

## Estudio sobre las herramientas de TRIZ para determinar su utilidad en la automatización con el diseño funcional

Vicente Chulvi <sup>1\*</sup>

Rosario Vidal <sup>1</sup>

<sup>1</sup>GID, Dpt. Enginyeria Mecànica I Construcció, Universitat Jaume I, Castellón, Spain

### Abstract

TRIZ is the Russian acronym for the Theory of Inventive Problem Solving (TIPS). It emerges from the study of hundreds of thousands patents, and it consists on a set of model-based tools for idea and innovation solutions generation, in order to solve problems. These tools have a creative and innovative potential that can be used for designers by means of Computer Aided Inventing (CAI) tools. The present work shows a study over different TRIZ tools and their capacity to be automated in design process, through a Knowledge Based System (KBS) based on the Behaviour-driven Function-Environment-Structure (B-FES) framework, and aided by NIST functional basis and B-Cube model terminologies.

*Keywords: TRIZ; functional design; FBS; Computer Aided Inventing (CAI)*

### Resumen

TRIZ, acrónimo ruso para la Teoría de Resolución de Problemas Inventivos, surge del estudio de cientos de miles de patentes y consiste en un conjunto de herramientas basadas en modelos para la generación de ideas y soluciones innovadoras para resolver problemas. Estas herramientas contienen un potencial creativo e innovador que puede ser aprovechado por los diseñadores a través de las herramientas de invención asistida por ordenador, CAI (Computer Aided Inventing). El presente trabajo presenta un estudio sobre diferentes herramientas de TRIZ y su capacidad para lograr su automatización en el proceso de diseño, a través de la vinculación con un KBS (Knowledge Based System) basado en el esquema B-FES (Behaviour-driven Function-Environment-Structure), auxiliado de las terminologías de la funcional basis del NIST y del modelo B-Cube.

*Palabras clave: TRIZ; diseño funcional; FBS; Invención Asistida por Ordenador (CAI)*

## 1 Introducción

TRIZ es el acrónimo ruso de la Teoría de Resolución de Problemas Inventivos (Teoriya Resheniya Izobretatelskikh Zadatch) (Altshuller, 1984), que surgió del estudio de cientos de miles de patentes. De dicho trabajo han surgido un conjunto de herramientas basadas en modelos para la generación de ideas y soluciones innovadoras para resolver problemas (Altshuller, 1997a,1999; Altshuller, 1997b; Belski, 2007). El uso de las herramientas de TRIZ en aplicaciones de invención asistida por ordenador, CAI (Computer Aided Inventing), ha sido objeto de estudio por diversos investigadores (Cascini, 2004a; Cascini, 2005; Cavallucci, 2004; Ikoenko, 2004). Por un lado, una buena parte de estos estudios discuten sobre la forma de integrar la creatividad de TRIZ con el conocimiento semántico de un modo teórico, con la finalidad de asistir en la fase de diseño (Cavallucci, 2001; Mulet, 2005), mientras que por otro, algunos investigadores han intentado poner estos modelos en práctica (Soo, 2006; Verbitsky, 2004; Zhang, 2005). Estos últimos hacen un mayor énfasis en el uso de las taxonomías y las ontologías para poder trasladar los modelos de la teoría a la práctica. Es precisamente por este motivo que el estudio de las taxonomías necesarias para la vinculación de herramientas de TRIZ con aplicaciones informáticas mediante herramientas semánticas de Ingeniería Basada en el Conocimiento, KBE (Knowledge Based Engineering), ha sido objeto de numerosos trabajos (Cascini, 2004b; Chang, 2004; Chang, 2003; Dewulf, 2006).

La propuesta del presente estudio para automatizar el diseño funcional y vincularlo con aplicaciones CAI se centra en el marco FBS (Function – Behaviour – Structure) (Gero, 1990), diferenciando aquellas estructuras que pertenecen al diseño de aquellas que forman parte del medio, que denominaremos restricciones. Por tanto, es necesario establecer una taxonomía que relacione TRIZ con el modelo FBS, como ya han defendido otros autores (Cao, 2007; Hirtz, 2002; Pinyayev, 2006). Una mejor visualización de cómo se relacionan las definiciones de TRIZ con el marco FBS se puede conseguir con la ayuda de los términos utilizados en la jerarquía de abstracciones, AH (Abstract Hierarchy) (Bergmann, 1996; Lind, 1999). La AH se define como una descomposición funcional a cinco niveles, que se utiliza para modelar los marcos de trabajo o, más comúnmente, los dominios para sistemas socio-técnicos complejos. Dicho de un modo más sencillo, AH describe un sistema con varios niveles de abstracción utilizando relaciones de cómo y porqué (tabla 1).

<b>AH</b>	<b>FBS-TRIZ</b>	<b>Descripción</b>
Propósito funcional (FP)	Función	Funciones. Objetivos y metas del sistema.
Función Abstracta (AF)	Principio	Leyes y principios que rigen el sistema. Soluciones genéricas provistas por TRIZ.
Función generalizada (GF)	Comportamiento	Comportamiento de las funciones. Proceso que relaciona leyes y principios con la función que se desea conseguir.
Función física (PFn)	Estructura	Componentes o elementos con los cuales se identifica el comportamiento.
Forma física (PFo)	Cualidad	Propiedad, condición, característica de los elementos.

