

UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA
FACULTAD DE INGENIERÍA EN INFORMÁTICA Y SISTEMAS



TESIS PARA TÍTULO PROFESIONAL

**ALGORITMO GENÉTICO PARA LA ELABORACIÓN DE HORARIOS DE
CLASES EN UNIVERSIDADES CON RECURSOS COMPARTIDOS, CASO
DE ESTUDIO: UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA**

**PARA OBTENER EL TÍTULO PROFESIONAL DE INGENIERO EN
INFORMÁTICA Y SISTEMAS**

ELABORADO POR

NORRIS LEE BRAVO HERMITAÑO

ASESOR

ING. RONALD EDUARDO IBARRA ZAPATA

TINGO MARÍA – PERÚ

2019



PARTE 1. FASE INICIAL

Siendo las 09:45 am horas del día 25 de Julio de 2019; en la Sala de Grados de la FIIS, se instala el jurado calificador conformado por:

Jurado 1: Dr. Walter Rubén BERNUY BLANCO (Presidente)

Jurado 2: M.Sc. Javier RODRIGUEZ CRUZ

Jurado 3: Ing. Pedro Crisólogo TRUJILLO NATIVIDAD

Oficializado mediante **Resolución N.º 051-2019-D-FIIS-UNAS** del 29 de marzo de 2019, para el proceso de sustentación del informe final de Tesis del bachiller **Norris Lee BRAVO HERMITAÑO**, titulado: **“ALGORITMO GENETICO PARA LA ELABORACIÓN DE HORARIOS DE CLASES EN UNIVERSIDADES CON RECURSOS COMPARTIDOS. CASO DE ESTUDIO: UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA”**. ASESOR: Ing. Ronald Eduardo Ibarra Zapata.

Se manifiesta que el bachiller cumple con los requisitos exigidos de Ley y se le invita a disertar su Tesis por espacio de 30 minutos, asimismo se dispondrá de igual tiempo para la absolver preguntas y sugerencias.

PARTE 2. FASE DE PREGUNTAS Y RESULTADO

Culminada la exposición se inicia la fase de preguntas por parte del jurado calificador; también se invita a los asistentes a formular preguntas sobre el tema de Tesis.

Absueltas todas las peticiones, el jurado calificador procede a deliberar en privado la calificación y resultado.

Concluida la deliberación y en presencia del público asistente, el jurado calificador anuncia que el resultado de la Sustentación de Tesis es: Aprobada por Mayoría

(NOTA: consignar una de la siguientes: DESAPROBADO, APROBADO POR MAYORIA o APROBADO POR UNANIMIDAD)

Con calificativo de: Buena

(NOTA: consignar una de la siguientes: EXCELENTE, MUY BUENO, BUENO, DEFICIENTE, MUY DEFICIENTE)

Por lo que se comunicará a las instancias correspondientes para el trámite respectivo.

PARTE 3. CONFORMIDAD

De todo lo mencionado se firma al pie en señal de conformidad, siendo las 11:54 horas se da por finalizada la ceremonia de Sustentación de Tesis.

Firma:	Firma:	Firma:
Jurado 1: <u>Walter Bernuy B.</u>	Jurado 2: <u>Javier Rodriguez C.</u>	Jurado 3: <u>Pedro C. Trujillo Natividad</u>
Firma:		
Sustentante: <u>NORRIS LEE BRAVO HERMITAÑO</u>		
		Asesor: <u>Ronald Ibarra Z</u>

UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA

REGISTRO DE TESIS PARA LA OBTENCIÓN DEL TÍTULO UNIVERSITARIO, INVESTIGACIÓN
DOCENTE

I. DATOS GENERALES DE PREGRADO

Universidad : Universidad Nacional Agraria de la Selva
Facultad : Ingeniería en Informática y Sistemas
Título de Tesis : ALGORITMO GENÉTICO PARA LA ELABORACIÓN DE HORARIOS DE CLASES EN UNIVERSIDADES CON RECURSOS COMPARTIDOS, CASO DE ESTUDIO: UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA.
Autor : Norris Lee Bravo Hermitaño
Asesor de Tesis : Ing. Ronald Eduardo Ibarra Zapata
Escuela Profesional : Ingeniería en Informática y Sistemas
Programa de Investigación : Ingeniería de Software
Línea(s) de Investigación : Computación
Eje Temático de Investigación : Optimización de Algoritmos
Lugar de Investigación : Tingo María - Distrito Rupa Rupa – Provincia de Leoncio Prado – Región Huánuco
Duración : **Fecha de Inicio** : 16/04/2018
Término : 13/03/2019
Financiamiento :
FEDU :
Propio : X
Otros :


Firma

17/12/2019
Fecha

DEDICATORIA

A mi madre Alisol Hermitaño Chusing
por el apoyo incondicional y
enseñarme la persistencia para
lograr mis proyectos.

AGRADECIMIENTOS

A Dios por la vida y mostrarme que las casualidades fueron parte de un plan para encontrar mi propósito de vida y que los fracasos son parte del aprendizaje.

Al asesor Ing. Ronald Ibarra Zapata por el asesoramiento en el trabajo de investigación y mostrar una guía de profesionalismo frente a los proyectos referente a la carrera profesional.

Al doctor Walter Bernuy Blanco por el apoyo en asesoramiento del trabajo de investigación, al docente universitario Wilmer Bermudez Pino por el apoyo en asesoramiento de la metodología del trabajo de investigación y su amistad, al magister Noel Campo Juipa por su amistad y confianza impartida en el laboratorio de software libre, y a los demás docentes universitarios de la Facultad de Ingeniería en Informática y Sistemas (FIIS) por el conocimiento impartido.

A mis amigos Samuel Pardo y Cesar Inga por el apoyo en momentos buenos y malos a lo largo de la carrera universitaria en el cual la confianza y amistad mejoran con el tiempo.

A promoción FIIS 2007 por brindarme su amistad y compartir experiencias que quedarán en un buen recuerdo y a los demás amigos de la facultad en el cual compartí conocimiento y momentos.

ÍNDICE

I.	INTRODUCCIÓN.....	1
1.1.	Contexto del Problema.....	3
1.2.	Formulación del Problema	4
1.3.	Problemas.....	5
1.3.1.	Problema General.....	5
1.3.2.	Problemas Específicos	5
1.4.	Justificación	6
1.5.	Objetivos	7
1.5.1.	Objetivo General.....	7
1.5.2.	Objetivos Específicos	7
1.6.	Hipótesis y Variables.....	8
1.6.1.	Hipótesis General	8
1.6.2.	Hipótesis Específicas.....	8
1.6.3.	Operacionalización de Variables e Indicadores	9
II.	REVISIÓN DE LITERATURA.....	12
2.1.	Antecedentes	12
2.2.	Marco Teórico	17
2.2.1.	Algoritmo Genético.....	17
2.2.2.	Procedimiento del Algoritmo Genético	33
2.3.	Definiciones Operacionales	34
2.3.1.	Caso de Estudio	34
2.3.2.	Problema de Optimización	35
2.3.3.	Modelo probabilístico.....	35
2.3.4.	Modelo Determinista.....	35
2.3.2.	Generación del Algoritmo Genético.....	35

2.3.3.	Restricciones Duras	35
2.3.4.	Restricciones Blandas	36
2.3.5.	Espacio de Búsqueda.....	36
2.3.6.	Solución Óptima	36
2.3.7.	Locus.....	36
III.	MATERIALES Y MÉTODOS	38
3.1.	Tipo, Diseño y Nivel de Investigación	38
3.1.1.	Tipo de Investigación.....	38
3.1.2.	Diseño de Investigación.....	38
3.1.3.	Nivel de Investigación	39
3.2.	Unidad de Análisis	39
3.3.	Técnicas de Investigación	40
3.3.1.	Caso de Estudio	40
3.4.	Instrumentos de la Investigación	40
3.4.1.	Ficha de Observación:.....	40
3.5.	Procedimiento de la Investigación	41
3.5.1.	Implementación del Algoritmo Genético	41
3.5.2.	Construcción de la Aplicación Web	42
3.5.3.	Obtención de Datos del Proceso en una Situación Forma Tradicional	42
3.5.4.	Obtención de Datos del Proceso en una Situación Utilizando el Algoritmo Genético	42
3.5.5.	Procesamiento de Datos	43
3.5.6.	Análisis de Resultados	43
3.6.	Proceso de Elaboración de Horarios de Clases	43
3.6.1.	Información de Entrada	46
3.6.2.	Mecanismo del Proceso	48

3.6.3. Horario de Clases Terminado	49
3.7. Proceso en una Situación de Forma Tradicional	49
3.7.1. Tecnología Utilizada	50
3.8. Proceso en una Situación Utilizando el Algoritmo Genético	58
3.8.1. Tecnología Utilizada	63
IV. RESULTADOS	95
4.1. Eficiencia del Proceso de Elaboración de Horarios de Clases	95
4.2. Tiempo de Elaboración del Proceso de Elaboración de Horarios de Clases	97
4.2.1. Forma Tradicional.....	98
4.2.2. Utilizando el Algoritmo Genético.....	100
4.3. Esfuerzo del Proceso de Elaboración de Horarios de Clases	100
4.3.1. Forma Tradicional.....	102
4.3.2. Utilizando el Algoritmo Genético.....	104
4.4. Incumplimiento de Restricciones Duras del Horario de Clases	104
4.4.1. Forma Tradicional.....	107
4.4.2. Utilizando el Algoritmo Genético.....	107
V. DISCUSIÓN.....	109
VI. CONCLUSIONES.....	112
RECOMENDACIONES	116
BIBLIOGRAFÍA	117
ANEXO	120
GLOSARIO	137

ÍNDICE DE TABLAS

Tabla 1. Operacionalización de variables del Proceso de Elaboración de Horarios de Clases (PEHC).....	9
Tabla 2. Explicación de variables.....	10
Tabla 3: Diseño No Experimental Transversal o Transeccional.....	39
Tabla 4. Registro de información de entrada - carga académica para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.	60
Tabla 5. Registro de información de entrada - aula universitaria para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.	61
Tabla 6. Registro de información de entrada - grupo de estudiantes para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.....	61
Tabla 7. Registro de información de entrada de limitaciones para horario de clases para el proceso en una situación utilizando el algoritmo genético.	62
Tabla 8. Registro de días de la semana para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.	62
Tabla 9. Registro de información de entrada de timeslot para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.	63
Tabla 10. Términos del algoritmo genético para el proceso de elaboración de horarios de clases.	64

Tabla 11. Coste de penalización de violación de restricciones duras de los criterios del horario de clases.....	67
Tabla 12. Eficiencia del proceso de elaboración de horarios de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.	95
Tabla 13. Tiempo de elaboración de horarios de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.	97
Tabla 14. Tiempo de elaboración del proceso de elaboración de horarios de clases de la UNAS en una situación de forma tradicional.	98
Tabla 15. Esfuerzo del proceso de elaboración de horarios de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.	101
Tabla 16. Esfuerzo del proceso de elaboración de horarios de clases de la UNAS en una situación de forma tradicional.....	102
Tabla 17. Incumplimiento de restricciones duras de horario de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.	105
Tabla 18. Porcentaje de conflicto de timeslot de los criterios de horario de clases de la UNAS en una situación de forma tradicional.	107
Tabla 19. Porcentaje de conflicto de timeslot de los criterios de horario de clases de la UNAS en una situación utilizando el algoritmo genético.	108
Tabla 20. Resumen de la ficha de recolección de datos del proceso de elaboración de horarios de clases (EHC) y otras actividades.	121

Tabla 21. Número de cursos, timeslot y cursos sin timeslot por escuelas profesionales en el horario de clases de forma tradicional.	125
Tabla 22. Número de cursos, timeslot y cursos sin timeslot por escuelas profesionales en el horario de clases utilizando el algoritmo genético.	126

ÍNDICE DE FIGURAS

Figura 1. Ecuación de <i>fitness</i> puro. Fuente: Ahumada (2014).	19
Figura 2. Ecuación de <i>fitness</i> estandarizado. Fuente: Ahumada (2014).....	20
Figura 3. Ecuación de <i>fitness</i> ajustado. Fuente: Ahumada (2014).....	21
Figura 4. Ecuación de <i>fitness</i> normalizado. Fuente: Ahumada (2014).....	21
Figura 5. Ecuación de probabilidad de ser seleccionado por el método de selección por ruleta. Fuente: Arroyo (2013).	23
Figura 6. Pseudocódigo del método de selección por ruleta. Fuente: Arroyo (2013).....	24
Figura 7. Ejemplo de procedimiento de selección por ruleta. Fuente: Arroyo (2013).....	25
Figura 8. Ejemplo de procedimiento de selección por torneo. Fuente: Arroyo (2013).....	26
Figura 9. Pseudocódigo del método de selección por torneo. Fuente: Arroyo (2013).....	27
Figura 10. Probabilidad de selección del individuo basado en el <i>ranking</i> . Fuente: Arroyo (2013).	27
Figura 11. Ordenamiento de individuos basado en su <i>fitness</i> . Fuente: Arroyo (2013).....	28
Figura 12. Ordenamiento de individuos basados en su <i>ranking</i> . Fuente: Arroyo (2013).....	29
Figura 13. Pseudocódigo del método selección basado en <i>ranking</i> . Fuente: Arroyo (2013).	29
Figura 14. Cruzamiento de un punto. Fuente: Eiben y Smith (2015).	30

Figura 15. Cruzamiento de m puntos con $m = 2$. Fuente: Eiben y Smith (2015).	31
Figura 16. Cruzamiento uniforme. Fuente: Eiben y Smith (2015).	32
Figura 17. Pseudocódigo del procedimiento del algoritmo genético convencional. Fuente: Jat (2012).	34
Figura 18. Diseño y formato de la ficha de observación para la recolección de datos del proceso de elaboración de horarios de clases.	41
Figura 19. Esquema general del proceso de elaboración de horarios de clases.	44
Figura 20. Proceso genérico de elaboración de horarios de clases en la UNAS.	45
Figura 21. Proceso de elaboración de horarios de clases en una situación de forma tradicional.....	50
Figura 22. Diagrama UML de asignación Manual de Horario de Clases del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	52
Figura 23. Vista de selección de filtros en el sistema de gestión de horarios del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	53
Figura 24. Asignación manual de un evento en el sistema de gestión de horarios del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	54

Figura 25. Diagrama de Despliegue del sistema de Gestión de Horarios del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	55
Figura 26. Sistema de Gestión de Horario versión 3.7 del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	56
Figura 27. Carga académica en el sistema de gestión de horarios versión 3.7 del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	56
Figura 28. Registro de aulas universitarias en el sistema de gestión de horarios versión 3.7 del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	57
Figura 29. Asignación manual de horario de clases en el sistema de Gestión de Horarios del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	57
Figura 30. Proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.	59
Figura 31. Representación no binaria de un gen del cromosoma de la tecnología utilizando el algoritmo genético.....	65
Figura 32. Representación del cromosoma en la tecnología utilizando el algoritmo genético.....	66
Figura 33. Representación de <i>fitness</i> detallado del cromosoma.....	68
Figura 34. Diagrama UML del Algoritmo genético implementado en SEAHC..	72
Figura 35. Diagrama UML de método selección por ruleta.	75
Figura 36. Pseudocódigo de selección de individuos con el método ruleta.	78

Figura 37. Proceso de cruzamiento y mutación en el algoritmo genético de SEAHC.....	79
Figura 38. Pseudocódigo de los operadores genéticos de cruzamiento y mutación.	81
Figura 39. Representación de una solución óptima	82
Figura 40. Pseudocódigo del método hacinamiento determinista (o <i>Deterministic Crowding</i>). Fuente: Lozano et al. (2008).....	84
Figura 41. Diagrama UML del método hacinamiento determinista aplicado en la presente investigación.	85
Figura 42. Diagrama de despliegue de SEAHC.....	87
Figura 43. Modelo en capas de SEAHC.	88
Figura 44. Vista de carga de información de entrada de SEAHC para elaborar los horarios de clases.....	91
Figura 45. Vista de configuración de SEAHC para elaborar los horarios de clases utilizando el algoritmo genético.....	92
Figura 46. Vista de característica de solución óptima u horario de clases terminado.	92
Figura 47. Vista de horario de clases en el SEAHC para los diferentes criterios.	93
Figura 48. Horario de clases en archivo XML terminado con la tecnología utilizando el algoritmo genético. Herramienta utilizada: Notepad++.....	94
Figura 49. Eficiencia del proceso de elaboración de horarios de clases en la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.....	96

Figura 50. Tiempo de elaboración del proceso de elaboración de horarios de clases en la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.....	98
Figura 51. Esfuerzo del proceso de elaboración de horarios de clases en la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.	102
Figura 52. Incumplimiento de restricciones duras del horario de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.....	106
Figura 53. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	127
Figura 54. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	128
Figura 55. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	129
Figura 56. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	130
Figura 57. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	131
Figura 58. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	132
Figura 59. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	133

Figura 60. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	134
Figura 61. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.....	135
Figura 62. Ficha de observación del proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.....	136

RESUMEN

La investigación se enfoca en resolver el problema de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, para determinar si es posible o no, mejorar la eficiencia del proceso de elaboración de horarios de clases respecto a minimizar el tiempo de elaboración y esfuerzo como también evitar el incumplimiento de restricciones duras. La investigación es de tipo aplicada, diseño no experimental transversal y nivel descriptivo para el proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético, es de enfoque cuantitativo y la técnica de investigación es un caso de estudio que mediante la ficha de observación se recolectó los datos de la eficiencia del proceso en las dos situaciones para realizar la comparación entre sí. Como resultado la diferencia de la eficiencia del proceso en las dos situaciones respecto al tiempo de elaboración y esfuerzo es 96.97% y 96.46% en favor de la tecnología utilizando el algoritmo genético frente a la forma tradicional y el incumplimiento de restricciones duras es 0% para las dos tecnologías; por lo tanto, se construyó una aplicación web que permitió implementar el algoritmo genético como metaheurística y se determinó que si es posible mejorar la eficiencia del proceso y se recomienda utilizar esta tecnología a las universidades con recursos compartidos que elaboran los horarios de clases de forma tradicional como también realizar investigaciones para complementar el incumplimiento de restricciones blandas.

Palabras claves: Algoritmo genético. Problema de elaboración de horarios de clases, Universidades con recursos compartidos. Metaheurística.

ABSTRACT

The research focuses on resolving the problem of the creation of class schedules at universities with shared resources using the genetic algorithm to determine if it is possible or not to improve the efficiency of the process of creating class schedules, with respect to minimizing the work time and effort, as well as avoiding the lack of fulfillment of hard restrictions. The research is of the applied type, with a non-experimental, cross-sectional design and a descriptive level for the process in a traditional situation and a situation using the genetic algorithm, it is of a quantitative focus and the research technique is a case study, which, through observation, data was collected on the efficiency of the process in the two situations in order to do a comparison between the two. For the results, the difference of the efficiency of the process in the two situations with respect to the elaboration time and effort is 96.97% and 96.46%, in favor of the technology using the genetic algorithm instead of the traditional way and the lack of compliance with hard restrictions is 0% for both technologies; thus a web application was constructed, which allows for a genetic algorithm to be implemented as a metaheuristic and it was determined that it is possible to improve the efficiency of the process and it is recommended that this technology be used in the universities with shared resources which have traditional class schedules, as well as doing research to compliment the lack of compliance with soft restrictions.

Keywords: Genetic algorithm, Problem of class schedule creation.

University with shared resources. Metaheuristic.

I. INTRODUCCIÓN

El problema de elaboración de horarios de clases en una situación de forma tradicional genera gastos en la economía del proceso respecto al tiempo de elaboración y esfuerzo, más aún cuando los recursos como aulas y docentes universitarios se comparten entre carreras profesionales de la universidad y con la finalidad de evitar el incumplimiento de restricciones duras en el horario de clases terminado el proceso es más dificultoso. En el caso de estudio Universidad Nacional Agraria de la Selva no cuenta con metaheurística aplicados a sistemas informáticos para resolver el problema de optimización en el proceso de elaboración de horarios de clases, así mismo no se cuenta con evidencia empírica en investigaciones utilizando el algoritmo genético para solucionar el problema de elaboración de horarios de clases en universidades con recursos compartidos.

El contenido de la presente tesis consta de seis capítulos: En el capítulo I se encuentra la introducción a la investigación, donde se describe el problema a solucionar, la justificación, los objetivos, la hipótesis y las variables de la investigación; en el capítulo II se encuentra la revisión de literatura la tesis, antecedentes similares a la investigación, conceptos teóricos y definiciones operacionales para la investigación, en el capítulo III se encuentra los materiales y métodos donde se describe el tipo, diseño, nivel, técnicas, instrumentos y

procedimiento de la investigación y la descripción del proceso de elaboración de horarios de clases del caso de estudio en una situación de forma tradicional y una situación utilizando el algoritmo genético, ambas situaciones se describieron de manera general a manera específica, en el capítulo IV se muestra los resultados de la investigación de la eficiencia del proceso de elaboración de horarios de clases y de las dimensiones: Tiempo de elaboración y esfuerzo del proceso como también el incumplimiento de restricciones en una situación de forma tradicional y una situación utilizando el algoritmo genético; estos resultados son descritos de manera general a manera específica, en el capítulo V se tiene la discusión, donde se ha realizado la comparación del proceso de elaboración de horarios de clases del caso de estudio en las dos situaciones: Forma tradicional y utilizando el algoritmo genético; también se ha realizado una comparación con dos antecedentes donde se ha utilizado el algoritmo genético para resolver problemas de elaboración de horarios de clases (o *course university timetabling problem*), en el capítulo VI se ha realizado la conclusión de la investigación donde se menciona la adecuación del algoritmo genético, la eficiencia más las tres dimensiones y una reflexión sobre la solución algorítmica; también se incorpora las recomendaciones, para implementar el algoritmo genético, implementar asignación de *timeslot* múltiple y *fitness* blando y explorar en otros campos para problemas de optimización del mismo, las referencias bibliográficas similares y/o importantes para la investigación y por último los anexos correspondientes de la investigación donde se encuentra el cuadro de resumen de resultados y las fichas de observación de datos del mismo obtenidos en las dos situaciones: forma tradicional y utilizando el algoritmo genético.

1.1. Contexto del Problema

Las universidades realizan cada semestre académico los horarios de clase y se encuentran con el problema de elaboración de horarios de clases (o UCTP, por sus siglas en inglés de *University Course Timetabling Problem*). La UCTP es la distribución eficiente de los grupos de estudiantes, cursos y docentes universitarios en las aulas universitarias y *timeslot*, evitando la violación de restricciones duras para definir la viabilidad del horario de clases y minimizando la violación de restricciones blandas para definir la calidad del horario de clases de la universidad (Soliman y Keshk, 2015).

La investigación considera principalmente a las universidades que comparten sus recursos (aulas y docentes universitarios) entre varias carreras profesionales y sujeto a evitar el incumplimiento de restricciones duras del horario de clases terminado, donde existe un problema de optimización en el proceso de elaboración de horarios de clases en una situación de forma tradicional, teniendo como consecuencia gastos en la economía del proceso respecto al tiempo de elaboración y esfuerzo.

A fin de determinar si es posible o no, mejorar la eficiencia del proceso de elaboración de horarios de clases como minimizar el tiempo de elaboración y esfuerzo del proceso y también evitar el incumplimiento de restricciones duras del horario de clases en universidades con recursos compartidos como aulas y docentes universitarios, utilizando el algoritmo genético, y realizar una comparación del proceso en una situación de forma

tradicional y una situación utilizando el algoritmo genético, se realizará la presente investigación de tesis titulado “ALGORITMO GENÉTICO PARA LA ELABORACIÓN DE HORARIOS DE CLASES EN UNIVERSIDADES CON RECURSOS COMPARTIDOS, CASO DE ESTUDIO: UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA”.

1.2. Formulación del Problema

Se desconoce si el algoritmo genético mejorará la eficiencia del proceso de elaboración de horarios de clases en universidades con recursos compartidos respecto a minimizar el tiempo de elaboración y esfuerzo del proceso como también evitar el incumplimiento de restricciones duras del horario de clases terminado; la mejora de la eficiencia del proceso se determinará mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético.

¿Será posible mejorar la eficiencia del proceso de elaboración de horarios de clases (minimizar el tiempo de elaboración y esfuerzo de proceso como también evitar el incumplimiento de restricciones duras del horario de clases terminado) en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético?

1.3. Problemas

1.3.1. Problema General

¿Será posible mejorar la eficiencia del proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético?

1.3.2. Problemas Específicos

¿Será posible minimizar el tiempo del proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético?

¿Será posible minimizar el esfuerzo del proceso de elaboración de horarios de clase en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético?

¿Será posible evitar el incumplimiento de restricciones duras del horario de clase en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del horario de clases terminado en una situación de forma tradicional y una situación utilizando el algoritmo genético?

1.4. Justificación

A pesar que se cuenta con antecedentes en los que se ha utilizado el algoritmo genético para resolver el problema de elaboración de horarios de clases en universidades, y en general se han reportado casos de estudio en carreras profesionales o universidades que mayormente utilizan recursos exclusivos (universidades que no comparten sus recursos como aulas y docentes universitarios), por lo que es necesario aportar evidencia empírica en un contexto más complejo en el que las universidades comparten sus recursos entre varias carreras profesionales, de modo que determinará si es posible o no, mejorar la eficiencia del proceso de elaboración de horarios de clases respecto a minimizar el tiempo de elaboración y esfuerzo del proceso como también evitar el incumplimiento de restricciones duras del horario de clases terminado.

La presente investigación permitirá a universidades con recursos compartidos contar con una alternativa de solución que, en caso de así demostrarse, mejorará la eficiencia del proceso de elaboración de horarios de clases, es decir, minimizará el tiempo de elaboración y esfuerzo del proceso como también evitará el incumplimiento de restricciones duras del horario de clases.

Como valor agregado la presente investigación generará conocimiento empírico de un modelo propuesto en el algoritmo genético para la solución del problema de elaboración de horarios de clases en universidades con recursos compartidos.

1.5. Objetivos

1.5.1. Objetivo General

Determinar si es posible o no, mejorar la eficiencia del proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético.

1.5.2. Objetivos Específicos

Determinar si es posible o no, minimizar el tiempo de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético.

Determinar si es posible o no, minimizar el esfuerzo en el proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético.

Determinar si es posible o no, evitar el incumplimiento de restricciones duras del horario de clases terminado en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del horario de clases terminado en una situación de forma tradicional y una situación utilizando el algoritmo genético.

1.6. Hipótesis y Variables

1.6.1. Hipótesis General

H₀: Es posible mejorar la eficiencia del proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético.

1.6.2. Hipótesis Específicas

H₁: Es posible minimizar el tiempo de elaboración del proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético.

H₂: Es posible minimizar el esfuerzo del proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del proceso en una situación de forma tradicional y una situación utilizando el algoritmo genético.

H₃: Es posible evitar el incumplimiento de restricciones duras del horario de clases en universidades con recursos compartidos utilizando el algoritmo genético, mediante la comparación del horario de clases terminado en una situación de forma tradicional y una situación utilizando el algoritmo genético.

1.6.3. Operacionalización de Variables e Indicadores

Tabla 1. Operacionalización de variables del Proceso de Elaboración de Horarios de Clases (PEHC).

	Independiente	Dependiente		
Variable	X: Mecanismo del PEHC.	Y: eficiencia del PEHC		
Dimensión	Tecnología.	Tiempo de elaboración.	Esfuerzo.	Incumplimiento de restricciones duras.
Indicador	Tipo de tecnología: Forma tradicional y utilizando el algoritmo genético.	Número de días.	Horas-persona.	Porcentaje de conflictos de <i>timeslot</i> de los criterios del horario de clases: Grupo de estudiantes, docente, aula y curso universitario.

