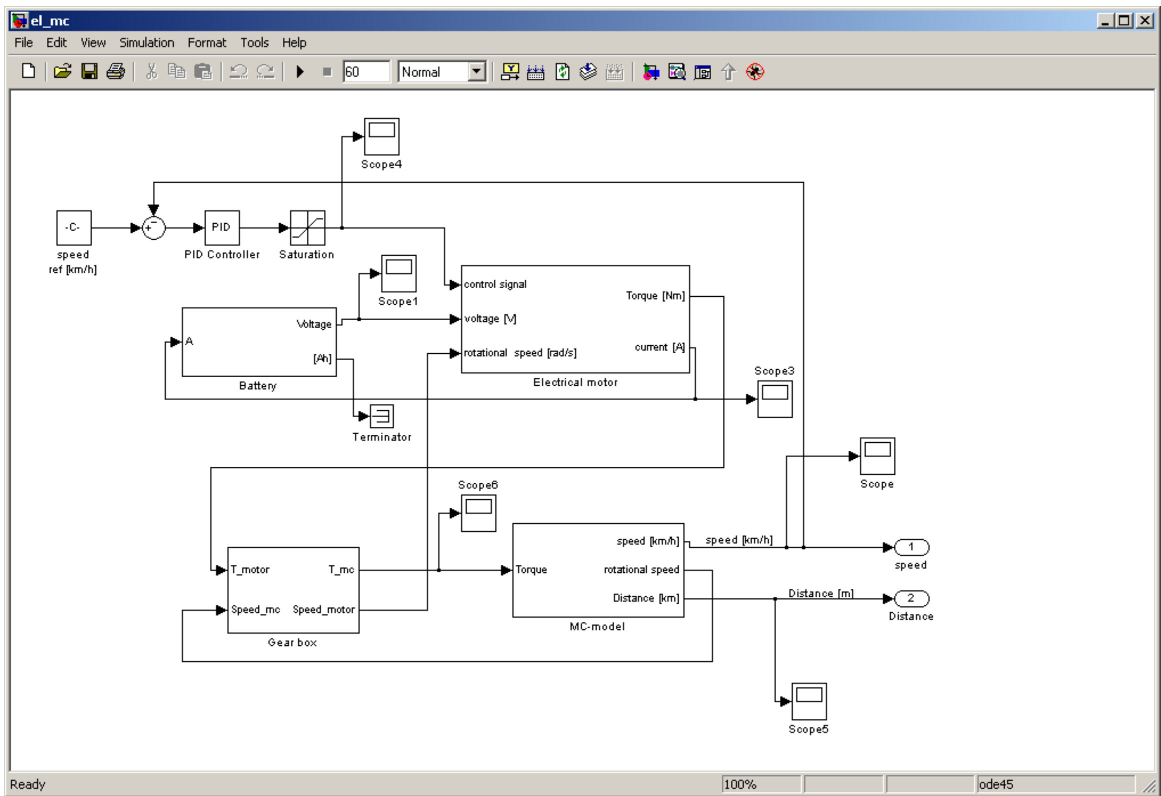
Simulation and optimization

Now we should study some simulation based design problems



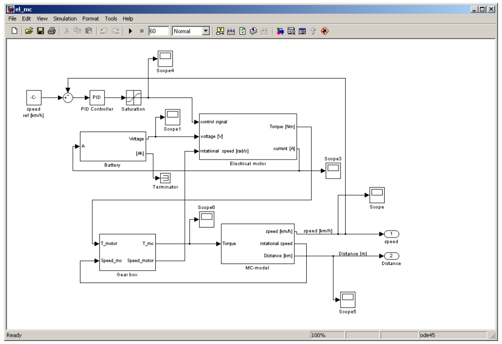
Contents

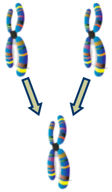
- How does it work
- MATLAB/Simulink example
  - Controller design
  - Electrical motorcycle
  - Two bar truss
- Research application examples

How does it work?

- The simulation program goes in where the objective function was in the optimization code
- Iterative process where a simulation is performed each time a new design has to be evaluated.

$$f(\mathbf{x}) = \int_{t_1}^{t_2} \left| y - y_{ref} \right| dt$$





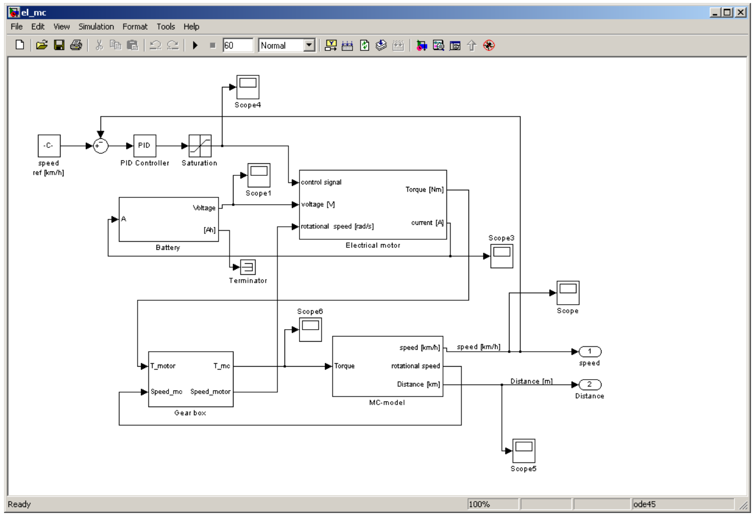
How does it work?

Objective function

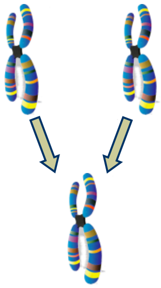
$$F(\mathbf{x}) = \alpha f_1(\mathbf{x}) + \beta f_2(\mathbf{x})$$

System characteristics/ simulation results

Objective function value



Simulation model



Optimization algorithm

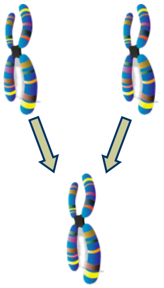
Design variables

How does it work?

Objective function value, f(**x**)

(constraint violation, gradients)

Optimization algorithm



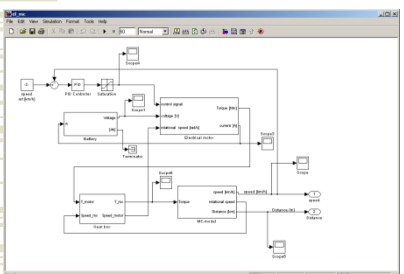
- **Complex**
- **GA**
- **fmincon**

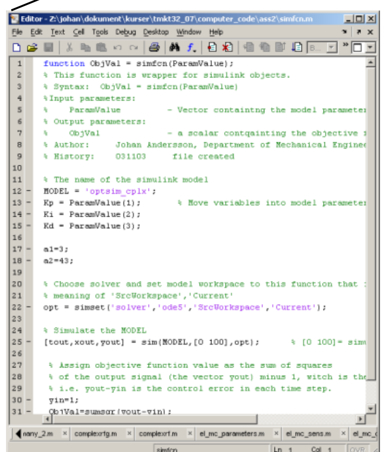
New design, i.e.
new values on
$\mathbf{x}=[x_1, x_2, \dots x_n]$

How does it work?

Objective function value, f(**x**)