Tabla 1. Términos y definiciones de la AH y su relación con TRIZ y el esquema FBS

Si comparamos los términos del marco FBS y TRIZ con los cinco niveles de abstracción propuestos por la AH, se puede apreciar que una función (FP) necesita estar relacionada a través de un comportamiento (GF) para poder conseguir soluciones con las herramientas de TRIZ, ya que estas trabajan al nivel de AF. También se sabe que algunas de estas herramientas de TRIZ requieren de las estructuras (PFn) con sus cualidades (PFo), que están ligadas a los comportamientos por definición. De esta comparación y de un trabajo previo sobre la materia (Chulvi, 2007) se ha llegado a la conclusión de que las herramientas de TRIZ funcionan principalmente con comportamientos, y por tanto su vinculación con el diseño funcional debe realizarse a través del nivel de comportamientos.

Así pues, el objetivo del presente estudio es el de determinar el nivel de utilidad de las diferentes herramientas de TRIZ para lograr su automatización en el proceso de diseño, a través de la vinculación con un KBS (Knowledge Based System) basado en el esquema B-FES. Para representar el nivel de funciones se han empleado las funcional basis provistas por el NIST (Hirtz, 2002), mientras que para representar el nivel de comportamientos se ha utilizado la terminología del modelo B-Cube (Chulvi, 2008). Este modelo propone un planteamiento tridimensional que utiliza definiciones a modo de conceptos de los comportamientos. Es decir, el comportamiento queda definido como un vector tridimensional (X,Y,Z) determinado por sus características y cualidades, en lugar de ser expresado como una sola palabra o taxón. Para la representación del nivel de estructuras o de las restricciones del medio no se ha empleado ninguna taxonomía más allá de la propia definición de las palabras empleadas para tal efecto, al tratarse de elementos más concretos.

## 2 Métodos

### 2.1 TRIZ

TRIZ comenzó hace más de seis décadas con la hipótesis de que existen unos principios inventivos universales que son la base para las creaciones innovadoras que hacen avanzar la tecnología, i que estos principios pueden ser identificados y codificados para hacer el proceso inventivo más previsible. Desde entonces hasta hoy, más de dos millones y medio de patentes han sido analizadas y clasificadas para conocer sus principios de innovación, para concluir en la elaboración un conjunto de herramientas para guiar en el proceso de búsqueda de soluciones innovadoras. Las principales herramientas de TRIZ sobre las que se ha basado el presente estudio, dónde se han utilizado los programas informáticos Creax Innovation Suite (<http://www.creaxinnovationsuite.com/>) y Techoptimizer (<http://www.invention-machine.com/>), son las siguientes:

#### **Matriz de contradicciones y principios inventivos**

El concepto de contradicción es la parte fundamental, y probablemente más popular, del conjunto de herramientas de TRIZ. Una contradicción queda definida como un impedimento físico o técnico para encontrar una solución. La matriz de contradicciones, compuesta por 39 factores o parámetros ingenieriles, ayuda a identificar una contradicción, bien de tipo técnica donde mejorar un factor empeora otro, bien de tipo física donde el mismo factor debe existir y no existir en el mismo espacio y/o tiempo, y orienta hacia cual de los 40 principios inventivos se ha de mirar para resolver las contradicciones encontradas. Los principios inventivos consisten en los enfoques clásicos para resolver problemas resolutivos. Estos provienen del estudio de millones de patentes y todas las invenciones se pueden catalogar dentro de ellos.

#### **Líneas de evolución**

Las líneas de evolución, también conocidas como leyes de la evolución de los sistemas técnicos, proceden de la idea de que todos los sistemas técnicos siguen los mismos patrones de evolución a pesar de que pertenezcan a campos muy dispares. Así, existen 31 líneas evolutivas que permiten predecir los saltos tecnológicos capaces de resolver el problema inventivo. El estado de evolución del elemento objeto del estudio debería ser analizado en cada una de dichas 31 líneas evolutivas con la finalidad de priorizar las líneas sobre las que se debe actuar.

#### **Bases de datos de funciones**

Esta es una herramienta basada en el conocimiento presente en la mayoría de los programas informáticos basados en TRIZ existentes en el mercado. Consiste en una base de datos que enumera las diferentes maneras conocidas de llevar a cabo ciertas funciones clave. La realización de las búsquedas en dicha base generalmente requiere de “que” quiere hacerse y “sobre que” sustancia quiere hacerse, es decir, función objetivo y elemento objeto. Nótese la similitud de estos dos elementos con los campos F y las sustancias  $S_2$  en el siguiente punto.

#### **Su-Fields**

La herramienta de Su-Fields requiere de una conversión del sistema a campos y sustancias interrelacionados entre sí formando triadas, es decir, conjuntos de tres elementos interconexos, frecuentemente dos sustancias y un campo (figura 1). El análisis de este sistema representado por campos y sustancias nos permitirá identificar aquellas triadas con conflictos para actuar sobre ellas por medio de cinco reglas:

La regla 1 nos dice que hay que sustituir la sustancia  $S_2$  por otra sustancia  $S_3$ , de modo que genere un nuevo campo  $F_2$  cuya acción no sea perjudicial o no deseada sobre  $S_1$ . La regla 2

indica que hay que añadir una nueva sustancia  $S_3$  que genere un nuevo campo  $F_2$  que actúe sobre  $S_2$  y lo modifique, de modo que este no ejerza negativamente sobre  $S_1$ . La regla 3 pide añadir una nueva sustancia  $S_3$  que genere un nuevo campo  $F_2$  que actúe sobre  $S_1$  y lo modifique, de modo que la acción de  $S_2$  sobre este no sea negativa. Por su parte, la regla 4 dice de añadir una nueva sustancia  $S_3$  intermedia entre  $S_1$  y  $S_2$ , a través de la cual actúe  $F_1$ , que genere un nuevo campo  $F_2$  que actúe sobre  $S_1$  o  $S_2$  y lo modifique para evitar su interacción negativa. Por último, la regla 5 nos dice que hay que añadir una nueva sustancia  $S_3$  que genere un nuevo campo  $F_2$  que actúe sobre  $S_1$  y  $S_2$  y los modifique para evitar su interacción negativa.

