Kinergy: Creating 3D Printable Motion using Embedded Kinetic Energy

Liang He
Paul G. Allen School of Computer Science & Engineering, University of Washington
lianghe@cs.washington.edu

Xia Su
Paul G. Allen School of Computer Science & Engineering, University of Washington
xiasu@cs.washington.edu

Huaishu Peng
Computer Science, University of Maryland
huaishu@cs.umd.edu

Jeffrey I. Lipton
Mechanical Engineering, University of Washington
jilipton@uw.edu

Jon E. Froehlich
Paul G. Allen School of Computer Science & Engineering, University of Washington
jonf@cs.uw.edu

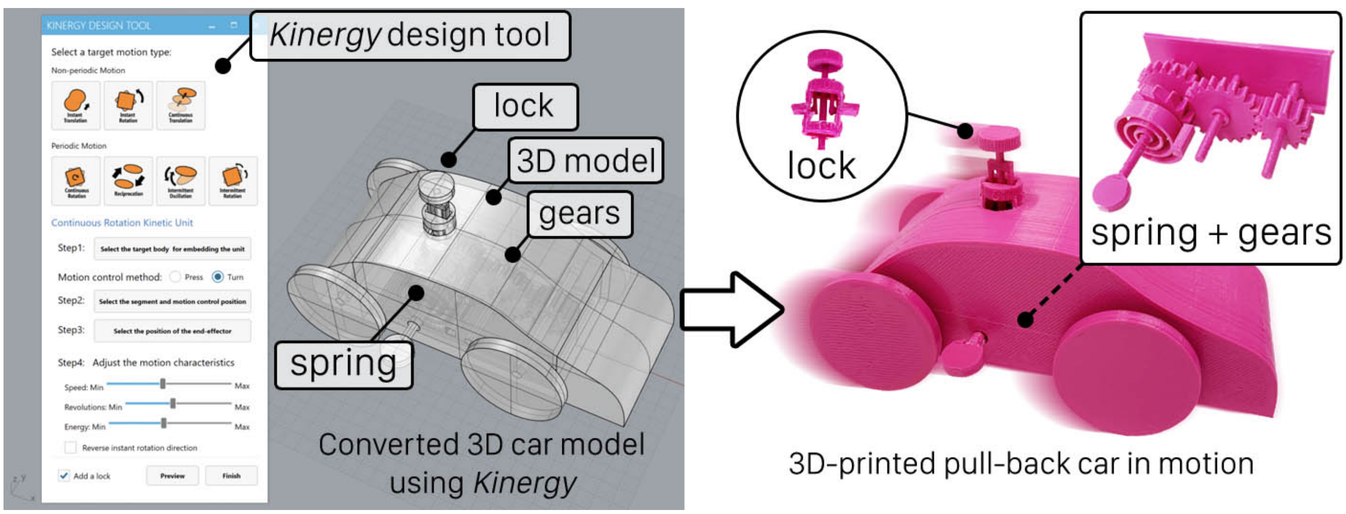


Figure 1: We introduce *Kinergy*, an interactive design tool to rapidly create 3D printable energy-powered motion. Above, we show a 3D-printed pull-back car created with *Kinergy*: the static car model is converted into a motion-enabled model with an auto-generated and embedded spring, a spring lock, and a set of gears (left). All the parts in the converted 3D car model are printed in place with a commercial 3D printer and the printed car is ready to move without post-print part assembly (right).

ABSTRACT

We present *Kinergy*—an interactive design tool for creating self-propelled motion by harnessing the energy stored in 3D printable springs. To produce controllable output motions, we introduce 3D printable *kinetic units*, a set of parameterizable designs that encapsulate 3D printable springs, compliant locks, and transmission mechanisms for three non-periodic motions—*instant translation, instant rotation, continuous translation*—and four periodic motions—*continuous rotation, reciprocation, oscillation, intermittent rotation.* *Kinergy* allows the user to create motion-enabled 3D models by embedding kinetic units, customize output motion characteristics by parameterizing embedded springs and kinematic elements, control energy by operating the specialized lock, and preview the resulting motion in an interactive environment. We demonstrate the potential of our techniques via example applications from spring-loaded cars to kinetic sculptures and close with a discussion of key challenges such as geometric constraints.

CCS CONCEPTS

• **Human-centered computing** → Human computer interaction (HCI); Interactive systems and tools.

UIST '22, October 29–November 02, 2022, Bend, OR, USA

ACM ISBN 978-1-4503-9320-1/22/10.
https://doi.org/10.1145/3526113.3545636

KEYWORDS

Digital fabrication, 3D printing, kinetic objects, spring, gear, computer-aided design, mechanical lock, parametric design

ACM Reference Format:

Liang He, Xia Su, Huaishu Peng, Jeffrey I. Lipton, and Jon E. Froehlich. 2022. Kinergy: Creating 3D Printable Motion using Embedded Kinetic Energy. In *The 35th Annual ACM Symposium on User Interface Software and Technology (UIST '22), October 29–November 02, 2022, Bend, OR, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3526113.3545636

1 INTRODUCTION

3D printable kinematics opens new exciting opportunities for 3D printing: designers can create and control printable *movement* enabling a wide variety of applications from 3D printable rotational solar system models[1] to robotic quadrupeds[2]. Despite substantial recent work on 3D printable kinematic elements [2, 5, 45, 47] and actuation methods [3, 19], designing fully-functional, 3D printable kinematic objects with controllable movements remains challenging. Expert skills, time, and labor are needed to design kinematic models [9, 38], assemble multiple 3D-printed parts, and interface components with external power sources for actuation [12, 35, 41]. To address these challenges, we introduce *Kinergy*—an open-sourced[3] interactive design tool to create, control, and print 3D objects with self-propelled motions.

Central to *Kinergy*'s approach is a suite of custom, 3D printable mechnical structures called *kinetic units*, which encapsulate complex kinematic and parametric mechanisms as "black box" units, abstracting the design and control of complex 3D motions into a direct manipulation interface. Kinetic units integrate 3D-printed helical or spiral springs as self-contained energy sources, convert spring deformations into controllable motion behaviors via mutually engaged kinematic transmission mechanisms (*e.g.,* gears, rack-and-pinions), and provide triggable motion via embeddable compliant lock mechanisms. Kinetic units offer three key benefits: (i) the embedded components are parameterizable—allowing end-users to control the embedded energy and motion characteristics such as movement speed [8, 35] based on their design needs; (ii) kinematic mechanisms are auto-generated—end-users do not require advanced mechanical design knowledge; and (iii) all embedded parts are 3D printed in place—reducing the need for manual assembly [8, 11].

Kinergy is aimed at lowering the barrier for novice and expert 3D modelers to create 3D models with desired motions. To create a kinematic 3D-printable model in *Kinergy*, the user selects a target motion from seven supported motion types, customizes the motion characteristics (*e.g.,* motion direction, energy, movement) through a set of graphical user controls, adds a lock mechanism for motion actuation control, previews the action of the 3D model, and prepares the converted model for 3D printing on desktop 3D printers. For example, Figure 1 shows a screenshot of *Kinergy* and the resulting 3D-printed model of a pull-back car that uses a *continuous rotation* kinetic unit for self-propelled motion. We demonstrate the potential of our approach with a series of kinematic applications created with *Kinergy*, such as a self-opening umbrella (Figure 11), a custom pull-back car (Figure 14), and a self-lifting trash bin (Figure 13).

In summary, this paper contributes: (i) custom kinetic units for the parametric design and control of seven self-propelled motion behaviors with 3D printable springs, locks, and transmission mechanisms; (ii) an interactive design tool called *Kinergy* that lowers the barrier to creating highly custom motion-enabled models for 3D printing; and (iii) a variety of applications that show the potential of *Kinergy* to create motion-enabled devices.

2 RELATED WORK

Our work builds upon prior work in 3D printable motion, including 3D printable kinematic components, fabrication techniques, and design tools.

2.1 Motion Types for 3D Printing

Traditional mechanical assemblies that interconnect kinematic elements includes toothed racks, gears, cams, and levers—all which support specialized movement such as translation, rotation, reciprocation, and oscillation [21, 29]. Amongst these motion types, *translation* describes motion along a fixed path, such as a sliding door; *rotational* motion is circular movement around an axis like a wheel on an axle; *reciprocating* motion describes an object repeatedly moves back and forth like a piston; and *oscillation* combines rotational and reciprocating motion like a pendulum.

In the 3D printing literature, past work has explored creating kinematic objects with 3D printable motion behaviors including translation [8, 35], rotation [38, 45], reciprocation [15, 37], and oscillation [35, 47]. However, these approaches support only one or a few motion types. In contrast, our work explores three non-periodic 3D printable motions by producing a one-time, ephemeral movement following a line (instant and continuous translation) and an arc path (instant rotation). Additionally, we support four repeated motion types by making the 3D-printed object rotate around an axis (continuous rotation), move back and forth following a linear trajectory (reciprocation), and move along an arc repeatedly (intermittent rotation and oscillation).

2.2 Fabrication Techniques for Making 3D Printable Kinematic Objects

To create self-activated kinematic 3D-printed objects, an energy source and a method to translate that energy to motion are both needed. Towards the former, researchers have explored interfacing 3D-printed parts with external electric motors [12, 16, 27, 41] and creating composites that reacts to environmental stimuli like heat [13, 31, 40] and pressure [22, 32]. While these methods require additional programing and specialized environmental control to trigger 3D objects to move, our work presents a coherent tool-supported workflow that solely use 3D-printed parts as the key to powering the motion of 3D-printed objects. Similarly, while prior work has explored embedding 3D printable kinematic components such as joints [2, 5, 38, 44], hinges [9, 25], linkages [3, 24], telescoping structures [43], and structured metamaterials [10, 26, 34], we use 3D printable springs as self-contained energy sources to activate motion output because springs can store and transmit energy [8, 11, 20, 46]. For example, a solid cone is converted into a

[1]https://www.thingiverse.com/thing:3928677
[2]https://www.thingiverse.com/thing:38159
[3]The open-sourced GitHub repository: https://github.com/makeabilitylab/Kinergy

springy rocket launcher that compresses and extends to fly the 3D-printed rocket [8]. In our work, we explored two primary spring types—helical and spiral springs—as the energy source to trigger movements.

