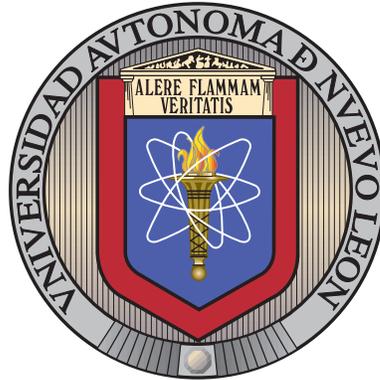


UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO



GENERANDO TRAYECTORIAS DE APRENDIZAJE
QUE CONSIDEREN DATOS BIOMÉTRICOS

POR

JUAN PABLO ROSAS BALDAZO

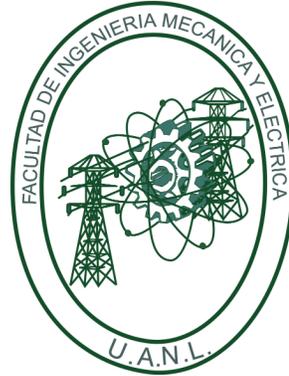
COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE
DOCTOR EN INGENIERÍA DE SISTEMAS

JUNIO 2023

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO



GENERANDO TRAYECTORIAS DE APRENDIZAJE
QUE CONSIDEREN DATOS BIOMÉTRICOS

POR

JUAN PABLO ROSAS BALDAZO

COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE
DOCTOR EN INGENIERÍA DE SISTEMAS

JUNIO 2023

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
Facultad de Ingeniería Mecánica y Eléctrica
Posgrado

Los miembros del Comité de Evaluación de Tesis recomendamos que la Tesis “Generando Trayectorias de aprendizaje que consideren datos biométricos”, realizada por el estudiante Juan Pablo Rosas Baldazo, con número de matrícula 1266973, sea aceptada para su defensa como requisito parcial para obtener el grado de Doctorado en Ingeniería de Sistemas.

El Comité de Evaluación de Tesis

Dr. Romeo Sánchez Nigenda
Director

Dra. Yasmín Á. Ríos Solís
Co-director

Dra. Cristina Maya Padrón
Revisor

Dra. Edith Lucero Ozuna Espinosa
Revisor

Dra. Iris Abril Martínez Salazar
Revisor

Dra. María Angélica Salazar Aguilar
Revisor

Dra. Sara Elena Garza Villarreal
Revisor

Vo.Bo.



Dr. Simón Martínez Martínez
Subdirector de Estudios de Posgrado

Institución 190001

Programa 557620

Acta Núm. 353

Ciudad Universitaria, a 13 de septiembre de 2024.

Para mi familia ...

*Soy un gato, un ser extremadamente sensible a los más sutiles cambios en la mente
o el alma del mundo. Y, naturalmente, necesito dormir más que el resto.*

Natsume Soseki - Soy un gato

ÍNDICE GENERAL

| | |
|---|-------------|
| Agradecimientos | XIII |
| Resumen | XIV |
| 1. Introducción | 1 |
| 1.1. Descripción del problema | 4 |
| 1.2. Hipótesis | 6 |
| 1.3. Objetivos | 7 |
| 1.4. Solución propuesta | 7 |
| 1.5. Estructura de la tesis | 8 |
| 2. Antecedentes | 10 |
| 2.1. Planeación y secuenciamiento de tareas | 10 |
| 2.2. Programación Lineal | 13 |
| 2.3. Meta-heurísticas | 14 |
| 3. Revisión de literatura | 16 |

| | |
|--|-----------|
| 3.1. Trayectorias de aprendizaje personalizadas como método para afrontar la deserción educativa | 16 |
| 3.2. Estrés | 18 |
| 3.2.1. Cómo medir el estrés | 19 |
| 4. Problema de Planificación Educativa considerando el estrés | 22 |
| 4.1. La capacidad de aprendizaje del estudiante afecta la duración de las actividades | 24 |
| 4.2. El estrés deteriora la duración de las actividades | 25 |
| 4.3. Modelo matemático | 28 |
| 4.3.1. Evaluación del modelo matemático | 31 |
| 4.4. VNS-GRASP | 41 |
| 4.4.1. Calibración de parámetros | 45 |
| 4.4.2. Evaluación del <i>GRASP – VNS</i> y los modelos matemáticos | 50 |
| 4.5. Comparación de resultados | 51 |
| 5. Conclusiones | 61 |
| 5.1. Trabajo a futuro | 62 |
| 5.1.1. Problema | 62 |
| 5.1.2. Instancias y datos | 63 |
| 5.1.3. Método de solución | 64 |
| A. Conceptos | 65 |

| | |
|---|----|
| A.1. Ramificación y Acotamiento | 65 |
|---|----|

ÍNDICE DE FIGURAS

| | |
|--|----|
| 1.1. Ejemplo de estructura de una retícula educativa. | 2 |
| 2.1. Ejemplo del diagrama de Gantt de la programación de tres trabajos por máquina. | 11 |
| 3.1. Ejemplo de la relación entre los niveles de estrés y el desempeño de una persona. | 21 |
| 4.1. Ejemplo de estructura de una retícula educativa. | 23 |
| 4.2. Modelos de duración de actividades para una sola actividad con una duración promedio de 10. | 27 |
| 4.3. Ejemplo de restricciones de precedencia. | 30 |
| 4.4. Rendimiento de los modelos EPP-SL y EPP-PSL respecto al EPP. El eje vertical es el makespan; en el eje horizontal, variamos la puntuación mínima deseada por el alumno. | 32 |
| 4.5. Impacto de cantidad de actividades de precedencia en las instancias en el makespan de las soluciones de los modelos MILP EPP-SL y EPP-PSL con respecto al EPP. El eje de las ordenadas es el makespan; en el eje de las abscisas, variamos la cantidad de actividades de precedencia. | 33 |

4.6. Impacto de cantidad del coeficiente intelectual de los estudiantes en el makespan de las soluciones de los modelos EPP-SL y EPP-PSL con respecto al EPP. El eje vertical es el makespan; en el eje horizontal, variamos el nivel de coeficiente intelectual. 34

4.7. Tiempos de ejecución de GUROBI para los modelos EPP-PSL y EPP-SL. El eje de las ordenadas representa el tiempo en segundos y el eje de las abscisas el número de actividades predecesoras requeridas. . . 35

4.8. Tiempos de ejecución de GUROBI para los modelos EPP-PSL y EPP-SL. El eje de las ordenadas representa el tiempo en segundos y el eje de las abscisas es la puntuación mínima deseada por el alumno. . . . 36

4.9. Tiempos de ejecución de GUROBI para los modelos EPP-PSL y EPP-SL. El eje de las ordenadas representa el tiempo en segundos y el eje de las abscisas es la tasa de aprendizaje del alumno. 36

4.10. Tiempos de ejecución de GUROBI para los modelos EPP-PSL y EPP-SL. El eje de las ordenadas representa el tiempo en segundos y el ejes de las abscisas es el foco de estrés al final o al principio del periodo. . 37

4.11. Comparación entre el modelo EPP, el modelo EPP-PSL y el modelo EPP→EPP-PSL que utiliza el mismo conjunto de actividades obtenidas por el EPP. 38

4.12. Similitud entre las soluciones obtenidas resolviendo la misma instancia pero con distintos niveles de estrés con el modelo EPP-PSL. Los paneles de la izquierda muestran la similitud en función de las actividades que conforman las soluciones. Los paneles de la derecha corresponden a la similitud entre la posición de las actividades. 40

4.13. Ejemplo de los vecindarios de la búsqueda local del VNS. 45

| | |
|---|----|
| 4.14. El eje de abscisas representa la cantidad de elementos en la <i>LRC</i> , mientras que el eje de ordenadas corresponde al makespan de las soluciones en el gráfico a) y al tiempo computacional en el gráfico b) de acuerdo a la cantidad de iteraciones del algoritmo <i>GRASP</i> | 46 |
| 4.15. El eje de abscisas representa el índice de aprendizaje, mientras que el eje de ordenadas corresponde al makespan de las soluciones en el gráfico a) y al tiempo computacional en el gráfico b) de acuerdo a la cantidad de iteraciones del algoritmo <i>VNS</i> | 47 |
| 4.16. Porcentaje de mejora del <i>VNS</i> comparado con la solución <i>GRASP</i> en relación el número de requisitos de precedencia de las actividades. | 48 |
| 4.17. Porcentaje de mejora del <i>VNS</i> comparado con la solución <i>GRASP</i> en relación a la calificación mínima esperada $k_{\text{mín}}$ | 49 |
| 4.18. Porcentaje de mejora del <i>VNS</i> comparado con la solución <i>GRASP</i> en relación con índice de aprendizaje de los estudiantes. | 50 |
| 4.19. Calidad en términos de makespan (eje de ordenadas) del modelo exacto comparado con el <i>GRASP – VNS</i> en relación al número de los requisitos de la actividad de precedencia (eje de abscisas). | 52 |
| 4.20. Calidad en términos de makespan (eje de ordenadas) del modelo exacto comparado con el <i>GRASP – VNS</i> en relación con el $k_{\text{mín}}$ esperado. | 53 |
| 4.21. Calidad en términos de makespan (eje de ordenadas) del modelo exacto comparado con el <i>GRASP – VNS</i> en relación con el índice de aprendizaje. | 54 |
| 4.22. Tiempo de ejecución (eje de ordenadas) en segundos de los modelos exactos en comparación con <i>GRASP – VNS</i> con respecto al número de los requisitos de precedencia de las actividades (eje de abscisas). . | 55 |

| | |
|---|----|
| 4.23. Tiempo de ejecución de los modelos exactos en comparación con la meta-heurística <i>GRASP-VNS</i> con respecto a la calificación mínima esperada. | 55 |
| 4.24. Tiempo de ejecución de los modelos exactos en comparación el <i>GRASP-VNS</i> con respecto al foco del estrés, al inicio o al final del periodo. . | 56 |
| 4.25. Tiempo de ejecución de los modelos exactos en comparación con el <i>GRASP-VNS</i> con respecto al índice de aprendizaje de los estudiantes. | 57 |
| 4.26. La compensación de Pareto para una estudiante con carga de estrés tiende a ser al final del periodo académico y su tasa de aprendizaje es alta. | 58 |

ÍNDICE DE TABLAS

| | |
|--|----|
| 1.1. EPP : trayectoria de aprendizaje estimada para todos los alumnos; $EPP(S)$: trayectoria de aprendizaje estimada aplicada a un alumno en concreto con estrés creciente al final del semestre; $EPP - SL(S)$: trayectoria de aprendizaje específica para un alumno teniendo en cuenta sus niveles de estrés y sus capacidades de aprendizaje. | 5 |
| 4.1. Resultados promedio para el $GRASP - VNS$ y el $EPP - PSL$ | 60 |

AGRADECIMIENTOS

A mi esposa Citlali Olvera, alguien que conocí en este camino y con quien se que siempre puedo contar para hablar, aclarar mis dudas y apoyarme cuando todo sea difícil.

A mi familia que siempre me ha demostrado su apoyo incondicional en cada proyecto. A la Dra. Yasmín Ríos por tenerme tanta paciencia y siempre tener palabras de aliento en cada paso que dimos y cada decisión que tomamos.

Al Dr. Romeo Sánchez por su consejos y guía en la realización de este trabajo.

A mis compañeros y profesores con los que pasé muchas horas de trabajo y diversión.

Y a mis revisores, por enriquecer con sus comentarios este trabajo.

RESUMEN

Juan Pablo Rosas Baldazo.

Candidato para obtener el grado de Doctor en Ingeniería de Sistemas .

Universidad Autónoma de Nuevo León.

Facultad de Ingeniería Mecánica y Eléctrica.

Título del estudio: GENERANDO TRAYECTORIAS DE APRENDIZAJE QUE CONSIDEREN DATOS BIOMÉTRICOS.

Número de páginas: 75.

OBJETIVOS Y MÉTODO DE ESTUDIO: El objetivo de este trabajo fue investigar el impacto en el desempeño académico de características subjetivas personales como el estrés y el coeficiente intelectual. De igual manera validar si mediante la generación de trayectorias de aprendizaje personalizadas que consideren estos factores podríamos mejorar dicho desempeño.

En particular, nos enfocamos en desarrollar e implementar un modelo de optimización que refleje el impacto del estrés y el coeficiente intelectual de un estudiante en el tiempo total de una trayectoria de aprendizaje personalizada con actividades de duración variable. Además, de un algoritmo meta-heurístico como una opción más rápida para generar soluciones a este problema.

CONTRIBUCIONES Y CONCLUSIONES: Propusimos dos modelos matemáticos y un algoritmo meta-heurístico *GRASP – VNS* para calcular las trayectorias de aprendizaje personalizadas considerando factores subjetivos como el estrés y el índice de aprendizaje de los estudiantes.

Nuestros resultados mostraron la importancia de tener en cuenta dichas características a la hora de crear trayectorias de aprendizaje. En particular, mostramos

que conforme aumenta el índice de aprendizaje del alumno, el tiempo total promedio de las soluciones disminuye debido a la mayor capacidad de aprendizaje, donde el tiempo total es la suma de las duraciones de las actividades pertenecientes a la ruta de aprendizaje. Además, nuestros resultados muestran que a pesar de que el conjunto de actividades que conforman las soluciones sea similar, la secuencia es completamente diferente debido a los niveles de estrés de cada estudiante y por lo tanto el tiempo total de la solución también lo es.

Por último, demostramos que el algoritmo *GRASP – VNS* propuesto genera soluciones de buena calidad en un menor tiempo que los métodos exactos.

Firma del asesor:



Dr. Romeo Sánchez Nigenda

CAPÍTULO 1

INTRODUCCIÓN

Uno de los principales retos en el área educativa que afrontan países de América Latina es el abandono escolar. Según el INEGI (2019) en México existen algunos estados como Quintana Roo, Chihuahua, Sonora donde el porcentaje de abandono supera el 15% en educación media superior y superior. Varios estudios indican que las principales razones para que un alumno abandone los estudios son la falta de adecuación entre las expectativas de los alumnos y el programa de estudios, las capacidades del alumno y la duración y los requisitos de las actividades (Meeuwisse *et al.*, 2010; Vossensteyn *et al.*, 2015; Lowis y Castley, 2008; Wang *et al.*, 2020).

Existen diferentes técnicas para combatir este tipo de situaciones, una de ellas son las *trayectorias de aprendizaje*, las cuales, consideran la selección y secuenciación de actividades educativas necesarias para alcanzar un objetivo académico. Trabajos recientes en investigación educativa destacan la importancia de la educación basada en este tipo de técnicas (De Smet *et al.*, 2016; Sanchez Nigenda *et al.*, 2018; Nabizada *et al.*, 2020; Smet *et al.*, 2016a) lo que sugiere podrían ser una buena herramienta para ayudar a prevenir el abandono escolar y el bajo rendimiento en el aprendizaje (Smith y Throne, 2007; Dynarski *et al.*, 2008; Takači *et al.*, 2017).

La generación de *trayectorias de aprendizaje* implica la secuenciación de actividades educativas para personalizar la enseñanza académica. Por tanto, el problema

de planificación educativa (*EPP* por sus siglas en inglés de *Educational Planning Problem*) presentado por Sanchez Nigenda *et al.* (2018) considera la generación de trayectorias de aprendizaje personalizadas mediante la selección de las actividades educativas necesarias para satisfacer las calificaciones esperadas minimizando la duración total (*makespan*) de las trayectorias de aprendizaje. Usualmente en problemas de secuenciación de actividades el *makespan* se define como el tiempo de fin de las operaciones asociadas a un trabajo. Sin embargo, para este trabajo, *makespan* será la suma de la duración de las actividades presentes en la trayectoria de aprendizaje y no el tiempo de finalización de la última actividad, esto debido a que en nuestras trayectorias de aprendizaje pueden existir tiempos muertos, lo que extendería el *makespan* de la solución. En la Figura 1.1 podemos ver una descripción gráfica de una retícula educativa para describir el *EPP*. Las asignaturas tienen asociados temas, subtemas y actividades que los estudiantes deben completar. Cada actividad tiene una duración prevista y un valor que representa la contribución a la calificación final.

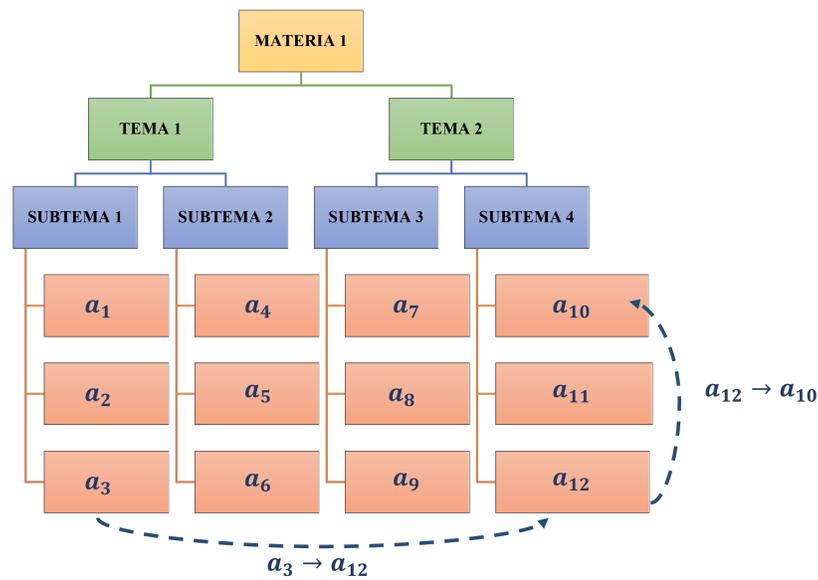


FIGURA 1.1: Ejemplo de estructura de una retícula educativa.

Muchas áreas de investigación proporcionan soluciones a *EPP* utilizando una amplia variedad de técnicas para modelar la información de los planes de estudios y automatizar la generación de trayectorias de aprendizaje (Klašnja-Milićević *et al.*,

2011; Tzouveli *et al.*, 2008a; Garrido *et al.*, 2016; Jeong *et al.*, 2012; Özpolat y Akar, 2009; Benmesbah *et al.*, 2021). Un punto a resaltar es que la eficacia con la que los estudiantes realizan las actividades de las trayectorias de aprendizaje depende de muchos factores. Sin embargo, hasta donde sabemos, no existen soluciones que tengan en cuenta el estrés y los efectos del aprendizaje (es decir, el aumento del rendimiento en el aprendizaje a medida que los estudiantes completan sus planes educativos) para personalizar sus trayectorias de aprendizaje.

Para este trabajo decidimos incorporar el estrés en la generación de trayectorias de aprendizaje dado que el estrés es uno de los principales factores que afectan negativamente al rendimiento académico de los estudiantes (Wolf *et al.*, 2015; Vogel *et al.*, 2018). Además, durante la reciente pandemia mundial de COVID-19, los niveles de estrés y ansiedad de los estudiantes aumentaron. En parte debido al miedo al contagio y a las sensibles pérdidas de familiares (Haikalis *et al.*, 2022), así como a la transición al aprendizaje a distancia con nuevas tecnologías (Durbas *et al.*, 2021), entre otras causas (Hamaideh *et al.*, 2022). Aunque la importancia de incorporar el estrés en las trayectorias de aprendizaje es evidente, no es tan obvio cómo transformar su naturaleza cualitativa en un factor cuantitativo que permita medir su impacto en las actividades de aprendizaje de un plan educativo (Wang, 2006).

Parte de las aportaciones de este trabajo consisten en proponer los modelos adecuados para incorporar los efectos del estrés y el aprendizaje en la secuenciación de las actividades educativas de los alumnos; a este problema lo denominamos *EPP – SL* (Por sus siglas en inglés de *Educational Planning Problem with Stress and Learning*). El *EPP – SL* es un problema combinatorio discreto NP-difícil ya que puede reducirse al problema de programación de una sola máquina con restricciones de precedencia, fechas de entrega y minimización del makespan (Lawler, 1973). Además, también proponemos técnicas que tienen en cuenta efectos de aprendizaje que consideran que realizar ciertas actividades al principio del plan puede reducir el tiempo de procesamiento de las actividades programadas al final.

1.1 DESCRIPCIÓN DEL PROBLEMA

Para este trabajo nos inspiramos en los problemas teóricos de secuenciación de actividades que consideran los efectos del deterioro y el aprendizaje en el contexto de los tiempos de procesamiento de un trabajo (Wang, 2006; Biskup, 1999; Mosheiov, 2001; Cheng *et al.*, 2008). Proponemos dos modelos para representar cómo afecta el estrés la duración de las actividades de aprendizaje en función de la posición de la actividad en el plan educativo. Por supuesto, la secuenciación de trabajos dentro de una planta de producción es muy diferente de una trayectoria de aprendizaje para un estudiante, pero la analogía es que la duración de las actividades de aprendizaje se ven afectadas negativamente (es decir, se deterioran) por su posición durante el semestre dado el factor de estrés e influenciadas positivamente por la capacidad de aprendizaje del estudiante para considerar cambios en sus tiempos de procesamiento.

Ilustremos un ejemplo de $EPP-SL$. La tabla 1.1 describe un semestre académico de 12 periodos. La línea *Estrés* es el comportamiento esperado del estrés de un estudiante s específico durante el periodo académico. Observe que el estrés del estudiante aumenta debido quizá a la carga de trabajo del curso a medida que avanza el período académico. EPP representa la trayectoria de aprendizaje esperada por el profesor que todos los estudiantes deben seguir para adquirir un aprendizaje perfecto sin tener en cuenta ningún factor que pueda afectar al rendimiento del estudiante. Pero cuando esta trayectoria de aprendizaje esperada se aplica a un alumno concreto s , obtenemos la recta $EPP(s)$. Obsérvese que no se realizan todas las actividades, el tiempo total del alumno es superior al estimado por el profesor y la nota final es inaceptable. Nuestra metodología pretende obtener la línea $EPP-SL(s)$ que considera el aumento del estrés a lo largo del periodo académico. Nótese que no todas las actividades se ejecutan en el orden que el profesor estableció, considerando menos actividades al final donde el estrés es mayor. $EPP-SL$ puede devolver planes de mayor duración, pero satisface los requisitos de grado.

