Mereological Knowledge Representation for the Chemical Formulation

Stefania Bandini, Alessandro Mosca

University of Milano-Bicocca
Department of Computer Science, Systems and Communication

Artificial Intelligence Lab

Overview

[1] Motivations

- The Chemical Formulation Problem
- Expert Knowledge in Chemical Formulation

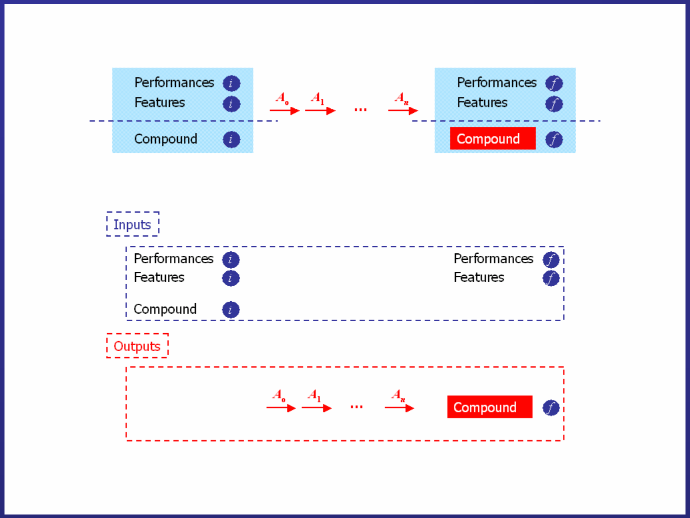
[2] The Knowledge Model

- A Description Logic Axiomatization
- One Example of... *Mereology in Action!*

[3] Concluding Remarks and Future Works

Problem Definition

Chemical Formulation Problem as *Product Innovation*

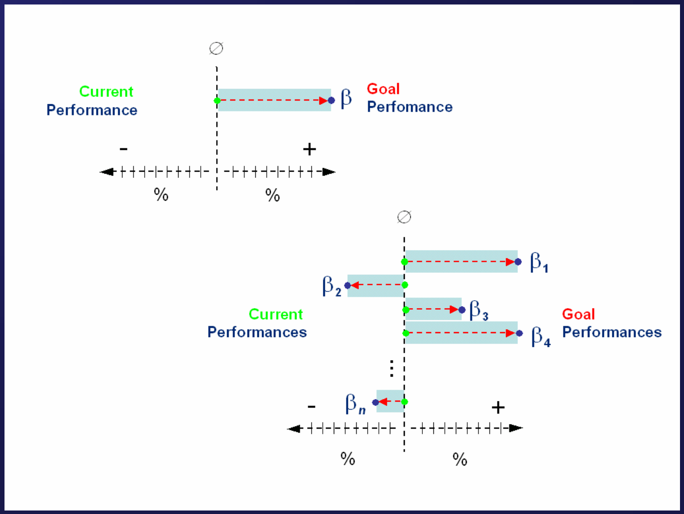


The Chemical Formulation Problem, whether for agrochemicals, pharmaceutical, or speciality chemicals industrial areas, deals with the possibility of *modifying the formulation* of an *existing* chemical compound, in order to gain new compound formulations showing *final desired behaviors* (e.g. *features* and *performances*).

Given the formulation of a preexistent compound, together with the evaluation of its behaviors, and the expression of new desired behaviors, the problem concerns the discovery of a set of "applicable" modifications that lead to a new compound formulation meeting the desired behaviors.

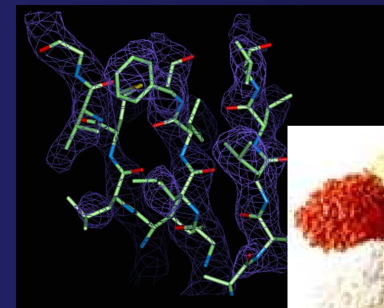
New desired behaviors for a chemical compound can be originated by specific *marketing commitments*, authorities *standards*, *design and cost requirements*, and by several constraints coming from *specific phases of the production process*.

Given the formulation of a preexistent compound together with the evaluation of its physical and chemical behaviors, new desired behaviors are usually expressed as percentage variations of the starting ones.



