



# **Extremal Optimization aplicado al problema multi-objetivo de planificación permutativa de producción en línea utilizando una evaluación de aptitud inseparable.**

**Miguel Ángel Briones Méndez**

INFORME DE PROYECTO DE TÍTULO PARA OPTAR AL TÍTULO DE  
**INGENIERÍA CIVIL INFORMÁTICA**

Profesor guía: **Pedro Gómez Meneses**

Profesor Informante: **José Abreu Salas**

Departamento de Ingeniería Informática

Facultad de Ingeniería

Concepción, agosto, 2017

---

## RESUMEN

El *Permutation Flow-Shop Scheduling Problem* (PFSSP) es una clase de problemas de programación en el cual un conjunto de trabajos permutables (el orden de ingreso puede ser modificado) debe ser procesado por un conjunto de máquinas fijas optimizando el tiempo de ejecución por máquina y el tiempo total del proceso completo (desde el primer hasta el último trabajo finalizado) minimizando los tiempos de espera y de inactividad en cada uno de ellos. Este tipo de problemas puede aplicarse tanto a las instalaciones de producción como a los diseños informáticos.

En el presente informe se investigan los resultados para el PFSSP utilizando la innovadora metaheurística denominada *Extremal Optimization* (EO), desarrollada recientemente y basada en la teoría de co-evolución y el fenómeno del equilibrio puntuado de Bak-Sneppen que no considera la evolución de forma lineal sino más bien ramificada, e incluyendo, como método de selección, el concepto de feromona extraído del Algoritmo de Colonia de Hormigas (ACO).

Como el informe es investigativo se realiza análisis a 4 variantes de EO. Se comienza aplicando EO en su forma pura y luego con búsqueda local, ambas con un criterio de selección propio, y luego se utiliza como método de selección el concepto de feromona analizando los datos obtenidos tanto en su forma pura y búsqueda local (con feromona). Con el procedimiento anterior se logra estudiar con más detenimiento el cambio que surge al aplicar la feromona a EO. Si bien el proceso de búsqueda puede variar dependiendo del individuo a cargo, la feromona aplicada a EO no se ve afectada.

---

## ABSTRACT

Permutation Flow-Shop Scheduling Problem (PFSSP) is a class of programming problems in which a set of interchangeable tasks (the order of entry can be modified) must be processed by a set of static machines optimizing the execution time by machine and the total time of the complete process (from the first to the last completed task) minimizing the waiting and inactivity times in each of them. This type of problem can be applied to both production facilities and computer designs.

In the present report, the results are investigated for the PFSSP using the innovative metaheuristic called Extremal Optimization (EO), recently developed and based on the co-evolutionary theory and Bak-Sneppen's punctuated equilibrium phenomenon that does not consider the evolution of form Linear but rather branched, and including, as a selection method, the pheromone concept extracted from the Ant Colony Algorithm (ACO).

As the report is investigative, analyzes are performed on four EO variants. It starts by applying EO in its pure form and then with local search, both with its own selection criteria, and then the pheromone concept is used as a selection method, analyzing the data obtained both in its pure form and in local search (with pheromone). With the above procedure, it is possible to study with more detail the change that arises when applying the pheromone to EO. While the search process may vary depending on the individual in charge, the pheromone applied to EO is not affected.

---

## AGRADECIMIENTOS

*Agradecer a todos los profesores y ayudantes que me brindaron sus valiosos conocimientos a lo largo de mi estadía por la carrera, por nombrar a algunos; Maritza Galindo, Marcela Torrejón, Ruth Sandoval, Denise Chamorro entre otros y, en especial, a los pertenecientes al Departamento de Ingeniería Informática; Marcia Muñoz, Mariella Gutiérrez, Emma Chávez, Pedro Rossel y a mi profesor guía Pedro Gómez los cuales me apoyaron en momentos muy complicados. No puedo olvidar a una profesora que, a pesar de no haber tenido la dicha de ser su alumno, siempre sentí su gran apoyo, me refiero a Claudia Martínez. A mis compañeros de universidad, de carrera, de coro y en este último año los de equipo de fútbol. A mi hermana Maritza que siempre ha sido una inspiración, superando todas las dificultades que le ha puesto la vida, espero algún día llegar a tener su fortaleza. No puedo olvidar a aquéllos que me criaron, enseñaron valores y estilos de vida, de los cuales aprendí el significado de luchar por lo que deseas, cumplir tus promesas, dar todo hasta el final sin arrepentimientos, honestidad, respeto, empatía, proteger a todos los seres vivos y que amar no es una emoción efímera sino un estilo de vida, ellos son mis queridos Animés (Dragon Ball, Los Súper Campeones, Los caballeros del Zodiaco, Cazador X, Final Fantasy, entre otros).*

***“Si de verdad te esfuerzas para conseguir algo no hay nada que pueda detenerte. Parece un fracaso porque la gente se rinde a la mitad. ¡Carga con la derrota, la desesperación y el fracaso! todo es parte de la próxima victoria. Nada es en vano, ¡Tus esfuerzos no son en vano!”*** (Una de las tantas enseñanzas que me ha dado el animé)

*Finalmente, a todos los seres que se han cruzado en mi vida, pues todos ellos tuvieron influencia en el yo de hoy, les digo **Gracias, Gracias y Gracias.***

---

## ÍNDICE

1	Introducción .....	1
1.1	Descripción .....	1
1.2	Justificación .....	2
1.3	Delimitación .....	3
1.4	Objetivo General .....	3
1.5	Objetivos Específicos.....	3
1.6	Metodología .....	4
2	Marco Teórico .....	6
2.1	Extremal Optimization (EO).....	6
2.2	Permutation Flow-Shop Scheduling Problem (PFSSP) .....	8
2.3	Optimización Multi-Objetivo (MOP) .....	9
2.4	Algoritmo Colonia de Hormigas (ACO).....	10
3	Estado del Arte .....	13
3.1	Balance between Genetic Search and Local Search in Memetic Algorithms for Multiobjective Permutation Flow-Shop Scheduling. ....	13
3.2	Aplicación de un algoritmo ACO al PFSSP con tiempos de preparación dependientes de la secuencia y minimización de Makespan. ....	14
3.3	EO aplicado a Multi-Objective Permutation Flow-Shop Scheduling Problem .....	15
4	Implementación .....	18
4.1	Modelamiento del Problema .....	18
4.2	Fitness .....	18
4.3	Búsqueda Local.....	22
4.4	EO Aplicado a MOPFSSP.....	23
5	Experimentos .....	24
5.1	Introducción .....	24
5.2	Set de Pruebas ( <i>Benchmarks</i> ).....	24
5.3	Resultados .....	24
5.3.1	EO_Puro.....	26
5.3.2	EO_Puro_Busqueda.....	50
5.3.3	EO_Feromonas .....	77
5.3.4	EO_Feromonas_Busqueda .....	101
5.4	Análisis.....	126
5.4.1	Makespan .....	126

---

5.4.2	Tardiness .....	130
5.4.3	Análisis de feromonas .....	134
5.4.4	Dos métodos de Búsqueda Local .....	135
6	Conclusión .....	138
6.1	Contribuciones .....	138
6.2	Trabajo Futuro.....	138
7	Bibliografía .....	140

---

## ÍNDICE DE FIGURAS

FIGURA 2.1	SEUDO-CÓDIGO EO DE MÍNIMO COSTE ( $T = \tau$ ).....	7
FIGURA 2.2	EJEMPLO DE DOMINANCIA PARA 2 OBJETIVOS (ELABORACIÓN PROPIA).....	10
FIGURA 4.1	EJEMPLO DEL VECTOR SOLUCIÓN PARA EL MOPFSSP. (GÓMEZ MENESES P. S., 2012) .....	18
FIGURA 4.2	EJEMPLO DE ACTUALIZACIÓN DE FEROMONAS CON N=6.....	21
FIGURA 4.3	MATRIZ QUE MEMORIZA LA POSICIÓN DE LAS SECUENCIAS Y SUS JOB.....	21
FIGURA 5.1	VARIACIÓN DE ITERACIONES PARA EO_PURO 20-20.....	26
FIGURA 5.2	VARIACIÓN DE SEMILLAS PARA EO_PURO 20-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	27
FIGURA 5.3	VARIACIÓN DE $\tau$ PARA EO_PURO 20-20 CON SEMILLAS 15 Y 20 Y 1.000.000 DE ITERACIONES .....	29
FIGURA 5.4	VARIACIÓN DE $\tau$ PARA EO_PURO CON SEMILLA 20, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	29
FIGURA 5.5	VARIACIÓN DE $\tau$ PARA EO_PURO CON SEMILLA 15, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES.....	30
FIGURA 5.6	VARIACIÓN DE ITERACIONES PARA EO_PURO 40-20 CON SEMILLA 0 Y $\tau = 1,4$	32
FIGURA 5.7	VARIACIÓN DE SEMILLAS PARA EO_PURO 40-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	33
FIGURA 5.8	VARIACIÓN DE $\tau$ PARA EO_PURO 40-20 CON SEMILLAS 1 Y 11 Y 1.000.000 DE ITERACIONES .....	35
FIGURA 5.9	VARIACIÓN DE $\tau$ PARA EO_PURO 40-20 CON SEMILLA 11, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	35
FIGURA 5.10	VARIACIÓN DE $\tau$ PARA EO_PURO 40-20 CON SEMILLA 1, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES.....	36
FIGURA 5.11	VARIACIÓN DE ITERACIONES PARA EO_PURO 60-20 CON SEMILLA 0 Y $\tau = 1,4$	38
FIGURA 5.12	VARIACIÓN DE SEMILLAS PARA EO_PURO 60-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	39
FIGURA 5.13	VARIACIÓN DE $\tau$ PARA EO_PURO 60-20 CON SEMILLAS 20 Y 14 Y 1.000.000 DE ITERACIONES .....	41
FIGURA 5.14	VARIACIÓN DE $\tau$ PARA EO_PURO 60-20 CON SEMILLA 14, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	41
FIGURA 5.15	VARIACIÓN DE $\tau$ PARA EO_PURO 60-20 CON SEMILLA 20, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES.....	42
FIGURA 5.16	VARIACIÓN DE ITERACIONES PARA EO_PURO 80-20 CON SEMILLA 0 Y $\tau = 1,4$	44
FIGURA 5.17	VARIACIÓN DE SEMILLAS PARA EO_PURO 80-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	45
FIGURA 5.18	VARIACIÓN DE $\tau$ PARA EO_PURO 80-20 CON SEMILLAS 6 Y 18 Y 1.000.000 DE ITERACIONES .....	47
FIGURA 5.19	VARIACIÓN DE $\tau$ PARA EO_PURO 80-20 CON SEMILLA 18, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	47
FIGURA 5.20	VARIACIÓN DE $\tau$ PARA EO_PURO 80-20 CON SEMILLA 6, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES.....	48

---

FIGURA 5.21 VARIACIÓN DE ITERACIONES PARA EO_PURO_BUSQUEDA 20-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	50
FIGURA 5.22 VARIACIÓN DE SEMILLAS PARA EO_PURO_BUSQUEDA 20-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	51
FIGURA 5.23 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 20-20 CON SEMILLA 8, MEJOR MAKESPAN Y TARDINESS, Y 1.000.000 DE ITERACIONES .....	53
FIGURA 5.24 VARIACIÓN DE ITERACIONES PARA EO_PURO_BUSQUEDA 40-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	55
FIGURA 5.25 VARIACIÓN DE SEMILLAS PARA EO_PURO_BUSQUEDA 40-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	56
FIGURA 5.26 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 40-20 CON SEMILLAS 12 Y 15 Y 1.000.000 DE ITERACIONES .....	58
FIGURA 5.27 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 40-20 CON SEMILLA 15, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES .....	58
FIGURA 5.28 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 40-20 CON SEMILLA 12, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES .....	59
FIGURA 5.29 VARIACIÓN DE ITERACIONES PARA EO_PURO_BUSQUEDA 60-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	61
FIGURA 5.30 VARIACIÓN DE SEMILLAS PARA EO_PURO_BUSQUEDA 60-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	62
FIGURA 5.31 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 60-20 CON SEMILLAS 15 Y 20 Y 1.000.000 DE ITERACIONES .....	65
FIGURA 5.32 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 60-20 CON SEMILLA 6, MEJOR TARDINESS, Y 1000000 DE ITERACIONES .....	65
FIGURA 5.33 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 60-20 CON SEMILLA 7, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES .....	66
FIGURA 5.34 VARIACIÓN DE ITERACIONES PARA EO_PURO_BUSQUEDA 80-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	69
FIGURA 5.35 VARIACIÓN DE SEMILLAS PARA EO_PURO_BUSQUEDA 80-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	70
FIGURA 5.36 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 80-20 CON SEMILLAS 10 Y 11 Y 1.000.000 DE ITERACIONES .....	73
FIGURA 5.37 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 80-20 CON SEMILLA 10, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES .....	73
FIGURA 5.38 VARIACIÓN DE $\tau$ PARA EO_PURO_BUSQUEDA 80-20 CON SEMILLA 11, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES .....	74
FIGURA 5.39 VARIACIÓN DE ITERACIONES PARA EO_FEROMONAS 20-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	77
FIGURA 5.40 VARIACIÓN DE SEMILLAS PARA EO_FEROMONAS 20-20 CON 1000000 DE ITERACIONES Y $\tau = 1,4$ .....	78
FIGURA 5.41 VARIACIÓN DE $\tau$ PARA EO_FEROMONAS 20-20 CON SEMILLAS 5 Y 15 Y 1.000.000 DE ITERACIONES .....	80
FIGURA 5.42 VARIACIÓN DE $\tau$ PARA EO_FEROMONAS 20-20 CON SEMILLA 15, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES .....	80

---



---

FIGURA 5.43 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 20-20 CON SEMILLA 5, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES.....	81
FIGURA 5.44 VARIACIÓN DE ITERACIONES PARA EO_ FEROMONAS 40-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	83
FIGURA 5.45 VARIACIÓN DE SEMILLAS PARA EO_ FEROMONAS 40-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	84
FIGURA 5.46 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 40-20 CON SEMILLAS 3 Y 17 Y 1.000.000 DE ITERACIONES.....	86
FIGURA 5.47 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 40-20 CON SEMILLA 17, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	86
FIGURA 5.48 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 40-20 CON SEMILLA 3, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES.....	87
FIGURA 5.49 VARIACIÓN DE ITERACIONES PARA EO_ FEROMONAS 60-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	89
FIGURA 5.50 VARIACIÓN DE SEMILLAS PARA EO_ FEROMONAS 60-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	90
FIGURA 5.51 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 60-20 CON SEMILLAS 2 Y 13 Y 1.000.000 DE ITERACIONES.....	92
FIGURA 5.52 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 60-20 CON SEMILLA 13, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	92
FIGURA 5.53 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 60-20 CON SEMILLA 2, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES.....	93
FIGURA 5.54 VARIACIÓN DE ITERACIONES PARA EO_ FEROMONAS 80-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	95
FIGURA 5.55 VARIACIÓN DE SEMILLAS PARA EO_ FEROMONAS 80-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	96
FIGURA 5.56 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 80-20 CON SEMILLAS 7 Y 6 Y 1.000.000 DE ITERACIONES.....	98
FIGURA 5.57 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 80-20 CON SEMILLA 6, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	98
FIGURA 5.58 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS 80-20 CON SEMILLA 7, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES.....	99
FIGURA 5.59 VARIACIÓN DE ITERACIONES PARA EO_ FEROMONAS_BUSQUEDA 20-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	101
FIGURA 5.60 VARIACIÓN DE SEMILLAS PARA EO_ FEROMONAS_BUSQUEDA 20-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	102
FIGURA 5.61 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 20-20 CON SEMILLAS 8 Y 1 Y 1.000.000 DE ITERACIONES.....	104
FIGURA 5.62 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 20-20 CON SEMILLA 1, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	104
FIGURA 5.63 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 20-20 CON SEMILLA 8, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES .....	105
FIGURA 5.64 VARIACIÓN DE ITERACIONES PARA EO_ FEROMONAS_BUSQUEDA 40-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	107

---

---

FIGURA 5.65 VARIACIÓN DE SEMILLAS PARA EO_ FEROMONAS_BUSQUEDA 40-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	108
FIGURA 5.66 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 40-20 CON SEMILLAS 11 Y 10 Y 1.000.000 DE ITERACIONES .....	110
FIGURA 5.67 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 40-20 CON SEMILLA 10, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	110
FIGURA 5.68 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 40-20 CON SEMILLA 11, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES .....	111
FIGURA 5.69 VARIACIÓN DE ITERACIONES PARA EO_ FEROMONAS_BUSQUEDA 60-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	113
FIGURA 5.70 VARIACIÓN DE SEMILLAS PARA EO_ FEROMONAS_BUSQUEDA 60-20 CON 1.000.000 DE ITERACIONES Y $\tau = 1,4$ .....	114
FIGURA 5.71 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 60-20 CON SEMILLAS 10 Y 16 Y 1.000.000 DE ITERACIONES .....	117
FIGURA 5.72 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 60-20 CON SEMILLA 16, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	117
FIGURA 5.73 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 60-20 CON SEMILLA 10, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES .....	118
FIGURA 5.74 VARIACIÓN DE ITERACIONES PARA EO_ FEROMONAS_BUSQUEDA 80-20 CON SEMILLA 0 Y $\tau = 1,4$ .....	120
FIGURA 5.75 VARIACIÓN DE SEMILLAS PARA EO_ FEROMONAS_BUSQUEDA 80-20 CON 1000000 DE ITERACIONES Y $\tau = 1,4$ .....	121
FIGURA 5.76 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 80-20 CON SEMILLAS 18 Y 12 Y 1.000.000 DE ITERACIONES .....	123
FIGURA 5.77 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 80-20 CON SEMILLA 12, MEJOR TARDINESS, Y 1.000.000 DE ITERACIONES.....	123
FIGURA 5.78 VARIACIÓN DE $\tau$ PARA EO_ FEROMONAS_BUSQUEDA 80-20 CON SEMILLA 18, MEJOR MAKESPAN, Y 1.000.000 DE ITERACIONES .....	124
FIGURA 5.79 GRÁFICO DE COMPARACIÓN MAKESPAN 20-20 CON MEJOR SEMILLA, ITERACIÓN Y T, LAS 4 METODOLOGÍAS UTILIZADAS Y REFERENCIA .....	126
FIGURA 5.80 GRÁFICO DE COMPARACIÓN MAKESPAN 40-20 CON MEJOR SEMILLA, ITERACIÓN Y T, LAS 4 METODOLOGÍAS UTILIZADAS Y REFERENCIA .....	127
FIGURA 5.81 GRÁFICO DE COMPARACIÓN MAKESPAN 60-20 CON MEJOR SEMILLA, ITERACIÓN Y T, LAS 4 METODOLOGÍAS UTILIZADAS Y REFERENCIA .....	128
FIGURA 5.82 GRÁFICO DE COMPARACIÓN MAKESPAN 80-20 CON MEJOR SEMILLA, ITERACIÓN Y T, LAS 4 METODOLOGÍAS UTILIZADAS Y REFERENCIA .....	129
FIGURA 5.83 GRÁFICO DE COMPARACIÓN TARDINESS 20-20 CON MEJOR SEMILLA, ITERACIÓN Y T, LAS 4 METODOLOGÍAS UTILIZADAS Y REFERENCIA .....	130
FIGURA 5.84 GRÁFICO DE COMPARACIÓN TARDINESS 40-20 CON MEJOR SEMILLA, ITERACIÓN Y T, LAS 4 METODOLOGÍAS UTILIZADAS Y REFERENCIA .....	131
FIGURA 5.85 GRÁFICO DE COMPARACIÓN TARDINESS 60-20 CON MEJOR SEMILLA, ITERACIÓN Y T, LAS 4 METODOLOGÍAS UTILIZADAS Y REFERENCIA .....	132
FIGURA 5.86 GRÁFICO DE COMPARACIÓN TARDINESS 80-20 CON MEJOR SEMILLA, ITERACIÓN Y T, LAS 4 METODOLOGÍAS UTILIZADAS Y REFERENCIA .....	133
FIGURA 5.87 ANÁLISIS DE FEROMONA.....	134

---

---

FIGURA 5.88 GRÁFICO DE COMPARACIÓN CON BÚSQUEDA LLAMADA COMBINACIÓN 2 CON Y SIN CATALIZADOR.....	135
FIGURA 5.89 GRÁFICO DE COMPARACIÓN CON BÚSQUEDA LLAMADA BÚSQUEDA_L CON Y SIN CATALIZADOR.....	136

---

## ÍNDICE DE TABLAS.

TABLA 5.1	RESULTADOS PARA SEMILLA 15 Y 20 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS EN EO_PURO 20-20.....	28
TABLA 5.2	MEJORES T PARA SEMILLAS 15 Y 20 EN EO_PURO 20-20 .....	31
TABLA 5.3	RESULTADOS PARA SEMILLA 1 Y 11 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_PURO 40-20.....	34
TABLA 5.4	MEJORES T PARA SEMILLAS 1 Y 11 EN EO_PURO 40-20 .....	37
TABLA 5.5	RESULTADOS PARA SEMILLA 20 Y 14 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_PURO 60-20.....	40
TABLA 5.6	MEJORES T PARA SEMILLAS 14 Y 20 EN EO_PURO 60-20. ....	43
TABLA 5.7	RESULTADOS PARA SEMILLA 6 Y 18 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_PURO 80-20.....	46
TABLA 5.8	MEJORES T PARA SEMILLAS 6 Y 18 EN EO_PURO 80-20 .....	49
TABLA 5.9	RESULTADOS PARA SEMILLA 8 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_PURO_BUSQUEDA 20-20.....	52
TABLA 5.10	MEJOR T PARA SEMILLA 8 EN EO_PURO_BUSQUEDA 20-20 .....	54
TABLA 5.11	RESULTADOS PARA SEMILLA 12 Y 15 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_PURO_BUSQUEDA 40-20 .....	57
TABLA 5.12	MEJORES T PARA SEMILLAS 12 Y 15 EN EO_PURO_BUSQUEDA 40-20.....	60
TABLA 5.13	RESULTADOS PARA SEMILLA 7 Y 6 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_PURO_BUSQUEDA 60-20.....	64
TABLA 5.14	MEJORES T PARA SEMILLAS 6 Y 7 EN EO_PURO_BUSQUEDA 60-20.....	68
TABLA 5.15	RESULTADOS PARA SEMILLAS 11 Y 10 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_PURO_BUSQUEDA 80-20 .....	72
TABLA 5.16	MEJORES T PARA SEMILLAS 10 Y 11 EN EO_PURO_BUSQUEDA 80-20.....	76
TABLA 5.17	RESULTADOS PARA SEMILLA 5 Y 15 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_FEROMONAS 20-20.....	79
TABLA 5.18	MEJORES T PARA SEMILLAS 5 Y 15 EN EO_FEROMONAS 20-20 .....	82
TABLA 5.19	RESULTADOS PARA SEMILLA 3 Y 17 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_FEROMONAS 40-20.....	85
TABLA 5.20	MEJORES T PARA SEMILLAS 3 Y 17 EN EO_FEROMONAS 40-20 .....	88
TABLA 5.21	RESULTADOS PARA SEMILLAS 2 Y 13 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_FEROMONAS 20-20.....	91
TABLA 5.22	MEJORES T PARA SEMILLAS 2 Y 13 EN EO_FEROMONAS 60-20 .....	94
TABLA 5.23	RESULTADOS PARA SEMILLA 7 Y 6 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_FEROMONAS 80-20 .....	97
TABLA 5.24	MEJORES T PARA SEMILLAS 6 Y 7 EN EO_FEROMONAS 80-20 .....	100
TABLA 5.25	RESULTADOS PARA SEMILLA 8 Y 1 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_FEROMONAS_BUSQUEDA 20-20.....	103
TABLA 5.26	MEJORES T PARA SEMILLAS 1 Y 8 EN EO_FEROMONAS_BUSQUEDA 20-20.....	106
TABLA 5.27	RESULTADOS PARA SEMILLA 11 Y 10 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_FEROMONAS_BUSQUEDA 20-20 .....	109
TABLA 5.28	MEJORES T PARA SEMILLAS 10 Y 11 EN EO_FEROMONAS_BUSQUEDA 40-20.	112

---

TABLA 5.29	RESULTADOS PARA SEMILLA 10 Y 16 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_FEROMONAS_BUSQUEDA 60-20 .....	116
TABLA 5.30	MEJORES T PARA SEMILLAS 10 Y 16 EN EO_FEROMONAS_BUSQUEDA 60-20.	119
TABLA 5.31	RESULTADOS PARA SEMILLA 18 Y 12 EN LAS PRUEBAS DE VARIACIÓN DE SEMILLAS CON EO_FEROMONAS_BUSQUEDA 80-20 .....	122
TABLA 5.32	MEJORES T PARA SEMILLAS 12 Y 18 EN EO_FEROMONAS_BUSQUEDA 80-20.	125

---

# 1 Introducción

## 1.1 Descripción

El problema de secuenciación de trabajos (de ahora en adelante llamados Job) en sistemas de producción, *Flow-Shop*, ha sido un tema importante en la investigación de operaciones, del cual se desea establecer la programación óptima de ellos en máquinas, dentro de un proceso de producción en general. Para esto se han empleado variadas técnicas y estrategias de optimización: Algoritmo de Chu-Beasley (Jiménez Morales, 2012), Algoritmos Evolutivos (Rodríguez Quiñones, 2014), Algoritmos Genéticos Modificados (Toro Ocampo, Restrepo Grisales, & Granada Echeverr, 2006), entre muchas otras (Peña & Zunelzu, 2006); con las que se pretende determinar el orden de procesamiento de los Job de forma óptima y que apoyen la toma de decisiones a nivel operativo.

Los sistemas *Flow-Shop-Scheduling* son un modelo de planificación de tareas, en el cual una cantidad  $n$  de Job ( $J_0, J_1, J_2, \dots, J_n$ ) son procesados por un conjunto  $m$  de máquinas ( $M_0, M_1, M_2, \dots, M_m$ ). Éstas siguen características básicas como:

- Cada máquina realiza una tarea específica y para un Job a la vez.
- El orden de las máquinas es fijo.
- Los Job pasan por cada máquina una única vez.
- Por último, los Job requieren un proceso completo para ser terminados, es decir, que todas las máquinas ejerzan su tarea correspondiente sobre el Job, desde el primero hasta el último.

El objetivo es ordenar la secuencia de ingreso de los Job, cumpliendo con los tiempos requeridos dadas las restricciones como: minimizar el tiempo para completar todos los Job y terminar la serie con la diferencia mínima de tiempo con respecto a la fecha de entrega. Considerando lo anterior, el problema de optimización de tiempos presenta una gran complejidad. Esto queda reflejado aún más cuando este procesamiento se realiza para una cantidad  $n$ , con  $n \geq 2$  de Job sobre una cantidad  $m$ , con  $m \geq 2$  de máquinas de tal forma que se optimicen las medidas de efectividad anteriores.

Algunas de las características del problema son:

- La secuencia de los Job es la misma para todo el proceso.
- Un Job no podrá ingresar a una máquina si ésta está siendo utilizada.

- 
- Los Job no pueden ser interrumpidos.
  - Se debe respetar la secuencia de los Job, es decir, una vez definido el orden de entrada a la secuencia de máquinas este orden no variará a lo largo del proceso.

Resolver problemas complejos (tanto problemas de optimización como de búsqueda) ha sido una de las áreas importantes de investigación en el campo de las ciencias de la computación. El objetivo perseguido en esta área ha sido el desarrollo de nuevos métodos capaces de resolver estos problemas con el menor esfuerzo computacional posible, mejorando así los algoritmos existentes. En consecuencia, esto no sólo permite afrontar los problemas de forma más eficiente, sino afrontar tareas vedadas en el pasado debido a su alto costo computacional.

La investigación está enfocada en aplicar una reciente metaheurística conocida como *Extremal Optimization* (EO) basada en el modelo de ecosistemas de Bak-Sneppen de auto-organización crítica, para ello se utilizará una evaluación de aptitud (*fitness*) inseparable (La complejidad de EO radica en encontrar un *fitness* apropiado para cada elemento de la solución, a diferencia de otros algoritmos genéticos, que la buscan para la solución completa, por ello a esta función de aptitud se le denomina inseparable, pues es exclusiva para cada especie involucrada) la cual será aplicada al problema multi-objetivo de planificación permutativa de producción en línea (MOPFSSP del inglés *Multiobjective Permutation Flow-shop Scheduling Problem*).

EO ya ha sido utilizado para resolver este problema, sin embargo, los resultados no han sido los esperados, por ello, el nuevo método que se propone en esta investigación es crear un algoritmo híbrido, es decir, a EO se le añadirá el concepto de feromona extraído desde Optimización Colonia de Hormigas (ACO) para así, con esta hibridación, lograr un método más generalizado en la construcción del *fitness* para este algoritmo. Para ello será necesario estudiar los conceptos de EO y ACO, como también profundizar en lo referente a multi-objetivo y el problema de *Flow-Shop Scheduling*.

## 1.2 Justificación

La principal motivación para el desarrollo de este proyecto de investigación es encontrar un método de evaluación de *Fitness* para EO cuando el problema a ser resuelto no entrega información suficiente y/o necesaria para su formulación.

De tener éxito con este método de evaluación EO podría convertirse en un mecanismo híbrido capaz de resolver no sólo el tipo de problemas *Flow-Shop Scheduling*, sino también podría ser aplicado a problemas de optimización combinatoria complejos.

---

### 1.3 Delimitación

La información con la que se cuenta de EO es relativamente escasa, menos de 20 años (Boettcher & Percus, 1999), por lo que la investigación tratará de ayudar en su argumentación, para ello se contará con *benchmarks* públicos, en particular los desarrollados por Ishibushi (Ishibushi, Yoshida, & Murata, 2003) que contiene pruebas referentes al problema en sí. También, de ser necesario, se procederá a simular buscando muestras representativas para realizar *test* lo más generalizado posible.

La cantidad de *tests* realizados serán 20, pero al tratarse de un problema multi-objetivo (dos objetivos a optimizar para ser más específicos) estas pruebas se realizarán variando el valor de semilla de la función aleatoria y el valor de tau característico en el algoritmo EO, siempre tratando de mantener el margen de realismo y comparación, además se evaluará, entre estos mismos *test*, el comportamiento del algoritmo. Como se dijo anteriormente, todo esto estará sujeto a la cantidad de información encontrada.

El lenguaje de programación escogido será C, debido a su menor tiempo de ejecución comparado con lenguajes de nivel superior y a su flexibilidad.

### 1.4 Objetivo General

- Aplicar EO combinado con ACO para la resolución del problema multi-objetivo *Permutation Flow-Shop Scheduling*.

### 1.5 Objetivos Específicos

- Estudiar los conceptos algorítmicos de *Extremal Optimization* y Colonia de Hormigas, así como los problemas de *Flow-Shop Scheduling* (FSS) y multi-objetivo.
- Modelar el problema de MOPFSSP.
- Implementar el algoritmo híbrido.
- Aplicar el algoritmo creado al problema.
- Analizar resultados y hacer comparaciones.



---

## 1.6 Metodología

- Para el estudio de los diferentes conceptos se procederá a través de búsquedas en *Google Académico*, artículos referentes a la materia en cuestión, así como investigaciones públicas que se han hecho al respecto y estudio de libros que se encuentren en la biblioteca de la UCSC.
- El modelado será aplicando programación matemática al problema, de la forma método simplex, con sus respectivas funciones objetivo y restricciones. Lo anterior ayudará a guiar de mejor forma el modelo algorítmico, pues se sabrá si es posible representarlo a través de estructuras, vectores, matrices entre otros.
- La creación del algoritmo híbrido y su posterior implementación se hará utilizando la metodología por prototipos, pues no se conoce a cabalidad las restricciones que pudiese presentar (tiempos de ejecución, restricciones de hardware y/o software, entre otros) y así asegurar la finalización de la investigación. Se crearán prototipos que ayuden y guíen a un camino más estructurado, reconociendo los requisitos reales al momento de ejecutar las pruebas. La cantidad de prototipos serán 3 comenzando con el que podría ser completamente descartable, sin embargo, ayudará a comprender el funcionamiento del algoritmo EO. Continuando con aquél que entregará la mayor cantidad de retroalimentación, pues aquí se implementará EO, del cual se entendió el funcionamiento en el prototipo anterior, directamente a PFSSP. Finalmente, gracias a la información recibida del prototipo anterior se creará uno completamente funcional en pro del problema establecido en esta investigación.
- La codificación se hará en lenguaje C ya que es eficiente en el código, además dispone de las estructuras típicas de los lenguajes de alto nivel, pero, a su vez, dispone de construcciones del lenguaje que permiten un control a muy bajo nivel, por otro lado, es un lenguaje muy flexible que permite programar con múltiples estilos e impide operaciones sin sentido.

- 
- Una vez que se ha llegado al prototipo final se procederá con las pruebas al problema en concreto, es decir, aplicado a MOPFSSP. Se realizarán 20 pruebas tabulando sus resultados, así como también se anotarán los tiempos de ejecución para futuras comparaciones y mediciones.
  - Los análisis se efectuarán utilizando herramientas estadísticas y gráficos, en particular la media y desviación estándar, estos valores darán una idea general del comportamiento del algoritmo creado y su real efectividad comparándolo con resultados desarrollados por Ishibushi (Ishibushi, Yoshida, & Murata, 2003).
  - La conclusión se hará en base a los análisis y comparaciones efectuadas y los resultados logrados.

---

## 2 Marco Teórico

### 2.1 Extremal Optimization (EO)

*Extremal Optimization* (Boettcher & Percus, 1999) es una técnica de búsqueda estocástica que tiene la propiedad de ser tanto local como global. Esta técnica está inspirada en el modelo de auto-organización crítica de co-evolución propuesto en el campo de física estadística por Bak-Sneppen (Bak & Sneppen, 1993). Este modelo sugiere que algunos sistemas dinámicos tienen un punto crítico como atractor, donde el sistema exhibe periodos de movimientos lentos o acumulativos seguidos de cortos periodos de avalanchas o inestabilidad. Ejemplos de estos sistemas son la formación de islas, terremotos y montañas de arenas dinámicas.

El dinamismo del sistema causa la mejora estable de la solución candidata a través de grandes y repentinos choques en la calidad de la misma los cuales permiten explotar soluciones de calidad más altas en una búsqueda local y evitar posibles óptimos locales. La estrategia de este proceso de búsqueda es identificar, iterativamente, la peor componente de una solución dada y sustituirlo o, como se hará en esta investigación, cambiarlo de lugar con otro componente. Esto es alcanzado a través de la asignación de coste (*Fitness*) dadas a las componentes de la solución. Una vez que los componentes son evaluados ellos son ordenados y el componente más débil es sustituido o cambiado por un componente seleccionado al azar o considerando alguna restricción propia del problema.

La gran complejidad está en visualizar de forma acertada esta asignación de costes y evaluar correctamente al candidato a sustituir, para ello, se debe conocer a la perfección las restricciones y limitaciones del problema, así EO tendrá una mayor precisión en la búsqueda de óptimos. Esta investigación intenta dar un indicio para esta función de coste y para evaluar en primera instancia a el o los candidatos a ser permutados utilizando el concepto de feromonas.

El algoritmo de la Figura 2.1 proporciona un pseudo-código para la versión canónica de EO. En la versión original de EO la selección, en el caso de minimizar una función de coste, del peor componente en la función **SelecciónComponenteDebil** y el reemplazo de la función de **SelecciónComponenteReemplazo** era determinista lo que provocaba que EO quedase atrapado en algún óptimo local, esta situación se mejoró haciendo uso del parámetro probabilístico  $\tau$ , entonces se menciona como  $\tau$ -Extremal Optimization.

```

Entrada: TamañoProblema, iteracionesmax , T
Salida: Smejor
Snormal ← CrearSolucionInicial(TamañoProblema)
Smejor ← Snormal
For (i = 1 to iteracionesmax)
  For (Componentei ∈ Snormal)
    Componenteicost ← Coste(Componentei , Snormal)
  End
  ComponentesOrdenadas ← Rank(Sicomponentes)
  Componentei ← SeleccionComponenteDebil(ComponentesOrdenadas, Componentei, T)
  Componentej ← SeleccionComponenteReemplazo(ComponentesOrdenadas, T)
  Scandidato ← Reemplazar(Snormal, Componentei , Componentej)
  If (Coste(Scandidato) ≤ Coste(Smejor))
    Smejor ← Scandidato
  End
End
Return (Smejor)

```

Figura 2.1 Seudo-código EO de mínimo coste ( $T = \tau$ )

- Extremal Optimization fue creada para problemas de optimización combinatorios, sin embargo, se han desarrollado variaciones que han sido aplicadas a la optimización de funciones continuas (Duch & Arenas, 2008).
- La selección de la peor componente y su respectivo reemplazo en cada iteración puede ser determinístico o probabilístico, la última versión referencia a  $\tau$ -Extremal Optimization pues se utiliza el parámetro  $\tau$ .
- Las funciones de evaluación y selección de las componentes de una solución apropiada son las consideraciones más complejas en el uso de esta técnica.
- Para  $\tau$ -Extremal Optimization, los valores  $\tau$  más bajos usados son  $\tau \in [1,2; 1,6]$ .
- Para esta investigación las pruebas se harán para  $\tau \in [1,0; 2,5]$  incrementando en 0,1.

A la versión anterior se le denominará, en esta investigación, como EO en estado Puro. Además, se utilizará un EO con búsqueda local, el cual consiste en añadir una función de búsqueda a EO.

---

## 2.2 Permutation Flow-Shop Scheduling Problem (PFSSP)

El *Scheduling* es una tarea con gran necesidad de cálculo y estos problemas son de optimización combinatoria (Saghafian & Hejazi, 2005).

La función de *Scheduling* se fundamenta en asignar recursos limitados a trabajos (de ahora en adelante se denominarán *Job*) a lo largo del tiempo y su finalidad es la optimización de uno o más objetivos.

En los sistemas *Flow-Shop* hay  $m$  máquinas en serie, cada *Job* se procesa en cada una de ellas. Todos los *Job* tienen la misma trayectoria, es decir, primero se procesan sobre la máquina 1, luego sobre la máquina 2, y así sucesivamente (Un modelamiento matemático general fue descrito por Sadjadi (Sadjadi, Aryanezhad, & Ziaee, 2008)). Cuando un *Job* termina su tarea en una máquina se agrega a la cola de la próxima máquina. Usualmente, se asume que todas las colas trabajan bajo la disciplina primero en entrar primero en salir (FIFO). Si se aplica la disciplina FIFO, al *Flow-Shop* se lo denomina como *Flow-Shop* con permutación (FSSP). Por todo lo anterior se puede decir que *Permutation Flow-shop Scheduling* es un modelo que consiste en programar un conjunto de  $n$  *Job*, en un ambiente de línea de flujo de  $m$  máquinas no relacionadas que cumplen con características básicas como que, cada máquina realiza una tarea específica y para un *Job* a la vez, los *Job* requieren finalizar un proceso (desde la máquina 1 hasta la  $m$ ) para ser completados, los *Job* pasan por cada máquina una única vez, cada *Job* tiene variados tiempos en concretar sus tareas en cada máquina así como un tiempo de entrega del *Job*, es decir, necesita haber terminado todo su proceso, convenientemente, antes de una fecha indicada y el orden de las máquinas es el mismo para todos los *Job*.

El objetivo es ordenar la secuencia de ingreso de los *Job*, cumpliendo con los tiempos asignados dadas las restricciones, como minimizar el tiempo requerido para completar todos los *Job* y/o terminar la serie con un margen mínimo de tiempo de diferencia a la fecha de entrega; considerando lo anterior, el problema de optimización de tiempos presenta una gran complejidad.

Matemáticamente el problema consiste en encontrar una permutación de los *Job* que resulte en un tiempo mínimo de procesamiento de todos ellos. Este problema está clasificado técnicamente como *np-Hard* (Cook, 1971), es decir, se ha demostrado que cualquier algoritmo de solución tiene un tiempo de ejecución que aumenta, en el peor de los casos, exponencialmente con el tamaño del problema.

Para medir la eficiencia en cualquier tipo de problema de *Flow-Shop*, se busca la optimización de medidas de desempeño o métricas de efectividad, esta investigación se centrará particularmente en dos que se definen a continuación y conforme al lenguaje utilizado en esta investigación:

---

**Makespan:** tiempo total requerido para terminar el proceso de un Job.

**Tardiness:** diferencia positiva entre el tiempo de término de un Job y su fecha de entrega.

El fin es minimizar ambos objetivos.

## 2.3 Optimización Multi-Objetivo (MOP)

La optimización multi-objetivo, es un problema de optimización definido para dos o más funciones objetivo. El principal inconveniente reside en la subjetividad de las soluciones encontradas, es decir, la solución dependerá de qué factores se considerarán más importantes para elegir, de entre un conjunto de soluciones, las “óptimas”, esto conlleva a especificar claramente estos factores, los cuales pueden variar dependiendo de las prioridades del individuo que las realice. Un problema multi-objetivo genera una gama de soluciones que no pueden ser negadas entre los múltiples objetivos a los cuales va dirigido. Por consiguiente, este conjunto de soluciones óptimas es denominado Frente de Pareto (FP). Esta frontera de soluciones contiene cada uno de los puntos que no son superados en todos los objetivos por otra solución. Este concepto se denomina dominancia, por ello el FP consiste sólo de soluciones no dominadas. Una solución domina a otra sí y sólo sí, es al menos tan buena como la otra en todos sus objetivos y es mejor en al menos uno de ellos.

En la optimización multi-objetivo se debe acordar un cierto esquema de mejoría de una solución sobre otra, es decir, cuáles se elegirán para ser más aptas; a esta relación de mejoría de un individuo sobre otro se le conoce como esquema de dominación y se basa en que la solución de un problema multi-objetivo no es única y el responsable debe elegir subjetivamente desde una gran variedad de posibles soluciones que no se pueden mejorar entre sí.

Así, la respuesta al problema de hallar las soluciones no dominadas en un problema multi-objetivo es a lo que se le llama el conjunto solución del problema y el conjunto de valores de la función objetivo con dominio restringido a los vectores del conjunto solución (los vectores no dominados) es lo que conocemos como Frente de Pareto.

En resumen, el conjunto de óptimos de Pareto es el espacio solución del problema, y el Frente de Pareto es su imagen con respecto a la función a optimizar.

Un concepto relacionado con el Frente de Pareto es el Óptimo de Pareto. Tanto el Óptimo como el Frente de Pareto son el marco sobre el cual trabaja la toma de decisiones multicriterio.

**Definición:** Dado el problema multi-objetivo, minimizar  $f(x)$ , donde

---

$f: F \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^q$ ,  $q \geq 2$  y  $A \subseteq F$  la región factible. Decimos que un vector  $\mathbf{x}^* \in A$  es no dominado o un óptimo Pareto si no existe un vector  $\mathbf{x} \in A$  tal que  $\mathbf{x} < \mathbf{x}^*$ .

**Definición:** El conjunto  $E(A; f)$  de soluciones de Pareto eficientes (también conocido como conjunto de óptimos de Pareto) se define de la manera siguiente:

$$E(A, f) = \{ \mathbf{a} \in A : \nexists \mathbf{b} \in A \text{ que cumpla } f(\mathbf{b}) < f(\mathbf{a}) \}$$

Es decir, el conjunto de todos los vectores no dominados bajo el esquema de Pareto.

A continuación, se grafica el Frente de Pareto representado por la línea roja. Los cuadrados representan una posible solución, mientras que los cuadrados remarcados son las soluciones dentro del Frente de Pareto. F2 corresponde al eje de las abscisas, mientras que F1 a las ordenadas. Cada solución está compuesta por un valor de F1 y F2, entonces, en la solución A (F1, F2) el valor de F1 es mayor que la solución en B (F1, F2) y la solución de F2 en A es menor que la solución de F2 en B, es decir, si el objetivo fuese minimizar, la solución A tendría un óptimo para F2, pero no para F1, mientras que la solución B tiene un óptimo para F1, mas no para F2.

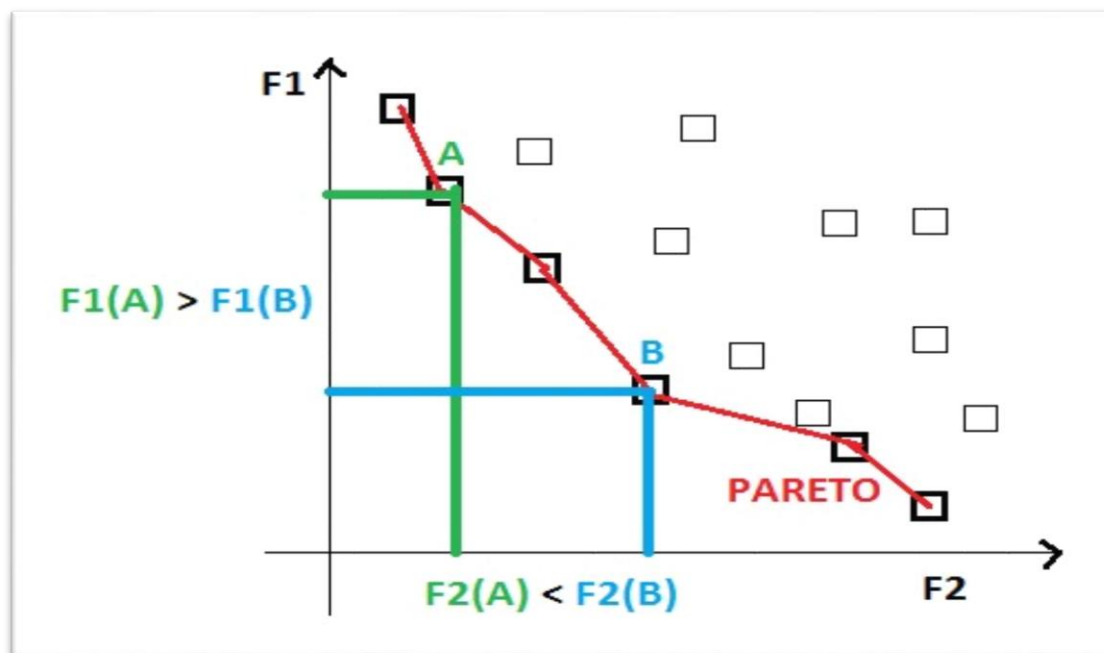


Figura 2.2 Ejemplo de Dominancia para 2 objetivos (Elaboración propia)

## 2.4 Algoritmo Colonia de Hormigas (ACO)

La metaheurística ACO (Duarte Muñoz, Pantrigo Fernández, & Gallego Carrillo, 2007) está inspirada en la observación del comportamiento de hormigas reales.

---

Las hormigas son insectos sociales que viven en colonias y que, debido a la colaboración, son capaces de mostrar comportamientos complejos y realizar tareas difíciles desde el punto de vista de una hormiga individual (Li, Peng, Kurths, Yang, & Schellnhuber, 2014). Un aspecto interesante es su habilidad para encontrar los caminos más cortos entre su nido y las fuentes de alimento. Este hecho es interesante si se tiene en cuenta que muchas de las especies de hormigas son casi ciegas, lo que evita el uso de rutas visuales.

Las hormigas, al momento de recorrer el nido hacia una fuente de comida y volver al nido, depositan una sustancia llamada feromona. Otras hormigas logran olfatear esta sustancia, y seguir el mismo recorrido depositando más feromona, siempre y cuando este recorrido sea menor al que están recorriendo, así las concentraciones de feromona aumentan en la vía más corta como también la cantidad de hormigas que la recorren. Esto les permite encontrar buenas fuentes de comida que han sido previamente identificadas por otras hormigas.

Las hormigas comienzan explorando los alrededores hasta alcanzar alguna fuente de comida. Depositán la feromona sobre el camino que siguieron entre el nido y la fuente. Inicialmente, cada hormiga elige aleatoriamente los caminos, sin embargo, conforme ocurren cambios aleatorios, al cabo de algún tiempo la feromona depositada en uno de los caminos será mayor. Puesto que las hormigas tienen la tendencia (en probabilidad) a seguir un rastro de feromona más fuerte, el camino con mayor cantidad de feromona atraerá más hormigas. Esto hace que el rastro de feromona aumente, hasta que, finalmente, la colonia de hormigas converge hacia el uso de un mismo camino.

El comportamiento anteriormente descrito está basado en la auto-catálisis (retroalimentación positiva) y es sabiamente utilizado por las hormigas para encontrar el camino más corto entre el nido y su fuente de comida.

En resumen, las hormigas convergen rápidamente al uso del camino más corto gracias a la capacidad mecánica de seguir y depositar feromona.

Esta naturalidad de las hormigas puede ser usada como inspiración para la creación de hormigas artificiales que resuelvan problemas de optimización. Ellas podrían simular la acción de depositar feromona cambiando apropiadamente las variables asociadas con estados del problema que visitan, a medida que construyen soluciones cada vez mejores. Destacar que utilizan el depósito de feromonas para recordar su comportamiento, es decir, acumular el conocimiento que adquieren del problema a resolver. También, las hormigas artificiales, sólo tendrían acceso en forma local a estas variables de feromona. Otro punto que tomar en cuenta es la relación entre el mecanismo auto-catalítico y el hecho de que caminos más cortos (los cuales corresponden a soluciones de menor costo) son completados antes que los más largos, y, por lo tanto, se deposita feromona con mayor constancia, por ende, mientras más corto es el camino, más pronto se deposita la feromona, así no acaba por evaporarse, y más hormigas la captan y utilizan esta vía.



---

Si todo lo anterior es utilizado apropiadamente, puede transformarse en un mecanismo poderoso en algoritmos de optimización basados en población.

---

## 3 Estado del Arte

Se presentarán tres investigaciones (todos con sus títulos originales) que se han considerado influyentes para el estudio de esta investigación. En particular se abordarán los temas del investigador Hisao Ishibushi (Ishibushi, Yoshida, & Murata, 2003) desde el cual se han considerado sus pruebas para la comparación, luego se presentará el tema de los investigadores Eduardo Salazar Hornig y Natalia Pavón Weber (Salazar Hornig & Pavón Weber, 2011) que corresponde a un estudio realizado en Chile, para tener una referencia local y finalizando con el tema investigado por el profesor Pedro Gómez Meneses (Gómez Meneses P. S., 2012) pues gran parte de esta investigación se basa en sus estudios.

### 3.1 Balance between Genetic Search and Local Search in Memetic Algorithms for Multiobjective Permutation Flow-Shop Scheduling.

