AMBIENTE TESIS

VUE esquema documento tesis

VUE diagramas de flujo

Zotero

Focus

Firefox

Whiteboard con algoritmos

Spyder 3

StarUML

Inkscape Esquemas\_tesis

Texmaker

QA2Do

Word Criterios AOCE.docx

Excel

El código tiene dos partes fundamentales:

1. Problemas mono\_objetivo

2. Problemas multi\_objetivo

En Gendrau 2010 p. 42, explican muy bien la diferencia entre espacio de búsqueda y estructura de vecindario (operadores). Mi espacio de búsqueda está definido por la asignación de servidores a los nodos de servicio (los flujos son hallados por PL). De allí se desprenden los operadores o estructuras de vecindario que he usado.

Debería exportar también:

Landscape: Me dice cuántas soluciones factibles se evaluaron. Tengo que incluir landscape en network.file para que al generar salida\_medicion, quede dentro de la información exportada.

También deseo explorar el fitness landscape. El libro de Talbi tiene indicadores de distribución de locales en el espacio de búsqueda, entropía, distribución en el espacio objetivo, longitud de caminos, función de autocorrelación, correlación de la distancia de fitness. Las medidas que se proponen pueden ser calculadas tomando cada solución como un nodo de la red. Un arco existe si hay una permutación que lleva de una solución a otra. Puedo usar NetworkX para explorar el Landscape como un grafo con medidas de distribución y correlación.

A tener en cuenta:

La relación entre sigma\_max y los s\_jk es muy importante.

Si sigma\_max es más grande que la suma de los s\_jk significa que tengo más servidores que capacidad. Esto da lugar a que se llenen los nodos de servicio y el problema alcanza su solución fácilmente.

Si sigma\_max es menor a la suma de los s\_jk, significa que tengo menos servidores que capacidad. Ahí el problema se complejiza porque debo tomar decisiones de dónde asignar.

Un cuello de botella se forma cuando un lambda es mayor a s\_jk y el nodo se congestiona con un rho mayor a 1. En ese caso, el problema es infactible. En tal caso habría que aumentar las capacidades de los nodos de servicio.

Para una sola heurística.

Estudio de las medidas de desempeño:

Hay estas medidas:

1. Q: Calidad de la solución - Desviación porcentual óptimo(%)

2. E: Esfuerzo computacional relativo - Tiempo por nodo (s/nodo)

3. R: Robustez del algoritmo - Variabilidad de desviación porcentual (%)

4. F: Frecuencia de veces que se alcanzó el óptimo. (Veces)

Gráficas:

Q vs K vs Operador (Calidad por profundidad)

Q vs J vs Operador (Calidad por amplitud)

Q vs I\*J\*K vs Operador (Calidad por tamaño de instancia)

Q vs J/K vs Operador (Calidad por razón de aspecto)

E vs K vs Operador (Esfuerzo por profundidad)

E vs J vs Operador (Esfuerzo por amplitud)

E vs IJK vs Operador (Esfuerzo por tamaño de instancia)

E vs J/K vs Operador (Esfuerzo por razón de aspecto)

R \_ K

R \_ J

R \_ IJK

R \_ Operador

Mostrar la trayectoria de las soluciones. Un gráfico de F.O. vs tiempo que vaya descendiendo.

Ya tengo los resultados de tabú search y de local Search. Puedo corroborar cuáles son los mejores vecindarios.

Estoy ejecutando GVNS. Cuando hice VND me di cuenta que como el criterio de parada es el número de vecindarios, el algoritmo no realiza una buena exploración del espacio de soluciones y termina rápidamente con soluciones que no son las mejores. Con GVNS logré el óptimo en una instancia 555. Propongo centrarme en el documento en TS y GVNS y compararlos.

\*Terminé de revisar toda la tesis hasta el numeral 2.1. de la página 18.

\*Sigue esta sección que trae el modelo HFLNDP con costo.

\*Luego el capítulo 3 trae las métricas AOCE y una medición en un conjunto de redes.

Luego el capítulo 4 trae el modelo de optimización HFLNDP con criterios AOCE.

1. Terminé la ejecución de optimización para instancias grandes. Los resultados están en 20240626 Experimento Optima. Puedo llevarlos a una tabla para saber los óptimos de las redes, los tiempos de ejecución y la cantidad de nodos de cada red. Podría adicionar los resultados de las redes pequeñas que están en 20240626 Experimento Optima Bloque baja dimensionalidad.

