

## Augmented Lagrangian Particle Swarm Optimization in Mechanism Design\*

Kai SEDLACZEK\*\* and Peter EBERHARD\*\*

\*\*Institute of Engineering and Computational Mechanics,  
University of Stuttgart,  
Pfaffenwaldring 9, 70569 Stuttgart, Germany  
E-mail: sedlaczek, eberhard@itm.uni-stuttgart.de

### Abstract

The problem of optimizing nonlinear multibody systems is in general nonlinear and nonconvex. This is especially true for the dimensional synthesis process of rigid body mechanisms, where often only local solutions might be found with gradient-based optimization methods. An attractive alternative for solving such multimodal optimization problems is the Particle Swarm Optimization (PSO) algorithm. This stochastic solution technique allows a derivative-free search for a global solution without the need for any initial design. In this work, we present an extension to the basic PSO algorithm in order to solve the problem of dimensional synthesis with nonlinear equality and inequality constraints. It utilizes the Augmented Lagrange Multiplier Method in combination with an advanced non-stationary penalty function approach that does not rely on excessively large penalty factors for sufficiently accurate solutions. Although the PSO method is even able to solve nonsmooth and discrete problems, this augmented algorithm can additionally calculate accurate Lagrange multiplier estimates for differentiable formulations, which are helpful in the analysis process of the optimization results. We demonstrate this method and show its very promising applicability to the constrained dimensional synthesis process of rigid body mechanisms.

**Key words:** Particle Swarm Optimization, Nonlinear Constraints, Mechanism Synthesis, Multibody Systems

### 1. Introduction

A simple problem formulation for the process of dimensional synthesis of rigid body mechanisms can be written as

$$\mathbf{p}^* = \arg \min f(\mathbf{p}), \quad f: \mathbb{R}^n \mapsto \mathbb{R}, \quad \text{s.t. } \mathbf{p}_l \leq \mathbf{p} \leq \mathbf{p}_u, \quad (1)$$

where the objective function  $f$  is to be minimized with respect to the vector of design variables  $\mathbf{p} \in \mathbb{R}^n$  that represents the geometry parameters of the mechanism (lengths, angles, positions). The design variables are usually bounded by physically meaningful lower and upper bounds  $\mathbf{p}_l$  and  $\mathbf{p}_u$  that limit the size of the individual linkages of the mechanism.

The objective function can be defined as the structural error determined by the deviation of the desired performance and the performance obtained using the actual design. The performance measure depends on the functional requirement which can be classified into the three main problems *function generation*, *path generation* and *motion generation*,

see Erdman and Sandor (7). As illustrated in Fig. 1, the objective function can be defined as

$$f = \frac{1}{N_T} \sum_{i=1}^{N_T} \rho_1 (x_o^i(\mathbf{p}) - x_t^i)^2 + \rho_2 (y_o^i(\mathbf{p}) - y_t^i)^2 + \rho_3 r (\alpha_o^i(\mathbf{p}) - \alpha_t^i)^2, \quad (2)$$

where  $N_T$  is the total number of measurement points (time steps) and the indices  $o$  and  $t$  describe the output and target trajectory, respectively. The angular deviation can be multiplied with a characteristic length  $r$  in order to obtain comparable terms in the objective function formulation. The coefficients  $\rho_i$  are chosen according to the functional requirement. For the problem of path generation, e.g., only the path of the output point is considered and thus the rotational orientation can be neglected ( $\rho_3 = 0$ ), see Fig. 1. In Eq. (2), the structural error is measured point wise. Thus, the objective function does not only include some positional deviation from the target trajectory, it also incorporates some coarse velocity information. If this is not desired, the input rates might be parameterized in time or more sophisticated formulations could be used, see Hansen (9).

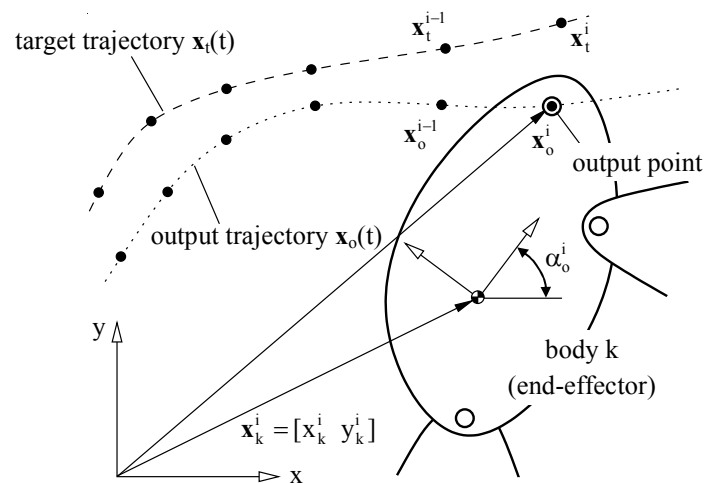


Figure 1: The problem of dimensional synthesis of mechanisms. An output link (end-effector) has to follow a prescribed trajectory. This can include both angular and positional orientation of the output point/body according to the functional requirement.

In addition to the simple bounds of the design variables specified in Eq. (1), the feasible design space might be further limited by additional nonlinear equality and inequality constraints

$$\mathbf{g}(\mathbf{p}) = \mathbf{0}, \quad \mathbf{g} : \mathbb{R}^n \mapsto \mathbb{R}^{m_e}, \quad \mathbf{h}(\mathbf{p}) \leq \mathbf{0}, \quad \mathbf{h} : \mathbb{R}^n \mapsto \mathbb{R}^{m_i}. \quad (3)$$

For the problem of dimensional synthesis, these constraints might represent workspace restrictions, kinematic constraints or other requirements.

