

Supplementary Material for “Computational Design of Planar Multistable Compliant Structures”

RAN ZHANG, IST Austria, Austria
 THOMAS AUZINGER, IST Austria, Austria
 BERND BICKEL, IST Austria, Austria

This document provides additional results and analyzes the robustness and limitations of our approach. Please refer to the main article for the details of our algorithm.

Section 1 shows the result for an EXTENDED FLAG SEMAPHORE example with 4 end effectors (2 fixed end effectors) and 6 stable poses. With this example, we demonstrate the quality of the result for an input with more than 3 stable poses. Section 2 shows an ASYMMETRIC ARMS example with different scales for the lengths of edges in two end effector groups. This example demonstrates the robustness of our algorithm on an input with different scales of edge lengths. In section 3, we tested our algorithm on a BEETLE example with 4 stable poses and 6 end effector groups, and analyzed the performance in the presence of more than 4 end effector groups. Section 4 presents an EXTENDED FIGURINE example similar to the figurine example in the main article but with 4 stable poses. Finally, we discuss the layering post processing and problems we encountered when solving the layering for complicated cases. Please see the detailed statistics in Table 1. All the tests were computed on a regular workstation with an quad-core Intel Xeon E3-1230 v5 and 32GB RAM.

Table 1. Results statistics. We provide the geometric complexity of both in- and output as vertex and edge counts (#V and #E) as well as the target approximation and rigidity energies E_{tar} and E_{rig} of each result. The maximal deviation of any end effector joint from its user-specified target position is presented with Δ_{max} [mm] and the maximum dimension of the input model is D_{max} [mm]. The last column reports the runtime of our design method.

Result	Poses	End effector		Coupler		Switch		E_{tar}	E_{rig}	Δ_{max}	D_{max}	Time
		#V (fixed)	#E	#V	#E	#V	#E					
EXTENDED FLAG SEMAPHORE	6	4 (2)	2	10	14	10	7	16.53	10.97	0.47	100	53 s
ASYMMETRIC ARMS	5	6 (4)	4	14	20	8	6	6598.58	61.87	9.76	263	4 min
BEETLE	4	12 (6)	6	29	30	6	5	5253.02	116.94	9.26	153	1.5 h
EXTENDED FIGURINE	4	12 (4)	8	29	42	6	5	5641.90	57.09	9.20	211	38 min

Authors' addresses: Ran Zhang, IST Austria, Am Campus 1, 3400, Klosterneuburg, Austria, rzhang@ist.ac.at; Thomas Auzinger, IST Austria, Am Campus 1, 3400, Klosterneuburg, Austria, thomas.auzinger@ist.ac.at; Bernd Bickel, IST Austria, Am Campus 1, 3400, Klosterneuburg, Austria, bernd.bickel@ist.ac.at.

