

Sintonización de un control difuso aplicado a una celda termoeléctrica: un enfoque basado en algoritmos genéticos con codificación real

Tuning of a fuzzy control applied to a thermoelectric cell:
an approach based on genetic algorithms with real coding

Juan Fernando García-Mejía ^{1*}, Allan Antonio Flores-Fuentes ¹, José Arturo Pérez
Martínez ¹

¹ Centro Universitario UAEMex, Atlacomulco, Universidad Autónoma del Estado de México.
Carretera Toluca-Atlacomulco km 60. CP 50450. Estado de México, México
* Correo-e: fgarciam@uaemex.mx

PALABRAS CLAVE:

celda termoeléctrica, algoritmo
Mamdani, algoritmo genético

RESUMEN

Una celda termoeléctrica es un dispositivo semiconductor de dos capas que basa su funcionamiento en los efectos Thompson y Seebeck; por lo tanto, es factible su uso en la refrigeración móvil y es posible regular su funcionamiento por técnicas de ingeniería de control. En este trabajo se documenta un controlador difuso tipo Mamdani de una entrada (porcentaje de error) y una salida (pendiente de control), cuyos conjuntos difusos se calcularon por medio de un algoritmo genético con codificación real, con una función de costo implementada con base en la integral del error absoluto. Cabe destacar que la presente propuesta se basa en Scilab, un software de licencia GNU.

KEYWORDS:

thermoelectric cell, Mamdani
algorithm, genetic algorithm

ABSTRACT

A thermoelectric cell is a two layer semiconductor device which operation is based on the Thompson and Seebeck effects; hence, its use on mobile refrigeration is feasible and its operation can be regulated by control engineering techniques. In this paper, an one input (error rate) and one output (slope control) Mamdani type fuzzy controller is documented, whose fuzzy sets were calculated by a genetic algorithm with real codification, with a cost function implementation based on the integral of the absolute error. It must be emphasized that the present proposal is based on Scilab, a GNU licensed software.

1 INTRODUCCIÓN

En esta sección se exponen los fundamentos teóricos que sustentan la propuesta de este trabajo. En primer término, se describen aplicaciones y el modelo matemático de una celda termoeléctrica; posteriormente se documentan los principios de las técnicas empleadas en esta propuesta.

1.1 Celda termoeléctrica

Existen dispositivos que se utilizan como generadores del efecto de enfriamiento, denominados *thermoelectrics coolers* (TEC) que funcionan mediante la aplicación de una corriente a una placa que está constituida de un material semiconductor. Las aplicaciones de los TEC se orientan en la microrrefrigeración y refrigeradores móviles, por ejemplo Tarter [1]. Su comportamiento matemático se describe en la ecuación 1 [2, 3], donde $\tilde{T}(s)$ es la temperatura de la celda e $\tilde{I}(s)$ es la corriente de alimentación. Una respuesta típica mostrada en la figura 1 puede manipularse por medio de técnicas de ingeniería de control, como los controladores difusos, basados en la teoría desarrollada por Lofti Zadeh.

$$\frac{\tilde{T}_L(s)}{\tilde{I}(s)} = G_I(s) = -6.4061 \left(\frac{0.064s + 0.00854}{s^2 + 0.5964s + 0.00855} \right) \quad (1)$$

1.2 Lógica difusa

Lofti Zadeh es uno de los pioneros en el uso de las tecnologías de información para el control de procesos, al postular la lógica difusa en un artículo titulado "Fuzzy Sets" en la revista *Information and Control* (1965), donde proponía un algoritmo fundamentado en el razonamiento difuso. Años después, en 1972, Michio Sugeno, del Instituto de Tecnología de Tokio, propuso los conceptos de medición e integral difusa a partir de las ideas de Zadeh. En 1974, Ebrahim Mamdani, de la Universidad de Londres, utilizó las técnicas desarrolladas por Zadeh para controlar una máquina de vapor simple [4, 5].