The optimization problem defined by Eqs. (1) to (3) is in general nonlinear, nonconvex and can be non-differentiable. Sophisticated methods like SQP algorithms might fail or the result – representing only a local minimum – depends on the initial design. Thus, stochastic solution methods have become an interesting alternative with the benefit of increasing computing power available nowadays. A relatively new stochastic optimization method is the Particle Swarm Optimizer which was introduced by Kennedy and Eberhart (12). Its functionality is based on the observation, interpretation and simulation of bird flocks or fish

schools. The movement of individuals/particles relies on their cognitive and social behavior as well as on their collective behavior as a swarm and enables a powerful search for an optimal place in the design space. The social interaction of the particles and their individual behavior is assigned stochastically. This makes PSO a random search technique that can be implemented with only a few lines of computer code.

PSO has attained a lot of attention in recent years and many PSO variants and applications have been developed. Besides the basic concept of PSO as presented by Kennedy and Eberhart (12), a more in-depth study of the PSO functionality with convergence analysis can be found in the doctoral thesis of Bergh (1) and a comprehensive text book about PSO was recently presented by Clerc (5). Many variants were developed tackling different problem formulations and characteristics. Venter and Sobieszczanski-Sobieski (21) as well Laskari et al. (13) presented some modifications towards integer programming. Clerc (4) modified the PSO algorithm in order to solve combinatorial problems like the Travelling Salesman Problem. He also developed an adaptive, parameter free version of PSO (3). Coello and Lechuga (6) proposed a PSO algorithm to solve multi-objective optimization problems by computing the Pareto-optimal front. Some examples for engineering applications using Particle Swarm Optimization are presented by Hu et al. (11), Shutte and Groenwold (17) or Sedlaczek and Eberhard (18).

Like many other engineering optimization problems, the process of dimensional synthesis of mechanisms often requires methods that are capable of handling the problem immanent equality and inequality constraints. Whereas gradient-based methods usually include constraints efficiently by the use of Lagrange multipliers, stochastic methods are often unable to solve constrained problems in a reasonable way avoiding simple trial and error. Some methods make use of penalty functions in order to reduce the constrained problem to an unconstrained problem by penalizing the objective function despite ill-conditioning. But this work utilizes the simple structure of PSO and combines this technique with an extended non-stationary penalty function approach, the Augmented Lagrange Multiplier Method, where ill-conditioning is a far less harmful problem and the correct solution can be obtained even for finite penalty factors. The resulting Augmented Lagrangian Particle Swarm Optimizer (ALPSO) represents a robust optimization algorithm, which additionally provides accurate Lagrange Multiplier estimates for differentiable problems. These Lagrange multipliers might be helpful in the analysis process of the optimization results and can be interpreted as a cost measure of the corresponding constraints. The robust and easy-to-use algorithm and the derivative-free search without the restriction to local solutions make ALPSO an attractive solution method especially in the process of dimensional synthesis of rigid body mechanisms.

In the following Section 2, we present the ALPSO technique including the basic concept of PSO and its combination with the Augmented Lagrange Multiplier Method. In Section 3, we demonstrate the basic problem of dimensional synthesis designing a slider-crank mechanism which is used as a benchmark problem in this paper. We show the solution of an unconstrained problem of path generation as well as the results of the synthesis process with nonlinear workspace restrictions. Some concluding remarks are made at the end.

## **2. Augmented Lagrangian Particle Swarm Optimization**

Particle Swarm Optimization is an iterative, population based method similar to Evolutionary Algorithms. It utilizes swarm intelligence in order to find the best place in the design space. This can be interpreted as a swarm of particles that are flying through the design space while interchanging information about the hyper-surface that is defined by the objective function and possible constraints. The actual position of a particle  $i$  is associated

with the design vector  $\mathbf{p}_i$ . The trajectory of the  $i^{\text{th}}$  particle through the design space can be described by only two simple equations that compose entirely the basic algorithm of PSO. These are the position update

$$\mathbf{p}_i^{k+1} = \mathbf{p}_i^k + \Delta \mathbf{p}_i^{k+1} \quad (4)$$

and the velocity update equation

$$\Delta \mathbf{p}_i^{k+1} = w \Delta \mathbf{p}_i^k + c_1 r_{1,i}^k (\mathbf{p}_i^{\text{best},k} - \mathbf{p}_i^k) + c_2 r_{2,i}^k (\mathbf{p}_{\text{swarm}}^{\text{best},k} - \mathbf{p}_i^k), \quad (5)$$

where the change in position  $\Delta \mathbf{p}_i^{k+1}$  is often referred to as velocity in PSO references. At the current iteration  $k$ ,  $\mathbf{p}_i^{\text{best},k}$  represents the best position a particle has experienced by itself and  $\mathbf{p}_{\text{swarm}}^{\text{best},k}$  is the best position of the entire swarm. The individual movement of a particle depends on three components. The first term in Eq. (5) is proportional to the inertia factor  $w$  and it incorporates some information from previous iterations into the further movement of the particle. The second term is referred to as the cognitive performance of the particle which leads to a position change towards its own best position. Finally, the social interaction is represented by the third term, which is responsible for the information exchange between all particles of the swarm. The cognitive and the social scaling factor  $c_1$  and  $c_2$  – together with the inertia factor  $w$  – determine the behavior of the particles, whereas  $r_1$  and  $r_2$  are uniformly distributed numbers in  $[0,1]$  necessary for a stochastic and diverse search. Figure 2 illustrates the basic steps of the PSO algorithm, which can be implemented with only a few lines of computer code.