Simulation model

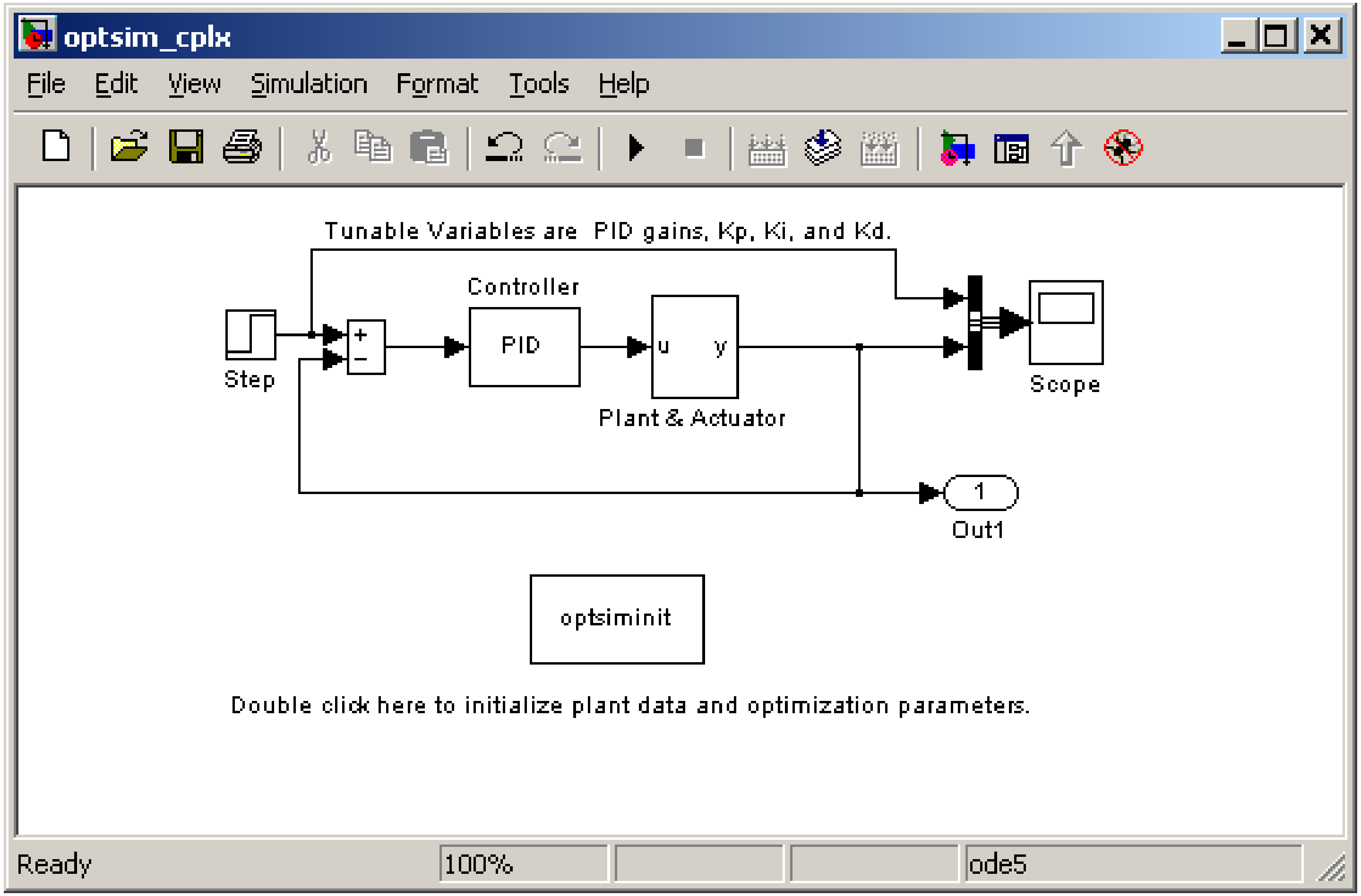




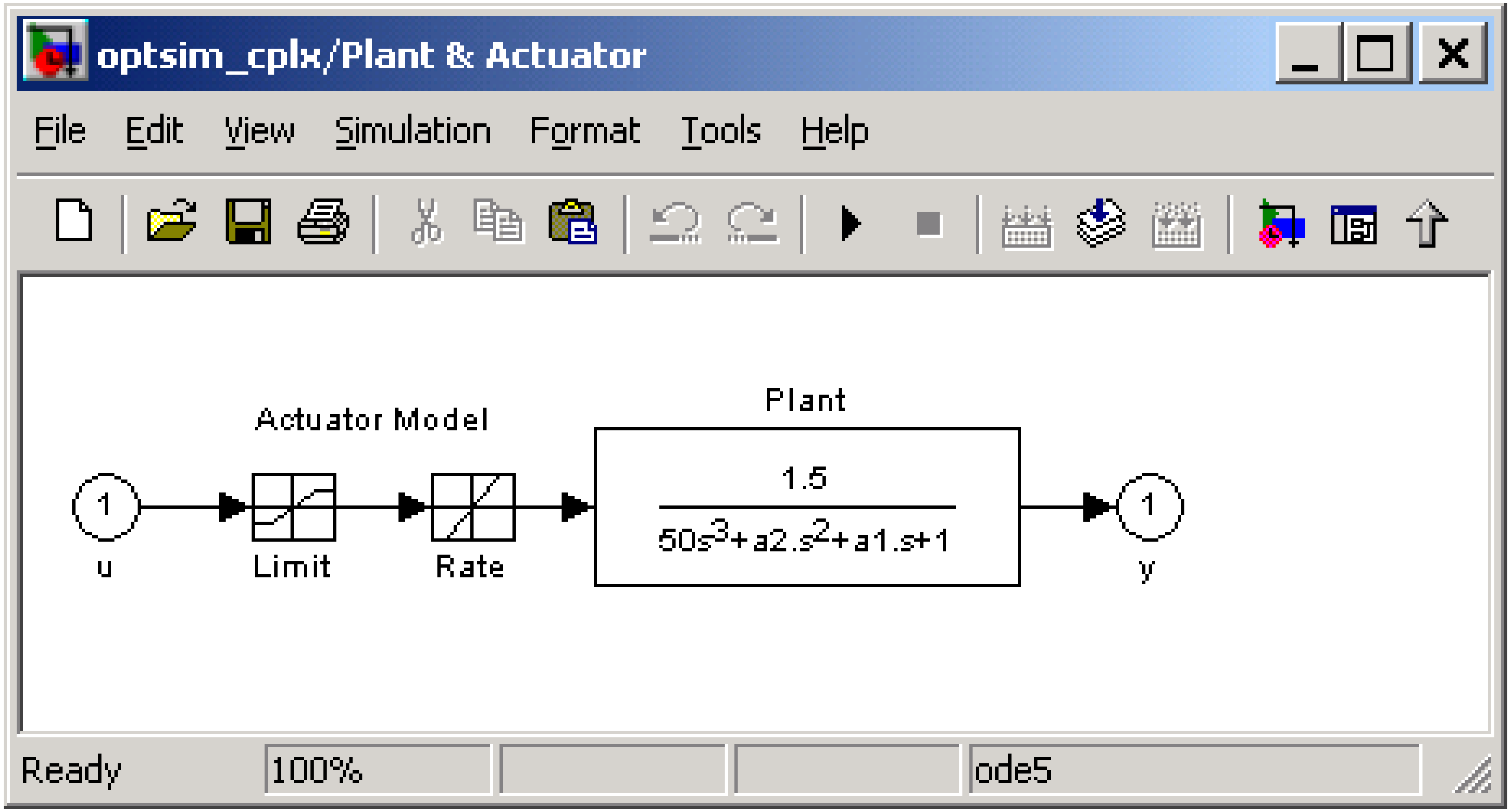
```
function ObjVal = simfcn(x);

% Set parameters for the simulation
MODEL = 'simulation model';
Param1 = x(1);
Param2 = x(2);

% Simulate the MODEL
[tout,xout,yout] = sim(MODEL,[0 100]);

% Calculate objectives
ObjVal=yout(1,end);
```

New design, $\mathbf{x}=[x_1, x_2, \ldots x_n]$

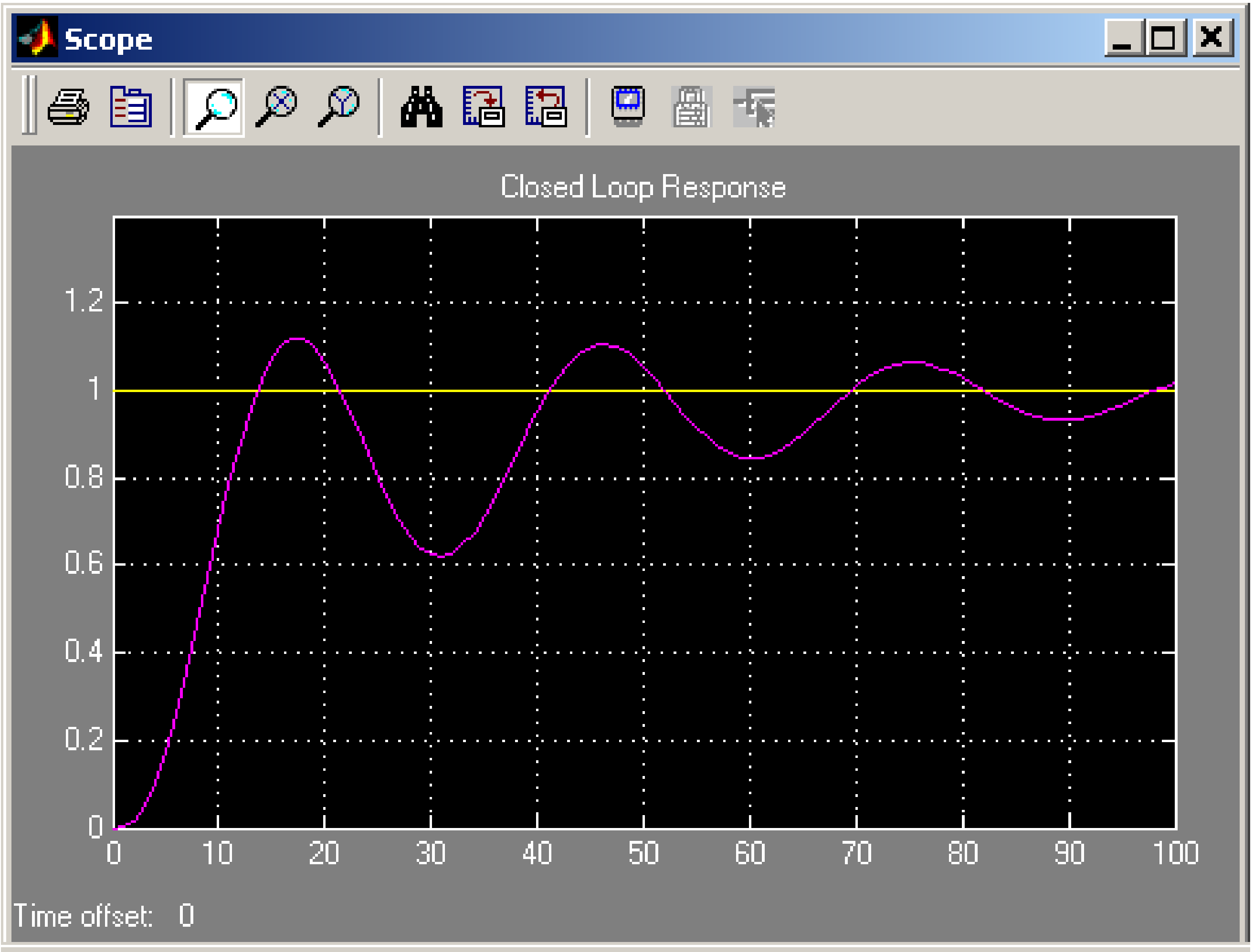
A MATLAB/Simulink model



Inside a Simulink block



Simulation results



Problem formulation

- Find the PID gains that minimize the control error.
- The control error is calculated as the sum of the squares of the control error for each time step.

$$\min_{x} f(x) = \sum_{t=t\_start}^{t\_stop} \left(y - y_{ref}\right)^2$$

$$s.t. \quad K_p, K_i, K_d \in [0,15]$$

Start a simulation from the MATLAB command window

Output signals

Name of Simulink model

Options to Simulink

Time

State variables

[tout,xout,yout]=sim('modelname',[tstart tstop], options)

Simulation output to workspace

MATLAB command

Simulation start time

Simulation stop time

Objective function for simulation

```
function ObjVal = simfcn(ParamValue);
% This function is a wrapper for simulink objects.
% Syntax:  ObjVal = simfcn(ParamValue)
%Input parameters:
%    ParamValue         - Vector containtng the model parameter values,
%                         PID-parameters Kp, Ki and Kd
% Output parameters:
%    ObjVal             - a scalar contqainting the objective function value.
% Author:     Johan Andersson, Department of Mechanical Engineering, Linköping University
% History:    031103     file created
%             050210     file updated

% The name of the simulink model
MODEL = 'optsim_cplx';

% Move variables into model parameter names
% These are the names used in the simulink model
Kp = ParamValue(1);
Ki = ParamValue(2);
Kd = ParamValue(3);

a1=3;       % plant parameters for the simulink model
a2=43;

% Choose solver and set model workspace to this function, i.e. when it says
% Kp, Ki and Kd in the simulink model use the parameters defined above.
opt = simset('solver','ode5','SrcWorkspace','Current');
% Simulate the MODEL
[tout,xout,yout] = sim(MODEL,[0 100],opt);      % [0 100]= simulation start and stop time,
                                                % opt is the options defined above.

 % Assign objective function value as the sum of the squares of the control
 % error. yout is the output signal for each timestep and the input signal is 1.
 ObjVal=sumsqr(yout-1);
```

MATLAB commands

MATLAB

File Edit View Web Window Help

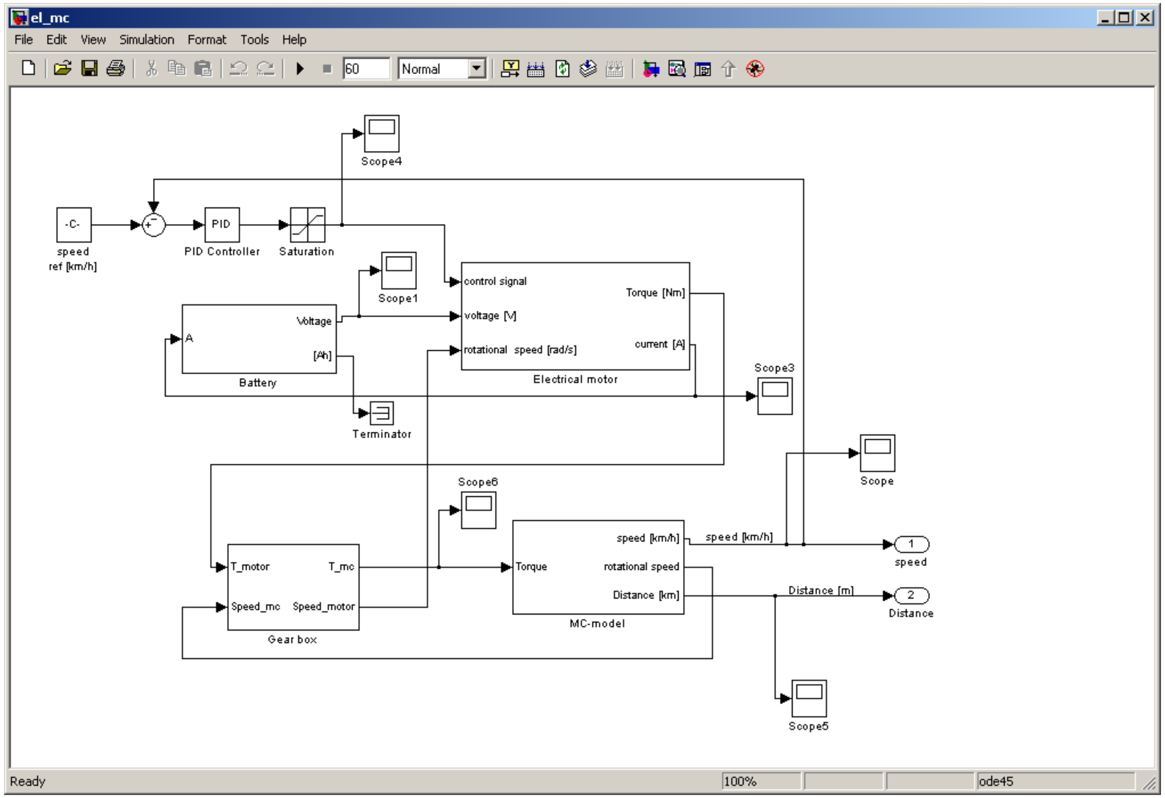
Current Directory: C:\MATLAB6p5\

```
>> simulink
Done initializing optsim.
>> whos
  Name      Size                   Bytes  Class

  Kd        1x1                        8  double array
  Ki        1x1                        8  double array
  Kp        1x1                        8  double array
  a1        1x1                        8  double array
  a2        1x1                        8  double array

Grand total is 5 elements using 40 bytes

>> [x,f]=complexrf('simfcn',[0 0 0],[15 15 15])

** Complex optimization started **

Number of evaluations
  257

** Optimization stopped **

Convergence in function values

Number of evaluations = 257.
Number of iterations = 88.
Minimum function value = 4.46955.
Maximum function value = 4.46959.
Best point found at:          min(x)     max(x)
x(1) =     2.993              2.985      2.994
x(2) =    0.1447             0.1443     0.1457
x(3) =     13.32              13.31      13.32
```

