Toward natural optimization into CAD software

Or how to simply integrate an optimization tool into a CAD software for solving a whole class of problems

Jean - Christophe FAUROUX
LaRAMA - Laboratoire de Recherches et Applications en Mécanique Avancée
IFMA - Clermont-Ferrand - FRANCE

IDMME'2000
3rd International Conference on Integrated Design and Manufacturing in Mechanical Engineering
May 16-19, 2000
Montréal, CANADA

Summary

Introduction

Engineering in the 90’s
=
CAD **+** FEM **???**

Lack of CAD tools for **optimizing** products

Optimizing : a major concern in industry

- User-friendly GUI
- Deeper integration into CAD software

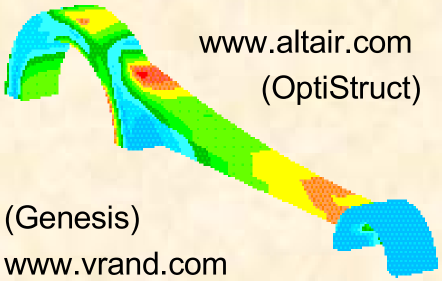
[Vanderplaats 97]

Introduction :
Optimization for Engineering

Several types of **optimizing tools**

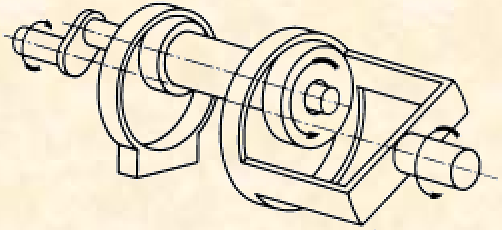
Structural optimization

- One FEM calculation at each optimization iteration
- General method for every type of structure
- Heavy computations



Optimization for **preliminary design**

- Not the same purpose
- Upstream in the design process
- Faster computations

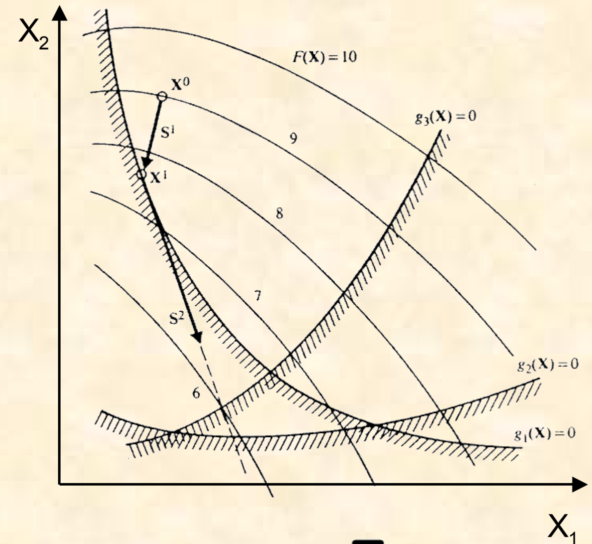


Our vision of integration of CAD + optimization tools for preliminary design

Nature of the problem
Optimization programming

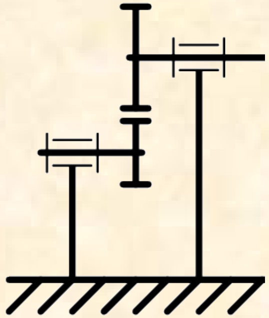
Using optimizing tools for mechanism design

- Hard to use for beginners
- Rigorous method and formalism :
  - Model of the mechanical problem
  - Objective function, constraint functions, solving method
  - Compiling and linking with optimization library
  - Running
  - Reading results
  - Graphic representation of optimized mechanism



Suitable for specific problems

- Ex : speed reducer (1 cylindrical gearing)
- For simple and standard mechanisms

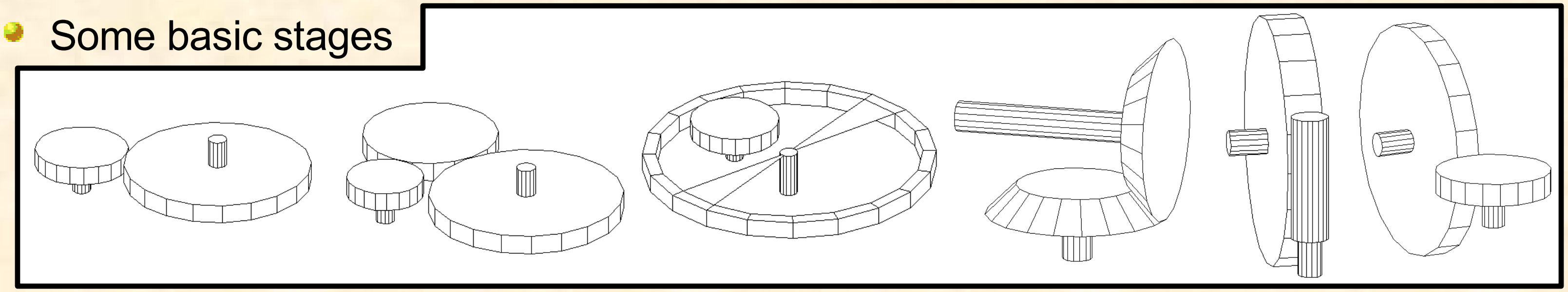


Nature of the problem

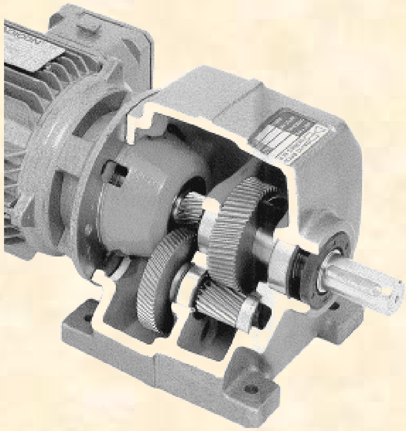
- Using optimizing tools for **general shape optimization problems**

