



**ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA
INFORMÁTICA**

**GRADO EN INGENIERÍA INFORMÁTICA – TECNOLOGÍAS
INFORMÁTICAS**

ASC - COMPETICIÓN MULTIOBJETIVO

Realizado por FRANCISCO JESÚS BELMONTE PINTRE

Departamento de Electrónica y Electromagnetismo

Sevilla, Marzo de 2020

ESPECIFICACIONES DE LA IMPLEMENTACIÓN

Método basado en agregación por descomposición en sub problemas de Tchebycheff.

Aplicación de operador de cruce bajo las siguientes especificaciones:

- Evolución diferencial, 3 vectores escogidos aleatoriamente de entre los k vecinos más cercanos. K un valor obtenido de **redondear** $|P| * (\text{Porcentaje de vecindad})$.
- $F = 0.5$, para todos los experimentos.
- $CR = 0.5$, para todos los experimentos.

Aplicación del operador de mutación bajo la siguiente especificación:

- Mutación Gaussiana, para todos los experimentos.
- $SIG = 20$, para todos los experimentos.

Aplicación de la reproducción bajo los siguientes pasos, de manera secuencial y en orden:

1. Cruce bajo operador EA.
2. Mutación bajo operador Gaussiano.
3. Acotamiento entre límite superior e inferior definidos por el espacio de búsqueda de la variable x , esto es $\{0, 1\}$ en zdt3, $\{0, 1\}$ para x_1 y $\{-2, 2\}$ para x_n con $n > 1$ para cf6. Si sobrepasamos, tomamos los extremos.

Punto de referencia para las métricas tomado de manera manual, bajo las siguientes especificaciones y para todos los experimentos:

Para cada semilla $(1, \dots, 10)$, calculamos:

- $f1' = \text{worst}\{f1i, i \in \{1, \dots, |P| * |G|\}\}, \text{para AGG}$
- $f2' = \text{worst}\{f2i, i \in \{1, \dots, |P| * |G|\}\}, \text{para AGG}$
- $f1'' = \text{worst}\{f1i, i \in \{1, \dots, |P| * |G|\}\}, \text{para NSGAI}$
- $f2'' = \text{worst}\{f2i, i \in \{1, \dots, |P| * |G|\}\}, \text{para NSGAI}$

Nuestro punto de referencia final para las métricas será:

- $HV_{f1} = \text{worst}\{f1', f1''\}$
- $HV_{f2} = \text{worst}\{f2', f2''\}$
- $HV_{final} = (HV_{f1}, HV_{f2})$

Dado que estamos minimizando:

$$- \text{worst}(a_1, a_2, \dots, a_n) = \max \{a_1, a_2, \dots, a_n\}$$

Manejo de restricciones basado en técnicas de penalti, bajo las siguientes especificaciones y para todos los experimentos:

$$f'i(X) = fi(X) + \sum_j CVj, \quad i \in \{1, 2\}, j \in \{1, 2\}$$

$$CVj = Wj * |Rj|, \quad Wj \in \{0, 1\}, Rj \in (-\infty, 0]$$

$$Wj \equiv Rj < 0, \quad Wj \in \{0, 1\}$$

Cálculo del **IGD** bajo la siguiente especificación para todos los experimentos:

$$IGD(A, P) = \frac{\sum_{p \in P} MIN_{distance}(p, A)}{|P|}$$

Experimentos llevados a cabo:

1. Comparación frente de agregación y frente de nsgaii, para zdt3 con P = 100, G = 100, y 10 semillas. % Vecindad y % Mutación = 15% y 1/30.
2. Evolución Hipervolumen, Coverage Set y Spacing de frente por agregación y nsgaii, para el caso anterior.
Resultados en el directorio **./zdt3_P_100_G_100_agg_vs_nsgaii**.
3. Comparación frente de agregación y frente de nsgaii, para zdt3 con P = 40, G= 250 y 10 semillas. % Vecindad y % Mutación = 15% y 1/30.
4. Evolución Hipervolumen, Coverage Set y Spacing de frente por agregación y nsgaii, para el caso anterior.
Resultados en el directorio **./zdt3_P_40_G_250_agg_vs_nsgaii**.
5. Comparación IGD entre agregación y nsgaii, para zdt3 con P = 100, G = 100 y 10 semillas. % Vecindad y % Mutación = 15% y 1/30.
Resultados en el directorio **./zdt3_P_100_G_100_agg_vs_nsgaii/IGD**.
6. Comparación frente alcanzado por agregación para zdt3 con P = 100, G = 100 y 10 semillas, y frente alcanzado por agregación para zdt3 con P = 40, G = 250 y 10 semillas. % Vecindad y % Mutación = 15% y 1/30.
7. Evolución Hipervolumen, Coverage Set y Spacing de frente por agregación y nsgaii, para el caso anterior.

Resultados en el directorio ***./zdt3_P_100_G_100_agg_vs_P_40_G_250_agg.***

8. Comparación frente de agregación y frente de nsgaii, para zdt3 con $P = 80$, $G = 50$ y 10 semillas. % Vecindad y % Mutación = 30% y 1/30.
9. Evolución Hipervolumen, Coverage Set y Spacing de frente por agregación y nsgaii, para el caso anterior.

Resultados en el directorio ***./zdt3_P_80_G_50_agg_vs_nsgaii.***

10. Comparación 5 frentes de cf6 para semilla = 1: IDEAL vs P100G100N4 vs P100G100N16 vs P40G250N4 vs P40G250N16.
% Vecindad y % Mutación = 15%, 1/4 ($N = 4$) y 1/16 ($N = 16$).

Resultados en el directorio ***./cf6_quick_view.***

11. Comparación frente de agregación y frente de nsgaii, para cf6 con $P = 100$, $G = 100$, $N = 4$ y 10 semillas. % Vecindad y % Mutación = 15%, 1/4 ($N = 4$).
12. Evolución Coverage Set y Spacing de frente por agregación y nsgaii, para el caso anterior.

Resultados en el directorio ***./cf6_P_100_G_100_agg_vs_nsgaii.***

13. Comparación frente de agregación y frente de nsgaii, para cf6 con $N = 16$, $P = 80$, $G = 50$ y 10 semillas. % Vecindad y % Mutación = 15%, 2/16 ($N = 16$).
14. Evolución Hipervolumen, Coverage Set y Spacing de frente por agregación y nsgaii, para el caso anterior.

Resultados en el directorio ***./cf6_P_80_G_50_agg_vs_nsgaii.***

Puntos de referencia de ***HV***, ***CS*** y ***SPA*** almacenados en archivos ***HV.out***:

- ***./zdt3_P_100_G_100_agg_vs_nsgaii/HV.out.***
- ***./zdt3_P_40_G_250_agg_vs_nsgaii/HV.out.***
- ***./cf6_P_100_G_100_agg_vs_nsgaii/HV.out.***
- ***./zdt3_P_100_G_100_agg_vs_P_40_G_250_agg/HV.out.***
- ***./zdt3_P_80_G_50_agg_vs_nsgaii/HV.out.***
- ***./cf6_P_80_G_50_agg_vs_nsgaii/HV.out.***

Se proporciona los archivos ***HV.out*** con 7 columnas:

- La 1ª corresponde a la semilla empleada.
- La 2ª y 3ª corresponde a $f1'$ y $f1''$.
- La 4ª corresponde a HV_{f1} .
- La 5ª y 6ª corresponde a $f2'$ y $f2''$.
- La 7ª corresponde a HV_{f2} .

Archivos de configuración **.in files** empleados para obtener las métricas:

- ***./zdt3_P_100_G_100_agg_vs_nsgaii/in files.***
- ***./zdt3_P_40_G_250_agg_vs_nsgaii/in files.***
- ***./cf6_P_100_G_100_agg_vs_nsgaii/in files.***
- ***./zdt3_P_100_G_100_agg_vs_P_40_G_250_agg/in files.***
- ***./zdt3_P_80_G_50_agg_vs_nsgaii/in files.***
- ***./cf6_P_80_G_50_agg_vs_nsgaii/in files.***

EXPERIMENTOS 1 Y 2

Vamos a tratar de representar gráficamente los tres frentes: el **ideal**, el encontrado por **agregación**, y el proporcionado para **nsgaii**.

Para ello, nos basaremos en la última generación de la ejecución del algoritmo, para 10000 evaluaciones, con parámetros de tamaño para la población equivalente a 100 y número de generaciones equivalente a 100.

Tomamos los valores f1 y f2 para los últimos 100 valores de los archivos **all_popm_seedn.out** y **P_100_G_100_DIM_30_Prob.Neigh_0.15_Prob.Mutat_0.033333333333333333_seed_n.out**.

