011001	6	8
6	294	100100110	7	7
7	383	101101011	4	8
8	355	101100011	9	3
9	294	100100110	1	1
10	423	110100111	9	8

## Para x2

### Genotipo pareja y posición de cruce

No	X2		Pareja	Posición
	Población	Población		
	inicial	inicial	cruce	cruce
	decimal	binario		
1	1774	011011101110	1	6
2	1156	010010000100	1	7
3	1844	011100110100	1	11
4	1378	010101100010	9	7
5	1287	010100000111	8	6
6	1963	011110101011	3	7
7	1835	011100101011	8	11
8	1140	010001110100	3	6
9	1616	011001010000	6	9
10	1521	010111110001	1	5

### Para x3

#### Genotipo pareja y posición de cruce

No	X3		Pareja	Posición
	Población	Población		
	inicial	inicial	cruce	cruce
	decimal	binario		
1	18	10010	1	4
2	20	10100	1	2
3	11	01011	1	3
4	13	01101	10	2
5	16	10000	4	1
6	4	00100	9	3
7	12	01100	8	3
8	3	00011	6	1
9	11	01011	6	1
10	9	01001	3	4

### Para x4

#### Genotipo pareja y posición de cruce

No	X4		Pareja	Posición
	Población	Población		
	inicial	inicial	cruce	cruce
	decimal	binario		
1	10	0001010	1	6
2	81	1010001	1	5
3	99	1100011	1	1
4	95	1011111	6	1
5	17	0010001	10	2
6	35	0100011	3	5
7	52	0110100	10	3
8	35	0100011	4	4
9	69	1000101	4	1
10	47	0101111	6	4

#### 3.7.1.2.2.4. ALMACENAMIENTO DE INFORMACIÓN BINARIA

Cada gen es almacenado en una matriz con la cual nos permite en manejo individual de los datos del material genético de cada individuo.

#### 3.7.1.2.2.5. REALIZAR CRUCE DE INDIVIDUOS.

Para explicar el proceso de cruce vamos a tomar el genotipo de la posición 8 del ambiente X1, es decir el elemento ubicado en la fila número 8 del ambiente X1.

Genotipo No.8 decimal: 355

Genotipo No.8 binario: 101100011

Pareja de cruce: 9

Posición de cruce: 3

Nuevo individuo 1 decimal: 358

Nuevo individuo 1 binario: 101100110

Nuevo individuo 2 decimal: 291

Nuevo individuo 2 binario: 100100011

Dado el cruce del material genético de los dos genotipos (a los cuales denominaremos padres) el resultado son dos nuevos individuos (llamados hijos), lo que da una población ya no de 10 sino de 20. A este proceso de intercambio de material genético o podemos graficar de la siguiente manera:

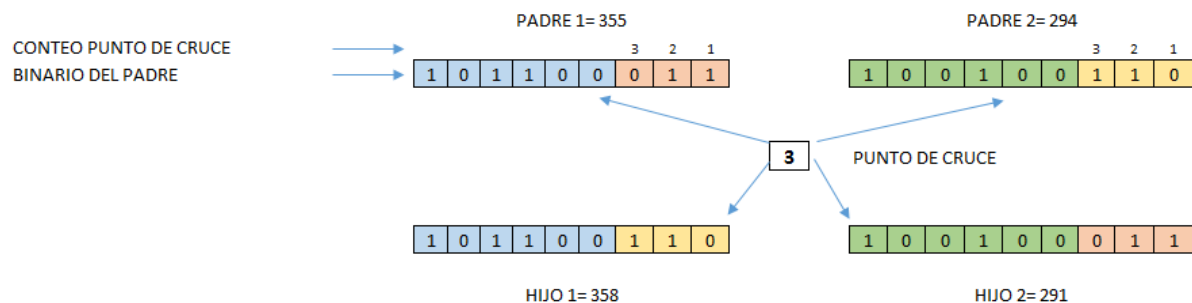


Figura 3.2 Modelo Gráfico de Cruce

El resultado para cada ambiente es el siguiente:

### Para x1

#### Generación de nuevos individuos mediante cruce.

No	X1		Pareja	Posición	Hijo 1	Hijo 1	Hijo 2	Hijo 2
	Población	Población						
	inicial	inicial						
	decimal	binario						
1	211	011010011	1	2	011010011	211	011010011	211
2	484	111100100	1	3	111100011	483	011010100	212
3	457	111001001	1	4	111000011	451	011011001	217
4	270	100001110	10	6	100100111	295	110001110	398
5	345	101011001	6	8	100100110	294	101011001	345
6	294	100100110	7	7	101101011	363	100100110	294
7	363	101101011	4	8	100001110	270	101101011	363
8	355	101100011	9	3	101100110	358	100100011	291
9	294	100100110	1	1	100100111	295	011010010	210
10	423	110100111	9	8	110100110	422	110100111	423

### Para x2

#### Generación de nuevos individuos mediante cruce.

No	X2		Pareja	Posición	Hijo 1	Hijo 1	Hijo 2	Hijo 2
	Población	Población						
	inicial	inicial						
	decimal	binario						
1	1774	011011101110	1	6	011011101110	1774	011011101110	1774
2	1158	010010000100	1	7	010011101110	1282	011010000100	1688
3	1844	011100110100	1	11	011011101110	1774	011100110100	1844
4	1378	010101100010	9	7	010101010000	1380	011001100010	1634
5	1287	010100000111	8	6	010100110100	1332	010001000111	1095
6	1983	011110101011	3	7	011110110000	1988	011100101011	1835
7	1835	011100101011	8	11	010001110100	1140	011100101011	1835
8	1140	010001110100	3	6	010001110100	1140	011100110100	1844
9	1616	011001010000	6	9	011110101011	1983	011001010000	1616
10	1521	010111110001	1	5	010111101110	1518	011011110001	1777

### Para x3

#### Generación de nuevos individuos mediante cruce.

No	X3		Pareja	Posición	Hijo 1	Hijo 1	Hijo 2	Hijo 2
	Población	Población						
	inicial	inicial						
	decimal	binario						
1	18	10010	1	4	10010	18	10010	18
2	20	10100	1	2	10110	22	10000	16
3	11	01011	1	3	01010	10	10011	19
4	13	01101	10	2	01101	13	01001	9
5	16	10000	4	1	10001	17	01100	12
6	4	00100	9	3	00011	3	01100	12
7	12	01100	8	3	01011	11	00100	4
8	3	00011	6	1	00010	2	00101	5
9	11	01011	6	1	01011	11	00101	5
10	9	01001	3	4	01011	11	01001	9

## Para x4

### Generación de nuevos individuos mediante cruce.

No	X4		Pareja	Posición	Hijo 1	Hijo 1	Hijo 2	Hijo 2
	Población	Población						
	inicial	inicial	cruce	cruce	binario	decimal	binario	decimal
	decimal	binario						
1	10	0001010	1	8	0001010	10	0001010	10
2	81	1010001	1	5	1001010	74	0010001	17
3	99	1100011	1	1	1100010	98	0001011	11
4	95	1011111	6	1	1011111	95	0100011	35
5	17	0010001	10	2	0010011	19	0101101	45
6	35	0100011	3	5	1100011	99	1100011	99
7	52	0110100	10	3	0110111	55	0101100	44
8	35	0100011	4	4	0101111	47	1010011	83
9	69	1000101	4	1	1000101	69	1011111	95
10	47	0101111	6	4	0100011	35	0101111	47

#### 3.7.1.2.2.6. REALIZAR MUTACIÓN.

De lo establecido en la sección 2.3.8.4.1 y 2.3.8.4.6 alteramos el código genético, esto se lo realiza a nivel del ADN como ocurriría en la naturaleza, para generar una diversidad en la población, solo mutara un porcentaje de la población en función de un 5% normalmente, pero se puede dar mayor diversidad aumentando este porcentaje, solo mutará un alelo del genotipo seleccionado en cada ambiente.

Al ser aleatoria la generación del gen a mutar en base a la longitud del genotipo, mientras más cerca al gen de la posición 1 más dramático es el cambio, y mientras más se acerca al último el será más parecido al individuo normal, las mutaciones se las puede visualizar de la siguiente manera.

Los cambios por mutación (5%) se produjeron de la siguiente manera:

Para X1, se mutó el individuo de la posición 10

Para X2, se mutó el individuo de la posición 6

Para X3, se mutó el individuo de la posición 9

Para X4, se mutó el individuo de la posición 6

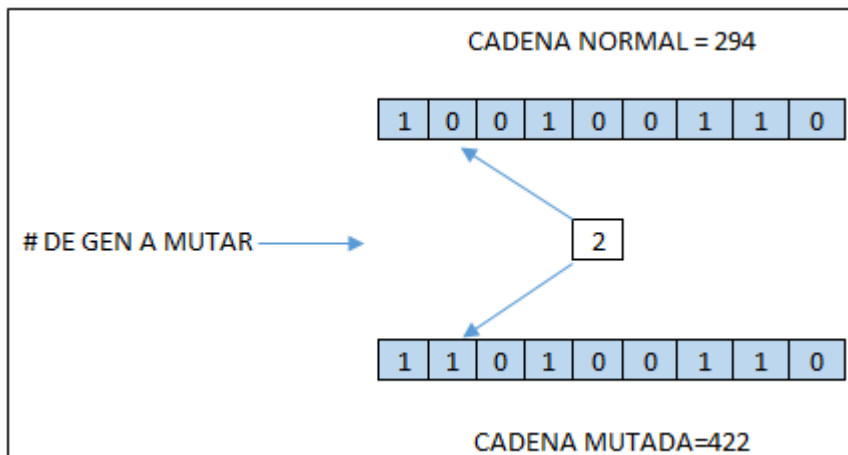


Figura 3.3 Esquema de Mutación de la posición 16 y 10 en la población X1

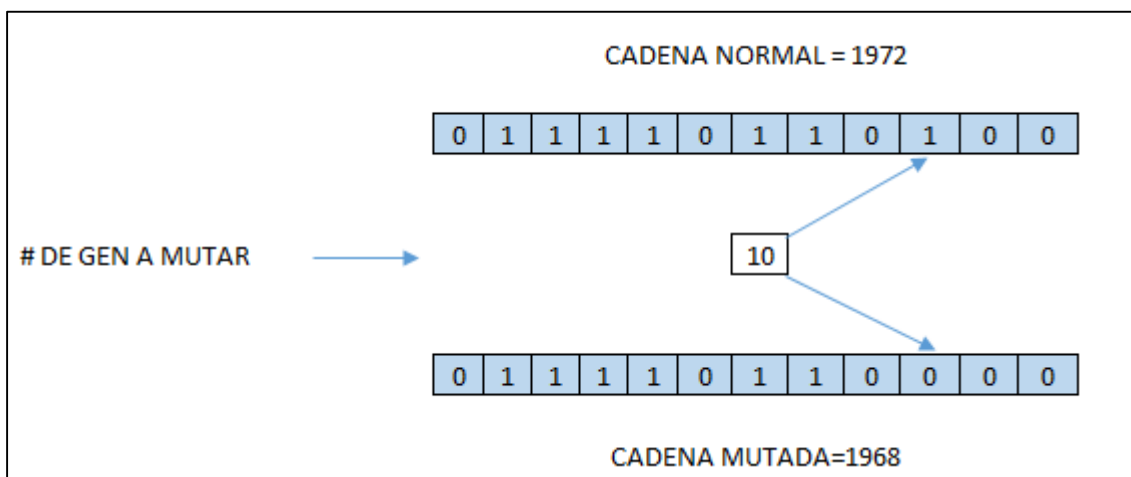


Figura 3.4 Esquema de Mutación de la posición 6 y 12 de la población X2

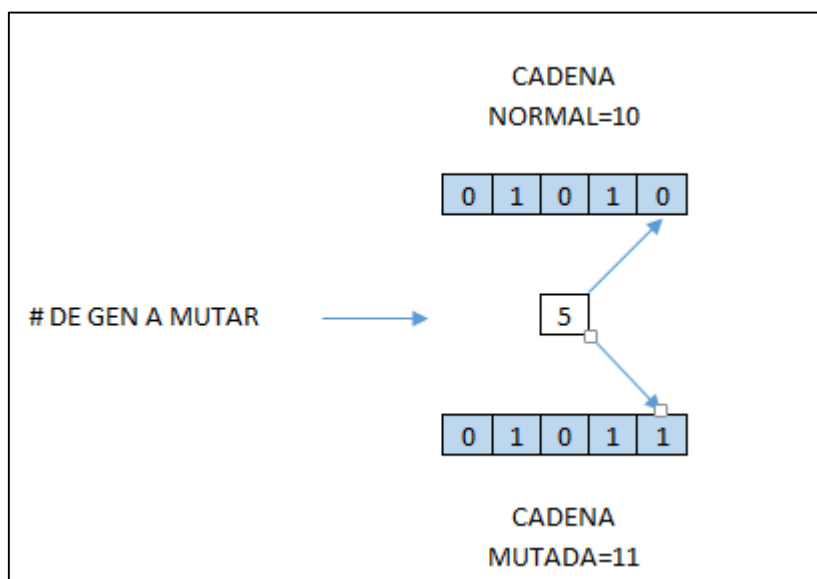


Figura 3.5 Esquema de Mutación de la posición 3 y 7 de la población X3

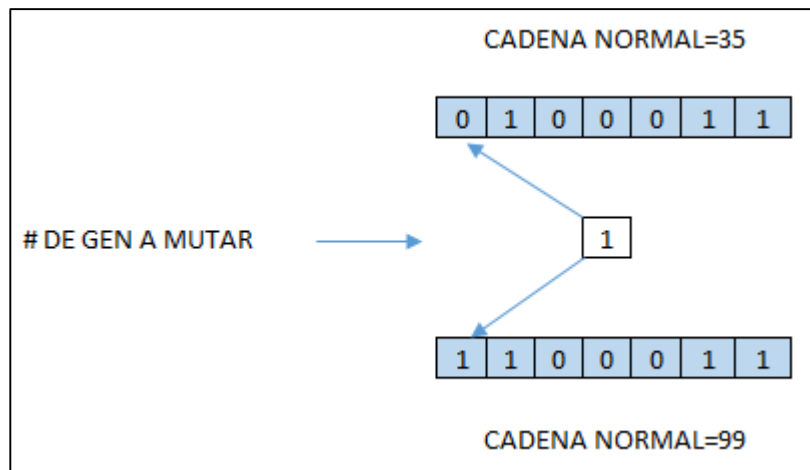


Figura 3.6 Esquema de Mutación de la posición 6 y 10 de la población X4

### 3.7.1.2.2.7. TRANSFORMAR DE BINARIO A DECIMAL

Siguiendo el proceso después de la mutación transformamos la información binaria a decimal de cada genotipo, para poder realizar la evaluación.

De la siguiente manera:

**Para x1 lista de individuos después del proceso de mutación en binario y decimal**

No	X1	X1	Pareja	Posición	Hijo 1	Hijo 1	Hijo 2	Hijo 2
	Población	Población			Hijo 1	Hijo 1	Hijo 2	Hijo 2
	inicial	inicial			binario	decimal	binario	decimal
	decimal	binario			cruce	cruce	binario	decimal
1	211	011010011	1	2	011010011	211	011010011	211
2	484	111100100	1	3	111100011	483	011010100	212
3	457	111001001	1	4	111000011	451	011011001	217
4	270	100001110	10	6	100100111	295	110001110	398
5	345	101011001	6	8	100100110	294	101011001	345
6	294	100100110	7	7	101101011	383	100100110	294
7	383	101101011	4	8	100001110	270	101101011	383
8	355	101100011	9	3	101100110	358	100100011	291
9	294	100100110	1	1	100100111	295	011010010	210
10	423	110100111	9	8	110100110	422	110100111	423

No	Individuo	Gen	Mutado
10	100100110	2	110100110



**Para x2 lista de individuos después del proceso de mutación en binario y decimal**

No	X2		Pareja	Posición	Hijo 1	Hijo 1	Hijo 2	Hijo 2
	Población	Población						
	inicial	inicial						
	decimal	binario						
		cruce	cruce	binario	decimal	binario	decimal	
1	1774	011011101110	1	6	011011101110	1774	011011101110	1774
2	1156	010010000100	1	7	010011101110	1262	011010000100	1688
3	1844	011100110100	1	11	011011101110	1774	011100110100	1844
4	1378	010101100010	9	7	010101010000	1380	011001100010	1634
5	1287	010100000111	8	6	010100110100	1332	010001000111	1095
6	1983	011110101011	3	7	011110110000	1988	011100101011	1835
7	1835	011100101011	8	11	010001110100	1140	011100101011	1835
8	1140	010001110100	3	6	010001110100	1140	011100110100	1844
9	1616	011001010000	6	9	011110101011	1983	011001010000	1616
10	1521	010111110001	1	5	010111101110	1518	011011110001	1777

No	Individuo	Gen	Mutado
6	011110110100	10	011110110000

**Para x3 lista de individuos después del proceso de mutación en binario y decimal**

No	X3		Pareja	Posición	Hijo 1	Hijo 1	Hijo 2	Hijo 2
	Población	Población						
	inicial	inicial						
	decimal	binario						
		cruce	cruce	binario	decimal	binario	decimal	
1	18	10010	1	4	10010	18	10010	18
2	20	10100	1	2	10110	22	10000	16
3	11	01011	1	3	01010	10	10011	19
4	13	01101	10	2	01101	13	01001	9
5	16	10000	4	1	10001	17	01100	12
6	4	00100	9	3	00011	3	01100	12
7	12	01100	8	3	01011	11	00100	4
8	3	00011	6	1	00010	2	00101	5
9	11	01011	6	1	01011	11	00101	5
10	9	01001	3	4	01011	11	01001	9

No	Individuo	Gen	Mutado
9	01010	5	01011

**Para x4 lista de individuos después del proceso de mutación en binario y decimal**

No	X4		Pareja	Posición	Hijo 1	Hijo 1	Hijo 2	Hijo 2
	Población	Población						
	inicial	inicial						
	decimal	binario						
		cruce	cruce	binario	decimal	binario	decimal	
1	10	0001010	1	6	0001010	10	0001010	10
2	81	1010001	1	5	1001010	74	0010001	17
3	99	1100011	1	1	1100010	98	0001011	11
4	95	1011111	6	1	1011111	95	0100011	35
5	17	0010001	10	2	0010011	19	0101101	45
6	35	0100011	3	5	1100011	99	1100011	99
7	52	0110100	10	3	0110111	55	0101100	44
8	35	0100011	4	4	0101111	47	1010011	83
9	69	1000101	4	1	1000101	69	1011111	95
10	47	0101111	6	4	0100011	35	0101111	47

No	Individuo	Gen	Mutado
6	0100011	1	1100011

### 3.7.1.2.2.8. EVALUAR LA FUNCIÓN DE APTITUD

Una vez que el proceso ha logrado concebir una población primaria, desarrollar hijos, generar mutaciones en un porcentaje de ella, se considera hasta este punto que se posee una población diversa, prosiguiendo en este paso al proceso de evaluación, en la cual ingresa cada fenotipo de cada ambiente X1, X2, X3, X4 en la función objetivo o llamada de aptitud que es la siguiente:

$$Z = \left[ [(X2 - (2 * X4)) * X3] + [2 * (X1 * X4)] \right] * \frac{1}{100^3} * 7850 * P1$$

Donde:

Z= costo por metro lineal del Trabe diseñado por el algoritmo genético.

X1= Base del ala del perfil de acero o BF.

X2= Altura del perfil de acero o H.

X3= Espesor del alma del perfil de acero o TW.

X4= Espesor del ala del perfil de acero o TF.

P1= Costo de un Kg de acero estructura en dólares.

El factor  $\frac{1}{100^3}$  transforma el área del perfil de  $mm^2$  a  $m^2$ , y el factor 7850 Kg/m<sup>3</sup> es el peso específico del acero.

Uno de los principales objetivos al implementar un algoritmo genético al diseño de vigas-trabe tipo I armadas mediante placas de acero es el de encontrar la viga más económica y esto se refleja en la ecuación de la función objetivo que es minimizar el costo total que implica la combinación eficiente de materiales.

En forma práctica y siguiendo el proceso actual para una viga los elementos X1, X2, X3, X4 ubicados en la primera fila los siguientes datos:

Para X1=211

Para X2=1774

Para X3=18

Para X4=10

P1= \$3.5(es un valor promedio de una viga soldada entre las principales Mecánicas Industriales en la ciudad de Riobamba cotizados al mes de Marzo del 2015)

Ingresando en la ecuación de la función objetivo para Vigas-Trabe tenemos:

Z= 983.39 dólares

Es decir para estos datos de la primera generación se obtiene un valor de 983.39 dólares por cada metro lineal de viga de las dimensiones descritas (X1, X2, X3, X4), más adelante se presentaran resultados cuando han transcurrido varias generaciones, la información haya pasado por el proceso de evaluación de diseño y se minimice la función objetivo hasta su convergencia.