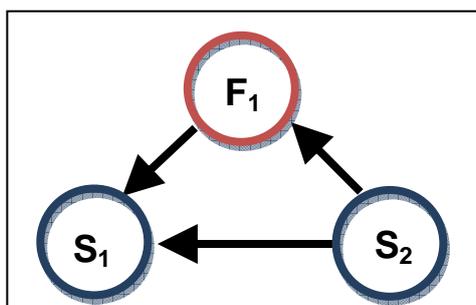


Figura 1. Triada: conjunto básico del análisis Su-Field

### Principio de idealidad (IFR)

El principio de idealidad consiste en plantear la solución final óptima, aunque esta sea imposible de realizar. Según este planteamiento, la máquina ideal es aquella que permite llevar a cabo la función requerida sin consumir recursos, generar residuos, ni utilizar espacio. Puesto que este nivel de idealidad no se puede alcanzar, la aplicación del IFR nos dice que el sistema se debe de ir desevolucionando desde este máximo óptimo, hasta llegar al punto en que ya sea viable el nuevo diseño. De este modo nos hemos asegurado que nuestro diseño es el más cercano a la idealidad, y por tanto el más evolucionado.

### 2.2 Desarrollo del ejemplo usado

El ejemplo desarrollado para analizar la aplicabilidad de las herramientas de TRIZ en la automatización del diseño funcional consiste en un útil sencillo para limpiar los cascos de los barcos cuando están en el agua. El KBS desarrollado debe partir de unos inputs por parte del diseñador. Estos serán los requerimientos del útil a diseñar, es decir, las funciones que se piden al mismo, y las restricciones que nos encontraremos en el medio en el cual va a funcionar el útil y que van a afectar al mismo. A partir de estos datos se elabora el nivel de comportamientos, y de este el de estructuras, que pueden, a su vez, requerir de nuevos comportamientos.

La figura 1 representa el desarrollo del modelo en el marco B-FES. El nivel de funciones viene expresado por la taxonomía del NIST (en la figura 2 representado por su terminología completa: función primaria / función secundaria / función terciaria), por lo que el requisito de limpiar viene determinado por "remove", mientras que el requisito de ser movido por un operario se expresa como "translate". De estas dos funciones derivan los comportamientos, expresados en términos del modelo B-Cube, (2,1,1) y (1,1,2), respectivamente. (2,1,1) se refiere a que el comportamiento afecta a la conexión topológica de las estructuras ( $x=2$ ), de un modo acumulativo y no homeomérico ( $y=1$ ), dónde dicha conexión topológica se encuentra al principio y se pierde a través del comportamiento ( $z=1$ ). Por su parte, (1,1,2) implica un comportamiento que afecta a la localización espacial, de un modo acumulativo y no homeomérico, con indiferencia del sentido en que se ve afectada dicha localización espacial. El nivel de comportamientos se ve completado por la pérdida de conexión topológica de modo acumulativo y homeomérico (2,2,1) producida por la corrosión del

medio, y por la acción de mantener separada de modo acumulativo y homeomérico la localización espacial (1,2,1) de los componentes que requieren actuar en los dos medios diferentes (tierra y agua). El esquema se ve completado por el nivel de las estructuras, donde se ve que el útil está formado por una tubería (pipe 1) que lleva unidos un cepillo (brush) y un mango (handle).

