

Un Sistema Inmune Artificial para Solucionar Problemas de Optimización Multiobjetivo

Nareli Cruz Cortés y Carlos A. Coello Coello
Grupo de Computación Evolutiva
Departamento de Ingeniería Eléctrica
Sección de Computación
Av. Instituto Politécnico Nacional No. 2508
Col. San Pedro Zacatenco
México D.F. 07300, MÉXICO
nareli@computacion.cs.cinvestav.mx
ccoello@cs.cinvestav.mx

Resumen

El sistema inmune ha servido como inspiración para solucionar exitosamente problemas del mundo real de diverso tipo. Sin embargo, el uso de sistemas inmunes artificiales para resolver problemas multiobjetivo ha sido un tema muy poco abordado en la literatura. En este artículo se propone el uso de una emulación del sistema inmune para resolver problemas de optimización multiobjetivo (con o sin restricciones), la cual se basa en el principio de selección clonal.

1. Introducción

La mayoría de los problemas de optimización del mundo real suelen tener varios objetivos que deben satisfacerse simultáneamente, los cuales generalmente están en conflicto entre sí. Las técnicas poblacionales han demostrado tener éxito en la solución de éstos, debido principalmente a que pueden encontrar varias soluciones en una sola ejecución, en vez de una a la vez como es en el caso de las técnicas que han sido desarrolladas por los estudiosos del área de investigación de operaciones.

El sistema inmune ha servido como inspiración para solucionar problemas complejos de ingeniería y la ciencia con gran éxito, debido principalmente a que es un sistema de aprendizaje distribuido con interesantes características.

En este trabajo se propone un algoritmo para solucionar problemas de optimización multiobjetivo usando un modelo inspirado en el sistema inmune.

2. Definición del problema

El problema que deseamos solucionar, es el general de optimización multiobjetivo y se define como sigue [2]:

Encontrar el vector $\vec{x}^* = [x_1^*, x_2^*, \dots, x_n^*]^T$ tal que satisfaga las m restricciones de desigualdad

$$g_i(\vec{x}) \geq 0 \quad i = 1, 2, \dots, m \quad (1)$$

las p restricciones de igualdad

$$h_i(\vec{x}) = 0 \quad i = 1, 2, \dots, p \quad (2)$$

y optimiza el vector de funciones

$$\vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_k(\vec{x})]^T \quad (3)$$

donde $\vec{x} = [x_1, x_2, \dots, x_n]^T$ es el vector de variables de decisión.

Otro concepto importante es el de optimalidad de Pareto [2], el cual fue propuesto por Vilfredo Pareto [1] en el siglo XIX y constituye el origen del área de optimización multiobjetivo. Formalmente se define como sigue:

Decimos que un punto $\vec{x}^* \in \Omega$ (Ω es la zona factible) es un **óptimo de Pareto** si para toda $\vec{x} \in \Omega$ e $I = \{1, 2, \dots, k\}$,

$$\forall_{i \in I} (f_i(\vec{x}) = f_i(\vec{x}^*)) \quad (4)$$

o, hay al menos una $i \in I$ tal que

$$f_i(\vec{x}) > f_i(\vec{x}^*) \quad (5)$$

$$\Omega \subseteq S = \mathbb{R}^n \rightarrow \mathbb{R}, \Omega \neq \emptyset$$

3. Sistema Inmune Artificial

Una de las principales tareas del sistema inmune es mantener al organismo sano. Algunos microorganismos (llamados patógenos) que invaden al organismo pueden resultar dañinos para éste. Los *antígenos* son moléculas que se encuentran expresadas en la superficie de los patógenos que pueden ser reconocidos por el sistema inmune y que además son capaces de dar inicio a la respuesta inmune para eliminarlos.

Esta respuesta defensiva del sistema inmune presenta interesantes características desde el punto de vista del procesamiento de información. Es por ello que se ha usado como inspiración para crear soluciones alternativas a problemas complejos de ingeniería y la ciencia. Esta es un área relativamente nueva a la cual se le llama *sistema inmune artificial* [8].

Principio de selección clonal

Uno de los mecanismos del sistema inmune que intenta explicar el proceso por el cual se combate contra un antígeno es el llamado *principio de selección clonal*, el cual establece la idea de que sólo aquellas células inmunes (*linfocitos B*) que mejor reaccionen ante el estímulo de un antígeno se clonarán. A continuación se describe de manera muy simplificada [8] este principio:

Cuando se ha detectado la presencia de algún invasor, aquellos *linfocitos* con afinidad más alta con respecto al invasor, sufren un proceso de clonación seguido de una mutación de gran magnitud o hipermutación. Después de esta clonación y mutación los linfocitos se convierten en grandes productores de anticuerpos que eliminarán y neutralizarán a los antígenos invasores.

