

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 665 072**

51 Int. Cl.:

G05B 13/02 (2006.01)

G06N 3/08 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **16.04.2015 PCT/EP2015/058239**

87 Fecha y número de publicación internacional: **29.10.2015 WO15162050**

96 Fecha de presentación y número de la solicitud europea: **16.04.2015 E 15716062 (3)**

97 Fecha y número de publicación de la concesión europea: **31.01.2018 EP 3117274**

54 Título: **Método, controlador y producto de programa informático para controlar un sistema objetivo mediante el entrenamiento por separado de unos modelos de red neuronal recurrente primero y segundo, que se entrenan inicialmente usando datos operativos de sistemas fuente**

30 Prioridad:
22.04.2014 US 201414258740

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
24.04.2018

73 Titular/es:
**SIEMENS AKTIENGESELLSCHAFT (100.0%)
Werner-von-Siemens-Straße 1
80333 München, DE**

72 Inventor/es:
**DÜLL, SIEGMUND;
SPIECKERMANN, SIGURD;
UDLUFT, STEFFEN y
MUNSHI, MRINAL**

74 Agente/Representante:
LOZANO GANDIA, José

ES 2 665 072 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

MÉTODO, CONTROLADOR Y PRODUCTO DE PROGRAMA INFORMÁTICO PARA CONTROLAR UN SISTEMA OBJETIVO MEDIANTE EL ENTRENAMIENTO POR SEPARADO DE UNOS MODELOS DE RED NEURONAL RECURRENTE PRIMERO Y SEGUNDO, QUE SE ENTRENAN INICIALMENTE USANDO DATOS OPERATIVOS DE SISTEMAS FUENTE

5	DESCRIPCIÓN
	Antecedentes de la invención
10	El control de sistemas técnicos dinámicos complejos, por ejemplo turbinas de gas, turbinas eólicas u otras plantas, puede optimizarse por medio de los denominados enfoques basados en datos. Con ello, pueden mejorarse diversos aspectos de tales sistemas dinámicos, por ejemplo para turbinas de gas, su eficiencia, dinámica de combustión o emisiones, y por ejemplo para turbinas eólicas, su consumo durante la vida útil, eficiencia u orientación.
15	La optimización basada en datos moderna utiliza métodos de aprendizaje de máquina para mejorar estrategias o políticas de control de sistemas dinámicos con respecto a objetivos de optimización generales o específicos. Tales métodos de aprendizaje de máquina permiten a menudo superar las estrategias de control convencionales. En particular, si el sistema controlado está cambiando, un enfoque de control adaptativo que puede producir aprendizaje y ajuste de una estrategia de control según la nueva situación y las nuevas propiedades del sistema dinámico es ventajoso a menudo con respecto a estrategias de control sin aprendizaje convencionales.
20	
25	Sin embargo, para optimizar sistemas dinámicos complejos, por ejemplo turbinas de gas u otras plantas, ha de recopilarse una cantidad suficiente de datos operativos para hallar o aprender una buena estrategia de control. Por tanto, en caso de poner en servicio una nueva planta, de mejorarla o modificarla, puede llevar cierto tiempo recopilar suficientes datos operativos del sistema nuevo o que ha cambiado antes de que se disponga de una buena estrategia de control. Los motivos para tales cambios podrían ser desgaste, piezas cambiadas después de una reparación, o diferentes condiciones ambientales.
30	Los métodos conocidos para el aprendizaje de máquina comprenden métodos de aprendizaje por refuerzo que se centran en el aprendizaje eficiente de datos para un sistema dinámico especificado. Sin embargo, incluso cuando se usan estos métodos, puede llevar cierto tiempo hasta que se dispone de una buena estrategia de control basada en datos después de un cambio del sistema dinámico. Hasta entonces, el sistema dinámico que ha cambiado funciona fuera de una envolvente posiblemente optimizada. Si la tasa de cambio del sistema dinámico es muy alta, sólo pueden lograrse resultados subóptimos para una optimización basada en datos puesto que nunca puede disponerse de una cantidad suficiente de datos operativos.
35	
40	El documento US 7 330 804 B2 da a conocer un aproximador no lineal para el control de procedimientos empíricos. El aproximador se desarrolla basándose en tres etapas que incluyen el aprendizaje de una red neuronal basándose en datos empíricos.
45	El documento de Chan <i>et al.</i> : "Training recurrent network with block-diagonal approximated Levenberg-Marquardt algorithm", INTERNATIONAL JOINT CONFERENCE ON NEURONAL NETWORKS IJCNN '99, vol. 3, 10 de julio de 1999, páginas 1521 a 1526, da a conocer el aprendizaje de una red neuronal recurrente usando una aproximación basándose en una matriz diagonal por bloques en el método de Levenberg-Marquardt usado para el entrenamiento de la red neuronal.
50	El documento de Puskorius <i>et al.</i> : "Neurocontrol of Nonlinear Dynamical Systems with Kalman Filter Trained Recurrent Networks", IEEE TRANSACTIONS ON NEURONAL NETWORKS, vol. 5, n.º 2, 1 de marzo de 1994, páginas 279 a 297, y el documento de Feldkamp <i>et al.</i> : "Enhanced Multi-Stream Kalman Filter Training for Recurrent Networks" en: "Nonlinear Modeling", 31 de diciembre de 1998, páginas 29 a 53, Springer US, dan a conocer el entrenamiento de redes neuronales recurrentes usando un filtro de filtro de Kalman extendido.
	Sumario de la invención
55	En vista de lo anterior, un objeto de la presente invención es crear un método, un controlador y un producto de programa informático para controlar un sistema objetivo que permitan un aprendizaje más rápido de estrategias de control en particular para un sistema objetivo cambiante.
60	Según la presente invención, un método, un controlador o un producto de programa informático para controlar un sistema objetivo, por ejemplo una turbina de gas o eólica u otro sistema técnico, se basa en datos operativos de una pluralidad de sistemas fuente. El método, controlador o producto de programa informático está configurado para recibir los datos operativos de los sistemas fuente, distinguiéndose los datos operativos mediante identificadores específicos de sistema fuente. Por medio de una red neuronal, se somete a entrenamiento un modelo neuronal basado en los datos operativos recibidos de los sistemas fuente teniendo en cuenta los identificadores específicos de sistema fuente, en la que se somete a entrenamiento una primera componente de modelo neuronal sobre las propiedades compartidas por los sistemas fuente y se somete a entrenamiento una segunda componente de modelo
65	

