

## Research Article

# A Framework for Multidisciplinary Optimization of a Balancing Mechanism for an Industrial Robot

Johan A. Persson,<sup>1</sup> Xiaolong Feng,<sup>2</sup> Daniel Wappling,<sup>2</sup> and Johan Ölvander<sup>1</sup>

<sup>1</sup>Linköping University, S581 83 Linköping, Sweden

<sup>2</sup>ABB Corporate Research, S722 26 Västerås, Sweden

Correspondence should be addressed to Johan A. Persson; [joan.persson@liu.se](mailto:joan.persson@liu.se)

Received 11 December 2014; Revised 5 May 2015; Accepted 19 May 2015

Academic Editor: Rene V. Mayorga

Copyright © 2015 Johan A. Persson et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The paper presents a framework that can be used to design and optimize a balancing mechanism for an industrial robot. The framework has the capability to optimize three different concepts: a mechanical, a pneumatic, and a hydropneumatic. Several disciplines are included in the framework, such as dynamic and static analyses of the robot performance. Optimization is performed for each concept and the obtained optimal designs are all better than the reference design. This means that the framework can be used as a tool both to optimize the balancing mechanism and also to support concept selection.

## 1. Introduction

An industrial robot is a manipulator that is constructed by combining several links, gearboxes, and actuators as seen in Figure 1. A problem when heavy loads are operated is that the resulting torque that affects the gearbox of the second axis can be very high. This is especially true when the load is extended far away from the robot base. This is prevented by constructing a balancing mechanism that produces a torque that acts in the opposite direction of the torque that originates from the load.

The design of the balancing mechanism should ideally fulfil several criteria as good as possible. Some of the objectives are listed below:

- (1) The magnitude of the balancing torque from the balancing mechanism should vary depending on the required balancing.
- (2) The resulting dynamic torque that affects the gearbox of axis two during a robot cycle should be minimal.
- (3) The resulting static torque that affects the gearbox of the second axis when the robot is stopped with a load in an arbitrary position should be minimized.

- (4) The robot should not drop the load if the balancing mechanism malfunctions.
- (5) The balancing mechanism should be made as compact as possible.
- (6) The robot should consume as little energy as possible during a robot cycle.

The optimization problem turns into a multiobjective optimization (MOO) problem since there are several criteria that should be fulfilled [1]. The different criteria can be calculated from different disciplines such as a dynamic analysis, static analysis, solid mechanics, and weight estimation. This means that the optimization also is a multidisciplinary optimization (MDO) problem [2].

Previous work on balancing mechanisms exists. Five different methods to optimize the static balancing of a six-degree-of-freedom industrial manipulator were compared by Saravanan et al. [3].

Simionescu and Ciupitu [4, 5] are using cylindrical helical springs to balance the weight forces for robot arms. A dynamic model of an industrial robot was created by Xiao et al. [6]. They design a mechanical balancing cylinder with the help of the Lagrangian function.

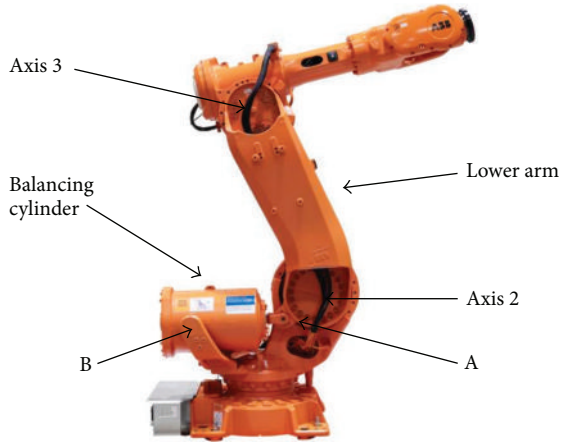


FIGURE 1: An ABB IRB6640 robot courtesy of ABB.

The human interaction with robot arms is investigated by Vermeulen and Wisse [7]. They design a mechanically safe robot arm by using mechanical springs and calculate the static balancing needed to ensure that the human is not hurt.

An optimization framework that optimizes a mechanical balancing cylinder for an industrial robot is presented by Weström et al. [8]. The framework calculates the unbalanced static torque of the second axis of an industrial robot by using the mounting positions and the spring coefficient as design variables. This information is then used to design three concentric springs to ensure proper balancing and low stresses in the springs.

This work is based on the generic idea of MOO and MDO presented in [9] and the initial studies performed in [10, 11]. We propose a formal MOO framework which is used to design and analyse a balancing mechanism in a thorough manner.

*1.1. Paper Outline.* Three different concepts that can be used to balance an industrial robot are presented in Section 2. Section 3 presents different ways to perform MDO and MOO, whereas Section 4 presents the proposed framework. The results from the optimization of the framework are presented in Section 5, and the conclusion summarizes the paper in Section 6.

## 2. Balancing Mechanisms for Industrial Robots

A balancing mechanism with a variable torque can be achieved by several different concepts. It is desirable to choose a concept that requires no external power supply to function. This means that concepts that involve electronics are undesirable. It also means that no sensors are used.

The size of the balancing torque can instead be dimensioned mechanically by attaching a cylinder and a moveable piston to the robot base and the second axis, respectively. This can be seen in Figures 1 and 2. The distance between the attachment points changes when the second axis rotates.

