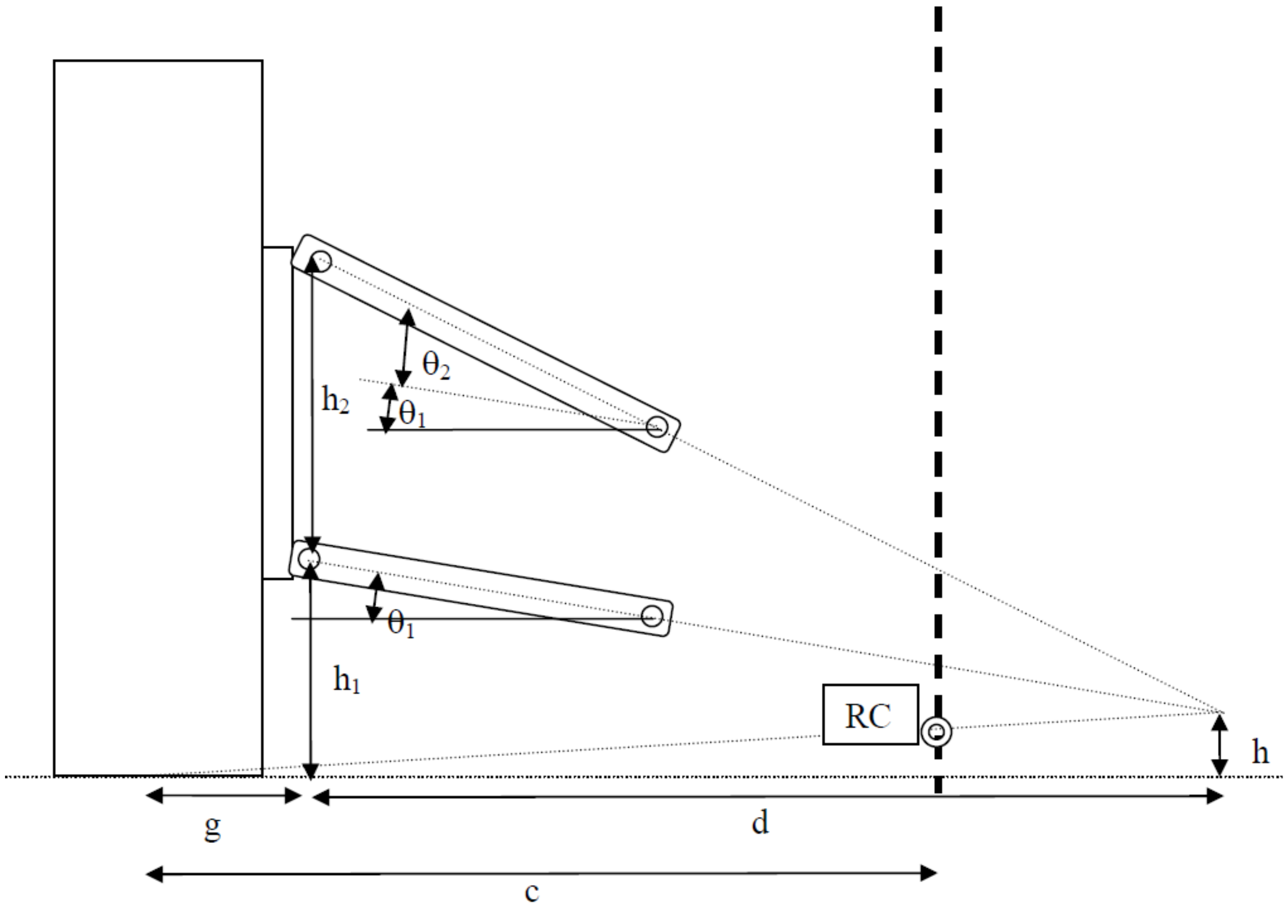
OPTIMAL DESIGN

TRAINING LAB. 7

MULTI-OBJECTIVE OPTIMISATION: DOUBLE WISHBONE SUSPENSION DESIGN

Simplified scheme of a double wishbone suspension



- The camber rate of change is given by:

$$\frac{\Delta\gamma}{\Delta z} = \frac{1}{d+g}$$

- The height of the roll centre (RC) is:

$$RCH = \frac{h}{d+g}c$$

with

$$d = \frac{h_2}{\tan(\theta_1 + \theta_2) - \tan(\theta_1)}$$
$$h = h_1 - d\tan(\theta_1)$$

Data:

$$h_1 = 0.190 \quad [m]$$
$$h_2 = 0.270 \quad [m]$$
$$g = 0.050 \quad [m]$$
$$c = 0.870 \quad [m]$$

The design variables are the angles θ_1 and θ_2 that define the inclination of the two wishbone arms.

The range of variation of the design variables are reported below and mainly depend on the available space in the vehicle.

