Computer-Aided Kinematic Design of a Torsional Ratcheting Actuator

Elisha Sacks
Computer Science Department
Purdue University
West Lafayette, IN 47907, USA
eps@cs.purdue.edu
http://www.cs.purdue.edu/visctr/cad.html

Stephen M. Barnes
Intelligent Micromachine Department
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185, USA
smbarne@sandia.gov, http://mems.sandia.gov

Abstract

We have developed an interactive computer-aided design program that supports the mechanical design of devices fabricated in surface micro-machining processes. The program automates kinematic analysis via a novel configuration space computation code, performs real-time simulation and kinematic tolerance analysis, and supports kinematic synthesis. Designers can visualize system function under a range of operating conditions, can find and correct design flaws, and can optimize performance. We used the program to redesign and validate a torsional ratcheting actuator fabricated at Sandia with the SUMMiT V process.

Keywords: MEMS, micro-mechanical design, configuration space, Summit.

1 Introduction

We have developed an interactive computer-aided design program that supports the mechanical design of devices fabricated in surface micro-machining processes, such as the Sandia SUMMiT technology. Typical devices include gears, ratchets, and transmissions [1]. Surface micro-machining poses significant design and analysis challenges that do not arise in traditional, macroscopic devices. It is a batch fabrication method, so mechanical elements must be fabricated in place, which subjects their clearances and tolerances to the limitations of the fabrication process. The clearance and tolerance effects fall into three categories: lithography line width and space rules, sacrificial oxide spacing, and unilateral size tolerances on the mechanical elements due to release processing. These factors influence the device kinematics and dynamics in complex ways. Comprehensive computer modeling is needed to assure correct function without excessive prototyping, which is extremely slow and expensive.

Traditional computer-aided design softwares is inappropriate for micro-mechanism design because of the curved geometry, contact changes, and large clearances. VLSI design software is not meant for moving parts. Finite element analysis is difficult and slow due to the large number of parts, the curved geometry, and the many part contact changes. It is computational overkill for designs that can be modeled as rigid-body systems. Mechanical system simulators offer an efficient alternative to finite element codes, but are limited to systems with permanent part contacts, such as pin joints, prismatic joints, and involute gears. These conditions are unrealistic for micro-mechanisms because the fabrication process cannot produce ideal joints and because contact changes (higher kinematic pairs) are common.

Our program automates kinematic analysis [2] [3], dynamical simulation [4], tolerance analysis [5], and kinematic synthesis [6]. Designers can visualize system function under a range of operating conditions, can find and correct design flaws, and can optimize performance. The program handles planar systems of curved, rigid parts with custom pairs, open and closed kinematic chains, and contact changes. An early version, which lacked tolerancing and synthesis modules, was used to analyze a MEMS indexing pair [7].

In this paper, we show how our design program supports the design of a complex micro-mechanism using the Sandia SUMMiT V surface micro-machining process. The mechanism is a torsional ratcheting actuator: a novel electrostatic actuator that is compact, easy to operate, and provides accurate position control without feedback [8]. The mechanism was redesigned to increase robustness and to optimize performance. The design program provided comprehensive performance evaluations, detected potential design flaws, and interactively computed modifications that repair the flaws. These functions reduce design time by an order of magnitude and improve design quality.

2 Torsional ratcheting actuator

Figure 1 shows the original torsional ratcheting actuator. The mechanism consists of a driver, a ratchet, a ring gear, and an anti-reverse. The gear and the anti-reverse are mounted on the substrate with pin joints, while the ratchet is attached to the driver with a pin joint. (These are really flexures; pin joints are an accurate, simple approximation.) The driver is attached to the substrate by springs that allow planar rotation, but prevent translation. The driver is rotated 2.5° counterclockwise by an electrostatic comb drive. The ratchet engages the inner teeth of the gear and rotates it counterclockwise. When the voltage drops, the springs restore the driver to its start orientation, which disengages the ratchet. The anti-reverse prevents the gear from rotating clockwise. Its outer, involute teeth drive an external load.