La investigación desarrollada por el profesor Hisao Ishibushi (Ishibushi, Yoshida, & Murata, 2003) muestra cómo el funcionamiento de algoritmos de optimización evolutiva multi-objetivo (EMO) puede ser mejorado por la hibridación con la búsqueda local. El principal efecto positivo de la hibridación es la mejora en la velocidad de convergencia al frente de Pareto. Por otro lado, el principal efecto negativo es el aumento en el tiempo de cómputo por cada generación. Así el número de generaciones disminuye conforme el tiempo de cómputo disponible se limita. Por consiguiente, la capacidad global de búsqueda de este algoritmo no es utilizada por completo. Estos efectos, positivo y negativo, son examinados por experimentos computacionales sobre los problemas de planificación con permutación multi-objetivo Flow-Shop. Los resultados de dichos experimentos muestran claramente la importancia de encontrar un justo equilibrio entre búsqueda genética y búsqueda local. En esta investigación se modificó el antiguo algoritmo de búsqueda genética multi-objetivo local (MOGLS) (Schaffer, 1985) escogiendo sólo a buenos individuos como soluciones iniciales para la búsqueda local y asignando una dirección de búsqueda local apropiada a cada solución inicial. Luego se demostró la importancia de encontrar un justo equilibrio entre la búsqueda genética y la local a través de experimentos computacionales. Entonces se comparó MOGLS modificado con algoritmos EMO desarrollados recientemente: SPEA (Zitzler & Thiele, 1998) y NSGA-II (Srinivas & Deb, 1995). Finalmente, se manifestó que la búsqueda local puede ser fácilmente combinada con aquellos algoritmos EMO para diseñar algoritmos meméticos multi-objetivo.

En conclusión, se demostró, a través de experimentos computacionales, que el funcionamiento de SPEA y el NSGA-II fue considerablemente mejor para algunos

---

problemas de prueba gracias a la hibridación con la búsqueda local. También se demostró que la hibridación redujo considerablemente el tiempo de cómputo en los algoritmos EMO para muchos problemas de prueba.

Para ajustar el equilibrio entre la búsqueda genética y local en experimentos computacionales sobre los problemas de planificación con permutación multi-objetivo *flow-shop* se usaron tres parámetros que pueden disminuir el número de soluciones examinadas por la búsqueda local. Los valores de estos parámetros eran constantes durante la ejecución de estos experimentos. El control dinámico de estos parámetros es un asunto de futuras investigaciones.

Por último y no por ello menos importante es destacar que el algoritmo de búsqueda local utilizado fue el mecanismo de *Simple Hill Climbing* para esta investigación.

### **3.2 Aplicación de un algoritmo ACO al PFSSP con tiempos de preparación dependientes de la secuencia y minimización de Makespan.**

En este trabajo, desarrollado por Eduardo Salazar Hornig y Natalia Pavón Weber (Salazar Hornig & Pavón Weber, 2011), se estudió el problema de secuenciamiento de trabajos en el taller de flujos de permutación con tiempos de preparación dependientes de la secuencia y minimización de Makespan. Para ello se llevó el problema original a una estructura semejante al problema del vendedor viajero *Traveling Salesman Problem* (TSP) (Merz & Freisleben, 1997) asimétrico, proponiendo un algoritmo de optimización mediante colonia de hormigas y utilizando problemas propuestos en la literatura para su evaluación los que luego son comparados con una adaptación de la heurística *Nawaz-Enscore-Ham* (NEH) (Nawaz, Enscore Jr., & Ham, 1983). Finalmente, a la solución encontrada tanto por ACO como NEH se le aplica una búsqueda en vecindad.

En esta investigación se analizó el comportamiento de un algoritmo ACS de optimización de colonia de hormigas aplicado al problema antes mencionado con minimización de Makespan ( $C_{max}$ ). Se infirió que, con un nivel de significancia  $\alpha = 0,05$  obtenido luego del análisis de varianza realizado para la variable %MSC, ninguno de los factores considerados tenía una incidencia significativa en el desempeño del algoritmo ACS. Por tanto, el algoritmo ACS muestra estabilidad en su rendimiento para los valores de los parámetros  $\rho = [0,4 - 0,5]$  y  $\beta = [3 - 4 - 5]$ , por lo que se recomienda su uso.

Se utilizó una extensión de la heurística NEH para la comparación en el problema. Además, se implementó una búsqueda en vecindad IP para mejorar la

---

solución entregada por ambos algoritmos (ACS y NEH). Para conceptos de evaluación se utilizó la extensión del *benchmark* de Taillard utilizado por Ruiz, Maroto y Alcaraz.

El comportamiento de los algoritmos es inverso con respecto al porcentaje máximo de tiempo de preparación y tiempo de proceso el uno del otro, sin embargo, este comportamiento no se mantiene cuando se utiliza en ellos la búsqueda en vecindad IP. Además, ACS+BV muestra mejor desempeño promedio que NEH+BV en problemas con mayor porcentaje máximo de tiempo de preparación respecto del tiempo de proceso.

En particular al aplicar ACS+BV el promedio de mejora es de un 2,38%, con resultados entre 0,90% y 7,65% promediando 4,48%. Por tanto, no se puede afirmar que sea competitivo para todo tipo de problemas de taller de flujo con permutación.

Se puede afirmar que ACS+BV mejora las soluciones con bajo costo computacional pues los tiempos de ejecución son del orden de un minuto por réplica para las instancias de mayor tamaño y esto no influye en el orden de magnitud del tiempo de ejecución.

NEH+BV es mejor en desempeño en 1,04%. El bajo tiempo de ejecución de NEH+BV (cerca de 0,5 segundos en problemas de tamaño mayor) compensa el bajo rendimiento del aumento de porcentaje en el tiempo máximo de preparación respecto del proceso comparado con ACS+BV.

Considerar de forma explícita la estructura del sistema productivo en estudio puede dar mejoras al rendimiento de ACS+BV, pues se hizo una simplificación redefiniendo la matriz costo/distancia para dar estructura TSP al problema original, es decir, no considerar a cabalidad el tipo de configuración productiva tratada.

En resumen, se debe analizar concienzudamente el problema a tratar, teniendo en cuenta los métodos escogidos y considerando su desempeño en términos de calidad de la solución y tiempo computacional requerido, ya que, para problemas como los presentados anteriormente las buenas soluciones en un tiempo breve son primordiales. En este sentido la heurística NEH puede ser de gran utilidad pues es fácil de comprender e implementar.

### **3.3 EO aplicado a Multi-Objective Permutation Flow-Shop Scheduling Problem**

En esta investigación, desarrollada por Pedro Gómez Meneses (Gómez Meneses P. S., 2012), cada problema fue controlado diez veces. Para cada ejecución la semilla, arbitraria, fue cambiada por un nuevo valor. El número de iteraciones, para completar un proceso de EO multi-objetivo híbrido era 100.000. Las pruebas realizadas para cada

---

problema con un número grande de iteraciones no arrojaron mejoras significativas en los resultados. El parámetro de  $\tau$  fue probado dentro de la gama de valores entre 1.1 y 2.8 en los incrementos de 0.1. Para valores fuera de esta gama de  $\tau$  los resultados no eran competitivos. Por consiguiente,  $\tau$  fue puesta en 1.4 para todos los problemas de este caso, que dieron los mejores resultados totales. El mismo  $\tau$  ha sido probada en investigaciones anteriores demostrando ser un valor bueno para obtener soluciones eficientes (Gómez Meneses P. S., 2012).

Los tres conceptos básicos que considerar en esta investigación fueron:

- 1- Reducir al mínimo la convergencia, que es la diferencia entre el Frente de Pareto obtenido con el real (asumiendo que éste es conocido).
- 2- Maximizar la diversidad, que es la distribución de soluciones de modo que el Frente de Pareto sea lo más uniforme posible.
- 3- Maximizar la cobertura, es decir, que las soluciones del Frente de Pareto abarquen el mayor espacio de búsqueda posible.

En los resultados de la métrica  $S$  medidos en esta investigación se aprecia que la diferencia en el tamaño del espacio dominado por el acercamiento de EO multi-objetivo híbrido y los valores reales del Pareto Optimal Set (POS) es mínimo en favor de POS. En este caso, la convergencia, la diversidad y rasgos de cobertura trabajaron bastante bien. Una situación similar puede ser observada para los datos de la métrica  $C$  que fueron analizados. Para ambos casos la diferencia era menor al 7%.

En esta investigación se aplicó EO híbrido a un conjunto de casos para *Permutation Flow-shop Scheduling* multi-objetivo. Este conjunto de pruebas es propuesto por Ishibushi, Yoshida y Murata (Ishibushi, Yoshida, & Murata, 2003) y consiste en cuatro casos bi-objetivo.

Los casos son definidos con  $n$ -Job y  $m$ -máquinas y descritos de la siguiente forma: el tiempo que tarda cada Jobs en cada máquina fue especificado como un número entero arbitrario en el intervalo de (1, 99). El vencimiento de cada trabajo fue especificado añadiendo un número entero arbitrario en el intervalo  $(-100, 100)$  a su tiempo real de término en una lista generada al azar. Todos los problemas de prueba tienen 20 máquinas ( $m = 20$ ). El número  $n$  de jobs puede tomar los siguientes valores:

$n = 20, 40, 60, 80$ . El número de objetivos  $k$  es 2 (Tardiness y Makespan).

Los resultados para  $n = 20$  y  $m = 20$  muestran que estaban muy cerca, en cuanto al Frente de Pareto de referencia (pruebas de Ishibushi), sin embargo, los resultados eran más favorables al objetivo Tardiness que al Makespan. Este presenta un

---

logro eficaz en relación con los rasgos de cobertura, diversidad y convergencia, aunque el Frente de Pareto encontrado no empareje con el Frente de Pareto de referencia.

Para los valores de la métrica  $S$  se encontró que el área que cubre la búsqueda es un 4% menor que el Frente de Pareto de referencia. Además, la métrica  $C$  indica que las soluciones obtenidas no son muy aceptables.

Por otra parte, los resultados para  $n = 40, 60, 80$  manifestaron un logro relativamente eficiente en lo que concierne a la cobertura y diversidad, sin embargo, demostró una evidente dificultad en el acercamiento al Makespan. Esto conduce a la incapacidad de alcanzar una convergencia apropiada hacia el Frente de Pareto de referencia que también es evidenciado en el valor del parámetro  $C$ . Ahora en lo que refiere a la métrica  $S$  se observó, para los tres casos, que el espacio cubierto es, por regla general un 14% menor al Frente de Pareto de referencia.

La razón de obtener estos resultados puede ser explicada debido a la imposibilidad de alcanzar una evaluación de contribución adecuada para cada componente en la solución. Esta dificultad es provocada por la falta de información o datos asociados con cada componente, como un valor de coste, beneficio o peso. Se cree que estos valores deberían ser integrados, más bien que valores aislados, en las funciones objetivas o coacciones.

Para esta investigación los datos provenientes de cada problema incluyen el número de Jobs, el número de máquinas, el vencimiento para cada Job y el tiempo de tratamiento para cada Job en cada máquina. Este último es el valor más relevante para evaluar la contribución en cada componente. Sin embargo, este valor pierde significado pues todos los Job están presentes en la solución. Así, tanto funciones objetivas, Makespan como el Tardiness, deben tratar de encontrar un modo de reducir al mínimo los espacios de tiempo ociosos que podrían ser generados en las máquinas, considerando la permutación corriente de Jobs. En el caso del objetivo basado en el Tardiness, el tiempo de proceso tiene una parte más funcional como este valor puede ser usado para calcular el Tardiness asociado con cada Job en la solución. Este valor de Tardiness puede ser usado para definir el componente que degrada la solución, sin embargo, para la función objetiva basada en el Makespan, no es posible calcular un valor que puede ser asociado a cada trabajo. Esto es porque el Makespan está relacionado con el último trabajo en la permutación. De ahí, este último trabajo, que es representado por el último componente de la solución, siempre será designado como el componente que degrada la solución. Esto quiere decir que la variabilidad para las soluciones en la función Makespan objetiva será menos. Por esta razón la convergencia asociada con el Makespan es deficiente.

---

## 4 Implementación

### 4.1 Modelamiento del Problema

Como se explicó en el capítulo anterior EO intenta representar un ecosistema en el cual interactúan variadas especies. Para el PFSSP el ecosistema estará representado por un vector de tamaño  $n$  (Ver Figura 4.1), donde  $n$  es la cantidad de tareas en el problema, luego las especies serán los Job.

Gráficamente se verá de la siguiente forma.

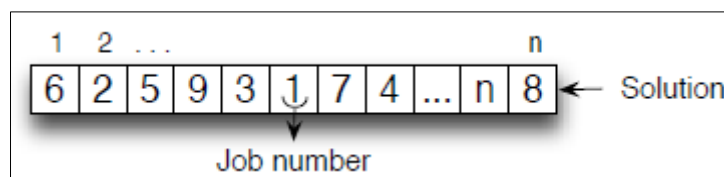


Figura 4.1 Ejemplo del vector solución para el MOPFSSP. (Gómez Meneses P. S., 2012)

Así se tiene que el orden de entrada está dado por la posición del Job en el vector, es decir, según figura 4.1 el primer Job en entrar al proceso es el Job 6, luego el Job 2 y así sucesivamente hasta llegar al Job 8 que será el último en ingresar (recordar que el orden de las máquinas es fijo).

### 4.2 Fitness

Fitness corresponde al criterio particular para elegir una especie o Job que “degrada” el ecosistema e intercambiarlo por otra especie escogida aleatoriamente o con algún método más riguroso.

Debido a que PFSSP es un problema de permutación las especies escogidas no son retiradas del ecosistema sino cambiadas de posición, es decir, se permuta el orden en que entra a ser procesado por las máquinas.

A continuación, se explicarán los dos criterios escogidos:

➤  $Fitness\ Global = Makespan + Tardiness$  , donde

$$Tardiness = \sum_{i=0}^{n-1} D_i - \sum_{i=0}^{n-1} C_{i-max}$$

$$Makespan = \sum_{i=0}^{n-1} TM_i - \sum_{i=0}^{n-1} C_{i-max}$$

$D_i$ : Límite de tiempo para entregar el Job  $i$ .

$C_{i-max}$ : Tiempo que se tarda el Job  $i$  en ser procesado por todas las máquinas, se podría decir que este es el tiempo real de proceso.

$TM_i$ : Tiempo que tarda el Job  $i$ , si fuese el único, en ser procesado por todas las máquinas. Éste podría ser el tiempo ideal el cual sólo corresponde al Job que entra primero, pues éste es procesado sin ninguna espera al cambiar de máquina.

Con el anterior criterio se puede tener en consideración algunos puntos:

- Makespan dará valores comprendidos entre  $]-\infty, 0]$ .
- Valores de Makespan arroja la siguiente información: el tiempo de proceso de cada Job es igual o mayor al tiempo de proceso ideal. Valores más negativos indican un alejamiento mayor del tiempo ideal.
- Tardiness dará valores  $]-\infty, +\infty[$ .
- Valores de Tardiness: informan si el Job ha sido entregado a tiempo. Valores negativos indican que se ha retrasado en la entrega no así los valores positivos.
- Finalmente, Fitness Global tendrá tanto valores positivos como negativos, siendo estos últimos los que “degradan” el ecosistema.

Este criterio se utilizó en la implementación de EO en su estado Puro y EO con búsqueda local.

#### ➤ Fitness con Feromona

Para este caso, que es lo novedoso en este Proyecto de Investigación, se pretende utilizar el concepto de feromona para la evaluación del candidato a ser permutado en EO, como se dijo en la sección 2.1 esto es lo que resulta más complejo en su implementación más aún si las restricciones particulares del problema no están claras. Entonces, se implementará una matriz de tamaño  $n \times n$ , donde  $n$  corresponde a la cantidad de Job presentes, con la cual se evaluará positivamente la secuencia de trabajos ingresados como solución al Frente de Pareto cada vez que se encuentre alguna. La forma de evaluación será la siguiente:

$$Feromonas[i][j] = Feromonas[i][j] + \alpha \text{ Si la secuencia } i, j \in \text{Solución actual.}$$

$\alpha$ : Es un valor constante, para este caso de 0.1.

Solución actual: Corresponde a aquella que acaba de formar parte en el frente de Pareto.



---

$Feromonas[i][j]$ : Corresponde a la matriz que “guarda” la feromona de aquellas soluciones que han formado o forman parte del frente de Pareto.

También se propuso una Feromona Global, a la anterior se le denominará Normal, la cual tiene la propiedad de desvanecer la feromona y se definió de la siguiente forma:

$$Feromonas[i][j] = (1 - \alpha) * Feromonas[i][j] + \alpha * \Delta$$

Donde: 
$$\Delta = \begin{cases} \frac{n}{menor}, & \text{si } i, j \in \text{Solución actual} \\ 0, & \text{en otro caso} \end{cases}$$

Con:  $menor = Cmax/Tmax$

Donde Cmax corresponde al Makespan y Tmax al Tardiness de la solución que se está analizando.

$n$ : Cantidad de Job en el problema.

El cálculo de feromona normal se utiliza al ingresar una nueva solución al frente de Pareto, mientras que el cálculo global se utiliza cada vez que la nueva solución es el mejor Makespan y/o el mejor Tardiness entre todas las soluciones presentes hasta ese momento en el frente de Pareto.

Estos criterios fueron implementados en EO con fitness feromona y EO con búsqueda local y fitness feromona.

A partir de la matriz Feromona se analiza el fitness correspondiente como sigue:

Supongamos el siguiente vector  $Solución Actual = [5, 2, 3, 1, 4, 0]$  para  $n = 6$  que acaba de ingresar al Frente de Pareto. Entonces la matriz feromona será actualizada de la siguiente forma: La primera secuencia (se denominará secuencia al par de Job ligados uno del otro en una solución) 5-2 serán los valores  $i$ - $j$  respectivamente para añadir el valor de feromona en la matriz  $Feromonas[i][j]$ , esta secuencia se guarda en otro arreglo (ver figura 4.3) el cual tiene la función de recordar la posición de la secuencia y los Job que la componen para posteriormente aplicar la permutación (Se explica más adelante). Luego será el turno de la segunda secuencia 2-3 para continuar con la tercera 3-1 hasta llegar a la última secuencia 4-0 como muestra la figura 4.2. Cabe destacar que el número de secuencias guardadas será de  $n - 1$ .

X	0	0	0	0	0
0	X	0	0	0.1	0
0	0	X	0.1	0	0
0	0.1	0	X	0	0
0.1	0	0	0	X	0
0	0	0.1	0	0	X

Figura 4.2 Ejemplo de actualización de feromonas con  $n=6$

5	2
2	3
3	1
1	4
4	0

Figura 4.3 Matriz que memoriza la posición de las secuencias y sus Job

El siguiente paso es ordenar estos valores de feromona para luego escoger, a través del método de la ruleta, uno de ellos, en especial al que “degrada” el sistema, el cual representará a 2 Job.

En este punto ocurrió una leve disyuntiva. ¿Cómo escoger los 2 Job a permutar? ¿Es conveniente permutar la secuencia escogida? Así surgieron algunas alternativas que fueron implementadas y se describen a continuación:

- i. Permutar la secuencia escogida: El método de la ruleta escoge un valor el cual involucra a 2 Job en cierto orden. La idea entonces es simplemente permutar ese orden, es decir, aquel que comienza la secuencia, ahora la finaliza y el que la finaliza ahora la comienza.
- ii. De la secuencia favorecida se escoge un Job aleatoriamente (con igual probabilidad para ambos), luego se escoge el siguiente Job de forma aleatoria entre el resto de Job no escogidos en la secuencia anterior y se procede a la permutación correspondiente.
- iii. De la secuencia favorecida se escoge un Job aleatoriamente (con igual probabilidad para ambos), luego se escoge el siguiente Job con el fitness utilizado en EO Puro (EO sin búsqueda local) y se efectúa la permutación correspondiente.

---

### 4.3 Búsqueda Local

Vectores que tener en consideración para comprender el proceso de Búsqueda local y Catalizador, el algoritmo creado tiene un vector encargado de Makespan y otro para el Tardiness que trabajan en cada iteración:

**VectorS[i][j]:** Vector que guarda el conjunto j de Job que pertenecen a las i-ésimas soluciones del Frente de Pareto encontrados en las iteraciones.

**Multi00[i]:** Vector que mantiene el Makespan de las i-ésimas soluciones guardadas en el Frente de Pareto.

**Multi01[i]:** Vector que mantiene el Tardiness de las i-ésimas soluciones guardadas en el Frente de Pareto.

**Contador:** Variable que mantiene la cuenta actualizada de las soluciones agregadas al VectorS.

Cabe mencionar que las soluciones se guardan en orden ascendente en el valor de Makespan, es decir, en  $i = 0$  se guarda la solución con el menor Makespan y así sucesivamente hasta llegar a  $i = Contador - 1$  donde se guarda la solución con mayor Makespan que, al tratarse de una optimización de 2 objetivos se produce un conflicto entre ellos, es decir, la optimización de uno causa el empeoramiento del otro, por lo que en  $i = Contador - 1$  también se encontrará el menor Tardiness.

Se propuso un formato de búsqueda a través de los algoritmos genéticos, considerando que, en el Frente de Pareto, se están guardando las soluciones temporales (VectorS[i][j]) a medida que se ejecutan las iteraciones. Así a partir del vector solución actual, éste se "cruza" con los vectores en el frente de Pareto para crear hijos, los cuales son analizados y añadidos, si cumple con los requisitos, al frente de Pareto. Este método ha resultado bastante rápido al momento de ejecutar, sin embargo, sus resultados no son del todo favorables, por ello se añadió un "Catalizador" al problema. Éste consiste en dar un límite a la cantidad de veces que no se añaden soluciones al VectorS, es decir, si en cierta cantidad de iteraciones seguidas no se produce un cambio en la variable Contador se extrae desde el VectorS, la primera solución VectorS[0][j] y última solución VectorS[contador - 1][j] hasta aquel momento, y analiza como si fuese el vector actual, logrando así una mejora bastante favorable, no solo para la búsqueda, sino también para EO, sin embargo, este catalizador resulta elitista, pues al centrarse en el mejor Makespan y Tardiness obliga a que los resultados encontrados fluctúen en esas vecindades, dejando de lado los valores obtenidos en el término medio los que pueden ser de igual importancia si se pretende un equilibrio entre ambos.

---

## 4.4 EO Aplicado a MOPFSSP

Se ejecutaron 4 diferentes algoritmos para analizar qué tan efectivo es el fitness a partir de la feromona, en cada uno de ellos se probaron los 4 set de datos utilizados por Ishibushi estos son: 20 Jobs y 20 Máquinas, 40 Jobs y 20 Máquinas, 60 Jobs y 20 Máquinas y 80 Jobs y 20 Máquinas.

Los algoritmos se nombran a continuación, recordar que a todos ellos les fue añadida la función de catalizador explicada en la sección 4.3:

- **EO\_Puro:** Corresponde al análisis del problema MOPFSSP aplicando el algoritmo de EO tal como está descrito en la literatura, con la aptitud de fitness descrita en la sección 4.2.
- **EO\_Puro\_Búsqueda:** Como el anterior, pero se le añadió la función de búsqueda descrita en la sección 4.3.
- **EO\_Feromona:** Aquí se opta por utilizar el concepto de feromona en la función de fitness en EO\_Puro. Este concepto está descrito en la sección 4.2.
- **EO\_Feromona\_Búsqueda:** Al algoritmo anterior se le añade la función de búsqueda local descrita en la sección 4.3.

---

## 5 Experimentos

### 5.1 Introducción

Como se dijo en un comienzo esta investigación fue hecha por prototipos. En su primera fase se utilizó EO aplicado a *knapsack problem* (Karp, 1972), Gracias a esto se logró entender la funcionalidad de EO y la gran importancia de escoger una función apropiada de fitness para lograr buenos resultados y así aprovechar al máximo las propiedades de este algoritmo. La segunda fase consistió en comprender y modelar algorítmicamente el PFSSP y cómo lograr que EO fuese aplicado a este problema con la función de feromona. La tercera etapa y final fue aplicar EO y EO con feromona al problema directamente y analizar los resultados. Estos análisis se efectúan en base a las pruebas obtenidas por Ishibushi (Ishibushi, Yoshida, & Murata, 2003) en su investigación. La idea es conseguir acercarse lo más posible a sus resultados, pero utilizando la metodología de EO con el fitness de feromonas extraídos del algoritmo de Colonia de Hormigas.

### 5.2 Set de Pruebas (*Benchmarks*)

Para esta investigación se utilizaron los resultados obtenidos de los experimentos efectuados por Ishibushi (Ishibushi, Yoshida, & Murata, 2003) para los distintos sets de prueba:

- **20J20M:** Datos de 20 Jobs y 20 Máquinas (20-20).
- **40J20M:** Datos de 40 Jobs y 20 Máquinas (40-20).
- **60J20M:** Datos de 60 Jobs y 20 Máquinas (60-20).
- **80J20M:** Datos de 80 Jobs y 20 Máquinas (80-20).

### 5.3 Resultados

Los resultados se dividirán en 4 etapas con la finalidad de analizar cada algoritmo utilizado y con cada set de pruebas (EO\_Puro, EO\_Puro\_Busqueda, EO\_Feromonas y EO\_Feromonas\_Busqueda).

Se realizó un total de 3 pruebas diferentes con cada algoritmo y para cada set de pruebas las que se describen continuación.

- 
- i. **Variar Iteraciones:** Las iteraciones corresponde a la cantidad de veces que se ejecuta EO en busca de una solución, ésta puede como no pertenecer al Frente de Pareto. Se iteró con 4 cantidades diferentes: 25.000, 250.000, 500.000 y 1.000.000. Valor de semilla 0 y  $\tau = 1,4$ . Gracias a los datos obtenidos en estas pruebas, se confirma que 1.000.000 de iteraciones arroja las mejores soluciones, por lo que las siguientes etapas se ejecutarán con dicha cantidad.
  - ii. **Variar Semillas:** La semilla corresponde al valor que se asigna como parámetro de entrada a la función *Rándom* utilizada para obtener valores al azar. Se hizo variar desde 1 a 20 y considerando 1.000.000 de iteraciones con valor de  $\tau = 1,4$ .  
De los datos obtenidos en esta prueba se consigue obtener aquellas semillas que tiene un mejor comportamiento en la optimización de los objetivos que se propusieron (Makespan y Tardiness) las cuales serán utilizadas en la prueba siguiente.
  - iii. **Variar Tau:** La variable Tau ( $\tau$ ) fue explicada en la sección 2.1. De la prueba anterior se seleccionó la semilla que arrojaba el mejor Makespan y la de mejor Tardiness y se les aplicó 1.000.000 de iteraciones variando el valor de  $\tau$  desde 1,0 hasta 2,5 incrementando en 0,1 en cada prueba.

---

### 5.3.1 EO\_Puro.

#### 5.3.1.1 Análisis 20 Jobs y 20 Máquinas.

##### 5.3.1.1.1 Variar las iteraciones.

En la Figura 5.1 se grafican los resultados obtenidos en el algoritmo de EO\_Puro con los datos del *benchmark* 20 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que los mejores resultados se logran con 1.000.000 de iteraciones.

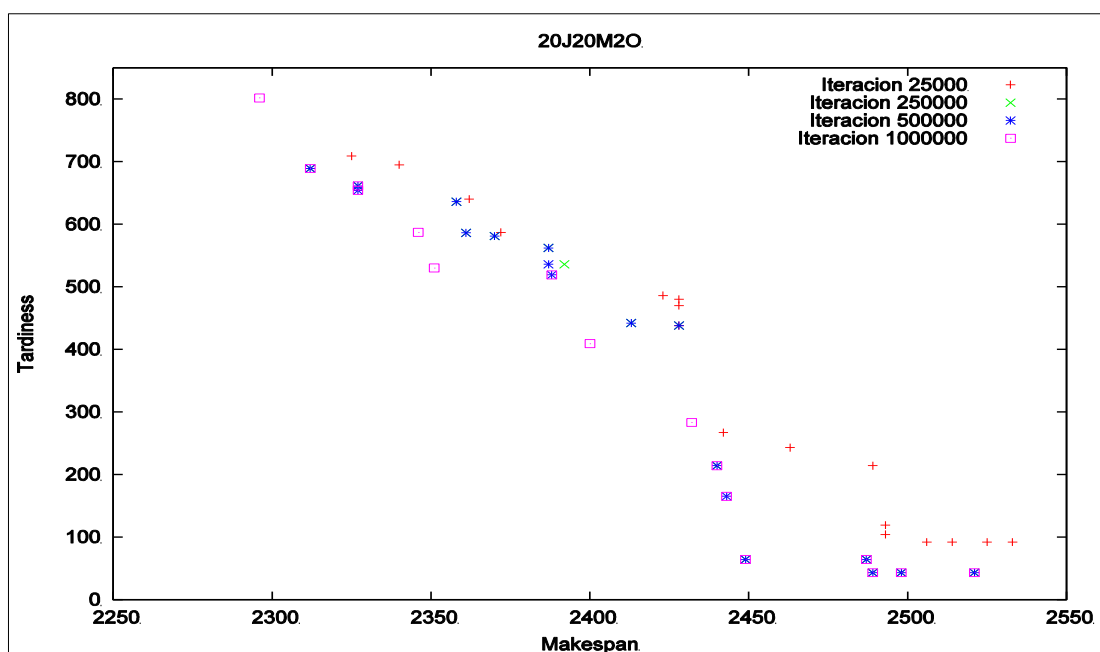


Figura 5.1 Variación de iteraciones para EO\_Puro 20-20

---

### 5.3.1.1.2 Variar el valor de Semillas.

En la Figura 5.2 se grafican los resultados obtenidos en el algoritmo de EO\_Puro con los datos del *benchmark* 20 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness son muy parecidos para la mayor cantidad de semillas, mientras que en el Makespan los valores se dispersan para las distintas semillas.

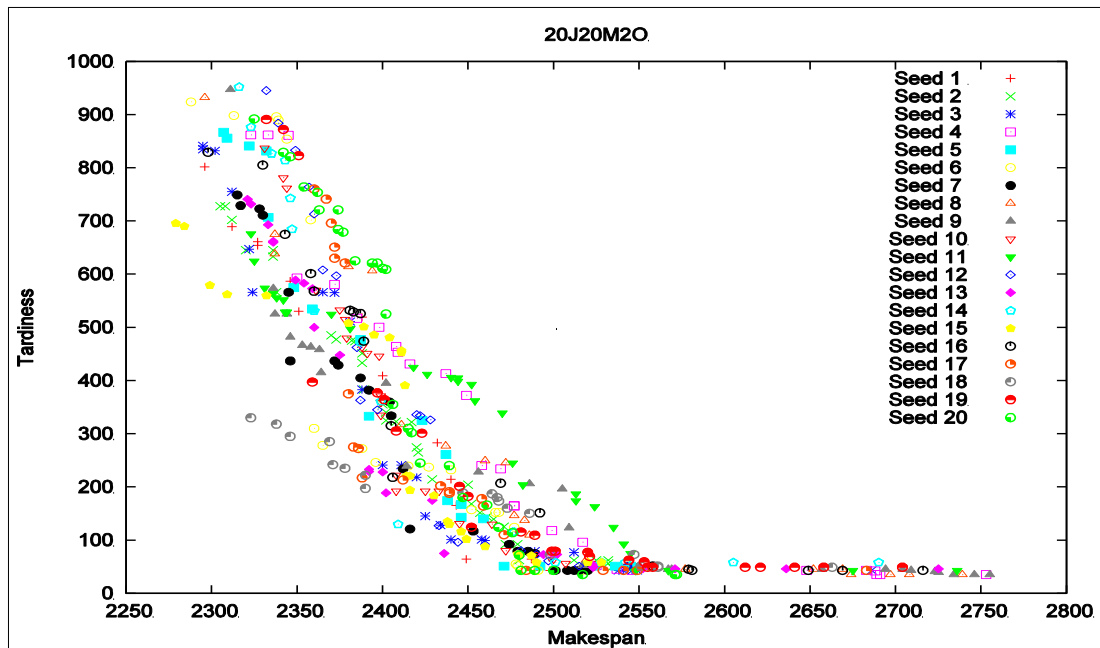


Figura 5.2 Variación de Semillas para EO\_Puro 20-20 con 1.000.000 de iteraciones y  $\tau = 1,4$



De acuerdo con el análisis de los datos el mejor resultado en Makespan lo obtiene la semilla 15 con 2279, mientras que en Tardiness es de la semilla 20 con 35:

Semilla 15		Semilla 20	
Makespan	Tardiness	Makespan	Tardiness
<b>2279</b>	696	2325	892
2284	690	2342	829
2299	579	2346	821
2309	562	2354	764
2332	560	2362	754
2380	508	2363	721
2389	501	2374	721
2395	486	2374	684
2404	481	2377	679
2411	453	2384	625
2411	456	2394	621
2413	391	2397	621
2414	221	2400	611
2416	221	2402	609
2416	194	2402	525
2430	184	2406	355
2438	135	2415	310
2439	130	2417	302
2446	116	2422	245
2449	102	2439	240
2460	88	2447	180
2479	74	2461	166
2480	72	2468	124
2487	71	2476	114
2490	58	2480	72
2520	58	2481	43
2528	58	2489	43
		2500	43
		2517	35
		2571	35
		2572	<b>35</b>

Tabla 5.1 Resultados para semilla 15 y 20 en las pruebas de variación de semillas en EO\_Puro 20-20

### 5.3.1.1.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.3 se muestran los resultados de la variación de  $\tau$  para las semillas 15 y 20, luego en los gráficos 5.4 y 5.5 están los resultados individuales de la semilla 20 que obtuvo el mejor Tardiness y la semilla 15 con el mejor Makespan respectivamente.

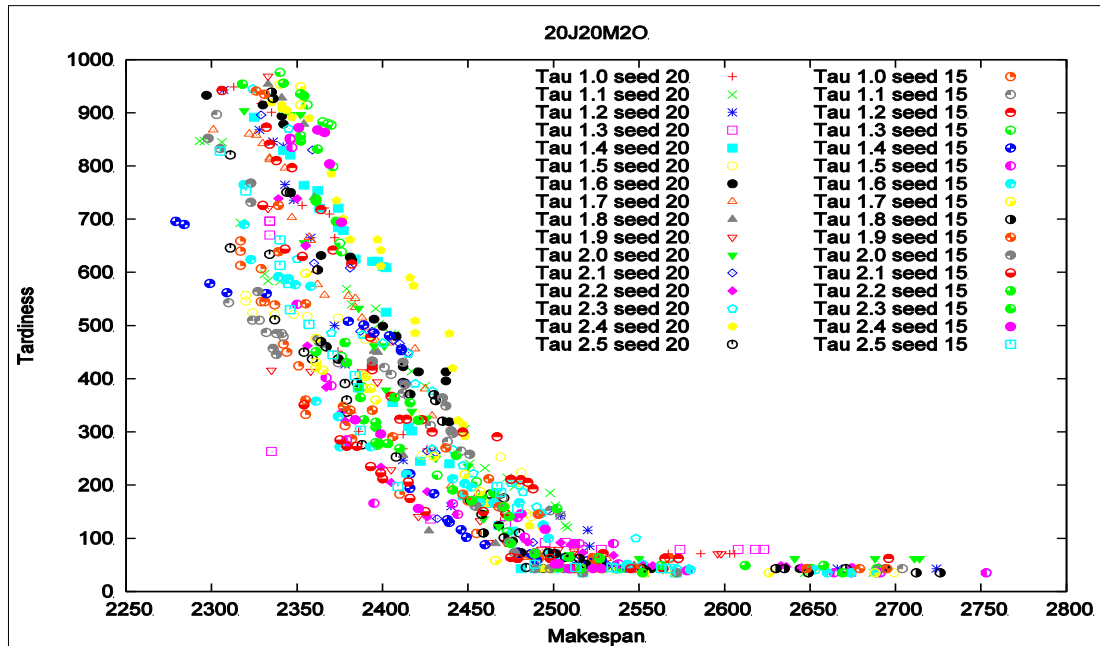


Figura 5.3 Variación de  $\tau$  para EO\_Puro 20-20 con semillas 15 y 20 y 1.000.000 de iteraciones

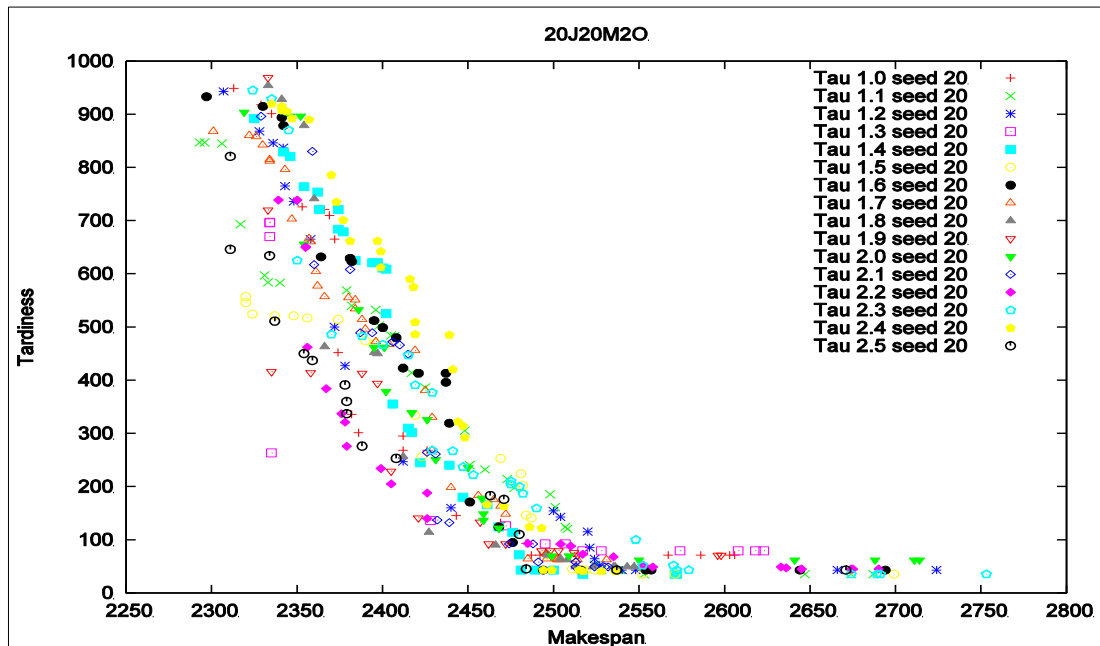


Figura 5.4 Variación de  $\tau$  para EO\_Puro con semilla 20, mejor Tardiness, y 1.000.000 de iteraciones

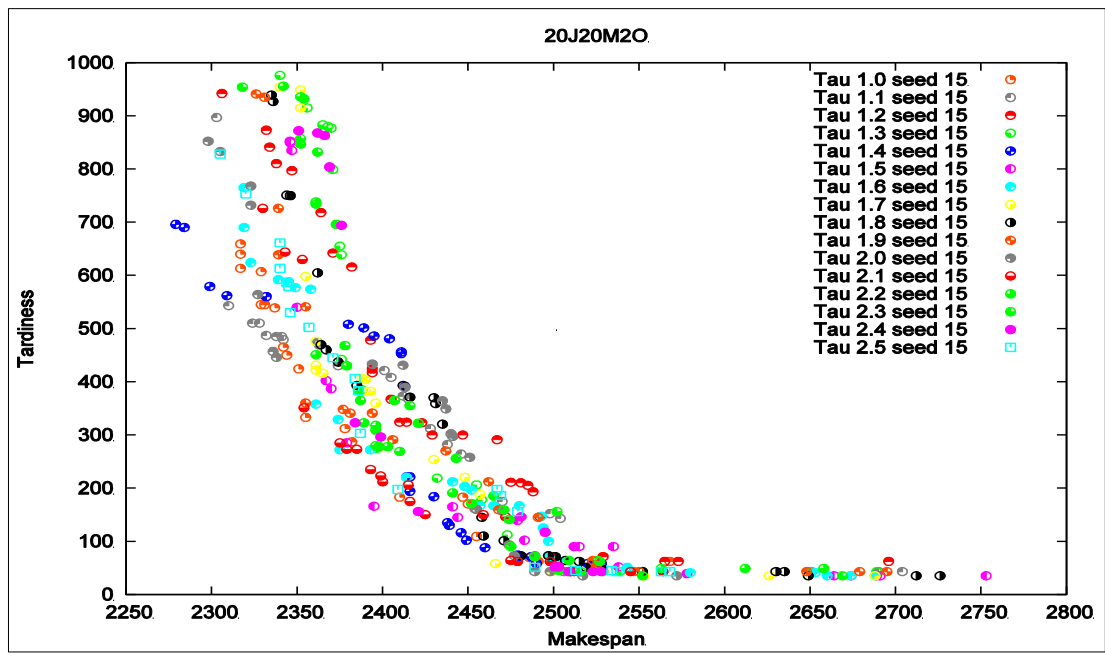


Figura 5.5 Variación de  $\tau$  para EO\_Puro con semilla 15, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 15:** Mejor Makespan para  $\tau = 1,4$  con 2279, mientras que el Tardiness es para  $\tau = 1,5; 1,6; 1,7; 1,8; 2,0; 2,3$  con valor 35, de ellos se escogerá el que tiene la mejor solución (2517, 35) que es  $\tau = 2,0$ .
- ✓ **Semilla 20:** Mejor Makespan para  $\tau = 1,1$  con 2293, mientras que el mejor Tardiness es para  $\tau = 1,1; 1,4; 1,5; 2,3$  con valor 35, se escogerá el que tiene la mejor solución (2517, 35) para  $\tau = 1,4$

Semilla 15				Semilla 20			
$\tau = 1,4$		$\tau = 2,0$		$\tau = 1,1$		$\tau = 1,4$	
<u>2279</u>	696	2298	852	<u>2293</u>	847	2325	892
2284	690	2305	833	2296	847	2342	829
2299	579	2323	768	2306	845	2346	821
2309	562	2323	732	2317	693	2354	764
2332	560	2327	564	2331	597	2362	754
2380	508	2336	457	2333	584	2363	721
2389	501	2338	446	2340	583	2374	721
2395	486	2394	433	2379	569	2374	684
2404	481	2412	431	2382	540	2377	679
2411	453	2413	389	2396	532	2384	625
2411	456	2435	364	2398	503	2394	621
2413	391	2437	349	2405	484	2397	621
2414	221	2440	302	2407	483	2400	611
2416	221	2441	297	2417	414	2402	609
2416	194	2451	258	2425	386	2402	525
2430	184	2454	161	2448	305	2406	355
2438	135	2455	160	2451	240	2415	310
2439	130	2470	157	2460	232	2417	302
2446	116	2474	93	2473	214	2422	245
2449	102	2477	73	2477	197	2439	240
2460	88	2489	43	2498	185	2447	180
2479	74	2489	73	2501	161	2461	166
2480	72	2498	43	2507	124	2468	124
2487	71	2506	43	2508	120	2476	114
2490	58	2510	43	2524	53	2480	72
2520	58	<u>2517</u>	<u>35</u>	2527	52	2481	43
2528	58	2572	35	2528	43	2489	43
				2538	43	2500	43
				2539	43	<u>2517</u>	<u>35</u>
				2553	35	2571	<u>35</u>
				2647	35	2572	<u>35</u>
				2674	35		
				2687	35		

Tabla 5.2 Mejores  $\tau$  para semillas 15 y 20 en EO\_Puro 20-20

---

### 5.3.1.2 Análisis 40 Jobs 20 Máquinas

#### 5.3.1.2.1 Variar las iteraciones.

En la Figura 5.6 se grafican los resultados obtenidos en el algoritmo de EO\_Puro con los datos del *benchmark* 40 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que la diferencia entre 1.000.000 y 500.000 iteraciones es ínfima en cuanto a los mejores resultados obtenidos, sin embargo, se trabajará en las siguientes pruebas con 1.000.000 de iteraciones para mantener homogeneidad en las pruebas.

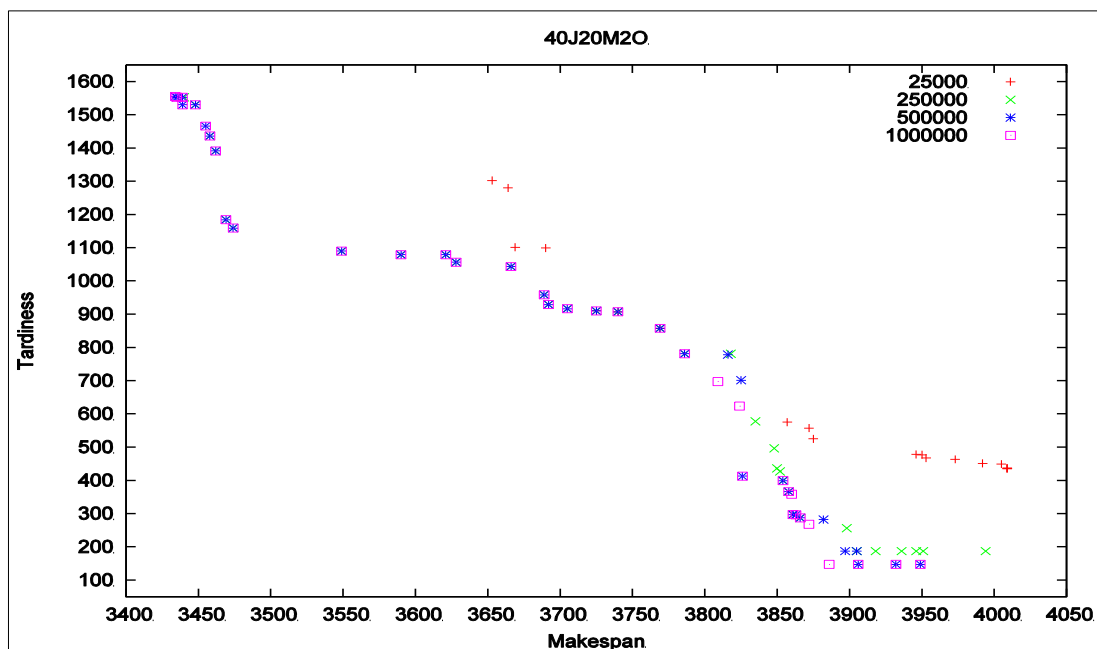


Figura 5.6 Variación de iteraciones para EO\_Puro 40-20 con semilla 0 y  $\tau = 1,4$

### 5.3.1.2.2 Variar el valor de Semillas.

En la Figura 5.7 se grafican los resultados obtenidos en el algoritmo de EO\_Puro con los datos del *benchmark* 40 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness y Makespan se dispersan para las distintas semillas, aún así los valores de Tardiness convergen levemente.

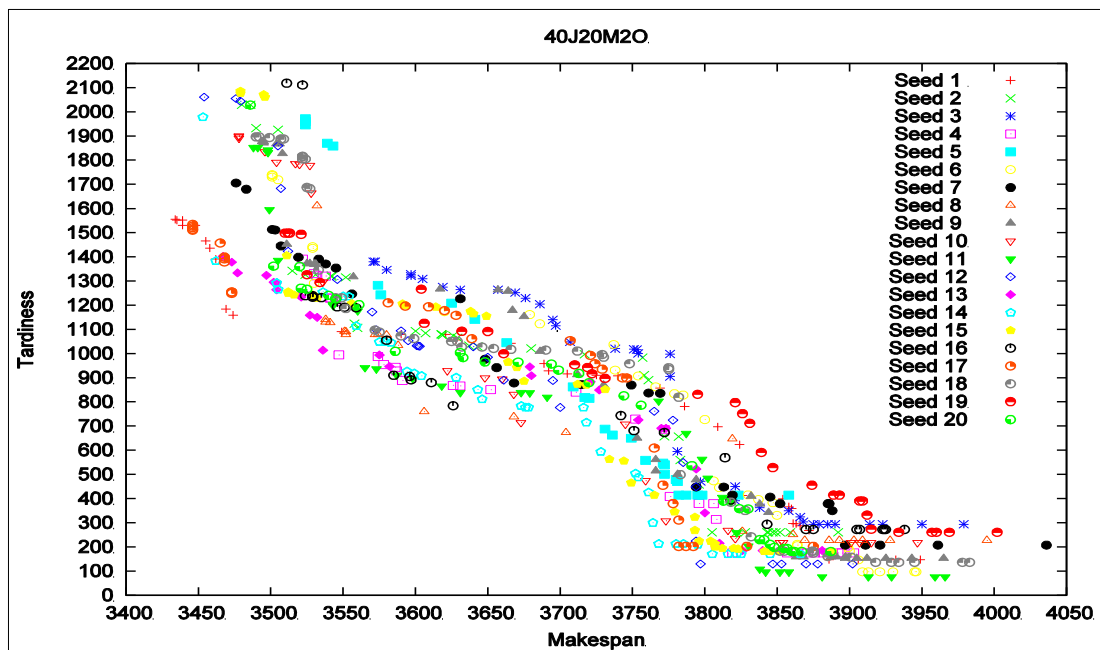


Figura 5.7 Variación de Semillas para EO\_Puro 40-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos el mejor resultado en Makespan lo obtiene la semilla 1 con valor 3434, mientras que en Tardiness es de la semilla 11 con valor 77:

Semilla 1		Semilla 11	
Makespan	Tardiness	Makespan	Tardiness
<b>3434</b>	1555	3488	1853
3435	1552	3491	1853
3439	1530	3498	1843
3439	1552	3498	1832
3448	1530	3499	1596
3455	1466	3505	1386
3458	1436	3543	1204
3462	1391	3543	1199
3469	1184	3558	1193
3474	1159	3560	1177
3549	1090	3565	942
3590	1079	3573	937
3621	1079	3587	911
3628	1056	3598	897
3666	1043	3618	866
3689	958	3631	839
3692	929	3673	839
3705	916	3679	838
3725	910	3691	820
3740	907	3768	804
3769	857	3787	670
3786	781	3798	563
3809	697	3802	485
3824	623	3812	405
3826	413	3818	385
3854	399	3822	259
3858	366	3830	204
3860	358	3838	109
3861	297	3842	97
3863	296	3852	97
3866	287	3858	97
3872	268	3881	<b>77</b>
3886	147	3913	<b>77</b>
3906	147	3929	<b>77</b>
3932	147	3959	<b>77</b>
3949	147	3966	<b>77</b>

Tabla 5.3 Resultados para semilla 1 y 11 en las pruebas de variación de semillas con EO\_Puro 40-20

### 5.3.1.2.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.8 se muestran los resultados de la variación de  $\tau$  para las semillas 1 y 11, luego, en los gráficos 5.9 y 5.10, están los resultados individuales de la semilla 11 que obtuvo el mejor Tardiness y la semilla 1 con el mejor Makespan respectivamente.