neuronal sobre las propiedades que varían entre los sistemas fuente. Después de recibir datos operativos del sistema objetivo, el modelo neuronal sometido a entrenamiento se somete a entrenamiento adicional basándose en los datos operativos del sistema objetivo, en el que se le da preferencia a un entrenamiento adicional de la segunda componente de modelo neuronal es sobre un entrenamiento adicional de la primera componente de modelo neuronal. El sistema objetivo se controla por medio de la red neuronal sometida a entrenamiento adicional.

Dado que la invención usa datos operativos de una pluralidad de sistemas fuente y usa modelos neuronales aprendidos por medio de estos datos operativos, se dispone de un buen punto de partida para un modelo neuronal del sistema objetivo. En realidad, son necesarios muchos menos datos operativos del sistema objetivo para obtener un modelo neuronal exacto para el sistema objetivo que en el caso de aprendizaje de un modelo neuronal para el sistema objetivo desde cero. Así, pueden aprenderse estrategias o políticas de control eficaces en poco tiempo incluso para sistemas objetivos con datos escasos.

En una realización preferida de la invención, la primera componente de modelo neuronal puede estar representada por primeros pesos adaptativos, y la segunda componente de modelo neuronal puede estar representada por segundos pesos adaptativos. Tales pesos adaptativos también pueden indicarse como parámetros de la componente de modelo neuronal respectiva.

Preferiblemente, el número de los segundos pesos adaptativos puede ser varias veces menor que el número de los primeros pesos adaptativos. Dado que se le da preferencia al entrenamiento de la segunda componente de modelo neuronal representada por los segundos pesos adaptativos sobre el entrenamiento de la primera componente de modelo neuronal representada por los primeros pesos adaptativos, el número de pesos que ha de adaptarse durante el entrenamiento adicional con el sistema objetivo puede reducirse significativamente. Esto permite un aprendizaje más rápido para el sistema objetivo.

Además, los primeros pesos adaptativos pueden comprender una primera matriz de pesos y los segundos pesos adaptativos pueden comprender una segunda matriz de pesos. La segunda matriz de pesos puede ser una matriz diagonal. Para determinar pesos adaptativos del modelo neuronal, la primera matriz de pesos puede multiplicarse por la segunda matriz de pesos.

Según una realización preferida, la primera componente de modelo neuronal puede no someterse a entrenamiento adicional. Esto permite centrarse en el entrenamiento de la segunda componente de modelo neuronal que refleja las propiedades que varían entre los sistemas fuente.

Alternativamente, cuando se somete a entrenamiento adicional el modelo neuronal sometido a entrenamiento, un primer subconjunto de los primeros pesos adaptativos puede mantenerse sustancialmente constante mientras que un segundo subconjunto de los primeros pesos adaptativos puede someterse a entrenamiento adicional. Esto permite un ajuste preciso de la primera componente de red neuronal que refleja las propiedades compartidas por los sistemas incluso durante la fase de entrenamiento adicional.

Según una realización preferida de la invención, el modelo neuronal puede ser un modelo de aprendizaje por refuerzo, que permite un aprendizaje eficiente de estrategias de control para sistemas dinámicos.

Ventajosamente, la red neuronal puede funcionar como una red neuronal recurrente. Esto permite mantener un estado interno que facilita una detección eficiente de patrones dependientes del tiempo cuando se controla un sistema dinámico. Además, pueden manipularse muchos de los denominados procesos de decisión de Markov parcialmente observables como los denominados procesos de decisión de Markov por medio de una red neuronal recurrente.

Según una realización preferida de la invención, durante el entrenamiento del modelo neuronal, puede determinarse si el modelo neuronal refleja una distinción entre las propiedades compartidas por los sistemas fuente y las propiedades que varían entre los sistemas fuente. Dependiendo de esa determinación, puede verse afectado el entrenamiento del modelo neuronal. En particular, el entrenamiento del modelo neuronal basado en los datos operativos de los sistemas fuente puede terminarse si se detecta tal distinción con una fiabilidad predeterminada.