Un fuzzificador tiene como función transformar la información de entrada de un sistema difuso a variables lingüísticas mediante el uso de una operación denominada fuzzificación, la cual está sustentada en tres valores:

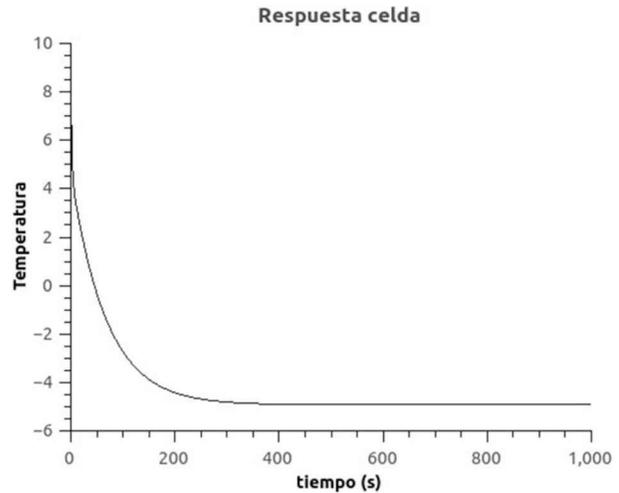


Figura 1. Respuesta de una celda termoeléctrica a lazo abierto

- V es una variable (como área o tamaño) definida sobre un conjunto de referencia X
- X es el universo de discurso (campo de variación de V)
- T_V es el vocabulario para describir en forma simbólica los valores de V (pequeño, grande)

El conjunto $T_V = \{A_1, A_2, \dots\}$, finito o infinito, contiene subconjuntos difusos normalizados de X , es usado para caracterizar V . Cada subconjunto difuso A_i es definido por un grado de membresía $\mu_{A_i}(x)$.

La inferencia difusa consiste en la interpretación de los resultados mediante un conjunto de sentencias lógicas del tipo *if-then* formadas por un secuyente y un consecuente; lo cual determina el tipo de algoritmo empleado, que puede ser del tipo Mamdani, si el consecuente es numérico o Takagi-Sugeno, si está constituido por un vector.

El defuzzificador es el elemento que se encarga de convertir el resultado de la inferencia en una variable de salida acorde al universo de discurso del problema. Es empleado cuando se utiliza el algoritmo de Mamdani, y está basado en los conjuntos de salidas difusos.

1.3 Algoritmo genético

Ahora bien, los algoritmos evolutivos (EA, por sus siglas en inglés) son series de pasos definidos aplicables en la optimización, o de aprendizaje con la capacidad de evolucionar. Tienen tres características principales [6]:

- Basado en la población. Un EA mantiene un conjunto de soluciones, llamado población, para optimizar o aprender el problema en forma paralela. Este es un principio básico del proceso evolutivo.
- Orientada en *fitness*. Toda solución en una población se llama individuo, tiene una representación llamada código y la evaluación de desempeño se denomina aptitud (o *fitness*). Un EA privilegia a los individuos más aptos que permitan la optimización y la convergencia de los algoritmos.
- Variación de motor. Los individuos de una población se someten a una serie de operaciones con la finalidad de introducir variaciones en su código; esto es fundamental para buscar un espacio de soluciones.

Un algoritmo genético (GA, por sus siglas en inglés) simula algunos aspectos propios de la teoría de la evolución de las especies de Darwin y de las leyes de la herencia de Gregory Mendel [7]. El primer algoritmo genético fue desarrollado por John Holland, en la década de 1960, en la Universidad de Michigan; el pseudocódigo de su trabajo se lista a continuación [8, 9]:

1. definir una función de aptitud o función objetivo;
2. generar una serie de posibles soluciones de manera aleatoria (población);
3. codificar la población;
4. evaluar la población, iniciando así la i-esima generación;
5. seleccionar soluciones que se reproducirán;
6. aplicar una operación de cruzamiento;
7. mutar algunos elementos resultantes del cruzamiento;
8. reemplazar elementos de la población de la i-esima generación con los mejores elementos de 6 y 7;
9. detener si se cumple criterio de paro, y en caso contrario, saltar a 4.

La lógica difusa y los algoritmos genéticos son partes de las técnicas que se engloban bajo el concepto de *soft computing*, término acuñado por Lofti Zadeh en 1992 [10]. Cuando existe una sinergia entre las técnicas antes mencionadas se denomina inteligencia computacional [11].

2 METODOLOGÍA

En esta sección se muestran las consideraciones de diseño del controlador difuso utilizado; se detalla el número de variables de entrada y salida, así como los conjuntos difusos de cada una de ellas, el tipo de fuzzificador y el defuzzificador. Asimismo, se desglosa la aplicación del algoritmo genético propuesto, cuya función de costo se basa en la integral del error absoluto.