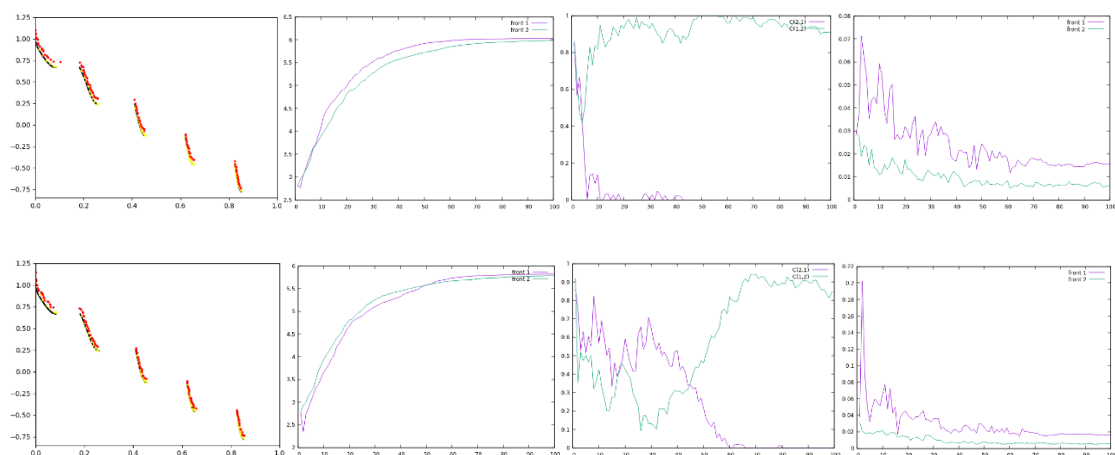
- En **negro**: frente ideal proporcionado por **PF.dat**.
- En **amarillo**: frente encontrado en la última generación de la ejecución del algoritmo de agregación.
- En **rojo**: frente encontrado en la última generación de la ejecución del algoritmo de nsgaii.

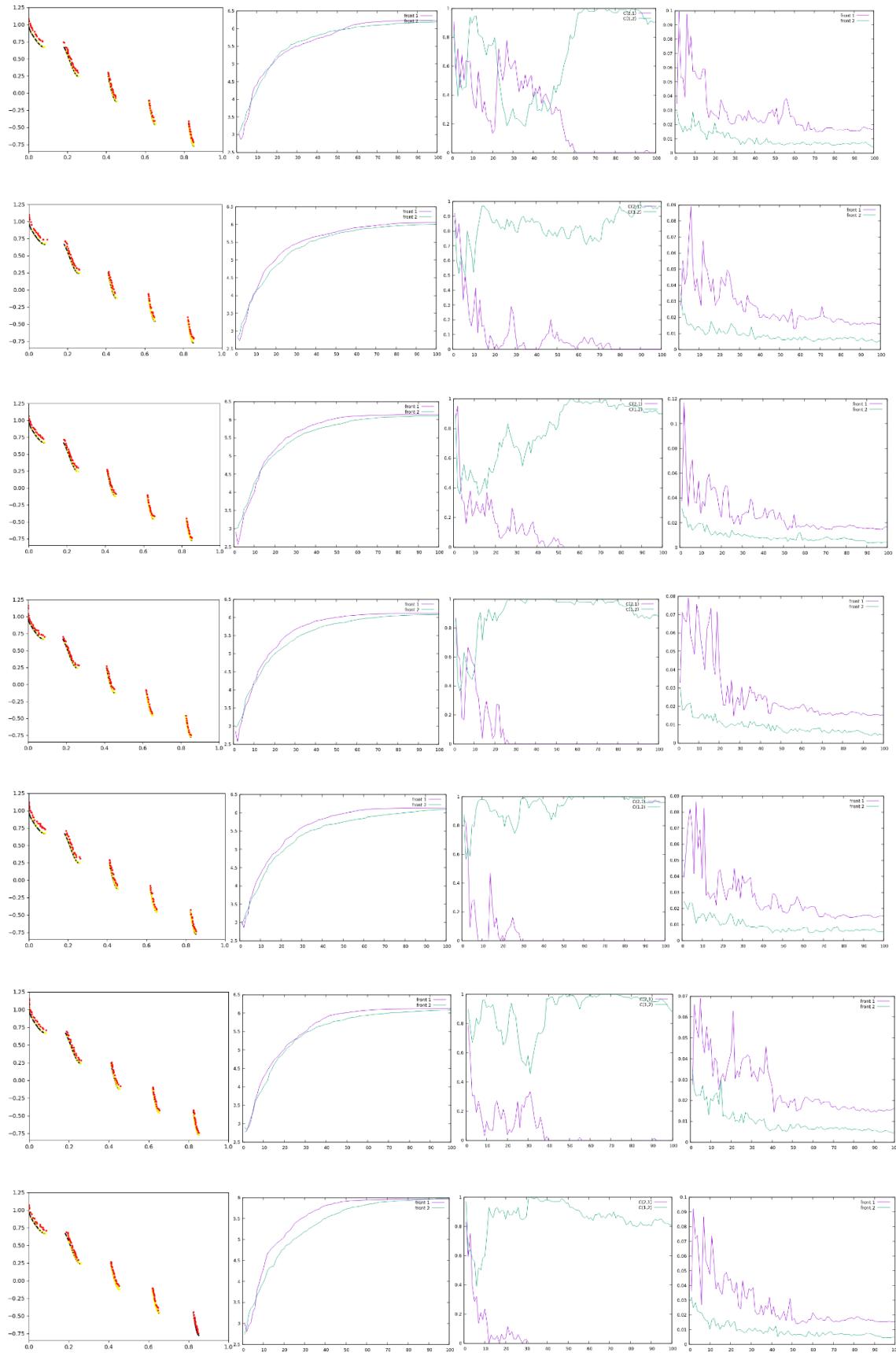
Además, vamos a tratar de analizar la progresión temporal que sigue el cálculo del Hipervolumen, el Coverage Set y el Spacing conforme pasan las generaciones.

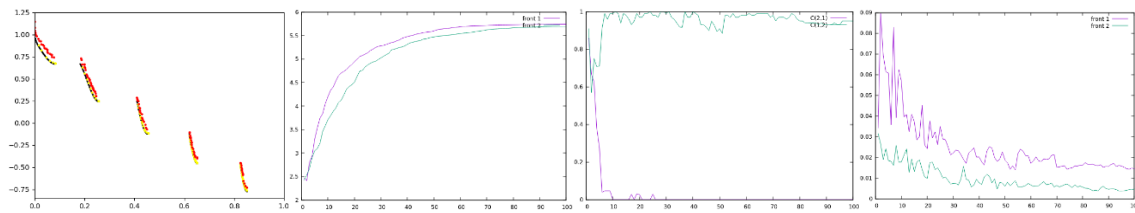
Veamos el avance para semilla (1, ..., 10), población 100 y generación 100:

- **Violeta**: frente obtenido por agregación.
- **Azul turquesa**: frente obtenido por nsgaii.

Los resultados son los siguientes:







Para todas las semillas, se observa que el frente encontrado por agregación es más cercano al ideal que el encontrado mediante nsgaii.

Para las semillas número 2 y número 3, existe un “sube y baja” en ciertos tramos donde ambos algoritmos compiten, y en un tramo uno lo supera para en el tramo siguiente ser superado por el otro.

Mención especial para la representación gráfica correspondiente a la semilla número 9, donde una parte del frente no es alcanzada por agregación. El hipervolumen del frente por nsgaii termina superando al de agregación si hacemos zoom, lo cual puede deberse al último rango de puntos del frente ideal que sí es encontrado por nsgaii pero no por agregación.

El hipervolumen aumenta cuando lo hace la uniformidad de soluciones y cuando estas se alejan del punto de referencia por acercarse al frente Pareto ideal. Valora conjuntamente convergencia, extensión del frente y uniformidad.

Dicho esto, en general se acerca más rápidamente el frente asociado a agregación que el frente asociado a nsgaii.

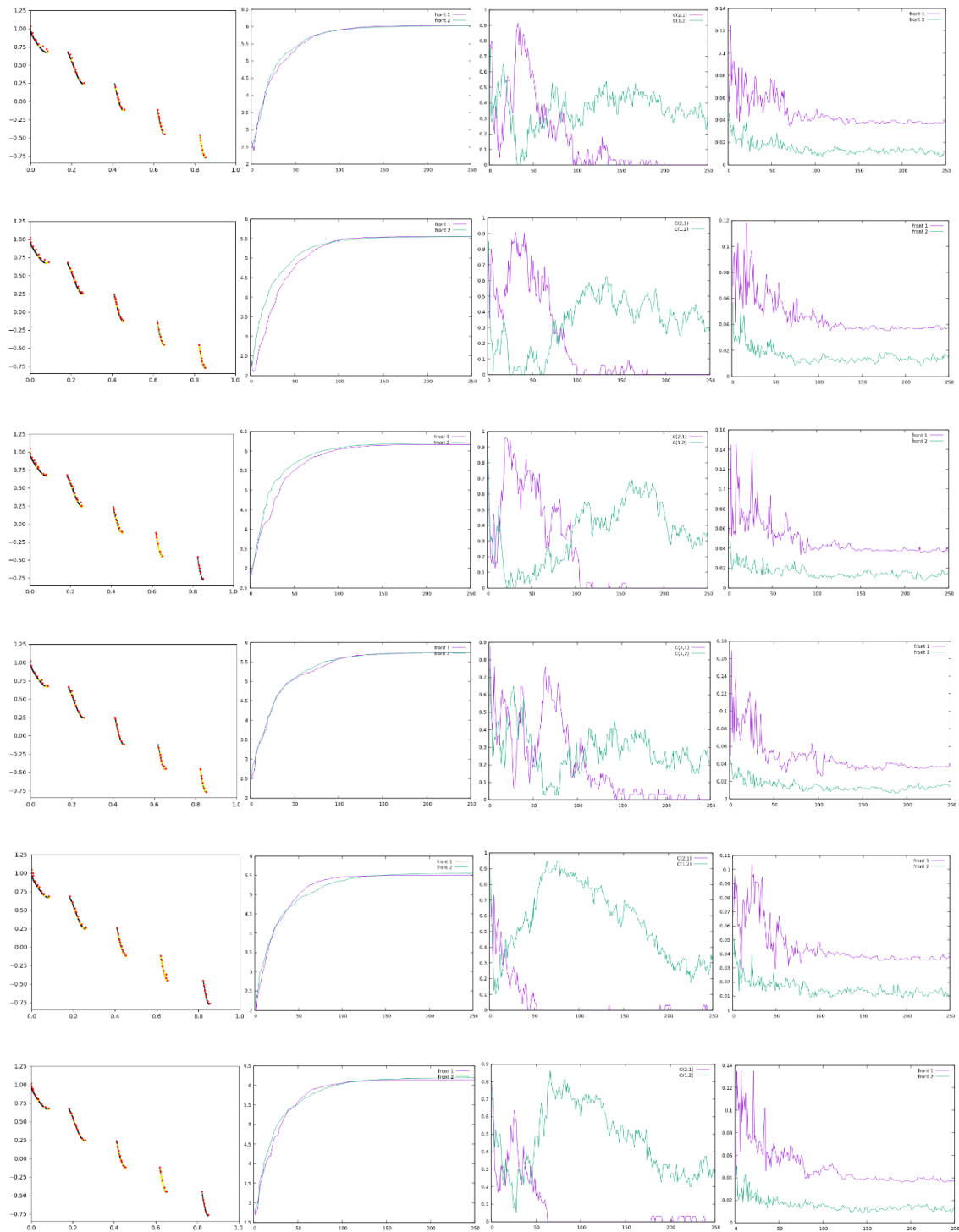
En cuanto a la interpretación del Coverage Set, hay que tener en cuenta lo siguiente:

