Robotica (2017) volume 35, pp. 2238–2255. © Cambridge University Press 2017
doi:10.1017/S0263574716000850

Robust design of Passive Assist Devices for multi-DOF robotic manipulator arms

W. Robert Brown* and A. Galip Ulsoy

Department of Mechanical Engineering, University of Michigan, Ann Arbor, Michigan 48109, USA. E-mail: ulsoy@umich.edu

(Accepted December 20, 2016. First published online: February 10, 2017)

SUMMARY

A comparison of series, parallel, and dual Passive Assist Devices (PADs) designed using energy minimization based on a known maneuver is presented. Implementation of a PAD can result in an improvement in system performance with respect to efficiency, reliability, and/or utility. We introduce a new initial design using a weighted force displacement curve fit. A robust design approach for a family of maneuvers is developed and presented. Applications to a 3-link manipulator arm show that PADs could reduce energy consumption between 60% and 80%.

KEYWORDS: Design, Serial Manipulator Design and Kinematics, Mechatronic Systems, Mobile Robots, Sensor or Actuator Design

1. Introduction

Today, there are over 190,000 industrial robots operating in the United States and each typically consumes over 300 kWh per day.[1] A mere 1% improvement in efficiency would yield tremendous benefits. Our previous work demonstrated that adding a Passive Assist Device (PAD) (e.g., spring), which is designed based on a known maneuver, to a single link robot arm performing a basic task could realize an energy savings of 25%.[2] In this paper, we have expanded our method so that it can be applied to multi-DOF robot arms as well.

This approach could be particularly useful for unmanned ground vehicle (UGV) manipulator arms because the energy available on any given mission is limited. Reducing the energy requirements of arm manipulators allows for more power to be spent traversing ground, communicating, performing additional arm maneuvers, or increasing mission length. In a UGV manipulator, PADs can be designed to reduce the maximum motor torques required for a particular maneuver, thereby increasing energy efficiency. Reducing torques required by the motors will have the additional benefit of improving motor life and reliability, another mechanical shortcoming of UGVs.[2] Furthermore, we present a robust design approach where, if the maneuver can be characterized by a known distribution, a PAD can minimize the effects of maneuver variation on system performance.

1.1. Literature review

There are three noted design approaches for parallel PAD designs as follows: (a) combined maneuver-PAD optimization, (b) static balancing, and (c) a maneuver-based force displacement curve fit (FDCF). The walking robot ERNIE[3] uses springs acting parallel to the knee joints to reduce average power consumption. The design of ERNIE's springs is a three-step process that involves simultaneously optimizing the gait and one spring parameter. By contrast, we simultaneously optimize both spring variables based on a known maneuver in one step.

Much of the energy used to manipulate a robot arm is spent overcoming the weight of the arm itself. Parallel springs can be added to an existing manipulator, such that the potential energy of the arm is constant regardless of configuration. This results in an arm that is effectively weightless (i.e., statically balanced), drastically reducing peak torques and energy consumption.[4,5] Applications for

* Corresponding author. E-mail: wrbrown@umich.edu

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

static balancing include industrial robots,[6,7] exoskeletons,[8,9] as well as non-robotic devices, including an architect's desk lamp and a garage door assembly. Compared to statically balanced designs, our PAD designs are typically simpler and easier to implement while remaining effective because we consider the dynamics and external loading that define a typical maneuver in addition to the weight distribution and kinematics of the arm itself.[2]

Mettin *et al.* describe a parallel spring design process for augmenting active joints based on a prescribed trajectory and external forcing.[10] The spring design is a FDCF—a linear least squares approximation to the required force–displacement data. Adding a spring designed this way to a system performing this maneuver will reduce the required effort of the actuator, resulting in a decrease in energy consumption. While we typically use a force displacement curve fit (FDCF) design to initialize the optimization step in our design method, an energy minimizing design will avoid some of the potential limitations of an FDCF design, including the fact that for certain maneuvers the FDCF design could actually increase energy consumption.

A series PAD cannot be used for static balancing but can effectively improve energy efficiency by reducing kinetic energy losses.[11] In walking robots, this is done by attempting to mimic biologically equivalent systems[12,13] or by choosing the PAD stiffness, such that the natural frequency is typical of efficient natural gaits.[14] By contrast, we consider a prescribed maneuver (e.g., gait) and design a PAD that minimizes the energy required to execute such a maneuver without any of the approximation of the other methods and without altering the final maneuver. A closer approach to ours would be a robotic fish that uses a series spring to reduce energy consumption by aligning the phase of the harmonic tail velocity with the harmonic external hydrodynamic forces.[15] We consider a similar design approach, except the trajectory and forcing we are considering are more general than out of phase harmonic oscillations. Series PADs can also be designed to reduce backlash and shock loads while maintaining a necessary degree of control bandwidth.[16] These objectives are different from energy minimization but elements of this design approach could be incorporated into our methodology in the form of optimization constraints. Our method could also be extended to work on parallel linkage mechanisms.[17]

1.2. Original contributions

Our previous work described a novel, energy minimization, approach to design parallel PADs to augment active components (e.g., motors) and showed the benefits of our approach compared to two other state-of-the-art approaches: a linear FDCF based on a known maneuver[10] and a statically balanced design that utilizes non-linear components but does not consider the trajectory or loading.[4] Furthermore, we experimentally validated our approach on a single link robot arm.[2] Implementing a PAD designed with this approach will directly improve system energy efficiency and may also improve reliability and/or utility.[2]

The next three sections illustrate our design approach and highlight several original contributions that extend upon previous work. In Section 2, we provide a general six-step methodology to design PADs. This methodology is explained via a generic mass–spring–actuator system. We extend our existing maneuver-based design method for parallel PADs to series and dual PAD designs and provide a superior initial design for optimization beyond the traditional FDCF.

In Section 3, we show how our approach can be used on multi-DOF systems, such as a 3-link robot manipulator arm. We also consider a family of possible maneuvers and design PADs that are robust with respect to maneuver variation.

Finally, we provide a summary, conclusions, and a brief discussion of future work.

2. Method

In this section, we present a general, maneuver-based methodology to design PADs, demonstrate the analogous nature of series and parallel design, and provide a superior initial design for optimization compared to FDCF. We explain the steps of this method via a mass–spring–actuator example (see Fig. 1). In Section 3, this method is applied to a more complex system demonstrating its general utility.