2.1 Formulación del controlador difuso

Para la regulación de la temperatura de la celda termo-eléctrica se utiliza un controlador difuso Mamdani (MCF, por sus siglas en inglés) del tipo SISO (una entrada una salida), el cual consiste de cuatro etapas: la fuzzificación, la inferencia, la defuzzificación y la agregación. Los conjuntos difusos asociados a la variable de entrada *er*, (cuyo rango es $-20 < er < 110$) se describen en la tabla 1 y los de la pendiente de control *pe* (con rango $-2 < pe < 2$) se enlistan en la tabla 2. Las reglas de inferencia propuestas se muestran en la tabla 3. Cabe destacar que el fuzzificador empleado es del tipo Singleton, los valores e_1, \dots, e_{12} pertenecen al universo de discurso de la variable *er* y los valores c_1, \dots, c_{10} pertenecen al universo de la pendiente de control. En cuanto al defuzzificador, se propone el uso de media de centros.

Tabla 1. Variable de entrada al sistema difuso y sus conjuntos difusos

VARIABLE	CONJUNTOS DIFUSOS	FUNCIÓN DE MEMBRESÍA
er (% de error)	ne error negativo	$\mu_L(er; e_1, e_4)$
	c error cero	$\mu_{triangular}(er; e_3, e_4, e_5)$
	ep error pequeño	$\mu_{triangular}(er; e_6, e_7, e_8)$
	em error medio	$\mu_{triangular}(er; e_9, e_{10}, e_{11})$
	ae error alto	$\mu_{triangular}(er; e_{12}, e_{13})$

La razón para usar un controlador MCF del tipo SISO radica en la simplicidad de expresar conocimiento en la etapa de inferencia; se eligieron las funciones de membresía porque son las más usadas en la literatura especializada.

Tabla 2. Variable de salida del sistema difuso y sus conjuntos difusos

VARIABLE	CONJUNTOS DIFUSOS	FUNCIÓN DE MEMBRESÍA
pe (pendiente de control)	nc corriente negativa	$\mu_L(pe; c_1, c_2)$
	bc baja corriente	$\mu_{triangular}(pe; c_3, c_4, c_5)$
	mc media corriente	$\mu_{triangular}(pe; c_6, c_7, c_8)$
	ac alta corriente	$\mu_{triangular}(pe; c_9, c_{10})$

Tabla 3. Reglas de inferencia

Nº	REGLA
1	Si <i>er</i> es <i>ne</i> entonces <i>pe</i> es <i>nc</i>
2	Si <i>er</i> es <i>c</i> entonces <i>pe</i> es <i>bc</i>
3	Si <i>er</i> es <i>ep</i> entonces <i>pe</i> es <i>bc</i>
4	Si <i>er</i> es <i>em</i> entonces <i>pe</i> es <i>mc</i>
5	Si <i>er</i> es <i>ae</i> entonces <i>pe</i> es <i>ac</i>

2.2 Aplicación del algoritmo genético

Es posible sintonizar un controlador difuso por medio de la variación de los conjuntos difusos de las variables *er* y *pe*. Con esta finalidad, se propone un algoritmo genético con una función objetivo en términos de las señales etiquetadas en el esquema de control propuesto en la figura 2; esto se describe en (2). Cabe destacar que la señal salida de temperatura está determinada por la señal error y los elementos del MCF descritos en 2.1.

$$f_{obj} = \left(\frac{1}{\int \left| \frac{\text{Referencia} - \text{Salida de temperatura}}{\text{Referencia}} \right| * 100 \right)} \quad (2)$$

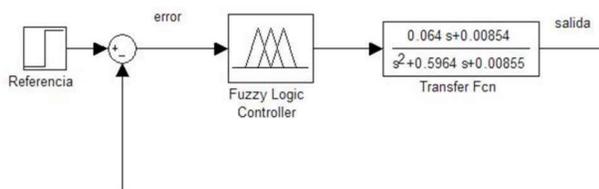


Figura 2. Esquema de control propuesto

Los posibles conjuntos difusos que sintonizan al controlador difuso constituyen una colección de sujetos propuestos como posibles soluciones y son generados de manera aleatoria (100 en total), con una distribución uniforme (mediante las condiciones

mostradas en la tabla 4) codificando los cromosomas con números reales, lo cual fue propuesto por Lucasius [12] como una alternativa a la representación binaria original propuesta por Holland cuando se tienen problemas de optimización numérica en dominio continuo.