Tabla 2. Explicación de variables.

Variable	Dimensión	Indicador	Definición
X: Mecanismo del proceso de elaboración de horarios de clases.	Tecnología.	Tipo de tecnología: Forma tradicional utilizando algoritmo genético.	Se refiere a la tecnología que se utiliza en el proceso de elaboración de horarios de clases: en una situación de forma tradicional y una situación utilizando el algoritmo genético.
Y: Eficiencia del proceso de elaboración de horarios de clases.	Tiempo de elaboración.	Número de días.	Hace referencia a la cantidad de días, desde el día de inicio hasta el día final del proceso de elaboración de horarios de clases.
	Esfuerzo.	Horas-persona.	Hace referencia las horas de trabajo del personal asignado dedicados al proceso de elaboración de horarios de clases.

Incumplimiento de restricciones duras

Porcentaje de conflictos de *timeslot* de los criterios horario de clases: Grupo de estudiantes, docente, aula y curso universitario.

Hace referencia a la cantidad porcentual de conflictos de *timeslot* por cada criterio del horario de clases; cuyo resultado es el producto de la cantidad total del criterio de horario de clases, la cantidad total de *timeslot* del horario de clases y el 100%, dividido entre la cantidad de conflictos de *timeslot* del criterio de horario de clases; este método se aplica para cada criterio de horario de clases: Grupo de estudiantes, docente, aula y curso universitario.

II. REVISIÓN DE LITERATURA

2.1. Antecedentes

Babaei, Karimpour y Hadidi (2015) en su artículo científico **UN ESTUDIO DE ENFOQUES PARA EL PROBLEMA DE ELABORACIÓN DE HORARIOS DE CURSOS UNIVERSITARIOS** realiza un estudio de diferentes enfoques para el problema de elaboración de horarios de cursos universitarios y una presentación de datos confiables para probar y evaluar algunos algoritmos para resolver problemas de elaboración de horarios de cursos universitarios. Indicando el número de restricciones duras y blandas se diferencian de una institución a otra, en el cual el objetivo de todos los algoritmos a estudiar es maximizar el número de restricciones blandas satisfechas en los horarios finales. Donde uno de sus experimentos con pequeños conjuntos de datos resulta que los algoritmos genéticos tendrían un mejor rendimiento en términos de eficiencia frente a otras heurísticas, pero no es el mismo resultado con medianos y/o grandes conjuntos de datos.

Khonggamnerd e Innet (2009) en su artículo científico **SOBRE LA MEJORA DE EFECTIVIDAD EN LA ELABORACIÓN DE HORARIOS UNIVERSITARIOS AUTOMÁTICO CON EL ALGORITMO GENÉTICO APLICADO** ha implementado el algoritmo genético para elaborar los horarios universitarios con la finalidad de ahorrar horas-persona en el proceso y se ha

utilizado los datos del plan curricular de un año académico de la escuela de Ingeniería de la universidad de la Cámara de Comercio de Tailandia. La información de entrada para el algoritmo genético es: cuatro restricciones duras y blandas, una carrera profesional, un grupo de estudiantes universitarios para el semestre I y II, cuatro aulas de teoría y dos de práctica, 25 docentes, cinco días de clases y seis *timeslot*. En la investigación se ha realizado experimentos donde se hizo pruebas con los parámetros de probabilidad de cruzamiento y mutación del algoritmo genético para encontrar la solución óptima, y para el caso de éxito los parámetros del algoritmo genético es: dos de tamaño de población inicial, 0.7 probabilidad de cruzamiento, 0.3 probabilidad de mutación y se seleccionan los dos mejores cromosomas en base al rango de fitness entre los dos padres y dos hijos; la solución óptima fue encontrado en la generación 445 con *fitness* 0 en restricciones duras y 34 en restricciones blandas.

Junn, Obit y Alfred (2018) en su artículo científico **EL ESTUDIO DEL ENFOQUE DE ALGORITMO GENÉTICO PARA RESOLVER EL PROBLEMA DE ELABORACIÓN DE HORARIOS DE CURSOS UNIVERSITARIOS** destaca el rol del algoritmo genético, y utiliza conjuntos de datos de dos semestres académico de la universidad de Malasia Sabah Labuan Campus Internacional. El objetivo de la investigación es satisfacer las restricciones duras y minimizar las violaciones de restricciones blandas, y obtener una solución viable y de alta calidad para la institución, como también su propósito es estudiar y comparar su desempeño del Algoritmo Genético frente a otros algoritmos de Gran Diluvio y Simulado Recocido de otras investigaciones. La investigación realizó dos

experimentos, el primero investiga el desempeño del Algoritmo Genético entre los dos semestres y un análisis basado en la calidad de solución, donde la mejora de la calidad de solución promedio es de 35% de la población inicial. El segundo experimento se compara el desempeño del Algoritmo Genético, Gran diluvio y Recocido Simulado, donde demostró que el desempeño de la calidad de solución de Algoritmo Genético de 35% es mucho mejor que Gran Diluvio y Recocido Simulado de 25%, pero el costo de funcionamiento es menor que el Algoritmo Genético. Concluyendo que el Algoritmo Genético logró superar a los otros algoritmos, en obtener calidad de soluciones en el dominio de proceso de elaboración de horarios.

Johnston (2000) en su tesis **APLICACIÓN DE ALGORITMOS GENÉTICOS PARA LA ASIGNACIÓN DE CARGA ACADÉMICA EN INSTITUCIONES DE EDUCACIÓN SUPERIOR** busca resolver el problema de asignación de carga académica de forma manual o tradicional para reducir el tiempo de elaboración del proceso en el caso de estudio al Instituto Tecnológico de Nuevo Laredo de México en el cual no obtiene resultados óptimos, la importancia de la investigación es encontrar distribución óptima del horario de clases de acuerdo al perfil académico de los docentes, el horario de carga académica consta de dos restricciones duras (penalización de tres por conflicto de *timeslot* de grupo de estudiantes y dos por docente) y una restricción blanda (penalización de uno por *timeslot* libre entre clases de un grupo de estudiantes). En la investigación implementó el algoritmo genético en un Sistema de Asignación de Carga Académica en Instituciones de Educación Superior, por

límite de recursos tecnológicos la información de entrada es: primeros cuatro ciclos académicos de un departamento académico en una carrera profesional, 25 cursos, 30 docentes y excluye la asignación de aulas; los parámetros para el algoritmo genético es: tamaño de población inicial 150, método de selección por ruleta, probabilidad de cruzamiento uniforme 0.9, probabilidad de mutación 0.001 y elitismo para reemplazar individuos de la población inicial. El algoritmo genético encuentra la mejor solución con *fitness* 31 (10 conflictos de *timeslot* de docente y 11 de *timeslot* libre) en la generación 272 con *timeslot* fijo de una hora frente a la forma tradicional con *fitness* 105 (cuatro conflictos de *timeslot* de grupo de estudiantes y 93 de *timeslot* libre) que fueron evaluados en las mismas condiciones.

Yang y Jat (2011) en su artículo científico **ALGORITMOS GENÉTICOS CON ESTRATEGIAS DE BÚSQUEDA LOCAL Y GUIADA PARA ELABORACIÓN DE HORARIOS DE CLASE CURSOS UNIVERSITARIOS** investiga algoritmos genéticos con una estrategia de búsqueda guiada que sirve para crear descendencia basado en estructura de datos con información de buenos individuos de generaciones anteriores; y la técnica de búsqueda local, utiliza su capacidad de búsqueda exploratoria para mejorar la eficiencia de búsqueda de los algoritmos genéticos propuesto y la calidad de los individuos; la investigación presenta un Algoritmo Genético de Búsqueda Guiada (GSGA, por sus siglas en inglés de *Guided Search Genetic Algorithm*), donde la estrategias de búsqueda y la técnica de búsqueda local son integradas al Algoritmo Genético de Estado Estacionario (SSGA, por sus siglas en inglés de *Steady State Genetic*

Algorithm); denominado GSGA Extendida (EGSGA, por sus siglas en inglés de “*Extended GSGA*”). Realizó un estudio comparativo de EGSGA con otros algoritmos implementados por los investigadores: Búsqueda Tabú (TS, por sus siglas en inglés de *Tabú Search*), SSGA, algoritmo genético con búsqueda local (GALS, por sus siglas en inglés de *Genetic Algorithm with Local Search*) y GSGA; donde EGSGA produce un bajo promedio y desviación estándar del valor objetivo, y los peores objetivos producidos por EGSGA son mejores entre los valores objetivos de los otros algoritmos, debido a la estructura de datos y esquema de búsqueda local, resultando que EGSGA es más confiable que los otros algoritmos descritos.

Badoni, Gupta y Mishra (2014) en su artículo científico **UN NUEVO ALGORITMO HÍBRIDO PARA PROBLEMAS DE ELABORACIÓN DE HORARIOS DE CURSOS UNIVERSITARIOS USANDO EVENTOS BASADOS EN AGRUPAMIENTO DE ESTUDIANTES** combina un algoritmo genético con búsqueda local (GALS, por sus siglas en inglés de *Genetic Algorithm with Local Search*) y el uso de eventos basado en agrupamiento de estudiantes, obteniendo un Nuevo Algoritmo Híbrido (NHA, por sus siglas en inglés de *New Hybrid Algorithm*) para solucionar el problema de elaboración de horarios de cursos universitarios. realizó experimentos en base al tiempo para encontrar mejores soluciones, utilizando los mismos parámetros de tamaño de población, probabilidad de cruzamiento y mutación, compara el NHA con el GALS y otros algoritmos considerados en la investigación para resolver problemas de elaboración de horarios de cursos universitarios; resultando que el NHA

proporciona soluciones óptimas, superando a GALS que no los proporciona, en periodos de tiempo muy corto para conjunto de datos de problemas pequeños; y para algunas instancias de problemas mediano NHA supera a los otros algoritmos considerados en la investigación.

2.2. Marco Teórico

2.2.1. Algoritmo Genético

El algoritmo genético es un método de búsqueda estocástica basada en la teoría de la evolución de Darwin¹ popularizado por Holland² que propuso de forma explícita el cruzamiento y otros operadores de recombinación que se utilizaron para resolver problemas de optimización y búsqueda (Jat, 2012). El algoritmo genético es un algoritmo de búsqueda basado en la selección y genética natural, combinan la supervivencia del más apto con intercambio de información estructurado de forma aleatoria, y en cada generación se crea un conjunto de individuos artificiales utilizando elementos de los individuos antiguos y más aptos (Goldberg, 1989). En el algoritmo genético existen dos elementos indispensables: La representación de la solución del problema o individuo el cual se denomina cromosoma y la función *fitness* para medir la aptitud del individuo; como también tres operaciones fundamentales aplicado a los cromosomas, selección de individuos para candidatos a reproducirse, cruzamiento entre dos

¹ Charles Darwin en su libro "El origen de las especies por medio de la selección natural" (1859).

² John Holland en su libro "Adaptación en sistemas artificiales y naturales" (1975).

cromosomas y mutación para modificar un cromosoma en alguno de sus puntos y explorar espacios de soluciones (Ahumada, 2014).

2.2.1.1. Representación y Codificación del Problema

El cromosoma es el vínculo entre el problema original (mundo real) y el espacio de solución del problema donde la evolución tiene lugar (mundo algoritmo evolutivo), mediante la abstracción de algunos aspectos del mundo real se crea un contexto de la solución del problema bien definido y tangible que pueda ser evaluado, como también se debe almacenar el cromosoma de manera que pueda ser manipulada por el computador; en el contexto del problema original una solución se denomina fenotipo mientras en el contexto de algoritmos evolutivos la codificación de la solución o individuo se denomina genotipo (Eiben y Smith, 2015). El cromosoma es la representación de la solución al problema que se conforma de estructura de datos que contiene la información relevante del problema y debe ocupar el menor espacio posible para ser evaluado de forma simple por la función *fitness* del algoritmo genético, y su codificación puede estar dada en dos tipos: codificación binaria, conformada por una cadena de bits y codificación no binaria, conformada por una cadena de caracteres (Ahumada, 2014).

2.2.1.2. *Fitness* o Valor de Aptitud

El *fitness* es la medida numérica para determinar la aptitud de un individuo en el problema, por lo tanto, la función *fitness* debe estar diseñado para calcular y sumar todas las penalizaciones de violación de restricciones duras y

blandas del cromosoma para obtener el *fitness* del individuo, siendo así los individuos más aptos aquellos que tienen el menor número de incumplimiento de restricciones (Ahumada, 2014).

Ahumada (2014) menciona cuatro tipos de función *fitness*:

- ***Fitness Puro.***

Es la medida de ajuste establecida en la terminología natural del problema, el cálculo de *fitness* de un individuo *i* en un instante *t* como se muestra en la ecuación de la Figura 1.

$$r(i, t) = \sum_{j=1}^{Nc} |s(i, j) - c(i, j)|$$

Figura 1. Ecuación de *fitness* puro. Fuente: Ahumada (2014).

Siendo:

$s(i, j)$ = Valor deseado para el individuo *i* en el caso *j*.

$c(i, j)$ = Valor obtenido por el individuo *i* para el caso *j*.

Nc = Número de casos.

En los problemas de maximización los individuos con un *fitness* puro con mayor valor serán los más aptos mientras en los problemas de minimización los individuos con *fitness* puro con menor valor serán los más aptos.

- ***Fitness* Estandarizado.**

Para solucionar la dualidad del *fitness* estandarizado en problemas de minimización o maximización se modifica el *fitness* puro de acuerdo a la ecuación de la Figura 2.

$$s(i, t) = \begin{cases} r(i, t), & \text{Minimización} \\ r_{max} - r(i, t), & \text{Maximización} \end{cases}$$

Figura 2. Ecuación de *fitness* estandarizado. Fuente: Ahumada (2014).

En la Figura 2 se observa que en casos de problemas de minimización se emplea la medida de *fitness* puro y si el problema es de maximización se resta una cuota o un valor superior r_{max} del error del *fitness* puro. Aplicando esta medida, la aptitud de un individuo es mayor siempre en cuando el valor del ajuste se aproxime a cero, por lo tanto, dentro de la generación t un individuo i es mejor que uno individuo j si se cumple $s(i, t) < s(j, t)$.

- **Fitness Ajustado.**

Se obtiene aplicando la transformación reflejada en la ecuación del *fitness* estandarizado de la Figura 2 como se muestra en la Figura 3.

$$a(i, t) = \frac{1}{1 + s(i, t)}$$

Figura 3. Ecuación de *fitness* ajustado. Fuente: Ahumada (2014).

En la Figura 3 se observa que de esta forma el *fitness* ajustado tomará siempre los valores del intervalo [0,1] cuando más se aproxime el *fitness* de un individuo a 1 mayor será su aptitud.

- **Fitness Normalizado.**

Los diferentes tipos de *fitness* vistos en los puntos anteriores hacen referencia únicamente a la aptitud del individuo en cuestión, el *fitness* normalizado introduce un nuevo aspecto donde indica la aptitud de una solución con respecto al resto de soluciones representadas en la población, considerando una población de tamaño N se obtiene la siguiente ecuación de la Figura 4.

$$n(i, t) = \frac{a(i, t)}{\sum_{k=1}^N a(i, t)}$$

Figura 4. Ecuación de *fitness* normalizado. Fuente: Ahumada (2014).

En la Figura 4 se observa, así como el *fitness* ajustado toma valores del intervalo $[0,1]$ y los mejores individuos son aquellos cuyo *fitness* se aproxime a 1. Pero a diferencia del otro tipo de *fitness* un valor cercano a 1 no sólo indica que el individuo representa una buena solución al problema, sino que además es una solución destacadamente mejor que las proporcionadas por el resto de la población. La suma de los valores de *fitness* normalizado de todos los individuos de una población siempre resulta la unidad o el valor de 1. Este tipo de ajuste es empleado en la mayoría de los métodos de selección proporcionadas al *fitness*.

2.2.1.3. Operador de Selección

Es la función encargada de seleccionar individuos candidatos a reproducirse concediendo mayor oportunidad de reproducción a los individuos más aptos según su *fitness*, tomando en consideración que los individuos con *fitness* deficiente no se deben descartar de la población porque restaría diversidad de la población (Pose, 2000).

Arroyo (2013) menciona algunos métodos de selección como selección por ruleta, selección por torneo, selección basada en ranking y selección sexual, pero se describe alguno de ellos:

- Selección por Ruleta

A cada uno de los individuos de la población se les asigna una parte proporcional de la ruleta que sumando los porcentajes obtiene el valor de uno o la unidad, los mejores individuos reciben una mayor porción de la ruleta frente a los peores; la probabilidad P_i de ser seleccionado un individuo i está dado de acuerdo a su fitness f_i entre el fitness de la población f_j de N individuos como se muestra en la ecuación de la Figura 5.

$$P_i = \frac{f_i}{\sum_{j=1}^N f_j}$$

Figura 5. Ecuación de probabilidad de ser seleccionado por el método de selección por ruleta. Fuente: Arroyo (2013).

Para seleccionar un individuo se inicia un conteo numérico n con un valor de cero (0) y mientras el tamaño de la población sea mayor a n hacer: Se genera un número aleatorio r en el intervalo $[0,1]$, calcular el fitness acumulado, fitness total P_t y la suma del fitness proporcional (SUM), por lo tanto, surge la condición si **SUM** es menor que el número aleatorio r , se selecciona el individuo en el cual devuelve un individuo situado en la posición de la ruleta recorriendo los individuos de la población y acumulando sus proporciones de ruleta hasta que la suma exceda el valor obtenido del valor aleatorio; el pseudocódigo del procedimiento del método de selección por ruleta se muestra en la Figura 6.

1. **Generar** un valor de n con un valor de 0
2. **Mientras** tamaño de la población $> n$ **hacer**
3. **Generar** el número aleatorio r
4. **Calcular** el *fitness* acumulado, *fitness* total (P_t), y la suma del *fitness* proporcional (SUMA)
5. **Si** SUMA $< r$ entonces
6. **Seleccionar** el primer individuo, de otra manera, seleccionar i -ésimo individuo
7. **Fin de si**
8. **Incrementar** el valor de n en uno
9. **Fin de mientras**
10. **Devolver** los individuos seleccionados con *fitness* proporcionales al tamaño de la sección de la ruleta

Figura 6. Pseudocódigo del método de selección por ruleta. Fuente: Arroyo (2013).

En la Figura 6 se observa el procedimiento del método de selección por ruleta y para entender el procedimiento se realiza el siguiente ejemplo, se tiene una población de cuatro individuos con su *fitness*, se obtiene el *fitness* de la población P_t , luego el *fitness* del primer individuo es el *fitness* acumulado del mismo, a partir de ello para obtener el *fitness* acumulado del siguiente individuo se suma el *fitness* del individuo en mención y el *fitness* acumulado del individuo anterior, la suma del *fitness* proporcional es el resultado del *fitness* acumulado entre el *fitness* de la población, en el cual el último individuo obtiene el valor de la unidad y así se forma la ruleta, la probabilidad de selección está asociada con el *fitness* del individuo entre el *fitness* total de la población el cual es una porción de la ruleta, el número aleatorio es generado para seleccionar individuos

situados en la ruleta o la suma de fitness proporcional como se muestra en la Figura 7.

Individuo i	Fitness $f(i)$	Fitness acumulado	Suma del fitness proporcional	Probabilidad de selección P_i en la ruleta	Probabilidad de selección y suma de fitness proporcional de individuos para formar la ruleta	Número aleatorio	Individuo i seleccionado
i1	22	22	0.27	0.27		0.57	i2
i2	34	56	0.68	0.41		0.77	i3
i3	12	68	0.83	0.15		0.21	i1
i4	14	82	1	0.17		0.42	i2
	$P_T = 82$			1			

Figura 7. Ejemplo de procedimiento de selección por ruleta. Fuente: Arroyo (2013).

En la Figura 7 se observa una población de cuatro individuos con su *fitness* 22,34,12 y 14; el *fitness* de la población es 82, es decir la suma de *fitness* de los cuatro individuos, el *fitness* acumulado de i1 es 22, para el *fitness* acumulado del segundo individuo es la suma del *fitness* acumulado del primer individuo más el *fitness* del segundo individuo que resulta 56, de la misma forma se obtiene el *fitness* acumulado del tercer y cuarto individuo; luego se divide el *fitness* acumulado del primer individuo con el *fitness* de la población obteniendo 0.27 y así se obtiene el *fitness* proporcional hasta llegar el valor de la unidad que representa la ruleta, donde se genera un número de forma aleatoria que representa rodar la ruleta y el primer

valor es 0.57 donde es menor que el segundo individuo de la suma del fitness proporcional, por lo tanto, el individuo seleccionado es el segundo individuo i_2 .

- Selección por Torneo

Se realiza mediante un torneo o comparación entre un subconjunto de individuos elegidos al azar de la población como se muestra en la Figura 8, los beneficios de este método es la velocidad de aplicación y capacidad de prevenir de alguna forma la convergencia prematura pero su desventaja es establecer los parámetros correspondientes al tamaño del subconjunto.

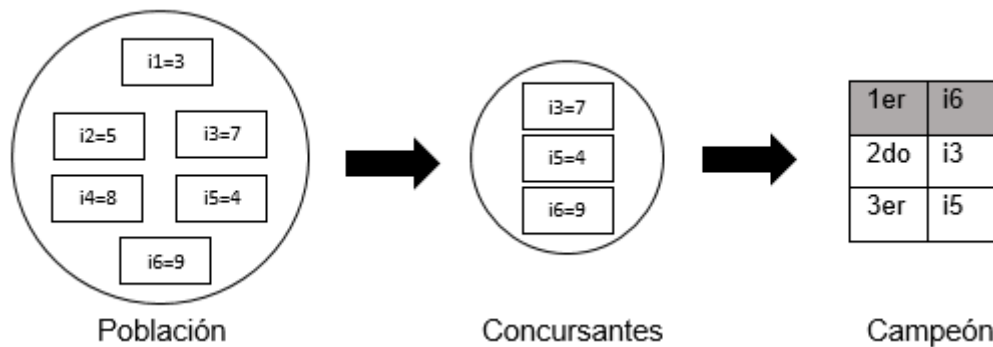


Figura 8. Ejemplo de procedimiento de selección por torneo. Fuente: Arroyo (2013).

Cuando participan muchos individuos en cada torneo la presión selectiva es elevada y los peores individuos tienen menos oportunidades y cuando el tamaño de torneo es reducido la presión selectiva disminuye y los peores individuos tienen mayor

oportunidad de ser seleccionados, mientras el elitismo global es la participación de toda la población y la selección es determinista; el pseudocódigo de este método se muestra en la Figura 9.

1. **Definir** tamaño del subconjunto
 2. **Calcular** el *fitness* acumulado, *fitness* total (P_i)
 3. **Mientras** tamaño del subconjunto > 0 **hacer**
 4. **Elegir** concursante
 5. **Fin de mientras**
 6. **Ordenar los** concursantes de acuerdo a su *fitness* y elegir el ganador del torneo
- Figura 9. Pseudocódigo del método de selección por torneo. Fuente: Arroyo (2013).

- Selección basada en *Ranking*

En este método se realiza mediante el ordenamiento de individuos de la población según su *fitness*, por lo tanto, la probabilidad de selección se asigna basado en el *ranking* como se muestra en la Figura 10.

$$Ranking_i = 2 - PS + \left(2(PS - 1) \times \frac{Pos - 1}{N - 1} \right)$$

Figura 10. Probabilidad de selección del individuo basado en el *ranking*. Fuente: Arroyo (2013).

Donde se considera a N al número de individuos de la población, Pos como la posición del individuo en la población y la PS como la

presión de selección en el intervalo $\langle 1,2 \rangle$. La necesidad de ordenar la población basado en su *ranking* es porque si se ordena con su *fitness*, los mejores individuos poseerían mayor probabilidad de selección como se muestra en la Figura 11.

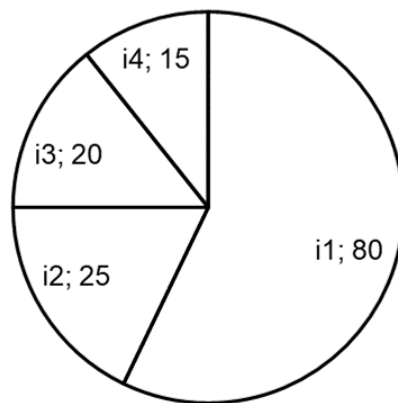


Figura 11. Ordenamiento de individuos basado en su *fitness*. Fuente: Arroyo (2013).

En la Figura 11 se observa cuatro individuos ordenados de acuerdo a su *fitness* en un problema de maximización, en el cual el mejor individuo es el mayor fitness, por lo tanto, el primer individuo tiene un *fitness* de 80 el cual lo convierte en un super individuo y con mayor probabilidad para ser seleccionado; para evitar los super individuos se asigna la probabilidad de selección basado en su *ranking* como se muestra en la Figura 12.

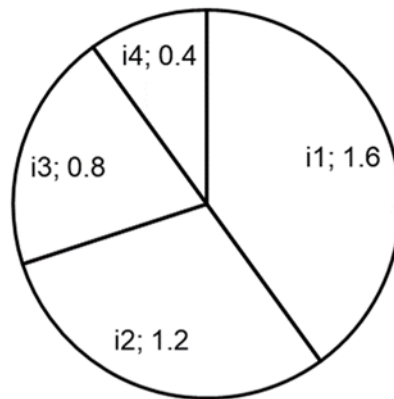


Figura 12. Ordenamiento de individuos basados en su *ranking*. Fuente: Arroyo (2013).

En la Figura 12 se observa que el *ranking* de cada individuo no muestra mayor diferencia para seleccionar individuos, evitando así los super individuos. El algoritmo del procedimiento de selección basada en *ranking* se muestra en la Figura 13.

1. **Generar** el valor n con un valor de 0
2. **Mientras** tamaño de la población $> n$ **hacer**
3. **Generar** el número aleatorio r
4. **Ordenar** población de acuerdo a su ranking
5. **Asignar** los valores de fitness a los individuos de acuerdo a su ranking
6. **Calcular** el *fitness* acumulado, *fitness* total (P_i), y la suma del *fitness* proporcional (SUM)
7. **Si** SUM $< r$ **entonces**
8. **Seleccionar** el primer cromosoma, de otra manera, seleccionar el i -ésimo cromosoma
9. **Fin de si**
10. **Incrementar** el valor de n en una unidad
11. **Fin de mientras**
12. **Devolver** los cromosomas con valores de *fitness* proporcionales al tamaño de la sección de la ruleta seleccionada

Figura 13. Pseudocódigo del método selección basado en *ranking*. Fuente: Arroyo (2013).

2.2.1.4. Operador de Cruzamiento

Es la función más importante del algoritmo genético por la diversidad de la población en el cual se reproducen dos cromosomas padres que concierne el intercambio de genes con el objetivo de reproducir cromosomas hijos y buscar nuevas o mejores soluciones al problema (Eiben y Smith, 2015).

Eiben y Smith (2015) menciona tres tipos de cruzamiento:

- Cruzamiento de un Punto

Se elige un punto de forma aleatoria para dividir cada cromosoma padre en dos segmentos de genes, y los cromosomas descendientes se forman realizando un intercambio del segundo segmento de los cromosomas padres como se muestra en la Figura 14.