The main purpose of the redesign is to ensure that the gear advances by one tooth per cycle. The original design exhibits overrun where the ratchet/gear impact causes the gear to jump two or more teeth. An anti-overrun and a riser were

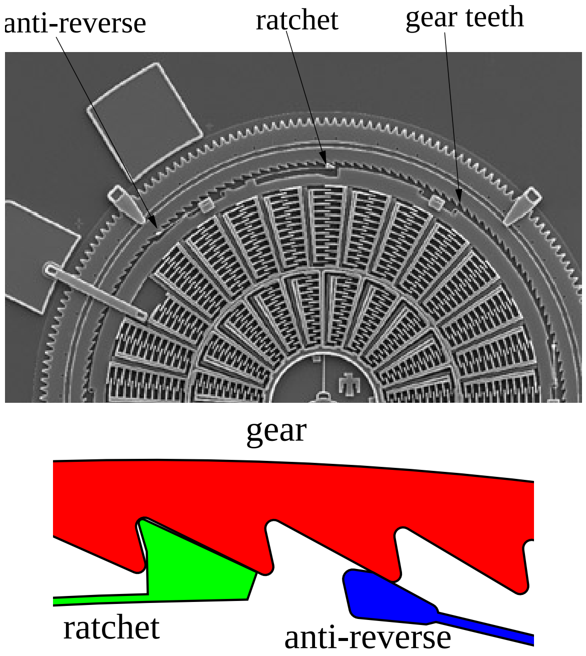

Figure 1: Torsional ratcheting actuator: (top) SEM image courtesy of Sandia National Laboratories Intelligent Micromachine Initiative www.mems.sandia.gov; (bottom) Detail of CAD model.

added to the design to prevent this. A second goal is to ensure correct function despite gear hub play, which is estimated at 0.5 microns. A third goal is to optimize performance factors, including voltage, power, wear, and area. This paper addresses the first two goals, which mostly involve the kinematic function of the mechanism.

Figure 2 illustrates the modified design. The riser is fixed to the substrate and the anti-overrun is attached to the driver with a pin joint. It is rotated $2.375°$ counterclockwise by the electrostatic comb drive (a–c). The ratchet rotates with the driver, engages the inner teeth of the gear (b), and rotates it counterclockwise (c). The anti-overrun rotates with the driver, rides over the riser (b), engages the gear (c), and prevents further gear motion. When the voltage drops, torsional springs restore the drive wheel to its start orientation, which disengages the ratchet and the anti-overrun (d–e). As before, the anti-reverse prevents the gear from rotating clockwise.

