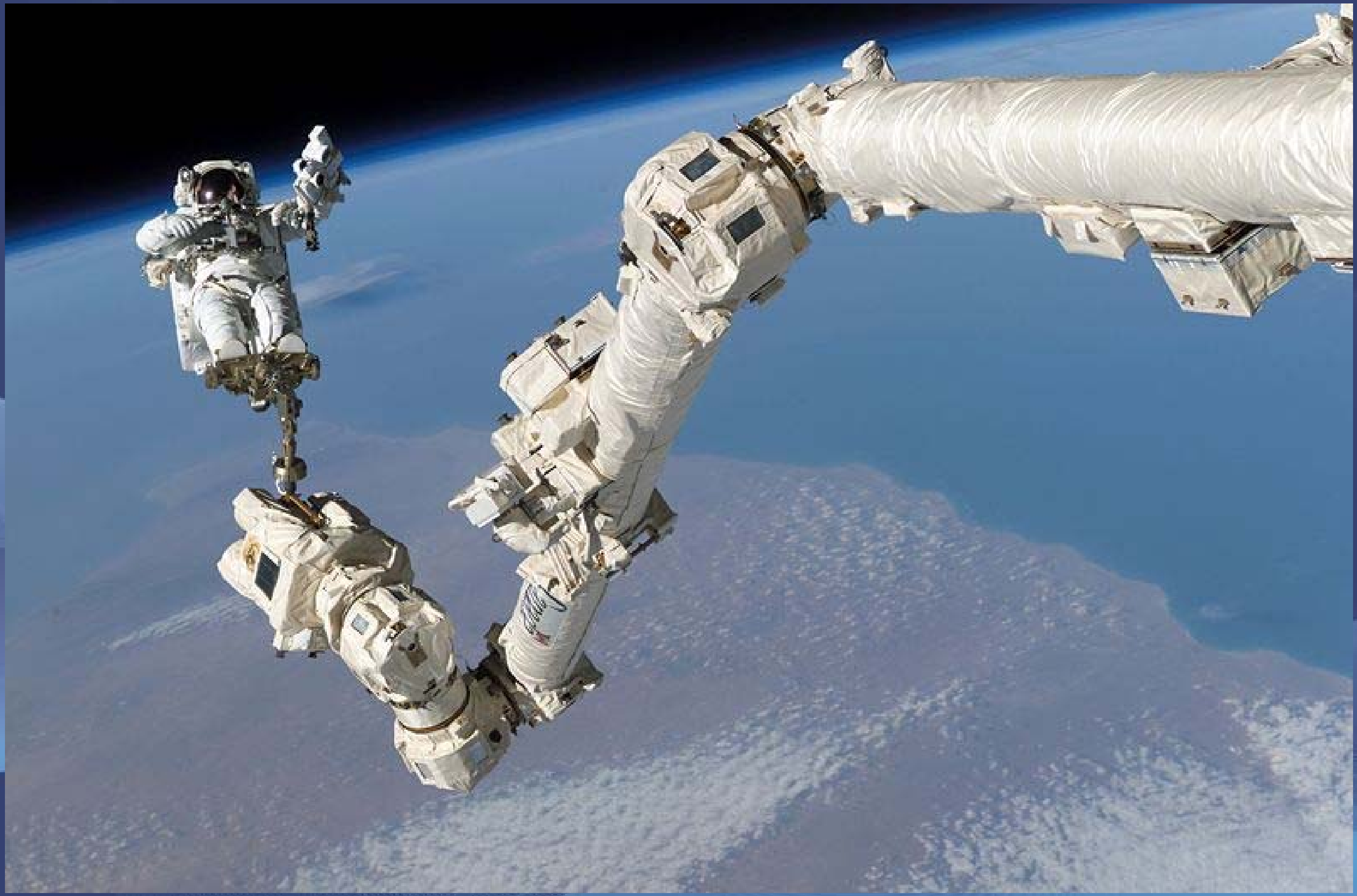
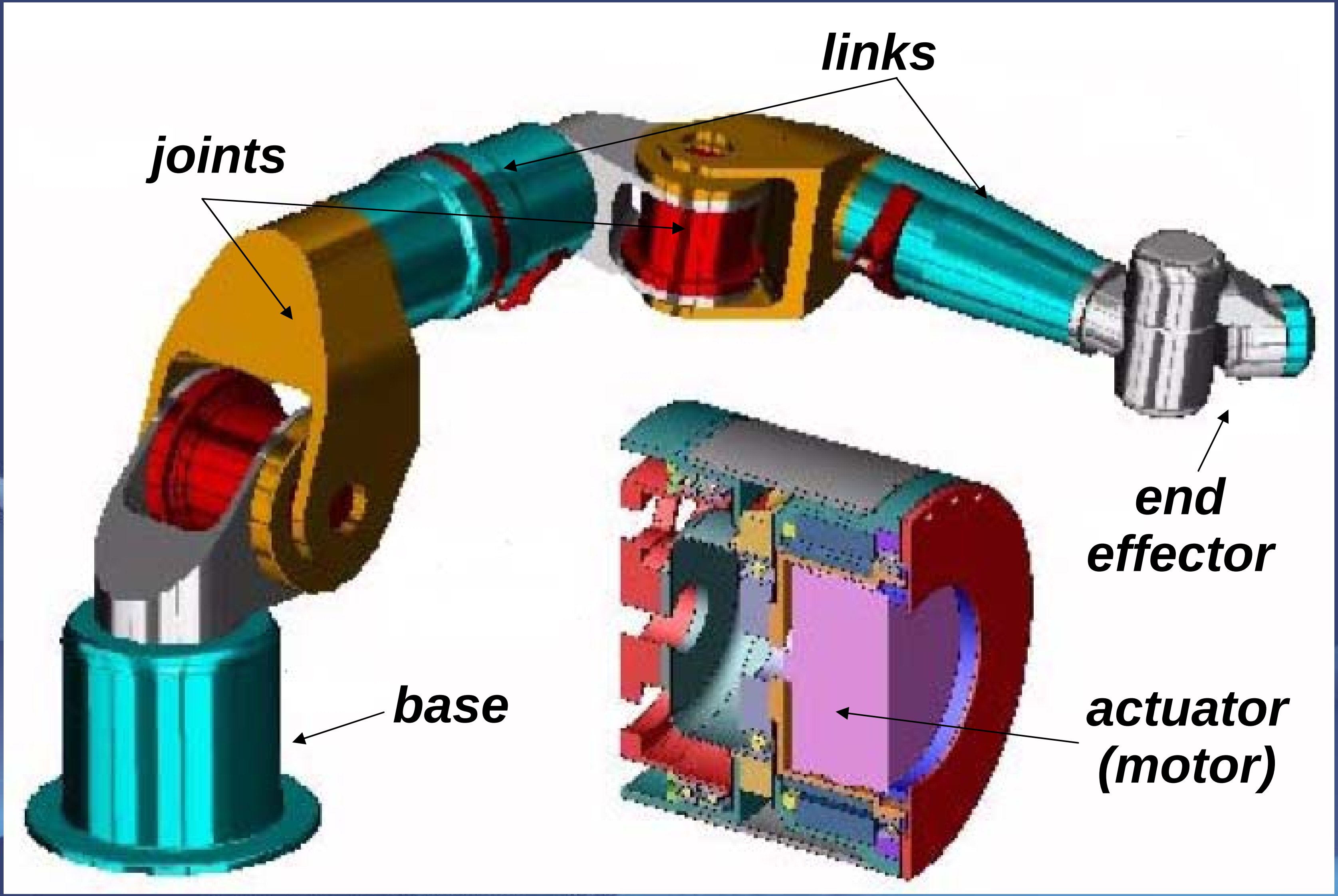
Multiobjective Optimal Trajectory Planning of Space Robot Using Particle Swarm Optimization