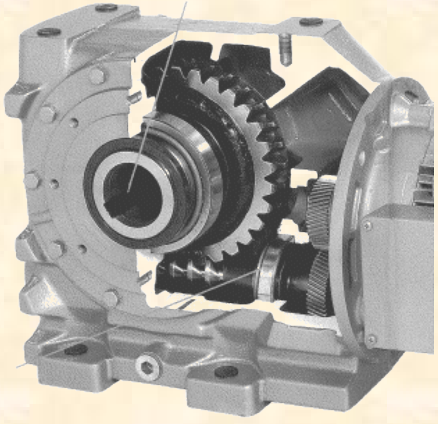
- Example : Optimization of **any type** of geared speed reducer

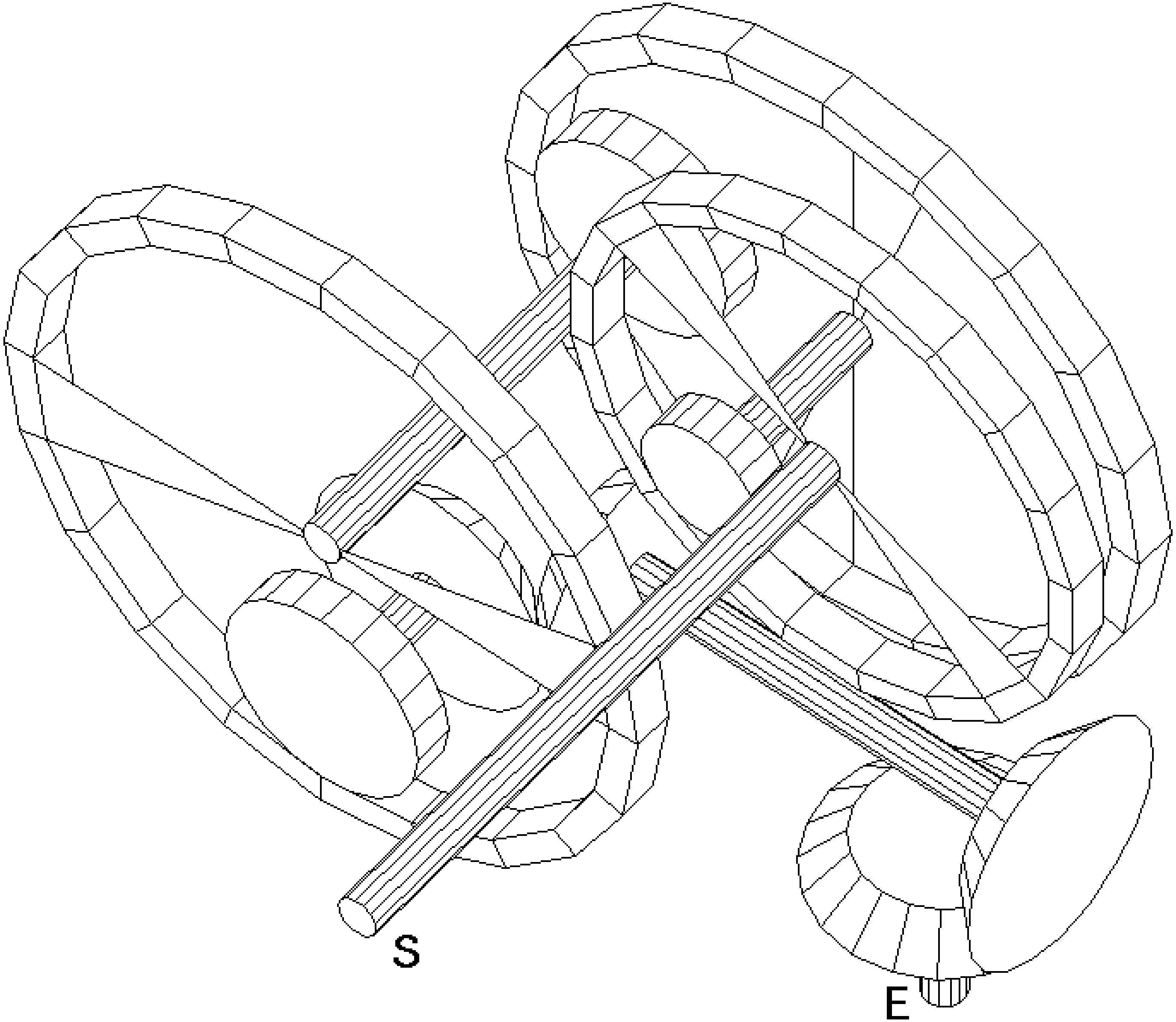
  - Some basic stages



  - Two types of geared speed reducers :