For translating power to specific motion output, researchers have also combined 3D printable kinematic elements with transmission mechanisms such as gears [18, 36, 45, 47] and compliant structures [24, 42]. For example, Zhu *et al.* created animated mechanical toys with computationally generated mechanical parts that perform linear and circular motion using gears, cams, crank-sliders in a box beneath the toy characters [47]. Similarly, our work leverages various transmission mechanisms, including geartrains, rack-and-pinions, Scotch yokes, Geneva drives, and crank and slotted levers, to achieve both non-periodic like continuous translating and periodic such as reciprocating motions. Most relevant to our work, Song *et al.* combined a set of 3D-printed elemental mechanisms such as cams and followers and off-the-shelf steel spring motors in wind-up toys of arbitrary shapes for expressive part motions such as translation and oscillation [35]. In contrast, our work embeds all springs and transmission mechanisms in the 3D models and all kinematic parts are 3D printed in place, reducing post-print manual assembly.

2.3 Design Tools for Creating Kinematic 3D Models

Custom interactive computer-aided design (CAD) tools are designed to support end-users to create and render movable 3D parts that enable kinematic behaviors and meet user-defined requirements [6, 10, 16, 28]. The challenge in these tools is that generating the enabling 3D structures requires substantial domain knowledge from mechanical engineering [24] and robotics [23]. To lower the barrier to designing such enabling structures, our work employs a similar approach to [35, 45, 47] by embedding predefined 3D printable kinematic mechanisms in custom shapes for desired motion behaviors. For example, Zhang *et al.* developed a tool for novices to retarget and place predefined mechanical templates in arbitrary 3D shapes and adjust each element for desired motions [45]. Similarly, Song *et al.* also predefines elemental structures for a set of target motion that can be parameterizable and integrated into wind-up toys in arbitrary forms to produce expressive motion patterns [35]. While other approaches are possible, such as remixing mechanical elements from existing 3D creations [30] and converting 3D bodies into movable parts with structured cell units [10], our approach allows users to parameterize the embedded kinematic designs in place for characterizing motion behaviors. In addition, we expand prior work by allowing the customization of motion power via spring parameters and the control of motion power with a custom compliant lock mechanism. Like those tools that provide a preview of the generated behaviors [14, 15, 17, 23, 33], we use a similar graph-based approach to [25] and simulate how the driving force produced by the motion power transforms into output motion through a series of kinematic elements. This approach has been used for parsing part movement and the interactions between parts in a conjunction [35, 47]. In our work, we apply this approach to the spring motor and transmission mechanism for an animated motion sequence from the initial configuration to the result state.

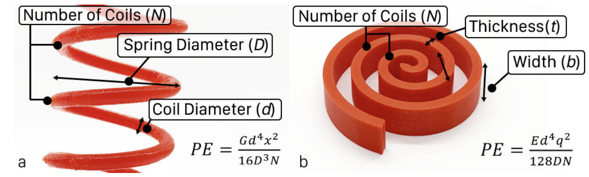


Figure 2: Spring energy source and related parameters: (a) helical spring and (b) spiral spring. Material-related parameters are used for calculating the potential energy in springs: Shear modulus G for helical springs and Young's modulus E for spiral springs.

3 KINERGY

To enable the design and control of motion behaviors in 3D models with self-contained energy sources, we create *Kinergy*—an interactive *design tool* that allows end-users to create self-propelled motion behaviors. *Kinergy* builds upon the notion of *kinetic units*, a set of custom mechanical structures that harness the energy stored in 3D printable springs, contain custom locks/triggers, and translate energy into motion behaviors. Below, we first describe the seven kinetic unit designs that enable controllable motion. We then introduce how these kinetic units can be integrated into *Kinergy*, the design tool's user interface.

3.1 Kinetic Units

Kinetic units support seven controllable output motions that are 3D printable: instant translation, instant rotation, continuous translation, continuous rotation, reciprocation, intermittent oscillation, and intermittent rotation. The instant and continuous kinetic units are categorized by how quickly the output motion lasts rather than how fast the energy is released from the spring. For the instant kinetic units, the motion is executed immediately. For the continuous kinetic units, energy is dissipated over time, leading to a lasting output motion. Each kinetic unit consists of an embedded energy source—either a helical or spiral spring, a compliant lock mechanism, and a transmission mechanism. Table 1 shows primary kinetic unit compositions in rendered 3D models and cutaways for target output motion types.

3.1.1 Energy Sources. To enable self-propelled and controllable motion output, we use 3D printable and embeddable mechanical springs. As described by He *et al.* [8], springs are attractive yet often overlooked energy sources for 3D printing. Inborn spring parameters such as coil diameter and number of coils can be used to control the energy stored in deformed springs (Figure 2). The springs themselves can be 3D printed within an object, eliminating the need for other external actuators. As in prior work [8, 35, 45], we use embedded helical and spiral springs, which can be customized based on spring parameters (Figure 2). To store the potential energy in a spring, the user needs to either manually press on a helical spring or wind up a spiral spring. We demonstrate the capability of supporting these two energy-charging methods in two example applications: a pull-back car is activated by a winding spiral spring (Figure 14) and a handheld flashlight lights up by hand pressing (Figure 15).

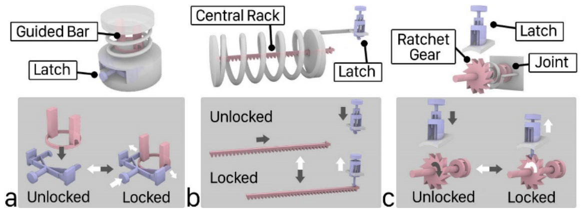


Figure 3: Compliant lock mechanisms used in kinetic units: (a) guided bars and the compliant latch for the locking control of the helical spring in instant kinetic units, (b) the central rack and the compliant latch for the locking control of the helical spring in non-instant kinetic units, and (c) ratchet gear and compliant latch design for the locking control of the spiral spring.

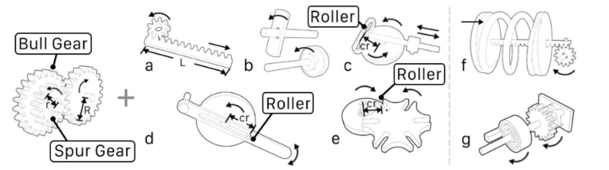


Figure 4: For non-instant motion types, kinetic units combine geartrains (pairs of bull and spur gears) and kinematic transmission mechanisms: (a) rack-and-pinions, (b) axels or revolute joints, (c) Scotch yokes, (d) crank-and-slotted-levers, and (e) Geneva drives. Helical springs engage with gears through (f) rack-and-pinions and (g) spiral springs are co-axial with gears for driving the geartrain.

3.1.2 Lock Mechanisms. To store the potential energy in a spring, kinetic units include unique in-place compliant structures as locks (Figure 3). The locks differ depending on the spring type. These locks were designed iteratively through brainstorming and experimentation to arrive at robust, consistent solutions that have relatively small footprints.

For a *helical spring* energy source, there are two different designs in kinetic units. For the kinetic unit that supports instant motion, two protruded guiding bars with notches are attached to the moving end of the spring, which snaps to two latch hooks situated in the stationary part when the spring is compressed (Figure 3a). The latch hooks are controlled via a compliant two-bar mechanism—pressing the button makes the two hooks move apart, releasing the two bars and unleashing the stored spring energy. For the kinetic units that enables non-instant motion, the helical spring has a central rack that acts as a moving hook and contacts and locks with a sliding latch when the spring is at its maximum compression (Figure 3b).

For a *spiral spring*, the spring can be locked at a rotation angle with a ratchet mechanism, which consists of a gear with asymmetrical teeth, in parallel with the spring. A sliding compliant latch—working as the "pawl"—is mounted on the stationary part (Figure 3c). The latch uses an identical compliant two-bar design—two-sided hooks retract and move under button pressing and pulling and stop the sliding lock at positions when engaging grooves in the guided wall. Spring rotation is prevented by engaging one gear tooth with the inserted latch. By pulling the latch and freeing the ratchet gear, the spiral spring turns and releases the stored potential energy.

3.1.3 Transmission Mechanisms. To transform the driving force created from a spring into an output motion, we combine the spring energy source with specialized transmission mechanisms made of kinematic pairs. These pairs are joints between two contacting rigid mechanical components under relative motion [1], including geartrains, rack-and-pinions, Scotch yoke, crank and slotted levers, and Geneva drives (Figure 4a-e). The specific mechanism is based on the desired motion enabled and spring type used by the kinetic unit. For those non-instant motion types, a geartrain that is constructed with a series of two paralleled gears (a bigger bull gear and a smaller spur gear) is the crucial component to transmit the motion from the spring energy source to the end-effector. While the speed of energy release is hard to control [7], the output motion speed can be controlled through geartrain parameters such as gear ratios (*e.g.,* to control how fast a pull-back car travels).

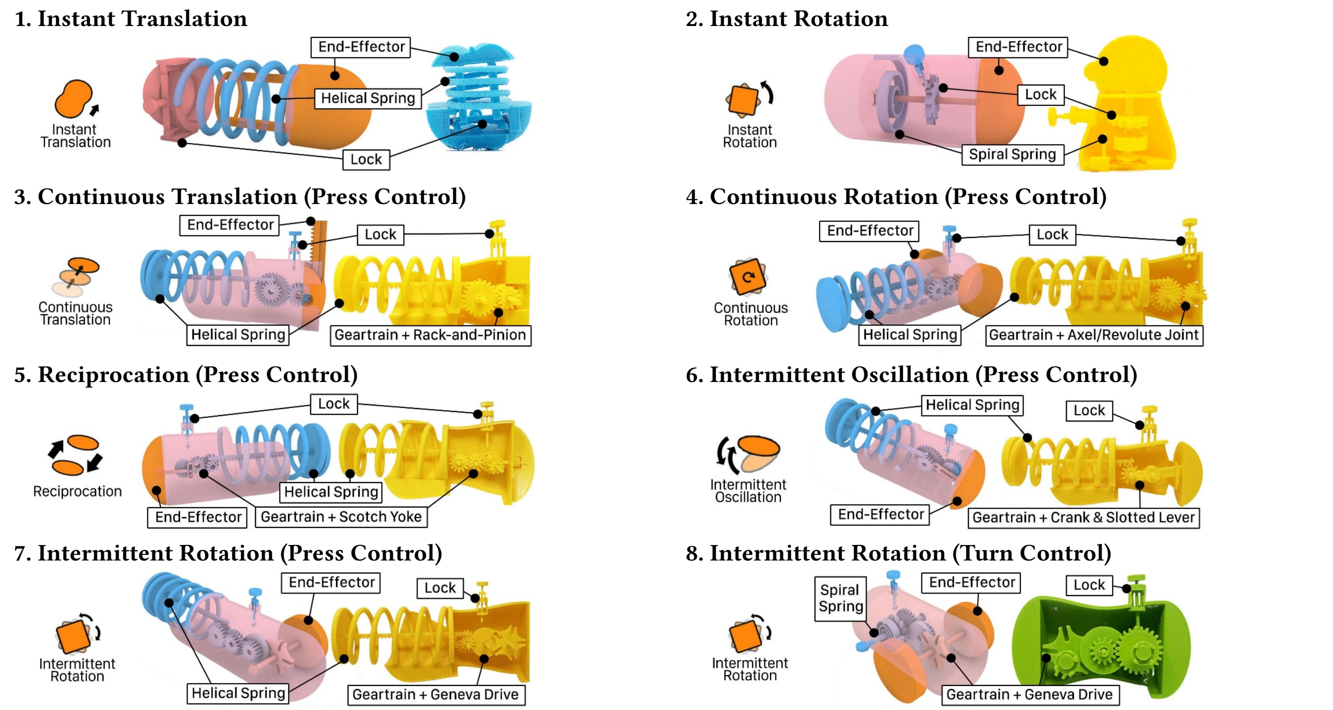
To engage the spring with the geartrain, we use two approaches. For a helical spring, we attach a central rack to the helical spring and engage the rack with the first gear in the geartrain, which commits a rack-and-pinion mechanism (Figure 4f and Table 13-7) For a spiral spring, the first gear in the geartrain and the spiral spring are coaxial, and the gear rotates when the spring is turned (Figure 4g and Table 18).