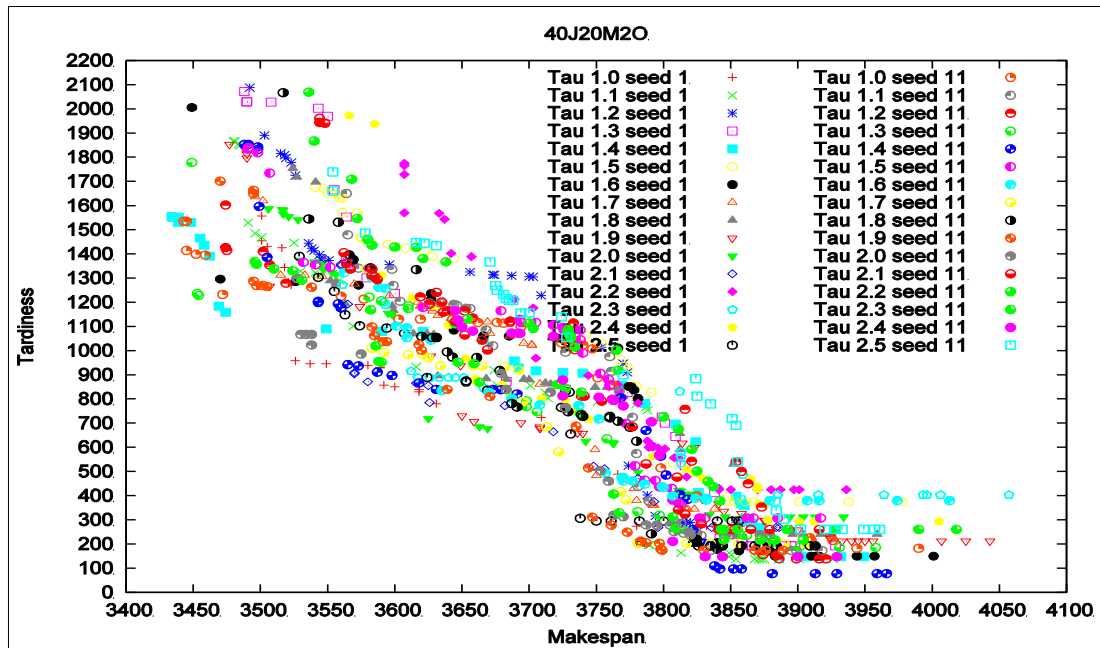


Figura 5.8 Variación de  $\tau$  para EO\_Puro 40-20 con semillas 1 y 11 y 1.000.000 de iteraciones

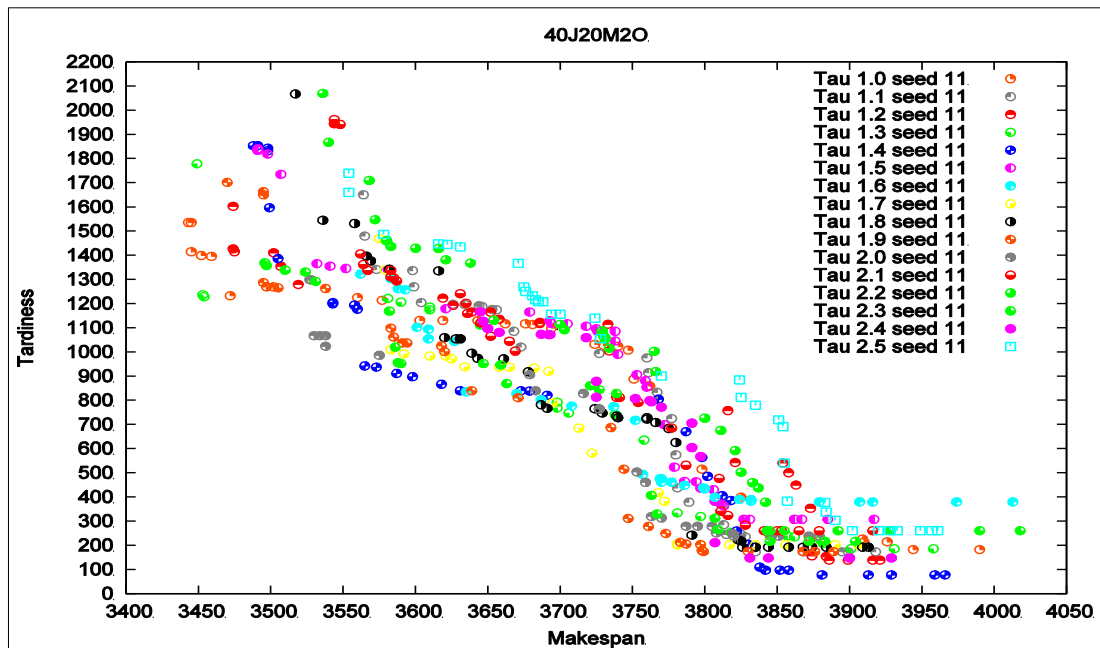


Figura 5.9 Variación de  $\tau$  para EO\_Puro 40-20 con semilla 11, mejor Tardiness, y 1.000.000 de iteraciones



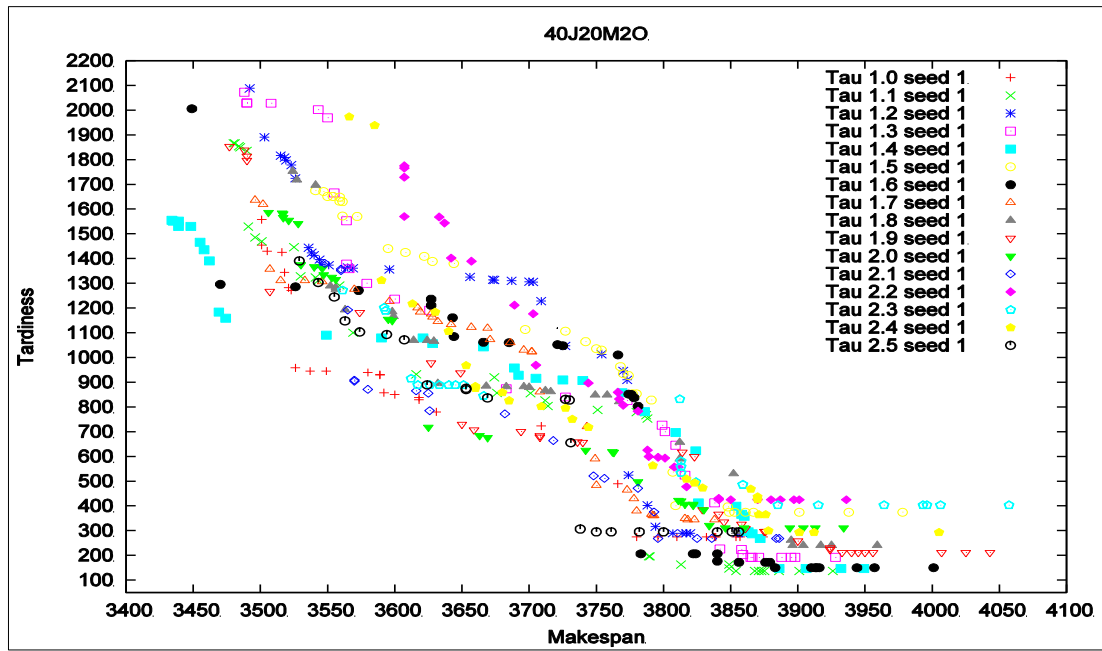


Figura 5.10 Variación de  $\tau$  para EO\_Puro 40-20 con semilla 1, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 1:** Mejor Makespan para  $\tau = 1,4$  con valor 3434, mientras que el Tardiness es para  $\tau = 1,1$  con valor 137.
- ✓ **Semilla 11:** Mejor Makespan para  $\tau = 1,0$  con 3443, mientras que el mejor Tardiness es para  $\tau = 1,4$  con valor 77.

Semilla 1				Semilla 11			
$\tau = 1.4$		$\tau = 1.1$		$\tau = 1.0$		$\tau = 1.4$	
3434	1555	3480	1866	3443	1535	3488	1853
3435	1552	3481	1866	3445	1414	3491	1853
3439	1530	3484	1854	3445	1535	3498	1843
3439	1552	3485	1850	3452	1399	3498	1832
3448	1530	3490	1835	3459	1395	3499	1596
3455	1466	3491	1529	3472	1232	3505	1386
3458	1436	3496	1485	3560	1225	3543	1204
3462	1391	3501	1469	3577	1213	3543	1199
3469	1184	3525	1446	3603	1130	3558	1193
3474	1159	3530	1328	3619	1130	3560	1177
3549	1090	3541	1323	3643	1129	3565	942
3590	1079	3543	1311	3662	1116	3573	937
3621	1079	3559	1291	3676	1116	3587	911
3628	1056	3569	1100	3681	1113	3598	897
3666	1043	3616	932	3694	1098	3618	866
3689	958	3674	919	3724	1031	3631	839
3692	929	3676	856	3730	1027	3673	839
3705	916	3701	855	3740	1023	3679	838
3725	910	3712	826	3747	1007	3691	820
3740	907	3714	804	3751	887	3768	804
3769	857	3751	788	3761	860	3787	670
3786	781	3780	779	3762	858	3798	563
3809	697	3787	767	3798	515	3802	485
3824	623	3788	753	3825	399	3812	405
3826	413	3789	196	3845	240	3818	385
3854	399	3790	196	3873	226	3822	259
3858	366	3813	163	3909	226	3830	204
3860	358	3849	163	3910	214	3838	109
3861	297	3849	147	3926	214	3842	97
3863	296	3854	137	3944	182	3852	97
3866	287	3868	137	3990	182	3858	97
3872	268	3871	137			3881	77
3886	147	3873	137			3913	77
3906	147	3875	137			3929	77
3932	147	3886	137			3959	77
3949	147	3901	137			3966	77
		3926	137				

Tabla 5.4 Mejores  $\tau$  para semillas 1 y 11 en EO\_Puro 40-20

### 5.3.1.3 Análisis 60 Jobs 20 Máquinas

#### 5.3.1.3.1 Variar las iteraciones.

En la Figura 5.11 se grafican los resultados obtenidos en el algoritmo de EO\_Puro con los datos del *benchmark* 60 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que los mejores resultados se logran con 1.000.000 de iteraciones, aunque las 500.000 iteraciones se acercan bastante a resultados similares. Se continuará trabajando con 1.000.000 de iteraciones.

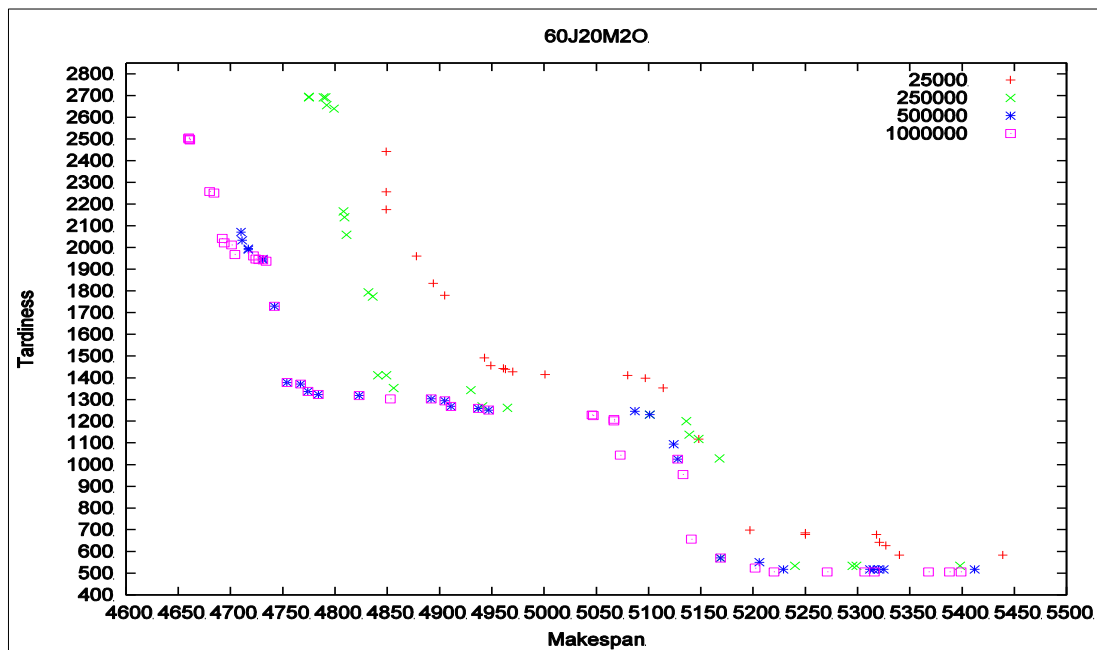


Figura 5.11 Variación de iteraciones para EO\_Puro 60-20 con semilla 0 y  $\tau = 1,4$

### 5.3.1.3.2 Variar el valor de Semillas.

En la Figura 5.12 se grafican los resultados obtenidos en el algoritmo de EO\_Puro con los datos del *benchmark* 60 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness y Makespan se dispersan para las distintas semillas, sin embargo, esta tendencia es mayor para los valores de Makespan.

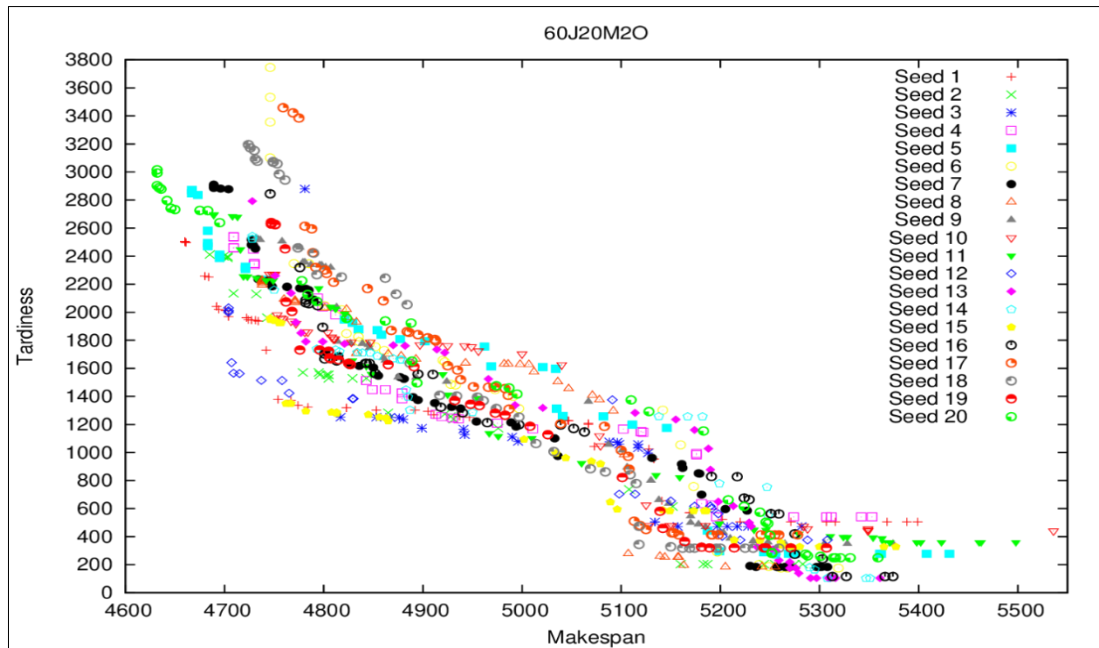


Figura 5.12 Variación de Semillas para EO\_Puro 60-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos el mejor resultado en Makespan lo obtiene la semilla 20, mientras que en Tardiness es de la semilla 14:

Semilla 20		Semilla 14	
Makespan	Tardiness	Makespan	Tardiness
<b>4632</b>	2903	4728	2542
<b>4632</b>	2993	4728	2524
<b>4632</b>	3016	4728	2522
4634	2888	4750	2158
4636	2878	4793	1738
4642	2797	4812	1733
4646	2741	4816	1719
4650	2732	4834	1714
4675	2726	4839	1711
4683	2725	4847	1710
4695	2638	4854	1688
4778	2225	4870	1670
4784	2126	4877	1662
4793	2077	4879	1660
4823	1957	4883	1448
4862	1938	4885	1396
4888	1923	4887	1303
4889	1652	4950	1286
4894	1496	5138	1257
4974	1476	5167	1256
4987	1460	5182	1256
4995	1417	5199	779
5110	1375	5247	753
5128	1292	5248	552
5183	1153	5256	268
5208	662	5290	180
5224	616	5295	165
5240	574	5308	<b>103</b>
5246	503	5347	<b>103</b>
5248	494	5351	<b>103</b>
5250	409		
5252	286		
5252	281		
5286	275		
5288	263		
5301	260		
5314	250		
5316	250		
5321	250		
5331	250		
5359	250		

Tabla 5.5 Resultados para semilla 20 y 14 en las pruebas de variación de semillas con EO\_Puro 60-20

### 5.3.1.3.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.13 se muestran los resultados de la variación de  $\tau$  para las semillas 20 y 14, luego en los gráficos 5.14 y 5.15 están los resultados individuales de la semilla 14 que obtuvo el mejor Tardiness y la semilla 20 con el mejor Makespan respectivamente.

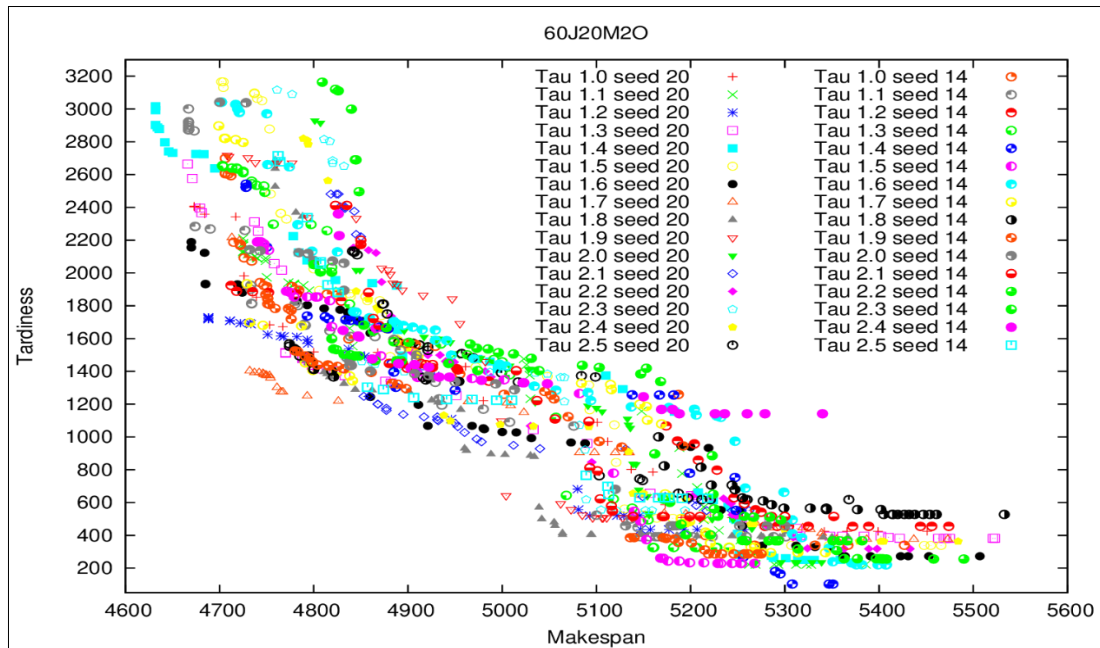


Figura 5.13 Variación de  $\tau$  para EO\_Puro 60-20 con semillas 20 y 14 y 1.000.000 de iteraciones

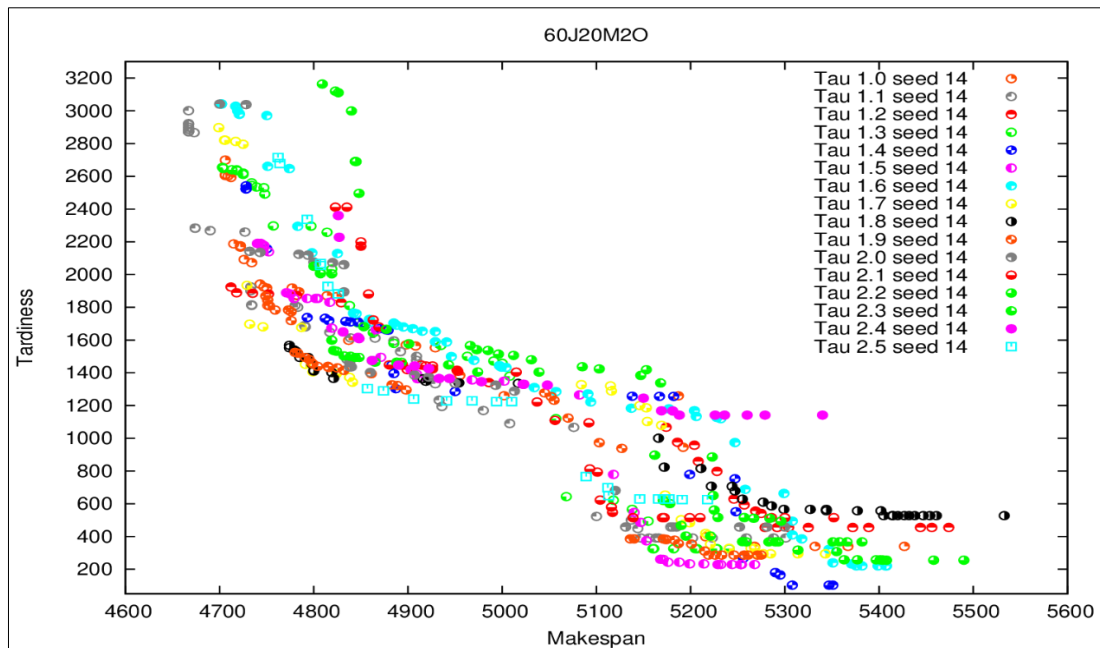


Figura 5.14 Variación de  $\tau$  para EO\_Puro 60-20 con semilla 14, mejor Tardiness, y 1.000.000 de iteraciones

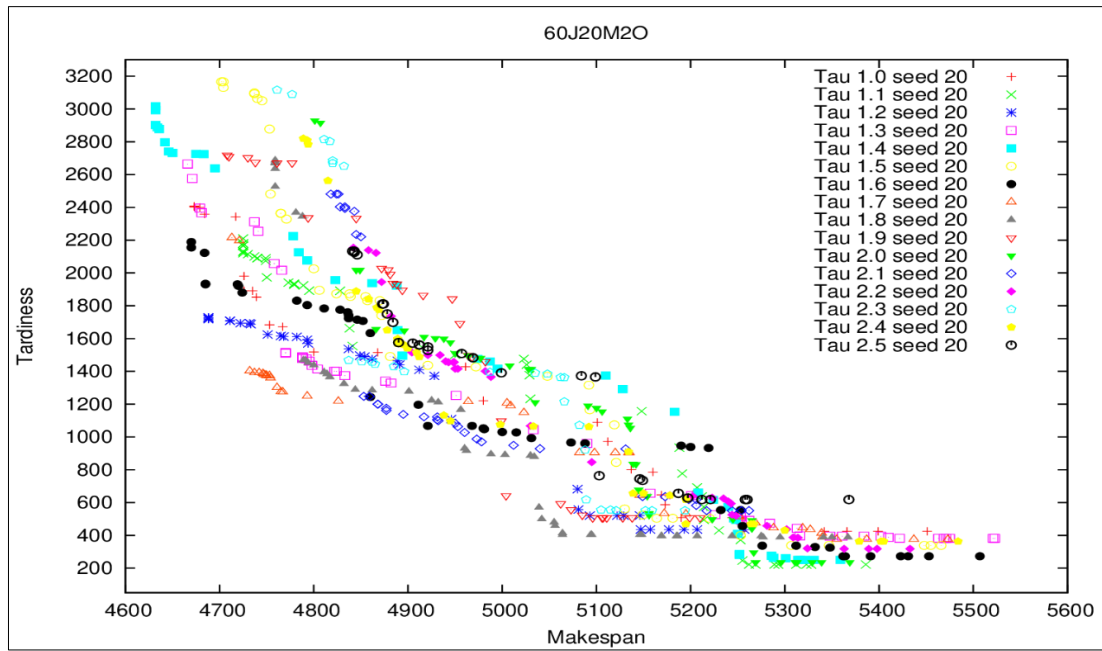


Figura 5.15 Variación de  $\tau$  para EO\_Puro 60-20 con semilla 20, mejor Makespan, y 1.000.000 de iteraciones

Al analizar ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 14:** Mejor Makespan para  $\tau = 1,1$  con 4667, mientras que el Tardiness es para  $\tau = 1,4$  con valor 103.
- ✓ **Semilla 20:** Mejor Makespan para  $\tau = 1,4$  con 4632, mientras que el mejor Tardiness es para  $\tau = 1,1$  con valor 222.

Semilla 14				Semilla 20			
$\tau = 1,1$		$\tau = 1,4$		$\tau = 1,4$		$\tau = 1,1$	
4667	3001	4728	2542	4632	2903	4725	2167
4667	2900	4728	2524	4632	2993	4725	2179
4667	2888	4728	2522	4632	3016	4725	2166
4667	2873	4750	2158	4634	2888	4725	2129
4667	2872	4793	1738	4636	2878	4725	2210
4667	2875	4812	1733	4642	2797	4725	2180
4667	2916	4816	1719	4646	2741	4725	2132
4667	2922	4834	1714	4650	2732	4725	2120
4673	2867	4839	1711	4675	2726	4730	2115
4674	2283	4847	1710	4683	2725	4737	2101
4674	2284	4854	1688	4695	2638	4740	2090
4690	2268	4870	1670	4778	2225	4740	2091
4727	2260	4877	1662	4784	2126	4749	2074
4733	1925	4879	1660	4793	2077	4749	2090
4733	1901	4883	1448	4823	1957	4750	1973
4734	1815	4885	1396	4862	1938	4773	1941
4734	1812	4887	1303	4888	1923	4779	1935
4780	1811	4950	1286	4889	1652	4779	1929
4783	1801	5138	1257	4894	1496	4779	1930
4790	1688	5167	1256	4974	1476	4789	1924
4791	1681	5182	1256	4987	1460	4795	1892
4817	1647	5199	779	4995	1417	4828	1891
4839	1613	5247	753	5110	1375	4838	1663
4865	1612	5248	552	5128	1292	4841	1555
4885	1595	5256	268	5183	1153	4915	1547
4892	1530	5290	180	5208	662	4950	1502
4909	1500	5295	165	5224	616	5023	1476
4909	1476	5308	103	5240	574	5026	1441
4911	1367	5347	103	5246	503	5029	1377
4929	1333	5351	103	5248	494	5030	1233
4933	1233			5250	409	5148	1157
4936	1195			5252	286	5188	934
4980	1170			5252	281	5191	777
5008	1090			5286	275	5207	692
5076	1067			5288	263	5212	637
5100	523			5301	260	5214	495
5137	500			5314	250	5214	574
5144	449			5316	250	5230	430
5147	391			5321	250	5253	372
5156	391			5331	250	5255	246
5163	391			5359	250	5262	222
5166	391					5269	222
5202	391					5289	222
5223	391					5293	222
5259	391					5312	222
5288	391					5323	222
5301	391					5328	222
						5386	222

Tabla 5.6 Mejores  $\tau$  para semillas 14 y 20 en EO\_Puro 60-20.



### 5.3.1.4 Análisis 80 Jobs 20 Máquinas

#### 5.3.1.4.1 Variar las iteraciones.

En la Figura 5.16 se grafican los resultados obtenidos en el algoritmo de EO\_Puro con los datos del *benchmark* 80 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que los mejores resultados se logran con 1.000.000 de iteraciones.

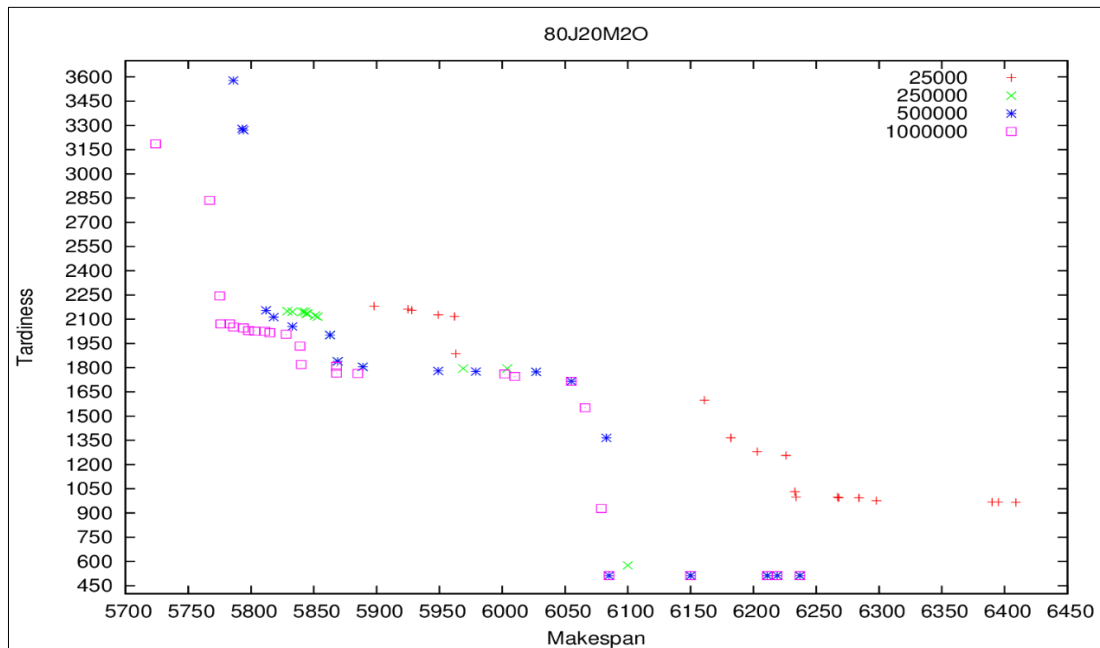


Figura 5.16 Variación de iteraciones para EO\_Puro 80-20 con semilla 0 y  $\tau = 1,4$ .

### 5.3.1.4.2 Variar el valor de Semillas.

En la Figura 5.17 se grafican los resultados obtenidos en el algoritmo de EO\_Puro con los datos del *benchmark* 80 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness y Makespan son dispersos para las distintas semillas.

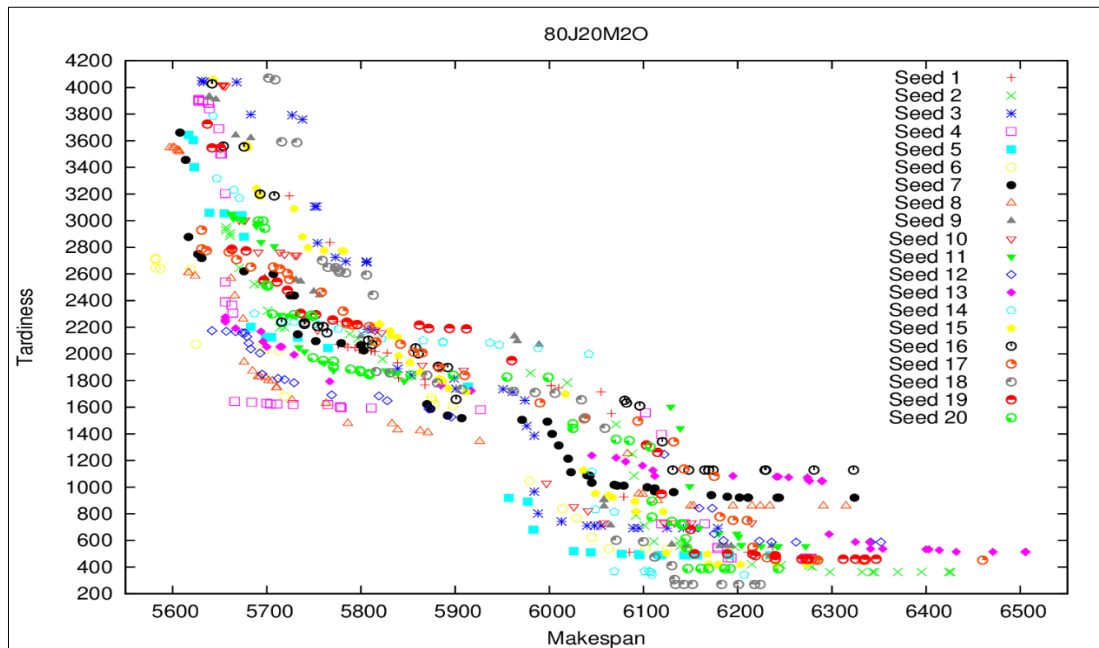


Figura 5.17 Variación de Semillas para EO\_Puro 80-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo al análisis de los datos el mejor resultado en Makespan lo obtiene la semilla 6, mientras que en Tardiness es de la semilla 18:

Semilla 6		Semilla 18	
Makespan	Tardiness	Makespan	Tardiness
<b>5582</b>	2709	5702	4071
<b>5582</b>	2716	5709	4058
<b>5582</b>	2644	5716	3592
5587	2641	5732	3586
5619	2636	5759	2702
5625	2073	5765	2650
5695	2051	5773	2647
5699	2033	5776	2629
5713	2023	5777	2614
5719	1682	5784	2608
5875	1672	5806	2593
5875	1652	5813	2441
5878	1624	5816	1867
5898	1610	5831	1858
5979	1047	5870	1840
6014	840	5881	1782
6029	767	5907	1733
6045	623	5962	1719
6063	539	5985	1710
6102	539	6006	1705
6125	539	6034	1656
6145	539	6038	1521
		6059	1442
		6071	602
		6100	591
		6112	475
		6130	411
		6133	307
		6134	<b>270</b>
		6143	<b>270</b>
		6152	<b>270</b>
		6183	<b>270</b>
		6201	<b>270</b>
		6218	<b>270</b>
		6224	<b>270</b>

Tabla 5.7 Resultados para semilla 6 y 18 en las pruebas de variación de semillas con EQ\_Puro 80-20

### 5.3.1.4.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.18 se muestran los resultados de la variación de  $\tau$  para las semillas 6 y 18, luego en los gráficos de las Figuras 5.19 y 5.20 están los resultados individuales de la semilla 18 que obtuvo el mejor Tardiness y la semilla 6 con el mejor Makespan respectivamente.

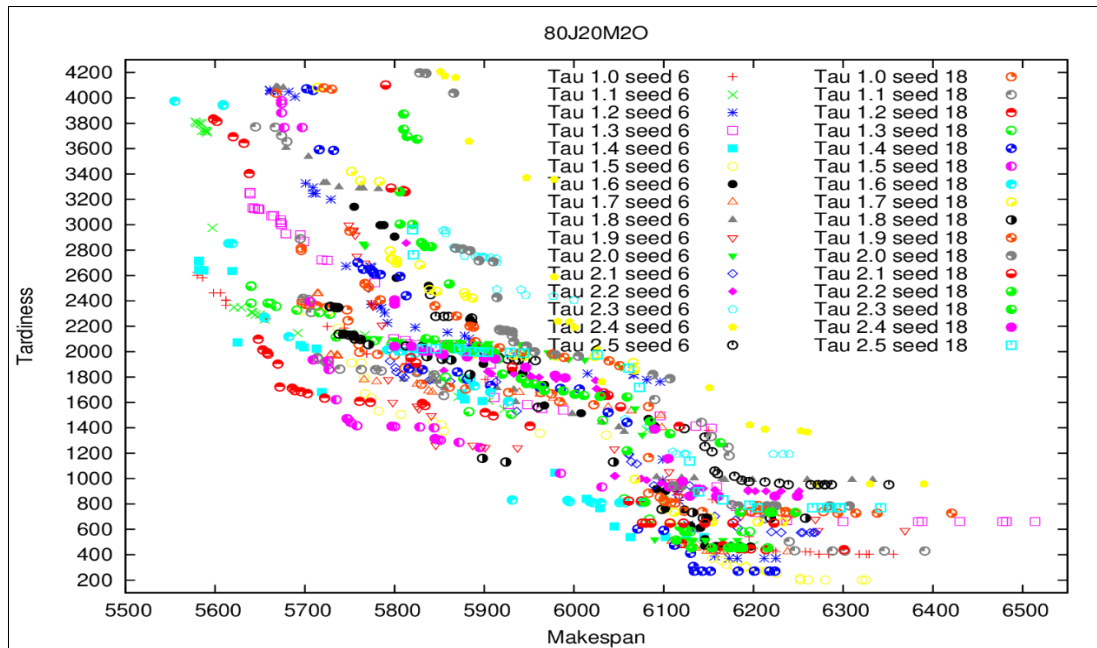


Figura 5.18 Variación de  $\tau$  para EO\_Puro 80-20 con semillas 6 y 18 y 1.000.000 de iteraciones

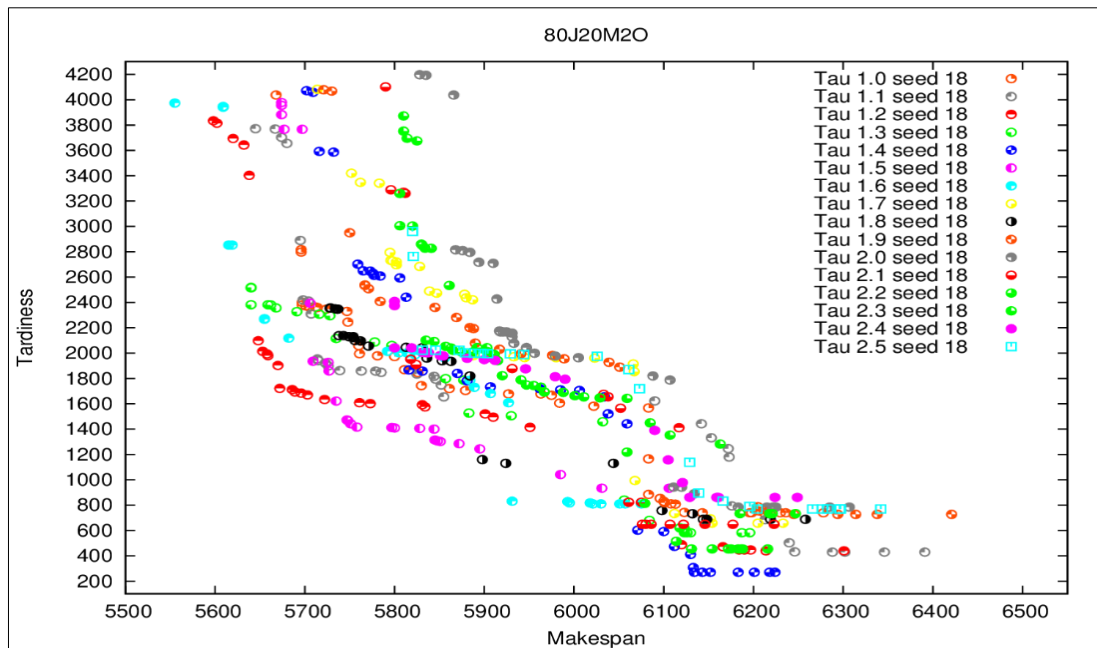


Figura 5.19 Variación de  $\tau$  para EO\_Puro 80-20 con semilla 18, mejor Tardiness, y 1.000.000 de iteraciones

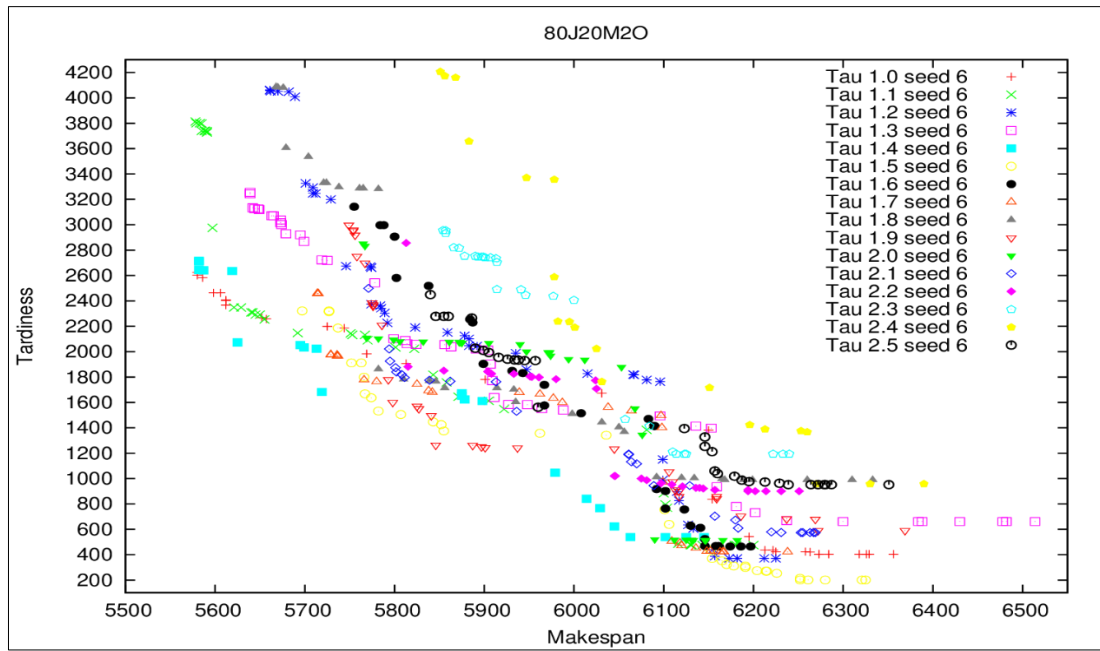


Figura 5.20 Variación de  $\tau$  para EO\_Puro 80-20 con semilla 6, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 6:** Mejor Makespan para  $\tau = 1,1$  con 5578, mientras que el Tardiness es para  $\tau = 1,5$  con valor 201.
- ✓ **Semilla 18:** Mejor Makespan para  $\tau = 1,6$  con 5555, mientras que el mejor Tardiness es para  $\tau = 1,4$  con valor 270.

Semilla 6				Semilla 18			
$\tau = 1,1$		$\tau = 1,5$		$\tau = 1,6$		$\tau = 1,4$	
5578	3814	5697	2321	5555	3974	5702	4071
5579	3801	5727	2321	5609	3941	5709	4058
5583	3801	5727	2315	5609	3947	5716	3592
5585	3740	5737	2186	5615	2853	5732	3586
5585	3798	5752	1912	5619	2853	5759	2702
5588	3740	5763	1912	5655	2272	5765	2650
5589	3740	5766	1799	5655	2267	5773	2647
5591	3727	5767	1668	5682	2118	5776	2629
5591	3737	5774	1637	5792	2014	5777	2614
5597	2976	5782	1532	5805	2003	5784	2608
5621	2349	5807	1505	5814	2001	5806	2593
5629	2348	5843	1446	5823	1987	5813	2441
5640	2309	5852	1425	5856	1962	5816	1867
5642	2309	5855	1375	5884	1786	5831	1858
5645	2292	5962	1357	5889	1730	5870	1840
5650	2292	6036	1342	5907	1681	5881	1782
5655	2254	6101	755	5927	1608	5907	1733
5692	2147	6106	639	5931	831	5962	1719
5751	2144	6152	441	5993	828	5985	1710
5753	2132	6154	370	5995	818	6006	1705
5767	2132	6165	352	6018	818	6034	1656
5770	2090	6170	323	6021	810	6038	1521
5801	2034	6178	310	6030	810	6059	1442
5822	2022	6191	310	6051	810	6071	602
5843	1818	6191	299	6057	810	6100	591
5858	1757	6204	274	6075	810	6112	475
5871	1645	6214	270			6130	411
5905	1612	6215	270			6133	307
5922	1547	6226	253			6134	270
6081	1384	6252	214			6143	270
6100	884	6252	201			6152	270
6102	796	6261	201			6183	270
6104	770	6280	201			6201	270
6116	487	6321	201			6218	270
6130	482	6325	201			6224	270
6130	476						
6131	476						
6143	476						
6179	476						
6181	476						
6200	476						

Tabla 5.8 Mejores  $\tau$  para semillas 6 y 18 en EO\_Puro 80-20

---

## 5.3.2 EO\_Puro\_Busqueda

### 5.3.2.1 Análisis 20 Jobs 20 Máquinas

#### 5.3.2.1.1 Variar las iteraciones.

En la Figura 5.21 se grafican los resultados obtenidos en el algoritmo de EO\_Puro\_Busqueda con los datos del *benchmark* 20 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que los resultados de las distintas iteraciones son similares. Se continuará utilizando 1.000.000 de iteraciones para las siguientes pruebas.

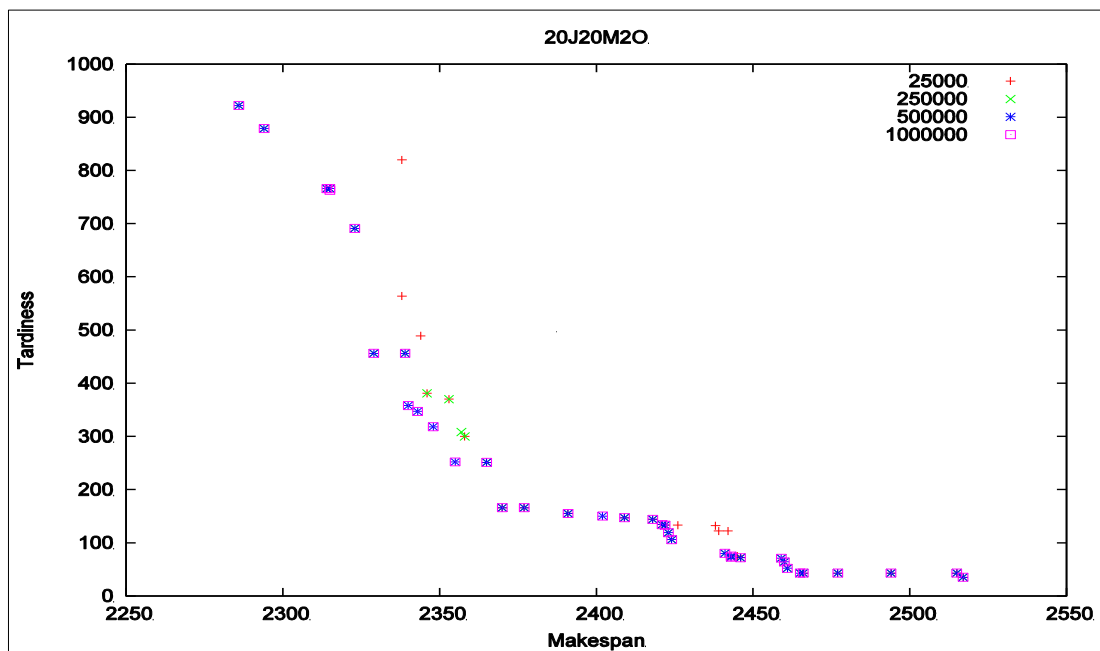


Figura 5.21 Variación de iteraciones para EO\_Puro\_Busqueda 20-20 con semilla 0 y  $\tau = 1,4$

### 5.3.2.1.2 Variar el valor de Semillas.

En la Figura 5.22 se grafican los resultados obtenidos en el algoritmo de EO\_Puro\_Busqueda con los datos del *benchmark* 20 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness son muy parecidos para la mayor cantidad de semillas, mientras que en el Makespan los valores se dispersan levemente para las distintas semillas.

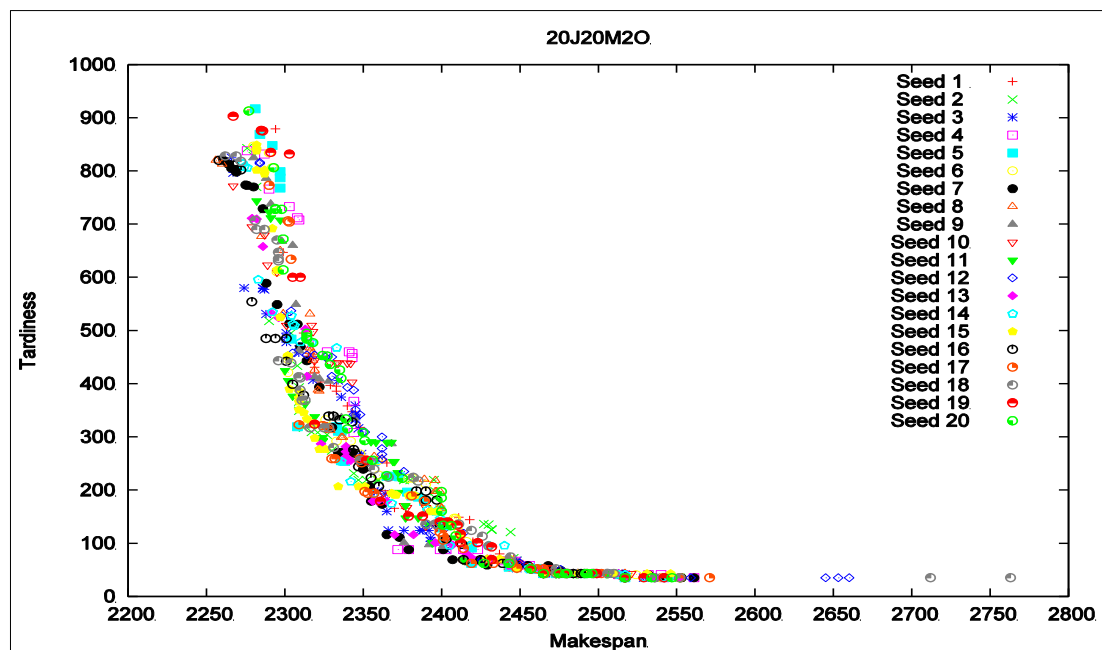


Figura 5.22 Variación de Semillas para EO\_Puro\_Busqueda 20-20 con 1.000.000 de iteraciones y  $\tau = 1,4$



De acuerdo al análisis de los datos el mejor resultado en Makespan lo obtiene la semilla 8 con valor 2256, mientras que en Tardiness las mejores semillas son 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 11, 16, 17, 18, 19, 20 con valor 35 y, utilizando el criterio de la mejor solución, las semillas se reducen a 1, 2, 5, 8, 9, 11, 12, 13, 16, 17, 18, 19, 20 con el par [2517, 35] y como decisión final se escoge aquella que tiene el mejor Makespan global que es la semilla 8 con el valor 2256, por tanto se utiliza sólo esta semilla para la siguiente prueba:

Makespan	Semilla 8	Tardiness
<b>2256</b>		820
2260		813
2285		677
2297		650
2299		531
2316		460
2316		531
2316		467
2318		458
2318		444
2319		425
2321		388
2322		386
2327		319
2329		312
2329		316
2336		299
2337		299
2357		263
2362		258
2389		220
2396		197
2396		218
2397		197
2399		169
2399		197
2400		166
2403		108
2414		69
2419		69
2436		67
2441		67
2443		55
2443		62
2456		54
2461		52
2466		43
2477		43
2480		43
2487		43
2493		43
2498		43
2517		<b>35</b>
2517		43
2536		<b>35</b>
2552		<b>35</b>

Tabla 5.9 Resultados para semilla 8 en las pruebas de variación de semillas con EO\_Puro\_Busqueda 20-20

---

### 5.3.2.1.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.23 se muestran los resultados de la variación de  $\tau$  para la semilla 8, la cual obtuvo los mejores resultados de Tardiness y Makespan.