En las conclusiones de esta sección no hay que demostrar que el problema demora bastante tiempo, sino que es np HARD. Ser np-hard no depende de la función objetivo sino del espacio de búsqueda. Encontré artículos que demuestran que el FLP, el NDP y FLNDP son np-completo. (Están en las ventanas abiertas de Firefox)

Luego tendría que escribir el capítulo 5 con la metaheurística

2. Para hacer el capítulo de metaheurísticas tengo que tener en cuenta estas recomendaciones de Gustavo:

Tengo que describir el uso de algoritmos de búsqueda local con énfasis en FLP y NDP. También con AOC.

- Ahmadi Javid. p. 234

- Location Science in Healthcare Chapter 21

- Radman y Esghi

- Reuniones director p. 80-84

- Revisión de 52 artículos sobre accesibilidad y congestión

Entregable: https://drive.google.com/open?id=1MrfX2m45z\_9gomk872yak9GNOQ8KE\_jx&authuser=eduartef%40unal.edu.co&usp=drive\_fs

- Revisión de literatura presentada en la tabla 2-1 de la tesis

- Otros artículos en Zotero

Análisis de solución por metaheurística usando VNS únicamente. No incluir tabú, local search.

Trabajar en la mejora del VNS para que encuentre siempre la solución óptima.

Revisión de literatura con apoyo IA: How To Write An Exceptional Literature Review With AI [NEXT LEVEL Tactics] (<https://www.youtube.com/watch?v=wz8lg_3j3Ok>)

1. Pedir a Chat GPT una estructura del capítulo.
2. Cada sección de la estructura es llevada como una pregunta a elicit.com
3. Elicit presenta artículos relevantes. Escojo los más recientes y con mejor afinidad a lo que quiero (reviso abstract).
4. Los ubico en connected papers y busco artículos derivados.
5. Guardo sus pdf en Zotero
6. Construyo un colección de PDF docanalyzer.ai
7. Para cada sección de la estructura creo un label que me servirá para hacer un chat con la ai y me generará los párrafos que puedo corregir, referenciar y mejorar para llevar a Texmaker

ESTRUCTURA DEL CAPÍTULO

**2. Fundamentos teóricos de Metaheurísticas de búsqueda local**: Explicar los conceptos básicos y el funcionamiento de las metaheurísticas de búsqueda local, incluyendo ejemplos de algoritmos relevantes (Búsqueda Tabú, Enfriamiento Simulado, Búsqueda Local Iterada, VNS) y su aplicabilidad a redes de salud.

**3. Metaheurísticas de búsqueda local en problemas de localización de instalaciones (FLP) y diseño de redes (NDP) en salud**

3.1. Localización de instalaciones (FLP) en salud

3.1.1. Principios y desafíos del FLP en salud: Abordar los objetivos y retos comunes en la localización de instalaciones de salud, como la optimización de la cobertura y la accesibilidad de los servicios de salud.

3.1.2. Aplicaciones de metaheurísticas de búsqueda local en FLP en salud: Revisar estudios donde se han aplicado metaheurísticas como Búsqueda Tabú, Enfriamiento Simulado, Búsqueda Local Iterada y VNS para resolver problemas de localización de instalaciones de salud.

3.1.3. Resultados y análisis de estudios relevantes: Analizar y comparar los resultados obtenidos en diferentes estudios, destacando la eficacia de las metaheurísticas de búsqueda local en mejorar la localización de instalaciones de salud.

3.2. Diseño de redes (NDP) en salud

3.2.1. Principios y desafíos del NDP en salud: Explicar los objetivos y problemas típicos en el diseño de redes de salud, incluyendo la optimización de la conectividad y la minimización de la congestión en redes de salud.

3.2.2. Aplicaciones de metaheurísticas de búsqueda local en NDP en salud: Examinar investigaciones donde se han implementado metaheurísticas como Búsqueda Tabú, Enfriamiento Simulado, Búsqueda Local Iterada y VNS en problemas de diseño de redes de salud.

3.2.3. Resultados y análisis de estudios relevantes: Evaluar los resultados reportados en la literatura, comparando la eficiencia y eficacia de las metaheurísticas de búsqueda local en el diseño de redes de salud.

**4. Metaheurísticas de búsqueda local para problemas de accesibilidad, congestión y continuidad en redes de salud**

4.1. Accesibilidad en redes de salud

4.1.1. Definición y relevancia de la accesibilidad en redes de salud: Definir qué se entiende por accesibilidad en redes de salud y por qué es un objetivo crucial en la planificación de redes de salud.