Searching in a Problem Space

Problem-dependent motivations



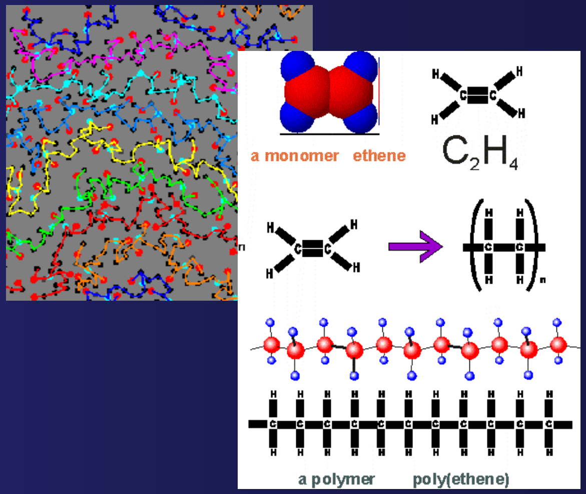
- The hardness of the problem consists in <u>*controlling the effects*</u> of the application of transformations to compound formulations in terms of compound behaviors (i.e. features and final performances) transformations.

- The solution searching process in compounding essentially is a <u>*combinatorial process*</u>, i.e. a solution is a suitable combination of transformations of the starting compound formulation that leads to a new compound with the desired final behaviors.

- Since a *holistic* perspective on the effects of compound structural transformations is natural and necessary (i.e. the application of a structural transformation to a compound translates to a modification of *all* the associated compound behaviors), searching for a solution of a compounding problem means always to search for a <u>*compromise*</u>: the *optimum* does not exist!

The Need for Ontologies in Compounding

Opening new perspectives



- Although *mathematical (quantitative) methods* are well-known and there are a number of systems based on them, their *computational cost grows largely* with the number of electrons and, therefore, with the chemical and physical complexity of the investigated substances.

- The limitations of the actual Computational Chemistry systems suggest that we must rely on systems that are based on different modeling techniques.

- As far as the problem of designing and implementing systems that drive transformations of complex chemical substances is concerned, it becomes necessary *to overcome the computational intractability* of *quantitative*, *mathematical* compound representations.

- Reason on chemical compounds taking advantage of a formal model representing *not-quantitative expert compounding knowledge*.

- Expert knowledge on chemical formulation is often *not immediately quantifiable*, *not directly math-based*, and *not microscopic*.

“*Knowledge is the Key*”... but, what Kind of Knowledge?

Ontological knowledge

- It is the result of the analysis of *entities* along different dimensions (functional and behavioral): it specifies what *entities* have to be considered as *consistent* compounds’ formulations and compounds’ behaviors in a given domain.

- Ontological knowledge establishes the “integrity conditions” of a compound in a given domain of interest, and it guarantees that the applications of transformations to those entities preserve their ontological status.

Causal knowledge

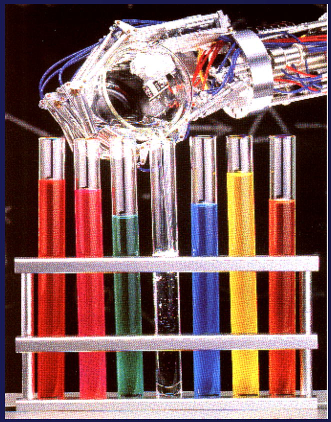
- It is knowledge concerning the mapping between compound transformations at the formulation level to transformations at the behavioral one.

- Causal knowledge in compounding allows to expect the effects of a compound transformation tn term of the transformations occurring to its behaviors.

Causal and ontological compounding knowledge do not have to be extracted in some computational way: it represents a heritage that lies in *expert practices* and communities, and that needs to be elicited, formally represented, and integrated.