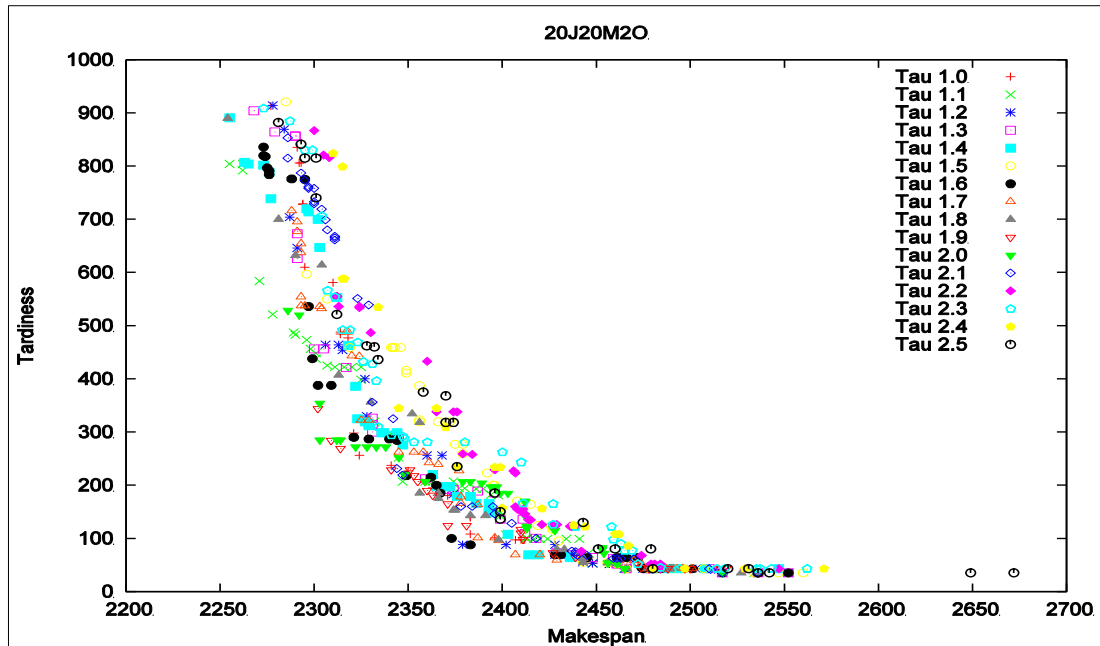


Figura 5.23 Variación de  $\tau$  para EO\_Puro\_Busqueda 20-20 con semilla 8, mejor Makespan y Tardiness, y 1.000.000 de iteraciones

---

En el análisis realizado a la semilla 8 se encontraron los siguientes resultados:

- ✓ **Semilla 8:** Mejor Makespan para tau 1.8 con 2254, mientras que el mejor Tardiness es para  $\tau = 1,0; 1,1; 1,2; 1,3; 1,5; 1,6; 1,8; 2,0; 2,1$  y 2,5 con valor 35, como segundo criterio el que tiene la mejor solución (2517, 35) para el  $\tau = 1,0; 1,1; 1,2; 1,3; 1,6; 1,8; 2,0$  y 2,1. Finalmente se escogerá el que tiene el mejor Makespan que es el  $\tau = 1,8$ .

Semilla 8	
$\tau=1,8$	
2254	890
2281	700
2290	632
2304	614
2313	407
2330	356
2352	334
2356	185
2356	318
2366	185
2366	175
2374	153
2375	153
2383	143
2391	143
2398	97
2429	79
2433	79
2443	62
2443	55
2456	54
2461	52
2466	43
2477	43
2481	43
2489	43
2490	43
2491	43
2517	35
2527	35
2536	35

Tabla 5.10 *Mejor  $\tau$  para semilla 8 en EO\_Puro\_Busqueda 20-20*

### 5.3.2.2 Análisis 40 Jobs 20 Máquinas

#### 5.3.2.2.1 Variar las iteraciones.

En la Figura 5.24 se grafican los resultados obtenidos en el algoritmo de EO\_Puro\_Busqueda con los datos del *benchmark* 40 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . Se logra observar que los mejores resultados son para 1.000.000 de iteraciones.

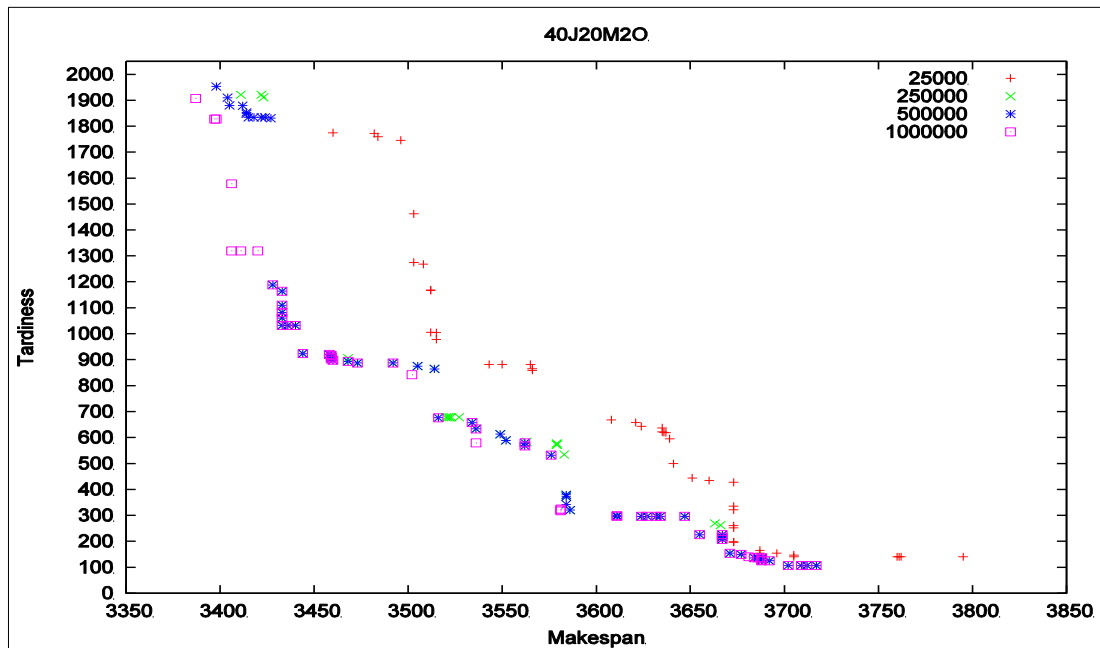


Figura 5.24 Variación de iteraciones para EO\_Puro\_Busqueda 40-20 con semilla 0 y  $\tau = 1,4$

### 5.3.2.2.2 Variar el valor de Semillas.

En la Figura 5.25 se grafican los resultados obtenidos en el algoritmo de EO\_Puro\_Busqueda con los datos del *benchmark* 40 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness son muy parecidos para la mayor cantidad de semillas, mientras que en el Makespan los valores se dispersan para las distintas semillas.

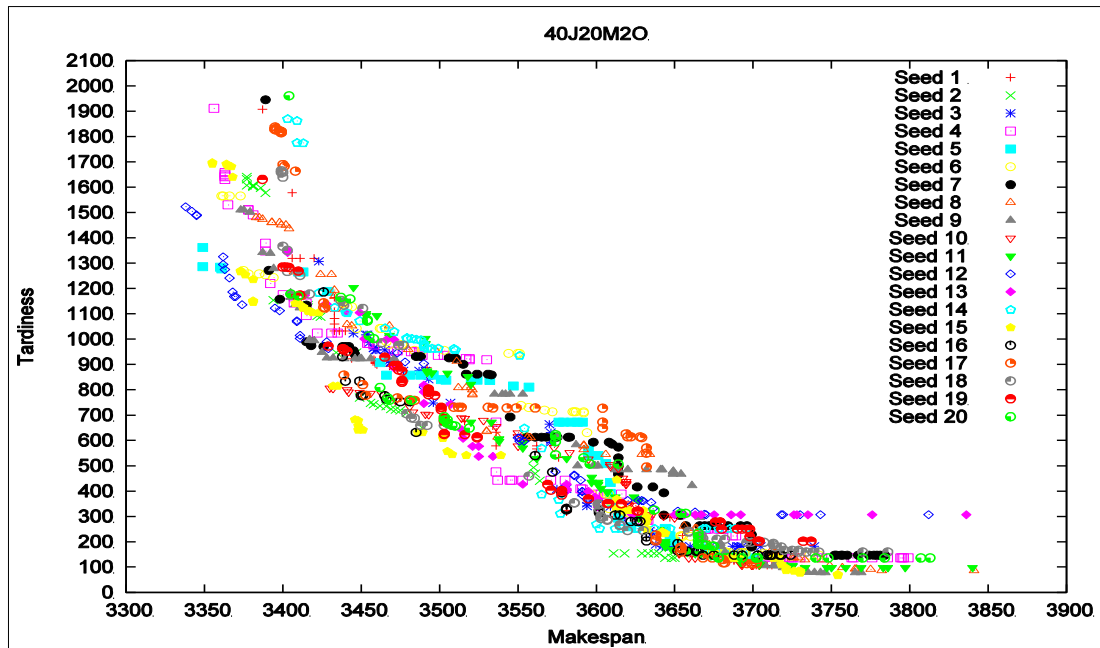


Figura 5.25 Variación de Semillas para EO\_Puro\_Busqueda 40-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos el mejor resultado en Makespan lo obtiene la semilla 12 con valor 3338, mientras que en Tardiness la mejor semilla es la 15 con valor 67:

Semilla 12		Semilla 15	
Makespan	Tardiness	Makespan	Tardiness
<b>3338</b>	1523	3355	1697
3342	1505	3355	1692
3345	1488	3364	1692
3345	1491	3365	1684
3362	1295	3367	1682
3362	1325	3368	1641
3363	1275	3373	1270
3366	1241	3373	1267
3368	1186	3376	1255
3369	1168	3381	1151
3370	1168	3381	1146
3374	1136	3381	1236
3395	1123	3408	1142

3398	1112	3410	1142
3409	1073	3413	1127
3409	1069	3413	1123
3411	1003	3416	1111
3411	1015	3417	1111
3428	990	3419	1105
3429	962	3423	1105
3473	945	3432	815
3478	910	3435	815
3490	810	3446	682
3490	903	3448	679
3550	610	3448	642
3551	610	3448	658
3551	597	3451	642
3554	597	3486	632
3574	477	3489	632
3586	460	3502	618
3586	464	3502	609
3590	443	3505	557
3591	399	3508	545
3591	398	3517	542
3591	384	3539	542
3595	368	3613	368
3595	363	3613	445
3601	363	3613	331
3620	363	3613	330
3627	363	3617	324
3629	363	3621	324
3630	361	3630	324
3635	355	3632	299
3636	331	3632	279
3652	321	3632	250
3663	317	3642	240
3669	309	3642	233
3670	307	3644	233
3718	307	3675	159
3719	307	3685	136
3730	307	3689	136
3743	307	3690	136
3812	307	3691	136
		3713	136
		3717	136
		3718	111
		3721	97
		3721	87
		3726	85
		3729	85
		3730	85
		3730	79
		3730	77
		3754	72
		3754	<b>67</b>

Tabla 5.11 *Resultados para semilla 12 y 15 en las pruebas de variación de semillas con EO\_Puro\_Busqueda 40-20*

### 5.3.2.2.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.26 se muestran los resultados de la variación de  $\tau$  para las semillas 12 y 15, luego en los gráficos de las Figuras 5.27 y 5.28 están los resultados individuales de la semilla 15 que obtuvo el mejor Tardiness y la semilla 12 con el mejor Makespan respectivamente.

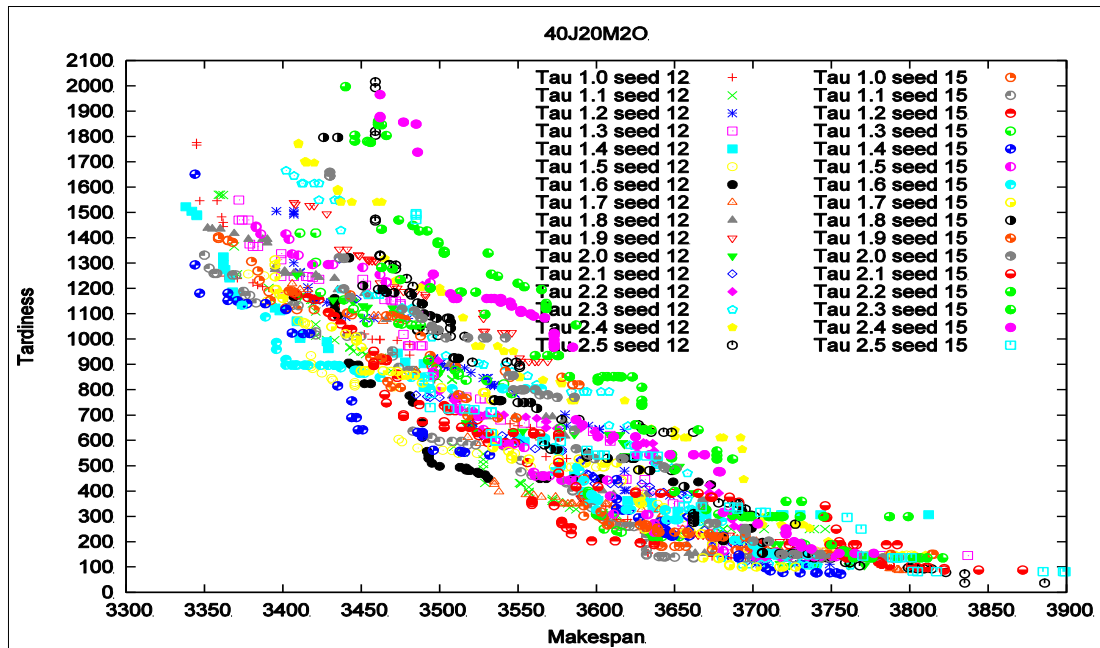


Figura 5.26 Variación de  $\tau$  para EO\_Puro\_Busqueda 40-20 con semillas 12 y 15 y 1.000.000 de iteraciones

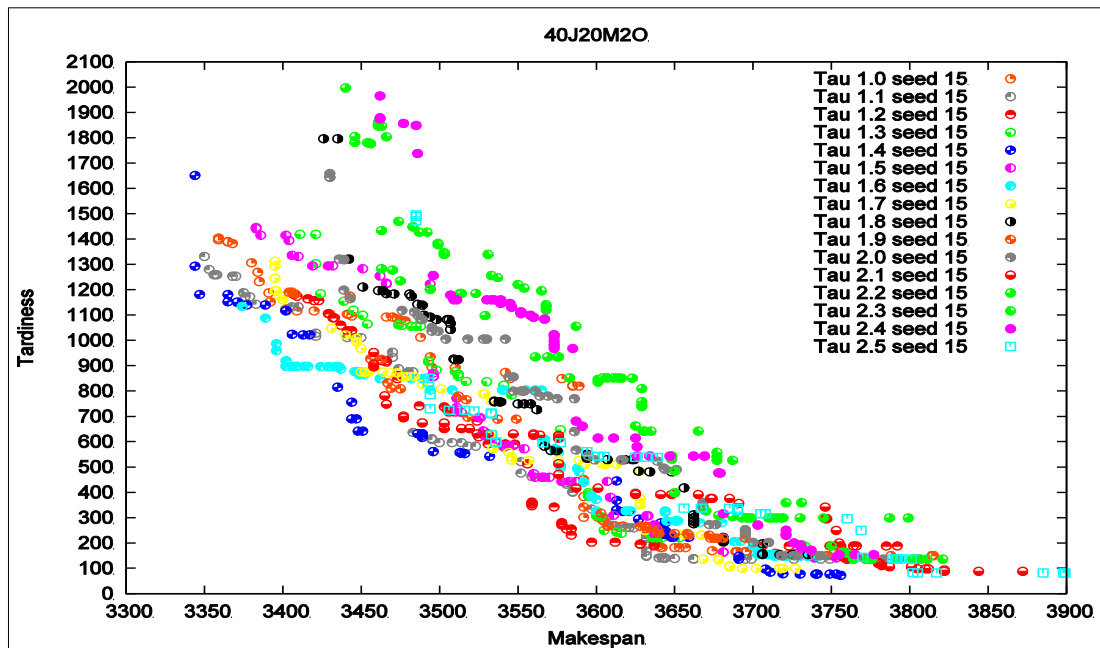


Figura 5.27 Variación de  $\tau$  para EO\_Puro\_Busqueda 40-20 con semilla 15, mejor Tardiness, y 1.000.000 de iteraciones

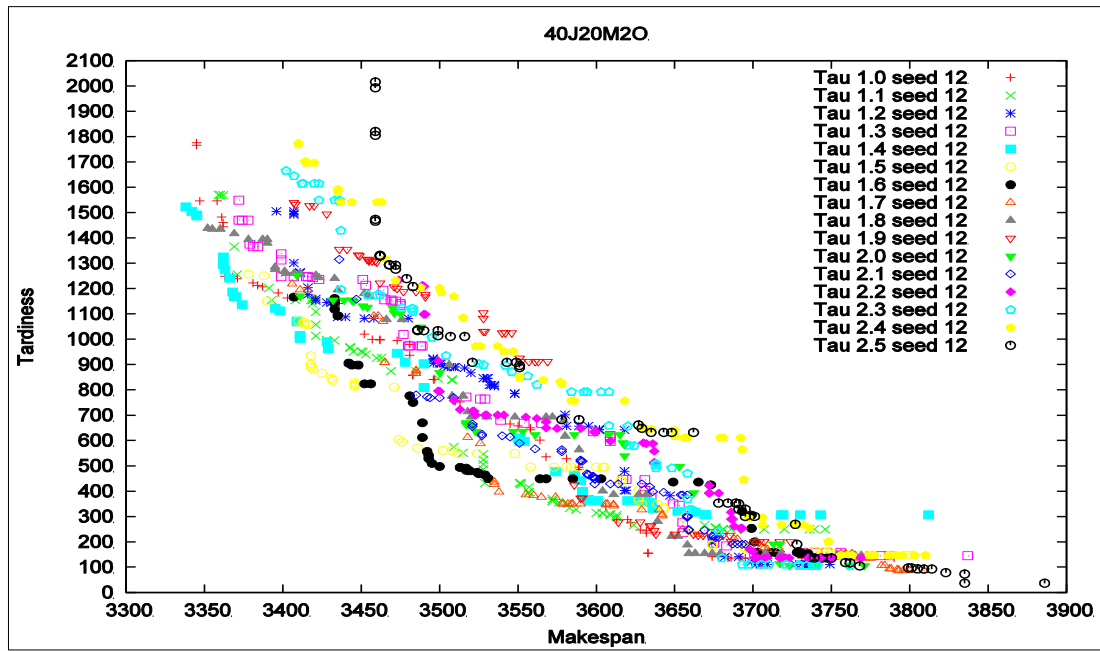


Figura 5.28 Variación de  $\tau$  para EO\_Puro\_Busqueda 40-20 con semilla 12, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 12:** Mejor Makespan para  $\tau = 1,4$  con 3338, mientras que el Tardiness es para  $\tau = 2,5$  con valor 36.
- ✓ **Semilla 15:** Mejor Makespan para  $\tau = 1,4$  con 3344 y mejor Tardiness es para  $\tau = 1,4$  con valor 72.

Semilla 12				Semilla 15	
$\tau = 1.4$		$\tau = 2.5$		$\tau = 1.4$	
3338	1523	3459	1466	3344	1293
3342	1505	3459	1993	3344	1651
3345	1488	3459	1805	3347	1181
3345	1491	3459	1473	3365	1181
3362	1295	3459	2016	3365	1153
3362	1325	3459	1820	3371	1151
3363	1275	3462	1329	3377	1140
3366	1241	3462	1328	3389	1140
3368	1186	3462	1331	3402	1118
3369	1168	3468	1294	3402	1116
3370	1168	3468	1293	3406	1024
3374	1136	3472	1292	3413	1022
3395	1123	3472	1277	3417	1022
3398	1112	3479	1239	3435	815
3409	1073	3483	1208	3444	756



3409	1069	3486	1037	3444	690
3411	1003	3486	1034	3447	690
3411	1015	3490	1034	3448	642
3428	990	3499	1014	3451	642
3429	962	3499	1034	3486	632
3473	945	3507	1011	3489	617
3478	910	3516	1011	3489	625
3490	810	3521	909	3489	632
3490	903	3543	909	3496	561
3550	610	3549	909	3513	557
3551	610	3551	896	3514	554
3551	597	3551	888	3516	553
3554	597	3578	682	3532	542
3574	477	3589	682	3613	331
3586	460	3627	661	3613	330
3586	464	3629	648	3613	445
3590	443	3635	632	3613	368
3591	399	3643	632	3617	324
3591	398	3648	632	3621	324
3591	384	3662	632	3627	294
3595	368	3678	353	3641	279
3595	363	3684	353	3644	233
3601	363	3689	353	3644	247
3620	363	3691	329	3644	279
3627	363	3691	350	3644	240
3629	363	3695	329	3644	246
3630	361	3695	300	3648	231
3635	355	3701	300	3648	222
3636	331	3727	269	3649	222
3652	321	3728	190	3659	222
3663	317	3729	160	3681	222
3669	309	3730	154	3691	136
3670	307	3739	136	3691	147
3718	307	3750	136	3708	97
3719	307	3759	118	3711	85
3730	307	3762	117	3719	78
3743	307	3768	104	3730	77
3812	307	3799	97	3740	77
		3801	97	3741	77
		3805	93	3749	77
		3809	92	3753	77
		3814	92	3756	72
		3823	78		
		3835	73		
		3835	36		
		3886	36		

Tabla 5.12 Mejores  $\tau$  para semillas 12 y 15 en EO\_Puro\_Busqueda 40-20

### 5.3.2.3 Análisis 60 Jobs 20 Máquinas

#### 5.3.2.3.1 Variar las iteraciones.

En la Figura 5.29 se grafican los resultados obtenidos en el algoritmo de EO\_Puro\_Busqueda con los datos del *benchmark* 60 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que los mejores resultados se logran con 1.000.000 de iteraciones.

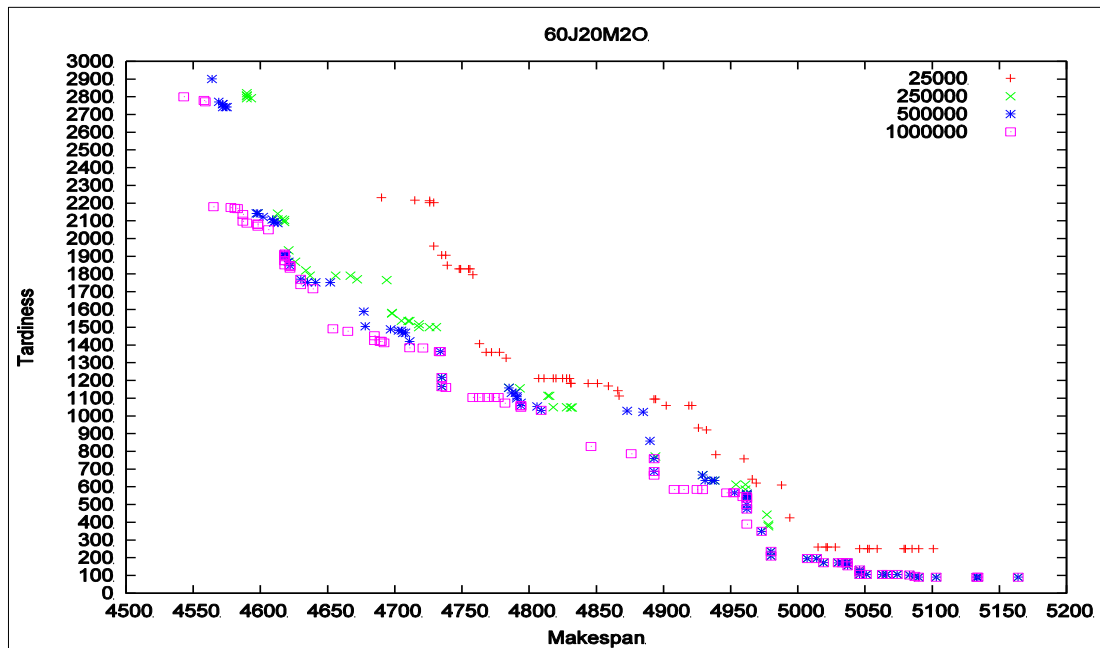


Figura 5.29 Variación de iteraciones para EO\_Puro\_Busqueda 60-20 con semilla 0 y  $\tau = 1,4$

### 5.3.2.3.2 Variar el valor de Semillas.

En la Figura 5.30 se grafican los resultados obtenidos en el algoritmo de EO\_Puro\_Busqueda con los datos del *benchmark* 60 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness son muy parecidos para la mayor cantidad de semillas, mientras que en el Makespan los valores se dispersan.

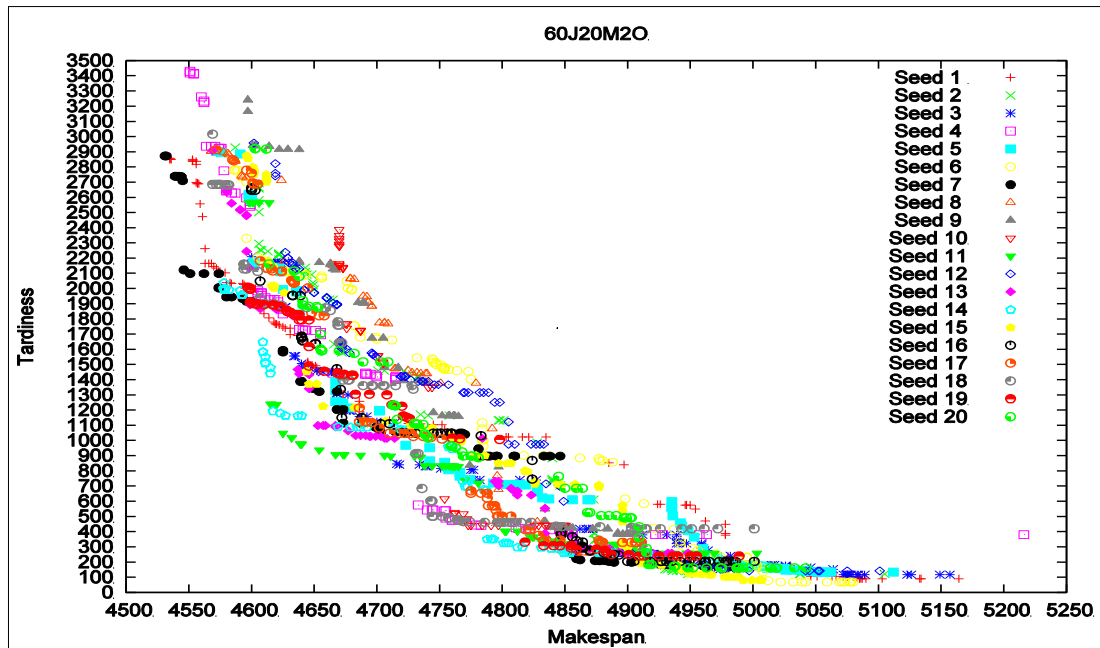


Figura 5.30 Variación de Semillas para EO\_Puro\_Busqueda 60-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

El mejor resultado en Makespan lo obtiene la semilla 7 con valor 4531, mientras que en Tardiness la mejor semilla es la 6 con valor 68:

Semilla 7		Semilla 6	
Makespan	Tardiness	Makespan	Tardiness
<b>4531</b>	2872	4586	2780
4532	2872	4587	2780
4539	2740	4589	2780
4540	2740	4592	2733
4542	2740	4596	2700
4544	2737	4596	2330
4544	2735	4605	2207
4545	2710	4608	2190
4546	2123	4616	2147
4551	2099	4617	2142
4562	2099	4629	2109
4574	2006	4629	2118
4574	2003	4631	2108
4574	2099	4632	2108
4580	1945	4643	2093
4584	1945	4643	2088
4593	1934	4656	2069
4596	1916	4656	2083
4600	1916	4656	2070
4625	1593	4676	2048
4625	1583	4677	2001
4639	1389	4680	1994
4640	1389	4680	1993
4649	1343	4680	1990
4654	1322	4682	1756
4668	1204	4682	1754
4668	1322	4687	1718
4673	1114	4687	1679
4673	1204	4699	1679
4688	1110	4700	1666
4699	1087	4712	1661
4703	1056	4732	1543
4715	1056	4744	1537
4721	1056	4745	1514
4766	1050	4745	1530
4768	1050	4747	1500
4770	1040	4747	1511
4781	946	4751	1495
4782	898	4752	1480
4784	898	4754	1472
4788	898	4754	1467
4790	898	4754	1480
4795	898	4754	1479
4796	898	4757	1466
4809	898	4764	1460
4810	898	4773	1454
4833	898	4775	1403
4838	898	4784	1002
4846	898	4784	1116
4854	311	4784	1077
4856	253	4785	963
4859	222	4785	1002
4860	222	4817	901
4862	214	4833	901
4874	209	4862	885

4877	209	4877	880
4878	209	4878	866
4882	209	4880	866
4885	203	4888	858
4889	198	4898	616
4903	198	4913	585
4926	184	4921	422
4929	184	4937	415
4930	184	4942	314
4933	184	4942	336
4944	184	4952	289
4947	184	4956	275
4948	182	4970	252
4959	180	4974	233
4961	169	4995	229
4964	169	4995	232
4972	169	4995	231
4981	169	4999	186
4985	169	5003	180
		5003	169
		5008	135
		5008	162
		5012	77
		5022	<u>68</u>
		5032	<u>68</u>
		5039	<u>68</u>
		5042	<u>68</u>
		5059	<u>68</u>
		5069	<u>68</u>
		5071	<u>68</u>
		5073	<u>68</u>
		5077	<u>68</u>
		5078	<u>68</u>
		5080	<u>68</u>

Tabla 5.13 *Resultados para semilla 7 y 6 en las pruebas de variación de semillas con EO\_Puro\_Busqueda 60-20*

### 5.3.2.3.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.31 se muestran los resultados de la variación de  $\tau$  para las semillas 6 y 7, luego en los gráficos de las Figuras 5.32 y 5.33 están los resultados individuales de la semilla 6 que obtuvo el mejor Tardiness y la semilla 7 con el mejor Makespan respectivamente.

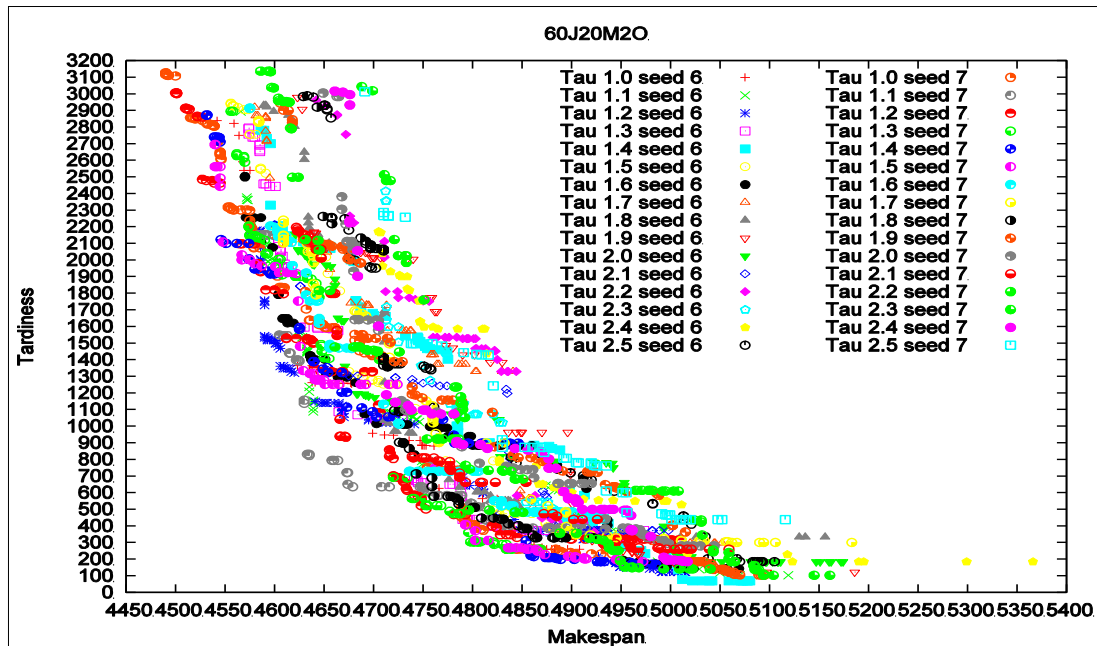


Figura 5.31 Variación de  $\tau$  para EO\_Puro\_Busqueda 60-20 con semillas 15 y 20 y 1.000.000 de iteraciones

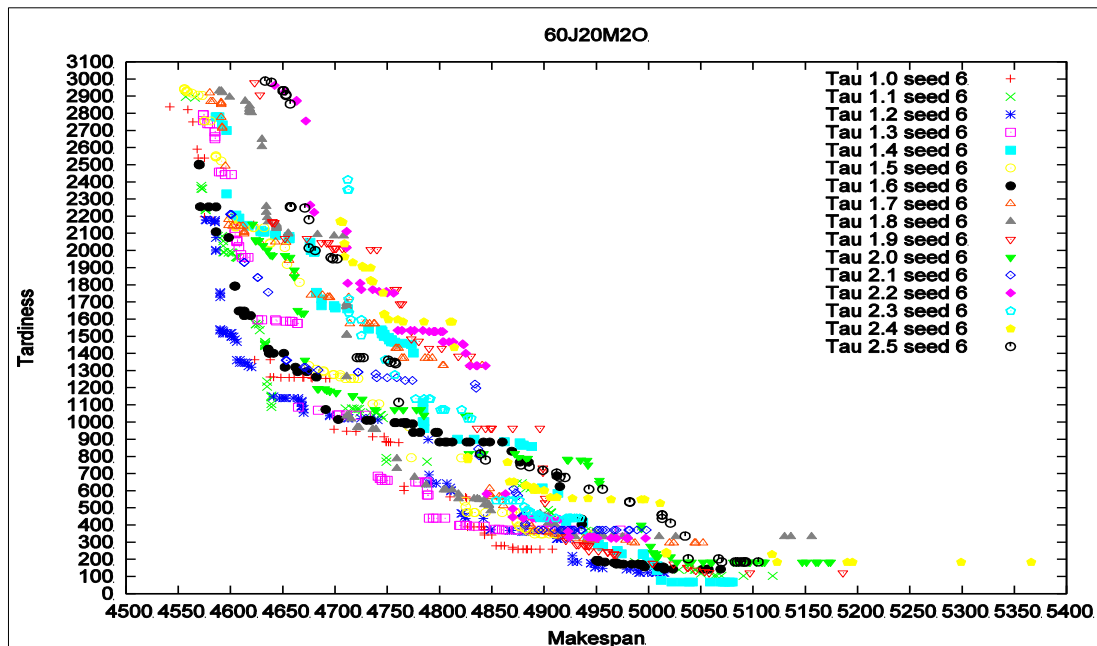


Figura 5.32 Variación de  $\tau$  para EO\_Puro\_Busqueda 60-20 con semilla 6, mejor Tardiness, y 100000 de iteraciones

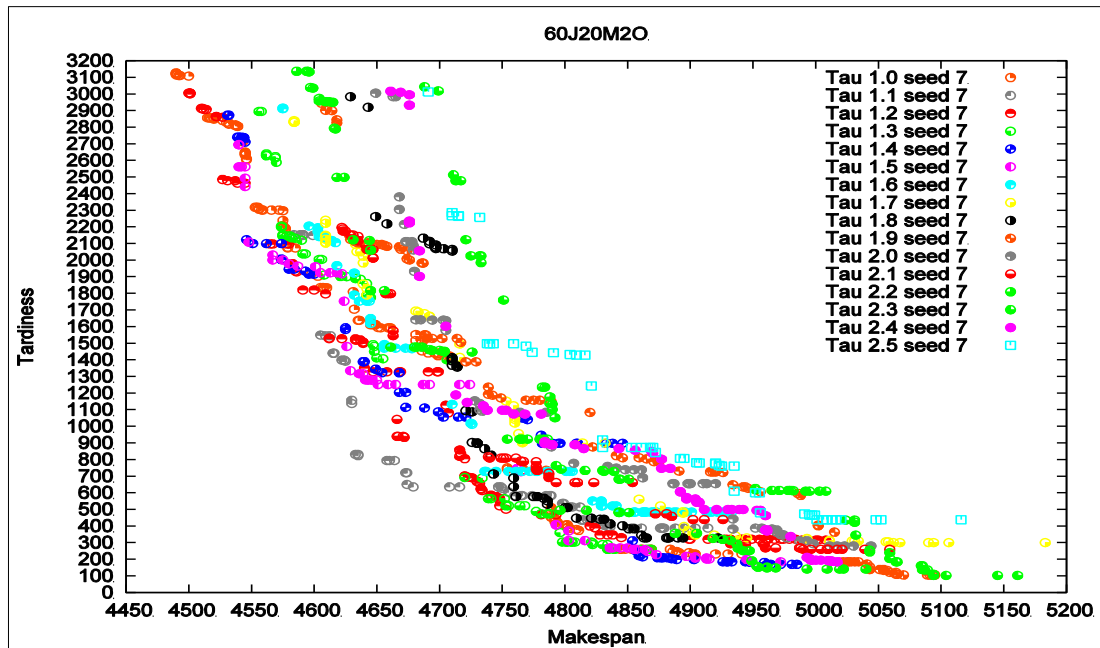


Figura 5.33 Variación de  $\tau$  para EO\_Puro\_Busqueda 60-20 con semilla 7, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 6:** Mejor Makespan para  $\tau = 1,0$  con 4542, mientras que el Tardiness es para  $\tau = 1,4$  con valor 68.
- ✓ **Semilla 7:** Mejor Makespan para  $\tau = 1,0$  con 4490, mientras que el mejor Tardiness es para  $\tau = 2,3$  con valor 103.

Semilla 6				Semilla 7			
$\tau = 1.0$		$\tau = 1.4$		$\tau = 1.0$		$\tau = 2.3$	
4542	2838	4586	2780	4490	3124	4688	3043
4559	2821	4587	2780	4490	3122	4699	3018
4564	2749	4589	2780	4490	3115	4711	2513
4568	2590	4592	2733	4490	3126	4713	2478
4569	2539	4596	2700	4491	3115	4717	2478
4575	2196	4596	2330	4492	3110	4721	2123
4575	2539	4605	2207	4493	3110	4725	2026
4584	2173	4608	2190	4500	3107	4732	2026
4606	2168	4616	2147	4515	2856	4733	1985
4606	2165	4617	2142	4516	2856	4733	2026
4606	2166	4629	2109	4517	2856	4751	1760
4608	2165	4629	2118	4518	2855	4754	922
4622	2150	4631	2108	4520	2853	4764	922
4623	1363	4632	2108	4520	2847	4766	922
4638	1262	4643	2093	4520	2855	4770	922
4638	1363	4643	2088	4527	2840	4780	922
4641	1262	4656	2069	4530	2829	4783	922
4648	1259	4656	2083	4532	2818	4786	922
4656	1259	4656	2070	4532	2821	4793	763

4661	1259	4676	2048	4532	2825	4797	742
4669	1259	4677	2001	4537	2814	4816	734
4670	1259	4680	1994	4537	2809	4830	734
4677	1259	4680	1993	4538	2809	4831	728
4678	1254	4680	1990	4539	2804	4839	728
4691	1254	4682	1756	4539	2806	4843	682
4695	1254	4682	1754	4542	2713	4844	682
4699	957	4687	1718	4543	2713	4851	682
4711	947	4687	1679	4545	2632	4952	625
4720	945	4699	1679	4545	2640	4952	618
4736	914	4700	1666	4545	2645	4965	614
4747	914	4712	1661	4545	2637	4971	614
4749	886	4732	1543	4545	2650	4978	614
4749	883	4744	1537	4546	2604	4980	614
4751	883	4745	1514	4546	2609	4982	612
4753	883	4745	1530	4553	2318	4992	609
4761	881	4747	1500	4554	2318	4995	609
4766	626	4747	1511	4555	2318	5003	609
4766	601	4751	1495	4556	2306	5008	609
4810	563	4752	1480	4556	2311	5024	435
4825	563	4754	1472	4558	2302	5031	435
4826	556	4754	1467	4559	2301	5031	423
4827	388	4754	1480	4566	2301	5032	345
4833	388	4754	1479	4572	2301	5043	244
4840	388	4757	1466	4575	2297	5043	255
4842	388	4764	1460	4575	2191	5059	240
4843	372	4773	1454	4575	2196	5059	204
4843	352	4775	1403	4575	2238	5064	184
4843	371	4784	1002	4575	2207	5065	184
4843	341	4784	1116	4577	2183	5084	160
4850	341	4784	1077	4577	2191	5085	138
4854	279	4785	963	4579	2077	5085	155
4859	279	4785	1002	4579	2073	5085	160
4862	279	4817	901	4585	2073	5086	138
4870	259	4833	901	4598	1918	5087	136
4870	270	4862	885	4603	1903	5089	136
4875	259	4877	880	4605	1834	5094	108
4880	259	4878	866	4607	1834	5094	109
4883	259	4880	866	4608	1833	5094	103
4884	259	4888	858	4610	1833	5104	103
4888	259	4898	616	4631	1810	5145	103
4896	259	4913	585	4632	1704	5161	103
4908	259	4921	422	4635	1637		
		4937	415	4636	1637		
		4942	314	4645	1611		
		4942	336	4645	1637		
		4952	289	4647	1611		
		4956	275	4648	1611		
		4970	252	4649	1611		
		4974	233	4651	1594		
		4995	229	4653	1592		
		4995	232	4659	1592		
		4995	231	4660	1592		
		4999	186	4661	1592		
		5003	180	4681	1552		
		5003	169	4681	1516		
		5008	135	4686	1510		
		5008	162	4686	1506		
		5012	77	4686	1478		
		5022	68	4686	1454		
		5032	68	4691	1441		
		5039	68	4696	1441		
		5042	68	4699	1424		



5059	68	4699	1437
5069	68	4699	1428
5071	68	4699	1441
5073	68	4706	1423
5077	68	4708	1423
5078	68	4709	1414
5080	68	4709	1413
		4754	750
		4754	742
		4767	735
		4767	739
		4785	735
		4787	705
		4787	711
		4789	511
		4789	495
		4789	453
		4789	455
		4792	439
		4792	434
		4792	426
		4792	438
		4792	443
		4794	407
		4794	422
		4801	407
		4802	392
		4802	388
		4802	407
		4804	385
		4809	379
		4811	373
		4844	259
		4845	259
		4848	259
		4853	259
		4861	259
		4862	259
		4868	259
		4869	259
		4884	259
		4886	246
		4889	246
		4891	239
		4899	239
		4902	232
		4904	232
		4920	232
		4930	232
		4941	232

Tabla 5.14 Mejores  $\tau$  para semillas 6 y 7 en EO\_Puro\_Busqueda 60-20

### 5.3.2.4 Análisis 80 Jobs 20 Máquinas

#### 5.3.2.4.1 Variar las iteraciones.

En la Figura 5.34 se grafican los resultados obtenidos en el algoritmo de EO\_Puro\_Busqueda con los datos del *benchmark* 80 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que los mejores resultados se logran con 1.000.000 de iteraciones.

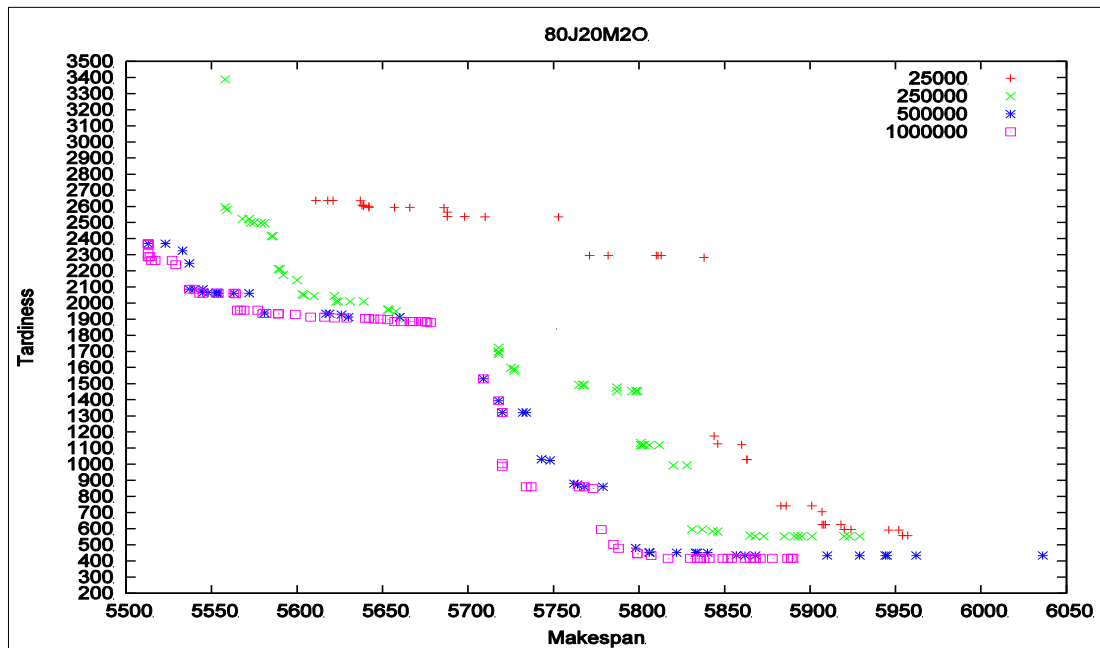


Figura 5.34 Variación de iteraciones para EO\_Puro\_Busqueda 80-20 con semilla 0 y  $\tau = 1,4$

### 5.3.2.4.2 Variar el valor de Semillas.

En la Figura 5.35 se grafican los resultados obtenidos en el algoritmo de EO\_Puro\_Busqueda con los datos del *benchmark* 80 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness son muy parecidos para la mayor cantidad de semillas, mientras que en el Makespan los valores se dispersan bastante.

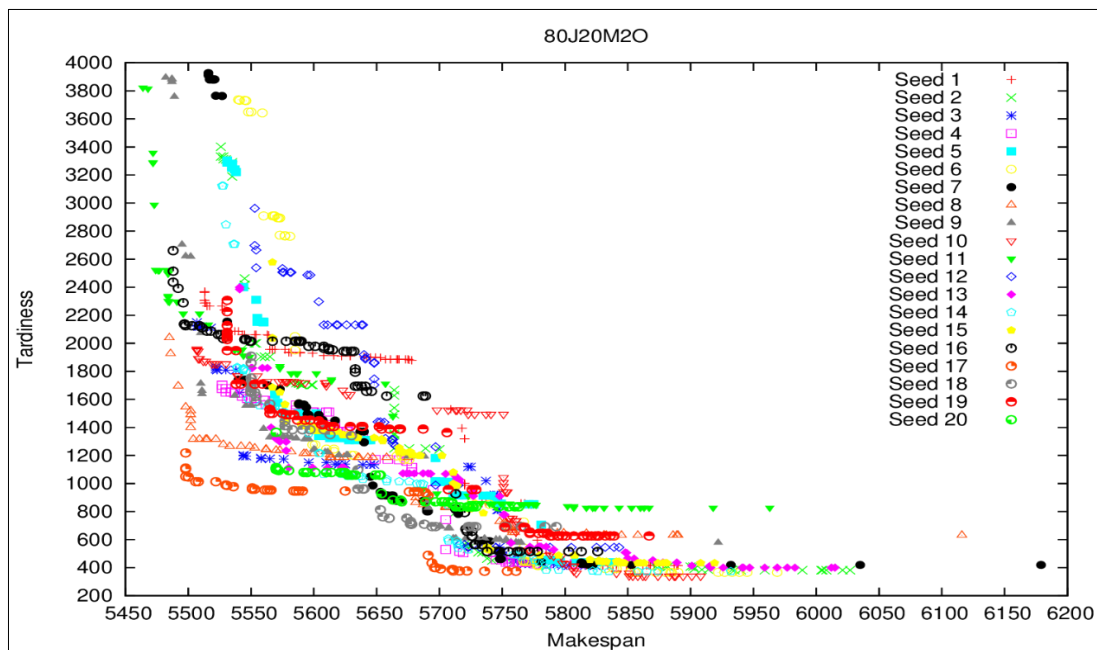


Figura 5.35 Variación de Semillas para EO\_Puro\_Busqueda 80-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos el mejor resultado en Makespan lo obtiene la semilla 11 con valor 5464, mientras que en Tardiness la mejor semilla es la 10 con valor 340:

Semilla 11		Semilla 10	
Makespan	Tardiness	Makespan	Tardiness
<b>5464</b>	3823	5507	1959
5468	3816	5507	1949
5472	3287	5507	1954
5472	3290	5508	1891
5472	3359	5508	1945
5473	2988	5509	1891
5474	2523	5512	1873
5476	2521	5514	1873
5477	2521	5519	1854
5483	2521	5521	1854
5484	2337	5523	1854
5484	2500	5530	1854
5484	2521	5539	1769
5484	2301	5540	1769
5484	2520	5555	1768
5484	2497	5563	1729
5484	2330	5572	1729
5485	2297	5575	1729
5491	2297	5577	1729
5496	2212	5578	1729
5509	2212	5579	1729
5516	2133	5584	1729
5537	1955	5587	1719
5537	1958	5593	1719
5540	1955	5609	1719
5543	1955	5610	1698
5544	1913	5610	1719
5549	1904	5623	1666
5573	1823	5626	1634
5573	1837	5629	1634
5582	1785	5698	1525
5586	1785	5707	1525
5587	1785	5713	1525
5602	1785	5714	1525
5614	1733	5717	1525
5614	1742	5722	1525
5657	1712	5723	1495
5663	1481	5728	1495
5663	1482	5737	1495
5686	951	5741	1495
5686	956	5751	1495
5686	960	5751	980
5698	930	5751	1008
5723	912	5751	1043
5726	861	5751	940
5733	861	5753	940
5740	857	5754	940
5742	857	5755	752
5749	857	5756	752
5756	857	5758	752
5765	852	5761	732
5767	852	5761	736
5771	852	5761	655
5775	852	5761	711
5777	849	5762	524
5802	839	5771	524

5807	828	5772	524
5807	838	5777	524
5817	828	5788	494
5822	828	5791	483
5824	828	5797	483
5831	828	5798	432
5843	828	5801	427
5853	828	5807	413
5860	828	5807	377
5863	828	5807	408
5867	828	5809	361
5918	828	5841	361
5963	828	5851	361
		5851	342
		5854	<b><u>340</u></b>
		5858	<b><u>340</u></b>
		5862	<b><u>340</u></b>
		5864	<b><u>340</u></b>
		5874	<b><u>340</u></b>
		5883	<b><u>340</u></b>
		5887	<b><u>340</u></b>
		5888	<b><u>340</u></b>
		5894	<b><u>340</u></b>
		5901	<b><u>340</u></b>
		5908	<b><u>340</u></b>

Tabla 5.15 *Resultados para semillas 11 y 10 en las pruebas de variación de semillas con EO\_Puro\_Busqueda 80-20*

### 5.3.2.4.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.36 se muestran los resultados de la variación de  $\tau$  para las semillas 10 y 11, luego en los gráficos de las Figuras 5.37 y 5.38 están los resultados individuales de la semilla 10 que obtuvo el mejor Tardiness y la semilla 11 con el mejor Makespan respectivamente.