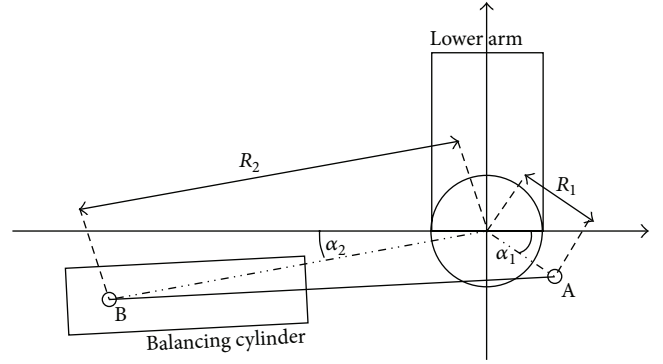


FIGURE 2: Schematic of the mounting positions A and B for a pushing cylinder.

This means that the position of the piston changes as well. A variable axial force and therefore torque are achieved if the cylinder is filled with a medium that generates a force when it is compressed.

This principle is used in the three concepts that are considered in this paper. The three different concepts are a mechanical spring, a gas spring, and a hydropneumatic concept. Each concept is presented in its own subsection.

*2.1. Mechanical Spring.* An easy way to achieve a varying force is to place a mechanical spring inside the cylinder. The spring will produce a larger axial force the more it is compressed as can be seen in the following:

$$F = k_{\text{spring}} \Delta x. \quad (1)$$

The spring stiffness can be calculated according to the following equation as a function of the thread diameter,  $d$ , the spring diameter,  $D$ , the shear modulus,  $G$ , and the number of free turns,  $n$  [12]:

$$k_{\text{spring}} = \frac{Gd^4}{8nD^3}. \quad (2)$$

It is also possible to use several concentric springs in parallel. The overall spring stiffness can then be calculated as the sum of the stiffness of all springs. This is exemplified for three springs in

$$k_{\text{tot}} = \sum_{i=1}^3 k_{\text{spring},i}. \quad (3)$$

This has the benefit that the load will not be dropped even if one of the springs breaks under the assumption that the other springs do not malfunction. This introduces some degree of redundancy in the system and the other springs should be enough to keep the load until the robot can be stopped safely. It is however important to ensure that the outer diameter of the inner springs is smaller than the inner diameter of the outer springs if several concentric springs are used.

2.2. *Gas Spring.* Another possibility is to fill one chamber of the cylinder with gas. The axial force in the piston will then be the pressure of the gas multiplied by the cross-sectional area.

The gas pressure will increase as the gas is compressed. The movements of an industrial robot are assumed to be so fast that the heat transfer with the surroundings is minimal. This means that the pressure change as the volume changes can be assumed to follow a polytropic curve as seen in (4) [13].  $C$  is a constant in this equation which means that the pressure,  $p$ , changes exponentially:

$$C = p_{\text{gas}} V_{\text{gas}}^q. \quad (4)$$

The current gas volume can be expressed as the cross-sectional area of the piston multiplied by the position of the piston (the length between the cylinder bottom and piston), denoted by  $x$ . This makes it possible to express the axial force in the piston as in

$$F = A_{\text{piston}} \frac{C}{(A_{\text{piston}} x)^q}. \quad (5)$$

A drawback with a gas spring solution is that the construction is susceptible to leakages. Close maintenance times of the balancing mechanism are undesirable and the performance of the robot deteriorates if the gas pressure inside the gas spring is reduced. A large leakage might also lead to a sudden pressure drop which introduces a risk that the robot drops the carried load.

2.3. *Hydropneumatic Concept.* It is also possible to use a hydropneumatic concept as a balancing mechanism. The idea is to fill a cylinder with a fluid and connect it to an accumulator. The benefits compared to a gas spring are that the gas is contained in a vessel with only one moving surface and that the accumulator can be placed wherever space is available.

The gas pressure inside the accumulator can be assumed to be equal to the pressure exerted on the piston if it is assumed that the fluid is incompressible compared to the gas. The size of the accumulator can be calculated with (6), where the minimum and maximum working pressures are needed as well as the working volume,  $\Delta V$ :

$$V_0 = \frac{\Delta V (p_1/p_0)}{1 - (p_1/p_2)^{1/n}}. \quad (6)$$

The working volume can be estimated from the geometry of the balancing mechanism as the stroke length of the cylinder multiplied with the cross-sectional area. The latter can be estimated to equal the maximum allowed force in the piston divided by the maximum allowed pressure.

The pressure inside the gas can then be estimated according to (7) where it is a function of the position of the piston,  $x$ . This is in turn a function of the angle of the second axis,  $\theta$ , and the mounting positions of the balancing cylinder:

$$p_1 = \frac{p_0 V_0^q}{(V_0 - A_{\text{piston}} x)^q}. \quad (7)$$

### 3. Optimization

The following two subsections briefly describe state-of-the-art for MDO and MOO. Knowledge about both these sub-fields of optimization is needed to optimize the balancing mechanism efficiently.