1 EXTENDED FLAG SEMAPHORE

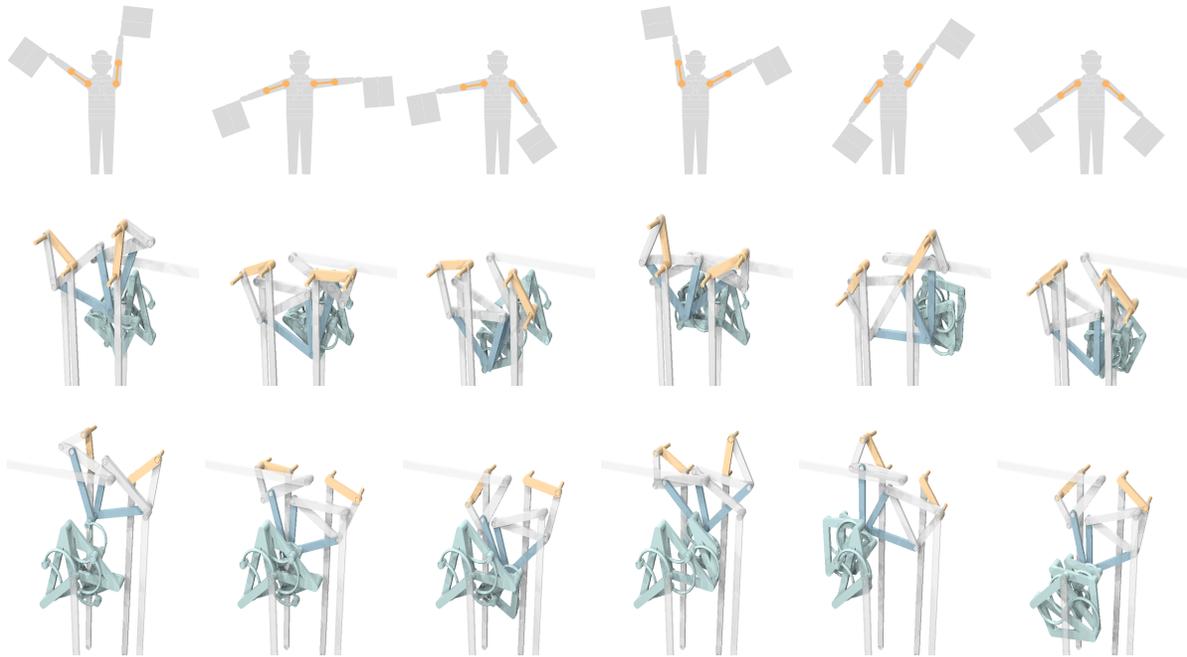


Fig. 1. EXTENDED FLAG SEMAPHORE result (top row: input poses (gray) and generated end effector positions in generated graph realizations, mid row: rendered from front view, bottom row: rendered from back view).

This example demonstrates the result of our algorithm on a 6-stable flag semaphore input. Although, in comparison to the examples shown in the main article, this example contains a larger number of poses, our result demonstrates that we can achieve close approximations to the desired poses. The required processing time was 53 seconds.

2 ASYMMETRIC ARMS

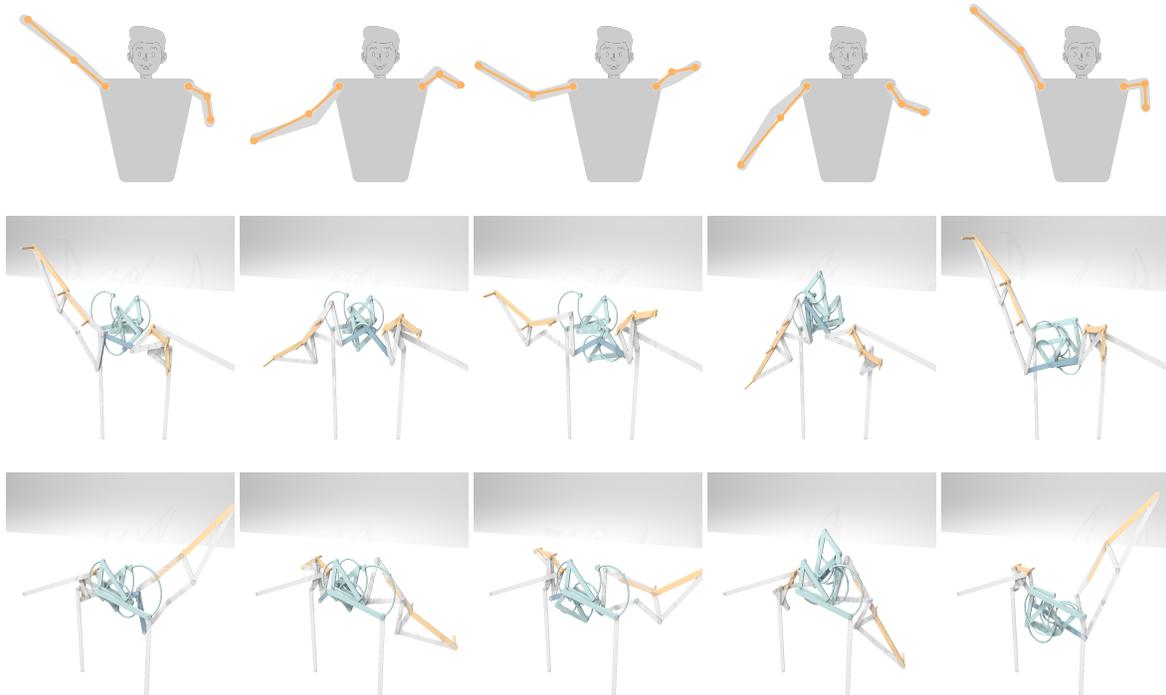


Fig. 2. ASYMMETRIC ARMS result (top row: input poses (gray) and generated end effector positions in generated graph realizations, mid row: rendered from front view, bottom row: rendered from the back view).