Robotics: definitions



Robotics: definitions

- Modeling
  - ✓ Kinematic, static and dynamic description
  - ✓ Sensor and actuators
- Control
  - ✓ Trajectory planning
  - ✓ Motion control
  - ✓ Hardware & Software

Robotics: definitions

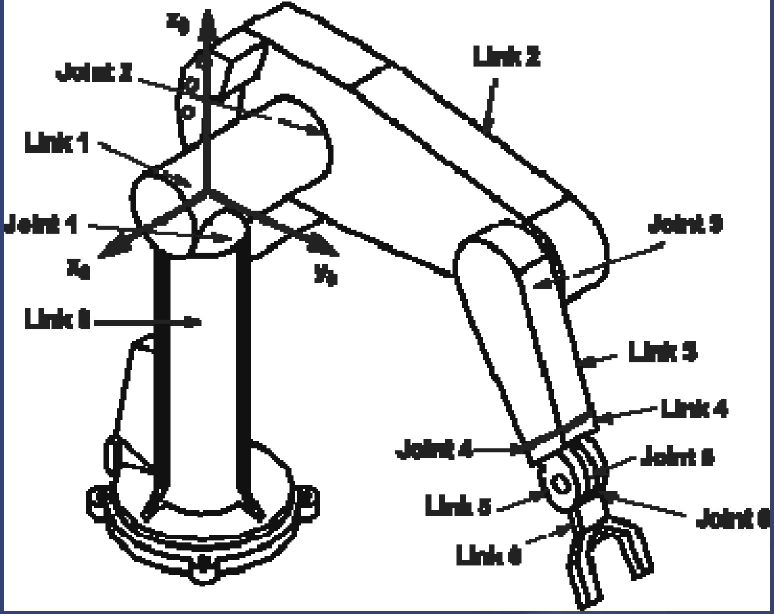
- Kinematics

$\underline{p} = [x, y, z, \alpha, \beta, \gamma]$

$\underline{\theta} = [\theta_1, \theta_2 \ldots \theta_n]$

$\underline{p} = \underline{f}(\underline{\theta})$ (forward kin.)