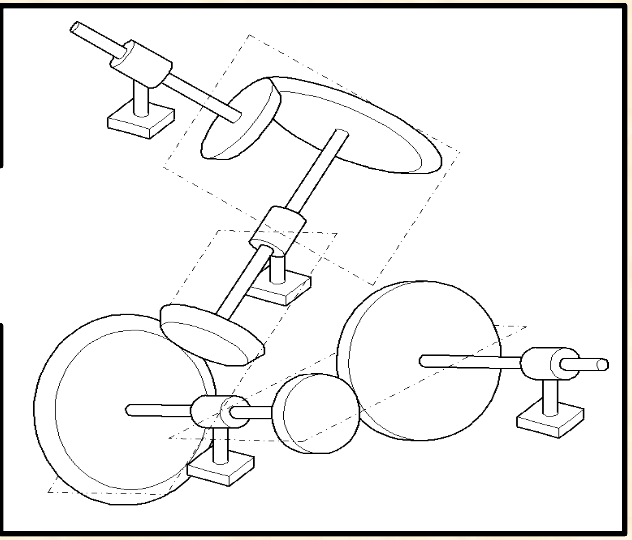





S
E

Nature of the problem

- How to solve **a complete class of problems**
- Similarities between problems :
  - Same **objective function** (volume, cost, ...)
  - Same **constraint types**
    - Geometrical ⇒ Non-interference, Continuity, Closure condition
    - Technological ⇒ Speed ratio, Efficiency, Part resistance
    - Local level ⇒ Tooth resistance, Shaft torsion, Maximal tangential speed — depend on **EACH STAGE**
    - Global level ⇒ Speed ratio, Efficiency, Closure constraint — depend on **ALL STAGES**



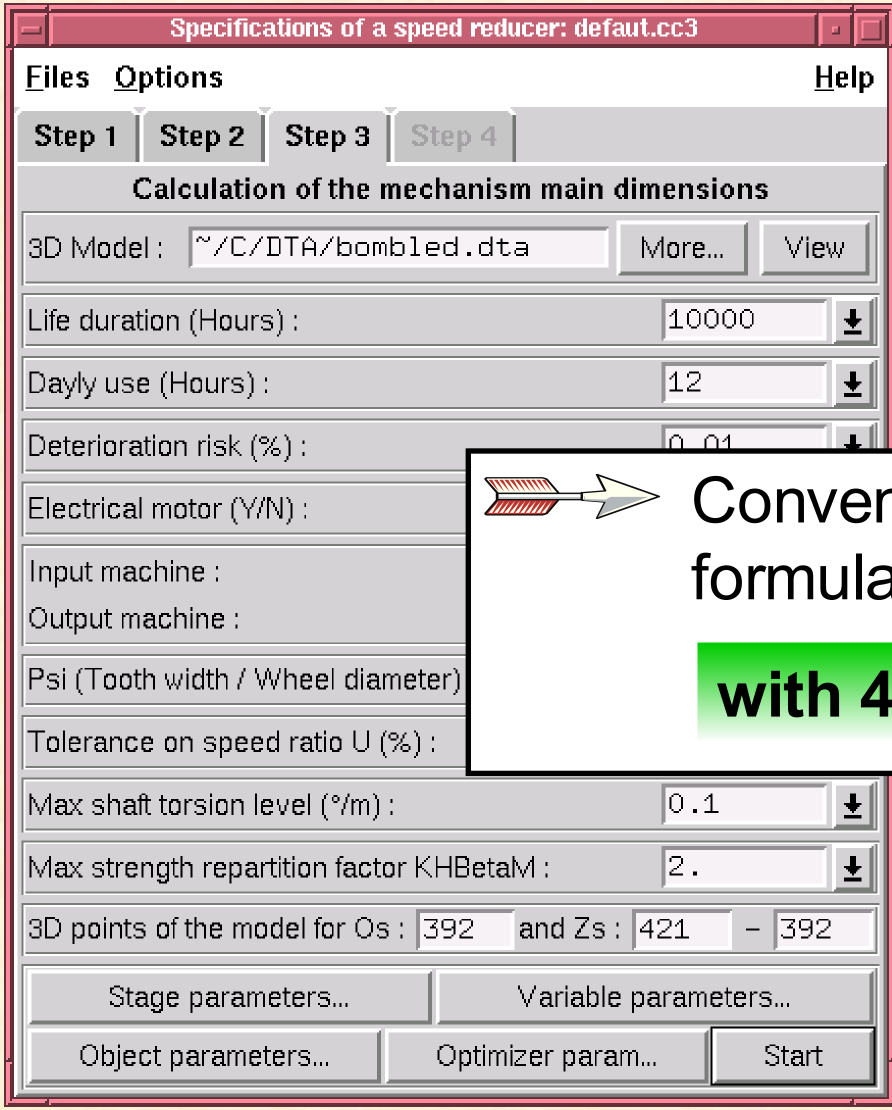
Nature of the problem : So what ?