This example demonstrates the result of our algorithm on a 5-stable-6-end effector input, with asymmetric length of edges in the input vary for different end effector groups. The result shows close approximation to the input stable poses with a maximum deviation of 9.76mm. The longest dimension of the input is 263mm. The optimization required 4 minutes computational time. In our tests, we did not notice that the scales of different end effector groups influence the quality of the results in the sense of numerical deviation.

3 BEETLE

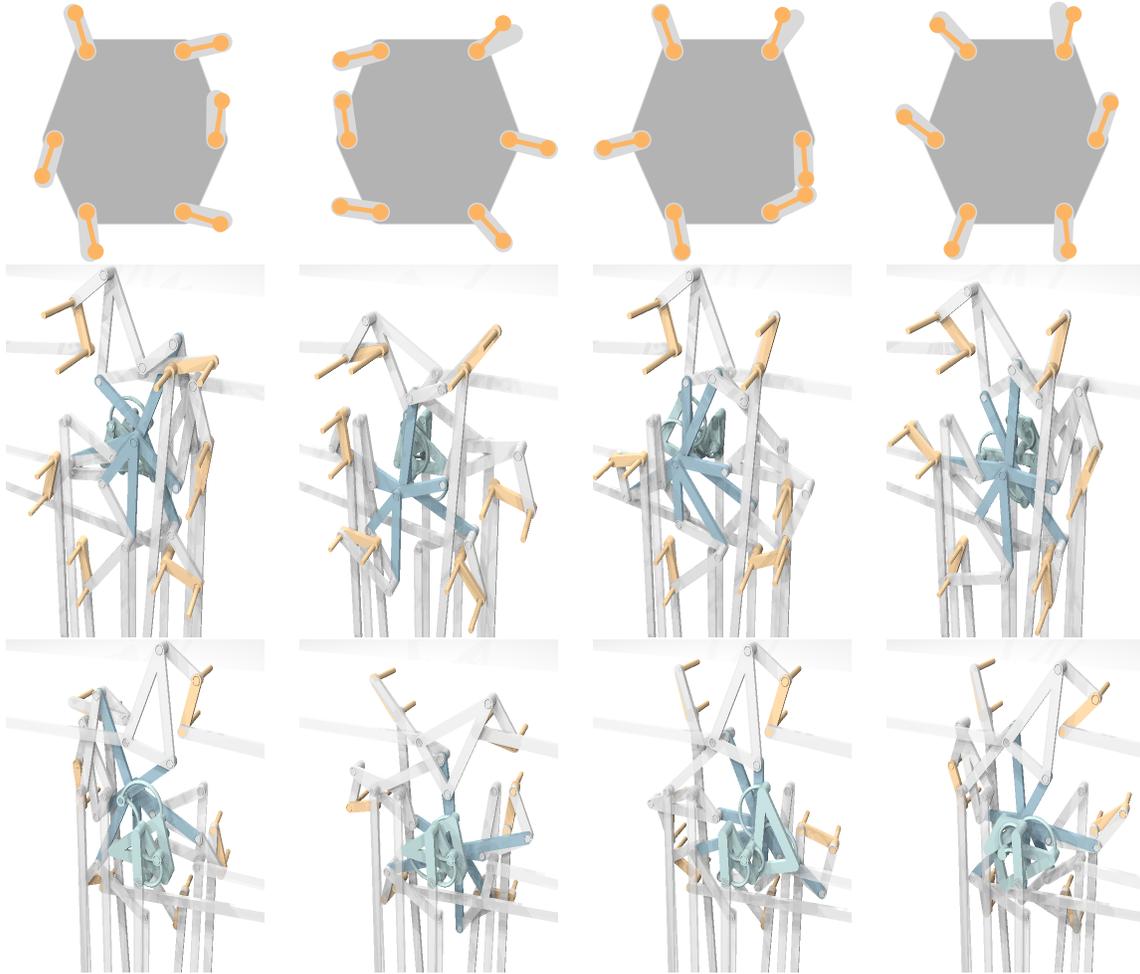


Fig. 3. BEETLE result (top row: input poses (gray) and generated end effector positions in generated graph realizations, mid row: rendered from front view, bottom row: rendered from the back view).