TABLA 1.1: EPP : trayectoria de aprendizaje estimada para todos los alumnos; $EPP(S)$: trayectoria de aprendizaje estimada aplicada a un alumno en concreto con estrés creciente al final del semestre; $EPP - SL(S)$: trayectoria de aprendizaje específica para un alumno teniendo en cuenta sus niveles de estrés y sus capacidades de aprendizaje.

| Posición | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | Tiempo | Calificación |
|---------------|------------|------------|------------|------------|------------|------------|------------|------------|---------------|---------------|---------------|---------------|--------|--------------|
| Estrés | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.2 | 0.2 | 0.2 | 0.3 | 0.3 | 0.3 | 0.3 | total | final |
| EPP | α_1 | α_2 | α_3 | α_4 | α_5 | α_6 | α_7 | α_8 | α_9 | α_{10} | α_{11} | α_{12} | 100 | 100 |
| $EPP(S)$ | α_2 | α_3 | α_4 | α_5 | α_1 | | α_8 | α_9 | | α_{10} | α_{11} | | 140 | 65 |
| $EPP - SL(S)$ | α_1 | α_2 | α_6 | α_3 | α_5 | α_4 | α_7 | α_8 | α_{11} | | | | 120 | 80 |

Hasta donde sabemos, no existen enfoques en la literatura que incluyan los índices de aprendizaje y niveles de estrés para generar trayectorias de aprendizaje personalizadas. Nuestras principales contribuciones son dos modelos de secuenciación de actividades para el *EPP – SL* con dos nuevas funciones de duración de las actividades que incorporan el estrés y la posición dentro de la secuencia para imitar los efectos del aprendizaje. Además, proponemos un algoritmo adaptativo aleatorio codicioso (*GRASP Greedy Randomize Adaptative Search Procedure*), mejorado por una búsqueda de vecindad variable (*VNS Variable Neighborhood Search*), para resolver instancias de tamaño real. Reconocemos que puede haber diferentes formas de modelar los efectos del estrés y el aprendizaje. Aún así, nuestro objetivo es proporcionar una metodología para incorporarlos a la generación de trayectorias de aprendizaje y, al mismo tiempo, estudiar los efectos que estos factores tienen en las trayectorias de aprendizaje personalizadas. Creemos que los modelos propuestos considerando el estrés constituyen un primer paso hacia la generación de trayectorias de aprendizaje con medidas subjetivas, que aumentan la personalización del aprendizaje.

1.2 HIPÓTESIS

Como ya mencionamos, el considerar factores subjetivos como el estrés en la generación de trayectorias de aprendizaje nos permite obtener secuencias de actividades personalizadas y plantear las siguientes hipótesis:

- Si el estrés repercute de forma negativa en el desempeño de las personas, y el desempeño lo medimos en función del tiempo necesario para realizar una actividad, entonces, una persona puede necesitar más tiempo del necesario para hacer una actividad cuando sus niveles de estrés son elevados.
- Si el *CI* (*Coefficiente Intelectual*) es una forma de medir la inteligencia de una persona, entonces, una persona con un *CI* mayor al promedio, puede necesitar

menos tiempo para realizar una actividad ya que su desempeño es superior al de los demás.

- Personalizar las trayectorias de aprendizaje de acuerdo a la situación mental de cada estudiante mejora el makespan de las soluciones y, por tanto, el desempeño de los estudiantes.

1.3 OBJETIVOS

El principal objetivo de este trabajo es investigar el impacto del estrés de una persona en su desempeño académico, mediante la implementación de trayectorias de aprendizaje personalizadas, lo cuál se ve reflejado en el tiempo necesario para realizar las actividades que la conforman. Para probar las hipótesis propuestas definimos los siguientes objetivos:

- Desarrollo e implementación de un modelo de optimización que refleje el impacto del estrés en la generación de trayectorias de aprendizaje personalizadas (*EPP – SL*).
- Desarrollo e implementación de un algoritmo heurístico como una opción más rápida para generar trayectorias de aprendizaje personalizadas y para comparar con el modelo de optimización (*VNS – GRASP*).
- Evaluar el desempeño del modelo de optimización y el algoritmo heurístico con las instancias generadas.

1.4 SOLUCIÓN PROPUESTA

Para generar las trayectorias de aprendizaje personalizadas que consideran el estrés mental de una persona y su capacidad intelectual, proponemos la implemen-

tación de un modelo matemático al que llamamos *EPP – SL* que nos proporcione la secuencia de actividades necesarias para satisfacer los objetivos de las materias incorporando las características de cada alumno. Además, la implementación de un algoritmo híbrido *VNS – GRASP* nos ofrecería soluciones de calidad en un tiempo computacional más corto.

Para evaluar los algoritmos mencionados anteriormente, se generaron instancias que simulan períodos de estrés. Cada instancia debe tener un conjunto de actividades que representen una retícula educativa. Se están considerando restricciones de precedencia y actividades obligatorias. Cada actividad debe tener asociado su valor de beneficio y duración. Hasta ahora, se han implementado instancias simétricas; es decir, cada asignatura tiene el mismo número de temas, cada tema tiene el mismo número de subtemas y cada subtema tiene el mismo número de actividades que los demás.

1.5 ESTRUCTURA DE LA TESIS

El capítulo 2 describe los antecedentes de nuestro problema y el área de la ciencia en el que nos estamos ubicando. A pesar de ser un problema enfocado en el área educativa, hicimos una analogía con problemas de planeación y secuenciamiento de tareas, por lo que es necesario definir algunos conceptos de este tipo de problemas. De igual manera, en esta sección se describe de forma muy general problemas de optimización entera y algoritmos meta-heurísticos utilizados para resolver el *EPP – SL*.

El capítulo 3 detalla literatura reciente relacionada con el área educativa en la actualidad. Se describen los efectos negativos de altos niveles de estrés en los estudiantes, los tipos de estrés que existen y que datos biométricos podríamos utilizar para medirlo. También se menciona porque las trayectorias de aprendizaje pueden ser una herramienta de apoyo para los estudiantes durante su vida académica.

A continuación en el capítulo 4 describimos formalmente el $EPP - SL$ junto con los dos modelos matemáticos de programación lineal entera propuestos y un algoritmo meta-heurístico $VNS - GRASP$ para resolver el $EPP - SL$. La experimentación realizada en cada uno de los métodos de solución propuestos y una discusión de los resultados obtenidos se muestran en una subsección del capítulo 4.

Las conclusiones generales de la tesis y las contribuciones hechas se describen en el capítulo 5, así como el trabajo a futuro con el que pretendemos continuar.

De éste trabajo resultó una publicación en la revista *Interactive Learning Environments* con el nombre de “Scheduling personalized study plans considering the stress factor” en el cual abordamos el problema $EPP - SL$ y el algoritmo meta-heurístico $VNS - GRASP$.

Éste trabajo fue presentado en las conferencias siguientes durante los últimos 2 años:

- XXI Latin-Iberoamerican Conference on Operations Research - CLAIO. Con el tema: “A personalized activity scheduling problem considering stress, deadlines, and teamwork”.
- X Congreso de la Sociedad Mexicana de Investigación de Operaciones - CSMIO. Con el tema: “Diferentes métodos de solución para el problema de programación de actividades considerando el estrés”.
- Escuela Latinoamericana de Verano en Investigación Operativa - ELAVIO Con el tema: “Scheduling personalized study plans considering stress factor”.
- IX Congreso de la Sociedad Mexicana de Investigación de Operaciones - CSMIO. Con el tema: “Educational Planning Problem with stress”.

CAPÍTULO 2

ANTECEDENTES

En este capítulo describimos de forma general algunas áreas de conocimiento importantes para el desarrollo de este trabajo. Recordemos que el *EPP – SL* está basado en problemas de planeación y secuenciamiento de actividades (*scheduling*) por lo que es importante describir de forma muy general dichos problemas y algunos de sus componentes. Otro punto importante es que se esta presentando un modelo de optimización, en particular, un problema de programación entera binario. Y por último, en este trabajo propusimos un algoritmo meta-heurístico *GRASP – VNS* por lo que es importante definir a que nos referimos al hablar de meta-heurísticas.

2.1 PLANEACIÓN Y SECUENCIAMIENTO DE TAREAS

Pinedo (2016) define los problemas de programación de actividades (a partir de aquí nos referiremos a la programación de actividades como «scheduling», por el idioma inglés) como un proceso de toma de decisiones que tiene una gran importancia en la mayoría de los procesos de manufactura y sistemas de producción. Se ocupa de la asignación de recursos o tareas en un periodo de tiempo, teniendo como meta optimizar uno o más objetivos. Los objetivos varían de acuerdo al problema, en algunos casos puede ser la minimización del tiempo de terminación de la última tarea realizada, otro puede ser la maximización de las tareas completadas en un

tiempo límite. De igual manera, las tareas y los recursos pueden tomar diferentes formas, un recurso puede ser una máquina en un taller, una pista de aterrizaje en un aeropuerto, ejecución de programas, entre otros.

Según Brucker (2004) podemos ver el problema de scheduling como un conjunto de trabajos $J_i (i = 1, 2, \dots, n)$ que requieren ser procesados en un determinado número de máquinas $M_j (j = 1, 2, \dots, m)$. Un trabajo J_i se conforma de un número n_i de operaciones O_{i1}, \dots, O_{in} con un tiempo de procesamiento cada una. Además, cada operación está asociada con un conjunto de máquinas en las cuales puede ser procesada $\mu_{ij} \subseteq \{M_1, \dots, M_m\}$. Típicamente los problemas de scheduling pueden ser representados mediante diagramas de Gantt como en la figura 2.1, donde, contamos con tres máquinas (M_1, M_2, M_3) y tres trabajos (J_1, J_2, J_3) los cuales tienen diferente número de operaciones, el trabajo J_1 y J_2 tienen tres operaciones para ser completados y el trabajo J_3 tiene solo dos.

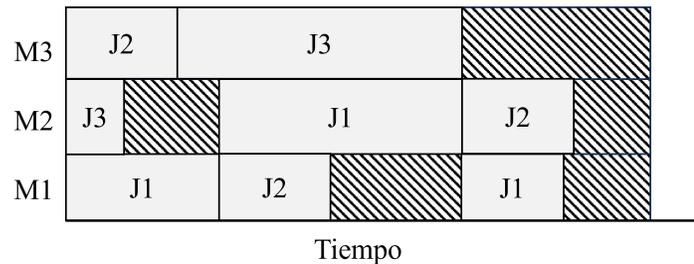


FIGURA 2.1: Ejemplo del diagrama de Gantt de la programación de tres trabajos por máquina.

El objetivo de este tipo de problemas es secuenciar las operaciones de dichos trabajos respetando un número de restricciones y tratando de optimizar uno o más criterios de desempeño.

De forma análoga, el *EPP – SL* es similar a los problemas de scheduling en el sentido de que un conjunto de actividades/trabajos necesitan ser procesadas por un conjunto de máquinas/estudiantes en un cierto orden, de manera que les permita alcanzar uno o varios objetivos.

Para el problema desarrollado en este trabajo utilizamos cierta notación relacionada con problemas de scheduling la cuál se describe a continuación. Para los trabajos es necesario conocer:

- Tiempo de procesamiento (*processing time*). Normalmente se denota como p_{ij} que es el tiempo de procesamiento del trabajo j en la máquina i . El sub índice i se puede omitir si no es relevante en el problema la máquina en la que se procesa o solamente existe una máquina.
- Fecha de término (*due date*). Representa el tiempo en la que se debe de completar la tarea o el trabajo j y se denota como d_j . Normalmente se puede realizar el trabajo después de la fecha de término con cierta penalidad. Sin embargo, si el trabajo no se puede realizar después de la fecha de término se conoce como fecha límite (*deadline*) y se denota como \bar{d}_j .
- Máquinas idénticas en paralelo. Si nos referimos al ambiente en el que serán procesadas las actividades, significa que existen m máquinas idénticas dispuestas en paralelo y el trabajo j que tiene solo una operación puede ser procesado en cualquiera de dichas máquinas.
- Máquinas en paralelo no relacionadas. Existen m máquinas diferentes dispuestas en paralelo, donde la máquina i puede procesar el trabajo j a la velocidad v_{ij} . Por lo tanto, el tiempo de procesamiento p_{ij} del trabajo i en la máquina j es igual a $\frac{p_{ij}}{v_{ij}}$. En el caso de que la velocidad de la máquina sea independiente de los trabajos $v_{ij} = v_i, \forall(i, j)$.
- Restricciones de precedencia. Las restricciones de precedencia pueden aparecer en ambientes de una sola máquina o en máquinas paralelas, requiriendo que uno o más trabajos deban de ser completados antes de que otro trabajo sea admitido.

2.2 PROGRAMACIÓN LINEAL

Programación lineal es una rama de las matemáticas aplicadas en donde un problema de programación lineal maximiza o minimiza una función lineal sujeta a un conjunto finito de restricciones lineales.

Wolsey (2020) define programación entera como una forma de resolver problemas de optimización con variables enteras. Dichas variables sirven para modelar indivisibilidades y variables binarias usadas usualmente para representar decisiones. Existe una gran variedad de problemas que se pueden modelar con programación entera, por ejemplo: Problemas de schedulling, planeación, telecomunicaciones, problemas de corte y mucho más.

Un problema lineal se puede definir de la siguiente manera, supongamos que tenemos un problema lineal:

$$\max\{cx : Ax \leq b, x \geq 0\} \quad (2.1)$$

Donde A es una matriz $m \times n$, c es un vector n -dimensional de renglones y b un vector m -dimensional de columnas de variables. Si algunas de dichas variables pero no todas son enteras, tenemos un problema de Programación Entero Mixto (*MIP* Mixed Integer Program por sus siglas en inglés):

$$\max cx + hy \quad (2.2)$$

$$Ax + Gy \leq b \quad (2.3)$$

$$x \geq 0, y \geq 0, x, y \in Z \quad (2.4)$$

Donde A otra vez es una matriz de tamaño $m \times n$ y G es una matriz de tamaño $m \times p$, h es un p -vector de renglones y y es un p -vector de columnas de variables

enteras. Si todas las variables son enteras tenemos un problema de programación entera (*IP* Integer Program por sus siglas en inglés), el cual se describe como:

$$\text{máx } cx \tag{2.5}$$

$$Ax \leq b \tag{2.6}$$

$$x \geq 0, x \in Z \tag{2.7}$$

Y si todas sus variables enteras pueden tomar solo valores entre $\{0,1\}$ tenemos un problema de programación entera binaria (*BIP* Binary Integer Program por sus siglas en inglés):

$$\text{máx } cx \tag{2.8}$$

$$Ax \leq b \tag{2.9}$$

$$x \in \{0, 1\}^n \tag{2.10}$$

El *EPP – SL* descrito en este trabajo es un problema entero binario, ya que sus variables de decisión solo pueden tomar valores entre 0 y 1, que representa si una actividad de aprendizaje se agrega en la trayectoria de aprendizaje o no.

2.3 META-HEURÍSTICAS

Una meta-heurística es una estructura algorítmica de alto nivel independiente del problema que da un conjunto de estrategias para realizar algoritmos de optimización heurística. Las meta-heurísticas son desarrolladas para encontrar soluciones lo suficientemente buenas en un tiempo computacional lo suficientemente pequeño. Las meta-heurísticas han demostrado en muchas ocasiones que pueden llegar a ser viables y casi siempre superiores a los métodos exactos más tradicionales como ra-

mificación y acotamiento (*branch and bound*, ver apéndice A) especialmente para problemas muy complicados o instancias muy grandes Sörensen y Glover (2013).

Los algoritmos meta-heurísticos intentan encontrar la mejor solución factible posible para un problema de optimización evaluando potenciales soluciones y desarrollando una serie de operaciones en ellas tratando de encontrar diferentes y mejores soluciones. Estos algoritmos trabajan con soluciones codificadas de tal manera que puedan ser fácilmente almacenadas por la memoria de una computadora y a la cual se le puedan aplicar una serie de operaciones. Existen tres categorías de meta-heurísticas de acuerdo a la forma en como estos algoritmos manipulan las soluciones. El primero, meta-heurísticas de búsqueda local, las cuales de forma iterativa aplican operadores sobre la solución de tal manera que genere pequeños cambios, tratando de encontrar una mejor solución. La segunda son meta-heurísticas constructivas las cuales se enfocan en la construcción de cada uno de los componentes de la solución. Y la última, meta-heurísticas basadas en poblaciones, las cuales combinan soluciones en nuevas. Cabe mencionar que estas categorías no son mutuamente excluyentes, existen algoritmos que combinan estas estrategias. De hecho en este trabajo, combinamos dos meta-heurísticas, un algoritmo *GRASP* que es un algoritmo constructivo y un algoritmo *VNS* que es un algoritmo que considera búsquedas locales en varios vecindarios, ambos se explican a detalle más adelante.

No hay que confundir un algoritmo heurístico con un algoritmo meta-heurístico. Los primeros usualmente son utilizados para un problema en particular o para un tipo de instancia. Mientras que los algoritmos meta-heurísticos son de propósito general y se pueden aplicar a casi cualquier problema de optimización Talbi (2009).

CAPÍTULO 3

REVISIÓN DE LITERATURA

Esta sección está dividida en tres partes, la primera describe trabajos relacionados con el ámbito educativo. La segunda parte describe trabajos aplicados a la generación de rutas de aprendizaje. En la tercera parte se muestran algunos trabajos relacionados con el estrés y por qué es importante considerarlo en problemas de secuenciación de actividades donde interviene el factor humano.

A partir de la revisión de la literatura, observamos que aunque existen trabajos de diferentes áreas de investigación que tratan el problema de generar trayectorias de aprendizaje, no hemos podido identificar enfoques que incluyan los ritmos de aprendizaje y el comportamiento del estrés para proponer trayectorias de aprendizaje personalizadas. Además, determinar el impacto de estos factores en la generación de trayectorias de aprendizaje sigue siendo un área de investigación abierta.

3.1 TRAYECTORIAS DE APRENDIZAJE

PERSONALIZADAS COMO MÉTODO PARA AFRONTAR LA DESERCIÓN EDUCATIVA

Tzouveli *et al.* (2008b) mencionan que los sistemas de aprendizaje no suelen

adaptarse a los perfiles de aprendizaje de los alumnos. Introducen una metodología de e-learning que se adapta a los niveles de tecnología de la información y la comunicación de los estudiantes. Klašnja-Milićević *et al.* (2011) proponen un sistema de tutoría de programación que se adapta a los intereses y niveles de conocimiento del alumno. Smet *et al.* (2016b) diseñaron un análisis multinivel para estudiar el impacto en los resultados del aprendizaje según el diseño de los itinerarios de aprendizaje, el entorno individual/colaborativo y el género.

El uso de herramientas de tutoría inteligentes es cada vez más común. Este tipo de herramientas proporcionan a los usuarios recursos educativos que les permiten desarrollar un aprendizaje en cierta área (Nabizadeh *et al.*, 2017). Un ejemplo de este tipo de herramientas puede ser el uso de *trayectorias de aprendizaje*. Una *trayectoria de aprendizaje* se puede definir como la implementación de un diseño curricular, la cual guía a los usuarios para alcanzar un objetivo de aprendizaje Nabizadeh *et al.* (2020).

De Smet *et al.* (2016) encontraron que el uso de este tipo de herramientas efectivamente mejoran el desempeño de los estudiantes en comparación con los métodos de enseñanza tradicional. Sin embargo, en la mayoría de los casos, estos sistemas ofrecen una misma solución para todos, la cual, generalmente no se adapta al perfil de todos ya que deja de lado los conocimientos previos de los usuarios y su capacidad de aprendizaje lo que conlleva a invertir más tiempo del necesario antes de alcanzar su objetivo de aprendizaje (Tzouveli *et al.*, 2008b). Por lo que, generar trayectorias de aprendizaje personalizadas significaría una mejora en este tipo de herramientas.

Klašnja-Milićević *et al.* (2011) define un aprendizaje personalizado cuando un sistema de tutorías inteligente destina deliberadamente esfuerzos a diseñar soluciones que se ajustan a las necesidades, objetivos, talentos e intereses de los aprendices.

Durand *et al.* (2013) mencionan que el diseño de trayectorias de aprendizaje personalizadas es una tarea compleja. Proponen un enfoque basado en la teoría de grafos que utiliza el concepto de camarillas para recomendar rutas de aprendizaje.

Nabizadeh *et al.* (2020) señalan la dificultad de recomendar materiales de aprendizaje que los usuarios puedan completar a tiempo. Por ello, presentan dos enfoques para maximizar las calificaciones de los estudiantes en un curso y, al mismo tiempo, satisfacer sus limitaciones de tiempo. Utilizan heurísticas de búsqueda local para resolver sus modelos.

Hwang *et al.* (2010) detectan los comportamientos de aprendizaje de los estudiantes con sensores y proponen un problema de optimización que modela los objetivos y criterios para determinar rutas de aprendizaje personalizadas para maximizar la eficacia del aprendizaje de los estudiantes individuales. Al igual que hacemos en este trabajo, proponen un algoritmo heurístico para resolver su modelo. Takači *et al.* (2017) utilizan la búsqueda de vecindario variable para formar grupos heterogéneos para el aprendizaje colaborativo.