3.1. *Multidisciplinary Optimization.* The problem turns into an MDO problem if more than one discipline is involved. It is undesirable to perform optimizations of a few disciplines individually without analysing how the suggested design performs for the other disciplines. There is, for example, usually no use in trying to minimize the weight of a structure without regarding the stresses.

Martins and Lambe [2] have searched through previous works for methods for MDO. They classify the methods and present them together with diagrams and lists of advantages and disadvantages. The methods include the early works by Schmit [14] and Haftka [15], the all-at-once problem statement [16], Simultaneous Analysis and Design (SAND) [16], Individual Discipline Feasible (IDF) [16], Multidisciplinary Feasible (MDF) [16], Concurrent Subspace Optimization (CSSO) [17], Collaborative Optimization [18], Bilevel Integrated System Synthesis (BLISS) [19], Analytical Target Cascading (ATC) [20], Exact and Inexact Penalty Decomposition (EPD and IPD) [21], MDO of Independent Subspaces (MDOIS) [22], Quasiseparable Decomposition (QSD) [23], and Asymmetric Subspace Optimization (ASO) [24].

Regardless of which method is chosen, a problem with MDO is that the wall-clock-time of an optimization may be unrealistically long since several analyses need to be performed. The different methods try to remedy this and two commonly used and easily implemented features are the following.

(1) The first is analysing one of the fastest disciplines first. The rest of the disciplines can be omitted if the suggested simulation performs extremely poor for this discipline. It is instead given a bad objective function value to indicate to the optimization algorithm that it was an unsatisfactory design.

(2) The second is replacing the demanding models with computationally effective surrogate models (SMs) [25]. This introduces additional errors into the optimization, but the reduced optimization time might make up for it.

3.2. *Multiobjective Optimization.* Multiobjective optimization is used to let the computer find Pareto optimal solutions to optimization problems with more than one objective. The Pareto optimal solutions are of interest since there exist no solutions that are better for all objectives.

The procedure of finding the Pareto optimal points can be performed in several different ways. A popular method is to use an optimization algorithm that is tailored for multiobjective optimization, for example, a multiobjective genetic algorithm (MOGA) such as NSGA-II (Non-Dominated Sorting Genetic Algorithm-II) [26]. These algorithms have the benefit that they can identify the whole Pareto optimal front in one optimization run without any input from the decision maker regarding the preferences among the objectives.

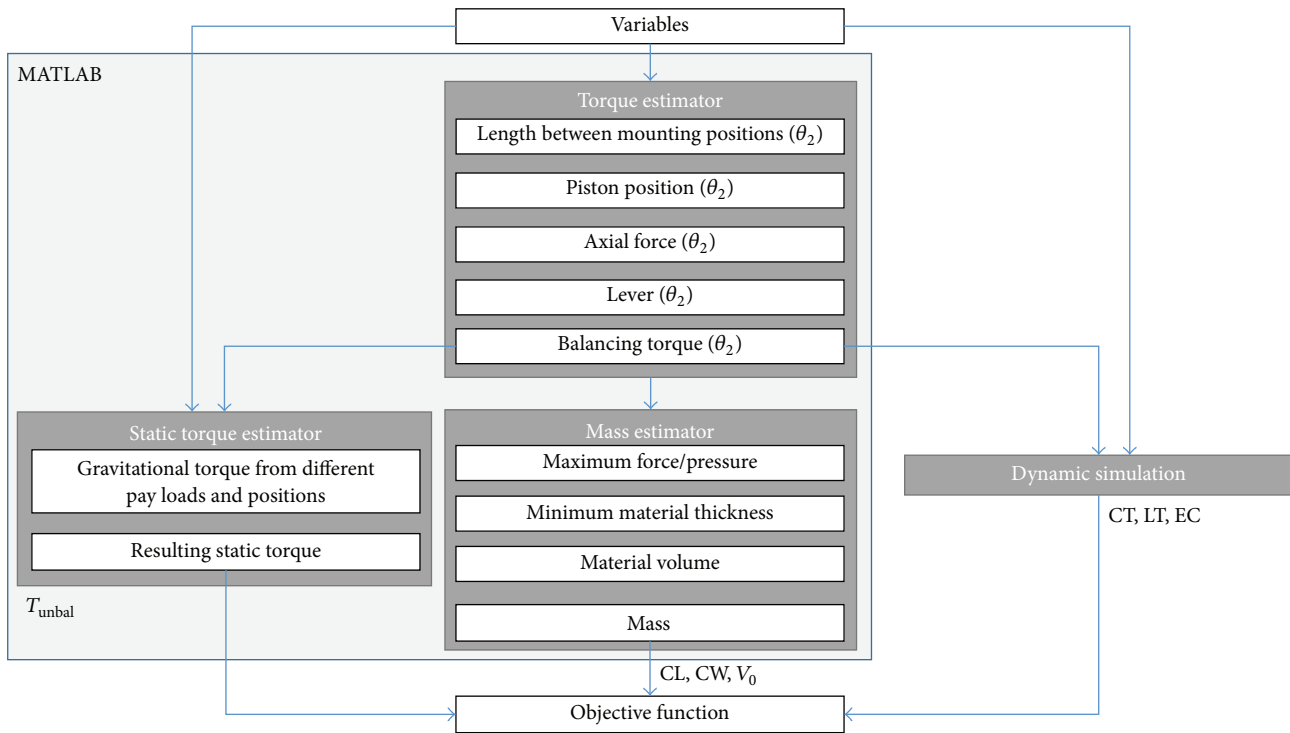


FIGURE 3: Schematic of the proposed framework with focus on how the information is passed between the optimization algorithm and the four tools. They are the four squares with grey background.