### 3.7.1.2.2.9. REALIZAR SELECCIÓN DE INDIVIDUOS MEJOR ADAPTADOS

Una vez realizado el proceso anterior se procede a tomar los valores que menor valor produjeron en la función de evaluación, de los ambientes X1, X2, X3, X4, aquí presentamos los individuos mejor adaptados:

#### Individuos mejor adaptados en la función objetivo cumpliendo los requerimientos de diseño LRFD

No	X1	X2	X3	X4	\$
1	211	1774	18	10	983.39
2	483	1262	22	74	2637.38
3	451	1774	10	98	2862.24
4	295	1360	13	95	1957.87
5	294	1332	17	19	911.35
6	363	1968	3	99	2120.63
7	270	1140	11	55	1127.3
8	358	1140	2	47	982.07
9	295	1963	11	69	1670.07
10	422	1518	11	35	1249.23
11	211	1774	18	10	983.39
12	212	1668	16	17	916.35
13	217	1844	19	11	1082.3
14	398	1634	9	35	1152.19
15	345	1095	12	45	1184.45
16	294	1835	12	99	2139.09
17	363	1835	4	44	1069.66
18	291	1844	5	83	1557.72
19	210	1616	5	95	1292.15
20	423	1777	9	47	1508.62

Estos individuos mejor adaptados cada uno y en conjunto han sido verificados que cumplan con las restricciones establecidas, y con el método de diseño proyectado que en nuestro caso es el planteado por el AISC-LRFD. Las restricciones son las que detallamos a continuación:

$$X1 > 0$$

$$X2 > 0$$

$$X3 > 0$$

$$X4 > 0$$

Donde

X1= Base del ala del perfil de acero o BF.

X2= Altura del perfil de acero o H.

X3= Espesor del alma del perfil de acero o TW.

X4= Espesor del ala del perfil de acero o TF.

Los fenotipos de cada ambiente ingresarán en el proceso de diseño detallado en la sección 3.1.6 para proceder a su evaluación, y ser aceptados o no.

#### **3.7.1.2.2.10. PASAR LA POBLACIÓN SELECCIONADA COMO NUEVA POBLACIÓN.**

Finalizando se debe ingresar a la primera fase en la cual se generó la población, con la diferencia de que no se la creará de manera aleatoria, sino que la nueva población son los mejores individuos elegidos en el proceso anterior. Este conjunto de procesos continúa por n generaciones hasta llegar al punto de convergencia de todos los ambientes. Hay que recalcar que el mejor individuo de la generación anterior se mantendrá para la generación venidera, esto para mantener el mejor material genético de cada generación.

A continuación vamos a analizar en diferentes generaciones como la población va evolucionando.

### Individuos de la tercera generación

No	X1	X2	X3	X4	\$
1	423	1776	4	34	978.0
2	363	1802	4	34	868.76
3	359	1840	4	34	865.46
4	363	1840	9	54	1505.41
5	394	1570	4	34	901.18
6	430	1579	5	47	1314.54
7	362	1840	4	46	1107.13
8	430	1776	9	44	1457.05
9	299	1872	1	84	1426.94
10	302	1762	11	67	1603.88
11	423	1776	4	34	978.0
12	431	1777	4	44	1227.69
13	427	1771	4	44	1217.36
14	362	1762	13	67	1914.24
15	366	1634	9	45	1286.82
16	270	1872	9	76	1552.89
17	398	1760	5	45	1213.57
18	295	1824	5	35	808.31
19	366	1570	13	42	1375.45
20	299	1838	4	47	963.88

Se puede observar que ya existen valores que se van repitiendo y minimizando la función objetivo.

### Individuos de la sexta generación

No	X1	X2	X3	X4	\$
1	327	1840	4	34	805.68
2	327	1840	4	34	805.68
3	327	1840	4	34	805.68
4	327	1840	12	34	1195.16
5	355	1776	4	34	850.96
6	359	1840	4	34	865.46
7	327	1840	4	34	805.68
8	363	1785	4	34	866.89
9	325	1840	4	34	801.94
10	363	1840	4	34	872.94
11	327	1840	4	34	805.68
12	327	1840	4	34	805.68
13	327	1840	4	34	805.68
14	327	1840	4	34	805.68
15	359	1835	4	34	864.91
16	327	1840	4	34	805.68
17	359	1840	4	34	865.46
18	327	1840	4	34	805.68
19	363	1840	4	34	872.94
20	363	1840	4	50	1188.57

En la tabla se puede observar que en esta etapa los valores se van uniformizando y la función de aptitud se ha minimizado, y posterior mente con 11 generaciones se ha llegado a la convergencia total, con una función de aptitud minimizada y dimensiones X1, X2, X3, X4 que pasan todos los chequeos del diseño del LRFD.

### Individuos de la novena generación

No	X1	X2	X3	X4	\$
1	325	1808	4	34	798.42
2	327	1840	4	34	805.68
3	325	1808	4	32	763.15
4	325	1840	4	34	801.94
5	325	1904	4	34	808.97
6	325	1840	4	34	801.94
7	325	1840	4	34	801.94
8	325	1840	4	34	801.94
9	327	1840	4	34	805.68
10	325	1840	12	34	1191.43
11	325	1808	4	34	798.42
12	325	1808	4	34	798.42
13	325	1840	4	34	801.94
14	327	1840	4	34	805.68
15	325	1840	4	34	801.94
16	325	1840	4	34	801.94
17	325	1840	4	34	801.94
18	327	1840	4	34	805.68
19	325	1840	4	34	801.94
20	327	1840	4	34	805.68

### Individuos de la onceava generación

No	X1	X2	X3	X4	\$
1	325	1808	4	34	798.42
2	325	1808	4	34	798.42
3	325	1808	4	34	798.42
4	325	1808	4	34	798.42
5	325	1808	4	34	798.42
6	325	1808	4	34	798.42
7	325	1808	20	34	1583.33
8	325	1808	4	34	798.42
9	325	1840	4	34	801.94
10	325	1808	4	34	798.42
11	325	1816	4	32	764.02
12	325	1808	4	34	798.42
13	325	1808	4	34	798.42
14	325	1808	4	34	798.42
15	325	1808	4	34	798.42
16	325	1840	4	34	801.94
17	69	1808	4	34	320.14
18	325	1808	4	34	798.42
19	325	1808	4	34	798.42
20	325	1840	4	34	801.94

En la tabla anterior se muestran los resultados finales del algoritmo genético cuando ha convergido con 11 generaciones, en las que se ha obtenido una sección de base( $b_f$ ) $\times$ 1=325mm, altura ( $h$ )  $\times$ 2=1808mm, con un espesor ( $t_w$ )  $\times$ 3= 4mm, con un espesor ( $t_f$ )  $\times$ 4= 34mm y con un costo por metro lineal de 798.42 dólares, estos datos pasan el diseño requerido por el LRFD para vigas-trabe con solicitaciones de carga requeridas.

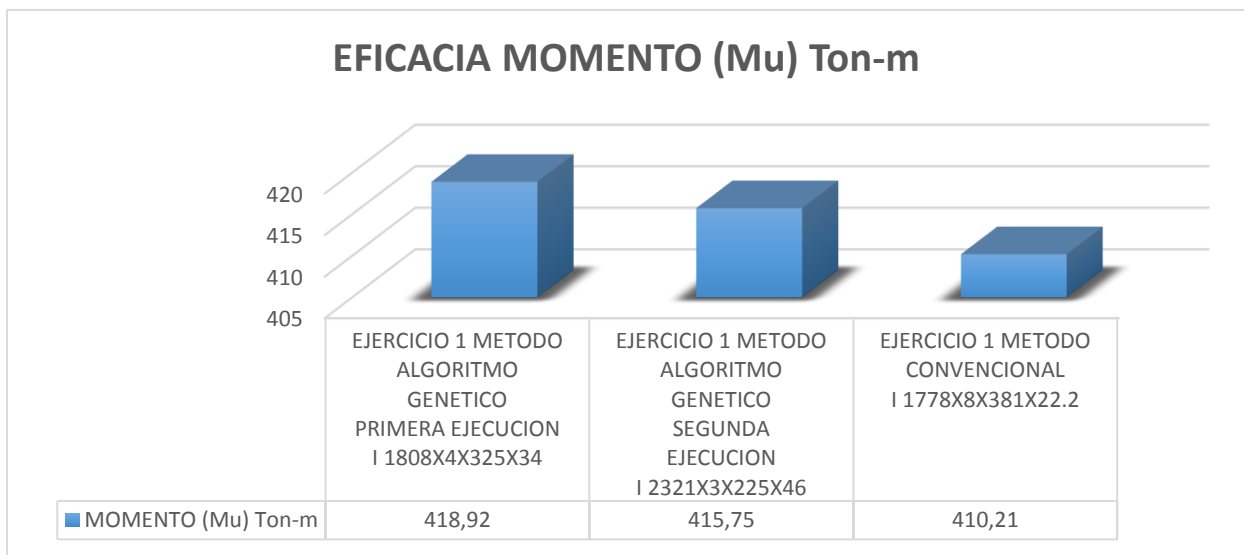
### **3.8. ANALISIS ESTADÍSTICO DE LOS RESULTADOS.**

Una vez obtenida la información necesaria en el cual se aceptará o rechazara la hipótesis para esta investigación se realizara un análisis estadístico de los resultados, tomando como parámetro en primera instancia la similitud de resultados de los ejercicios planteados en la sección 3.6.1 con el resuelto mediante nuestro algoritmo para determinar la idoneidad del proceso de diseño, y en segundo caso comparar el diseño manual de control con el diseño ejecutado mediante el algoritmo genético para de esta manera medir el nivel de eficacia, eficiencia y sobretodo económico de estos dos procesos de diseño.

