

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

Informe Tarea 2
Desarrollo de instancias utilizando algoritmos genéticos

Joaquín Saldivia

Sebastián Sandoval

Emanuel Vega

Noviembre 2024

Índice

1. Introducción	3
2. Descripción	4
2.1 Descripción de la metaheurística elegida	4
2.2 Tipo de la metaheurística	4
2.3 Ventajas y desventajas	5
2.4 Razón elección	6
3. Implementación	7
3.1 Pseudocódigo	7
3.2 Descripción de los componentes	8
3.3 Valores de los parámetros utilizados	9
4. Mejores soluciones	9
4.2 Instancia 2	10
4.3 Instancia 3	11
5. Resultados	13
5.1 Tablas de ejecución	13
5.2 Gráfico de cajas instancia 1	14
5.4 Gráfico de cajas instancia 3	15
6. Conclusiones	16

1. Introducción

Introducción

La resolución de problemas complejos en la optimización combinatoria, como el problema de la mochila multidimensional (MKP), requiere enfoques capaces de explorar grandes espacios de soluciones respetando múltiples restricciones. Entre las técnicas más destacadas para abordar estos desafíos se encuentran las metaheurísticas, que combinan flexibilidad y eficiencia para encontrar soluciones óptimas o cercanas al óptimo en contextos donde los métodos exactos serían ineficaces.

En este informe, se presenta la implementación de un algoritmo genético, una metaheurística poblacional inspirada en los principios de la evolución natural. Este tipo de algoritmo se caracteriza por su capacidad de generar soluciones de calidad mediante procesos iterativos de selección, cruzamiento y mutación, permitiendo una exploración balanceada del espacio de búsqueda. A lo largo del desarrollo, se han considerado estrategias específicas para optimizar su desempeño frente a las características particulares del MKP, incorporando mecanismos de evaluación y ajustes dinámicos que garantizan un enfoque robusto.

El objetivo principal es analizar el rendimiento del algoritmo en diversas instancias del problema, evaluando su efectividad y consistencia. Los resultados obtenidos permiten reflexionar sobre las ventajas y limitaciones de este enfoque, así como explorar oportunidades para futuras mejoras y aplicaciones en escenarios prácticos, destacando su potencial en áreas como la logística y la planificación de recursos.

2. Descripción

2.1 Descripción de la metaheurística elegida

Para resolver el problema de la mochila multidimensional (MKP), se ha elegido el algoritmo genético como metaheurística de optimización. Los algoritmos genéticos son métodos poblacionales inspirados en la evolución darwiniana, donde las soluciones son tratadas como individuos de una población que “evoluciona” con el objetivo de encontrar soluciones cada vez más óptimas a través de varias generaciones.

En este contexto, cada solución del problema se representa como un vector binario (genotipo) en el que cada posición del vector indica si un objeto específico es seleccionado (1) o no (0). Esta representación es natural y eficiente para el problema de la mochila, ya que facilita el cálculo del valor total y el peso de los objetos seleccionados.

El algoritmo comienza generando una población inicial que, en este caso, está compuesta por individuos seleccionados aleatoriamente y otros generados en base a una heurística de valor/peso. A lo largo de varias generaciones, los individuos de esta población inicial son sometidos a operadores genéticos como lo son: la selección de padres, favorece a los individuos con mejor fitness; el cruzamiento combina características de dos individuos padres para generar descendencia, y la mutación introduce cambios aleatorios en los individuos para mantener la diversidad genética.