It is also possible to use regular singular optimization algorithms even for MOO. The MOO aspect of the optimization needs to be handled in these cases; see, for example, Andersson [27]. One way is to treat all objectives except one as constraints and perform an optimization of the remaining objective. Different solutions will be found depending on the magnitude of the constraint limits. Another method is to aggregate all objectives into a single objective function. All objectives need to be normalized for this method to ensure that they have the same order of magnitude.

#### 4. The Proposed Framework

The framework that is used to optimize the balancing mechanism consists of four different tools as seen in Figure 3. The torque estimator is called first, followed by the three other tools. They can be run in parallel since they only need entities from the torque estimator. More information about the tools is presented in the following subsections.

**4.1. Torque Estimator.** The design variables are sent to the torque estimator that estimates the torque from the balancing cylinder as a function of the angular position of axis two of the industrial robot. One example of such a function can be seen in Figure 4. This function is sent to the dynamic simulation and the static estimator. The torque estimator also estimates the maximum pressure in the balancing system and axial force in the piston. This information is sent to the weight

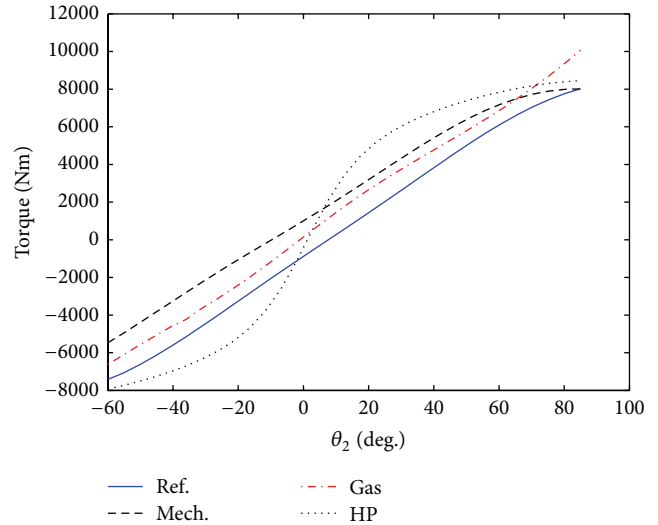


FIGURE 4: Example of how the torque from the balancing cylinder may vary when the angular position of axis two varies. These curves represent the solutions from Table 2.

estimation tool and used to suggest material thickness for the balancing mechanism.

**4.2. Dynamic Simulation.** The dynamic simulation tool calculates numerous dynamic entities during a robot cycle. It comprises a kinematic analysis of the mechanism, dynamic

equations of motion together with the actual path planner and controller used in the robot itself. It uses one input file that defines the robot cycle path and one input file that contains the data of all components in the robot. This includes entities such as lengths and moment of inertias of all links, the size of the actuators, and much more. It is possible to see how a suggested balancing mechanism affects the robot performance by changing the torque data in this input file.

This is the most computationally demanding of the four tools and requires 18 seconds of wall-clock-time on a Dell Precision T1600 with an Intel(R) Xeon(R) CPU E31225 @ 3.10 GHz processor. The other tools are implemented in MATLAB [28] and can be evaluated in under one second each. The fast evaluation of everything except the dynamic model means that an AAO framework can be used.

A problem with the dynamic simulation is that it might return an error if the balancing is inadequate since the mechanism will not be able to move. This needs to be handled by the optimization framework.

The output extracted from this framework is the time it takes to perform a certain cycle in seconds, the energy consumption during the cycle, and the estimated lifetimes of all gearboxes.

**4.3. Static Estimator.** The static estimator estimates the unbalanced static torque on the second axis if the robot is stopped in a position with a load. This is done by varying the angular positions of the second and third axes in steps of five degrees and the load in steps of 50 kg between 0 and 250 kg. This yields a three-dimensional matrix of torques that should be reduced by the torques from the balancing mechanism.

The torques from the balancing mechanism only change with the angular position of the second axis in this framework, which means that the static torque cannot be balanced perfectly for all positions and loads. The static estimator therefore returns the maximum unbalanced static torque as its output. This information is important since it determines the maximum torque that the gearbox of the second axis needs to handle when the robot is standing still.

**4.4. Weight Estimation.** It is also desirable to estimate the weight of the balancing mechanism since it is desirable that the mechanism is light, compact, and inexpensive. A simple estimation is to multiply the material density with the material volume of the mechanism. The volume therefore needs to be estimated.

The three concepts have different weight equations for the contents inside the balancing cylinder. The mechanical concept has three springs that add metal volume. The other two concepts have fluids that add weight. The hydropneumatic concept also adds an accumulator that contributes to the weight.