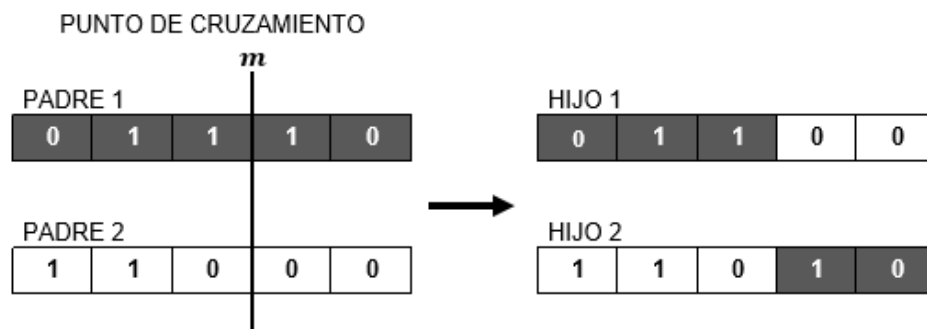


Figura 14. Cruzamiento de un punto. Fuente: Eiben y Smith (2015).

En la Figura 14 se observa el punto de cruzamiento m que está dado de forma aleatoria en el intervalo $[1, l-1]$, donde l es la longitud del cromosoma, por lo tanto, es necesario evitar el punto de cruzamiento

antes de la primera y después de la última posición del cromosoma ($r > 0$) \vee ($r < l$).

- Cruzamiento de m Puntos

Se elige dos o más puntos de cruzamiento de forma aleatoria para dividir el cromosoma en más de dos segmentos de genes contiguos, y los cromosomas descendientes se forman concatenando los segmentos de genes alternativos de los cromosomas padres como se muestra en la Figura 15.

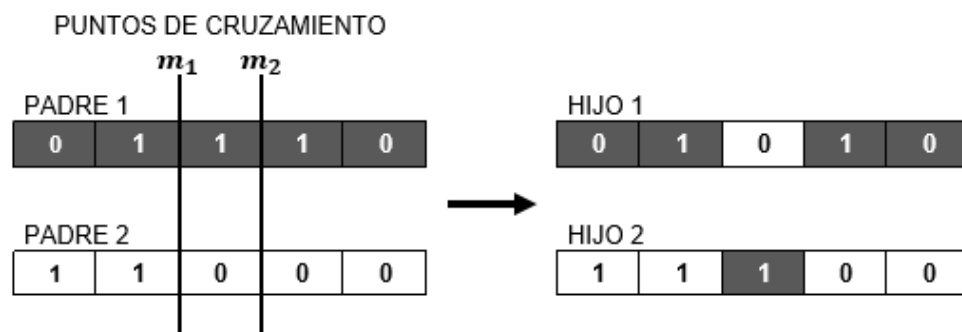


Figura 15. Cruzamiento de m puntos con $m = 2$. Fuente: Eiben y Smith (2015).

En la Figura 15 se observa dos puntos de cruzamiento m_1 y m_2 de forma aleatoria en el intervalo $[1, l-1]$ para dividir el cromosoma en tres segmentos de genes, los cromosomas descendientes se forman con el primer segmento de los cromosomas padres, luego se intercambia el segundo segmento de los cromosomas padres y se adiciona el tercer segmento del cromosoma padre inicial.

- **Cruzamiento Uniforme.**

Este método de cruzamiento manipula cada gen del cromosoma de forma independiente, generando una cadena de variables aleatorias V_j en el intervalo $[0,1]$ comparando con probabilidad de herencia PH que generalmente es 0.5, entonces si $V_j < PH$ se hereda los genes de los padres a los hijos en orden directo, caso contrario se alternan la herencia de los genes como se muestra en la Figura 16.

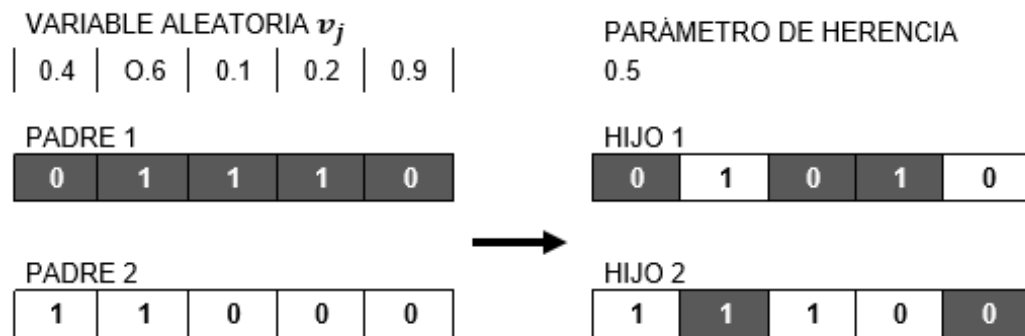


Figura 16. Cruzamiento uniforme. Fuente: Eiben y Smith (2015).

En la Figura 16 se observa que para formar los cromosomas hijos se realiza una herencia de genes de los cromosomas padres, por lo tanto, V_0 es 0.4 y es menor que PH , entonces el primer gen del padre 1 se hereda al hijo 1 y el primer gen del padre 2 se hereda al hijo 2, para la herencia del segundo gen, V_1 es 0.6 y es mayor PH , por lo tanto, el segundo gen del padre 1 se hereda al hijo 2 y el segundo gen del padre 2 se hereda al hijo 1, para la herencia del tercer gen, V_2 es 0.1 entonces el tercer gen del padre 1 y padre 2 se hereda al hijo 1 e hijo 2, para la herencia del cuarto gen, V_3 es 0.2 entonces el

cuarto gen del padre 1 y padre 2 se hereda al hijo 1 e hijo 2, para la herencia del quinto gen, V_4 es 0.9 entonces se alterna la herencia del quinto gen del padre 1 y padre 2 hacia el hijo 2 e hijo 1.

Existe otros métodos de selección como selección con rango lineal, selección con rango exponencial, selección por truncamiento y muestreo estocástico (Jebari y Madiafi, 2013).

2.2.1.5. Operador de Mutación

La mutación es la función encargada de realizar una modificación al gen del cromosoma de forma aleatoria que por lo general solo se elige a uno, la probabilidad de mutación es muy baja pero ayuda a explorar nuevas soluciones y el diseño del algoritmo genético es responsable de elegir la mutación, seleccionado uno o dos cromosomas descendientes producidos por el operador de cruzamiento; en codificaciones de tipo binario consiste en negar el bit del gen como también otra técnica es el intercambio entre dos valores de una matriz (Ahumada, 2014). La mutación es un restablecimiento aleatorio en cada posición del cromosoma de forma independiente, en el cual los genes del cromosoma están codificados con atributos cardinales y tienen la misma probabilidad de ser elegidos para modificar al azar sus valores en lo permitido (Eiben y Smith, 2015).

2.2.2. Procedimiento del Algoritmo Genético

El procedimiento del algoritmo genético convencional inicializa con un número de población inicial (conjunto de individuos) generalmente aleatoria y

cada uno es evaluado de acuerdo a la función objetivo de un problema específico llamada función *fitness*, luego mediante el operador de selección se seleccionan los mejores individuos para someterse al proceso de recombinación; en la fase de recombinación los operadores de cruce y mutación se utilizan para crear nuevos individuos y explorar el espacio de búsqueda o posibles soluciones; los nuevos individuos creados reemplazan a los peores individuos de la población en base a su *fitness*, este proceso se repite hasta alcanzar un criterio de terminación que puede ser dado por el número máximo de generaciones o tiempo límite; el pseudocódigo del procedimiento del algoritmo genético convencional se muestra en la Figura 17 (Jat, 2012).

1. Inicializa población inicial
2. Evalúa individuos de la población
3. **Mientras** la condición de terminación no es alcanzada **Hacer**
4. Seleccionar padres mediante un esquema de selección
5. Cruzamiento de padres para crear descendientes
6. Aplicar mutación a descendiente
7. Reemplazar peor(es) miembro(s) de la población para la siguiente generación
8. **Fin de mientras**

Figura 17. Pseudocódigo del procedimiento del algoritmo genético convencional.

Fuente: Jat (2012).

2.3. Definiciones Operacionales

2.3.1. Caso de Estudio

También llamado estudio de caso, es un método o técnica de investigación que estudia o indaga en profundidad una situación en particular.

2.3.2. Problema de Optimización

Son aquellos que se ocupan de elegir la decisión óptima de un problema, es decir, encontrar cual es el máximo o mínimo de una variable determinado criterio o función sujeto a unas condiciones que da el problema.

2.3.3. Modelo probabilístico

Llamado también modelo estadístico donde es la forma que puedan tomar un conjunto de datos obtenidos de muestreos de datos con comportamiento que se supone aleatorio.

2.3.4. Modelo Determinista.

Modelo matemático donde las mismas entradas o condiciones iniciales producirán las mismas salidas o resultados, no contemplando la existencia del azar o incertidumbre en el proceso.

2.3.2. Generación del Algoritmo Genético.

Representa una reproducción de la población inicial mediante los operadores genéticos.

2.3.3. Restricciones Duras

Reglas de la universidad respecto a los horarios de clases que se tiene que cumplir, el cual no pueden ser obviados bajo ninguna circunstancia.

2.3.4. Restricciones Blandas

Consideraciones por parte de los interesados de la universidad respecto a los horarios de clase, el cual determinará la satisfacción de los interesados y la calidad de la solución.

2.3.5. Espacio de Búsqueda

Cuando los genes pueden tomar distintos valores existentes de otros cromosomas mediante la reproducción

2.3.6. Solución Óptima

La solución que no contiene violación de restricciones duras y algunas violaciones de restricciones blandas.

2.3.7. Locus

Posición fija en un cromosoma que determina la posición de un gen.

2.3.8. Elitismo.

Método que asegura a uno o varios individuos con mejores aptitudes o *fitness* de una población para sobrevivir y continuar participando o formar parte de la siguiente generación.

2.3.9. Convergencia

Es la coincidencia de datos de varios objetos que no puede producir cambios si se realiza alguna recombinación.

2.3.10. Timeslot

Periodo de clases, rango de hora de inicio y final de clases en un día de la semana.

2.3.11. Conflicto de Timeslot

Cuando cierto periodo de clases coincide más de una vez para los criterios del horario de clases tales como: Grupo de estudiantes, aula, docente y curso universitario.

2.3.12. Evento

Horario de clases de un determinado grupo de estudiantes, curso universitario, docente universitario, aula universitaria, *timeslot* y día de la semana.

III. MATERIALES Y MÉTODOS

3.1. Tipo, Diseño y Nivel de Investigación

3.1.1. Tipo de Investigación

Investigación Aplicada.

Según Hernández, Fernández, y Baptista (2014) se realizó una investigación de tipo aplicada porque se utilizó el algoritmo genético para determinar que si es posible mejorar la eficiencia del proceso de elaboración de horarios de clases en universidades con recursos compartidos.

3.1.2. Diseño de Investigación

Diseño No Experimental Transversal o Transeccional.

Según Hernández et al. (2014) se realizó un diseño no experimental, porque no se manipuló la variable independiente, sólo contemplar los fenómenos en su estado natural del mecanismo del proceso de elaboración de horarios de clases en una situación de forma tradicional y una situación utilizando el algoritmo genético; y transversal o transeccional porque se ha recolectado datos del proceso un momento dado o un semestre académico, a partir de los cuales se obtuvieron los datos de la eficiencia del proceso para realizar la comparación de los resultados entre las dos tecnologías como se muestra en la Tabla 3.

Tabla 3: Diseño No Experimental Transversal o Transeccional

Proceso de elaboración de horarios de clases	Semestre académico X	Eficiencia del proceso
Forma tradicional	Observación 1	Y1
Utilizando el algoritmo genético	Observación 2	Y2

3.1.3. Nivel de Investigación

Investigación Descriptiva.

Según Hernández et al. (2014) es de nivel descriptivo porque la investigación solo se ha limitado a indagar la incidencia de la variable independiente, es decir, describir de forma individual sin vincular y otras intervenciones del proceso de elaboración de horarios de clases en una situación de forma tradicional y una situación utilizando el algoritmo genético.

3.2. Unidad de Análisis

La unidad de análisis es una universidad con recursos compartidos en un semestre académico, porque los datos comparativos en una situación de forma tradicional y una situación utilizando el algoritmo genético comprenden el proceso de elaboración de horarios de clases.

3.3. Técnicas de Investigación

3.3.1. Caso de Estudio

Se ha empleado esta técnica para observar el proceso de elaboración de horarios de clases en la Universidad Nacional Agraria de la Selva (UNAS) en una situación de forma tradicional y una situación utilizando el algoritmo genético tal cual ocurre de manera cotidiana sin alterar el proceso, es decir, no realizar experimento alguno. El uso de esta técnica permitió contar con datos reales que constituyen la evidencia empírica.

3.4. Instrumentos de la Investigación

3.4.1. Ficha de Observación:

La ficha de observación sirvió para recolectar los datos de la eficiencia del proceso de elaboración de horarios de clases en una situación de forma tradicional y una situación utilizando el algoritmo genético, que consta de cinco columnas, la primera columna es el registro de la fecha de observación para obtener el número de días que ha definido el tiempo de elaboración del proceso, la segunda columna es para registrar al participante del proceso, porque existe la posibilidad de más de un participante, en la tercera columna se registra la actividad laboral de los participantes designados a realizar los horarios de clases, puesto que existe la posibilidad de realizar otras actividades laborales no consideradas como datos importantes para la investigación y se ha registrado como otros, en la cuarta columna se registra la hora de inicio y final dedicadas a cada actividad laboral, para poder obtener el indicador hora para la dimensión horas-personas dedicadas al proceso de elaboración de horarios de clases,

consultando la tercera columna de actividad laboral y la segunda columna de los participantes y por último la quinta columna, la descripción de la actividad laboral, en el caso del proceso de elaboración de horarios de clases se ha registrado un detalle resumido de los realizado, y para el caso otras actividades laborales, solo se ha considerado otros; el diseño y formato de la ficha de observación se muestra en la Figura 18.

Fecha	Participante	Actividad laboral	Hora inicio y fin	Descripción

Figura 18. Diseño y formato de la ficha de observación para la recolección de datos del proceso de elaboración de horarios de clases.

3.5. Procedimiento de la Investigación

La investigación se ha desarrollado en seis fases en las cuales se describen a continuación:

3.5.1. Implementación del Algoritmo Genético

Se ha realizado la revisión de literatura sobre el algoritmo genético, su ventaja y desventaja para solucionar problemas de optimización, se ha realizado algunas variantes del algoritmo genético, el diseño de codificación del problema, la identificación de las restricciones duras, se determina los métodos de los operadores genéticos.

3.5.2. Construcción de la Aplicación Web

El algoritmo genético está implementado en un sistema web bajo el IDE visual estudio 2017, y cada generación es guardada en archivos XML, para no utilizar más recursos de memoria RAM, se ha desarrollado los módulos de carga y validación de información de entrada, el registro de parámetros para poder elaborar los horarios de clases mediante el algoritmo genético, debe tener una vista del horario de clases terminado con filtros de cada criterio de horario de clases tales como: grupo de estudiantes, curso, docente y aulas universitarias; y poder descargar el archivo del horario de clases.

3.5.3. Obtención de Datos del Proceso en una Situación Forma Tradicional

Se ha registrado los ítems de la ficha de observación del proceso de elaboración de horarios de clases de forma tradicional del semestre académico 2018 – I de la UNAS, desde el momento en que inicia y culmina el proceso, sin intervenir y alterar del mismo.

3.5.4. Obtención de Datos del Proceso en una Situación Utilizando el Algoritmo Genético

Se ha registrado los ítems de la ficha de observación del proceso de elaboración de horarios de clases utilizando el algoritmo genético, del semestre académico 2018 – I de la UNAS, desde el momento en que inicia y culmina el proceso, sin realizar alguna intervención o manipulación de datos.

3.5.5. Procesamiento de Datos

Los datos obtenidos por medio de la ficha de observación del proceso de elaboración de horarios de clases en las dos situaciones para el semestre académico 2018–I de la UNAS, con las dos tecnologías: de forma tradicional y utilizando el algoritmo genético, deben ser procesados en base a los indicadores y dimensiones de la investigación.

3.5.6. Análisis de Resultados

Una vez procesados los datos en base a las dimensiones se realizará la comparación del proceso de elaboración de horarios de clases en las dos situaciones del semestre académico 2018 – I de la UNAS, en base a las dimensiones de la investigación, con la finalidad de determinar si es posible o no, mejorar la eficiencia del proceso.

3.6. Proceso de Elaboración de Horarios de Clases

En el caso de estudio UNAS cada semestre académico se realizan los horarios de clases; la información de entrada pasó por las dos tecnologías del proceso de elaboración de horarios de clases: forma tradicional y una situación utilizando el Algoritmo Genético; obteniendo como salida el horario de clases terminado para cada situación, como se observa en la Figura 19.

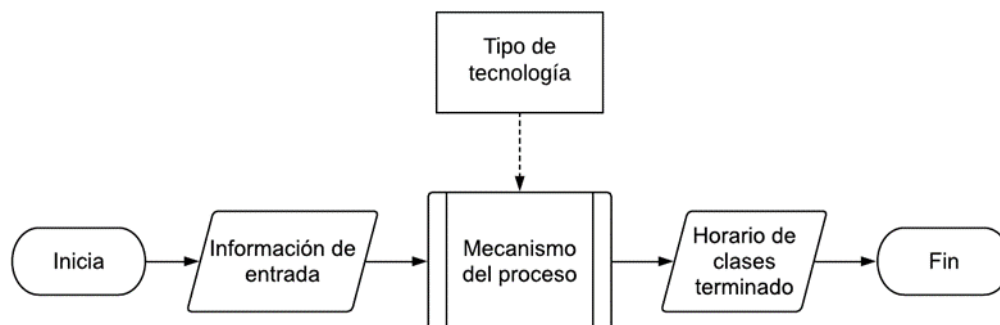


Figura 19. Esquema general del proceso de elaboración de horarios de clases.

En la Figura 19 se observa que el proceso de elaboración de horarios de clases inicia con la entrega de información de entrada del departamento académico de las escuelas académicas a la Dirección de Coordinación y Desarrollo Académico (DICDA), la entrega se realiza en Mesa de Partes para dar parte al Área de Registro y registrar en el Sistema de Gestión de Horarios versión 3.7 Módulo Registro de Carga Académica, el documento de carga académica pasa al área de informática, y el área de informática elabora los horarios de clases, finalizando el proceso de elaboración de horarios de clases obteniendo como resultado el horario de clases terminado, como se observa en la Figura 20.

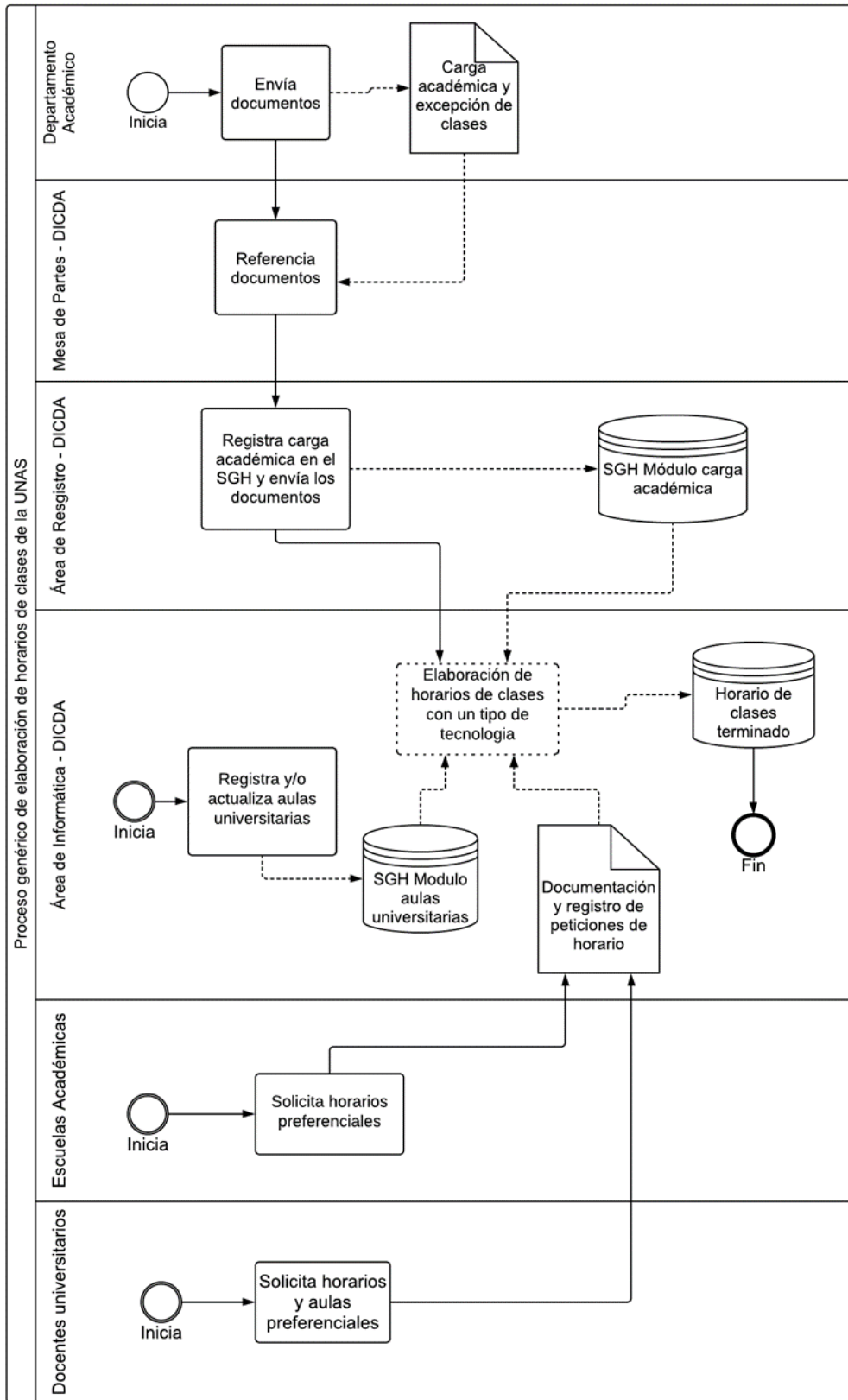


Figura 20. Proceso genérico de elaboración de horarios de clases en la UNAS.

La información de entrada contiene datos necesarios para elaborar los horarios de clases, incluyendo algunas restricciones del horario de clases, y el output debe mostrar la información necesaria de horario de clases para los recursos de la universidad.

3.6.1. Información de Entrada

La información de entrada que necesita el proceso de elaboración de horarios de clases, en cada tecnología de elaboración de horarios de clases los datos son el mismo. La información de entrada son los siguientes:

3.6.1.1 Carga Académica

Los departamentos académicos elaboraron la carga académica el cual contiene los datos del curso universitario, horas de teoría y práctica, escuela profesional a la que pertenece el curso, departamento académico del curso universitario, docente a cargo del curso universitario, departamento académico del docente universitario al que pertenece. Los documentos de la carga académica emitidos por los departamentos académicos, llega a mesa de partes de DICDA, el mismo pasa al área de registro de DICDA, el área de registro, registra la carga académica en el sistema de registro SGH.

3.6.1.2 Aulas Universitarias

En el Sistema de Gestión de Horarios existe un módulo de registro de aulas universitarias, donde están registradas las aulas universitarias que incluye las aulas para teoría, centro de cómputo y laboratorios; el registro y

actualización de aulas universitarias lo realizan los responsables del área de informática de DICDA - UNAS.

3.6.1.3 Restricciones Duras

Se ha considerado cuatro restricciones duras y son las siguientes:

- Un aula universitaria sólo puede ser asignado a un grupo de estudiantes, curso y docente universitario en un determinado día y hora de clases.
- Un docente universitario sólo puede dictar una clase de un curso universitario a un grupo de estudiantes en un determinado día y hora de clases.
- Un grupo de estudiantes sólo puede tener asignado una clase de un curso, docentes y aula en un determinado día y hora de clases.
- Un curso universitario sólo puede estar asignado a un grupo de estudiantes, docente y aula universitaria en un determinado día y hora de clases.

3.6.1.4 Horas y Días de Clases

Para realizar el proceso de elaboración de horarios de clases se debe asignar la hora y día de la semana al horario de clases, ya sea de conocimiento heurístico por parte de los participantes del proceso de forma tradicional o por medio de un algoritmo que utilizará el algoritmo genético.

3.6.1.5 Rango de Horarios de Clases y de Alimentación Definidos por la Universidad

No se ha considerado entre las restricciones duras porque en el proceso en una situación utilizando el algoritmo genético la asignación aleatoria de día de clases y *timeslot* siempre respeta estas consideraciones y desde el primer momento no existe horarios de clases en estas condiciones para ser penalizadas por la función *fitness*, en el cual se detallan lo siguiente:

- Los horarios de clases deben estar asignados entre límite de días y horas de clases determinado por la universidad.
- La asignación de horarios de clases debe respetar los horarios de alimentación en una determinada hora y todos los días de clases.

3.6.2. Mecanismo del Proceso

El proceso de elaboración de horarios de clases utiliza dos mecanismos, uno con el sistema de asignación manual y otro de asignación automática, que utilizan las siguientes tecnologías forma tradicional y con el algoritmo genético.

3.6.2.1 Tecnología de Forma Tradicional

La elaboración de horarios de forma tradicional se realiza en el Sistema de Gestión de Horarios (SGH) Versión 3.7 de asignación manual a cargo del Área de Informática de DICDA - UNAS.

3.6.2.2 Tecnología Utilizando el Algoritmo Genético

La elaboración de horarios de clases de forma automática mediante el algoritmo genético está implementada en el Sistema de Elaboración Automática de Horarios de Clases (SEAHC).

3.6.3. Horario de Clases Terminado

Como salida del proceso de elaboración de horarios de clases en las dos situaciones es el horario de clases terminado y debe evitar el incumplimiento de restricciones duras para las dos tecnologías, donde el porcentaje de conflictos de *timeslot* de los criterios del horario de clases: Grupo de estudiantes, docentes, aula y cursos universitario; debe resultar 0%.

3.7. Proceso en una Situación de Forma Tradicional

El proceso inicia cuando los documentos de carga académica elaborados por los departamentos académicos de la UNAS son recibidos por el área de informática y registrados en el SGH Módulo Cursos Activos, como se observa en la Figura 27. Las aulas universitarias se encuentran registradas en el SGH Módulo registro de aulas, como se observa en la Figura 28. El proceso para elaborar los horarios de clases debe estar basado en las restricciones documentarias y se tiene el conocimiento de procesos anteriores. Los responsables del proceso desde un ordenador de la oficina, ingresan al SGH de asignación de *timeslot* manual, entendiéndose que este sistema permite elaborar los horarios de clases de forma manual, programando cada carga académica en base al comportamiento heurístico del usuario, una vez finalizado el proceso, se

obtiene como salida el horario de clases terminado como se muestra en la Figura 21.