La función de fitness que evalúa cada individuo incorpora una penalización para las soluciones que no cumplen con las restricciones de capacidad del MKP. De esta forma, el algoritmo puede evaluar y comparar soluciones tanto factibles como infactibles, favoreciendo naturalmente aquellas que se acercan a una solución óptima respetando las restricciones del problema. A medida que el algoritmo avanza, los individuos mejoran su calidad de manera progresiva, permitiendo al algoritmo genético explorar y explotar el espacio de soluciones para maximizar el valor total de la mochila

2.2 Tipo de la metaheurística

El algoritmo genético se clasifica como una metaheurística poblacional. A diferencia de los métodos de trayectoria, que exploran una única solución en cada iteración, las metaheurísticas poblacionales manejan múltiples soluciones en paralelo (población). En el caso del algoritmo genético, esta población evoluciona colectivamente, generando una nueva generación de soluciones en cada ciclo mediante operadores genéticos como la selección, el cruzamiento y la mutación. Esta característica permite explorar de forma más amplia el espacio de búsqueda y facilita la diversidad en las soluciones, incrementando la probabilidad de encontrar una solución óptima.

Además, los algoritmos genéticos son una metaheurística de optimización estocástica. Este tipo de optimización se caracteriza por incorporar elementos aleatorios, como la selección aleatoria de padres o la mutación en individuos, lo cual permite que el algoritmo no solo explote las mejores soluciones en cada generación, sino que también explore nuevas áreas del espacio de búsqueda. Esta aleatoriedad reduce el riesgo de quedar atrapado en óptimos locales y fomenta una búsqueda más exhaustiva en problemas complejos como el MKP.

Finalmente, debido a su capacidad de manejar restricciones complejas a través de funciones de penalización y ajuste en el fitness, los algoritmos genéticos son adecuados para problemas como el de la mochila multidimensional, donde se deben considerar múltiples restricciones sin limitar la exploración del espacio de soluciones

2.3 Ventajas y desventajas

Ventajas:

- **Capacidad para manejar restricciones complejas:** Los algoritmos genéticos son flexibles al incluir funciones de evaluación que penalizan soluciones infactibles, permitiendo avanzar hacia soluciones factibles sin restringir drásticamente el espacio de búsqueda. Esta característica es especialmente útil en problemas como el MKP, donde se deben respetar múltiples restricciones.
- **Exploración y explotación simultáneas:** Al trabajar con una población de soluciones en cada generación, el algoritmo genético explora de manera amplia el espacio de búsqueda, mientras que la selección de los mejores individuos permite concentrarse en las regiones más prometedoras. Esto genera un equilibrio entre descubrir nuevas soluciones y refinar las existentes.
- **Evita el estancamiento en óptimos locales:** Gracias a los operadores de cruzamiento y mutación, los algoritmos genéticos pueden escapar de óptimos locales, lo cual es una ventaja frente a métodos de optimización que exploran una única trayectoria.
- **Facilidad de adaptación y ajuste:** Los algoritmos genéticos son adaptables a diferentes problemas al ajustar sus operadores y parámetros, como la tasa de mutación o el tipo de cruzamiento. Esta flexibilidad permite experimentar y mejorar el rendimiento del algoritmo según el problema específico.

Desventajas:

- **Requiere alto costo computacional:** La evaluación de múltiples soluciones en cada generación puede ser costosa en términos de tiempo y recursos computacionales, especialmente para problemas de gran tamaño o alta dimensionalidad.
- **Dependencia de parámetros:** El rendimiento de los algoritmos genéticos depende en gran medida de una adecuada configuración de parámetros como el tamaño de la población, la probabilidad de mutación y el número de generaciones. Una mala elección de estos parámetros puede llevar a una exploración insuficiente o a una convergencia prematura.
- **Convergencia prematura:** Existe el riesgo de que la población se vuelva homogénea demasiado rápido, llevando al algoritmo a quedar “atrapado” en soluciones subóptimas. Para mitigar este efecto, es necesario mantener un balance adecuado en la diversidad de la población, especialmente en etapas tempranas.
- **Posible dificultad para garantizar factibilidad:** Aunque los algoritmos genéticos pueden penalizar soluciones infactibles, en algunos problemas con restricciones muy estrictas o complejas, puede ser difícil asegurar que las soluciones generadas cumplan con todas las restricciones, lo cual puede requerir ajustes adicionales en la función de evaluación

2.4 Razón elección

La elección de un algoritmo genético para resolver el problema de la mochila multidimensional (MKP) se basa en varias de sus características que lo hacen adecuado para este tipo de problemas de optimización combinatoria con restricciones.