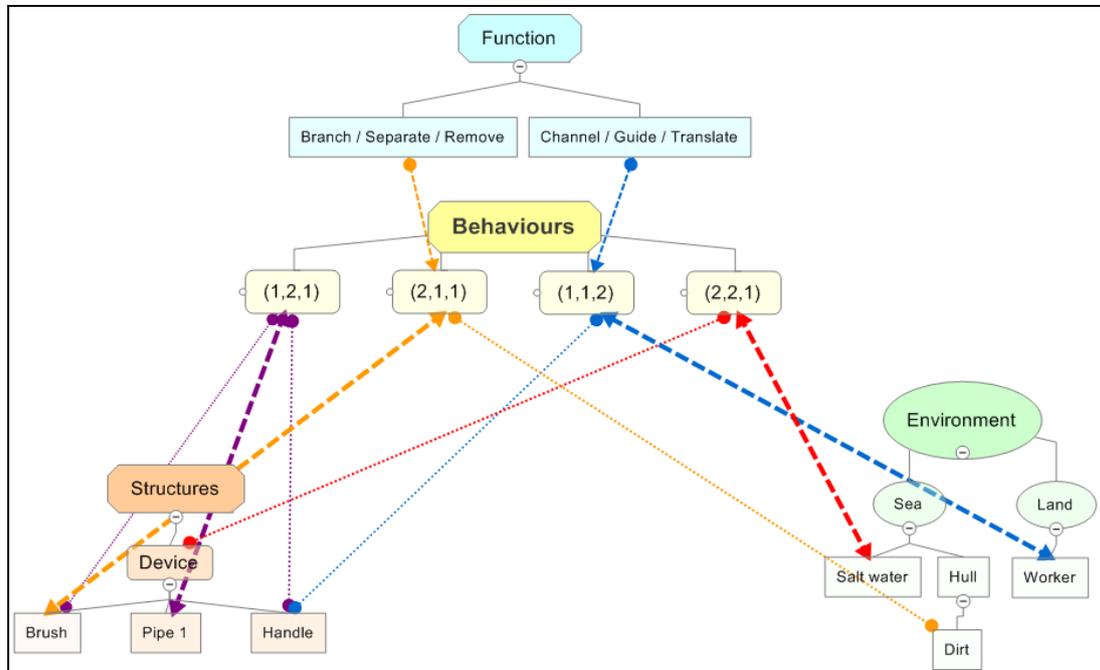


Figura 2. Desarrollo del modelo en el marco B-FES

### 3 Discusión

#### 3.1 Principio de idealidad

Esta herramienta se debe tener en cuenta antes de comenzar el desarrollo del modelo, pues más bien consiste en un replanteamiento del problema. En nuestro ejemplo, el diseñador pide un útil para limpiar los cascos de los barcos manejado por un operario. Según el principio de idealidad (figura 3), se debe partir de aquello que es lo más óptimo: que el barco se limpie solo. A partir de este punto, y puesto que de momento se ve imposible de alcanzar, se puede ir hacia atrás por diferentes ramas que nos permiten acercarnos a la idealidad. Por un lado, se puede ver que si no se incrusta suciedad al casco del barco no hace falta limpiarlo, lo que puede requerir de impregnar el casco con algún aditivo para evitar que se incruste la suciedad o que se incruste con menor fuerza. Mientras que otro camino puede ser el de reducir el esfuerzo del operario, es decir, sustituyendo o auxiliando al operario por otro medio de mover el útil.

Con esto se comprueba que lo que se han cambiado son los parámetros de entrada o requisitos de diseño. En el primero de los casos vemos como hemos eliminado la función “remove” referida a quitar la suciedad por otra referida a añadir un aditivo, “couple”. Por su parte, en el segundo de los casos hemos modificado (o eliminado) una restricción, “worker”, que el enunciado lo defina como el sujeto que iba a mover el útil. Por tanto, con esta herramienta lo que hemos conseguido es un enunciado nuevo, que posiblemente nos conducirá a una solución mejor, pero que lo que ha modificado es la entrada en el sistema. Así pues, esta herramienta se considera de mucha utilidad para ayudar al diseñador sobre que parámetros de entrada debe introducir en el sistema, pero no sirve una vez definidos los objetivos y puesto en marcha el proceso de diseño.

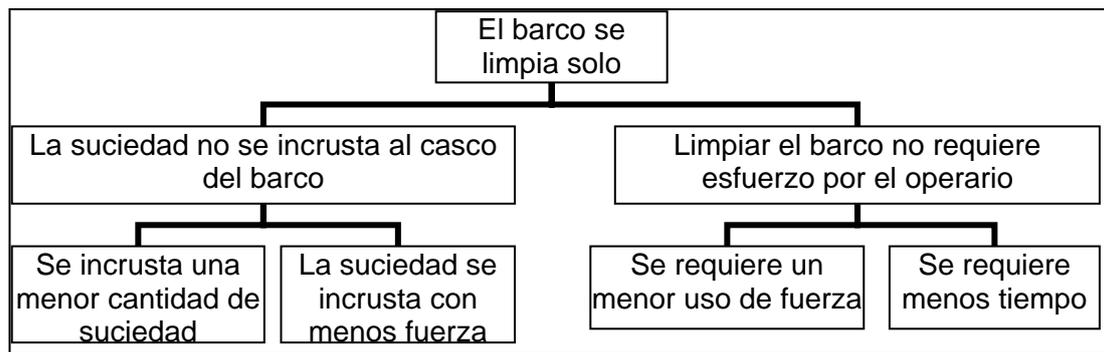


Figura 3. Aplicación del principio de idealidad

### 3.2 Bases de datos de funciones

Puesto que dentro de un KBS, las búsquedas de información trabajan con bases de datos, la herramienta de bases de datos de funciones parece más fácil de conectar con el sistema. Para analizar este caso utilizamos la base de datos de funciones que aparece en el programa informático Creax Innovation Suite. Lo primero en que nos fijamos es en que dicha herramienta pide como entradas la función a realizar y el elemento sobre el que actúa, esto es, la estructura o restricción con la que interactúa. Como hemos visto en la imagen 1, en el esquema B-FES estos elementos no están directamente relacionados con las funciones, que son más abstractas, sino que van relacionadas a los comportamientos. Esto induce a pensar que la base de datos de funciones coincide más con una base de datos de comportamientos, según la terminología empleada.

Siguiendo con el ejemplo, tomamos para analizar el comportamiento (2,1,1), que aparece vinculado a la restricción suciedad contenida en la restricción casco en el medio mar. Se puede deducir que el resultado de la aplicación de la base de datos de funciones será la estructura capaz de llevar a cabo dicho comportamiento, por lo que esta es una herramienta a emplear durante el desarrollo del modelo. Este comportamiento, junto con las restricciones vinculadas a él, se puede expresar como “un proceso mediante el cual la suciedad pierda su conexión topológica con el casco del barco, que se encuentra sumergido en el mar”. Al acceder manualmente a la base de datos de funciones de la aplicación CREAX, esta definición es asimilable tanto a la palabra *clean* (limpiar, separar o quitar impurezas) como a *polish* (pulir, separar o quitar la capa superficial irregular). Además, la suciedad, que es aquello que queremos quitar, añade la condición de sólido. De esto podemos deducir que, para combinar esta aplicación con el diseño funcional se hace necesario, por un lado establecer una correlación entre las taxonomías del modelo B-Cube y la empleada por la aplicación de TRIZ que vayamos a utilizar para identificar sus funciones, con la desventaja de que cada aplicación de TRIZ posee su propia taxonomía de funciones. Y, por otro lado, la base de datos de estructuras (y de restricciones) del sistema deberá incorporar información sobre el estado físico (sólido, líquido, gas o campo) de cada estructura para que la librería de TRIZ sea capaz de recuperarla y aplicarla en la base de datos de funciones.