The Entry of the Ontological Knowledge - I

Exploiting *ontological* knowledge in controlling the expansion rate of CoPS

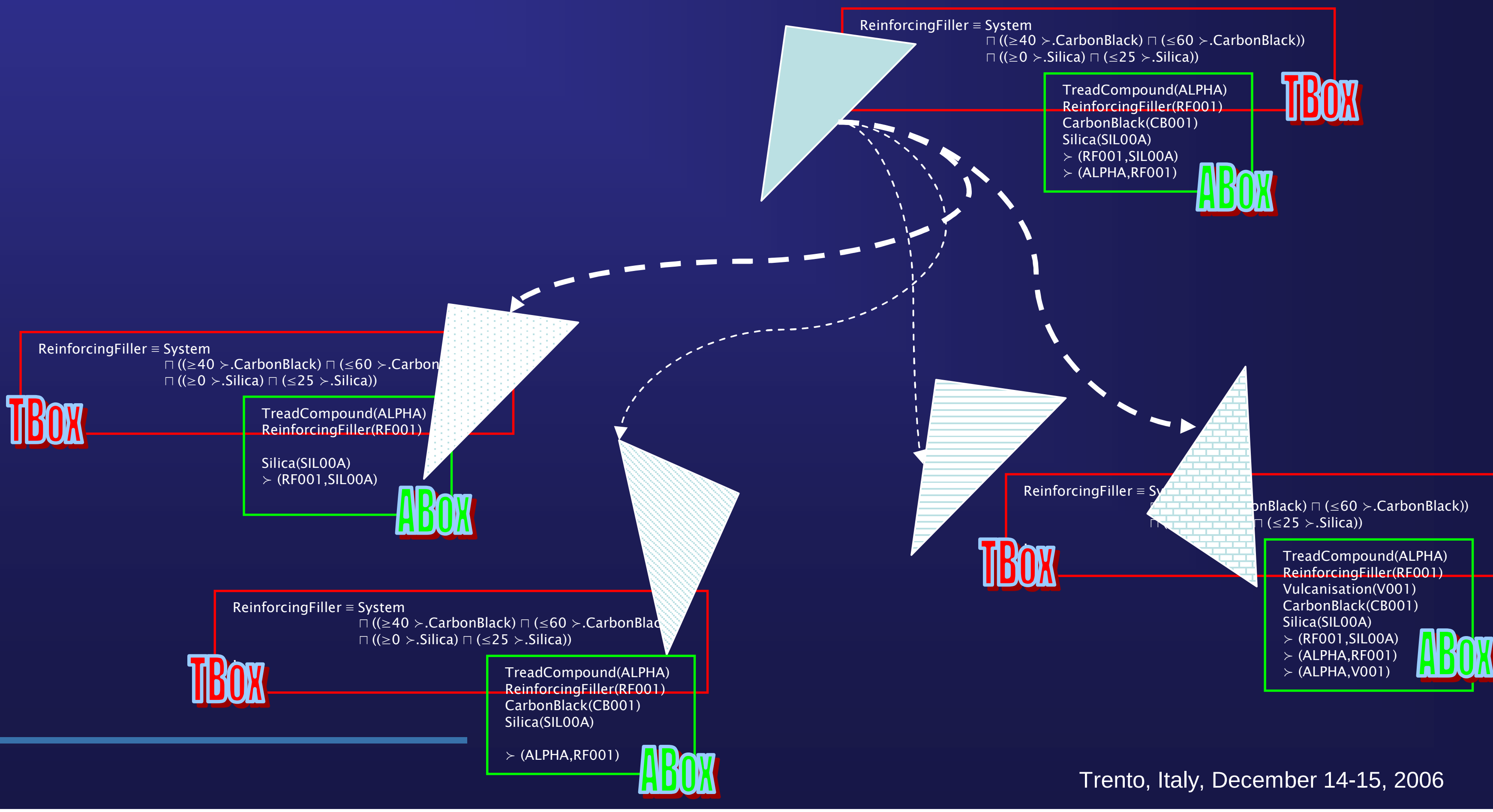


- Reducing the problem space essentially means to cut transitions.
- Given a state of the space in which the problem solving process is occurring, it is possibile to cut the outgoing transitions if:
  - [1] their <u>pre-conditions</u> at that state are not satisfied (i.e. their application to the state is *ontologically inconsistent*), and
  - [2] their <u>post-conditions</u> are *ontologically inconsistent* (i.e. the new generated states are ontologically inconsistent).
- *<u>Ontological knowledge</u>* is the key factor, i.e. the understanding and the formal specification of *<u>what a compound is</u>* in a given domain.
- The proposed solution consists in adding to the definition of the problem space (i.e. to states and transitions) a formal representation of the ontological knowledge w.r.t. the compounding domain of interest.

