



HAL
open science

Formalisation des connaissances capitalisées en conception inventive.

François Rousselot, Alexis Bultey, Cécilia Zanni

► **To cite this version:**

François Rousselot, Alexis Bultey, Cécilia Zanni. Formalisation des connaissances capitalisées en conception inventive.. Actes des 19es Journées Francophones d'Ingénierie des Connaissances (IC 2008) 19es Journées Francophones d'Ingénierie des Connaissances (IC 2008), Jun 2008, Nancy France, France. pp.173-182. hal-00541093

HAL Id: hal-00541093

<https://hal.science/hal-00541093>

Submitted on 29 Nov 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Formalisation des connaissances capitalisées en conception inventive

François Rousselot¹, Alexis Bultey¹, et Cécilia Zanni¹

¹Laboratoire LGECO, INSA de Strasbourg,
{francois.rousselot, alexis.bultey, cecilia.zanni}
@insa-strasbourg.fr

Résumé : Cet article présente une analyse des bases de connaissances de TRIZ¹ les plus utilisées dans la phase de résolution. Ces bases de connaissances ont été conçues à partir d'une étude ontologique des sources de connaissances informelles utilisées, à la base, par les praticiens de la théorie. Cette approche a permis de concevoir une architecture logicielle qui facilite la mise en œuvre de ces bases par leur manipulation déclarative.

Mots-clés : Ingénierie des connaissances, Ontologies, Conception inventive, Systèmes à base de connaissances, capitalisation de connaissances.

1 Introduction

Après avoir remarqué de nombreux aspects intéressants et des concepts proches de ceux utilisés en Intelligence Artificielle dans TRIZ (Altshuller, 1988, 1999), nous avons décidé de modéliser formellement cette approche méthodologique. La raison est que la TRIZ peut être déclinée dans un ensemble d'outils et de méthodes basés sur des concepts qui ne sont pas formellement définis.

Nous travaillons depuis plusieurs années dans ce sens et notre but est d'éclaircir ses points peu clairs pour avoir une meilleure compréhension de son utilisation, des connaissances et des raisonnements mis en œuvre.

L'originalité d'une méthode de conception inspirée de TRIZ tient au fait qu'elle voit la conception comme de la résolution de problème, elle fournit des outils pour aider à formuler correctement le problème et à construire un modèle de celui-ci, centré sur les principales contradictions à résoudre. La résolution peut ensuite commencer et suit un processus algorithmique s'appuyant sur diverses bases de connaissances. Nos travaux ont porté sur la modélisation du processus de formulation du problème et du processus de résolution (Zanni & Rousselot, 2006).

L'originalité d'une telle méthode par rapport aux autres méthodes de conception tient au fait qu'elle refuse le compromis au cœur par exemple des méthodes basées sur l'optimisation et également au fait qu'elle incite à briser l'inertie psychologique qui conduit souvent les experts à ne chercher la solution que dans leur domaine et dans ce qu'ils connaissent bien. Cette approche s'oppose encore aux méthodes où l'espace de recherche est très grand comme le brain-storming, car elle utilise des principes qui contraignent la recherche.

¹ Acronyme russe pour théorie de résolution des problèmes inventifs

Nous allons, ici, nous intéresser à l'exploitation des bases de connaissances (BDC) utilisées dans la phase de résolution. Nous avons conçu une architecture logicielle pour les exploiter plus facilement et permettre de les étendre en rendant possible la capitalisation de nouvelles connaissances². Ces BDC sont le résultat de la compilation manuelle de grosses masses d'informations extraites de plusieurs milliers de brevets.

Nous aborderons en premier lieu les notions principales de TRIZ, puis nous présenterons les différents modèles à partir desquels on peut lancer la résolution. Nous présenterons un de ces modèles appelé 'analyse Vépole', et nous présenterons l'ontologie qui formalise ce modèle. Finalement, nous présenterons notre approche qui repose sur une architecture logicielle à base de Logique de Description (LD) qui permet de manipuler de façon déclarative les connaissances et qui exploite les mécanismes d'abstraction des LD. Le système gère les « effets physiques » souplement et dans l'avenir facilitera l'intégration de nouvelles connaissances.