Al cruzar estos parámetros, las funciones *clean* y *polish* y el estado sólido de la suciedad, en la herramienta de TRIZ, se obtiene una batería combinada de 40 posibles soluciones. Obviamente, el número de posibles soluciones es muy elevado, y se puede observar que no todas van a ser aplicables al caso. Por ejemplo, la explosión no es posible, pues dañariamos el casco del barco, aplicar un disolvente no parece muy factible, pues el medio líquido del mar diluiría su concentración (aparte de ser dañino para el medio ambiente), o quemar tampoco es conveniente por ambas razones, barco y mar. Así pues, la aplicación de esta herramienta de TRIZ requerirá de un método de filtrado de soluciones para determinar cuáles son viables y cuáles no. Como hemos podido ver, el resto de restricciones afectadas

y el mismo medio sirven como condiciones para efectuar este filtrado. En este caso, la función debe de poder realizarse dentro de un medio líquido y no debe dañar al barco. Con las funciones resultantes del filtrado es con las que se deberá acceder nuevamente a la base de datos de estructuras, para ver que estructuras son capaces de llevar a cabo dichas funciones y devolverlas al sistema.

Para el caso del ejemplo podemos ver la variedad de soluciones viables que surgen de la aplicación de la herramienta: Cepillo (ya existente en el modelo original), cavitación acústica, lijadora; aunque también existen otras teóricamente viable, pero más difíciles de aplicar: campos magnéticos, electrolisis.

### 3.3 Su-Fields

Para actuar con la herramienta de los campos-sustancia lo primero es fijarse en la fácil asimilación de los comportamientos con los campos y de las estructuras y restricciones con las sustancias. Con todo ello, al igual que en el punto anterior, se requiere establecer una correlación entre las taxonomías del modelo B-Cube y la correspondiente a los campos. La ventaja de los Su-Field frente a las bases de datos de funciones en este aspecto es que todas las aplicaciones de TRIZ usan los mismos campos, mientras que cada una tiene su base de datos de funciones diferente, por lo que se puede establecer una correlación de términos única. Además, el hecho de que existan cinco reglas predefinidas para resolver los conflictos en un diagrama Su-Field induce a prever que será más fácil su automatización.

Siguiendo con el ejemplo anterior, partimos de la suciedad, que es lo que deseamos eliminar. Según la solución general de la regla 1, si tenemos una sustancia  $S_1$  es necesario añadir una sustancia  $S_2$  al sistema que genere un campo  $F_1$  para que influya sobre la sustancia  $S_1$  en el modo en que deseamos (figura 4a). En este caso tenemos un paso más avanzado, pues ya sabemos que el comportamiento asimilable al campo es el (2,1,1). Las dos primeras columnas de la tabla 2 muestran los campos y sub-campos capaces de llevar a cabo este comportamiento.

Campo	Sub-campo	Ejemplo solución	Viabilidad
Mecánico	Fricción	Cepillo Lijadora	Viable Viable
	Contacto directo	Buzo	Viable, más costoso
	Vibración	Vibrar el barco	Inviabile
	Shocks	Sacudir el barco	Inviabile
	Procesado mecánico	Fresadora	Difícil
Acústico	Cavitación	Cavitación	Viable
Térmico	Fuego	Fuego	Inviabile
Químico	Disolventes	Disolvente	Inviabile
Intermolecular	Evaporación	Resistencia Convector	Inviabile Inviabile
	Seres vivos	Peces come-suciedad	Muy difícil
Biológico	Bacterias	Bacterias come-suciedad	Muy difícil

Tabla 2. Campos aplicables al ejemplo y soluciones que pueden proporcionar

En este caso, el número de campos-solución es de 11, número bastante más reducido que en el caso de la base de datos de funciones, lo cual conlleva a un proceso de análisis más reducido. Así pues, aprovechando la idea de utilizar las restricciones y medios relacionados como condiciones para efectuar el filtrado, se puede comprobar que también se producen casos de soluciones inviables, viables y teóricamente viables pero difíciles de aplicar, como aparece detallado en la tabla 2. Para poder llevar a cabo la vinculación de esta herramienta de TRIZ con el diseño funcional será necesario además añadir a la base de datos de

estructuras del sistema información sobre qué estructuras son capaces de cumplir con cada sub-campo. También será necesario, al igual que en el caso anterior, el crear un sistema de filtrado fiable.

El resto de reglas para resolver conflictos que nos proporciona la herramienta Su-Fields permite, además de la búsqueda de estructuras-solución para nuestro diseño, el resolver los conflictos que vayan surgiendo durante la elaboración del mismo. Esto podemos comprobarlo tomando como ejemplo el modo en el que el agua salada afecta a los componentes del útil (figura 4b). Como se puede observar, en este caso la herramienta se aplica después del proceso de diseño, y no al mismo tiempo como hemos visto en el ejemplo anterior. En este caso será necesario, además, establecer que regla será más conveniente utilizar para solucionar el sistema, lo cual puede hacer más compleja la vinculación de la herramienta. También se hará necesario diferenciar entre que campos (o comportamientos) son deseados y cuales causan conflictos.