Start

Electrical motorcycle problem

Find the gear ratios and the shifting times so that the motorcycle gets as good properties as possible



Gear box parameters:
Two gear ratios and a shifting speed

- The first gear ratio (u_1) need to be larger than the ratio of the second gear (u_2).
- Create a parameterization that guarantees that, e.g.

$$u_1 = x_1 + x_2$$

$$u_2 = x_2$$

$$x_1, x_2 \geq 1$$

- The speed at which shifting from first to second gear is *v_shift* (rotational speed of the gearbox [rev/s])

Objective function high acceleration

As high velocity as possible after 5 seconds

$$\max F = V_5(\mathbf{x})$$

$$s.t.$$

$$x_i^l \leq x_i \leq x_i^u, i = 1\ldots3$$

2 gear ratios and 1 shifting time yields 3 optimization variables

Objective function high top speed

As high velocity as possible after 60 seconds

$$\max F = V_{60}(\mathbf{x})$$

$$s.t.$$

$$x_i^l \leq x_i \leq x_i^u, i = 1 \ldots 3$$

Objective function long range

As high range as possible traveling at 50 km/h

$$\max F = Range(\mathbf{x})$$

$$s.t.$$

$$V(\mathbf{x}) \geq 50\ km/h$$

$$x_i^l \leq x_i \leq x_i^u, i = 1...3$$

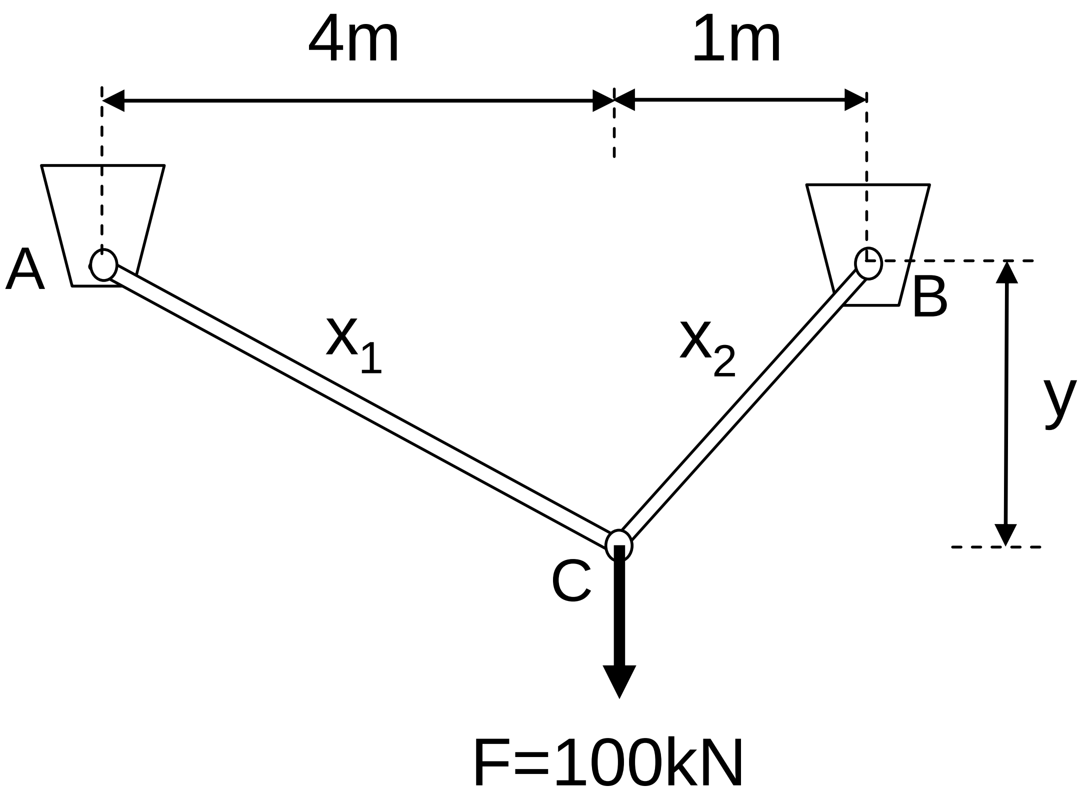
Compromise MC

Find a gearbox that is a good compromise between acceleration, top speed and range.

$$\max F = f\left(acc(\mathbf{x}), top(\mathbf{x}), Range(\mathbf{x})\right)$$
$$s.t.$$
$$x_i^l \le x_i \le x_i^u, i = 1...3$$

Two bar truss

Find the tradeoff between low weight and low stress



Objective function formulation

$$\min\left(f_1(\mathbf{x}), f_2(\mathbf{x})\right) = \left(Volume(\mathbf{x}), Stress(\mathbf{x})\right)$$

$$s.t.$$

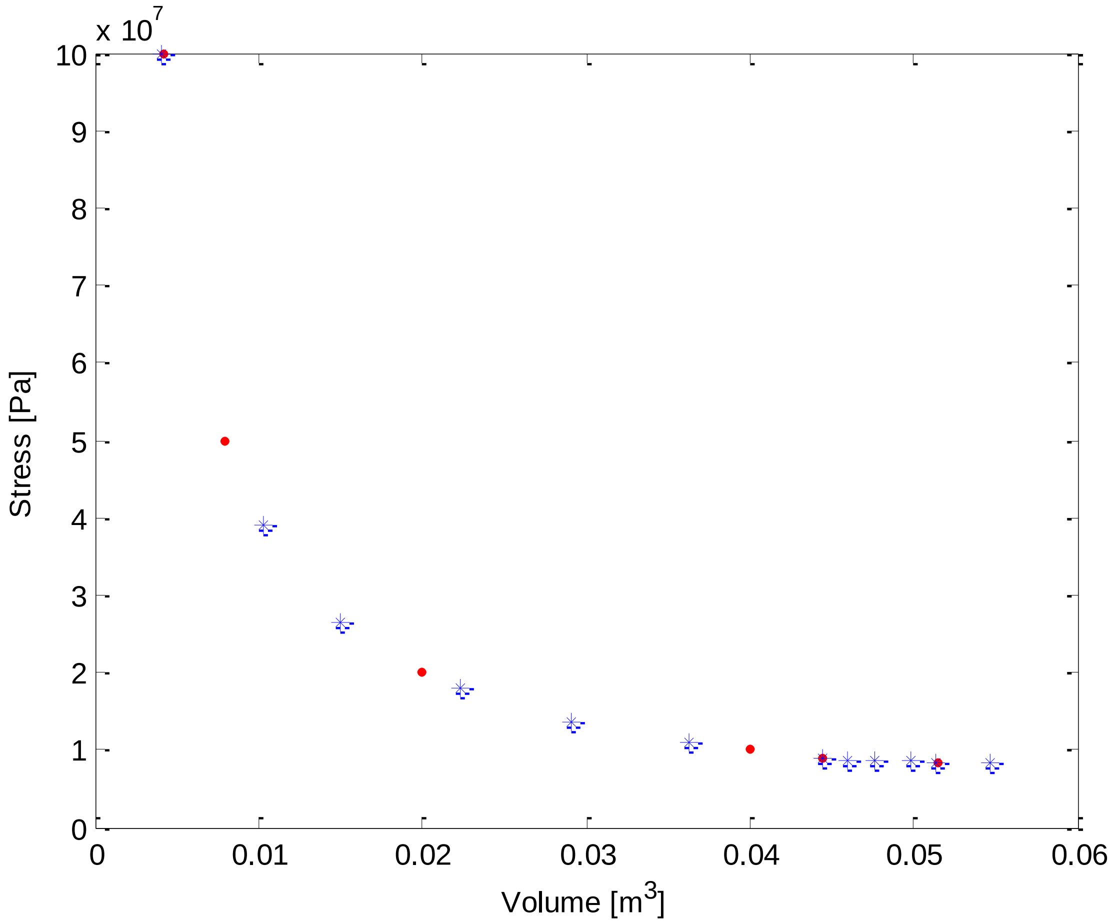
$$Stress \leq \sigma_{\max}$$

$$0.001 \leq x_i \leq 0.01, i = 1, 2$$

$$1 \leq x_3 \leq 3$$

2 cross-sectional areas and 1 length yields 3 optimization variables

Pareto optimal front



Industrial robot application

Simulation
results

Evaluation

$$f_1(\mathbf{x}),\ f_2(\mathbf{x}),\ldots,\ f_k(\mathbf{x})$$

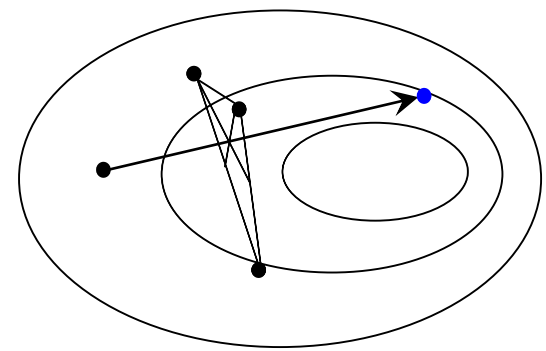
- Cycle time
- Lifetime
- Cost

Figure of
merit

ABB Simulation program

- Gearboxes
- Torque–speed curve

Optimization



Design
variables

Problem formulation

Find the gearboxes and the torque-speed curves that minimize the cost for a certain cycle-time and lifetime of the gearboxes.