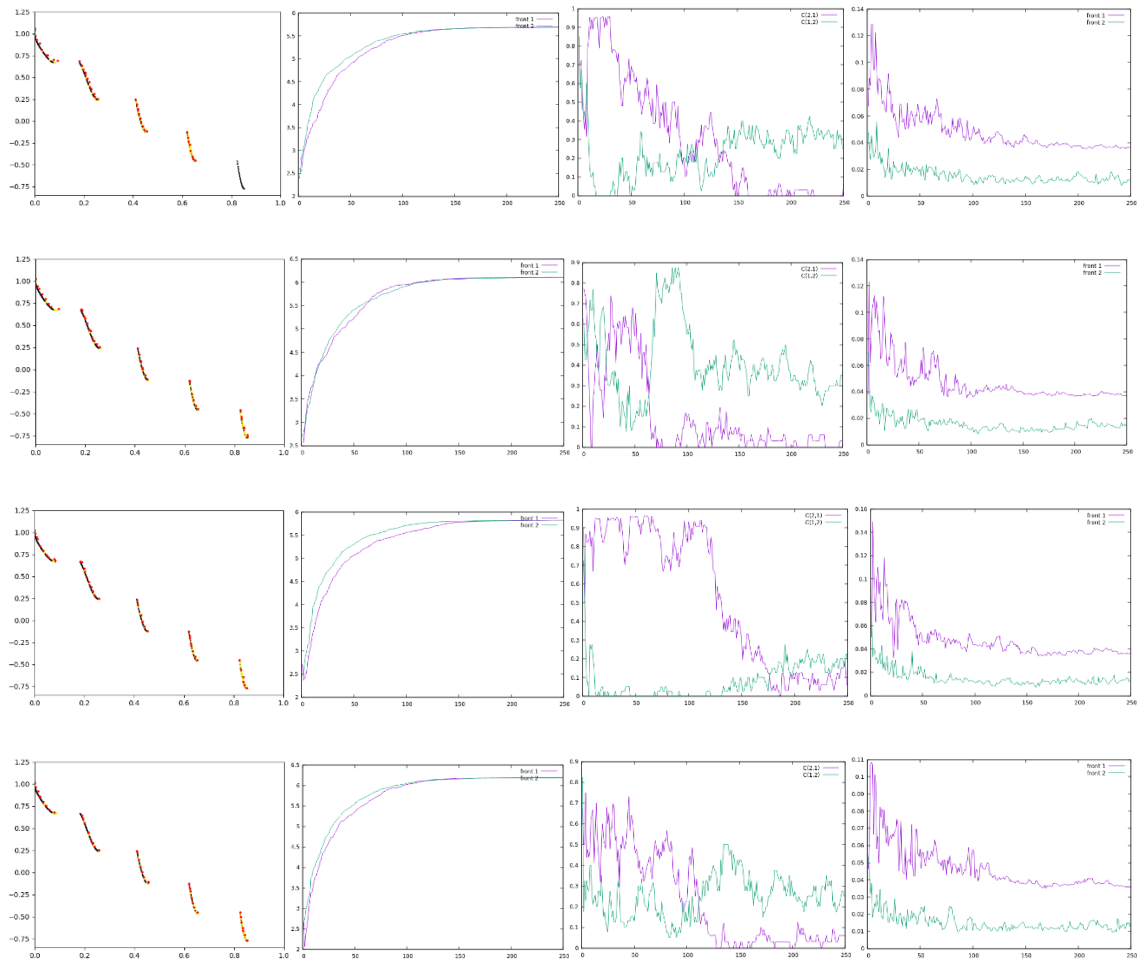
- **Azul turquesa:** % de puntos del frente 2 (nsgaii) cubierto por el frente 1 (agg).
- **Violeta:** % de puntos del frente 1 (agg) cubierto por el frente 2 (nsgaii).

El caso del azul turquesa es muy alto para una buena parte de valores de semilla, a partir de unas pocas generaciones ya va por delante la mayoría de los puntos de agregación frente al nsgaii, reflejando lo que ya vimos gráficamente durante la comparación de frentes entre agregación y nsgaii para población 100 y generación 100, donde la mayoría de puntos amarillos (agregación) estaban por delante, en aproximación al frente ideal, de nsgaii. Al ir por delante, cubren al otro frente. Lo que se refleja es que ese proceso lo hace muy rápidamente, esto es, en muy pocas generaciones.

EXPERIMENTOS 3 Y 4

Veamos ahora si ocurre lo mismo variando los parámetros poblacionales y de generación a 40 individuos y 250 generaciones:





En este experimento, si sólo nos fijásemos en el gráfico de frentes, no quedaría del todo claro que algoritmo funciona mejor. En términos de cercanía al frente pareto ideal, parece que los dos son muy parecidos, quizás algo mejor el de agregación, pero ya no se ve tanta diferencia como la que se veía en el experimento anterior.

Atendiendo a lo que dicen las métricas, el Coverage Set indica que los frentes obtenidos por agregación en las últimas generaciones tienen mayor número de puntos sin cubrir por puntos del frente nsgaii, luego en términos de cercanía al frente Pareto ideal, se han obtenido mejores resultados con agregación.

Sin embargo, no hay esa diferencia tan grande que percibíamos en gran parte de las semillas en el experimento anterior donde con poco número de generaciones, el de agregación ya se ponía por delante del nsgaii. Aquí el proceso es más comedido.

Lo más llamativo es que el último rango de puntos del frente pareto ideal cuesta encontrarlo, y para las semillas que cuesta encontrarlo, hasta en 3 de ellas ha dado mejor resultado con el nsgaii, ofreciendo nulos resultados con el de agregación.

Sin embargo, atendiendo a lo que nos dice el gráfico de Spacing, no podemos afirmar que el frente obtenido por agregación esté más distribuido uniformemente que el obtenido por nsgaii. Es de suponer que nsgaii tiene cierta ventaja en este aspecto frente a agregación, por lo ya mostrado en los ejemplos gráficos donde ciertas zonas del frente ideal eran complicadas de encontrar mediante agregación, a diferencia de nsgaii que sí era competente:

Cuanto más pequeña es la desviación estándar de individuos de la población, menor el espaciado y, en consecuencia, mayor es la uniformidad. Es lo que vemos con esta gráfica, el espaciado asociado al frente por nsgaii indica que el conjunto de individuos de nsgaii está distribuido más uniformemente que el asociado a agregación

El bajo número de individuos para la población limita de alguna manera la búsqueda, aunque parece que penaliza más en agregación que en nsgaii. En el anterior experimento, para 100 individuos y 100 generaciones, también tuvimos un caso parecido, concretamente para la semilla número 9, donde se volvió a ver reflejado que hay ciertas zonas del frente que cuesta más encontrarlas con agregación, pero disminuyendo la población parece que se acrecienta dicho fenómeno.

EXPERIMENTO 5

Comprobemos pues, la velocidad de convergencia del algoritmo basado en agregación mediante la descomposición de Tchebycheff frente a la de NSGAII.

Para ello, nos ayudaremos del archivo **PF.dat** para calcular la distancia generacional invertida (**IGD**):

$$IGD(A, P) = \frac{\sum_{p \in P} MIN_{distance}(p, A)}{|P|}$$

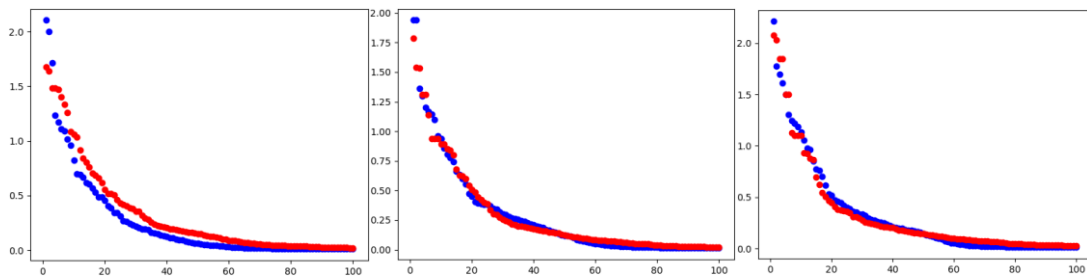
Para cada punto del frente ideal hallaremos la mínima distancia al conjunto de puntos del frente obtenido, esto es, la distancia al punto del frente obtenido que se encuentra más cerca del punto del frente ideal. Se sumaran todas las distancias y se dividirá por el número de sumandos.