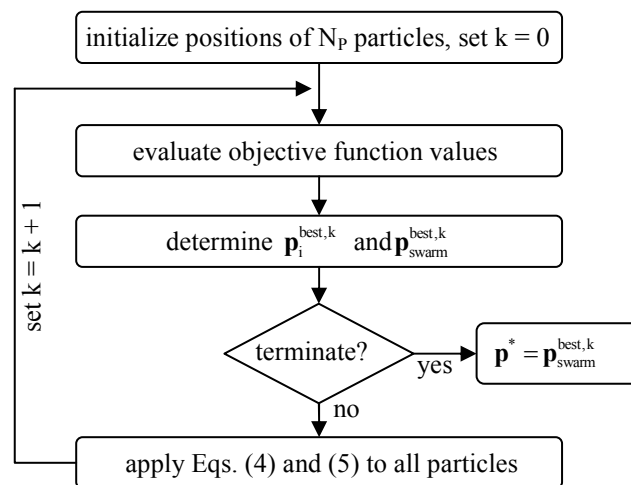


Figure 2: Flowchart of the basic PSO algorithm.

The inertia factor as well as the cognitive and social scaling factors can be used to control the swarm behavior, especially the convergence and search diversity properties of the algorithm. Appropriate values guarantee convergence, but the optimal values regarding convergence rates and diversity are problem dependent. The inertia factor is usually decreased as the search progresses in order to improve the convergence rate. On the other hand, premature convergence to a local solution might be unwanted. Therefore, using so-called neighborhood models can improve the diverse search. The basic idea of the neighborhood models is to replace  $\mathbf{p}_{\text{swarm}}^{\text{best},k}$  in Eq. (5) by  $\mathbf{p}_{\text{nh}}^{\text{best},k}$ , that is the best position only in a somehow defined neighborhood of particle  $i$ , and thus to maintain multiple attractors in the swarm. More information about PSO, its variants and convergence and diversity characteristics are provided by Bergh (1) and Clerc (5).

### 2.1 Constrained PSO and the Augmented Lagrange Multiplier Method

Due to its simplicity and its diverse search capabilities, the PSO method is an attractive solution algorithm in the process of dimensional synthesis of mechanisms. However, engineering optimization tasks often consist of problem immanent equality and inequality constraints. Hence, the basic PSO algorithm must be extended in order to handle such constraints, which can be done by introducing a penalty function that maps the objective to a pseudo objective function

$$\psi(\mathbf{p}) = f(\mathbf{p}) + r_p \phi(\mathbf{p}). \quad (6)$$

The most common penalty function is the quadratic penalization that violates any deviation from the feasible design subspace. For equality constraints, e.g., this quadratic penalty function is given by

$$\phi(\mathbf{p}) = \mathbf{g}(\mathbf{p}) \cdot \mathbf{g}(\mathbf{p}). \quad (7)$$

However, the accuracy of the solution with active constraints depends on the value of the penalty factor  $r_p$ , which must be infinitely large for an exact solution. Alternatively, a more sophisticated penalty function is given by the Augmented Lagrange Multiplier Method, where a finite penalization is sufficient. The generalized Lagrange function can be written as

$$L(\mathbf{p}, \boldsymbol{\lambda}) = f(\mathbf{p}) + \sum_{i=1}^{m_e} \lambda_i g_i(\mathbf{p}) + \sum_{j=1}^{m_i} \lambda_{j+m_e} h_j(\mathbf{p}), \quad (8)$$

where the solution of the original constrained minimization problem is a stationary point of  $L$  for the correct Lagrange multipliers  $\boldsymbol{\lambda}^*$ . However, it is not necessarily a minimum. Therefore, the Augmented Lagrange function consists of a third term that alters the stationary point into a minimum by adding a quadratic penalization,

$$L_A(\mathbf{p}, \boldsymbol{\lambda}, r_p) = f(\mathbf{p}) + \sum_{i=1}^{m_e+m_i} \lambda_i \theta_i(\mathbf{p}) + \sum_{i=1}^{m_e+m_i} r_{p,i} \theta_i^2(\mathbf{p}), \quad (9)$$

with

$$\theta_i(\mathbf{p}) = \begin{cases} g_i(\mathbf{p}), & i = 1, \dots, m_e, \\ \max \left[ h_{i-m_e}(\mathbf{p}), \frac{-\lambda_i}{2r_{p,i}} \right], & i = m_e + 1, \dots, m_e + m_i. \end{cases} \quad (10)$$

The definition of the penalty function  $\theta(\mathbf{p})$  is based on a continuous and differentiable problem formulation. The term  $-\lambda_i/2r_{p,i}$  in Eq. (10) is chosen in order to have continuous derivatives  $\partial L_A / \partial \mathbf{p}$  at  $\hat{\mathbf{p}}$  where  $h_{i-m_e}(\hat{\mathbf{p}}) = -\lambda_i/2r_{p,i}$ . This is reasonable but arbitrary since PSO does not rely on any derivatives or smooth functions. But it is straightforward and fully functional for the proposed method.

Figure 3 depicts the functionality of the Augmented Lagrange function. As shown for the illustrated example with the objective function  $f(\mathbf{p}) = p^3$  and the equality constraint  $g(\mathbf{p}) = p + 1 = 0$ , a finite penalty value is sufficient for an exact solution of the constrained minimization problem.