3.2 ESTRÉS

Selye (1975) define *estrés* como una respuesta al cambio, en términos de mantener la estabilidad, frente a estímulos que alteran el equilibrio físico y mental del cuerpo. También se puede definir como un estado no adaptativo en el que el sistema nervioso simpático está hiper-activado, provocando un deterioro físico y psicológico agudo o crónico, derivando en enfermedades psicológicas como la ansiedad o la depresión y enfermedades médicas como la hipertensión arterial y la diabetes Campkin (2000).

Por ello, el estrés es uno de los principales factores que inciden en el rendimiento académico de los estudiantes (Vogel *et al.*, 2018) (Beehr y Newman, 1978) (Alzahem *et al.*, 2015) (Özpolat y Akar, 2009), y en la última década ha sido descrito como uno de las causas de las enfermedades cardiovasculares.

Wolf *et al.* (2015) relata una serie de factores relacionados con la generación de niveles altos de estrés, entre ellos factores personales como relaciones problemáticas

entre estudiantes, falta de apoyo de familiares y amigos; y factores profesionales como la falta de capacidad y el miedo al fracaso. También se menciona que si bien el estrés puede tener un efecto positivo en el rendimiento, como un aumento de la motivación o la generación de nuevas ideas, también menciona que los altos niveles de estrés provocan una disminución del rendimiento y depresión. El estrés se puede presentar de diferentes formas:

- Agudo: Es principalmente causado por emociones y es de muy corta duración.
- Episodios de estrés agudo: Normalmente ocurre en el ámbito profesional, por ejemplo, cuando se tienen fechas límite muy justas. Estos episodios dan la percepción de que se pasa de un problema a otro impidiendo volver a un estado de tranquilidad.
- Estrés crónico: En este tipo de situaciones no se puede ver la manera en como salir del estado de estrés.

3.2.1 CÓMO MEDIR EL ESTRÉS

Recordemos que el objetivo de este trabajo no es proponer un nuevo método de medición del estrés, si no una forma de incorporarlo en problemas de secuenciación de actividades. Por lo que, no se profundiza en la relación biológica entre los datos biométricos y el estrés.

Uno de los principales inconvenientes de incorporar características subjetivas como el estrés es, ¿Cómo podemos darle un valor numérico?.

Brown *et al.* (2020) presentó un método que permite medir la reducción en las variaciones de la frecuencia cardíaca. Lo hace mediante el uso de sensores de *ECG* (Electrocardiograma) durante un período de 24 horas. El procedimiento se describe en Brown *et al.* (2018) donde se hace una comparación entre diferentes formas de medir el estrés psicológico; uno de ellos es a través de la frecuencia cardíaca y el

otro midiendo la variación de la frecuencia cardíaca; en dicho trabajo, toman valores de VFC en diferentes momentos del día y en diferentes situaciones, para descartar momentos como, por ejemplo, cuando duerme o cuando hace ejercicio.

Alberdi *et al.* (2016) muestra un novedoso modelo multimodal para evaluar los niveles de estrés en una persona tomando en cuenta características psicológicas obtenidas mediante cuestionarios, datos biométricos como presión sanguínea, electroconductividad de la piel, variación de la frecuencia cardíaca, entre otros.

Existen otros trabajos de investigación donde se menciona que ciertas características como la ansiedad y el estrés se pueden reflejar en expresiones faciales como micro-movimientos en los ojos, boca y mejillas. Y en particular para el estrés, en movimientos de la cabeza y el cuello Giannakakis *et al.* (2017).

Podemos notar que existen diferentes formas de estimar el estrés mental en un momento específico mediante cambios en datos biométricos, siendo la variación de la frecuencia cardíaca VFC (HRV por sus siglas en inglés de Heart rate variability) el más aceptado, quizá debido a ser un método no invasivo (Trotman *et al.*, 2019) (Ribeiro y Cunha, 2018). Sin embargo, hasta ahora no existe una única forma establecida y reconocida por la comunidad científica para evaluar el estrés (Kim *et al.*, 2018).

El tener una forma de evaluar el estrés nos permitiría conocer la relación entre el desempeño de una persona y sus niveles de estrés. Sanders (1983) sugiere varias formas de representar dicha relación, sin embargo, solo consideraremos dos. La primera es la forma de U-invertida de la función *Yerkes – Dodson* (Westman y Eden, 1996) (ver figura 3.1, lado izquierdo), donde el estrés a un nivel óptimo coincide con un desempeño óptimo, contrario a lo que pasa cuando los niveles son muy bajos o muy altos, lo que repercute en un desempeño menor. La segunda forma de ver la relación entre el desempeño y el estrés es una forma lineal negativa, donde a un mayor nivel de estrés, menor desempeño.

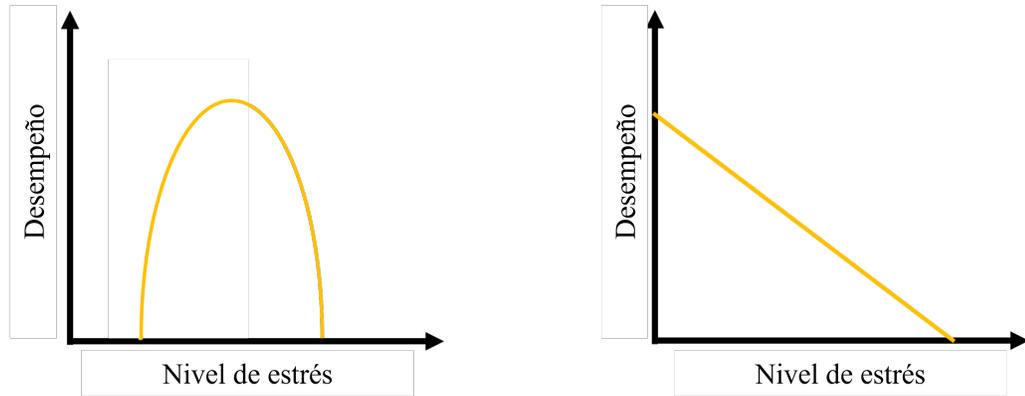


FIGURA 3.1: Ejemplo de la relación entre los niveles de estrés y el desempeño de una persona.

CAPÍTULO 4

PROBLEMA DE PLANIFICACIÓN EDUCATIVA CONSIDERANDO EL ESTRÉS

En esta sección, presentamos la notación utilizada para el modelo matemático y el algoritmo meta-heurístico que proponemos para resolver el problema de planificación educativa con efectos de estrés y aprendizaje *EPP-SL*. Parte de éste capítulo fue obtenido del trabajo Baldazo *et al.* (2023).

Como se ilustra en la figura 4.1, un conjunto de asignaturas Φ deben ser cubiertas durante un periodo escolar. Cada asignatura $\phi \in \Phi$ consta de un conjunto de temas T^ϕ que a su vez tienen un conjunto de subtemas $\Delta^{\phi\tau}$, para $\phi \in \Phi$ y $\tau \in T^\phi$. Por último, cada conjunto de subtemas $\Delta^{\phi\tau}$ se compone de un conjunto de actividades $A^{\phi\tau\delta}$, de las cuales algunas son obligatorias $\alpha \in M$, tal que $M \subseteq A^{\phi\tau\delta}$ y algunas de ellas tienen restricciones de precedencia, para $\phi \in \Phi, \tau \in T^\phi, \delta \in \Delta^{\phi\tau}$. Las características de una actividad $\alpha \in A^{\phi\tau\delta}$ forman un tupla (d_α, g_α) donde d_α es la duración media de la actividad estimada por el instructor, y g_α es la importancia o el valor de la actividad en la calificación de la asignatura.

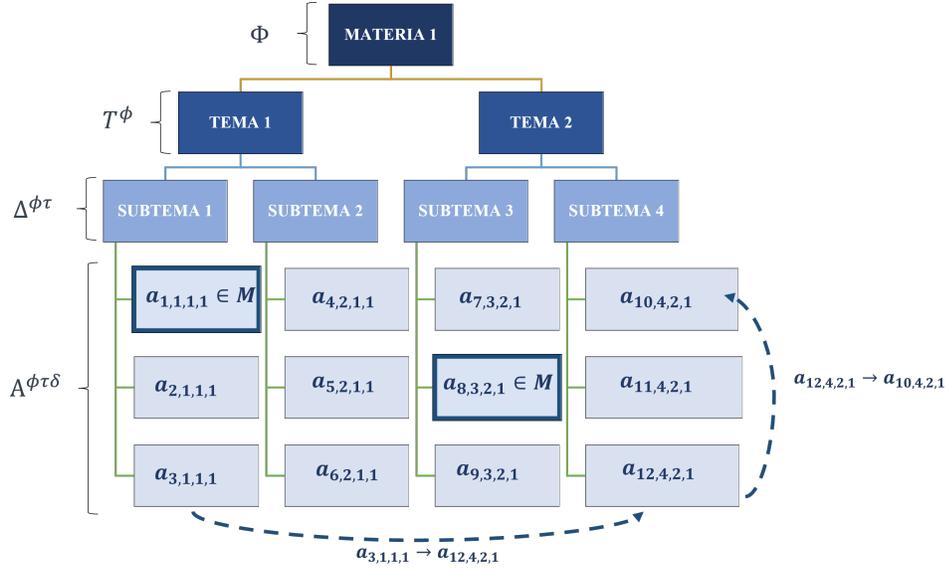


FIGURA 4.1: Ejemplo de estructura de una retícula educativa.

El $EPP-SL$ debe elegir y secuenciar un conjunto de actividades de los diferentes conjuntos $A^{\phi\tau\delta}$ para alcanzar la nota esperada de una asignatura y minimizar el tiempo total de realización de las actividades elegidas, para $\phi \in \Phi, \tau \in T^\phi, \delta \in \Delta^{\phi\tau}$. Las actividades elegidas se programan en posiciones para ser realizadas en ese orden durante el curso académico, como se ve en el ejemplo de la tabla 1.1. Algunas de las posiciones podrían estar vacías en el plan educativo. Una solución al $EPP-SL$ es una secuencia de actividades asignadas en una determinada posición, esta secuencia es de tamaño

$$n = \left| \bigcup_{\phi \in \Phi, \tau \in T^\phi, \delta \in \Delta^{\phi\tau}} A^{\phi\tau\delta} \right| = |A|.$$

Algunas actividades son obligatorias y pertenecen al conjunto $M \subset A^{\phi\tau\delta}$. La decisión para estas actividades consiste únicamente en determinar su posición en el periodo académico.

Algunas otras actividades tienen un conjunto de actividades obligatorias que deben ejecutarse previamente. En estos casos, las actividades encadenadas tienen que

programarse en orden. A este conjunto de actividades de precedencia lo denominamos $H(\alpha)$ las cuales deben ejecutarse antes de α donde $\alpha \in A^{\phi\tau\delta}$.

Como se ha mencionado en la introducción, nos inspiramos en los problemas teóricos de scheduling que consideran los efectos del deterioro y el aprendizaje en los tiempos de procesamiento de los trabajos. En otras palabras, modelamos los efectos del deterioro y el aprendizaje que afectan a los tiempos de procesamiento de las actividades de aprendizaje dadas sus posiciones durante el curso académico. A continuación presentamos nuestros dos modelos para las duraciones de las actividades que dependen de las capacidades de aprendizaje y del estrés.

4.1 LA CAPACIDAD DE APRENDIZAJE DEL ESTUDIANTE AFECTA LA DURACIÓN DE LAS ACTIVIDADES

Dado que cada estudiante es diferente, la duración prevista de las actividades puede variar entre sí debido a que la capacidad intelectual de cada estudiante influye como un efecto de aprendizaje en el rendimiento global durante el curso académico.

Denotamos la duración de una actividad en función de la posición de la secuencia y de una persona específica como $d_{\alpha p}$ para la actividad $\alpha \in A^{\phi\tau\delta}$, posición $p = 1, \dots, n$, para $\phi \in \Phi$, $\tau \in T^\phi$, y $\delta \in \Delta^{\phi\tau}$. Como se ha mencionado, no podemos utilizar la función de duración propuesta para los problemas de programación (Biskup, 2008), pero heredamos los conceptos principales y los adaptamos a nuestro problema $EPP - SL$.

Si existen efectos de aprendizaje en los alumnos, pueden tener enormes consecuencias en su plan de actividades, por un lado, y en el cálculo del tiempo total de las actividades, por otro. El aprendizaje se modela frecuentemente utilizando la

fórmula de potencia (Biskup, 2008):

$$d_{\alpha,p} = d_{\alpha} p^a,$$

donde $a = \log_2 LR \leq 0$ es el índice de aprendizaje que depende de la tasa de aprendizaje LR .

Cuanto menor sea la tasa de aprendizaje LR , mayor será el efecto del aprendizaje. Para modelizar los distintos índices de aprendizaje de los alumnos, suponemos que $LR \in [0.95, 1.05]$. De esta forma, los alumnos que tardan más en terminar sus actividades son los que tienen 1.05, y el efecto de aprendizaje más rápido es para los que tienen 0.95. La sustitución de los tiempos de procesamiento conduce a $LR = 2^a$ o a $a = \log_2 LR$.

Los valores supuestos para el ritmo de aprendizaje LR fueron ajustados al modelo, para evitar obtener datos exagerados como que tenía un exceso de aprendizaje o muy bajo.

4.2 EL ESTRÉS DETERIORA LA DURACIÓN DE LAS ACTIVIDADES

Dado que cada estudiante es diferente, los tiempos de duración previstos de las actividades difieren, especialmente cuando están sujetos a un cierto nivel de estrés mental (revisar capítulo 3.2). Por ejemplo, un estudiante puede estar más estresado al final o al principio del semestre. Así, el estrés relacionado con la posición de la actividad en la secuencia puede afectar a la duración media de la actividad. Por ejemplo, las actividades programadas durante un periodo con altos niveles de estrés pueden tener una duración más significativa que las mismas actividades posicionadas en un periodo con un nivel de estrés más bajo.

Cuando la duración de una actividad se ve afectada por el estrés encontrado a lo largo del tiempo, utilizamos el concepto de los problemas de scheduling con

deterioro de tiempos (Wang, 2006; Wang y Cheng, 2007), que corresponde a un aumento del tiempo de procesamiento de las actividades causado por su posición en la secuencia. La forma en como modificamos la duración de las actividades de acuerdo a su posición es similar al trabajo propuesto por Wang (2006) lo que nos permite obtener un nuevo modelo de duración de las actividades educativas:

$$\bar{d}_{\alpha,p} = d_{\alpha}(1 + s_p)p^a, \quad (4.1)$$

donde $s_p \in [0, 1]$ es el valor de estrés esperado en la posición p relacionada con un alumno en concreto. Esta función refleja que el tiempo medio de duración de las actividades se ve incrementado por el estrés al que se enfrenta el estudiante en el periodo p , pero también se tiene en cuenta el índice de aprendizaje.

Otro modelo para la duración de las actividades implica que cuanto más se acerca el final del semestre del curso académico, más cargados y presionados están los estudiantes. Para ello, consideramos la siguiente función

$$\hat{d}_{\alpha,p} = d_{\alpha}(1 + s_p \log_{\beta} p)p^a, \quad (4.2)$$

donde el parámetro β (igual a 10 en este trabajo) influye en la duración de la secuencia de planificación, y lo utilizamos en el término $\log_{\beta} p$ para suavizar el impacto del estrés sobre la posición p . Para esta función, el estrés en cada posición p aumenta con la proximidad del final del periodo académico. La principal ventaja de las ecuaciones de duración de la actividad (4.1) y (4.2) es que consideran simultáneamente ambas características de los estudiantes. Por ejemplo, un estudiante que presenta más estrés al principio del semestre puede compensar esta situación gracias a su coeficiente intelectual.

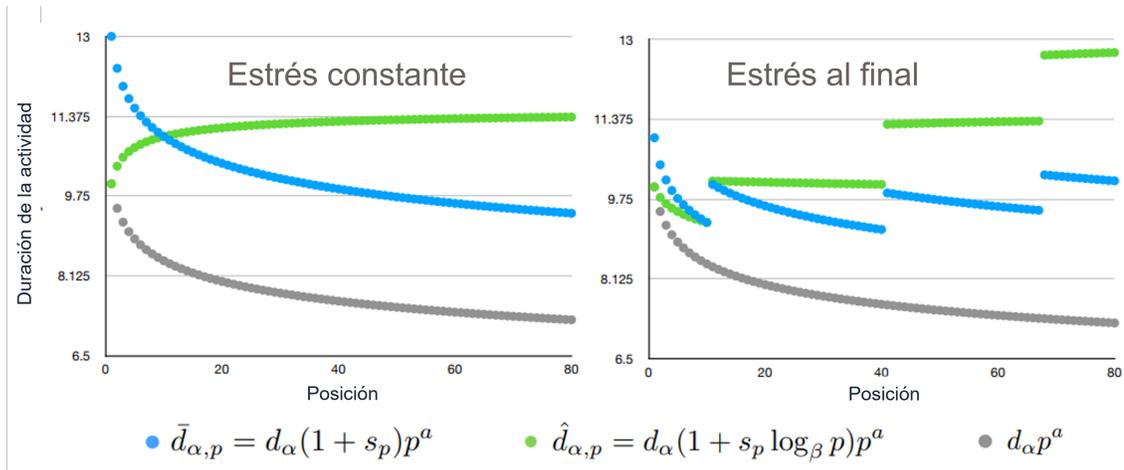


FIGURA 4.2: Modelos de duración de actividades para una sola actividad con una duración promedio de 10.

En la figura 4.2, mostramos las dos funciones de duración de la actividad que proponemos, $\hat{d}_{\alpha,p}$ y $\bar{d}_{\alpha,p}$, y la comparamos con la que sólo considera un efecto de aprendizaje $d_{\alpha}p^a$ propuesta por Biskup (1999). Este ejemplo es para una única actividad con una duración estimada de 10 unidades de tiempo y un alumno con un alto ritmo de aprendizaje. El gráfico de la izquierda considera un estrés moderado constante a lo largo del periodo académico, mientras que el de la derecha simula un aumento del estrés al final del periodo. Las posiciones que pueden ocupar las actividades en el plan de aprendizaje están en el eje de abscisas, mientras que la duración de la actividad está en el eje de ordenadas.

Para el caso de estrés constante (gráfico izquierdo de la figura 4.2), la función de duración de la actividad $\bar{d}_{\alpha,p}$ se ve muy afectada por el estrés en sus primeras posiciones. Sin embargo, cuando las posiciones de las actividades aumentan, el efecto de aprendizaje supera el impacto del estrés constante porque aumenta la capacidad de aprendizaje del alumno. Obsérvese que la función de duración de las actividades $\hat{d}_{\alpha,p}$ acentúa el estrés a lo largo de los periodos. Cuando el estrés no es costante sino que aumenta al final del periodo académico, obtenemos funciones de duración de las actividades que enfatizan el efecto de aprendizaje para $\bar{d}_{\alpha,p}$ sobre el estrés o el

otro comportamiento con $\hat{d}_{\alpha,p}$ para el que el estrés es más significativo.

4.3 MODELO MATEMÁTICO

En esta sección se describen los dos modelos matemáticos de programación lineal (ver capítulo 2.2) que proponemos en este trabajo.

A partir del conjunto de actividades $A^{\phi\tau\delta}$, nuestros modelos matemáticos deben seleccionar una secuencia de actividades que les permita alcanzar una calificación mínima. La variable principal, en todos nuestros modelos, es $x_{\alpha,p}$, que es igual a uno si la actividad α va a ser realizada por el alumno en la posición p . Nuestros modelos se parametrizan con las calificaciones mínimas y máximas esperadas del estudiante $K = [K_{\min}, K_{\max}]$. De hecho, un estudiante puede desear obtener al menos 70/100, la nota mínima para aprobar, y menos de 90/100 porque quiere centrarse en otras actividades extracurriculares.

Tenemos dos modelos de duración de las actividades en función del periodo en el que se realizan:

- $EPP - SL(K)$: Modelo de planificación educativa considerando $\bar{d}_{\alpha,p} = d_{\alpha}(1 + s_p)p^{\alpha}$ como la duración de la actividad α en la posición p con estrés constante.
- $EPP - PSL(K)$: Modelo de planificación educativa considerando $\hat{d}_{\alpha,p} = d_{\alpha}(1 + s_p \log_{\beta} p)p^{\alpha}$ como la duración de la actividad α en la posición p con un impacto del estrés variable β .

Una de nuestras principales aportaciones es considerar la nota del alumno como una restricción, considerando, en la función objetivo, el tiempo total de realización (makespan) del plan. Para el modelo $EPP - SL(K)$, la función objetivo es:

$$\min \sum_{\phi \in \Phi} \sum_{\tau \in T^{\phi}} \sum_{\delta \in \Delta^{\tau}} \sum_{\alpha \in A^{\delta}} \sum_{p \in P} \bar{d}_{\alpha,p} x_{\alpha,p}. \quad (4.3)$$

Del mismo modo, para el modelo $EPP - PSL(K)$ tenemos la siguiente función objetivo:

$$\text{mín} \sum_{\phi \in \Phi} \sum_{\tau \in T^\phi} \sum_{\delta \in \Delta^\tau} \sum_{\alpha \in A^\delta} \sum_{p \in P} \hat{d}_{\alpha,p} x_{\alpha,p}. \quad (4.4)$$

En ambos casos, la función objetivo sólo tiene en cuenta la duración de las actividades elegidas en su posición específica.