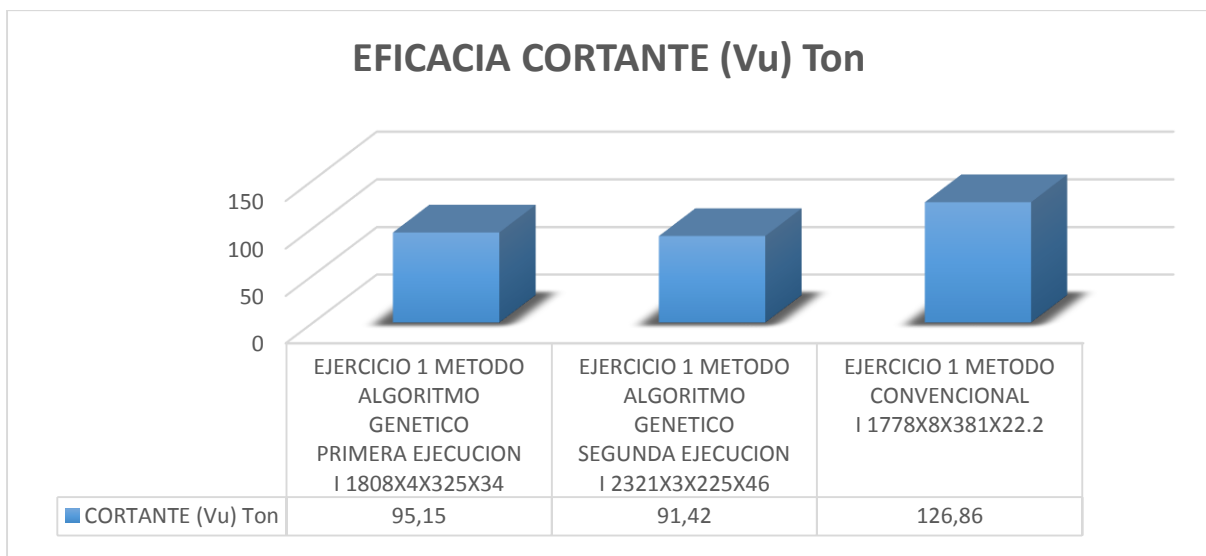
## CAPITULO IV

### 4. RESULTADOS Y DISCUSIÓN

De primera mano se realiza un análisis comparativo entre lo realizado en el programa y ejercicios realizados por autores, para lo cual se tomó como parámetro de referencia los resultados obtenidos de Momentos Últimos y Cortantes Últimos que equivale al total del requerimiento de la sección diseñada del apartado 3.6.1, con los obtenidos aplicando nuestro proceso de diseño que se encuentran en la sección 3.1.6, obteniendo los siguientes resultados:

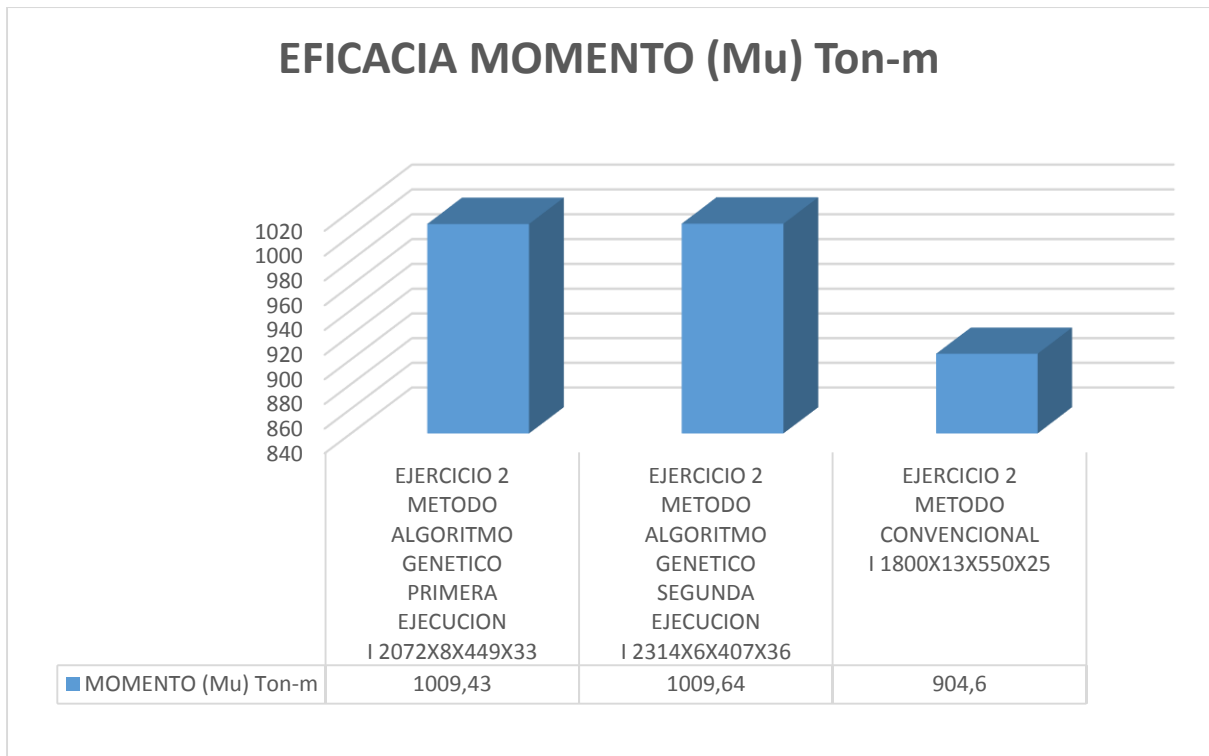


*Figura 4.1 Eficacia Momento*

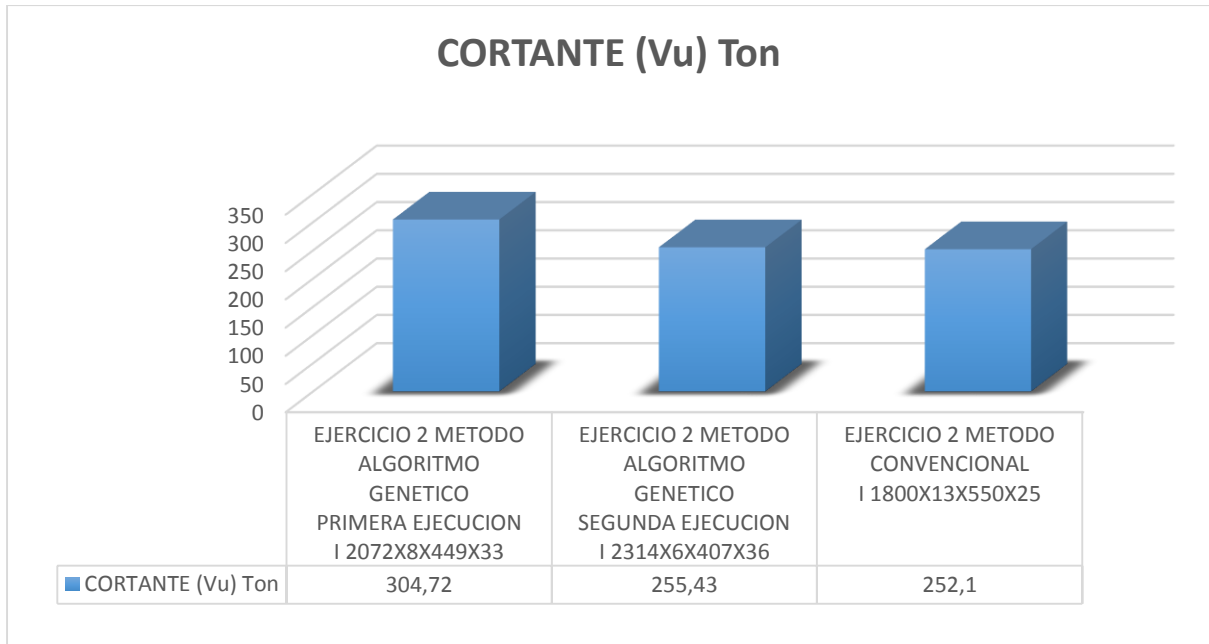


*Figura 4.2. Eficacia Cortante*





*Figura 4.3 Eficacia Momento*



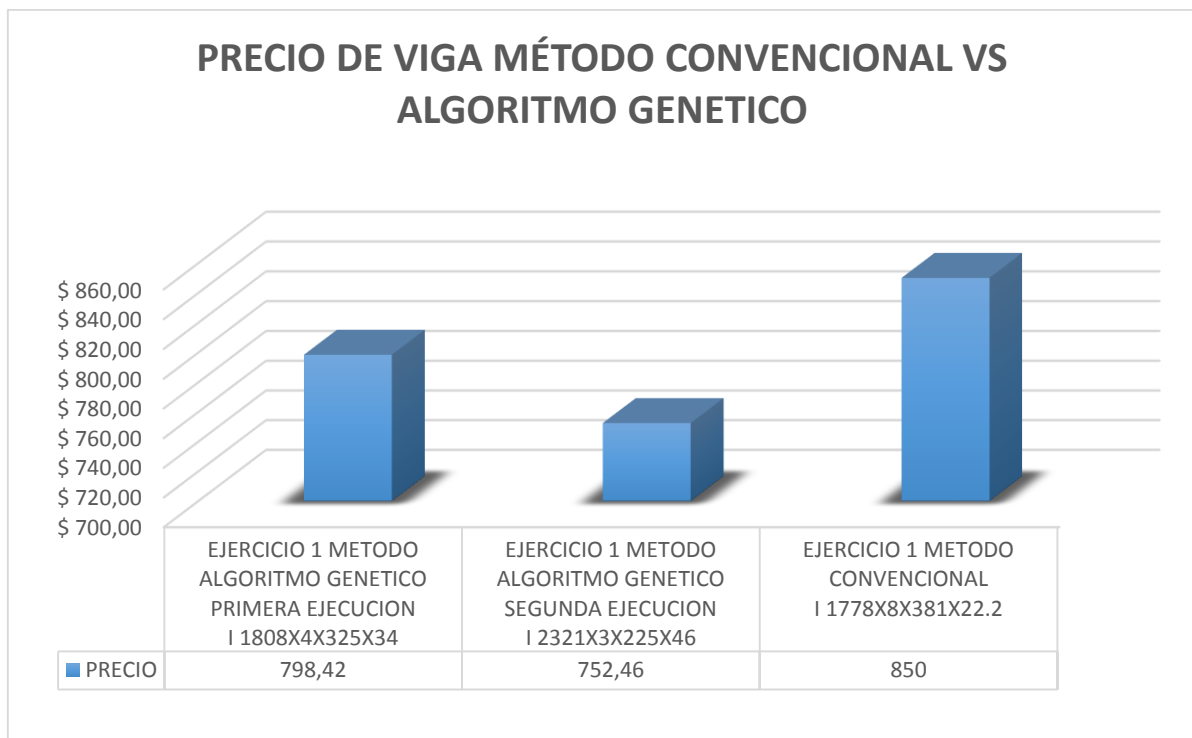
*Figura 4.4. Cortante*

De los resultados conseguidos se puede apreciar que el método de diseño aplicado a nuestro programa arroja resultados similares y/o superiores a los solicitados comparando con los

realizados por otros autores cuyo proceso se detallan en la sección 3.6.1 con lo cual se acepta el proceso de diseño.