The maximum pressure inside the cylinder can be used to estimate how thick the cylinder walls need to be. The cylindrical hoop stress is usually larger than the axial stress. Formulas for thin-walled assumptions can be used if the ratio between the cylinder diameter and the wall thickness is sufficiently large. The ratio is not large enough for these

applications, but it should give a decent estimation of the required cylinder wall thickness.

The analytical function for the cylindrical hoop stress can be seen in (8), where it is a function of the internal pressure, cylinder radius, and material thickness:

$$\sigma_{\theta} = \frac{Pr}{t}. \quad (8)$$

This stress should be lower than the yield strength of the material divided by a safety factor. This means that the required material thickness can be estimated according to (9)

$$t \geq \frac{PrF_{\text{safety}}}{\sigma_{\text{yield}}}. \quad (9)$$

## 5. Multidisciplinary Optimization of the Balancing Mechanism

The framework is implemented in the software modeFRONTIER [29] which is an integration platform for MOO and MDO. It has a few features that make it suitable for this optimization. The MOO algorithm NSGA-II is included in the software which means that the optimization algorithm itself handles the multiple objectives. It is also possible to impose a time-out on the dynamic simulation so it aborts if the calculations have been too long, indicating a numerical failure in the simulation of an infeasible design.

**5.1. Optimization Settings.** The optimization problem consists of multiple objectives. It is desirable to minimize the cycle time, CT, the energy consumption, EC, the maximum unbalanced static torque,  $T_{\text{unbal}}$ , the size of an eventual accumulator,  $V_0$ , and the weight, CW, and length, CL, of the balancing mechanism. This implies that the objective function in (10) is a combination of these entities. The size of the accumulator is only used as an objective for the hydropneumatic concept, whereas the weight is only an objective for the mechanical concept since the concepts that involve gas have a much smaller weight compared to it. A constraint is set to ensure that the lifetime of the gearbox, LT, of the second axis is higher than a reference value:

$$\begin{aligned} \min \quad & f(x) = \{CT, EC, T_{\text{unbal}}, CW, CL, V_0\} \\ \text{s.t.} \quad & g_1(x) = LT_{\text{min}} - LT \leq 0 \\ & x_j^L \leq x_j \leq x_j^U. \end{aligned} \quad (10)$$

The design variables and their limits can be seen in Table 1.

The mounting positions of the balancing mechanism are optimization variables for all concepts. The distance to point B is the sum of  $R_1$  and  $R_{12}$  to ensure that B lies further away from the rotational center than A. The mechanical concept also uses the dimensions of the springs as optimization variables. The concepts that involve gas instead use the initial gas pressure as an optimization variable. This means that the concepts that involve gas have five optimization variables, whereas the mechanical concept has 16.



TABLE 1: The optimization variables and their upper and lower limits.

Notation	Low. bound	Up. bound	Entity
$\alpha_1$	0	$\frac{\pi}{2}$	Angle for mounting A
$\alpha_2$	0	$\frac{\pi}{2}$	Angle for mounting B
$R_1$	0.15	0.3	Radius to A [m]
$R_{12}$	0.1	0.7	Extra distance to B [m]
$p_0$	50e5	200e5	Initial pressure [Pa]
$d_j$	0.015	0.03	Thread diameter of spring $j$ [m]
$D_j$			Middle diameter of spring $j$ [m]
$n_j$	6.5	10.5	Number of turns of spring $j$
$l_{0j}$	0.65	0.8	Unloaded length of spring $j$ [m]

This parametrization of the concepts that include gas works under the assumption that the maximum gas pressure and axial force in the piston are set to fixed values. The dimensions of the cylinder (and accumulator) can be derived from these two fixed values and the five optimization variables.

It would be possible to use the dimensions of the cylinder (and accumulator) as optimization variables instead, but the maximum gas pressure and axial force in the piston would have to be added as constraints then. The number of optimization variables would also increase and would together with the two additional constraints lead to a more complex optimization problem.

The chosen optimization algorithm is NSGA-II and it is run with 40 individuals and 100 generations. This means that 4000 designs need to be evaluated. This implies a total wall-clock-time of less than 70 hours with a time-out set to abort if the dynamic simulation takes more than one minute.

**5.2. Results.** Each concept has been optimized to demonstrate the capabilities of the framework.

The optimization problem is quite challenging and many designs lead to a bad balancing. This is shown in Figure 5, where the dynamic simulations that return errors are represented by red dots. Yellow dots represent solutions that violate the lifetime constraint whereas the green dots represent valid designs. The axes represent the angles  $\alpha_1$  and  $\alpha_2$  in Figure 2. These are just two of the optimization variables, but the important thing to notice is that there are red dots present among the green ones. This means that it is difficult to classify where the valid and invalid designs are. But the variable limits have at least been adjusted so that the areas with no feasible points have been removed from the design space. The starting generation has been seeded by a few feasible solutions to prevent the risk that the first generation contains no feasible solutions. The rest of the generation has been filled with a Latin Hypercube Sampling [30].