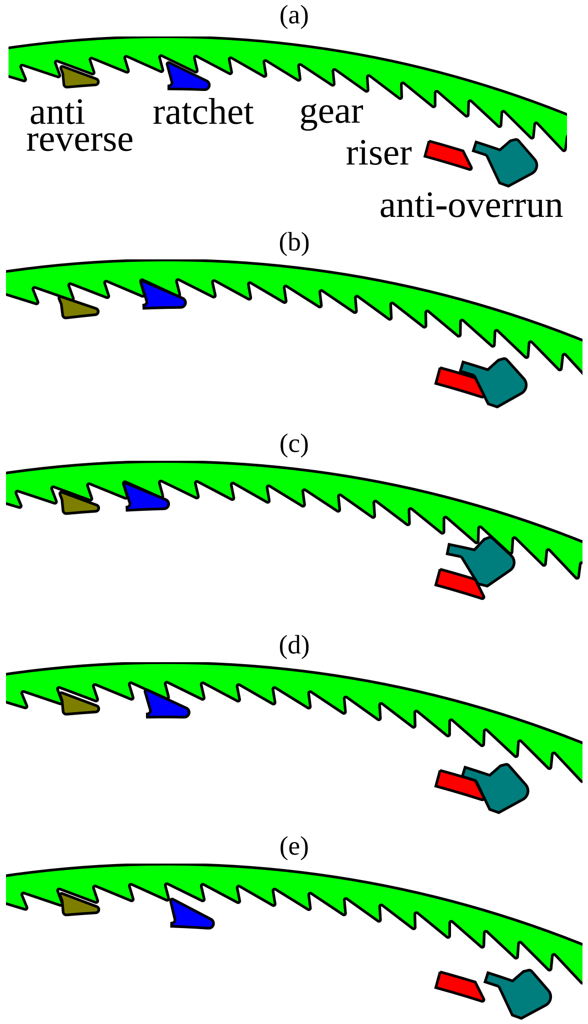

Figure 2: Simulation of modified design.

3 Configuration space

Kinematic analysis is performed by constructing configuration spaces for the interacting pairs of parts. These spaces encode the part interactions in a uniform geometric format that supports function validation, simulation, tolerancing, and synthesis. The configuration space of a pair is a manifold with one coordinate per part degree of freedom. A general planar pair has a three-dimensional configuration space whose coordinates are the position and orientation of one part relative to the other. A pair of rotating parts can be modeled with a simpler, two-dimensional space whose coordinates are the part orientation angles. Although our program handles both cases, we will discuss only the simpler case.

Figure 3 shows the gear/anti-reverse configuration space. The coordinates are the gear orientation g and the anti-reverse orientation a in degrees. The dot marks the displayed configuration where the parts are engaged. The part contacts are encoded by partitioning configuration space into three disjoint classes: blocked space (grey) where the parts overlap, free space (white) where they do not touch, and contact space (black) where they touch without overlap. Blocked space represents the illegal configurations, free space represents the independent part motions, and contact space represents motion constraints induced by part contacts. Contact space consists of many short contact curves that represent contacts between pairs of part features (points, line segments, and arcs). The displayed configuration lies at the meeting point of two contact curves.

The configuration space shows that the pair functions correctly. The slanted contact curve to the right of the displayed

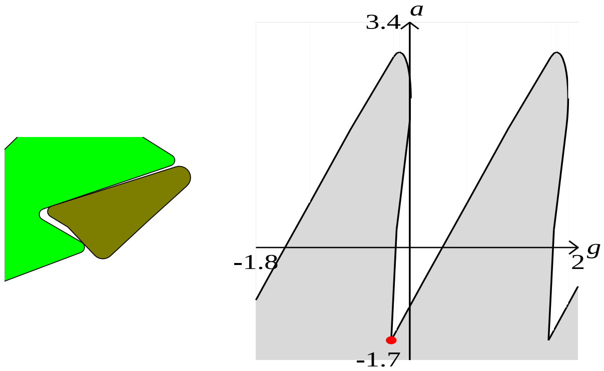

Figure 3: Gear/anti-reverse configuration space.

configuration represents the contact between the long side of a gear tooth and the top of the anti-reverse. If the gear rotates counterclockwise (g increases), this contact causes the anti-reverse to rotate counterclockwise (a increases), as the configuration of the pair follows the contact curve. The near vertical contact curve to the left of the displayed configuration represents the contact between the short side of a gear tooth and the side of the anti-reverse. This contact prevents the gear from rotating clockwise, since every direction in which g decreases lies in blocked space.

The gear/ratchet pair has a three-dimensional configuration space because the ratchet rotates around a rotating point, hence has three degrees of freedom. The interactions are invariant when both parts are rotated by the same angle, so we can analyze the pair in a two-dimensional space whose coordinates are the angle between the parts and the ratchet orientation. Equivalently, we freeze the driver at zero degrees, which also freezes the ratchet center of rotation.