By combining the geartrain with different kinematic mechanisms, the end-effector can achieve desired output motions (Figure 4a-e and Table 1): a rack that connects to the end-effector engages with the last gear in the geartrain to commit a *rack-and-pinion* for the output translation; a *revolute joint* or an *axel* rotates together with the last gear in the geartrain to drive the connected end-effector to rotate; the end-effector equipped with a *Scotch yoke* moves back and forth on a linear path; a *crank and slotted lever* design can transfer the rotary movement of the last gear in the geartrain into repeatedly oscillating movement, where the end-effector moves along an arc path; and finally, inserting a *Geneva drive* between the last gear in the geartrain and the end-effector leads to intermittent rotary movements of the end-effector.

3.1.4 Kinetic Unit Types. The energy sources, locks, and transmission mechanisms described above are combined to create seven kinetic units (Table 1). The non-instant kinetic units can be controlled by pressing a helical spring or by turning a spiral spring. Each kinetic unit allows customization by parameterizing both the spring energy source and the transmission mechanism. Below, we enumerate the kinetic unit compositions and how they support distinct motion types.

Instant Translation. To enable an object to extend its body, the kinetic unit auto-replaces a portion of a selected 3D shape with a helical spring. The spring connects to an end-effector and a stationary segment of the body (Table 1-1). The converted spring is controlled by a compliant lock shown in Figure 3a. For example, a self-popping Halloween pumpkin décor has one instant translation kinetic unit embedded and the sectional view is shown in Table 1-1. Because the body directly executes the motion with a spring, no transmission mechanism is needed in this kinetic unit.

Table 1: Motion types and Kinergy kinetic unit examples (3D models and cutaways).



Instant Rotation. To enable an object to rotate, the kinetic unit auto-embeds a spiral spring in a selected object body, which then separates into two parts: an end-effector and a stationary segment (Table 1-2). The spring center connects to the end-effector via a central shaft and the spring coil end connects to the stationary segment of the body via a solid rod. The central shaft extends toward the stationary part and connects via a *revolute joint* consisting of a bearing socket and a circular disc, which avoids extraneous shaft movement caused by turning. The converted spiral spring is controlled by a compliant lock shown in Figure 3c. Similar to *instant translation,* no transmission mechanism is needed.

Continuous Translation. To prolong the output translation motion, the kinetic unituses a geartrain that joins with the spring energy source for motion transmission. A rack that connects to the end-effector mated with the last gear in the geartrain is used to perform as a *rack-and-pinion* for the output translation (Table 1-3). This kinetic unit works with both helical and spiral spring energy sources. For helical spring control, the compliant lock uses the central rack as the guided bar rather than the original two-bar design in Figure 3b. For the spiral spring control, the first gear shaft extends and drills through the 3D body to provide a handler for spiral winding. The other end of the shaft is connected to the body via a revolute joint so that the shaft rotates in place. The placements of spring energy source and lock mechanism are reused in other non-instant kinetic units.

Continuous Rotation. To support a continuous rotation output, the last gear in the geartrain and the end-effector are coaxial on a shaft, which drives the end-effector to rotate. If multiple end-effectors residing on the opposite sides of the 3D body, the last gear shaft extends and drills through the body as an *axel* to connect end-effectors; otherwise, the shaft only extends on one direction and the other end connects to the body via a *revolute joint* (Table 1-4).

Reciprocation. To create a reciprocating motion, a Scotch yoke (*a.k.a.,* slotted link mechanism) is used to connect the last gear in the geartrain and the end-effector (Table 1-5). The Scotch yoke is composed of a circular disk, a roller, a yoke, and a connecting rod. The circular disk resides coaxially with the last gear and the connecting rod connects to the end-effector. The linear motion guides are fixed to the 3D body. When the circular disk rotates along with the gear, the roller slides inside the yoke, making the connecting rod and end-effector move repeatedly.

Intermittent Oscillation. To enable an intermittent oscillation, a crank and slotted lever (one type of quick-return mechanism) is driven by the last gear in the geartrain to actuate the end-effector to oscillate along an arc path (Table 1-6). The crank and slotted lever consists of a pivot that is fixed inside the 3D body, a bull gear that resides coaxially with the last gear in the geartrain, a crank pin, and a slotted bar that connects to the end-effector. When the bull gear turns, the crank pin slides back and forth in the slotted bar and swings the end-effector around the pivot for intermittent oscillating motion.

Intermittent Rotation. To create an intermittent rotation, a Geneva drive, which translates a continuous rotational motion into an intermittent rotational motion, resides coaxially with the last

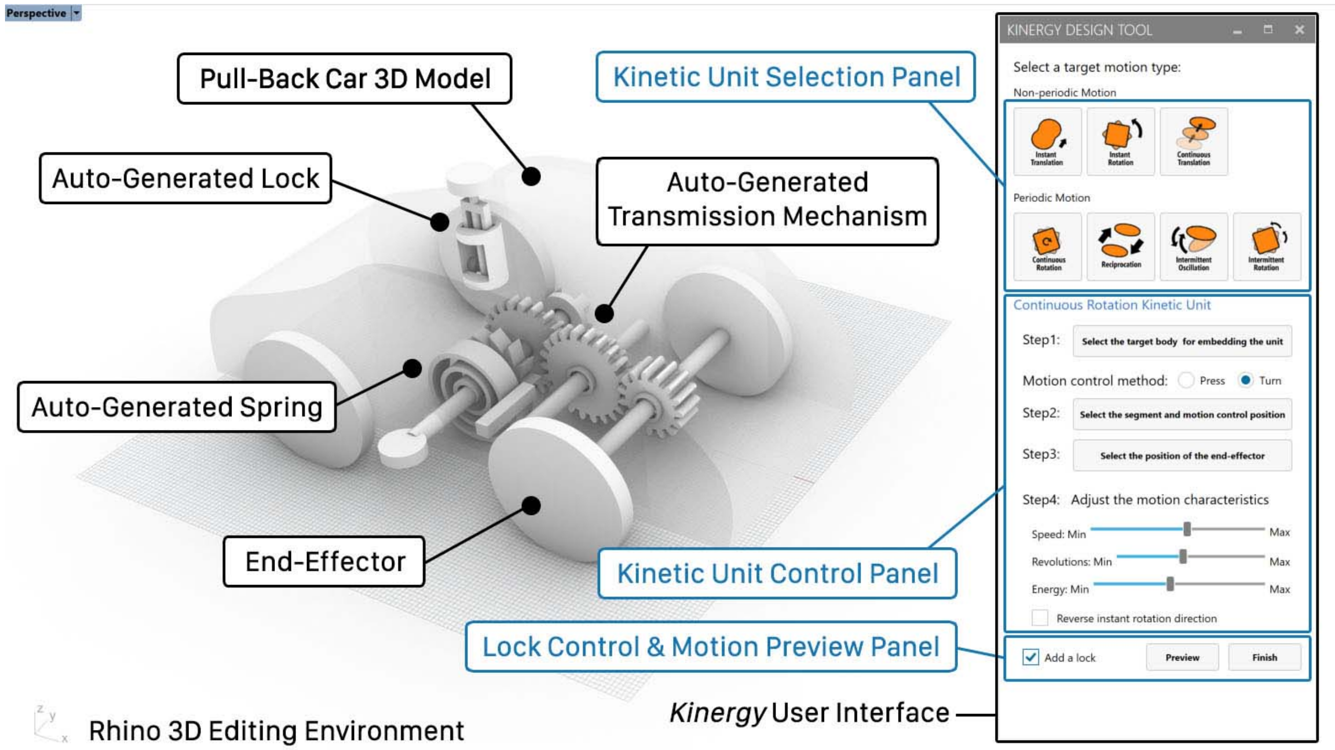


Figure 5: The user interface of *Kinergy* design tool.

gear in the gear train (Table 1-7 and Table 1-8). The Geneva drive consists of two parts: a driving wheel and a driven wheel [39]. When the driving wheel rotates, the protruded roller on the driving wheel goes in and out the slot on the driven wheel repeatedly, resulting in an intermittent rotational motion of the driven wheel. The end-effector also rotates intermittently since it shares the same shaft with the driven wheel.

3.2 Kinergy Design Tool

Kinetic units provide the foundation to lower design barriers for complex mechanical motions. However, to integrate these motions into 3D models, kinetic units need to be customizable with a front-end interactive design tool, *Kinergy*. Kinergy is an open-source plugin for Rhino 6 (Figure 5), with the front-end user interface built with *Grasshopper*[4] and *Human UI*[5], and the backend written in C# using the *RhinoCommon API*[6].