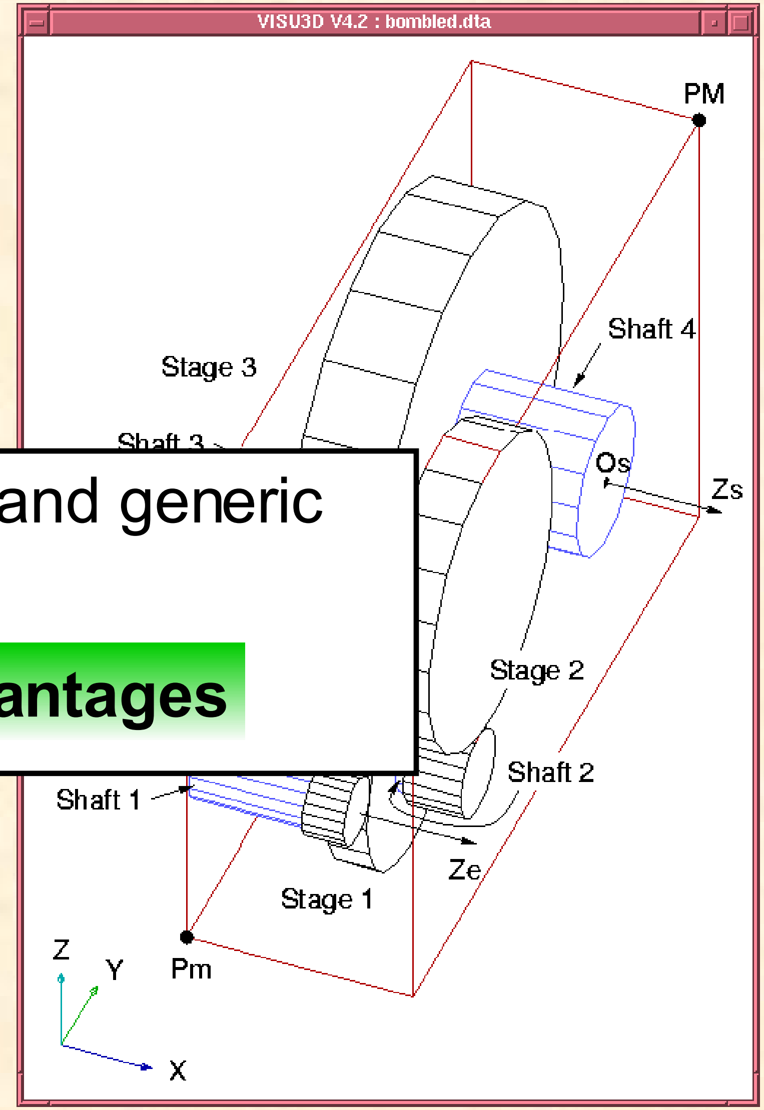
- Constructing each optimization problem is **tedious**
- Many constraints change :
  - Geometrical constraints must all be rewritten
  - Global constraints change $\Rightarrow$ **speed ratio**
  - Some local constraints do not change $\Rightarrow$ **tooth resistance**
- Representing the optimized mechanism is **boring**

The challenge :

Integrating CAD and optimization tools for solving a complete class of problems

Solving method



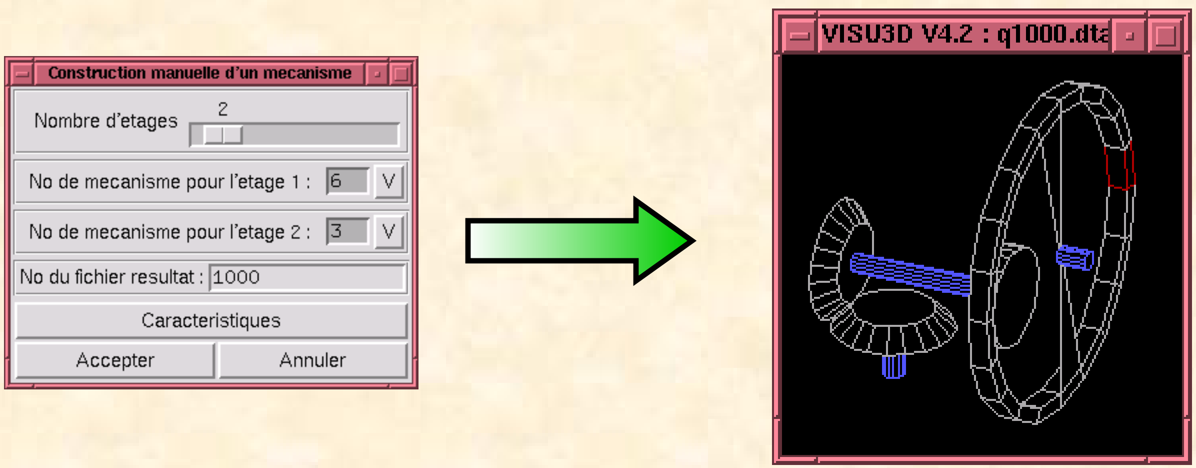


Convenient and generic formulation

with 4 advantages

Solving method
Automatic mechanism drawing

- Object oriented representation
- Fully parametered model
- Automatic mechanism continuity

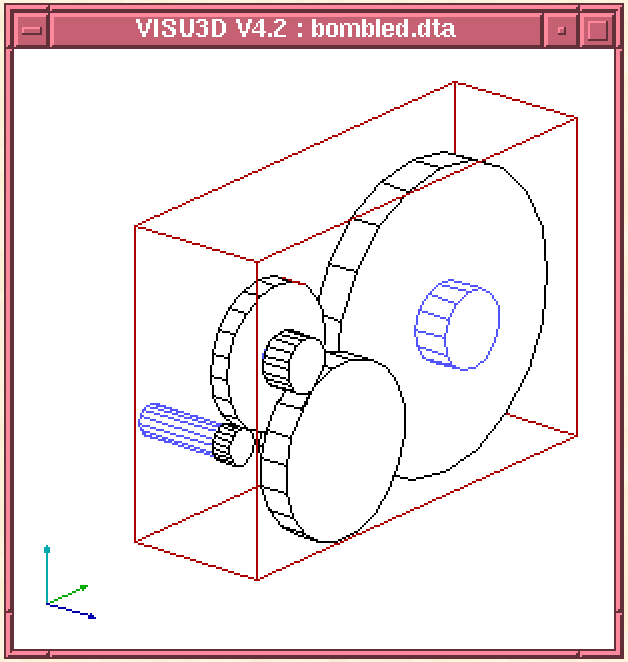


Solving method
Linking CAD and Optimization tools (1)