Figure 4 shows the configuration space. The coordinates are the gear orientation g and the ratchet orientation r. The dot marks the displayed configuration where the ratchet is driving the gear. The near vertical contact curve to the left represents the contact between the short side of a gear tooth and the ratchet tip, which prevents the gear from rotating clockwise relative to the driver. The configuration space reveals a design flaw: the curve slope is positive (and very large), which implies that the gear can rotate clockwise, escape the ratchet, and jump to the next tooth. Friction will prevent this until the driver torque reaches a critical value. Load variation and hub play can exacerbate the problem.

4 Simulation

Configuration space analysis reveals the kinematic constraints due to contacts among the parts of the actuator. We combine this analysis with rigid body simulation to predict the system function. Kinematic analysis is a prerequisite for simulation because contacts create forces that effect part motion. We have developed a simulator that computes the configuration spaces of the interacting pairs before the simulation and queries them at each time step for part collisions and contact changes [4]. When these occur, the simulator backs up to the change time and updates the contact equations. Running on a personal computer, the simulator handles systems with tens of moving parts at interactive speeds. It generated the simulation shown in Figure 2, which validates the initial nominal design. It also validated the revised design described below.

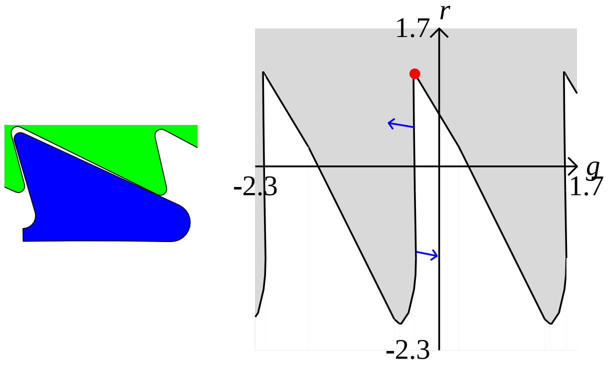

Figure 4: Gear/ratchet configuration space.

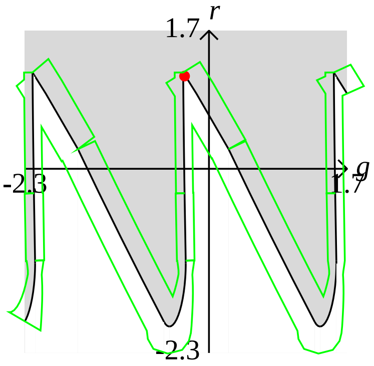

Figure 5: Gear/ratchet contact zone.

5 Tolerance analysis

Our analysis ignores shape variation and hub play. The dominant factor is gear hub play, which is estimated at 0.5 microns. Flexures have minimal play and shape variations are negligible in the SUMMiT V process. We model the gear hub play by parameterizing the center of rotation (x_r, y_r) and allowing these parameters to vary in the interval $[-0.5, 0.5]$. Each choice of parameter values generates a different part with its own gear/ratchet, gear/anti-reverse, and gear/anti-overrun configuration spaces. We compute zones around the nominal contact spaces, called contact zones, that bound their worst-case variation [9] [10]. Figure 5 shows the gear/ratchet contact zone, which is displayed as light bands around the contact curves. The contact curves are guaranteed to lie in the zone whenever the parameters lie in the specified interval.

The gear/ratchet contact zone shows that hub play can

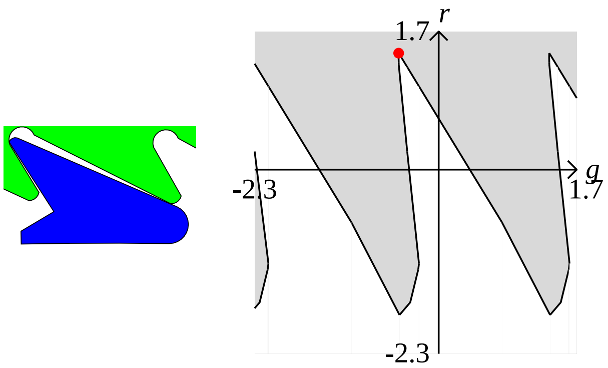

Figure 6: Modified gear/ratchet configuration space.

cause incorrect function. The near vertical curve, which is meant to prevent jumping, can slope to the left without leaving the contact zone. The gear will then be able to rotate clockwise and jump. The gear/anti-reverse and gear/anti-overrun contact zones show correct function despite hub play.