En primer lugar, los algoritmos genéticos son especialmente efectivos en problemas complejos donde es difícil encontrar una solución óptima mediante métodos exactos debido a la gran cantidad de combinaciones posibles. Al trabajar con una población de soluciones y utilizar operadores de selección, cruzamiento y mutación, los algoritmos genéticos pueden explorar una gran cantidad de posibles combinaciones de objetos en la mochila sin necesidad de evaluar exhaustivamente todas las opciones.

Además, el algoritmo genético permite incorporar una función de evaluación con penalización, lo cual es clave en el MKP. Este problema tiene restricciones de capacidad en varias dimensiones, y el uso de penalizaciones permite que el algoritmo navegue temporalmente por soluciones infactibles, promoviendo una búsqueda flexible y adaptativa hacia soluciones factibles de alto valor.

Otra razón importante para elegir un algoritmo genético es su capacidad de equilibrar exploración y explotación. Esta propiedad es fundamental para evitar que el algoritmo se

quede en óptimos locales y, al mismo tiempo, permite refinar las mejores soluciones encontradas en cada generación. Gracias a su naturaleza poblacional, el algoritmo genético facilita la búsqueda de soluciones óptimas o cercanas al óptimo al trabajar con múltiples individuos que representan diferentes combinaciones posibles de objetos.

Por último, los algoritmos genéticos son flexibles y fáciles de adaptar a distintos parámetros y operadores según las necesidades específicas del problema. Esta adaptabilidad permite ajustar parámetros como la tasa de mutación y el tamaño de la población para mejorar el rendimiento del algoritmo en el contexto particular del MKP.

3. Implementación

3.1 Pseudocódigo

Data: Archivo con las instancias del problema

Result: Mejor solución encontrada para cada instancia

$instancias \leftarrow leer_instancias(archivo);$

$resultados \leftarrow \emptyset;$

foreach *instancia en instancias* **do**

$G \leftarrow generar_población_inicial(instancia);$

$s^+ \leftarrow mejor_solución(G);$

while *no criterio_de_termino()* **do**

$G^* \leftarrow \emptyset;$

while $|G^*| < tamaño_población$ **do**

$p_1, p_2 \leftarrow seleccionar_padres(G, fitness(G));$

$hijos \leftarrow cruzamiento_dos_puntos(p_1, p_2);$

$hijos \leftarrow mutar_población(hijos, tasa_mutación);$

$G^* \leftarrow G^* \cup hijos;$

end

$G \leftarrow aplicar_elitismo(G, G^*);$

$s^+ \leftarrow mejor_solución(s^+, G);$

end

if *no factible*(s^+) **then**

$s^+ \leftarrow reparar_individuo(s^+, valores, pesos, capacidades);$

end

$resultados \leftarrow resultados \cup \{s^+\};$

end

mostrar(resultados);

3.2 Descripción de los componentes

- **Población inicial:** Se inicia con 100 individuos, de los cuales 75 son generados aleatoriamente y los otros 25 se seleccionan según la heurística de relación valor/peso de los objetos. Esta mezcla de soluciones permite una buena diversidad inicial y una base de soluciones con alta probabilidad de factibilidad.
- **Función de evaluación:** La función de evaluación calcula el valor total de los objetos seleccionados para cada individuo. Además, penaliza las soluciones que exceden las capacidades máximas del problema. La penalización se calcula considerando el exceso de peso acumulado en cada dimensión, escalado por un factor basado en la relación valor/peso promedio de los objetos. Esto permite manejar de manera efectiva las restricciones del MKP, promoviendo soluciones factibles sin excluir por completo aquellas que no lo son.
- **Selección de padres:** Se emplea el método de torneo, en el cual un grupo aleatorio de individuos compite, y el individuo con el mejor fitness es seleccionado como padre. Este método equilibra la selección de soluciones prometedoras con la preservación de la diversidad en la población, lo que permite evitar la convergencia prematura hacia un óptimo local.
- **Cruzamiento:** Se emplea un operador de cruzamiento en dos puntos, que combina fragmentos de genes de dos padres para producir nuevos descendientes. Este tipo de cruzamiento facilita el intercambio de características entre los individuos, generando soluciones con combinaciones potencialmente mejores de objetos seleccionados.
- **Mutación:** Se aplica una probabilidad de mutación baja (0.2%) a cada gen de los descendientes, cambiando un 0 por un 1 o viceversa. La mutación introduce variabilidad en la población y previene la convergencia prematura del algoritmo hacia un óptimo local.
- **Selección de la nueva generación:** La nueva población se selecciona aplicando un elitismo del 10%, lo que significa que los mejores individuos de la generación anterior pasan directamente a la nueva generación. El elitismo garantiza que las soluciones de alta calidad se preserven y aumenten la probabilidad de mejorar continuamente el fitness global de la población.