Con lo anterior en mente, es posible definir que el formato del cromosoma tiene la siguiente forma:

$$cromosoma = [e_1, e_2, \dots, e_{13}, c_1, c_2, \dots, c_{10}]^T$$

Tabla 4. Condiciones de generación de cromosomas

CROMOSOMAS DE <i>er</i>	CROMOSOMAS DE <i>pe</i>
$e_1 = -(20 + (\text{random}[0,1]) * 2)$	$c_1 = -(0.1 + (\text{random}[0,1]) * 1.9)$
$e_2 = (e_1 + (\text{random}[0,1]) * (220 - e_1))$	$c_2 = (\text{random}[0,1]) * 0.5$
$e_3 = (e_1 + (\text{random}[0,1]) * (e_2 - e_1))$	$c_3 = (\text{random}[0,1]) * 0.5$
$e_4 = (e_2 + (\text{random}[0,1]) * (220 - e_2))$	$c_4 = (\text{random}[0,1]) * 0.5$
$e_5 = (e_4 + (\text{random}[0,1]) * (220 - e_4))$	$c_5 = (c_4 + (\text{random}[0,1]) * (0.5 - c_4))$
$e_6 = (e_5 + (\text{random}[0,1]) * (220 - e_5))$	$c_6 = c_4$
$e_7 = (e_1 + (\text{random}[0,1]) * (e_6 - e_1))$	$c_7 = (c_4 + (\text{random}[0,1]) * (0.5))$
$e_8 = (e_7 + (\text{random}[0,1]) * (220 - e_7))$	$c_8 = (0.5 + (\text{random}[0,1]) * (c_7))$
$e_9 = (e_1 + (\text{random}[0,1]) * (e_8 - e_1))$	$c_9 = (0.3 + (\text{random}[0,1]) * (0.7))$
$e_{10} = (e_8 + (\text{random}[0,1]) * (220 - e_8))$	$c_{10} = c_8$
$e_{11} = (e_{10} + (\text{random}[0,1]) * (220 - e_{10}))$	
$e_{12} = (e_1 + (\text{random}[0,1]) * (e_{11} - e_1))$	
$e_{13} = (e_{11} + (\text{random}[0,1]) * (220 - e_{11}))$	

Los cromosomas que se seleccionaron para el cruzamiento en sucesivas generaciones fueron escogidos mediante un operador de ruleta, dado que favorece y preserva la característica estocástica de los algoritmos genéticos.

Para la codificación usada para representar las variables de entrada y salida del controlador difuso existen diversas opciones, como el cruzamiento *BLX- α* y el cruzamiento aritmético, tanto no uniforme como uniforme. En el caso de este trabajo se utilizó el último.

El operador de cruce aritmético puede definirse de la siguiente forma:

Sean $C_1 = [e_1^1, e_2^1, \dots, e_{13}^1, c_1^1, c_2^1, \dots, c_{10}^1]$ y $C_2 = [e_1^2, e_2^2, \dots, e_{13}^2, c_1^2, c_2^2, \dots, c_{10}^2]$ que fueron seleccionados mediante un procedimiento de ruleta. Los descendientes $H_k = [e_1^k, e_2^k, \dots, e_{13}^k, c_1^k, c_2^k, \dots, c_{10}^k]$, fueron generados a través de las ecuaciones 3 y 4 con un $\alpha = 0.5$

$$H_1 = \alpha (C_1 + ((1 - \alpha) * C_2)) \quad (3)$$

$$H_2 = \alpha (C_2 + ((1 - \alpha) * C_1)) \quad (4)$$

En la mutación se alternan de manera aleatoria dos individuos por cada generación. Para esto se utiliza el operador genético de mutación con distribución normal, donde a partir de un cromosoma C_i' se puede obtener un cromosoma C_i'' transformado o mutado desde la siguiente expresión; donde el tamaño de paso de la mutación es $\beta=[0,1]$ y la dirección $d=[0,1]$. Esto se representa por (5):

$$C_i'' = C_i' + \beta * d \quad (5)$$

3 RESULTADOS

La aplicación de la metodología propuesta en este documento permite observar una respuesta de la celda termoeléctrica controlada por el MCF a una referencia de -5°C , que se grafica en la figura 3(a) y se sintetiza en la tabla 5. Por otra parte, la figura 3(b) muestra la respuesta del algoritmo genético, donde en el eje x se muestra el número de generaciones usadas, y el eje y se constituye con los valores de la integral de error absoluto, cuya convergencia se observa alrededor de la generación 110.