2 Les notions principales de TRIZ

Nous avons défini dans différents travaux une ontologie couvrant la plupart des notions de TRIZ et notamment celles qui permettent de formaliser les différents modèles utilisés en conception inventive (Rousselot et al., 2007). Nous en donnons ici les grandes lignes et renvoyons pour de détails à (Zanni & Rousselot, 2007).

La méthodologie TRIZ est souvent présentée comme un processus montant puis descendant. Elle part d'abord de la situation initiale et en produit un modèle abstrait par rapport au niveau du domaine (processus montant), la résolution consiste ensuite à associer au modèle abstrait une solution générique puisée dans une base de connaissances préconstruite. Il convient finalement d'évaluer la faisabilité et la qualité de cette solution dans le domaine spécifique de l'application en cours (processus descendant).

Après avoir observé le processus d'acquisition-reformulation, nous avons un point de vue différent: nous considérons qu'elle propose un déroulement principalement **descendant** de construction du modèle de l'artefact initié à partir de la formulation initiale du problème. L'étape de formulation et de reformulation du problème est primordiale, elle conduit à préciser le problème et à construire un modèle de celui-ci. Le succès de la résolution qui intervient ensuite est bien évidemment fonction de la qualité du modèle et de sa finesse. En réalité, on peut parler de la construction de différents modèles de niveaux de plus en plus bas développés selon le contexte³.

Précisons que ces modélisations sont toutes élaborées sur des concepts indépendants du domaine de l'application. Le problème initial étant souvent mal posé, TRIZ incite le concepteur à le structurer d'abord en sous-ensembles (voir l'ontologie dans Rousselot et al., 2007) suivant le **modèle systémique**. Cette modélisation conditionne l'organisation du système en sous-parties (**contrôle, outil, produit**, etc.) et isole sa **fonction principale utile**. Puis, un second modèle fait ressortir la

²Il faut savoir que depuis la mort de G. Altshuller (le créateur de TRIZ), les bases de connaissances n'évoluent plus, car il n'est plus là pour valider les changements.

³L'expert TRIZ a le choix de déclencher la résolution à tous les niveaux quitte à continuer à développer ensuite des modèles plus précis s'il en a envie.

contradiction clé à l'origine du problème. Dans TRIZ, il y a deux types importants de contradictions et pour contourner celles-ci deux bases de connaissances différentes.

a) **La contradiction technique** : l'évolution d'un paramètre A dans le sens d'une amélioration conduit à la dégradation de la valeur d'un paramètre utile B (et réciproquement).

b) **La contradiction physique** : le paramètre d'un système doit présenter deux valeurs contradictoires à la fois : forte et faible, dur et mou ...

Le concepteur a le choix d'arrêter son analyse au niveau de la contradiction technique, il a alors à sa disposition la « Matrice de résolution » et ses 40 principes. Ceux-ci lui proposent des pistes de solutions le plus souvent énoncées sous forme de procédés généraux qui résolvent la même contradiction ou une contradiction analogue. Les praticiens de TRIZ préfèrent généralement approfondir leur recherche en utilisant deux autres modèles : celui de la contradiction physique et celui de l'analyse **Vépole** encore appelée analyse Substance-Champ. La résolution de la contradiction physique est supportée par une base de connaissances, appelée "méthodes de séparation". Dans le cadre de cet article, nous nous intéressons exclusivement au modèle Vépole et aux bases de connaissances associées: la modélisation Vépole, les 76 Standards Inventifs et le « pointeur des effets scientifiques ».

Le modèle Vépole orientera le concepteur vers la manipulation successive des Standards et du pointeur. Les 76 Standards Inventifs sont des connaissances de raisonnement qui permettent de passer d'un modèle de problème, appelé **Vépole Problème**, vers un modèle de solution, appelé **Vépole Solution**. Ensuite intervient le pointeur des effets qui concourt à l'interprétation du Vépole Solution et donne des pistes vers des réalisations physiques du modèle de solution.