Los resultados encuentran sustentados en forma resumida en la sección 3.6.1 y en el respaldo de la sección de apéndices.

Vamos a mostrar resultados obtenidos a través de la aplicación del método de optimización de algoritmos genéticos a dos ejercicios planteados en la sección 3.6.1, el respaldo técnico se encuentra en la sección de apéndices, a continuación un análisis comparativo marcado como objetivo en esta investigación y como medio de validación del estudio en discusión.

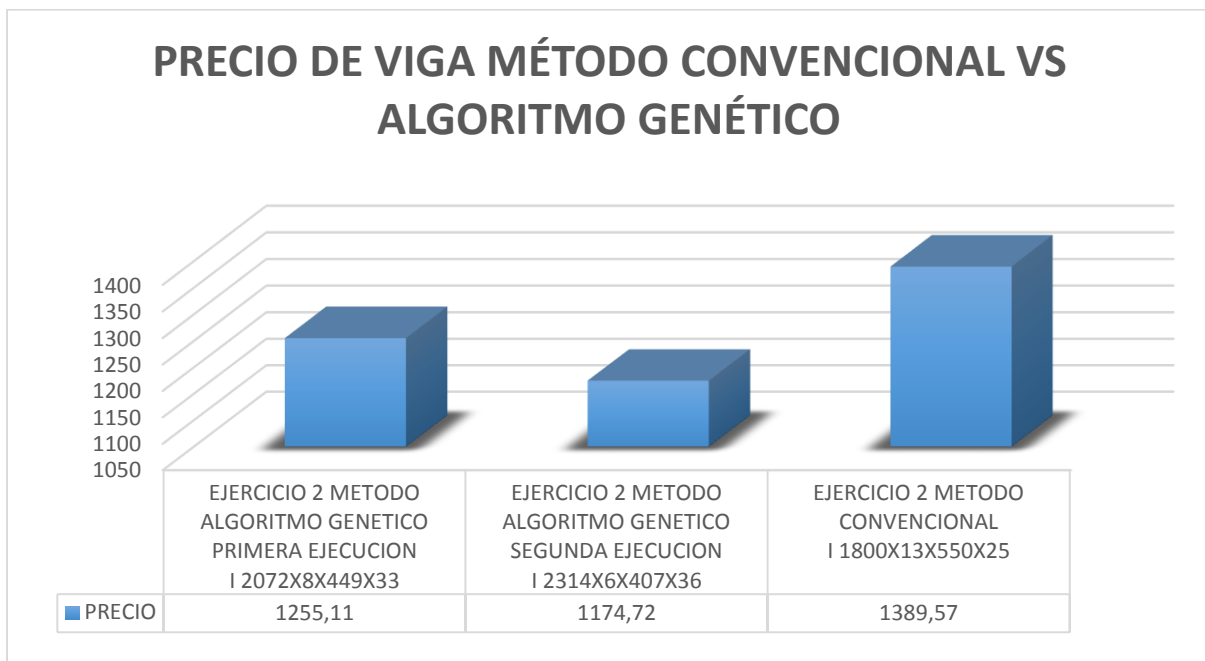


*Figura 4.5. Precio de viga convencional vs Algoritmo Genético*

En este grafico que resuelve el ejercicio 1 resuelto por un autor diferente a esta investigación en la sección 3.6.1 , visualizamos los tres resultados que se obtiene del proceso de diseño, en primer lugar empezando de izquierda a derecha se encuentra el resultado generado por el programa el cual presenta un trabe de 1808mm x 4mm x 325mm x 34mm, en el cual podemos observar que en la primera ejecución del programa nos presenta un precio menor al presentado por el autor, generando un costo por metro lineal de viga de 798.42 dólares. El grafico central posee el resultado calculado mediante nuestro programa en la segunda ejecución, una viga de 2321mm x 3mm x 225mm x 46mm obteniendo un costo por metro lineal viga de 752.46 dólares

mucho menor al presentado por la primera ejecución y al del autor. El grafico final de la derecha es el resultado generado por el autor del cual se calculó el precio por metro lineal cuyas dimensiones del perfil de acero de 1778mm x 8mm x 381mm x 22.2mm con un precio por metro lineal de viga de 850 dólares.

Comparando el costo por metro lineal de viga diseñada a través de la optimización de algoritmos genéticos se obtiene un ahorro de 51.58 dólares en el primer caso y 97.54 dólares en el segundo caso en comparación con la viga-trabe diseñada a través del proceso manual de control. La viga-trabe óptima cumple todas las condiciones que requiere el LRFD para diseñar traves armados mediante placas de acero.



*Figura 4.6. Precio de viga convencional vs Algoritmo Genético*

En este grafico que resuelve el ejercicio 1 resuelto por un autor diferente a esta investigación en la sección 3.6.1 , visualizamos los tres resultados que se obtiene del proceso de diseño, en primer lugar empezando de izquierda a derecha se encuentra el resultado generado por el programa el cual presenta un trabe de 2072mm x 8mm x 449mm x 33mm, en el cual podemos observar que en la primera ejecución del programa nos presenta un precio menor al presentado por el autor, generando un costo por metro lineal de viga de 1255.11 dólares. El grafico central posee el resultado calculado mediante nuestro programa en la segunda ejecución, teniendo una viga de 2314mm x 6mm x 407mm x 36mm obteniendo un costo por metro lineal de viga de

1174.72 dólares mucho menor al presentado por la primera ejecución y al del autor. El gráfico final de la derecha es el resultado generado por el autor del cual se calculó el precio por metro lineal cuyas dimensiones del perfil de acero de 1800mm x 13mm x 550mm x 25mm con un precio por metro lineal de viga de 1389.57 dólares.

Comparando el costo por metro lineal de viga diseñada a través de la optimización de algoritmos genéticos se obtiene un ahorro de 134.43 dólares en el primer caso y 214.85 dólares en el segundo caso en comparación con la viga-trabe diseñada a través del proceso manual de control.

Se debe aclarar que los valores arrojados de las dimensiones de acero tanto del proceso de diseño como de la optimización a través de algoritmos genéticos no trabajan en función de dimensiones de acero estructural comerciales, sino más bien para saber qué nivel de optimización se puede llegar a obtener mediante la aplicación de estos dos métodos, pero tales perfiles se los puede llegar a fabricar en taller.

#### **4.1. COMPARACIÓN MÉTODO CONVENCIONAL VS ALGORITMO GENÉTICO**

Con la finalidad de comprobar la eficacia del algoritmo genético se ha propuesto una serie de ejercicios cuyos resultados serán plasmados en una gráfica con el fin de visualizar su comportamiento y realizar su respectivo análisis.

<b>DESCRIPCIÓN</b>	<b>EJERCICIO 1</b>	<b>MOMENTO</b>	<b>CORTANTE</b>	<b>PRECIO \$</b>
<b>PRIMERA EJECUCIÓN</b>	716x202x20x4	68.26	36.97	296.29
<b>SEGUNDA EJECUCIÓN</b>	696x150x27x4	59.91	35.11	293.1
<b>MÉTODO CONVENCIONAL</b>	905x140x30x5	93.63	57.76	346.87
	<b>EJERCICIO 2</b>	<b>MOMENTO</b>	<b>CORTANTE</b>	<b>PRECIO \$</b>
<b>PRIMERA EJECUCIÓN</b>	785x176x16x2	25.18	20.59	196.12
<b>SEGUNDA EJECUCIÓN</b>	840x173x16x2	25.22	22.09	196.5
<b>MÉTODO CONVENCIONAL</b>	990x160x35x2	126.53	37.73	383.55

	EJERCICIO 3	MOMENTO	CORTANTE	PRECIO \$
<b>PRIMERA EJECUCIÓN</b>	1926x337x16x4	157.84	103.58	504.44
<b>SEGUNDA EJECUCIÓN</b>	1927x450x21x9	408.25	231.94	985.39
<b>MÉTODO CONVENCIONAL</b>	1620x450x40x8	700.46	168.43	1327.59
	EJERCICIO 4	MOMENTO	CORTANTE	PRECIO \$
<b>PRIMERA EJECUCIÓN</b>	1769x215x21x2	61.75	47.22	343
<b>SEGUNDA EJECUCIÓN</b>	1738x185x24x8	66.52	184.84	615.44
<b>MÉTODO CONVENCIONAL</b>	1730x140x65x3	359.31	65.62	631.92
	EJERCICIO 5	MOMENTO	CORTANTE	PRECIO \$
<b>PRIMERA EJECUCIÓN</b>	430x108x17x1	16.36	5.41	111.17
<b>SEGUNDA EJECUCIÓN</b>	443x57x38x2	18.87	10.03	139.19
<b>MÉTODO CONVENCIONAL</b>	520x30x75x2	20.44	10.12	143.97
	EJERCICIO 6	MOMENTO	CORTANTE	PRECIO \$
<b>PRIMERA EJECUCIÓN</b>	704X129X17X4	39.99	36.64	194.14
<b>SEGUNDA EJECUCIÓN</b>	1155X97X19X2	56.07	30.54	162.65
<b>MÉTODO CONVENCIONAL</b>	1060X80X20X3	48.6	41.83	171.99
	EJERCICIO 7	MOMENTO	CORTANTE	PRECIO \$
<b>PRIMERA EJECUCIÓN</b>	1045X100X25X2	63.87	27.21	192.05
<b>SEGUNDA EJECUCIÓN</b>	1196X108X16X2	55.84	31.83	158.92
<b>MÉTODO CONVENCIONAL</b>	1020X130X25X2	78.67	26.52	231.89

*Tabla 4.1. Resultados Comparación Método Convencional Vs Algoritmo Genético*

EJERCICIO #	PRIMERA EJECUCIÓN	SEGUNDA EJECUCIÓN	MÉTODO CONVENCIONAL
1	296.29	293.1	346.87
2	196.12	196.5	383.55
3	504.44	985.39	1327.59
4	343	615.44	631.92
5	111.17	139.19	143.97
6	194.14	162.65	171.99
7	192.05	158.92	231.89