La figura 3(a) muestra la referencia o *set point*, así como la salida de temperatura regulada en la celda termoeléctrica. Se puede observar que la temperatura ambiente que se propone es de 10°C . La figura 3(b) muestra la convergencia del algoritmo, donde se observan algunas variaciones, las cuales tienden a desaparecer por la mutación.

La figura 4 muestra los conjuntos difusos de la variable pe , y la figura 5, los conjuntos difusos de la variable er que se obtuvieron de la aplicación del algoritmo genético. Estos conjuntos permitieron obtener la respuesta que se muestra en la figura 3.

4 CONCLUSIONES

Como muestran los resultados de la tabla 5 y la figura 2(a), es posible regular el comportamiento de una celda termoeléctrica mediante un MCF, cuyos conjuntos difusos fueron construidos por medio de un algoritmo genético. Esto permite mejorar las actividades de

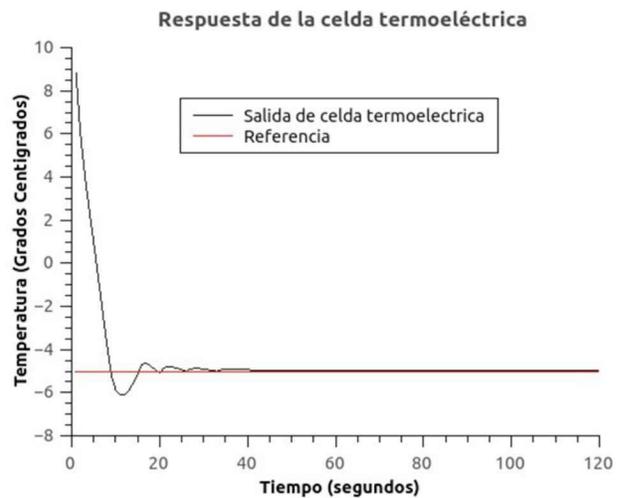


Figura 3(a). Respuesta de la celda y del algoritmo genético

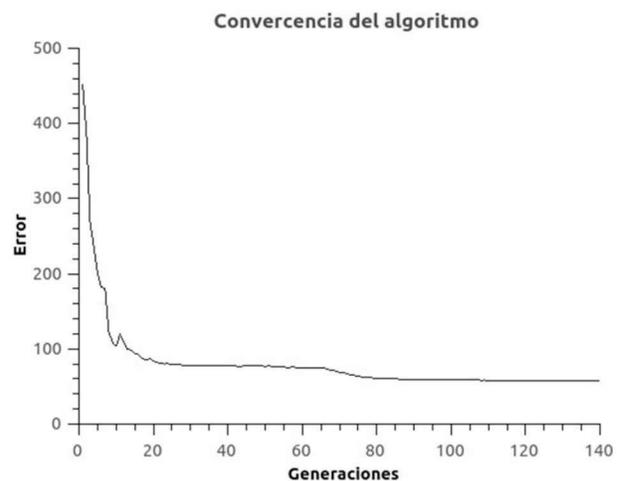


Figura 3(b). Respuesta de la celda y del algoritmo genético

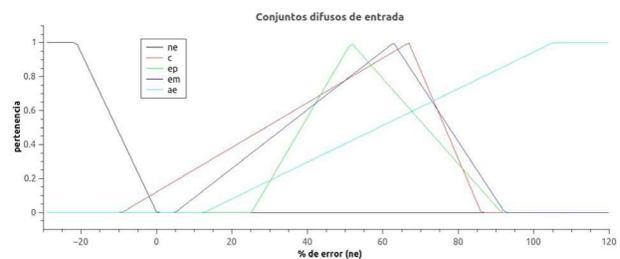


Figura 4. Conjunto difuso de salida

diseño de este tipo de controlador; cabe destacar que actualmente los autores trabajan sobre un estudio de repetitividad que permita constituirse como comparativa del presente trabajo. Por otra parte, el desempeño de la celda debe de contrastarse con otro controlador como Takagi-Sugeno, así como implementar un MCF de tipo MISO (múltiples entradas, una salida), ambas estrategias de control con conjuntos difusos diseñados mediante algoritmos genéticos, ya sean simples o con restricciones.

