

Capítulo 6

Conclusiones del estudio

6.1 Conclusiones y aportaciones principales

Tras centrar, en la introducción, la familia de sistemas que se desea modelar y el marco de modelado, en el segundo capítulo se ha justificado el uso de las redes neuronales en el ámbito del control de sistemas dinámicos y, en concreto, en el modelado e identificación de sistemas no lineales. Se ha visto que tienen ventajas sobre otros modelos e inconvenientes como la imposibilidad de interpretar físicamente sus resultados o el de añadir conocimiento previo para acelerar el proceso de modelado.

Se ha propuesto un modelo que aprovecha las ventajas de las redes neuronales y minimiza sus inconvenientes. Se trata de las redes modulares, que están compuestas por módulos neuronales.

Un módulo neuronal es una red neuronal que aprovecha el uso de restricciones estructurales para forzar un tipo de comportamiento en el modelo. Este concepto se ha creado a propósito en este estudio, apoyado por el argumento de que las restricciones topológicas constituyen un método más versátil y efectivo que el propio mecanismo de aprendizaje para facilitar comportamientos deseados en una ANN.

De esta forma, una vez aplicado el proceso de identificación, el modelo resultante es una red neuronal compuesta por módulos, cada uno de los cuales representa un bloque funcional del sistema con un significado fácilmente interpretable.

Puesto que los módulos neuronales son paradigmas nuevos en el ámbito de las redes neuronales, se ha propuesto una serie de pautas para el diseño de NMs que aproximen una determinada

familia de funciones. Se ha descrito con detalle el diseño de módulos que permiten discriminar la pertenencia del valor de entrada a un rango y el de módulos que aproximan funciones lineales a tramos.

También se describe un conjunto de módulos neuronales que resulta útil en el modelado de sistemas dinámicos no lineales. Este conjunto está formado por nueve no-linealidades duras y los sistemas lineales sin restricción de orden. Las primeras corresponden a los comportamientos más comunes en los modelos de sistemas físicos, como la saturación, la zona muerta, la histéresis o la fricción. Los sistemas lineales se han modelado siguiendo la representación de entrada y salida.

En el tercer capítulo se ha realizado un estudio formal en el que se han caracterizado los sistemas que se pueden aproximar mediante redes de módulos neuronales, el conjunto Σ_{NM} , y se ha establecido una cota del error de esta aproximación.

Σ_{NM} está formado por la unión infinita de conjuntos NLN^i que, a su vez, están ordenados jerárquicamente según $NLN \subset NLN^2 \subset \dots \subset NLN^i$. El conjunto NLN contiene sistemas lineales, sin restricción de orden, modificados en sus entradas y salida por características no lineales estáticas, continuas y derivables.

Σ_{NM} es un conjunto interesante porque cuando se modela un proceso físico es habitual dividirlo en subsistemas más sencillos, de manera que el producto de uno de ellos, su salida, es la entrada de otro. Estos subsistemas suelen considerarse lineales, aunque si contienen características no lineales es habitual que se trate no-linealidades duras, como las tratadas en este estudio. De esta forma, las partes del proceso corresponden a elementos de NLN y su composición corresponde a un elemento de NLN^i , que está en Σ_{NM} .

Se ha demostrado, en la sección 3.3, que si un sistema no lineal se puede representar mediante sumas y composiciones de sistemas más sencillos, elementos de NLN , entonces existe una red de módulos neuronales que lo aproxima con un grado arbitrario de precisión.

Este resultado es fundamental porque sienta una base formal para plantear el modelado de sistemas no lineales con redes modulares de módulos neuronales. Hay que tener presente que el comportamiento de los sistemas no lineales es mucho más imprevisible que el de los lineales y que dos sistemas con idéntica estructura y parámetros ligeramente diferentes pueden tener comportamientos radicalmente distintos. Este resultado nos asegura que, cuanto más precisa sea la aproximación de las diferentes partes del sistema, más precisa será la aproximación del sistema global.

El conjunto Σ_{NM} está construido solamente con dos módulos neuronales. Por ello, en la última sección de este capítulo se discute la utilidad de los restantes módulos. Se concluye, primeramente, que los módulos estáticos (saturación, zona muerta, valor absoluto y fricción) cumplen las condiciones necesarias para no alterar los resultados anteriores. En segundo lugar se presentan resultados experimentales que permiten suponer que los módulos dinámicos (juego, tope, limitador de velocidad e histéresis) pueden resultar muy útiles en el modelado de ciertos sistemas no lineales.

En el cuarto capítulo se consideran los aspectos de creación y optimización del modelo propuesto en los capítulos anteriores.

Primeramente, y puesto que se trata de una red neuronal, se repasan los mecanismos existentes en la literatura para adaptar los parámetros del modelo al problema. En este sentido, se ha diseñado un algoritmo de aprendizaje específico para redes neuronales modulares, el *modular backpropagation*.