3.3 Valores de los parámetros utilizados

Para la implementación del algoritmo genético en el MKP, se han utilizado los siguientes valores de parámetros:

- **Tamaño de la población:** 100 individuos.
- **Generación de la población inicial:** 75 individuos aleatorios y 25 basados en la heurística relación valor/peso.
- **Número de generaciones:** 100 generaciones
- **Probabilidad de mutación:** 0.2%.
- **Elitismo:** 10%, lo que permite preservar los 10 mejores individuos de cada generación.
- **Método de selección de padres:** Selección por torneo.
- **Cruzamiento:** Cruzamiento en dos puntos entre padres seleccionados.
- **Función de penalización:**

$$f_{evaluada}(x) = \sum_{i=1}^n x_i v_i - \rho \cdot \max(0, W(x) - W)$$

Penalización dinámica en función de ρ para asegurar que las soluciones infactibles sean penalizadas adecuadamente

4. Mejores soluciones

El código al finalizar, obtiene los mejores valores obtenidos del problema.

4.1 Instancia 1

Mejor instancia: N° 22

Mejor valor total después de reparación de todas las instancias: 61675

Vector de selección después de reparación:

[1,1,1,0,1,0,1,0,0,1,0,1,1,1,1,1,1,1,1,0,1,1,0,1,1,1,0,1,1,1,1,1,1,1,0,1,1,0,1,1,1,1,0,1,1,1,0,1,1,1,0,1,1,0,1,1,0,0,1,0,1,1,1,1,0,0,1,0,1,1,0,1,1,0,1,1,0,1,1,1,1,1,1,1,1,1,1,1]

Pesos obtenidos / Capacidades por dimensión (después de reparación):

Dimensión 1: 42757 / 42813

Dimensión 2: 36071 / 36845

Dimensión 3: 35747 / 36146

Dimensión 4: 37032 / 37149

Dimensión 5: 38649 / 39151

Valores de variables seleccionadas: [982 1009 1105 697 641 613 1059 569 976 774 755 842
972 761 625 880 804 851 514 853 1004 572 944 732 830 476 824 758
728 739 474 428 835 581 632 892 1035 401 839 600 1066 834 857 1061 710 795 976
1154 827 1082 943 557 834 695 655 630 614 935 1003 836 808 464 1079 781 861 1230
927 494 1096 673 626 861 734 825 842 995 709]

4.2 Instancia 2

Mejor instancia: N° 26

Mejor valor total después de reparación de todas las instancias: 150433

Vector de selección después de reparación:
[0,1,1,1,1,1,1,1,1,0,1,1,1,0,1,0,0,1,1,0,0,1,1,1,1,0,1,1,1,1,0,1,1,
1,1,1,1,1,1,1,1,1,0,1,1,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,1,1,
1,1,1,1,1,0,1,1,0,1,1,1,0,0,1,1,1,1,0,1,1,0,1,1,0,1,1,1,1,0,1,1,1,
0,0,1,1,1,0,0,1,1,0,1,0,1,0,1,0,1,1,0,0,1,1,1,1,0,0,1,1,1,1,0,1,
0,1,1,1,1,1,0,1,1,0,1,1,1,0,1,1,1,1,1,1,1,0,1,1,1,0,0,1,0,0,1,1,
1,1,1,0,1,1,1,1,1,0,1,0,0,0,1,0,1,1,0,1,1,1,1,1,1,1,1,0,0,0,1,1,1,0,
1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,1,1,1,0]