$$\min \ f(\mathbf{x}) = Cost(\mathbf{x})$$

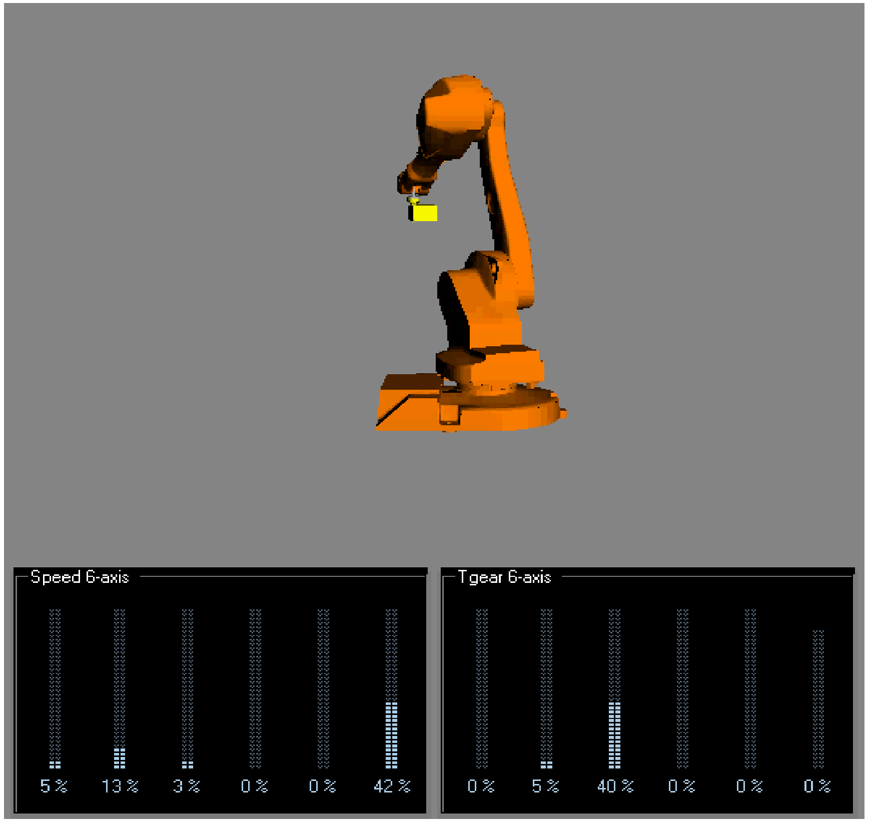
$$s.t.$$

$$\text{lifetime}(\mathbf{x}) \geq L10_0$$

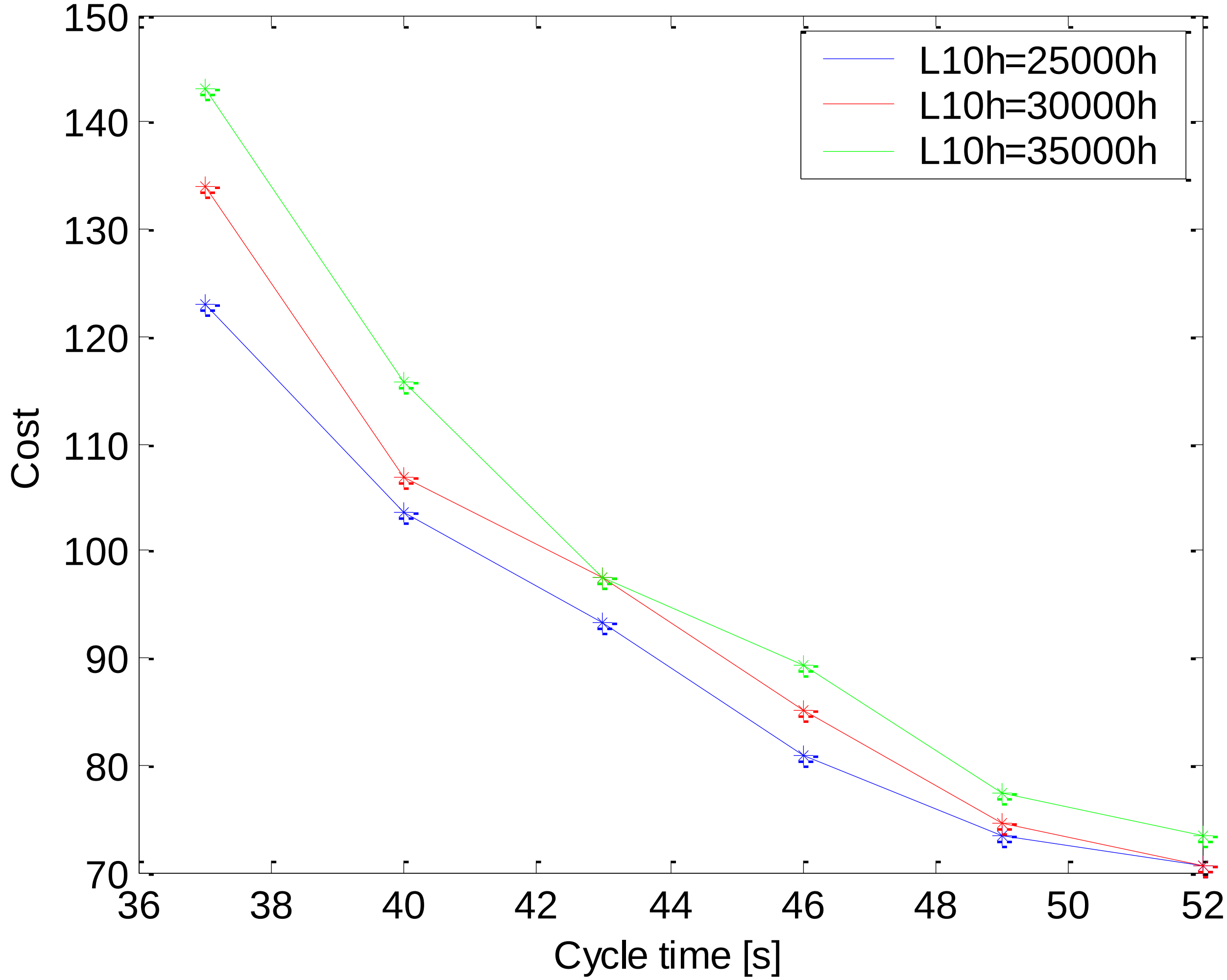
$$\text{cycltetime}(\mathbf{x}) \leq CT_0$$

$$x_i^l \leq x_i \leq x_i^u$$

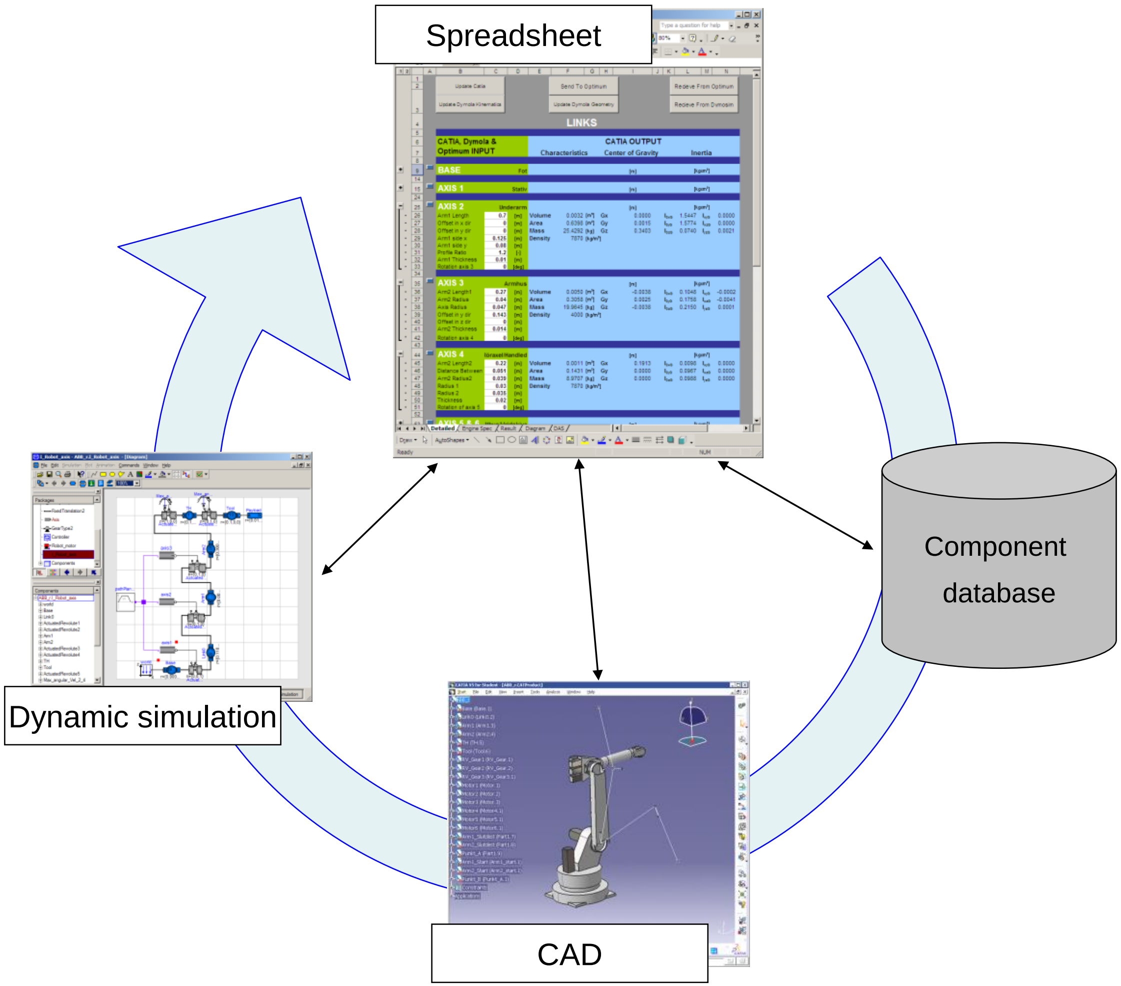
Robot in motion



Cost as a function of cycle-time and lifetime



Integrated framework

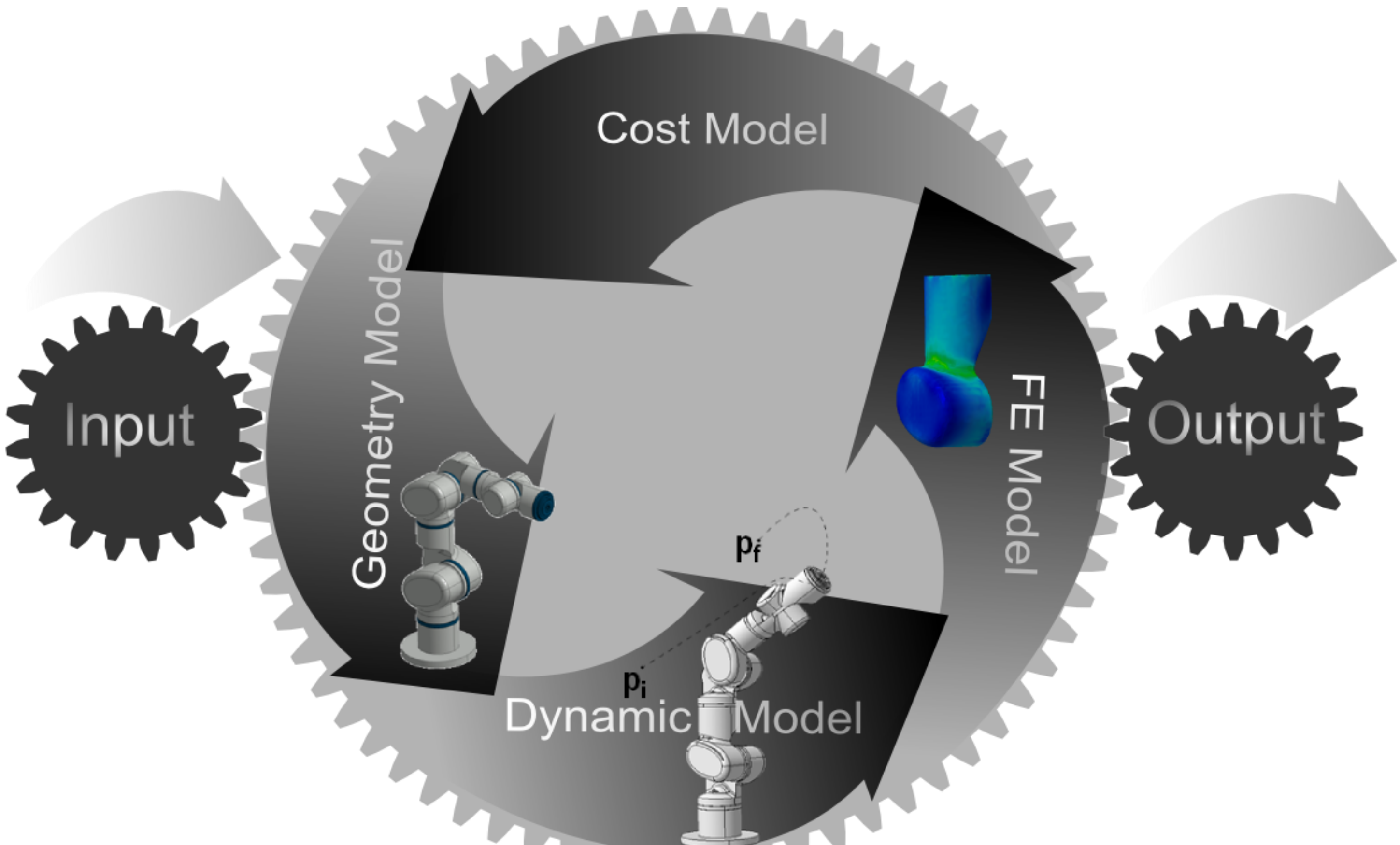


Practical MDO for Modular Robots

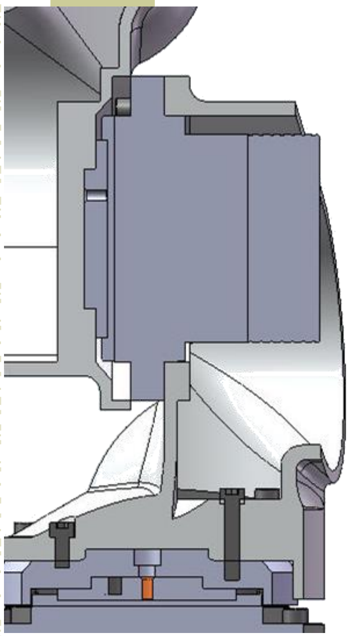
1. Which design tools and models are required?
2. How to generate the geometry parametrically?
3. How to achieve design integration?
4. How to achieve time efficient concept evaluations?