3 L'étude de l'analyse Vépole

3.1 Introduction

La modélisation Vépole repose sur une hypothèse simple. Les problèmes d'un système peuvent être ramenés à un nombre réduit de parties élémentaires en interaction. Ces parties élémentaires sont appelées substances. Un champ est à l'origine de l'interaction entre deux substances. Une des substances, l'**outil**, agit sur l'autre, le **produit** ; le **champ** est un scalaire de force qui permet l'interaction (de l'outil sur le produit). Une fois, le problème modélisé, il s'apparente à un Vépole problème.

Les 76 Standards inventifs sont exprimés sous forme de règles. Ils permettent de passer d'un Vépole problème à un Vépole solution. Si la description du Vépole problème vérifie les prémisses d'un Standard, alors le Standard peut être appliqué. Un Standard est souvent formulé de la sorte :

Si <Condition 1> et <Condition 2> alors <Transformation>

La condition 1 est souvent la description d'un type de Vépole problématique et la condition 2 est un ensemble de propositions restrictives dédiées à situer le problème,

comme le suggère l'exemple du Standard 1.2.2 - « *S'il y a un effet utile et un effet néfaste entre deux substances (condition 1) et qu'il n'est pas requis que ces deux substances soient en contact, et s'il n'est pas interdit ou malvenu d'ajouter une substance étrangère condition 2), le problème peut être résolu avec l'introduction d'une modification d'une ou des deux substances (transformation).* »

Les Standards sont divisés en 5 classes qui correspondent à autant de types de problèmes. Les 3 premières classes prennent en compte les tendances d'évolutions des artefacts techniques, la 4^e est dédiée aux systèmes de mesure et de détection. La dernière classe est une sorte de métaclasse qui guide l'application des 4 premières.

Le pointeur des effets physiques constitue une source de connaissances importante, car il comporte 4 000 phénomènes de domaines variés liés à la physique, à l'optique et à la chimie classifiés suivant leurs intérêts pratiques, par exemple : produire un aérosol, produire une substance gazeuse, changer des propriétés mécaniques, joindre des corps, créer un champ magnétique, etc. Cette source de connaissances n'est pas liée à un domaine technique ou industriel particulier.

3.2 Le choix de représentation : l'ontologie

Nous avons donc construit l'ontologie de l'analyse Vépole, ceci a induit le choix d'un nouveau mode de représentation des Standards et des pointeurs aux effets physiques. L'architecture logicielle résultante rend leur emploi plus flexible, favorise leur extension, soit par l'adjonction de nouveaux effets soit par la possibilité de prise en compte de nouveaux domaines d'application. La méthodologie de construction de l'ontologie se base sur les principes différentiels énoncés dans (Bachimont, 2000) et sur les critères de classification des propriétés proposées par (Guarino & Welty, 2000).

Nos travaux de conceptualisation de l'analyse Substance-Champ s'appuient sur un corpus de textes de référence (Altshuller, 1988, 1999) (Savransky, 2000) (Salamatov, 2000) et sur des discussions avec les experts TRIZ. Ils ont donné naissance à l'ontologie présentée ici brièvement, pour plus de détails, voir (Bultey et al., 2006). La cohérence logique de l'ontologie est assurée grâce au système à base de Logiques de Description, CICLOP (de Bertrand de Beuvron et al., 2000).

Il a fallu ajouter une surcouche à CICLOP pour organiser les étapes de construction du modèle et de la résolution, car une logique de description manque de possibilités procédurales. Afin de préserver le caractère déclaratif de l'ensemble de l'implémentation, nous avons choisi un système à base de règles JSNARK⁴ qui communique avec CICLOP et permet d'exécuter des actions spécifiques. Ce moteur permet d'inspecter les concepts de la base de connaissances, de poser des questions à l'utilisateur et de lancer des classifications à tout moment.

