

**Materia:**

**DISEÑO ELECTRÓNICO BASADO EN  
SISTEMAS EMBEBIDOS**

**Alumno:**

Posadas Pérez Isaac Sayeg

Paniagua Rico Juan Julian

García Azzúa Jorge Roberto

**Grado y grupo:**

8°G

**Profesor:**

Garcia Ruiz Alejandro Humberto

**Unidad 3 - Tarea 8:**

Función objetivo ponderada

# Función objetivo ponderada en sistemas embebidos

## Introducción

En muchos problemas de optimización en sistemas embebidos es común que existan múltiples objetivos o criterios que deben ser considerados simultáneamente, como minimizar el consumo energético, reducir el tiempo de respuesta y maximizar la confiabilidad. La **función objetivo ponderada** surge como una técnica para abordar este tipo de problemas de manera unificada, combinando varios objetivos en una sola función escalar mediante el uso de **pesos**.

Este enfoque es útil en entornos embebidos donde las decisiones deben optimizarse bajo restricciones de recursos limitados. Permite priorizar ciertos objetivos sobre otros dependiendo de las necesidades del sistema, logrando un equilibrio funcional que sería difícil de obtener con un único objetivo fijo.

## Desarrollo

### 1. ¿Qué es una función objetivo ponderada?

Una función objetivo ponderada es una expresión matemática que combina múltiples funciones objetivo individuales en una sola fórmula, utilizando coeficientes de ponderación para asignarles importancia relativa:

$$F(x) = w_1 \cdot f_1(x) + w_2 \cdot f_2(x) + \dots + w_n \cdot f_n(x)$$

Donde:

- $F(x)$ : función objetivo combinada
- $f_i(x)$ : funciones objetivo individuales
- $w_i$ : pesos asociados a cada función, tal que  $\sum w_i = 1$

Esto convierte un problema multiobjetivo en uno de optimización escalar, que puede resolverse usando metaheurísticas o algoritmos clásicos.

## 2. Justificación de su uso en sistemas embebidos

- **Multicriterio natural del diseño embebido:** Se debe balancear tiempo de ejecución, consumo energético, uso de memoria, precisión, etc.
- **Simplificación del modelo de optimización:** Permite usar algoritmos tradicionales sin necesidad de técnicas especializadas para múltiples objetivos.
- **Adaptabilidad:** Los pesos se pueden ajustar dinámicamente según el contexto del sistema (por ejemplo, priorizar batería en modo ahorro).

## 3. Diseño y normalización de funciones objetivo

### a) Normalización de objetivos

Los objetivos deben ser llevados a una escala común antes de ser combinados, ya que pueden tener unidades o magnitudes diferentes.

Ejemplo de normalización:

$$f_i^{norm}(x) = \frac{f_i(x) - f_i^{min}}{f_i^{max} - f_i^{min}}$$

Esto asegura que ningún objetivo se domine artificialmente debido a su escala.

### b) Asignación de pesos

Los pesos determinan la importancia relativa de cada objetivo. Se pueden definir:

- **Manual o experta:** Basado en experiencia del diseñador del sistema.
- **Automática o adaptativa:** Mediante técnicas de aprendizaje o heurísticas dinámicas.

## 4. Aplicaciones en sistemas embebidos

### a) Gestión de energía y rendimiento

En sistemas alimentados por batería (como sensores IoT), se pondera el rendimiento y la duración energética:

$$F(x) = 0.7 \cdot (\text{consumo normalizado}) + 0.3 \cdot (\text{tiempo de respuesta normalizado})$$

#### b) Optimización de asignación de tareas

En sistemas multiprocesador embebidos, se busca minimizar el tiempo total de ejecución y el uso de recursos.

#### c) Diseño de hardware embebido

La síntesis de arquitecturas puede incluir área ocupada, frecuencia máxima y consumo:

$$F(x) = 0.4 \cdot \text{área} + 0.4 \cdot \text{energía} + 0.2 \cdot \text{retardo}$$

#### d) Controladores adaptativos

El diseño de controladores PID embebidos puede equilibrar estabilidad, precisión y velocidad de respuesta.

### 5. Limitaciones y desafíos

- **Elección de pesos:** Requiere conocimiento profundo del sistema o un proceso de ajuste cuidadoso.
- **Objetivos en conflicto:** A veces no es posible encontrar un balance óptimo; los compromisos deben aceptarse.
- **Cambios dinámicos:** En tiempo de ejecución, las prioridades pueden variar, y la función objetivo debería adaptarse en consecuencia.
- **No captura todo el conjunto de Pareto:** Una sola función combinada pierde visibilidad sobre soluciones alternativas igualmente válidas (óptimos de Pareto).

### Ejemplos

- **Sistema embebido en dron:** Se prioriza autonomía sobre precisión de navegación cuando la batería está baja. Los pesos cambian dinámicamente.
- **Dispositivo médico portátil:** Equilibrio entre precisión de medición, tiempo de operación y vida útil de la batería.
- **Sistema de monitoreo ambiental:** Red de sensores que optimiza intervalo de muestreo y consumo energético con una función objetivo ponderada dependiente de la frecuencia de eventos.

## Conclusión

La función objetivo ponderada es una técnica efectiva para abordar problemas de optimización multicriterio en sistemas embebidos. Al permitir la combinación flexible de distintos objetivos, facilita el diseño de soluciones balanceadas que se ajustan a restricciones reales de hardware y software.

No obstante, su efectividad depende críticamente de una correcta normalización y asignación de pesos. En sistemas embebidos dinámicos o autónomos, esta función puede evolucionar durante el tiempo de ejecución para adaptarse a condiciones cambiantes. En conjunto con heurísticas y metaheurísticas, esta técnica potencia la capacidad de diseñar sistemas más inteligentes y eficientes.

## Bibliografía

- ❖ Marwedel, P. (2011). *Embedded System Design*. Springer.
- ❖ Talbi, E. G. (2009). *Metaheuristics: From Design to Implementation*. Wiley.
- ❖ Deb, K. (2001). *Multi-objective Optimization using Evolutionary Algorithms*. Wiley.
- ❖ Zitzler, E., et al. (2004). *Indicator-based Selection in Multiobjective Search*.
- ❖ Givargis, T., & Vahid, F. (2002). *Embedded System Design*. Wiley.
- ❖ Wolf, W. (2012). *Computers as Components*. Morgan Kaufmann.