Tabla 5. Parámetros de la salida regulada

PARÁMETRO	VALOR
Sobre impulso	-6.1
Valor final	-4.995°C
Tiempo de establecimiento	18 seg.
% de error en estado estacionario	0.1%

REFERENCIAS

1. Tarter, R. *Solid-state power conversion handbook*. Nueva York: John Wiley and Sons, 1993.
2. Song, S. Temperature Control of Thermoelectric Cooler Based on Adaptive NN-PID, *Electrical and Control Engineering (ICECE)*, 2010 *International Conference on*, junio de 2010, 2245-2248.
3. García, F. Diseño de controlador proporcional-integral-derivativo de celda termoeléctrica mediante algoritmo genético con codificación real. *Prognat*, 2014, 6(1),55-60.
4. [Pickover, C. A. *The Math Book: From Pythagoras to the 57th Dimension, 250 Milestones in the History of Mathematics*. Nueva York: Sterling Publishing Company, 2009.
5. Tanaka, K. *An introduction to fuzzy logic for practical applications*. Nueva York: Springer-Verlag, 1997.
6. Burger, C. *Propeller performance analysis and multidisciplinary optimization using a genetic algorithm* ProQuest, 2007.
7. Yang, X. S. *Nature-Inspired Metaheuristic Algorithms*. Cambridge: Luniver Press, 2010.
8. Mitchell, M. *An introduction to genetic algorithms*. Cambridge, Massachusetts: MIT Press, 1998.
9. Gen, M. y Cheng, R. *Genetic algorithms and engineering optimization*. Nueva York: John Wiley and Sons, 2000.
10. Konar, A. *Computational intelligence principles, techniques, and applications*. Berlín: Springer-Verlag, 2005.
11. Yu, X. y Gen, M. *Introduction to Evolutionary Algorithms*. Londres: Springer-Verlag, 2010
12. Lucasius C. B. y Kateman, G. Applications of genetic algorithms in chemometrics. J. David Schaffer (ed.), *Proceedings of the Third International Conference on Genetic Algorithms*. 1989, 170–176.

Acerca de los autores



Juan Fernando García Mejía es Ingeniero en Electrónica por parte del Instituto Tecnológico de Toluca desde 2002; por la misma institución, obtuvo el grado de Maestro en Ciencias en Electrónica en 2004. Desde 2004 labora en el Centro Universitario UAEM Atlacomulco, donde es profesor de tiempo completo con perfil

PRODEP y líder del cuerpo académico Desarrollo de Software, Dispositivos y Sistemas Aplicados a la Innovación Tecnológica. Su línea de investigación está centrada en el desarrollo de instrumentos virtuales y el uso de técnicas de *soft-computing* aplicadas a la ingeniería de control.



Allan Antonio Flores-Fuentes es Ingeniero en Electrónica por parte del Instituto Tecnológico de Toluca, Metepec, desde 2004. Obtuvo el grado de Doctor en Ciencias en Ingeniería Electrónica por el Instituto Tecnológico de Toluca en 2009. Actualmente es miembro SNI, nivel

candidato, por parte del CONACyT de 2012-2014. Desde 2011 se desempeña en la Universidad Autónoma del Estado de México como Profesor-Investigador en el área de tópicos selectos de electrónica de potencia aplicada para el desarrollo de conversión de energía, implementando técnicas de control mediante uso de *soft-computing*.



José Arturo Pérez Martínez es Ingeniero en Electrónica desde 2004 y en 2010 recibió el grado de Doctor en Ciencias en Ingeniería Electrónica, ambos por parte del Instituto Tecnológico de Toluca, México. Actualmente su investigación se enfoca en el diseño de convertidores estáticos de

radio frecuencia aplicados en la generación de plasmas a presión atmosférica en reactores con diferentes configuraciones; labora en la Universidad Autónoma del Estado de México, donde es miembro del cuerpo académico Desarrollo de Software, Dispositivos y Sistemas Aplicados a la Innovación Tecnológica, en el Centro Universitario UAEM Atlacomulco.