Several subscripts are used throughout this chapter. The subscripts $_a$ and $_k$ correspond to the actuator and spring. The subscripts $_s$, $_p$, and $_d$ refer to specific series, parallel, and dual configurations,

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

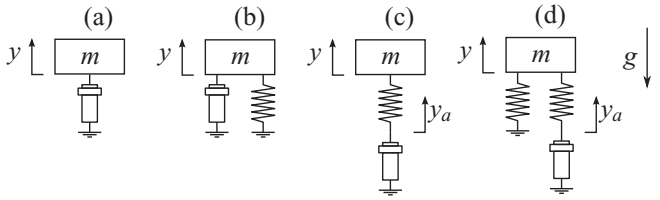

Fig. 1. An unsprung, actuated mass (a), the same system augmented with a parallel spring (b), the same system as (a) augmented with a spring in a series configuration (c), and the system augmented with a dual spring configuration (d).

respectively. For example, F is the force required to move the mass, P_{a_p} is the power required of the actuator in a parallel configuration, and $\dot{y}_{k_s}$ is the spring velocity for a series spring.

2.1. General maneuver-based design methodology

The primary objective of our research is to develop a method to improve the performance of active systems with optimized PADs. The key aspect of our approach is that the design is based on a known maneuver (i.e., trajectory and loading). This general method can be applied to a wide range of specific systems performing specific tasks by following well-defined steps. Depending on the complexity of the system being optimized, some of the steps may be trivial but for a general case they are all important. These steps are described below and illustrated via a mass–spring–actuator example (see Fig. 1).

2.1.1. Define machine architecture. Define the architecture of the machine that you wish to augment with a PAD: number of degrees of freedom, link or axis mass and inertia information, and link or axis dimensions.

The simplest machine we could augment is a one-dimensional translational mass–actuator system as depicted in Fig. 1(a) with mass, m.

2.1.2. Define trajectory and external forcing. The trajectory and external forcing of the machine (collectively referred to as the maneuver in this paper) must be known in order to design PADs with this method. This method is well suited to improve dedicated manufacturing machines, where the maneuver is precisely repeated thousands or millions of times. However, a substantial increase in performance may still be realized if the maneuver is not precisely known, so long as the prescribed maneuver is typical of actual maneuvers.[2,18]

To demonstrate this method on the simple machine of Fig. 1, we consider a specific trajectory, $y(t)$, to be a piece-wise smooth mix of sinusoidal acceleration zones (where the mass reverses direction) and constant velocity zones. The trajectory is parameterized by the frequency, ω, and amplitude, a, of the sinusoidal portions, and the ratio of constant velocity duration to sinusoidal duration, n.

$$y(t) = y_0 + \begin{cases} a\sin(\omega t) & 0 \le t < \pi/\omega \\ -a(\omega t - \pi) & \pi/\omega \le t < (n+1)\pi/\omega \\ a\sin(\omega t - n\pi) - an\pi & (n+1)\pi/\omega \le t < (n+2)\pi/\omega \\ a(\omega t - 2(n+1)\pi) & (n+2)\pi/\omega \le t < 2(n+1)\pi/\omega \end{cases} \tag{1}$$

where $y_0 = a\omega n\pi/2$, so that the position is evenly distributed about the y-axis. Increasing ω or a increases the severity of the acceleration of the mass. As $n \to 0$, the trajectory becomes harmonic. The specific maneuver we are using to illustrate this method is comprised of both the trajectory $y(t)$ and a constant external load.

In this design approach, it is assumed that the manipulator exactly follows the prescribed maneuver. In reality, there will be some tracking error due to the controller. However, if the controller is well designed, this is small and can be ignored.[2]

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

2.1.3. Perform inverse dynamics/decouple joints. Given the machine's architecture, trajectories of all joints (or axes), and the external forcing, a set of joint torque-angle (or axis force–displacement) profiles can be calculated using inverse dynamics. This step transforms an n-degree of freedom (DOF) system into n single DOF systems. PADs can then be designed for each joint independently. This step is simple for single DOF systems but is critical for more complex machines, such as the 3-DOF robot arm presented in Section 3.

The force required from the actuator to execute the prescribed maneuver of the mass–actuator example is $F = m\ddot{y} + F_c$, where F_c is the constant external load (e.g., weight).

2.1.4. Select PAD topology. There are many different PAD configurations that could be considered, including springs, pistons, pressurized tanks, capacitors, etc. For a moderately sized mechanical system, a linear spring is a practical choice for a PAD.[2] More complex PADs, including non-linear magnetic springs[19] or compliant parallelogram mechanisms[20] could be designed with the same general procedure as demonstrated here for linear springs.

In the case of the mass–spring–actuator example, we will demonstrate the design methodology with parallel, series, and dual PAD configurations (see Fig. 1b–d). In each case, the PAD will be a linear extension spring with stiffness, k, and preload, F_0:

$$F_k = ky_k + F_0 \tag{2}$$

2.1.5. Define powertrain. In order to design a PAD that minimizes an objective function (see below), the parameters defining the joint powertrain (e.g., gears, non-backdrivable (NBD) elements, and friction losses) must be known beforehand to accurately predict and design PADs for energy saving purposes.[2]

In the simple mass–actuator example, the actuator is considered to be an ideal motor—it can always supply the required force and velocity and has no inefficiencies.

2.1.6. PAD optimization. The final step of this methodology is optimization of the PAD design. We define an objective function, $f(\mathbf{x})$, corresponding to some performance metric that we wish to minimize with the addition of an optimal PAD, where $\mathbf{x}$ is a vector containing the design variables of the PAD (e.g., spring stiffness and preload).

In this paper, we will typically look for PAD designs that minimize actuator energy consumption. Thus, we formulate the following constrained optimization problem:

$$\begin{aligned} \text{minimize} \quad & f(k, F_0) = \int_0^{t_f} |P_a(t, k, F_0)| \, \mathrm{d}t \\ \text{with respect to} \quad & k, F_0 \\ \text{subject to} \quad & k \geq 0 \\ \text{and} \quad & t_f \geq t \geq 0, \end{aligned} \tag{3}$$

where P_a is the required actuator power. The form of P_a will change depending on the type of PAD being optimized.

We solved Eq. (3) with MATLAB's built in function *fmincon*, a gradient-based constrained optimization solver, initialized with an FDCF design.

2.2. Parallel vs. series vs. dual spring design

In the following subsections, we will demonstrate the analogous approach to the design of PADs in parallel, series, and dual configurations. We will also demonstrate how to choose an initial design based on the known loading and trajectory of the joint.