Pesos obtenidos / Capacidades por dimensión (después de reparación):

Dimensión 1: 92015 / 93296
Dimensión 2: 95735 / 96520
Dimensión 3: 88988 / 92703
Dimensión 4: 94566 / 96615
Dimensión 5: 90079 / 91424
Dimensión 6: 96187 / 96421
Dimensión 7: 91045 / 92414
Dimensión 8: 91814 / 95669
Dimensión 9: 99327 / 101343
Dimensión 10: 91850 / 91852

Valores de variables seleccionadas: [677 787 904 887 781 745 923 838 741 809 992 673
685 825 890 971 795 648 632 823 882 651 960 966 830 794 957 914 760 766 863 876
732 759 844 607 463 1081 885 850 991 836 646 494 534 842 408 869 642 691 914 764
698 784 843 607 888 854 934 934 729 845 863 804 694 995 931 713 706 626 844 1013
724 434 923 889 652 820 679 800 1029 817 950 922 938 946 838 514 628 810 786 755
812 871 932 965 861 991 864 679 885 968 712 859 720 841 664 945 562 744 823 805
1192 784 697 572 664 638 990 855 793 863 1016 941 734 1060 658 942 698 896 775
1031 782 953 620 846 1014 603 885 990 1011 747 906 628 813 838 697 903 660 637
798 692 897 1090 708 598 909 691 758 744 727 632 610 957 910 950 734 946 821 878
1054 544 969 836 952 783 776 868 576 633 650 1076 619 1010 560 750 6]

4.3 Instancia 3

Mejor instancia: N° 28

Mejor valor total después de reparación de todas las instancias: 302941

Vector de selección después de reparación:

[illegible]

Pesos obtenidos / Capacidades por dimensión (después de reparación):

Dimensión 1: 192752 / 196838

Dimensión 2: 191454 / 193714

Dimensión 3: 181055 / 183808

Dimensión 4: 184213 / 184442

Dimensión 5: 193217 / 193526

Dimensión 6: 177633 / 182362

Dimensión 7: 181204 / 187913

Dimensión 8: 189668 / 199290

Dimensión 9: 180364 / 188159

Dimensión 10: 172753 / 178775

Dimensión 11: 197789 / 197855

Dimensión 12: 179179 / 181840

Dimensión 13: 183958 / 189272

Dimensión 14: 190171 / 191300

Dimensión 15: 180499 / 186557

Dimensión 16: 188700 / 191682

Dimensión 17: 176700 / 179195

Dimensión 18: 175369 / 184603

Dimensión 19: 191209 / 194892

Dimensión 20: 183371 / 184667

Dimensión 21: 178031 / 187349

Dimensión 22: 182428 / 190034

Dimensión 23: 169277 / 176715

Dimensión 24: 179127 / 183189

Dimensión 25: 187799 / 191394
Dimensión 26: 183859 / 186110
Dimensión 27: 180023 / 185426
Dimensión 28: 189710 / 194468
Dimensión 29: 178980 / 188483
Dimensión 30: 178992 / 186912