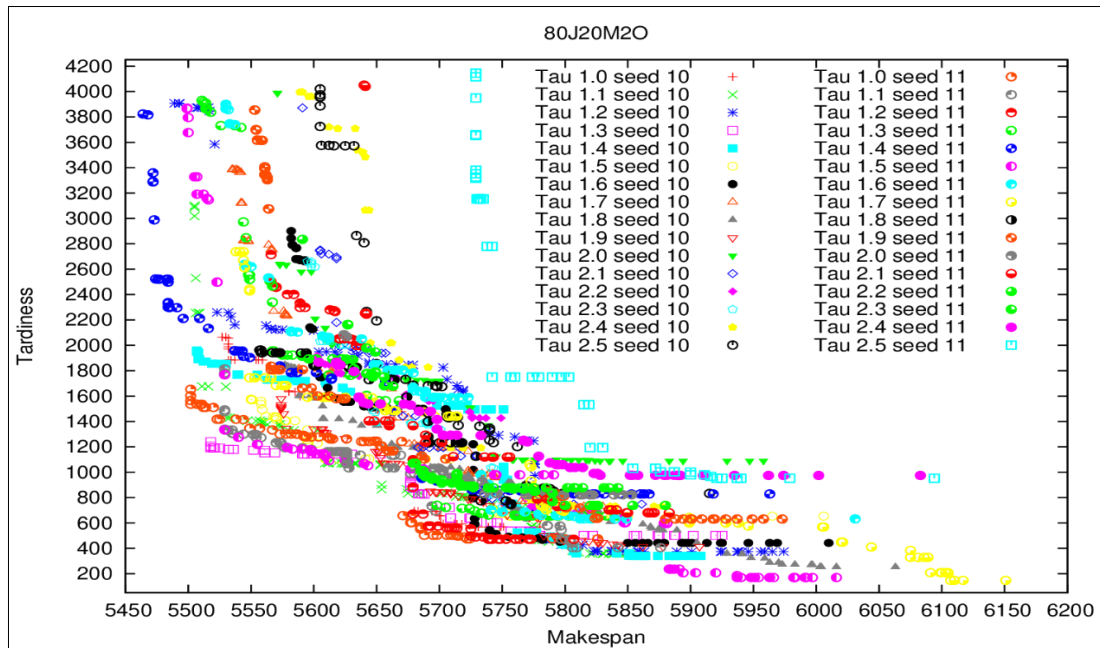


Figura 5.36 Variación de  $\tau$  para EO\_Puro\_Busqueda 80-20 con semillas 10 y 11 y 1.000.000 de iteraciones

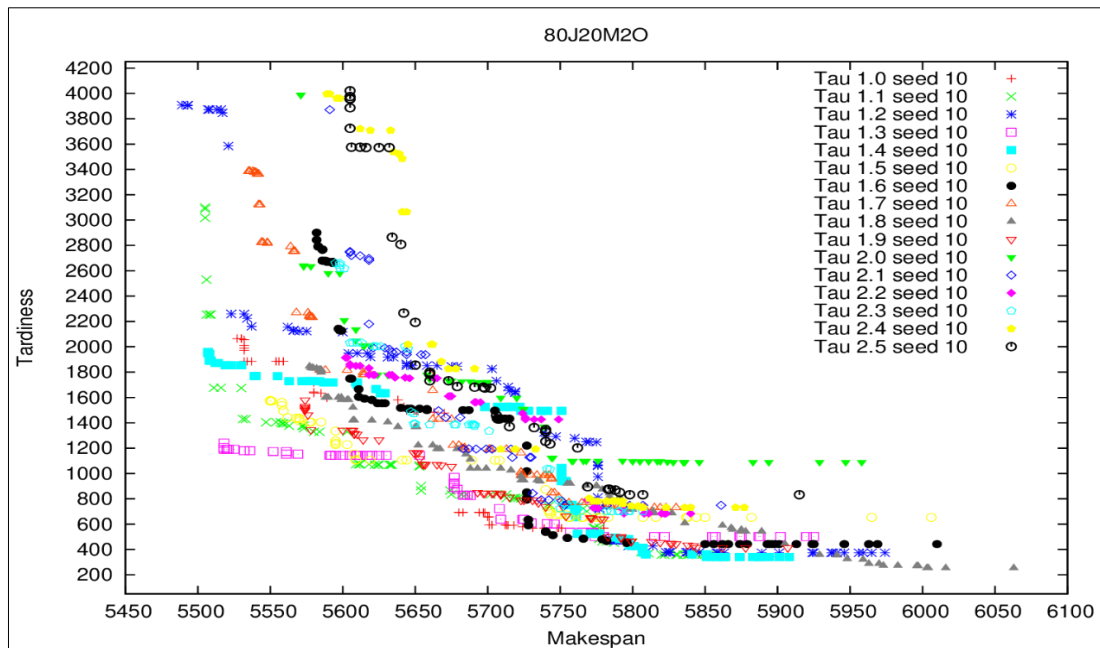


Figura 5.37 Variación de  $\tau$  para EO\_Puro\_Busqueda 80-20 con semilla 10, mejor Tardiness, y 1.000.000 de iteraciones

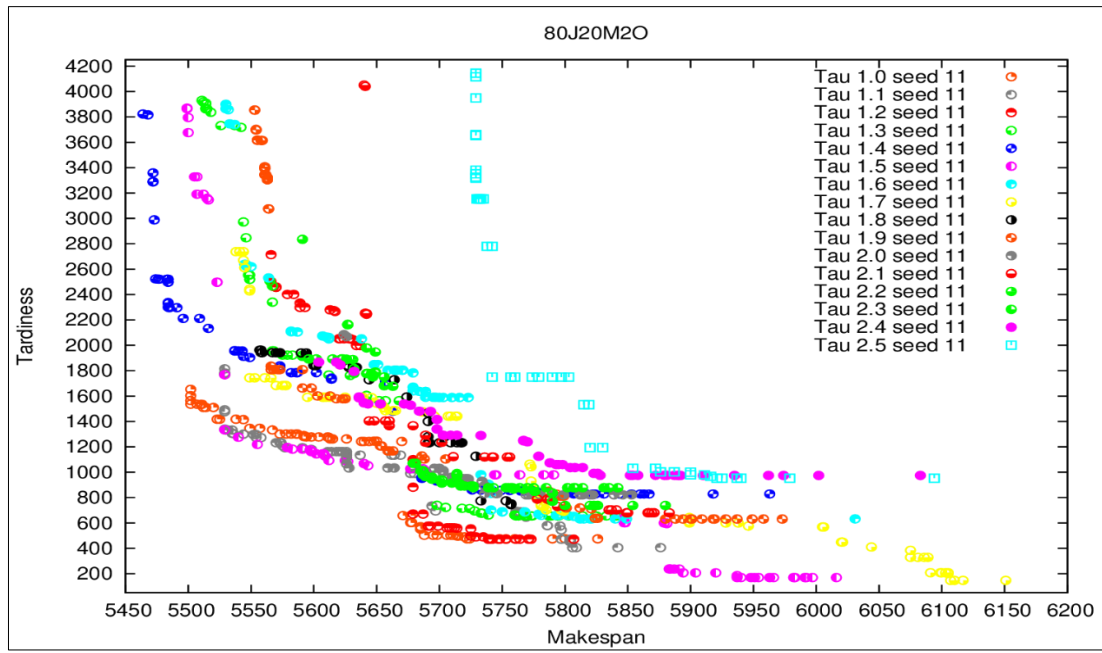


Figura 5.38 Variación de  $\tau$  para EO\_Puro\_Busqueda 80-20 con semilla 11, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 10:** Mejor Makespan para  $\tau = 1,2$  con 5489, mientras que el Tardiness es para  $\tau = 1,8$  con valor 253.
- ✓ **Semilla 11:** Mejor Makespan para  $\tau = 1,4$  con 5464, mientras que el mejor Tardiness es para  $\tau = 1,7$  con valor 148

Semilla 10				Semilla 11			
$\tau = 1,2$		$\tau = 1,8$		$\tau = 1,4$		$\tau = 1,7$	
5489	3908	5577	1845	5464	3823	5538	2738
5493	3904	5578	1833	5468	3816	5541	2738
5493	3908	5579	1831	5472	3287	5544	2670
5507	3872	5579	1833	5472	3290	5544	2738
5508	3872	5579	1832	5472	3359	5545	2612
5513	3872	5580	1828	5473	2988	5549	1744
5516	3872	5582	1828	5474	2523	5549	2428
5517	3846	5584	1819	5476	2521	5549	2443
5521	3584	5585	1819	5477	2521	5553	1744
5523	2259	5585	1797	5483	2521	5561	1744
5532	2259	5585	1810	5484	2337	5566	1744
5534	2224	5588	1600	5484	2500	5570	1683
5537	2161	5590	1600	5484	2521	5575	1683
5562	2157	5597	1600	5484	2301	5577	1683
5566	2131	5599	1600	5484	2520	5578	1683
5566	2127	5599	1582	5484	2497	5595	1591
5570	2123	5603	1582	5484	2330	5607	1591
5575	2123	5604	1582	5485	2297	5608	1591
5600	2115	5607	1419	5491	2297	5610	1591
5604	1949	5607	1520	5496	2212	5613	1591
5610	1949	5608	1419	5509	2212	5625	1591
5617	1949	5620	1418	5516	2133	5637	1591
5619	1920	5631	1404	5537	1955	5639	1591
5631	1917	5638	1367	5537	1958	5646	1588
5635	1917	5650	1367	5540	1955	5650	1543
5644	1865	5651	1367	5543	1955	5654	1531
5644	1851	5652	1222	5544	1913	5656	1510
5648	1851	5662	1222	5549	1904	5656	1505
5658	1851	5665	1194	5573	1823	5657	1485
5665	1851	5667	1194	5573	1837	5663	1485
5675	1845	5679	1194	5582	1785	5665	1485
5679	1845	5685	1151	5586	1785	5686	1476
5703	1825	5688	1110	5587	1785	5687	1476
5706	1731	5688	1039	5602	1785	5696	1442
5714	1683	5693	1039	5614	1733	5707	1442
5715	1654	5699	1039	5614	1742	5709	1442
5719	1637	5709	1039	5657	1712	5713	1442
5719	1647	5716	1025	5663	1481	5714	1442
5739	1324	5723	945	5663	1482	5772	1063
5740	1292	5723	977	5686	951	5773	1043
5747	1292	5725	945	5686	956	5773	930
5760	1278	5734	945	5686	960	5774	822
5768	1250	5736	938	5698	930	5774	870
5770	1250	5755	920	5723	912	5776	813
5775	1247	5757	920	5726	861	5781	749
5776	811	5768	903	5733	861	5781	736
5776	1067	5775	903	5740	857	5781	791
5776	974	5775	900	5742	857	5783	727



5776	1061	5779	873	5749	857	5785	704
5777	727	5784	844	5756	857	5797	704
5777	719	5784	818	5765	852	5797	700
5780	496	5791	815	5767	852	5799	691
5785	472	5792	774	5771	852	5899	612
5787	472	5796	714	5775	852	5899	600
5788	472	5800	714	5777	849	5899	643
5789	472	5814	704	5802	839	5928	600
5797	428	5814	687	5807	828	5938	600
5814	428	5814	691	5807	838	5946	576
5823	378	5823	687	5817	828	6005	569
5825	378	5827	687	5822	828	6006	569
5826	378	5836	608	5824	828	6020	449
5837	378	5862	589	5831	828	6021	449
5838	378	5868	570	5843	828	6044	410
5840	378	5869	556	5853	828	6075	384
5846	378	5875	556	5860	828	6075	329
5858	374	5875	542	5863	828	6081	329
5859	374	5878	542	5867	828	6086	329
5863	374	5889	542	5918	828	6089	329
5889	374	5905	444	5963	828	6091	208
5901	374	5929	358			6099	208
5924	374	5941	354			6100	208
5925	374	5948	325			6103	208
5937	374	5959	320			6105	208
5938	374	5963	288			6105	202
5941	374	5970	288			6107	148
5946	374	5973	273			6110	148
5956	374	5983	273			6117	148
5959	374	5996	270			6151	148
5965	374	5997	270				
5974	374	6003	253				
		6004	253				
		6016	253				
		6063	253				

Tabla 5.16 Mejores  $\tau$  para semillas 10 y 11 en EO\_Puro\_Busqueda 80-20

---

### 5.3.3 EO\_Feromonas

#### 5.3.3.1 Análisis 20 Jobs 20 Máquinas

##### 5.3.3.1.1 Variar las iteraciones.

En la Figura 5.39 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas con los datos del *benchmark* 20 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que los mejores resultados se logran con 1.000.000 de iteraciones.

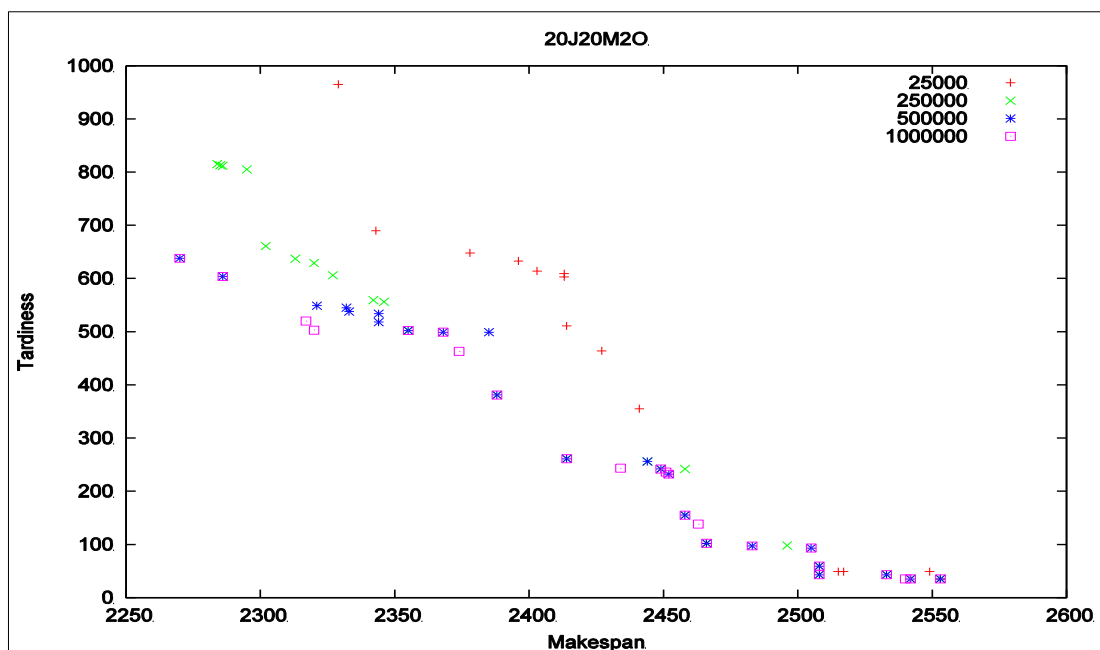


Figura 5.39 Variación de iteraciones para EO\_Feromonas 20-20 con semilla 0 y  $\tau = 1,4$

### 5.3.3.1.2 Variar el valor de Semillas.

En la Figura 5.40 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas con los datos del *benchmark* 20 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness son muy parecidos para la mayor cantidad de semillas, mientras que en el Makespan los valores se dispersan.

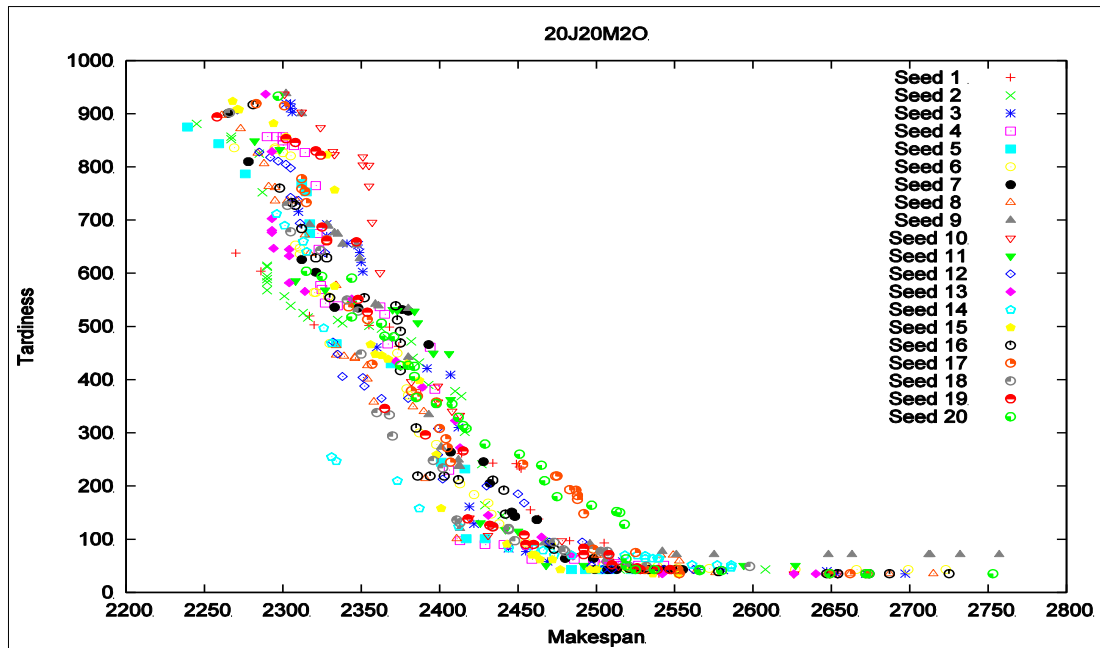


Figura 5.40 Variación de Semillas para EO\_Feromonas 20-20 con 1000000 de iteraciones y  $\tau = 1,4$

El mejor resultado en Makespan lo obtiene la semilla 5 con 2239, mientras que en Tardiness es de las semillas 1, 3, 8, 10, 13, 15, 16, 17 y 20 con valor 35. Se representará aquella que tiene la mejor solución, es decir entre los valores mínimos de Tardiness el Makespan correspondiente debe ser el menor, esta condición la cumple sólo la semilla 15 con valores de solución (2536, 35):

Semilla 5		Semilla 15	
Makespan	Tardiness	Makespan	Tardiness
<b><u>2239</u></b>	875	2268	924
2259	844	2271	909
2276	787	2272	908
2312	769	2294	882
2315	753	2302	857
2317	693	2328	823
2317	675	2333	757
2334	468	2333	576
2369	430	2356	466
2401	245	2359	448
2416	232	2363	446
2417	101	2367	439
2429	101	2379	430
2470	89	2387	399
2484	43	2398	259
2493	43	2401	158
2501	43	2443	91
2504	43	2459	71
2506	43	2461	71
2510	43	2464	62
2522	43	2472	62
2525	43	2477	43
		2496	43
		2500	43
		2521	43
		<b><u>2536</u></b>	<b><u>35</u></b>
		2552	<b><u>35</u></b>

Tabla 5.17 Resultados para semilla 5 y 15 en las pruebas de variación de semillas con EO\_Feromonas 20-20

### 5.3.3.1.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.41 se muestran los resultados de la variación de  $\tau$  para las semillas 5 y 15, luego en los gráficos de las Figuras 5.42 y 5.43 están los resultados individuales de la semilla 15 que obtuvo el mejor Tardiness y la semilla 5 con el mejor Makespan respectivamente.

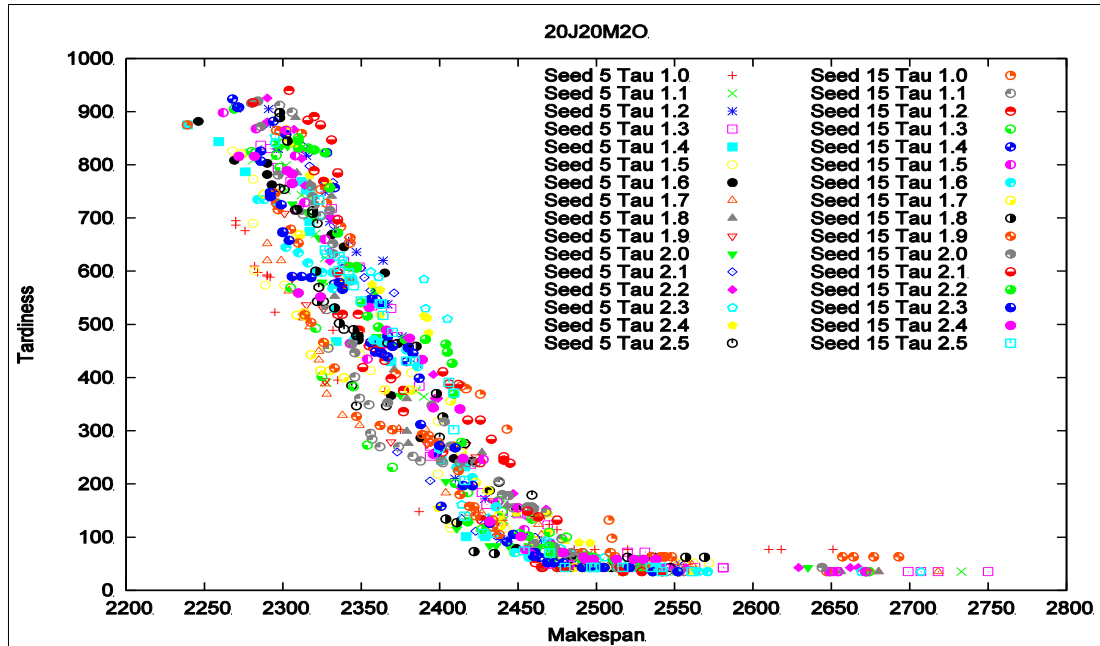


Figura 5.41 Variación de  $\tau$  para EO\_Feromonas 20-20 con semillas 5 y 15 y 1.000.000 de iteraciones

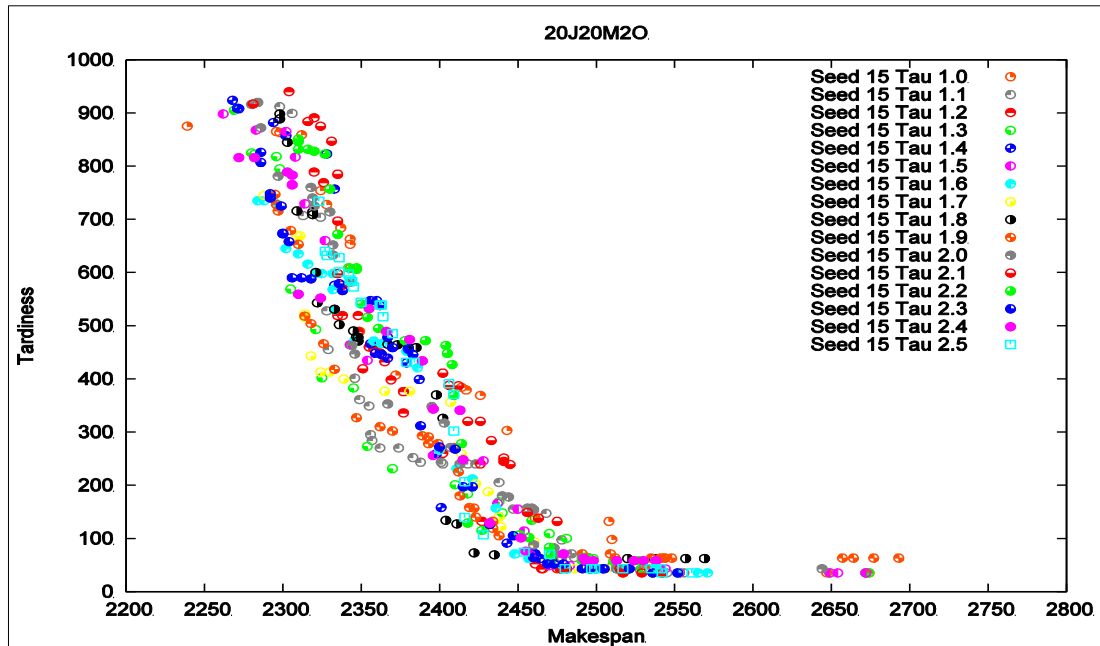


Figura 5.42 Variación de  $\tau$  para EO\_Feromonas 20-20 con semilla 15, mejor Tardiness, y 1.000.000 de iteraciones

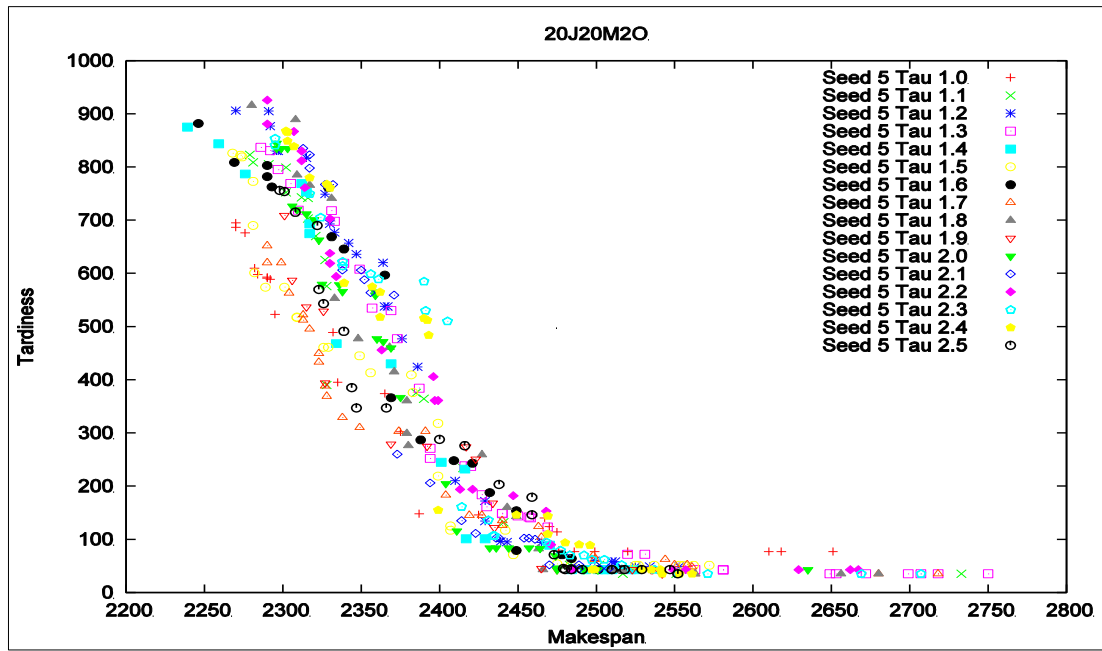


Figura 5.43 Variación de  $\tau$  para EO\_ Feromonas 20-20 con semilla 5, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 5:** Mejor Makespan para  $\tau = 1,4$  con 2239, mientras que el Tardiness es para  $\tau = 1,1; 1,3; 1,7; 1,8; 1,9; 2,3; 2,4$  y  $2,5$  con valor 35, de ellos se escogerá al que tiene la mejor solución (35, 2517) para  $\tau = 1,1$ .
- ✓ **Semilla 15:** Mejor Makespan para  $\tau = 1,0$  con 2239, mientras que el mejor Tardiness es para  $\tau = 1,0; 1,1; 1,3; 1,4; 1,5; 1,6; 1,7; 2,1; 2,3$  y  $2,5$  con valor 35, como segundo criterio los que tienen la mejor solución son  $\tau = 1,7$  y  $2,1$  y de estos el que tiene el mejor Makespan  $\tau = 2,1$  con 2281.

Semilla 5				Semilla 15			
$\tau = 1,4$		$\tau = 1,1$		$\tau = 1,0$		$\tau = 2,1$	
2239	875	2279	823	2239	875	2281	917
2259	844	2281	809	2296	865	2316	884
2276	787	2291	805	2298	863	2324	875
2312	769	2302	799	2312	859	2331	847
2315	753	2302	752	2324	754	2335	598
2317	693	2312	743	2328	728	2335	697
2317	675	2316	742	2337	684	2335	785
2334	468	2316	703	2343	653	2338	576
2369	430	2321	670	2343	608	2351	419
2401	245	2327	625	2343	663	2402	411
2416	232	2328	576	2349	489	2406	388
2417	101	2328	390	2372	407	2412	387
2429	101	2385	375	2414	384	2418	320
2470	89	2390	364	2417	379	2426	320
2484	43	2415	234	2426	369	2433	284
2493	43	2441	132	2443	303	2441	251
2501	43	2483	52	2461	138	2441	245
2504	43	2502	52	2508	132	2445	239
2506	43	2509	43	2510	98	2456	149
2510	43	2517	35	2510	47	2463	138
2522	43	2551	35	2518	43	2475	132
2525	43	2733	35	2532	43	2477	43
				2553	35	2481	43
				2647	35	2515	43
				2672	35	2517	35
				2674	35	2529	35
						2542	35

Tabla 5.18 Mejores  $\tau$  para semillas 5 y 15 en EO\_Feromonas 20-20

### 5.3.3.2 Análisis 40 Jobs 20 Máquinas

#### 5.3.3.2.1 Variar las iteraciones.

En la Figura 5.44 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas con los datos del *benchmark* 40 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que los mejores resultados se logran con 1.000.000 de iteraciones.

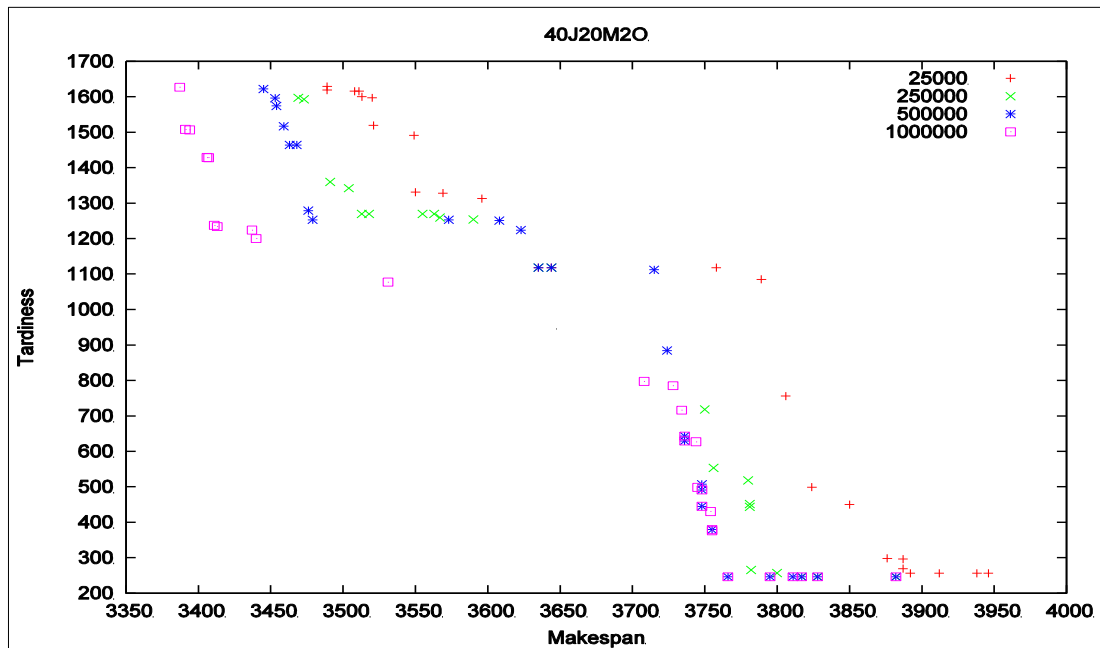


Figura 5.44 Variación de iteraciones para EO\_Feromonas 40-20 con semilla 0 y  $\tau = 1,4$



### 5.3.3.2.2 Variar el valor de Semillas.

En la Figura 5.45 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas con los datos del *benchmark* 40 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness y Makespan están dispersos para las distintas semillas.

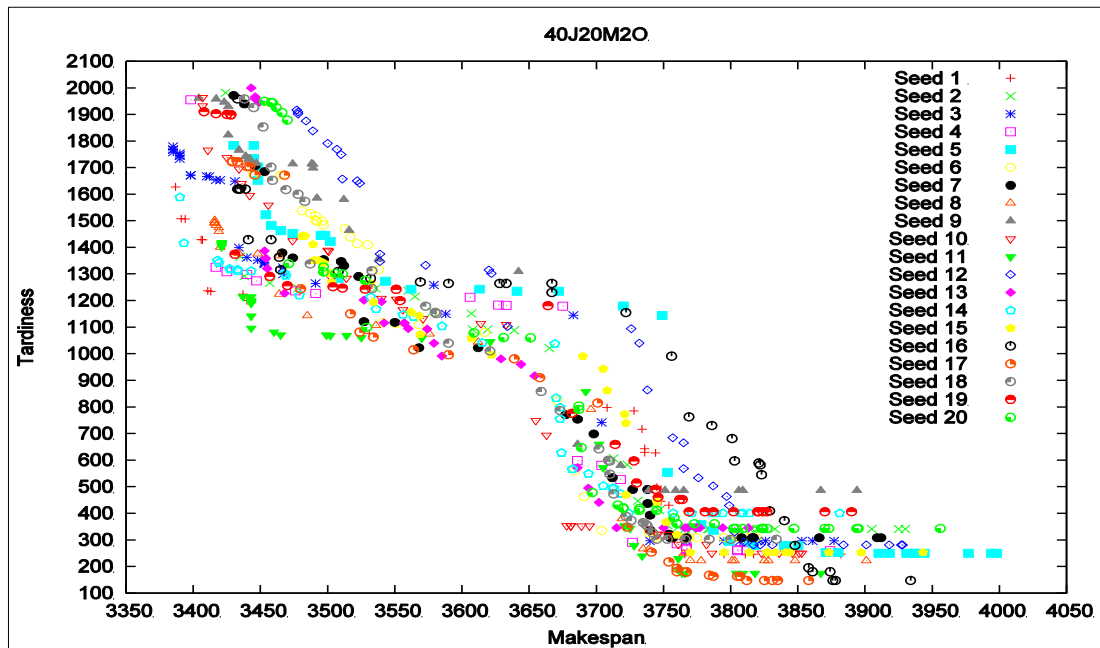


Figura 5.45 Variación de Semillas para EO\_Feromonas 40-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos el mejor resultado en Makespan lo obtiene la semilla 3 con valor 3385, mientras que en Tardiness es de la semilla 16 y 17 ambos con 147. Se representará el de mejor solución que corresponde a la semilla 17 con (3812, 147):

Semilla 3		Semilla 17	
Makespan	Tardiness	Makespan	Tardiness
<b>3385</b>	1779	3429	1723
<b>3385</b>	1768	3429	1724
<b>3385</b>	1758	3429	1722
<b>3385</b>	1767	3433	1722
3390	1746	3440	1704
3390	1753	3442	1703
3390	1745	3446	1673
3390	1733	3468	1672
3398	1672	3480	1244
3398	1671	3517	1150
3410	1667	3524	1081
3412	1666	3534	1063
3417	1654	3564	1015
3420	1653	3590	996
3431	1649	3639	981
3434	1399	3658	910
3440	1362	3701	815
3448	1352	3723	348
3453	1343	3741	254
3453	1341	3754	217
3453	1337	3760	193
3467	1299	3760	180
3467	1324	3767	180
3491	1264	3767	178
3579	1258	3784	167
3588	1149	3787	163
3683	1145	3805	163
3704	741	3807	163
3740	296	<b>3812</b>	<b>147</b>
3767	296	3825	<b>147</b>
3795	296	3831	<b>147</b>
3807	296	3835	<b>147</b>
3820	296	3858	<b>147</b>
3826	296		
3853	296		
3861	296		
3877	296		

Tabla 5.19 Resultados para semilla 3 y 17 en las pruebas de variación de semillas con EO\_Feromonas 40-20

### 5.3.3.2.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.46 se muestran los resultados de la variación de  $\tau$  para las semillas 3 y 17, luego en los gráficos de las Figuras 5.47 y 5.48 están los resultados individuales de la semilla 17 que obtuvo el mejor Tardiness y la semilla 3 con el mejor Makespan respectivamente.

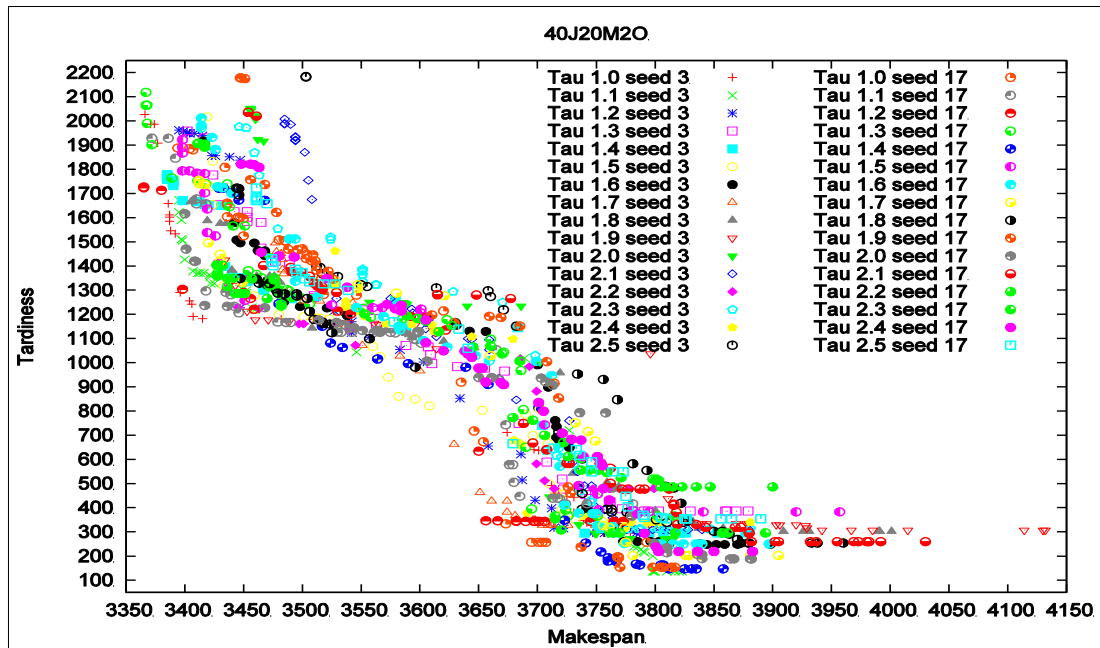


Figura 5.46 Variación de  $\tau$  para EO\_ Feromonas 40-20 con semillas 3 y 17 y 1.000.000 de iteraciones

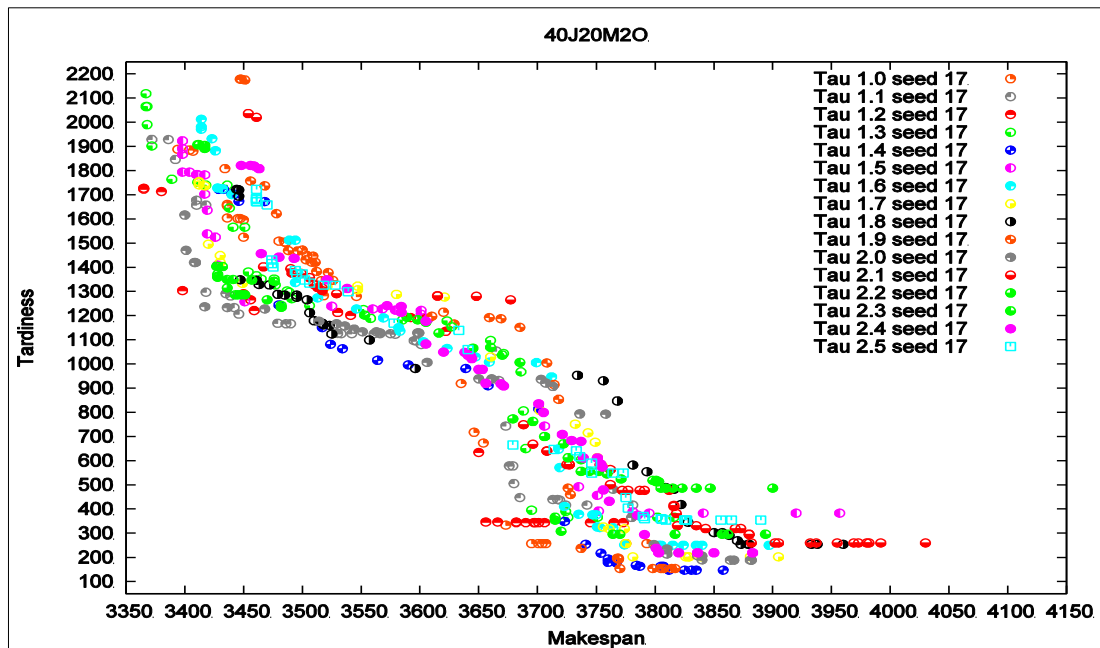


Figura 5.47 Variación de  $\tau$  para EO\_ Feromonas 40-20 con semilla 17, mejor Tardiness, y 1.000.000 de iteraciones

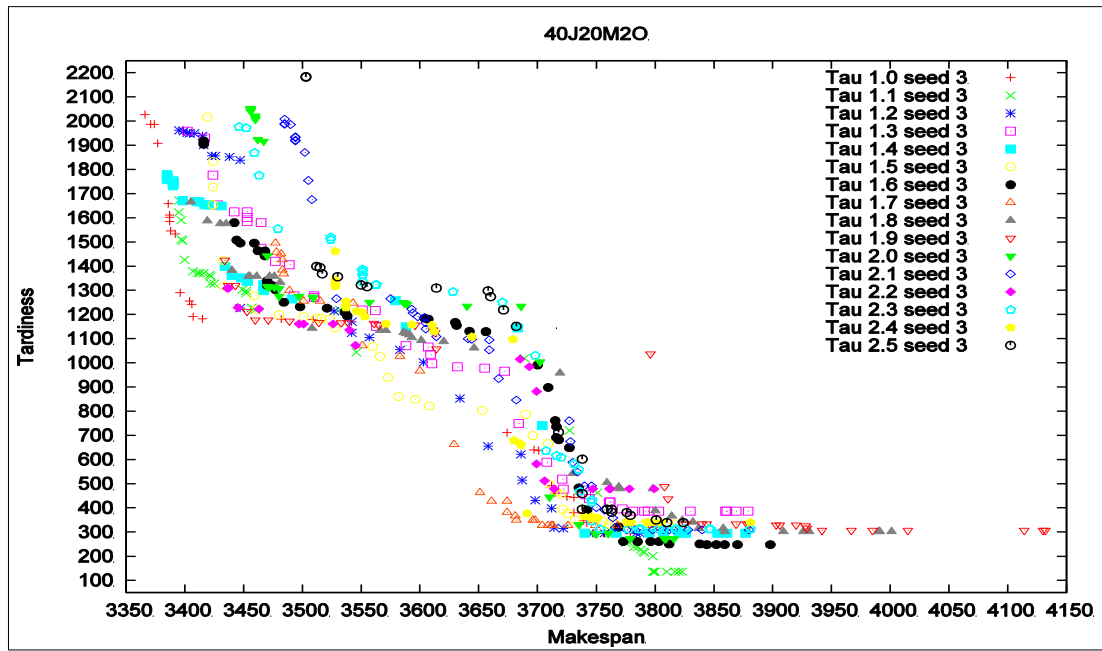


Figura 5.48 Variación de  $\tau$  para EO\_ Feromonas 40-20 con semilla 3, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 3:** Mejor Makespan para  $\tau = 1,0$  con 3366, mientras que el Tardiness es para  $\tau = 1,1$  con valor 136.
- ✓ **Semilla 17:** Mejor Makespan para  $\tau = 1,2$  con 3365, mientras que el mejor Tardiness es para  $\tau = 1,4$  con valor 147.

Semilla 3				Semilla 17			
$\tau = 1,0$		$\tau = 1,1$		$\tau = 1,2$		$\tau = 1,4$	
3366	2027	3395	1672	3365	1727	3429	1723
3371	1987	3395	1622	3365	1723	3429	1724
3374	1987	3397	1511	3380	1713	3429	1722
3377	1908	3397	1590	3398	1303	3433	1722
3385	1780	3398	1505	3450	1291	3440	1704
3386	1658	3400	1426	3456	1265	3442	1703
3387	1612	3407	1378	3459	1221	3446	1673
3387	1585	3411	1372	3530	1213	3468	1672
3387	1602	3413	1372	3541	1201	3480	1244
3388	1546	3416	1371	3592	1192	3517	1150
3392	1533	3421	1363	3603	1189	3524	1081
3396	1290	3422	1328	3622	1135	3534	1063
3404	1255	3422	1357	3650	634	3564	1015
3406	1242	3425	1328	3656	346	3590	996
3407	1191	3449	1303	3666	346	3639	981
3415	1182	3452	1298	3682	346	3658	910
3482	1181	3452	1288	3682	343	3701	815
3509	1179	3457	1228	3690	343	3723	348
3674	711	3460	1220	3697	343	3741	254
3697	640	3532	1220	3698	343	3754	217
3701	637	3546	1043	3701	343	3760	193
3712	492	3689	1019	3706	343	3760	180
3715	461	3727	720	3745	343	3767	180
3725	446	3736	473	3765	343	3767	178
3731	380	3751	462	3773	343	3784	167
3731	443	3761	296			3787	163
3738	379	3761	309			3805	163
3739	347	3768	296			3807	163
3742	338	3775	255			3812	147
3748	296	3782	239			3825	147
3756	296	3785	237			3831	147
3757	296	3788	227			3835	147
3779	296	3790	219			3858	147
3794	296	3791	215				
3807	296	3798	136				
3857	296	3798	200				
		3799	136				
		3800	136				
		3810	136				
		3817	136				
		3820	136				
		3823	136				

Tabla 5.20 Mejores  $\tau$  para semillas 3 y 17 en EQ\_Feromonas 40-20

### 5.3.3.3 Análisis 60 Jobs 20 Máquinas

#### 5.3.3.3.1 Variar las iteraciones.

En la Figura 5.49 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas con los datos del *benchmark* 60 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . Se observa que los mejores resultados se logran con 1.000.000 de iteraciones.

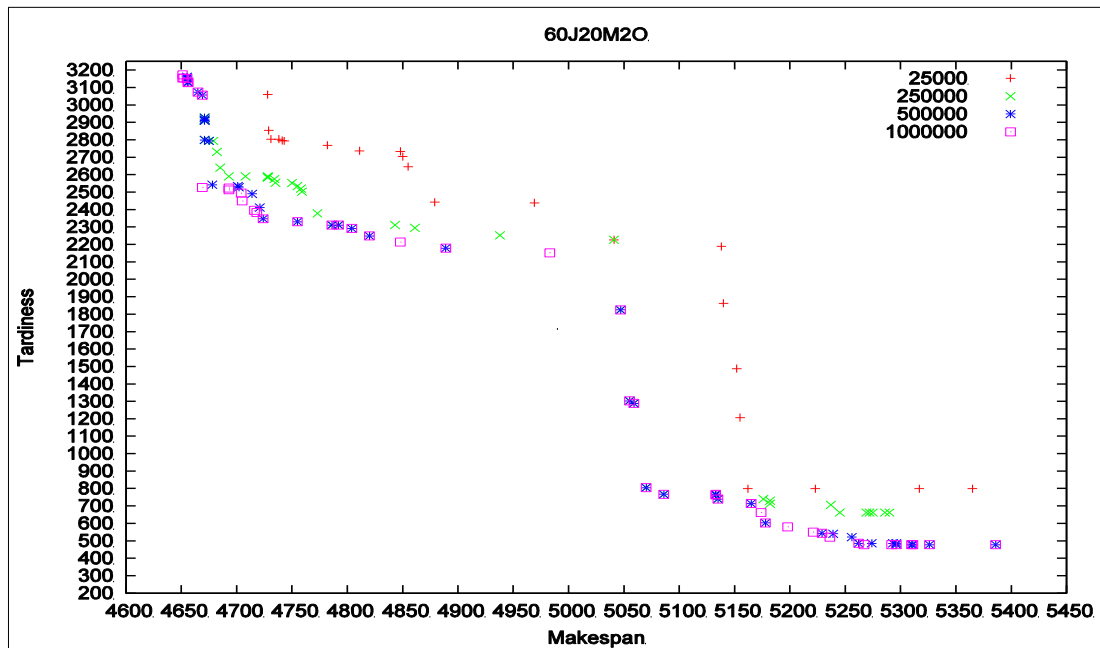


Figura 5.49 Variación de iteraciones para EO\_Feromonas 60-20 con semilla 0 y  $\tau = 1,4$

### 5.3.3.3.2 Variar el valor de Semillas.

En la Figura 5.50 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas con los datos del *benchmark* 60 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness y Makespan son dispersos, aunque en el caso de Tardiness hay una mejor congruencia de datos para las semillas analizadas.