4.1.2. Aplicaciones de metaheurísticas de búsqueda local para mejorar la accesibilidad en redes de salud: Revisar casos donde se han utilizado metaheurísticas de búsqueda local para optimizar la accesibilidad a servicios de salud dentro de una red.

4.1.3. Resultados obtenidos: Discutir los resultados de los estudios, evaluando cómo las metaheurísticas de búsqueda local han mejorado la accesibilidad en redes de salud.

4.2. Congestión en redes de salud

4.2.1. Definición y relevancia de la congestión en redes de salud: Explicar la importancia de abordar la congestión en redes de salud y cómo afecta la eficiencia del sistema.

4.2.2. Aplicaciones de metaheurísticas de búsqueda local para reducir la congestión en redes de salud: Presentar estudios que aplican metaheurísticas de búsqueda local para minimizar la congestión en redes de salud.

4.2.3. Resultados obtenidos: Analizar los hallazgos de estos estudios, destacando las mejoras en la reducción de la congestión en redes de salud.

4.3. Continuidad del servicio en redes de salud

4.3.1. Definición y relevancia de la continuidad del servicio en redes de salud: Definir la continuidad del servicio en redes de salud y su importancia para la atención ininterrumpida y de calidad.

4.3.2. Aplicaciones de metaheurísticas de búsqueda local para asegurar la continuidad del servicio en redes de salud: Examinar investigaciones que utilizan metaheurísticas de búsqueda local para garantizar la continuidad del servicio en redes de salud.

4.3.3. Resultados obtenidos: Evaluar los resultados de estos estudios y cómo han contribuido a mejorar la continuidad del servicio en redes de salud.

**5. Análisis comparativo de metaheurísticas de búsqueda local**

5.1. Comparación de eficiencia y eficacia en redes de salud: Comparar la eficiencia y eficacia de diferentes metaheurísticas de búsqueda local en la optimización de redes de salud.

5.2. Ventajas y desventajas en redes de salud: Analizar las ventajas y limitaciones de cada metaheurística de búsqueda local en el contexto de redes de salud, considerando los problemas y objetivos específicos.

5.3. Factores a considerar en redes de salud: Identificar factores prácticos y contextuales a tener en cuenta al seleccionar una metaheurística de búsqueda local para aplicaciones en redes de salud.

6. Casos de estudio destacados

6.1. Ejemplos de implementación en redes de salud: Describir detalladamente casos de estudio donde se hayan aplicado metaheurísticas de búsqueda local a los problemas y objetivos mencionados en el contexto de redes de salud.

6.2. Resultados y conclusiones en redes de salud: Discutir los resultados obtenidos en estos casos de estudio y las lecciones aprendidas en el ámbito de redes de salud.

7. Conclusiones

7.1. Resumen de hallazgos clave en redes de salud: Resumir los principales hallazgos de la revisión, destacando la contribución de las metaheurísticas de búsqueda local en la optimización de redes de salud.

7.2. Direcciones futuras en redes de salud: Proponer áreas de investigación futura basadas en las brechas identificadas y los resultados obtenidos, con un enfoque en redes de salud.

7.3. Impacto potencial en redes de salud: Reflexionar sobre el impacto potencial de las metaheurísticas de búsqueda local en la mejora de los sistemas de salud, especialmente en la optimización de redes de salud.

8. Referencias

Lista de todas las referencias citadas en el capítulo.

Una solución es un objeto que encapsula su información completa (congestión, accesibilidad y continuidad) VERIFICAR INTEGRIDAD DE DATOS

Una solución se codifica en un vecino que tiene estructura de diccionario así:

vecino1 = { k1: [ [sigma j1k1, sigma j2k1, sigma j3k1] , [rho\_max , sigma\_rho\_max] ] }

{ k2: [ [sigma j1k2, sigma j2k2, sigma j3k2} , [rho\_max , sigma\_rho\_max] ] }

{ k3: [ [sigma j1k3, sigma j2k3, sigma j3k3} , [rho\_max , sigma\_rho\_max] ] }

{ rho\_max:[ [rho\_max, sigma\_rho\_max, k\_rho\_max]}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*GESTIÓN DE COLORES \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Tomado de https://matplotlib.org/stable/users/explain/colors/colormaps.html#ware

Hay cuatro clases de mapas de colores:

Secuenciales: datos ordenados. Parten de un tono y va aumentando la luminosidad de forma monotónica. Los mapas secuenciales se pueden imprimir bien en blanco y negro (grises).