$$\begin{pmatrix} \alpha_{stage\_1} \\ D_{shaft\_1} \\ L_{shaft\_1} \\ D_{wheel\_1} \\ b_{wheel\_1} \\ \ldots \\ D_{wheel\_n} \\ b_{wheel\_n} \end{pmatrix}$$

- Variable vector
- CAD model :
- Correspondance :

- Pure geometrical problem
- Respective roles :



Optimization tool :

- Reads variables in the CAD model
- Improves variable vector with any numerical method
- Re-injects variable vector in CAD model

CAD software :

- Stores the current state of the model
- Performs graphical representation
- Updates model at each dimension change

Solving method
Linking CAD and Optimization tools (2)

The coupling offers **numerous advantages** :

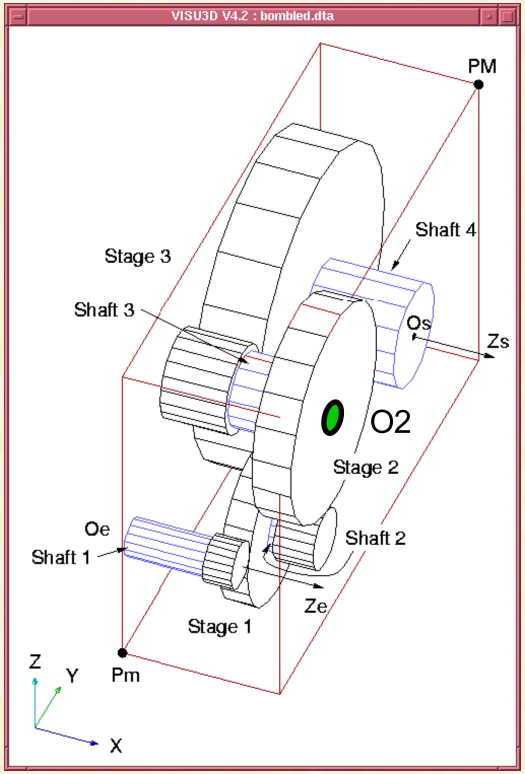
- Avoids duplicate data storage

- No more complex relations to write

  Example : wheel position of stage 2

- Generic writing of constraints

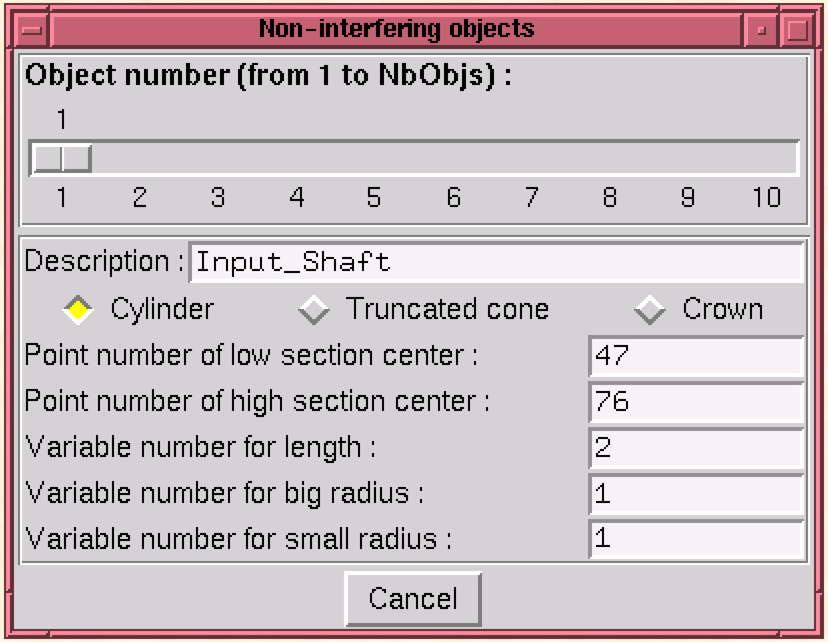
  When the nature of mechanism changes, no need to rewrite constraints



Solving method
Linking CAD and Optimization tools (3)