Kinergy consists of three parts (Figure 5): a *Kinetic Unit Selection* panel, *Kinetic Unit Control* panel, and *Lock Control and Motion Preview* panel. The *Kinetic Unit Selection* panel (Figure 5) provides seven buttons, each of which indicates the supported motion type and kinetic unit. The *Kinetic Control* panel (Figure 5) displays a series of steps to complete the embedding of the selected kinetic unit as well as to parameterize motion behaviors, such as energy strength and motion displacement adjustment. Finally, the *Lock Control and Motion Preview* (Figure 5) panel allow the user to add a lock to the spring for motion control and preview the generated motion behavior via a simulated animation in a separate window.

Currently, *Kinergy* captures four operation failures: a kinetic unit is not fitting in the model body, unexpected objects are selected, a wrong order of operations occurs, and a new kinetic unit is unpurposedly selected. For the first three error types, the tool shows an error message. The user can then go back to edit the 3D model. For the last, the tool clears all the results and renders the latest result in red (*i.e.,* unbaked results produced by Grasshopper). Below, we describe the workflow of the design tool through a pull-back car, which uses a continuous rotation kinetic unit. We also describe three key parts of the tool: the auto-generation of kinetic units, the parameterization of kinetic unit mechanisms, and motion previews.

3.2.1 Design Walkthrough: Creating a 3D Printable Pull-Back Car. In this section, we provide an example design walkthrough of using Kinergy to create a 3D-printable pull-back car. For this, the user needs to embed a continuous rotation kinetic unit into a 3D car model. See our video, which complements the following description.

(1) In the Rhino 3D editing environment, the user first creates a 3D pull-back car model that includes a car body and four wheels (Figure 6a). The four wheels are positioned in parallel with the car body. The user aligns the facing direction of the car body with the X-axis in the 3D environment.

[4]Grasshopper: https://www.rhino3d.com/6/new/grasshopper
[5]Human UI: https://www.food4rhino.com/app/human-ui
[6]RhinoCommon API. https://developer.rhino3d.com/api/

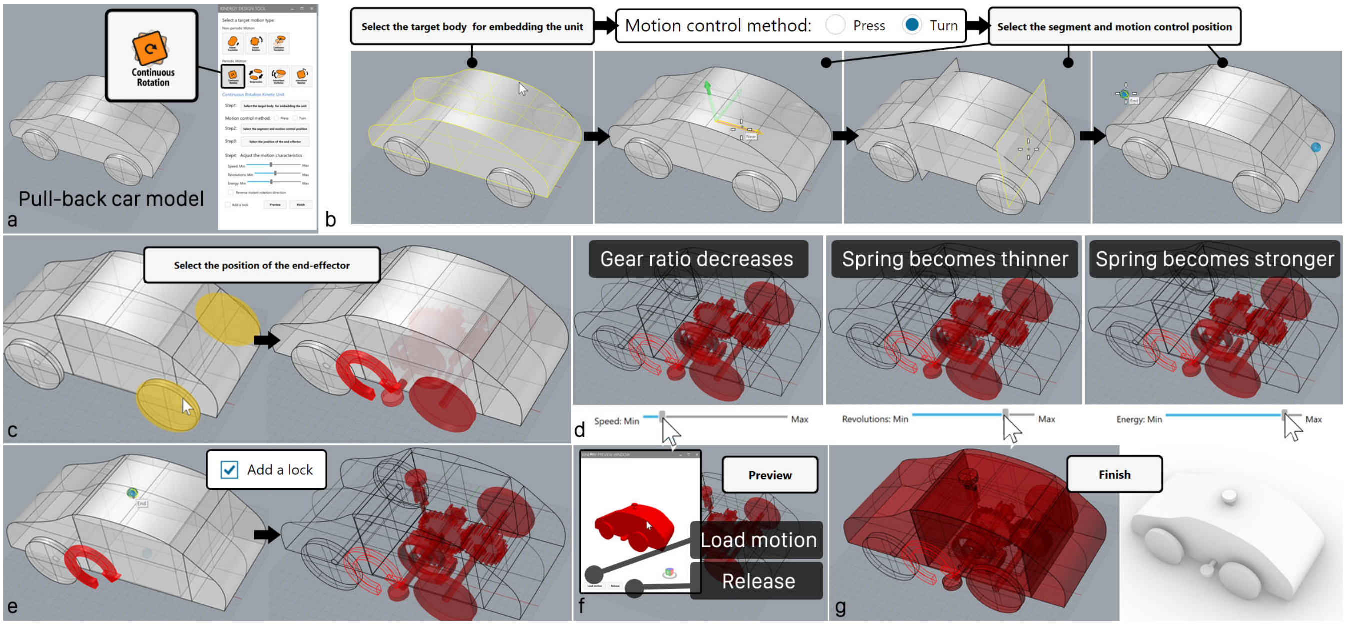


Figure 6: The workflow of making a pull-back car with *Kinergy*.

(2) To make the car move with a continuous rotation kinetic unit, the user clicks on the *Continuous Rotation Kinetic Unit* button form the *Kinetic Unit Selection* panel. The *Kinetic Unit Control* panel then displays the instructional user controls (see the buttons and sliders in Figure 6a).

(3) Then, the user selects the target car body in the Rhino 3D editing environment and confirms the selection by clicking on the *Select the target body for embedding the unit* button (Figure 6b).

(4) The user selects the motion control method—turn (supported by a spiral spring energy source).

(5) After the control method is confirmed, the user clicks on the *Select the segment and motion control position* button. Three colored axes appear for the user to select the car's body orientation, which is the X-axis (Figure 6b). Upon the selection of the body orientation, two adjustable paralleled planes in perpendicular to the body orientation show up. The user can select the target body region for embedding the kinetic unit by dragging these two planes separately and the planes move along the X-axis (Figure 6b). Once the portion is confirmed, the 3D body is split into three parts: two end parts and the middle body that contains the embedded kinetic unit. The user needs to select one end part to indicate which side the spring of the kinetic unit will be placed on (Figure 6b).

(6) Then, the user selects the two wheels next to the other end part (not the spring side) as the end-effectors after clicking on the *Select the position of the end-effector* button (Figure 6c). Upon the selection of the end-effector, the continuous rotation kinetic unit including the spring energy source, the geartrain, and the revolute joint is generated automatically inside the 3D body.

(7) After the kinetic unit is generated, the user can adjust the needed energy, speed, and rotating revolutions by dragging the sliders on the user interface (Figure 6d). The spring design changes based on the energy adjustments and the geartrain updates with user changes on the speed and revolution sliders in real time in the 3D editing environment.

(8) Optionally, the user can add a spring lock by checking the *Add a lock* checkbox on the interface (Figure 6e). Once the checkbox is checked, a ratchet gear and the compliant lock mechanism are auto-generated in place.

(9) Finally, the user examines how the pull-back model moves and the wheels rotate through an animation of the motion in a separately popped window after clicking on the *Preview* button (Figure 6f). In this window, the user views the converted pull-back car model in a set of interactive ways: rotating, panning, and zooming. To trigger the motion preview, the user first clicks on the *Load Motion* button to charge the energy in the spring. Then, the user releases the energy by clicking on the *Release* button and the simulated animation begins. All the generated 3D parts can be exported for 3D printing by hitting *Finish* button (Figure 6g).

3.3 Implementation

3.3.1 Generating Kinetic Units. Section 3.1 detailed all the kinetic unit designs. Here, we document the implementation of these units as part of the *Kinergy* design tool. Generating kinetic units requires three processing steps: determining the position and orientation of the embedded kinetic unit, generating the kinetic unit components (spring energy source, transmission mechanism, and lock), and interfacing the kinetic unit with the 3D body. Below, we describe these steps by starting with a set of terms that we use in our implementation: Dir_p denotes the pose direction of the embedded kinetic unit, $Axis_{trans}$ denotes the axis that a part translates along, $Axis_{rot}$ denotes the axis that a part rotates around, $Part_{st}$ denotes the stationary part that connects to the kinetic unit, and $Part_{ee}$ denotes the end-effector that connects to the kinetic unit (Figure 7).

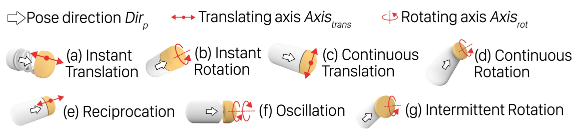


Figure 7: The unit orientation, translation axis, and rotation axis in each kinetic unit. The end-effectors are highlighted in light orange and the stationary parts are marked in grey.

The tool first decides the position and orientation of the embedded kinetic unit based on the user selection of the $Part_{ee}$ and user input directions. All the kinetic units are embedded in a user selected segment of the 3D body (see *Design Walkthrough* step 4). For kinetic units that have no geartrains, the selected body portion is converted with an embedded spring and the Dir_p of the kinetic unit is aligned with either the helical spring's translation axis or the spiral spring's rotation axis (Figure 7a-b). Once the Dir_p is determined, the kinetic unit is generated in place regardless of which direction the unit is oriented in. For kinetic units that use geartrains and transmission mechanisms, Dir_p is aligned with the propagating direction of the gears in the geartrain (Figure 7 c-g). For those non-instant kinetic units, the unit orientation also depends on the rotary angle along the Dir_p, which is input by the user in the tool.

After the position and orientation of the embedded kinetic unit are determined, *Kinergy* computes and generates the spring, the transmission mechanism, and the lock. For instant kinetic units, the spring resides close to the center of the selected segment. Then, by computing the user input energy and motion attributes (*e.g.,* translation displacement, rotation angle), the tool uses the *RhinoCommon* `spiral` function to create a spiral curve and the `sweep` function to create a solid spring. Upon selecting the lock positions by the user, *Kinergy* auto-generates the lock structures in place.

For those non-instant kinetic units, *Kinergy* also generates transmission mechanisms in addition to springs and locks. To generate the geartrain, the tool first determines the number of gear sets and gear positions based on the body segment volume and Dir_p. All the bull and spur gears in the geartrains share the same gear module, which is the unit of size that indicates how big and small a gear is, in our implementation. Then, *Kinergy* generates all the gear models using gear parameters such as gear diameter and the number of teeth, which are determined by the input motion speed, and the approach of calculating an involute spur gear[7]. All the gears automatically self-rotate to engage with each other. Once the gears are generated, *Kinergy* generats the springs based on user selected input method and engages the spring with the first gear of the geartrain. Finally, to engage the geartrain with the end-effector, *Kinergy* generates the parametric 3D models of specialized mechanisms (*e.g.,* Scotch Yoke). The $Part_{ee}$ connected to a specialized mechanism moves strictly following either a translation axis or a rotary axis (Figure 7c-g).