Divergentes: Datos alrededor de una media. Parten de un color, llegan al blanco, y finalizan en otro color. No se imprimen bien en grises.

Cíclicos: ???

Cualitativos: No hay relación entre los datos. No son útiles para datos perceptuales. Existen varias paletas disponibles.

Para facilitar la lectura de personas con deficiencias visuales se recomienda usar el verde y rojo en el mismo gráfico.

La herramienta https://colorbrewer2.org/ permite escoger mapas de color y verificar si son amigables para personas con problemas de visión o al fotocopiar.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*GESTIÓN DE DATOS \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

La estrategia de gestión de información en el código es:

1. Leo los datos desde un assSrchivo de Excel.

2. Los datos son guardados en un objeto netowrk y una copia se guarda en un objeto solution.

3. Para construir el modelo de optimización exacta no se requiere network\_repr, los datos pasan del objeto network\_original a un archivo datos.dat y de allí pasan al modelo de Pyomo (Create\_data\_dat).

4. Tras resolver el modelo por Gurobi (execute\_solver), los datos se guardan en Excel (Set\_solution\_excel salida\_optimizacion) y en un txt (set\_solution\_txt). Si estoy en "Exacta", los datos de la solución también se guardan en un dataframe "detailed\_solution" (Set\_solution\_excel). Hasta aquí, los datos de la solución no se han guardado ni en el objeto solution ni en el objeto network,

5. Paso a evauar KPI con la opción 4.

Si current\_solution.objective != "Nulo", llevo los datos de Excel a file.network\_copy (merge\_niveles\_capac,create\_df\_asignacion,create\_df\_probs\_kk,create\_df\_arcos)

Si tecnica != "Local\_Search", estas funciones toman datos del archivo de excel salida\_optimizacion.xlsx y los llevan a solution.file.network\_copy

Si técnica == Local\_Search,estas funciones toman datos de network.problem (Ver más adelante el uso de Aproximada)

6. Aplico kpi.calculate\_kpi (usando los datos de file.network\_copy). Todos los cálculos se realizan sobre network\_copy

Aquí termina el proceso cuando uso "Exacta"

Ahora miro qué sucede cuando uso "Aproximada"

Escojo el tipo de problema monoobjetivo escojo Aproximación.

optimizar=True, tecnica=Aproximacion

Al escoger Aproximación, se llama la función initial\_solution. Allí se crea un objeto network\_repr. Los nuevos sigma quedan guardados en network\_repr, y se construyen los df\_sigma, df\_f\_ijk, df\_l\_jk, df\_solucion, df\_prob\_fi\_ijkjk, df\_fi\_ijkjk, df\_prob\_fi\_jkjk que se guardan en network\_repr y en solution (quedan por fuera de network\_copy). Los nuevos lambda, phi, pi son construidos en el network\_repr.

Los datos quedan grabados en network\_repr. No se ha modificado network\_copy

No se han calculado KPIs.

Luego, llevo los datos a un Excel (set\_solution\_excel), y procedo ejecutar fix\_initial solution, se calculan los kpi con kpi\_local\_search. Al interior de kpi\_calculate, se deben halar los datos de la solución. Como la solución es \_post\_optima y no es LocalSearch, se halan desde el excel que se ha construido.

En todos los casos, al ejecutar kpi\_calculate, se actualizan los datos que están en solution.file.network\_copy

Ahora voy a escoger el problema monoobjetivo y escojo Local\_Search.

se ejecuta initial\_solution, se crea el objeto network\_repr

Al ejecutar initial\_solution y fix\_initial\_solution, el código toma los datos de network\_repr y construye nuevas matrices df\_sigma, df\_asignacion, df\_l\_jk, solution, df\_prob\_fi\_ijkjk, df\_fi\_ijkjk, df\_prob\_fi\_jkjk

Luego se ejecuta fix\_initial\_solution.

Allí se hace un kpi\_calculate y se leen los datos que quedaron en el objeto solution (porque estoy con tecnica=Local\_Search), no los que están en file.network\_copy, ni los que están en network\_repr. Los datos que están en el objeto solution fueron tomados de network\_repr.

Por lo tanto, si deseo evaluar un objeto neighbor y es la solución inicial, puedo usar los datos que están en solution, pero no necesariamente los que es´tan en network\_repr.