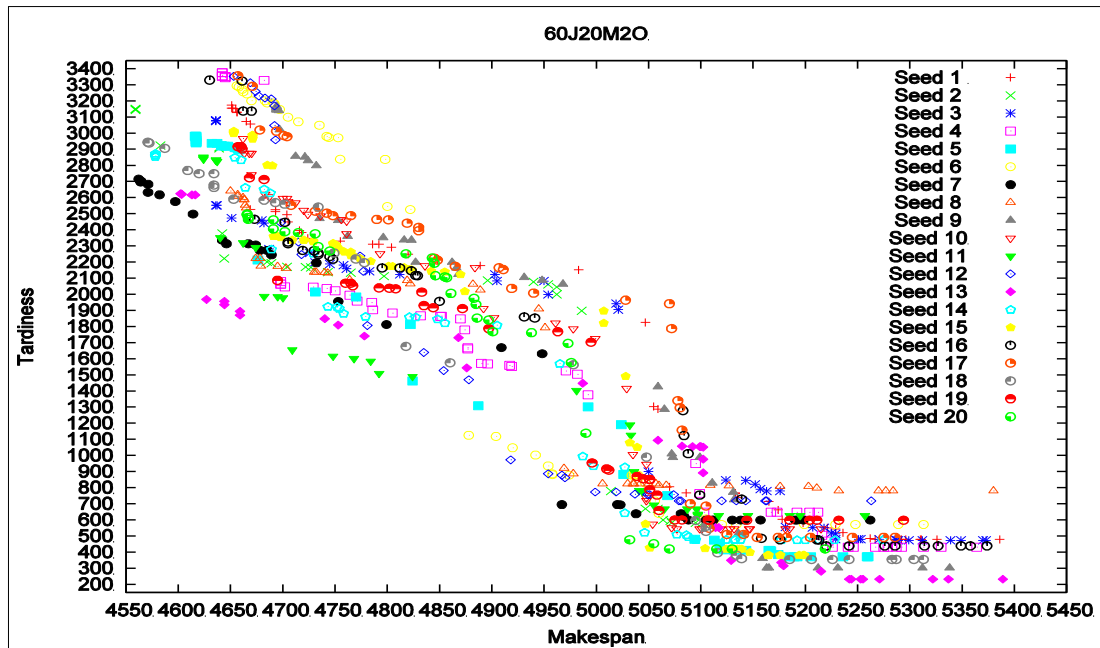


Figura 5.50 Variación de Semillas para EO\_Feromonas 60-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo al análisis de los datos El mejor resultado en Makespan lo obtiene la semilla 2 con valor 4559, mientras que en Tardiness es de la semilla 13 con valor 232:

Semilla 2		Semilla 13	
Makespan	Tardiness	Makespan	Tardiness
<b>4559</b>	3151	4602	2624
<b>4559</b>	3148	4603	2621
<b>4559</b>	3144	4603	2622
4583	2922	4613	2617
4639	2912	4616	2617
4639	2905	4627	1968
4642	2377	4644	1956
4644	2222	4644	1936
4685	2201	4659	1871
4689	2193	4659	1894
4704	2171	4740	1849
4722	2169	4753	1810
4727	2166	4778	1741
4743	2137	4868	1731
4765	2134	4876	1544
4797	2112	4987	1448
4896	2085	5059	1094
4940	2077	5082	1058
4954	2061	5092	1055
4962	1999	5099	1054
4962	2045	5100	1054
4986	1898	5102	892
5014	779	5102	1052
5047	669	5102	977
5059	668	5114	553
5064	599	5117	552
5071	595	5129	349
5072	582	5177	336
5095	582	5179	315
5102	582	5215	281
5105	582	5242	<b>232</b>
5114	582	5244	<b>232</b>
		5253	<b>232</b>
		5255	<b>232</b>
		5271	<b>232</b>
		5322	<b>232</b>
		5337	<b>232</b>
		5389	<b>232</b>

Tabla 5.21 Resultados para semillas 2 y 13 en las pruebas de variación de semillas con EO\_Feromonas 20-20



### 5.3.3.3.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.51 se muestran los resultados de la variación de  $\tau$  para las semillas 2 y 13, luego en los gráficos de las Figuras 5.52 y 5.53 están los resultados individuales de la semilla 13 que obtuvo el mejor Tardiness y la semilla 2 con el mejor Makespan respectivamente.

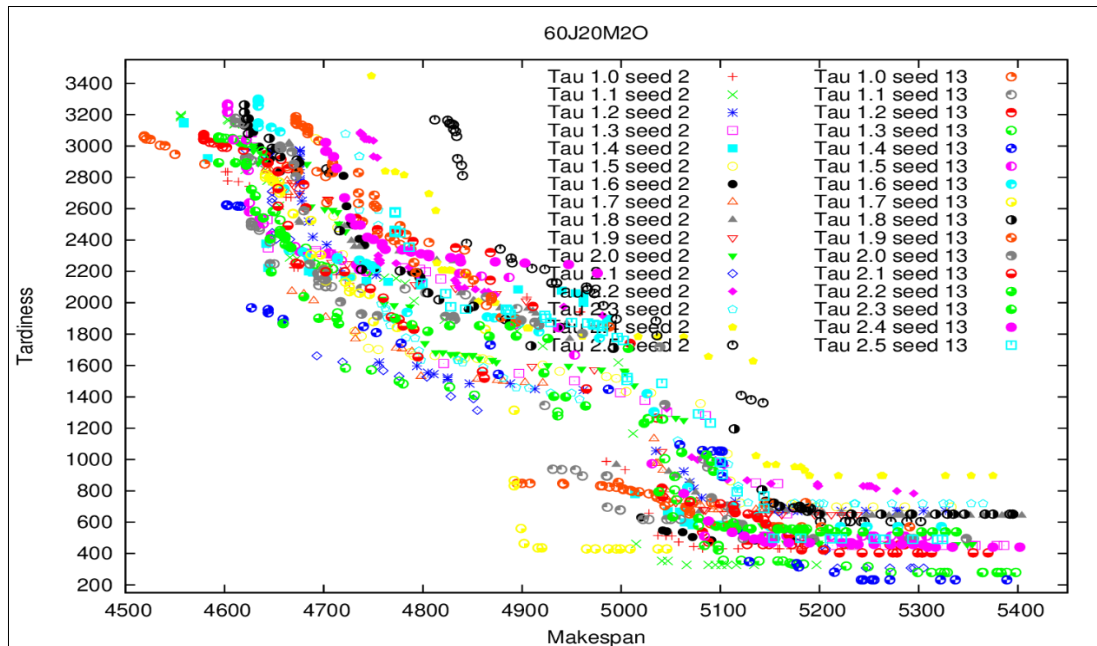


Figura 5.51 Variación de  $\tau$  para EO\_ Feromonas 60-20 con semillas 2 y 13 y 1.000.000 de iteraciones

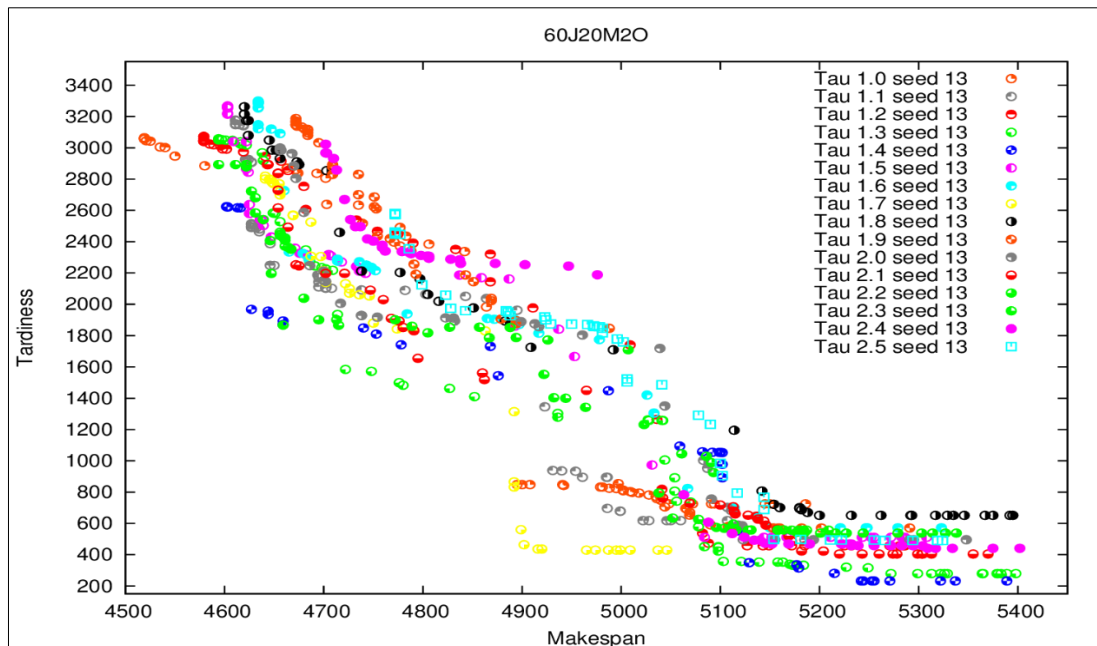


Figura 5.52 Variación de  $\tau$  para EO\_ Feromonas 60-20 con semilla 13, mejor Tardiness, y 1.000.000 de iteraciones

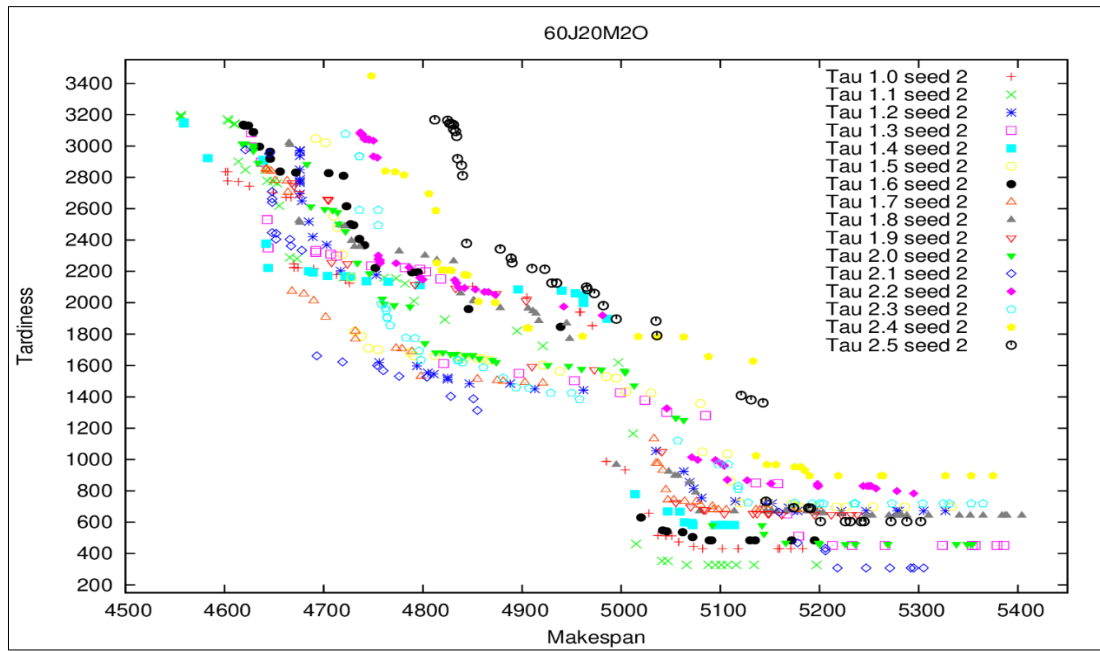


Figura 5.53 Variación de  $\tau$  para EO\_Feromonas 60-20 con semilla 2, mejor Makespan, y 1.000.000 de iteraciones

Al analizar ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 2:** Mejor Makespan para  $\tau = 1,1$  con 4556, mientras que el Tardiness es para  $\tau = 2,1$  con valor 308.
- ✓ **Semilla 13:** Mejor Makespan para  $\tau = 1,0$  con 4519, mientras que el mejor Tardiness es para  $\tau = 1,4$  con valor 232.

Semilla 2				Semilla 13			
$\tau = 1,1$		$\tau = 2,1$		$\tau = 1,0$		$\tau = 1,4$	
4556	3184	4621	2976	4519	3062	4602	2624
4556	3192	4645	2952	4519	3055	4603	2621
4556	3197	4648	2640	4519	3060	4603	2622
4603	3170	4648	2663	4519	3064	4613	2617
4604	3162	4648	2447	4521	3045	4616	2617
4610	3143	4648	2709	4525	3042	4627	1968
4610	3138	4652	2405	4535	3005	4644	1956
4614	2901	4652	2443	4540	3001	4644	1936
4621	2848	4666	2405	4550	2947	4659	1871
4641	2848	4667	2362	4580	2885	4659	1894
4643	2778	4678	2335	4663	2869	4740	1849
4649	2776	4693	1661	4672	2860	4753	1810
4653	2762	4719	1622	4673	2846	4778	1741
4655	2619	4754	1600	4674	2838	4868	1731
4666	2289	4760	1567	4693	2837	4876	1544
4673	2282	4776	1531	4702	2806	4987	1448
4720	2161	4804	1526	4703	2639	5059	1094
4761	2160	4828	1403	4735	2634	5082	1058
4773	2155	4851	1387	4739	2517	5092	1055
4782	2123	4855	1313	4777	2491	5099	1054
4791	2012	5159	665	4783	2436	5100	1054
4822	1893	5178	468	4806	2384	5102	892
4895	1820	5206	434	4842	2337	5102	1052
4921	1725	5206	417	4864	1983	5102	977
4997	1619	5218	308	4895	848	5114	553
5012	1166	5247	308	4900	848	5117	552
5015	461	5271	308	4908	848	5129	349
5041	353	5292	308	4941	848	5177	336
5047	353	5295	308	4942	842	5179	315
5066	328	5305	308	4979	832	5215	281
5088	328			4981	832	5242	232
5095	328			4988	825	5244	232
5101	328			4995	823	5253	232
5107	328			5002	813	5255	232
5116	328			5009	803	5271	232
5134	328			5011	798	5322	232
5197	328			5028	782	5337	232
				5035	772	5389	232
				5036	759		
				5040	748		
				5048	725		
				5072	725		
				5091	725		
				5094	725		
				5145	725		
				5150	725		
				5186	725		

Tabla 5.22 Mejores  $\tau$  para semillas 2 y 13 en EO\_Feromonas 60-20

### 5.3.3.4 Análisis 80 Jobs 20 Máquinas

#### 5.3.3.4.1 Variar las iteraciones.

En la Figura 5.54 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas con los datos del *benchmark* 80 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . Se aprecia que los mejores resultados se logran con 1.000.000 de iteraciones.

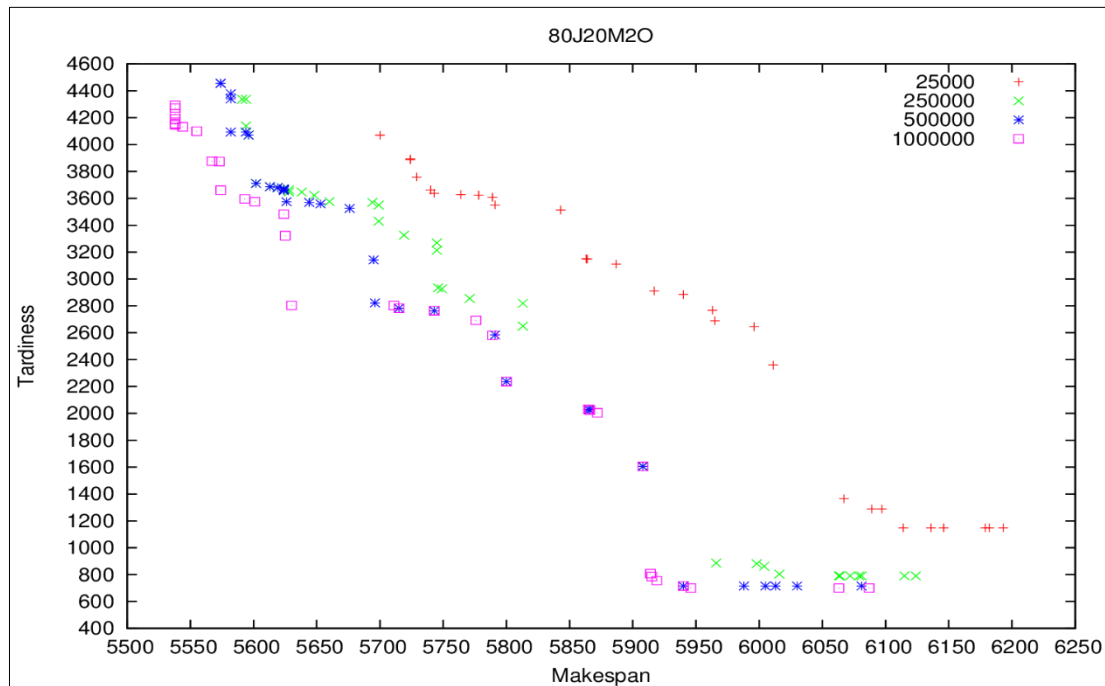


Figura 5.54 Variación de iteraciones para EO\_Feromonas 80-20 con semilla 0 y  $\tau = 1,4$

### 5.3.3.4.2 Variar el valor de Semillas.

En la Figura 5.55 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas con los datos del *benchmark* 80 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness y Makespan son dispersos para las distintas semillas.

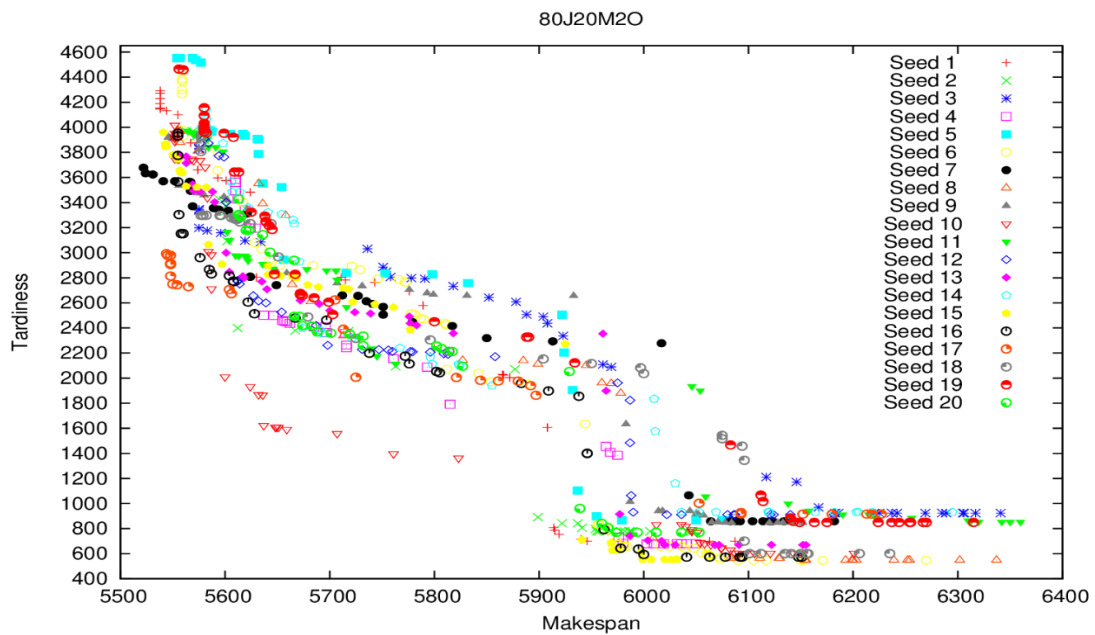


Figura 5.55 Variación de Semillas para EO\_Feromonas 80-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos El mejor resultado en Makespan lo obtiene la semilla 7 con valor 5522, mientras que en Tardiness es de la semilla 6 con valor 547:

Semilla 7		Semilla 6	
Makespan	Tardiness	Makespan	Tardiness
<b>5522</b>	3678	5559	4383
5524	3633	5559	4445
5531	3624	5559	4299
5541	3570	5559	3981
5552	3570	5559	4367
5566	3565	5559	4267
5567	3493	5560	3929
5567	3563	5593	3802
5569	3370	5593	3657
5589	3356	5617	3400
5594	3344	5628	3079
5603	3335	5635	3028
5621	3312	5643	2947
5624	2808	5652	2898

5624	2809	5684	2897
5649	2742	5721	2894
5712	2659	5729	2864
5727	2656	5746	2790
5735	2613	5766	2787
5741	2592	5772	2766
5751	2506	5772	2764
5751	2568	5778	2514
5779	2446	5781	2506
5817	2415	5787	2476
5850	2319	5790	2468
5913	2293	5810	2439
6017	2278	5811	2425
6043	1065	5812	2071
6064	859	5842	2039
6083	859	5854	1971
6088	859	5889	1953
6092	859	5944	1634
6104	859	5956	833
6114	859	5957	833
6120	859	5966	786
6123	859	5974	703
6134	859	5975	673
6182	859	5981	663
		5981	659
		5984	659
		6016	659
		6025	659
		6036	659
		6039	645
		6048	644
		6055	611
		6060	597
		6062	594
		6063	582
		6063	588
		6073	570
		6073	557
		6090	<u>547</u>
		6105	<u>547</u>
		6117	<u>547</u>
		6150	<u>547</u>
		6153	<u>547</u>
		6171	<u>547</u>
		6270	<u>547</u>

Tabla 5.23 Resultados para semilla 7 y 6 en las pruebas de variación de semillas con EO\_Feromonas 80-20

### 5.3.3.4.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.56 se muestran los resultados de la variación de  $\tau$  para las semillas 6 y 7, luego en los gráficos de las Figuras 5.57 y 5.58 están los resultados individuales de la semilla 6 que obtuvo el mejor Tardiness y la semilla 7 con el mejor Makespan respectivamente.

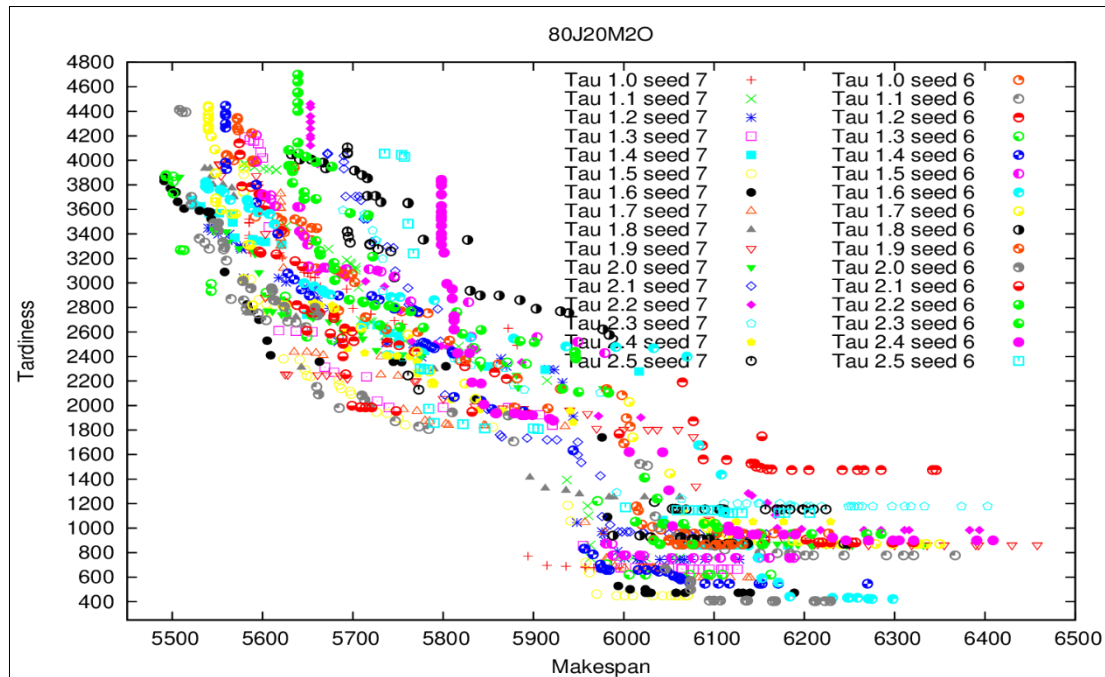


Figura 5.56 Variación de  $\tau$  para EO\_Feromonas 80-20 con semillas 7 y 6 y 1.000.000 de iteraciones

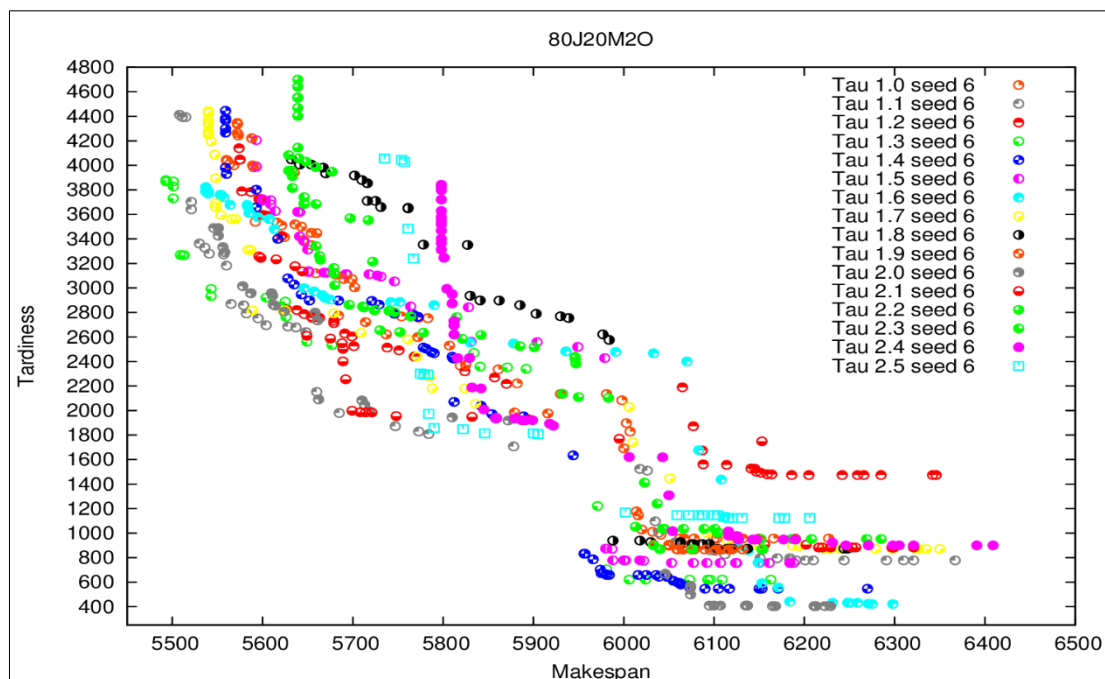


Figura 5.57 Variación de  $\tau$  para EO\_Feromonas 80-20 con semilla 6, mejor Tardiness, y 1.000.000 de iteraciones

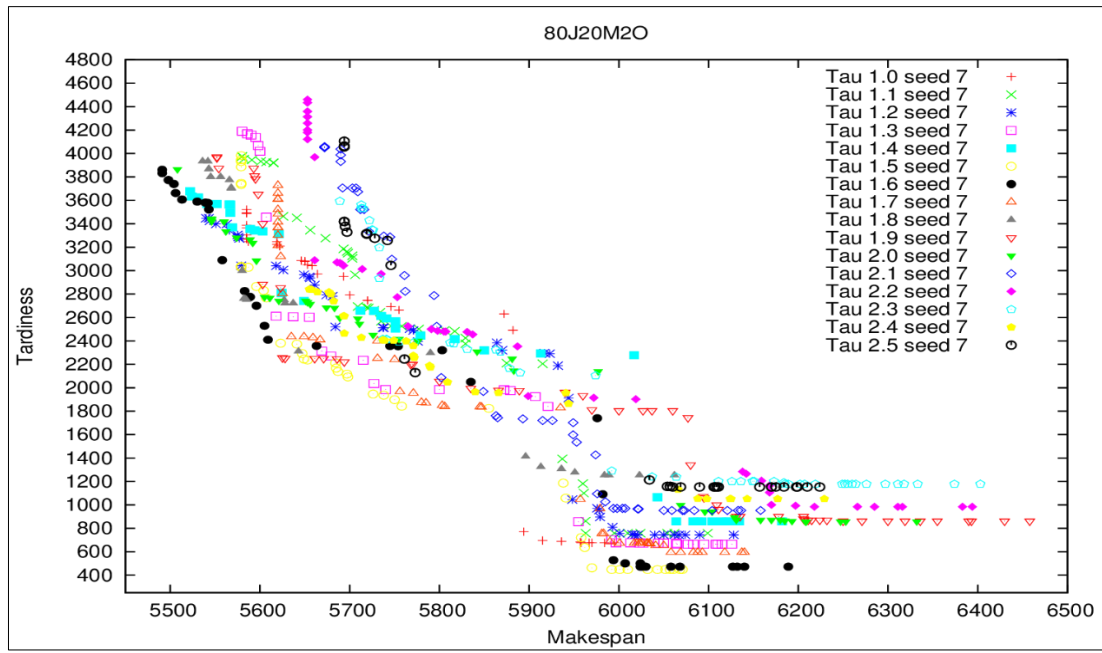


Figura 5.58 Variación de  $\tau$  para EO\_Feromonas 80-20 con semilla 7, mejor Makespan, y 1.000.000 de iteraciones



En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 6:** Mejor Makespan para  $\tau = 1,3$  con 5493, mientras que el Tardiness es para  $\tau = 2,0$  con valor 405.
- ✓ **Semilla 7:** Mejor Makespan para  $\tau = 1,6$  con 5491, mientras que el mejor Tardiness es para  $\tau = 1,5$  con valor 449.

Semilla 6				Semilla 7			
$\tau = 1,3$		$\tau = 2,0$		$\tau = 1,6$		$\tau = 1,5$	
5493	3868	5546	3489	5491	3861	5579	3946
5493	3880	5551	3427	5491	3832	5579	3746
5501	3828	5551	3489	5498	3773	5579	3932
5501	3868	5556	3334	5504	3740	5579	3735
5501	3731	5558	3292	5506	3662	5579	3037
5509	3269	5578	3015	5513	3607	5579	3028
5512	3267	5587	2958	5530	3589	5579	3886
5513	3267	5587	2957	5539	3580	5579	3982
5543	2992	5610	2957	5542	3578	5587	3028
5543	2934	5611	2937	5543	3523	5596	2865
5604	2922	5612	2863	5558	3089	5604	2830
5625	2889	5614	2856	5583	2826	5607	2767
5626	2763	5624	2807	5589	2775	5623	2380
5649	2562	5658	2799	5596	2700	5641	2373
5677	2534	5661	2736	5605	2529	5647	2292
5834	2470	5662	2091	5609	2410	5649	2245
5841	2359	5710	2083	5663	2358	5653	2236
5871	2352	5713	2048	5745	2356	5684	2211
5892	2340	5718	1994	5754	2354	5685	2168
5971	1222	5810	1944	5803	2321	5687	2138
5981	705	5859	1941	5835	2050	5697	2117
6006	621	5872	1919	5976	1740	5698	2092
6024	621	6046	669	5982	1091	5726	1946
6073	621	6074	570	5994	528	5738	1936
6093	621	6074	500	6007	501	5750	1898
6096	621	6074	556	6024	501	5758	1843
6109	621	6095	409	6024	473	5855	1824
6163	621	6100	409	6026	473	5938	1186
		6107	409	6030	473	5941	1058
		6135	409	6058	473	5958	720
		6137	409	6068	473	5962	636
		6165	405	6127	473	5970	463
		6168	405	6132	473	5992	449
		6211	405	6140	473	6001	449
		6213	405	6189	473	6010	449
		6223	405			6032	449
		6229	405			6043	449
						6052	449
						6057	449
						6061	449
						6068	449
						6071	449

Tabla 5.24 Mejores  $\tau$  para semillas 6 y 7 en EO\_Feromonas 80-20

---

### 5.3.4 EO\_Feromonas\_Busqueda

#### 5.3.4.1 Análisis 20 Jobs 20 Máquinas

##### 5.3.4.1.1 Variar las iteraciones.

En la Figura 5.59 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas\_Busqueda con los datos del *benchmark* 20 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . Se observa que todas las iteraciones obtienen resultados similares desprendiéndose levemente la de 1.000.000 con la que se trabajará en las pruebas siguientes.

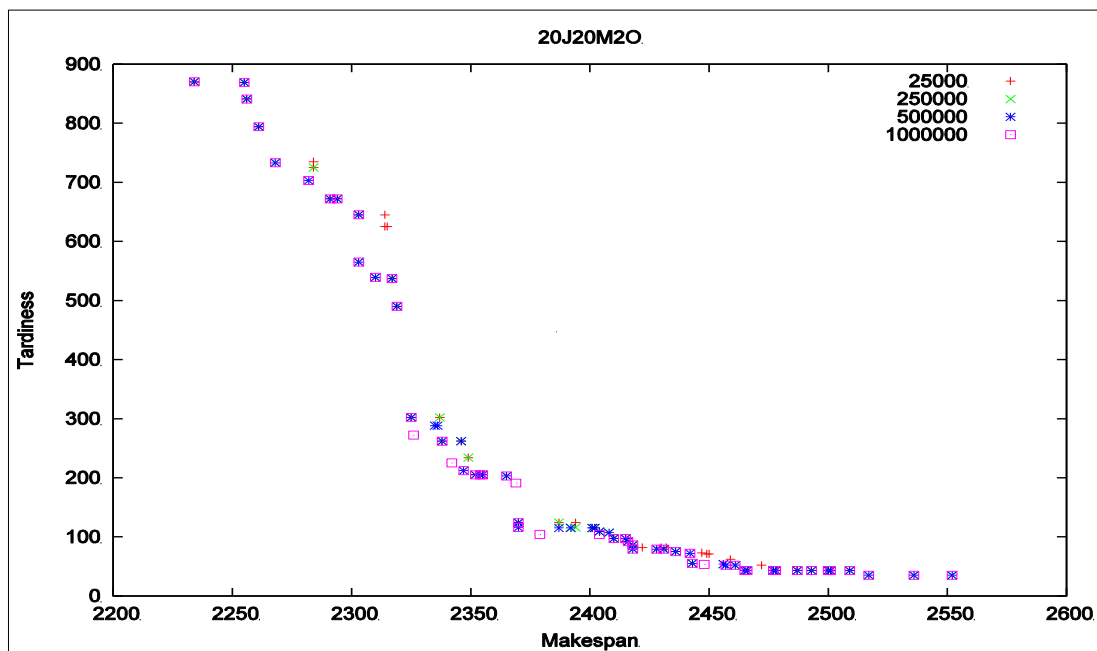


Figura 5.59 Variación de iteraciones para EO\_Feromonas\_Busqueda 20-20 con semilla 0 y  $\tau = 1,4$

### 5.3.4.1.2 Variar el valor de Semillas.

En la Figura 5.60 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas\_Busqueda con los datos del *benchmark* 20 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness son muy parecidos para la mayor cantidad de semillas, mientras que en el Makespan los valores se dispersan levemente.

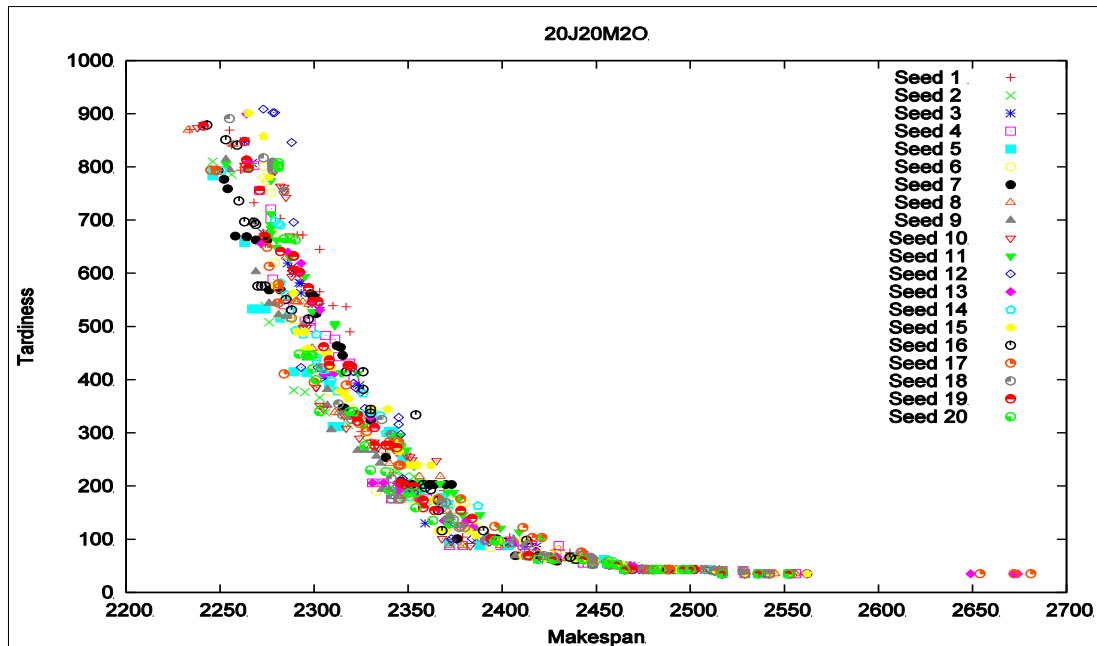


Figura 5.60 Variación de Semillas para EO\_Feromonas\_Busqueda 20-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos El mejor resultado en Makespan lo obtiene la semilla 8 con valor 2233, mientras que en Tardiness es de la semilla 1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18 19 20 con valor 35. Se representará aquella con la mejor solución que lo obtienen las semillas 1, 3, 5, 9, 10, 11, 12, 13, 14, 16, 17, 19 y 20 con el par (2517, 35) y de ellos el que tiene el mejor Makespan global, por llamarlo de alguna forma que es la semilla 1 con valor 2234:

Semilla 8		Semilla 1	
Makespan	Tardiness	Makespan	Tardiness
<b>2233</b>	869	<b>2234</b>	870
2257	842	2255	869
2263	803	2256	841
2265	798	2261	794
2273	675	2268	733
2285	631	2282	703
2286	554	2291	672
2290	547	2294	672
2291	547	2303	645
2291	546	2303	565
2294	546	2310	539
2297	541	2317	537
2299	459	2319	490
2306	459	2325	302
2311	338	2326	272
2313	338	2338	262
2317	331	2342	225
2318	331	2347	212
2325	331	2352	205
2329	269	2354	205
2340	242	2355	205
2345	239	2365	203
2352	239	2369	191
2356	218	2370	116
2367	218	2370	124
2371	152	2379	104
2372	88	2404	104
2379	88	2410	97
2399	88	2415	97
2401	88	2416	92
2411	69	2418	79
2413	69	2418	86
2415	67	2428	79
2425	67	2431	79
2426	67	2436	75
2434	64	2442	72
2435	64	2443	55
2448	62	2448	53
2448	53	2457	52
2457	52	2461	52
2465	45	2465	43
2465	43	2466	43
2469	43	2477	43
2475	43	2478	43
2492	43	2487	43
2501	43	2493	43
2503	43	2500	43
2506	43	2501	43
2513	41	2509	43
2517	41	<b>2517</b>	<b>35</b>
2528	41	2536	<b>35</b>
2529	35	2552	<b>35</b>
2540	35		
2542	35		
2545	35		
2553	35		
2556	35		

Tabla 5.25 Resultados para semilla 8 y 1 en las pruebas de variación de semillas con EO\_Feromonas\_Busqueda 20-20

5.3.4.1.3 Variar el valor de  $\tau$ .

En el gráfico de la Figura 5.61 se muestran los resultados de la variación de  $\tau$  para las semillas 1 y 8, luego en los gráficos de las Figuras 5.62 y 5.63 están los resultados individuales de la semilla 1 que obtuvo el mejor Tardiness y la semilla 8 con el mejor Makespan respectivamente.

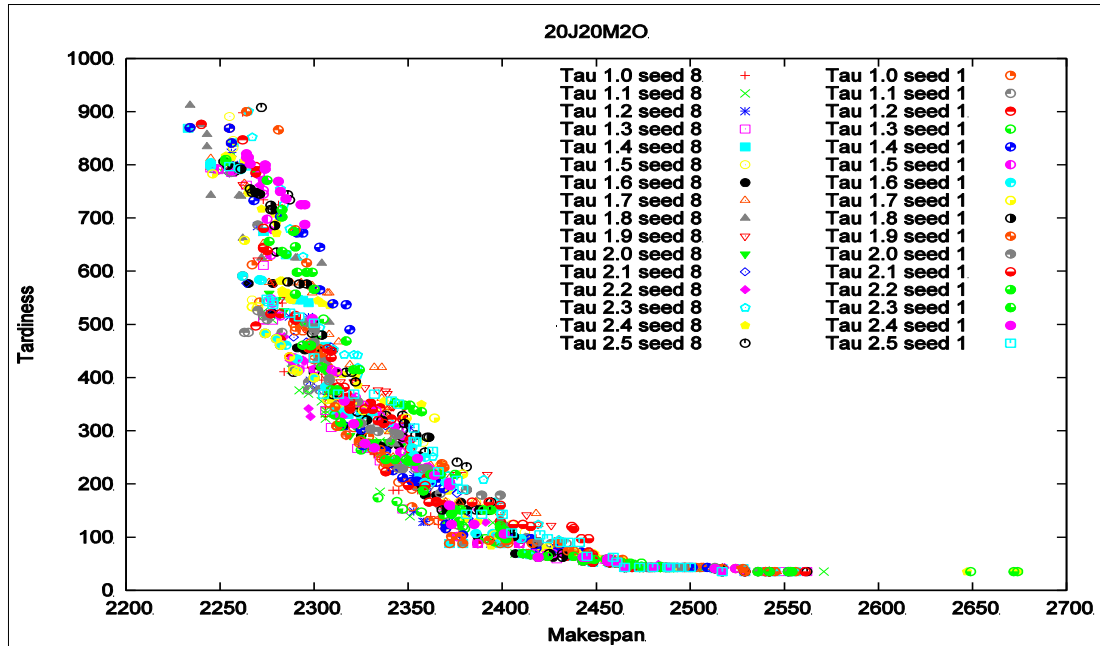


Figura 5.61 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 20-20 con semillas 8 y 1 y 1.000.000 de iteraciones

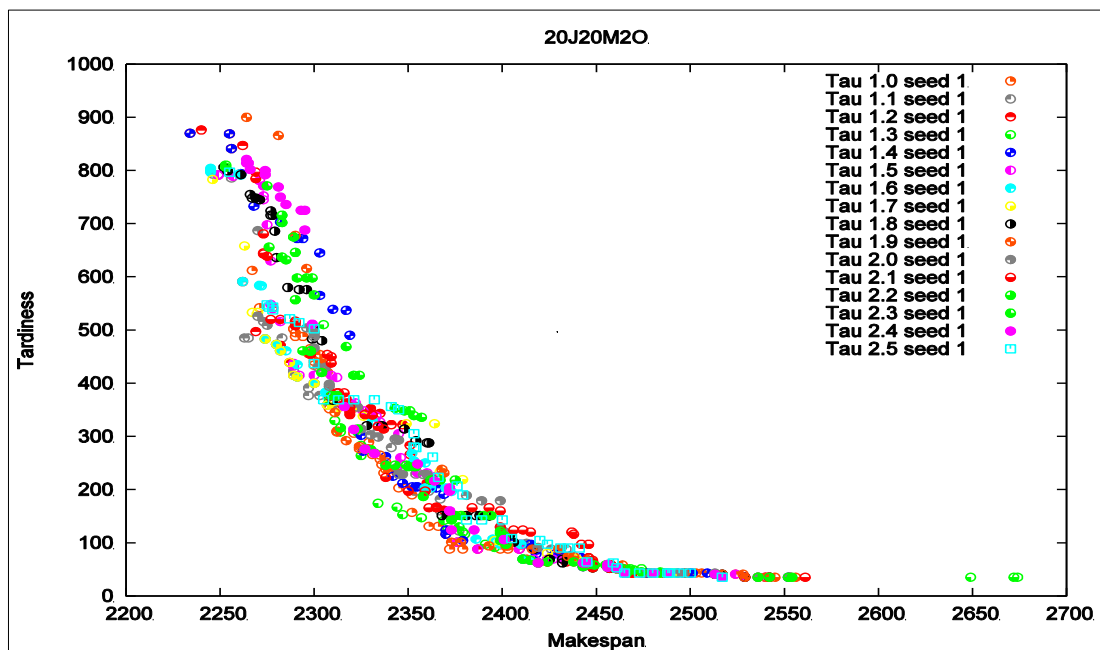


Figura 5.62 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 20-20 con semilla 1, mejor Tardiness, y 1.000.000 de iteraciones

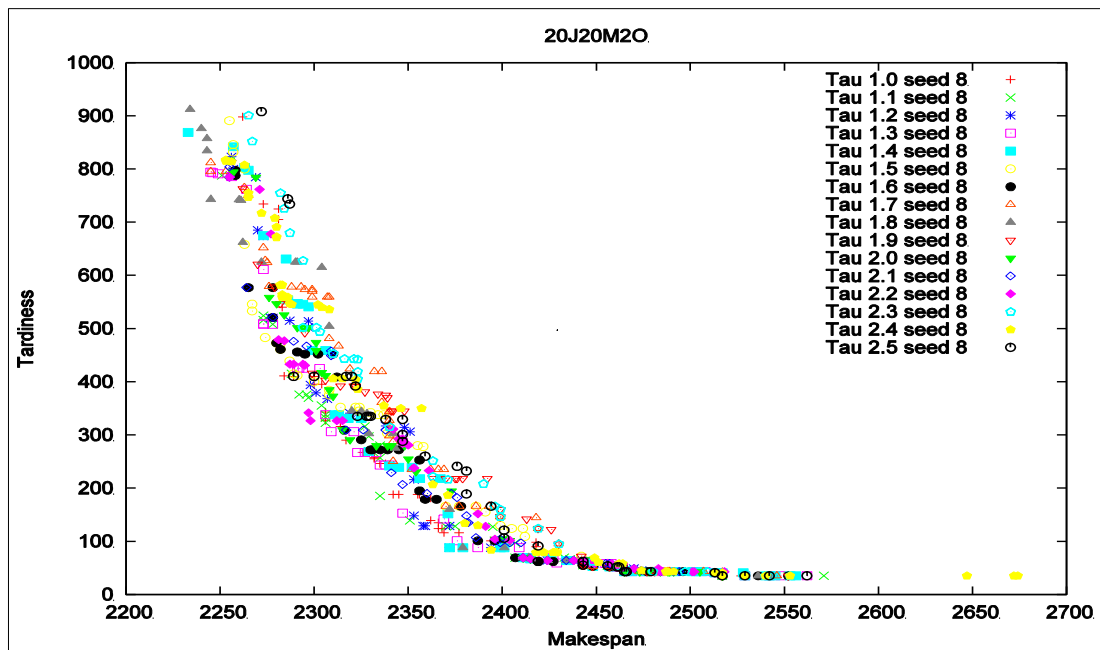


Figura 5.63 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 20-20 con semilla 8, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 1:** Mejor Makespan para  $\tau = 1,0$  y  $1,4$  con valor 2234, con el criterio de la mejor solución ambos poseen la misma (2234, 870) luego analizando el Tardiness es para ambos igual con valor 35, sin embargo, la mejor solución la tiene  $\tau = 1,4$  con (2517, 35). Por otro lado, el mejor Tardiness es para todos los  $\tau$  con valor 35 excepto  $\tau = 2,3$ , luego la mejor solución la tienen  $\tau = 1,1; 1,3; 1,4; 1,6; 1,7; 1,8; 2,0; 2,1; 2,2; 2,4$  y  $2,5$  con (2517, 35) y de ellos el que tiene el mejor Makespan  $\tau = 1,4$  con valor 2234.
- ✓ **Semilla 8:** Mejor Makespan para  $\tau = 1,4$  con 2233, mientras que el mejor Tardiness es para:
- ✓  $\tau = 1,0; 1,1; 1,3; 1,4; 1,5; 1,6; 1,7; 1,8; 1,9; 2,0; 2,3; 2,4$  y  $2,5$  con valor 35, de ellos la mejor solución (2517, 35) es para:
- ✓  $\tau = 1,0; 1,1; 1,5; 1,6; 1,7; 1,8; 1,9; 2,0; 2,3; 2,4$  y  $2,5$  donde el mejor Makespan lo tiene  $\tau = 1,8$  con valor 2234.

Semilla 1		Semilla 8			
$\tau = 1,4$		$\tau = 1,4$		$\tau = 1,8$	
2234	870	2233	869	2234	912
2255	869	2257	842	2240	876
2256	841	2263	803	2243	857
2261	794	2265	798	2243	834
2268	733	2273	675	2245	743
2282	703	2285	631	2260	742
2291	672	2286	554	2261	741
2294	672	2290	547	2262	662
2303	645	2291	547	2272	625
2303	565	2291	546	2290	625
2310	539	2294	546	2291	625
2317	537	2297	541	2304	615
2319	490	2299	459	2308	504
2325	302	2306	459	2320	345
2326	272	2311	338	2325	344
2338	262	2313	338	2328	340
2342	225	2317	331	2329	302
2347	212	2318	331	2342	302
2352	205	2325	331	2343	275
2354	205	2329	269	2372	160
2355	205	2340	242	2379	88
2365	203	2345	239	2401	88
2369	191	2352	239	2411	69
2370	116	2356	218	2414	69
2370	124	2367	218	2415	67
2379	104	2371	152	2424	64
2404	104	2372	88	2434	64
2410	97	2379	88	2438	64
2415	97	2399	88	2444	63
2416	92	2401	88	2446	59
2418	79	2411	69	2449	58
2418	86	2413	69	2461	51
2428	79	2415	67	2464	51
2431	79	2425	67	2465	43
2436	75	2426	67	2475	43
2442	72	2434	64	2484	43
2443	55	2435	64	2513	41
2448	53	2448	62	2517	35
2457	52	2448	53	2517	41
2461	52	2457	52	2536	35
2465	43	2465	45	2542	35
2466	43	2465	43		
2477	43	2469	43		
2478	43	2475	43		
2487	43	2492	43		
2493	43	2501	43		
2500	43	2503	43		
2501	43	2506	43		
2509	43	2513	41		
2517	35	2517	41		
2536	35	2528	41		
2552	35	2529	35		
		2540	35		
		2542	35		
		2545	35		
		2553	35		
		2556	35		

Tabla 5.26 Mejores  $\tau$  para semillas 1 y 8 en EO\_Feromonas\_Busqueda 20-20

---

### 5.3.4.2 Análisis 40 Jobs 20 Máquinas

#### 5.3.4.2.1 Variar las iteraciones.

En la Figura 5.64 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas\_Busqueda con los datos del *benchmark* 40 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . En él se logra observar que los mejores resultados se obtienen con 1.000.000 de iteraciones.