The Entry of Ontological Knowledge - II

More formally

- Associate a TBox for the compound formulation problem to a ***State Space***, whose axioms define what a compond compound formulation is in a given domain of interest (e.g. the tire production).
- ***States*** collect sets of *concept* and *role assertions* of a respective ABox, whose aim is to specify concrete instances of compound formulations.
- ***Transitions*** represent *insertions* and *deletions* of concept and role assertions of a state.



The Entry of Ontological Knowledge - III

Exploiting the semantic notion of *model*

- An interpretation *satisfies* an ABox A *with respect to* a TBox T if, in addition to being a model of A, it is model of T

> A model of A and T is an *abstraction of a concrete world* where the concepts are interpreted as subsets of the domain as required by the TBox and where the membership of the individuals to concepts and their relationships with one another in terms of roles respect the assertions in the ABox.

[Baader & Nutt, *Basic Description Logics*, 2003]

- **LTS⁺**, an augmented definition of labelled transition system.

Definition (LTS$^+$) *Let A and $\mathcal{T}$ be a set of modification labels and a terminology, respectively. An* augmented *transition system* **LTS$^+$** *over A and $\mathcal{T}$ is a structure*

$$(S, i, A, \rightarrow, \models)$$

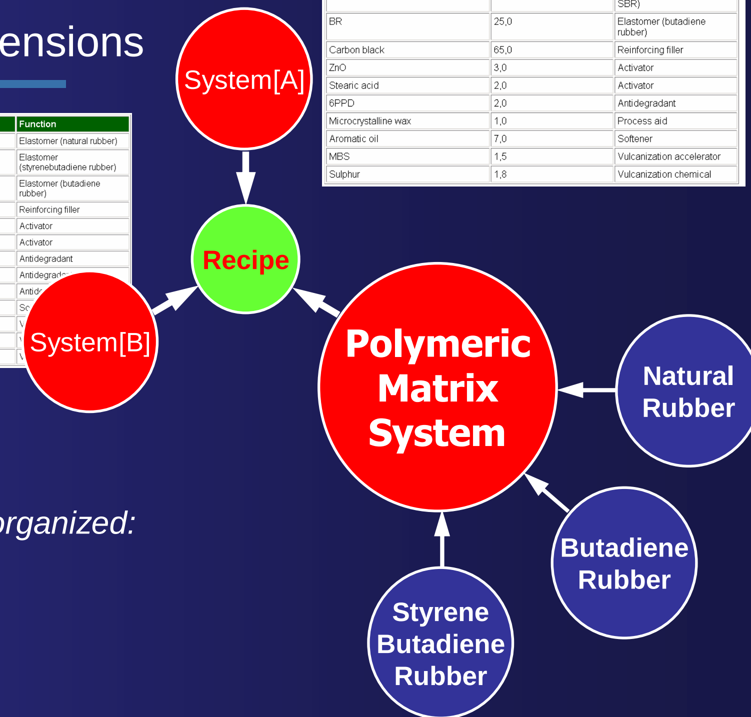
1. *S is a set of* states *with initial state i*
2. *A is a set of* modification labels
3. *$\rightarrow \subseteq S \times A \times S$ is the* transition relation, *and*
4. *$\models \subseteq S \times \mathcal{T}$, such that $\models (s_\Gamma, \mathcal{T})$* holds *iff the state s bears a set of individual assertions Γ that admit a* model with respect to *the terminology $\mathcal{T}$.*

Compound Formulations

Representing *structural* and *behavioral* compound dimensions

Ingredient	Amount (phr)	Function
OE-SBR	103,1	Elastomer (oil extended SBR)
BR	25,0	Elastomer (butadiene rubber)
Carbon black	65,0	Reinforcing filler
ZnO	3,0	Activator
Stearic acid	2,0	Activator
6PPD	2,0	Antidegradant
Microcrystalline wax	1,0	Process aid
Aromatic oil	7,0	Softener
MBS	1,5	Vulcanization accelerator
Sulphur	1,8	Vulcanization chemical

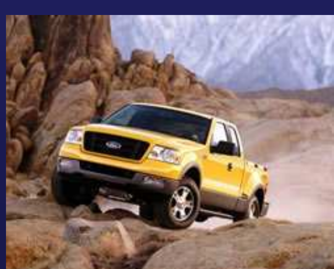
Ingredient	Amount (phr)	Function
SMR 20	25	Elastomer (natural rubber)
SBR 1712 Kralex	62	Elastomer (styrenebutadiene rubber)
BR Nipol 1220	30	Elastomer (butadiene rubber)
N-115 (Carbon black)	65	Reinforcing filler
ZnO	4,0	Activator
Stearic acid	2,0	Activator
6PPD	2,0	Antidegradant
TMQ	1,0	[illegible]
Seresine	1,5	[illegible]
Oil Nysolvex 830	21,0	[illegible]
CBS	1,0	[illegible]
DPG	0,2	[illegible]
Sulphur	1,9	[illegible]