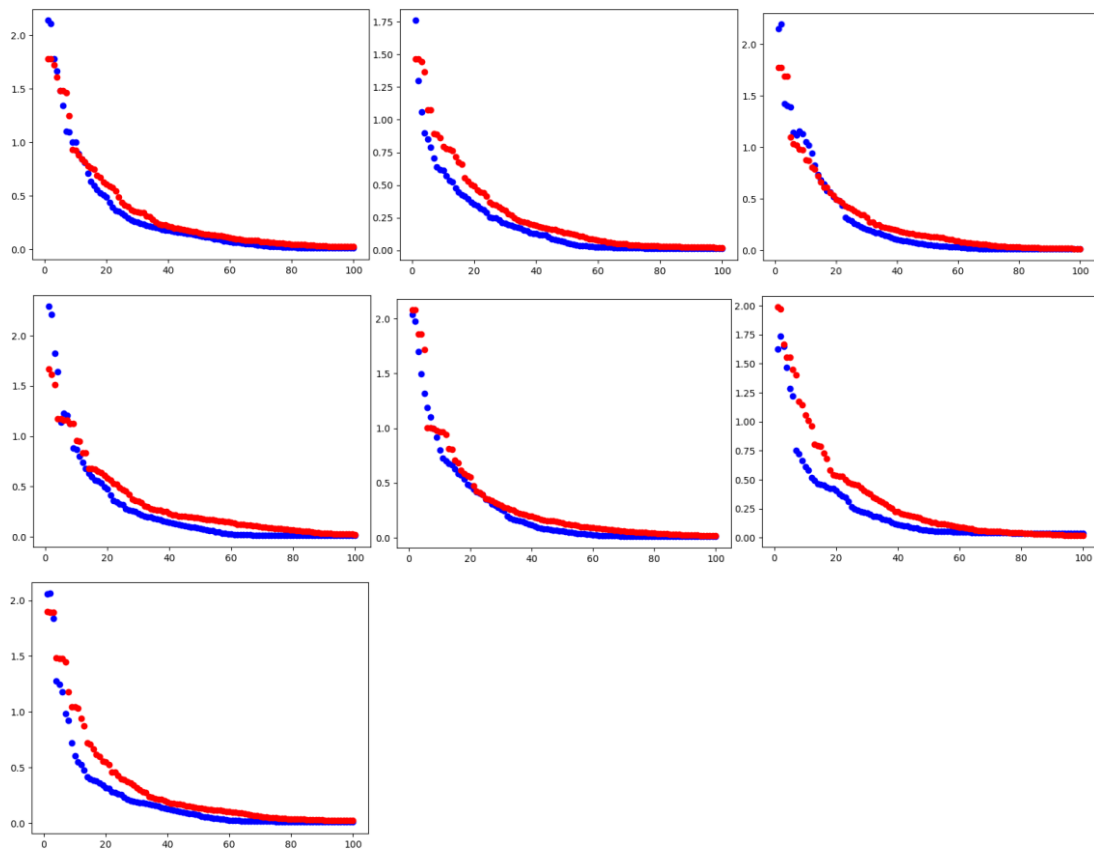
Dicho proceso algorítmico se aplicará tanto para el frente obtenido por el algoritmo basado en agregación mediante la descomposición de Tchebycheff, como el frente obtenido por el algoritmo NSGAII, a partir de los datos proporcionados por el profesor. Ambos cálculos se llevarán a cabo en cada una de las generaciones (**100 generaciones, 100 individuos**), y se aplicará para 10 ejecuciones del algoritmo, una para cada semilla.

Los resultados obtenidos son los siguientes:

- **Azul:** algoritmo basado en agregación mediante la descomposición de Tchebycheff
- **Rojo:** algoritmo NSGA II

De izquierda a derecha, de arriba hacia abajo, variando la semilla en (1, ..., 10):





Como puede observarse en las gráficas, las curvas azules se hacen pequeñas más rápidamente que las curvas rojas, estando estas últimas casi siempre por encima de las azules, lo cual refleja que los individuos del algoritmo basado en agregación se aproximan más rápidamente al frente pareto ideal que el NSGAI.

Para la semilla número 2 y 3, se puede observar que en ciertos tramos uno es mejor que en otro, y en otros tramos es a la inversa. En ciertos instantes uno está más cerca del frente ideal, para a continuación revertirse la situación. Hemos visto una situación similar durante el estudio del Hipervolumen en el experimento 1.

Para la semilla número 9, vemos como al final del todo la curva roja termina por quedar por debajo de la curva azul. Este fenómeno también lo hemos visto durante el estudio del hipervolumen del experimento 1, donde haciendo zoom el hipervolumen de nsgai terminaba por superar al de agregación, y esgrimimos que podía deberse a que nsgai sí encontraba el último conjunto de puntos del frente ideal para dicha semilla, a diferencia de agregación.

EXPERIMENTO 6 Y 7

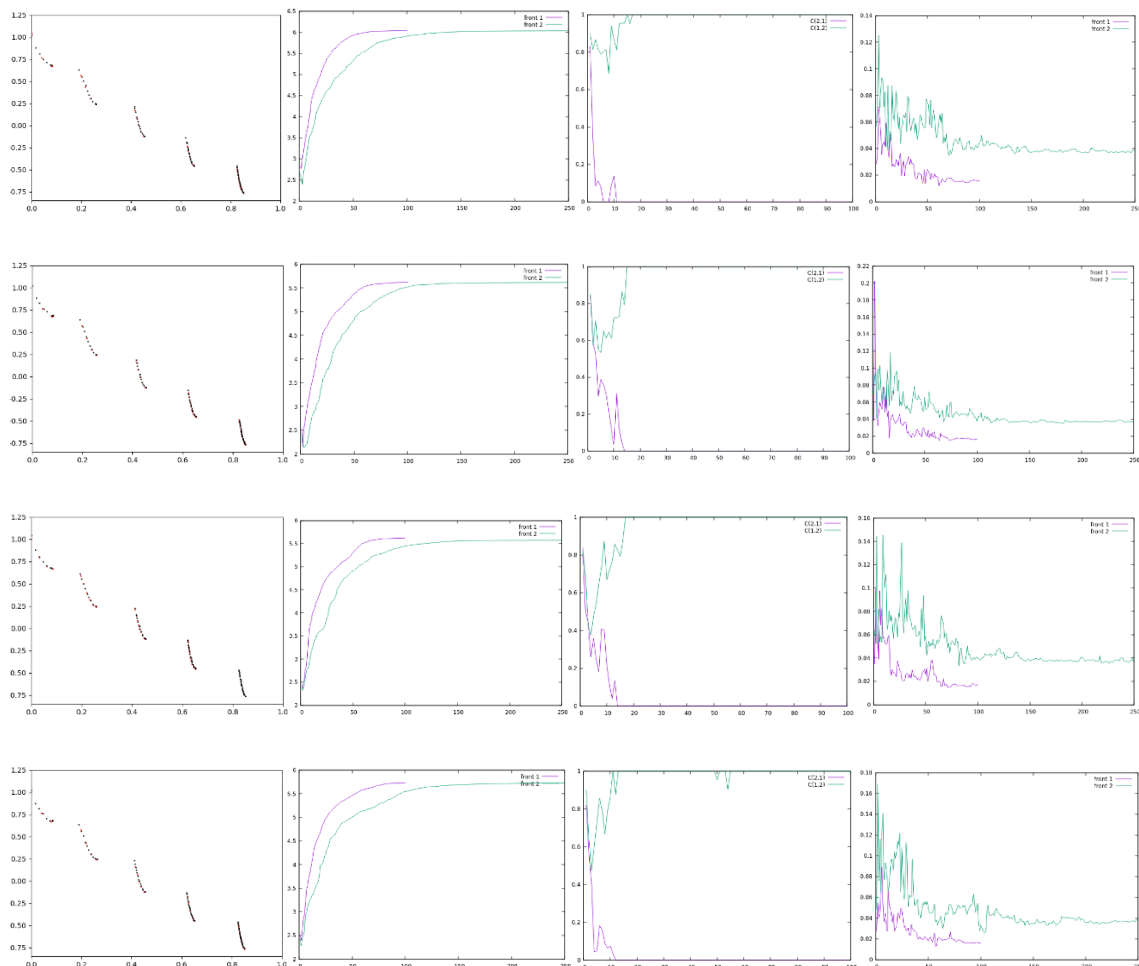
Vamos a tratar de analizar, dentro de lo que es el método de agregación, qué valores poblacionales y de generación dan mejores resultados