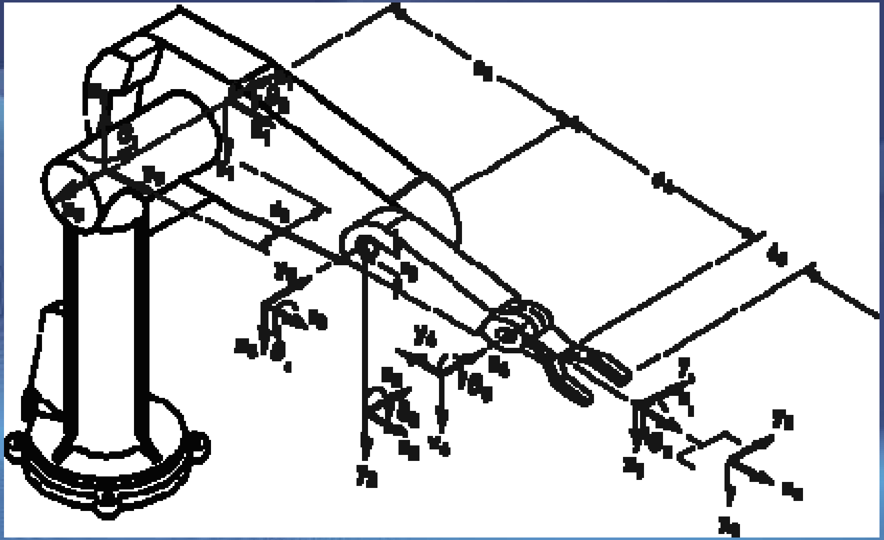
$\underline{\theta} = \underline{f}^{-1}(\underline{p})$ (inverse kin.)



- Differential Kinematics

$[v, \omega] = J(\underline{\theta})\underline{\theta}'$ (forward)

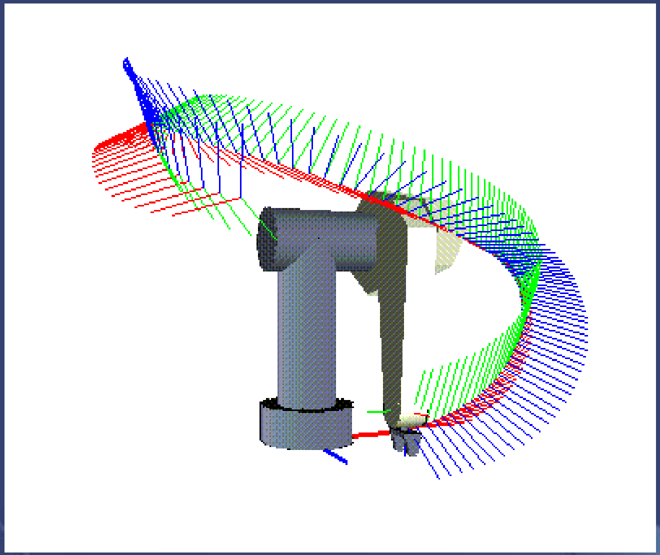
$\underline{\theta}' = J^{-1}(\underline{\theta})\,[v, \omega]$ (inverse)



- Dynamics

$\tau = H(\underline{\theta})\underline{\theta}'' + \underline{c}(\underline{\theta}, \underline{\theta}', \underline{F}_{ex})$