El exceso de células es eliminado. Sin embargo, una parte de ellas permanece en el organismo circulando por un largo periodo como células de memoria, de manera que, cuando se presenta el mismo tipo de antígeno o uno similar, la respuesta del sistema inmune es más rápida y eficiente. A esta respuesta se le llama *respuesta secundaria*.

4. Algoritmo propuesto

1. La población inicial se crea de manera que los individuos estén distribuidos a lo largo del espacio de búsqueda de manera uniforme.
2. Inicializar la memoria secundaria (externa) de manera tal que esté vacía.
3. Determinar para cada individuo de la población, si es dominado (en el sentido de Pareto) o no y, para el caso de problemas con restricciones, determinar si es factible o no.
4. Determinar cuáles son los mejores linfocitos, de acuerdo al siguiente criterio:
Para un espacio sin restricciones: Todos los individuos no dominados son los mejores.
Para un espacio restringido:
 - Si existen individuos factibles: Se selecciona a los no dominados (con respecto al conjunto de individuos factibles)
 - Si no existe ningún individuo factible: Todos los individuos no dominados.
5. Copiar a los mejores linfocitos (resultantes del paso anterior) en la memoria secundaria.
6. Se determina para cada “mejor linfocito” la cantidad de clones que se deben crear, de manera tal que el total de clones sea igual al 60% de la población. Además, si la memoria secundaria está saturada, se aplica el siguiente criterio:
 - Si al individuo no se le permitió ingresar a la memoria secundaria por ser repetido o pertenecer a la celda más saturada, entonces se crean cero clones.

- Si el individuo pertenece a una celda de la memoria secundaria, tal que su población es menor al promedio (con respecto a la cantidad de individuos en cada celda de la memoria secundaria), entonces la cantidad de clones se duplica.
 - Si el individuo pertenece a una celda con población mayor al promedio (con respecto a todas las celdas), entonces la cantidad de clones es dividida entre 2.
7. Se realiza el proceso de clonación de los mejores linfocitos de acuerdo a la información creada en el paso anterior.
 8. Se aplica el operador de mutación a los clones. El número de genes que se mutan de cada cadena cromosómica, es igual a la cantidad de variables existentes.
 9. A aquellos linfocitos que no resultaron ser los mejores, se les aplica el operador de mutación uniforme. El porcentaje de mutación inicial es alto y se va decrementando a lo largo de las generaciones de manera lineal (desde 0.9 hasta 0.3)
 10. Si la memoria secundaria está saturada, se aplica cruza a una fracción de ella (cruza de dos puntos). Los individuos resultantes que sean no dominados ingresan a la memoria.
 11. El tamaño de la población debe regresarse a su valor original, de manera que se eliminan a los individuos excedentes, dando preferencia de sobrevivir a los nuevos clones.
 12. Se repite el proceso desde el paso 3 un número predeterminado de veces.

La selección de los parámetros de los pasos 6 y 9 del algoritmo fueron determinados de manera experimental.

4.1. Memoria secundaria

Usamos una memoria secundaria o externa como mecanismo de elitismo con la finalidad de mantener las mejores soluciones a lo largo del proceso. Los individuos almacenados en esta memoria son no dominados entre ellos, y además con respecto a individuos previos que intentaron ingresar a la memoria. De manera que, esta memoria secundaria almacena el conjunto solución de nuestro algoritmo al problema.

Como la memoria secundaria no puede crecer de manera indefinida, puede llegar un momento en que se sature, y existan individuos que intentan ingresar a ella. En tal situación es necesario determinar si se les permite entrar o no, de manera que únicamente se permitirá el ingreso a aquellos individuos que de alguna manera mejoren la información contenida en la memoria. Dado que todos los elementos son no dominados, entonces se debe considerar otro criterio de ingreso, que en este caso es mejorar la distribución de las soluciones a lo largo del frente de Pareto. Para ello usamos la rejilla adaptativa propuesta por Knowles y Corne [4], la cual divide el espacio de la memoria secundaria por regiones de densidad.

5. Experimentos

Nuestro algoritmo multiobjetivo basado en un sistema inmune artificial (SIAM) fue validado usando un conjunto de funciones de prueba que son comúnmente adoptadas por los investigadores del área de optimización multiobjetivo; en este trabajo se reportan los resultados obtenidos para tres de ellas debido a restricciones de espacio.