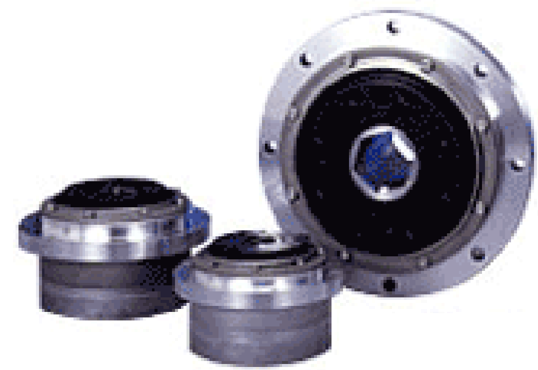
Design Automation to enable MDO

1. Which design tools and models are required?



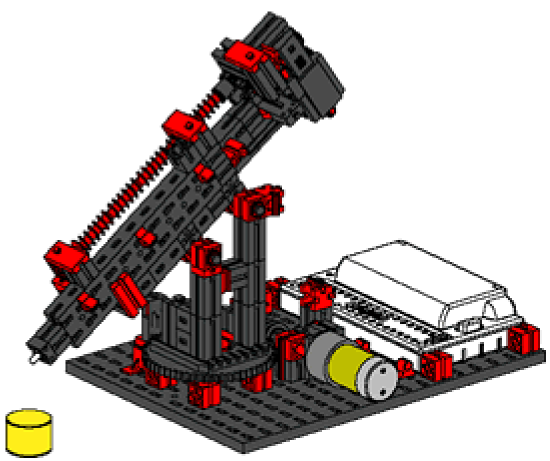
Creating the Geometry Model

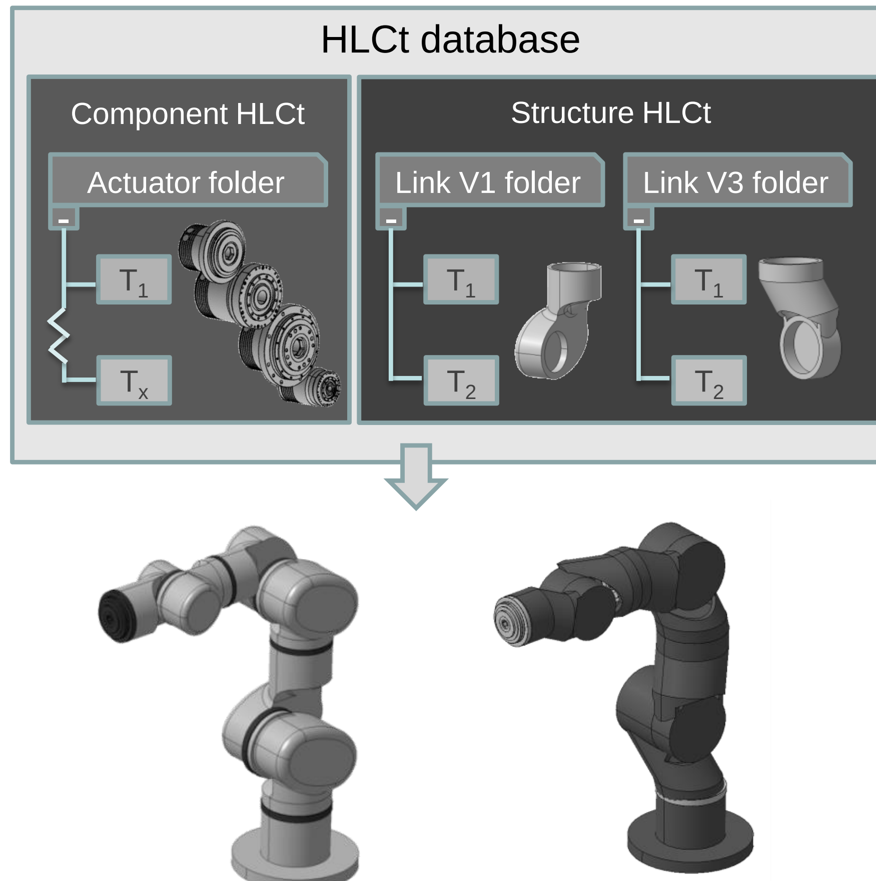




High Level CAD template -> Virtual LEGO

2. How to generate the geometry parametrically?

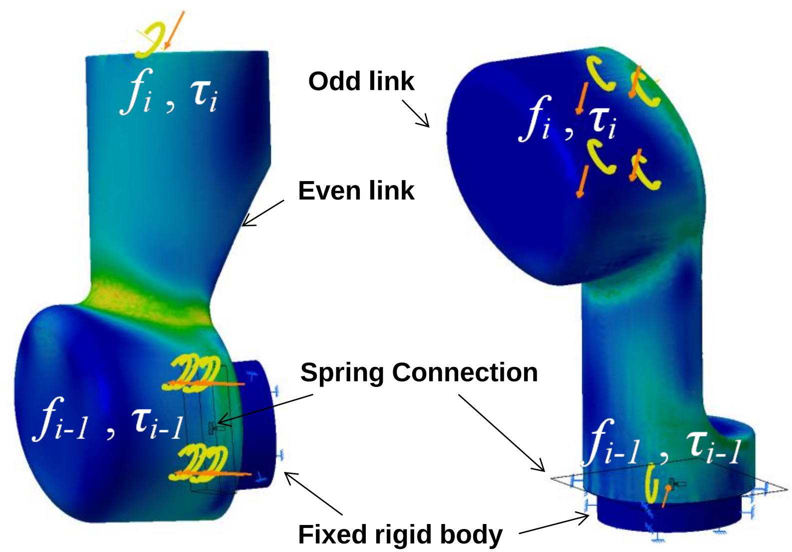




Integrated Design

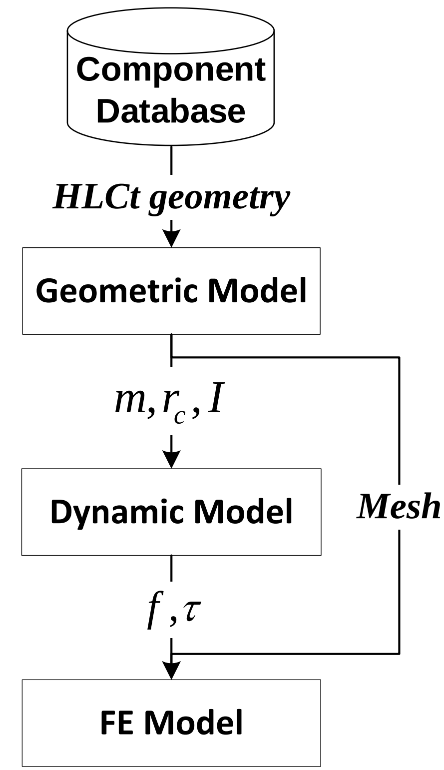
3. How to achieve design integration?

modeFRONTIER
the multi-objective optimization and design environment



HLCt enables MDO for Modular Robots

1. Which design tools and models are required?
2. How to generate the geometry parametrically?
3. How to achieve design integration?
4. How to achieve time efficient concept evaluations?



The Models are Expensive

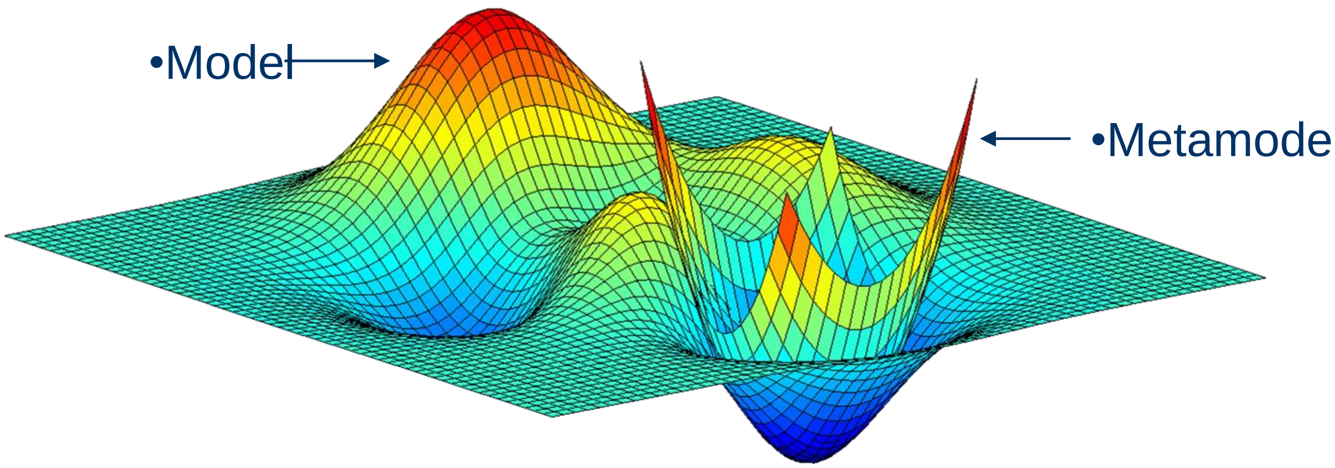
- Time needed for each evaluation is ~ **6 minutes**.
- **5 seconds** for the Dynamic model
- **20 seconds** for the CAD model
- **60 seconds** for FE analysis for each link
- Total optimization time > **1 month** for ~10 000 evaluations

4. How to achieve time efficient concept evaluations?

Use of Metamodels

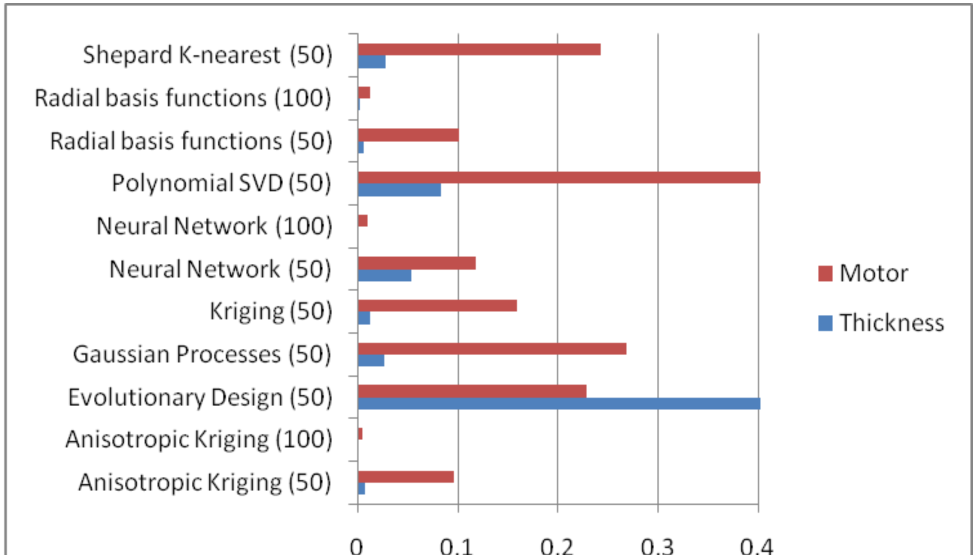
Metamodels

- Computationally efficient approximations of the output of a model



Metamodels

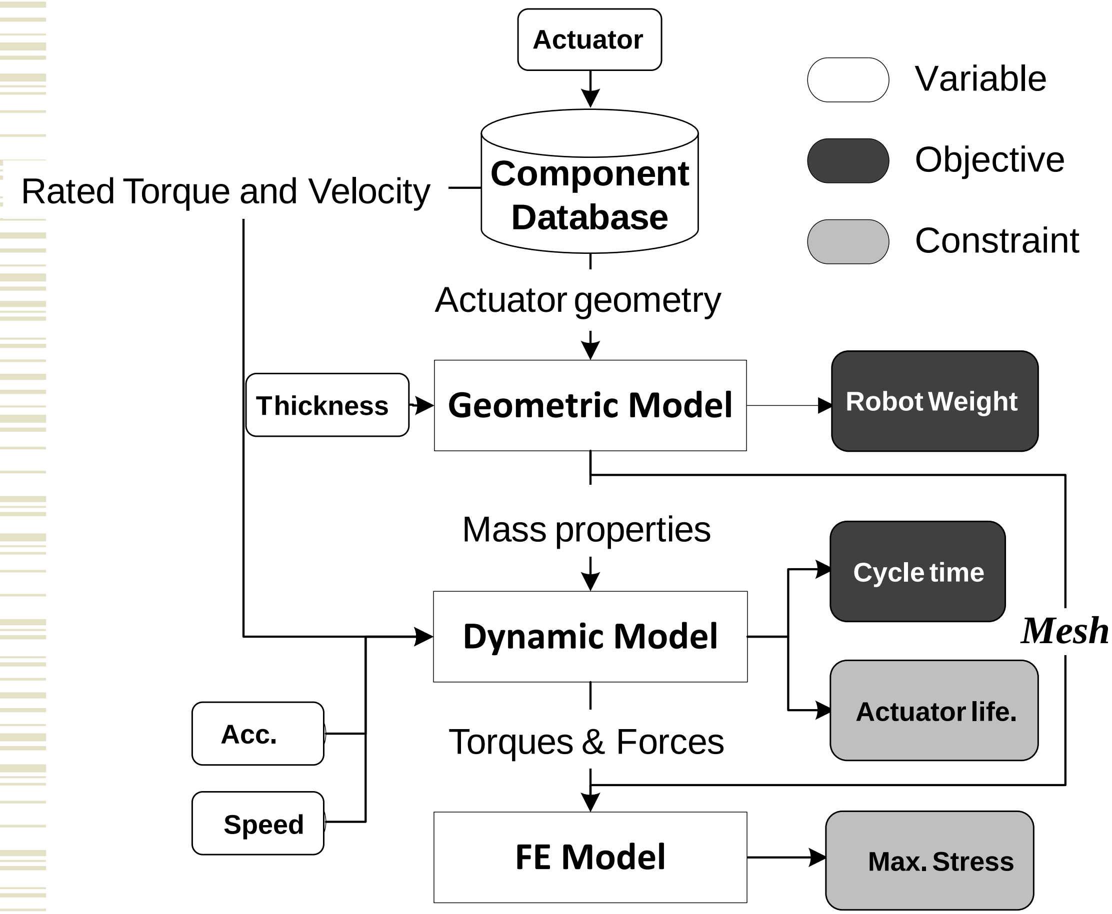
- Various metamodels were evaluated
- Geometry model: **NRSME of 0.5% (**anisotropic kriging,100 samples).
  3 input variables: *mixed continuous and discrete*
- FE model: **NRSME of 9% (**anisotropic kriging,1400 samples).
  14 input variables: *mixed continuous and discrete*