A continuación exponemos las restricciones de nuestros modelos. Ambos modelos necesitan las siguientes restricciones.

$$\sum_{\phi \in \Phi} \sum_{\tau \in T^\phi} \sum_{\delta \in \Delta^\tau} \sum_{\alpha \in A^\delta} x_{\alpha p} \leq 1 \quad p \in P, \quad (4.5)$$

$$\sum_{p \in P} x_{\alpha p} \leq 1 \quad \alpha \in A, \quad (4.6)$$

$$\sum_{p \in P} x_{\alpha,p} = 1 \quad \alpha \in M. \quad (4.7)$$

Las restricciones 4.5 garantizan que no haya más de una actividad por posición $p \in P$, mientras que las restricciones (4.6) garantizan que ninguna actividad se sitúe en más de una posición. Las restricciones (4.7) obligan a que todas las actividades de aprendizaje obligatorias estén en una de las posiciones del período académico.

Para una actividad α con un conjunto de requisitos de precedencia $H(\alpha)$, las restricciones (4.8) obligan a todas las actividades $\alpha' \in H(\alpha)$ a ejecutarse en cualquier posición p' antes de p , que es la posición para α cuando la variable $x_{\alpha,p} = 1$. La restricción es redundante cuando esta variable es cero. Las restricciones (4.9) prohíben que la actividad α se coloque en una posición menor que el número de sus actividades de precedencia en $H(\alpha)$. Esta restricción es una desigualdad válida que mejora la relajación lineal del modelo.

$$\sum_{\alpha' \in H(\alpha)} \sum_{p' < p} x_{\alpha' p'} \geq |H(\alpha)| x_{\alpha p} \quad p \in P, \alpha \in A, \quad (4.8)$$

$$x_{\alpha p} = 0 \quad \alpha \in A, p \leq |H(\alpha)| \quad (4.9)$$

Por ejemplo, en la figura 4.3 podemos ver que la actividad I requiere que la actividad J se realice primero, pero, la actividad J requiere a su vez, que se realicen

primero las actividades K y L , por lo que la actividad I debe de programarse después de que estas tres actividades se programen.

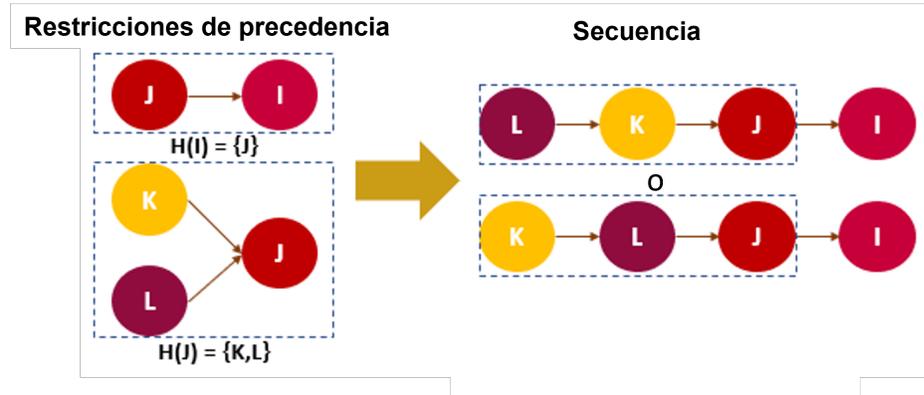


FIGURA 4.3: Ejemplo de restricciones de precedencia.

Para garantizar que cada uno de los valores ponderados de las actividades es igual o superior a la nota final mínima deseable $K_{\text{mín}}$ e igual o inferior a la nota máxima $K_{\text{máx}}$, tenemos las siguientes restricciones:

$$\sum_{\tau \in T^\phi} \sum_{\delta \in \Delta^\tau} \sum_{\alpha \in A^\delta} \sum_{p \in P} v_\alpha x_{\alpha,p} \geq K_{\text{mín}} \quad \phi \in \Phi, \quad (4.10)$$

$$\sum_{\tau \in T^\phi} \sum_{\delta \in \Delta^\tau} \sum_{\alpha \in A^\delta} \sum_{p \in P} v_\alpha x_{\alpha,p} \leq K_{\text{máx}} \quad \phi \in \Phi. \quad (4.11)$$

Este trabajo esta basado en el *EPP* (Educational Planning Problem) modelo propuesto por Sanchez Nigenda *et al.* (2018). Sin embargo, dicho modelo no considera que la duración de la actividad puede variar con la posición de la secuencia debido al nivel de estrés en dicha posición y no considera el efecto de aprendizaje derivado del coeficiente intelectual del estudiante. Así, considerando $p = 0$ para todas las actividades, podemos adaptar su modelo para utilizarlo como comparación para nuestros modelos propuestos en la Sección experimental 4.3.1:

$$\begin{aligned} & \min \sum_{\phi \in \Phi} \sum_{\tau \in T^\phi} \sum_{\delta \in \Delta^\tau} \sum_{\alpha \in A^\delta} d_\alpha x_{\alpha,0} \\ \text{s.t.} \quad & \sum_{p \in P} x_{\alpha,p} = 1 \quad \alpha \in M, \end{aligned}$$

$$\begin{aligned}
 x_{\alpha p} &= 0 & \alpha \in A, p \leq |H(\alpha)| & & \text{con } p = 0 \\
 \sum_{\tau \in T^\phi} \sum_{\delta \in \Delta^\tau} \sum_{\alpha \in A^\delta} \sum_{p \in P} v_\alpha x_{\alpha, p} &\geq K_{\text{mín}} & \phi \in \Phi, & & \\
 \sum_{\tau \in T^\phi} \sum_{\delta \in \Delta^\tau} \sum_{\alpha \in A^\delta} \sum_{p \in P} v_\alpha x_{\alpha, p} &\leq K_{\text{máx}} & \phi \in \Phi. & & \text{con } p = 0 \\
 x_{\alpha p} &\in \{0, 1\} & \alpha \in A & & \text{con } p = 0.
 \end{aligned}$$

4.3.1 EVALUACIÓN DEL MODELO MATEMÁTICO

Evaluamos los dos modelos propuestos *EPP-SL* y *EPP-PSL* y los comparamos con el *EPP* (Sanchez Nigenda *et al.*, 2018). Dicho modelo *EPP* no tiene en cuenta ni el estrés ni el ritmo de aprendizaje del alumno, tampoco proporciona una secuencia de actividades, sólo un conjunto de actividades a realizar en el orden que el alumno desee. Sin embargo, usamos el makespan de las soluciones generadas con el *EPP* como punto de referencia y vemos cómo cambian las soluciones al incorporar una secuencia y características subjetivas. La figura 4.4 muestra en el eje horizontal el $K_{\text{mín}}$ esperado por el estudiante y el eje vertical muestra el makespan obtenido por cada algoritmo correspondiente. El valor de $K_{\text{máx}}$ no varía ya que es un problema donde tratamos de minimizar la duración del plan educativo. Podemos notar que los modelos propuestos obtienen resultados similares al modelo *EPP* con la diferencia que nuestros modelos consideran la personalización de la secuencia.

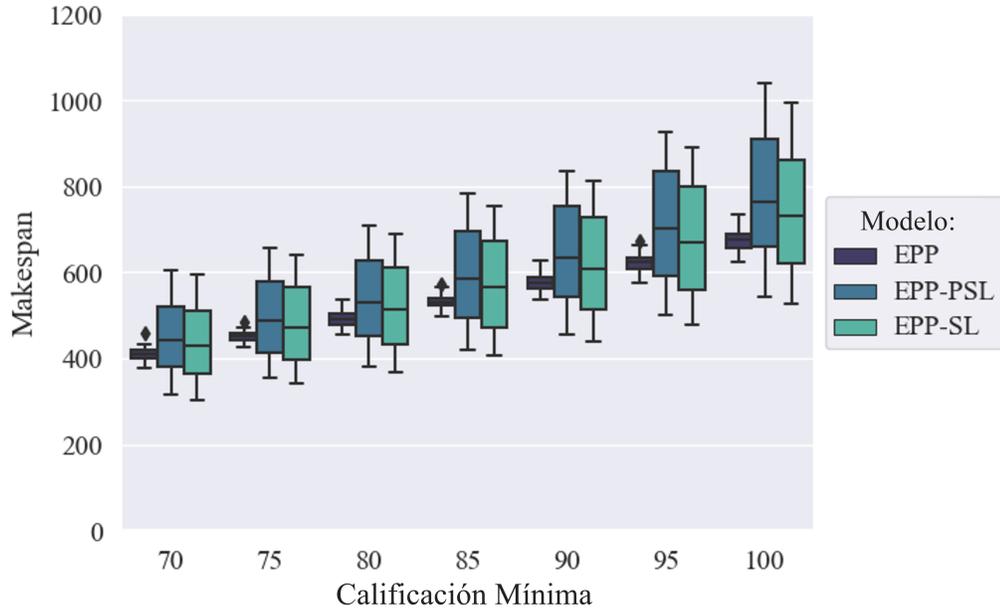


FIGURA 4.4: Rendimiento de los modelos EPP-SL y EPP-PSL respecto al EPP. El eje vertical es el makespan; en el eje horizontal, variamos la puntuación mínima deseada por el alumno.

En la figura 4.5 muestra el impacto que tiene el tener actividades de precedencia en las instancias en el makespan de las soluciones en los diferentes modelos propuestos en comparación con el *EPP*. El eje horizontal representa la cantidad de restricciones de precedencia que pueden tener las actividades (0, 1, 2) Cabe resaltar que solo el 30 % de las actividades presentes en las instancias pueden tener alguna actividad de precedencia, el tener un número mayor podría generar instancias infactibles o con una duración mayor. Podemos notar que el makespan es similar en los tres modelos y variar la cantidad de restricciones de precedencia no impacta en gran medida.

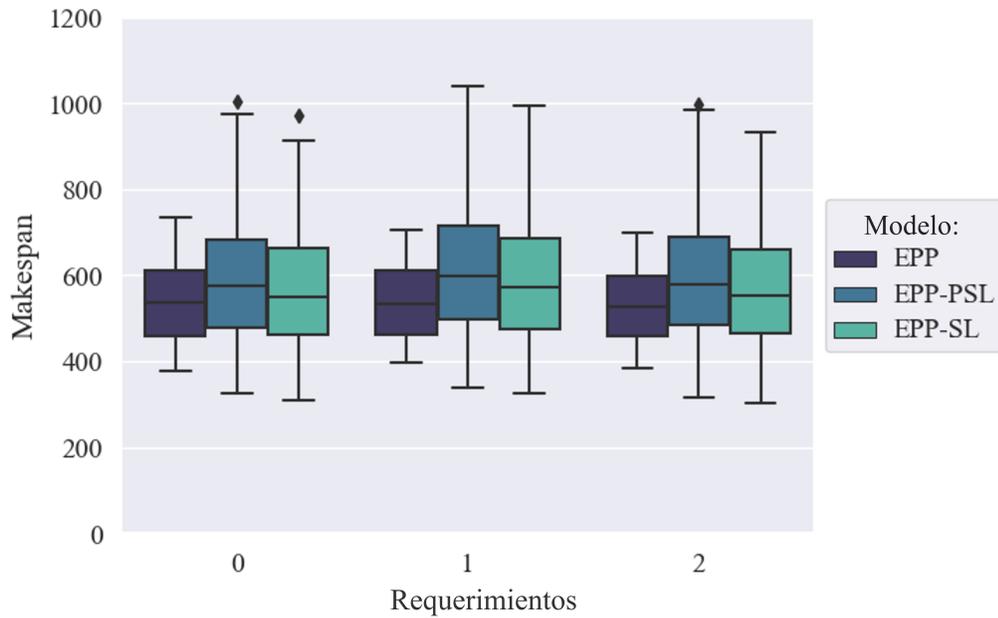


FIGURA 4.5: Impacto de cantidad de actividades de precedencia en las instancias en el makespan de las soluciones de los modelos MILP EPP-SL y EPP-PSL con respecto al EPP. El eje de las ordenadas es el makespan; en el eje de las abscisas, variamos la cantidad de actividades de precedencia.

La figura 4.6 muestra el impacto que puede tener el coeficiente intelectual de un estudiante en el makespan de las soluciones. Se puede observar que cuando se tiene un coeficiente alto el makespan es inclusive menor al modelo *EPP* lo que implica que este tipo de características subjetivas impacta positivamente el desempeño de los estudiantes. El nivel de coeficiente intelectual lo relacionamos con el índice de aprendizaje *LR* presentado anteriormente y propuesto por Biskup (1999).

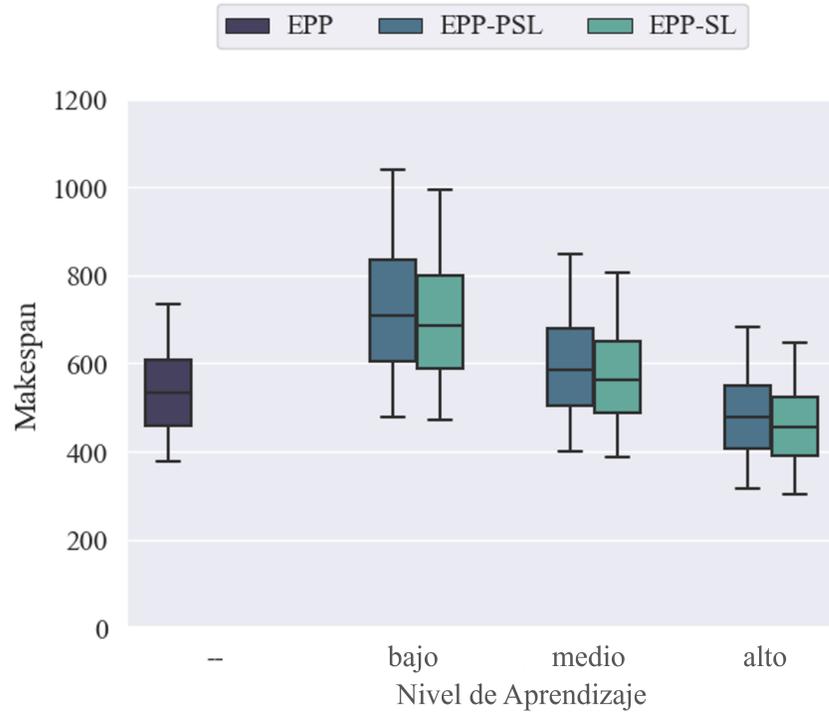


FIGURA 4.6: Impacto de cantidad del coeficiente intelectual de los estudiantes en el makespan de las soluciones de los modelos EPP-SL y EPP-PSL con respecto al EPP. El eje vertical es el makespan; en el eje horizontal, variamos el nivel de coeficiente intelectual.

Podemos observar en las figuras 4.4, 4.5 y 4.6 que el modelo *EPP*, en promedio, produce una mejor duración de la solución. La razón principal es que *EPP* no considera una secuencia de actividades que dependa de factores externos. Una vez incorporamos efectos de aprendizaje a los modelos (figura 4.6), observamos que se ven influenciados positivamente por este factor, lo cual tiene sentido debido a la mayor capacidad de aprendizaje del alumno. La tendencia es que los modelos con mayores tasas de aprendizaje tienen mejores soluciones de makespan en promedio que el *EPP*.

Además, parece que la duración media del plan no depende del número de requisitos de las actividades de precedencia (figura 4.5). Sin embargo, como era de esperar, observamos que cuanto mayor es la puntuación mínima deseada por

el alumno, más considerable es el makespan para todos los modelos (figura 4.4); obsérvese que la varianza en el makespan es mayor para los modelos que tienen en cuenta el estrés y el índice de aprendizaje.

Las figuras 4.7,4.8,4.9,4.10 muestran los tiempos de ejecución del solucionador *GUROBI* para encontrar una solución a nuestros modelos. El tiempo de cómputo del *EPP* es la primer gráfica de barras que debería de visualizarse, sin embargo, no se distingue en las gráficas porque se resuelve en pocos segundos ya que no considera las restricciones de secuencia.

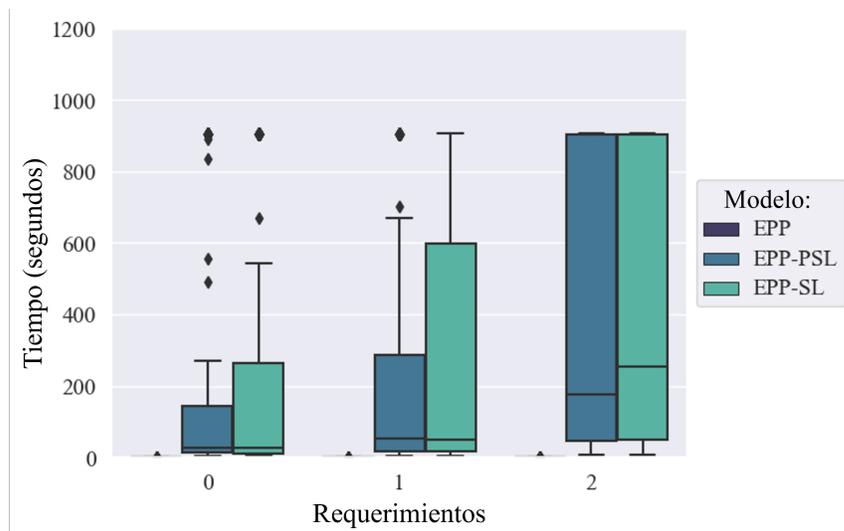


FIGURA 4.7: Tiempos de ejecución de *GUROBI* para los modelos *EPP-PSL* y *EPP-SL*. El eje de las ordenadas representa el tiempo en segundos y el eje de las abscisas el número de actividades predecesoras requeridas.

Es importante mencionar que para todos los modelos se impuso un tiempo límite de 900 segundos por lo que ninguna solución sobrepasa ese tiempo. En la figura 4.7 observamos un impacto significativo en los tiempos de ejecución siempre que existan requisitos de precedencia de actividades. En efecto, el modelo tiene que posicionar varias actividades que están relacionadas entre sí.

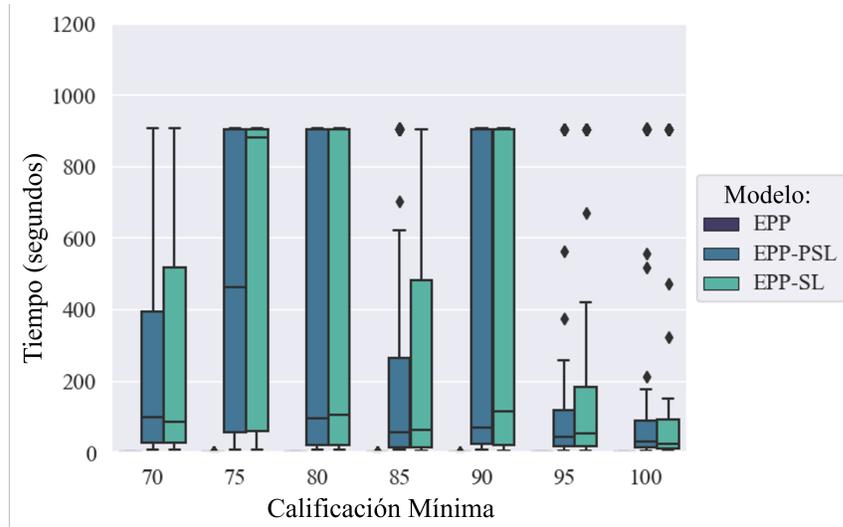


FIGURA 4.8: Tiempos de ejecución de GUROBI para los modelos EPP-PSL y EPP-SL. El eje de las ordenadas representa el tiempo en segundos y el eje de las abscisas es la puntuación mínima deseada por el alumno.

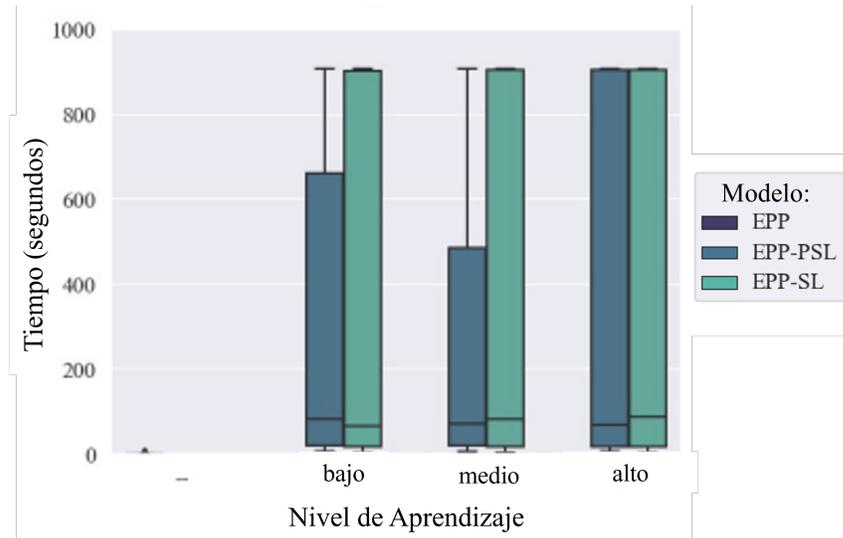


FIGURA 4.9: Tiempos de ejecución de GUROBI para los modelos EPP-PSL y EPP-SL. El eje de las ordenadas representa el tiempo en segundos y el eje de las abscisas es la tasa de aprendizaje del alumno.

La figura 4.8 muestra que cuando un estudiante desea obtener una puntuación alta, este estudiante debe realizar la mayoría de las actividades; por lo tanto, los

modelos se resuelven fácilmente ya que se suprime la selección de actividades, dejando únicamente su programación en la secuencia. Es interesante que la figura 4.9, cuanto más capacidad tiene un alumno, más tiempo lleva resolver los modelos. Esto podría deberse a que se pueden considerar más actividades simultáneamente. Por lo tanto, el modelo debe evaluar más variables de decisión.