2.2.1. Parallel spring design. The basic design objective described in this paper is to minimize required actuator energy (see Eq. (3)) to execute a given maneuver (see Eq. (1)) by adding a PAD. Because the actuator is ideal in the mass–actuator example, the required power is as follows:

$$P_a = \dot{y}_a F_a \tag{4}$$

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

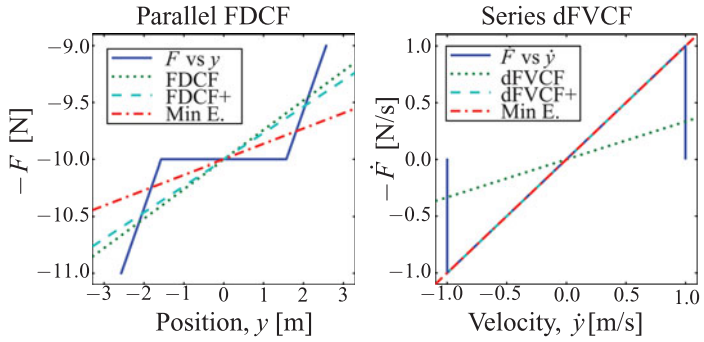

Fig. 2. (a) The required force to move the mass through the prescribed trajectory vs. the corresponding position of the mass (solid blue). Three parallel linear extension springs designs are overlaid: a force–displacement curve fit (FDCF; dot green), weighted FDCF (FDCF+; dash cyan), and energy minimization (dash–dot red). (b) The analogous data and spring designs for a series configuration.

For a parallel configuration, the spring extension, y_k, and the actuator position, y_a, are both equal to the mass position, y, while the actuator force, $F_a = F + F_{k_p}$.

$$P_{a_p} = \dot{y}(F + F_{k_p}) \tag{5}$$

As y is prescribed, we can only reduce the energy required by making $F + F_{k_p}$ as close to 0 as possible as often as possible:

$$-F \sim k_p y + F_0 \tag{6}$$

F and y are both prescribed by the maneuver and can be plotted against each other as in Fig. 2(a). The linear least-squares curve fit that best fits the F vs. y data will have a slope of k_p and a constant offset of F_0. We refer to this design as an FDCF.[10]

Implementing an energy minimizing parallel PAD can greatly increase efficiency. Figure 3(a) shows the power profiles of the actuator without a parallel PAD as well as with three different PAD designs for a maneuver with $n = 1$ and $C = 10$. Figure 3(b) more clearly shows the effect of the different PAD designs. As expected, the optimal design performs the best, whereas FDCF is not quite as good. The weighted force displacement curve fit (FDCF+) is described in the next section and provides a performance that is intermediate. When the maneuver is altered such that $n = 0$, the maneuver becomes harmonic and the PAD completely eliminates actuator energy consumption. However, if the maneuver is altered such that $n = 1$ and $C = 0$, a parallel PAD can only achieve modest energy savings. This suggests that parallel PADs are well suited for maneuvers where external forces are large compared to inertial forces.

2.2.2. Series spring design. A series PAD can be designed in a manner completely analogous to the parallel design. A series PAD transmits the force required to move the mass, F, through the spring directly to the actuator, and therefore has no effect on actuator force, $F_a = F$. The series spring extension, $y_{k_s} = y - y_a$; thus, the required actuator power is as follows:

$$P_{a_s} = (\dot{y} - \dot{y}_{k_s})F \tag{7}$$

As F is prescribed, we reduce the energy required by making $\dot{y}_{k_s}$ as close to $\dot{y}$ as possible as often as possible. To get $\dot{y}_{k_s}$ in terms of the series spring parameters, we take the time derivative of the spring constitutive relation (see Eq. (2)):

$$\dot{F}_{k_s} = k_s \dot{y}_{k_s} \tag{8}$$

Note that the constant spring preload, F_0, disappears. The extension force in the spring is equal and opposite to the force required to move the mass, $F_{k_s} = -F$. This yields a derivative of force velocity

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

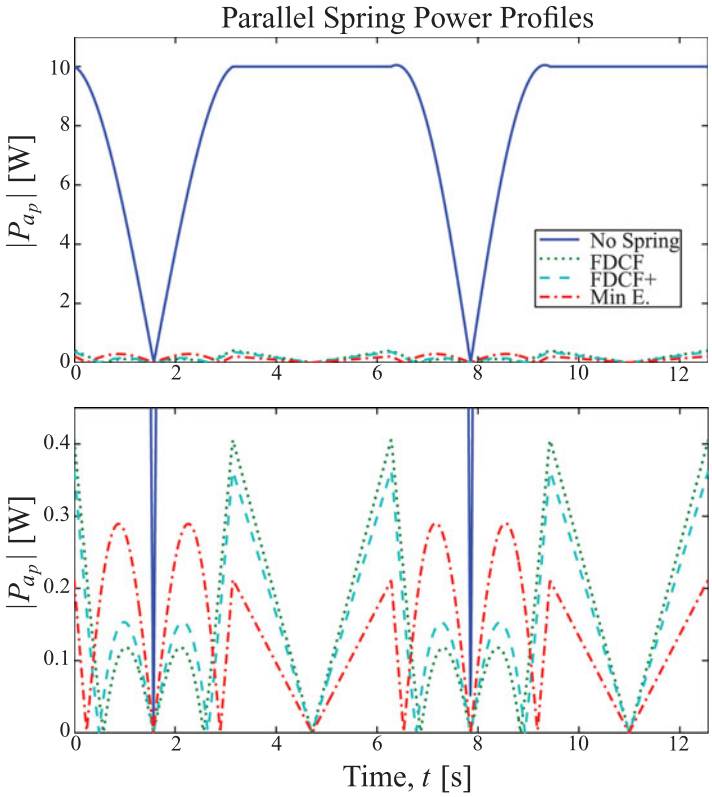

Fig. 3. Required actuator power to execute the maneuver with no spring (solid blue) and three parallel springs. The top figure depicts the total power, whereas the bottom figure zooms in to examine the different performances of the three spring designs.

curve fit (dFVCF):

$$-\dot{F} \sim k_s \dot{y} \tag{9}$$

This is the same form as Eq. (6) except with $\dot{F}$ and $\dot{y}$ instead of F and y and there is no F_0 term. The series spring preload could be important for implementation and packaging reasons but has no effect on actuator power consumption. The dFVCF design will often perform well and can be used as an initial guess in a gradient-based optimization routine.