- The representation of a ***Compound Formulation***

  A "formulation" is a set of atomic chemical elements mereologically organized:
  (A) A formulation consists of one or more ***functional systems****;*
  (B) A functional system is an aggregate of ***atomic ingredients****.*

- A set of specific ***lab tests*** are carried out in correspondence to any formulated compound recipe.
- Lab tests return a ***quantitative evaluation*** of the compound recipe.
- Lab tests evaluate the chemical compound in isolation.



- A set of specific ***outdoor tests*** are carried out in correspondence to any formulated compound recipe.
- Outdoor tests return a ***qualitative evaluation*** of the compound recipe.
- Outdoor tests evaluate the chemical compound once integrated in the whole tire.

Ontological Compounding Knowledge - I

TBox

Terminological Knowledge
(Background Knowledge)

General Concepts — e.g. Compound

Domain Axioms — e.g. TreadCompound

ABox

Assertional Knowledge
(Knowledge about individuals)

Concept and Role Assertions — e.g. xyz *is a compound*

Ontological Compounding Knowledge - II

Basic Description Logics

The core part of any Description Logics is the **concept language**, containing: **concept names** (assigning a name to groups of objects), **atomic roles** (assigning a name to relations between objects), **constructors** (allow to relate concept names and role names).

$\top$	[universal concept]	$\top^{\mathfrak{I}}$	$=$	$\Delta^{\mathfrak{I}}$
$\bot$	[bottom concept]	$\bot^{\mathfrak{I}}$	$=$	$\varnothing$
$(C \sqcap D)$	[conjunction]	$(C \sqcap D)^{\mathfrak{I}}$	$=$	$C^{\mathfrak{I}} \cap D^{\mathfrak{I}}$
$(C \sqcup D)$	[disjunction]	$(C \sqcup D)^{\mathfrak{I}}$	$=$	$C^{\mathfrak{I}} \cup D^{\mathfrak{I}}$
$\neg C$	[atomic negation]	$\neg C^{\mathfrak{I}}$	$=$	$(\Delta^{\mathfrak{I}} \backslash C^{\mathfrak{I}})$
$(\exists R.\top)$	[limited existential quantification]	$(\exists R.\top)^{\mathfrak{I}}$	$=$	$\{d \in \Delta^{\mathfrak{I}} \mid \text{there exists an } e \in \Delta^{\mathfrak{I}}, \text{with } (d,e) \in r^{\mathfrak{I}}\}$
$(\forall R.C)$	[value restriction]	$(\forall R.C)^{\mathfrak{I}}$	$=$	$\{d \in \Delta^{\mathfrak{I}} \mid \text{for all } e \in \Delta^{\mathfrak{I}}, \text{if } (d,e) \in r^{\mathfrak{I}} \text{ then } e \in C^{\mathfrak{I}}\}$

\+ [full existential quantification]

$(\exists R.C)$ $\quad (\exists R.C)^{\mathfrak{I}} = \{d \in \Delta^{\mathfrak{I}} \mid \text{there exists a } e \in \Delta^{\mathfrak{I}}, \text{with } (d,e) \in R^{\mathfrak{I}} \text{ and } e \in C^{\mathfrak{I}}\}$

\+ [full qualified number restriction]

$(\geq n\ R.C)$ $\quad (\geq n\ R.C)^{\mathfrak{I}} = \{ d \in \Delta^{\mathfrak{I}} \mid \|\{e \mid (d,e) \in R^{\mathfrak{I}}\}\| \geq n \}$

$(\leq n\ R.C)$ $\quad (\leq n\ R.C)^{\mathfrak{I}} = \{ d \in \Delta^{\mathfrak{I}} \mid \|\{e \mid (d,e) \in R^{\mathfrak{I}}\}\| \leq n \}$

\+ [inverse role]