This example demonstrates the result of our algorithm on a 4-stable-6-end effector-group input. This result shows visually plausible approximations to the input poses with a maximum deviation of 9.26mm. The longest dimension of the input model is 153mm. Among all the end effectors, we observe one with a larger deviation, while the rest provides a close approximation of the target. The deviation is due to the increased number of end effector groups. In comparison, this is not as accurate as in the other examples because it has more end effector groups.

Regarding the computational time, the complexity of our joint optimization is exponential with respect to the total number of end effector groups.

4 EXTENDED FIGURINE

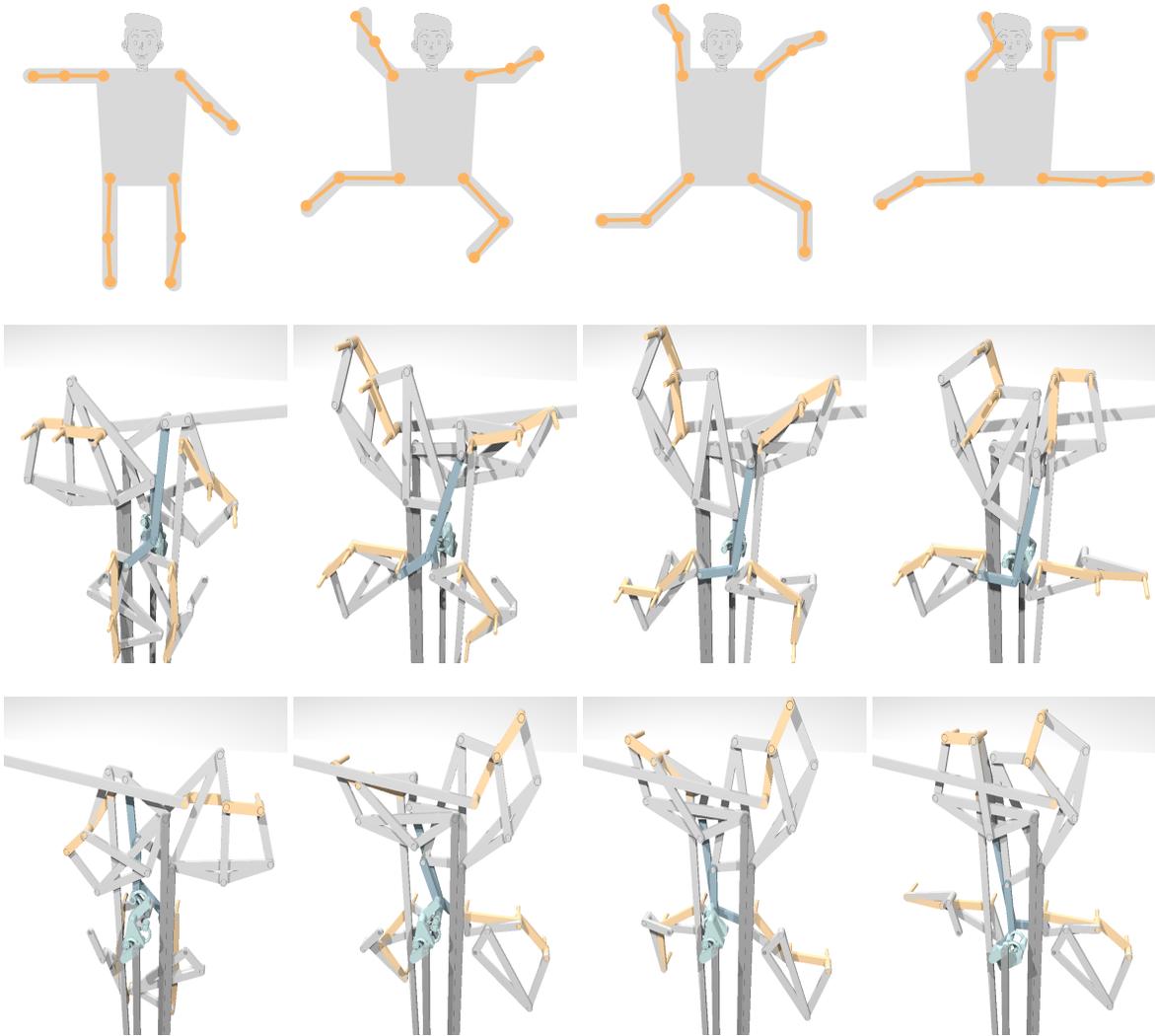


Fig. 4. EXTENDED FIGURINE result (top row: input poses (gray) and generated end effector positions in generated graph realizations, mid row: rendered from front view, bottom row: rendered from the back view).