Since the correct Lagrange multipliers  $\boldsymbol{\lambda}^*$  are unknown, the Augmented Lagrange function (9) cannot be used directly as a penalized pseudo objective function. The correct Lagrange multipliers and the appropriate penalty factors are problem dependent and are a priori unknown. It rather must be solved a sequence of unconstrained problems (9) with subsequent updates of  $\boldsymbol{\lambda}$  and  $r_p$ . According to differentiable problems, we apply an update scheme based on the solution of the stationary condition of the  $v^{\text{th}}$  subproblem. If  $\mathbf{p}^v$  is a

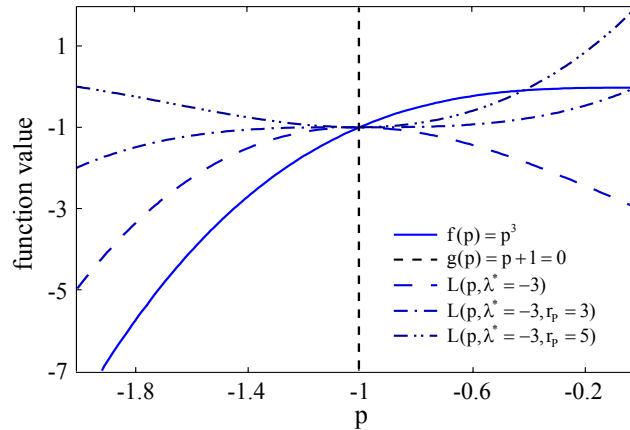


Figure 3: The Lagrange function  $L$  has a stationary point at  $\mathbf{p}^*$ , which is a minimum of the Augmented Lagrange function  $L_A$  using a finite penalty factor  $r_p$ .

solution estimate of such subproblem, it holds for  $\mathbf{p}^v \approx \mathbf{p}^{*v}$  that

$$\left[ \frac{\partial f(\mathbf{p})}{\partial \mathbf{p}} + \sum_{i=1}^{m_g+m_h} \lambda_i^v \frac{\partial \theta_i(\mathbf{p})}{\partial \mathbf{p}} + \sum_{i=1}^{m_g+m_h} 2r_{p,i}^v \theta_i(\mathbf{p}) \frac{\partial \theta_i(\mathbf{p})}{\partial \mathbf{p}} \right]_{\mathbf{p}=\mathbf{p}^v} = \boldsymbol{\varepsilon}^v \approx \mathbf{0}. \quad (11)$$

If the residual  $\boldsymbol{\varepsilon}^v$  is small, an update scheme can be formulated based on the comparison of Eq. (11) with the stationarity condition of the Lagrange function (8). This yields

$$\lambda_i^{v+1} = \lambda_i^v + 2r_{p,i}^v \theta_i(\mathbf{p}). \quad (12)$$

As long as the penalty factors  $r_p$  are large enough, the particles will move towards the feasible region. Although excessively large penalty factors do not alter the stationarity condition of the pseudo problem (9), the reduction of the corresponding penalization is essential for convergent and accurate Lagrange multiplier estimates. Regarding Eq. (11), it cannot be expected that the solution of the  $v^{\text{th}}$  subproblem is exact and thus  $\boldsymbol{\varepsilon}^v > 0$ . Moreover, the magnitude of the residual is proportional to the penalty factor. Therefore it is necessary for accurate Lagrange multiplier updates to have not excessively large penalty factors, since the update equation (12) is based on the assumption of a negligible residual  $\boldsymbol{\varepsilon}^v \approx 0$ . The heuristic updates of the penalty factors can be summarized as

$$\mathbf{r}_{p,i}^{v+1} = \begin{cases} 2r_{p,i}^v & \text{if } |g_i(\mathbf{p}^v)| > |g_i(\mathbf{p}^{v-1})| \wedge |g_i(\mathbf{p}^v)| > \varepsilon_g \\ \frac{1}{2}r_{p,i}^v & \text{if } |g_i(\mathbf{p}^v)| \leq \varepsilon_g \\ r_{p,i}^v & \text{else} \end{cases} \quad i = 1, \dots, m_g, \quad (13)$$

$$\mathbf{r}_{p,j+m_g}^{v+1} = \begin{cases} 2r_{p,j+m_g}^v & \text{if } h_{j_i}(\mathbf{p}^v) > h_{j_i}(\mathbf{p}^{v-1}) \wedge h_{j_i}(\mathbf{p}^v) > \varepsilon_h \\ \frac{1}{2}r_{p,j+m_g}^v & \text{if } h_{j_i}(\mathbf{p}^v) \leq \varepsilon_h \\ r_{p,j+m_g}^v & \text{else} \end{cases} \quad j = 1, \dots, m_h,$$

where the appropriate penalty factor  $r_{p,i}$  is increased if the intermediate solution  $\mathbf{p}^v$  is not closer to the feasible region defined by the  $i^{\text{th}}$  constraint. The penalty factor is reduced if the best position of the particles moves towards the feasible subspace or the  $i^{\text{th}}$  constraint is satisfied within the user defined tolerances  $\varepsilon_g$  and  $\varepsilon_h$ , respectively.

The main steps of the ALPSO algorithm are illustrated in Fig. 4. More information about the algorithm and a detailed description is given by Sedlaczek and Eberhard (18).