(R^{-1}) $\quad (R^{-1})^{\mathfrak{I}} = \{ (d,e) \in \Delta^{\mathfrak{I}} \times \Delta^{\mathfrak{I}} \mid (e,d) \in R^{\mathfrak{I}} \}$

Relation *has-part* "$\succ$" (*Finite*, *irreflexive*, *asymmetric*, and *intransitive*)

$\prec \; =_{\text{def}} \; \succ^{-1}$

General Concepts and Axioms Compound, System, GroundElement

Disjointness of concepts

$$\bot \sqsupseteq \text{Compound} \sqcap \text{System}$$
$$\bot \sqsupseteq \text{Compound} \sqcap \text{GroundElement}$$
$$\bot \sqsupseteq \text{System} \sqcap \text{GroundElement}$$

Basic Mereological Set-up

$$\text{Compound} \equiv ((\geq 1 \; \succ.\text{System}) \sqcap (\leq n \succ.\text{System})) \sqcap \forall \prec.\bot$$

$$\text{System} \equiv ((\geq 1 \; \succ.\text{GroundElement}) \sqcap (\leq n \succ.\text{GroundElement})) \sqcap (=1 \prec.\text{Compound})$$

$$\text{GroundElement} \equiv (=1 \prec.\text{System}) \sqcap f_1.\text{NonNegativeInteger} \sqcap \ldots \sqcap f_m.\text{ NonNegativeInteger} \sqcap \forall \succ.\bot$$

TBox Compounding Axioms

A TreadCompound is a domain entity that is a Compound, having exactly 5 systems as its parts (i.e. PolymericMatrix, Vulcanisation, ProcessAid, ReinforcingFiller, and Antidegradant). It may or may not have a Softener system as its part

$$\begin{aligned} \text{TreadCompound} \equiv \text{Compound} \; & \sqcap (=1 \succ.\text{PolymericMatrix}) \\ & \sqcap (=1 \succ.\text{Vulcanisation}) \\ & \sqcap (=1 \succ.\text{ProcessAid}) \\ & \sqcap (=1 \succ.\text{ReinforcingFiller}) \\ & \sqcap (=1 \succ.\text{Antidegradant}) \\ & \sqcap ((\geq 0 \succ.\text{Softener}) \sqcap (\leq 1 \succ.\text{Softener})) \end{aligned}$$

A PolymericMatrix is a domain entity that is a System, having exactly 100 ground elements as its parts, that are instances of the concepts NaturalRubber and SyntheticRubber

$$\text{PolymericMatrix} \equiv \text{System} \sqcap (=100 \succ.(\text{NaturalRubber} \sqcup \text{ButadieneRubber}))$$

A ReinforcingFiller is a domain entity that is a System, whose parts are CarbonBlack and Silica

$$\begin{aligned} \text{ReinforcingFiller} \equiv \text{System} \; & \sqcap ((\geq 40 \succ.\text{CarbonBlack}) \sqcap (\leq 60 \succ.\text{CarbonBlack})) \\ & \sqcap ((\geq 0 \succ.\text{Silica}) \sqcap (\leq 25 \succ.\text{Silica})) \end{aligned}$$

CarbonBlack is a domain entity that is a GroundElement. It is an exclusive part of the ReinforcingFiller system and has two distinctive attributes: surface area and porosity, usually expressed by numerical values

$$\begin{aligned} \text{CarbonBlack} \equiv \text{GroundElement} \; & \sqcap \exists \prec \text{ReinforcingFiller} \\ & \sqcap \mathit{hasSurfaceArea}.\text{NonNegativeInteger} \\ & \sqcap \mathit{hasPorosity}.\text{NonNegativeInteger} \end{aligned}$$

ABox Knowledge about individuals

- By means of the ABox, one describes a specific state of affairs of an application domain (e.g. a compound recipe) in terms of *concepts* and *roles.*

- In the ABox, one introduces *individuals*, by giving them names, and one asserts properties of these individuals.