Además, pueden ejecutarse políticas o estrategias de control que resultan del modelo neuronal sometido a entrenamiento en un bucle cerrado de aprendizaje con el sistema objetivo técnico.

Se describen características y ventajas adicionales de la presente invención en, y resultarán evidentes a partir de, la siguiente descripción detallada de la invención y las figuras.

Breve descripción de los dibujos

La figura 1 muestra una ilustración gráfica de una arquitectura de una red neuronal recurrente según una realización a modo de ejemplo de la presente invención.

La figura 2 muestra un esquema de una realización a modo de ejemplo de la invención que comprende un sistema objetivo, una pluralidad de sistemas fuente y un controlador.

Descripción detallada de la realización preferida

5 Según la presente invención, un sistema objetivo se controla no sólo por medio de datos operativos de ese sistema objetivo sino también por medio de datos operativos de una pluralidad de sistemas fuente. El sistema objetivo y los sistemas fuente pueden ser turbinas de gas o eólicas u otros sistemas dinámicos que incluyen herramientas de simulación para simular un sistema dinámico.

10 Preferiblemente, los sistemas fuente se eligen para que sean similares al sistema objetivo. En ese caso, los datos operativos de los sistemas fuente y un modelo neuronal sometido a entrenamiento por medio de los mismos son un buen punto de partida para un modelo neuronal del sistema objetivo. Con el uso de datos operativos u otra información de otros sistemas técnicos similares, puede reducirse considerablemente la cantidad de datos operativos requeridos para el aprendizaje de una estrategia o política de control eficiente para el sistema objetivo. El enfoque de la invención aumenta la eficiencia de datos global del sistema de aprendizaje y reduce significativamente la cantidad de datos requeridos antes de que pueda derivarse una nueva estrategia de control basada en datos para un sistema objetivo recién puesto en servicio.

20 Según una realización preferida de la invención, una turbina de gas debe controlarse como sistema objetivo por medio de una red neuronal sometida previamente a entrenamiento con datos operativos de una pluralidad de turbinas de gas similares como sistemas fuente. Los sistemas fuente pueden comprender el sistema objetivo en un momento diferente, por ejemplo antes del mantenimiento del sistema objetivo o antes del recambio de un componente del sistema, etc. A la inversa, el sistema objetivo puede ser uno de los sistemas fuente en un momento posterior. La red neuronal se implementa preferiblemente como una red neuronal recurrente.

En vez de someter a entrenamiento un modelo neuronal distinto para cada uno de los sistemas fuente por separado, se somete a entrenamiento un modelo neuronal conjunto para la familia de sistemas fuente similares basándose en los datos operativos de todos los sistemas. Ese modelo neuronal comprende como primera componente de modelo neuronal un módulo global que permite que se comparta el conocimiento operativo a través de todos los sistemas fuente. Además, el modelo neuronal comprende como segunda componente de modelo neuronal módulos específicos de sistema fuente que facilitan que se ajuste con precisión el modelo neuronal para cada sistema fuente individualmente. De este modo, es posible un mejor aprendizaje de modelos neuronales, y por tanto, estrategias o políticas de control incluso para sistemas con datos escasos, en particular para un sistema objetivo similar a los sistemas fuente.

35 Supongamos que I_{fuente} e $I_{objetivo}$ indican dos conjuntos de identificadores específicos de sistema de sistemas dinámicos similares. Los identificadores del conjunto I_{fuente} identifican, cada uno, uno de los sistemas fuente mientras que los identificadores del conjunto $I_{objetivo}$ identifican el sistema objetivo. Se supone que los sistemas fuente se han observado durante un tiempo suficientemente largo de tal manera que hay suficientes datos operativos disponibles para el aprendizaje de un modelo neuronal exacto de los sistemas fuente mientras que, en cambio, sólo hay una pequeña cantidad de datos operativos del sistema objetivo disponibles. Puesto que los sistemas tienen propiedades dinámicas similares, la transmisión de conocimientos desde los sistemas fuente bien observados al sistema objetivo observado de manera escasa es un enfoque ventajoso para mejorar la calidad del modelo de estos últimos.

45 Supongamos que $s_1 \in S$ indica un estado inicial de los sistemas dinámicos considerados, donde S indica un espacio de estados de los sistemas dinámicos, y supongamos que a_1, \dots, a_T indican una secuencia de acciones de T escalones, siendo $a_t \in A$ una acción en un espacio de acciones A de los sistemas dinámicos en un escalón de tiempo t . Además, supongamos que h_1, \dots, h_{T+1} indican una secuencia de estados ocultos de la red neuronal recurrente. Entonces, puede definirse un modelo de red neuronal recurrente de un único sistema dinámico, que produce una secuencia de estados sucesores $\hat{s}_2, \dots, \hat{s}_{T+1}$, mediante las siguientes ecuaciones

$$h_1 = \sigma_n(W_{hs}s_1 + b_1)$$

$$55 \quad h_{t+1} = \sigma_n(W_{ha}a_t + W_{hh}h_t + b_h)$$

$$\hat{s}_{t+1} = W_{sh}h_{t+1} + b_s$$

60 donde $W_{vu} \in \mathbb{R}^{n_v \times n_u}$ es una matriz de pesos desde la capa u hasta la capa v , siendo estas últimas capas de la red neuronal recurrente. $b_v \in \mathbb{R}^{n_v}$ es un vector de sesgo de la capa v , n_v es el tamaño de la capa v y $\sigma(\cdot)$ es una función no lineal por elementos, por ejemplo $\tanh(\cdot)$. W_{uv} y b_v pueden considerarse pesos adaptativos que se adaptan durante el procedimiento de aprendizaje de la red neuronal recurrente.