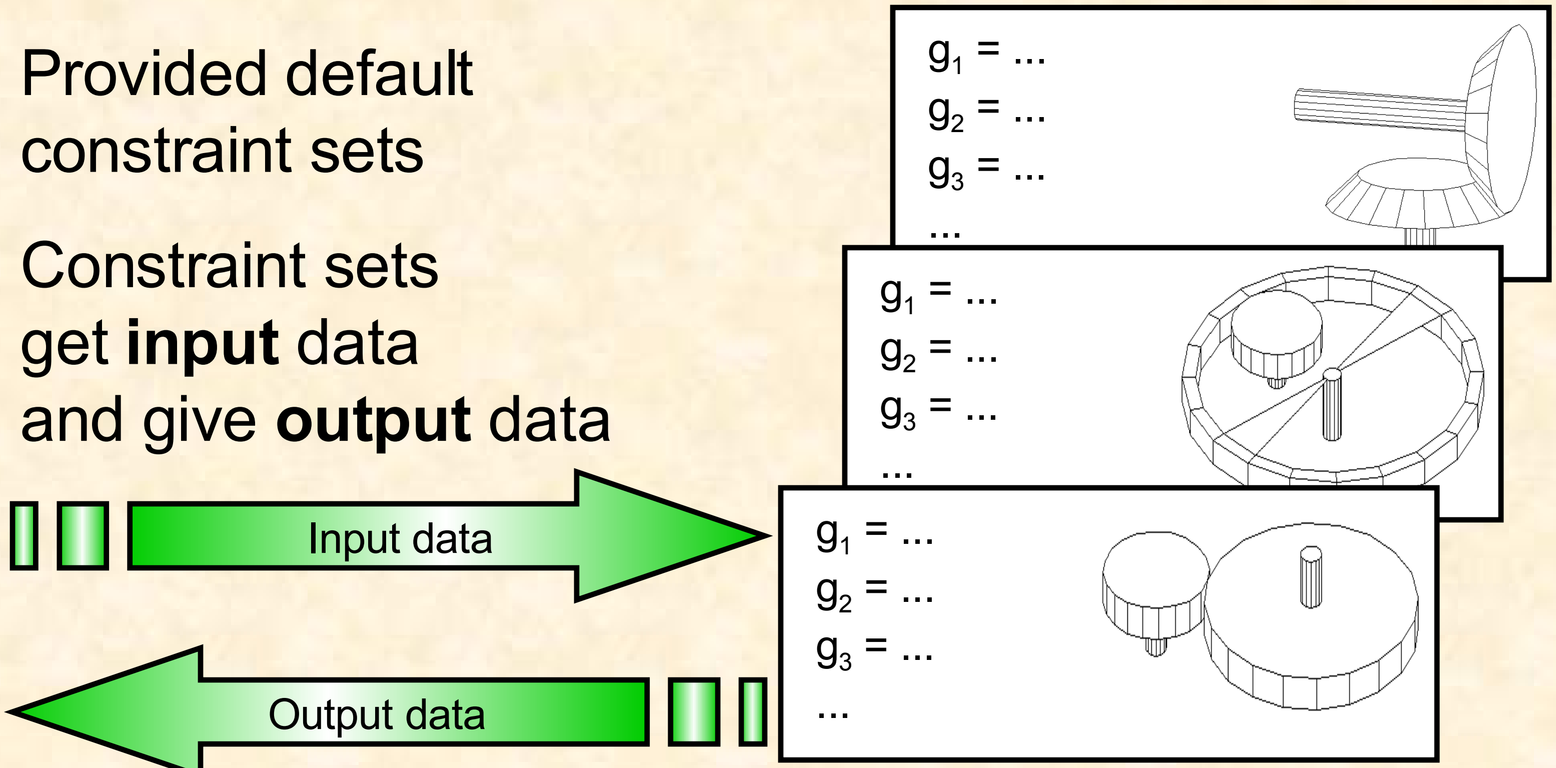
From the user point of view :

- Characteristics of mechanism stages
- Nature of each variable
- Nature of each object of CAD model