Así pues, vemos como la regla 1 no se puede aplicar, pues el agua salada es una restricción del medio y no puede ser sustituida. También vemos que la regla 2 no se puede aplicar, pues modificar el agua del mar resulta inviable, tanto por su cantidad como por sus implicaciones medioambientales. La regla 3 si que puede ser aplicada, pues si que parece viable añadir un elemento a nuestro diseño que modifique una parte del diseño. Del mismo modo, la regla 4 puede ser aplicada desde el punto de vista de que el elemento intermedio que añadamos modifique a la parte del diseño (pipe) y no a la restricción (sea wáter). La regla 5 no podrá ser aplicada por el mismo motivo de que modificar el agua del mar resulta inviable. A partir de este ejemplo se puede deducir que el criterio principal que determinará si una regla se puede o no se puede aplicar vendrá determinado por si el elemento a modificar es una estructura o una restricción. Este criterio debería de poder ser fácilmente evaluado por el KBS.

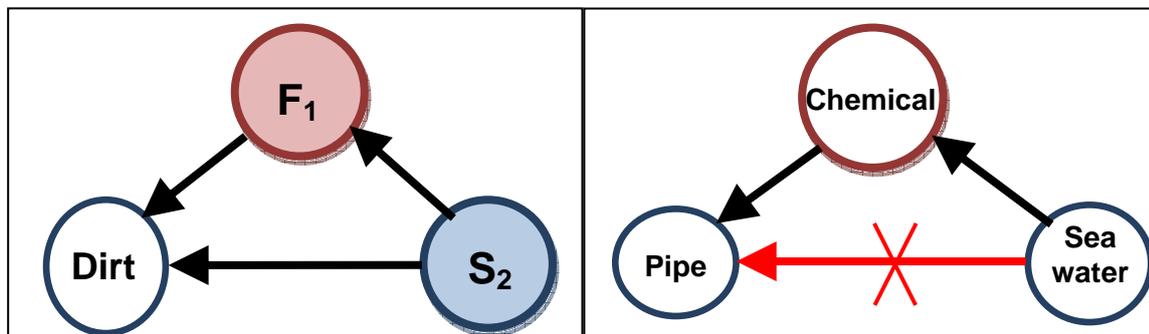


Figura 4. A) Modelo de solución aplicando la regla 1 de los Su-Fields. B) Efecto nocivo de la restricción “agua de mar” sobre la estructura “tubo”

### 3.4 Matriz de contradicciones y principios inventivos

La matriz de contradicciones se percibe en un principio como de posible aplicación tanto durante el proceso de diseño, como después del proceso de diseño. Sin embargo, en el caso de la automatización no es tan sencillo, pues aunque el diseñador pueda prever las contradicciones mientras elabora el diseño, es razonable pensar que un KBS no detectará la contradicción hasta que la vea. Así, por ejemplo, el diseñador puede deducir que cuanto más largo sea el elemento que una el mango (handle) del útil con el elemento que limpia (brush), mayor peso tendrá y más difícil será de manipular, pero sin embargo dicho elemento ha de ser largo para poder limpiar todo el casco del barco. Sin embargo un sistema automatizado no percibirá la contradicción hasta el momento en que ubique el tubo (pipe), y vea que para la longitud necesaria marcada por la restricción casco del barco (hull) el peso puede resultar excesivo, y por tanto la manipulación difícil. Es decir, hasta que el

proceso de diseño no haya concluido, el sistema no será capaz de detectar las contradicciones, y por lo tanto no podrá actuar sobre ellas.

El modo de identificarlas será al comparar los requisitos impuestos por las restricciones sobre algunos de los atributos de las estructuras. En este caso, los atributos enfrentados son “longitud” frente a “peso” de la estructura tubo (pipe). Como se puede apreciar, estos dos atributos se corresponden con sendos parámetros de la matriz de contradicciones. La matriz nos guía hasta cuatro principios inventivos, que son los siguientes:

28 – Sustitución de la mecánica: Sustituir las partes mecánicas de la estructura por otras que lleven a cabo la misma función. En este caso, al ser el tubo completamente mecánico, la opción pasa por la sustitución de la estructura por otra no mecánica. Las búsquedas por campos mostradas en el punto anterior pueden guiarnos hacia la estructura objetivo. Ejemplo: un campo electromagnético puede mantener ambos elementos a la distancia deseada.

29 – Fluido: Utilizar gases o líquidos para sustituir partes mecánicas. Al igual que el anterior, la estructura es completamente mecánica. Una solución intuitiva es hacer el tubo hueco (relleno de aire), pero para que el sistema pudiera considerar la opción debería de existir en la base de datos una estructura “tubo hueco” diferenciada de la estructura “tubo”.

30 – Cambio de parámetro: El sistema buscará estructuras iguales pero de diferente material, cuyos atributos “longitud” y/o “peso” mejoren según nuestro propósito, a cambio de perder en propiedades de otros atributos (parámetros) del mismo, como pueden ser la rigidez o la elasticidad.

40 – Compuestos: El sistema buscará estructuras iguales pero de materiales compuestos (aleaciones, fibras, etc.), de modo que los atributos “longitud” y/o “peso” mejoren.

### **3.5 Líneas de evolución**

Esta herramienta es de clara aplicación posterior al proceso de diseño, pues se trata de analizar el útil para ver en qué partes es optimizable. Este paso se prevé como más costoso de vincular con el diseño funcional, no solo por la cantidad de conocimiento que requiere la metodología, sino también por el elevado número de análisis que se deberían de realizar, pues tanto el útil como todas sus estructuras deberían de ser comparadas con la totalidad de las líneas de evolución (31 en total). Un primer paso que se puede hacer de cara a optimizar este proceso es el de diferenciar que líneas de evolución son aplicables sólo a estructuras (por ejemplo: segmentación espacial), cuales son sólo aplicables a útiles completos (por ejemplo: moni-bi-poly-varios objetos), y cuáles de ellas pueden ser aplicadas a ambos grupos (por ejemplo: materiales inteligentes). El resultado de esta primera distinción se muestra en la tabla 3, donde se puede ver que se puede llegar a reducir el número de análisis en un tercio.