Para permitir la transmisión de conocimientos desde los sistemas fuente al sistema objetivo, pueden modificarse la transición de estados $W_{hh}h_t$, que describe la evolución temporal de los estados ignorando las fuerzas externas, y el efecto de una fuerza externa $W_{ha}a_t$ para compartir conocimientos comunes de todos los sistemas fuente mientras que aún puede distinguirse entre las peculiaridades de cada sistema fuente. Por tanto, la matriz de pesos W_{hh} se factoriza produciendo

$$W_{hh} \approx W_{hf_h} \text{diag}(W_{f_h z} z) W_{f_h h}$$

donde $z \in \{e_1, \dots, e_{|I_{fuente} \cup I_{objetivo}|}\}$ es un vector de base euclidiana que tiene un "1" en la posición $i \in I_{fuente} \cup I_{objetivo}$ y "0" en otros lugares. Es decir, el vector z porta la información por medio de la que la red neuronal recurrente puede distinguir los sistemas fuente específicos. En consecuencia, z actúa como selector de columna de $W_{f_h z}$ de tal manera que hay un conjunto distinto de parámetros $W_{f_h z}$ asignados para cada sistema fuente. Por tanto, la transformación es una composición de los pesos adaptativos W_{hf_h} y $W_{f_h h}$, que se comparten entre todos los sistemas fuente, y los pesos adaptativos $W_{f_h z}$ específicos para cada sistema fuente.

Se aplica la misma técnica de factorización a W_{ha} produciendo

$$W_{ha} \approx W_{hf_a} \text{diag}(W_{f_a z} z) W_{f_a a}$$

La red neuronal recurrente de tensores factorizados resultante se describe entonces mediante las siguientes ecuaciones:

$$h_1 = \sigma_h(W_{hs}s_1 + b_1)$$

$$h_{t+1} = \sigma_h(W_{hf_a} \text{diag}(W_{f_a z} z) W_{f_a a} a_t + W_{hf_h} \text{diag}(W_{f_h z} z) W_{f_h h} h_t + b_h)$$

$$s_{t+1} = W_{sh} h_{t+1} + b_s$$

Por tanto, los pesos adaptativos W_{hf_h} , $W_{f_h h}$, W_{hf_a} , $W_{f_a a}$, b_h , W_{sh} y b_s se refieren a propiedades compartidas por todos los sistemas fuente y los pesos adaptativos de las matrices diagonales $\text{diag}(W_{f_h z} z)$ y $\text{diag}(W_{f_a z} z)$ se refieren a propiedades que varían entre los sistemas fuente. Es decir, los pesos adaptativos W_{hf_h} , $W_{f_h h}$, W_{hf_a} , $W_{f_a a}$, b_h , W_{sh} y b_s representan la primera componente de modelo neuronal mientras que los pesos adaptativos $\text{diag}(W_{f_h z} z)$ y $\text{diag}(W_{f_a z} z)$ representan la segunda componente de modelo neuronal. Como estos últimos pesos adaptativos son matrices diagonales, comprenden muchos menos parámetros que los primeros pesos adaptativos. Es decir, el entrenamiento de la segunda componente de modelo neuronal requiere menos tiempo y/o menos datos operativos que el entrenamiento de la primera componente de modelo neuronal.

La figura 1 ilustra una representación gráfica de la arquitectura de red neuronal recurrente de tensores factorizados descrita anteriormente. Los nodos con línea de puntos en la figura 1 indican nodos idénticos que se reproducen por conveniencia. Los nodos que tienen el símbolo \odot en su centro son "modos de multiplicación", es decir se multiplican los vectores de entrada de los nodos componente a componente. En cambio, los nodos convencionales implican el sumatorio de todos los vectores de entrada. Los nodos con el borde en negrita indican el uso de una función de activación, por ejemplo "tanh" (\cdot).

Aparte de las factorizaciones descritas anteriormente de las matrices de pesos, pueden usarse representaciones adicionales o alternativas. Por ejemplo:

- Las matrices de pesos W_{hf_h} , $W_{f_h h}$, W_{hf_a} y/o $W_{f_a a}$ pueden restringirse a la forma simétrica.
- Puede añadirse una matriz específica de sistema $\text{diag}(W_{f_h z} z)$ a la matriz de pesos W_{hh} compartida por los sistemas fuente. Esta última puede restringirse a una representación de bajo rango $W_{hh} \approx W_{hu} W_{uh}$. Además, W_{uh} puede restringirse a la forma simétrica.
- Puede hacerse que el vector de sesgo b_h sea específico de sistema, es decir depender de z .
- Cuando se fusiona información de múltiples sistemas fuente u objetivo en un modelo neuronal, pueden producirse problemas debido a sensores mal calibrados de los que se derivan los datos operativos o mediante los que se controlan las acciones. Para lidiar con artefactos que resultan de sensores mal calibrados, puede hacerse que la matriz de pesos W_{sh} y/o el vector de sesgo b_s sean específicos de sistema, es decir depender del vector z . En particular, estas matrices de pesos pueden comprender una matriz diagonal dependiente de z .