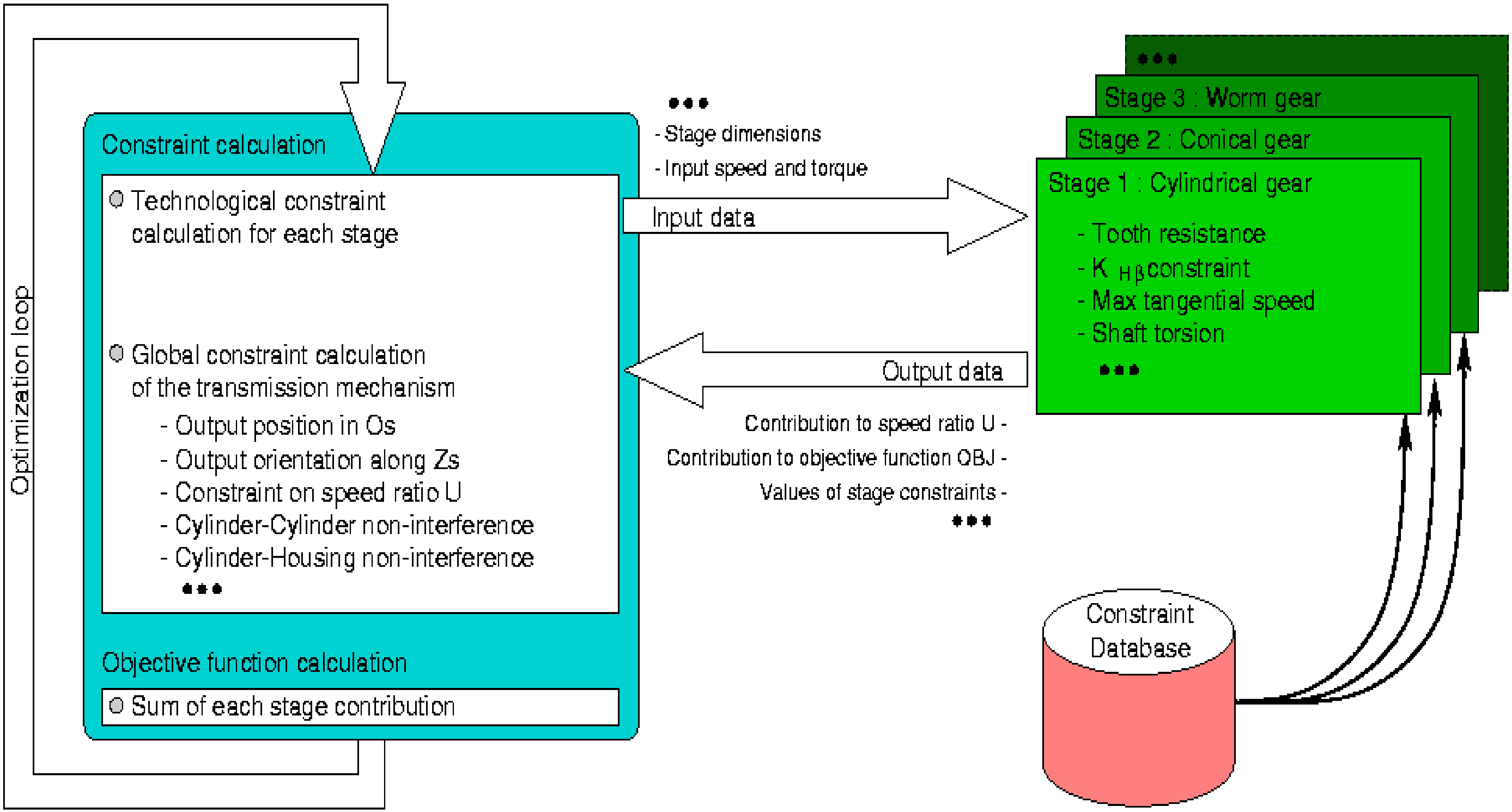


Solving method
Automatic constraint construction (1)

- A **technological constraint set** for each stage
  - Stored in a database
  - C language
- Provided default constraint sets
- Constraint sets get **input** data and give **output** data

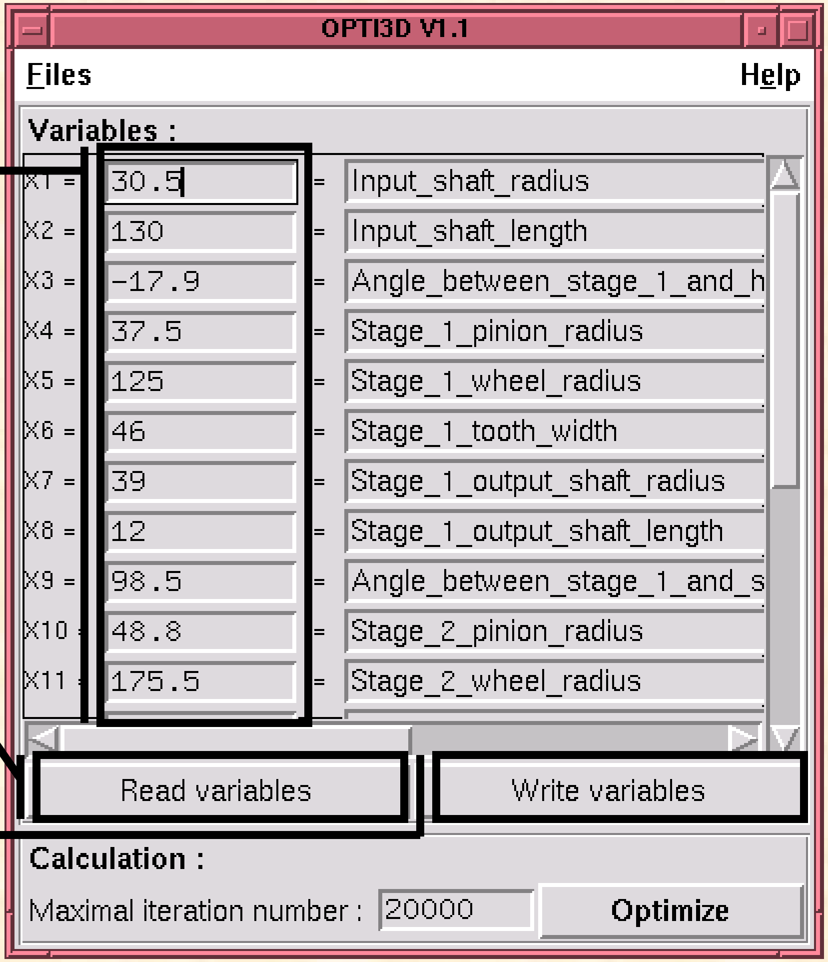


Solving method
Automatic constraint construction (2)



Solving method
Interactive resolution (1)