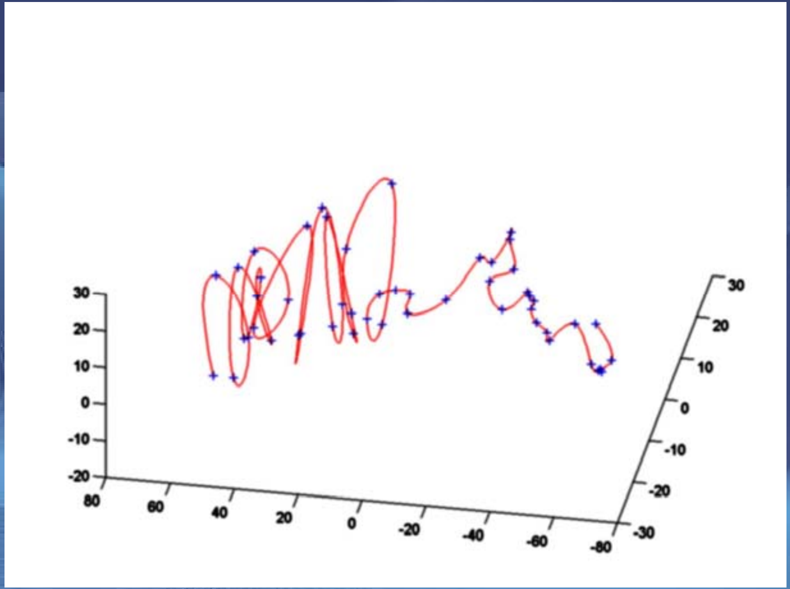
Trajectory planning



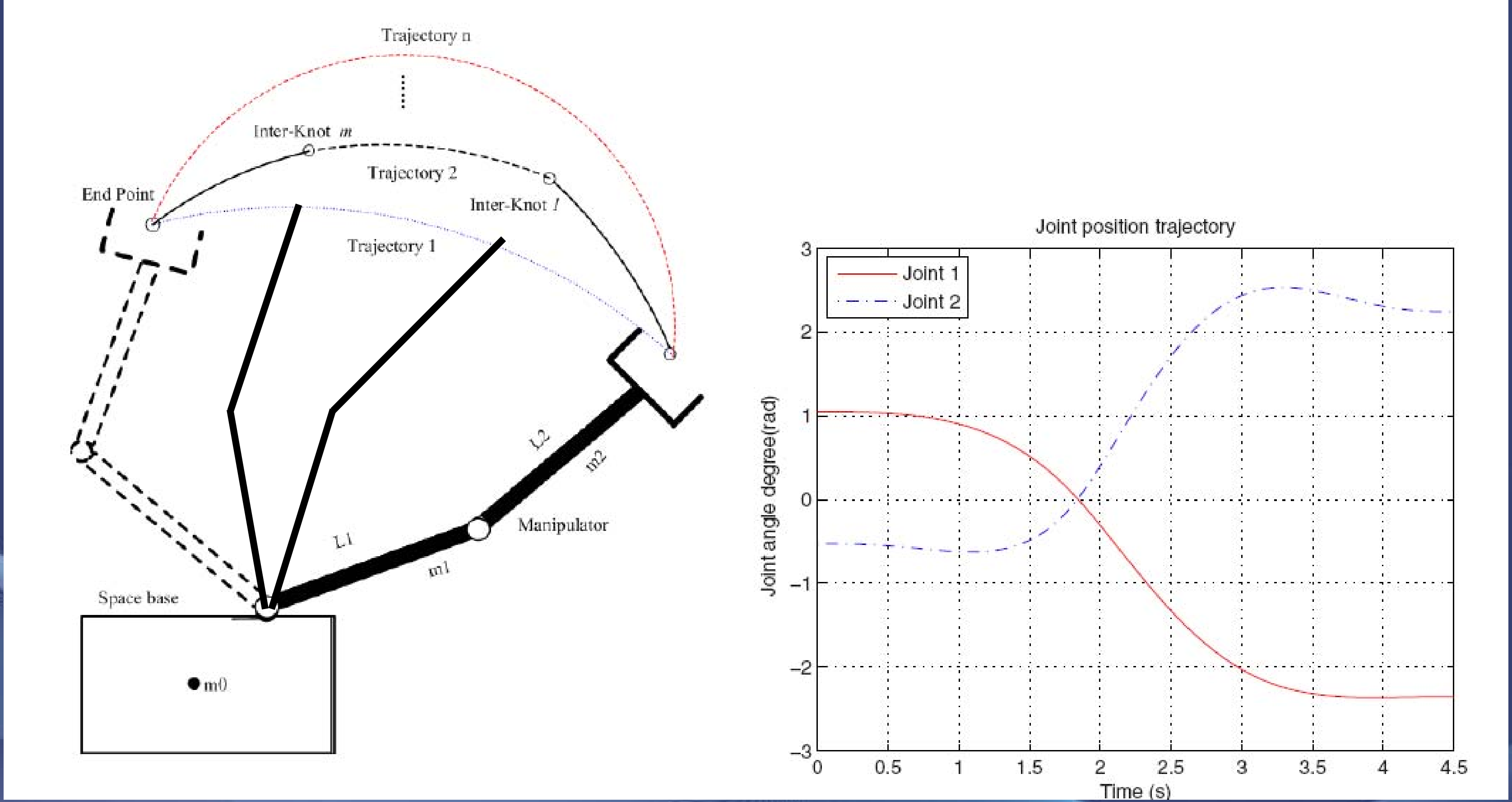
Given a specific task $\underline{p}(t)$, evaluate $\underline{\theta}(t)$, $\underline{\theta}'(t)$, $\underline{\theta}''(t)$ so that:

- the end effector follows the desired trajectory
- the trajectory is smooth (without discontinuities)

- geometric, velocity, acceleration, and torque constraints are satisfied
- the trajectory is OPTIMA according to some criteria



Trajectory planning



- Point-to-point problem: define inter-knot points and interpolate (linear interpolation, spline, etc.)
- Motion control: define the torques to be applied

Trajectory planning

Different conflicting criteria:

- minimize trajectory time
- minimize mechanical energy of actuators
- minimize fuel consumption
- minimize disturbances
- etc.