La figura 2 muestra un esquema de una realización a modo de ejemplo de la invención que comprende un sistema

TS objetivo, una pluralidad de sistemas S1, ..., SN fuente, y un controlador CTR. El sistema TS objetivo puede ser por ejemplo una turbina de gas y los sistemas S1, ..., SN fuente pueden ser, por ejemplo, turbinas de gas similares al sistema TS objetivo.

5 Cada uno de los sistemas S1, ..., SN fuente se controla mediante un controlador RLC1, RLC2, ... o RLCN de aprendizaje por refuerzo, respectivamente, basándose este último en una estrategia o política P1, P2, ... o PN de control, respectivamente. Se almacenan datos DAT1, ..., DATN operativos específicos de sistema fuente de los sistemas S1, ..., SN fuente en bases de datos DB1, ..., DBN. Los datos DAT1, ..., DATN operativos se distinguen mediante identificadores ID1, ..., IDN específicos de sistema fuente de I_{fuente} . Además, los datos DAT1, DAT2, ... o DATN operativos respectivos, se procesan según la política P1, P2, ... o PN respectiva en el controlador RLC1, RLC2, ... o RLCN de aprendizaje por refuerzo respectivo. La salida de control de la política P1, P2, ... o PN respectiva se retroalimenta al sistema S1, ... o SN fuente respectivo mediante un bucle CL de control, dando como resultado un bucle cerrado de aprendizaje para el controlador RLC1, RLC2, ... o RLCN de aprendizaje por refuerzo respectivo.

15 Por consiguiente, el sistema TS objetivo se controla mediante un controlador RLC de aprendizaje por refuerzo basado en una estrategia o política P de control. Se almacenan datos DAT operativos específicos para el sistema TS objetivo en una base DB de datos. Los datos DAT operativos se distinguen de los datos DAT1, ..., DATN operativos de los sistemas S1, ..., SN fuente mediante un identificador ID específico de sistema objetivo de $I_{objetivo}$. Además, los datos DAT operativos se procesan según la política P en el controlador RLC de aprendizaje por refuerzo. La salida de control de la política P se retroalimenta al sistema TS objetivo mediante un bucle CL de control, dando como resultado un bucle cerrado de aprendizaje para el controlador RLC de aprendizaje por refuerzo.

25 El controlador CTR comprende un procesador PROC, una red RNN neuronal recurrente, y un generador PGEN de política de aprendizaje por refuerzo. La red RNN neuronal recurrente implementa un modelo neuronal que comprende una primera componente NM1 de modelo neuronal que va a someterse a entrenamiento sobre las propiedades compartidas por todos los sistemas S1, ..., SN fuente y una segunda componente NM2 de modelo neuronal que va a someterse a entrenamiento sobre las propiedades que varían entre los sistemas S1, ..., SN fuente, es decir sobre propiedades específicas de sistema fuente. Tal como ya se mencionó anteriormente, la primera componente NM1 de modelo neuronal está representada por los pesos adaptativos W_{hfh} , W_{fth} , W_{hfa} , W_{faa} , b_h , W_{sh} y b_s mientras que la segunda componente NM2 de modelo neuronal está representada por los pesos adaptativos $diag(W_{fhz} z)$ y $diag(W_{faz} z)$.

35 Por medio de la red RNN neuronal recurrente, el generador PGEN de política de aprendizaje por refuerzo genera las las políticas o estrategias P1, ..., PN, y P de control. Una política P1, ..., PN, P generada respectiva se retroalimenta entonces a un controlador RLC 1, ..., RLCN, o RLC de aprendizaje por refuerzo respectivo, tal como se indica por medio de una flecha FB en negrita en la figura 2. Con ello, se cierra un bucle de aprendizaje y las políticas P1, ..., PN y/o P generadas se ejecutan en bucle cerrado con los sistemas S1, ..., SN y/o TS dinámicos.

40 El entrenamiento de la red RNN neuronal recurrente comprende dos fases. En una primera fase, se somete a entrenamiento un modelo neuronal conjunto sobre los datos DAT1, ..., DATN operativos de los sistemas S1, ..., SN fuente. Con este fin, los datos DAT1, ..., DATN operativos se transmiten junto con los identificadores ID1, ..., IDN específicos de sistema fuente desde las bases DB1, ..., DBN de datos al controlador CTR. En esta primera fase de entrenamiento, la primera componente NM1 de modelo neuronal se somete a entrenamiento sobre las propiedades compartidas por todos los sistemas S1, ..., SN fuente y la segunda componente NM2 de modelo neuronal se somete a entrenamiento sobre las propiedades que varían entre los sistemas S1, ..., SN fuente. En este caso, los sistemas S1, ..., SN fuente y su datos DAT1, ..., DATN operativos se distinguen por medio de los identificadores ID1, ..., IDN específicos de sistema de I_{fuente} representados por el vector z.