Tabla 4.2. Resumen de Resultados

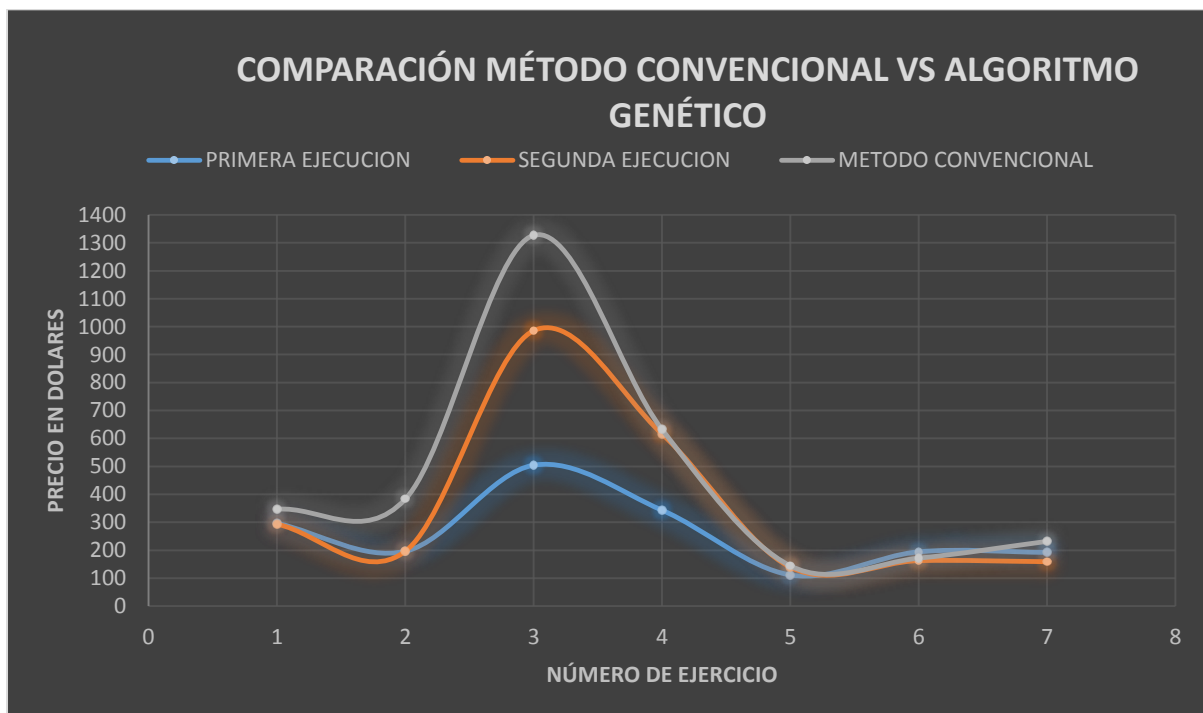


Figura 4.7. Comparación Método Convencional Vs Algoritmo Genético

El gráfico nos demuestra la efectividad del algoritmo genético, aunque en ciertas ejecuciones el método convencional resultara mejor como se puede observar en el ejercicio 6, en el cual el método convencional supero a la primera ejecución del algoritmo genético, lo cual se resolvió volviendo a ejecutar el algoritmo genético, obteniendo el resultado más óptimo.

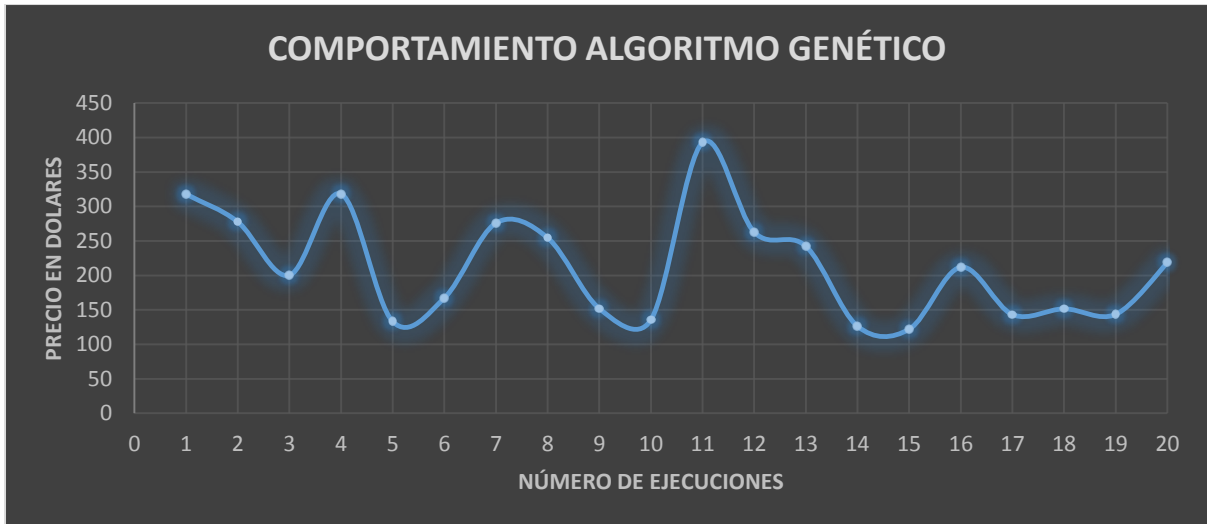
#### 4.2. COMPORTAMIENTO DEL ALGORITMO GENÉTICO

Si bien es cierto al realizar la ejecución del algoritmo genético presenta una serie de resultados, hasta que la función objetivo determine el más óptimo, pero al realizar una nueva ejecución los resultados pueden variar por el hecho de que las parejas que escoge el algoritmo genético para el cruce no siempre son las mismas.

Por los cual en el siguiente ejercicio se ha plasmado en una gráfica el comportamiento del algoritmo genético en sus diferentes ejecuciones.

<b>EJERCICIO #</b>	<b>SECCIÓN</b>	<b>MOMENTO</b>	<b>CORTANTE</b>	<b>PRECIO \$</b>
<b>1</b>	1664X97X35X3	95.1	65.38	317.94
<b>2</b>	1536X128X16X4	71.23	82.25	277.83
<b>3</b>	1242X85X36X1	51.96	16	200.29
<b>4</b>	817X130X33X4	79.72	41.07	318.27
<b>5</b>	794X104X16X2	28.29	20.84	133.31
<b>6</b>	785X128X18X2	38.74	20.48	167.33
<b>7</b>	1536X65X32X4	31	80.5	276.07
<b>8</b>	1122X62X50X3	32.01	41.92	254.58
<b>9</b>	1459X180X12X1	30.07	19.62	152.12
<b>10</b>	1174X164X8X2	25.27	31.66	135.73
<b>11</b>	853X125X45X4	103.91	41.73	392.95
<b>12</b>	903X66X48X4	31.72	44.13	262.77
<b>13</b>	1117X132X17X4	72.65	59.22	242.33
<b>14</b>	1024X144X9X2	37.25	27.51	126.41
<b>15</b>	837X128X11X2	31.06	22.28	122.15
<b>16</b>	1075X60X48X2	71.04	26.77	212.05
<b>17</b>	1024X134X12X2	44.05	27.34	143.31
<b>18</b>	1110X112X15X2	49.96	29.53	151.66
<b>19</b>	1160X93X16X2	47.64	30.84	143.75
<b>20</b>	1544X176X14X2	102.23	41.45	218.7

*Tabla 4.3. Ejecuciones Método con Algoritmo Genético*



*Figura 4.8. Comportamiento Algoritmo Genético*

Como se puede observar en el gráfico la ejecución del algoritmo genético no va a proporcionar datos regulares por el mismo hecho de la elección aleatoria de las parejas para el cruce, por lo cual no se puede determinar una lógica para el programa y su utilización está destinada para profesionales con criterio, los cuales optaran por el resultado más óptimo.



## CAPITULO V

### 5. CONCLUSIONES Y RECOMENDACIONES.

#### 5.1. CONCLUSIONES

Mediante la aplicación de la técnica de algoritmos genéticos al diseño vigas-trabe tipo I armada mediante placas de acero se ha llegado a las siguientes conclusiones:

1. Un algoritmo genético ha sido desarrollado para diseñar una viga-trabe tipo I armada mediante placas de acero ya que cumple con las condiciones de aplicabilidad las mismas que son:
  - Su espacio de búsqueda está delimitado, ya que las dimensiones que se generan en el diseño, como son base altura y se las puede limitar mediante restricciones, las mismas que están en función de la relación base altura y de los requerimientos arquitectónicos para el dimensionamiento.
  - Se puede definir la función aptitud, la misma que se basa en el costo de la viga-trabe.
  - Las posibles soluciones o individuos pueden ser codificados mediante código binario.
2. Se ha establecido el proceso de diseño de vigas-trabe tipo I armadas mediante placas de acero, con la aplicación de algoritmos genéticos y cumpliendo con la normativa AISC-LRFD, habiendo descrito en forma ordenada y practica cada paso para la implementación del mismo en la sección 3.1.6, aclarando que para agilizar el proceso de cálculo se recurrió a la aplicación de un paquete informático que permita el manejo de gran cantidad de operaciones.

3. Se desarrolló un completo estudio acerca del funcionamiento y aplicación de un algoritmo genético en aplicaciones de ingeniería, prueba de ello es la generación de herramientas computacionales que se ha desarrollado para la aplicación didáctica de esta investigación
4. Se fundamentó la interrelación de las ecuaciones fundamentales del diseño de vigas-trabe tipo I armadas mediante placas de acero del método AISC-LRFD con el proceso que conlleva la estructura de un algoritmo genético, generando un esquema base para aplicaciones ingenieriles planteado en la sección 6.5.
5. Se ha implementado con éxito la aplicación del algoritmo genético al diseño de vigas-trabe tipo I armadas mediante placas de acero en el programa denominado DEAC-TB, el mismo que desde su inicio fue planteado para para que sea utilizado de forma práctica por el usuario aplicado a elementos reales, proveyendo de esta manera una herramienta al ingeniero que le permita aumentar la eficacia en el diseño estructural, ya que para el mismo elemento pueden existir varias soluciones alternativas pero no siempre económicas, objetivo que se logra con la aplicación de este sistema de optimización.
6. Se confirmó la hipótesis planteada mediante el análisis realizado en el capítulo 4 de esta investigación, obteniéndose datos con alto porcentaje de validez, ya que se verificó que el método aplicado produjo resultados con un alto grado de eficacia en relación a Momentos y Cortantes que resistan las solicitaciones planteadas por otros autores. También se pudo analizar que las vigas-trabe armadas diseñadas mediante algoritmos genéticos fueron las más económicas tanto en tiempo como en dinero, en comparación con las diseñadas con el procedimiento de control tradicional, ya que lo que se busca es generar las dimensiones que minimizan la función objetivo del algoritmo genético y que cumplan con el método de diseño propuesto por la AISC-LRFD.

- 7.** Analizando el grafico 4.5 se llega a lo siguiente:  
Que el método de optimización de algoritmos genéticos produce un ahorro de 97.54 dólares comparados con el diseño de control manual.
  