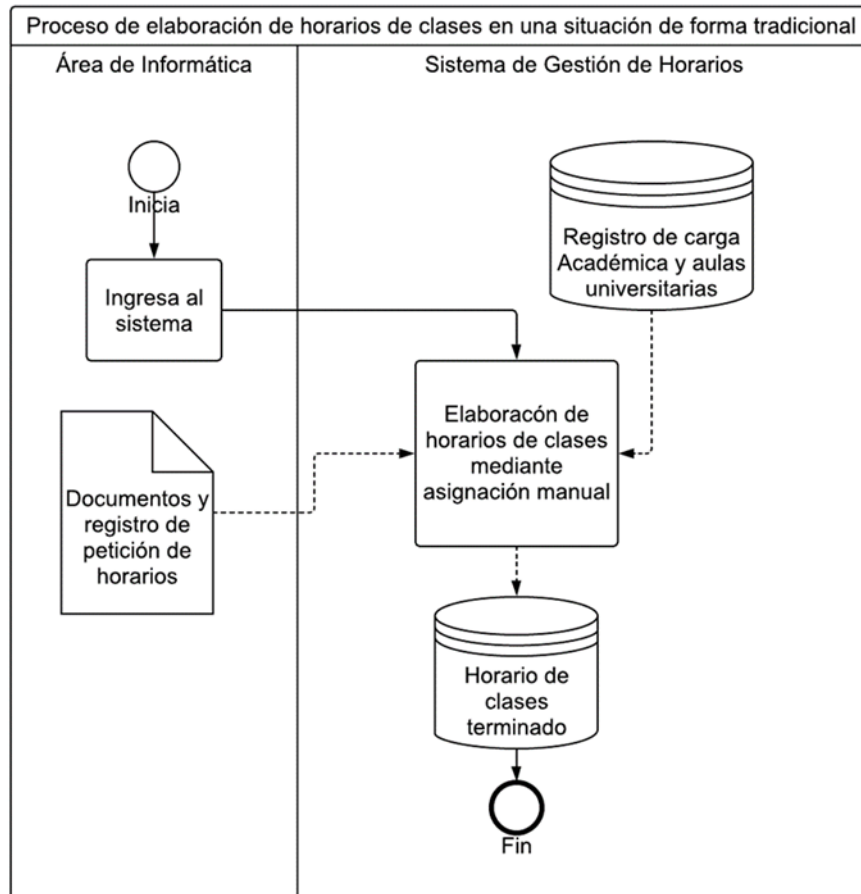


Figura 21. Proceso de elaboración de horarios de clases en una situación de forma tradicional.

3.7.1. Tecnología Utilizada

El Sistema de Gestión de horarios permite elaborar de forma manual los horarios clases por el usuario de forma heurística y guiándose de horarios de clases de semestres académicos anteriores, asignando un *timeslot* y día de clases para los diferentes filtros: Ciclo académico, Curso universitario, docente universitario y aula universitaria; teniendo conocimiento de las restricciones para

elaborar los horarios de clases en base a conocimiento heurística y revisión documentaria.

3.7.1.1 Descripción de Asignación Manual

El usuario selecciona alguno de los siguientes filtros de asignación de horarios de clases, seguido selecciona un *timeslot* y asigna manualmente el horario de clases, define tipo de clases y asigna recursos de la universidad, los recursos universitarios solo mostrarán datos pertenecientes al filtro de asignación de horarios seleccionado, continuando así hasta completar los recursos de horario de clases de filtro seleccionado. Este bucle continuará hasta completar el horario de clases de la universidad para todos los cursos universitarios como se muestra en la Figura 22.

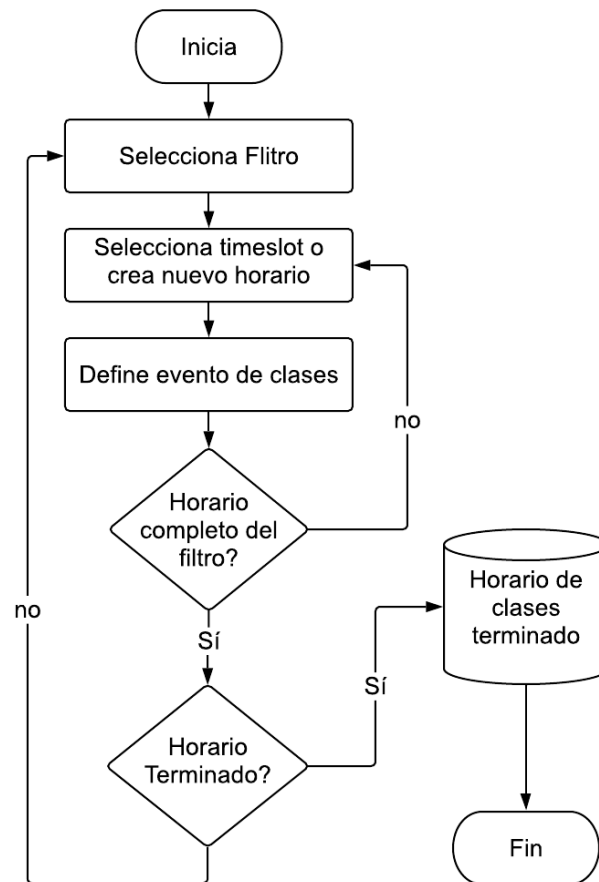


Figura 22. Diagrama UML de asignación Manual de Horario de Clases del proceso de elaboración de horarios de clases en una situación de forma tradicional.

a. Selección de filtros

Existen cinco tipos de filtros para asignar de forma manual los horarios de clases tales como: Aula, docente, ciclo, alumno y curso universitario como se muestra en la Figura 23.

Procesar por		Ciclo 01 ACTIVIDADES LIBRES		
Hor \ Dia		MARTES	MIERCOLES	JUEVES
06-07 am.				
07-08 am.				
08-09 am.				
09-10 am.				
10-11 am.				
11-12 am.				
12-01 pm.				

Figura 23. Vista de selección de filtros en el sistema de gestión de horarios del proceso de elaboración de horarios de clases en una situación de forma tradicional.

b. Define evento de clases

Con el filtro correspondiente te muestra un registro de un horario de clases con *timeslot* múltiple para poder seleccionar y mostrar algunas opciones, muestra los datos correspondientes para asignar el horario de clases, donde se asigna el curso, docente, aula, día, hora, duración en horas de clases, tipo de clases (teoría o Práctica) y la opción de omitir cruces de horarios como se muestra en la Figura 24.

The screenshot shows a window titled "Registrar Horario" with the following fields and controls:

- Curso:** NIS103B TECNICAS DE PROGRAMACION
- Docente:** 4205 JUIPA CAMPO, NOEL
- Aula:** IS02 LABORATORIO DE SISTEMAS DE INFORMACION - FIIS 20
- Dia:** Jueves
- Hora:** 19
- Duración:** 2
- Nro Grupo de Práctica:** (empty)
- Radio buttons:** Hora de Teoría (selected), Hora de Práctica
- Checkbox:** Omitir Cruces de Horario (unchecked)
- Buttons:** Registrar, Cancelar

Figura 24. Asignación manual de un evento en el sistema de gestión de horarios del proceso de elaboración de horarios de clases en una situación de forma tradicional.

3.7.1.2 Sistema de Gestión de Horarios (SGH)

- **Tipo de Software**

El Sistema de Gestión de horarios versión 3.7 de asignación de *timeslot* manual de entorno escritorio (o *Desktop*) es clasificado como un sistema de procesamiento de transacciones desde un punto de vista empresarial en el sistema de información; y está alojado en el Centro de Tecnologías de Información y Comunicación (CTIC) de la UNAS en cual se puede tener acceso mediante la red corporativa en la oficina DICDA y el centro de mando de las aulas universitarias como se muestra en la Figura 25.

- **Arquitectura**

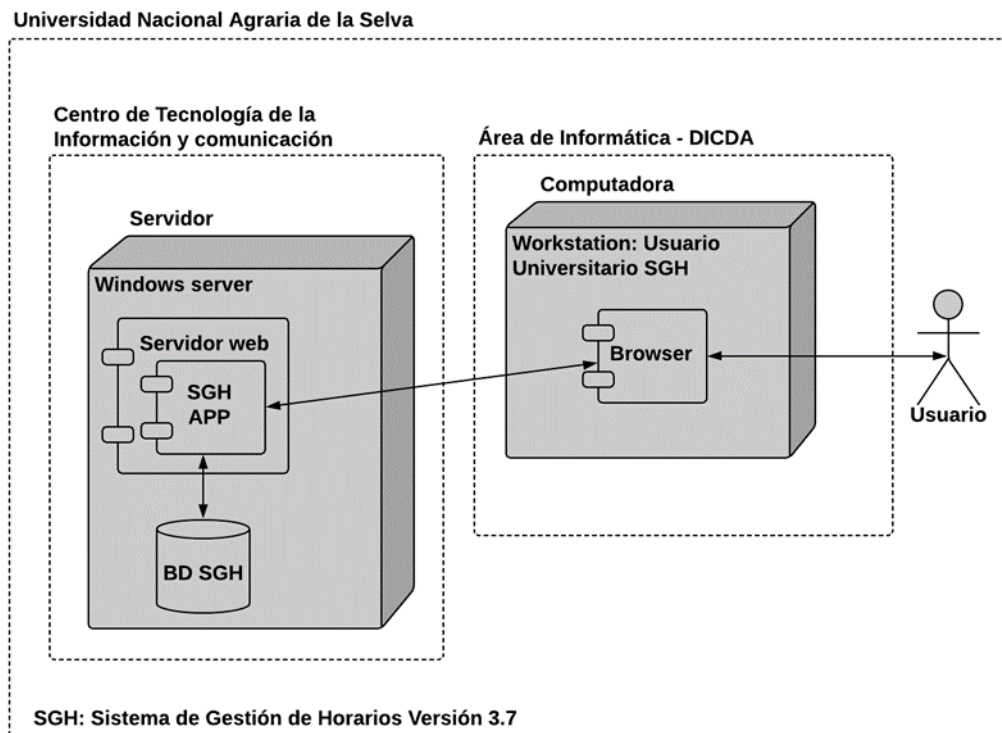


Figura 25. Diagrama de Despliegue del sistema de Gestión de Horarios del proceso de elaboración de horarios de clases en una situación de forma tradicional.

- **Funcionalidades**

Se puede asignar manualmente los horarios de clases por cinco tipos filtros: ciclo académico, docente universitario, aula universitaria y alumnos; cuenta con un módulo de registro de las aulas universitarias, disponibilidad de aula universitarias, horarios de los alumnos universitarios y el módulo de registro de aulas universitarias, donde se registra y actualiza las aulas universitarias. El sistema horario de clases, muestra de color celeste un evento en la vista del horario de clases un criterio cuando existe un conflicto de *timeslot* de un curso, docente o aula universitaria.

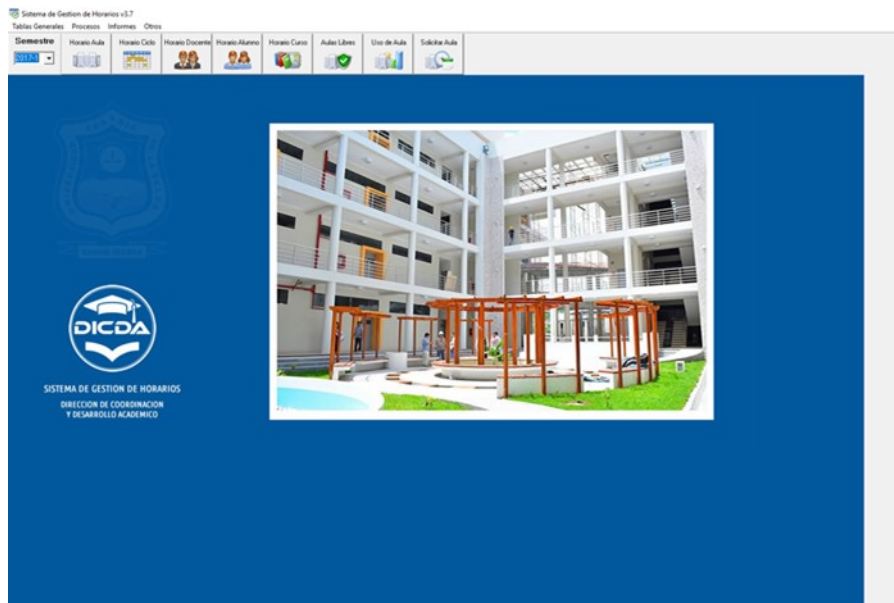


Figura 26. Sistema de Gestión de Horario versión 3.7 del proceso de elaboración de horarios de clases en una situación de forma tradicional.

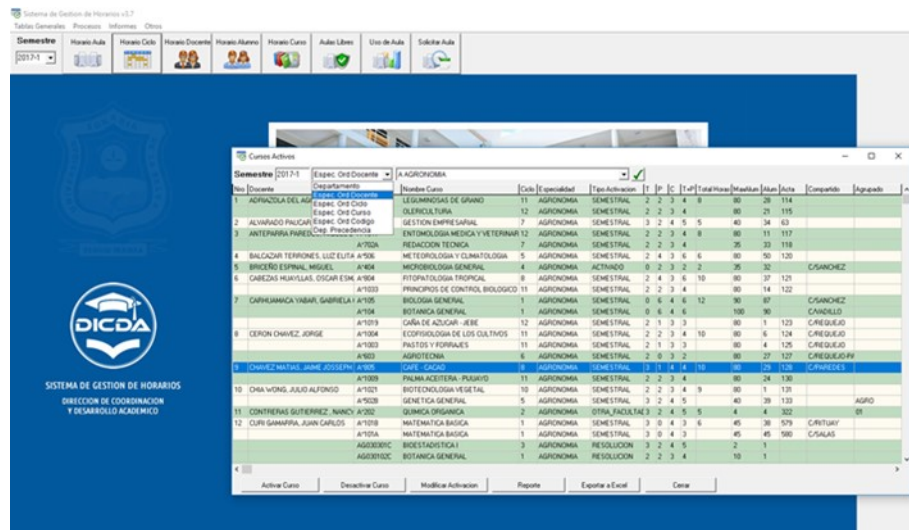


Figura 27. Carga académica en el sistema de gestión de horarios versión 3.7 del proceso de elaboración de horarios de clases en una situación de forma tradicional.

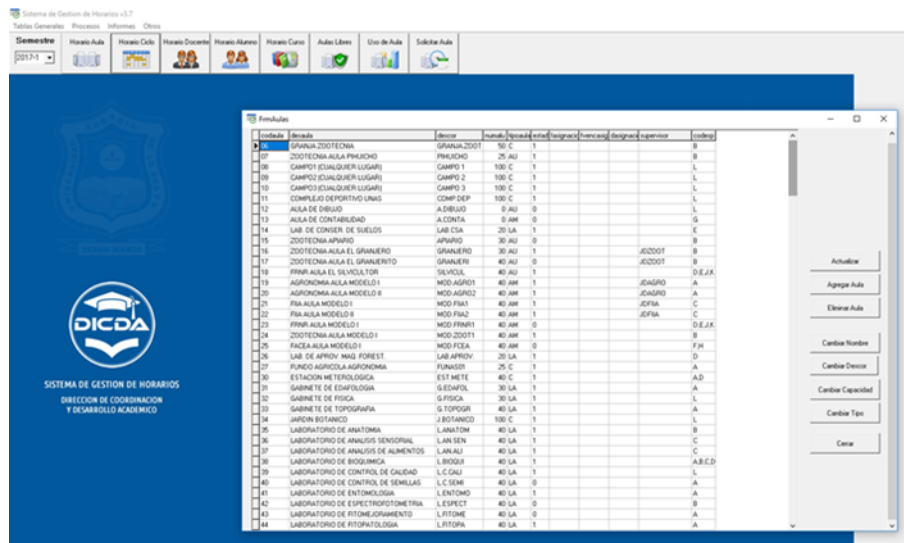


Figura 28. Registro de aulas universitarias en el sistema de gestión de horarios versión 3.7 del proceso de elaboración de horarios de clases en una situación de forma tradicional.

Procesar por **CICLO** Ciclo 05 INGENIERIA EN INFORMATICA Y SISTEMAS I Colorear Tipo de Hora Aula Mínimo de 20 Alumnos

Hor \ Día	LUNES	MARTES	MIÉRCOLES	JUEVES	VIERNES	SABADO	DOMINGO
06-07 am.							
07-08 am.							
08-09 am.	IS030505 ESTADÍSTICA Y PRO 0842 BERMUDEZ PINO, WILME E204 PABELLON NUEVO E-12	IS030503 INGENIERIA DE REQ 3035 GARCIA VILLEGAS, CHRI IS05 AUDITORIO FIIS		IS030503 INGENIERIA DE REQ 3035 GARCIA VILLEGAS, CHRI IS05 AUDITORIO FIIS			
09-10 am.	IS030505 ESTADÍSTICA Y PRO 0842 BERMUDEZ PINO, WILME E204 PABELLON NUEVO E-12	IS030503 INGENIERIA DE REQ 3035 GARCIA VILLEGAS, CHRI IS05 AUDITORIO FIIS		IS030503 INGENIERIA DE REQ 3035 GARCIA VILLEGAS, CHRI IS05 AUDITORIO FIIS			
10-11 am.		IS030501B INACTIVO ? 9999112 NO ACTIVO ? IS03 LABORATORIO DE SISTE	IS030504 REDES Y CONECTIV 71018 SANTILLAN RUIZ, JOSE A304 LAS ORQUIDEAS A304 ->	IS030505 ESTADÍSTICA Y PRO 0842 BERMUDEZ PINO, WILME A305 LAS ORQUIDEAS A305 ->			
11-12 am.	IS030504 REDES Y CONECTIV 71018 SANTILLAN RUIZ, JOSE IS04 LABORATORIO DE REDES	IS030501B INACTIVO ? 9999112 NO ACTIVO ? IS03 LABORATORIO DE SISTE	IS030504 REDES Y CONECTIV 71018 SANTILLAN RUIZ, JOSE A304 LAS ORQUIDEAS A304 ->	IS030505 ESTADÍSTICA Y PRO 0842 BERMUDEZ PINO, WILME A305 LAS ORQUIDEAS A305 ->			
12-01 pm.	IS030504 REDES Y CONECTIV 71018 SANTILLAN RUIZ, JOSE IS04 LABORATORIO DE REDES	IS030501B INACTIVO ? 9999112 NO ACTIVO ? IS03 LABORATORIO DE SISTE	IS030504 REDES Y CONECTIV 71018 SANTILLAN RUIZ, JOSE A304 LAS ORQUIDEAS A304 ->	IS030505 ESTADÍSTICA Y PRO 0842 BERMUDEZ PINO, WILME A305 LAS ORQUIDEAS A305 ->			
01-02 pm.							
02-03 pm.							
03-04 pm.		IS030502 GESTION DE BASE D 4205 JUJIRA CAMPO, NOEL IS02 LABORATORIO DE SISTE	IS030501 COSTOS Y PRESUP 1280 CHUCOS BAQUERIZO, N IS03 LABORATORIO DE SISTE				
04-05 pm.	IS030504B INACTIVO ? 9999115 NO ACTIVO ? IS04 LABORATORIO DE REDES	IS030502 GESTION DE BASE D 4205 JUJIRA CAMPO, NOEL IS02 LABORATORIO DE SISTE	IS030501 COSTOS Y PRESUP 1280 CHUCOS BAQUERIZO, N IS03 LABORATORIO DE SISTE	IS030502 GESTION DE BASE D 4205 JUJIRA CAMPO, NOEL IS02 LABORATORIO DE SISTE	IS030501B INACTIVO ? 9999112 NO ACTIVO ? IS03 LABORATORIO DE SISTE		
05-06 pm.	IS030504B INACTIVO ? 9999115 NO ACTIVO ? IS04 LABORATORIO DE REDES	IS030502 GESTION DE BASE D 4205 JUJIRA CAMPO, NOEL IS02 LABORATORIO DE SISTE	IS030501 COSTOS Y PRESUP 1280 CHUCOS BAQUERIZO, N IS03 LABORATORIO DE SISTE	IS030502 GESTION DE BASE D 4205 JUJIRA CAMPO, NOEL IS02 LABORATORIO DE SISTE	IS030501B INACTIVO ? 9999112 NO ACTIVO ? IS03 LABORATORIO DE SISTE		
06-07 pm.							

Figura 29. Asignación manual de horario de clases en el sistema de Gestión de Horarios del proceso de elaboración de horarios de clases en una situación de forma tradicional.

3.8. Proceso en una Situación Utilizando el Algoritmo Genético

El proceso inicia cuando la información de entrada de aula universitaria y Carga Académica de los departamentos académicos de la UNAS, están registradas en el Sistema de Gestión Horarios. Los responsables de elaborar los horarios de clases descargan la información de entrada de Carga Académica y el registro de aulas universitarias del Sistema de Gestión de Horarios como se muestra en la Figura 27 y Figura 28, para construir la información de entrada de tipo XML, basados en el formato de la Tabla 4 y Tabla 5, así mismo se ha construido la información de entrada de Grupo de Estudiantes y restricciones basados en el formato de la Tabla 6 y Tabla 7. El responsable ingresa a SEAHC y carga los la información de entrada de tipo XML en la sección carga de información de entrada como se muestra en la Figura 44, para luego ser validado el formato de información de entrada, y no la confidencialidad del mismo, si el formato de información de entrada es correcto, SEAHC permite continuar e ingresar la configuración para elaborar los horarios de clases utilizando el algoritmo genético como se muestra en la Figura 45; caso contrario muestra observaciones de información de entrada. SEAHC contiene archivos de tipo XML de registro de día de la semana y *timeslot* como se muestra en la Tabla 8 y Tabla 9. El responsable ha ejecutado el ítem de elaborar horarios de clases y SEAHC ha elaborado utilizando el algoritmo genético encontrando una solución óptima u horario de clases sin violación de restricciones duras finalizando el proceso de elaboración de horarios de clases y muestra el horario de clases terminado para visualizar en los distintos criterios y descargar el archivo de tipo XML, caso contrario no muestra el horario de clases. Para visualizar las

actividades del proceso en una situación utilizando el algoritmo genético observar la Figura 30.

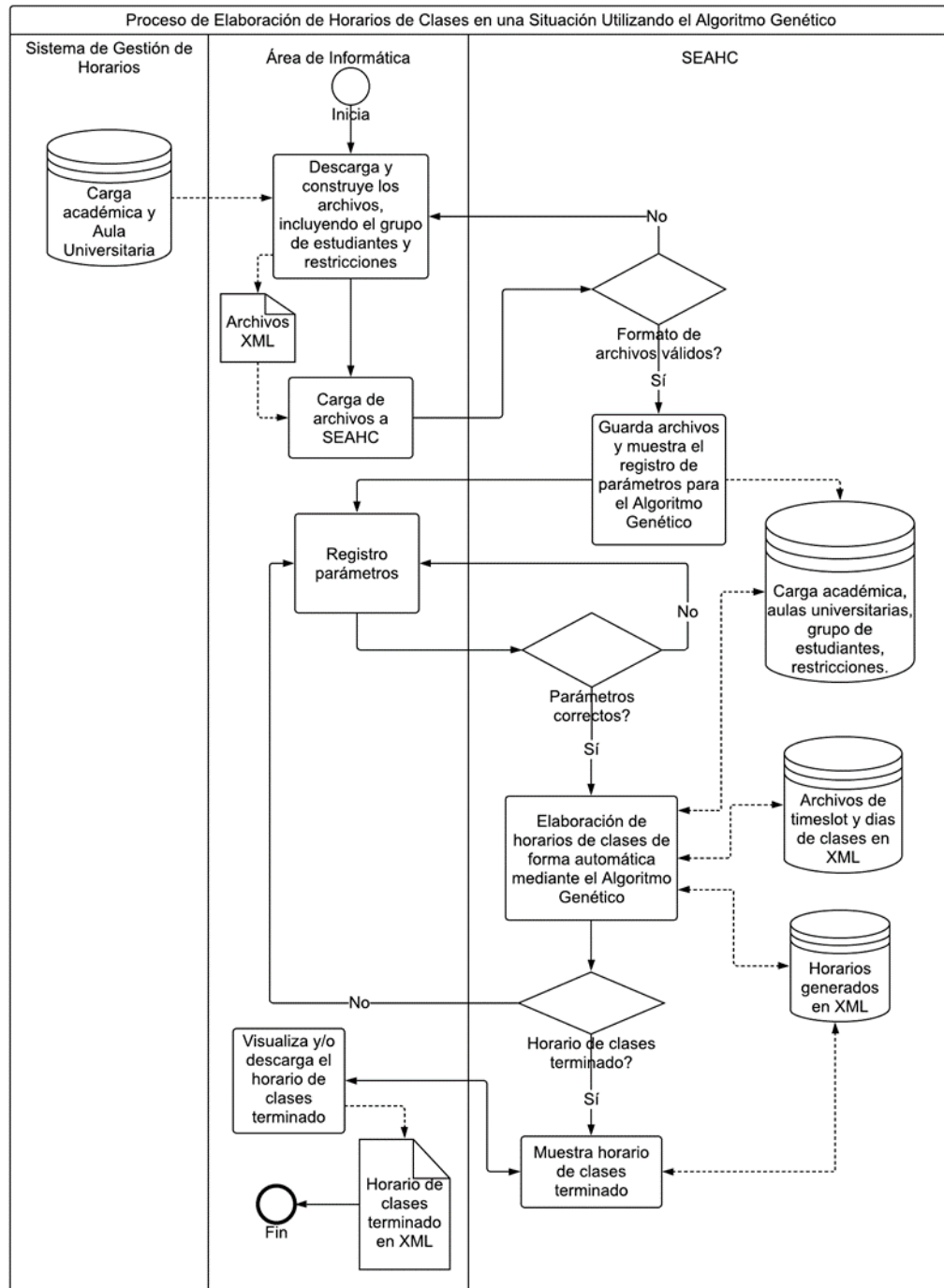


Figura 30. Proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.

Tabla 4. Registro de información de entrada - carga académica para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.

Campo	Descripción
Código del curso universitario	Tipo de dato: carácter
Nombre del curso universitario	Tipo de dato: carácter
Código de la escuela académica	Tipo de dato: carácter
Número de Horas de teoría	Tipo de dato: entero
Número de horas de práctica	Tipo de dato: entero
Código del docente universitario	Tipo de dato: carácter
Datos del docente universitario	Apellidos y nombres del docente universitario; tipo de dato: carácter
Código de la entidad (departamento) académico del docente universitario	Tipo de dato: carácter
Curso aula de práctica	0= No utilizará aula de práctica 1= Sí utilizará aula de práctica Nota: si el valor es igual a cero, los datos posteriores son inválidas.
Curso total de clases aula de práctica	0 = No, 1= Sí. Nota: si el valor es cero, las clases de práctica es en un aula de práctica y si es uno, todas las clases son realizadas en el aula de práctica
Código del curso de práctica	Tipo de dato: carácter

Tabla 5. Registro de información de entrada - aula universitaria para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.

Campo	Descripción
Código del aula universitaria	Tipo de dato: entero
Nombre del aula universitaria	Tipo de dato: carácter
Capacidad del aula universitaria	Tipo de dato: entero
Tipo del aula universitaria	0 = teoría, 1 = práctica. Tipo de dato: entero

Tabla 6. Registro de información de entrada - grupo de estudiantes para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.

Campo	Descripción
Código de grupo de estudiantes	Tipo de dato: carácter
Código de escuela académica	Tipo de dato: carácter
Ciclo universitario	Tipo de dato: entero
Cantidad de estudiantes	Valor aproximado. Tipo de dato: entero
Cursos universitarios	Arreglo de cursos de la carga académica Tipo de dato: carácter.

Tabla 7. Registro de información de entrada de limitaciones para horario de clases para el proceso en una situación utilizando el algoritmo genético.

Horario de la Universidad	
Día de la semana de inicio de clases	Tipo de dato: entero
Día de la semana de final de clases	Tipo de dato: entero
Hora de inicio de clases	Tipo de dato: entero
Hora de culminación de clases	Tipo de dato: entero

Hora de libre de la semana	
Hora libre	Arreglo de horas libres si lo prefiere. Tipo de dato: entero

Tabla 8. Registro de días de la semana para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.

Código	Día de clases
1	Lunes
2	Martes
3	Miércoles
4	Jueves
5	Viernes
6	Sábado
7	Domingo

Tabla 9. Registro de información de entrada de *timeslot* para el proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.