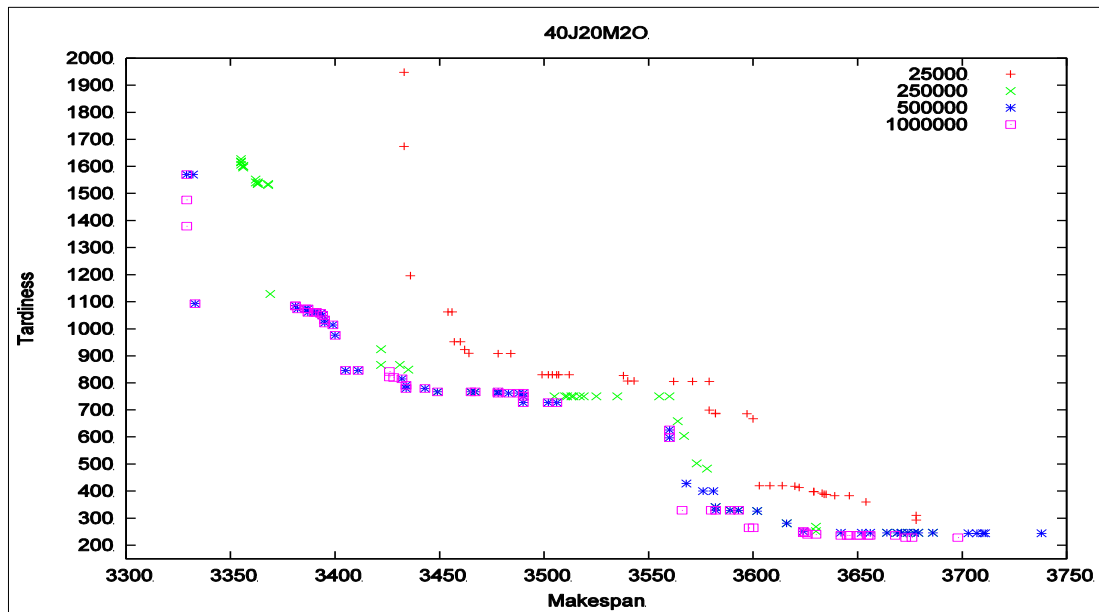


Figura 5.64 Variación de iteraciones para EO\_Feromonas\_Busqueda 40-20 con semilla 0 y  $\tau = 1,4$



### 5.3.4.2.2 Variar el valor de Semillas.

En la Figura 5.65 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas\_Busqueda con los datos del *benchmark* 40 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness y Makespan tienen una leve tendencia a agruparse para las distintas semillas.

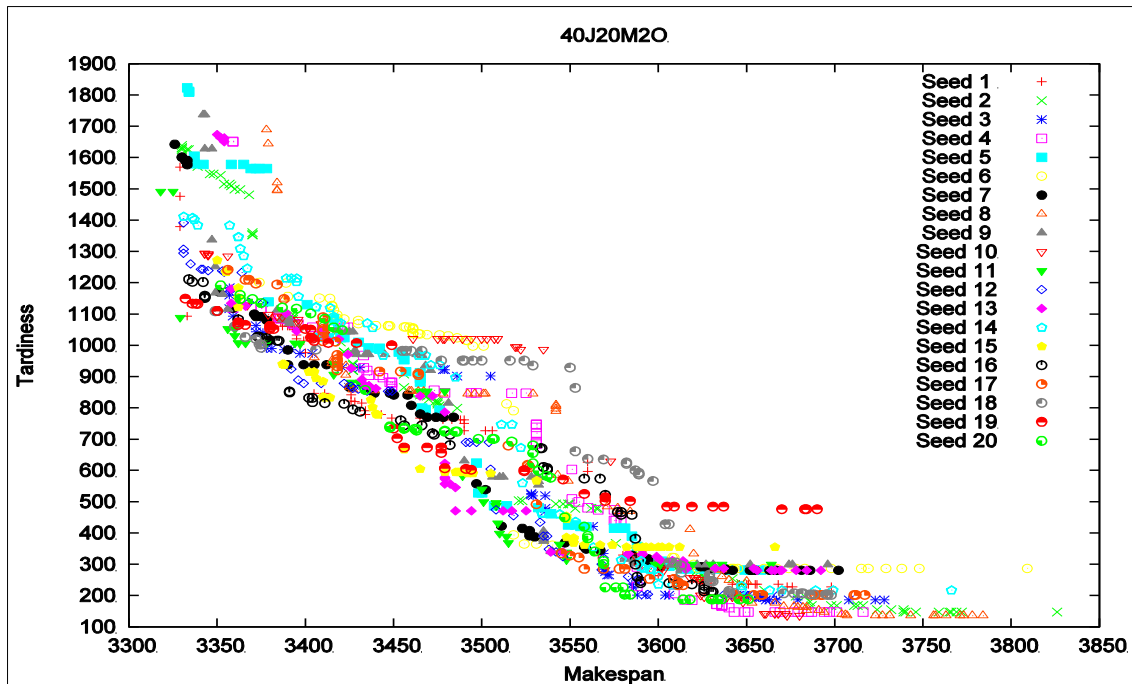


Figura 5.65 Variación de Semillas para EO\_Feromonas\_Busqueda 40-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos El mejor resultado en Makespan lo obtiene la semilla 11 con valor 3318, mientras que en Tardiness es de la semilla 8 y 10 con valor 136. Se decide mostrar la semilla 10 pues tiene la mejor solución con valor (2673, 136):

Semilla 11		Semilla 10	
Makespan	Tardiness	Makespan	Tardiness
<u>3318</u>	1492	3343	1294
3325	1492	3343	1293
3329	1089	3345	1287
3356	1053	3345	1291
3360	1052	3345	1292
3360	1031	3356	1241
3362	1008	3356	1285
3366	1007	3366	1128

3378	1007	3366	1129
3394	1007	3367	1127
3397	1005	3379	1098
3416	907	3384	1093
3416	941	3387	1090
3426	882	3396	1071
3431	882	3396	1082
3431	876	3409	1068
3465	854	3411	1035
3469	854	3461	1021
3479	854	3475	1021
3489	583	3478	1021
3500	540	3479	1021
3501	500	3484	1021
3508	497	3489	1021
3508	487	3494	1021
3508	488	3500	1021
3509	432	3503	1021
3510	399	3507	1021
3514	389	3509	1021
3515	369	3519	995
3544	364	3520	995
3548	339	3522	987
3548	315	3535	987
3598	300	3573	630
3601	300	3587	326
3610	300	3589	294
3614	300	3596	286
3619	300	3601	286
3630	300	3601	281
3636	300	3603	281
3664	300	3608	257
		3609	257
		3618	257
		3620	257
		3621	257
		3623	251
		3624	251
		3624	201
		3626	196
		3631	196
		3639	196
		3642	196
		3644	196
		3645	196
		3646	196
		3648	192
		3648	179
		3660	142
		3661	142
		3666	142
		3668	142
		3672	142
		<b>3673</b>	<b>136</b>
		3680	<b>136</b>

Tabla 5.27 Resultados para semilla 11 y 10 en las pruebas de variación de semillas con EO\_Feromonas\_Busqueda 20-20

### 5.3.4.2.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.66 se muestran los resultados de la variación de  $\tau$  para las semillas 10 y 11, luego en los gráficos de las Figuras 5.67 y 5.68 están los resultados individuales de la semilla 10 que obtuvo el mejor Tardiness y la semilla 11 con el mejor Makespan respectivamente.

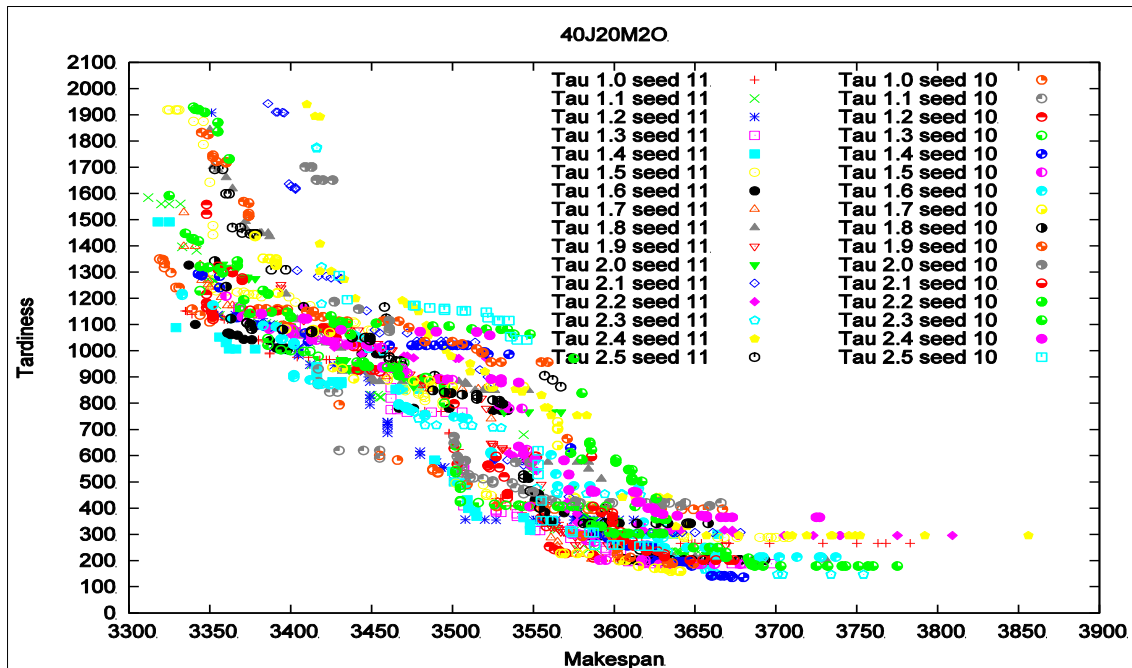


Figura 5.66 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 40-20 con semillas 11 y 10 y 1.000.000 de iteraciones

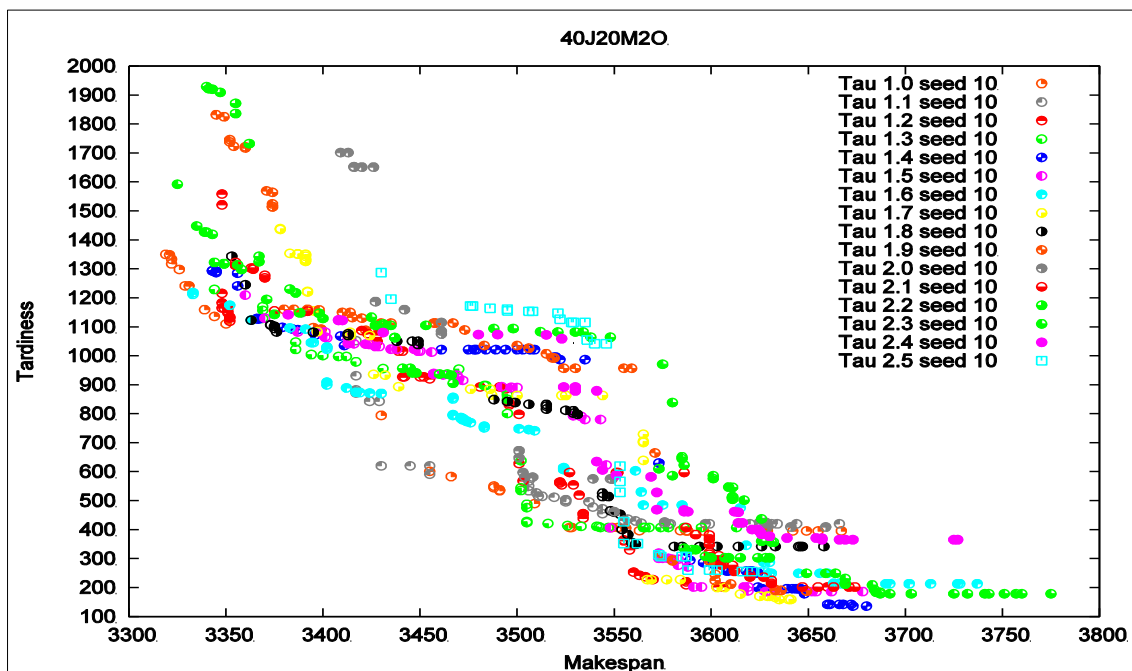


Figura 5.67 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 40-20 con semilla 10, mejor Tardiness, y 1.000.000 de iteraciones

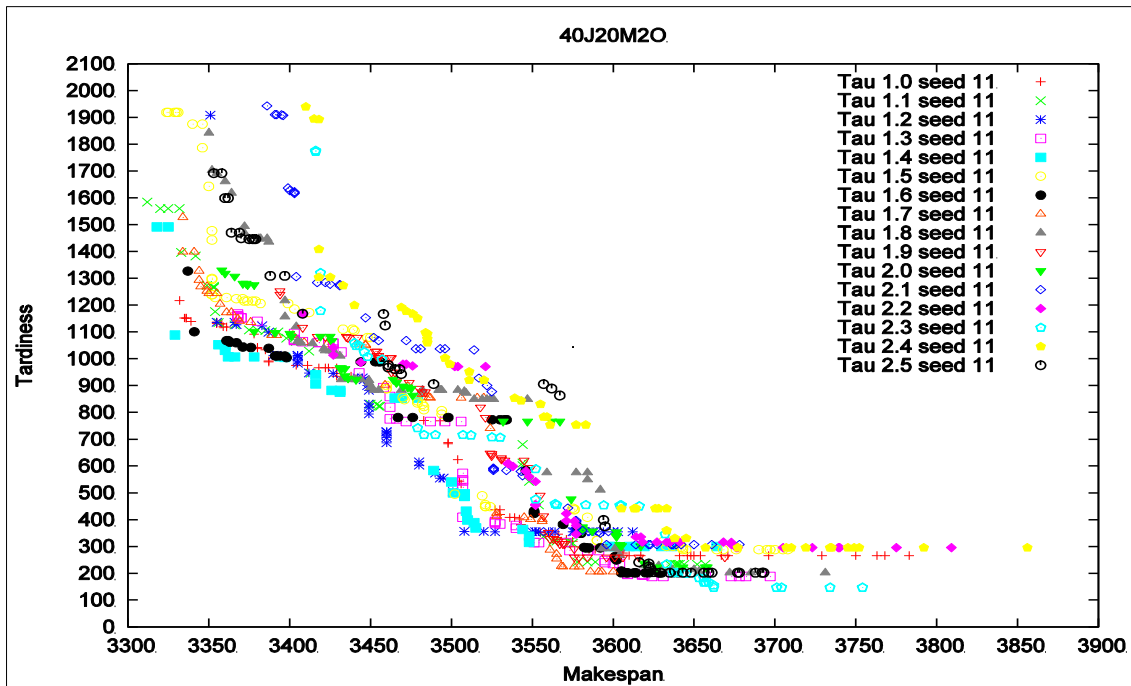


Figura 5.68 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 40-20 con semilla 11, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 10:** Mejor Makespan para  $\tau = 1,0$  con valor 3319, mientras que el mejor Tardiness es para  $\tau = 1,4$  con valor 136.
- ✓ **Semilla 11:** Mejor Makespan para  $\tau = 1,1$  con 3312, mientras que el mejor Tardiness es para  $\tau = 2,3$  con valor 147.

Semilla 10				Semilla 11			
$\tau = 1,0$		$\tau = 1,4$		$\tau = 1,1$		$\tau = 2,3$	
3319	1350	3343	1294	3312	1584	3416	1777
3321	1348	3343	1293	3320	1560	3416	1772
3322	1333	3345	1287	3325	1560	3419	1320
3322	1317	3345	1291	3332	1560	3419	1179
3326	1298	3345	1292	3333	1396	3440	1059
3329	1241	3356	1241	3342	1383	3441	1048
3331	1241	3356	1285	3349	1274	3443	1048
3339	1159	3366	1128	3353	1266	3446	1025
3344	1136	3366	1129	3353	1264	3448	1008
3350	1110	3367	1127	3353	1270	3457	998
3395	1097	3379	1098	3354	1175	3479	742
3399	1089	3384	1093	3356	1135	3483	717
3399	1092	3387	1090	3366	1126	3490	717
3422	1088	3396	1071	3375	1107	3507	717
3426	1086	3396	1082	3380	1107	3512	715
3430	794	3409	1068	3385	1105	3525	708
3455	601	3411	1035	3389	1100	3530	707
3466	583	3461	1021	3393	1100	3552	476

3488	550	3475	1021	3398	1077	3552	589
3488	544	3478	1021	3404	1065	3564	459
3491	534	3479	1021	3412	1028	3565	455
3491	537	3484	1021	3413	946	3583	455
3509	490	3489	1021	3450	830	3594	454
3527	407	3494	1021	3454	830	3604	454
3528	407	3500	1021	3455	823	3605	454
3549	407	3503	1021	3456	823	3616	450
3552	407	3507	1021	3544	611	3632	346
3556	405	3509	1021	3544	680	3633	234
3563	396	3519	995	3544	601	3637	202
3584	396	3520	995	3548	542	3639	202
3587	395	3522	987	3548	576	3645	199
3588	395	3535	987	3554	455	3653	183
3596	395	3573	630	3557	337	3656	166
3627	395	3587	326	3563	320	3656	183
3628	395	3589	294	3565	320	3657	166
3629	395	3596	286	3573	316	3659	166
3642	395	3601	286	3575	312	3662	147
3649	395	3601	281	3578	242	3662	155
3655	395	3603	281	3584	242	3701	147
3667	395	3608	257	3589	242	3704	147
		3609	257	3635	233	3734	147
		3618	257	3638	233	3754	147
		3620	257	3639	233		
		3621	257	3640	233		
		3623	251	3641	233		
		3624	251	3644	233		
		3624	201	3652	233		
		3626	196	3657	233		
		3631	196				
		3639	196				
		3642	196				
		3644	196				
		3645	196				
		3646	196				
		3648	192				
		3648	179				
		3660	142				
		3661	142				
		3666	142				
		3668	142				
		3672	142				
		3673	136				
		3680	136				

Tabla 5.28 Mejores  $\tau$  para semillas 10 y 11 en EO\_Feromonas\_Busqueda 40-20

### 5.3.4.3 Análisis 60 Jobs 20 Máquinas

#### 5.3.4.3.1 Variar las iteraciones.

En la Figura 5.69 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas\_Busqueda con los datos del *benchmark* 60 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . Se observa que los mejores resultados se logran con 1.000.000 de iteraciones.

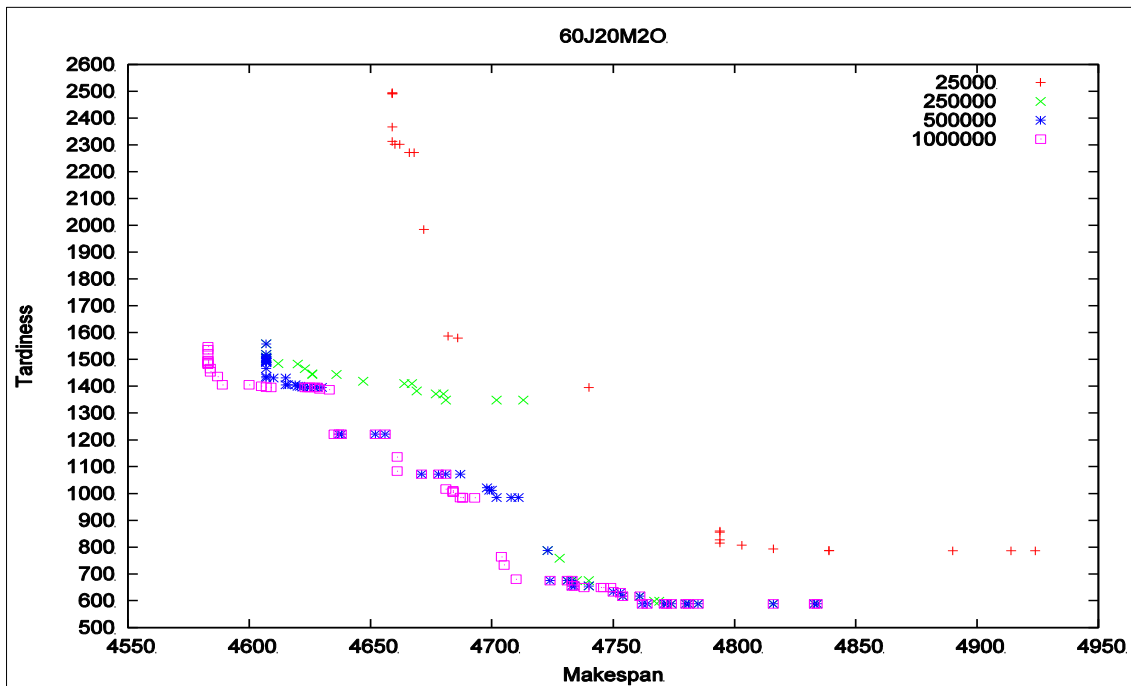


Figura 5.69 Variación de iteraciones para EO\_Feromonas\_Busqueda 60-20 con semilla 0 y  $\tau = 1,4$

### 5.3.4.3.2 Variar el valor de Semillas.

En la Figura 5.70 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas\_Busqueda con los datos del *benchmark* 60 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . Se observa que los valores de Tardiness y Makespan son dispersos para las distintas semillas.

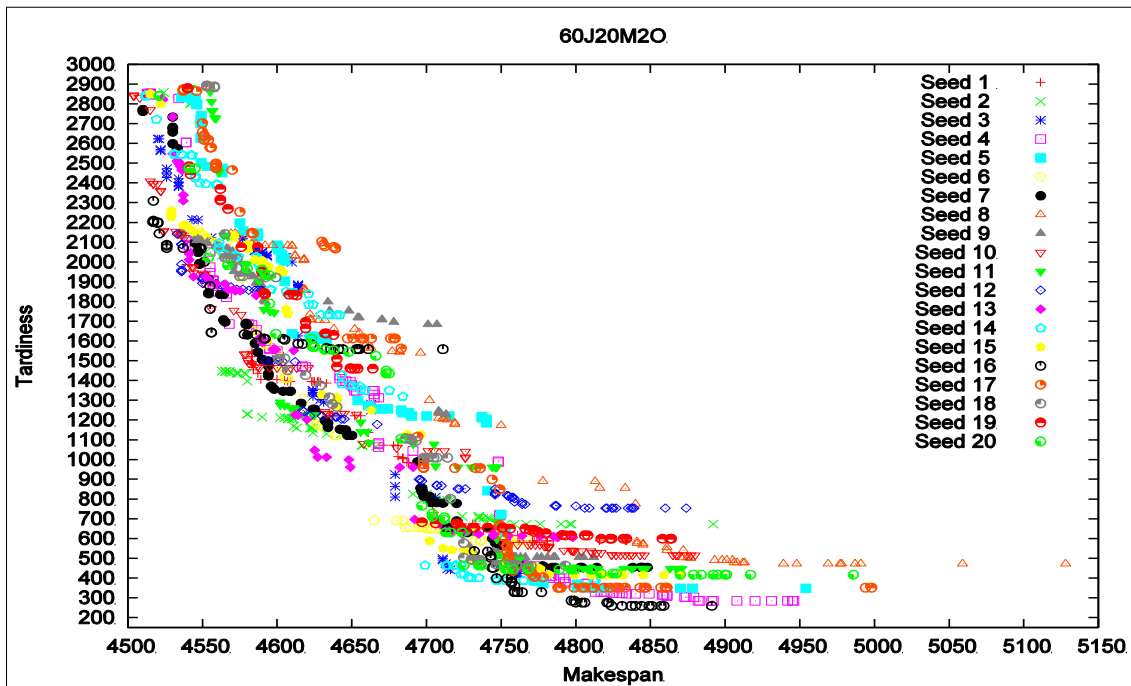


Figura 5.70 Variación de Semillas para EO\_Feromonas\_Busqueda 60-20 con 1.000.000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos El mejor resultado en Makespan lo obtiene la semilla 10 con valor 4504, mientras que en Tardiness es de la semilla 16 con valor 259:

Semilla 10		Semilla 16	
Makespan	Tardiness	Makespan	Tardiness
<b>4504</b>	2843	4517	2203
<b>4504</b>	2844	4517	2202
<b>4504</b>	2841	4517	2209
4508	2831	4517	2309
4515	2770	4517	2201
4515	2409	4520	2199
4517	2398	4520	2198
4517	2395	4520	2200
4520	2395	4521	2144
4522	2358	4526	2071
4522	2361	4526	2082
4524	2157	4526	2087
4524	2154	4537	2071
4532	2154	4548	2070
4534	2143	4549	2068
4536	2138	4551	2001
4538	2123	4555	1762
4540	1977	4555	1879
4543	1975	4556	1642
4544	1973	4556	1643
4544	1944	4578	1634
4550	1942	4586	1632
4551	1893	4591	1612
4551	1927	4592	1611
4551	1935	4604	1611
4551	1925	4605	1606
4551	1937	4614	1586
4556	1764	4617	1584
4571	1755	4623	1584
4576	1733	4633	1565
4578	1534	4636	1558
4579	1533	4641	1558
4580	1502	4644	1558
4581	1489	4653	1558
4587	1472	4654	1558
4588	1465	4655	1558
4595	1465	4661	1558
4600	1465	4711	1558
4606	1465	4714	646
4609	1465	4714	654
4613	1465	4727	631
4621	1465	4732	536
4632	1241	4741	534
4633	1234	4743	512
4634	1234	4743	502
4643	1234	4744	452
4645	1234	4745	452
4653	1229	4747	399
4658	1080	4754	399
4680	1059	4757	399
4701	1044	4757	375
4705	1044	4757	379
4712	1044	4757	382



4726	1040	4759	360
4726	1012	4759	328
4726	1008	4759	332
4750	584	4761	328
4753	584	4764	328
4755	569	4777	328
4761	569	4797	285
4768	569	4799	285
4770	569	4800	276
4773	569	4804	276
4776	569	4806	276
4779	561	4821	276
4780	561	4822	276
4782	561	4824	<u>259</u>
4782	554	4831	<u>259</u>
4782	557	4837	<u>259</u>
4792	540	4840	<u>259</u>
4797	540	4844	<u>259</u>
4798	540	4849	<u>259</u>
4798	536	4851	<u>259</u>
4799	536	4857	<u>259</u>
4804	536	4859	<u>259</u>
4810	536	4891	<u>259</u>
4811	527		
4814	527		
4817	527		
4824	518		
4824	516		
4827	516		
4830	516		
4833	516		
4836	516		
4838	516		
4843	516		
4846	516		
4865	516		
4867	516		
4868	516		
4874	516		
4877	516		
4880	516		

Tabla 5.29 Resultados para semilla 10 y 16 en las pruebas de variación de semillas con EO\_Feromonas\_Busqueda 60-20

### 5.3.4.3.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.71 se muestran los resultados de la variación de  $\tau$  para las semillas 10 y 16, luego en los gráficos de las Figuras 5.72 y 5.73 están los resultados individuales de la semilla 16 que obtuvo el mejor Tardiness y la semilla 10 con el mejor Makespan respectivamente.

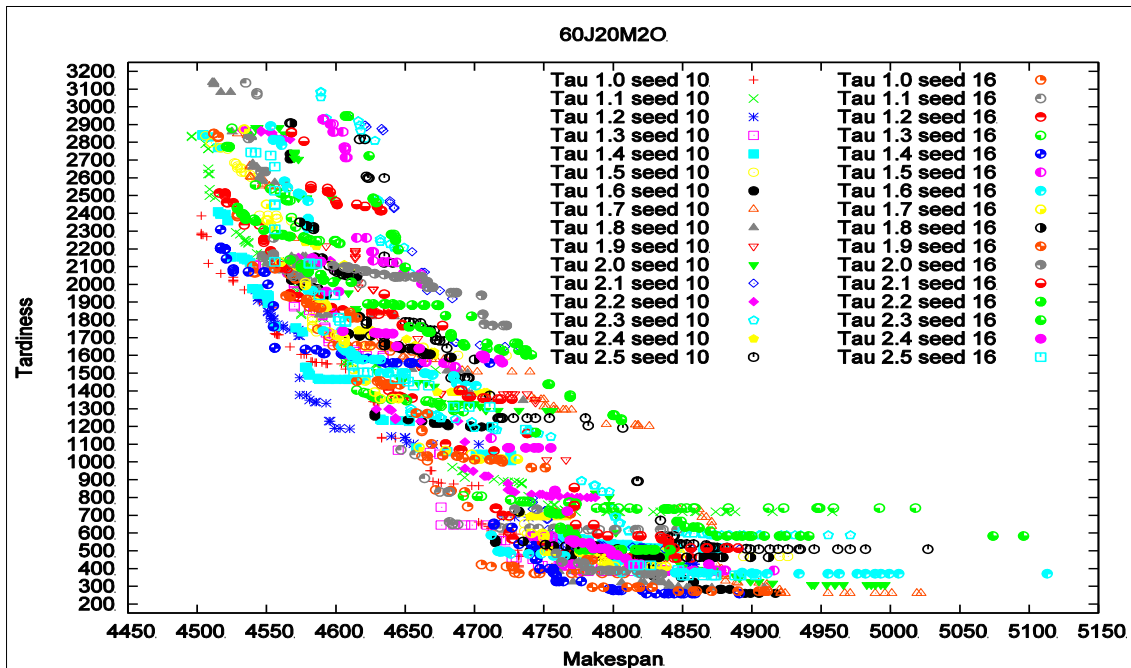


Figura 5.71 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 60-20 con semillas 10 y 16 y 1.000.000 de iteraciones

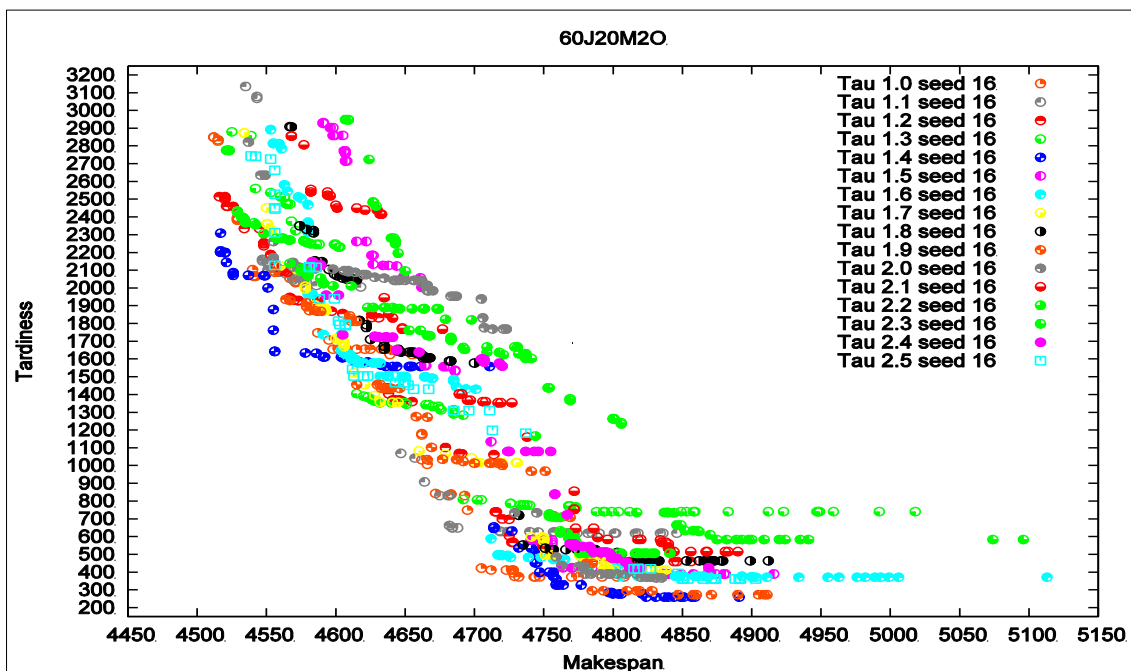


Figura 5.72 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 60-20 con semilla 16, mejor Tardiness, y 1.000.000 de iteraciones

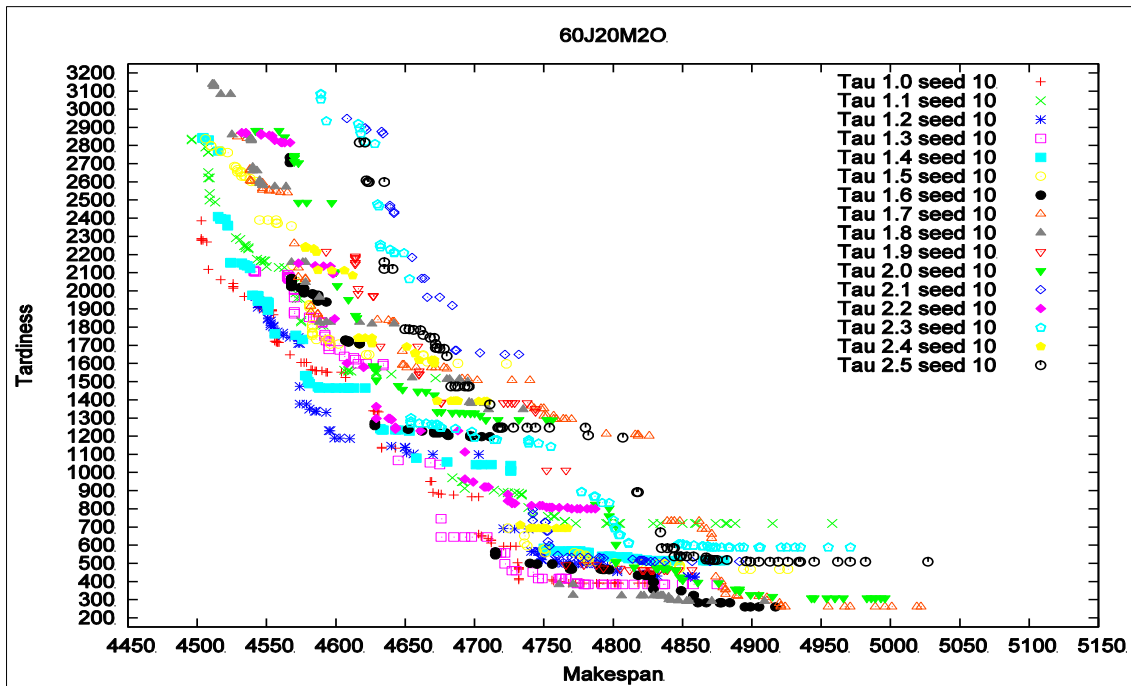


Figura 5.73 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 60-20 con semilla 10, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 10:** Mejor Makespan  $\tau = 1,1$  valor 4496 y el mejor Tardiness  $\tau = 1,6$  y 1,7 valor 261. Mejor solución para  $\tau = 1,6$  con valor (4895, 261).
- ✓ **Semilla 16:** Mejor Makespan para  $\tau = 1,0$  con 4512, mientras que el mejor Tardiness es para  $\tau = 1,4$  con valor 259.

Semilla 10				Semilla 16			
$\tau = 1,1$		$\tau = 1,6$		$\tau = 1,0$		$\tau = 1,4$	
4496	2836	4567	2707	4512	2849	4517	2203
4496	2831	4567	2734	4512	2850	4517	2202
4506	2792	4568	2069	4515	2834	4517	2209
4508	2763	4568	2059	4515	2828	4517	2309
4508	2651	4568	2054	4529	2384	4517	2201
4508	2622	4568	2028	4529	2392	4520	2199
4508	2764	4568	2023	4529	2396	4520	2198
4509	2536	4568	2030	4529	2381	4520	2200
4509	2499	4568	2055	4540	2103	4521	2144
4509	2622	4568	2047	4541	2068	4526	2071
4513	2488	4568	2067	4541	2069	4526	2082
4528	2292	4574	2018	4541	2065	4526	2087
4531	2282	4577	2013	4542	2065	4537	2071
4534	2248	4577	1994	4567	2055	4548	2070
4535	2248	4577	1996	4571	2030	4549	2068
4535	2238	4577	1989	4571	2036	4551	2001
4537	2231	4583	1984	4571	2033	4555	1762
4537	2238	4584	1980	4579	1990	4555	1879

4543	2174	4587	1951	4585	1932	4556	1642
4546	2166	4587	1944	4587	1748	4556	1643
4546	2167	4587	1976	4595	1707	4578	1634
4548	2165	4587	1949	4598	1655	4586	1632
4549	2165	4587	1968	4602	1655	4591	1612
4550	2133	4587	1975	4608	1655	4592	1611
4559	2129	4593	1939	4615	1655	4604	1611
4563	2129	4607	1730	4621	1655	4605	1606
4566	2082	4607	1727	4625	1655	4614	1586
4567	2062	4609	1724	4633	1655	4617	1584
4570	2039	4609	1719	4636	1655	4623	1584
4571	1989	4609	1722	4638	1655	4633	1565
4572	1958	4617	1710	4639	1625	4636	1558
4574	1958	4617	1709	4655	1625	4641	1558
4575	1837	4628	1259	4662	1180	4644	1558
4575	1828	4628	1272	4662	1033	4653	1558
4589	1810	4652	1238	4662	1172	4654	1558
4591	1810	4662	1229	4666	1033	4655	1558
4606	1563	4671	1229	4666	1007	4661	1558
4609	1563	4671	1218	4672	843	4711	1558
4609	1557	4674	1218	4683	839	4714	646
4611	1557	4679	1216	4683	836	4714	654
4629	1542	4679	1218	4693	831	4727	631
4629	1548	4681	1204	4695	749	4732	536
4629	1554	4697	1204	4705	422	4741	534
4640	1542	4697	1198	4712	411	4743	512
4672	1520	4698	1198	4723	411	4743	502
4684	971	4705	1198	4727	411	4744	452
4691	950	4711	1198	4728	395	4745	452
4692	950	4715	547	4728	411	4747	399
4693	912	4715	563	4729	376	4754	399
4714	902	4740	501	4732	373	4757	399
4721	891	4745	496	4743	373	4757	375
4724	889	4756	496	4744	373	4757	379
4729	888	4768	487	4747	373	4757	382
4732	887	4769	469	4770	373	4759	360
4734	882	4770	469	4775	373	4759	328
4734	875	4791	469	4792	373	4759	332
4734	879	4792	469	4803	373	4761	328
4739	807	4793	466	4808	373	4764	328
4742	759	4797	465			4777	328
4753	759	4818	435			4797	285
4757	759	4823	435			4799	285
4758	759	4825	435			4800	276
4765	731	4827	435			4804	276
4765	729	4829	396			4806	276
4773	720	4829	357			4821	276
4795	720	4829	397			4822	276
4796	720	4849	349			4824	259
4804	720	4858	323			4831	259
4805	720	4861	283			4837	259
4829	720	4867	283			4840	259
4834	720	4877	283			4844	259
4850	720	4878	283			4849	259
4859	720	4884	283			4851	259
4860	720	4895	261			4857	259
4874	720	4899	261			4859	259
4881	720	4905	261			4891	259
4882	720	4917	261				
4888	720						
4915	720						
4958	720						

Tabla 5.30 Mejores  $\tau$  para semillas 10 y 16 en EO\_Feromonas\_Busqueda 60-20

### 5.3.4.4 Análisis 80 Jobs 20 Máquinas

#### 5.3.4.4.1 Variar las iteraciones.

En la Figura 5.74 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas\_Busqueda con los datos del *benchmark* 80 Jobs y 20 Máquinas haciendo variar la cantidad de iteraciones con valor de semilla 0 y  $\tau = 1,4$ . Se observa que los mejores resultados se logran con 1.000.000 de iteraciones.

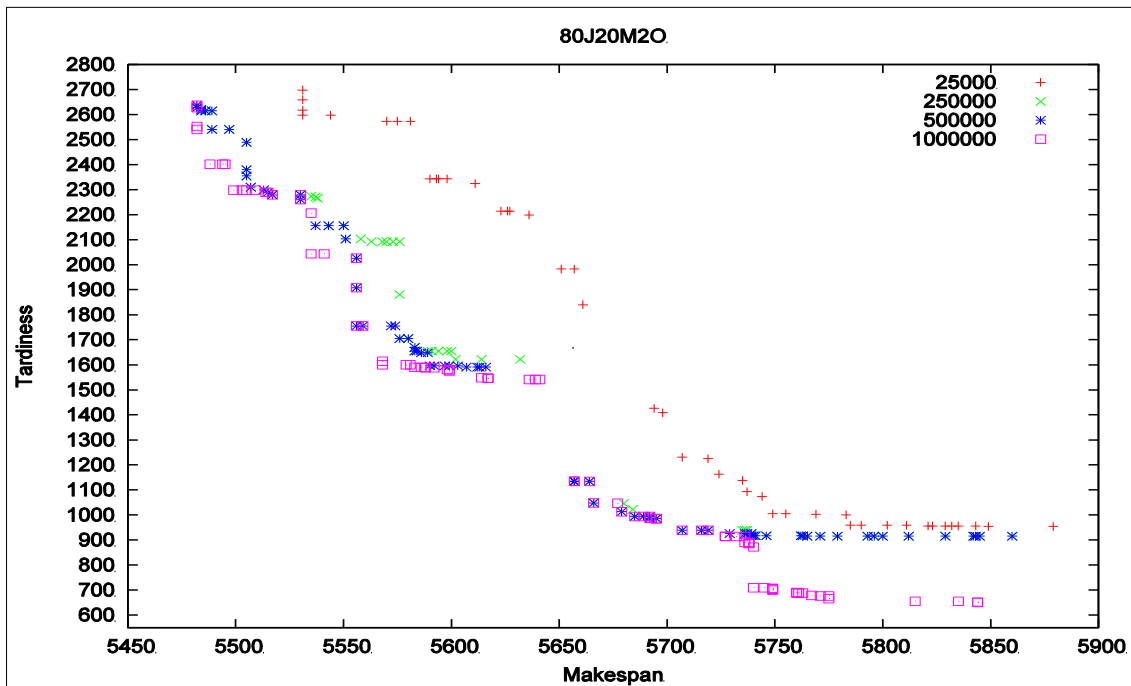


Figura 5.74 Variación de iteraciones para EO\_Feromonas\_Busqueda 80-20 con semilla 0 y  $\tau = 1,4$

### 5.3.4.4.2 Variar el valor de Semillas.

En la Figura 5.75 se grafican los resultados obtenidos en el algoritmo de EO\_Feromonas\_Busqueda con los datos del *benchmark* 80 Jobs y 20 Máquinas haciendo variar el valor de semilla. En esta prueba se utilizó el mejor resultado de la variación de iteraciones, es decir, 1.000.000 y  $\tau = 1,4$ . En él se logra observar que los valores de Tardiness y Makespan están dispersos para las distintas semillas.

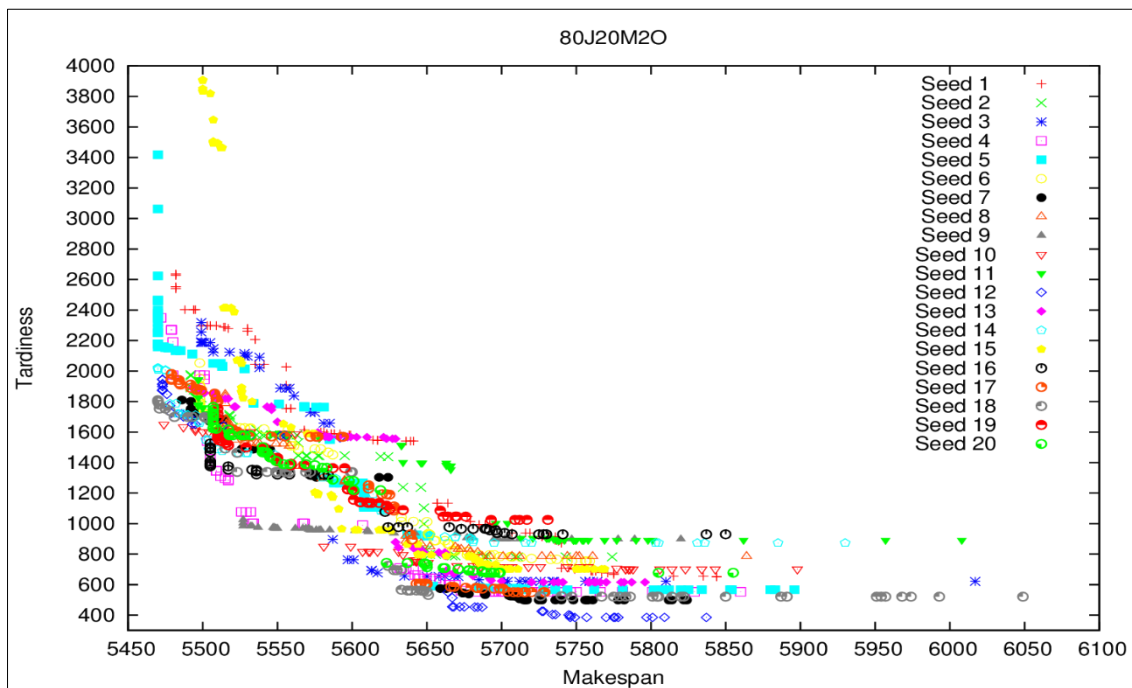


Figura 5.75 Variación de Semillas para EO\_Feromonas\_Busqueda 80-20 con 1000000 de iteraciones y  $\tau = 1,4$

De acuerdo con el análisis de los datos El mejor resultado en Makespan lo obtienen las semillas 5, 14 y 18 con valor 5470. Se representa la 18 pues tiene mejor solución con valor (5470, 1798), mientras que en Tardiness es de la semilla 12 con valor 385:

Semilla 18		Semilla 12	
Makespan	Tardiness	Makespan	Tardiness
<b>5470</b>	1809	5473	1878
<b>5470</b>	<b>1798</b>	5473	1944
5471	1754	5473	1916
5471	1779	5473	1899
5475	1754	5473	1946
5478	1739	5473	1912
5481	1699	5473	1872
5490	1699	5476	1848
5493	1699	5478	1773
5501	1699	5479	1724

5502	1694	5484	1719
5504	1685	5486	1719
5506	1685	5487	1700
5511	1558	5491	1700
5515	1492	5493	1683
5515	1558	5493	1652
5523	1464	5495	1652
5523	1339	5514	1641
5543	1339	5524	1582
5549	1339	5530	1582
5550	1339	5532	1576
5555	1339	5552	1576
5556	1339	5556	1576
5558	1338	5592	1562
5569	1338	5664	550
5572	1338	5667	514
5599	1338	5667	454
5600	1338	5668	454
5624	723	5672	454
5628	695	5675	454
5630	695	5682	453
5633	695	5684	453
5633	566	5687	452
5637	562	5727	425
5639	562	5728	425
5643	562	5734	404
5646	562	5736	404
5647	560	5745	401
5648	560	5746	398
5650	553	5746	<b><u>385</u></b>
5651	532	5749	<b><u>385</u></b>
5726	524	5757	<b><u>385</u></b>
5740	524	5770	<b><u>385</u></b>
5747	521	5775	<b><u>385</u></b>
5767	521	5776	<b><u>385</u></b>
5775	521	5778	<b><u>385</u></b>
5779	521	5797	<b><u>385</u></b>
5783	521	5801	<b><u>385</u></b>
5786	521	5809	<b><u>385</u></b>
5801	521	5837	<b><u>385</u></b>
5804	521		
5805	521		
5814	521		
5820	521		
5822	521		
5850	521		
5887	521		
5891	521		
5951	521		
5954	521		
5957	521		
5968	521		
5974	521		
5993	521		
6049	521		

Tabla 5.31 Resultados para semilla 18 y 12 en las pruebas de variación de semillas con EO\_Feromonas\_Busqueda 80-20

### 5.3.4.4.3 Variar el valor de $\tau$ .

En el gráfico de la Figura 5.76 se muestran los resultados de la variación de  $\tau$  para las semillas 12 y 18, luego en los gráficos de las Figuras 5.77 y 5.78 están los resultados individuales de la semilla 12 que obtuvo el mejor Tardiness y la semilla 18 con el mejor Makespan respectivamente.