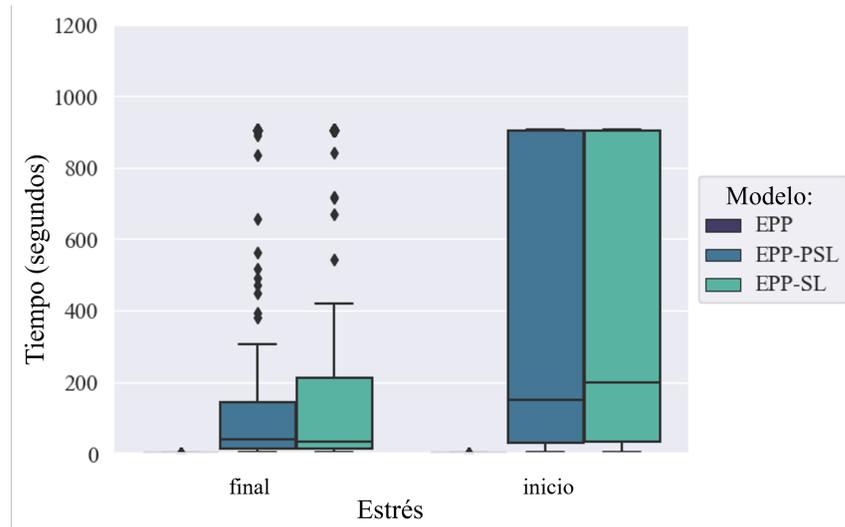


FIGURA 4.10: Tiempos de ejecución de GUROBI para los modelos EPP-PSL y EPP-SL. El eje de las ordenadas representa el tiempo en segundos y el ejes de las abscisas es el foco de estrés al final o al principio del periodo.

En la ultima figura 4.10 observamos que las instancias más difíciles son las que presentan más estrés al principio del periodo, quizá debido a que hay un menor efecto de aprendizaje en las primeras posiciones.

En la figura 4.11, tenemos el *makespan* en el eje vertical, mientras que el índice de aprendizaje está en el eje horizontal. En este caso, comparamos los modelos *EPP*, *EPP – PSL* y *EPP→EPP-PSL*. El último modelo corresponde al *EPP – PSL* el cual, recibe el conjunto de actividades obtenido por *EPP* y simplemente le asigna la posición en la secuencia. Esta figura muestra que, aunque consideremos el mismo conjunto de actividades que el *EPP*, la duración se ve afectada por el estrés y el nivel de aprendizaje del estudiante. Obsérvese que nuestros modelos se comportan

como es de esperar. Una tasa de aprendizaje elevada implica una menor duración de las actividades. Estos resultados también son importantes porque demuestran que aunque el conjunto de actividades sea el mismo, el orden de las actividades cambia, al tener en cuenta el estrés del estudiante durante el periodo académico, lo que causa que la duración media del plan educativo se incremente o se reduzca.

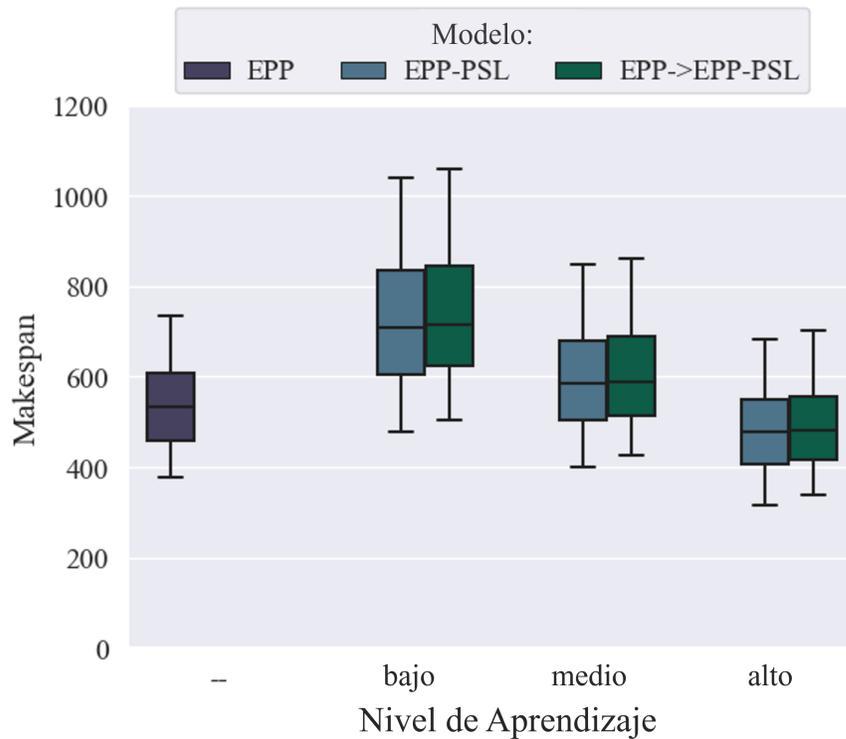


FIGURA 4.11: Comparación entre el modelo EPP, el modelo EPP-PSL y el modelo EPP→EPP-PSL que utiliza el mismo conjunto de actividades obtenidas por el EPP.

Uno de los puntos importantes a investigar en este trabajo fue un caso más cercano a la vida real, donde todos los estudiantes tienen el mismo conjunto de actividades disponibles y esperan alcanzar la misma calificación mínima, sin embargo, la solución para cada uno puede cambiar debido a que cada estudiante tiene diferentes niveles de estrés; ¿Qué tan parecidas son las soluciones entre ellos?. En la figura 4.12, se representa la similitud entre las soluciones obtenidas con el modelo EPP-PSL (paneles de la izquierda) de acuerdo a las actividades que las conforman. Los paneles de la derecha corresponden a la similitud entre la posición de las actividades

en la solución. Los paneles superiores a) y b) corresponden a $k_{\min} = 70$ con una tasa de aprendizaje baja. Los paneles inferiores c) y d) también tienen una tasa de aprendizaje baja pero desean una nota más alta de $k_{\min} = 80$. Cuanto más oscura es la intersección entre las soluciones, más diferentes son. Cada uno de los 20 cuadrados corresponde a 20 estudiantes en comparación con los otros 20 escenarios. Las primeras 10 instancias (renglones y columnas) tienen estrés al principio, mientras que las 10 últimas lo tienen hacia el final del periodo, por ello, hay una marcada división justo en la mitad de la imagen.

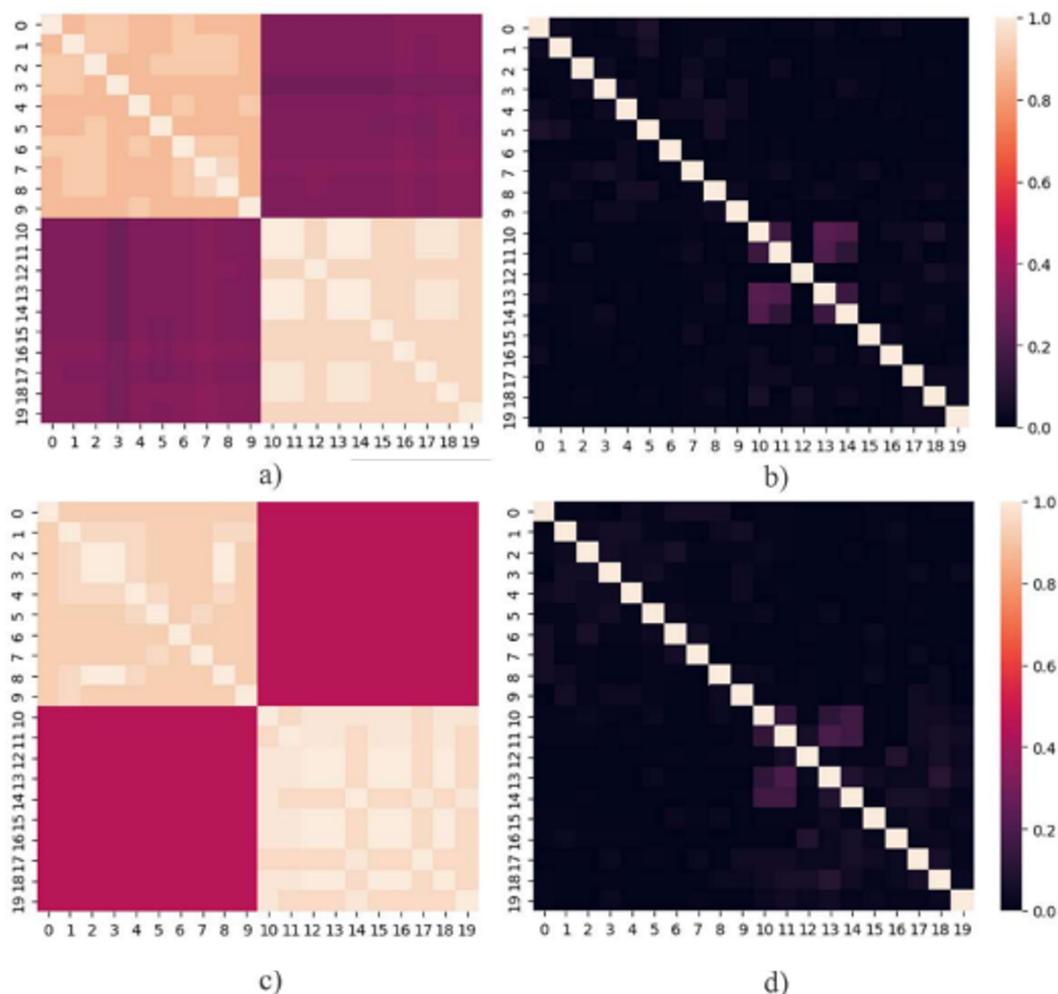


FIGURA 4.12: Similitud entre las soluciones obtenidas resolviendo la misma instancia pero con distintos niveles de estrés con el modelo EPP-PSL. Los paneles de la izquierda muestran la similitud en función de las actividades que conforman las soluciones. Los paneles de la derecha corresponden a la similitud entre la posición de las actividades.

La figura 4.12 muestra la gran similitud entre las actividades que el modelo elige para los estudiantes con estrés al principio o al final del periodo (cuadrados de 10×10 en los paneles de la izquierda de color más claro). Por otro lado, las soluciones de los estudiantes con estrés al principio comparadas con los que tienen estrés al final del periodo tienden a tener menos similitudes (cuadrados 10×10 más oscuros en los paneles de la izquierda). Sin embargo, los paneles de la derecha no muestran casi

ninguna similitud para ningún par de instancias en lo que respecta a la posición de las actividades (zonas oscuras en los paneles de la derecha). Este experimento muestra el impacto del estrés en la planificación de las actividades, dando una mayor importancia a la secuencia de las mismas por encima de la selección.

4.4 VNS-GRASP

Con el fin de obtener soluciones de calidad en menor tiempo se propuso un algoritmo meta-heurístico GRASP-VNS que combina una fase de construcción de soluciones factibles basada en un algoritmo GRASP (Feo y Resende, 1995; Festa y Resende, 2009) y una fase de búsqueda diversificada basada en la Búsqueda por Vecindarios Variables (VNS) (Hansen y Mladenović (2001)) para obtener soluciones de alta calidad en un tiempo computacional corto. La combinación de estas dos meta-heurísticas ha dado lugar a resultados de alta calidad en el campo de la programación de actividades (Báez *et al.*, 2019; Aqil y Allali, 2021; Liu *et al.*, 2018).

Una solución para nuestro problema es una secuencia $S = [\alpha_{[1]}, \alpha_{[2]}, \dots, \alpha_{[n]}]$ donde la notación $\alpha_{[i]}$ es la actividad ejecutada en la posición i (recordemos que puede haber posiciones con actividad nula). Una solución factible S debe satisfacer las restricciones de precedencia entre las actividades (ecuaciones (4.8)), debe garantizar la puntuación mínima (K_{min}) y máxima (K_{max}) (ecuaciones (4.10) y (4.11) e incluir todas las actividades obligatorias en la secuencia (ecuación (4.7)).

El algoritmo 1 es el pseudocódigo de nuestro algoritmo GRASP-VNS. El primer ciclo (pasos 2 a 6) es el algoritmo GRASP que construye en cada iteración una solución factible S con la función *constructivo()* descrita más adelante. A continuación, S se mejora mediante una búsqueda local, haciendo un seguimiento de la mejor secuencia en términos de la suma de las duraciones de las actividades incluidas en S . El VNS se encuentra en los pasos 8 a 16. El algoritmo perturba la solución actual con la función *perturbar(S)* en el paso 11, y cada vecindad k intenta mejorarla. Este

proceso termina cuando se alcanza el número máximo de iteraciones VNS.

Algoritmo 1 GRASP-VNS($Iteraciones_{GRASP}, Iteraciones_{VNS}, \beta$)

$mejorSeq \leftarrow constructivo(\beta)$

mientras no se alcance $Iteraciones_{GRASP}$ **hacer**

$S \leftarrow constructivo(\beta)$

$S \leftarrow BusquedaLocalGRASP(S)$

si $makespan(S) \leq makespan(mejorSeq)$ **entonces**

$mejorSeq \leftarrow S$

fin

fin

$S \leftarrow mejorSeq$

mientras no se alcance $Iteraciones_{VNS}$ **hacer**

$k = 0$

mientras $k \leq NumVecindarios$ **hacer**

$S \leftarrow perturba(S)$

$S \leftarrow BusquedaLocalVNS(S, k)$

si $makespan(S) \leq makespan(mejorSeq)$ **entonces**

$mejorSeq \leftarrow S$

en otro caso

$k \leftarrow k + 1$

fin

fin

fin

devolver $mejorSeq$

La fase constructiva GRASP se muestra en el Algoritmo 2. A partir de una solución de secuencia vacía, la función $constructivo(\beta)$ agrega aleatoriamente a S una actividad de la Lista Restringida de Candidatos (LRC) (paso 9) sólo si no viola ninguna restricción de precedencia. En caso contrario, la actividad se elimina de la LRC . En cualquier caso, la siguiente actividad de $lista$ se añade a la LRC . Tenga en cuenta que la función de prioridad $g(\alpha)$ del paso 3 hace una lista no creciente de las actividades. Tiene en cuenta la proporción estimada de duración de la actividad y su

importancia en la nota final. Sin embargo, para diversificar las soluciones obtenidas, añadimos a esta función el estrés s_p de la posición p , que en este caso no está necesariamente relacionada con la secuencia.

Algoritmo 2 *constructivo*(β)

$p = 0$

para $\alpha \in A$ **hacer**

 | $g(\alpha) = (\frac{d_\alpha}{d_{max}} / \frac{v_\alpha}{v_{max}} + s_p), p = p + 1$

fin

lista \leftarrow ordena las $\alpha \in A$ en orden decreciente con respecto a sus valores en $g(\alpha)$

LRC se conforma de las primeras β actividades de *lista*

$S \leftarrow [\alpha_{[i]} = \emptyset]$ para $i = 1, \dots, n$; ,secuencia vacía

mientras $K_{min} \leq \text{puntaje}(S) \leq K_{max}$ o $LRC = \emptyset$ **hacer**

 | $S \leftarrow S \cup \text{aleatorio}(LRC)$ solo si las restricciones de precedencia son válidas
 | actualiza *LRC*

fin

regresa S

El algoritmo 3 es la búsqueda local GRASP. La actividad α^* de la solución actual se selecciona aleatoriamente. Supongamos que esta actividad no es una restricción de precedencia para otras actividades $\alpha^* \notin H(\alpha)$ en la solución (paso 3). De las actividades que no están en la solución actual, elegimos la primera actividad α' (paso 4) y la insertamos en el lugar α^* . Si hemos obtenido una solución factible con igual o mejor makespan, actualizamos la solución actual S y devolvemos la secuencia. En caso contrario, probamos con otra actividad α' hasta alcanzar el límite *Iteraciones* BL_{GRASP} .

Algoritmo 3 *BusquedaLocalGRASP(S)*

```

mientras no se alcance IteracionesBLGRASP hacer
  |  $S' \leftarrow S$ ; crea una copia de la secuencia
  | elige un  $\alpha^* \in S'$  aleatorio tal que  $\alpha^* \notin H(\alpha)$  para todo  $\alpha \in S$ 
  | para  $\alpha' \in A \setminus S$  hacer
  | | remueve  $\alpha^*$  inserta  $\alpha'$ 
  | | si  $S'$  es factible y  $makespan(S') \leq makespan(S)$  entonces
  | | |  $S \leftarrow S'$ 
  | | | devolver  $S$ 
  | | en otro caso
  | | | remueve  $\alpha'$  de  $S'$  y reinserta  $\alpha^*$ 
  | | fin
  | fin
fin
devolver  $S$ 

```

La función $shaking(S)$ del Algoritmo 1 (paso 10) es una fase de diversificación para evitar mínimos locales. Esta función elige aleatoriamente una actividad de la solución actual S y la inserta aleatoriamente en otra posición de la secuencia. Si este movimiento produce una solución no factible, entonces lo deshacemos. La función $localSearchVNS(S, k)$ es la búsqueda local de la etapa VNS (paso 11) que considera tres vecindarios diferentes ($k = 1, 2, 3$). El objetivo de la VNS es perturbar las posiciones de las actividades en la solución para reducir el tiempo de espera. De hecho, cada posición tiene un nivel de estrés diferente. Por lo tanto, cambiar las posiciones de las actividades puede conducir a una mejor secuencia final. El primer vecindario se obtiene intercambiando la posición de dos actividades en la secuencia actual. La segunda vecindad intercambia tres actividades: la primera actividad pasa a la posición de la tercera actividad, la segunda actividad pasa a la posición de la primera actividad y la tercera actividad pasa a la posición de la segunda actividad. El tercer vecindario se obtiene intercambiando cuatro actividades: la primera actividad va a la última posición de actividad, la segunda actividad va a la primera, la tercera

actividad va a la segunda y la cuarta actividad va a la tercera posición, un ejemplo de dichos vecindarios se puede ver en la figura 4.13. La mejor solución obtenida de cada vecindario se considera la nueva solución actual. Si no se ha encontrado ninguna solución mejorada, se construye el siguiente vecindario como indica el Algoritmo 1.

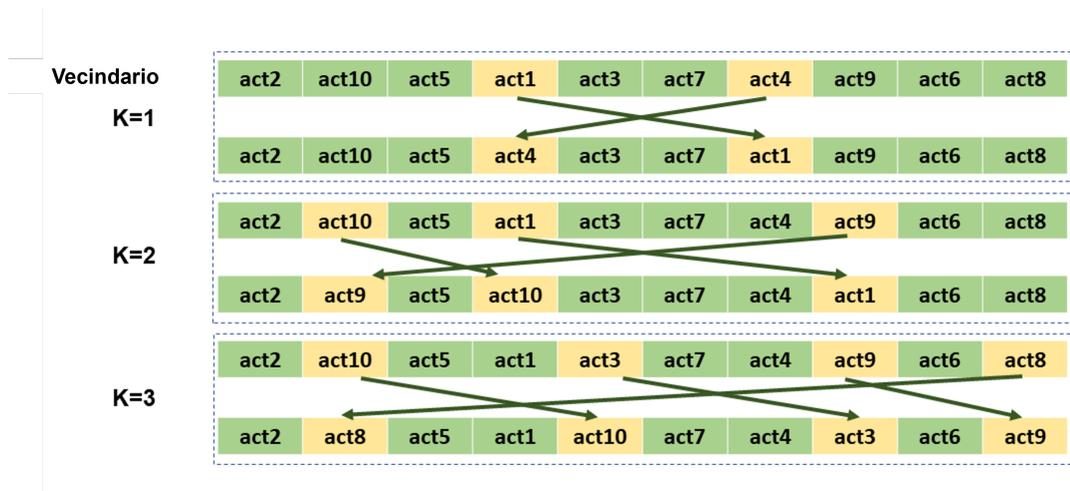
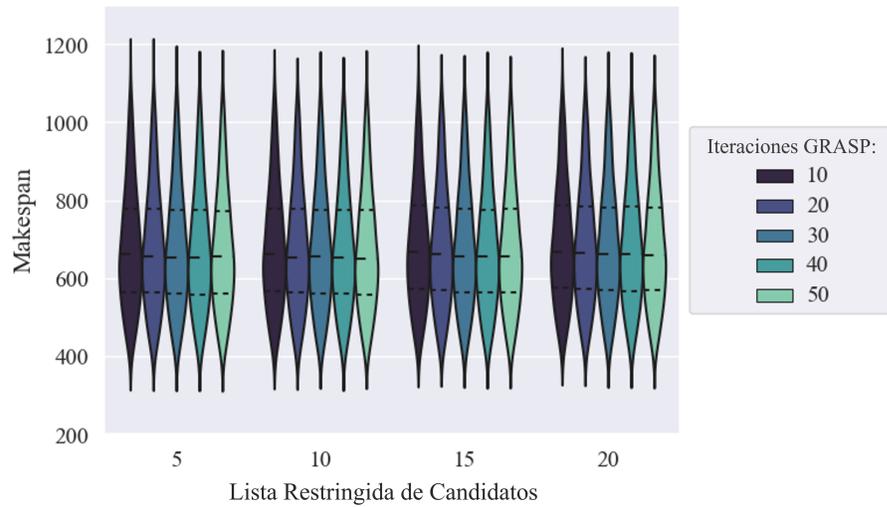


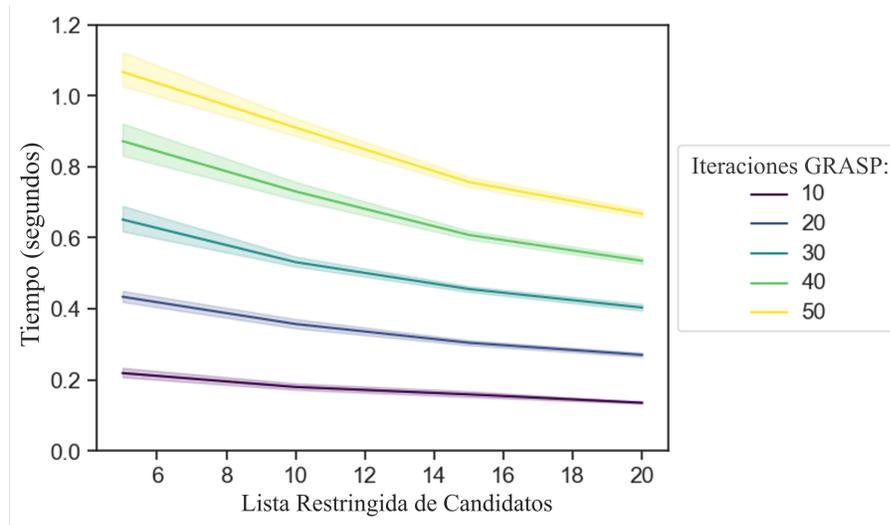
FIGURA 4.13: Ejemplo de los vecindarios de la búsqueda local del VNS.