Finally, *Kinergy* also generates extra structures to secure the kinetic unit in the 3D model. For example, the gear shafts in the geartrains are auto-extended to connect both ends to the solid $Part_{st}$. A solid pole is auto-generated to fixate the outer end of the spiral spring to the $Part_{st}$ in the kinetic units. To prevent the movable gears from sliding on the shafts, a pair of spacers are auto-generated on both sides of the gear. For non-instant kinetic units, the tool creates handler button to operate the springs (Figure 5).

[7]Calculation of involute gears: https://www.tec-science.com/mechanical-power-transmission/involute-gear/calculation-of-involute-gears/

3.3.2 Parameterizing Embedded Energy and Motion Properties. *Kinergy* allows the user to control stored energy in 3D models by adjusting the spring parameters and to characterize desired motions by changing the parameters of the geartrains and the specialized kinematic elements. Table 2 shows all the parameters that are used for energy control and motion characterization in the design tool.

Based on spring theory (p. 156 and p. 537 in [4]), the potential energy stored in the deformed springs are impacted by spring parameters and the amount of spring deformation. For the helical spring, the potential energy that the spring can achieve is proportional to the fourth power of coil thickness d and the square of the compression/extension displacement X, while inversely proportional to the third power of spring diameter D and the number of coils N (Eq. 1). For the spiral spring, the potential energy that the spiral spring can achieve is proportional to the spring width b, the third power of the thickness t, and the square of rotary angle θ, while inversely proportional to the number of coils N (Eq. 2). These parameters inform the energy customization for 3D models created with *Kinergy*.

Besides energy control, *Kinergy* also allows the user to customize motion characteristics (Figure 5). For instant translation kinetic unit, spring compression displacement can be directly specified through a slider. Similarly, the rotating revolutions of the spiral spring is proportional to the rotary angle θ (Eq. 3) in an instant rotation kinetic unit. For non-instant motions, the design tool enables the user to control the motion speed, which relates to gear ratio of the geartrain. As the speed slider in the user interface moves, the tool recalculates and updates the gears in the geartrain, including the bull gear radius R, spur gear radius r, and the number of bull and spur gear sets N in the geartrain (Eq. 4). In addition to speed control, the design tool also supports specialized motion characterizations for those non-instant kinetic units via sliders. For the continuous translation and rotation, the translating distance and rotation revolutions are proportional to the spring deformation amount and the gear ratio of the geartrain (Eq. 5-6). For reciprocation and oscillation, the reciprocating distance and oscillating amplitude are also proportional to the crank length in their Scotch yoke and crank-and-slotted-lever designs (Eq. 8-9). Finally, for intermittent rotation, the rotary angle at each rotation step is inversely proportional to the number of opening slots in the driven wheel of the Geneva drive (Eq. 10).

3.3.3 Previewing Generated Motion. *Kinergy* provides the motion preview by encoding all the part types and interactions between parts in a graph, which is similar to [25, 47]. In the graph, each node stores the part type (*e.g.,* spring, gear, rack-and-pinion, non-kinematic connectors), part parameters (*e.g.,* spring wire diameter, number of coils), and its motion attributes (*e.g.,* rotation axis). Each edge encodes one of the three interaction types between two mechanical parts: fixation (*e.g.,* a gear shaft is fixed to the solid 3D body), engagement (*e.g.,* two gears mate with each other), and locking (*e.g.,* a compliant latch stays in the lock groove). The kinetic unit created with our tool is represented using a graph, for example,

Table 2: Parameters for energy control and motion characterization in the kinetic units.

Kinetic Unit Type	Adjustable Parameters	Controllable Output	Relationship
Instant translation	Spring diameter D, coil thickness d, the number of coils N, compression/extension displacement X	Helical spring energy P_e, spring displacement X	$P_e \propto \frac{d^4 x^2}{D^3 N}$ (1)
Instant rotation	Spring width b, thickness t, rotary angle θ, the number of coils N	Spiral spring energy P_e, rotating revolutions Rev	$P_e \propto \frac{bt^3\theta^2}{N}$ (2) $Rev \propto \theta$ (3)
Continuous translation	Gear ratio of the geartrain G_{ratio} (bull gear radius R, spur gear radius r, number of bull and spur gear sets N), spring parameters (see above)	Spring energy P_e, motion speed S, translating distance Dis	$S \propto G_{ratio} = (\frac{R}{r})^N$ (4) $Dis \propto \{ \begin{matrix} X \times G_{ratio} \\ \theta \times G_{ratio} \end{matrix}$ (5)
Continuous rotation	Gear ratio of the geartrain G_{ratio}, spring parameters	Spring energy P_e, motion speed S, rotating revolution R_r	$R_r \propto \{ \begin{matrix} X \times G_{ratio} \\ \theta \times G_{ratio} \end{matrix}$ (6)
Reciprocation	Gear ratio of the geartrain G_{ratio}, spring parameters, Scotch yoke crank length r	Spring energy P_e, motion speed S, stroke Str, reciprocating distance Dis_{rec}	$Str \propto \{ \begin{matrix} X \times G_{ratio} \\ \theta \times G_{ratio} \end{matrix}$ (7) $Dis_{rec} \propto r$ (8)
Intermittent Oscillation	Gear ratio of the geartrain G_{ratio}, spring parameters, quick-return crank length r	Spring energy P_e, motion speed S, stroke Str, oscillating amplitude Amp	$Amp \propto r$ (9)
Intermittent Rotation	Gear ratio of the geartrain G_{ratio}, spring parameters, number of opening slots on the driven wheel n	Spring energy P_e, motion speed S, stroke Str, interval angle θ_{int}	$\theta_{int} \propto \frac{1}{n}$ (10)

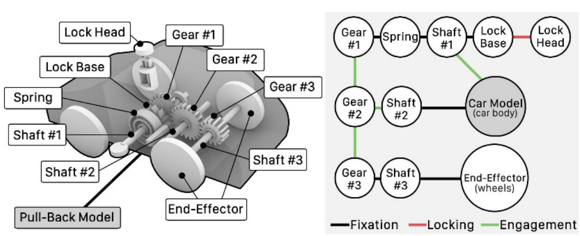


Figure 8: The pull-back car 3D model is represented by a graph for motion preview.

Figure 8 shows a graph representation of a pull-back car with a continuous rotation kinetic unit embedded.

With the graph-based representations, *Kinergy* creates an animation preview that shows the idealized predicted motion behavior enabled by the embedded kinetic unit, rather than a realistic physics-based simulation. This preview is created under three assumptions: (i) the driving force solely originates from the spring deformations; (ii) external forces such as frictions and object weight are negligible in the simulated animation; and (iii) the springs deform in a realistic and strict manner, *e.g.,* helical springs only compress/extend along a linear path and spiral springs only rotate around their central axes. The created animation is rendered at 20 FPS frame rate and displayed in a separate window. In each frame, the tool begins with the spring node by calculating the spring movement using *Hooke's Law*—capturing the amount of spring deformation, which derives motion acceleration and speed. Then, spring deformation changes are propagated to all the graph nodes through edges using depth-first searching (DFS). During the search, the part position and speed stored in each node are updated based on its part type and interaction with a neighbor node, as well as position changes from the neighbor. The transmissions terminate when either the spring returns to its equilibrium or a new locking is applied, *i.e.,* the user clicks on the *Load Motion* button and the spring is locked again.

4 FABRICATION

Kinergy aims to add self-propelled motion behaviors that can be 3D printed without post-print manual assembly. To achieve this goal, we provided calibrated tolerances for the printability of kinetic units as well as guidance for slicing. The calibration data is based on a dual-extruder *Ultimaker* 3D printer, which uses polylactic (PLA) as the printing material and polyvinyl alcohol (PVA) as the support material. First, we identified three types of mechanical gaps in a kinetic unit design that may be problematic for consumer-grade 3D printing (Figure 9a-c): the gap between two engaged gears (Type 1), between a gear and its nearby spacer on the shaft (Type 2), and between a gear and its residing shaft (Type 3). To ensure that robust and functional gears and spacers are printed on the shafts in one shot, we took a trial-and-error approach and found that the printing orientation, as shown in Figure 9d, is superior to the other directions due to anisotropic 3D printing. With a printer that uses a 0.4mm-sized printing nozzle and 0.15mm printing layer height, we found that a 0.3mm Type 1 gap makes sufficient room for the gear backslash, which is the distance between the involutes of the mating gear teeth, and a 0.25mm Type 2 gap and a 0.35mm Type 3 gap appropriate as minimal distances. Further, the 3D model is sliced with an *Exclusive* slicing tolerance and a minimal negative 0.04mm horizontal expansion in a slicer software (*e.g., Cura*) to

Table 3: Overview of *Kinergy* applications.

Application	Umbrella	Game Controller	Trash Can	Pull-Back Cars	Flashlight	Maneki Neko	Cutter	Zoetrope
Applied Kinetic Unit Type	Instant Translation	Instant Rotation	Continuous Translation	Continuous Rotation *(turn)*	Continuous Rotation*(press)*	Reciprocation	Oscillation	Intermittent Rotation

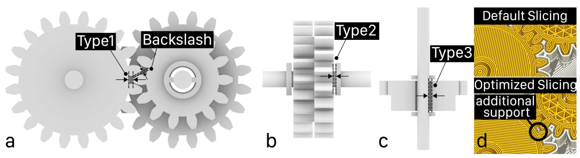


Figure 9: Three types of tolerance identified as problematic for one-shot printing: (a) the gap between two mating gears, (b) the gap between a gear and its nearby spacer, and (c) the gap between a gear and its residing shaft. The 3D model is sliced in (d) an optimized orientation and the slicing settings are curated to create clean part contours for intermating elements.

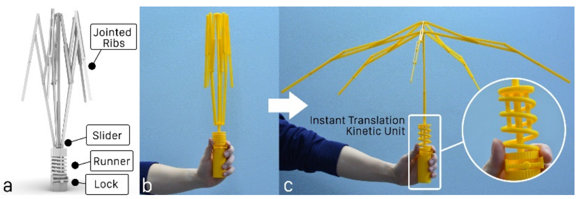


Figure 11: An auto-opening umbrella prototype created with the instant translation kinetic unit: (a) the rendered 3D model of the umbrella, (b) the printed and assembled umbrella in the locked stat, and (c) the opened umbrella after the runner is unlocked.