The Pareto front is exemplified for four characteristics for an optimization of the hydropneumatic concept in Figures 6 and 7. It can be seen that the unbalanced static torque decreases as the size of the accumulator increases. This is due to the fact that the gas pressure is assumed to follow a

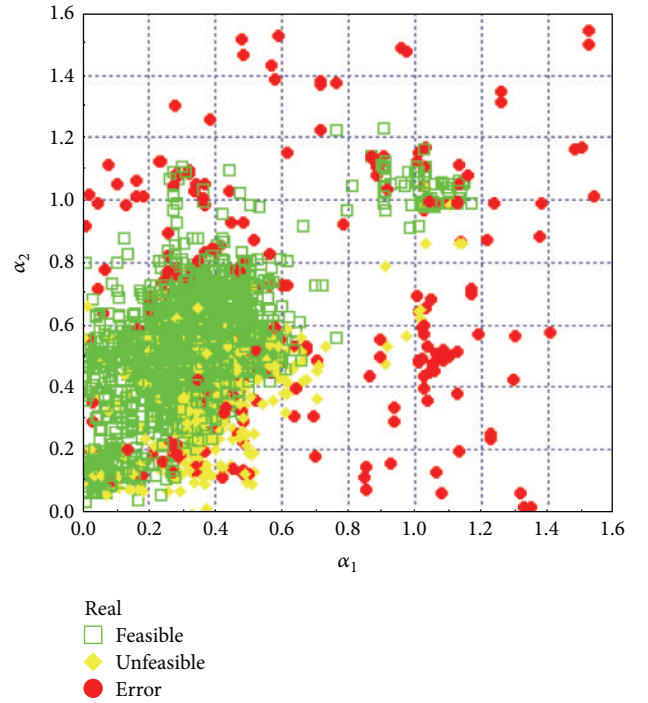


FIGURE 5: A scatter point that shows feasible, infeasible, and erroneous designs. These are coloured green, yellow, and red, respectively, and positioned according to the angles  $\alpha_1$  and  $\alpha_2$ .

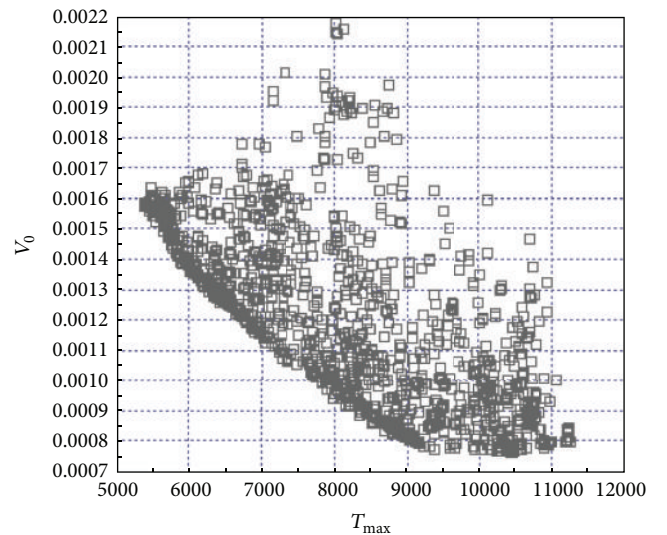


FIGURE 6: The Pareto front plotted as the volume of the accumulator versus the unbalanced static torque.

polytropic relation as seen in (7). A smaller accumulator leads to larger pressure differences, which means that it is more difficult to achieve an adequate balancing torque for all load cases. It can be seen in Figure 7 that the size of the balancing mechanism increases tremendously if a cycle time of close to, or less than, 0.97 of the reference cycle time is desired.

It is difficult for the human brain to process a Pareto front consisting of more than three objectives [31]. A large number

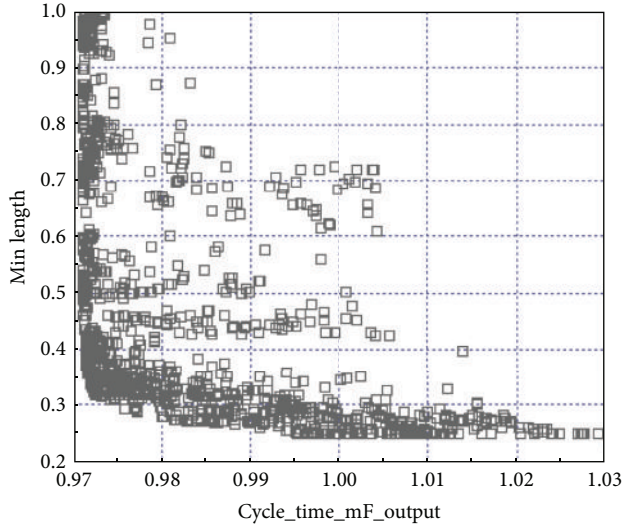


FIGURE 7: The Pareto front plotted as the robot cycle time versus the cylinder length.

TABLE 2: Percentage values of the balancing properties for three optimized concepts with respect to a reference balancing cylinder.

Property	Ref.	Mech.	Gas	HP
Cycle time	100	96	98	98
Cylinder length	100	110	115	79
Cylinder weight	100	123	14	18
Energy consumption	100	91	96	90
Lifetime second axis	100	187	135	190
Maximum axial force	100	139	129	119
Unbalanced static torque	100	92	91	88

of points may also be Pareto optimal for many objectives since a point just needs to be better than another point for one objective to be Pareto optimal. One representative of each concept has therefore been manually chosen from the database and presented in Table 2. They are compared to an existing balancing cylinder that is used as reference for this comparison.