6 Synthesis

We modified the gear/ratchet pair to prevent jumping, using our interactive kinematic synthesis program [6]. We specified the kinematic changes by mapping nominal contact configurations to desired configurations. In Figure 4, the nominal configurations are the base points of the two arrows and the desired configurations are the arrow heads. The program modifies the design parameters to achieve the changes, while preventing unintended changes. Figure 6 shows the modified pair. The gear tooth slope and the matching ratchet slope are larger. In configuration space, the near horizontal contact has rotated enough that its slope is negative, and stays negative despite hub play. The gear/ratchet modifications alter the kinematics of the other pairs. We validated the entire mechanism via configuration space computation, simulation, and tolerance analysis.

7 Conclusions

We have presented configuration space methods of kinematic analysis, tolerancing, and synthesis. We have demonstrated their efficacy (and our implementation) on the redesign of a torsional ratcheting actuator that is fabricated in the Sandia SUMMiT V process. We plan to exploit this methodology to validate future MEMS designs electronically, thereby reducing prototyping effort and design time.

Acknowledgments

Sacks is supported in part by NSF grants IIS-0082339 and CCR-9617600, a Ford Motors Research Grant, the Israeli Academy of Sciences, and the Purdue Center for Computational Image Analysis and Scientific Visualization. Barnes is now employeed at MEMX, Inc (barnes@memx.org). Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

REFERENCES

[1] J. J. Sniegowski, S. M. Miller, G. F. LaVigne, M.S. Rodgers, and P. J. McWhorter. Monolithic geared-mechanisms driven by a polysilicon surface-micromachined on-chip electrostatic microengine. In *Solid-State Sensor and Actuator Workshop*, pages 178–182, Hilton Head Is., South Carolina, 1996.

[2] Elisha Sacks and Leo Joskowicz. Computational kinematic analysis of higher pairs with multiple contacts. *Journal of Mechanical Design*, 117(2(A)):269–277, June 1995.

[3] Elisha Sacks. Practical sliced configuration spaces for curved planar pairs. *International Journal of Robotics Research*, 18(1):59–63, January 1999.

[4] Elisha Sacks and Leo Joskowicz. Dynamical simulation of planar systems with changing contacts using configuration spaces. *Journal of Mechanical Design*, 120(2):181–187, June 1998.

[5] Elisha Sacks and Leo Joskowicz. Parametric kinematic tolerance analysis of general planar systems. Technical Report 97-027, Purdue University, 1997. To appear in *Computer-Aided Design*.

[6] Min-Ho Kyung and Elisha Sacks. Computer-aided synthesis of higher pairs via configuration space manipulation. Technical report, Purdue University, 2000.

[7] Elisha Sacks and James Allen. Mems functional validation using the configuration space approach to simulation and analysis. In *Second International Conference on Modeling and Simulation of Microsystems, Semiconductors, Sensors, and Actuators*, 1999.

[8] Stephen M. Barnes, Samuel L. Miller, M. Steven Rodgers, and Fernando Bitsie. Torsional ratcheting actuation system. In *Third International Conference on Modeling and Simulation of Microsystems*, San Diego, CA, 2000.

[9] Elisha Sacks and Leo Joskowicz. Parametric kinematic tolerance analysis of general planar systems. *Computer-Aided Design*, 30(9):707–714, August 1998.

[10] Leo Joskowicz and Elisha Sacks. Kinematic tolerance analysis with configuration spaces. In *Workshop on the Algorithmic Foundations of Robotics*, Dartmouth, NH, 2000.