MBP es un algoritmo de gestión del aprendizaje en redes modulares. Su característica principal es que permite la utilización de otros algoritmos, como *backpropagation* o *backpropagation through time*, de forma local en los módulos de la red. Además, se ha comparado su coste computacional frente al de otros algoritmos clásicos y es más eficiente en estructuras modulares.

El problema de determinar la estructura del modelo, es decir, los módulos que forman parte de él y su patrón de interconexión, es extremadamente complejo. Se han sugerido, para resolverlo, las técnicas de optimización evolutiva y, en concreto, la programación evolutiva.

Finalmente, se describe una herramienta de modelado, diseñada a propósito, como método para crear y optimizar, de forma automática, redes de módulos neuronales. Esta herramienta combina la programación evolutiva, algoritmos clásicos de aprendizaje neuronal y el gestor de aprendizaje, MBP, con la finalidad de resolver problemas de modelado de sistemas no lineales mediante redes de módulos neuronales.

En el capítulo cinco se describe el proceso de modelado de sistemas no lineales utilizando las herramientas desarrolladas en este estudio. Se ha creado una aplicación (*miga*) que permite sistematizar este proceso y se ha realizado el modelado de tres sistemas no lineales para comprobar su utilidad.

El proceso de modelado consta de cuatro pasos:

- Obtención y análisis de datos. Se recomienda obtener secuencias de datos significativas respecto a los regímenes de operación del sistema y suficientemente largas, para poder crear los conjuntos de aprendizaje, test y validación.

En este punto es destacable que este método permite incorporar información previa para sesgar y acelerar la búsqueda del modelo.

- Diseño y ejecución de las pruebas. Se sugiere organizarlo jerárquicamente en pruebas, experimentos y ejecuciones.

La aplicación *miga* realiza todas las pruebas programadas de forma automática en un entorno distribuido.

- Selección de resultados. Cada prueba desprende un gran número de resultados. Estos se ordenan para realizar una selección según la complejidad del modelo y los errores de test.
- Validación, análisis y refinado de resultados. Los modelos resultantes se someten a un proceso de validación convencional para seleccionar el mejor.

Uno de estos modelos es una estructura manipulable, al mismo nivel, que un diagrama de bloques convencional, con bloques no lineales.

Finalmente, los problemas que se han sometido al proceso de modelado con redes neuronales modulares son: motor de corriente continua, sistema no lineal con histéresis y elemento piezoeléctrico. En los tres casos se ha llegado a una solución satisfactoria que permite confirmar la utilidad de las herramientas desarrolladas en este estudio.

6.2 Vías futuras de trabajo

Este estudio representa un punto de partida que ha abierto direcciones de investigación prometedoras. Seguidamente se apuntan algunas de las más interesantes.

En el ámbito de las redes neuronales

- Completar el esquema de modelado modular para abarcar sistemas más complejos: sistemas con no-linealidades de más de una entrada (como el producto de señales), sistemas con características no lineales analíticas (como las trigonométricas, por ejemplo), estudiar la viabilidad del modelado de sistemas realimentados, etc.
- Plantear estructuras de modelado más ricas, como las que modelan el ruido.

- Extender MPB a redes modulares de propagación recurrente entre módulos. Esta extensión es perfectamente viable, pero no se ha realizado por no ser necesaria para el tipo de redes que se tratan en este estudio.

En el ámbito de los algoritmos de optimización evolutiva

- El estudio de criterios más potentes de optimización de modelos. En este trabajo se han utilizado los criterios clásicos de la teoría de identificación de sistemas (AIC, FPE, ISE, MDL, etc.) pero la naturaleza de los algoritmos evolutivos permite el uso de criterios arbitrarios. En este sentido parece interesante profundizar en criterios que ponderen la ‘interpretabilidad’ del modelo, además de su precisión.
- Introducción de reglas para simplificar los modelos en el proceso de búsqueda. Se pueden incluir reglas invariantes respecto al comportamiento del modelo o reglas de ‘podado’, como algunos algoritmos de aprendizaje neuronal, que simplifican el modelo a costa de modificar ligeramente su comportamiento.

En la automatización del proceso de modelado existen múltiples vías de continuidad

- Creación de herramientas más potentes de análisis de los modelos resultantes.
Por un lado, es importante eliminar de forma automática los modelos poco interesantes mediante técnicas de validación más abiertas y, por otro, conviene mejorar y explotar al máximo la característica más propia del modelado con módulos neuronales: la extracción de información sobre el sistema del modelo resultante.
- La ejecución de programas de cálculo intensivo en un entorno distribuido es un campo de investigación muy activo y la herramienta de modelado es un programa claramente paralelizable. Por ello, el abanico de posibilidades en este sentido es muy amplio. Una de las más efectivas podría ser la de distribuir una ejecución a través de Internet, siguiendo un protocolo cliente-servidor, en que los últimos provean servicios de aprendizaje neuronal.