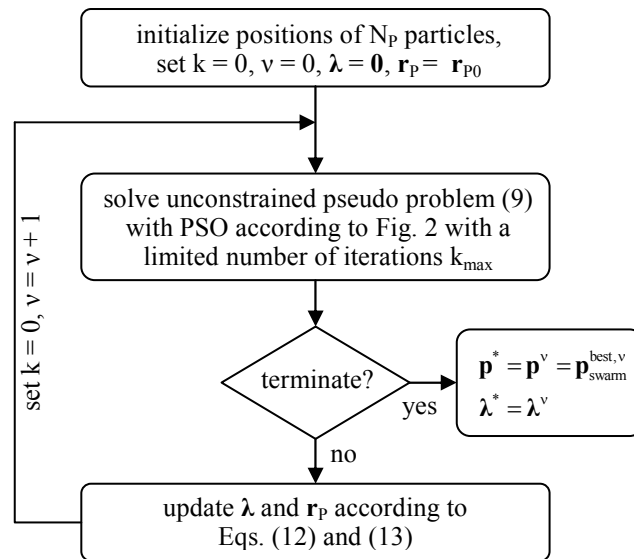


Figure 4: Flowchart of the Augmented Lagrangian Particle Swarm Optimization algorithm for constrained problems.

In order to demonstrate the functionality of ALPSO and the typical convergence characteristics of the Lagrange multiplier estimates and penalty factors, Figure 5 shows the optimization history of the method solving the simple problem

$$\begin{aligned}
 & \min_{\mathbf{p}} p_1^2 + p_2^2, \\
 \text{s.t.} \quad & g_1 = p_1 - 3 = 0, \\
 & h_1 = 2 - p_2 \leq 0.
 \end{aligned} \tag{14}$$

Certainly, this problem could be solved with an efficient gradient-based method or, preferably, the solution might be derived analytically by a simple hand calculation. However, it demonstrates clearly the functioning of the two-loop structured ALPSO algorithm. The penalty factors are consecutively increased until the particles move towards the feasible region. As soon as the residual becomes small, the Lagrange multipliers are estimated more accurately. Finally, a reduction to necessary but not superfluous large penalty factors yields a stationary and accurate estimation of the corresponding Lagrange multipliers. Within a few iterations, the constrained problem is reduced to a quasi-stationary unconstrained problem with appropriate multipliers and penalty factors. Thus, the nested loop structure of ALPSO implies only a slight increased computational burden until appropriate values for the Lagrange multipliers and penalty factors are identified. More benchmark results of ALPSO are presented by Sedlaczek and Eberhard (18). It is important to mention that ALPSO benefits from relatively small numbers for the social and cognitive scaling factors  $c_{1,2} \approx 0.5$ . Since the Lagrange multipliers and the penalty factors are dynamically updated, the PSO algorithm has to track the changing minimum of the non-stationary pseudo objective function (9). Premature convergence would impede this tracking behavior. For the same reason, we do not apply a linearly decreasing inertia factor  $w$ , we rather extend the right hand side of the velocity update equation (5) by a fourth vector  $\Delta \mathbf{p}_i^{\text{crazy},k}$ , whose  $j^{\text{th}}$  element is a random number drawn from a normal distribution with mean value 0 and standard deviation  $r_{s,j}$  ( $r_{s,j} \ll p_{u,j} - p_{l,j}$ ). This can be interpreted as some craziness that is stochastically assigned to each particle. As we have experienced in many experiments, this additional stochastic search in the surrounding neighborhood with radius  $r_s$  yields an improved tracking behavior. Finally, it turned out that for the test

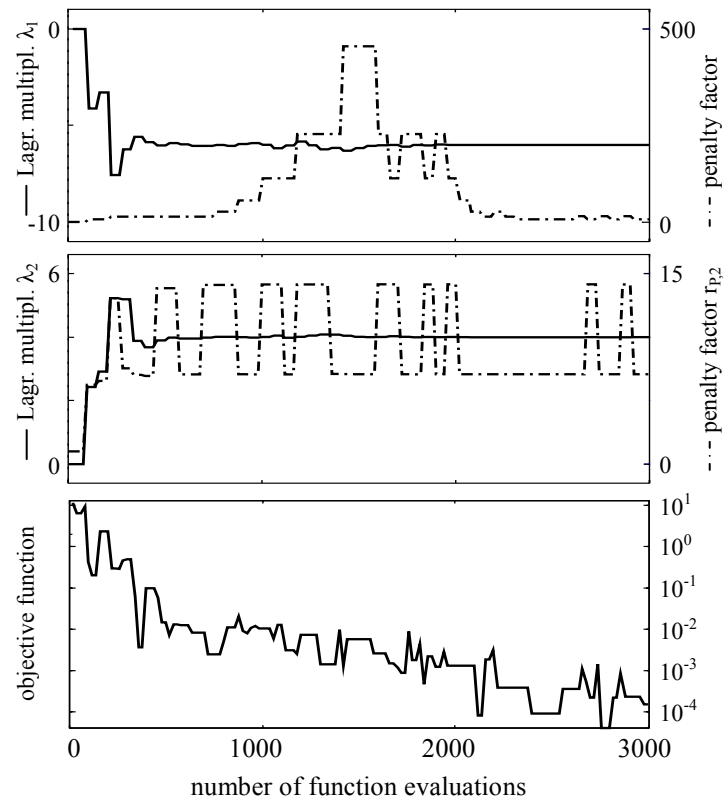


Figure 5: Convergence characteristics of Lagrange multiplier estimates, penalty factors and objective function value of ALPSO solving the test problem (14).

problems conducted during the development,  $k_{\max} = 3$  subsequent basic PSO iterations are sufficient. Most problems could even reliably be solved with  $k_{\max} = 2$ , which almost eliminates the computational overhead of the two-loop structure.