El algoritmo (SIAM) se comparó contra dos algoritmos representativos del área que son: NSGA-II [3] y PAES [4]. Se realizaron 20 corridas de cada función para cada uno de los algoritmos. La comparación se efectuó utilizando tres métricas: dispersión, tasa de error y distancia generacional. Además se realizó una inspección visual de las soluciones graficando el verdadero frente de Pareto (obtenido por enumeración) contra la solución obtenida por los diferentes algoritmos (la que se encuentra en la media de acuerdo a la métrica de distancia generacional):

Dispersión (D): Esta métrica fue propuesta por Schott [5], y mide la varianza de la distancia de cada miembro del conjunto de óptimos de Pareto con respecto a su vecino más cercano:

$$D \leq \sqrt{\frac{1}{n-1} \sum_{i=1}^n (\bar{d} - d_i)^2}, \quad (6)$$

donde $d_i = \min_j (|f_1^i(\vec{x}) - f_1^j(\vec{x})| + |f_2^i(\vec{x}) - f_2^j(\vec{x})|)$, $i, j = 1, \dots, n$, \bar{d} es el promedio de todos d_i , y n es el número de elementos del conjunto de Pareto obtenidos hasta el momento. Si $D = 0$ significa que nuestro algoritmo ha encontrado la distribución ideal de vectores no dominados.

Tasa de error (TE): Propuesta por Van Veldhuizen [6] para indicar el porcentaje de soluciones del frente Pareto actual (FP_{actual}) que no son miembros del verdadero frente de Pareto ($FP_{verdadero}$):

$$TE = \frac{\sum_{i=1}^n e_i}{n} \quad (7)$$

donde n es el número de vectores en FP_{actual} , $e_i = 0$ si el vector i es un miembro de $FP_{verdadero}$ y $e_i = 1$ de lo contrario. Un valor de $TE = 0$ indica el comportamiento ideal del algoritmo.

Distancia Generacional (DG): El concepto de distancia generacional fue introducido por Van Veldhuizen [6] como una manera de estimar qué tan lejos están los elementos FP_{actual} de $FP_{verdadero}$ y se define como:

$$DG = \frac{\sum_{i=1}^n d_i}{n} \quad (8)$$

donde n es el número de vectores no dominados encontrados por el algoritmo que está siendo analizado, y d_i es la distancia Euclidiana entre cada uno de ellos y el miembro más cercano del verdadero frente de Pareto. Si $DG = 0$ indica que todos los elementos generados están en el verdadero frente de Pareto de la función.

Ejemplo 1

El primer ejemplo es una función con 2 objetivos cuyo frente de Pareto es desconectado y formado por cuatro segmentos:

Minimizar: $F = (f_1(x, y), f_2(x, y))$, donde

$$f_1(x, y) = x,$$

$$f_2(x, y) = (1 + 10y) * [1 - (\frac{x}{1+10y})^\alpha - \frac{x}{1+10y} \sin(2\pi qx)]$$

$$\text{y } 0 \leq x, y \leq 1, q = 4, \alpha = 2.$$

La figura 1 muestra el verdadero frente de Pareto graficado contra la solución encontrada por SIAM, NSGA-II y PAES.

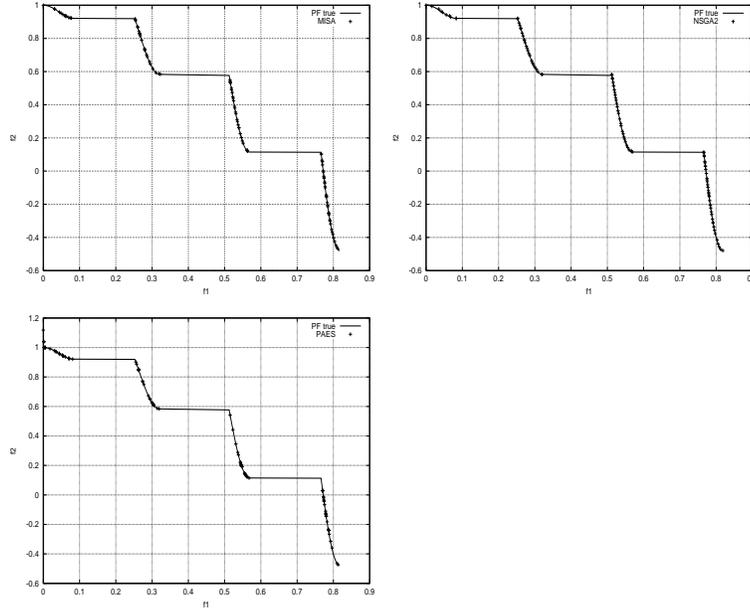


Figura 1: Frentes de Pareto obtenidos por SIAM, NSGA-II y PAES respectivamente, para el ejemplo 1.