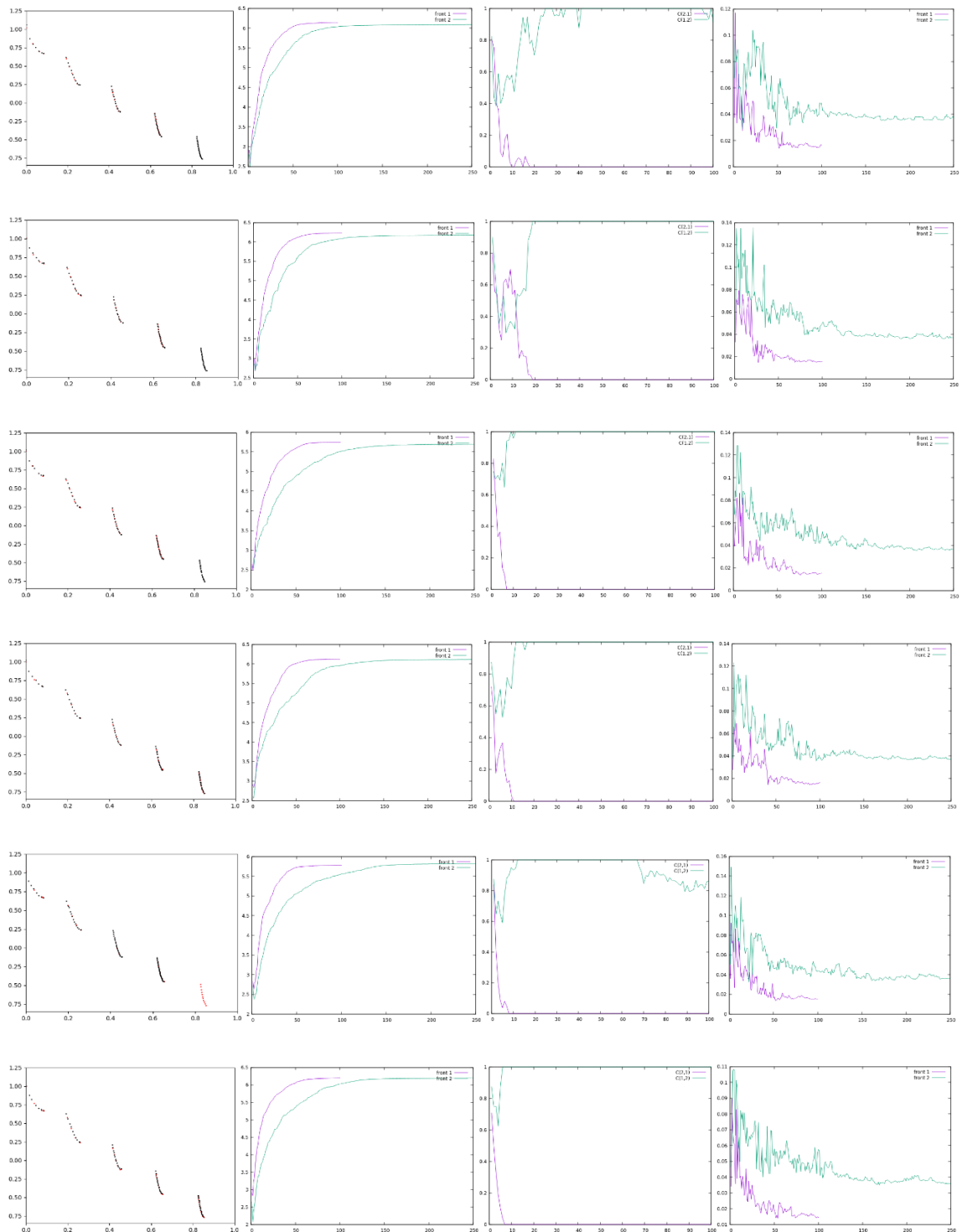
Se tratará pues de comprender si este método penaliza el tener pocos o muchos individuos por generación.

Enfrentemos dos ejecuciones:

- En **rojo**: 40 individuos, 250 generaciones
- En **negro**: 100 individuos, 100 generaciones

Para semillas (1, ..., 10):





A simple vista no parece que merezca la pena la configuración P40G250 teniendo la configuración P100G100 que parece dar resultados similares, pero con un mayor número de puntos, cubriendo por tanto mayor cantidad de frente, salvo alguna semilla como la 9 donde puede ocurrir que no encontremos cierta zona del frente ideal.

En cuanto a las métricas de calidad, en uniformidad gana P100G100 frente a P40G250. Lo mismo en Coverage Set.

En Hipervolumen, se muestra una convergencia más rápida por parte de P100G100.

En definitiva, no ofrece ninguna ventaja la configuración P40G250 teniendo la configuración P100G100.

EXPERIMENTO 8 Y 9

Vamos a probar ahora con 4000 evaluaciones, y comparar qué algoritmo es mejor con dicha limitación en el número de evaluaciones, si nsgaii o agregación.

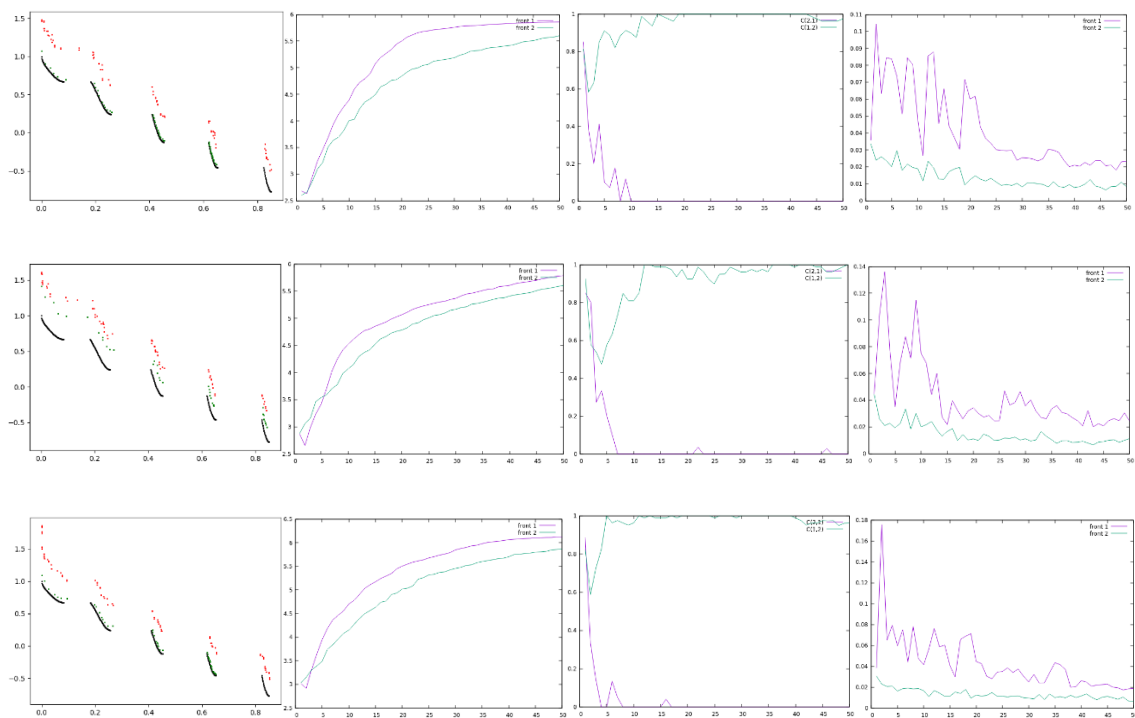
Datos a tener en cuenta del experimento:

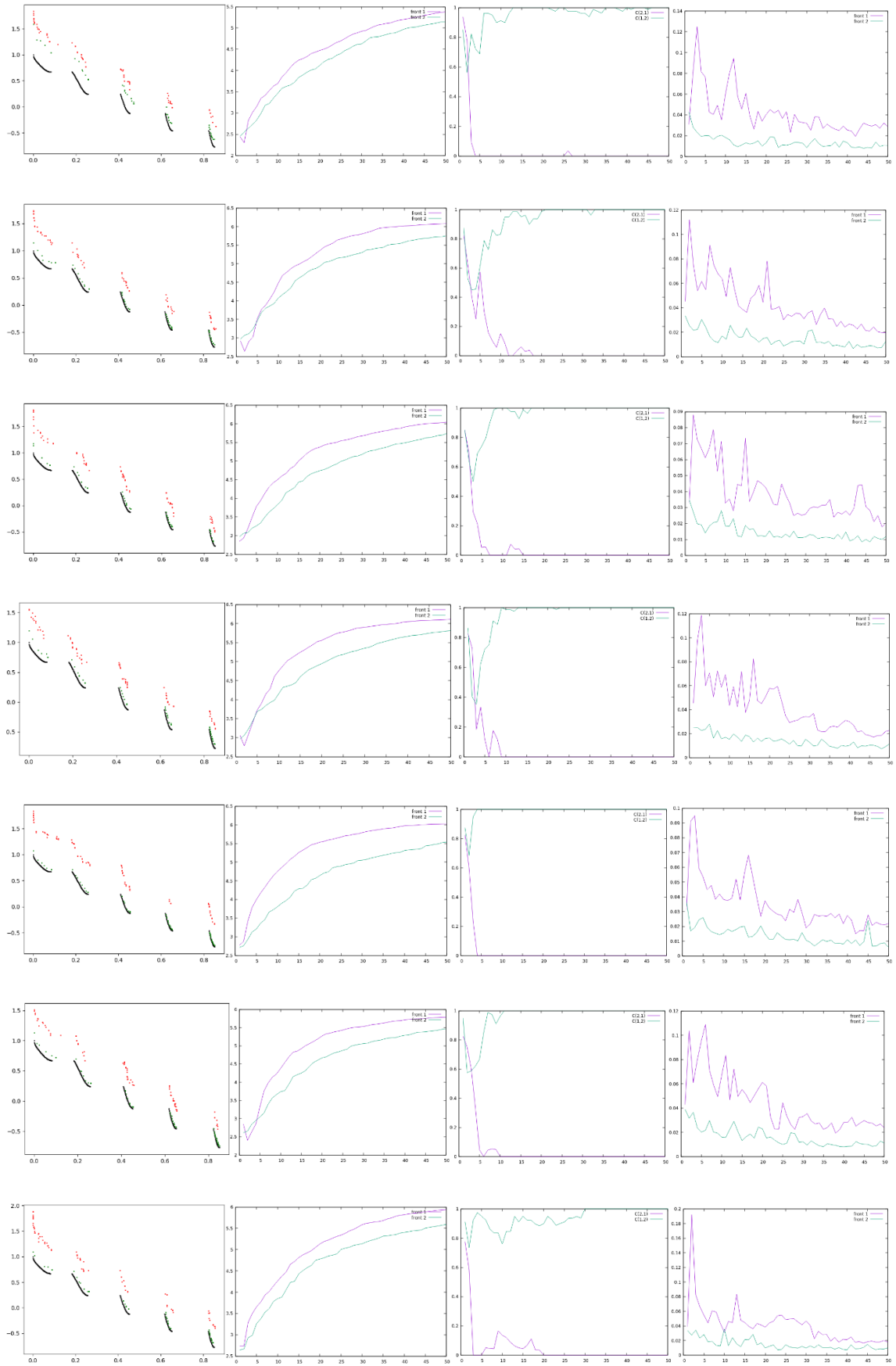
- Problema zdt3
- 80 individuos
- 50 generaciones
- 30% vecindad
- 3'3% mutación (1/30)