This example demonstrates the result of our algorithm on a 4-stable figurine example input. This result shows a more practical case with close approximations to the input poses with 38 minutes computational time.

5 REGARDING THE LAYERING PROBLEM

5.1 Algorithm

In order to make a generated bar-joint mechanism properly work in 3D space, we have to solve for a layer assignment of each bar in the bar-joint mechanism, and ensure that no collision happens in the movements of the mechanism.

Typically, this process is formulated as a Constrained Satisfaction Problem(CSP), we follow the same formal definition of the CSP problem and compose our optimization problem. The variables of our CSP problem consist of an individual layer x_i for every bar, a top layer t_j and a bottom layer b_j spanned by each of the joint axes in the bar-joint mechanism. We first calculated the bar-bar collisions and bar-joint collisions in the forward simulation of the bar-joint mechanism assuming all of the bars are located on the same layer. Based on the collisions information, we build the collision constraints for the CSP problem to ensure that there is no collision for the optimized layer arrangements of the mechanism. Besides the collision constraints, we also consider two boundary conditions. The end effector joints connect to the topmost layer where the input end effector bars are located. On the other hand, the fixed joints connect to the bottommost layer so that they are fixed to the pedestal of the model. The complete mathematical definition of the CSP problem is presented below.

$$\begin{aligned}
 & \arg \min_{\{x_0, \dots, x_{n-1}, t_0, \dots, t_{m-1}, b_0, \dots, b_{m-1}\} \in \mathbb{N}^{n+2m}} \max(x_0, \dots, x_{n-1}) \\
 \text{s.t. } & \begin{cases} x_i \neq x_j, & \text{if } i\text{-th bar collides with } j\text{-th bar} \\ (x_i < l_k) \vee (x_i > u_k), & \text{if } i\text{-th bar collides with } k\text{-th joint} \\ u_k = \max_{i \in \{\text{bars connected with } k\text{-th joint}\}} (x_i) & \text{if } k\text{-th joint is not an end effector joint} \\ l_k = \min_{i \in \{\text{bars connected with } k\text{-th joint}\}} (x_i) & \text{if } k\text{-th joint is not a fixed joint} \\ u_k = \max(x_i) + 1 & \text{if } k\text{-th joint is an end effector joint} \\ l_k = 0 & \text{if } k\text{-th joint is a fixed joint} \end{cases} \quad (1)
 \end{aligned}$$

This CSP problem can be solved with various Constraint Programming-Satisfaction SATisfaction(CP-SAT) solvers, where we employed the solver provided by Google’s OR-Tools [Perron and Furnon 2019].

5.2 Observation

Solving our CSP problem with a CP-SAT solver is fast since the structures only require at most decades of layers. However the inputs (i.e. the realizations of the bar-joint mechanisms) of the solver determine the existence of the solutions. We categorize the results of our algorithm into different types:

- (1) A valid layering is found without adding additional supporting structures to the pedestal (see the GRIPPER result in the main paper).
- (2) A valid layering is found by adding additional supporting structures to the pedestal (See the FLAG SEMAPHORE, FIGURINE, ICONIC POSES results in the main paper and all the results in this supplemental material).
- (3) A valid layering is found without the boundary condition constraints of end effector joints and fixed joints.
- (4) No valid layering found even without the boundary condition constraints of end effector joints and fixed joints.

According to our test with a large number of different inputs, we observed the situations below,

- All the inputs can get valid layerings when removing boundary condition constraints of end effector joints and fixed joints (type 3).

- In the `FLAG SEMAPHORE` example, most of the inputs can get valid layering without adding additional supporting structures (type 1), all the other inputs can get valid layering with adding additional supporting structures (type 2).
- For the more complicated examples, the results largely depend on the inputs, if the input end effector groups are close to each other or have overlapping regions, the solver is most likely to fail (type 3). If the input end effector groups are sufficiently distant from each other, the solver often gets valid layering with adding additional supporting bars (type 2).