50 En una segunda fase, la red RNN neuronal recurrente se somete a entrenamiento adicional por medio de los datos DAT operativos del sistema TS objetivo. En este caso, los parámetros W_{hfh} , W_{fth} , W_{hfa} , W_{faa} , b_h , W_{sh} y b_s compartidos que representan la primera componente NM1 de modelo neuronal y adaptados en la primera fase se usan de nuevo y permanecen fijos mientras que los parámetros específicos de sistema $diag(W_{fhz} z)$ y $diag(W_{faz} z)$ que representan la segunda componente NM2 de modelo neuronal se someten a entrenamiento adicional por medio de los datos DAT operativos del sistema TS objetivo. La red RNN neuronal recurrente distingue los datos DAT operativos del sistema TS objetivo de los datos DAT1, ..., DATN operativos de los sistemas S1, ..., SN fuente por medio del identificador ID específico de sistema objetivo.

60 Debido al hecho de que la estructura general de la dinámica de la familia de sistemas S1, ..., SN fuente similares se aprende en la primera fase de entrenamiento, puede completarse la adaptación de los parámetros específicos de sistema de un sistema TS objetivo posiblemente no observado en un plazo de segundos a pesar de la alta complejidad del modelo global. Al mismo tiempo, sólo se requieren unos pocos datos DAT operativos para lograr un bajo error del modelo en el sistema TS objetivo. Además, el modelo neuronal del sistema TS objetivo es más robusto frente a sobreajuste, lo cual aparece como problema común cuando sólo están disponibles pequeñas cantidades de datos DAT operativos, en comparación con un modelo que no aprovecha el conocimiento anterior de los sistemas S1, ..., SN fuente. Con la presente invención sólo quedan por determinar las peculiaridades en las que el sistema TS

objetivo difiere de los sistemas S_1, \dots, S_N fuente.

Hay varias maneras de diseñar los procedimientos de entrenamiento para obtener transferencia de conocimiento desde sistemas S_1, \dots, S_N fuente hasta el sistema TS objetivo incluyendo, pero sin limitarse a, las siguientes variantes:

Dado un modelo neuronal conjunto que se sometido a entrenamiento con datos DAT_1, \dots, DAT_N operativos procedentes de un número suficiente de sistemas S_1, \dots, S_N fuente, y dado un nuevo sistema TS objetivo que es similar a los sistemas S_1, \dots, S_N fuente con los que se sometió a entrenamiento el modelo neuronal conjunto, se vuelve muy eficiente en cuanto a datos obtener un modelo neuronal exacto para el sistema TS objetivo similar. En este caso, los parámetros $W_{hfh}, W_{fhh}, W_{hfa}, W_{faa}, b_h, W_{sh}$ y b_s compartidos del modelo neuronal conjunto pueden congelarse y sólo los parámetros específicos de sistemas $diag(W_{fhz}, z)$ y $diag(W_{faz}, z)$ se someten a entrenamiento adicional con los datos DAT operativos del nuevo sistema TS objetivo. Dado que el número de parámetros específicos de sistema es normalmente muy pequeño, sólo se requieren muy pocos datos operativos para la segunda fase de entrenamiento. La idea subyacente es que los datos DAT_1, \dots, DAT_N operativos de un número suficiente de sistemas S_1, \dots, S_N fuente usados para el entrenamiento del modelo neuronal conjunto contienen suficiente información para el modelo neuronal conjunto para distinguir entre la dinámica general de la familia de sistemas S_1, \dots, S_N fuente y las características específicas de sistema fuente. La dinámica general se codifica en los parámetros $W_{hfh}, W_{fhh}, W_{hfa}, W_{faa}, b_h, W_{sh}$ y b_s compartidos permitiendo una transferencia eficiente del conocimiento al nuevo sistema TS objetivo similar para el que sólo se necesita aprender los pocos aspectos característicos en la segunda fase de entrenamiento.

Para un nuevo sistema TS objetivo que no es lo suficientemente similar a los sistemas S_1, \dots, S_N fuente con los que se sometió a entrenamiento el modelo conjunto, la dinámica general aprendida por el modelo neuronal conjunto puede diferir demasiado de la dinámica del nuevo sistema TS objetivo para transferir el conocimiento al nuevo sistema TS objetivo sin adaptación adicional de los parámetros compartidos. Este también puede ser el caso si el número de sistemas S_1, \dots, S_N fuente usados para someter el modelo neuronal conjunto a entrenamiento es demasiado pequeño como para extraer conocimiento suficiente de la dinámica general de la familia global de sistemas.

En ambos casos, puede resultar ventajoso adaptar los pesos $W_{hfh}, W_{fhh}, W_{hfa}, W_{faa}, b_h, W_{sh}$ y b_s adaptativos compartidos también durante la segunda fase de entrenamiento. En este caso, los datos DAT_1, \dots, DAT_N operativos usados para el entrenamiento del modelo neuronal conjunto se extienden por los datos DAT operativos del nuevo sistema TS objetivo y todos los pesos adaptativos permanecen libres para la adaptación también durante la segunda fase de entrenamiento. Los pesos adaptativos sometidos a entrenamiento en la primera fase de entrenamiento del modelo neuronal conjunto se usan para inicializar un modelo neuronal del sistema TS objetivo, siendo ese modelo neuronal una simple extensión del modelo neuronal conjunto que contiene un conjunto adicional de pesos adaptativos específicos para el nuevo sistema TS objetivo. Por tanto, el tiempo requerido para la segunda fase de entrenamiento puede reducirse significativamente porque la mayoría de los parámetros ya están inicializados a valores correctos en el espacio de parámetros y sólo se necesita un poco de entrenamiento adicional para que el modelo neuronal conjunto extendido alcance la convergencia.