Notre BDC se décompose en deux bases de connaissances terminologiques (BCT) et les bases d'assertions (ou faits) associées) contenues dans CICLOP et une base de connaissances inférentielles contenue dans JSNARK. Les BCT décrivent respectivement les connaissances concernant les Vépoles et leurs transformations

⁴ Ce moteur JSNARK est une implémentation en Java d'un système à base de règles proche de SNARK (Laurière & Vialatte, 1986). C'est un moteur d'inférence d'ordre 1 qui fonctionne en marche avant par saturation, il n'utilise que des prédicats binaires donc proches des rôles utilisés en LD.

suggérées dans les Standards et le pointeur des effets physiques. Elles constituent respectivement l'Ontologie des Vépoles et l'Ontologie des Transformations, évidemment liées. Nous les présentons séparément pour des raisons de clarté. La base inférentielle se compose des connaissances sur les raisonnements exprimés par les Standards, des règles d'application des Standards, et des « règles projet ».

La base terminologique des Vépoles définit les concepts « actifs » de l'analyse Vépole : Substance, Champ, Interaction, etc. Les concepts fils sont des Vépoles plus spécifiques rencontrés dans les prémisses et les conclusions des Standards, ils caractérisent les différents types de modèles de problème et de modèles de solution. Nous donnons ici la description des niveaux hauts de l'ontologie des Vépoles en LD⁵.

$$\begin{aligned} \text{Vepole} &= \text{TrizModel} \cap \exists a\text{Element.Element} \\ \text{Vepole_Incomplet_S} &= \text{Vepole} \cap \exists a\text{Substance_S1.Substance} \\ &\quad \cap \exists a\text{Substance_S2.Bottom} \\ \text{Vepole_Complet} &= \text{Vepole} \cap \exists a\text{Substance_S1.Substance} \\ &\quad \cap \exists a\text{Substance_S2.Substance} \cap \exists a\text{Champ_C1.Champ} \end{aligned}$$

Un Vépole est défini comme un sous-concept d'un modèle TRIZ ayant au moins un rôle aElement dont le codomaine est le concept atomique Element. Element décrit les deux composants substance et champ d'un Vépole. Les concepts fils d'Element sont donc Substance et Champ et sont définis comme disjoints. Les concepts fils de Champ caractérisent les différents champs dits « technologiques » que nous rencontrons dans la TRIZ⁶. Au concept de Champ est associé celui d'Interaction par l'intermédiaire du rôle aInteraction.

Enfin, dans TRIZ, la substance pouvant se définir comme un Vépole, elle est donc subsumée par Vepole et Element⁷. Par exemple, un aimant peut être représenté comme un Vépole constitué d'un matériau ferromagnétique et d'un champ magnétique. La décomposition la plus fine d'une substance est un Vépole appelé Matière, qui possède une substance Corps_Pur et un champ Champ_De_Liaison.

L'ontologie est ainsi construite afin de coïncider le plus possible avec l'analyse Vépole, mais également afin de s'adapter aux caractéristiques de l'ontologie des Transformations présentée plus loin.

$$\begin{aligned} \text{Aimant} &= \text{Substance} \cap \exists a\text{Substance_S1.Materiau_Mineral_} \\ &\quad \text{Ferromagnetique} \cap \exists a\text{Champ_C1.Champ_Magnetique} \\ \text{Matiere} &= \text{Substance} \cap \exists a\text{Substance_S1.Corps_Pur} \\ &\quad \cap \exists a\text{Champ_C1.Champ_de_Liaison} \end{aligned}$$

⁵ Nous utilisons ici la syntaxe la plus communément utilisée en Logique de Description (Nardi 2002).

⁶En TRIZ, la notion de champ est plus large qu'en physique. En plus des quatre formes de champs que sont les champs électromagnétiques, gravitationnels ceux de faibles et de fortes interactions, TRIZ considère par exemple des champs dits « de pression » ou « de force centrifuge ».

⁷ Il n'y a pas de cycle dans les définitions Vépole n'étant pas défini comme un concept fils de Substance, le modèle Vépole s'utilise ici pour modéliser la composition d'une substance.