Los resultados obtenidos por las métricas se muestran en las tablas 1 y 2 para la función 1. De acuerdo con la métrica de distribución, en promedio, los mejores resultados los obtienen SIAM y NSGA-II con valores muy similares cercanos a cero. El mejor valor para la métrica de distancia generacional lo obtiene SIAM, mientras que para la tasa de error el mejor desempeño lo tienen PAES y NSGA-II, quedando SIAM en tercer lugar. Gráficamente, SIAM y NSGA-II muestran una mejor aproximación del verdadero frente de Pareto.

Ejemplo 2

El segundo ejemplo es una función de prueba propuesta por Kita [7]:

	Distribución			DG		
	SIAM	NSGA-II	PAES	SIAM	NSGA-II	PAES
Media	0.06772	0.0671	0.0780	0.0002	0.0004	0.0030
Mejor	0.067339	0.0669	0.0257	0.0001	0.0002	0.0001
Peor	0.068934	0.06749	0.2587	0.0003	0.0044	0.0251
Dsv.Est.	0.0003	0.0001	0.0438	0.00002	0.0009	0.0075

Cuadro 1: Distribución y distancia generacional (DG) para el ejemplo 1

	SIAM	NSGA-II	PAES
Media	0.327228	0.1430695	0.1342229
Mejor	0.158416	0.0792080	0.0594060
Peor	0.495050	0.2871290	0.2673270
Dsv.Est.	0.086998	0.0498141	0.0522345

Cuadro 2: Tasa de error para el ejemplo 1

Maximizar $F = (f_1(x, y), f_2(x, y))$
donde:

$$\begin{aligned} f_1(x, y) &= -x^2 + y, \\ f_2(x, y) &= \frac{1}{2}x + y + 1 \end{aligned}$$

$$x, y \geq 0, 0 \geq \frac{1}{6}x + y - \frac{13}{2}, 0 \geq \frac{1}{2}x + y - \frac{15}{2}, 0 \geq 5x + y - 30.$$

La figura 2 muestra el verdadero frente de Pareto graficado contra la solución encontrada por SIAM, NSGA-II y PAES.

Los resultados obtenidos por las métricas se muestran en las tablas 3 y 4. Podemos ver en las gráficas que la aproximación al verdadero frente de Pareto de SIAM es mejor que NSGA-II y PAES, pues estos últimos pierden una buena fracción del mismo. PAES muestra un mejor desempeño claramente según la métrica de distribución. Para la distancia generacional, los mejores valores se obtuvieron por SIAM seguido por NSGA-II con valores muy aproximados, y con respecto a la tasa de error, los tres algoritmos muestran valores muy cercanos al cero, siendo en este caso NSGA-II el que mejor valor obtuvo.

Ejemplo 3

El tercer problema es una función de 2 objetivos, donde el frente de Pareto está dividido en 2 segmentos. Esta función fue propuesta por Schaffer [9]:

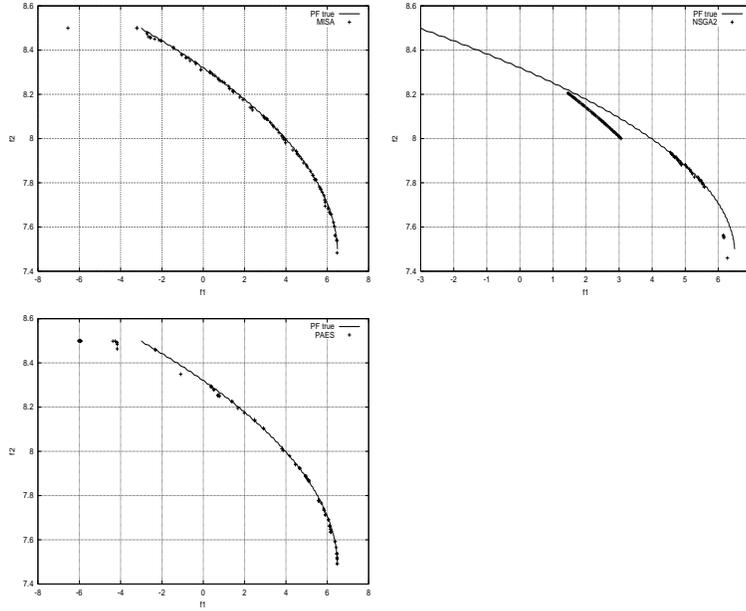


Figura 2: Frentes de Pareto obtenidos por SIAM, NSGA-II y PAES respectivamente, contra el verdadero frente de Pareto para el ejemplo 2