Código	Descripción
215	07:00 – 09:00
219	09:00 – 11:00
223	11:00 – 13:00
229	14:00 – 16:00
233	16:00 – 18:00
239	19:00 – 21:00

3.8.1. Tecnología Utilizada

SEAHC ha permitido elaborar de forma automática los horarios de clases mediante el algoritmo genético asignando *timeslot* aleatorio a un evento del criterio del horario de clases tales como: grupo de estudiantes, curso universitario, docente universitario y aula universitaria; respetando las restricciones de horario inicial, final de clases y horarios de alimentación establecidos por la universidad. En cada generación del algoritmo genético las violaciones de restricciones duras disminuyen en su mayoría hasta encontrar un horario de clases óptimo, es decir, sin violaciones de restricciones duras.

3.8.1.1. Descripción del Algoritmo Genético

El algoritmo genético funciona con un conjunto de soluciones o individuos, el individuo es codificado de tipo no binario (número y letra) llamado genotipo. Se realiza una serie de fases organizadas tales como: generación de

la población inicial aleatoria, cálculo de la solución óptima, criterio de terminación, para dar solución al problema de horario de clases de la universidad con recursos compartidos, el algoritmo hace evolucionar a una población de individuos con acciones aleatorias, seleccionando a los individuos más aptos para la reproducción y mutación; luego los individuos descendientes se guardan en la población inicial de la siguiente generación, la generación anterior es guardada en espacio de disco del servidor de forma comprimida con la opción de mantener ciertas generaciones y los demás son eliminadas, para disminuir el uso de espacio de disco del servidor.

Componentes de Algoritmo Genético:

Los términos que se utilizará para el algoritmo genético para resolver el problema de elaboración de horarios de clases se muestran en el Tabla 10.

Tabla 10. Términos del algoritmo genético para el proceso de elaboración de horarios de clases.

Términos de algoritmo genético	Descripción para la investigación
Gen	Datos no binarios del evento de clases para formar el horario de clases.
Cromosoma	Información genética representada en eventos de clases que conforma el horario de clases de la universidad.
<i>Fitness</i> detallado	Coste de penalización de violación de restricciones duras de los criterios del horario de clases.

Individuo	Horario de clases y su <i>fitness</i> detallado.
Población inicial	Conjunto de horarios de clases para iniciar el funcionamiento del algoritmo genético.
<i>Fitness</i> ajustado	<i>fitness</i> de tipo decimal para resolver problemas de minimización, mientras más cercano a uno es él mejor individuo.
Solución óptima	Horario de clases sin violación de restricciones duras para los criterios de horario de clases o el valor numérico del <i>fitness</i> duro igual a cero.
Generación	Conjunto de horarios de clases producto de la reproducción de una población inicial.

- **Representación y Codificación del Cromosoma**

Para codificar la solución del problema de horarios de clases para universidades con recursos compartidos, se diseña el cromosoma que contiene la información genética en cadena no binario. El gen está representado por: Grupo de estudiante, Curso universitario, docente universitario, aula universitaria, día de clases y *timeslot* como se observa en la Figura 31.

Grupo de estudiantes	Curso universitario	Docente universitario	Aula universitaria	Día de clases	<i>Timeslot</i>
FIIS01	IS030101	7184	D304	4	219

Figura 31. Representación no binaria de un gen del cromosoma de la tecnología utilizando el algoritmo genético.

En la Figura 31 se observa a un gen del cromosoma indicando el grupo de estudiantes de código FIIS01, curso universitario de código IS030101, docente universitario de código 7184, aula universitaria de código D3014, día de clases de código 4 y hora de clases de código 219. Cada gen generado se va concatenando para formar el cromosoma y al final se adiciona el *fitness* detallado para obtener la representación de un individuo como se muestra en la Figura 32.

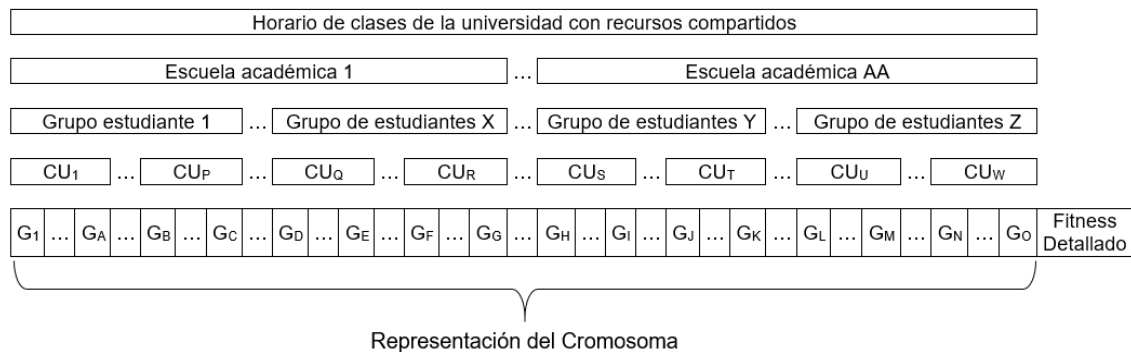


Figura 32. Representación del cromosoma en la tecnología utilizando el algoritmo genético.

En la Figura 32 se observa los genes concatenados para formar un cromosoma, donde el G_1 (gen 1) hasta G_A forman parte de CU_1 (Curso universitario 1); el GU_1 (Grupo de estudiantes 1) contiene de 1 a varios cursos universitarios, la Escuela académica 1, contiene uno o varios grupos de estudiantes, y uno o varias escuelas académicas pertenecen a la universidad con recursos compartidos. El cromosoma pasa por un proceso de cálculo de *fitness* obteniendo el *fitness* detallado que se concatena al final del cromosoma.

- **Representación y Cálculo del *Fitness***

Llamado también función de aptitud, es el método que evalúa y mide la calidad del individuo en el ambiente, es decir, que tan apto es la solución para el problema. El cromosoma generado de forma aleatoria o mediante operadores los genéticos de cruzamiento y mutación, pasa por un proceso de cálculo de *fitness* detallado de la violación de restricciones duras, cada restricción dura violada tiene un costo de penalización como se muestra en la Tabla 11.

Tabla 11. Coste de penalización de violación de restricciones duras de los criterios del horario de clases.

Criterio del horario de clases	Restricciones duras	Costo de penalización
Grupo de estudiantes	RD1. Conflicto de <i>timeslot</i> de clases	1
Docente universitario	RD2. Conflicto de <i>timeslot</i> de clases	1
Aula universitaria	RD3. Conflicto de <i>timeslot</i> de clases	1
Curso universitario	RD4. Cuando un curso se agenda más de un <i>timeslot</i> en un día	1

Los costes de penalización de violación de restricciones duras son acumulado y distribuido por cada criterio de horarios de clases en el *fitness* detallado como se muestra en la Figura 33.

$VRD1$	$VRD2$	$VRD3$	$VRD4$
--------	--------	--------	--------

Figura 33. Representación de *fitness* detallado del cromosoma.

Donde:

$$VRD1 = \sum_{k=1}^{ge} \sum_{i=1}^{dc} RD1_{ki} \quad (1)$$

$VRD1$ es costo total de violación de restricción dura del grupo de estudiantes, conflicto de *timeslot* entre diferentes cursos del grupo de estudiantes; ge es total de grupo de estudiantes en el cromosoma u horario de clases, y k es un grupo de estudiantes, dc son los días de clases del grupo de estudiantes e i es un día de clases del grupo de estudiantes; y $RD1_{ki}$ son los conflictos de *timeslot* de un grupo de estudiantes k , en su día i de clases.

$$VRD2 = \sum_{k=1}^{du} \sum_{i=1}^{dc} RD2_{ki} \quad (2)$$

$VRD2$ es costo total de violación de restricción dura de los docentes universitarios, conflicto de *timeslot* entre los cursos de un docente universitario; du es total de docentes universitarios en el cromosoma u horario de clases, y k es un docente universitario, dc son los días de clases del del docente universitario k , e i es un día de clases del docente universitario k ; entonces para

$RD2_{ki}$ son los conflictos de *timeslot* de un docente universitario k , en su día i de clases.

$$VRD3 = \sum_{k=1}^{au} \sum_{i=1}^{dc} RD3_{ki} \quad (3)$$

$VRD3$ es costo total de violación de restricción dura de aulas universitarias, conflicto de *timeslot* entre las clases de un aula universitario; au es total de aulas universitarios en el cromosoma u horario de clases, y k es un aula universitaria, dc son los días de clases en el aula universitaria k , e i es un día de clases en el aula universitaria k ; entonces para $RD3_{ki}$ son los conflictos de *timeslot* en el aula universitaria k , en su día i clases.

$$VRD4 = \sum_{k=1}^{cu} \sum_{i=1}^{dc} RD4_{ki} \quad (4)$$

$VRD4$ es costo total de violación de restricción dura de cursos universitarios, cuando se agenda más de un *timeslot* de un curso universitario por día; cu es total de cursos universitarios en el cromosoma u horario de clases, y k es un curso universitario, dc son los días de clases en el curso universitario k , e i es un día de clases en el curso universitario k ; entonces para $RD4_{ki}$ son los conflictos de *timeslot* en el curso universitario k , en su día i de clases.

- **Parámetros del Algoritmo Genético**

Se recomienda un tamaño de población moderado, una tasa muy alta para la probabilidad de cruzamiento de 0.6 a 0.95, y una tasa muy baja para la probabilidad de mutación de 0.001 a 0.1 (De Jong, 1975); por otra parte se sugiere en base a evidencia empírica que una población pequeña converge prematuramente y una población muy grande tiene un mejor espacio de búsqueda, pero costos computacionales muy altos, por eso define el tamaño de población inicial entre l y $2l$, suficiente para tener éxito en el problema, donde l es la longitud del cromosoma (Alander, 1992). Por otra parte, cuando no se encuentra la solución óptima debe existir criterios de terminación para detener el funcionamiento del algoritmo genético y se sugiere tres condiciones de terminación cuando no encuentra la solución óptima y están definidas por el tiempo límite de funcionamiento, número máximo de iteraciones que coincide con el número de generaciones del algoritmo genético y la convergencia de la soluciones, es decir, cuando no existe cambios significativos entre la población inicial y los descendientes; si cumple estas condiciones de término se elige la mejor solución encontrada (Ahumada, 2014).

En base a los antecedentes y pruebas heurísticas en la construcciones del aplicativo web implementado la tecnología utilizando algoritmo genético y la longitud del cromosoma para este caso de estudio es de 818 (número de genes) se determinó los siguientes parámetros: Tamaño de población inicial es 150, probabilidad de cruzamiento es 0.95, probabilidad de mutación es 0.001, los criterios de terminación son: 1000 generaciones, 300

minutos y la convergencia en 20 generaciones del promedio del *fitness* de la población cuando no sufra cambios.

Procedimiento del algoritmo genético

El algoritmo genético inicializa con una población inicial, conformada por una cantidad entera de individuos, cada individuo representa una posible solución al problema de horarios de clases, en la primera generación del algoritmo genético la población inicial se genera de forma aleatoria, luego cada cromosoma pasa por un proceso de evaluación de *fitness*, para ver su aptitud con el ambiente, concatenando el *fitness* detallado al cromosoma, obteniendo el individuo; seguido se aplica los operadores genético: Operación de selección, utilizando el método de selección por ruleta a cada individuo de la población se le asigna un valor proporcional de la ruleta, para luego girar la ruleta y seleccionar al individuo para reproducirse; el operador genético de cruzamiento se realiza con cierta probabilidad, para recombinar los cromosomas padres obteniendo como resultado a dos cromosomas hijos o descendientes; el operador genético de mutación es aplicado a cada gen del cromosoma con cierta probabilidad muy baja, pero se realiza al momento de hacer el cruzamiento entre los genes de los cromosomas padres. Luego los cromosomas pasan por un proceso de evaluación de objetivo, para ver si la solución es óptima, si es de éxito termina el funcionamiento del algoritmo genético, caso contrario, guarda los individuos a la siguiente generación, mientras la selección de la población no se completa; completada la selección de población inicial se realiza el elitismo para mantener a los mejores individuos de la población inicial en la siguiente generación; luego

pasa por un criterio de terminación del algoritmo genético para decidir si continúa o se detiene. El diagrama del procedimiento del algoritmo genético en la presente investigación se muestra en la Figura 34.

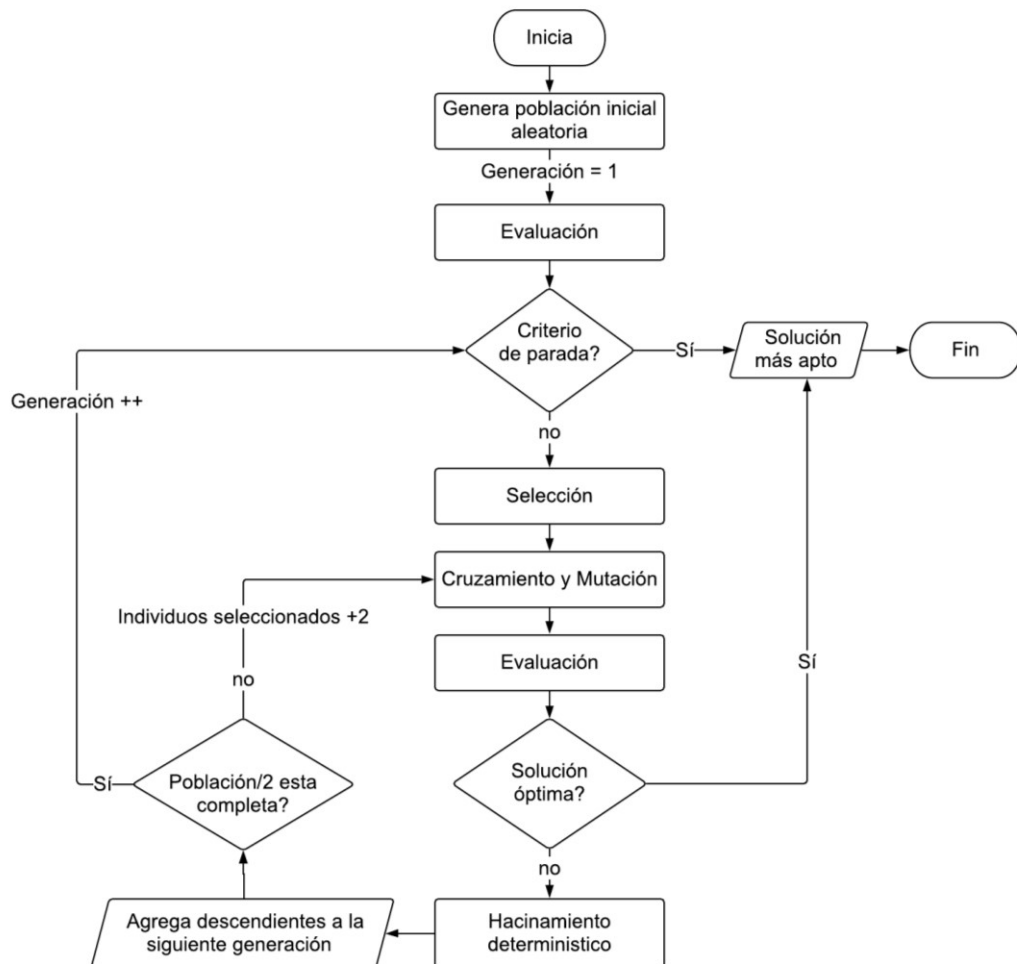


Figura 34. Diagrama UML del Algoritmo genético implementado en SEAH.

En la Figura 34 se observa el procedimiento del algoritmo genético implementado en SEAH en el cual el procedimiento se detalla en el siguiente orden:

a. Generación de la Población Inicial Aleatorio

El algoritmo genético ha iniciado generando la población inicial aleatoria de N individuos, donde los genes del cromosoma son construidos de forma aleatoria una vez terminado de construir el cromosoma se calcula el *fitness* detallado y se guarda en un archivo XML, generando un individuo de nombre de *individual* más el número correspondiente (*individual1.xml*) ocurriendo para toda la población y son comprimidos en un solo archivo correspondiente a la población inicial.

Para el obtener el cromosoma aleatorio, se debe concatenar los genes construidos de forma aleatoria. Un gen es construido a partir de los archivos cargados al sistema, inicia con la lectura del archivo de grupo de estudiantes de la Tabla 6; y para cada curso universitario del grupo de estudiantes se busca en el archivo de carga académica de la Tabla 4, se hace lectura del curso y docente universitario asignado, luego se consulta horas de clases de teoría y práctica; para la hora de teoría se le asigna aula universitaria de tipo teórico, día y *timeslot* de forma aleatoria de los archivos cargados al sistema correspondientes al

Tabla 5, Tabla 8 y Tabla 9, siempre y cuando el ítem curso universitario necesita aula de práctica es igual a cero, si es uno se le asigna aula de práctica descrita en el ítem de aula de práctica en la carga académica; si el aula práctica de la carga académica, si es si, se consulta aula si el curso universitario necesita sus todas sus clases en el aula de práctica, en caso en

ítem del total de clases es en el aula de práctica, todas las clases son asignados aula de práctica.

b. Selección

Se ha aplicado el método de selección por ruleta por su forma sencilla de desarrollo y presenta similitud de procedimiento con la frecuencia relativa acumulada en estadística, además en el funcionamiento del método el peor individuo puede ser seleccionado más de una vez para recombinar los cromosomas garantizando la variabilidad de datos y no convergiendo prematuramente.

En la presente investigación el algoritmo genético está diseñado para resolver problemas de minimización, encontrando las mejores soluciones con menor valor *fitness*, es decir, cromosomas con menos violaciones de restricciones duras; si el valor *fitness* es mayor, el método de ruleta asigna una mayor parte de la ruleta a los peores individuos, y por eso se necesita calcular el *fitness* ajustado, valor *fitness* numérico de tipo decimal, donde el individuo con menor *fitness* es asignado a una mayor proporción de la ruleta.

El método de selección por ruleta aplicado en el algoritmo genético de la presente investigación inicia con el cálculo de *fitness*, *fitness* duro, *fitness* puro, *fitness* estandarizado y *fitness* ajustado de la población inicial (Paso 1); luego se calcula el *fitness* acumulado y la suma del *fitness* proporcional o ruleta de la población inicial (Paso 2), luego se rueda la ruleta, y se selecciona los

individuos de la suma del *fitness* proporcional o ruleta hasta completar la población (Paso 3); los pasos del procedimiento se muestra en la Figura 35.

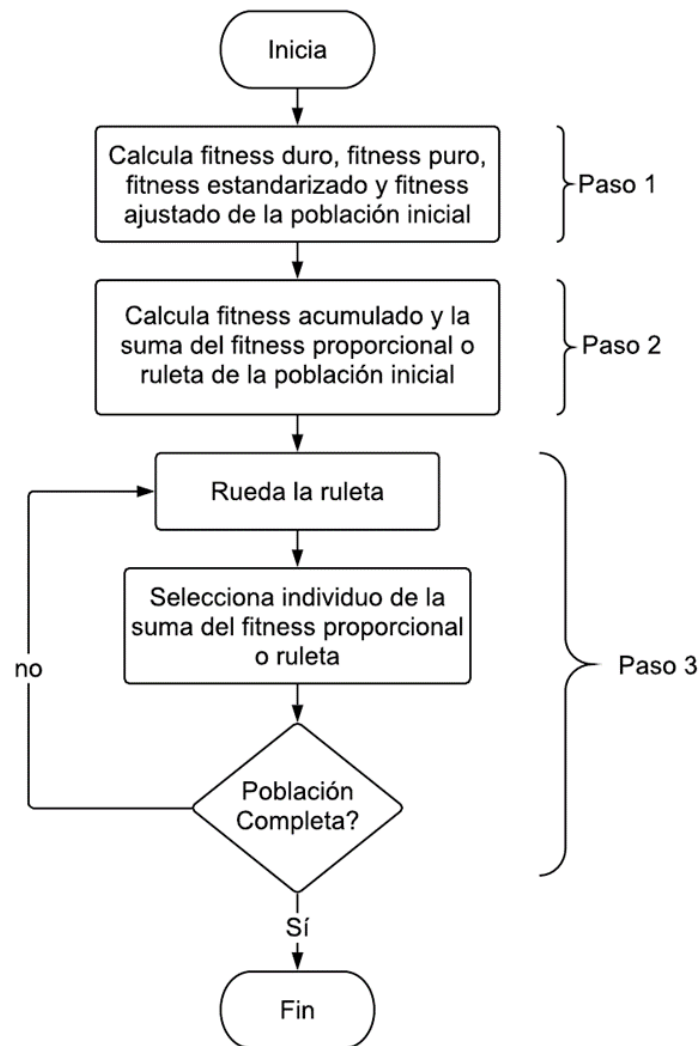


Figura 35. Diagrama UML de método selección por ruleta.

En la Figura 35 se observa tres pasos para seleccionar individuos y sean candidatos a reproducirse en el cual se detalla en el siguiente orden:

Paso 1

Se realiza la lectura del *fitness* detallado de cada individuo para calcular del *fitness* duro, el *fitness* puro de la Figura 1, el *fitness* estandarizado de la Figura 2; el *fitness* ajustado de la Figura 3 de la población inicial, este paso ocurre para cada individuo de la población inicial y de forma paralela, guardando los datos en listas.

$$\forall i \in P_g / FD_i = VRD1 + VRD2 + VRD3 + VRD4 \quad (5)$$

FD_i es el *fitness* duro del individuo i y se obtiene de la suma el *fitness* de restricciones duras en las ecuaciones (1), (2), (3) y (4), el individuo i ; Para todo individuo i que pertenece a la población inicial P de la generación g .

$$\forall FD_i \in P_g / FP_i = |0 - FD_i| \quad (6)$$

FP_i es el *fitness* puro del individuo i ; y se obtiene del valor absoluto de la diferencia entre el valor deseado cero y el FD_i , *fitness* duro del individuo i ; Para todo FD_i que pertenece a la población inicial P , de la generación g .

$$\forall FP_i \in P_g / FE_i = FP_i \quad (7)$$

FE_i es el *fitness* estandarizado del individuo i ; cómo es un problema de minimización se asigna el valor FP_i , el *fitness* puro del individuo i ; Para todo FP_i que pertenece a la población inicial P de la generación g .

$$\forall FE_i \in P_g / FA_i = \frac{1}{1 + FE_i} \quad (8)$$

FA_i es el *fitness* ajustado del individuo i ; y se obtiene de la división de la unidad entre la suma de la unidad y el FE_i , *fitness* estandarizado del individuo i ; Para todo FE_i que pertenece a la población inicial P de la generación g .

Paso 2

Se consulta la lista de *fitness* ajustado de la población inicial de una generación y se realiza la sumatoria total, luego se divide cada *fitness* ajustado de cada individuo con el sumatoria total para obtener el *fitness* acumulado, solo para el primer individuo el *fitness* acumulado se obtiene de la igualdad del *fitness* ajustado del primer individuo FA_1 como muestra la ecuación (9).

$$\forall i_1 \in P_g / FAC_1 = FA_1 \quad (9)$$

FAC_1 es el *fitness* acumulado del primer individuo de la población inicial P de la generación g , y para obtener el *fitness* acumulado de los siguientes individuos se aplica la ecuación (10).

$$\forall i \in P_g \wedge i > 1 / FAC_i = FAC_{i-1} + FA_i \quad (10)$$

FAC_i es el *fitness* acumulado a partir del segundo individuo i , es decir para todo individuo i mayor a uno hasta completar con la población P en la generación g .

Paso 3

Se genera un número aleatorio para la población inicial en el intervalo $[0,1]$ y recorre en la lista del *fitness* acumulado o ruleta, si el valor aleatorio es mayor *fitness* acumulado o ruleta del individuo i , se elige al siguiente individuo de la población inicial; esto significa que valor aleatorio está situada en la porción de la ruleta perteneciente al individuo $i + 1$. Este proceso se repite para cada individuo de la población inicial, como se muestra en la Figura 36.

Inputs: Población inicial, Lista fitnessAcumulado o Ruleta, Lista fitnessPuro
Outputs: Lista de númeroAleatorio, Lista de individuoSeleccionado, Lista de fitnessPadre

```

1. Begin
2. Foreach (individuo in Población inicial) do
3.   Número aleatorio ← Genera aleatorio // [0.0 – 1.0]
4.   Lista de númeroAleatorio ← Número aleatorio
5.   Posición del individuo ← 0 // La primera posición de lista es cero
6.   Foreach (itemAcumulado in Lista fitnessAcumulado o Ruleta) do
7.     bool Individuo seleccionado ← false
8.     If (Número aleatorio < itemAcumulado) then
9.       Lista de individuoSeleccionado ← Posición del individuo + 1
10.      Lista de fitnessPadre ← Lista fitnessPuro [Posición]
11.      Individuo seleccionado ← true
12.      If (individuo seleccionado is true) then
13.        break; // Salta del Foreach de la Lista fitnessAcumulado o Ruleta
14. End

```

Figura 36. Pseudocódigo de selección de individuos con el método ruleta.

c. Cruzamiento y Mutación

Los dos operadores genéticos ocurren de forma paralela, al realizar el cruzamiento uniforme con cierta probabilidad los dos cromosomas padres guiados por una variable de herencia de 0 o 1 para heredar los genes de los cromosomas padres a hijos en el mismo locus, pero antes de ser heredado el gen del cromosoma padre al cromosoma hijo puede sufrir una mutación de acuerdo a cierta probabilidad; este procedimiento se muestra en la Figura 37.

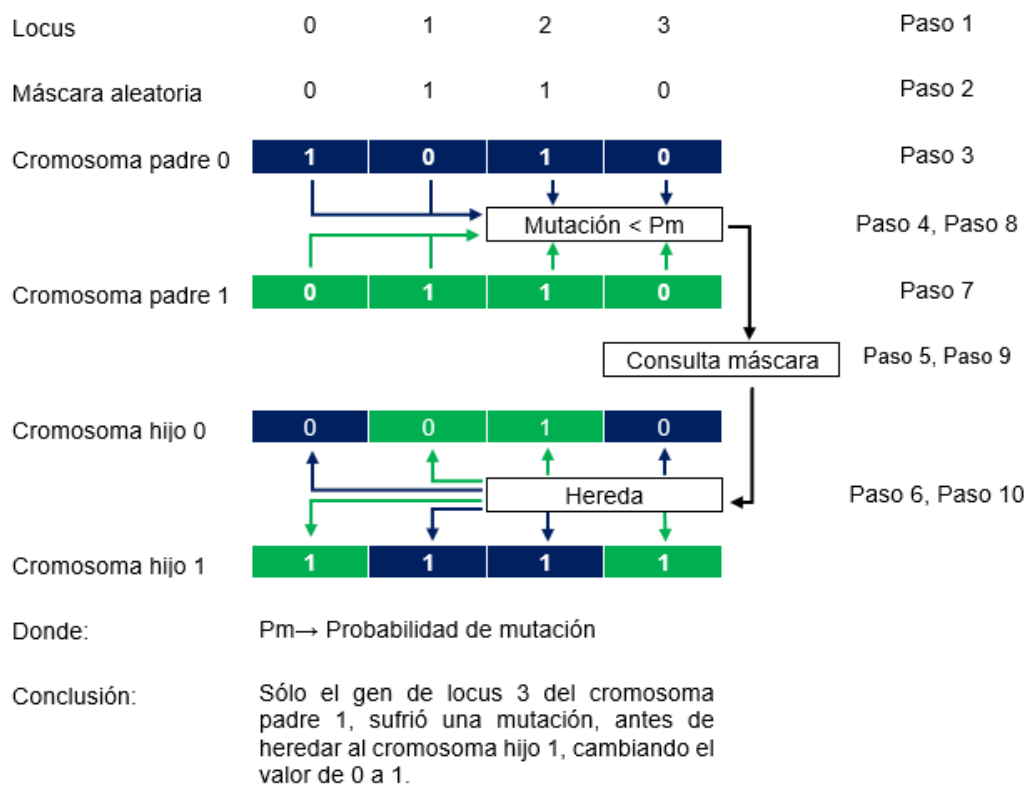


Figura 37. Proceso de cruzamiento y mutación en el algoritmo genético de SEAHC.