- 8.** Analizando el grafico 4.6 se llega a lo siguiente:  
Que el método de optimización de algoritmos genéticos produce un ahorro de 214.85 dólares comparados con el diseño de control manual.
  
- 9.** Se puede mencionar que el diseño de vigas tipo trabe armadas mediante placas de acero con la aplicación de algoritmos genético genera secciones más económicas que cumplen con los requerimientos del LRFD que el método convencional.
  
- 10.** Los valores arrojados del diseño planteado en esta investigación generan resultados similares a los de otros autores los cuales se demostró en sección 4, expresando la eficacia de nuestro diseño.
  
- 11.** A través de las etapas de prueba y etapa de aplicación del algoritmo genético se observa la rapidez proceso de diseño que nos otorga esta herramienta en comparación con el método de diseño convencional.
  
- 12.** Analizando el grafico 4.7 se llega a lo siguiente:  
Se realizó 7 ejercicios por el método convencional y utilizando algoritmos genéticos de lo cual se determinó que el algoritmo genético es el más eficaz tanto por los resultados que presenta, como la rapidez de calculo que se podría decir que es inmediata, no siempre los resultados que presenta son los más óptimos por lo cual se requiere de cierto criterio de diseño, como es el caso del ejercicio 6 en el cual en la primera ejecución presenta un costo mayor que el presentado por el método convencional y realizando una nueva ejecución del programa nos muestra un resultado mejor que el del método convencional.  
Cabe aclarar que para la ejecución de los 7 ejercicios se utilizó 10 individuos, un porcentaje de mutación del 5% y un precio por metro lineal de 3.5 dólares americanos

**13.** Analizando el grafico 4.8 se llega a lo siguiente:

Al observar la gráfica se puede notar que tienen una dispersión irregular por el hecho de que las parejas para el cruce se escogen aleatoriamente, por lo cual el resultado más óptimo estará regido por el criterio del usuario el cual realizara el número de ejecuciones necesarias para su diseño, por el hecho de que los resultados se obtengan de manera inmediata, para las 20 ejecuciones se tomó un tiempo aproximado de 3 minutos cuyo resultado más óptimo se presentó en la quinceava ejecución.

Todos los resultados presentados por el programa cumplen con los requerimientos para el diseño de una viga-trabe tipo I, a diferencia del método convencional que al no cumplir con algún parámetro en cualquiera de sus iteraciones se procede a descartar la sección del perfil y por ende realizar una nueva iteración la cual aunque cumpla con todos los parámetros de diseño no necesariamente será la más óptima.

## 5.2. RECOMENDACIONES

1. Para la aplicación de algoritmos genéticos a cualquier ámbito en el que se pretenda optimizar elementos, procesos o funciones, y estas tengan un alto grado de complejidad es imprescindible en primera instancia la aplicación de un algoritmo genético básico, y luego ir determinando experimentalmente los parámetros más adecuados de ajuste tales como el tipo de cruce, la tasa de mutación, el orden de las funciones de selección ya que no se pueden establecer parámetros generales, puesto que estos varían de acuerdo a varios parámetros propios de cada estudio .
2. Lo detallado en el proceso de resultados manifiesta que las dimensiones arrojadas por el algoritmo genético no están en base a dimensiones comerciales de perfiles, esto se lo realizó recomendablemente para conseguir la respuesta de la optimización que genera el algoritmo genético a su máximo estado minimizando la función objetivo que en este caso evalúa el costo.
3. La aplicación de métodos de optimización debería considerarse materia práctica en todos los diseños, ya que con esto se dota al ingeniero de un recurso fundamental para manejar el recurso económico y enrolarse en el campo competitivo, cualidades necesarias en nuestra sociedad.
4. Para la utilización del software DEAC-TB recomendamos que el navegador predeterminado sea Google Chrome en versión actualizada por poseer herramientas visuales que ayudan en el proceso de impresión de resultados.
5. El incentivo de la investigación es una necesidad de la universidad en los actuales momentos, y en particular dentro de la ingeniería, desarrollando investigaciones de carácter práctico que realmente sean aplicables a la realidad y fomenten la innovación tecnológica que el país demanda.

## **CAPITULO VI**

### **6. PROPUESTA**

#### **6.1.TÍTULO DE LA PROPUESTA**

“Desarrollo del software denominado DEAC-TB para el diseño de vigas-trabe tipo I armadas mediante placas de acero utilizando algoritmos genéticos “

#### **6.2.TÍTULO DE LA PROPUESTA**

Mediante la implementación del software DEAC-TB se dotará de una herramienta informática de utilidad práctica en el diseño estructural, la misma que será capaz realizar un proceso de optimización utilizando algoritmos genéticos (AG) para cargas y momentos requeridos en la viga-trabe ingresados al programa, logrando con este último proceso de AG reducir el tiempo de cálculo para el diseño y revisar los chequeos pedidos por el método del LRFD.

Mediante la aplicación de la técnica de optimización de algoritmos genéticos se obtendrán elementos económicos y que cumplan con la normativa LRFD.

#### **6.3.OBJETIVO GENERAL**

##### **6.3.1. GENERAL.**

Desarrollar el software DEAC-TB para el diseño de vigas-trabe tipo I armadas mediante placas de acero utilizando algoritmos genéticos “

##### **6.3.2. ESPECÍFICOS.**

- Obtener una herramienta informática que permita optimizar el costo económico del diseño de vigas-trabe tipo I armadas mediante placas de acero utilizando algoritmos genéticos.

- Obtener elementos eficientes en el campo estructural y minimizar el tiempo de cálculo mediante la aplicación del software DEAC-TB

## **6.4.DESCRIPCIÓN DE LA PROPUESTA**

### **6.4.1. PROGRAMA COMPUTACIONAL DEAC-TB PARA EL DISEÑO DE VIGAS-TRABE TIPO I ARMADAS MEDIANTE PLACAS DE ACERO UTILIZANDO ALGORITMOS GENÉTICOS.**

El software DEAC-TB es un programa computacional didáctico y práctico que diseña vigas-trabe tipo I armadas mediante placas de acero utilizando algoritmos genéticos, todas las herramientas en las que se diseñó el sistema informático son de última generación y de software libre orientados al a la web, teniendo como lenguaje de programación: JSF2 (Java Server Faces) y como servidor: GlassFish 4.

Contiene una interfaz gráfica de cómoda comprensión y manipulación de la información ingresada, así como de la información obtenida a través del análisis.

El análisis que se realiza en este programa es para elementos tipo viga-trabe de estructuras reales, partiendo de datos iniciales de carga y momentos. Maneja internamente el proceso de diseño manejado por la AISC –LRFD, dando la oportunidad de realizar un diseño óptimo económicamente basado en Algoritmos Genéticos.

### 6.4.1.1.PROCESO LÓGICO DEL SOFTWARE DEAC-TB

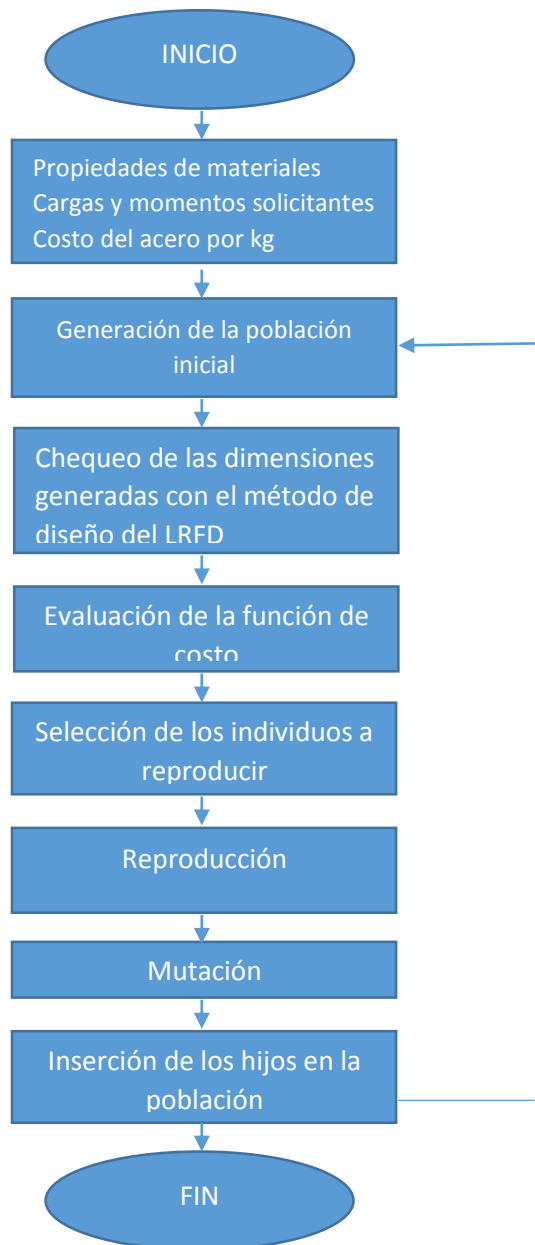


Figura 6.1. Esquema de un Algoritmo Genético Básico



## 6.4.2. ESTRUCTURACIÓN DEL SOFTWARE DEAC-TB.

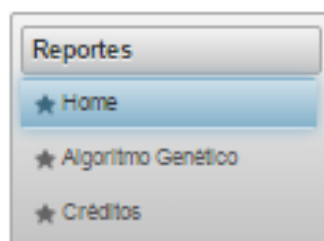


*Figura 6.2. Pantalla Principal del programa DEAC-TB*

En este apartado presentamos la interfaz del software DEAC-TB y explicamos cada una de sus partes.

Cabe resaltar que el control principal lo sigue manteniendo el criterio del ingeniero, ya que dentro de las convicciones con las que se realizó este software y en forma global se tiene la certeza de que los resultados que arroje un programa no tendrán validez si no están controlados por el buen juicio y conocimiento de quien lo usa.

El programa inicializa con un grupo de selección principal el cual es:



*Figura 6.3. Grupo de Selección*

Este menú permite seleccionar si se requiere hacer un diseño óptimo a través de algoritmos genéticos, regresar al menú principal o mostrar los créditos.