4.4.1 CALIBRACIÓN DE PARÁMETROS

Para evaluar nuestro algoritmo *GRASP – VNS* propuesto, comenzamos con la calibración de la fase *GRASP*. En la figura 4.14, variamos el número máximo de iteraciones del algoritmo *GRASP*. La variación del número de iteraciones o del tamaño de la *LRC* en la fase *GRASP* no reflejó un cambio significativo en la duración del plan. Sin embargo, el gráfico b) como era de esperarse muestra que estos dos factores influyen en el tiempo de cómputo. Cuanto mayor sea el tamaño del *LRC*, menos tiempo tardará el *GRASP* en converger a un mínimo local, esto debido a que entre más grande sea la *LRC* puede caer más fácilmente en elementos aleatorios y no necesariamente buenos.



(a)

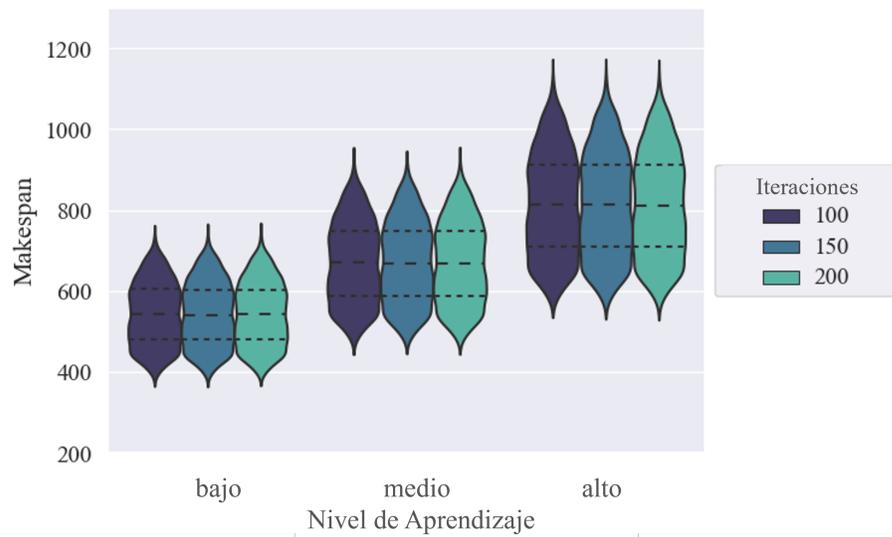


(b)

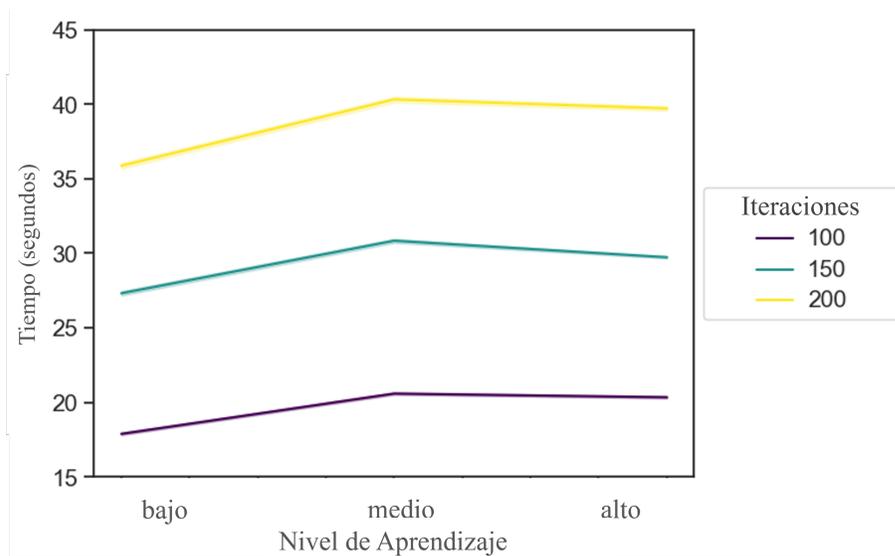
FIGURA 4.14: El eje de abscisas representa la cantidad de elementos en la *LRC*, mientras que el eje de ordenadas corresponde al makespan de las soluciones en el gráfico a) y al tiempo computacional en el gráfico b) de acuerdo a la cantidad de iteraciones del algoritmo *GRASP*.

Para el algoritmo *VNS* solo variamos la cantidad de iteraciones que realizaría. Como podemos ver en la figura 4.15 tal variación, no parece afectar mucho el makespan de las soluciones, esto quizá debido a que la principal función del *VNS*

es modificar la solución para encontrar la mejor secuencia de actividades, pero sin modificar el conjunto ya seleccionado por el GRASP.



(a)



(b)

FIGURA 4.15: El eje de abscisas representa el índice de aprendizaje, mientras que el eje de ordenadas corresponde al makespan de las soluciones en el gráfico a) y al tiempo computacional en el gráfico b) de acuerdo a la cantidad de iteraciones del algoritmo *VNS*.

Una vez calibrados los parámetros de nuestro algoritmo procedemos a evaluar el desempeño de forma independiente como combinados. Fijamos la cantidad de iteraciones del *VNS* en 100 ya que mas haya de esa cantidad solo incrementa el tiempo computacional, la *LRC* del *GRASP* lo fijamos en 20 para permitir cierta diversidad en las posibles actividades de la solución sin caer en la aleatoriedad y 50 iteraciones. Tras la fase *GRASP*, incluimos una búsqueda de vecindario variable *VNS*. En la figura 4.16, mostramos el porcentaje de mejora con respecto al makespan (eje de ordenadas) al incorporar el *VNS* en comparación con la solución obtenida con el *GRASP* en relación con el número de requisitos de precedencia de las actividades.

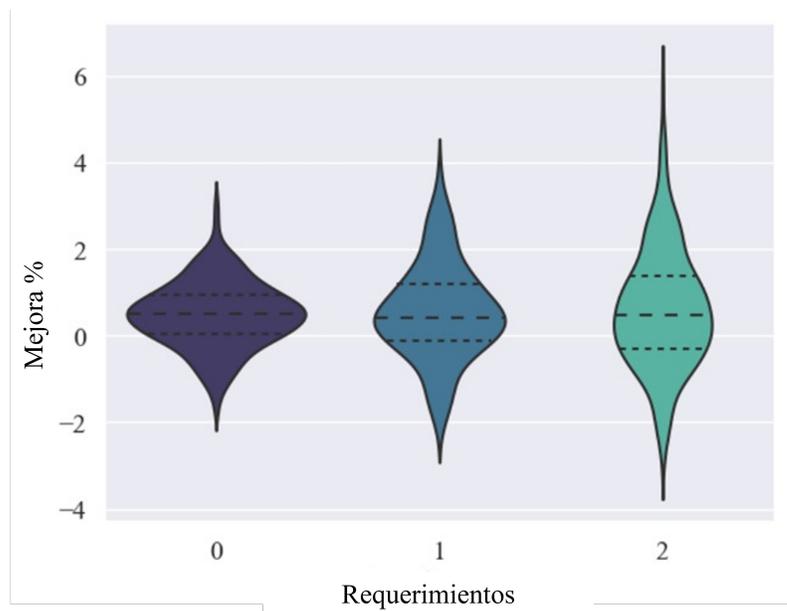


FIGURA 4.16: Porcentaje de mejora del *VNS* comparado con la solución *GRASP* en relación el número de requisitos de precedencia de las actividades.

En la figura 4.17, en el eje de las ordenadas mostramos el porcentaje de mejora al incorporar el *VNS* en comparación con la solución obtenida con el *GRASP* en relación con la calificación mínima esperada, vemos que no hay mucha diferencia en cuanto al porcentaje de mejora, sin embargo, los datos están menos dispersos cuando la calificación esperada es más alta.

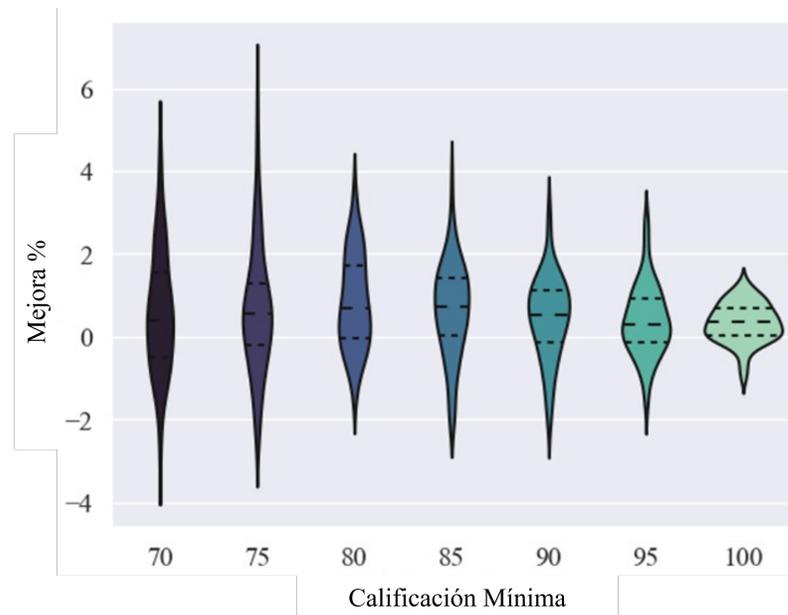


FIGURA 4.17: Porcentaje de mejora del *VNS* comparado con la solución *GRASP* en relación a la calificación mínima esperada k_{\min} .

En la figura 4.18, mostramos el porcentaje de mejora en relación con el índice de aprendizaje del estudiante. En general, observamos que el porcentaje de mejora se sitúa en torno al 1%, pero en algunos casos llega hasta el 5%. Las mejoras son más visibles cuando se tiene un alto índice de aprendizaje.

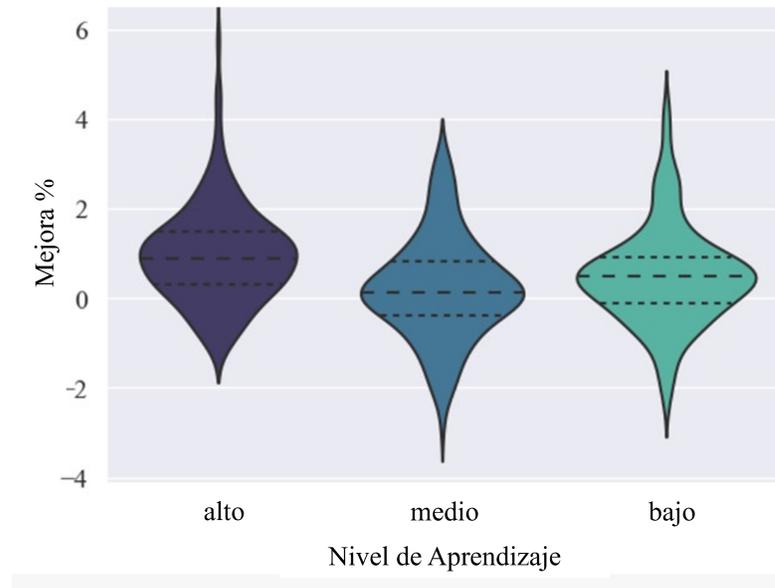


FIGURA 4.18: Porcentaje de mejora del *VNS* comparado con la solución *GRASP* en relación con índice de aprendizaje de los estudiantes.

Considerando cada parte del algoritmo de forma independiente, la fase de construcción de *GRASP* obtiene soluciones con un desfase medio de 35 % respecto al valor óptimo. Este valor se reduce en la fase de búsqueda local, en la que encontramos soluciones con un desfase medio inferior a 26 %. Finalmente, combinando *GRASP* con las fases *VNS*, el gap de las soluciones encontradas disminuye en otros 10 – 15 %, obteniendo soluciones con un *gap* medio inferior a 10 % respecto al valor óptimo obtenido con los modelos de optimización.

4.4.2 EVALUACIÓN DEL *GRASP – VNS* Y LOS MODELOS MATEMÁTICOS

Para la evaluación tanto de los modelos matemáticos *EPP–LS* y *EPP–LSP*, como del algoritmo *GRASP – VNS* se utilizaron 30 instancias con datos artificiales. Cada instancia generada tiene $|\Phi| = 2$ materias, $T^\phi = 4$ temas, $|\Delta^{\phi\tau}| = 8$ subtemas, y $|A| = 88$ actividades que representan una retícula educativa, para $\phi \in \Phi$, y $\tau \in T^\phi$.

Así, cada subtema tiene 11 actividades. Cada actividad $\alpha \in A^{\phi\tau\delta}$ tiene un valor de grado distribuido uniformemente entre:

$$\left[\frac{100}{|A|} \sum_{\phi \in \Phi} \sum_{\tau \in T^\phi} |\Delta^{\phi\tau}|, 6 + \frac{100}{|A|} \sum_{\phi \in \Phi} \sum_{\tau \in T^\phi} |\Delta^{\phi\tau}| \right] = [9, 15], \quad (4.12)$$

para $\phi \in \Phi$, $\tau \in T^\phi$, y $\delta \in \Delta^{\phi\tau}$. La duración estimada de las actividades d_α es uniformemente distribuida entre:

$$\left[-6 + \frac{100}{|A|} \sum_{\phi \in \Phi} \sum_{\tau \in T^\phi} |\Delta^{\phi\tau}|, 6 + \frac{100}{|A|} \sum_{\phi \in \Phi} \sum_{\tau \in T^\phi} |\Delta^{\phi\tau}| \right] = [3, 15]. \quad (4.13)$$

para $\alpha \in A^{\phi\tau\delta}$, $\phi \in \Phi$, $\tau \in T^\phi$, y $\delta \in \Delta^{\phi\tau}$.

Las calificaciones mínima y máxima deseadas por un alumno al final de la asignatura son $K_{\min} = 70$ y $K_{\max} = 100$, siendo 70 la calificación mínima para aprobar la asignatura. Menos del 30% de estas actividades tienen una o dos actividades precedentes. A diferencia de trabajos anteriores, nuestras instancias no tienen actividades obligatorias, ya que son más fáciles de resolver al estar ya fijadas varias decisiones y solo tener que asignarles un orden dentro del plan educativo. No obstante, nuestros códigos y modelos pueden manejar actividades obligatorias.

En cuanto al estrés, establecemos un intervalo uniforme entre $[0, 0.5]$, donde 0.5 es el valor de estrés más alto que puede tener un estudiante en un semestre. Nótese que no estamos considerando casos excepcionales que tendrían un estrés mayor. Generamos aleatoriamente 30 instancias; 15 tienen mayor estrés en el primer tercio del periodo académico, mientras que el resto en el último tercio. La tasa de aprendizaje de cada alumno está en $[0.95, 0.99, 1.03]$; cuanto mayor sea el valor, mayor capacidad de aprendizaje. Evaluamos el conjunto de 30 instancias con cada una de las tasas de aprendizaje, dando un total de 90 casos de evaluación.

4.5 COMPARACIÓN DE RESULTADOS

Las figuras 4.19, 4.20, 4.21 comparan la calidad de las soluciones en términos de makespan (eje de ordenadas) y mientras que las figuras 4.22 y 4.23 comparan

las soluciones respecto al tiempo de ejecución del modelo exacto comparado con la heurística *GRASP – VNS*.

Como era de esperar, podemos observar en las figuras 4.19, 4.20 y 4.21 que el algoritmo meta-heurístico (*GRASP – VNS*) obtienen un mayor makespan, pero las soluciones siguen siendo de alta calidad y tienden a estar cerca de las soluciones óptimas.

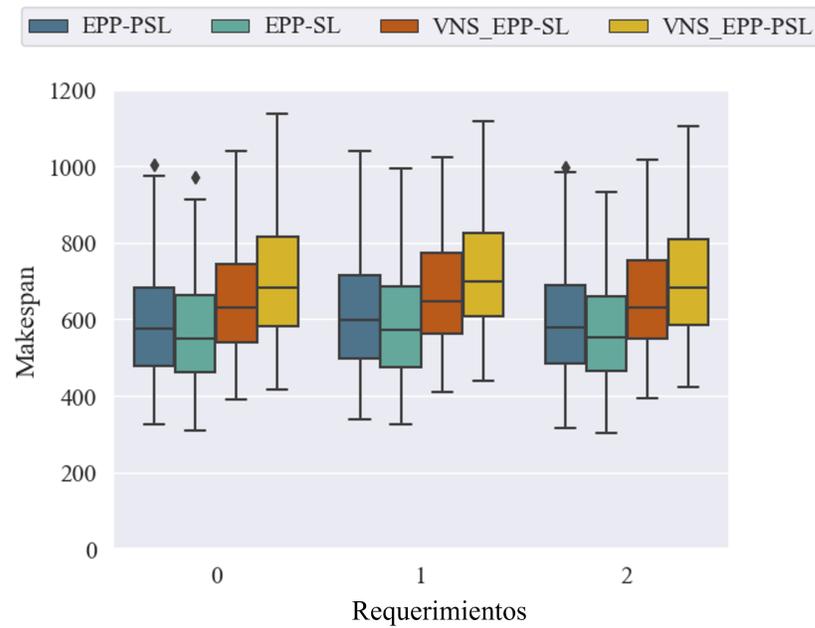


FIGURA 4.19: Calidad en términos de makespan (eje de ordenadas) del modelo exacto comparado con el *GRASP – VNS* en relación al número de los requisitos de la actividad de precedencia (eje de abscisas).

Podemos notar en la figura 4.19 que a pesar de incrementar el número de requisitos de precedencia en las actividades, el makespan de las soluciones parece no incrementar. Y como se esperaba, en la figura 4.20 vemos que al incrementar la calificación mínima esperada, si se incrementa el makespan, esto debido al aumento en el número de actividades en las soluciones, a mayor calificación esperada, se necesita una mayor cantidad de actividades.

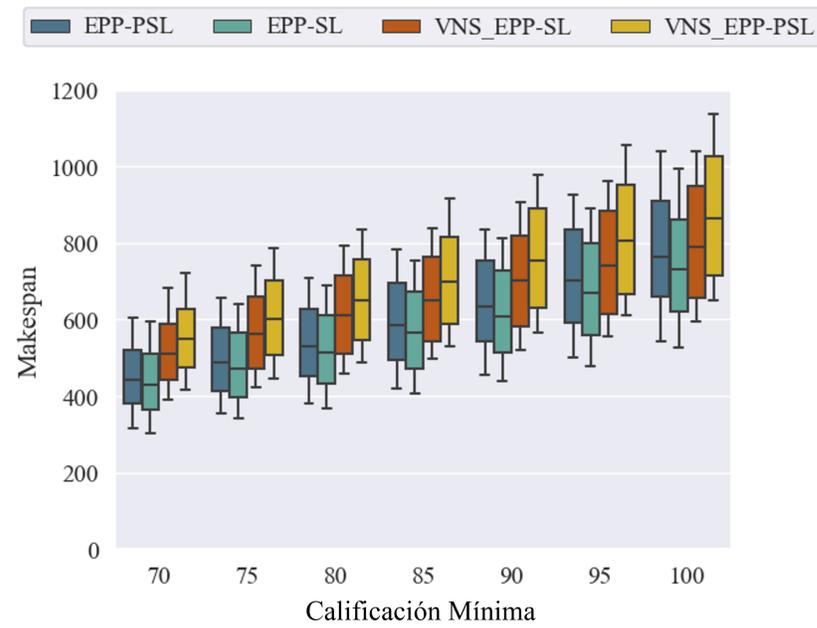


FIGURA 4.20: Calidad en términos de makespan (eje de ordenadas) del modelo exacto comparado con el *GRASP – VNS* en relación con el k_{\min} esperado.

Por último vemos que en la figura 4.21 el índice de aprendizaje se muestra como un factor importante en cuanto al *makespan* de las soluciones, a un mejor índice de aprendizaje un menor *makespan*.