- Using concepts and roles that has been defined in the TBox (e.g. *Compound* and *hasPart*), one can make *concept assertions* and *role assertions* of the following form:

TreadCompound(ALPHA)
ReinforcingFiller(RF001)
CarbonBlack(CB001)
Silica(SIL00A)
$\succ$ (RF001,SIL00A)
$\succ$ (ALPHA,RF001)
⋮

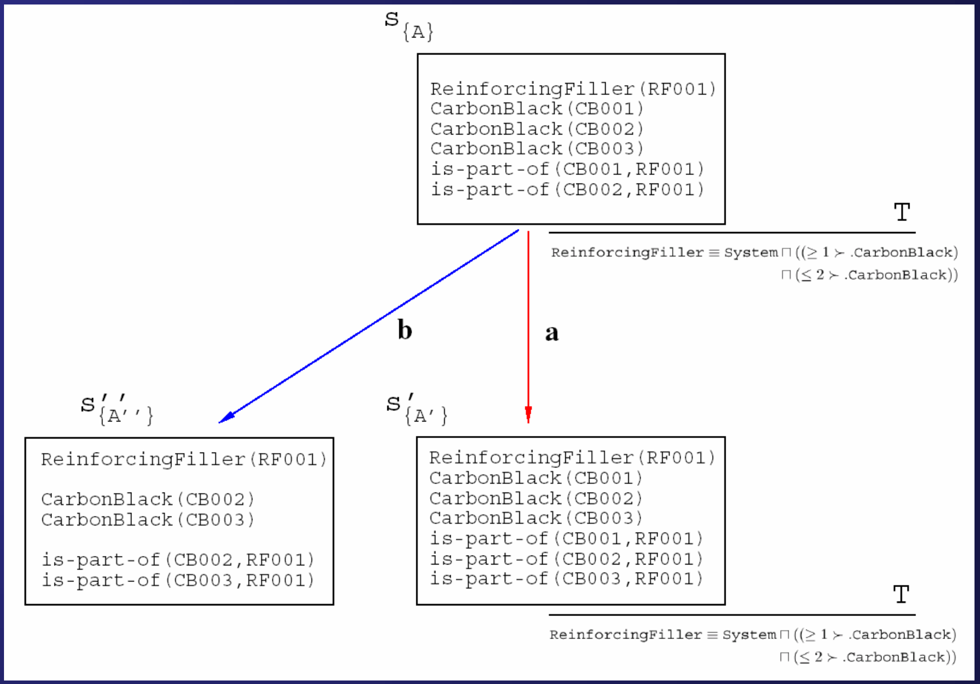
$\mathrm{ALPHA}^{\mathfrak{I}} \in \mathrm{TreadCompound}^{\mathfrak{I}}$
$\mathrm{RF001}^{\mathfrak{I}} \in \mathrm{ReinforcingFiller}^{\mathfrak{I}}$
$\mathrm{CB001}^{\mathfrak{I}} \in \mathrm{CarbonBlack}^{\mathfrak{I}}$
$\mathrm{SIL00A}^{\mathfrak{I}} \in \mathrm{Silica}^{\mathfrak{I}}$
$(\mathrm{RF001},\mathrm{SIL00A}) \in \succ^{\mathfrak{I}}$
$(\mathrm{ALPHA},\mathrm{RF001}) \in \succ^{\mathfrak{I}}$
⋮

Semantics

- An ABox is a finite set of such assertions.