2.2.3. Power weighted curve fit designs. For certain maneuvers, the dFVCF design is worse than having no spring at all. It dramatically increases energy consumption (see Fig. 4) and serves as a poor initial guess for optimization. The maneuver explored in this example is simple enough that the effect of a series spring design on actuator energy consumption, $f(k)$, can be calculated analytically by substituting the selected maneuver (1) into the series power Eq. (7) and integrating over the duration of the maneuver:

$$f(k_s) = \begin{cases} 2a\left|1 - \dfrac{\omega^2 m}{k_s}\right|\left(\dfrac{C^2}{B} + B\right) + 2aC\pi n & |C| \le B \\ 4a\left|1 - \dfrac{\omega^2 m}{k_s}\right| C + 2aC\pi n & |C| \ge B \end{cases} \tag{10}$$

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

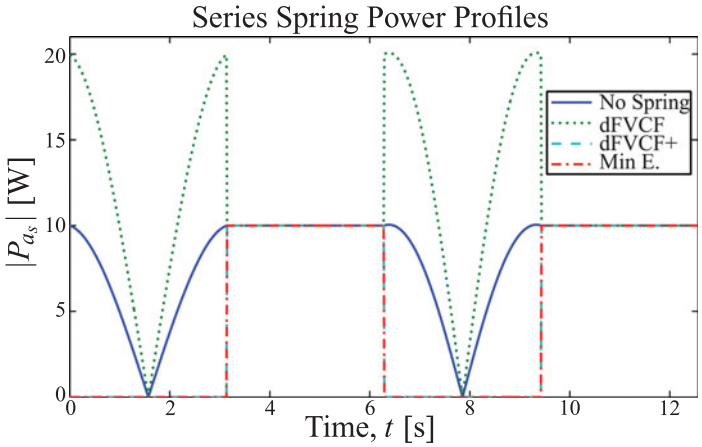

Fig. 4. Required actuator power to execute the maneuver with no spring (solid blue) and three series PADs. No series PAD can affect the power consumption during periods of constant velocity. Furthermore, implementing the unweighted dFVCF design (dot green) substantially increases energy consumption. The weighted derivative of force velocity curve fit (dFVCF+) (dash cyan) and energy minimizing design (dash–dot red) are the same and both eliminate actuator power during the sinusoidal regions of the trajectory.

where $B := a\omega^2 m$. This leads to different performance regimes based on the spring stiffness:

$$\begin{aligned} k_s &= 0 && \text{disconnected joint} \\ 0 < k_s &< \omega^2 m/2 && \text{reduced performance} \\ k_s &= \omega^2 m/2 && \text{neutral performance} \\ \omega^2 m/2 < k_s &< \infty && \text{improved performance} \\ k_s &= \omega^2 m && \text{optimal performance} \\ k_s &\to \infty && \text{no spring} \end{aligned} \tag{11}$$

The dFVCF design lies within the reduced performance regime. This is because the vertical "spikes" or "tails" in Fig. 2b correspond to points in the maneuver, where the mass is moving at constant speed ($\ddot{y} = 0$) and experiencing a constant external force ($\dot{F} = 0$). A series spring cannot affect actuator power under these conditions. Therefore, trying to fit the spring design to those data points only has the effect of pulling the design away from regions where the spring could effectively reduce power consumption. An analogous problem could occur for parallel designs if the prescribed maneuver had a stationary segment with changing external loads. In both cases, a curve fit spring design would perform much better if it were able to ignore data points where the spring would be ineffective.

The dFVCF design is the linear, least-squares curve fit to the derivative of force vs. velocity data (Eq. 9). We propose a new curve fit design, dFVCF+ (parallel analogue: FDCF+), which uses a linear, *weighted*, least-squares curve fit. By making the weighting, W, proportional to the corresponding spring power, P_k, we can find a PAD design that typically outperforms its unweighted counterpart because the weighting is able to de-emphasize data points where the PAD cannot be as effective (e.g., the "tails" in Fig. 2b).

For a series system, the equation for actuator power (Eq. 7) can be expressed as

$$P_{a_s} = P - P_{k_s} \tag{12}$$

where P_{a_s} is the actuator power in the series configuration, P is the power required to move the mass, and $P_{k_s} = \dot{y}_{k_s} F$. Combining this result with Eq. (8) yields

$$P_{k_s} = F\dot{F}/k_s \tag{13}$$

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

When P_{k_s} is small, it will have little effect on P_{a_s}; therefore, the curve fit should weight the data with

$$W_{i_s} = F_i \dot{F}_i \tag{14}$$

The analogous weighting for an FDCF+ design would be

$$W_{i_p} = y_i \dot{y}_i \tag{15}$$

We can then use the dFVCF+ design to initialize the series optimization.

Due to the simple nature of this example maneuver, the dFVCF+ design is the same as the optimal series PAD design. This is because the dFVCF+ design ignores the constant velocity data ($W_{i_s} = 0$ in those regions) and can completely compensate for the joint velocity in the sinusoidal regions. Regardless of constant external loading, an energy minimizing series spring can completely eliminate actuator velocity (and, thus, power) for a harmonic trajectory ($n = 0$). However, in the example maneuver where $n = 1$ and $C = 10$ N, a series PAD can only achieve modest energy savings. This suggests that series PADs are well suited for maneuvers where inertial forces are large compared to external forces.

2.2.4. Dual spring design. In addition to augmenting a joint with a parallel or series PAD, a joint may be augmented with both types of PADs simultaneously (see Fig. 1d). This would be the closest system to animal skeletal muscle systems that are well modeled by springs acting in parallel (internal to muscle tissue) and in series (tendons) to linear actuators (muscles).[21]

We can design the dual PAD in a manner similar to the parallel and series designs. The actuator power of a dual PAD system is as follows:

$$P_{a_d} = (\dot{y} - \dot{y}_{k_s})(F + F_{k_p}) \tag{16}$$

The actuator power, P_{a_d}, is affected by the parallel spring force, F_{k_p} in Eq. (2), and by the series spring velocity, $\dot{y}_{k_s}$ in Eq. (8). The energy consumed is now a function of three variables: parallel stiffness, parallel preload, and series stiffness. These variables can be initialized for an optimization routine by using the FDCF+ design for the two parallel variables and the dFVCF+ design for the series stiffness.

An optimized dual system can always outperform (or match) either an optimized parallel or series design because the dual system has additional variables compared to either single PAD design. As it turns out, for the class of trajectory described in this example (Eq. (1)) with constant external forcing, an energy minimizing dual PAD can always completely eliminate actuator energy consumption if the parallel PAD is a constant force spring with $F_0 = -C$ and the series stiffness is $k_s = \omega^2 m$. In general, this will not be the case.

2.2.5. Design selection and reduction. From an energy perspective, different PAD types are preferable for different maneuvers. Series PADs tend to be superior to parallel PADs for maneuvers where the inertial forces dominate the external ones and vice versa. Furthermore, a dual PAD can always outperform a single PAD design. We propose three possible ways to select a PAD type for a general maneuver.

The first decision may be made based on practical constraints. Is this PAD intended to augment an already existing system? And if so, can a series or dual PAD easily be implemented? If the maneuver has a large change in position, a parallel spring could encounter wind-up problems. On the other hand, if precise position control is essential, series and dual PADs may provide unacceptable performance.