### 3. Dimensional Synthesis of a Slider Crank Mechanism

The first benchmark problem for Augmented Lagrangian Particle Swarm Optimization consists of a slider crank mechanism that has to be designed in order to follow the prescribed elliptic target trajectory as illustrated in Fig. 6(a). For this path generation problem, the target path is parameterized in time and can be described by

$$x_1(t) = 3 \cos(t), \quad y_1(t) = 2 \sin(t). \quad (15)$$

The crank of the mechanism is driven counterclockwise with  $\alpha_d(t) = t, 0 \leq t \leq 2\pi$ , (one revolution) and the output point is attached to the connecting rod. As mentioned in Section 1, the objective function of this problem can be formulated as presented in Eq. (2) with  $\rho_3 = 0$ . For the benchmark problem, we used a total of  $N_T = 63$  time steps, which yield a step size of  $\Delta t = 2\pi/63 \approx 0.1$ .

In order to evaluate the objective function of the problem, it is necessary to compute the trajectory of the output link. The kinematic analysis used in this work is based on the absolute coordinate formulation, which has the advantage that it is general, well established and computationally efficient, see Haug (10), Nikravesh (14) or Shabana (20). The positions and angular orientations of the bodies are calculated for each time step by solving the nonlinear set of kinematic constraint equations in implicit form,

$$\mathbf{c}(\mathbf{x}, \mathbf{x}_0, t) = \mathbf{0}. \quad (16)$$

This is done here by applying Powell's method, see Powell (15). For more information about the kinematic analysis, especially about the handling of singular configurations such as lock-up and bifurcation, see Sedlaczek et al. (19).

The solution circuit of the nonlinear kinematic equation (16) depends on the initial position vector  $\mathbf{x}_0$ , see Chase and Mirth (2). Thus, the initial positions and orientations must be included into the vector of design variables for the process of the dimensional synthesis. Otherwise the optimization result is based only on one solution circuit/branch of Eq. (16), which is determined by the initial position  $\mathbf{x}_0$ . This might represent a local solution, not showing the full capability of the slider crank mechanism. However, we extend the design space not by adding the initial position vector but by an extended description of the positions and orientations of the joints with respect to the local reference system of each body. Figure 6(b) illustrates the extended geometric description of the slider crank mechanism. For a binary link, not only a single dimensional property – namely the length – but four geometry parameters are used to have a sufficient description. Hence, we have a total of 18 design variables for the benchmark problem. A more detailed description of the kinematic constraint equations can be found in Gaugele (8) and in Sedlaczek et al. (19).

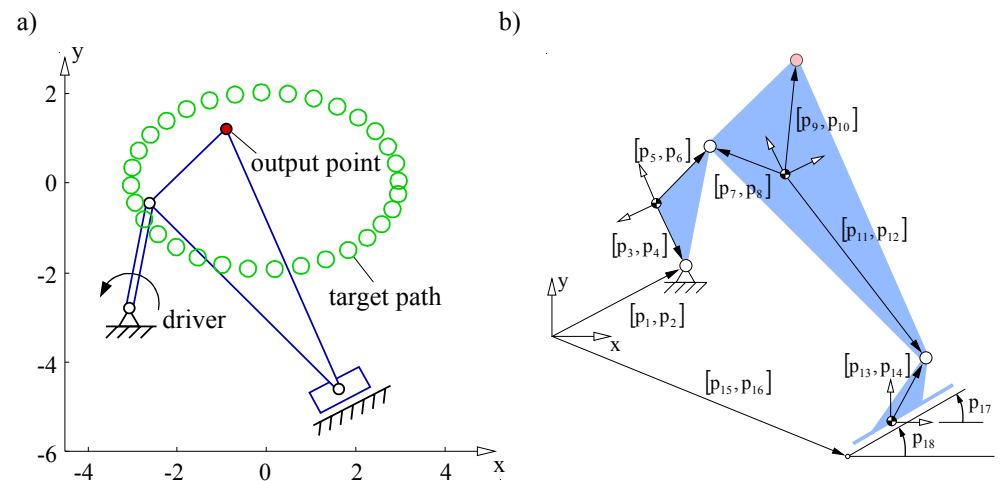


Figure 6: a) The benchmark path generation problem is to synthesize a slider crank mechanism whose output point follows the elliptical target path as close as possible. b) The design variables describing the geometry of the slider crank mechanism.

### 3.1 Unconstrained Synthesis of the Slider Crank Mechanism

The geometric description as mentioned above allows the formulation of an optimization based synthesis problem. This can be written as

$$\min_{\mathbf{p}} f(\mathbf{p}) \quad \text{with } f(\mathbf{p}) = \frac{1}{N_T} \sum_{i=1}^{N_T} (x_o^i(\mathbf{p}) - x_t^i)^2 + (y_o^i(\mathbf{p}) - y_t^i)^2 \quad (17)$$

$$\text{s.t. } \mathbf{p}_l \leq \mathbf{p} \leq \mathbf{p}_u.$$

The lower and upper bounds are chosen in order to limit the dimensions of the individual bodies. With the simple bounds  $-5 \leq p_i \leq 5$  for the lengths and  $0 \leq p_i \leq 2\pi$  for the angular variables, the Particle Swarm algorithm has found a slider crank mechanism as depicted in Fig. 7(a) with an objective function value of  $f = 6.983 \cdot 10^{-3}$ . The optimization was stopped after  $v_{\max} = 500$  iterations, which corresponds to 1500 basic PSO iterations ( $k_{\max} = 3$ ). The population size was set to 200 particles. The objective function value could only be marginally improved by increasing the maximum number of iterations or by using a

subsequent local search method. However, PSO does not guarantee the global solution. Thus, improved designs might be found, especially with extended simple bounds and increased populations.