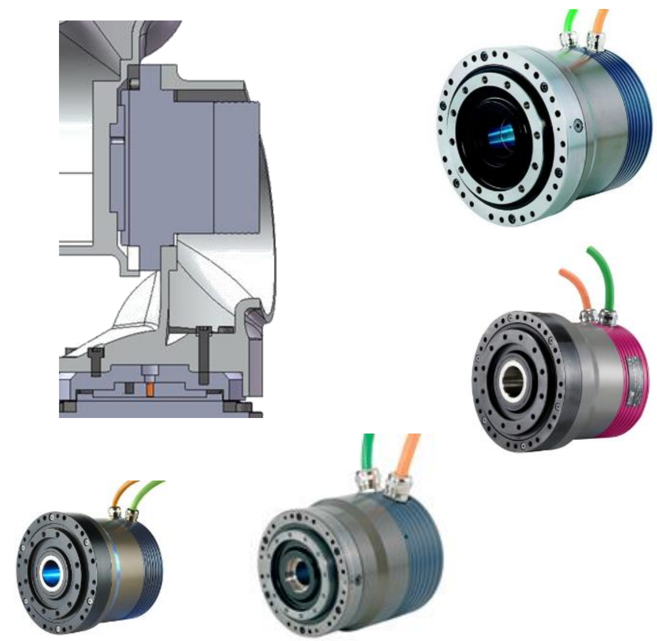


Model Speed with Metamodels

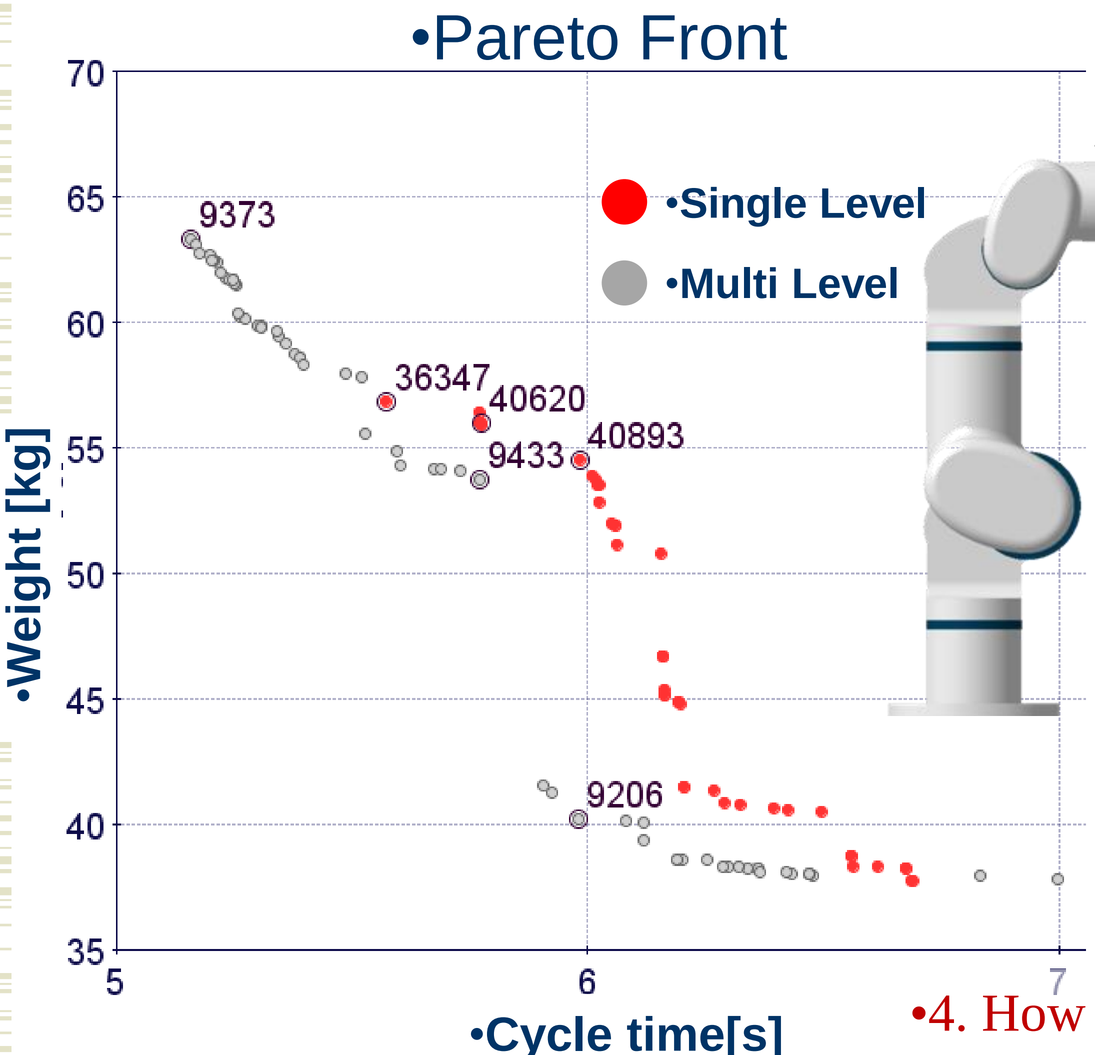
- **5 seconds** for the Dynamic model
- **5 seconds >>** for the CAD metamodels
- **5 seconds >>** for the FE metamodels

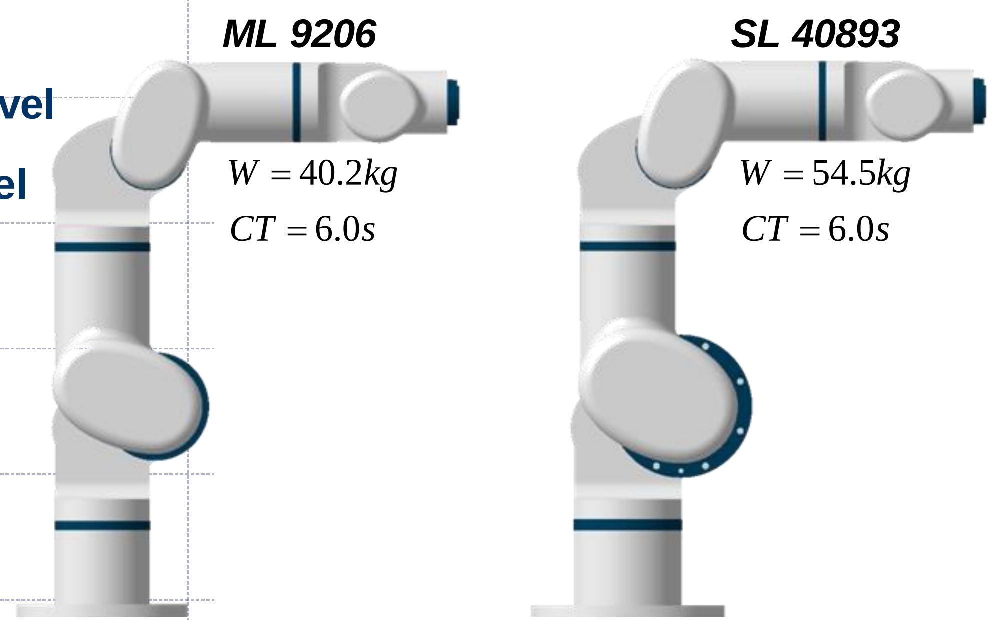
Multidisciplinary Optimization Framework





Single Robot Optimization



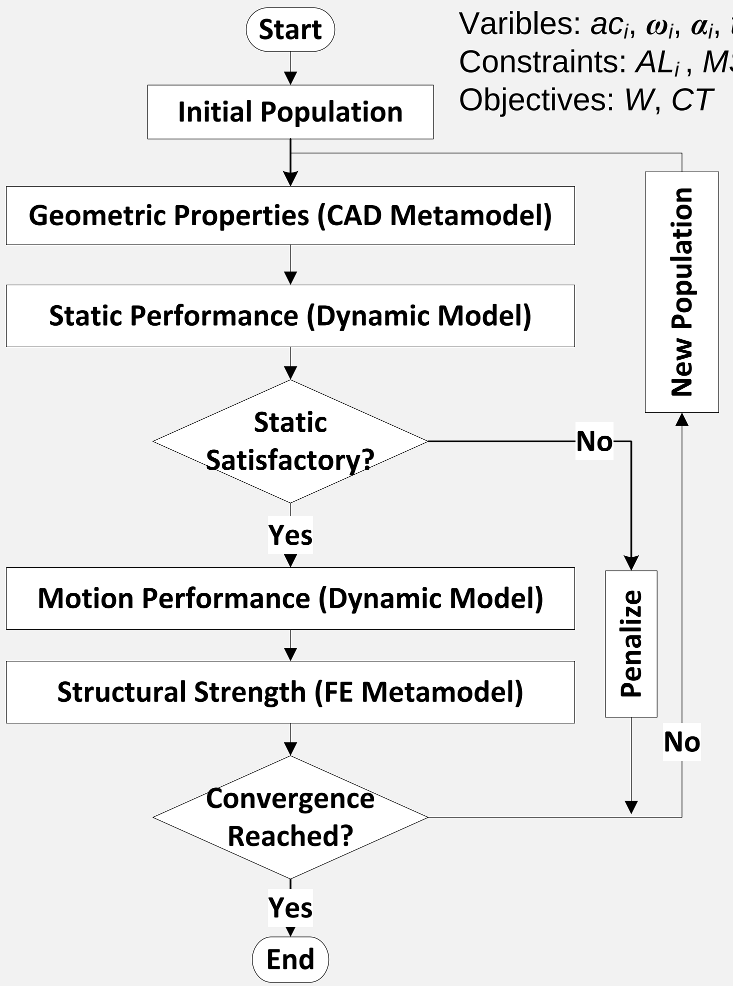


•4. How to achieve time efficient concept evaluations?