Mereological Knowledge *in Action!*

The Compound Formulation Space as an example of the integration



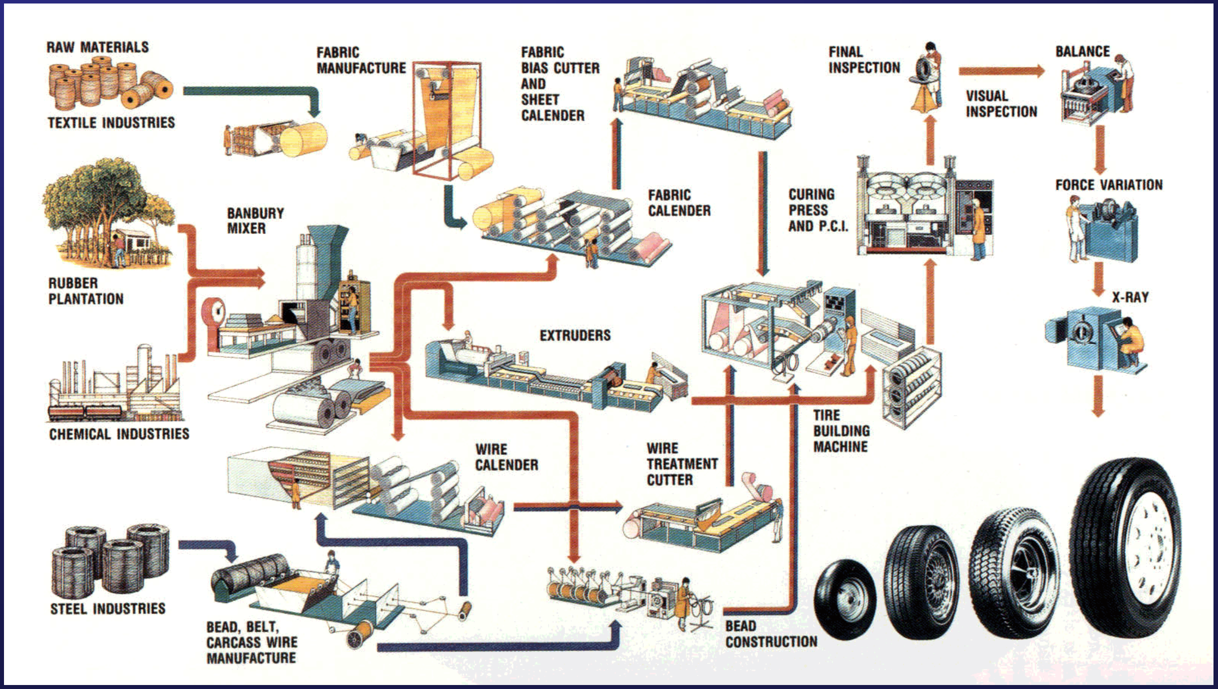
- The state *s* at the root collects the assertions of the ABox A and, since there is a model of A w.r.t. T, it is an <u>ontologically consistent state</u> of the problem space.
- The state *s*' collects the assertions of the ABox A' and, since there is no model of A' w.r.t. T, it is an <u>ontologically inconsistent state</u> of the problem space (i.e. *it cannot be an admissible solution*).
- The state *s*'' collects the assertions of the ABox A'' and, since there is a model of A'' w.r.t. T, it is an ontologically consistent state of the problem space (i.e. *it could be an admissible solution*).
- As a consequence, *red line* "a" denotes the <u>ontologically inconsistent transition</u> (*s,a,s'*), while the *blue line* denotes a <u>ontologically consistent transition</u> (*s,b,s*'').

Case Study

An AI system for compounding in tire industry: the **P-Truck** domain

Design by Adaptation of Rubber Compounds

A rubber compound is a blend of different ingredients, both natural (e.g. natural rubber, resins) and synthetic (e.g. carbon blacks, oils). This design phase has to decide the blend composition, identifying a set of ingredients and their amount, in order to achieve the performances that are required for the blend and for the tyre (e.g. tensile strength, resistance to fatigue).



Two Automatic Systems

- The first system is an *Iterative Deepening A** algorithm and it is based on the Heuristic Search Paradigm.
  - The *heuristic* that control the exploration of the space is a function that computes the distance between the current state and the goal state, and the distance between the current state and the starting one.
  - The function assigns a numerical value to each explored state on the basis of this computation, and the value represents the potential of this state in being a feasible solution for the problem.
- The second system is a *Genetic Algorithm*.
  - The system implement a population of individuals, that are arrays of possible compound transformations, and evolves by means of the usual evolutionary operators of *Crossover* and *Mutation*.
  - The system assigns the worst value of Fitness to the individuals in the population that produce ontologically illegal compound formulations and, therefore, the evolving process of the genetic algorithm manages in the usual way the individuals of the population.

Concluding Remarks

- The problem solving method for the Chemical Formulation Problem is the result of the formal integration of ***causal*** and ***ontological*** knowledge.
- ***Description Logics*** have been used as knowledge representation formalisms for modeling mereological expert knowledge on compounds.
- The integration of different knowledge sources and formal representations has enabled the definition of a ***unifying computational model*** of compounding.
- The application of the proposed computational model to the chemical domain of *rubber compounds design for tire industry* has leaded to a successful system implementation that has been ***tested on concrete chemical compounding problems***.

… and Future Directions

- Implement a computational model exploiting some existent DL reasoner (e.g. Fact++, RacerPRO, Pellet, …).
- Extend our ontology and connect it to structural ontology from Chemistry, and

  therefore…
- Defining a completely logic-based approach for Compounding.

Thank You!