### 6.4.2.1.PAGINA ALGORITMO GENETICO.

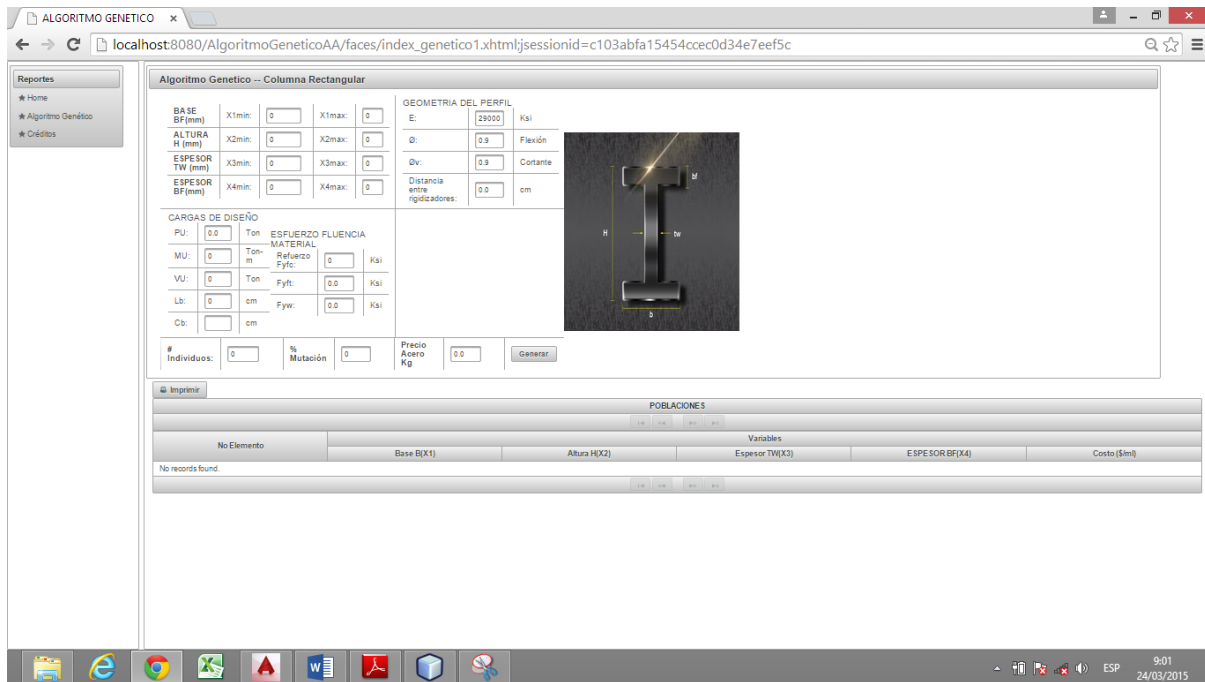


Figura 6.4 Página Control de Datos

Accediendo internamente a la página de **ALGORITMO GENETICO** de diseño se encuentra los siguientes grupos secundarios.

#### A. Ajuste del rango de generación de poblaciones

BASE BF(mm)	X1min:	<input type="text" value="0"/>	X1max:	<input type="text" value="0"/>
ALTURA H (mm)	X2min:	<input type="text" value="0"/>	X2max:	<input type="text" value="0"/>
ESPESOR TW (mm)	X3min:	<input type="text" value="0"/>	X3max:	<input type="text" value="0"/>
ESPESOR BF(mm)	X4min:	<input type="text" value="0"/>	X4max:	<input type="text" value="0"/>

Figura 6.5. Ajuste del rango de generación de poblaciones.

Estos son los límites para poder generar la población inicial que el algoritmo genético necesita para iniciar. La Fig. 6.5. Muestra donde se colocan los valores mínimo y máximo para generar dicha población para el algoritmo genético en el diseño.

## B. Datos de diseño.

DATOS DE DISEÑO		
E:	<input type="text" value="29000"/>	Ksi
$\phi$ :	<input type="text" value="0.9"/>	Flexión
$\phi_v$ :	<input type="text" value="0.9"/>	Cortante
Distancia entre rigidizadores:	<input type="text" value="0.0"/>	cm

Figura 6.6. Datos del diseño del perfil de acero

Esta parte del programa da la oportunidad de seleccionar las características de diseño con datos que son necesarios para determinar si nuestra viga cumple o no con los requerimientos planteados también nos pide ingresar distancia entre rigidizadores en caso que así lo disponga el diseño.

## C. Datos de materiales utilizados.

ESFUERZO FLUENCIA MATERIAL		
Refuerzo Fyfc:	<input type="text" value="50.0"/>	Ksi
Fyft:	<input type="text" value="0.0"/>	Ksi
Fyw:	<input type="text" value="0.0"/>	Ksi

Figura 6.7. Esfuerzo, Fluencia y Material del perfil de acero.

Aquí se ingresan las propiedades de los materiales acero teniendo en cuenta las limitantes que requiere el LRFD para el diseño de este tipo de vigas, en el caso de existir dos o más materiales se ingresarán tanto para el ala como el alma.

#### D. Datos de cargas y momentos.

CARGAS DE DISEÑO		
PU:	<input type="text" value="0.0"/>	Ton
MU:	<input type="text" value="0"/>	Ton-m
VU:	<input type="text" value="0"/>	Ton
Lb:	<input type="text" value="0"/>	cm
Cb:	<input type="text" value="0"/>	cm

Figura 6.8. Datos de Cargas y Momentos

#### E. Número de Individuos

# Individuos:	<input type="text" value="0"/>
---------------	--------------------------------

Figura 6.9 Número de Individuos

Aquí se ingresan la cantidad de individuos con los que va trabajar el algoritmo desde la generación de la población inicial, mientras más generoso sea este valor el programa tendrá una cantidad considerable de datos para trabajar, teniendo en cuenta que si lo sobrecargamos el tiempo de cálculo será mayor por la gran cantidad de proceso que realizará internamente, dependerá de la capacidad del computador en donde se lo ejecute.

#### F. Porcentaje de mutación.

% Mutación	<input type="text" value="0"/>
------------	--------------------------------

Figura 6.10 Porcentaje de Mutación

Este factor otorga variedad al proceso como se lo detallo en la sección 2.3.8.4.6

## G. Datos de costo de materiales.

Precio Acero Kg	<input type="text" value="0"/>	<input type="button" value="Generar"/>
-----------------	--------------------------------	--

Figura 6.11 Datos de costo de materiales del perfil de acero

Aquí se ingresa el precio de cada uno de los materiales, el precio del metro cubico de hormigón y el precio por kg de acero. Estos datos servirán para obtener el precio del metro lineal de viga diseñada. Y de los cual se vale el algoritmo genético para calcular la función de aptitud.

## H. Generar.

<input type="button" value="Generar"/>
--

Figura 6.12 Botón Generar

Este botón inicializa el proceso de cálculo y muestra los resultados

## I. Visualización de resultados.

POBLACIONES											
No Elemento	Base B(X1)				Altura H(X2)			Variables			
Generacion 1	Detalle x1	Detalle x2	Detalle x3	Detalle x4	Función	Propiedades	Ala	Pandeo	Optimo	Costo (\$/ml)	
1	180				2098				18	53	1954.46
2	209				2049				15	86	2415.88
3	180				2098				18	53	1954.46
4	209				2049				15	86	2415.88
5	180				2098				18	53	1954.46
6	209				2049				15	86	2415.88
7	180				2098				18	53	1954.46
8	209				2049				15	86	2415.88
9	180				2098				18	53	1954.46
10	209				2049				15	86	2415.88
11	180				2098				18	53	1954.46
12	209				2049				15	86	2415.88
13	180				2098				18	53	1954.46
14	209				2049				15	86	2415.88
15	180				2098				18	53	1954.46
16	209				2049				15	86	2415.88
17	180				2098				18	53	1954.46
18	209				2049				15	86	2415.88
19	180				2098				18	53	1954.46
20	209				2049				15	86	2415.88

Figura 6.13. Visualización de Resultados

El algoritmo genético muestra todas las generaciones calculadas hasta llegar a la convergencia del proceso encontrando la sección óptima que minimiza la función de costo, se pueden visualizar todos los pasos que realiza el algoritmo hasta llegar a la optimización. Las dimensiones óptimas se encuentran en la última generación y se puede visualizar todos los chequeos aceptados por el LRFD.

### **6.5. MONITOREO Y EVALUACION DE LA PROPUESTA.**

Con el desarrollo del software DEAC-TB se obtuvieron los siguientes avances en el sistema de diseño de vigas-trabe tipo I armadas mediante placas de acero.

- El costo de viga-trabe por metro lineal se reduce notablemente como se lo demostró en la sección 4 ya que se encontró la sección más económica.
- El tiempo empleado en el diseño de una viga-trabe tipo I armadas redujo de un tiempo de 8 a 10 minutos en una hoja de cálculo normal por un tiempo de 30 a 60 segundos completamente chequeado el diseño.
- Los resultados de las secciones obtenidas de los perfiles de acero no son dimensiones de perfiles comerciales, los valores arrojados por el programa comprueba que se puede llegar a optimizar un elemento, teniendo en cuenta que las dimensiones de los perfiles dado por el software se los puede llegar a fabricar en taller.
- Las dimensiones óptimas que se obtienen del diseño con algoritmos genéticos cumplen en su totalidad con los requerimientos del LRFD para el diseño de vigas-trabe tipo I armadas mediante placas de acero.

## CAPITULO VII

### 7. BIBLIOGRAFIA.

1. McCormac. Diseño de Estructuras de Acero 2da Edición. México, 2002
2. VALENCIA, Gabriel. Estructuras de Acero Diseño con Factores de Carga y Resistencia. 2da. Edición, Colombia 2004, p. 392-394.
3. TIMOSHENKO S. Resistencia de Materiales, Decimocuarta Edición
4. GALAMBOS Theodore V. Diseño de estructuras de acero con LRFD. 1999
5. AMERICAN INSTITUTE OF STEEL CONSTRUCTION. Manual of Steel Construction, Load & Resistance Factor Design.
6. GOLDBERG, David E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning
7. RODRÍGUEZ Piñero Piedad Tolmos, Introducción A Los Algoritmos Genéticos Y Sus Aplicaciones.
8. ASHQUI Laguna Felix Armando, Diseño a flexión de vigas rectangulares de hormigón armado basado en la normativa ecuatoriana y mediante la aplicación de algoritmos genéticos, 2011.
9. GÓMEZ Otero Patricia, Estudio de los Algoritmos genéticos para el desarrollo de sistemas de control.
10. Andrés Donoso-Stalin Parreño, Diseño a flexocompresión biaxial columnas de acero rellenas de hormigón basado en la normativa del AISC aplicando el método del LRFD con la aplicación de algoritmos genéticos, 2013.
11. TAPIA, Edison. Diseño de Estructuras en Acero, Universidad Nacional de Chimborazo 2011

## **CAPITULO VIII**

### **8. APENDICES Y ANEXOS.**