### 3.2 Slider Crank Mechanism with Workspace Constraints

In the dimensional process of mechanisms, it is often necessary to consider nonlinear constraints, such as workspace restrictions. There might be, e.g., a limited workspace for the working motion of a machine tool whose end-effector is guided by the slider crank linkage. For demonstration, we have chosen simple workspace restrictions as shown in Fig. 7(b). All vertices of the slider crank mechanism must stay inside the available workspace during operation. These restrictions can be formulated as shown exemplarily for the first constraint,

$$h_1 = x_{\max} - 4 \leq 0, \quad (18)$$

where  $x_{\max}$  is the right most position of all vertices at all time steps. This formulation is chosen to limit the number of constraints that have to be considered in the optimization process. The constraint function (18) is highly nonlinear due to the nonlinear kinematics involved in the calculation of  $x_{\max}$ . However, the constraints are piecewise differentiable for the presented problem, which allows the computation of Lagrange multiplier estimates. The optimization problem is defined as

$$\begin{aligned} \min_{\mathbf{p}} f(\mathbf{p}) \quad \text{with } f(\mathbf{p}) &= \frac{1}{N_T} \sum_{i=1}^{N_T} (x_o^i(\mathbf{p}) - x_t^i)^2 + (y_o^i(\mathbf{p}) - y_t^i)^2, \\ \text{s.t. } h_1 &= x_{\max} - 4 \leq 0, \\ h_2 &= y_{\max} - 3 \leq 0, \\ h_3 &= -x_{\min} - 4 \leq 0, \\ h_4 &= -y_{\min} - 3 \leq 0, \\ h_5 &= l_{\text{crank}} - l_{\text{con.rod}} + \varepsilon_1 \leq 0, \\ \mathbf{p}_l &\leq \mathbf{p} \leq \mathbf{p}_u, \end{aligned} \quad (19)$$

with an additional inequality constraint  $h_5$  that avoids singular configurations that might occur if the length of the connecting rod  $l_{\text{con.rod}} = \sqrt{(p_7 - p_{11})^2 + (p_8 - p_{12})^2}$  is equal or smaller than the length of the crank shaft  $l_{\text{crank}} = \sqrt{(p_3 - p_5)^2 + (p_4 - p_6)^2}$ . A small value of  $\varepsilon_1 = 0.1$  is added to have an additional safeguard.

The optimized slider crank mechanism as well as the workspace constraints is shown in Fig. 7(b). The structural error increased due to the constraints to a value of  $f = 7.063 \cdot 10^{-2}$ . The corresponding Lagrange multiplier estimates are listed in Table 1. The values are validated with the results obtained with a modification of the SQP method of Schittkowski (16). These Lagrange multipliers are useful in the analysis process of the optimization results. Since they map the gradient of the objective function to the corresponding gradients of the constraint functions,

$$\frac{\partial f}{\partial p_i} = -\sum_{j=1}^{m_c} \lambda_j \frac{\partial g_j}{\partial p_i} - \sum_{j=1}^{m_l} \lambda_{j+m_c} \frac{\partial h_j}{\partial p_i}, \quad (20)$$

they can be interpreted as a cost measure, e.g., a small release  $\delta h_i$  of the constraint  $h_i$  will result in a improvement of the objective function of  $\delta f \approx -\lambda_i \delta h_i$ . Thus, a large Lagrange multiplier identifies an expensive constraint, which might be reconsidered in the subsequent design process.

Table 1: The Lagrange multipliers at the optimum of problem (19) obtained using the ALPSO algorithm and a SQP method.

constraint	$h_1$	$h_2$	$h_3$	$h_4$	$h_5$
$\lambda_i$ (ALPSO)	0	$7.208 \cdot 10^{-2}$	$1.102 \cdot 10^{-1}$	0	$9.665 \cdot 10^{-2}$
$\lambda_i$ (SQP)	0	$8.069 \cdot 10^{-2}$	$1.096 \cdot 10^{-1}$	0	$9.805 \cdot 10^{-2}$

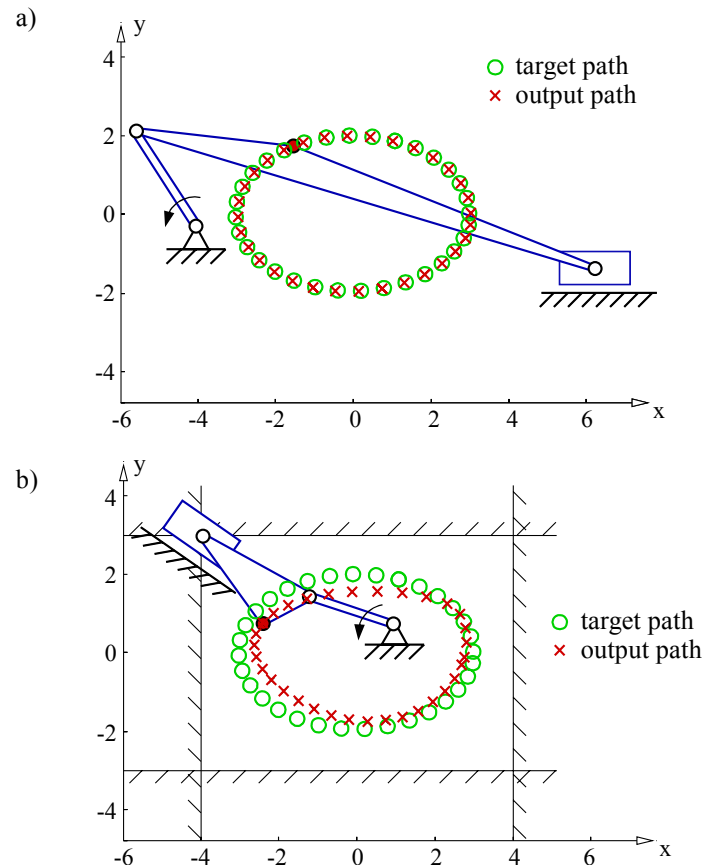


Figure 7: a) The optimal design of the slider crank mechanism without any workspace restrictions, b) optimal design with illustrated workspace constraints.