Para probar esta herramienta tomamos como ejemplo la estructura “pipe 1”. Al tratarse de una estructura podemos reducir el número de líneas a analizar a 20 de las 31 originales, según la diferenciación sugerida anteriormente. Otro medio para intentar reducir aún más el número de análisis a realizar puede hacerse estableciendo una correlación entre comportamientos y líneas de evolución, para ver si estas últimas son relevantes o no con los primeros. Por tanto, del esquema de la figura 2 puede verse que la estructura objeto está relacionada directamente con el comportamiento (1,2,1), e indirectamente con (2,2,1). Así, si una línea de evolución no presenta relevancia con ninguno de los comportamientos, esta se podría descartar, y por tanto el número de análisis a realizar por el sistema se vería considerablemente reducido. La tabla 4 muestra como se establece esta selección de líneas relevantes.

<b>Estructuras</b>	<b>Instrumentos</b>	<b>Estructuras e instrumentos</b>
Segmentación del objeto. Macro a nano escala. Segmentación espacial. Segmentación superficial. Coordinación del ritmo. Coordinación de la acción. Reducción de la amortiguación. Punto de diseño. Reducción de la densidad. Redes y fibras. Incremento de la transparencia.	Reducción de conversiones de energía de n a 0. Controlabilidad. Reducción de la participación humana. Reducción de fronteras. Mono-bi-poly-varis objetos. Mono-bi-poly incremento de diferencias. Reducción de la complejidad del sistema. Incremento del uso de los sentidos. Metodología de diseño. Objetivo de compra del consumidor. Evolución del mercado.	Evolución geométrica de construcciones lineales. Evolución geométrica de construcciones volumétricas. Dinamización. Mono-bi-poly objetos similares. Incremento de asimetría. Incremento del uso de color. Materiales inteligentes. Grados de libertad. No linealidad.

Tabla 3. Diferenciación de las líneas de evolución en función del tipo de elemento afectado

Con estos dos “filtrados” se han podido reducir el número de líneas a evaluar en un tercio de las iniciales aproximadamente. La salida del KBS en este caso podría ser únicamente en forma de sugerencias específicas para el diseñador sobre qué aspectos revisar en su diseño.

<b>Línea de evolución</b>	<b>(1, 2, 1)</b>	<b>(2, 2, 1)</b>	<b>Relevante</b>
Segmentación del objeto	✓	✓	Si
Macro a nano escala.	X	✓	Si
Segmentación espacial.	✓	✓	Si
Segmentación superficial.	X	✓	Si
Coordinación del ritmo.	X	X	No
...			

Tabla 4. Selección de las líneas de evolución

## 4 Conclusiones

En el análisis de las herramientas TRIZ se ha podido observar que para llevar a cabo su automatización con el diseño funcional a través de un KBS se requieren de su vinculación con otras herramientas auxiliares, como deben ser una herramienta semántica, una de evaluación, y las bases de datos necesarias. Cuanto más elaboradas estén estas últimas, mejores resultados se podrán obtener de las herramientas de TRIZ. Dichas herramientas se ha visto que pueden actuar antes, durante o después de la fase de diseño inicial.

El principio de idealidad se presenta como una herramienta a utilizar antes de la fase de diseño. Por tanto, es lógico prever que no puede valer para su automatización conjunta con el diseño funcional, pues actúa sobre el planteamiento del problema, y no sobre la resolución de mismo. En este caso solo sirve para proporcionar la información al diseñador sobre cómo plantear correctamente el enunciado del problema de diseño al sistema.

Aquellas herramientas cuya aplicación se puede realizar durante la misma fase de diseño parecen las más idóneas para su automatización. En este caso hemos visto las bases de datos de funciones y los Su-Fields. Ambas herramientas se pueden asimilar a las bases de datos con las que funcionan los KBS. Aunque las bases de datos de funciones ya sea en si una base de datos, las diversas fuentes consultadas no proporcionan una taxonomía común, por lo que falta un mejor análisis para determinar cuál de ellas es la óptima. También presenta el problema del filtrado de las soluciones, por lo que se requerirá de una herramienta de evaluación creada para tal efecto y con la misma taxonomía. Por otro lado, la herramienta de los Su-Fields presenta como ventaja de una taxonomía de campos única, perfectamente vinculable y asimilable al nivel de los comportamientos. En este caso, la misma información de que campos se asimilan a cada comportamiento puede estar implícito dentro de la misma base de datos de los comportamientos, como un atributo más de los mismos. En lo referente a sustancias se corresponden con las estructuras.

Por último, en las herramientas de aplicación posterior a la fase de diseño hemos visto la matriz de contradicciones y las líneas de evolución. La primera de ellas hemos visto que actúa sobre los atributos de las estructuras. En este caso se ve factible su vinculación con el KBS a través de dichos atributos, mientras que deberá de ser una herramienta de evaluación la que indique sobre qué estructuras hay que realizar dicho análisis de atributos en busca de contradicciones. En cuanto a las líneas de evolución, el estudio no la muestra como muy favorable para su automatización. Requiere de mucha información inicial que puede que un sistema genérico no posea, debe realizar muchos análisis y cálculos, pues hemos visto que para cada estructura de la solución se deben de analizar aproximadamente 10-12 estados evolutivos, y la información resultante del análisis, al igual que en el caso del principio de idealidad, proporciona información al diseñador de cómo replantear el problema. Una alternativa para este caso que se está estudiando es la de la creación de árboles de evolución específicos para cada empresa, que en este caso actuarían antes de la fase de diseño.