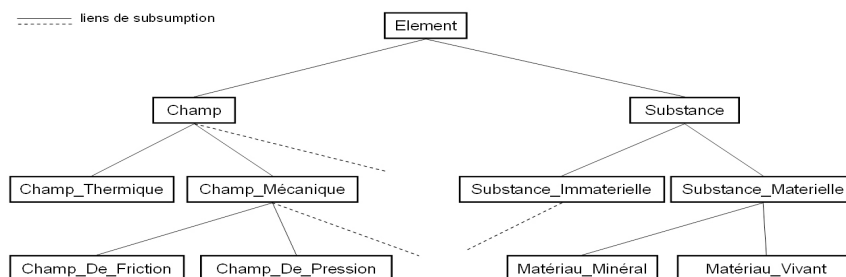


Fig. 1 – Extrait de l'ontologie des éléments d'un Vépole

La base terminologique des Transformations définit chaque transformation comme un couple de Vépoles, l'un correspondant à la description de l'état initial, l'autre de l'état final. Les transformations vues dans les Standards sont classifiées dans cette ontologie ainsi que celles citées dans les 200 effets physiques préconisés par Altshuller.

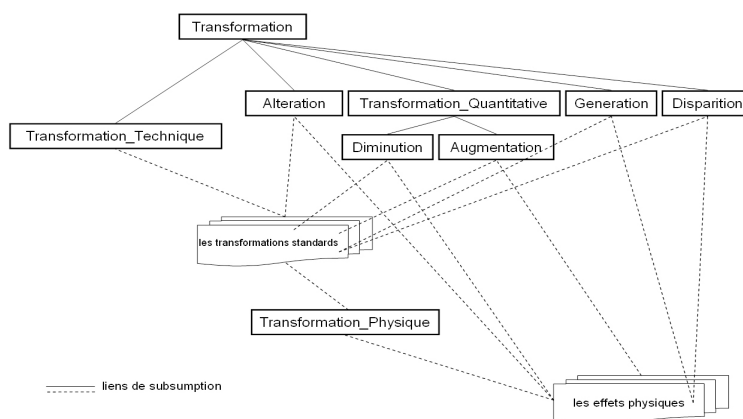


Fig. 2 – Extrait de l'ontologie des transformations

Cette classification opérée par la LD met en rapport des effets physiques (transformations spécifiées) avec les Standards (transformations abstraites). Comme le montre la **Fig. 2**, un effet physique peut réaliser plusieurs transformations.

Pour classifier un Standard, il suffit d'entrer sa définition LD. Par exemple on décrira le Standard 1.2.2, vu précédemment de la manière suivante :

```
Standard_1.2.2 = Transformation_Technique
                ∩ ∃initial.Vepole_Complet_Néfaste
                ∩ ∃final.Vepole_Chaine
```

C'est une Transformation_Technique ayant pour état initial un Vépole Complet possédant une interaction néfaste et pour état final un Vépole en chaîne⁸ possédant une nouvelle Substance S3 qui est une modification de S1 ou S2. Les concepts spécifiques Vépole_Complet_Néfaste et Vépole_Chaine représentent des Vépoles particuliers appartenant

⁸ Un Vépole en chaîne est un Vépole qui a plus de deux substances.

aux prémisses et aux conclusions des Standards, ils proviennent donc de la base des Vépôles. Les rôles « initial » et « final » sont des rôles liant les ontologies des Vépôles et des Transformations. Ils permettent de tirer profit des différents concepts de Vépôle spécifiés pour classifier plus finement les différents Standards, mais aussi les effets physiques. Voici, par exemple, la définition en logique de description de l'effet de cavitation acoustique :

```
Cavitation_Acoustique = Transformation_Physique
    ∩ ∃initial.Liquide
    ∩ ∃final.Vepole
    ∩ ∃substance.Melange_Liquide_Gaz
    ∩ ∃champ.Champ_Acoustique)
```

Qu'on peut interpréter comme : 'la cavitation acoustique transforme une substance liquide en un mélange liquide gazeux par l'intermédiaire d'un champ acoustique'.