#### 4. Conclusions

We have presented an extended Particle Swarm Optimization technique that is capable of handling problem immanent equality and inequality constraints. This is done by applying an advanced non-stationary penalty function based on the Augmented Lagrangian Multiplier Method, which allows accurate results without the need for excessively large penalty factors. The resulting optimization method solves nonconvex and nonsmooth problems, but for differentiable problems it additionally provides accurate Lagrange multiplier estimates, which can be of great interest in the analysis process of the optimization results.

The ALPSO method is perfectly applicable to the problem of dimensional synthesis of rigid body mechanisms, which was shown by synthesizing a slider crank mechanism with workspace constraints. These highly nonlinear and multimodal optimization problems can be solved with the stochastic ALPSO method, which is not restricted to local solutions. In

order to demonstrate the functionality and to carry out own numerical investigations, a web-based interface to ALPSO can be found at the address [www.itm.uni-stuttgart.de/research/alpsso](http://www.itm.uni-stuttgart.de/research/alpsso), which is limited to two design variables.

### References

- (1) Bergh, van den F., *An Analysis of Particle Swarm Optimizers*. Ph.D. thesis, University of Pretoria, South Africa, 2001.
- (2) Chase, T.R., Mirth, J.A., Circuits and Branches of Single-Degree-of-Freedom Planar Linkages. *Journal of Mechanical Design*, 115, 1993, pp. 223-230.
- (3) Clerc, M., The Swarm and the Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization. *Proceedings of the IEEE Congress on Evolutionary Computation*, 1999, pp. 1951-1957.
- (4) Clerc, M., Discrete Particle Swarm Optimization. *New Optimization Techniques in Engineering*, Springer, Berlin, 2004.
- (5) Clerc, M., *Particle Swarm Optimization*, ISTE, London, 2006.
- (6) Coello, C.A., Lechuga, M.S., *A Proposal for Multiple Objective Particle Swarm Optimization*. Technical Report EVOCINV-01-2001, Evolutionary Computation Group at CINVESTAV-IPN, Mexico, 2001.
- (7) Erdman, A.G., Sandor, G.N., *Mechanism Design*. Prentice-Hall, Englewood Cliffs, 1991.
- (8) Gaugele, T., *Topologieoptimierte Synthese ebener Starrkörpermechanismen* (in German). Dipl-105, Institute B of Mechanics, University of Stuttgart, 2005.
- (9) Hansen, J.H., Synthesis of Spatial Mechanisms Using Optimization and Continuation Methods. *Computational Dynamics in Multibody Systems*, 1995, pp. 183-196.
- (10) Haug, E.J., *Computer-Aided Kinematics and Dynamics of Mechanical Systems*. Allyn and Bacon, Needham Heights, 1989.
- (11) Hu, X., Eberhart, R., Shi, Y., Engineering Optimization with Particle Swarm. *Proceedings of the IEEE Swarm Intelligence Symposium*, Indianapolis, 2003, pp. 53-57.
- (12) Kennedy, J., Eberhart, R., Particle Swarm Optimization. *Proceedings of the International Conference on Neural Networks*, Perth, Australia, 1995, pp. 1942-1948.
- (13) Laskari, E.C., Parsopoulos, K.E., Vrahatis, M.N., Particle Swarm Optimization for Integer Programming. *Proceedings of the IEEE Congress on Evolutionary Computation*, Honolulu, 2002, pp. 1582-1587.
- (14) Nikravesh, P., *Computer-Aided Analysis of Mechanical Systems*. Prentice-Hall, Englewood Cliffs, 1988.
- (15) Powell, M.J.D., A Hybrid Method for Nonlinear Equations. *Numerical Methods for Nonlinear Algebraic Equations*, 7, 1970, pp. 87-114.
- (16) Schittkowski, K., A Fortran Subroutine Solving Constrained Nonlinear Programming Problems. *Annals of Operations Research*, 5, 1986, pp. 485-500.
- (17) Schutte, J.F., Groenwold, A.A., The Optimal Sizing Design of Truss Structures Using the Particle Swarm Optimization Algorithm. *In Proc. 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Atlanta, 2002.
- (18) Sedlaczek, K., Eberhard, P., Constrained Particle Swarm Optimization of Mechanical Systems. *Proceedings of the 6<sup>th</sup> WCSMO*, Rio de Janeiro, Brazil, 2005.
- (19) Sedlaczek, K., Gaugele, T., Eberhard, P., Topology Optimized Synthesis of Planar Kinematic Rigid Body Mechanisms. *Proceedings of the ECCOMAS Thematic Conference on Multibody Dynamics*, Madrid, Spain, 2005.
- (20) Shabana, A.A., *Computational Dynamics*. John Wiley & Sons, New York, 2001.
- (21) Venter, G., Sobieszczanski-Sobieski, J., Particle Swarm Optimization. *Proceedings of the 43<sup>rd</sup> AIAA Structures, Structural Dynamics and Materials Conference*, Denver, CO, 2002.