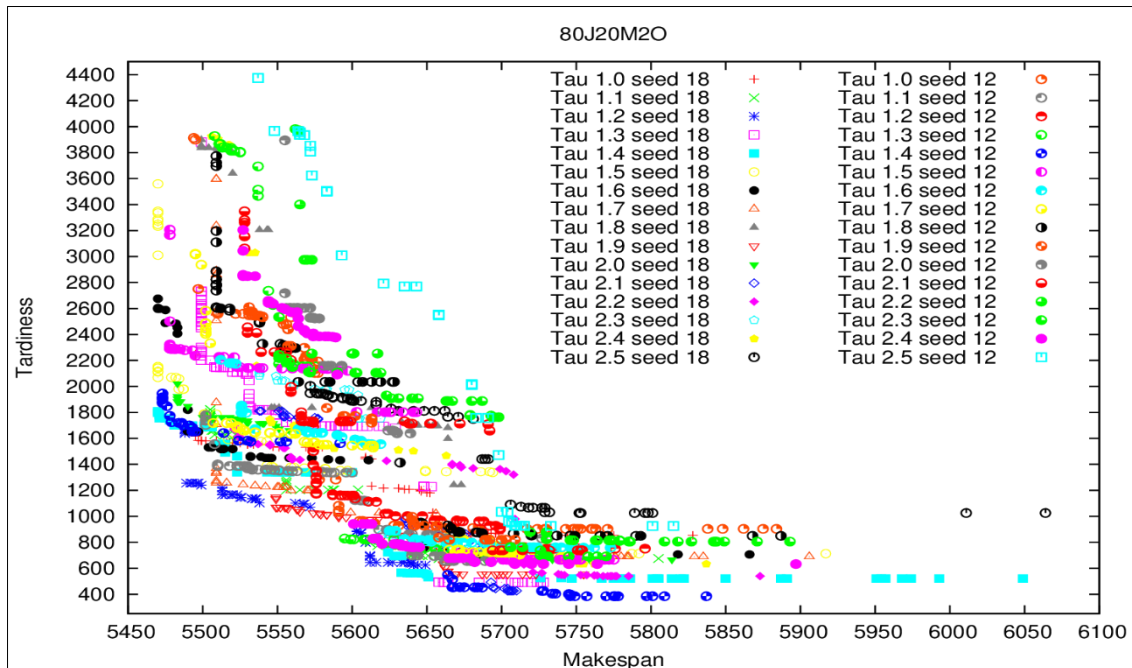


Figura 5.76 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 80-20 con semillas 18 y 12 y 1.000.000 de iteraciones

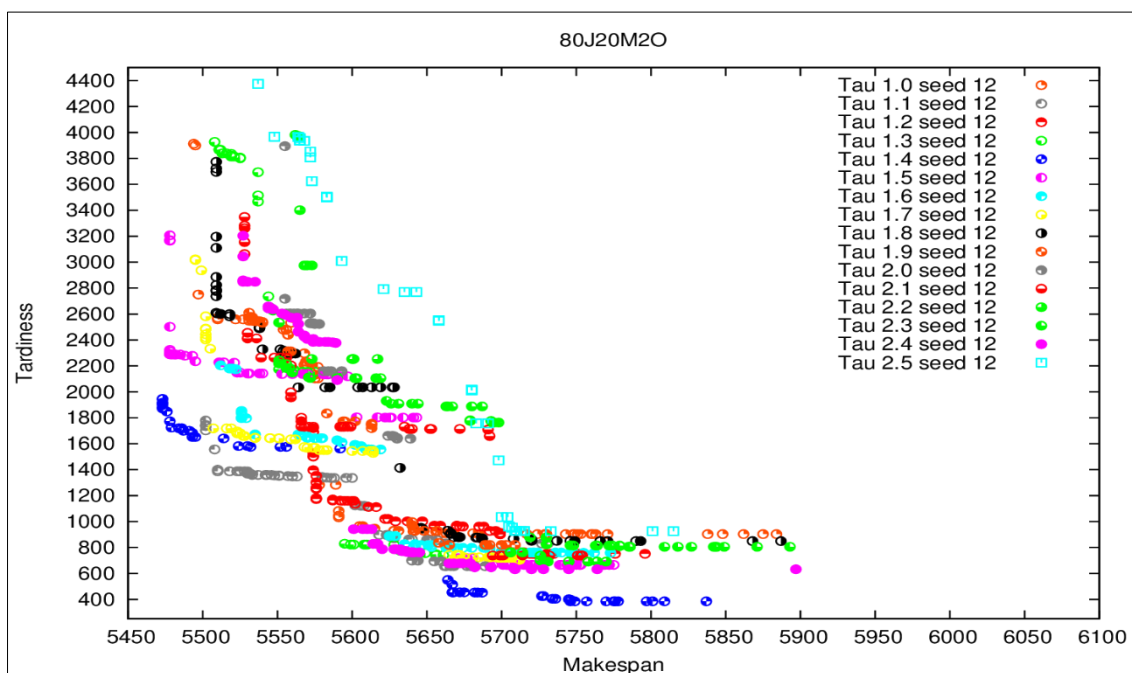


Figura 5.77 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 80-20 con semilla 12, mejor Tardiness, y 1.000.000 de iteraciones



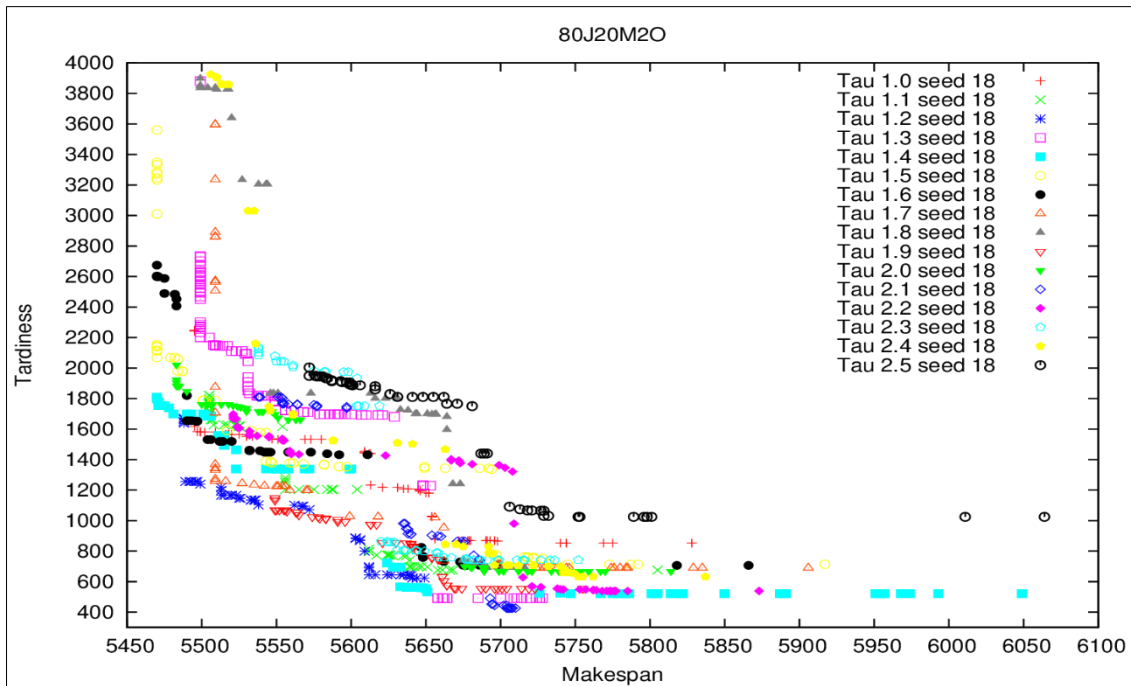


Figura 5.78 Variación de  $\tau$  para EO\_Feromonas\_Busqueda 80-20 con semilla 18, mejor Makespan, y 1.000.000 de iteraciones

En el análisis realizado a ambas semillas se encontraron los siguientes resultados:

- ✓ **Semilla 12:** Mejor Makespan para  $\tau = 1,4$  con valor 5473, mientras que el mejor Tardiness es también para  $\tau = 1,4$  con valor 385.
- ✓ **Semilla 18:** Mejor Makespan para  $\tau = 1,4; 1,5$  y  $1,6$  con 5470 la mejor solución la tiene  $\tau = 1,4$  con valor (5470, 1798), mientras que el mejor Tardiness es para  $\tau = 2,1$  con valor 426.

Semilla 12		Semilla 18			
$\tau = 1,4$		$\tau = 1,4$		$\tau = 2,1$	
5473	1878	5470	1809	5538	1810
5473	1944	5470	1798	5539	1810
5473	1916	5471	1754	5551	1810
5473	1899	5471	1779	5553	1810
5473	1946	5475	1754	5554	1766
5473	1912	5478	1739	5554	1802
5473	1872	5481	1699	5554	1779
5476	1848	5490	1699	5555	1766
5478	1773	5493	1699	5558	1764
5479	1724	5501	1699	5564	1764
5484	1719	5502	1694	5564	1759
5486	1719	5504	1685	5575	1759
5487	1700	5506	1685	5577	1755
5491	1700	5511	1558	5577	1747
5493	1683	5515	1492	5597	1736
5493	1652	5515	1558	5597	1746

5495	1652	5523	1464	5635	981
5514	1641	5523	1339	5636	981
5524	1582	5543	1339	5637	947
5530	1582	5549	1339	5638	914
5532	1576	5550	1339	5638	943
5552	1576	5555	1339	5640	910
5556	1576	5556	1339	5640	911
5592	1562	5558	1338	5654	904
5664	550	5569	1338	5658	900
5667	514	5572	1338	5660	896
5667	454	5599	1338	5671	864
5668	454	5600	1338	5674	864
5672	454	5624	723	5676	864
5675	454	5628	695	5682	773
5682	453	5630	695	5686	740
5684	453	5633	695	5693	492
5687	452	5633	566	5694	453
5727	425	5637	562	5695	453
5728	425	5639	562	5696	444
5734	404	5643	562	5702	444
5736	404	5646	562	5704	444
5745	401	5647	560	5704	426
5746	398	5648	560	5705	426
5746	385	5650	553	5706	426
5749	385	5651	532	5707	426
5757	385	5726	524	5708	426
5770	385	5740	524	5710	426
5775	385	5747	521		
5776	385	5767	521		
5778	385	5775	521		
5797	385	5779	521		
5801	385	5783	521		
5809	385	5786	521		
5837	385	5801	521		
		5804	521		
		5805	521		
		5814	521		
		5820	521		
		5822	521		
		5850	521		
		5887	521		
		5891	521		
		5951	521		
		5954	521		
		5957	521		
		5968	521		
		5974	521		
		5993	521		
		6049	521		

Tabla 5.32

*Mejores  $\tau$  para semillas 12 y 18 en EO\_Feromonas\_Busqueda 80-20*

## 5.4 Análisis

En esta sección se mostrarán los gráficos correspondientes a los 4 set de pruebas en sus versiones con mejor Makespan y mejor Tardiness, comparando cada una de ellas con las referencias extraídas de la literatura.

### 5.4.1 Makespan

En el gráfico de la Figura 5.79 se muestran los mejores resultados en Makespan para el set de pruebas 20 Jobs y 20 Máquinas con las 4 metodologías utilizadas. Se comparan entre sí y con el *benchmark* de referencia perteneciente a Ishibushi (Ishibushi, Yoshida, & Murata, 2003). Se puede observar que el uso de feromonas hace que EO tenga un comportamiento más favorable tanto en su versión pura como con búsqueda local.

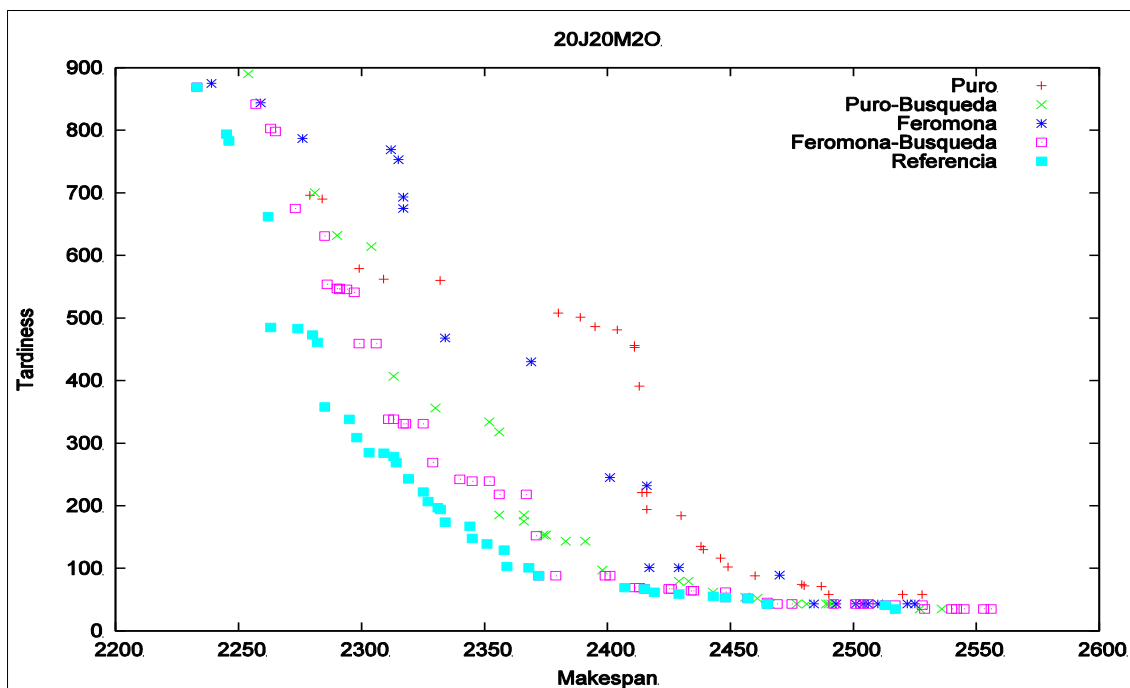


Figura 5.79 Gráfico de comparación Makespan 20-20 con mejor semilla, iteración y  $\tau$ , las 4 metodologías utilizadas y referencia

En el gráfico de la Figura 5.80 se muestran los mejores resultados en Makespan para el set de pruebas 40 Jobs y 20 Máquinas con las 4 metodologías utilizadas. Se comparan entre sí y con el *benchmark* de referencia perteneciente a Ishibushi (Ishibushi, Yoshida, & Murata, 2003). Nuevamente se nota que el uso de feromonas logra una mejora en el funcionamiento de EO.

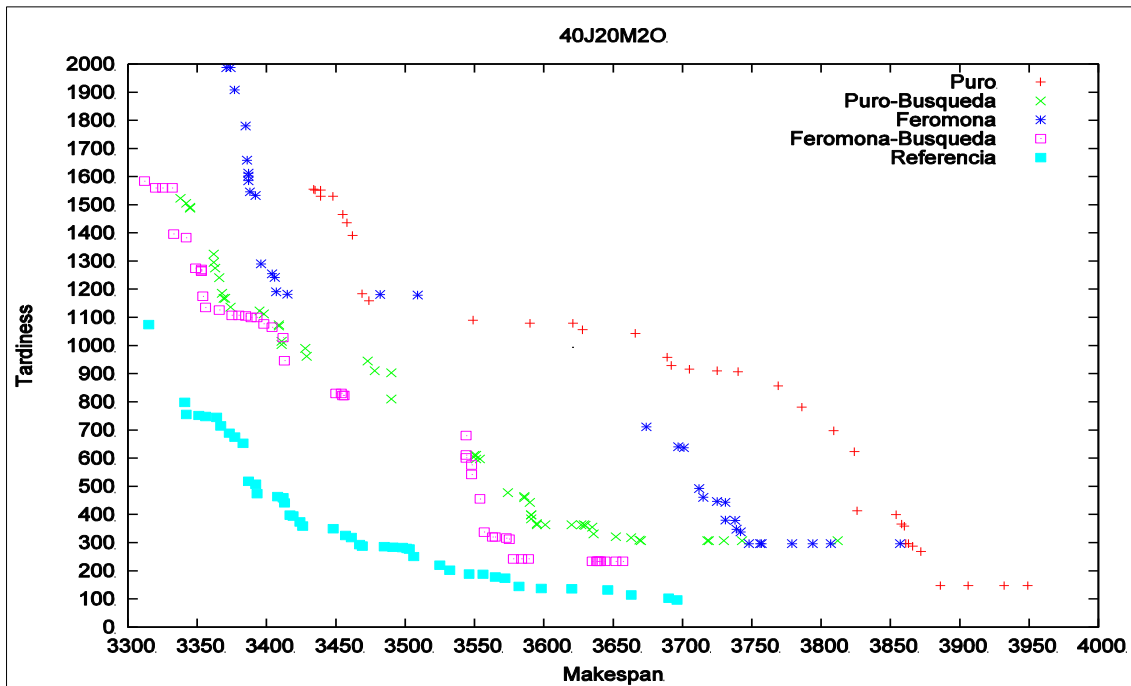


Figura 5.80 Gráfico de comparación Makespan 40-20 con mejor semilla, iteración y  $\tau$ , las 4 metodologías utilizadas y referencia

En el gráfico de la Figura 5.81 se muestran los mejores resultados en Makespan para el set de pruebas 60 Jobs y 20 Máquinas con las 4 metodologías utilizadas. Se comparan entre sí y con el *benchmark* de referencia perteneciente a Ishibushi (Ishibushi, Yoshida, & Murata, 2003). En este caso se observa una ambigüedad en las metodologías, es decir, en favor de Makespan son bastante similares, pero en Tardiness los datos de EO puro son mejores que con el uso de Feromona.

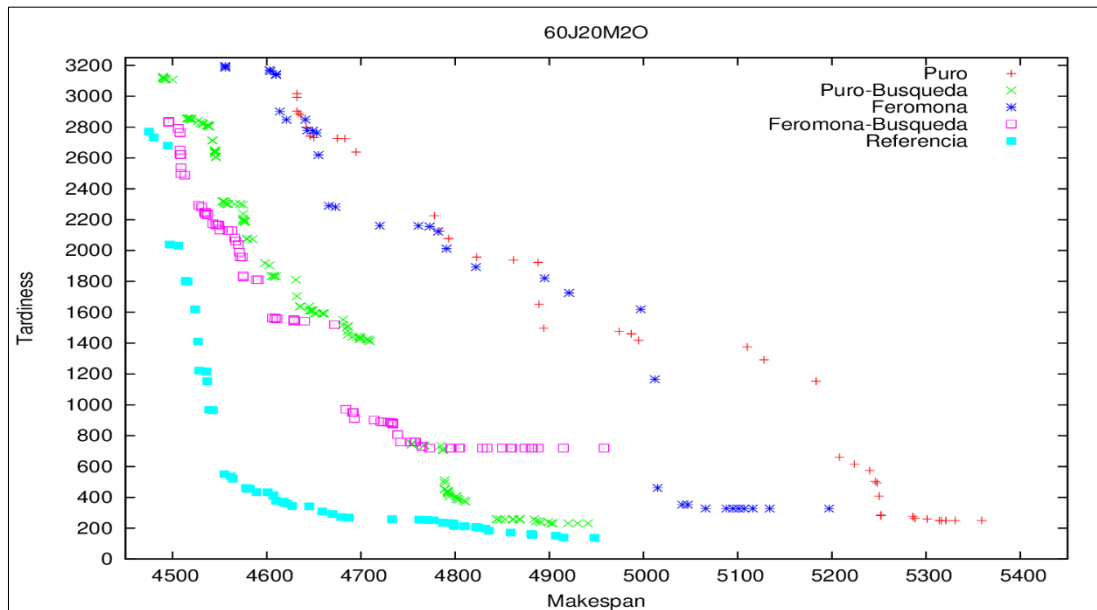


Figura 5.81 Gráfico de comparación Makespan 60-20 con mejor semilla, iteración y  $\tau$ , las 4 metodologías utilizadas y referencia

En el gráfico de la Figura 5.82 se muestran los mejores resultados en Makespan para el set de pruebas 80 Jobs y 20 Máquinas con las 4 metodologías utilizadas. Se comparan entre sí y con el *benchmark* de referencia perteneciente a Ishibushi (Ishibushi, Yoshida, & Murata, 2003). Acá se observa la diferencia entre el set de referencia y las metodologías utilizadas en esta investigación.

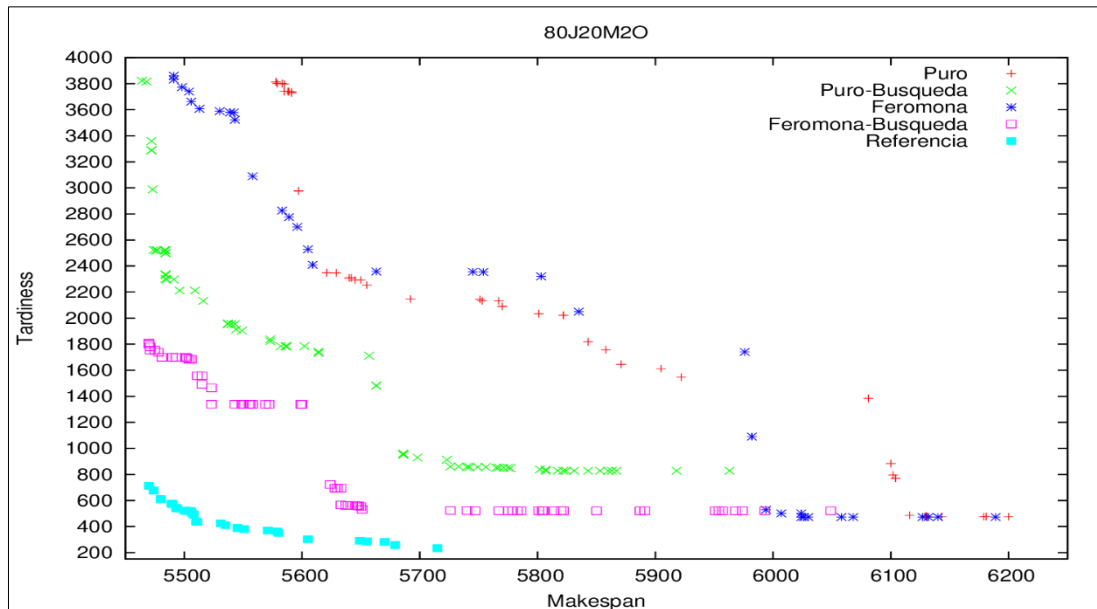


Figura 5.82 Gráfico de comparación Makespan 80-20 con mejor semilla, iteración y  $\tau$ , las 4 metodologías utilizadas y referencia

## 5.4.2 Tardiness

En el gráfico de la Figura 5.83 se muestran los mejores resultados en Tardiness para el set de pruebas 20 Jobs y 20 Máquinas con las 4 metodologías utilizadas. Se comparan entre sí y con el *benchmark* de referencia perteneciente a Ishibushi (Ishibushi, Yoshida, & Murata, 2003). Para este caso se observa que el uso de feromonas con búsqueda ha logrado igualar los extremos de la referencia.

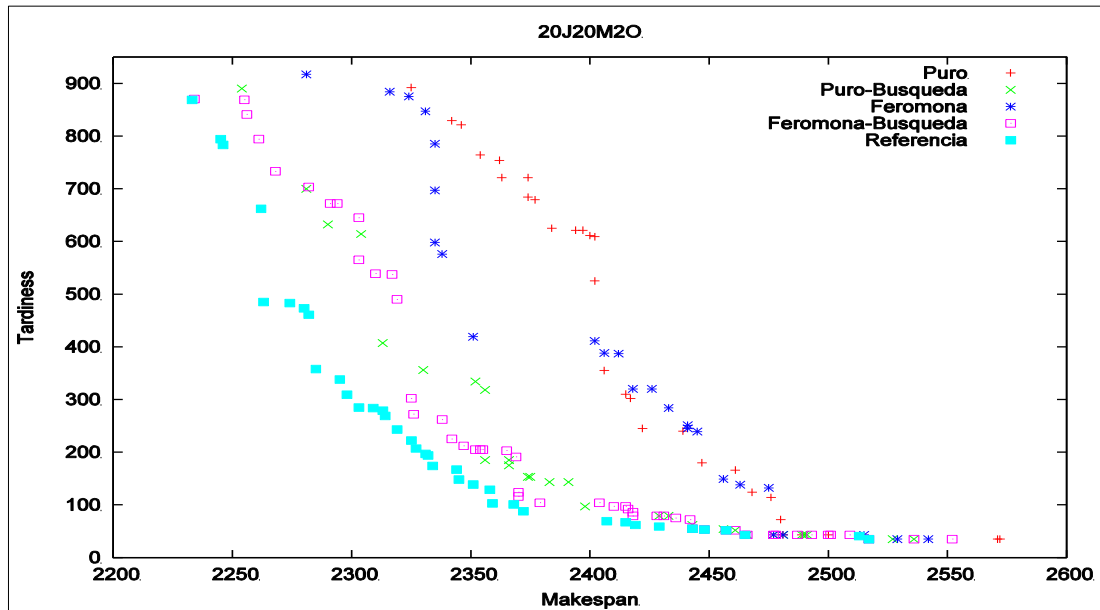


Figura 5.83 Gráfico de comparación Tardiness 20-20 con mejor semilla, iteración y  $\tau$ , las 4 metodologías utilizadas y referencia

En el gráfico de la Figura 5.84 se muestran los mejores resultados en Tardiness para el set de pruebas 40 Jobs y 20 Máquinas con las 4 metodologías utilizadas. Se comparan entre sí y con el *benchmark* de referencia perteneciente a Ishibushi (Ishibushi, Yoshida, & Murata, 2003). Búsqueda puro y con feromonas tienen comportamientos similares en los extremos.

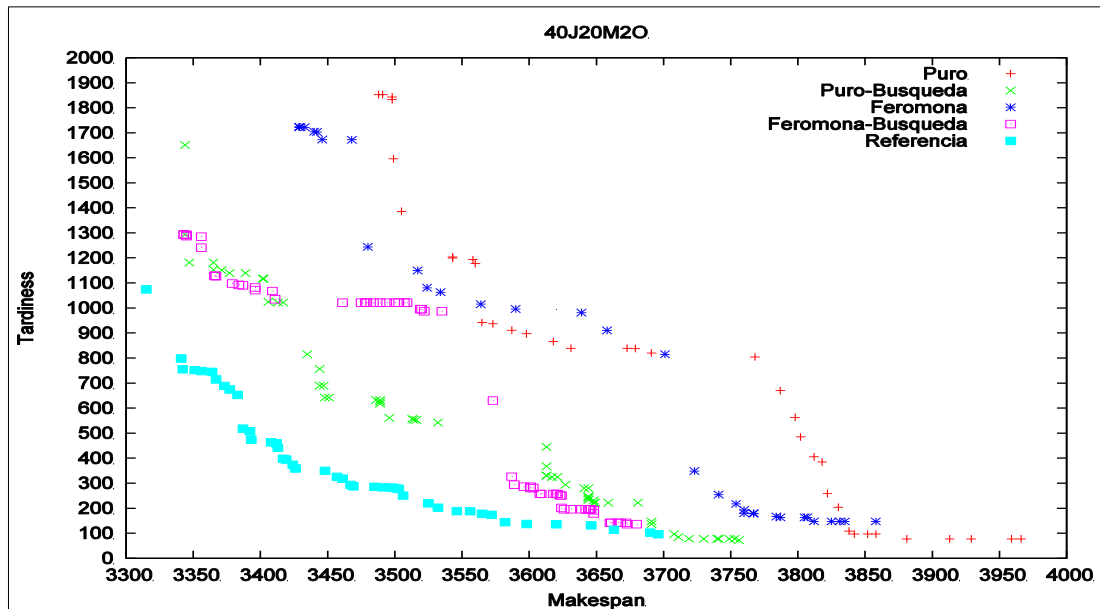


Figura 5.84 Gráfico de comparación Tardiness 40-20 con mejor semilla, iteración y  $\tau$ , las 4 metodologías utilizadas y referencia



En el gráfico de la Figura 5.85 se muestran los mejores resultados en Tardiness para el set de pruebas 60 Jobs y 20 Máquinas con las 4 metodologías utilizadas. Se comparan entre sí y con el *benchmark* de referencia perteneciente a Ishibushi (Ishibushi, Yoshida, & Murata, 2003). Aquí se ve que feromona con búsqueda local tiende a igualar en sus extremos a la referencia, además el algoritmo de búsqueda local puro logra obtener mejores resultados en el Tardiness que el de referencia.

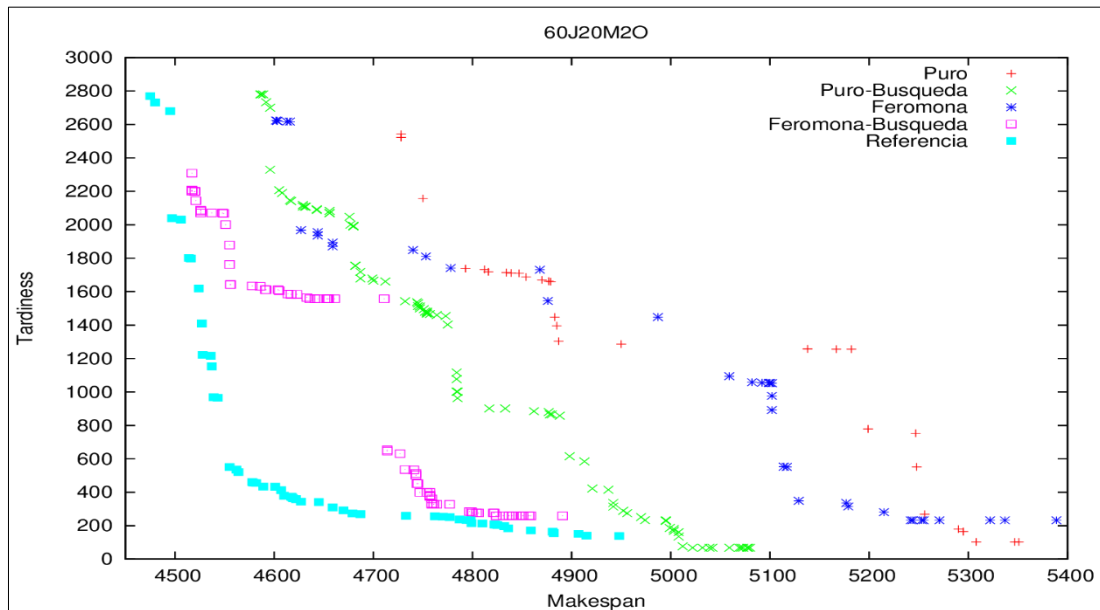


Figura 5.85 Gráfico de comparación Tardiness 60-20 con mejor semilla, iteración y  $\tau$ , las 4 metodologías utilizadas y referencia

En el gráfico de la Figura 5.86 se muestran los mejores resultados en Tardiness para el set de pruebas 80 Jobs y 20 Máquinas con las 4 metodologías utilizadas. Se comparan entre sí y con el *benchmark* de referencia perteneciente a Ishibushi (Ishibushi, Yoshida, & Murata, 2003). Acá se ve que feromona con búsqueda local tiene un comportamiento mucho mejor que el resto de metodologías, sin embargo, no logra igualar los resultados de referencia.

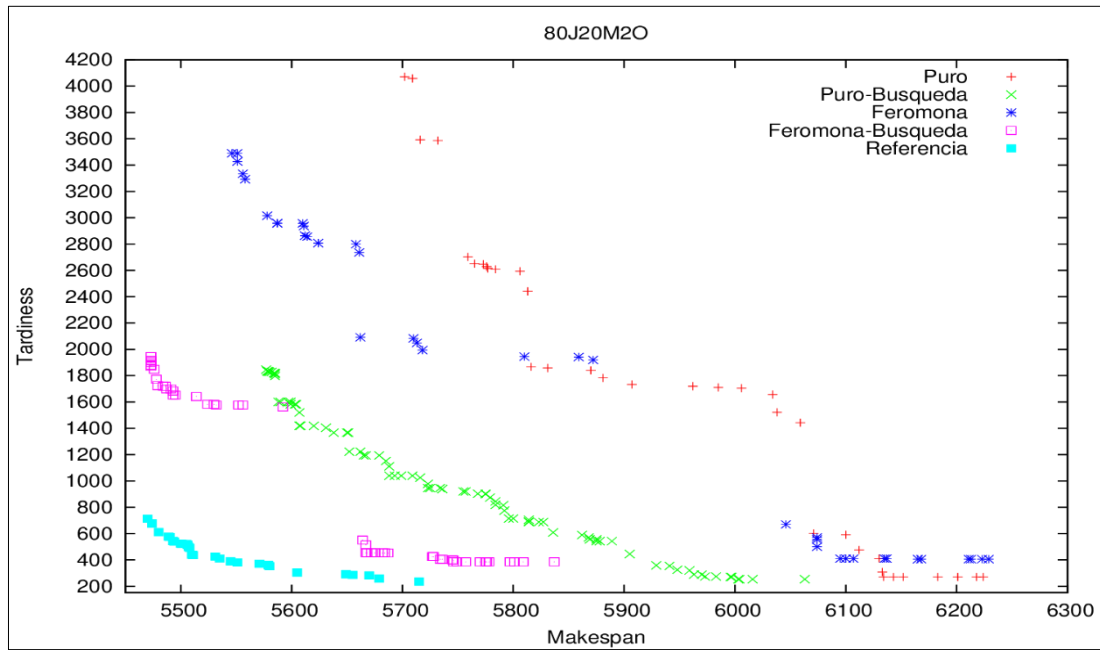


Figura 5.86 Gráfico de comparación Tardiness 80-20 con mejor semilla, iteración y  $\tau$ , las 4 metodologías utilizadas y referencia

### 5.4.3 Análisis de feromonas.

A continuación, se muestran cuatro formas diferentes de implementar las feromonas como método de selección para EO en esta investigación y sus correspondientes análisis gráficos.

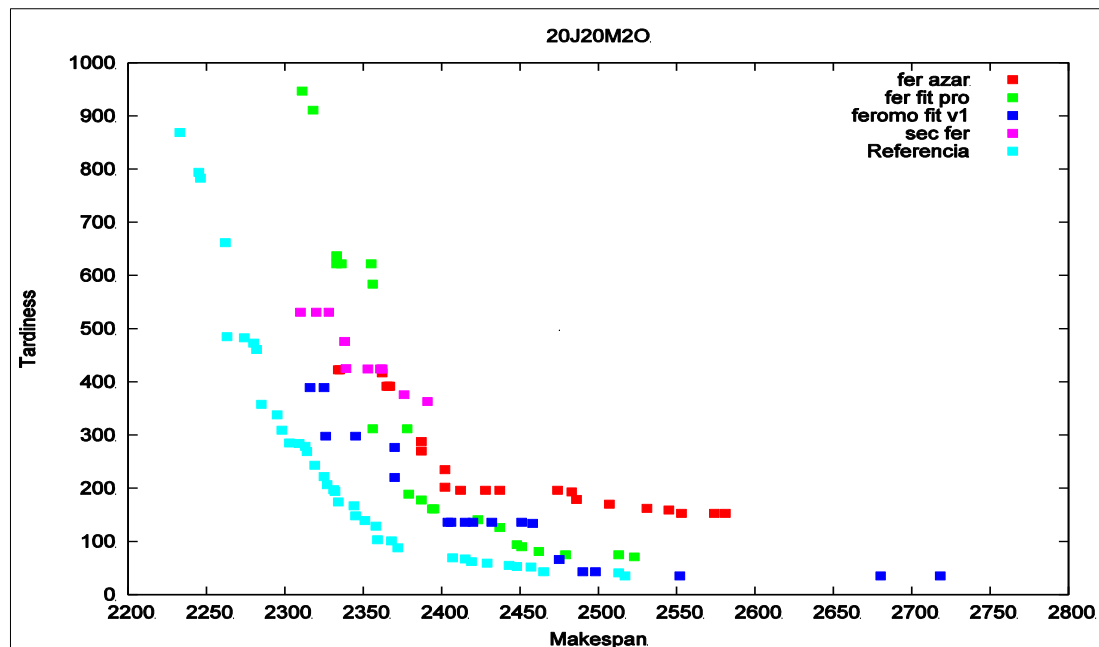


Figura 5.87 Análisis de feromona

Para escoger el par candidato, con feromona, a ser reemplazado se utilizaron varias técnicas que se describen a continuación todas ellas representadas en el gráfico de la Figura 5.87.

- i. **Fer\_azar:** Escoge un Job de la secuencia, al azar e igual probabilidad, con la feromona más baja según ruleta y el otro Job es escogido completamente al azar para la correspondiente permutación.
- ii. **Fer\_fit\_pro:** Escoge un Job de la secuencia con la feromona más baja, al azar e igual probabilidad, según ruleta y el otro Job es escogido del fitness denominado Fit\_pro que se explicará más abajo.
- iii. **Feromo\_fit\_V1:** Escoge un Job de la secuencia con la feromona más baja, al azar e igual probabilidad, según ruleta y el otro Job es escogido del fitness denominado Fit\_V1 que se explicará más abajo.
- iv. **Sec\_fer:** Escoge los dos Jobs en la secuencia, según feromonas y ruleta, y la permuta de posición.

**Fit\_pro:**

$$\text{Makespan}[i] = \text{Costo}[i][m - 1];$$

$$\text{Tardiness}[i] = \text{Costo}[i][m - 1] - \text{Due\_Date}[\text{array}[i]];$$

$$\text{Fitness\_Global}[i] = -\text{Makespan}[i] - \text{Tardiness}[i];$$

**Fit\_V1:**

$$\text{Makespan}[i] = T\_x\_Maq[\text{array}[i]] - \text{Costo}[i][m - 1];$$

$$\text{Tardiness}[i] = \text{Due\_Date}[\text{array}[i]] - \text{Costo}[i][m - 1];$$

$$\text{Fitness\_Global}[i] = \text{Makespan}[i] + \text{Tardiness}[i];$$

#### 5.4.4 Dos métodos de Búsqueda Local

En las siguientes gráficas se muestran dos métodos de búsqueda local probados en conjunto con el “Catalizador” implementado en este trabajo.

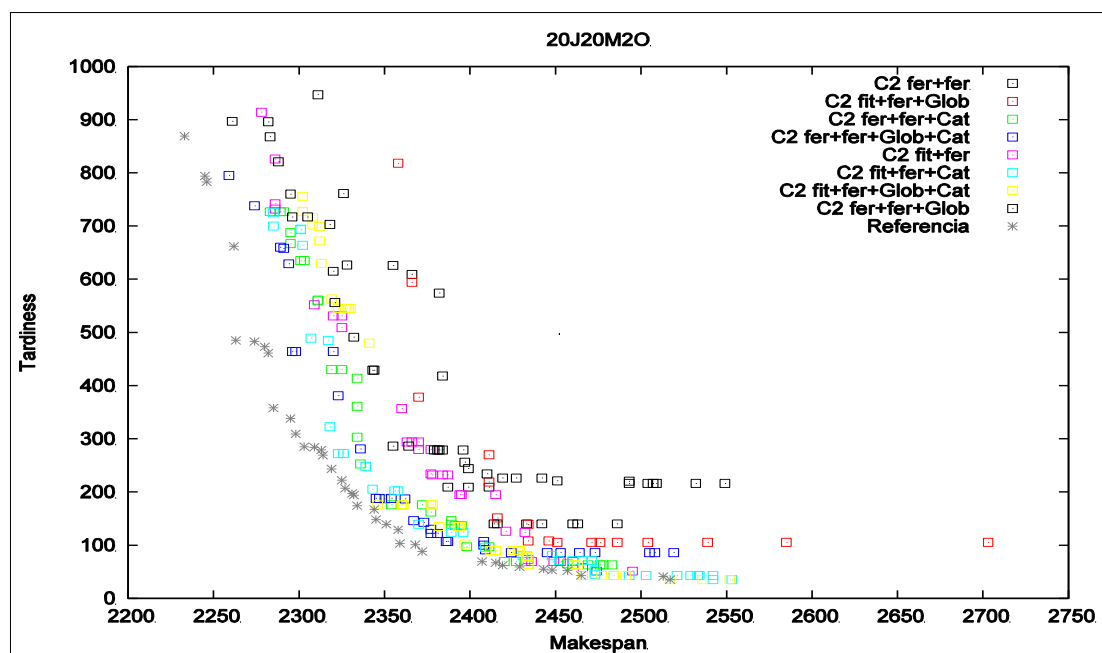


Figura 5.88 Gráfico de comparación con búsqueda llamada Combinación 2 con y sin catalizador

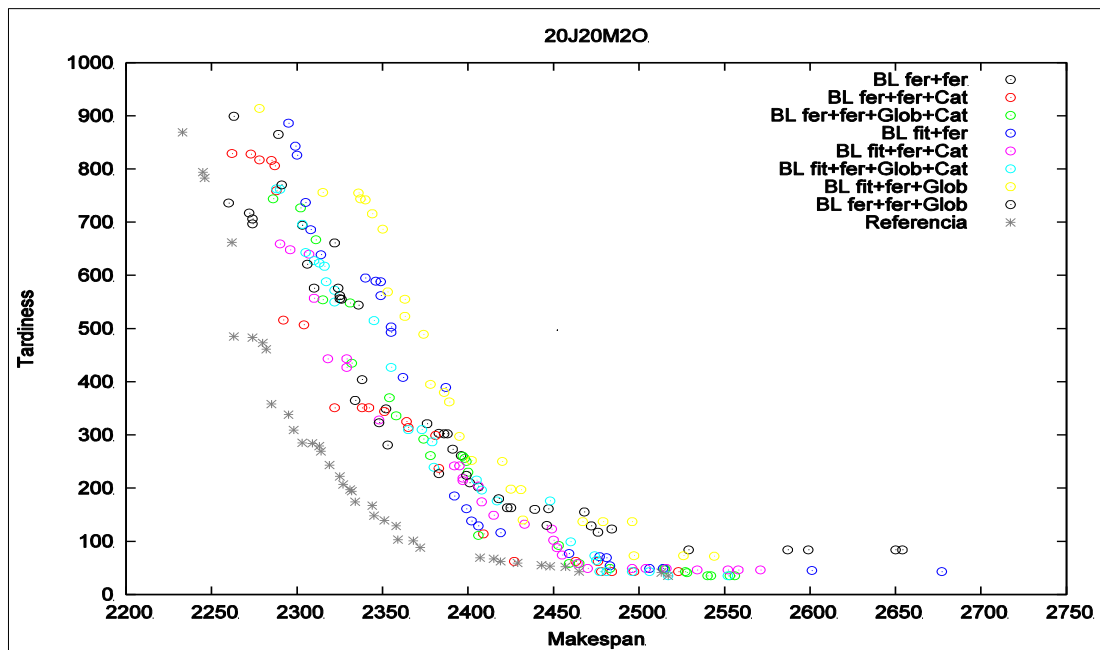


Figura 5.89 Gráfico de comparación con búsqueda llamada *Búsqueda\_L* con y sin catalizador

De la figura 5.4.2 y 5.4.3:

- **BL** y **C2** hacen referencia al tipo de búsqueda utilizado (BL=Búsqueda\_L y C2=Combinación 2).
- **Cat** hace referencia al uso del catalizador.
- **fit+fer** representan el criterio utilizado en la selección de los Jobs a permutar, es decir, se utilizó la feromona para escoger a uno de la secuencia y el fitness para escoger al otro.
- **Glob** se refiere a utilizar la feromona con evaporación.

- **Búsqueda\_L**: Al vector actual se le analiza de la siguiente forma: se permuta el Job en la posición 1 con cada uno de los Jobs en las otras posiciones. Luego al Job en la posición 2 con las posiciones restantes y así sucesivamente, analizando cada permutación para añadir soluciones al frente de Pareto.
- **Combinación2**: El vector actual se “cruza” con cada vector en el frente de Pareto creando “hijos” que son analizados e ingresados o rechazados en el frente de Pareto.
- **Catalizador**: De forma un tanto elitista se restringe la cantidad de veces que no se añade una solución al frente de Pareto. Cuando se llega a este límite, el vector actual es reemplazado por la mejor solución, hasta ese momento, en el frente de Pareto. Sin embargo, como el problema es de 2 objetivos, se opta por crear dos vectores actuales, es decir, un vector designado para Makespan y otro para Tardiness.

---

Finalmente se opta por utilizar Combinacion2 pues se ve en la gráfica que éste se aproxima más a los resultados de referencia.

---

## 6 Conclusión

Finalmente, luego de un extendido análisis de muchos datos se refleja que, aplicar feromona a EO es un método favorable en el caso de problemas donde encontrar una función de aptitud apropiada resulte complicado. Como se visualizó anteriormente, y comparando, tanto en su forma pura como con búsqueda local, los resultados mejoraron o a lo menos igualaron los del fitness propio. Además, se debe añadir que, aplicar feromona con evaporación resulta más conveniente. Estas conclusiones se respaldan con los gráficos de comparación en la sección 5.4 donde se refleja un mayor acercamiento a los resultados de referencia.

Analizando el caso particular de EO\_Búsqueda con los datos 60-20 en que mejoró los resultados de Tardiness encontrados en el *benchmark* correspondiente no se logra formular alguna explicación convincente al respecto, pues se utilizó las mismas funciones principales en todas las pruebas y sólo acá mejoraron.

Como se explicó en el capítulo 4, la selección de los individuos a permutar está dada tanto por la feromona como el fitness creado en esta investigación, por lo que este último es una variable que depende completamente de quien esté a cargo de la investigación. Es decir, es un factor que podría mejorar o arruinar los resultados finales.

### 6.1 Contribuciones

En la investigación se ha utilizado un método novedoso denominado EO e intentando dar una solución al principal inconveniente que presenta este algoritmo en su implementación (la búsqueda apropiada de un método de selección de acuerdo a los distintos factores presentes en los problemas a resolver) utilizando para ello las Feromonas. Éstas podrían funcionar como método estándar para la implementación en problemas con variados factores a considerar, como el visto en esta investigación, y así otorgar una idea generalizada de método de selección o bien un indicio para encontrar la función de selección más apropiada al problema.

### 6.2 Trabajo Futuro

Se consideraron los siguientes puntos.

- 
- En un futuro se podrá trabajar con el algoritmo sin limitarlo a los extremos (Makespan y Tardiness) como se realizó en esta investigación al utilizar el Catalizador y comprobar así los resultados, así se logrará una búsqueda menos elitista y con más resultados centrales lo que conlleva a un gráfico más suave en su curva.
  - Variar el valor de alpha ( $\alpha$ ) presente en las feromonas, acá se trabajó con valor 0,1 por lo que sería útil considerar variados valores y observar el comportamiento de los resultados.



---

## 7 Bibliografía

- Agarwal, A., Colak, S., & Eryarsoy, E. (2006). Improvement heuristic for the flow - shop scheduling problem. *European Journal of Operational Research*, 801-815.
- Angel A. Juan, H. R. (2012). *Combinando Randomización Sesgada y Búsqueda Local Iterativa para Resolver Problemas de Flow-Shop*. Albacete.
- Bak, P., & Sneppen, K. (1993). Punctuated Equilibrium and Criticality in a Simple Model of Evolution. *Physical Review Letters*, Volumen 7, Número 24.
- Boettcher, S., & Percus, A. G. (1999). *Extremal Optimization: Methods derived from Co-Evolution*. Orlando: GECCO 1999.
- Cook, S. A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the 3rd ACM Symposium Theory of Computing*, 151-158.
- Duarte Muñoz, A., Pantrigo Fernández, J. J., & Gallego Carrillo, M. (2007). *Metaheurísticas (pág. 114-120)*. Madrid: Dykinson S.L.
- Duch, J., & Arenas, A. (2008). *Community detection in complex networks using Extremal Opt.*
- Gómez Meneses, P. S. (2012). *Extremal Optimisation Applied to Constrained Combinatorial Multi-Objective Optimisation Problems*. Bond University.
- Gómez Meneses, P., Randall, M., & Lewis, A. (2010). *A hybrid multi-objective extremal optimisation approach for multi-objective combinatorial optimisation problems*. Barcelona.
- Ishibushi, H., Yoshida, T., & Murata, T. (2003). *Balance between genetic search local search in memetic algorithms for multi-objective Permutation flow-shop scheduling*. IEEE Computational Intelligence Society.
- Jiménez Morales, Á. P. (2012). *SOLUCIÓN DEL PROBLEMA DE PROGRAMACIÓN DE FLOW-SHOP FLEXIBLE EMPLEANDO EL ALGORITMO GENÉTICO DE CHU-BEASLEY*. Pereira: UNIVERSIDAD TECNOLÓGICA DE PEREIRA.
- Kacem, I. H. (2002). *Pareto-optimality Approach for Flexible Job-shop Scheduling Problems: Hybridization of Evolutionary Algorithms and Fuzzy Logic*. *Mathematics and Computers in Simulation*.
- Karp, R. M. (1972). Reducibility among Combinatorial Problems. En R. E. Miller, J. W. Thatcher, & J. D. Bohlinger, *Complexity of Computer Computations* (págs. 85-103). Berkeley: The IBM Research Symposia Series.
- Li, L., Peng, H., Kurths, J., Yang, Y., & Schellnhuber, H. J. (2014). Chaos–order transition in foraging behavior of ants. *Proceedings of the National Academy of Sciences*, 8392–8397.

- 
- Lin, T.-C. C., & Hsiao-Jou. (s.f.). *Flexible Job Shop Scheduling using a Multiobjective Memetic Algorithm*. Taiwan.
- Mailing, G. (2003). *Algoritmos heurísticos y el problema de job shop scheduling*. Buenos Aires.
- Merz, P., & Freisleben, V. (1997). Genetic local search for the TSP: New results. *Proc. of 4th IEEE International Conference on Evolutionary Computation*, 159-164.
- Nawaz, M., Ensco Jr., E., & Ham, I. (1983). A heuristic algorithm for the m - machine, n - job flow - shop sequencing problem. *Omega 11*, 91-95.
- Peña, V., & Zunelzu, L. (17 de Mayo de 2006). *Estado del Arte del Job Shop Scheduling Problem*. Valparaíso: Universidad Técnica Federico Santa María.
- Randall, M., Lewis, A., Hettenhausen, J., & Kipouros, T. (2014). Local Search Enabled Extremal Optimization for Continuous Inseparable Multi-objective Benchmark and Real-World Problems. *Procedia Computer Science Volume 29*, 1904-1914.
- Rodríguez Quiñones, T. A. (2014). *Solución de problemas tipo Flow-Shop mediante algoritmos evolutivos*. Bogotá: Universidad Nacional de Colombia.
- Sadjadi, S. J., Aryanezhad, M. B., & Ziaee, M. (2008). The General Flowshop Scheduling Problem: Mathematical Models. *Journal of Applied Sciences*, 3032-3037.
- Saghafian, S., & Hejazi, R. S. (2005). *Flowshop-scheduling problems with makespan criterion: a review*. Isfahan, Tehran: International Journal of Production Research.
- Salazar Hornig, E., & Pavón Weber, N. (2011). Aplicación de un algoritmo ACO al problema de taller de flujo de permutación con tiempos de preparación dependientes de la secuencia y minimización de makespan. *Revista chilena de ingeniería, vol. 19 N° 2*, 253-264.
- Schaffer, J. D. (1985). Multiple objective optimization with vector evaluated genetic algorithms. *Proc. of 1st International Conference on Genetic Algorithms and Their Applications*, 93-100.
- Šeda, M. (2007). *Mathematical Models of Flow Shop and Job*. World Academy of Science, Engineering and Technology.
- Srinivas, N., & Deb, K. (1995). Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation 2*, 221-248.
- Toro Ocampo, E. M., Restrepo Grisales, Y. S., & Granada Echeverr, M. (2006). ALGORITMO GENETICO MODIFICADO APLICADO AL PROBLEMA DE SECUENCIAMIENTO DE TAREAS EN SISTEMAS DE PRODUCCION LINEAL – FLOW SHOP. *Scientia Et Technica*, 285-290.
- Zitzler, E., & Thiele, L. (1998). *An evolutionary algorithm for multiobjective optimization*. Zurich: Swiss Federal Institute of Technology.