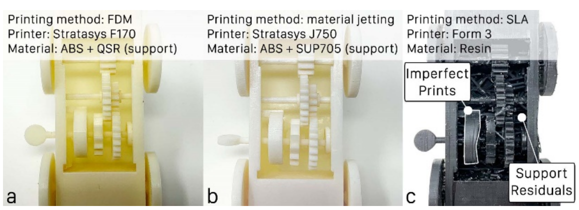


Figure 10: The pull-back cars created with *Kinergy* are printed with various 3D printers and printing technologies: (a) industrial-level FDM 3D printing, (b) PolyJet 3D printing, and (c) SLA 3D printing.

avoid over-fused gaps and to create a clean profile contour for the inter-engaged parts such as gears(Figure 9d).

To examine if our approach is applicable to other 3D printers and printing methods, we also printed pull-back car models created with *Kinergy* on 3D printers that offer different printing capabilities (Figure 10): an industrial FDM 3D printer (Stratasys F170; printing material: ABS; support material: water-soluble QSR), an industrial PolyJet 3D printer (Stratasys J750; printing material: ABS; support material: SUP705), and a desktop stereolithography (SLA)-based 3D printer (Form 3; printing and support material: Resin). As a result, the printed cars are functional except for the one printed with the single material-based approach (Figure 10c), which has support residuals in the car body and with some imperfect printing finishes. We also found that printing tolerance varies across all these methods. For example, the same set of Type1-3 tolerances works perfectly on the PolyJet-based print (Figure 10b), while the gears and joints are wiggly in the car printed with the industrial FDM 3D printer (Figure 10a). Based on these preliminary explorations, we anecdotally conclude that our approach is feasible with various multi-material 3D printing technologies with appropriate printing tolerances.

5 APPLICATIONS

To demonstrate the potential of *Kinergy*, we created eight functional models with *Kinergy* and showcased how each kinetic unit is used to support a specialized motion in these applications (Table 3). All the models are printed in one shot, and the dissolvable PVA support is fully removed before demonstration. Please see the video in our supplementary materials for full demonstrations.

5.1 Auto-Opening Umbrella

A spring-loaded umbrella or parasol includes a folding canopy supported by jointed ribs mounted to a pole, which automatically opens at the press of a button. To create an auto-opening umbrella prototype with *Kinergy*, we can use the *instant translation* kinetic unit. The prototype consists of a spring-loaded runner and a slider that moves along the pole and connects to the jointed ribs (Figure 11a). The runner and the slider with jointed ribs were printed separately, and the runner was mounted on the pole beneath the slider. With *Kinergy,* we created the runner by converting a cylinder into a compressible helical spring with an embedded lock. The spring is locked at its maximum compression and the slider stays on top of the spring when the umbrella is closed (Figure 11b). The user presses the built-in lock button to open the umbrella, and immediately, the spring in the runner extends, pushing the slider forward and unfolding the umbrella ribs (Figure 11c). A canopy cloth could be added but is not included to avoid occlusion.

5.2 Angle Adjustable Game Controller

We created a catapult-like launcher for an Angry Birds game to demonstrate the *instant rotation* kinetic unit and the ability to use

Figure 12: A catapult-like game controller is made to virtually projectile birds in an Angry Birds game: (a) the rendered 3D model of the game controller, (b) the printed game controller with external sensor and circuitry embedded, (c) flying the bird at a small angle, and (d) flying the bird at a bigger angle.

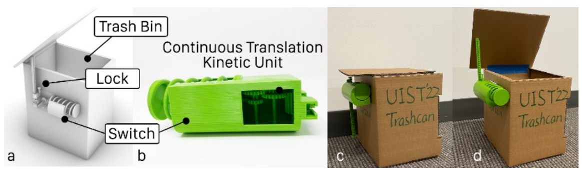


Figure 13: A continuous translation kinetic unit embedded switch is attached to a cardboard trash can: (a) the rendered 3D model of the switch and the trash can, (b) the printed switch, (c) the closed trash can with the switch attached and locked, and (d) opening the can's lid by unlocking the switch.

Kinergy to design custom game controllers quickly. First, we embedded a spiral spring in the launcher body to create a stationary base and a twistable shooting arm (Figure 12a). The arm can be rotated and locked at different positions for desired shooting angle. We mounted an accelerometer on the controller's arm to calculate the shooting angle, which was used to simulate the bird flying trajectory in a game built with *Processing* (Figure 12b-d). To play the game, the player first turns the shooter's arm to a certain angle based on how far the shooter is from the target and then locks the arm at the angle, waiting for a bird to fly into the launching area (Figure 12d). When a bird is in the launching area, the player releases the locked arm and projectiles the virtual bird toward the target.

5.3 Self-Actuated Trash Can

For the last non-periodic motion—continuous translation, we built a self-actuated cardboard trash can by attaching a *continuous translation* kinetic unit-embedded switch on one side (Figure 13). The switch has a helical spring-based button and the rear end of the switch is connected to the lid of the trash can (Figure 13 a-b). To close the can, the user presses the switch button and locks the switch by inserting the built-in latch (Figure 13c). When the user pulls the latch, the compressed switch button extends immediately to drive the link arm to translate upward, lifting the can's lid and opening the trash can (Figure 13d).

5.4 Motion Parameterizable Pull-Back Cars

To demonstrate how to control energy and motion with *Kinergy*, we created four different pull-back cars by varying the embedded energy, motion speed, and travel distance in separate *continuous rotation* kinetic units (Figure 14). The yellow car, as the baseline, embeds a continuous rotation kinetic unit with the lowest energy, lowest speed, and few rotary revolutions (Figure 14a). First, to show

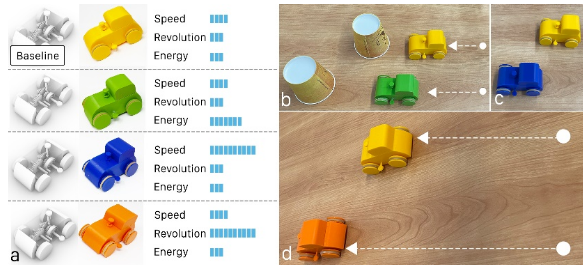


Figure 14: Four 3D-printed pull-back cars with different embedded energy, speed, and traveling distance, which the user interface abstracts details to make it easy for the user to set via sliders: (a) the rendered car models and the printed cars (the numbers of blue blocks in the bars correspond to the slider values for speed, revolution, and energy), (b) the comparison of cars with different energy, (c) the comparison of cars with different motion speed, and (d) the comparison of cars with different traveling distance.

how *Kinergy* allows for energy customization, we measured how far a papercup can be moved after it is hit by two different energy-loaded cars. In this case, we compared the baseline with a green printed car, which has double loaded energy as the baseline. As a result, the green car hit the papercup nearly double farther than the baseline model (Figure 14b). For the speed comparison, we built another car in blue with a higher speed and the blue car traveled faster than the baseline model after both cars were released at the same time (Figure 14c). Finally, we also created an orange car, which was designed to achieve more than three times the number of rotary revolutions as the baseline model, to show how rotary revolutions can be varied with *Kinergy* (Figure 14d). As predicted, the orange car travels a longer distance than the baseline model. These pull-back car examples validate that *Kinergy* allows the user to parameterize embedded energy and motion characteristics for 3D printing.

5.5 Human-Operated Handheld Flashlight

Unlike the pull-back cars that are driven by winding springs, we created a human-operated handheld flashlight to demonstrate pressing as the energy charging method with a *continuous rotation* kinetic unit (Figure 15). In this example, we used a helical spring motor that stores potential energy under compression. An electromotor wired to a LED was mounted and fixed in the flashlight head and the motor's axel was inserted into a socket that was driven by the gears in the kinetic unit (Figure 15b). When the user presses the spring button, the motor rotates its axle, generating current to charge and light up the LED light (Figure 15c). Without a built-in lock mechanism, the user can keep the LED light always on by pressing the button repeatedly and charging the flashlight.

5.6 Battery-Free Maneki Neko Sculpture

To demonstrate the oscillating movement in a 3D printable device, we created a waving arm actuated by an embedded *oscillation* kinetic unit in a Maneki Neko sculpture (Figure 16). The embedded kinetic unit was designed with a maximum oscillation amplitude

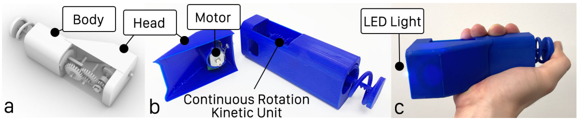


Figure 15: A human-operated handheld flashlight created with *Kinergy* and external electronics: (a) the rendered model of the handheld flashlight, (b) the flashlight head and body, and (c) the functioning flashlight with the embedded kinetic unit.

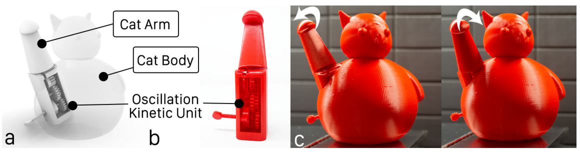


Figure 16: A battery-free Maneki Neko sculpture with an embedded oscillation kinetic unit: (a) the rendered model of the cat sculpture, (b) the printed kinetic unit-embedded arm, and (c) the waving arm in action.

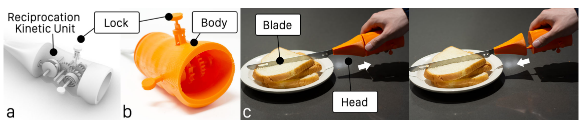


Figure 17: The user holds a 3D-printed cutter that uses an embedded reciprocation kinetic unit to move the blade back and forth repeatedly and cut the bread: (a) the rendered model of the cutter, (b) the printed cutter, and (c) the functioning cutter.

and as many as possible strokes to provide large arm waving movements. After the arm is inserted into the cat body, the user winds the spiral spring using the side handler to load the energy. Without a lock, the arm begins to swing as the user releases the winded spring and stops swinging when the loaded energy dissipates.

5.7 Semi-Automated Cutter

We created an assistive cutter that uses an embedded *reciprocation* kinetic unit to relieve people from the fatigue of the repeated back-and-forth cutting actions (Figure 17). The cutter embeds the kinetic unit into an organic, easy-to-hold cutter body which has a blade attached to the head (Figure 17a-b). The user first loads energy by turing the unit's spiral spring and then releases the lock at the top to execute the cutting job—the blade moves back and forth repetively (Figure 17c-d). Like the other applications, since the sole power source is the embedded spring, the user needs to re-charge the energy in the spring for repetitive cutting tasks.