	Lower bound	Upper bound
θ_1	-9 [deg]	-1 [deg]
θ_2	+5 [deg]	+15 [deg]

Targets on the height of the roll centre and on the gradient of camber angle are given

	TARGET value
TRCH	100 [mm]
$T\Delta\gamma\|_{\Delta z=50mm}$	2 [deg]

The task of the designer is therefore getting as close as possible to the targets. The two objective functions to minimize are therefore defined as

$$\min\left(\begin{array}{c} f_1 = \dfrac{(RCH - TRCH)^2}{TRCH^2} \\[2ex] f_2 = \dfrac{\left(\Delta\gamma|_{\Delta z=50mm} - T\Delta\gamma|_{\Delta z=50mm}\right)^2}{\left(T\Delta\gamma|_{\Delta z=50mm}\right)^2} \end{array}\right)$$

Request:

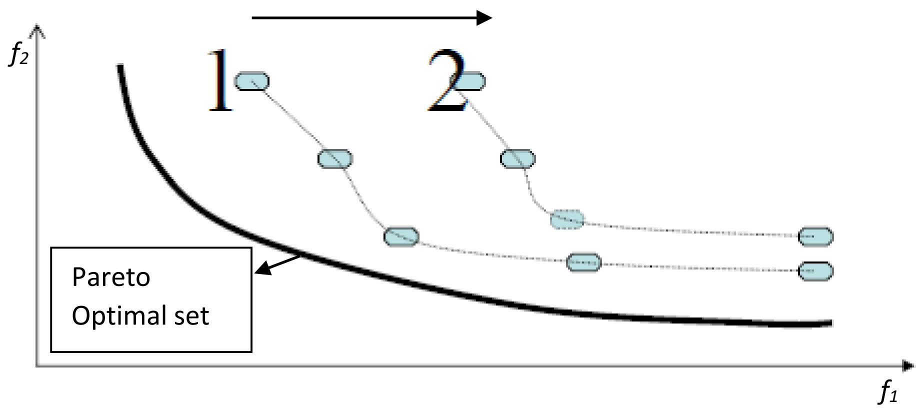
- Solve the minimization problem by using a genetic algorithm
- Plot the starting population and the final population in the design variables and objective functions domain.
- Plot the number of individuals of rank1 at each generation

Lab. 7 tutorial

The workflow of the algorithm is identical to the one of Lab. 6.

The main difference is in the computation of the fitness.

Rank calculation:



At each generation of the genetic algorithm we have a population of individuals. Each individual is characterized by a combination of values for the design variables and objective functions.

The set of individuals pertaining to a certain generation can be stored into a matrix.

Suppose for instance that at each generation a population of 6 individuals is obtained:

Individual 1	DV1	DV2	Fob1	Fob2
Individual 2	…	…	…	…
Individual 3	…	…	…	…
Individual 4	…	…	…	…
Individual 5	…	…	…	…
Individual 6	…	…	…	…

- Let's start by computing the Pareto-optimal set of the current population with the sorting method (suppose that the green ones are Pareto-optimal solutions):

Individual 1	DV1	DV2	Fob1	Fob2
Individual 2	…	…	…	…
Individual 3	…	…	…	…
Individual 4	…	…	…	…
Individual 5	…	…	…	…
Individual 6	…	…	…	…

- The individuals highlighted in green have therefore rank equal to 1.

- Now the obtained Pareto-optimal solutions are removed from the matrix, which now reads:

Individual 1	DV1	DV2	Fob1	Fob2
Individual 3	…	…	…	…
Individual 5	…	…	…	…
Individual 6	…	…	…	…

- The next step is now to compute the Pareto-optimal solutions of this new matrix (suppose now highlighted in red):

Individual 1	DV1	DV2	Fob1	Fob2
Individual 3	…	…	…	…
Individual 5	…	…	…	…
Individual 6	…	…	…	…

- To the individuals highlighted in red will be assigned rank equal to 2.

- Now the obtained Pareto-optimal solutions are removed from the matrix, which now reads:

Individual 1	DV1	DV2	Fob1	Fob2
Individual 5	…	…	…	…

- Again we choose the new Pareto-optimal solutions in the remaining individuals (suppose highlighted in yellow):

Individual 1	DV1	DV2	Fob1	Fob2
Individual 5	…	…	…	…

- To the yellow individuals will be assigned rank equal to 3.
- Now that the rank has been assigned to all the individuals the fitness is assigned with an inverse correlation w.r.t. the rank (a possible relation could be fitness=1/rank):

rank

Fitness = 1/rank

Individual 2	DV1	DV2	Fob1	Fob2	1	1
Individual 4	…	…	…	…	1	1
Individual 3	…	…	…	…	2	0.5
Individual 6	…	…	…	…	2	0.5
Individual 1	…	…	…	…	3	0.3333
Individual 5	…	…	…	…	3	0.3333