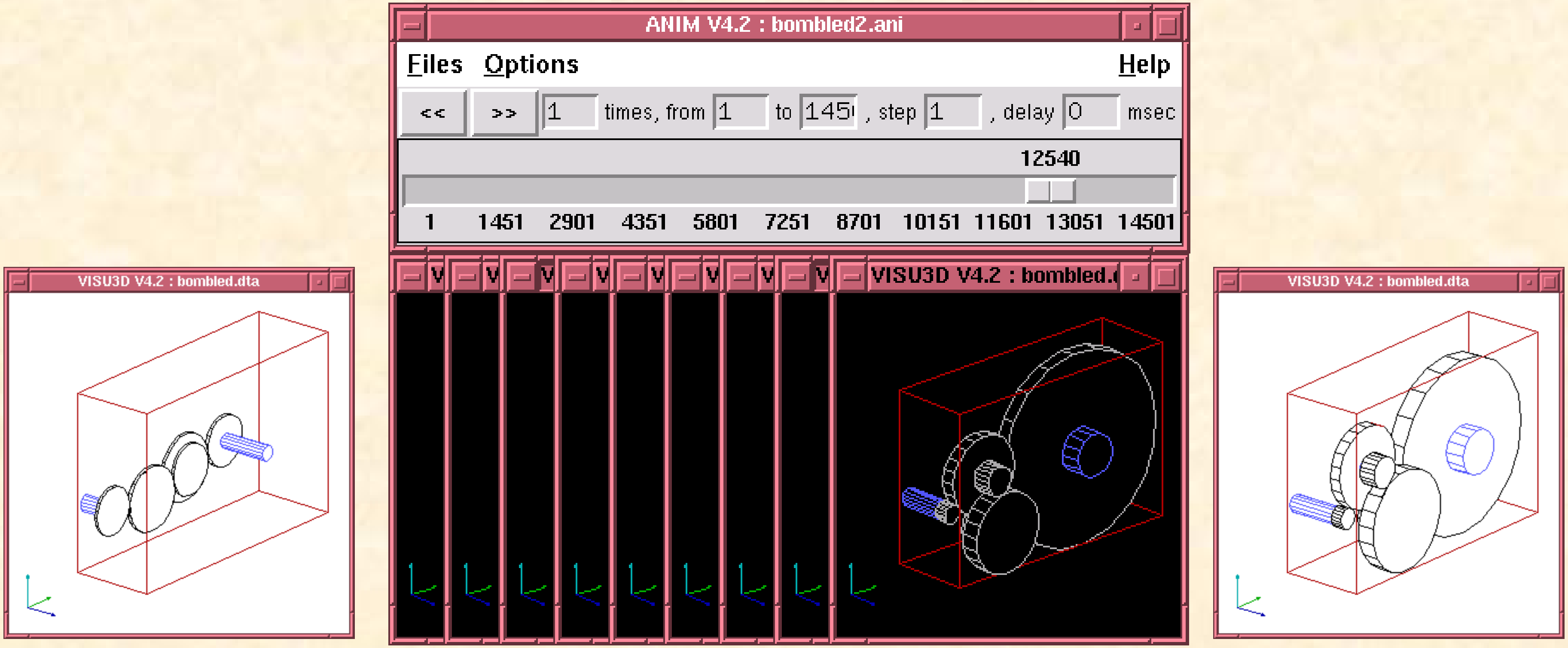
- Before starting resolution :
  - Setting each variable independently
  - Reading all values from CAD model
  - Writing all values to CAD model



- Interactivity for setting initial conditions

Solving method
Interactive resolution (2)

- Real time solving (typically half an hour on a Pentium Pro 233 running Linux)
- Virtual video recorder : **record** / **play** / **stop and edit**
- Watch convergence graphically



Solving method
Interactive resolution (3)

Displaying results

Result : constraint and objective functions

Iteration number : 11

Constraint functions :

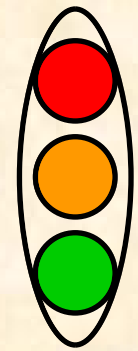
```
G[001] =      -0.4647887323943662 = EN = Closure position Os projected along  X
G[002] =      -0.5913901145000502 = EN = Idem along Y
G[003] =       0.1658369315775911 = EN = Idem along Z
G[004] =                        0 = EN = Closure orientation Zs projected along X
G[005] =                        0 = EN = Idem along Y
G[006] =                        0 = EN = Idem along Z
G[007] =        50.08643054218799 = IN = Max torsion ratio for input shaft
G[008] =                     -0.5 = IL = Input pinion diameter bigger than shaft diameter
G[009] =      -0.2499999999999999 = IN = Psi = b/d1 > PsiMin for stage 1
G[010] =      -0.8333333333333334 = IN = Psi = b/d1 < PsiMax for stage 1
G[011] =        8.943116724876999 = IN = Superficial pressure for stage 1
G[012] =      -0.4178750000000001 = IN = KHbeta < KHbetaMax for stage 1
G[013] =      -0.8661253130572532 = IN = Minimal required torque for stage 1
G[014] =                       -1 = IL = Tooth width bigger than 50mm for stage 1
G[015] =                    -0.98 = IL = Tooth width smaller than 500mm for stage 1
G[016] =      -0.8429203673205103 = IN = Tangential speed < VtMax for stage 1
...
G[038] =                        0 = IN = Object 1 upside plane X=XMin
G[039] =      -0.6056338028169014 = IN = Object 1 under plane X=XMax
...
G[098] =                        0 = IN = Interference obj. 2 / 1
...
G[142] =                        0 = IN = Interference obj. 10 / 9
```

Objective function (volume in mm3) :

238761.0416728243

Cancel

- Constraint status



Violated

Active

Verified

- Constraint nature

EN	=	Non linear Equality
EL	=	Linear Equality
IN	=	Non linear Inequality
IL	=	Linear Inequality

Conclusion

- Method for coupling Optimization + CAD tools
  Collaborative Optimization
- Suitable for solving classes of problems
- Little **drawback** : long solving
- Big **advantage** : extra-short problem implementation

Towards a more natural integration
of Optimization within CAD tools