5.8 3D-Printed Zoetrope

Finally, we embeded an *intermittent rotation* kinetic unit in a box that connects and spins a circular scaffording of six 3D-printed cat models to present a pyshical zoetrope installation—an animiation of a walking cat (Figure 18). To create the physical animation, we first

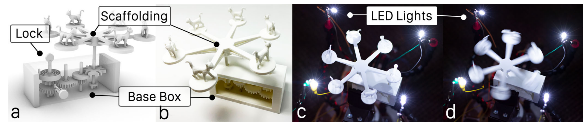


Figure 18: A 3D-printed zoetrope installation that is driven by an intermittent rotation kinetic unit embedded base box and animates a 3D walking cat: (a) the rendered model of the zoetrope, (b) the printed zoetrope, (c) the physical setup with programmable LED flashlights, and (d) the rotating zoetrope in motion.

used a Geneva drive with six opening slots to drive the scaffolding that also has six cat models on the circumference (Figure 18a-b). Then, we mounted the zoetrope model on a helping hand soldering station, where four bright LED lights were attached around the model. These four lights were programmed and controlled by an Arduino board to provide interval flashlight as the zoetrope spined (Figure 18c). Before the spinning, we loaded the energy in the base box by winding the spiral spring and started the LED control program. After the lock was released, the zoetrope model spinned and create the illusion of a walking cat (Figure 18d).

6 DISCUSSION AND FUTURE WORK

6.1 Geometry Complexity

Kinergy enables the customization of 3D printable motions by embedding springs and kinematic mechanical components in 3D-printed objects. However, the size of an object is limited to the minimum spring size and the minimal room for hosting printable kinematic parts in the transmission mechanism. This problem results in two limitations with our current tool: geometries with smaller sizes are not converted, and large gear ratios are hard to achieve (currently, the gear ratio range is 1/5-5). As we have validated the feasibility of *Kinergy* with other multi-material 3D printing technologies such as PolyJet 3D printers, we plan to explore higher resolution printers as alternatives to mitigate both concerns. For example, smaller gears with finer gear teeth are possible to create with metal 3D printing to fit in an object with smaller body volume.

Another limitation of our approach is that the component engagements in the embedded kinetic unit require a compatible object topology. For example, in the continuous rotation kinetic unit with pressing as the energy-charging method, the rack-and-pinion mechanism that couples the helical spring and the geartrain must follow a strict moving path, which leads to an unchangeable spatial configuration. This problem could be further addressed by introducing alternative, yet complex transmission mechanisms. For example, we can use bevel gears, which have conically shaped tooth-bearing faces and rotate around non-paralleled axes, in the geartrain to provide more arrangement options for engaging gears in a compact shape.

6.2 Printibility and Robustness of Mechanical Parts

Kinergy benefits from the in-place printing and the predictable movements of kinematic components (*e.g.,* springs, gears, axles, joints); however, compared to industrial manufacturing methods such as casting and forging, 3D-printed mechanical kinematic parts are limited to the anisotropy and resolution of 3D printing. For example, 3D-printed springs are more robust and less brittle when they are printed perpendicular to the 3D printer's Z-direction [8]. The amount of stored energy may also be influenced by how the spring is manufactured (*e.g.,* 3D-printed *v.s.* metal springs). While 3D-printed springs are limited in size and durability compared to off-the-shelf steel springs, we can still create springs with *Kinergy* and print them with advanced 3D printers, such as Stratasys J8 Series or Desktop Metal, for more robust and flexible energy control. With the current setup, a printable gear has a minimum 1 module and a minimal 0.3mm backslash. From our fabrication exploration, we find that gap tolerances may vary to apply *Kienrgy* to different 3D printers and materials. Additionally, we also find that objects printed with higher infill density (*e.g.,* 60% or above) are less brittle and more reliable. One solution is to use alternative material and printing processes that produce parts with high tensile and endurance strength against loads but less impacted by the anisotropicity of 3D printing. For example, 3D printing metal, a rising 3D printing industry, could be used to create more durable and sophisticated kinematic parts.

The main goal of *Kinergy* is to convert the potential energy stored in the embedded spring into the output motion. However, energy reduces due to the friction caused by the relative motion between mating kinematic parts. To minimize friction, we use kinematic components that have point or curve contacts between moving parts and thus cause less friction [1] in the kinetic units. Additionally, we also diminished friction by adding lubricant such as grease or oil to the engaging points in the kinematic parts.

6.3 Energy-Releasing Triggers

The current lock mechanism in a kinetic unit allows for only one way to release the stored energy in the spring. This prevents the customization of the interaction with the energy source, which may result in onerous human operations to trigger the action. For example, to launch the pull-back car, the user needs to pull the latch, which may interfere with the car movement. Additional sophisticated mechanical structures could offer a design space for the customization of human-operated energy-releasing triggers. For example, a bistable button that is pressable may be more practical in the pull-back car example.

Besides human-operated triggers for releasing energy, custom triggers can also be devised to connect multiple kinetic units and thus execute a series of motions in a controllable sequenc, increasing the expressiveness of the output motion behaivor. In the future, we plan to investigate feasible mechanical designs that allow the user to control when and how the energy can be passed from one kinetic unit to the next unit. One promising solution is to use the tourbillion mechanism as a mechanical timer to bridge two kinetic units. For example, to create an automated door opener, a button with an embedded continuous translation kinetic unit receives the pressing from the user and passes the translation to a tourbillon-based timer, which extends a crank to rotate a concatenating door hinge with an embedded instant rotation kinetic unit after a few seconds.

6.4 Design Tool Improvements

In the future, we plan to improve the design tool in the following aspects. First, to help the user avoid misoperations, we will add more instructional guidance to the tool, such as onboarding tutorials and easy-to-understand representations in the user interface, *e.g.,* using the graphics of weight with labels to indicate the energy strength. Second, as *Kinergy* is applicable to other 3D printing approaches, we plan to incorporate the compatible tolerances and printing-related metadata such as material types and infill density for a diverse range of priners and approaches in the tool. Third, the motion preview can be greatly improved by incorporating the effect of self-weight and friction between kinematic components into a fully interactive simulation environment, which also provides information about the simulated motion and even reports errors. Fourth, while the current tool provides seven kinetic units, we will develop a library of parameterizable mechanical part models for users to combine to create new kinetic unit templates. To use those newly added templates, the user needs to configure the relationship between the model parameters with the output motion characteristics. Finally, we will conduct a formal user evaluation with both novice and expert 3D modelers to identify the usability issues of the design tool for future improvements.

7 CONCLUSION

In this paper, we present *Kinergy*, an interactive design tool that allows the user to create self-propelled motions for 3D printing with a set of parameterizable kinematic designs. To enable the customization of 3D printable non-periodic and periodic motions, we introduced 3D printable kinetic units, which consist of embedded energy sources (either a helical or spiral spring), compliant locks, and transmission mechanisms. The user embeds a kinetic unit into a 3D model to create custom motion-enabled objects. By parameterizing embedded kinetic units, the user controls the energy and motion characteristics. We detailed the user interface of the design tool and the parameterization of energy and motion through custom user controls. Finally, we showcased how *Kinergy* supported the design and fabrication of 3D printable motions via a series of examples and discussed the improvements to our approach.

ACKNOWLEDGMENTS

This work was supported in part by an NSF CAREER Award #1834629. We acknowledge Yuebing Liang and Yawen Zheng for their assistance with the experimentations and prototyping of kinetic units, and Alexander Lefort for his help with 3D printing. We also thank reviewers for their valuable feedback.

REFERENCES

[1] Ashok G. Ambekar. 2007. Mechanism and Machine Theory.
[2] Moritz Bächer, Bernd Bickel, Doug L. James, and Hanspeter Pfister. 2012. Fabricating Articulated Characters from Skinned Meshes. ACM Transactions on Graphics 31, 4: 1–9. https://doi.org/10.1145/2185520.2185543

[3] Moritz Bächer, Stelian Coros, and Bernhard Thomaszewski. 2015. LinkEdit: Interactive Linkage Editing using Symbolic Kinematics. ACM Transactions on Graphics 34, 4: 1–8. https://doi.org/10.1145/2766985

[4] Richard G. Budynas and J. Keith Nisbett. 2008. Shigley's Mechanical Engineering Design, 8th Edition.

[5] Jacques Calì, Dan A. Calian, Cristina Amati, Rebecca Kleinberger, Anthony Steed, Jan Kautz, and Tim Weyrich. 2012. 3D-Printing of Non-Assembly, Articulated Models. ACM Transactions on Graphics 31, 6: 1–8. https://doi.org/10.1145/2366145.2366149

[6] Xiang "Anthony" Chen, Ye Tao, Guanyun Wang, Runchang Kang, Tovi Grossman, Stelian Coros, and Scott E. Hudson. 2018. Forte: User-Driven Generative Design. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18): 1–12. https://doi.org/10.1145/3173574.3174070

[7] Sathvik Divi, Xiaotian Ma, Mark Ilton, Ryan St. Pierre, Babak Eslami, S. N. Patek, and Sarah Bergbreiter. 2020. Latch-Based Control of Energy Output in Spring Actuated Systems. Journal of The Royal Society Interface 17, 168: 20200070. https://doi.org/10.1098/rsif.2020.0070

[8] Liang He, Huaishu Peng, Michelle Lin, Ravikanth Konjeti, François Guimbretière, and Jon E. Froehlich. 2019. Ondulé: Designing and Controlling 3D Printable Springs. In Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19): 739–750. https://doi.org/10.1145/3332165.3347951

[9] Jean Hergel and Sylvain Lefebvre. 2015. 3D Fabrication of 2D Mechanisms. Computer Graphics Forum 34, 2: 229–238. https://doi.org/10.1111/cgf.12555

[10] Alexandra Ion, Johannes Frohnhofen, Ludwig Wall, Robert Kovacs, Mirela Alistar, Jack Lindsay, Pedro Lopes, Hsiang Ting Chen, and Patrick Baudisch. 2016. Metamaterial Mechanisms. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16): 529–539. https://doi.org/10.1145/2984511.2984540

[11] Alexandra Ion, Ludwig Wall, Robert Kovacs, and Patrick Baudisch. 2017. Digital Mechanical Metamaterials. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17): 977–988. https://doi.org/10.1145/3025453.3025624

[12] Miyu Iwafune, Taisuke Ohshima, and Yoichi Ochiai. 2018. Coded skeleton: Shape Changing User Interface with Mechanical Metamaterial. SIGGRAPH Asia 2018 Technical Briefs, SA 2018: 4–7. https://doi.org/10.1145/3283254.3283255