Johan Ölvander

Single Level Optimization

Varibles: ac_i, ω_i, α_i, t_n
Constraints: AL_i , MS_n
Objectives: W, CT



$$\min f_1(\mathbf{x}) = CT(\mathbf{x})$$

$$\min f_2(\mathbf{x}) = \lambda_1 \sum W_i(\mathbf{x})$$

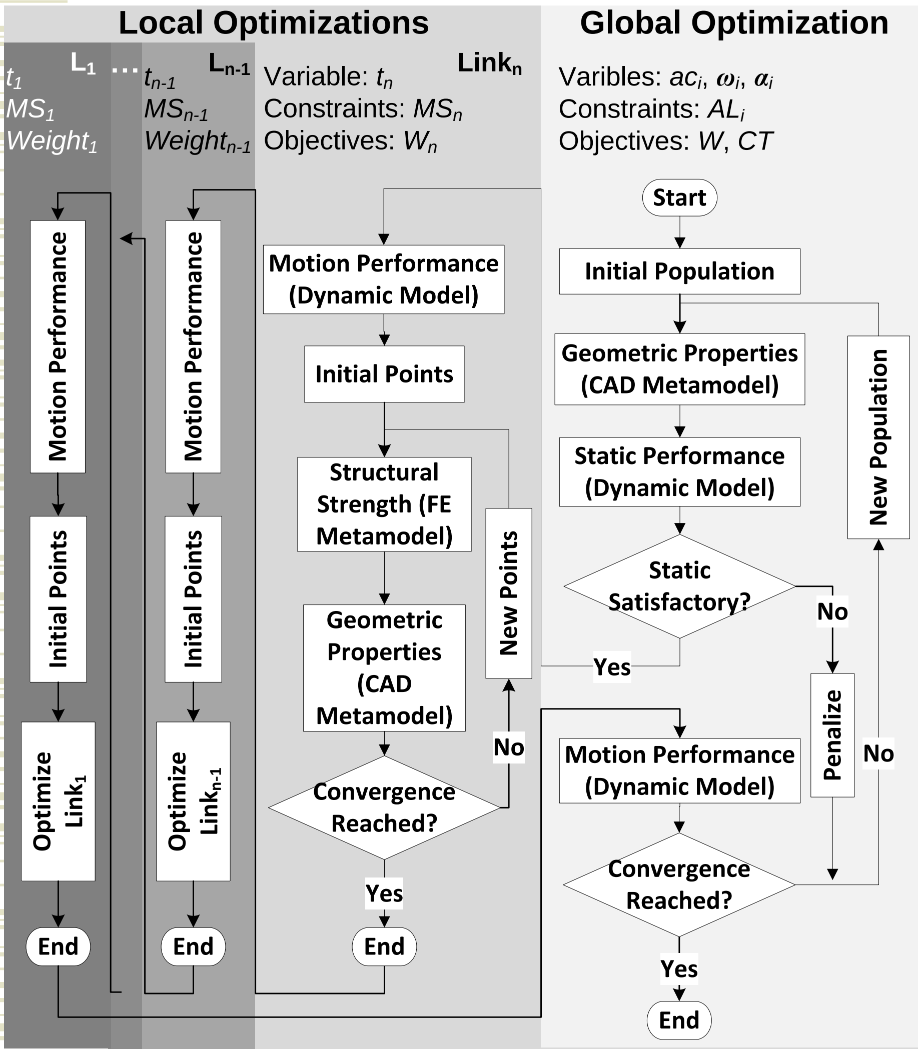
$$g_a(\mathbf{x}) = \frac{AL_a(\mathbf{x})}{AL_{req}} - 1 \leq 0, \quad a = 1:7$$

$$g_s(\mathbf{x}) = \frac{MS_i(\mathbf{x})}{MS_{req}} - 1 \leq 0, \quad s = 8:13$$

$$x_i \in \{1, 2, \ldots, n_{servo}\}, i = 1:7$$

$$x_j^{low} \leq x_j \leq x_j^{up}, j = 8:27$$

Multi Level Optimization



Global formulation

$$\min f_1(\mathbf{x}) = CT(\mathbf{x})$$

$$\min f_2(\mathbf{x}) = \sum f_y(\mathrm{t})$$

$$g_a(\mathbf{x}) = \frac{AL_a(\mathbf{x})}{AL_{req}} - 1 \le 0, \quad \mathbf{a} = 1:7$$

$$x_i \in \{1, 2, \ldots, n_{servo}\}, i = 1:7$$

$$x_j^{low} \le x_j \le x_j^{up}, j = 8:21$$

Local formulation

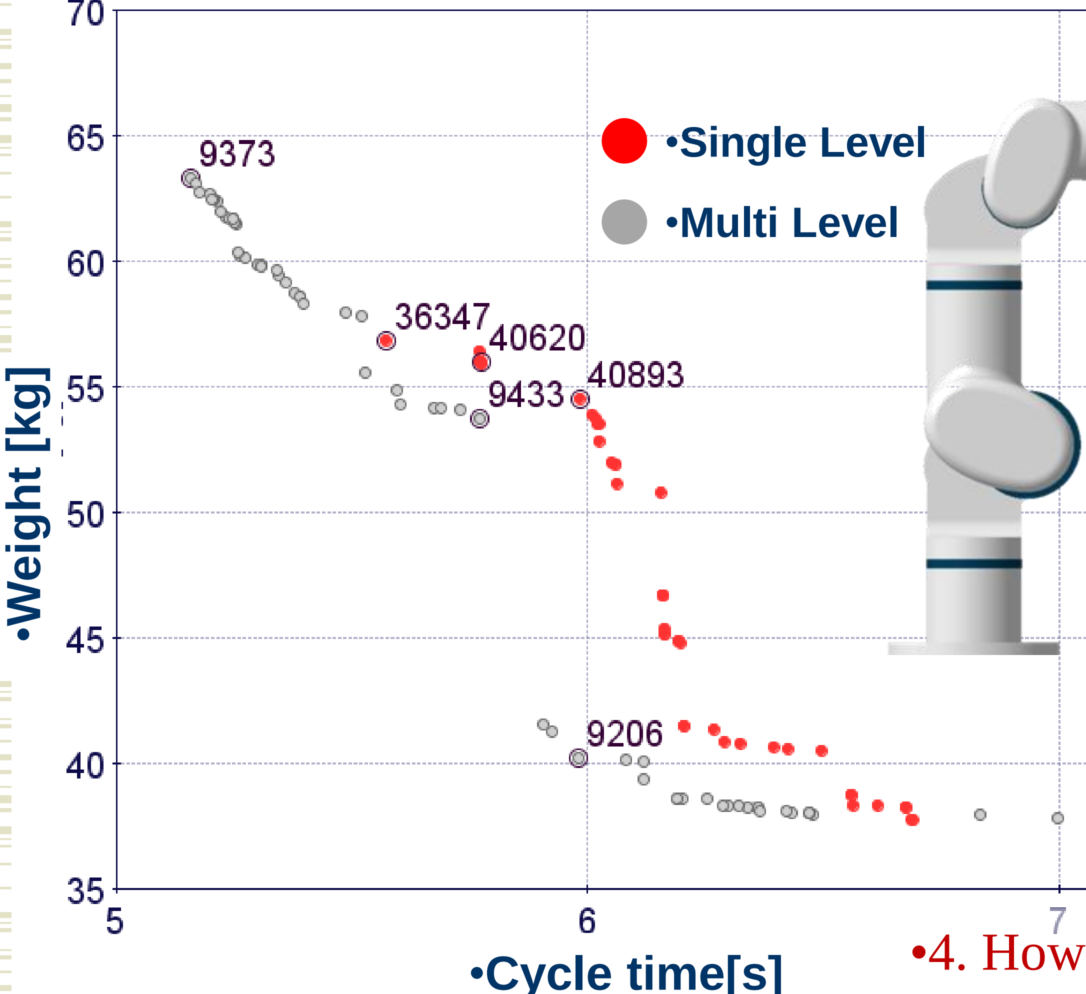
$$\min f_y(\mathrm{t}) = W_i\ (\mathrm{t}), \quad \mathrm{y} = 3:8$$

$$g_s(\mathrm{t}) = \frac{MS_j(\mathrm{t})}{MS_{req}} - 1 \le 0, \quad \mathrm{s} = 8:13$$

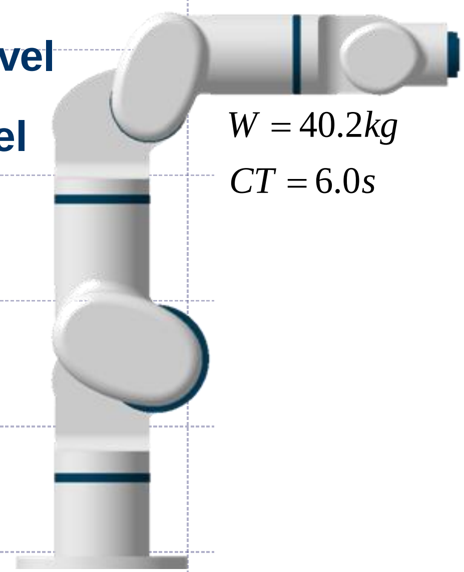
$$\mathrm{t}_j^{low} \le \mathrm{t}_j \le \mathrm{t}_j^{up}, j = 1:6$$