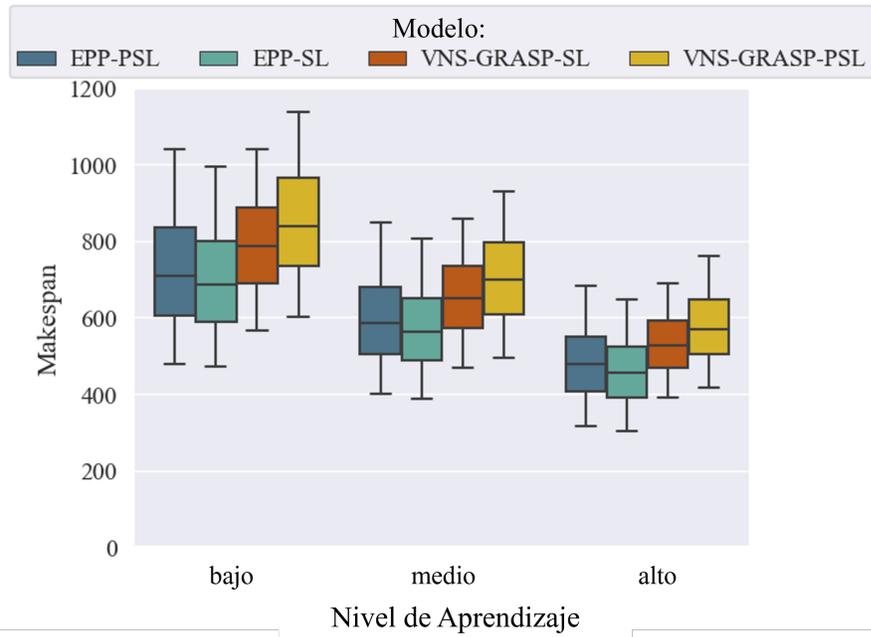


FIGURA 4.21: Calidad en términos de makespan (eje de ordenadas) del modelo exacto comparado con el *GRASP – VNS* en relación con el índice de aprendizaje.

Las figuras 4.22,4.23,4.24,4.25 muestran el tiempo computacional (eje de las ordenadas) de nuestra meta-heurística en segundos, mientras que para el método exacto, se puede tardar una hora entera en calcular el pareto de soluciones. Por tanto, aunque las soluciones de la meta-heurística *GRASP – VNS* no sean tan buenas como las de los métodos óptimos, la ganancia obtenida en tiempo computacional es enorme. No hay una ganancia visible para la meta-heurística en tiempo computacional en relación con las diferentes características que estamos trazando, pero la diferencia en el tiempo empleado por el método exacto valida la importancia de nuestro *GRASP – VNS*.

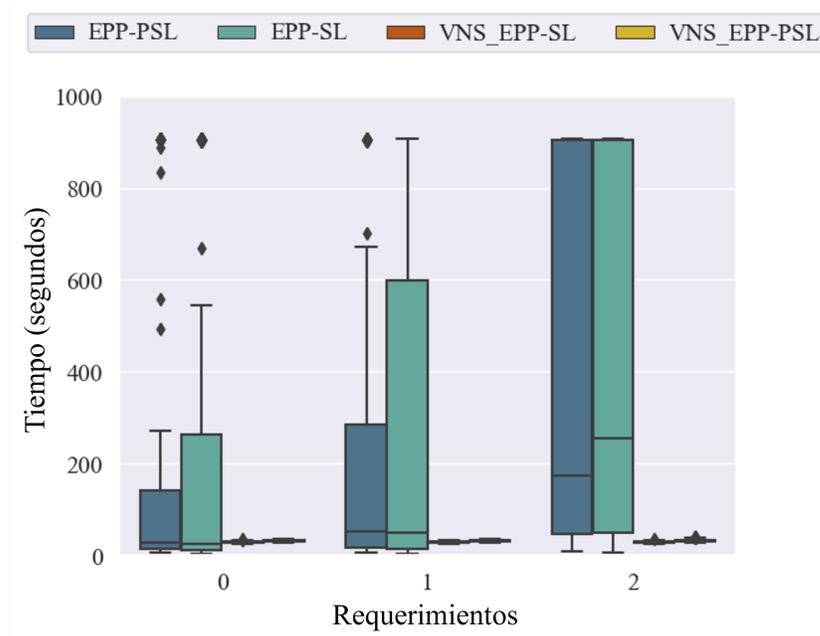


FIGURA 4.22: Tiempo de ejecución (eje de ordenadas) en segundos de los modelos exactos en comparación con *GRASP-VNS* con respecto al número de los requisitos de precedencia de las actividades (eje de abscisas).

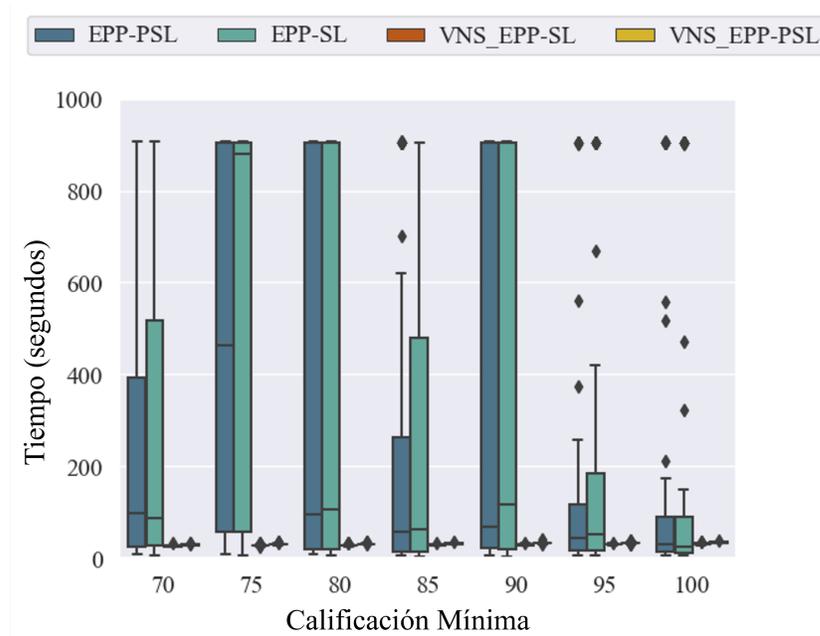


FIGURA 4.23: Tiempo de ejecución de los modelos exactos en comparación con la meta-heurística *GRASP-VNS* con respecto a la calificación mínima esperada.

Podemos notar en la figura 4.22 que el incremento en la cantidad de requerimientos de precedencia de las actividades muestra un incremento en el tiempo computacional de los modelos exactos, esto debido a que se generan cadenas de actividades las cuales se tienen que secuenciar en la mejor posición para minimizar el *makespan* de la solución. En la figura 4.23 vemos como al incrementar la calificación mínima esperada el tiempo de computo disminuye en las calificaciones más altas, esto debido a que este caso, se agregan casi todas las actividades, por lo que la cantidad de actividades opcionales dentro de la solución disminuye. En ambos casos los modelos exactos tienen un tiempo computacional mucho mayor con respecto a la meta-heurística.

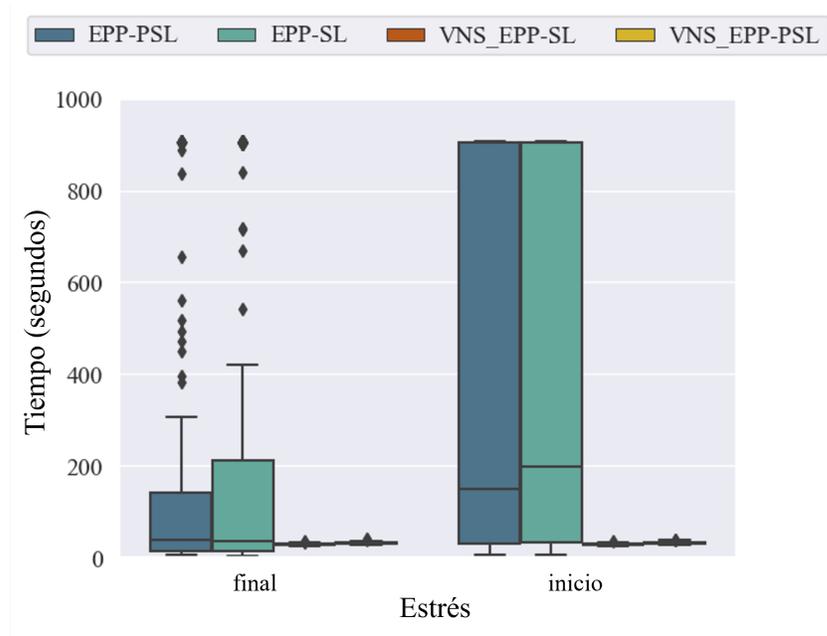


FIGURA 4.24: Tiempo de ejecución de los modelos exactos en comparación el *GRASP* – *VNS* con respecto al foco del estrés, al inicio o al final del periodo.

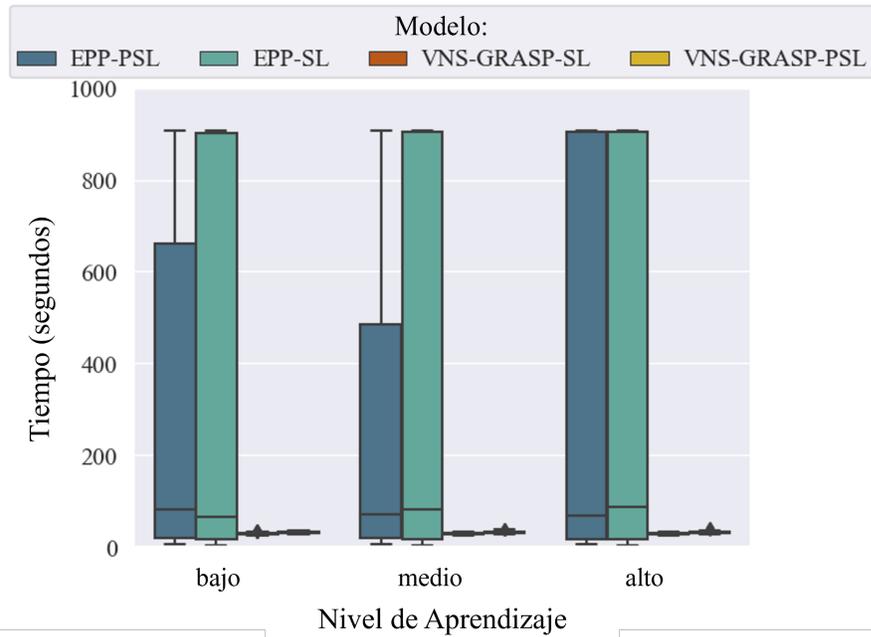


FIGURA 4.25: Tiempo de ejecución de los modelos exactos en comparación con el *GRASP – VNS* con respecto al índice de aprendizaje de los estudiantes.

Supongamos que un estudiante en concreto desea obtener una frontera de Pareto de la figura 4.26. El estrés de este alumno suele estar al final del periodo académico, y su ritmo de aprendizaje es alto. En el eje y, tenemos los valores k_{\min} correspondientes a la nota mínima que espera obtener al final del periodo. En el eje de abscisas tenemos el periodo académico dividido en posiciones que representan la secuencia de actividades que debe realizar el alumno. El gráfico superior tiene una escala de colores que muestra las actividades de aprendizaje con menor tiempo de procesamiento (las más oscuras), mientras que el gráfico inferior muestra las actividades en función de su calificación (las más oscuras son las de menor valor).

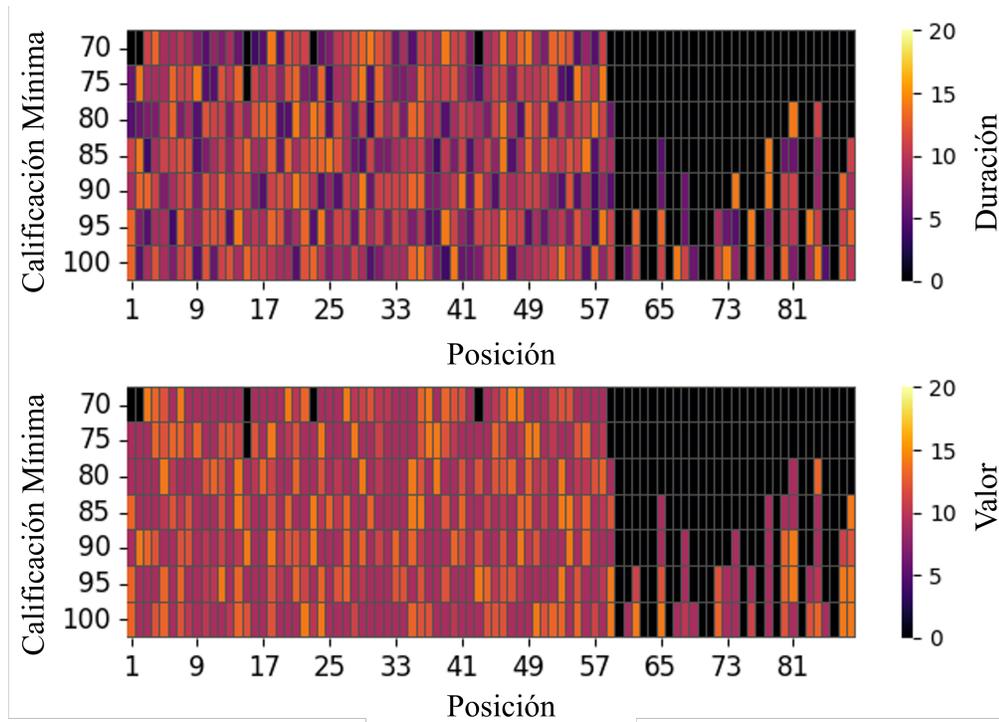


FIGURA 4.26: La compensación de Pareto para una estudiante con carga de estrés tiende a ser al final del periodo académico y su tasa de aprendizaje es alta.

La figura 4.26 muestra que si el alumno desea obtener una nota más alta, debe realizar más actividades. Sin embargo, como tiende a estar más estresada al final del periodo, la mayoría de las actividades se programan al principio del periodo académico. Las posiciones oscuras equivalen a tiempo libre con la garantía de obtener una nota mínima. Este gráfico indica la importancia de considerar el estrés en la generación de trayectorias de aprendizaje ya que permite ajustar y personalizar los planes educativos a las necesidades de los estudiantes.

Podemos concluir que el *GRASP – VNS* obtiene soluciones de alta calidad y es muy eficiente en términos de tiempo computacional. En la tabla 4.1 podemos ver un acumulado de los resultados promedio para las instancias utilizadas en este trabajo, el promedio se obtuvo sumando el valor del *makespan* obtenido para cada calificación mínima de 70 hasta 100 para cada instancia. Lo mismo hicimos pero con el tiempo computacional que se muestra en segundos y con la cuarta y quinta

columna que corresponden al *GRASP – VNS*. Las últimas dos columnas son el error porcentual, el cual obtuvimos de la siguiente manera:

$$Error \% = \frac{H_i - O_i}{O_i} * 100 \quad (4.14)$$

Donde H_1 corresponde al valor del makespan o tiempo computacional para la instancia i con el *GRASP – VNS* y O_i corresponde al valor del makespan o tiempo computacional para la instancia i con el *EPP – PSL*.

Podemos notar que a excepción de tres casos que se resaltan en color negro, el *VNS – GRASP* es por mucho mejor en cuanto a tiempo computacional, y en la mayoría de los casos la diferencia con respecto al valor óptimo es menor al 20 %.

TABLA 4.1: Resultados promedio para el *GRASP – VNS* y el *EPP – PSL*.

| Instancia | EPP-PSL | | VNS-GRASP | | Error % | |
|-----------|----------|--------|-----------|--------|----------|---------------|
| | Makespan | Tiempo | Makespan | Tiempo | Makespan | Tiempo |
| f 1 0 | 484.04 | 269.30 | 584.46 | 30.90 | 20.75 | 88.53 |
| f 1 1 | 516.89 | 272.58 | 594.36 | 32.03 | 14.99 | 88.25 |
| f 1 2 | 517.86 | 65.01 | 592.05 | 32.50 | 14.33 | 50.00 |
| f 2 0 | 479.98 | 27.98 | 575.61 | 31.17 | 19.92 | -11.39 |
| f 2 2 | 510.13 | 502.09 | 587.30 | 35.36 | 15.13 | 92.96 |
| f 3 0 | 504.32 | 216.06 | 595.34 | 31.23 | 18.05 | 85.55 |
| f 3 1 | 503.32 | 157.71 | 578.43 | 32.27 | 14.92 | 79.54 |
| f 3 2 | 483.32 | 195.33 | 547.86 | 32.46 | 13.35 | 83.38 |
| f 4 0 | 486.52 | 307.11 | 569.47 | 31.96 | 17.05 | 89.59 |
| f 4 1 | 538.84 | 16.55 | 621.44 | 32.51 | 15.33 | -96.44 |
| f 4 2 | 512.40 | 146.56 | 606.32 | 31.75 | 18.33 | 78.34 |
| f 5 0 | 461.45 | 310.57 | 546.15 | 31.49 | 18.36 | 89.86 |
| f 5 1 | 525.46 | 286.19 | 589.63 | 32.19 | 12.21 | 88.75 |
| f 5 2 | 489.37 | 301.60 | 566.79 | 31.79 | 15.82 | 89.46 |
| i 1 0 | 447.65 | 285.05 | 532.81 | 31.71 | 19.02 | 88.88 |
| i 1 1 | 493.71 | 416.22 | 567.29 | 34.19 | 14.90 | 91.79 |
| i 1 2 | 493.37 | 501.98 | 587.51 | 32.02 | 19.08 | 93.62 |
| i 2 0 | 463.08 | 274.18 | 568.98 | 31.66 | 22.87 | 88.45 |
| i 2 1 | 472.38 | 432.21 | 575.48 | 31.49 | 21.83 | 92.72 |
| i 3 0 | 448.29 | 28.91 | 538.51 | 32.46 | 20.12 | -12.30 |
| i 3 1 | 494.10 | 438.29 | 600.25 | 31.51 | 21.48 | 92.81 |
| i 3 2 | 425.01 | 467.17 | 543.91 | 31.26 | 27.98 | 93.31 |
| i 4 0 | 467.57 | 540.35 | 569.52 | 31.84 | 21.80 | 94.11 |
| i 4 1 | 456.98 | 186.43 | 577.17 | 31.46 | 26.30 | 83.13 |
| i 4 2 | 463.58 | 378.11 | 555.53 | 31.95 | 19.83 | 91.55 |
| i 5 0 | 495.89 | 79.52 | 605.52 | 32.08 | 22.11 | 59.65 |
| i 5 1 | 462.87 | 344.85 | 570.64 | 31.71 | 23.28 | 90.81 |
| i 5 2 | 449.39 | 807.29 | 545.30 | 31.09 | 21.34 | 96.15 |

CAPÍTULO 5

CONCLUSIONES

En este trabajo, estudiamos la generación de rutas de aprendizaje incorporando características subjetivas como el estrés y efectos de aprendizaje (Por ejemplo, la capacidad intelectual) de los estudiantes, los cuales nos permiten generar rutas de aprendizaje personalizadas para prevenir problemas académicos.

Tomamos inspiración de problemas teóricos de secuenciación de actividades que consideran efectos de deterioro y aprendizaje en los tiempos de procesamiento de los trabajos, propusimos dos modelos matemáticos y un algoritmo heurístico *GRASP – VNS* para calcular las trayectorias de aprendizaje con estrés y efectos de aprendizaje. Los modelos representan el impacto del estrés y el efecto del aprendizaje en la duración de las actividades para minimizar el makespan del plan educativo para alcanzar una calificación mínima.

Nuestros resultados experimentales mostraron la importancia de tener en cuenta las características de los estudiantes como el estrés y los efectos del aprendizaje a la hora de decidir las actividades a realizar. En concreto, demostramos que a medida que aumenta el ritmo de aprendizaje del alumno, el promedio de makespan de las soluciones disminuye debido a la mayor capacidad de aprendizaje del alumno. Además, nuestros resultados indican que considerar diferentes ordenaciones de las actividades de aprendizaje tiende a producir planes de makespan alternativos.

Por último, demostramos que el algoritmo GRASP-VNS propuesto calcula soluciones de buena calidad en una fracción del tiempo de los modelos exactos, lo que nos permite generar un frente de Pareto de trayectorias de aprendizaje. Disponer de un frente de Pareto de alternativas bajo diferentes niveles de estrés permite orientar las decisiones de los estudiantes al comienzo de un trimestre académico. Otras metodologías de resolución podrían ser exploradas para resolver el problema EPP-SL como programación de restricciones, modelos de dominio de planificación u otras meta-heurísticas.

Nuestro trabajo nos permitió incorporar medidas subjetivas, como el estrés y los efectos del aprendizaje, en el cálculo de las rutas de aprendizaje. Una extensión de nuestro trabajo considera la integración de recursos en las actividades de aprendizaje para estudiar las relaciones que podrían tener con los factores subjetivos presentados en este trabajo y su impacto en la personalización de las rutas de aprendizaje.

5.1 TRABAJO A FUTURO

Como se mencionó en un inicio, no existen muchos trabajos que incorporen características subjetivas dentro de problemas de optimización, puesto que no hay una forma estandarizada y aceptada por toda la comunidad científica para cuantificar o medir algunas de estas características. Por lo que para obtener mejores resultados es necesario establecer metodologías que nos permitan apegarnos a un marco de trabajo.

5.1.1 PROBLEMA

- En esta investigación logramos crear un modelo matemático que incluye algunas características subjetivas como el estrés y el coeficiente intelectual de una persona, sin embargo, aún existen muchos otros factores a considerar como el

tipo de estudiante, la forma de afrontar situaciones de estrés y muchas otras más, que pueden llegar a alterar la forma en como se desempeña una persona, ya sea en un ambiente laboral o académico.

- Dentro de las gráficas mostradas en el capítulo de experimentación, pudimos ver que los descansos dentro de las rutas de aprendizaje generan un papel importante en el tiempo necesario para completar la ruta. Por lo que, es necesario realizar más experimentación para saber que tanto tiempo debe de asignarse en cada descanso y conocer el impacto real de considerarlos dentro de las rutas de aprendizaje.

- La planeación de rutas educativas está estrechamente relacionado con problemas de secuenciación de actividades, los cuales han sido ampliamente estudiados, y aún más con el problema de ruteo de vehículos. En dicho problema, se busca encontrar la ruta que satisface la demanda de una cierta cantidad de clientes generando el menor costo, en el problema de generación de rutas de aprendizaje se busca encontrar la secuencia de actividades que satisfacen un objetivo final, en el menor tiempo posible. Por lo que es necesario profundizar en la similitud de estos problemas y aplicar metodologías de solución al problema de ruteo de vehículos en el problema de generación de rutas de aprendizaje.