It can be noted that the optimized concepts have a better balancing but this comes with higher maximum axial forces in the piston. These forces need to be handled by the bearings at the mounting positions, which therefore needs to be larger.

The optimized mechanical concept has better balancing than the reference, but this comes at the cost of a larger and heavier design.

The gas spring concept needs to use a long cylinder to get the desired balancing, which means that the end of the cylinder stretches further out from the second axis compared to the mechanical solution.

The hydropneumatic concept has great properties for every objective except the maximum axial force. It should however be pointed out that an accumulator needs to be added in the vicinity of the balancing mechanism. This needs a total gas volume of just above one litre, which corresponds to a radius of around seven centimetres for a spherical vessel.

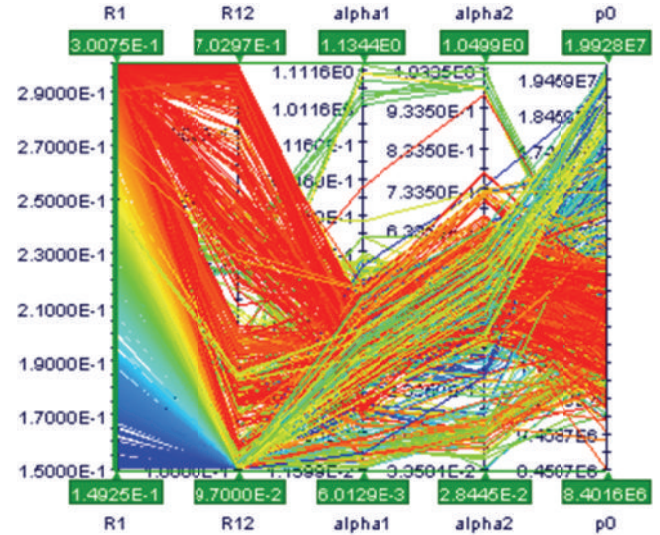


FIGURE 8: A parallel coordinate chart that displays the variable values of all Pareto optimal designs.

It is also interesting to investigate which variable values yield Pareto optimal points. This can be seen in Figure 8, where each line represents a Pareto optimal solution. The numbers on green background show the lower and upper variable limits that lead to these solutions. It can be seen that the whole design space contributes to the Pareto optimal points, with exceptions for the values of the angles of the mounting positions and the initial pressure. It seems that pressures below 100 Bar lead to almost no Pareto optimal points. The values of angles  $\alpha_1$  and  $\alpha_2$  are predominantly centered around 0.4 and 0.5. This indicates that the design space could be narrowed, for example, by raising the lower bound for the initial pressure to 80 Bar.

## 6. Conclusions

One optimization with the presented framework has been performed for each of the three concepts. All three optimal designs have better balancing properties than the reference balancing mechanism. All properties of the found solutions are credible which means that the framework can be used to optimize balancing mechanisms for industrial robots.

It is easy to switch between different concepts in the optimization framework. The only requirement is to replace the formula for how the torque varies with the position of the piston depending on the desired concept.

It is important to keep in mind that the concepts that involve gas come with disadvantages that the mechanical concept does not have. The gas pressure is critical for the functionality of the balancing mechanism and sudden pressure drops can be fatal. Additional maintenance may also be needed to counter leakage.

It is possible to use the framework to optimize the balancing of other robot models and duty cycles by changing the input files for the dynamic simulation and the geometrical

properties in the static estimator. This means that the framework can aid customer customization and the development of new robots.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## References