[13] Hiroki Kaimoto, Junichi Yamaoka, Satoshi Nakamaru, Yoshihiro Kawahara, and Yasuaki Kakehi. 2020. ExpandFab: Fabricating Objects Expanding and Changing Shape with Heat. In Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '18): 153–164. https://doi.org/10.1145/3374920.3374949

[14] Robert Kovacs, Alexandra Ion, Pedro Lopes, Tim Oesterreich, Johannes Filter, Philip Otto, Tobias Arndt, Nico Ring, Melvin Witte, Anton Synytsia, and Patrick Baudisch. 2018. TrussFormer: 3D Printing Large Kinetic Structures. In Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18): 113–125. https://doi.org/10.1145/3242587.3242607

[15] Robert Kovacs, Lukas Rambold, Lukas Fritzsche, Dominik Meier, Jotaro Shigeyama, Shohei Katakura, Ran Zhang, and Patrick Baudisch. 2021. Trusscillator: A System for Fabricating Human-Scale Human-Powered Oscillating Devices. In Proceedings of the 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21): 1074–1088. https://doi.org/10.1145/3472749.3474807

[16] Jiahao Li, Meilin Cui, Jeeeun Kim, and Xiang Anthony Chen. 2020. Romeo: A Design Tool for Embedding Transformable Parts in 3D Models to Robotically Augment Default Functionality. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20): 897–911. https://doi.org/10.1145/3379337.3415826

[17] Jiahao Li, Jeeeun Kim, and Xiang "Anthony" Chen. 2019. Robiot: A Design Tool for Actuating Everyday Objects with Automatically Generated 3D Printable Mechanisms. In Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19): 673–685. https://doi.org/10.1145/3332165.3347894

[18] Jiahao Li, Alexis Samoylov, Jeeeun Kim, and Xiang "Anthony" Chen. 2022. Roman: Making Everyday Objects Robotically Manipulable with 3D-Printable Add-On Mechanisms. In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22): 1–17. https://doi.org/10.1145/3491102.3501818

[19] Nianlong Li, Han Jong Kim, Luyao Shen, Feng Tian, Teng Han, Xing Dong Yang, and Tek Jin Nam. 2020. HapLinkage: Prototyping Haptic Proxies for Virtual Hand Tools using Linkage Mechanism. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20): 1261–1274. https://doi.org/10.1145/3379337.3415812

[20] Hongnan Lin, Liang He, Fangli Song, Yifan Li, Tingyu Cheng, Clement Zheng, Wei Wang, and HyunJoo Oh. 2022. FlexHaptics: A Design Method for Passive Haptic Inputs Using Planar Compliant Structures. In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22): 1–13. https://doi.org/10.1145/3491102.3502113

[21] David Macaulay and Neil Ardley. 1998. The New Way Things Work. HMH Books for Young Readers; Revised, Subsequent edition (October 26, 1998).

[22] Robert Maccurdy, Robert Katzschmann, Youbin Kim, and Daniela Rus. 2016. Printable Hydraulics: A Method for Fabricating Robots by 3D Co-Printing Solids and Liquids. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation: 3878–3885. https://doi.org/10.1109/ICRA.2016.7487576

[23] Vittorio Megaro, Bernhard Thomaszewski, Maurizio Nitti, Otmar Hilliges, Markus Gross, and Stelian Coros. 2015. Interactive Design of 3D-Printable Robotic Creatures. ACM Transactions on Graphics 34, 6: 1–9. https://doi.org/10.1145/2816795.2818137

[24] Vittorio Megaro, Eth Zurich, Disney Research, Jonas Zehnder, Moritz Bächer, Stelian Coros, Markus Gross, and Bernhard Thomaszewski. 2017. A Computational Design Tool for Compliant Mechanisms. ACM Transactions on Graphics 36, 4: 82:1-82:12. https://doi.org/10.1145/3072959.3073636

[25] Niloy J. Mitra, Yong Liang Yang, Dong Ming Yan, Wilmot Li, and Maneesh Agrawala. 2010. Illustrating How Mechanical Assemblies Work. ACM SIGGRAPH 2010 Papers, SIGGRAPH 2010 3. https://doi.org/10.1145/1778765.1778795

[26] Jifei Ou, Zhao Ma, Jannik Peters, Sen Dai, Nikolaos Vlavianos, and Hiroshi Ishii. 2018. KinetiX-Designing Auxetic-Inspired Deformable Material Structures. Computers and Graphics (Pergamon) 75: 72–81. https://doi.org/10.1016/j.cag.2018.06.003

[27] Huaishu Peng, François Guimbretière, James McCann, and Scott E. Hudson. 2016. A 3D Printer for Interactive Electromagnetic Devices. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16): 553–562. https://doi.org/10.1145/2984511.2984523

[28] Raf Ramakers, Fraser Anderson, Tovi Grossman, and George Fitzmaurice. 2016. RetroFab: A Design Tool for Retrofitting Physical Interfaces Using Actuators, Sensors and 3D Printing. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16): 409–419. https://doi.org/10.1145/2858036.2858485

[29] Franz Reuleaux and Alexander Blackie William Kennedy. 1876. The Kinematics of Machinery. Macmillan.

[30] Thijs Jan Roumen, Willi Mueller, and Patrick Baudisch. 2018. Grafter: Remixing 3D-Printed Machines. In Proceedings of the 2018 Conference on Human Factors in Computing Systems (CHI '18): 1–12. https://doi.org/10.1145/3173574.3173637

[31] Andrew O. Sageman-Furnas, Nobuyuki Umetani, and Ryan Schmidt. 2015. Meltables: Fabrication of Complex 3D Curves by Melting. SIGGRAPH Asia 2015 Technical Briefs, SA 2015: 1–4. https://doi.org/10.1145/2820903.2820915

[32] Harpreet Sareen, Udayan Umapathi, Patrick Shin, Yasuaki Kakehi, Jifei Ou, Hiroshi Ishii, and Pattie Maes. 2017. Printflatables: Printing Human-Scale, Functional and Dynamic Inflatable Objects. In Proceedings of the 2017 Conference on Human Factors in Computing Systems (CHI '17): 3669–3680. https://doi.org/10.1145/3025453.3025898

[33] Adriana Schulz, Cynthia Sung, Andrew Spielberg, Wei Zhao, Robin Cheng, Eitan Grinspun, Daniela Rus, and Wojciech Matusik. 2017. Interactive Robogami: An End-to-End System for Design of Robots with Ground Locomotion. The International Journal of Robotics Research 36, 10: 1131–1147. https://doi.org/10.1177/0278364917723465

[34] Christian Schumacher, Steve Marschner, Markus Gross, and Bernhard Thomaszewski. 2018. Mechanical Characterization of Structured Sheet Materials. ACM Transactions on Graphics 37, 4: 1–15. https://doi.org/10.1145/3197517.3201278

[35] Peng Song, Xiaofei Wang, Xiao Tang, Chi-Wing Fu, Hongfei Xu, Ligang Liu, and Niloy J. Mitra. 2017. Computational Design of Wind-Up Toys. ACM Transactions on Graphics 36, 6: 1–13. https://doi.org/10.1145/3130800.3130808

[36] Hiroshi Sugihara. 2016. Ready to Crawl. Retrieved 2016: https://www.youfab.info/2016/winners/ready-to-crawl.

[37] Md. Farhan Tasnim Oshim, Julian Killingback, Dave Follette, Huaishu Peng, and Tauhidur Rahman. 2020. MechanoBeat: Monitoring Interactions with Everyday Objects Using 3D Printed Harmonic Oscillators and Ultra-Wideband Radar. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20): 430–444. https://doi.org/10.1145/3379337.3415902.

[38] Francisca Gil Ureta, Chelsea Tymms, and Denis Zorin. 2016. Interactive Modeling of Mechanical Objects. Eurographics Symposium on Geometry Processing 35, 5: 145–155. https://doi.org/10.1111/cgf.12971

[39] Ronald A. Walsh. 2009. Handbook of Machining and Metalworking Calculations. McGraw-Hill Professional.

[40] Guanyun Wang, Ye Tao, Ozguc Bertug Capunaman, Humphrey Yang, and Lining Yao. 2019. A-line: 4D Printing Morphing Linear Composite Structures. In Proceedings of the 2019 Conference on Human Factors in Computing Systems (CHI '19): 1–12. https://doi.org/10.1145/3290605.3300656

[41] Hongyi Xu, Espen Knoop, Stelian Coros, and Moritz Bächer. 2018. Bend-it: Design and Fabrication of Kinetic Wire Characters. SIGGRAPH Asia 2018 Technical Papers, SIGGRAPH Asia 2018 37, 6. https://doi.org/10.1145/3272127.3275089

[42] Humphrey Yang, Tate Johnson, Ke Zhong, Dinesh Patel, Gina Olson, Carmel Majidi, Mohammad Islam, and Lining Yao. 2022. ReCompFig: Designing Dynamically Reconfigurable Kinematic Devices Using Compliant Mechanisms and Tensioning Cables. In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22): 1–14. https://doi.org/10.1145/3491102.3502065

[43] Christopher Yu, Keenan Crane, and Stelian Coros. 2017. Computational Design of Telescoping Structures. ACM Transactions on Graphics 36, 4: 83:1–83:9. https:

//doi.org/10.1145/3072959.3073673
[44] Ye Yuan, Changxi Zheng, and Stelian Coros. 2018. Computational design of transformables. Computer Graphics Forum 37, 8: 103–113. https://doi.org/10.1111/cgf.13516
[45] Ran Zhang, Thomas Auzinger, Duygu Ceylan, Wilmot Li, and Bernd Bickel. 2017. Functionality-Aware Retargeting of Mechanisms to 3D Shapes. ACM Transactions on Graphics 36, 4: 1–13. https://doi.org/10.1145/3072959.3073710
[46] Clement Zheng, Zhen Zhou Yong, Hongnan Lin, HyunJoo Oh, and Ching Chiuan Yen. 2022. Shape-Haptics: Planar & Passive Force Feedback Mechanisms for Physical Interfaces. In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22): 1–15. https://doi.org/10.1145/3491102.3501829
[47] Lifeng Zhu, Weiwei Xu, John Snyder, Yang Liu, Guoping Wang, and Baining Guo. 2012. Motion-Guided Mechanical Toy Modeling. ACM Transactions on Graphics 31, 6: 1–10. https://doi.org/10.1145/2366145.2366146