Disturbance to the space base

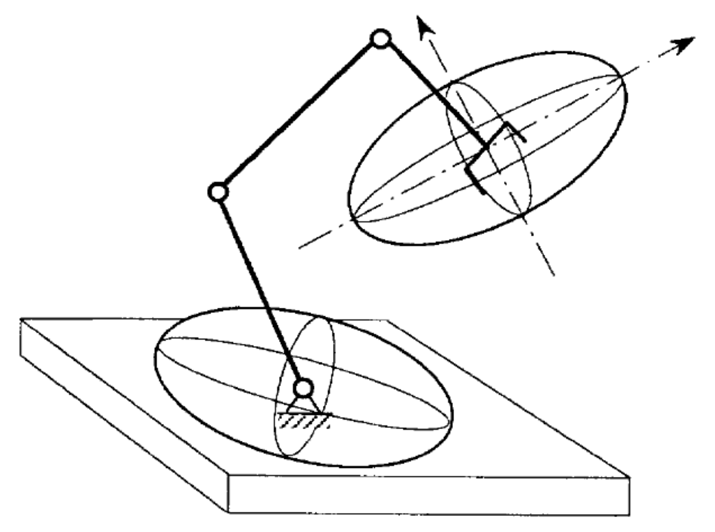
Free-floating environment

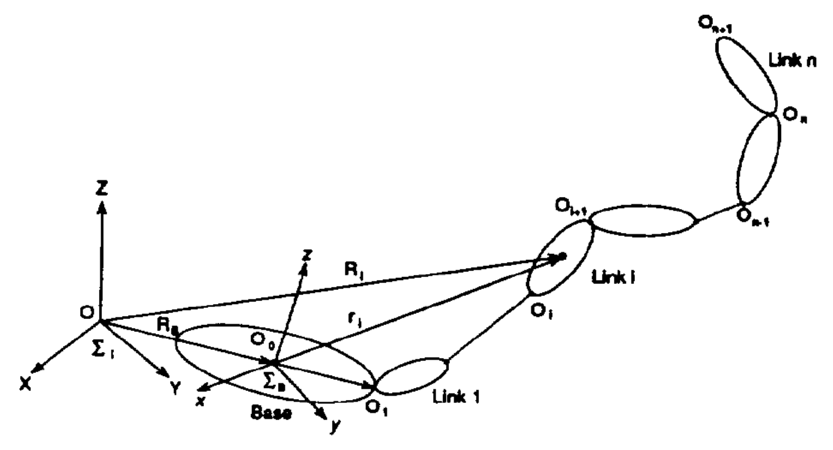
Mutual disturbance between base and end-effector:

$$F_B = \mathbf{N}^{-1}F_E \leftrightarrow F_E = \mathbf{N}F_B$$

N (dynamic coupling matrix) *"is a function of the robot configuration* [$\underline{\theta}$]*, the geometric and inertia parameters of the robot and the spacecraft, and the position of the robot base with respect to the spacecraft"* [2]

The shorter the motion time is, the greater the disturbance to the base will be





Multiobjective problem

$$\textbf{min}\ \left\{ \begin{array}{l} \Gamma = \dfrac{1}{N} \displaystyle\sum_{j=0}^{N-1} max(F_b(t_j))^2 \\[2ex] T = \displaystyle\int_{t_0}^{t_f} dt = \int_{t_0}^{t_f} \dfrac{1}{v} ds, s \in [p_0, p_f] \end{array} \right.$$

$$\textbf{s.t.}\ \left\{ \begin{array}{l} |\theta_i(t_j)| \leq \theta_{max}, 1 \leq j \leq N \\ |\dot{\theta}_i(t_j)| \leq \omega_{max} \\ |\ddot{\theta}_i(t_j)| \leq a_{max} \\ \tau^i_{min} \leq \tau_i \leq \tau^i_{max}, i = 1, 2, ..., n \end{array} \right.$$