$$\text{Minimizar } f_1(x) = \begin{cases} -x & \text{if } x \leq 1 \\ -2 + x & \text{if } 1 < x \leq 3 \\ 4 - x & \text{if } 3 < x \leq 4 \\ -4 + x & \text{if } x > 4 \end{cases} \quad (9)$$

$$\text{Minimizar } f_2(x) = (x - 5)^2 \quad (10)$$

y $-5 \leq x \leq 10$.

La figura 3, muestra el verdadero frente de Pareto graficado contra la solución encontrada por SIAM, NSGA-II y PAES.

	Distribución			DG		
	SIAM	NSGA-II	PAES	SIAM	NSGA-II	PAES
Media	0.905722	0.815194	0.135875	0.036707	0.049669	0.095323
Mejor	0.783875	0.729958	0.048809	0.002740	0.004344	0.002148
Peor	1.670836	1.123444	0.222275	0.160347	0.523622	0.224462
Dsv.Est.	0.237979	0.077707	0.042790	0.043617	0.123888	0.104706

Cuadro 3: Distribución y distancia generacional (DG) para el ejemplo 2

	SIAM	NSGA-II	PAES
Media	0.007431	0.002703	0.005941
Mejor	0.000000	0.000000	0.000000
Peor	0.010000	0.009009	0.009901
Dsv.Est.	0.003.52	0.004236	0.004976

Cuadro 4: Tasa de error para el ejemplo 2

	Distribución			DG		
	SIAM	NSGA-II	PAES	SIAM	NSGA-II	PAES
Media	0.2363	0.14528	0.26849	0.00037	0.00056	0.0026
Mejor	0.2158	0.03940	0.07496	0.00019	0.0004	0.0004
Peor	0.2564	0.21679	1.59285	0.00170	0.00174	0.0349
Dsv.Est	0.0135	0.07938	0.3367	0.0003	0.00033	0.007

Cuadro 5: Distribución y Distancia Generacional (DG) para el ejemplo 3

Los resultados obtenidos por las métricas de distribución y distancia generacional se muestran en la tabla 5 y la tasa de error en la tabla 6. El mejor valor obtenido para la métrica de distribución lo obtiene NSGA-II seguido por SIAM y PAES, con valores muy cercanos. Para el caso de la distancia generacional, la mejor aproximación la tiene SIAM seguida por NSGA-II con valores muy cercanos al ideal. Por otro lado la métrica de tasa de error, muestra que NSGA-II tiene el mejor valor, ocupando el segundo lugar SIAM y en último lugar PAES. En las gráficas podemos ver que SIAM y NSGA-II encuentran ambos segmentos del frente de Pareto, mientras que PAES sólo se aproxima a uno de ellos.

Como puede verse, este algoritmo usa la idea general del principio de selección clonal del sistema inmune, pero además toma “prestados” algunos de los operadores de los Algoritmos Evolutivos. Por ejemplo aplica el operador de cruza de dos puntos a la manera de los Algoritmos Genéticos con la finalidad de alcanzar puntos intermedios entre individuos de la memoria secundaria (paso 10 del algoritmo). Sin embargo, en la población principal el operador de reproducción que se aplica es la clonación (tomado del sistema inmune). Así mismo, como en los algoritmos evolutivos para optimización multiobjetivo de segunda generación, usa un archivo externo como mecanismo elitista.