La base inférentielle contenue dans JSNARK est organisée en 3 niveaux hiérarchiques : les « règles projet », les règles d'application des Standards, et les règles standards. Les règles standards aident à spécifier le Vépôle problème et le Vépôle solution de telle façon qu'une seule transformation standard soit choisie. En effet, à la fin de la modélisation du Vépôle Problème, celui-ci est souvent associé à plusieurs transformations standards, et les règles standards, par l'intermédiaire d'un jeu de question aident à faire un choix. Elles expriment directement une partie des prémisses des Standards (Condition 2). Soit, par exemple⁹ :

```
IF (Vépôle_Complet_Néfaste(X)) THEN askfor ( contact, "S1 et S2 doivent-elles être en contact ? ")
```

Si la réponse est oui le fait contact (oui) sera entré dans la base des faits du système.

Les règles d'application des Standards ont pour rôle de remplir les différents champs vides une fois la transformation standard choisie. Elles spécifient, par exemple, la transformation standard sélectionnée en définissant les substances et les champs qui composent le Vépôle solution. Mais aussi si l'ajout d'une substance S3 modifiée est nécessaire, elles cherchent un effet physique le permettant et l'ajoutent au Vépôle solution :

```
IF (substance modifiée(X)) then add(substance_S3(Y,X))
```

Les « règles projet » supervisent le processus de résolution et articulent les différentes phases telles la construction du modèle Vépôle problème, le lancement de la résolution, et la recherche d'effets. Soit par exemple la première règle lancée, à l'ouverture d'un nouveau projet qui permet de définir la valeur de l'attribut Nom du Vépôle problème :

```
IF (Nouveau projet(X)) THEN askfor (vépôle_problème, nom, "entrer le nom du nouveau Vépôle problème")
```

4 Exploitation de l'ontologie sur un exemple

Pour comprendre le fonctionnement général de l'architecture de résolution de problème, nous présentons l'utilisation de l'ontologie de l'analyse Vépôle au travers de la résolution d'un problème concret. Il s'agit du problème de pompage qui s'exprime comme suit : « Le pompage d'un liquide au travers d'un tuyau est souvent

⁹ Le langage utilisé est celui de JSNARK.

source de problème. Quand une vanne ou une soupape interrompt le flux du liquide, cela résulte souvent en ce qu'on appelle des coups de bélier (changements de pressions par à-coups) qui peuvent détruire le tuyau ».

Ce problème est se modélise en un Vépole de la façon suivante :

- S2 est le tuyau (la substance outil)
- S1 est le liquide (la substance produit)
- C1 est un champ mécanique qui s'exerce entre S2 et S1 à l'origine d'une interaction utile (contenir l'eau).
- C2 est une pression exercée par S1 sur S2 à l'origine d'une interaction néfaste (coup de bélier).

Ce Vépole problème est entré dans le système, traduit en LD suivant les concepts de l'ontologie Vépole, et incorporé à la base assertionnelle de CICLOP. En voici une définition simplifiée:

```
Vepole_probleme  ⊃ ∃substance_S1.Liquide
                  ⊃ ∃substance_S2.Solide_a__une_cavite
                  ⊃ ∃champ_C1.Champ_Mécanique...
                  ⊃ ∃champ_C2.Champ_de_Pression...
                  (Vepole_Probleme_Tuyau_1)
```

L'individu Vepole_Probleme_Tuyau_1 est défini comme instance du Vépole Problème spécifié ci-dessus. Sa définition terminologique est classifiée dans CICLOP et peut ainsi être rattachée à un Vépole_Problème décrit dans les prémisses des Standards. Le mécanisme inférentiel des logiques de description permet ce rattachement, et sélectionne une liste réduite de transformations standards ayant pour états initiaux un concept parent du Vépole_Problème_Tuyau.