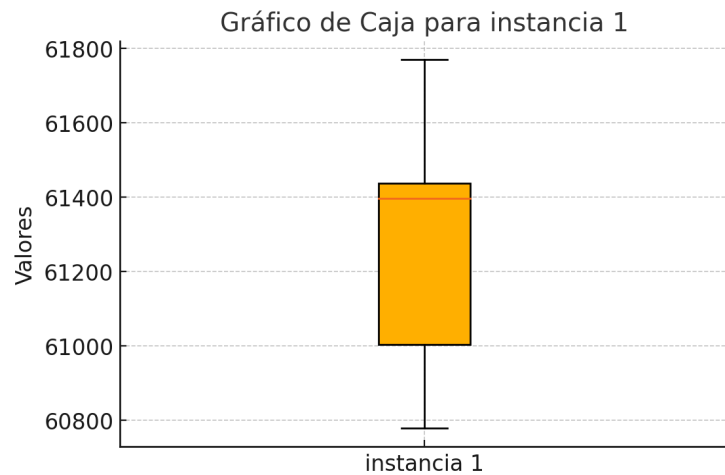
Valores de variables seleccionadas: [1004 797 839 902 928 653 711 942 653 962 645 764
879 782 935 837 997 1084 733 564 745 778 776 904 1030 820 820 849 859 912 708 687
875 1005 666 1008 860 942 628 646 832 779786 621 797 780 710 710 929 936 1017 809
821 688 902 886 835 921 862 942 976 1000 924 896 900 662 1009 785 967 665 733 575
779 955 657 700 993 789 797 813 793 967 647 989 749 754 960 641 939 881 739 928
750 926 666 919 992 627 1020 998 1048 702 1010 819 852 879 930 990 1000 583 748
970 1057 945 887 746 741 679 928 660 776 751 718 624 614 831 961 697 797 662 812
715 955 733 893 608 697 765 631 753 654 836 672 1058 675 851 759 907 825 674 804
878 932 867 869 789 888 986 812 830 959 678 851 623 925 948 783 805 867 669 858
914 867 828 773 940 918 863 771 593 940 985 868 1064 959 1047 771 766 791 696 878
753 993 814 1001 769 916 846 802 691 766 777 887 824 965 922 770 1025 763 1015 899
777 716 602 934 703 786 688 767 669 787 970 671 835 826 949 943 783 867 953 1002
936 743 1024 1024 875 627 977 958 596 753 738 718 723 830 904 824 1053 705 829
686 668 752 801 981 568 966 781 759 795 928 648 710 745 932 808 680 765 799 678
960 969 845 977 755 1033 808 994 844 759 1040 759 685 729 868 973 993 817 684
896 953 619 695 950 776 811 659 670 811 877 814 1017 837 689 908 853 959 962 695
727 690 637 727 806 898 886 916 847 895 715 719 759 869 789 615 902 966 910 996
960 1023 744 795 988 963 734 658 691 723 875 841 701 676 695 941 728 962 593 630
997 720 664 571 825 870 647 940 1014 672 871 877 668 836 848 621 740 684 947]

5. Resultados

5.1 Tablas de ejecución

ejecución	instancia 1	instancia 2	instancia 3
1	61450	151359	301417
2	60779	152460	300851
3	60938	152215	302651
4	61397	151860	302026
5	61398	151612	302661
6	61770	151878	302296
7	61584	151518	302533
8	60943	151118	301778
9	61186	151475	302565
10	61403	151363	302938
media	61284.8	151685.8	302171.6
desviación estándar	315.224	416.087	653.799

5.2 Gráfico de cajas instancia 1



Instancia 1:

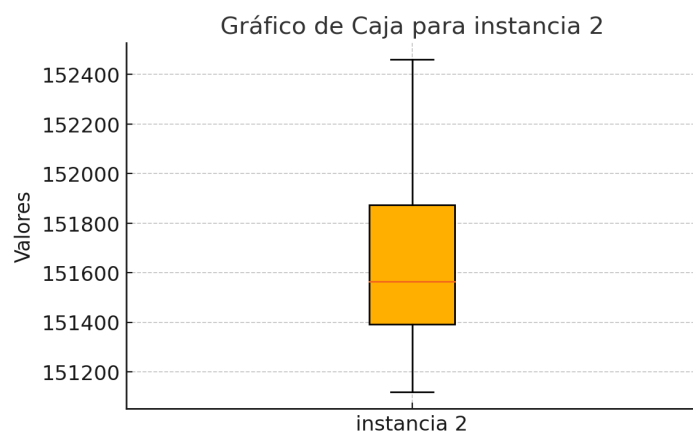
Mediana: La línea dentro de la caja está en aproximadamente 61.400, lo que indica que la mayoría de los valores están concentrados alrededor de este valor.