	SIAM	NSGA-II	PAES
Media		0.4100	0.2108
Mejor	0.3663	0.1782	0.2277
Peor	0.4455	0.2376	1.0000
Dsv.Est.	0.0254	0.0184	0.2732

Cuadro 6: Tasa de error para el ejemplo 3

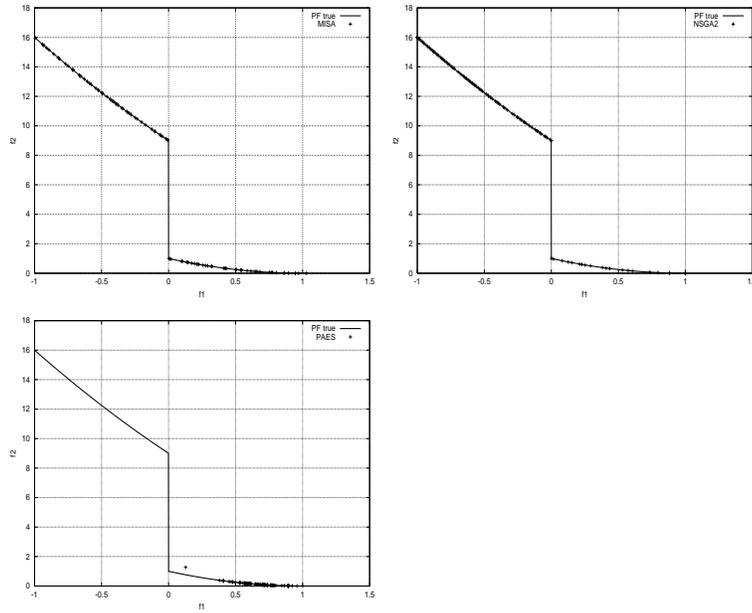


Figura 3: Frentes de Pareto obtenidos por SIAM, NSGA-II y PAES respectivamente, para el ejemplo 3.

De manera que, podríamos considerar esta propuesta como un híbrido que toma ideas del sistema inmune y de la computación evolutiva.

6. Conclusiones y trabajo futuro

Hemos presentado un nuevo algoritmo para optimización multiobjetivo que toma ideas del principio de selección clonal del sistema inmune. Este algoritmo representa el primer intento por solucionar problemas de optimización multiobjetivo (con o sin restricciones) usando el sistema inmune artificial, basado en el concepto de optimalidad de Pareto. Los resultados de las comparaciones muestran que es una opción competitiva, siendo éste un algoritmo relativamente sencillo. Como trabajo futuro, es necesario hacer ajustes al algoritmo con la finalidad de que la distribución de las soluciones a lo largo del frente de Pareto mejore. Además de realizar un estudio estadístico detallado para realizar una caracterización del tipo de problemas a los que puede ser aplicado el algoritmo de manera exitosa.

Agradecimientos

La primera autora agradece a CONACyT por la beca otorgada para realizar estudios de postgrado en la Sección de Computación del Departamento de Ingeniería Eléctrica

del CINVESTAV-IPN. El segundo autor agradece a CONACyT por el apoyo recibido a través del proyecto 34201-A.

Referencias

- [1] Pareto, Vilfredo. *Cours D'Economie Politique*, volume I and II. F. Rouge, Lausanne, 1896.
- [2] Coello Coello, Carlos A. A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques. *Knowledge and Information Systems. An International Journal*, 1(3):269–308, August 1999.
- [3] Deb, Kalyanmoy, Samir Agrawal, Amrit Pratab, and T. Meyarivan. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In Marc Schoenauer, Kalyanmoy Deb, Günter Rudolph, Xin Yao, Evelyne Lutton, Juan Julian Merelo, and Hans-Paul Schwefel, editors, *Proceedings of the Parallel Problem Solving from Nature VI Conference*, pages 849–858, Paris, France, 2000. Springer. Lecture Notes in Computer Science No. 1917.
- [4] Knowles, Joshua D. and David W. Corne. Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy. *Evolutionary Computation*, 8(2):149–172, 2000.
- [5] Schott, Jason R. Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization. Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts, May 1995.
- [6] David A. Van Veldhuizen. *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations*. PhD thesis, Department of Electrical and Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio, May 1999.
- [7] Kita, Hajime, Yasuyuki Yabumoto, Naoki Mori, and Yoshikazu Nishikawa. Multi-Objective Optimization by Means of the Thermodynamical Genetic Algorithm. In Hans-Michael Voigt, Werner Ebeling, Ingo Rechenberg, and Hans-Paul Schwefel, editors, *Parallel Problem Solving from Nature—PPSN IV*, Lecture Notes in Computer Science, pages 504–512, Berlin, Germany, September 1996. Springer-Verlag.
- [8] Leandro Nunes de Castro y Jon Timmis, *An Introduction to Artificial Immune Systems: A New Computational Intelligence Paradigm* Springer-Verlag, 2002
- [9] Schaffer, J. David, *Multiple Objective Optimization with Vector Evaluated Genetic Algorithms*, Vanderbilt University, 1984