En la Figura 37 se observa que el operador genético de cruzamiento, se recorre cada gen del cromosoma padre 0, se identifica el locus del gen (Paso

1), luego se aplica una máscara de cruzamiento de forma aleatoria entre cero y uno (Paso 2); se selecciona el gen de locus correspondiente del cromosoma padre 0 (Paso 3); luego se aplica la probabilidad de mutación (paso 4); si esto resulta de éxito, el gen sufre una mutación, que será aplicado antes de ser heredado en el paso 6, caso contrario no sufre mutación; se consulta la máscara de cruzamiento (paso 5); si la máscara es cero, el gen del cromosoma padre 0 se hereda al cromosoma hijo 0, o si la máscara es uno, el gen del cromosoma padre 0 se hereda al cromosoma hijo 1 (paso 6), luego se selecciona el gen en el locus del cromosoma padre 1, y se incrementa el locus en uno para la siguiente iteración (paso 7), se aplica la probabilidad de mutación, si esto resulta de éxito sufre cambios, caso contrario no sufre cambios (paso 8), se consulta la máscara (paso 9); si la máscara de cruzamiento es cero, el gen del cromosoma padre 1 se hereda al cromosoma hijo 1, o si la máscara es uno, se hereda al cromosoma hijo 0 (paso 10); la iteración se repite hasta terminar el cruce uniforme de todos los genes cromosoma. El Pseudocódigo del operador de cruzamiento y mutación de forma paralela se muestra en la Figura 38.

```

Inputs: CromosomaPadre0, CromosomaPadre1
Outputs: CromosomaHijo0, CromosomaHijo1
1. Begin
2. Locus ← 0 //posición del gen en el cromosoma
3. Foreach (GenPadre0 in CromosomaPadre0) do
4.   Máscara ← genera aleatorio // 0 y 1
5.   Mutación ← genera aleatorio // [0.0 – 1.0]
6.   If (Mutación <= Probabilidad de Mutación) then
7.     If (Máscara = 0) then
8.       CromosomaHijo0 ← Muta GenPadre0 y agrega
9.     Else
10.      CromosomaHijo1 ← Muta GenPadre0 y agrega
11.   Else
12.     If (Máscara = 0) then
13.       CromosomaHijo0 ← agrega GenPadre0
14.     Else
15.       CromosomaHijo1 ← agrega GenPadre0
// selecciona el Gen en el locus del cromosomaPadre1
16. GenPadre1 ← selecciona Gen de cromosomaPadre1[locus]
17. Locus++//para el siguiente cruzamiento de genes
18. Mutación ← genera aleatorio// [0.0-1.0]
19. If (Mutación <= Probabilidad de Mutación) then
20.   If (Máscara = 0) then
21.     CromosomaHijo1 ← Muta GenPadre1 y agrega
22.   Else
23.     CromosomaHijo0 ← Muta GenPadre1 y agrega
24.   Else
25.     If (Máscara = 0) then
26.       CromosomaHijo1 ← agrega GenPadre1
27.     Else
28.       CromosomaHijo0 ← agrega GenPadre1
29. End

```

Figura 38. Pseudocódigo de los operadores genéticos de cruzamiento y mutación.

En la Figura 38 se observa que el operador genético de mutación modifica el gen del cromosoma hijo cambiando el valor de aula universitaria, día de clases y *timeslot* de forma aleatoria, sin antes consultar los archivos cargados en el sistema de la Tabla 5, Tabla 8 y Tabla 9 respectivamente; para el caso de aula universitaria, solo aplica el cambio para los de tipo teórico y se mantiene el mismo dato para las aulas de tipo práctica.

d. Evaluación

Para el proceso de evaluación de solución óptima pasan los dos cromosomas hijos y su *fitness* detallado, provenientes de los operadores genéticos de cruzamiento y mutación, se realiza el cálculo del *fitness* duro de la ecuación (5) a los cromosomas hijos, si uno de ellos es igual a cero, es una solución óptima como se muestra en la Figura 39, y pasa por un proceso de transformación de *timeslot* de dos horas a una hora y se guarda la solución culminando con el algoritmo genético; caso contrario si el valor de *fitness* duro de los cromosomas es mayor a cero, pasa al siguiente proceso de agregar descendientes a la siguiente generación.

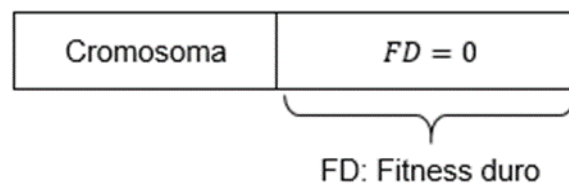


Figura 39. Representación de una solución óptima

e. Reemplazar población inicial para la Siguiete Generación

Se utiliza el método hacinamiento determinista (o *Deterministic Crowding*) para reemplazar población inicial de la siguiente generación. El método hacinamiento determinista es para realizar un concurso organizado contribuyendo a la variabilidad o diversidad entre los individuos padres e hijos para ser incluidos en la siguiente generación (Lozano, Herrera, y Ramón, 2008). (Toro, Aroba, & Ros, 2011) aplica el método hacinamiento determinista al diseño de su algoritmo genético para mejorar las posibilidades de obtener soluciones diversificadas.

Lozano et al. (2008) afirma:

El método de hacinamiento determinista consiste que dos cromosomas padres (P_i, P_j) mediante la recombinación produce dos cromosomas hijos (H_i, H_j) para que cada hijo reemplaza al padre más cercano si es más apto. Para realizar este procedimiento se define la distancia entre dos cromosomas que consiste el valor absoluto de la diferencia del fitness de los padres e hijos mediante la ecuación (11).

$$d(P, H) = |f(P) - f(H)| \quad (11)$$

Donde:

$f(P)$ es el *fitness* del cromosoma de P_i o P_j como también $f(H)$ es el *fitness* del cromosoma H_i o H_j , y $d(P, H)$ es la distancia entre el cromosoma padre e hijo.

El procedimiento del método de hacinamiento determinista se realiza mediante dos conjuntos de torneo: en el primer conjunto de torneo, Si la distancia de $d(P_i, H_i)$ más $d(P_j, H_j)$, es menor o igual que la distancia de $d(P_i, H_j)$ más $d(P_j, H_i)$, entonces se realiza el primer conjunto de torneo, si el $f(H_i)$ es mejor que $f(P_i)$ entonces reemplazar P_i con H_i , y si el $f(H_j)$ es mejor que $f(P_j)$ reemplazar P_j

con H_j ; de otro modo si de $d(P_i, H_i)$ más $d(P_j, H_j)$, no es menor o igual que la distancia de $d(P_i, H_j)$ más $d(P_j, H_i)$, se realiza si los siguientes torneos, si el $f(H_i)$ es mejor que el $f(P_j)$ entonces reemplazar P_j con H_i , y si el $f(H_j)$ es mejor que el $f(P_i)$ entonces reemplazar P_i con H_j ; como se muestra en la Figura 40.

1. **Si** $(d(P_i, H_i) + d(P_j, H_j)) \leq (d(P_i, H_j) + d(P_j, H_i))$ **entonces**
2. **Si** $f(H_i)$ es mejor que $f(P_i)$ **entonces** reemplazar P_i con H_i
3. **Si** $f(H_j)$ es mejor que $f(P_j)$ **entonces** reemplazar P_j con H_j
4. **Sino**
5. **Si** $f(H_i)$ es mejor que $f(P_j)$ **entonces** reemplazar P_j con H_i
6. **Si** $f(H_j)$ es mejor que $f(P_i)$ **entonces** reemplazar P_i con H_j

Figura 40. Pseudocódigo del método hacinamiento determinista (o *Deterministic Crowding*). Fuente: Lozano et al. (2008).

El algoritmo genético de la presente investigación el individuo más apto es el individuo con menor *fitness*, por lo tanto, el método de hacinamiento determinista es aplicado de la siguiente forma: los dos cromosomas padres y los dos cromosomas hijos con su respectivo *fitness* detallado ingresan a este método, para agregar descendientes a la siguiente generación, se calcula la distancia entre dos cromosomas en base al *fitness* duro, luego se realiza la competencia para reemplazar la población inicial de la siguiente generación guardando en una carpeta para ser utilizada en la siguiente generación como se muestra en la Figura 41.

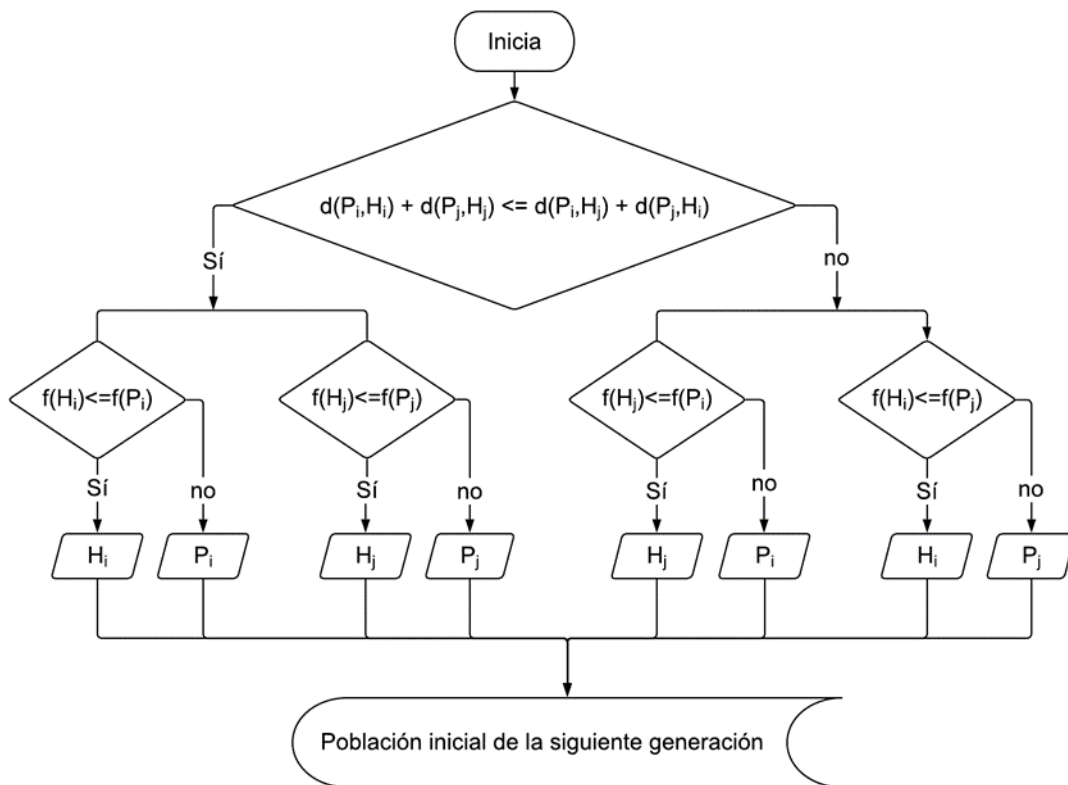


Figura 41. Diagrama UML del método hacinamiento determinista aplicado en la presente investigación.

f. Criterio de Terminación

Para continuar con la búsqueda de la solución óptima y pasar a la siguiente iteración se consulta el criterio de terminación definidas en los parámetros del algoritmo genético. En caso no cumpla con alguna de los tres criterios de terminación guarda la mejor solución obtenida hasta ese momento y vuelve a la selección de individuos para continuar el procedimiento del algoritmo genético en la siguiente iteración.

3.8.1.2. Sistema de Elaboración Automática de Horarios de Clases (SEAHC).

SEAHC fue construido con el IDE *Visual Studio Community* 2017 con el lenguaje de programación C# y se ha utilizado el *Team Services* de *Visual Studio online* como gestor de código fuente, el *framework* de aplicación web ASP.NET MVC 5 con .Net versión 4.5 y se ha utilizado el metalenguaje XML almacenar los datos del procedimiento del algoritmo genético implementado en SEAHC.

- **Tipo de software**

SEAHC de asignación de *timeslot* automático mediante el algoritmo genético de entorno web es clasificado como un sistema de procesamiento de transacciones desde un punto de vista empresarial en el sistema de información.

- **Arquitectura**

SEAHC está alojado en los servidores virtuales de la plataforma de la nube de Google (o *Google Cloud Platform*) en el cual se puede tener acceso desde cualquier ordenador con conexión a internet. El usuario accede a SEAHC y carga la información de entrada en archivos XML para elaborar los horarios de clases. El diagrama de despliegue de SEAHC se muestra en la Figura 42.

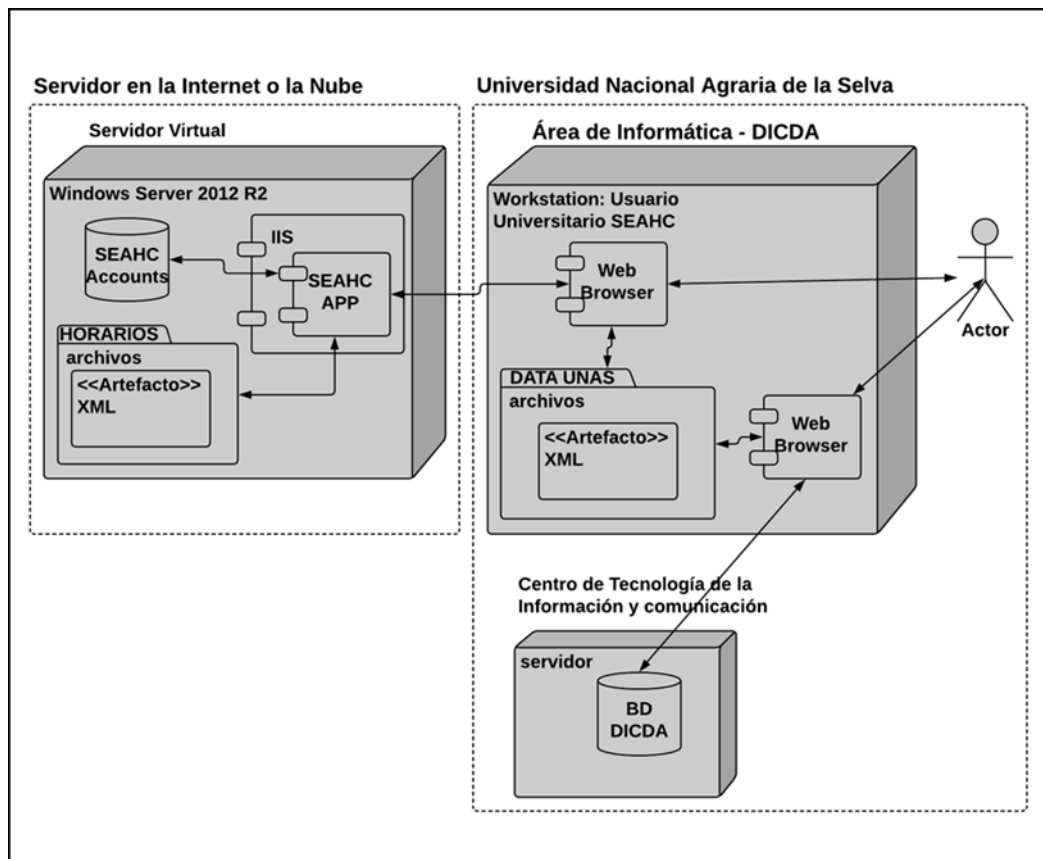


Figura 42. Diagrama de despliegue de SEAHC.

Modelo en Capas

SEAHC se ha construido en una arquitectura estructurada de N capas, utilizando seis capas tales como: Almacén de datos, lógica de acceso a datos, infraestructura y utilitarios, lógica de negocio, modelo, presentación; como se muestra en la Figura 43.

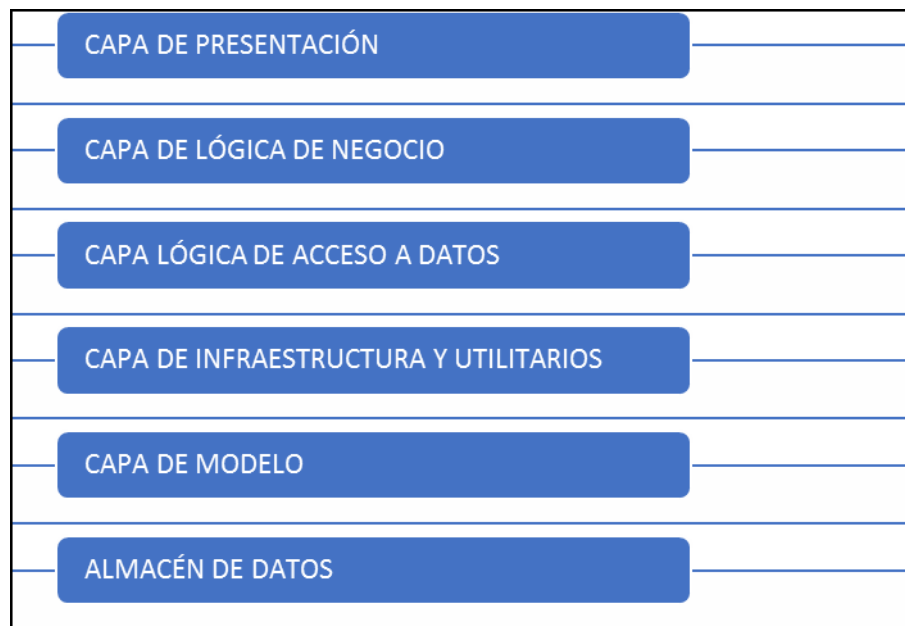


Figura 43. Modelo en capas de SEAHC.

En la Figura 43 se observa el modelo de m capas, en el cual se utilizó seis capas y son lo siguiente:

Almacén de Datos

Para almacenar datos de forma legible se ha utilizado XML en el disco duro virtual y cada individuo de la población se almacena en un archivo XML, como también el resumen de cada iteración del procedimiento del algoritmo genético y una colección del *fitness* de la población un resumen de la poblaciones o conjunto de horarios de clases que fueron evolucionando de y el horario de clases.

Capa Lógica de Negocio

En esta capa se ha construido los métodos de creación de archivos en el servidor virtual de SEAHC, los métodos serán de apoyo a la capa de la lógica de negocio y a la capa lógica de acceso a datos.

Lógica de acceso a datos

En esta capa se ha construido los métodos que serán usados en la capa lógica de negocio, los métodos planteados se comunican directamente con la base de datos de SEAHC haciendo las consultas a los archivos XML, por lo cual los métodos requieren usar LINQ10, LINQ para XML convierte un documento XML a una colección ordenada de nodos XmlNodeList.

Lógica de negocio

En la presente capa se ha desarrollado la clase de parte lógica, donde se usarán métodos para elaborar los horarios de clases, tales como el algoritmo genético y parte de ella, etc. La capa lógica de negocio realiza la petición a la capa de acceso a datos.

Modelo

La capa de modelo se divide en dos: entidades de negocio y de datos; las entidades de negocio son las clases modelo que la capa lógica de negocio necesita para elaborar los horarios de clases y las entidades de datos son las clases modelo que representa la base de datos de los archivos XML.

Presentación

Se ha utilizado el patrón de arquitectura de software Modelo Vista Controlador (MVC), para separar el código de lógica de aplicación y lógica de vista, desarrollando software más ordenado, ciclo de vida más adecuado, y de fácil mantenimiento.

En la vista de presentación se tiene las interfaces gráficas de usuario que realiza las peticiones a los servicios que brinda SEAHC, en la parte lógica de presentación se tiene al controlador que manipula los datos de acuerdo a los atributos de los datos de la vista modelo, la vista modelo responde al controlador y actualiza la vista de presentación. En la vista de presentación se diseñan las interfaces, y se ha utilizado los estilos de CSS, Bootstrap, JavaScript, funciones de angular.

- **Funcionalidades**

Carga de información de entrada y validación de formato de carga académica, grupo de estudiantes, aulas universitarias y las restricciones como se muestra en la Figura 44.

Figura 44. Vista de carga de información de entrada de SEAHC para elaborar los horarios de clases.

En la Figura 44 se observa la vista de carga de información de entrada de SEAHC en el cual solo admite archivos de tipo XML y los nombres correspondientes a cada ítem, luego pasa por un proceso de validación de formato de datos, pero no la confidencialidad del mismo, si es correcto pasa a la siguiente sección para registrar los datos de configuración de SEAHC.

En la configuración de SEHAC está incluido los parámetros del algoritmo genético, tamaño de población inicial, probabilidad de cruzamiento y de mutación, y por defecto trae seleccionado la única opción implementada el tipo de *timeslot* de dos horas, como también incluye el registro de los criterios de terminación del funcionamiento del algoritmo genético como el número de generaciones, el tiempo límite en minutos y numero de convergencia, como se muestra en la Figura 45.

CLASS TIMETABLING

Configuration parameters enter for class timetabling

PARAMETERS OF GENETIC ALGORITHM AND TIMESLOT TYPE

<p>POPULATION SIZE</p> <p>Número Entero, ej. 20; 20 horarios</p>	<p>TIMESLOT ASSIGNATION TYPE</p> <p>TIMESLOT FIJO: 2 HOURS</p>
<p>CROSSOVER PROBABILITY</p> <p>double number (0.0-0.1), ej. 0.7</p>	<p>MUTATION PROBABILITY</p> <p>double number (0.0-1.0), ej. 0.001</p>
<p>GENERATIONS NUMBER</p> <p>Número entero en minutos (20-200)</p>	<p>CONVERGENCE</p> <p>Número entero (10-50), ej. 20; 20 π</p>
<p>LIMIT TIME</p> <p>Número entero en minutos (20-200)</p>	

CLASS TIMETABLING

Figura 45. Vista de configuración de SEAHC para elaborar los horarios de clases utilizando el algoritmo genético.

Vista de la característica de los resultados del algoritmo genético si es o no, la solución óptima con la opción de descargar el archivo de tipo XML, como se muestra en la Figura 46.

Ver 10 Filas		Buscar:								
TIMETABLE	FITNESS	START	END	HH	MM	SS	GENERATION	INDIVIDUAL	GLOCATED	ACCTIONS
1	0	3/10/2019 6:01:41 PM	3/10/2019 7:42:14 PM	1	40	32	571	98	572	EXAMINE TIMETABLE

Mostrando 1 hasta 1 de 1 Filas

[Primero](#)
[Anterior](#)
[1](#)
[Siguiete](#)
[Ultimo](#)

Figura 46. Vista de característica de solución óptima u horario de clases terminado.

En la Figura 46 se observa el ítem EXAMINE al realizar clic muestra la vista del horario de clases para los diferentes criterios: grupo de estudiantes, curso, docente y aula universitario; como se muestra en la Figura 47.

CODE ENTER:		UNIVERSITY RESOURCE TYPE:			
<input type="text" value="ADMI01"/>		<input type="button" value="STUDENT GROUP"/>			
		<input type="button" value="VIEW"/>			
Ver 10	Filas				
Buscar: <input type="text"/>					
HOOR ^	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
06:00 - 07:00					
07:00 - 08:00	CURSO: DA080103 DOCENTE: 21069 AULA:C102	CURSO: DA080106 DOCENTE: 3750 AULA:A401	CURSO: DA080105 DOCENTE: 74972 AULA:C205	CURSO: DA080103 DOCENTE: 21069 AULA:E404	CURSO: DA080102 DOCENTE: 2650 AULA:E404
08:00 - 09:00	CURSO: DA080103 DOCENTE: 21069 AULA:C102	CURSO: DA080106 DOCENTE: 3750 AULA:A401	CURSO: DA080105 DOCENTE: 74972 AULA:C205	CURSO: DA080103 DOCENTE: 21069 AULA:E404	CURSO: DA080102 DOCENTE: 2650 AULA:E404
09:00 - 10:00				CURSO: DA080102 DOCENTE: 2650 AULA:D102	
10:00 - 11:00				CURSO: DA080102 DOCENTE: 2650 AULA:D102	
11:00 - 12:00			CURSO: DA080102 DOCENTE: 2650 AULA:E102		
12:00 - 13:00			CURSO: DA080102 DOCENTE: 2650 AULA:E102		
13:00 - 14:00					

Figura 47. Vista de horario de clases en el SEAHC para los diferentes criterios.

En la Figura 47 se observa el horario de clases del grupo de estudiantes del primer ciclo académico de administración de código ADMI01, en el cual está distribuido los horarios, código del curso, docente y aula de clases universitario. En la Figura 46 se observa el ítem de descarga TIMETABLE al realizar clic se descarga el horario de clases en archivo XML con *timeslot* de una hora como se muestra en la Figura 48.


```

1 <Chromosome>
2   <Gen>
3     <CodeStudentGroup>ADMI01</CodeStudentGroup>
4     <CodeAcademicCourse>DA080101</CodeAcademicCourse>
5     <CodeUniversityTeacher>315</CodeUniversityTeacher>
6     <CodeUniversityClassRoom>D101</CodeUniversityClassRoom>
7     <IdWeekDay>3</IdWeekDay>
8     <IdHour>119</IdHour>
9   </Gen>
10  <Gen>
11    <CodeStudentGroup>ADMI01</CodeStudentGroup>
12    <CodeAcademicCourse>DA080101</CodeAcademicCourse>
13    <CodeUniversityTeacher>315</CodeUniversityTeacher>
14    <CodeUniversityClassRoom>D101</CodeUniversityClassRoom>
15    <IdWeekDay>3</IdWeekDay>
16    <IdHour>120</IdHour>
17  </Gen>
18  <Gen>
19    <CodeStudentGroup>ADMI01</CodeStudentGroup>
20    <CodeAcademicCourse>DA080101</CodeAcademicCourse>
21    <CodeUniversityTeacher>315</CodeUniversityTeacher>
22    <CodeUniversityClassRoom>C205</CodeUniversityClassRoom>
23    <IdWeekDay>5</IdWeekDay>
24    <IdHour>114</IdHour>
25  </Gen>
26  <Gen>
27    <CodeStudentGroup>ADMI01</CodeStudentGroup>
28    <CodeAcademicCourse>DA080101</CodeAcademicCourse>
29    <CodeUniversityTeacher>315</CodeUniversityTeacher>
30    <CodeUniversityClassRoom>C205</CodeUniversityClassRoom>
31    <IdWeekDay>5</IdWeekDay>
32    <IdHour>115</IdHour>
33 </Gen>

```

Figura 48. Horario de clases en archivo XML terminado con la tecnología utilizando el algoritmo genético. Herramienta utilizada: Notepad++.

IV. RESULTADOS

4.1. Eficiencia del Proceso de Elaboración de Horarios de Clases

Se ha evaluado la eficiencia del proceso de elaboración de horarios de clases en una situación de forma tradicional y una situación utilizando el algoritmo genético basado en tres dimensiones: Tiempo de elaboración y esfuerzo del proceso como también el incumplimiento de restricciones duras del horario de clases terminado; y se ha realizado la comparación entre las dos tecnologías como se muestra en la Tabla 12.

Tabla 12. Eficiencia del proceso de elaboración de horarios de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.