Las variaciones de este enfoque incluyen congelar un subconjunto de los pesos adaptativos y usar subconjuntos de los datos DAT_1, \dots, DAT_N, DAT operativos para entrenamiento adicional. En vez de inicializar el modelo neuronal conjunto extendido con los pesos adaptativos del modelo neuronal conjunto inicial, esos pesos adaptativos pueden inicializarse aleatoriamente, y el modelo neuronal extendido puede someterse a entrenamiento adicional desde cero con datos de todos los sistemas S_1, \dots, S_N y TS.

La invención permite aprovechar información o conocimiento de una familia de sistemas S_1, \dots, S_N fuente con respecto a dinámica de sistema posibilitando un entrenamiento eficiente en cuanto a datos de una simulación de red neuronal recurrente para un conjunto completo de sistemas de tipo similar o igual. Este enfoque facilita un inicio con impulso cuando se despliega una red neuronal de aprendizaje en un nuevo sistema TS objetivo específico, es decir logra un rendimiento de optimización significativamente mejor con pocos datos DAT operativos del nuevo sistema TS objetivo en comparación con un modelo de aprendizaje sin tal transferencia de conocimiento.

Ventajas adicionales de tal compartición de información entre modelos de aprendizaje para sistemas similares comprenden una mejor capacidad de ajuste a condiciones ambientales, por ejemplo si los diferentes sistemas están ubicados en climas diferentes. El modelo de aprendizaje también puede generalizarse hacia diferentes clases de degradación, proporcionando capacidades de optimización mejoradas para situaciones raras o poco frecuentes porque puede usarse la información combinada recopilada a partir de todos los sistemas.

REIVINDICACIONES

1. Método para controlar un sistema (TS) objetivo basándose en datos (DAT1, ..., DATN) operativos de una pluralidad de sistemas (S1, ..., SN) fuente, que comprende:
 - a) recibir datos (DAT1, ..., DATN) operativos de los sistemas (S1, ..., SN) fuente, distinguiéndose los datos operativos mediante identificadores (ID1, ..., IDN) específicos de sistema fuente,
 - b) entrenar por medio de una red (RNN) neuronal un modelo neuronal basado en los datos (DAT1, ..., DATN) operativos recibidos de los sistemas (S1, ..., SN) fuente teniendo en cuenta los identificadores (ID1, ..., IDN) específicos de sistema fuente, en el que se entrena una primera componente (NM1) de modelo neuronal sobre propiedades compartidas por los sistemas (S1, ..., SN) fuente y se entrena una segunda componente (NM2) de modelo neuronal sobre propiedades que varían entre los sistemas (S1, ..., SN) fuente,
 - c) recibir datos (DAT) operativos del sistema objetivo,
 - d) someter a entrenamiento adicional el modelo (RNN) neuronal sometido a entrenamiento basándose en los datos (DAT) operativos del sistema (TS) objetivo, en el que se le da preferencia a un entrenamiento adicional de la segunda componente (NM2) de modelo neuronal sobre un entrenamiento adicional de la primera componente (NM1) de modelo neuronal, y
 - e) controlar el sistema (TS) objetivo por medio de la red (RNN) neuronal sometida a entrenamiento adicional.
2. Método según la reivindicación 1, en el que la primera componente (NM1) de modelo neuronal está representada por primeros pesos ($W_{hff}, W_{fhh}, W_{faa}, W_{hfa}, b_h$) adaptativos, y la segunda componente (NM2) de modelo neuronal está representada por segundos pesos (W_{fzZ}, W_{fazZ}) adaptativos.
3. Método según la reivindicación 2, en el que el número de los primeros pesos ($W_{hff}, W_{fhh}, W_{faa}, W_{hfa}, b_h$) adaptativos es varias veces mayor que el número de los segundos pesos (W_{fzZ}, W_{fazZ}) adaptativos.
4. Método según la reivindicación 2, en el que los primeros pesos ($W_{hff}, W_{fhh}, W_{faa}, W_{hfa}, b_h$) adaptativos comprenden una primera matriz de pesos y los segundos pesos (W_{fzZ}, W_{fazZ}) adaptativos comprenden una segunda matriz de pesos.
5. Método según la reivindicación 4, en el que para determinar pesos adaptativos ($W_{hff}, W_{fhh}, W_{faa}, W_{hfa}, b_h, W_{fzZ}, W_{fazZ}$) del modelo neuronal, se multiplica la primera matriz de pesos por la segunda matriz de pesos.
6. Método según la reivindicación 4, en el que la segunda matriz de pesos es una matriz diagonal.
7. Método según la reivindicación 1, en el que la primera componente (NM1) de modelo neuronal no se somete a entrenamiento adicional.
8. Método según la reivindicación 2, en el que cuando se somete a entrenamiento adicional el modelo (RNN) neuronal sometido a entrenamiento un primer subconjunto de los primeros pesos ($W_{hff}, W_{fhh}, W_{faa}, W_{hfa}, b_h$) adaptativos se mantiene sustancialmente constante mientras que un segundo subconjunto de los primeros pesos ($W_{hff}, W_{fhh}, W_{faa}, W_{hfa}, b_h$) adaptativos se somete a entrenamiento adicional.
9. Método según la reivindicación 1, en el que el modelo neuronal es un modelo de aprendizaje por refuerzo.
10. Método según la reivindicación 1, en el que

la red (RNN) neuronal funciona como red neuronal recurrente.