Enfrentemos dos ejecuciones:

- En **verde**: frente encontrado en la última generación de la ejecución del algoritmo de agregación.
- En **rojo**: frente encontrado en la última generación de la ejecución del algoritmo de nsgaii.
- En **negro**: frente ideal proporcionado por *PF.dat*.

Para semillas (1, ..., 10):





Agregación vence en cobertura, mientras que nsgaii lo hace en espaciado de las soluciones.

Según las representaciones de los frentes, agregación se acerca bastante más al frente Pareto ideal, aunque menos puntos parecen llegar, o más bien que gran parte de su población se termina concentrando en las franjas más hacia la derecha del frente Pareto ideal, dejando más sola a las de la izquierda.

Mención al hipervolumen, el cual mantiene una tendencia alcista para muchas semillas, lo que puede ser indicativo de que con unas cuantas generaciones más, veamos soluciones bastantes mejores.

EXPERIMENTO 10

Para el manejo de restricciones, se ha propuesto la técnica basada en penaltis:

$$f'i(X) = fi(X) + \sum_j CVj , \quad i \in \{1, 2\}, j \in \{1, 2\}$$

$$CVj = Wj * |Rj|, \quad Wj \in \{0, 1\}, Rj \in (-\infty, 0]$$

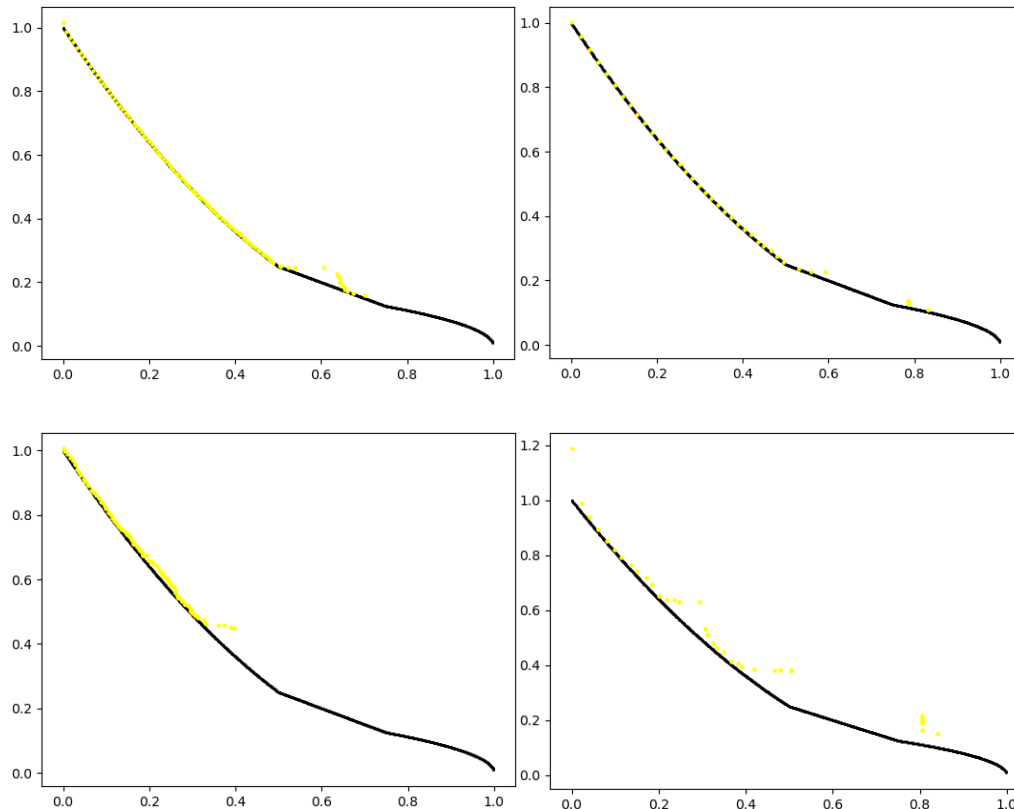
$$Wj \equiv Rj < 0, \quad Wj \in \{0, 1\}$$

Calculamos la magnitud de la violación cometida (cuánto se está violando), y empleamos el peso como función de activación de la aplicación de la penalización sobre la función objetivo.

Veamos 4 ejemplos rápidos:

1. Población: 100, Generaciones: 100, Dimensión: 4, % Vecindad: 15%, % Mutación: 1/4
2. Población: 40, Generaciones: 250, Dimensión: 4, % Vecindad: 15%, % Mutación: 1/4
3. Población: 100, Generaciones: 100, Dimensión: 16, % Vecindad: 15%, % Mutación: 1/16
4. Población: 40, Generaciones: 250, Dimensión: 16, % Vecindad: 15%, % Mutación: 1/16

De izquierda a derecha, de arriba abajo: 1, 2, 3 y 4:



Llevando a cabo una lectura rápida de los gráficos, parece existir un buen funcionamiento con 4 dimensiones, tanto para 100 como para 40 individuos. Con 16 dimensiones, parece que trabaja peor.

En cualquier caso, y para todos ellos, parece que cuesta encontrar ciertas zonas del frente. Las zonas que van hacia la derecha no las encuentra, mientras que para las zonas de la izquierda y las del centro no parece tener mucho problema.

EXPERIMENTO 11 Y 12

Vamos a tratar de medir comparativamente cuál de los 2 algoritmos funciona mejor para el problema CF6.

Para ello emplearemos las medidas de **Spacing** y **Coverage Set**, la primera para analizar cómo de uniforme están distribuidas las soluciones que vamos encontrándonos. Un espaciado bajo es una desviación estándar entre soluciones también baja, lo cual nos interesa. La segunda la emplearemos para dar una medida de qué frente, de entre los dos, tiene mayor cantidad de individuos mejores, entendiéndose por mejores aquellos que no son cubiertos por el frente contrario.

Datos a tener en cuenta del experimento:

- 100 individuos
- 100 generaciones
- 4 dimensiones

Para cada semilla (1, ..., 10), calculamos:

- $f1' = \text{worst}\{f1i, i \in \{1, 100 * 100\}\}, \text{para AGG}$
- $f2' = \text{worst}\{f2i, i \in \{1, 100 * 100\}\}, \text{para AGG}$
- $f1'' = \text{worst}\{f1i, i \in \{1, 100 * 100\}\}, \text{para NSGAI}$
- $f2'' = \text{worst}\{f2i, i \in \{1, 100 * 100\}\}, \text{para NSGAI}$

Nuestro punto de referencia final para Hipervolumen será:

- $HV_{f1} = \text{worst}\{f1', f1''\}$
- $HV_{f2} = \text{worst}\{f2', f2''\}$
- $HV_{final} = (HV_{f1}, HV_{f2})$

Dado que estamos minimizando:

- $\text{worst}(a_1, a_2, \dots, a_n) = \max \{a_1, a_2, \dots, a_n\}$

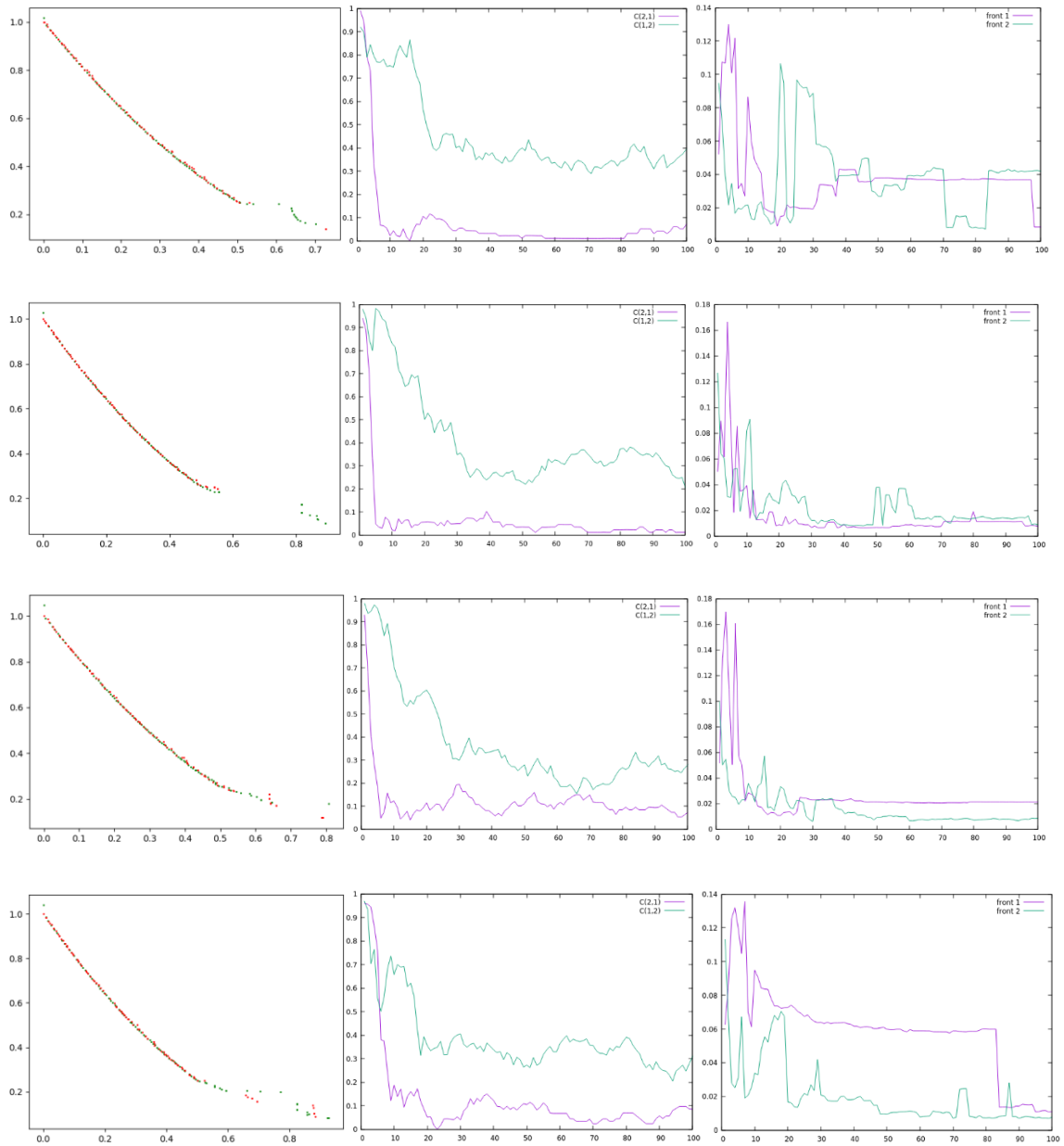
Se proporciona un archivo **HV.out** con 7 columnas:

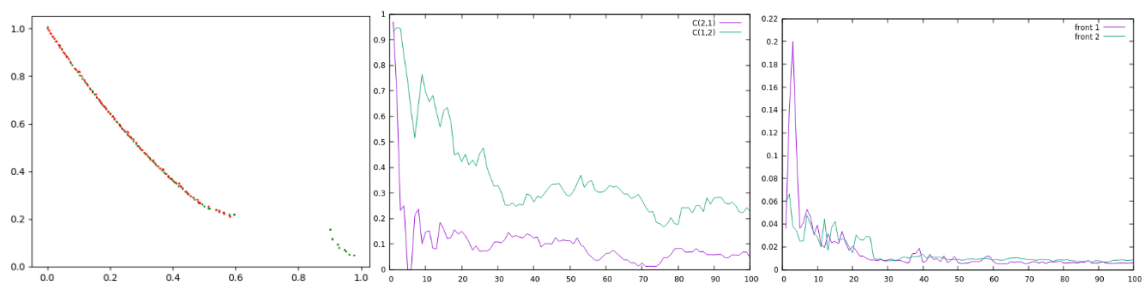
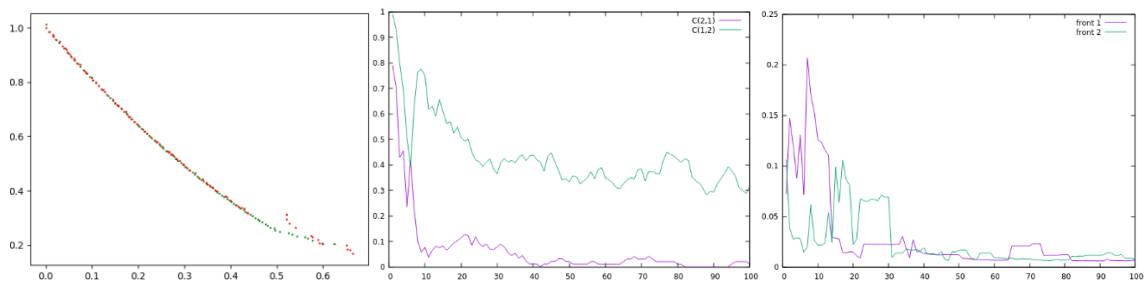
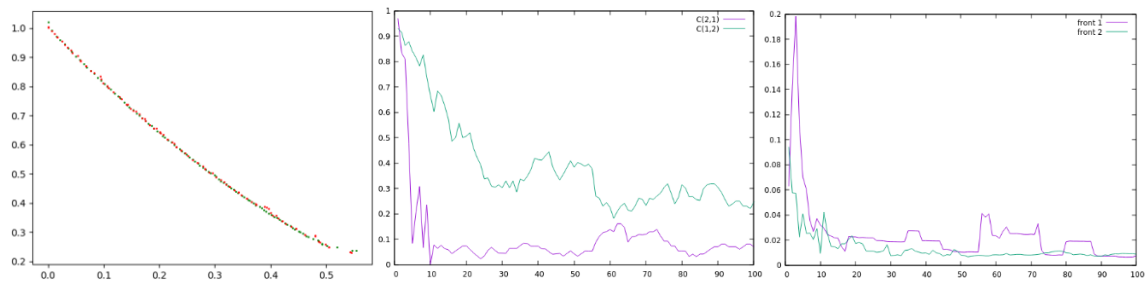
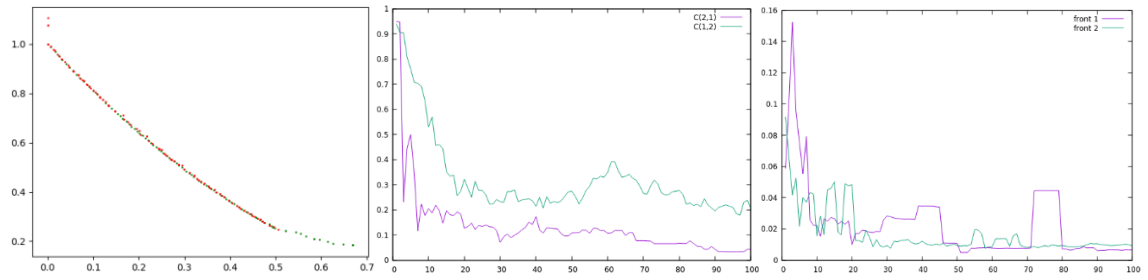
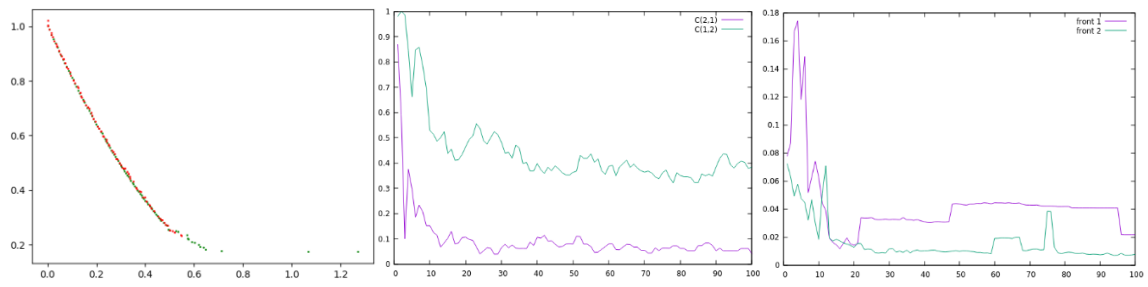
- La 1ª corresponde a la semilla empleada
- La 2ª y 3ª corresponde a $f1'$ y $f1''$

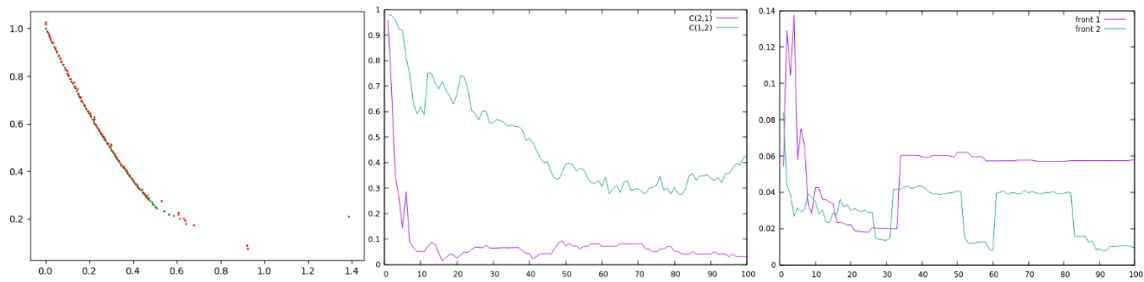
- La 4ª corresponde a HV_{f_1}
- La 5ª y 6ª corresponde a f_2' y f_2''
- La 7ª corresponde a HV_{f_2}

Se proporcionan 10 archivos con extensión **.in**, uno por cada semilla

- En **verde**: frente por agregación (frente 1)
- En **rojo**: frente por nsgaii (frente 2)







Según el análisis gráfico de Coverage Set, para las 10 semillas se cumple que el porcentaje de puntos del frente obtenido por agregación que son cubiertos por puntos del frente de nsgaii es menor que el caso contrario, ergo podemos concluir que el frente obtenido por agregación es más cercano al ideal en los 10 casos estudiados.