If any of the PAD types could be acceptable for non-energy reasons, then a decision needs to be made considering potential gains in performance against potential costs of design and implementation. A simple way to do this is to directly compare the three energy minimizing designs. The obvious advantage of this approach is that it provides a clear summary of effectiveness of each design. The disadvantage of this approach is the computational cost as three optimizations are required. Although this cost is trivial for a simple system and a single maneuver, it could be significant if non-linear PADs are considered, if the system has many joints that need PADs, or if the objective is to make a PAD robust against variations in maneuvers.

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

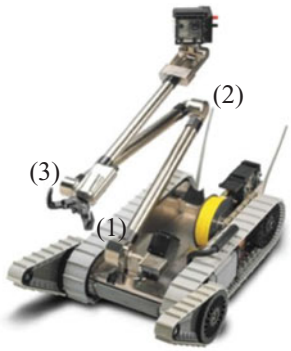

Fig. 5. A Packbot UGV with a 3-link manipulator arm. The arm has three joints: Joint 1 connects the arm to the chassis, Joint 2 is the first elbow, and Joint 3 is the second elbow. The arm has a camera at the end of the third link and a gripper located at Joint 3. Photo: iRobot.com

For the example presented here, this is a good method for comparing three spring designs because optimization can be done very quickly, even for the dual PAD. An energy minimizing series PAD can reduce energy consumption by 38.9%, a parallel PAD by 98.3%, and a dual PAD by 100%. The series PAD is clearly inferior based on this criteria but both the dual and parallel designs perform very well. If energy efficiency is paramount, then a dual PAD should be selected. However, the parallel design performs almost as well as the dual spring design and is simpler and easier to implement making it an attractive choice as well.

For more complex systems, PADs, maneuvers, or objective functions, performing three design optimizations in order to select a single design may be cost prohibitive. If this is the case, a PAD type can be selected based on the results of weighted curve fits. (FDCF+ for parallel, dFVCF+ for series, FDCF+ and dFVCF+ for dual). Any design that has a minimal effect on energy can be eliminated from contention. Finally, if the weighted curve fit dual design fails to provide at least a moderate improvement over the superior of the two single spring designs, it may not be worth optimizing.

3. Three-Link Robot Arm Example

Our general PAD design approach can be applied to more complex systems, such as a 3-link robotic manipulator arm. This method can be particularly useful for UGV manipulator arms because the energy available on any given mission is limited. Reducing the energy requirements of arm manipulation allows for more power to be spent traversing ground, communicating, performing additional arm maneuvers, or increasing mission length. In a UGV manipulator, a PAD (e.g., torsion spring) can be placed in each joint and designed to reduce the electrical energy required for a particular maneuver.

A UGV manipulator arm also distinguishes itself from its industrial cousins because a UGV arm operates in an open environment and performs a variety of tasks instead of precisely repeating maneuvers. However, such tasks can often be restricted to a family of similar trajectories and loads. Consequently, in this section we show that properly designed PADs can significantly reduce peak torques and, thus, improve reliability and energy efficiency so long as the maneuver being performed is close to the maneuver the PAD was designed for. Furthermore, we present a robust design approach where, if the maneuver can be characterized by a known distribution, a PAD can minimize the effects of maneuver variation on system performance.

3.1. Apply PAD design methodology

In this section, we apply the general, six-step design method described in Section 2.1 to improve the performance of a 3-link UGV manipulator arm with optimized PADs. The arm we will consider is similar to an iRobot Packbot arm (see Fig. 5). Parameters characterizing this system are listed in Table I. In this section, we explore the effect of maneuver variations on PAD design and performance.

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

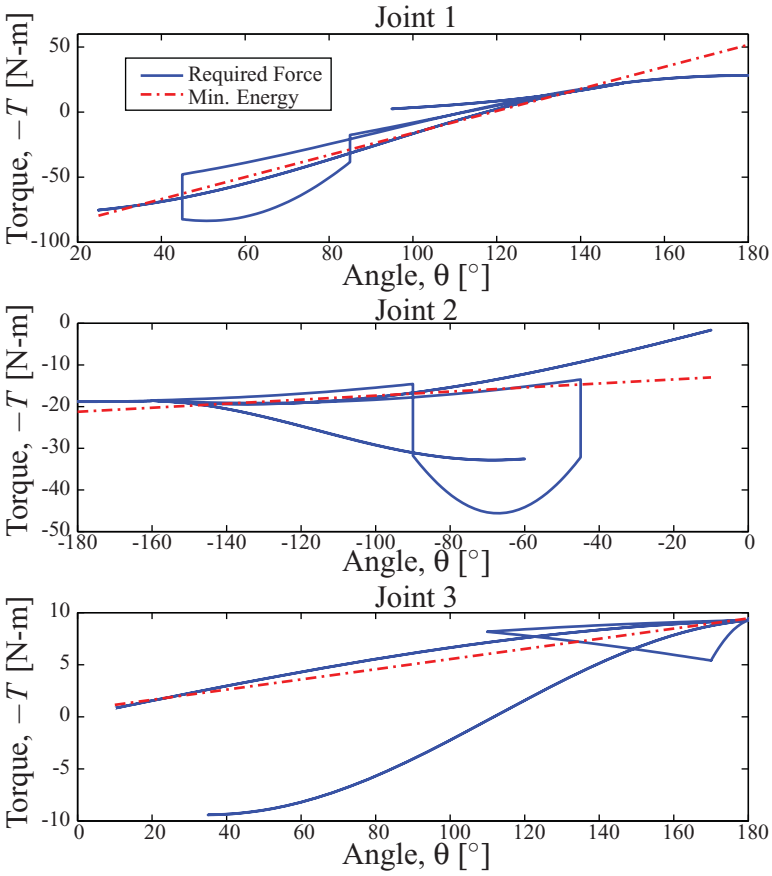

Fig. 6. For the 3-link UGV arm described in Table I to execute the maneuver described in Table II, the joints must follow the torque-angle profiles (solid blue). The PADs that minimize the energy consumed by each motor (dash–dot red) can be designed for each joint independently. An analogous set of derivative of torque–angular velocity plots can be produced for series PAD design.

The maneuvers considered in the chapter are described in Section 3.2 and are intended to be typical of tasks such an arm is likely to perform.