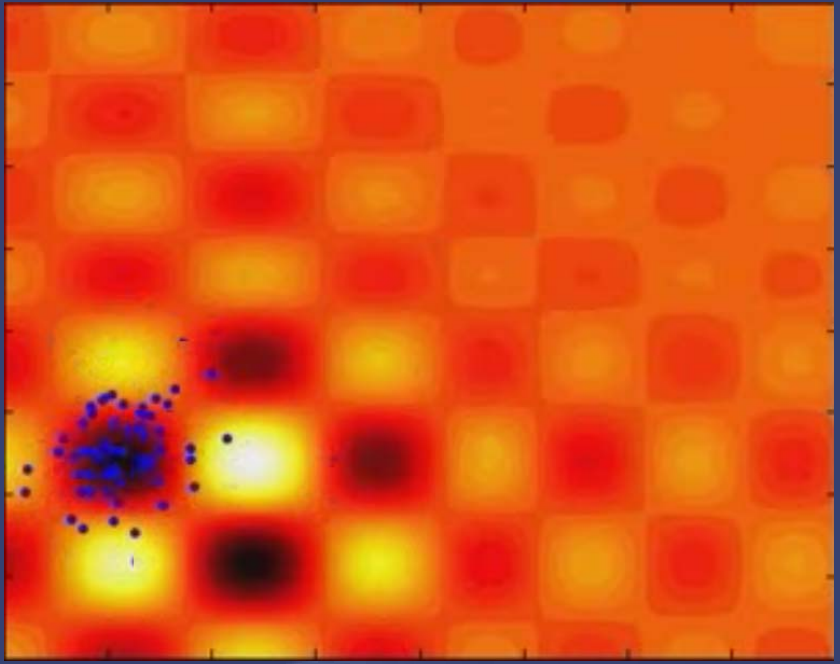
Particle Swarm Optimization

- Stochastic simulation of the social behavior of bird flocks (fish schools, particle swarms)
- Each solution is considered as a ***particle*** moving in $\mathbf{R}^n$ with the law (i = 1...N):

$$\underline{v}_i(k+1) = \alpha\underline{v}_i(k)+c_1\mathrm{rand}()(\underline{p}_i(k)-\underline{x}_i(k)) + c_2\mathrm{rand}()(\underline{p}_g(k)-\underline{x}_i(k))$$

$$\underline{x}_i(k+1) = x_i(k)+\beta\underline{v}_i(k+1)$$

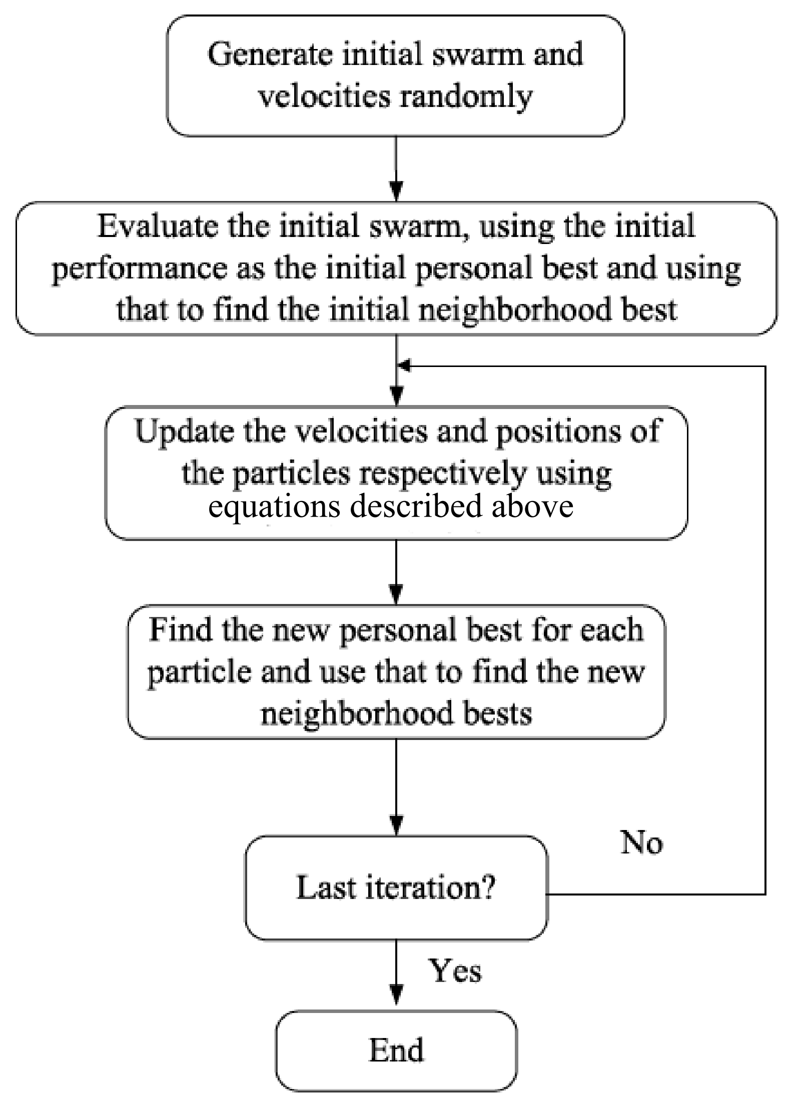
- Autobiographical memory (“simple nostalgia”) + shared knowledge of the swarm