Des questions (générées automatiquement grâce à l'ontologie) sont alors posées à l'utilisateur afin de restreindre la liste des transformations à une seule transformation standard : une substance peut-elle être ajoutée ?, S1 et S2 doivent elles rester en contact ?, ... Suivant les réponses données JSNARK va orienter la construction du Vépole_Solution_Tuyau. Un individu instance de transformation est alors généré, il comprend en état initial le Vépole_Problème_Tuyau_1 et en état final le Vépole_Solution_Tuyau_1. Cet individu bénéficiera de la même façon que précédemment des raisonnements assertionnels et de rapprochement pour être classifié dans l'ontologie des transformations et être ainsi rattaché à des transformations physiques proches s'il en existe. Dans notre exemple, les transformations physiques trouvées sont la cavitation acoustique, l'absorption et l'évaporation capillaire. Parmi cette liste d'effets, la cavitation acoustique a fait l'objet d'un brevet¹⁰ et constitue la solution proposée dans la littérature TRIZ. Elle se résume à l'ajout d'un émetteur à ultrason au niveau des zones sensibles comme les coudes, des bulles vont alors se former et amortir l'impact.

Nous avons reproduit ce test d'exploitation de l'ontologie sur plusieurs dizaines de problèmes énoncés et résolus dans la littérature TRIZ (Bultey et al., 2007). Nous retrouvons les résultats proposés par la littérature mais aussi d'autres non mentionnés qui sont parfois sujets à des réalisations intéressantes.

¹⁰ Le brevet russe 1078178.

5 Résultats

Dans cet article, nous n'avons donné que des extraits des différentes ontologies construites, car celles-ci sont très importantes : nous avons, en effet intégré 200 effets physiques sur les quelques 4 000 compilés par Altshuller.

L'ontologie des Vépoles, l'ontologie des transformations, l'ontologie des substances et l'ontologie des effets physiques sont un apport considérable à la normalisation nécessaire des connaissances TRIZ et relie de façon formelle les concepts « Effets physiques » et « Standard » qui étaient jusqu'alors employés séparément dans l'analyse Vépole.

Nous avons choisi d'utiliser la logique de description pour gérer ces ontologies. L'intérêt de ce formalisme déclaratif, outre un contrôle de cohérence, est de pouvoir, d'une part, rendre compte du continuum dans les différents niveaux d'abstraction entre les modèles plus ou moins renseignés, d'autre part, de prendre en compte toute modification ou d'ajout de concept en propageant les nouvelles contraintes introduites. Ainsi, l'ontologie permet de partir d'un niveau abstrait pour arriver à un niveau bas qui correspond à une solution possible utilisant un effet physique donné. D'autre part, en plus de la stratégie fiée induite par l'analyse Vépole, différentes stratégies de questionnement sont maintenant envisageables. Nous avons la possibilité, par exemple, de décider de poser au départ des questions concernant des situations particulières de sécurité (par exemple, faut-il éliminer les gaz ?). Le système de représentation des connaissances propageant les faits vrais connus, exclura automatiquement tous les effets physiques correspondants contenus dans la base des effets physiques. La sélection des effets physiques s'effectue maintenant sur des critères sémantiques gérés par la LD et non plus sur grâce à des règles prédéfinies et figées

Le système permet également de consulter la base des effets de différentes manières. Il est possible de partir d'un effet donné pour voir quelles sont les transformations où il peut intervenir. Il est également possible de contraindre de façon plus précise le Vépole Solution. En ajoutant à la définition de celui-ci que nous ne souhaitons pas, par exemple, avoir recours à un champ électrique, que nous voulons utiliser une substance bon marché, etc.

En ce qui concerne l'extensibilité de TRIZ à d'autres domaines, on constate que les niveaux hauts de l'ontologie produite ne présupposent pas que les composants du modèle Substance-champ soient de nature uniquement matérielle, ainsi on peut envisager des résolutions concernant des domaines non purement techniques, par exemple, des problèmes organisationnels, des problèmes communicationnels ou des problèmes de génie logiciel (Braunschweig, 2003).