The 3-link UGV arm described in Table I, executing the maneuver described in Table II, and Fig. 7 needs to produce the torque-angle profiles at each joint (solid blue) shown in Fig. 6. The PAD can supply some of the torque required to execute the maneuver (dash–dot red), reducing the load required of the motor and the energy consumed. However, the total torque acting on the joint remains the same and, thus, the PAD design of one joint has no effect on the design of PADs for other joints. Although performing inverse dynamics on a multi-link robot arm is not trivial, it can be easily and quickly accomplished using available techniques[22,23] or software packages such as Corke's robotics toolbox for MATLAB, which performs inverse dynamics with a Newton–Euler recursion algorithm.[24]

Because each joint is rotational, we will design PADs that are linear torsion springs of stiffness, k, with preload, T_0, with the following torque characteristic:

$$T_k = k\theta + T_0 \tag{17}$$

A typical UGV arm joint powertrain could include the following components: non backdrivable (NBD) worm gear, gear head, and DC motor. In addition to providing a speed reduction, the worm gear allows the arm to remain stationary in a raised configuration without expending any energy. The torque needed to drive the worm, T_w, of an NBD worm gear transmission given the required torque

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

Table I. Mass, length, and powertrain parameters for the worm gear and gear head were chosen by reverse engineering a single link robot arm. Motor parameters were chosen by examining data sheets for DC motors and selecting one that was big enough to handle the peak power, torque, and speed requirements.[26] The same parameters are used for each link and joint in the robot arm.

Parameter	Definition	Value
m_l	Link mass	3.1 kg
l	Link length	62 cm
i_{wg}	Worm gear speed ratio	18.7
C_1	Load-driven torque ratio	$-1/128$
C_2	Motor-driven torque ratio	1/8
n	Gear head speed ratio	30
R	Resistance	0.41 Ω
k_m	Torque constant	52 mN-m/A
T_f	Motor friction	4.9 mN-m

at the joint, T, can be calculated from the following equations of motion:[25]

$$T_w = \begin{cases} 0 & \text{if stuck} \\ (J_w i_{wg} + C_1 J_g)\ddot{\theta}_a + C_1(T_a) & \text{if load driven} \\ (J_w i_{wg} + C_2 J_g)\ddot{\theta}_a + C_2(T_a) & \text{if motor driven} \end{cases} \tag{18a}$$

$$\dot{\theta}_w = \dot{\theta}_a \times i_{wg} \tag{18b}$$

where J_w, J_g are the inertia of the worm and worm gear, respectively, i_{wg} is the speed ratio, and C_1, C_2 are the torque ratios of the load-driven and motor-driven cases, respectively. Which operating condition is active can be determined at each time step.[2] T_a and θ_a are the torque and position required, respectively, by the worm gear to maneuver the joint. Their values vary depending on whether the PAD is in a parallel or series configuration as described in Sections 2.2.1 and 2.2.2.

The gearhead and DC motor can be modeled as in our previous work for a single-joint robot arm:[2]

$$i = \frac{T_w/n + T_f \times \text{sgn}(\dot{\theta}_w)}{k_m} \tag{19a}$$

$$V = iR + k_m \times \dot{\theta}_w \times n \tag{19b}$$

$$P_e = V \times i \tag{19c}$$

where i, V, and P_e are the motor current, voltage, and electrical power required at each time step, n is the gear ratio, and R, k_m, and T_f are the motor resistance, torque constant, and internal friction torque, respectively.

In this example, we will initially seek a linear torsion spring that minimizes motor energy consumption. This is the same as the gradient-based optimization problem (3) except considering $f = P_e$ instead of P_a, and we add motor performance limit constraints to the problem. These constraints are typically inactive but including them ensures that we are only testing feasible maneuvers. We solve this problem with MATLAB's built in function *fmincon* initialized with an FDCF+ or dFVCF+ design for parallel and series PADs, respectively.

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

Table II. The maneuver is described by 10 parameters: 9 parameters describe the time to move from one stage to another (e.g., t_{AB} is the time to move from storage (A) to driving (B)) while the last parameter, m, is the mass lifted from stage (E_1) to (E_2).

Parameter	Mean	Std. Dev.	Parameter	Mean	Std. Dev.
t_{AB}	5 s	0.5 s	t_{BE_1}	15 s	3 s
t_{BC}	20 s	3 s	$t_{E_1E_2}$	30 s	5 s
t_{CB}	30 s	4 s	t_{E_2B}	25 s	4 s
t_{BD}	20 s	3 s	t_{BA}	10 s	1 s
t_{DB}	30 s	4 s	m	4 kg	1 kg

Table III. A performance comparison of how three different PAD types affect motor energy consumption in each joint of a three link manipulator arm. Each design minimizes motor energy consumption. The parallel torsional stiffness, k_p, parallel preload, T_0, series torsional stiffness, k_s, energy consumed, f, and energy saved compared to a springless design, S, are cataloged.

	PAD type	k_p $\left[\frac{\text{N-m}}{\text{rad}}\right]$	T_0 [N-m]	k_s $\left[\frac{\text{N-m}}{\text{rad}}\right]$	f [J]	S [%]
Joint 1	Parallel	49.6	−104.0	–	219.8	78.9
	Series	–	–	45.6	639.9	38.5
	Dual	46.8	−99.8	39.9	103.4	90.1
Joint 2	Parallel	1.1	−16.5	–	205.4	65.4
	Series	–	–	∞	593.4	0
	Dual	1.1	−16.5	∞	205.4	65.4
Joint 3	Parallel	2.8	0.5	–	98.5	39.8
	Series	–	–	7.1	109.9	32.8
	Dual	2.8	0.9	9.1	51.6	68.4

3.2. PAD design for a nominal maneuver

UGVs operate in uncertain environments and their manipulator arms are expected to be able to execute many different maneuvers. However, some types of maneuvers are much more common than others. For example, the arm is always stored in the same configuration (see (A) in Fig. 7) and the arm is typically raised slightly (B) while driving as to provide a better camera angle for the operator. The arm may also be used to lift the camera as high as possible (D) to achieve a superior point of view while stationary, to check under vehicles (C), and to lift an object from the ground (E_1) onto a higher surface (E_2). Of course, the arm could be used to perform other tasks, but it usually will be performing a task similar to the ones listed above. We define a nominal maneuver that progresses through the following configurations: A–B–C–B–D–B–E_1–E_2–B–A. When at each stage the arm is at rest and when moving between stages the joint angles follow a fifth-order polynomial trajectory. The time to execute each submaneuver and the lifted mass are listed in Table II. For this nominal maneuver, only the mean values of each parameter are used.

Table III summarizes the potential improvement of adding different energy minimizing PAD types to the various joints of a 3-link manipulator arm. Joint 1 connects the arm to the base, and the gripper is located at Joint 3. Figure 8 illustrates the effect that adding an energy minimizing parallel PAD has on the motor of Joint 1. The spring is able to supply much of the required torque, reducing the effort required from the motor. Consequently, the required power decreases and energy consumption is reduced by almost 80%. An energy-minimizing series PAD does not perform as well but can still reduce energy by almost 40%. The dual design combines a parallel and a series spring that are slightly softer than their single spring counterparts and reduces energy consumption by 90%.