En cuanto a la uniformidad que sigue las soluciones que se van encontrando, hay de todo. En algunos casos se muestra una clara tendencia disminuir el spacing más rápidamente en nsgaii, como se refleja en semillas 4, 5, 6 y 7. En otros casos baja más rápido el de agregación, como es la semilla 2. Otros casos son muy parejos, como la semilla 9 donde la progresión es muy parecida. La semilla 1 y 10 son las que muestran comportamientos más dispares entre ambos algoritmos, y al mismo tiempo son las que tienen las diferencias más abultadas en la última generación, puesto que para el resto no suele superar el 0.02 de valor.

Se puede concluir que, si bien para el Coverage Set parece haber un comportamiento homogéneo, esto no se da para Spacing que parece ser heterogéneo.

EXPERIMENTO 13 Y 14

Vamos a probar con qué algoritmo se consiguen mejores resultados, dentro de la limitación de 4000 evaluaciones como máximo, y para el problema de las 16 dimensiones.

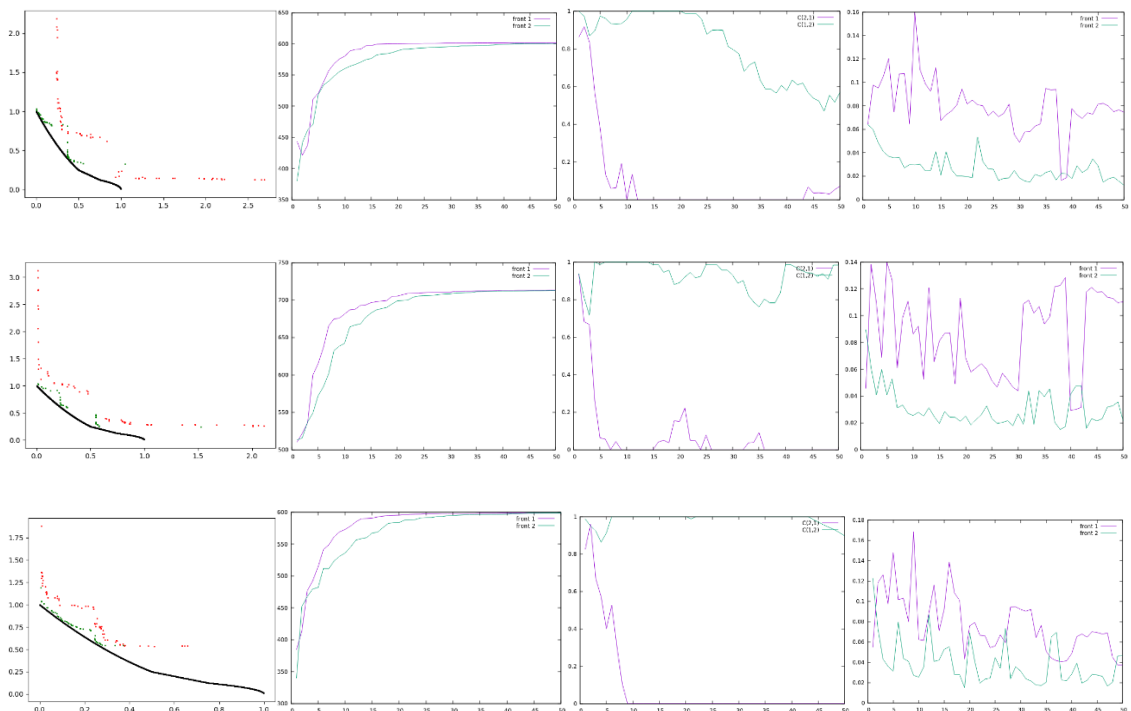
Datos a tener en cuenta del experimento:

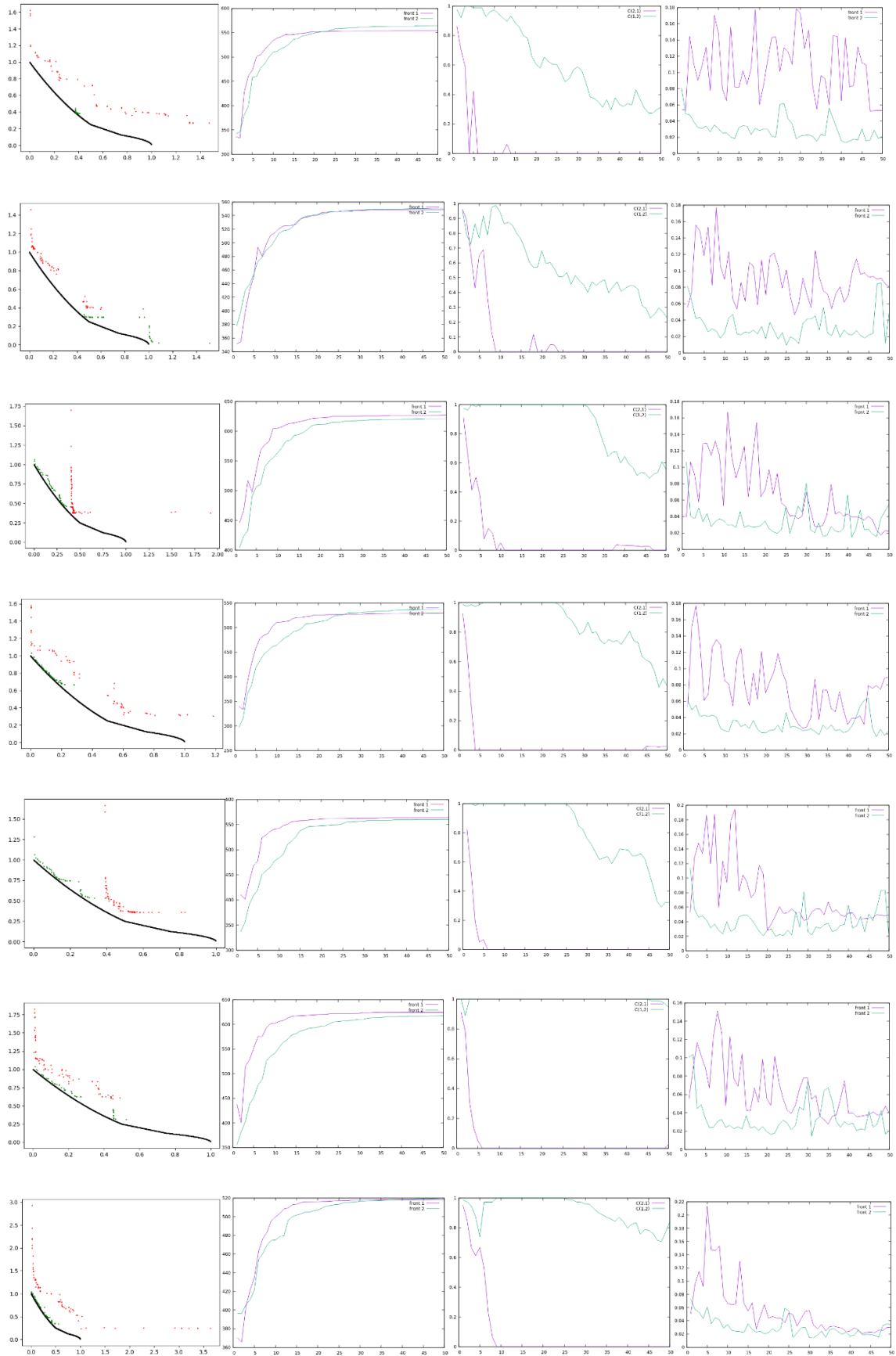
- Problema cf6
- 16 dimensiones
- 80 individuos
- 50 generaciones
- 15% vecindad
- 12'5% mutación

Enfrentemos dos ejecuciones:

- En **verde**: frente encontrado en la última generación de la ejecución del algoritmo de agregación.
- En **rojo**: frente encontrado en la última generación de la ejecución del algoritmo de nsgaii.
- En **negro**: frente ideal proporcionado por *cf6_16d_PF.dat*.

Para semillas (1, ..., 10):





Los resultados son muy dispares.

Independientemente de la semilla que se utilice, cuando el problema es de 16 dimensiones apenas se abarca un porcentaje pequeño del frente Pareto ideal.

Para algunas semillas, sí se consiguen buenos resultados acercándose a ese pequeño porcentaje del total del frente Pareto ideal.

Si tomamos en consideración la totalidad del frente Pareto ideal, los resultados no son buenos para ninguno de los 2 algoritmos

El nsgaii parece ganar en uniformidad, mientras que agregación parece ganar en cobertura, aunque hay que matizar que el hecho de que agregación avance concentrándose en una única zona frente a nsgaii, que parece estar más esparcido, tenga que ver a la hora de valorar la cobertura.

Para ciertos valores de la semilla, se puede ver un decaimiento en picado en el coverage set, a partir de cierta generación, de puntos del frente de nsgaii cubiertos por el de agregación, lo cual puede deberse a que al estar más esparcido nsgaii, termina por acercarse a zonas del frente Pareto ideal a las que ni tan siquiera agregación ha llegado por centrarse en una pequeña franja.