5.1.2 INSTANCIAS Y DATOS

- Para apegar el problema aún más a la realidad, es necesario contar con datos de planes de estudio de instituciones académicas, que se utilicen actualmente. Existen plataformas de aprendizaje como Canvas o Moodle que permiten administrar cursos y asignar tareas y trabajos a los estudiantes, sería interesante poder incorporar herramientas de generación de rutas de aprendizaje en estas

plataformas, de tal manera que se pueda mejorar el desempeño de los estudiantes y medir el impacto de las rutas de aprendizaje.

- Una forma de medir algunas características subjetivas como el estrés es mediante los signos vitales de una persona, esto se puede lograr mediante dispositivos como relojes inteligentes. Sin embargo, no siempre se tiene acceso a este tipo de recursos, esto sin considerar la necesidad de tener una conexión a internet para la recolección de información. Por lo que, es necesario desarrollar metodologías menos invasivas que nos permitan saber el estado mental de una persona.

5.1.3 MÉTODO DE SOLUCIÓN

- Uno de los principales problemas para la generación de las rutas de aprendizaje que consideren el estrés de las personas, es el limitado acceso a la información de los signos vitales, por lo que una forma de afrontar este problema es mediante el desarrollo de algoritmos predictivos, que nos permitan estimar la situación mental de un estudiante a lo largo del periodo escolar y poder ofrecerle una ruta académica adecuada a esa estimación.
- Idealmente, este tipo de herramientas para la generación de rutas de aprendizaje pueden ser cargados en la nube de las instituciones educativas sin ningún problema, solo es necesario el desarrollo de una interfaz web de preferencia, lo que facilitaría el acceso a la plataforma y que le permita a cada estudiante ingresar y planear su ruta académica, de acuerdo a sus características personales. Además, que se pueda ir ajustando conforme surjan cambios en la situación de cada persona.

APÉNDICE A

CONCEPTOS

A.1 RAMIFICACIÓN Y ACOTAMIENTO

El método de ramificación y acotamiento se basa en el principio de que el conjunto de soluciones factibles puede ser particionado en conjuntos más pequeños de tal manera que se puedan analizar dichos conjuntos hasta encontrar la mejor solución (Taylor, 2004). Se basa principalmente en cuatro fases: Ramificar, acotar, podar, replegar. La fase de *ramificación* consiste en la creación de uno o más subproblemas partiendo de uno inicial. La fase de *acotamiento* es usada para evaluar las soluciones parciales encontradas durante la fase de ramificación y tiene dos componentes principales, una cota superior encontrada por la solución directa del problema relajado y una cota inferior, que es un estimado de lo que podría ser un resultado favorable al redondear las variables. la fase de *poda* se refiere a la eliminación de nodos y las posibles ramas subsecuentes debido a que la solución encontrada no es factible ya que viola una o más restricciones. Y la última fase de *repliegue* corresponde a regresar a un nivel superior dentro del árbol de soluciones creado (Michael J. Brusco, 2005). Para ejemplificar las fases de este método, supongamos el siguiente problema:

$$\text{máx } Z = 100x_1 + 150x_2 \tag{A.1}$$

$$8000x_1 + 4000x_2 \leq 40000 \quad (\text{A.2})$$

$$15x_1 + 30x_2 \leq 200 \quad (\text{A.3})$$

$$x_1, x_2 \geq 0, x_1, x_2 \in \mathbb{Z} \quad (\text{A.4})$$

Los pasos para el algoritmo de ramificación y acotamiento son:

1. Encuentra una solución para un problema de programación lineal, relajando las restricciones de integralidad de las variables. Una solución para el problema anterior es: $x_1 = 2.22$, $x_2 = 5.56$, $Z = 1055.56$.
2. En el nodo inicial asigna como cota superior UB (por sus siglas en inglés de *Upper bound*) la solución encontrada en el paso 1 y como cota inferior LB (por sus siglas en inglés de *lower bound*) el valor encontrado redondeando hacia abajo el valor de las variables. Para este ejemplo sería: $UB = 1055.56(x_1 = 2.22, x_2 = 5.56)$, $LB = 950(x_1 = 2, x_2 = 5)$.
3. Selecciona la variable con la mayor proporción fraccionaria y crea dos restricciones para esta variable que reflejen los valores enteros particionados. El resultado es una restricción del tipo \leq y otra restricción \geq . Para el ejemplo anterior sería: $x_2 \geq 5$ y $x_2 \leq 6$.
4. Crea dos nuevos nodos, uno para cada una de las restricciones creadas anteriormente.

Modelo 1

$$\text{máx } Z = 100x_1 + 150x_1 \quad (\text{A.5})$$

$$8000x_1 + 4000x_2 \leq 40000 \quad (\text{A.6})$$

$$15x_1 + 30x_2 \leq 200 \quad (\text{A.7})$$

$$x_2 \geq 5 \quad (\text{A.8})$$

$$x_1, x_2 \geq 0, x_1, x_2 \in \mathbb{Z} \quad (\text{A.9})$$

Modelo 2

$$\text{máx } Z = 100x_1 + 150x_1 \quad (\text{A.10})$$

$$8000x_1 + 4000x_2 \leq 40000 \quad (\text{A.11})$$

$$15x_1 + 30x_2 \leq 200 \quad (\text{A.12})$$

$$x_2 \leq 6 \quad (\text{A.13})$$

$$x_1, x_2 \geq 0, x_1, x_2 \in \mathbb{Z} \quad (\text{A.14})$$

5. Resuelve los nuevos modelos de programación lineal que contiene las nuevas restricciones.
6. La solución relajada encontrada es la nueva UB de cada nodo y la solución entera máxima es la LB .
7. Si el proceso produce una solución entera factible igual a la UB de dicho nodo, se alcanzó la solución óptima. Si no se ha encontrado una solución entera factible, ramifica el nodo con la mayor UB .
8. Regresa al paso 3.

BIBLIOGRAFÍA

- ALBERDI, A., A. AZTIRIA y A. BASARAB (2016), «Towards an automatic early stress recognition system for office environments based on multimodal measurements: A review», *Journal of Biomedical Informatics*, **59**, págs. 49–75, URL <https://www.sciencedirect.com/science/article/pii/S1532046415002750>.
- ALZAHM, A. M., H. T. VAN DER MOLEN y B. J. DE BOER (2015), «Effectiveness of a Dental Students Stress Management Program», *Health Professions Education*, **1**(1), págs. 34–42, URL <https://www.sciencedirect.com/science/article/pii/S2452301115000048>.
- AQIL, S. y K. ALLALI (2021), «On VNS-GRASP and Iterated Greedy Metaheuristics for Solving Hybrid Flow Shop Scheduling Problem with Uniform Parallel Machines and Sequence Independent Setup Time», en *Heuristics for Optimization and Learning*, Springer, págs. 17–32.
- BÁEZ, S., F. ANGEL-BELLO, A. ALVAREZ y B. MELIÁN-BATISTA (2019), «A hybrid metaheuristic algorithm for a parallel machine scheduling problem with dependent setup times», *Computers & Industrial Engineering*, **131**, págs. 295–305.
- BALDAZO, J. P. R., Y. A. RÍOS-SOLÍS y R. S. NIGENDA (2023), «Scheduling personalized study plans considering the stress factor», *Interactive Learning Environments*, **0**(0), págs. 1–20, <https://doi.org/10.1080/10494820.2023.2191260>, URL <https://doi.org/10.1080/10494820.2023.2191260>.

- BEEHR, T. A. y J. E. NEWMAN (1978), «Job stress, employee health, and organizational effectiveness: A facet analysis, model, and literature review 1», *Personnel psychology*, **31**(4), págs. 665–699.
- BENMESBAH, O., M. LAMIA y M. HAFIDI (2021), «An improved constrained learning path adaptation problem based on genetic algorithm», *Interactive Learning Environments*, **0**(0), págs. 1–18, 10.1080/10494820.2021.1937659, URL 10.1080/10494820.2021.1937659.
- BISKUP, D. (1999), «Single-machine scheduling with learning considerations», *European Journal of Operational Research*, **115**(1), págs. 173–178.
- BISKUP, D. (2008), «A state-of-the-art review on scheduling with learning effects», *European Journal of Operational Research*, **188**(2), págs. 315–329.
- BROWN, S. B., J. F. BROSSCHOT, A. VERSLUIS, J. F. THAYER y B. VERKUIL (2018), «New methods to optimally detect episodes of non-metabolic heart rate variability reduction as an indicator of psychological stress in everyday life», *International Journal of Psychophysiology*, **131**, págs. 30–36, the Psychophysiology of Stress and Adaptation: Models, Pathways, and Implications, URL <https://www.sciencedirect.com/science/article/pii/S0167876017302325>.
- BROWN, S. B., J. F. BROSSCHOT, A. VERSLUIS, J. F. THAYER y B. VERKUIL (2020), «Assessing new methods to optimally detect episodes of non-metabolic heart rate variability reduction as an indicator of psychological stress in everyday life: A thorough evaluation of six methods», *Frontiers in neuroscience*, **14**, pág. 564123.
- BRUCKER, P. (2004), *Scheduling Algorithms*, Springer Berlin Heidelberg, URL <https://books.google.com.mx/books?id=X08fRT1yr58C>.
- CAMPKIN, M. (2000), «Stress management in primary care», *Family Practice*, **17**(1), págs. 98–99, URL <https://doi.org/10.1093/fampra/17.1.98--a>.

- CHENG, T. E., C.-C. WU y W.-C. LEE (2008), «Some scheduling problems with sum-of-processing-times-based and job-position-based learning effects», *Information Sciences*, **178**(11), págs. 2476–2487.
- DE SMET, C., B. DE WEVER, T. SCHELLENS y M. VALCKE (2016), «Differential impact of learning path based versus conventional instruction in science education», *Computers & Education*, **99**, págs. 53–67.
- DURAND, G., N. BELACEL y F. LAPLANTE (2013), «Graph theory based model for learning path recommendation», *Information Sciences*, **251**, págs. 10–21.
- DURBAS, A., H. KARAMAN, C. H. SOLMAN, N. KAYGISIZ y O. ERSOY (2021), «Anxiety and Stress Levels Associated With COVID-19 Pandemic of University Students in Turkey: A Year After the Pandemic», *Frontiers in Psychiatry*, **12**.
- DYNARSKI, M., L. CLARKE, B. COBB, J. FINN, R. RUMBERGER y J. SMINK (2008), «Dropout Prevention: A Practice Guide (NCEE 2008-4025)», *Informe técnico*, Washington, DC: National Center for Education Evaluation and Regional Assistance, Institute of Education Sciences, U.S. Department of Education.
- FEO, T. A. y M. G. C. RESENDE (1995), «Greedy Randomized Adaptive Search Procedures», *Journal of Global Optimization*, **6**(2), págs. 109–133.
- FESTA, P. y M. G. RESENDE (2009), «An annotated bibliography of GRASP—Part I: Algorithms», *International Transactions in Operational Research*, **16**(1), págs. 1–24.
- GARRIDO, A., L. MORALES y I. SERINA (2016), «On the use of case-based planning for e-learning personalization», *Expert Systems with Applications*, **60**, págs. 1–15.
- GIANNAKAKIS, G., M. PEDIADITIS, D. MANOUSOS, E. KAZANTZAKI, F. CHIARUGI, P. SIMOS, K. MARIAS y M. TSIKNAKIS (2017), «Stress and anxiety detection using facial cues from videos», *Biomedical Signal Processing and Control*, **31**, págs. 89–101, URL <https://www.sciencedirect.com/science/article/pii/S1746809416300805>.

- HAIKALIS, M., H. DOUCETTE, M. K. MEISEL, K. BIRCH y N. P. BARNETT (2022), «Changes in College Student Anxiety and Depression From Pre- to During-COVID-19: Perceived Stress, Academic Challenges, Loneliness, and Positive Perceptions», *Emerging Adulthood*, **10**(2), págs. 534–545.
- HAMAIDEH, S. H., H. AL-MODALLAL, M. TANASH y A. HAMDAN-MANSOUR³ (2022), «Depression, anxiety and stress among undergraduate students during COVID-19 outbreak and 'home-quarantine'», *Nursing Open*, **9**(2), págs. 1423–1431.
- HANSEN, P. y N. MLADENOVIĆ (2001), «Variable neighborhood search: Principles and applications», *European Journal of Operational Research*, **130**(3), págs. 449–467.
- HWANG, G.-J., F.-R. KUO, P.-Y. YIN y K.-H. CHUANG (2010), «A heuristic algorithm for planning personalized learning paths for context-aware ubiquitous learning», *Computers & Education*, **54**(2), págs. 404–415.
- INEGI (2019), *Abandono escolar en México*, URL <https://www.inegi.org.mx/app/tabulados/interactivos/?pxq=9171df60-8e9e-4417-932e-9b80593216ee#variables>.
- JEONG, H.-Y., C.-R. CHOI y Y.-J. SONG (2012), «Personalized Learning Course Planner with E-learning {DSS} using user profile», *Expert Systems with Applications*, **39**(3), págs. 2567 – 2577.
- KIM, H.-G., E.-J. CHEON, D.-S. BAI, Y. H. LEE y B.-H. KOO (2018), «Stress and heart rate variability: A meta-analysis and review of the literature», *Psychiatry investigation*, **15**(3), pág. 235.
- KLAŠNJA-MILIĆEVIĆ, A., B. VESIN, M. IVANOVIĆ y Z. BUDIMAC (2011), «E-Learning personalization based on hybrid recommendation strategy and learning style identification», *Computers & education*, **56**(3), págs. 885–899.

- LAWLER, E. L. (1973), «Optimal sequencing of a single machine subject to precedence constraints», *Management science*, **19**(5), págs. 544–546.
- LIU, X., S. LU, J. PEI y P. M. PARDALOS (2018), «A hybrid VNS-HS algorithm for a supply chain scheduling problem with deteriorating jobs», *International Journal of Production Research*, **56**(17), págs. 5758–5775.
- LOWIS, M. y A. CASTLEY (2008), «Factors affecting student progression and achievement: prediction and intervention. A two-year study», *Innovations in education and teaching international*, **45**(4), págs. 333–343.
- MEEUWISSE, M., S. E. SEVERIENS y M. P. BORN (2010), «Reasons for withdrawal from higher vocational education. A comparison of ethnic minority and majority non-completers», *Studies in higher education*, **35**(1), págs. 93–111.
- MICHAEL J. BRUSCO, S. S. (2005), *Branch-and-Bound Applications in Combinatorial Data Analysis*, Springer New York, NY, URL <https://doi.org/10.1007/0-387-28810-4>.
- MOSHEIOV, G. (2001), «Scheduling problems with a learning effect», *European Journal of Operational Research*, **132**(3), págs. 687–693.
- NABIZADEH, A. H., D. GONÇALVES, S. GAMA, J. JORGE y H. N. RAFSANJANI (2020), «Adaptive learning path recommender approach using auxiliary learning objects», *Computers & Education*, **147**, pág. 103777.
- NABIZADEH, A. H., A. MÁRIO JORGE y J. PAULO LEAL (2017), «RUTICO: Recommending Successful Learning Paths Under Time Constraints», en *Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization*, UMAP '17, Association for Computing Machinery, New York, NY, USA, pág. 153–158, URL <https://doi.org/10.1145/3099023.3099035>.
- PINEDO, M. (2016), *Scheduling: Theory, Algorithms, and Systems*, Springer International Publishing, URL <https://books.google.com.mx/books?id=M4-RCwAAQBAJ>.

- RIBEIRO, R. T. y J. P. S. CUNHA (2018), «A regression approach based on separability maximization for modeling a continuous-valued stress index from electrocardiogram data», *Biomedical Signal Processing and Control*, **46**, págs. 33–45.
- SANCHEZ NIGENDA, R., C. MAYA PADRÓN, I. MARTÍNEZ-SALAZAR y F. TORRES-GUERRERO (2018), «Design and evaluation of planning and mathematical models for generating learning paths», *Computational Intelligence*, **34**(3), págs. 821–838.
- SANDERS, A. (1983), «Towards a model of stress and human performance», *Acta psychologica*, **53**(1), págs. 61–97.
- SELYE, H. (1975), *Stress Without Distress*, Penguin Group (USA) Incorporated, URL <https://books.google.com.mx/books?id=0V0ym4SIPw4C>.
- SMET, C. D., T. SCHELLENS, B. D. WEVER, P. BRANDT-POMARES y M. VALCKE (2016a), «The design and implementation of learning paths in a learning management system», *Interactive Learning Environments*, **24**(6), págs. 1076–1096, 10.1080/10494820.2014.951059, URL 10.1080/10494820.2014.95105.
- SMET, C. D., T. SCHELLENS, B. D. WEVER, P. BRANDT-POMARES y M. VALCKE (2016b), «The design and implementation of learning paths in a learning management system», *Interactive Learning Environments*, **24**(6), págs. 1076–1096.
- SMITH, G. y S. THRONE (2007), *Differentiating Instruction with Technology in K-5 Classrooms*, International Society for Technology in Education (ISTE).
- SÖRENSEN, K. y F. GLOVER (2013), «Metaheuristics», *Encyclopedia of operations research and management science*, **62**, págs. 960–970.
- TAKAČI, ., M. MARIĆ, G. STANKOV y A. DJENIĆ (2017), «Efficiency of using VNS algorithm for forming heterogeneous groups for CSCL learning», *Computers & Education*, **109**, págs. 98–108.
- TALBI, E.-G. (2009), *Metaheuristics: From Design to Implementation*, Wiley Publishing.

- TAYLOR, B. (2004), *Introduction to Management Science*, Prentice Hall, URL <https://books.google.com.mx/books?id=wnevPuhs5IC>.
- TROTMAN, G. P., J. J. VELDHUIJZEN VAN ZANTEN, J. DAVIES, C. MÖLLER, A. T. GINTY y S. E. WILLIAMS (2019), «Associations between heart rate, perceived heart rate, and anxiety during acute psychological stress», *Anxiety, Stress, & Coping*, **32**(6), págs. 711–727.
- TZOUVELI, P., P. MYLONAS y S. KOLLIAS (2008a), «An Intelligent e-Learning System Based on Learner Profiling and Learning Resources Adaptation», *Computers & Education*, **51**(1), págs. 224–238.
- TZOUVELI, P., P. MYLONAS y S. KOLLIAS (2008b), «An intelligent e-learning system based on learner profiling and learning resources adaptation», *Computers & Education*, **51**(1), págs. 224–238.
- VOGEL, S., L. M. KLUEN, G. FERNÁNDEZ y L. SCHWABE (2018), «Stress affects the neural ensemble for integrating new information and prior knowledge», *NeuroImage*, **173**, págs. 176–187.
- VOSSENSTEYN, J. J., A. KOTTMANN, B. W. JONGBLOED, F. KAISER, L. CREMONINI, B. STENSAKER, E. HOVDHAUGEN y S. WOLLSCHIED (2015), «Dropout and completion in higher education in Europe: Main report», .
- WANG, J.-B. (2006), «A note on scheduling problems with learning effect and deteriorating jobs», *International Journal of Systems Science*, **37**(12), págs. 827–833.
- WANG, S., C. CHRISTENSEN, W. CUI, R. TONG, L. YARNALL, L. SHEAR y M. FENG (2020), «When adaptive learning is effective learning: comparison of an adaptive learning system to teacher-led instruction», *Interactive Learning Environments*, **0**(0), págs. 1–11.
- WANG, X. y T. E. CHENG (2007), «Single-machine scheduling with deteriorating jobs and learning effects to minimize the makespan», *European Journal of Operational Research*, **178**(1), págs. 57–70.

-
- WESTMAN, M. y D. EDEN (1996), «The inverted-U relationship between stress and performance: A field study», *Work & Stress*, **10**(2), págs. 165–173, 10.1080/02678379608256795, URL 10.1080/02678379608256795.
- WOLF, L., A. W. STIDHAM y R. ROSS (2015), «Predictors of stress and coping strategies of US accelerated vs. generic baccalaureate nursing students: An embedded mixed methods study», *Nurse education today*, **35**(1), págs. 201–205.
- WOLSEY, L. (2020), *Integer Programming*, segunda edición, Wiley.
- ÖZPOLAT, E. y G. B. AKAR (2009), «Automatic detection of learning styles for an e-learning system», *Computers & Education*, **53**(2), págs. 355–367.

RESUMEN AUTOBIOGRÁFICO

Juan Pablo Rosas Baldazo

Candidato para obtener el grado de
Doctor en Ingeniería de Sistemas

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica

Tesis:

GENERANDO TRAYECTORIAS DE APRENDIZAJE QUE CONSIDEREN
DATOS BIOMÉTRICOS

Mi nombre es Juan Pablo Rosas Baldazo, nací el 3 de julio de 1987 en Monterrey, Nuevo León. Hijo de María Magdalena Baldazo Barcenas y Pedro Pablo Rosas García, soy Ingeniero en Sistemas Computacionales por el Instituto Tecnológico de Nuevo León (*ITNL*) y Maestro en Ciencias con Orientación a Sistemas por el Posgrado de Ingeniería de Sistemas (*PISIS*) de la Facultad de Ingeniería Mecánica y Eléctrica (*FIME*) de la Universidad Autónoma de Nuevo León (*UANL*). He trabajado como desarrollador de software en consultorías de TI y como profesor de licenciatura en distintas universidades, impartiendo generalmente materias enfocadas a TI. Actualmente soy miembro de la Sociedad Mexicana de Investigación de Operaciones (*CSMIO*).