The motor in Joint 2 can be aided substantially by a parallel PAD. However, a series spring is unable to provide a benefit. Thus, the energy minimizing series spring has infinite stiffness, meaning that the best series spring is no spring. The dual spring design is the same as a parallel only design for this joint. The energy required of the Joint 3 motor can be reduced by 39.8% with a parallel PAD, 32.8% with a series PAD, and almost 70% with a dual PAD.

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

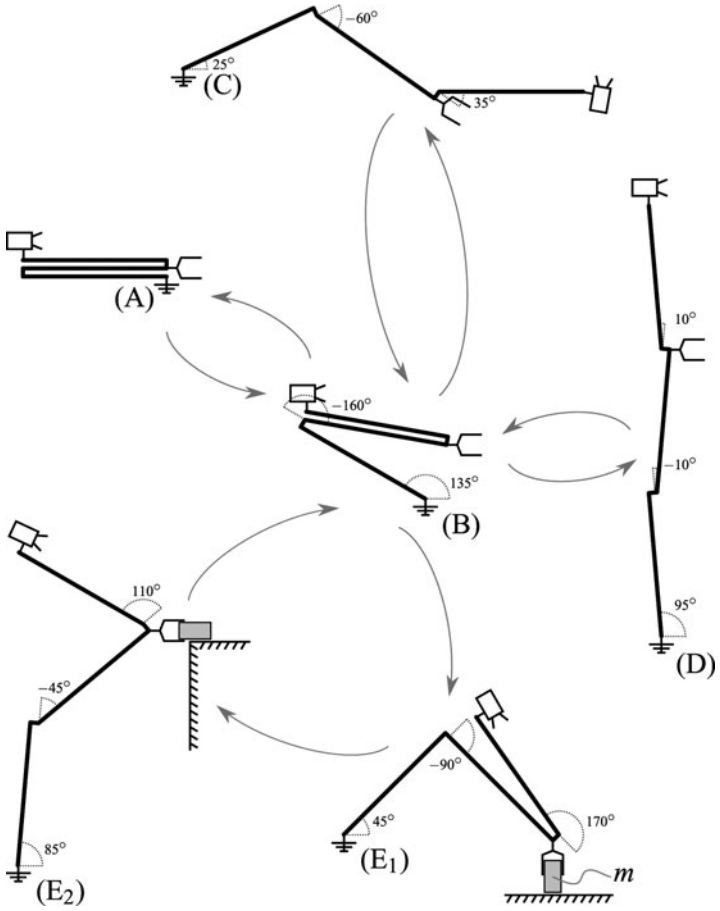

Fig. 7. Typical operating states of a UGV arm including storage (A), driving (B), looking under an object (C), looking from a raised vantage point (D), and lifting an object from (E_1) to (E_2). Joint 1 connects the arm to the base and the gripper is located at Joint 3.

Joint 2 should be augmented with a parallel PAD. Parallel springs are also effective options for Joints 1 and 3 resulting in a total energy savings of 67.9% for the entire manipulator arm. Alternatively, using dual PADs for Joints 1 and 3 results in a total energy savings of 77.9%. This further reduction may be enough to justify the increased complexity and cost of using dual PADs instead of parallel ones in Joints 1 and 3. However, for the remaining analysis in this section, we will only consider parallel PADs at each joint.

The relative energy savings are significantly higher than in previous work.[2] This is mostly because the torque angle data for the robot arm joints in this paper can be more closely approximated by a linear torsion spring than the corresponding data in previous work where a single joint was unloaded during descent and loaded during ascent. Different powertrain values also account for some of the difference. Optimizing the PAD and powertrain simultaneously will be a subject of future work.

3.3. Effect of increased maneuver variability

In order for a PAD to be of real use, it needs to provide significant assistance for all likely maneuvers, not just one. In general, as maneuver variability goes up, the number of maneuvers where the nominal energy minimizing PAD provide assistance will decrease. To demonstrate this point, we consider a maneuver distribution: each nominal submaneuver is altered such that each parameter is described by a normal distribution with mean and standard deviation as listed in Table II. Furthermore, instead of executing one raised arm submaneuver (B–C–B), one lowered arm submaneuver (B–D–B), and one lifting submaneuver (B–E_1–E_2–B), the arm executes three submaneuvers and each one has a 1/3 chance to be a raised arm, lowered arm, or lifting submaneuver (e.g., 1/27 of maneuvers in this

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

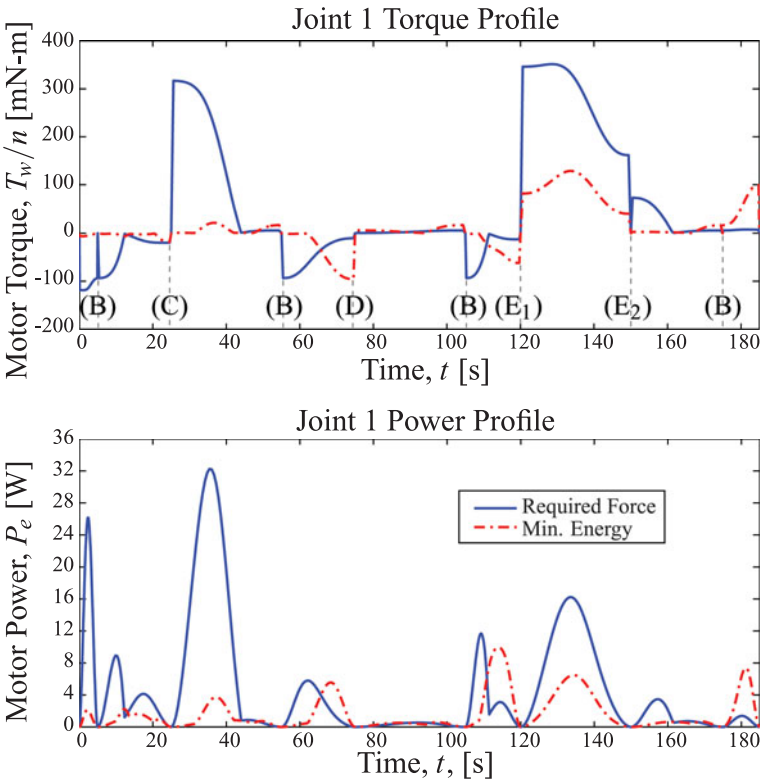

Fig. 8. Required motor torque (top) and power (bottom) to execute the nominal maneuver with no spring (solid blue) and with an energy-minimizing spring (dash–dot red). The maneuver starts and ends in a stored configuration (A) (see Fig. 7) and progresses between the other configurations as shown on the motor torque plot.

distribution involve lifting a load three times). Each distribution is implemented via a Monte Carlo approach, where performance metrics are calculated based on 1000 individual maneuvers.