Tecnología	Dimensión		
	Tiempo de elaboración	Esfuerzo	Incumplimiento de restricciones duras
Forma tradicional	100%	100%	0%
Utilizando el algoritmo genético	3.03%	3.54%	0%
Diferencia	96.97%	96.46%	0%

En la Tabla 12 se observa que el tiempo de elaboración del proceso con la tecnología de forma tradicional es 100% y con la tecnología utilizando el algoritmo genético es 3.03% con una diferencia de 96.97%; el esfuerzo del proceso en una situación de forma tradicional es 100% y con la tecnología utilizando el algoritmo genético es 3.54% con una diferencia de 96.46%; el incumplimiento de restricciones duras del horario de clases terminado con la tecnología de forma tradicional es 0% al igual que la tecnología utilizando el algoritmo genético de 0% y no existe diferencia alguna; la eficiencia del proceso utilizando las dos tecnologías se muestra en la Figura 49.

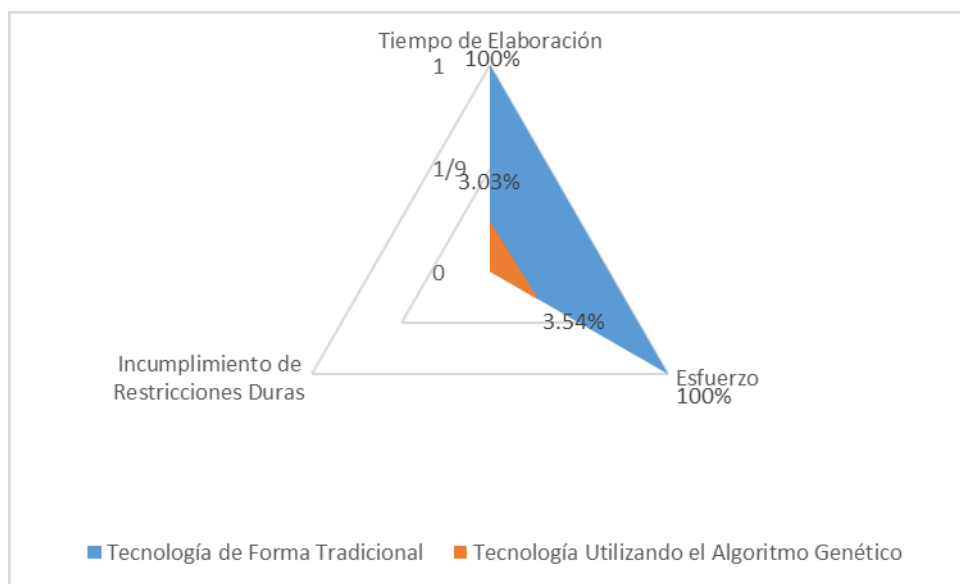


Figura 49. Eficiencia del proceso de elaboración de horarios de clases en la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.

4.2. Tiempo de Elaboración del Proceso de Elaboración de Horarios de Clases

Se ha evaluado el tiempo de elaboración del proceso de elaboración de horarios de clases en una situación de forma tradicional y una situación utilizando el algoritmo genético basado en el indicador número de días, y se ha realizado la comparación entre las dos tecnologías como se muestra en la Tabla 13.

Tabla 13. Tiempo de elaboración de horarios de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.

Tecnología	Dimensión	
	Tiempo de Elaboración	Valor Porcentual
Forma tradicional	33 días	100%
Utilizando el algoritmo genético	1 día	3.03%
Diferencia	32 días	96.97%

En la Tabla 13 se observa que el tiempo de elaboración del proceso con la tecnología de forma tradicional es 33 días que representa el 100% y con la tecnología utilizando el algoritmo genético es un día que representa el 3.03% respecto a la forma tradicional, con una diferencia de 32 días que representa el 96.97% en favor de la tecnología utilizando el algoritmo genético como se muestra en la Figura 50.

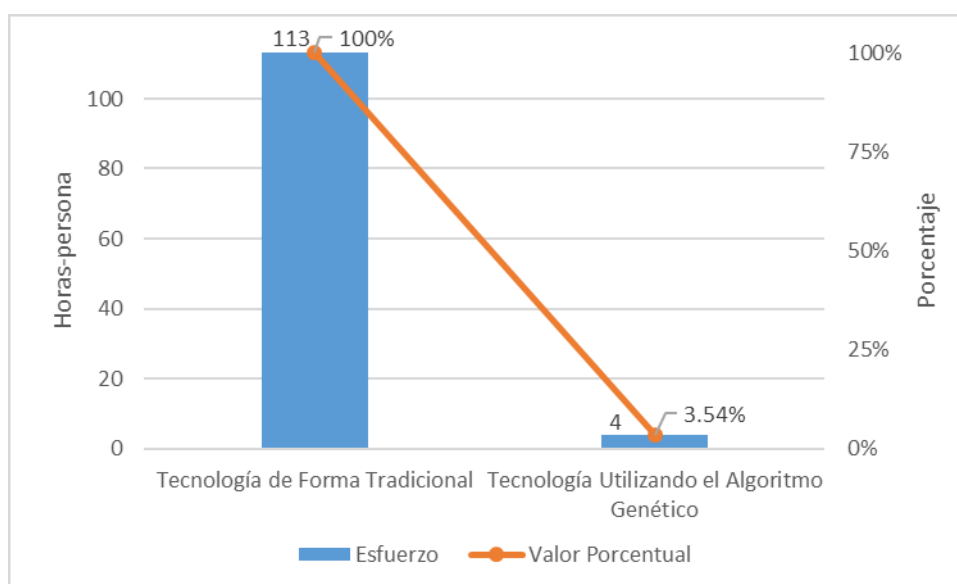


Figura 50. Tiempo de elaboración del proceso de elaboración de horarios de clases en la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.

4.2.1. Forma Tradicional

El tiempo de elaboración del proceso de elaboración de horarios de clases en una situación de forma tradicional es 33 días, iniciando el día 01 de febrero de 2018 y culminando el día 05 de marzo de 2018 como se muestra en la Tabla 14.

Tabla 14. Tiempo de elaboración del proceso de elaboración de horarios de clases de la UNAS en una situación de forma tradicional.

N°	Día, fecha y mes en el año 2018	Elaboración de horarios	Número de participantes
1	Jueves, 1 de febrero	Si	1
2	viernes, 2 de febrero	Si	1

3	sábado, 3 de febrero	no laborable	no laborable
4	domingo, 4 de febrero	no laborable	no laborable
5	lunes, 5 de febrero	Si	1
6	martes, 6 de febrero	Si	1
7	miércoles, 7 de febrero	Si	1
8	jueves, 8 de febrero	Si	1
9	viernes, 9 de febrero	Si	1
10	sábado, 10 de febrero	no laborable	No laborable
11	domingo, 11 de febrero	no laborable	No laborable
12	lunes, 12 de febrero	Si	1
13	martes, 13 de febrero	Si	1
14	miércoles, 14 de febrero	Si	1
15	jueves, 15 de febrero	Si	1
16	viernes, 16 de febrero	Si	1
17	sábado, 17 de febrero	no laborable	No laborable
18	domingo, 18 de febrero	no laborable	No laborable
19	lunes, 19 de febrero	Si	1
20	martes, 20 de febrero	Si	1
21	miércoles, 21 de febrero	Si	1
22	jueves, 22 de febrero	Si	1
23	viernes, 23 de febrero	Si	1
24	sábado, 24 de febrero	no laborable	no laborable
25	domingo, 25 de febrero	no laborable	no laborable
26	lunes, 26 de febrero	No se elaboró	0

27	martes, 27 de febrero	No se elaboró	0
28	miércoles, 28 de febrero	No se elaboró	0
29	jueves, 1 de marzo	Si	1
30	viernes, 2 de marzo	Si	2
31	sábado, 3 de marzo	no laborable	no laborable
32	domingo, 4 de marzo	no laborable	no laborable
33	lunes, 5 de marzo	Si	2

En la Tabla 14 se observa que 23 días son laborables entre lunes y viernes y 10 días no laborables entre sábado y domingos, de los 23 días laborables se dedicó 20 días al proceso de elaboración de horarios de clases con la participación de dos personas, para mayor detalle observar el Anexo A.

4.2.2. Utilizando el Algoritmo Genético

El tiempo de elaboración del proceso de elaboración de horarios de clases en situación utilizando el algoritmo genético es un día, realizado el día 10 de agosto de 2018 con la participación de una persona; para mayor detalle observar el Anexo E.

4.3. Esfuerzo del Proceso de Elaboración de Horarios de Clases

Se ha evaluado el esfuerzo del proceso de elaboración de horarios de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético basado en el indicador horas-persona, y se ha

realizado la comparación entre las dos tecnologías como se muestra en la Tabla 15.

Tabla 15. Esfuerzo del proceso de elaboración de horarios de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.

Tecnología	Dimensión Esfuerzo	Valor Porcentual
Forma tradicional	113 horas-persona	100 %
Utilizando el algoritmo genético	4 horas-persona	3.54 %
Diferencia	109 horas-persona	96.46 %

En la Tabla 15 se observa que el esfuerzo del proceso con la tecnología de forma tradicional es 113 horas-persona que representa el 100% y con la tecnología utilizando el algoritmo genético es cuatro horas-persona que representa el 3.54% respecto a la forma tradicional, con una diferencia de 109 horas-persona que representa el 96.46% en favor de la tecnología utilizando el algoritmo genético como se muestra en la Figura 51.

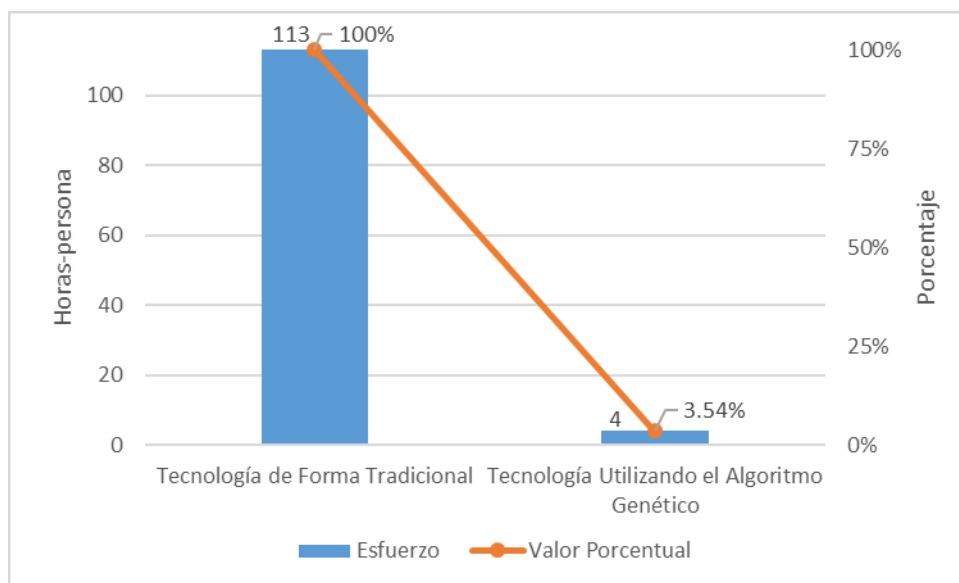


Figura 51. Esfuerzo del proceso de elaboración de horarios de clases en la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.

4.3.1. Forma Tradicional

Se ha tomado el esfuerzo de dos participantes responsables del proceso de elaboración de horarios de clases en una situación de forma tradicional basado en el indicador horas-persona.

Tabla 16. Esfuerzo del proceso de elaboración de horarios de clases de la UNAS en una situación de forma tradicional.

Día, fecha y mes de 2018	Responsable	Horas-persona
jueves, 1 de febrero	Pedro Saul Delgado Malpartida	5
viernes, 2 de febrero	Pedro Saul Delgado Malpartida	5
lunes, 5 de febrero	Pedro Saul Delgado Malpartida	5

martes, 6 de febrero	Pedro Saul Delgado Malpartida	5
miércoles, 7 de febrero	Pedro Saul Delgado Malpartida	5
jueves, 8 de febrero	Pedro Saul Delgado Malpartida	5
viernes, 9 de febrero	Pedro Saul Delgado Malpartida	5
lunes, 12 de febrero	Pedro Saul Delgado Malpartida	5
lunes, 12 de febrero	Pedro Saul Delgado Malpartida	3
martes, 13 de febrero	Pedro Saul Delgado Malpartida	5
martes, 13 de febrero	Pedro Saul Delgado Malpartida	3
miércoles, 14 de febrero	Pedro Saul Delgado Malpartida	5
miércoles, 14 de febrero	Pedro Saul Delgado Malpartida	3
jueves, 15 de febrero	Pedro Saul Delgado Malpartida	5
jueves, 15 de febrero	Pedro Saul Delgado Malpartida	3
viernes, 16 de febrero	Pedro Saul Delgado Malpartida	5
viernes, 16 de febrero	Pedro Saul Delgado Malpartida	3
lunes, 19 de febrero	Pedro Saul Delgado Malpartida	3
martes, 20 de febrero	Pedro Saul Delgado Malpartida	3
miércoles, 21 de febrero	Pedro Saul Delgado Malpartida	3
jueves, 22 de febrero	Pedro Saul Delgado Malpartida	3
viernes, 23 de febrero	Pedro Saul Delgado Malpartida	3
jueves, 1 de marzo	Pedro Saul Delgado Malpartida	5

viernes, 2 de marzo	Jorge Luis Jara Linares	5
viernes, 2 de marzo	Jorge Luis Jara Linares	3
viernes, 2 de marzo	Pedro Saul Delgado Malpartida	5
lunes, 5 de marzo	Jorge Luis Jara Linares	5

En la Tabla 16 se observa que el esfuerzo del proceso de elaboración de horarios de clases de forma tradicional es de 113 horas-persona con la participación de dos personas en el proceso y las horas laborales son de ocho horas diarias de lunes a viernes a excepción de los días feriados, de 08:00 a 17:00 horas, cinco y tres horas antes y después del almuerzo, se contempla una hora para almuerzo de 13:00 a 14:00 horas, para mayor detalle observar el Anexo A.

4.3.2. Utilizando el Algoritmo Genético

El esfuerzo del proceso de elaboración de horarios de clases en situación utilizando el algoritmo genético es de cuatro horas-persona con la participación de una persona, de las 8:00 a 12:00 horas en un día de la semana; para mayor detalle observar el Anexo E.

4.4. Incumplimiento de Restricciones Duras del Horario de Clases

Se ha evaluado el incumplimiento de restricciones duras del horario de clases terminado en una situación de forma tradicional y una situación utilizando el algoritmo genético basado en el indicador porcentaje de conflictos de *timeslot* de los criterios del horario de clases: Grupo de estudiantes, docente,

aula y curso universitario, y se ha realizado la comparación entre las dos tecnologías como se muestra en la Tabla 17.

Tabla 17. Incumplimiento de restricciones duras de horario de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.

Tecnología	Criterio de horario de clases				Me- dia
	Grupo de estudiantes	Docente universitario	Aula universitaria	Curso universitario	
Forma tradicional	0%	0%	0%	0%	0%
Utilizando el algoritmo genético	0%	0%	0%	0%	0%
Diferencia	0%	0%	0%	0%	0%

En la Tabla 17 se observa que la media del incumplimiento de restricciones duras con la tecnología de forma tradicional es 0% y con la tecnología utilizando el algoritmo genético es 0%, por lo tanto, no existe diferencia alguna como se muestra en la Figura 52.

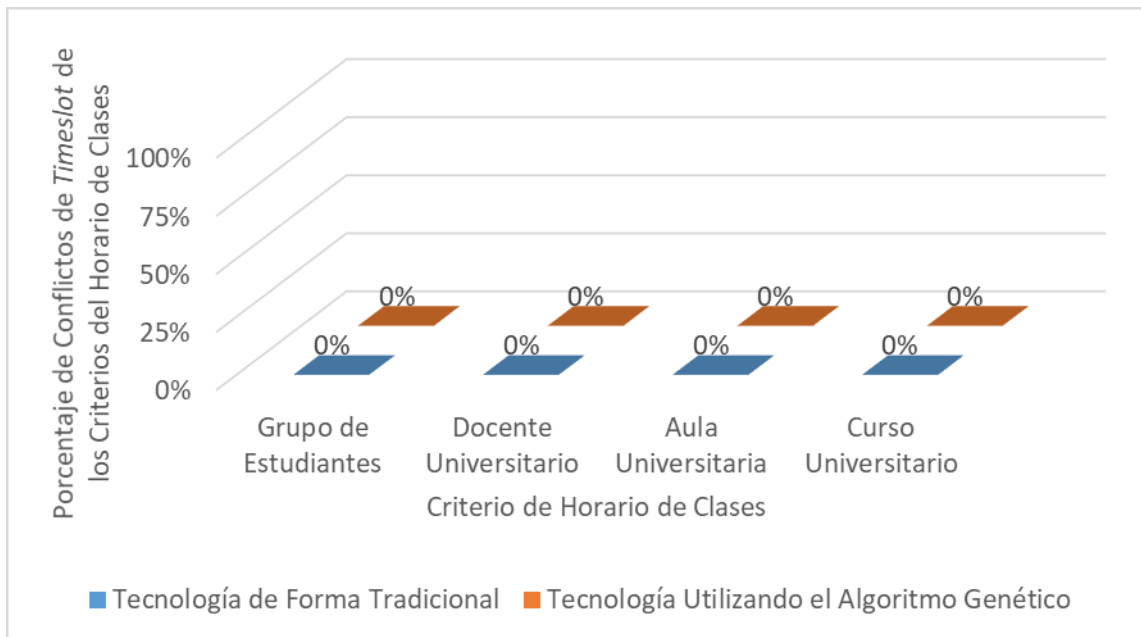


Figura 52. Incumplimiento de restricciones duras del horario de clases de la UNAS en una situación de forma tradicional y una situación utilizando el algoritmo genético.

En la Tabla 17 y Figura 52 el porcentaje de conflictos de *timeslot* para cada criterio de horario de clases se obtiene del producto de la cantidad de criterio de horario de clases, el número total de *timeslot* de horario de clases y el 100%; entre la cantidad de conflictos encontrados por cada criterio de horario de clases correspondiente, como se muestra en la ecuación (12).

$$PCT_i = \frac{C_i \times NT}{CCT_i} \times 100\% \quad (12)$$

PCT_i es el porcentaje de conflictos de *timeslot* de i para criterio de horario de clases; C_i es la cantidad del criterio de horario de clases, NT es el

número total de *timeslot* del horario de clases, CCT_i es la cantidad de conflictos de *timeslot* i para cada criterio de horario de clases.

4.4.1. Forma Tradicional

El porcentaje de conflictos de *timeslot* de los criterios del horario de clases de forma tradicional es 0%, es decir, para cada uno de ellos: Grupo de estudiantes, docente, aula y curso universitario; cuyo resultado se muestra en la *Tabla 18* y se obtiene con la ecuación (12).

Tabla 18. Porcentaje de conflicto de *timeslot* de los criterios de horario de clases de la UNAS en una situación de forma tradicional.

Criterio de horario de clases	Cantidad de	Número de <i>timeslot</i> en el horario	Número de Conflictos de <i>timeslot</i>	Porcentaje de conflictos de <i>timeslot</i>
Grupo de estudiantes	110	1382	0	0%
Docente universitario	240	1382	0	0%
Aula Universitario	101	1382	0	0%
Curso Universitario	629	1382	0	0%

4.4.2. Utilizando el Algoritmo Genético

El porcentaje de conflictos de *timeslot* de los criterios de horario de clases utilizando el algoritmo genético es de 0%, es decir, para cada uno de ellos:

Grupo de estudiantes, docente, aula y curso universitario; cuyo resultado se muestra en la Tabla 19 y se obtiene con la ecuación (12).

Tabla 19. Porcentaje de conflicto de *timeslot* de los criterios de horario de clases de la UNAS en una situación utilizando el algoritmo genético.

Criterio de horario de clases	Cantidad de	Número de <i>timeslot</i> en el horario	Número de Conflicto de <i>timeslot</i>	Porcentaje de conflictos de <i>timeslot</i>
Grupo de estudiantes	57	811	0	0%
Docente universitario	200	811	0	0%
Aula Universitario	101	811	0	0%
Curso Universitario	338	811	0	0%

V. DISCUSIÓN

A partir de los resultados obtenidos en la presente investigación se acepta la hipótesis planteada que si es posible mejorar la eficiencia del proceso utilizando el algoritmo genético, como también se expresa que los resultados del tiempo de elaboración y esfuerzo del proceso de forma tradicional se consideró el valor porcentual de 100% y respecto a ello obtener el resultado de proporcionalidad utilizando el algoritmo genético que determinó su efecto, mientras que el valor porcentual de los resultados de incumplimiento de restricciones duras se consideró de forma independiente porque se necesitaba conocer la viabilidad de las dos tecnología en el horario de clases terminado.

En el proceso en una situación de forma tradicional participaron dos personas que tenían asignado otras responsabilidades en el área de trabajo y muchas veces no participaron en el proceso, mientras en una situación utilizando el algoritmo genético participó una persona que se dedicó solo al proceso. Por otra parte la cantidad de información de entrada no se consideró en su totalidad en la tecnología utilizando el algoritmo genético a diferencia de la forma tradicional, por motivos de factibilidad porque la cantidad de información de entrada computacionalmente no sería de mayor variación en los resultados, otro motivo es que se presentan cargas académicas tentativas con modificaciones por el cual el acceso a datos actualizados es restringido y generan esfuerzo en

la construcción de información de entrada, por ello se ha trabajado con ciclos académicos oficiales al presente semestre, mientras en la forma tradicional las modificaciones de carga académica generan dificultad y esfuerzo en el proceso porque realizan modificaciones al avance del horario de clases.

El algoritmo genético presentado por Khonggamnerd e Innet (2009) que incluye restricciones duras y blandas encuentra la solución óptima en el menor número de generaciones que el algoritmo genético de la presente investigación que sólo incluye restricciones duras; un factor importante es que el primero cuenta con menor número de información de entrada por el cual el número de iteraciones de funcionamiento del algoritmo genético donde encuentra la solución óptima se expresa en 100% mientras que el segundo cuenta con mayor número de información de entrada, por lo tanto, el número de iteraciones de funcionamiento del algoritmo genético donde encuentra la solución óptima se expresa en 128.31% frente al primero.

El algoritmo genético presentado por Johnston (2000) encuentra la mejor solución aunque no la solución óptima en el menor número de generaciones que el algoritmo genético de la presente investigación que sí encuentra la solución óptima; se presume que el factor importante para encontrar la solución óptima es el método para reemplazar la población inicial, porque el primero utiliza elitismo a pesar que cuenta con menor complejidad (sin asignación de aulas y riesgo de conflicto de *timeslot* para los cursos) y menor información de entrada más aun cuando las condiciones tecnológicas de equipo

de cómputo son limitadas; mientras que el segundo utiliza el hacinamiento determinista y cuenta con mayor complejidad e información de entrada pero las condiciones tecnológicas son mayores.

VI. CONCLUSIONES

Se construyó una aplicación web que sirvió para implementar el algoritmo genético como metaheurística de una manera práctica para resolver el problema de elaboración de horarios de clases (o *University Course Timetabling Problem*) y se ha realizado su adecuación a un contexto de universidades con recursos compartidos, codificando la solución no binaria (números y letras) con *timeslot* fijo de dos horas y *fitness* entero solo para restricciones duras, realizando el operador genético de cruzamiento uniforme y mutación de forma paralela y se aplica el método hacinamiento determinista (o *Deterministic Crowding*) para reemplazar la población inicial, tomando la información de entradas del proceso y devolviendo como salida los horarios de clases para diferentes criterios tales como: Grupo de estudiantes, docente, aula y curso universitario; siendo posible de realizar el caso de estudio en la Universidad Nacional Agraria de la Selva en el semestre 2018-I.

En base al caso de estudio en un semestre académico se determinó que, si es posible mejorar la eficiencia del proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético, a pesar que solamente se trata de un caso de estudio cuyo resultado no puede generalizarse, pero queda demostrado de forma contundente por lo menos en este caso de estudio la mejora de la eficiencia del proceso con una

diferencia de 96.97% y 96.46% respecto al tiempo de elaboración y esfuerzo del proceso en favor de la tecnología utilizando el algoritmo genético frente a la forma tradicional, como también igualando con 0% el incumplimiento de restricciones duras del horario de clases terminado con la tecnología utilizando el algoritmo genético y de forma tradicional.

Se determinó que, si es posible minimizar el tiempo de elaboración en el proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético. El tiempo de elaboración del proceso en una situación de forma tradicional es 33 días que representa el 100% con la participación de dos personas y el tiempo de elaboración del proceso en una situación utilizando el algoritmo genético es un día que representa el 3.03% con la participación de una persona.

Se determinó que, si es posible minimizar el esfuerzo del proceso de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético. El esfuerzo del proceso en una situación de forma tradicional es 113 horas-persona que representa el 100% con la participación de dos personas y el esfuerzo del proceso en una situación utilizando el algoritmo genético es cuatro horas-persona que representa el 3.54% con la participación de una persona.

Se determinó que, si es posible evitar el incumplimiento de restricciones duras del horario de clases terminado en universidades con recursos compartidos utilizando el algoritmo genético, quedando demostrado la viabilidad del algoritmo genético utilizado para la elaboración de horario de clases al igual que la forma tradicional, al no tener conflictos de *timeslot* para los criterios del horario de clases: grupo de estudiantes, docente, aula y curso universitario.

Si bien es cierto, cualquier solución computacional tiende obviamente a minimizar tiempo y esfuerzo, sin embargo, no siempre es posible garantizar que una solución algorítmica sea igual o mejor a la solución heurística dada por el humano. A partir de esta reflexión se puede concluir que se logra una solución óptima capaz de resolver de manera práctica el problema de elaboración de horarios de clases en universidades con recursos compartidos utilizando el algoritmo genético.

Comparativamente entre el algoritmo genético de Khonggamnerd e Innet (2009) y el algoritmo genético de la presente investigación no se puede determinar cuál es más eficiente ya que son escenarios distintos por lo tanto no se puede concluir, sin embargo, se presume que el segundo es más eficiente que el primero porque sólo se expresa en un 28.31% adicional en el número de iteraciones de funcionamiento del algoritmo genético en el cual encuentra la solución óptima para un mayor número de información de entrada.

Para los casos estudiados el algoritmo genético de la presente investigación encuentra una solución óptima con *fitness* cero (sin violación de restricciones duras) en la elaboración de horarios de clases en universidades con recursos compartidos con mayor información de entrada y complejidad, por lo tanto se concluye que es más eficaz que el algoritmo genético implementado en Johnston (2000) para la asignación de carga académica de un instituto con menor información de entrada y complejidad que encuentra la mejor solución con *fitness* 31 (20 para restricciones duras y 11 para blandas).

RECOMENDACIONES

A las universidades con recursos compartidos que elaboran los horarios de clases de forma tradicional o de asignación manual, utilizar el algoritmo genético para mejorar la eficiencia del proceso de elaboración de horarios de clases, respecto al tiempo de elaboración y esfuerzo del proceso.

Se sugiere incorporar en el algoritmo genético una solución para la asignación de *timeslot* múltiple y el incumplimiento de restricciones blandas en el horario de clases, en futuras investigaciones donde las universidades tienen más claras sus modelos o políticas de horarios de clases, para obtener una solución de mejor calidad.

Se sugiere realizar investigaciones para explorar el algoritmo genético aplicado a otros campos de problemas de optimización que se pueda solucionar con métodos metaheurísticos, con la finalidad de aportar en la economía del proceso.

Se sugiere realizar investigaciones de otros algoritmos de búsqueda y optimización para determinar si es posible o no, solucionar el problema de elaboración de horarios de clases (o *university course timetabling problem*) en universidades con recursos compartidos y aportar evidencia empírica en un contexto más complejo.