Rango intercuartil: va aproximadamente desde 61,100 hasta 61,425. Esto significa que la dispersión de los valores centrales es de sólo 325 unidades.

Bigotes: Los valores mínimos y máximos están en torno a 60,800 y 61,800, lo que confirma que los valores no se alejan demasiado de la mediana.

Interpretación: Los datos son muy consistentes, con poca variación, lo que puede significar estabilidad en el comportamiento de la instancia. No se identifican valores atípicos.

5.3 Gráfico de cajas instancia 2



Instancia 2:

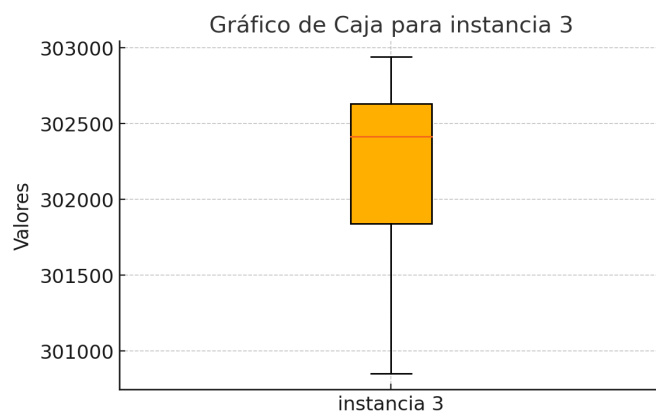
Mediana: La línea dentro de la caja está en aproximadamente 151.590, indicando que los valores tienden a estar ligeramente por debajo de los 152.000.

Rango intercuartil: va aproximadamente desde 151.400 hasta 151.840. Esto significa que la dispersión de los valores centrales es de sólo 440 unidades.

Bigotes: Los valores mínimos y máximos están en torno a 60.800 y 61.800, o que sugiere que los datos tienen una ligera mayor dispersión en comparación con la instancia 1.

Interpretación: Aunque los valores están concentrados en torno a la mediana, hay una mayor variabilidad en los datos. Esto podría indicar un comportamiento menos consistente o más influenciado por factores externos.

5.4 Gráfico de cajas instancia 3



Instancia 3:

Mediana: La línea dentro de la caja está en aproximadamente 302.400, lo que indica que la mayoría de los valores están concentrados alrededor de este valor.

Rango intercuartil: va aproximadamente desde 301.800 hasta 302.600 de aproximadamente 800 unidades, mucho mayor que en las instancias anteriores.

Bigotes: Los valores mínimos y máximos están en torno a 60,800 y 61,800, mostrando una mayor dispersión en comparación con las otras dos instancias.

Interpretación: La instancia 3 tiene una variabilidad más amplia que las instancias 1 y 2, lo que podría deberse a un comportamiento más dinámico de los datos o a condiciones menos controladas. Esto sugiere que es menos predecible, aunque todavía no hay outliers evidentes.

6. Conclusiones

La implementación de la metaheurística del algoritmo genético permitió comprender profundamente cómo las técnicas de optimización poblacional pueden abordar problemas complejos como el de la mochila multidimensional (MKP), destacando su capacidad para manejar restricciones y explorar eficientemente grandes espacios de búsqueda. El mayor desafío fue equilibrar la explotación de soluciones prometedoras y la exploración de nuevas regiones del espacio de soluciones, especialmente ajustando parámetros clave como la tasa de mutación y el elitismo para evitar la convergencia prematura y atascarse en óptimos locales. Aunque los resultados fueron satisfactorios, podrían mejorarse incorporando otras estrategias con otras metaheurísticas o adaptaciones dinámicas de parámetros en función del desempeño durante la ejecución. Estas técnicas tienen un gran potencial de aplicación en la realidad chilena, particularmente en áreas como la logística, la planificación de recursos, y la optimización de procesos industriales, donde se enfrentan restricciones similares y la búsqueda de soluciones eficientes es crítica.