Single Robot Optimization

•Pareto Front



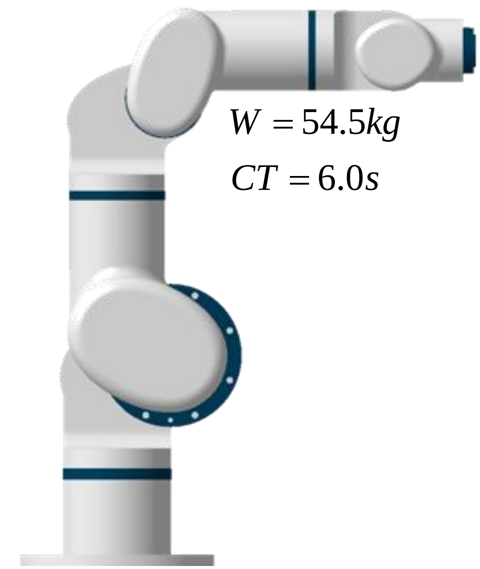
ML 9206



$W = 40.2kg$

$CT = 6.0s$

SL 40893

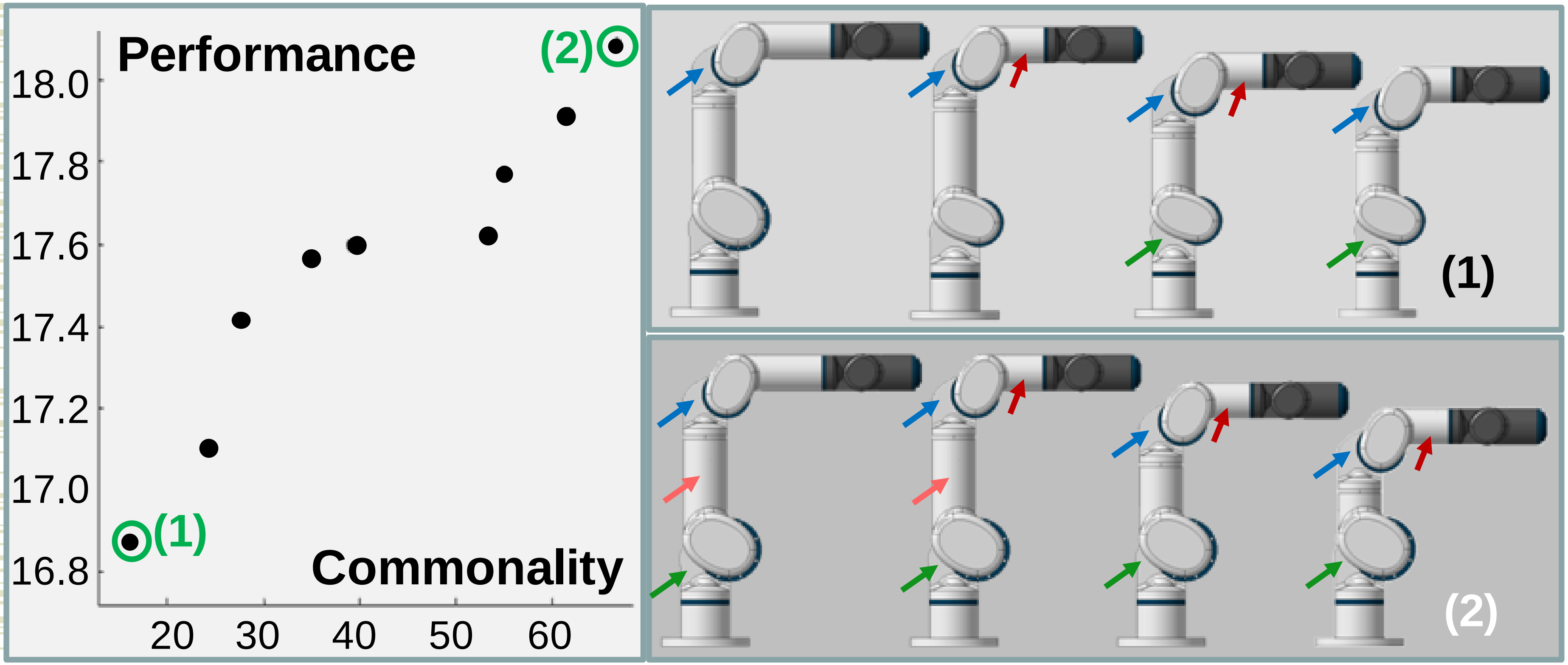


$W = 54.5kg$

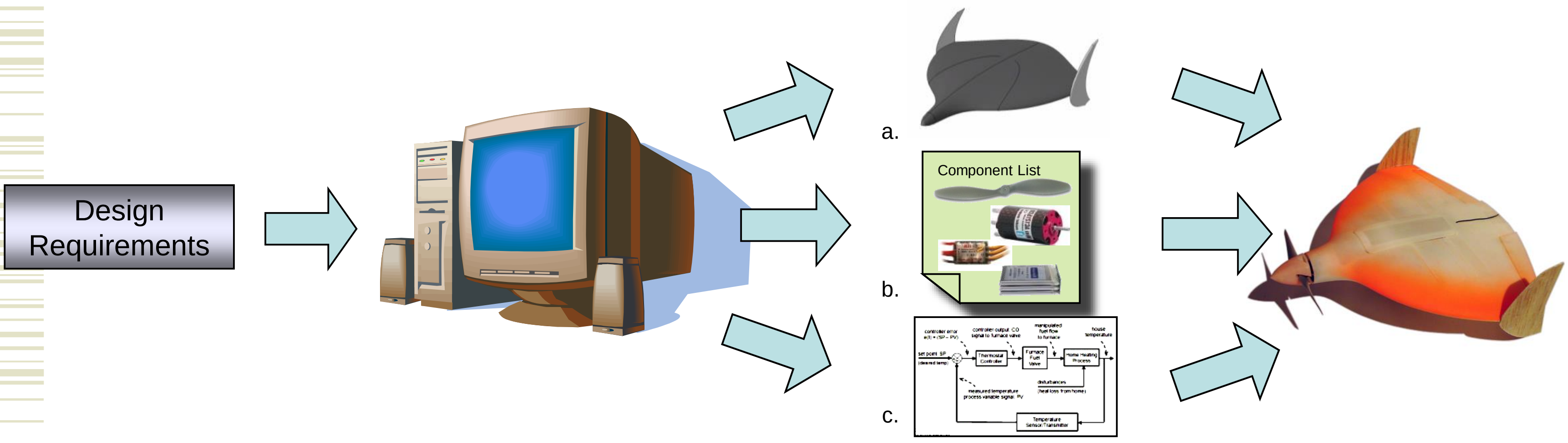
$CT = 6.0s$

•4. How to achieve time efficient concept evaluations?

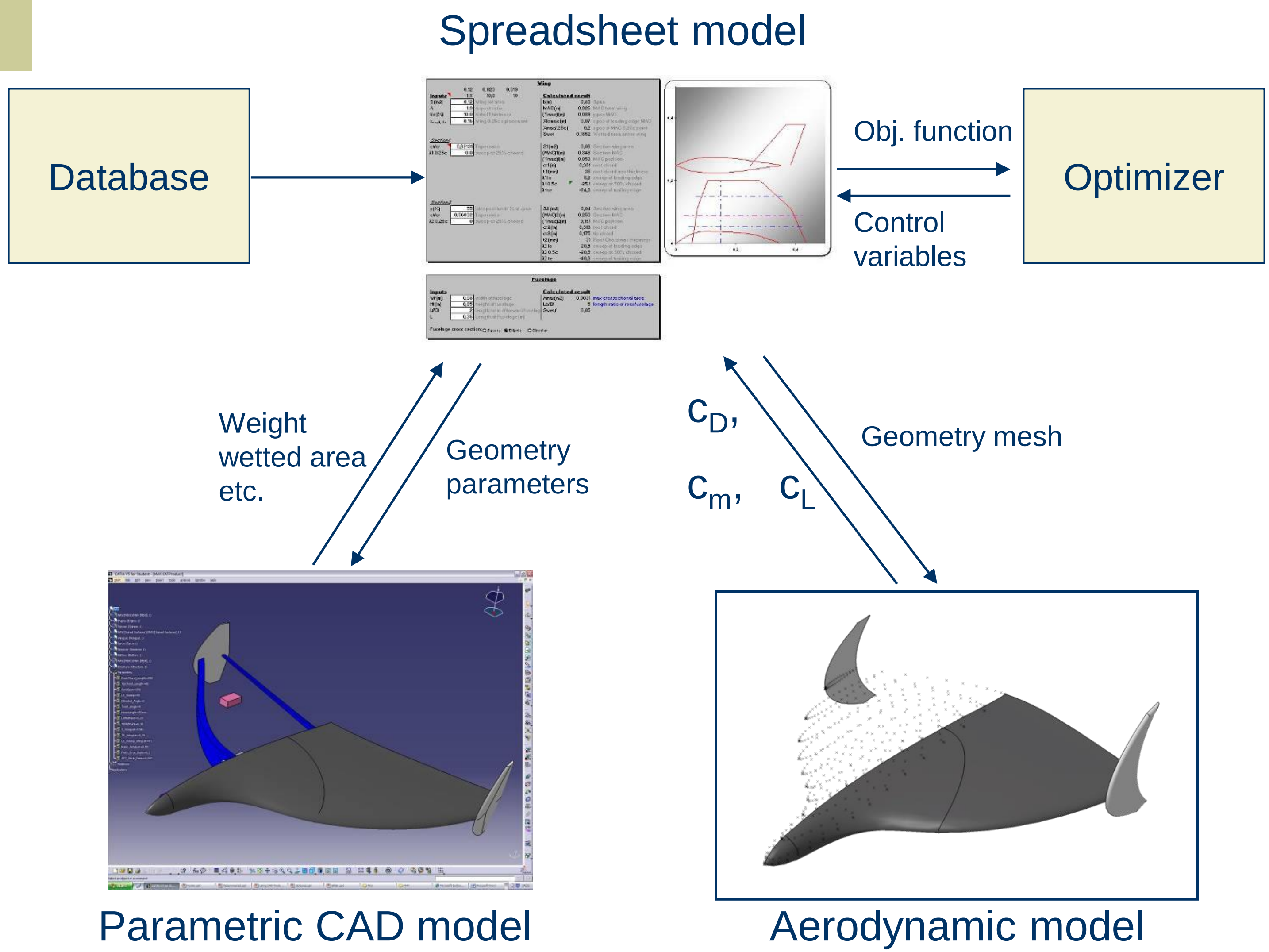
Industrial Robot Family Optimization



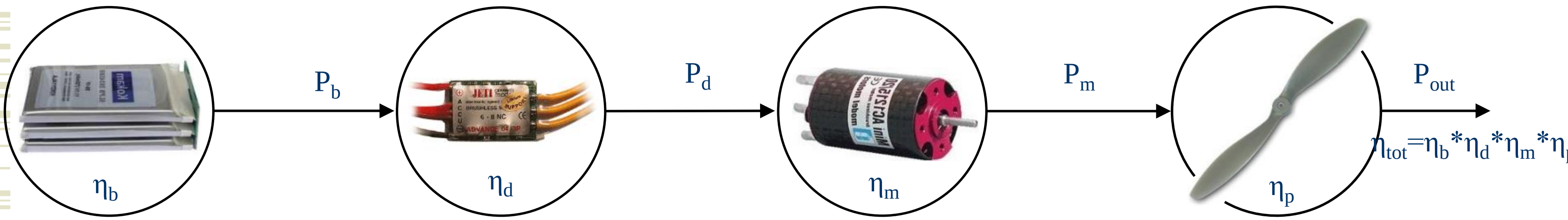
Rapid concept realization
MAV example



MAV - Design Framework



Modeling – Propulsion System



Battery

- Cell resistance
- Cell capacity
- Cell voltage
- Nr. of serial cells
- Nr. of parallel cells

Controller

- Resistive losses
- Losses depending on "throttle" position

$$L_{pwm} = 2\sum_{i=1}^{\infty}\left|{}^{i}V\right|^2 \frac{R_m}{R_m^2 + (2\pi i L_m / T)^2}$$

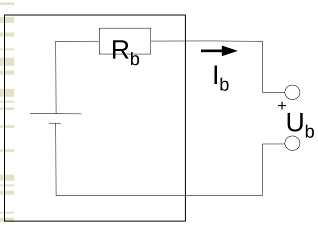
Classical electric motor model

- K_v, I_0, R_m

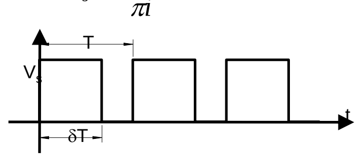
$$n = U_{emf} \cdot K_v \quad \text{(rpm)}$$
$$P_{in} = I_m \cdot U_m \quad \text{(W)}$$
$$P_{out} = I \cdot U_{emf} \quad \text{(W)}$$
$$P_{out} = \tau \cdot \omega \quad \text{(W)}$$

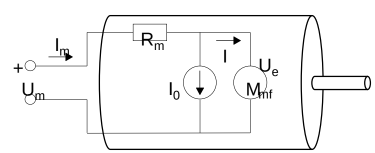
Blade element method

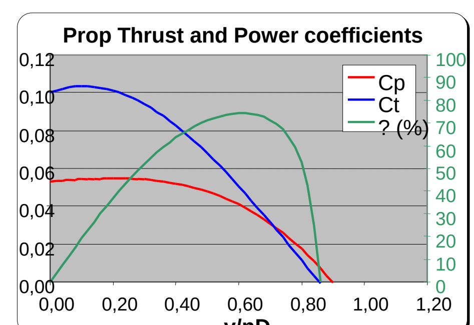
Performance characterized by Thrust and Power coefficients as function of advance ratio v/nD



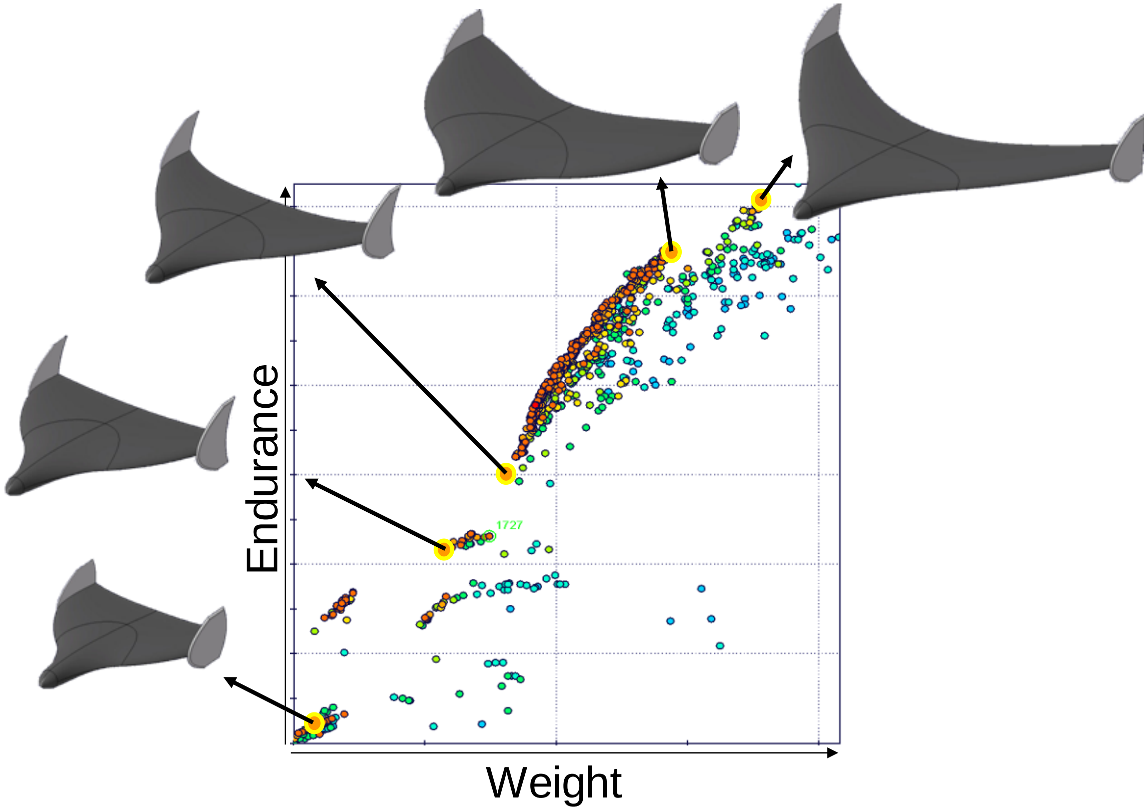
$${}^{i}V = V_s \frac{\sin(\delta\pi i)}{\pi i}$$







Multi-objective Optimization



The graph shows the trade of between the weight of the MAV and the endurance. The longer missions you want to fly, the heavier the MAV needs to be.

Closing the Loop - MAV Prototyping