Pour rajouter une nouvelle connaissance, par exemple un effet physique, il suffit de le décrire dans les termes de l'ontologie et CICLOP le classifie automatiquement, il est alors disponible pour fournir de nouvelles solutions.

Pour capitaliser de nouvelles connaissances de résolution, on va maintenant rajouter à nos BDC des connaissances correspondant à des domaines non techniques, par exemple, des vépoles solutions à des vépoles problèmes de génie logiciel.

Références

- ALTSHULLER G.S. (1988). Creativity as an exact science. *Gordon and Breach*, , New York.
- ALTSHULLER G.S. (1999). TRIZ the innovation algorithm, systematic innovation and technical creativity. *Technical innovation Center Inc.*, Worcester, Massachusset.
- BACHIMONT B. (2000) Engagement sémantique et engagement ontologique : conception et réalisation d'ontologies en ingénierie des connaissances. In J. CHARLET, M. ZACKLAD, G. KASSEL & D. BOURIGAUULT Eds. *Ingénierie des connaissances : évolutions récentes et nouveaux défis*, Eyrolles, p. 305-323.
- BULTEY A., DE BERTRAND DE BEUVRON F. & ROUSSELOT F. (2006). *A problem solving environment based on TRIZ ontologies. Proceedings of Virtual Concept 2006*, Mexico.
- BRAUNSCHWEIG B. (2003). *Vers la simulation numérique par agents apprenants*. Habilitation à diriger des recherches, Université Pierre et Marie Curie, Paris 6, France.
- BULTEY A., DE BERTRAND DE BEUVRON F. & ROUSSELOT F. (2007). *A substance-field ontology to support the TRIZ thinking approach. Int. J. Computer Applications in Technology*, Vol. 30, No 1/2, p 113-124.
- CAVALLUCCI D. & KHOMENKO N. (2006). *FromTRIZ to OTSM-TRIZ : Adressing complexity challenges in inventive design. Int. J. of Product Development*. Vol. 4, No 1-2.
- DE BERTRAND DE BEUVRON F., ROUSSELOT F., KULLMAN M., RUDOLF D. & SCHLICK(2000). *CICLOP. Proceedings of « les journées francophones de l'ingénierie des connaissances »*, Toulouse, France.
- GUARINO N. & WELTY C. (2000). *A formal ontology of properties*. In R. DIENG & O. CORBY Eds. *Knowledge Engineering and Knowledge Management : Methods, Models and Tools. Proceedings of International Conference EKAW'2000*, Springer-Verlag, p97-112.
- LAURIÈRE J.L. & VIALLATTE M. (1986). *SNARK : a language to represent declarative knowledge and inference engine which use heuristics. Proceedings of IFIP Congres*. p. 811-816, Dublin, Ireland.
- NARDI D., BRACHMAN R. J. (2002) An Introduction to Description Logics. In the *Description Logic Handbook*, edited by F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider, Cambridge University Press, pages 5-44.
- ROUSSELOT F. , ZANNI C. & CAVALLUCCI D. (2007). *Une ontologie pour l'acquisition et exploitation des connaissances pour la conception inventive. Revue des Nouvelles technologies de l'Information, numéro spécial sur la modélisation des Connaissances*. A paraître en 2008.
- SALAMATOV Y. (2000). TRIZ : the right solution at the right time. Insystec Ed. *Translated by Strogaiia M. & Yakovlev S*.
- SAVRANSKY S.D. (2000). *Engineering of Creativity : Introduction to TRIZ methodology of Inventive Problem Solving*. CRC Ed, Boca Raton.
- ZANNI C. & ROUSSELOT F. (2006). *Towards the Formalization of Innovating Design: The TRIZ example. Proceedings of KES2006- 10th International Conference on Knowledge-Based & Intelligent Information & Engineering Systems*, Bournemouth, United Kingdom.
- ZANNI C. & ROUSSELOT F. (2007). "New Directions in the Formalization of Inventive Design", *Proceedings of IMSM07- International Modeling and Simulation Multiconference 2007, Buenos Aires, Argentina, ISBN 978-2-9520712-6-0*.