Algorithm parameters

- c_1 (cognitive parameter)
- c_2 (social parameter)
- α (inertia weight: <u>tradeoff global vs local exploration</u>)
- β (constriction factor)

Particle Swarm Optimization



Multiobjective optimization

Personal and Neighborhood bests are non-dominated solutions lists (according to Pareto preference)

The comparison is made against a solution randomly selected from non-dominated solutions lists

When a new non-dominated solution is found, it is added to non-dominated solutions lists

The proposed method

- Weighted sum method:

min $\omega_1 T + \omega_2 \Gamma$ (s.t. constraints defined above)

- ω_i (randomly chosen in [0,1]) define the PSO search direction → solve PSO for different weights vectors

1. Define n inter-knot points and $t_{ik} \leq t_{max}$ (inter-knots travel time)
2. Define ω_i and solve PSO (for $n_{iter} < N_{max}$)
3. Update personal and neighborhood best solutions lists
4. Redefine ω_i and loop to 2. until a termination condition is reached

Simulation results

- Simulation validation with a planar 2 DOFs free-flying space robot model:

$m_0 = 40$ Kg, $m_1 = 4$ Kg, $m_2 = 3$ Kg

$L = L_1 = L_2 = 1$ m

$I_0 = 6.67$ Kg m^2, $I_1 = 0.33$ Kg m^2, $I_2 = 0.25$ Kg m^2

$-\pi \leq \theta_i \leq \pi$, $\theta'_{i\,max} = 5$rad/s, $\theta''_{i\,max} = 20$rad/s (i=1,2)

$\tau_{1\,max} = 100$Nm, $\tau_{2\,max} = 50$Nm

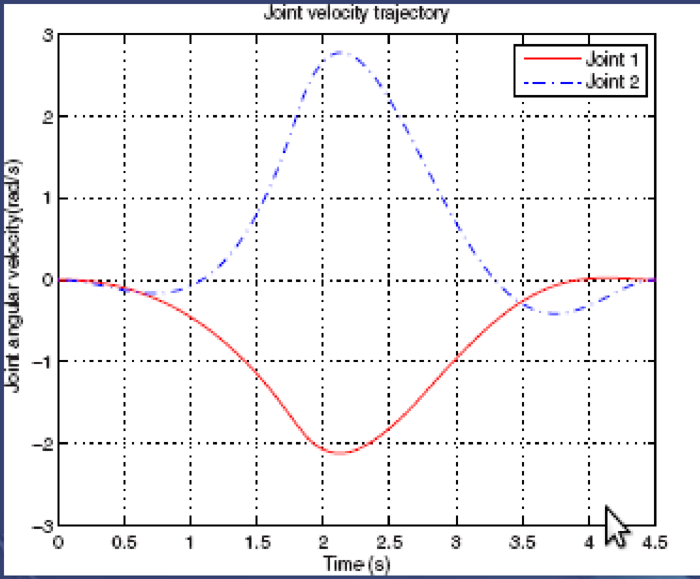
one inter-knot point, $t_{ik} \leq 2$s

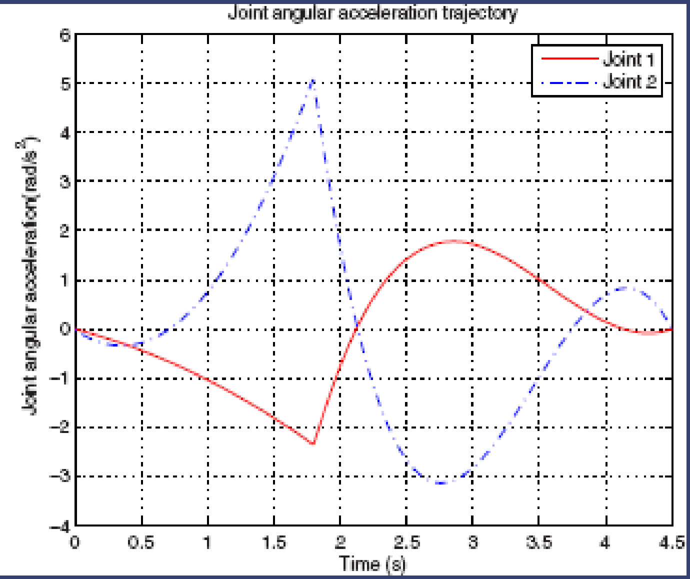
- Test trajectory:

$\underline{\theta}_0=[\theta_{10}, \theta_{20}] = [\pi/3, -\pi/6] \rightarrow \underline{\theta}_F = [\theta_{1F}, \theta_{2F}] = [-3\pi/4, 5\pi/7]$

$\underline{\theta}'_0 = \underline{\theta}''_F = \underline{0}$

Simulation results





one inter-knot point

↓

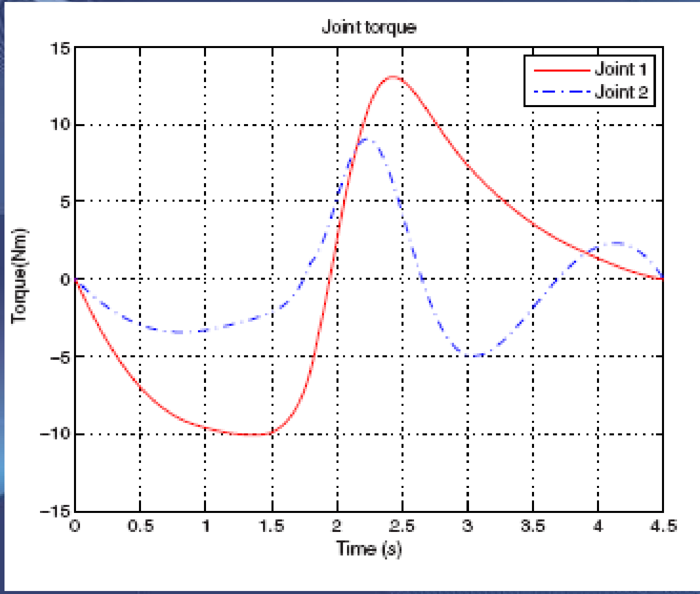
6 parameters:

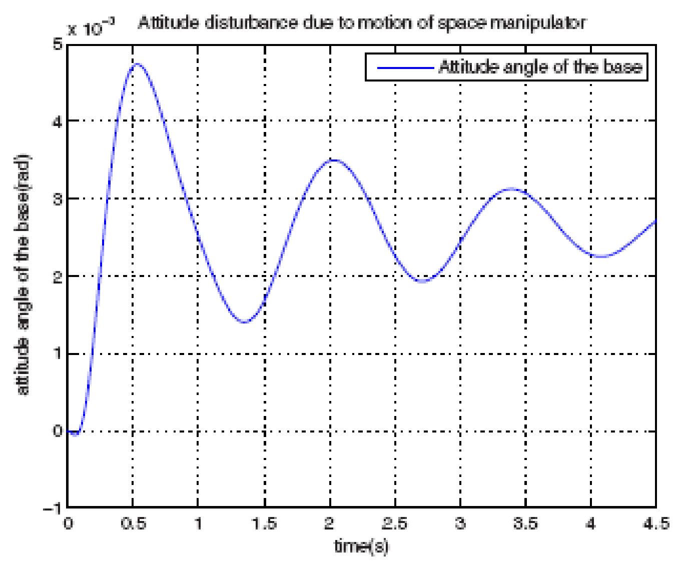
$\underline{\theta}_K = [\theta_{1K}, \theta_{2K}]$

$\underline{\theta}'_K = [\theta'_{1K}, \theta'_{2K}]$

$t_0\ [\theta_0 \rightarrow \theta_K]$

$t_1\ [\theta_K \rightarrow \theta_F]$





$\theta_{1K} = 0.6779$

$\theta_{2K} = -0.8802$

$\theta'_{1K} = -1.5203$

$\theta'_{2K} = 1.7563$

$t_0 = 1.8s$

$t_1 = 2.7s$

Conclusions

- Disadvantage of the proposed weighted sum method: many iterations on random weights
- Use instead the multiobjective variant of PSO (MOPSO [7]), or other MO heuristic methods (GA, DE, etc)
- Different scalarization approach: ε-constraint (disturbance to the base ≤ ε)
- Improvements: on-line optimization (look-ahead), more inter-knot points (computationally expensive)

References

[1] P. Huang et al.: Multi-Objective Optimal Trajectory of Space Robot Using Particle Swarm Optimization

[2] Y. Xu: The Measure of Dynamic Coupling of Space Robot Systems

[3] P. Huang, K. Chen, Y. Xu: Optimal Path Planning for Minimizing Disturbance of Space Robot

[4] P. Huang, Y. Xu: PSO-Based Time-Optimal Trajectory Planning for Space robot with Dynamic Constraints

[5] J. Kennedy, R. Eberhart: Particle Swarm Optimization

[6] J. Kennedy: The Behavior of Particles

[7] C. A. C. Coello, M. S. Lechuga: MOPSO: A proposal for Multiple Objective Particle Swarm Optimization