BIBLIOGRAFÍA

- Ahumada, J. A. (2014). Generación de Horarios en INACAP Utilizando Algoritmos Genético. Universidad de Chile. Recuperado de <http://repositorio.uchile.cl/bitstream/handle/2250/131197/Generación-de-horarios-académicos-en-INACAP-utilizando-algoritmos-genéticos.pdf?sequence=1&isAllowed=y>
- Alander, J. T. (1992). On optimal population size of genetic algorithms. In *CompEuro 1992, Proceedings Computer Systems and Software Engineering* (pp. 65–70). Espoo, Finlandia. <https://doi.org/10.1109/CMPEUR.1992.218485>
- Arroyo, V. M. (2013). Modelo de un Algoritmo Genético con Selección Discriminatoria de Individuos bajo un Esquema de Ponderación de Probabilidades de Mutación. Universidad Católica San Pablo. Recuperado de http://www.ucsp.edu.pe/~ytupac/research/advisor/Tesis_VictorArroyoUCSP2013.pdf
- Babaei, H., Karimpour, J., y Hadidi, A. (2015). A survey of approaches for university course timetabling problem. *Computers & Industrial Engineering*, 86(July), 43–59. <https://doi.org/10.1016/j.cie.2014.11.010>
- Badoni, R. P., Gupta, D. K., y Mishra, P. (2014). A new hybrid algorithm for university course timetabling problem using events based on groupings of students. *COMPUTERS & INDUSTRIAL ENGINEERING*, 78(November), 12–25. <https://doi.org/10.1016/j.cie.2014.09.020>
- DeJong, K. A. (1975). Analysis of the Behavior of a Class of Genetic Adaptive

- Systems. Universidad de Michigan. Retrieved from <https://deepblue.lib.umich.edu/handle/2027.42/4507>
- Eiben, A. E., y Smith, J. E. (2015). *Introduction to Evolutionary Computing* (2a ed.). Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-44874-8>
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Publishing Company, Inc. Recuperado de http://www2.fiit.stuba.sk/~kvasnicka/Freebooks/Goldberg_Genetic_Algorithms_in_Search.pdf
- Hernández, R., Fernández, C., y Baptista, P. (2014). *Metodología de la Investigación* (6th ed.). DF, México: McGRAW-HILL.
- Jat, S. N. (2012). *Genetic Algorithms for University Course Timetabling Problems*. University of Leicester. Recuperado de <https://lra.le.ac.uk/bitstream/2381/10997/1/2012NaseemJatSNPhD.pdf>
- Jebari, K., y Madiafi, M. (2013). Selection Methods for Genetic Algorithms. *Int. J. Emerg. Sci*, 3(4), 333–344. Recuperado de <https://www.researchgate.net/publication/259461147>
- Johnston, J. D. (2000). *Aplicación de Algoritmo Genético para la Asignación de Carga Académica en Instituciones de Educación Superior*. Universidad Autónoma de Nuevo León. Recuperado de <http://eprints.uanl.mx/7603/1/1020130069.PDF>
- Junn, K. Y., Obit, J. H., y Alfred, R. (2018). The Study of Genetic Algorithm Approach to Solving University Course Timetabling Problem. *Computational Science and Technology*, 488, 454–463. <https://doi.org/10.1007/978-981->

10-8276-4_43

- Khonggamnerd, P., e Innet, S. (2009). On Improvement of Effectiveness in Automatic University Timetabling Arrangement with Applied Genetic Algorithm. In Computer Sciences and Convergence Information Technology (pp. 1266–1270). Seúl, Corea del Sur: IEEE. <https://doi.org/10.1109/ICCIT.2009.202>
- Lozano, M., Herrera, F., y Ramón, J. (2008). Replacement strategies to preserve useful diversity in steady-state genetic algorithms. *Information Sciences*, 178(23), 4421–4433. <https://doi.org/10.1016/j.ins.2008.07.031>
- Pose, M. G. (2000). Introducción a los Algoritmos Genéticos. Recuperado de <http://sabia.tic.udc>.
- Soliman, S. E., y Keshk, A. E. (2015). Memetic Algorithm For Solving University Course Timetabling Problem. *International Journal of Mechanical Engineering and Information Technology*. <https://doi.org/10.18535/ijmeit/v2i8.01>
- Toro, F. De, Aroba, J., y Ros, E. (2011). Applied Artificial Intelligence: An COMPUTER-AIDED DIAGNOSIS OF THE PAROXYSMAL ATRIAL FIBRILLATION : A FUZZY-EVOLUTIONARY APPROACH, (October 2014), 37–41. <https://doi.org/10.1080/08839514.2011.595281>
- Yang, S., y Jat, S. N. (2011). Genetic Algorithms With Guided and Local Search Strategies for University Course Timetabling. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(1), 93–106. <https://doi.org/10.1109/TSMCC.2010.2049200>

ANEXO

Anexo A. Tecnología de Forma Tradicional

Tabla 20. Resumen de la ficha de recolección de datos del proceso de elaboración de horarios de clases (EHC) y otras actividades.

Día, fecha y mes de 2018	Participante	Horario laboral	Actividad laboral	Horas-persona
jueves, 1 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
viernes, 2 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
sábado, 3 de febrero	No laborable.	-	-	-
domingo, 4 de febrero	No laborable.	-	-	-
lunes, 5 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
martes, 6 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
miércoles, 7 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3

jueves, 8 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
viernes, 9 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
sábado, 10 de febrero	No laborable.	-	-	-
domingo, 11 de febrero	No laborable.	-	-	-
lunes, 12 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida	14:00 a 17:00	PEHC	3
martes, 13 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	PEHC	3
miércoles, 14 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	PEHC	3
jueves, 15 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	PEHC	3
viernes, 16 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	PEHC	3

sábado, 17 de febrero	No laborable.	-	-	-
domingo, 18 de febrero	No laborable.	-	-	-
lunes, 19 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
martes, 20 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
miércoles, 21 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
jueves, 22 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
viernes, 23 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
sábado, 24 de febrero	No laborable.	-	-	-
domingo, 25 de febrero	No laborable.	-	-	-
lunes, 26 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	Otras	5
	Malpartida.	14:00 a 17:00	Otras	3

martes, 27 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	Otras	5
	Malpartida.	14:00 a 17:00	Otras	3
miércoles, 28 de febrero	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	Otras	5
	Malpartida.	14:00 a 17:00	Otras	3
jueves, 1 de marzo	Jorge Luis Jara	08:00 a 13:00	Otras	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
viernes, 2 de marzo	Jorge Luis Jara	08:00 a 13:00	PEHC	5
	Linares.	14:00 a 17:00	PEHC	3
	Pedro Saul Delgado	08:00 a 13:00	PEHC	5
	Malpartida.	14:00 a 17:00	Otras	3
sábado, 3 de marzo	No laborable.	-	-	-
domingo, 4 de marzo	No laborable.	-	-	-
lunes, 5 de marzo	Jorge Luis Jara	08:00 a 13:00	PEHC	5
	Linares.	14:00 a 17:00	Otras	3
	Pedro Saul Delgado	08:00 a 13:00	Otras	5
	Malpartida.	14:00 a 17:00	Otras	3

Anexo B. Tecnología de forma tradicional

Tabla 21. Número de cursos, *timeslot* y cursos sin *timeslot* por escuelas profesionales en el horario de clases de forma tradicional.

Escuela profesional	Número de cursos	Número de <i>timeslot</i>	Cursos sin <i>timeslot</i>
Agronomía	81	161	15
Ingeniería Mecánica eléctrica	19	40	3
Administración	75	111	16
Ingeniería en Conservación de Suelos y Agua	72	123	16
Economía	63	114	9
Ingeniería en Recursos Naturales Renovables	70	86	33
Ingeniería Forestal	89	145	21
Ingeniería Ambiental	77	123	21
Contabilidad	61	121	4
Ingeniería en Informática y Sistemas	54	114	1
Ingeniería en Industrias Alimentarias	64	125	5
Zootecnia	72	119	24
Total	797	1382	168

Anexo C. Tecnología utilizando el algoritmo genético

Tabla 22. Número de cursos, *timeslot* y cursos sin *timeslot* por escuelas profesionales en el horario de clases utilizando el algoritmo genético.

Escuela profesional	Número de cursos	Número de <i>timeslot</i>	Cursos sin <i>timeslot</i>
Agronomía	32	75	0
Ingeniería Mecánica eléctrica	13	32	0
Administración	34	78	0
Ingeniería en Conservación de Suelos y Agua	29	65	0
Economía	28	71	0
Ingeniería en Recursos Naturales Renovables	29	66	0
Ingeniería Forestal	32	73	0
Ingeniería Ambiental	27	62	0
Contabilidad	32	82	0
Ingeniería en Informática y Sistemas	23	68	0
Ingeniería en Industrias Alimentarias	34	81	0
Zootecnia	25	65	0
Total	338	818	0


Anexo D. Ficha de recolección de datos del proceso de elaboración de horarios de clases de forma tradicional.

RECOLECCIÓN DE DATOS DEL PROCESO DE ELABORACIÓN DE HORARIOS DE CLASES Y DATOS DE ENTRADA PARA LA INVESTIGACIÓN

- 1. TRABAJO DE INVESTIGACIÓN:**
ALGORITMO GENÉTICO PARA LA ELABORACIÓN DE HORARIOS DE CLASES EN UNIVERSIDADES CON RECURSOS COMPARTIDOS, CASO DE ESTUDIO: UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA.
- 2. TIPO DE TRABAJO:** TESIS DE PREGRADO
- 3. ÁREA RESPONSABLE DE ELABORAR LOS HORARIOS DE CLASES:**
ÁREA DE INFORMÁTICA DE LA DIRECCIÓN DE COORDINACIÓN Y DESARROLLO ACADÉMICO (DICDA).
- 4. RESPONSABLES DE LA INVESTIGACIÓN:**
RONALD EDUARDO IBARRA ZAPATA
NORRIS LEE BRAVO HERMITAÑO
- 5. CONSENTIMIENTO Y COMPROMISO:**
Consentimiento otorgado por la DICDA, a favor de los responsables de la investigación para la recolección de datos, mediante el uso de la ficha de observación. Conste también, el compromiso de brindar los datos de entrada de aulas universitarias y de carga académica para el semestre 2018 – I, única y exclusivamente para fines de investigación.
- 6. FICHA DE OBSERVACIÓN DEL PROCESO DE ELABORACIÓN DE HORARIOS DE CLASES (PEHC)**


Fecha	Responsable	Actividad laboral	Hora inicio y fin	Descripción
01/02/2018	Jorge Luis Sosa Linares	Otras	08:00-15:00	Otras
01/02/2018	Pedro Saul Delgado Malpartida	PEHC	08:00-15:00	Escuela Académica de Economía Ciclo académico: 1, 2, 3
01/02/2018	Jorge Luis Sosa Linares	Otras	14:00-17:00	Otras
01/02/2018	Pedro Saul Delgado Malpartida	Otras	14:00-17:00	Otras

Figura 53. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.




02/02/2018	Jorge Luis Sara Linares	Otras	08:00-13:00	Otras
02/02/2018	Pedro Saul Delgado Malpartida	PEHC	08:00-13:00	Escuela académica de Economía Ciclo académico: 3,1
02/02/2018	Jorge Luis Sara Linares	Otras	14:00-19:00	Otras
02/02/2018	Pedro Saul Delgado Malpartida	Otras	14:00-19:00	Otras
03/02/2018	—	—	—	Sábado - No laborable
04/02/2018	—	—	—	Domingo - No laborable
05/02/2018	Jorge Luis Sara Linares	Otras	08:00-13:00	Otras
05/02/2018	Pedro Saul Delgado Malpartida	PEHC	08:00-13:00	Escuela académica de Contabilidad Ciclo académico: 4,2,5
05/02/2018	Jorge Luis Sara Linares	Otras	14:00-19:00	Otras
05/02/2018	Pedro Saul Delgado Malpartida	Otras	14:00-19:00	Otras
06/02/2018	Jorge Luis Sara Linares	Otras	08:00-13:00	Otras
06/02/2018	Pedro Saul Delgado Malpartida	PEHC	08:00-13:00	Escuela académica de Contabilidad Ciclo académico: 3,1
06/02/2018	Jorge Luis Sara Linares	Otras	14:00-19:00	Otras

Figura 54. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.




06/02/2018	Pedro Saul Delgado Malpartida	Otras	14:00-15:00	Otras
07/02/2018	Jorge Luis Sosa Linares	Otras	08:00-15:00	Otras
07/02/2018	Pedro Saul Delgado Malpartida	PEHC	08:00-15:00	Escuela Académica de Maestría en Administración ciclo académico: 4, 3, 5
07/02/2018	Jorge Luis Sosa Linares	Otras	14:00-17:00	Otras
07/02/2018	Pedro Saul Delgado Malpartida	Otras	14:00-17:00	Otras
08/02/2018	Jorge Luis Sosa Linares	Otras	08:00-15:00	Otras
08/02/2018	Pedro Saul Delgado Malpartida	PEHC	08:00-15:00	Escuela Académica de Maestría en Administración ciclo académico: 3, 2, 1
08/02/2018	Jorge Luis Sosa Linares	Otras	11:00-14:00	Otras
08/02/2018	Pedro Saul Delgado Malpartida	Otras	14:00-17:00	Otras
09/02/2018	Jorge Luis Sosa Linares	Otras	08:00-15:00	Otras
09/02/2018	Pedro Saul Delgado Malpartida	PEHC	08:00-15:00	Escuela Académica de Maestría en Zootecnia ciclo académico: 4, 3, 5
09/02/2018	Jorge Luis Sosa Linares	Otras	14:00-17:00	Otras
09/02/2018	Pedro Saul Delgado Malpartida	Otras	14:00-17:00	Otras

Figura 55. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.




10/02/2018	—	—	—	Sábado - No laborable
11/02/2018	—	—	—	Domingo - No laborable
12/02/2018	Jorge Luis Sara Linares	Otras	08:00-13:00	Otras
12/02/2018	Pedro Saul Delgado Malpartida	PEHL	08:00-13:00	Escuela académica de Zootecnia Ciclo académico: 3,1
12/02/2018	Jorge Luis Sara Linares	Otras	14:00-19:00	Otras
12/02/2018	Pedro Saul Delgado Malpartida	PEHL	14:00-17:00	Escuela académica de Ingeniería en Industrias Alimentarias. Ciclo académico: 4, 3, 5
13/02/2018	Jorge Luis Sara Linares	Otras	08:00-13:00	Otras
13/02/2018	Pedro Saul Delgado Malpartida	PEHL	08:00-13:00	Escuela académica de Ingeniería en Industrias Alimentarias. Ciclo académico: 4, 3, 1
13/02/2018	Jorge Luis Sara Linares	Otras	14:00-17:00	Otras
13/02/2018	Pedro Saul Delgado Malpartida	PEHL	14:00-17:00	Escuela académica de Ingeniería ambiental. Ciclo académico: 4, 7
14/02/2018	Jorge Luis Sara Linares	Otras	08:00-13:00	Otras
14/02/2018	Pedro Saul Delgado Malpartida	PEHL	08:00-13:00	Escuela académica de Ingeniería ambiental. Ciclo académico: 5, 3, 1
14/02/2018	Jorge Luis Sara Linares	Otras	14:00-17:00	Otras

Figura 56. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.



14/02/2018	Pedro Saul Delgado Malpartida	PEHL	14:00-17:00	Escuela académica de Ingeniería Mecánica eléctrica Ciclo académico: 1,5
15/02/2018	Serge Luis Sara Linares	Otras	08:00-13:00	Otras
15/02/2018	Pedro Saul Delgado Malpartida	PEHL	08:00-13:00	Escuela académica de Agronomía Ciclo académico: 9,7
15/02/2018	Serge Luis Sara Linares	Otras	14:00-17:00	Otras
15/02/2018	Pedro Saul Delgado Malpartida	PEHL	14:00-17:00	Escuela académica de Agronomía Ciclo académico: 9
16/02/2018	Serge Luis Sara Linares	Otras	08:00-13:00	Otras
16/02/2018	Pedro Saul Delgado Malpartida	PEHL	08:00-13:00	Escuela académica de Agronomía Ciclo académico: 3,1
16/02/2018	Serge Luis Sara Linares	Otras	14:00-17:00	Otras
16/02/2018	Pedro Saul Delgado Malpartida	PEHL	14:00-17:00	Escuela académica de Ingeniería Forestal Ciclo académico: 9,7
17/02/2018	—	—	—	Viernes - No laborable
18/02/2018	—	—	—	Domingo - No laborable
19/02/2018	Serge Luis Sara Linares	Otras	08:00-13:00	Otras
19/02/2018	Pedro Saul Delgado Malpartida	Otras	08:00-13:00	Otras

Figura 57. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.




19/02/2018	Jorge Luis Sara Linares	Otras	11:00-17:00	Otras
19/02/2018	Pedro Saul Delgado Malpartida	PEHL	14:00-17:00	Escuela Académica de Ingeniería Forestal: 7, 5 Ciclo académico
20/02/2018	Jorge Luis Sara Linares	Otras	08:00-13:00	Otras
20/02/2018	Pedro Saul Delgado Malpartida	Otras	08:00-13:00	Otras
20/02/2018	Jorge Luis Sara Linares	Otras	14:00-17:00	Otras
20/02/2018	Pedro Saul Delgado Malpartida	PEHL	14:00-17:00	Escuela Académica de Ingeniería Forestal: 3, 1 Ciclo académico
21/02/2018	Jorge Luis Sara Linares	Otras	08:00-13:00	Otras
21/02/2018	Pedro Saul Delgado Malpartida	Otras	08:00-13:00	Otras
21/02/2018	Jorge Luis Sara Linares	Otras	14:00-17:00	Otras
21/02/2018	Pedro Saul Delgado Malpartida	PEHL	14:00-17:00	Escuela Académica de Ingeniería Recurso Natural Renovables Ciclo académico: 9, 7
22/02/2018	Jorge Luis Sara Linares	Otras	08:00-13:00	Otras
22/02/2018	Pedro Saul Delgado Malpartida	Otras	08:00-13:00	Otras
22/02/2018	Jorge Luis Sara Linares	Otras	14:00-17:00	Otras

Figura 58. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.




22/02/2018	Pedro Saúl Dolgado Malpartida	PEHL	14:00-17:00	Escuela académica de Ingeniería Recursos Naturales Renovable ciclo académico: 7, 5
23/02/2018	Serge Luis Sara Linares	Otras	08:00-13:00	Otras
23/02/2018	Pedro Saúl Dolgado Malpartida	Otras	08:00-13:00	Otras
23/02/2018	Serge Luis Sara Linares	Otras	14:00-17:00	Otras
23/02/2018	Pedro Saúl Dolgado Malpartida	PEHL	14:00-17:00	Escuela académica de Ingeniería Recursos Naturales Renovable ciclo académico: 7, 2, 1
24/02/2018	—	—	—	Sábado - No laborable
25/02/2018	—	—	—	Domingo - No laborable
26/02/2018	Serge Luis Sara Linares	Otras	08:00-13:00	Otras
26/02/2018	Pedro Saúl Dolgado Malpartida	Otras	08:00-13:00	Otras
26/02/2018	Serge Luis Sara Linares	Otras	14:00-17:00	Otras
26/02/2018	Pedro Saúl Dolgado Malpartida	Otras	14:00-17:00	Otras
27/02/2018	Serge Luis Sara Linares	Otras	08:00-13:00	Otras
27/02/2018	Pedro Saúl Dolgado Malpartida	Otras	08:00-13:00	Otras

Figura 59. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.




27/02/2018	Serge Luis Sara Linares	Otras	14:00-17:00	Otras
27/02/2018	Pedro Saul Delgado Malpartida	Otras	14:00-17:00	Otras
28/02/2018	Serge Luis Sara Linares	Otras	08:00-13:00	Otras
28/02/2018	Pedro Saul Delgado Malpartida	Otras	08:00-13:00	Otras
28/02/2018	Serge Luis Sara Linares	Otras	14:00-17:00	Otras
28/02/2018	Pedro Saul Delgado Malpartida	Otras	14:00-17:00	Otras
01/03/2018	Serge Luis Sara Linares	Otras	08:00-13:00	Otras
01/03/2018	Pedro Saul Delgado Malpartida	PEML	08:00-13:00	Escuela académica: varios. Resolución de conflicto de horarios Ciclo académico: varios
01/03/2018	Serge Luis Sara Linares	Otras	14:00-17:00	Otras
01/03/2018	Pedro Saul Delgado Malpartida	Otras	14:00-17:00	Otras
02/03/2018	Serge Luis Sara Linares	PEML	08:00-13:00	Escuela académica de Ingeniería en Informática y Sistemas Ciclo académico: 4, 7, 5
02/03/2018	Pedro Saul Delgado Malpartida	PEML	08:00-13:00	Escuela académica: varios. Resolución de conflicto de horarios Ciclo académico: varios
02/03/2018	Serge Luis Sara Linares	PEML	14:00-17:00	Escuela académica de Ingeniería en Informática y Sistemas Ciclo académico: 3, 7


Figura 60. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.



02/03/2018	Pedro Saut Diegado Malpartida	Otros	14:00-17:00	Otros
03/03/2018	-	-	-	Sábado - No laborable
04/03/2018	-	-	-	Domingo - No laborable
05/03/2018	Serge Luis Sara Linares	PEHC	08:00-15:00	Exalta académica: Varías Feria y cambio de horario Club académico: Varías
05/03/2018	Pedro Saut Diegado Malpartida	Otros	08:00-15:00	Otros
 				
 				




RONALD EDUARDO IBARRA ZAPATA
RESPONSABLE




NORRIS LEE BRAVO HERMITAÑO
TESISTA

Figura 61. Ficha de observación del proceso de elaboración de horarios de clases en una situación de forma tradicional.

Anexo E. Ficha de recolección de datos del proceso de elaboración de horarios de clases utilizando el algoritmo genético.



Universidad
Nacional Agraria
de la Selva



Facultad de Ingeniería
en Informática y
Sistemas


Departamento Académico
de Ciencias en Informática y
Sistemas

Laboratorio de
Ingeniería de Software
(INSOFT)


RECOLECCIÓN DE DATOS DEL PROCESO DE ELABORACIÓN DE HORARIOS DE CLASES UTILIZANDO EL ALGORITMO GENÉTICO

- 1. TRABAJO DE INVESTIGACIÓN:**
ALGORITMO GENÉTICO PARA LA ELABORACIÓN DE HORARIOS DE CLASES EN UNIVERSIDADES CON RECURSOS COMPARTIDOS, CASO DE ESTUDIO: UNIVERSIDAD NACIONAL AGRARIA DE LA SELVA.
- 2. TIPO DE TRABAJO:** TESIS DE PREGRADO
- 3. ÁREA RESPONSABLE DE ELABORAR LOS HORARIOS DE CLASES:**
LABORATORIO DE INGENIERÍA DE SOFTWARE.
- 4. RESPONSABLES DE LA INVESTIGACIÓN:**
RONALD EDUARDO IBARRA ZAPATA
NORRIS LEE BRAVO HERMITAÑO
- 5. FICHA DE OBSERVACIÓN DEL PROCESO DE ELABORACIÓN DE HORARIOS DE CLASES (PEHC)**

Fecha	Responsable	Actividad laboral	Hora inicio y fin	Descripción
10/08/2018	Norris Lee Bravo Hermitaño	PEHC	08:00-12:00	Construcción de Laptops, Registro de Avances Pós al Algoritmo genético



RONALD EDUARDO IBARRA ZAPATA
RESPONSABLE



NORRIS LEE BRAVO HERMITAÑO
TESISTA

Figura 62. Ficha de observación del proceso de elaboración de horarios de clases en una situación utilizando el algoritmo genético.

GLOSARIO

Aleatoriedad. Proceso cuyo resultado no es previsible sino la consecuencia de la intervención del azar.

Algoritmo. Conjunto reescrito y organizado de instrucciones sistemáticas que describe un proceso a seguir y realizar cálculos para dar solución a un problema específico

Alojamiento. Servicio que provee a los usuarios de internet un sistema para almacenar información o archivos informáticos a través de la web.

Archivo. En la informática también llamado fichero informático, es un conjunto de bits que son almacenados en un dispositivo, llamado así son equivalentes a los archivos escritos, pero de forma digital.

Archivo Comprimido. Resultado de tratar un archivo o fichero informático con un programa específico para comprimir, es decir reducir el peso para reducir espacio de almacenamiento en un dispositivo sin perder los datos originales.

Bootstrap. Biblioteca multiplataforma para diseño de sitios y aplicaciones web que se adapta a todo tipo de dispositivo y pantalla.

Confidencialidad. Propiedad de la información garantizando el acceso al usuario autorizado

Costo computacional. El uso de los recursos del ordenador como espacio de almacenamiento, uso de RAM y CPU.

CPU. Unidad central de procesamiento (por sus siglas en inglés, *Central Processing Unit*), es el hardware dentro de un ordenador u otros dispositivos programables, que interpreta las instrucciones de un programa informático mediante la realización de las operaciones básicas aritméticas, lógicas y de entrada/salida del sistema.

CSS. Hojas de estilo en cascada (por sus siglas en inglés, *Cascading style sheets*), es un lenguaje de diseño gráfico utilizado para definir y crear la presentación de documentos estructurado escrito en un lenguaje de marcado.

Filtro. En informática se refiere a un programa diseñado para controlar la muestra de un contenido específico, con la finalidad de restringir el acceso a la información.

Formato. Forma, tamaño y modo de presentación de un objeto.

Hardware. En la informática se refiere a las partes físicas y tangibles de un sistema informático de componentes eléctricos, electrónicos y mecánicos

Heurística. Que tiene como significado hallar e inventar; considerado el arte de inventar por parte del humano, con la intención de procurar estrategias, métodos,

criterios, que permitan resolver problemas a través de la creatividad, pensamiento divergente o lateral.

JavaScript. Es un lenguaje de programación interpretado que se utiliza principalmente del lado del cliente para mejorar la interfaz del usuario y páginas web dinámicas.

Mecanismo. Forma de realizar una actividad, una función o un proceso.

Metaheurística. Es un método heurístico para resolver un problema computacional general, usando los parámetros dados por el usuario o humano sobre procedimientos genéricos y abstractos de una forma que se espera eficiente.

Módulo. En la informática (programación) una porción de programa de un ordenador.

Notepad++. Editor de texto sin formato y forma simple semejante al bloc de notas, de código de fuente libre.

Optimización. Proceso que consiste en encontrar los mínimos y máximos de una variable o función.

Parámetros. Dato imprescindible y orientativo para lograr evaluar o valorar una determinada situación.

Proceso. Una secuencia de pasos dispuesto con algún tipo de lógica que se enfoca en lograr algún resultado específico.

Pseudocódigo. Lenguaje de alto nivel para llevar de un algoritmo a un programa informático.

RAM. Memoria de acceso aleatorio (por sus siglas en inglés, *Random Access Memory*), se utiliza como memoria de trabajo de computadoras y otros dispositivos para el sistema operativo, los programas y donde se ejecuta la mayor parte del software.

Red Corporativa. Permite la conexión de todas las localizaciones de la institución de una forma permanente, privada, segura y fiable.

Servidor. Una aplicación en funcionamiento capaz de atender las peticiones de un cliente y devolver una respuesta en concordancia.

Software. Soporte lógico de un sistema informático, conjunto de programas

Tecnología. Conjunto de instrumentos, recursos técnicos o procedimiento empleados en un determinado campo o sector.

UML. Lenguaje unificado de modelado (por sus siglas en inglés, *Unified Modeling Language*), lenguaje de modelado de sistemas de software, de forma gráfica para visualizar, especificar, construir y documentar un sistema.

XML. Lenguaje de marcado extensible (por sus siglas en inglés, *eXtensible Markup Language*), es un meta – lenguaje utilizado para almacenar datos de forma legible.