- 5
11. Método según la reivindicación 1, en el que
- durante el entrenamiento del modelo neuronal
- 10
- se determina si el modelo neuronal refleja una distinción entre las propiedades compartidas por los sistemas (S1, ..., SN) fuente y las propiedades que varían entre los sistemas (SN, ..., SN) fuente, y
- se ve afectado el entrenamiento del modelo neuronal dependiendo de esa determinación.
12. Método según la reivindicación 1, en el que
- 15
- se ejecutan las políticas (P1, ..., PN, P) que resultan del modelo neuronal sometido a entrenamiento en un bucle cerrado de aprendizaje con el sistema (TS) objetivo técnico.
13. Controlador para controlar un sistema (TS) objetivo basándose en datos (DAT1, ..., DATN) operativos de una pluralidad de sistemas (S1, ..., SN) fuente, adaptado para realizar el método según la reivindicación 1.
- 20
14. Producto de programa informático para controlar un sistema objetivo basándose en datos (DAT1, ..., DATN) operativos de una pluralidad de sistemas (S1, ..., SN) fuente, adaptado para realizar el método según la reivindicación 1.

FIG 1

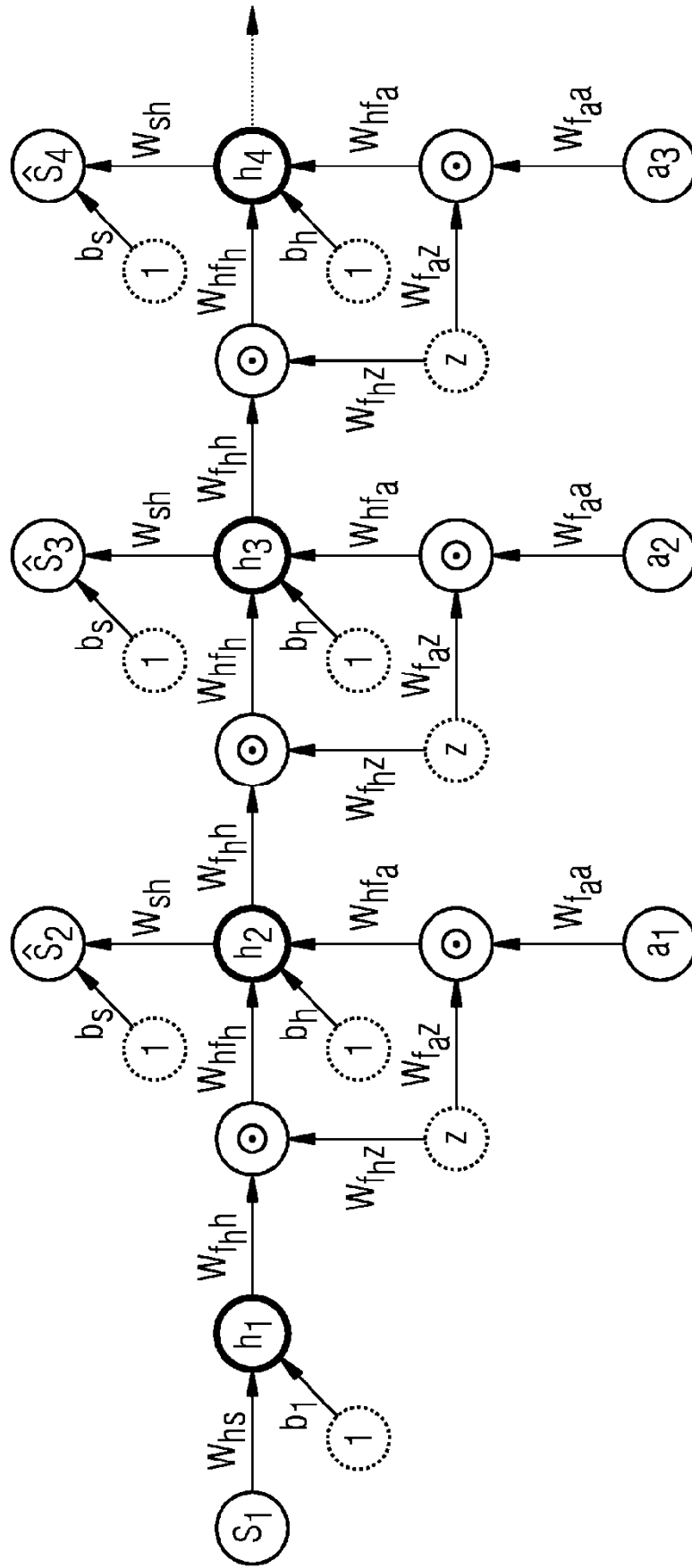


FIG 2