- [1] J. Andersson, *Multiobjective optimization in engineering design applications to fluid power systems [Ph.D. thesis]*, Linköping University, Linköping, Sweden, 2001.
- [2] J. R. R. A. Martins and A. B. Lambe, “Multidisciplinary design optimization: a survey of architectures,” *AIAA Journal*, vol. 51, no. 9, pp. 2049–2075, 2013.
- [3] R. Saravanan, S. Ramabalan, and P. Dinesh Babu, “Optimum static balancing of an industrial robot mechanism,” *Engineering Applications of Artificial Intelligence*, vol. 21, no. 6, pp. 824–834, 2008.
- [4] I. Simionescu and L. Ciupitu, “Static balancing of the industrial robot arms. Part I: discrete balancing,” *Mechanism and Machine Theory*, vol. 35, no. 9, pp. 1287–1298, 2000.
- [5] I. Simionescu and L. Ciupitu, “The static balancing of the industrial robot arms: part II: continuous balancing,” *Mechanism and Machine Theory*, vol. 35, no. 9, pp. 1299–1311, 2000.
- [6] Y. Xiao, J. Sun, T. Chen, and X. Fan, “Analysis and optimization for the counter-balancing method of heavy-load palletizing manipulators,” in *Proceedings of the International Conference on Advanced Mechatronic Systems (ICAMechS '13)*, pp. 716–719, IEEE, September 2013.
- [7] M. Vermeulen and M. Wisse, “Intrinsically safe robot arm: adjustable static balancing and low power actuation,” *International Journal of Social Robotics*, vol. 2, no. 3, pp. 275–288, 2010.
- [8] J. Weström, X. Feng, H. Andersson, and S. Lunderius, “Optimal spring balancing cylinder design of an industrial robot using multi-disciplinary and multi-objective optimization,” in *Proceedings of the 11th ASME Biennial Conference on Engineering Systems Design and Analysis*, pp. 133–139, American Society of Mechanical Engineers, Nantes, France, July 2012.
- [9] M. Tarkian, J. Persson, J. Ölvander, and X. Feng, “Multidisciplinary design optimization of modular industrial robots by utilizing high level CAD templates,” *Journal of Mechanical Design*, vol. 134, no. 12, Article ID 124502, 2012.
- [10] J. A. Persson, X. Feng, D. Wäppling, and J. Ölvander, “Multidisciplinary and multi-objective optimization of a hydro-pneumatic balancing cylinder for an industrial robot,” in *Proceedings of the 12th ASME Biennial Conference on Engineering Systems Design and Analysis (ESDA '14)*, Copenhagen, Denmark, July 2014.
- [11] J. Persson, X. Feng, D. Wäppling, and J. Ölvander, “Optimal design of a balancing system for a serial industrial robot,” in *Proceedings of the 14th Mechatronics Forum International Conference*, Karlstad, Sweden, June 2014.
- [12] S. R. Schmid, B. J. Hamrock, and B. O. Jacobson, *Fundamentals of Machine Elements*, CRC Press, 2013.
- [13] T. J. Viersma, *Analysis, Synthesis, and Design of Hydraulic Servosystems and Pipelines*, vol. 1 of *Studies in Mechanical Engineering*, Elsevier, Amsterdam, The Netherlands, 1980.
- [14] L. A. Schmit, “Structural design by systematic synthesis,” in *Proceedings of the 2nd ASCE Conference on Electronic Computation*, pp. 105–122, Pittsburgh, Pa, USA, 1960.
- [15] R. T. Haftka, “Automated procedure for design of wing structures to satisfy strength and flutter requirements,” Tech. Rep. NASA-TN-D-7264, NASA Langley Research Center, 1973.
- [16] E. J. Cramer, J. Dennis Jr., P. D. Frank, R. M. Lewis, and G. R. Shubin, “Problem formulation for multidisciplinary optimization,” *SIAM Journal on Optimization*, vol. 4, no. 4, pp. 754–776, 1994.
- [17] C. Bloebaum, P. Hajela, and J. Sobieszcanski-Sobieski, “Non-hierarchical system decomposition in structural optimization,” *Engineering Optimization*, vol. 19, no. 3, pp. 171–186, 1992.
- [18] R. D. Braun, *Collaborative optimization: an architecture for large-scale distributed design [Ph.D. thesis]*, Stanford University, Stanford, Calif, USA, 1996.
- [19] J. Sobieszcanski-Sobieski, J. Agte, and R. Sandusky, “Bi-level integrated system synthesis (BLISS),” in *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, Mo, USA, September 1998.
- [20] S. Tosserams, L. F. Etman, and J. E. Rooda, “Augmented Lagrangian coordination for distributed optimal design in MDO,” *International Journal for Numerical Methods in Engineering*, vol. 73, no. 13, pp. 1885–1910, 2008.
- [21] V. DeMiguel and W. Murray, “A local convergence analysis of bilevel decomposition algorithms,” *Optimization and Engineering*, vol. 7, no. 2, pp. 99–133, 2006.
- [22] M.-K. Shin and G.-J. Park, “Multidisciplinary design optimization based on independent subspaces,” *International Journal for Numerical Methods in Engineering*, vol. 64, no. 5, pp. 599–617, 2005.
- [23] R. T. Haftka and L. T. Watson, “Multidisciplinary design optimization with quasiseparable subsystems,” *Optimization and Engineering*, vol. 6, no. 1, pp. 9–20, 2005.
- [24] I. R. Chittick and J. R. R. A. Martins, “An asymmetric suboptimization approach to aerostructural optimization,” *Optimization and Engineering*, vol. 10, no. 1, pp. 133–152, 2009.
- [25] A. Forrester, A. Söbester, and A. Keane, *Engineering Design via Surrogate Modelling: A Practical Guide*, John Wiley & Sons, 2008.
- [26] K. Deb, S. Agarwal, A. Pratap, and T. Meyarivan, “A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II,” in *Parallel Problem Solving from Nature PPSN VI*, vol. 1917 of *Lecture Notes in Computer Science*, pp. 849–858, Springer, Berlin, Germany, 2000.
- [27] J. Andersson, “A survey of multiobjective optimization in engineering design,” Tech. Rep. lith-ikp-r-1097, Department of Mechanical Engineering, 2000.
- [28] Mathworks, 2014, <http://www.mathworks.se/products/matlab/>.
- [29] ESTECO, 2014, <http://www.esteco.com/modelfrontier>.
- [30] M. D. McKay, R. J. Beckman, and W. J. Conover, “Comparison of three methods for selecting values of input variables in the analysis of output from a computer code,” *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.
- [31] A. V. Lotov and K. Miettinen, “Visualizing the Pareto frontier,” in *Multiobjective Optimization*, vol. 5252 of *Lecture Notes in Computer Science*, pp. 213–243, Springer, Berlin, Germany, 2008.





**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