## **Bibliografía**

- Altshuller G., "*Creativity as an exact science: The theory of the solution of inventive problems*", Gordon and Breach Science Publishers, Luxembourg, 1984.
- Altshuller G., "*Introducción a la innovación sistemática: TRIZ*", Internet Global S.L., 1997a.
- Altshuller G., "*The innovation algorithm. TRIZ, systematic innovation and technical creativity*", Technical Innovation Center, Inc, Worcester, 1999.
- Altshuller G. and Shulyak L., "*40 principles: TRIZ keys to technical innovation*", Technical Innovation Center, Worcester, MA, 1997b.
- Belski I., "*Improve your thinking: Substance field analysis*", Melbourne, Australia, 2007.
- Bergmann R. and Wilke W., "On the role of abstraction in case-based reasoning", *Advances in case-based reasoning*, Springer Berlin / Heidelberg, 1996, pp 28-43.
- Cao G. and Tan R., "FBES model for product conceptual design", *International journal of Product Development*, Vol. 4, 2007, pp 22-36.
- Cascini G., "State-of-the-art and trends of computer-aided innovation tools", en *IFIP 18th World Computer Congress*, Toulouse, France, 2004a, pp 461-470.
- Cascini G., Agili A. and Zini M., "Building a patents small-world network as a means for computer-aided innovation", en *1ST IFIP TC5 Working Conference on CAI*, Ulm, Germany, 2005.
- Cascini G. and Rissone P., "Plastics design: Integrating TRIZ creativity and semantic knowledge portals", *Journal of Engineering Design*, Vol. 15, 2004b, pp 405-424.

Cavallucci D. and León N., "Towards "Inventiveness-oriented" CAI tools", en *IFIP 18th World Computer Congress*, Toulouse, France, 2004, pp 441-452.

Cavallucci D. and Weill R.D., "Integrating Altshuller's development laws for technical systems into the design process", *Annals of the CIRP*, Vol 50, 2001, pp 115-120.

Chang H.T. and Chen J.L., "The conflict-problem-solving cad software integrating TRIZ into eco- innovation", *Advances in Engineering Software*, Vol. 35, 2004, pp 553-566.

Chang H.T., Chen J.L. and Tainan T., "Eco-innovative examples for 40 TRIZ inventive principles", *TRIZ journal*, 2003.

Chulvi V. and Vidal R., "Vinculación de TRIZ con el diseño funcional", en *XI Congreso Internacional de Ingeniería de Proyectos (AEIPRO)*, Lugo, Spain, 2007, pp 613-621.

Chulvi V., Vidal R. and Cebrian-Tarrasón D., "B-cube", en *XII Congreso Internacional de Ingeniería de Proyectos (AEIPRO)*, Zaragoza, Spain, 2008.

Dewulf S., "Directed variation: Variation of properties for new or improved function product DNA, a base for 'connect and develop'", en *ETRIA TRIZ Futures*, Kortrijk, Belgium, 2006.

Gero J.S., "Design prototypes: A knowledge representation schema for design", *AI magazine*, Vol. 11, 1990, pp 26 - 36.

Hirtz J., Stone R., McAdams D., Szykman S. and Wood K., "A functional basis for engineering design: Reconciling and evolving previous efforts", *Research in Engineering Design*, Vol. 13, 2002, pp 65 - 82.

Ikovenko S., "TRIZ and computer aided inventing", en *IFIP 18th World Computer Congress*, Toulouse, France, 2004, pp 475 - 485.

Lind M., "Making sense of the abstraction hierarchy", *CSAPC'99*, Villeneuve d'Ascq, France, 1999.

Mulet E. and Vidal R., "Integración de principios inventivos de TRIZ en sistemas de diseño conceptual asistido por ordenador (CACD)", en *AEIPRO - Congreso Internacional de Ingeniería de Proyectos*, Málaga, España, 2005.

Pinyayev A., "Functional clues", en *ETRIA TRIZ Futures*, Kortrijk, Belgium, 2006.

Soo V.-W., Lin S.-Y., Yang S.-Y., Lin S.-N. and Shian-LuenCheng, "A cooperative multi-agent platform for invention based on patent document analysis and ontology", *Expert Systems with Applications*, Vol. 31, 2006, pp 766-775.

Verbitsky M. (), "Semantic TRIZ", *TRIZ journal - Español*, 2004.

Zhang M., Kirkovsky A. and Pesetsky S., "Engineering creativity and knowledge management powered by rca, ontology and triz", en *1st IFIP TC-5 Working Conference on CAI*, Ulm, Germany, 2005.

## **Agradecimientos**

Los autores muestran su gratitud al Ministerio de Educación y Ciencia por su financiamiento bajo el proyecto con referencia DPI2006-15570-C02-00 dentro del plan nacional de I+D+i (2004 – 2007) y a los fondos FEDER de la Unión Europea.

## **Correspondencia**

Vicente Chulvi Ramos  
Grupo de Ingeniería del Diseño (GID)  
Departamento de Ingeniería Mecánica y Construcción. Universitat Jaume I.  
Avenida Sos Baynat s/n. 12071 - Castellón (Spain)  
Teléfono: +34 964 72 9252 Fax: +34 964 72 8106  
E-mail: chulvi@emc.uji.es