To compare PAD performance across different maneuvers or maneuver distributions, we consider several performance metrics. First, the energy required to execute a single maneuver is as follows:

$$E = \int_0^{t_f} |P_e(t, k, T_0)| \, \mathrm{d}t \tag{20a}$$

Similarly, $\mathbf{E}$ is a vector containing the energy required to execute each simulated maneuver within a distribution such that

$$\mathbf{E}_i = \int_0^{t_f} |\mathbf{P}_{\mathbf{e}i}(t, k, T_0)| \, \mathrm{d}t \tag{20b}$$

The objective of our nominal optimization, (3), is to minimize E. $\bar{\mathbf{E}}$ and $\sigma(\mathbf{E})$ are the average and standard deviation of energy consumed over a distribution of maneuvers, respectively.

We also consider the percent energy saved by adding a PAD compared to a no-spring design for a single maneuver:

$$S = 1 - E(k, T_0)/E(0, 0) \tag{21a}$$

$\mathbf{S}$ is the analogous vector for a distribution of maneuvers:

$$\mathbf{S}_i = 1 - \mathbf{E}_i(k, T_0)/\mathbf{E}_i(0, 0) \tag{21b}$$

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

Finally, we measure the nth percentile of energy saved:

$$C(\mathbf{S}) = \bar{\mathbf{S}} + z(n)\sigma(\mathbf{S}) \tag{22}$$

For example, 10th percentile of energy saved corresponds to $z(0.10) = -1.28$.

The average energy saved stays nearly constant as variance increases because the distributed case varies evenly about the nominal maneuver by construction. A smaller $\sigma(\mathbf{E})$ corresponds to a narrower energy distribution—allowing for the energy consumed when executing a maneuver to be predicted more accurately. An alternative measure of variability, $C(\mathbf{S})$, is the energy saved in the nth percentile (22). In other words, as per this example, we can guarantee with 90% confidence that implementing energy minimizing springs on Joints 1, 2, and 3 executing the random submaneuvers distribution will use at least 63.7%, 60.1%, and 11.7% less energy than a springless design, respectively.

3.4. Robust optimization objective function

A robust PAD design would perform more consistently despite maneuver variability compared to a non-robust design. The traditional robust design objective looks to simultaneously minimize both an average objective and the variance in that objective.[27,28] For our purposes, we seek to minimize both average energy and variance in energy:

$$f(k, T_0) = (1 - w) \times \bar{\mathbf{E}} + w \times \sigma^2(\mathbf{E}) \tag{23}$$

where w is the weighting of the two objectives. If $w = 0$, then the goal is only to minimize the average energy, whereas $w = 1$ corresponds to minimizing variance in energy only. Although minimizing variance is good by making system performance more predictable, it is often at the expense of the average, resulting in a system that is predictably mediocre (or even bad).

When executing maneuvers that differ from the nominal design case, a PAD can save more energy on some maneuvers and less on others. If a maneuver being performed is much less demanding than the maneuver for which the PAD was designed, implementation of the PAD can actually decrease the performance of the system. Saving more than the average energy is fine, but we want to avoid situations where the PAD performs poorly or is a detriment to the system. This is similar to finding the nth percentile robust shortest path when estimating travel times.[29] Thus, we formulate the following robust objective function that balances minimizing mean energy consumption and maximizing the energy we are guaranteed to save with 90% confidence (a higher confidence level would result in a more robust system):

$$f(k, T_0) = (1 - w) \times \bar{\mathbf{E}} - w \times C(\mathbf{S}) \tag{24}$$

Table IV shows how increasing the weight, w, of the robust objective function shifts the emphasis from minimizing average energy consumption to maximizing the guaranteed energy savings at a 90% confidence level. The set of optimal designs form a Pareto curve. For Joint 3, a PAD can be designed to be significantly more robust against variations in the maneuver while only slightly sacrificing average energy savings. In Joints 1 and 2, the energy minimizing design is, conveniently, almost exactly the same as the most robust design.

Figure 9 illustrates why we chose an alternative robust design objective function (Eq. 24) instead of the standard one (Eq. 23). If we had used Eq. (23) for our robust design, we would have generated a Pareto curve that begins at the minimum point of the left plot (average energy) and goes to the minimum point on the right plot (standard deviation of energy). Minimizing average energy is a useful objective. However, minimizing standard deviation corresponds to an average energy savings of a mere 3.2% for Joint 3. Forcing the design in this direction effectively pulls the energy savings down to a low uniform level. By contrast, our robust design attempts to pull the performance of the poor performing maneuvers up to the average level.

4. Concluding Remarks

In this paper, we proposed a general methodology to increase the performance of active systems by augmenting them with optimized PADs. Given the ever-growing number of automated machines

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

Table IV. The performance of four parallel PAD designs: an energy-minimizing design (3) and three robust designs of various weights (24) based on the Random submaneuvers distribution. $\bar{\mathbf{E}}$ (20) and $\bar{\mathbf{S}}$ (21) are the average total energy consumed and the average percent saved relative to a no spring design, respectively. $C(\mathbf{S})$ is the minimum energy savings in 90% of maneuvers (Eq. 22).

	Parallel PAD design			Random submaneuvers		
Joint	Description	k [N-m/rad]	T_0 [N-m]	$\bar{\mathbf{E}}$ [J]	$\bar{\mathbf{S}}$[%]	$C(\mathbf{S})$ [%]
1	Min. energy, $w = 0$	49.6	−104	218.7	78.9	63.7
	Robust, $w = 1/3$	49.3	−103	218.7	78.9	63.7
	Robust, $w = 2/3$	48.4	−101	219.8	78.8	63.7
	Robust, $w = 1$	43.7	−89	248.6	76.0	63.8
2	Min. energy, $w = 0$	1.1	−16.5	204.8	65.3	60.1
	Robust, $w = 1/3$	1.3	−15.9	204.9	65.2	60.8
	Robust, $w = 2/3$	1.6	−15.3	205.1	65.2	61.2
	Robust, $w = 1$	1.6	−15.2	206.0	65.1	61.2
3	Min. energy, $w = 0$	2.8	0.5	101.5	38.4	11.7
	Robust, $w = 1/3$	3.9	−3.0	104.0	36.8	22.2
	Robust, $w = 2/3$	4.6	−4.8	105.8	35.7	24.0
	Robust, $w = 1$	4.8	−5.5	106.6	35.3	25.1

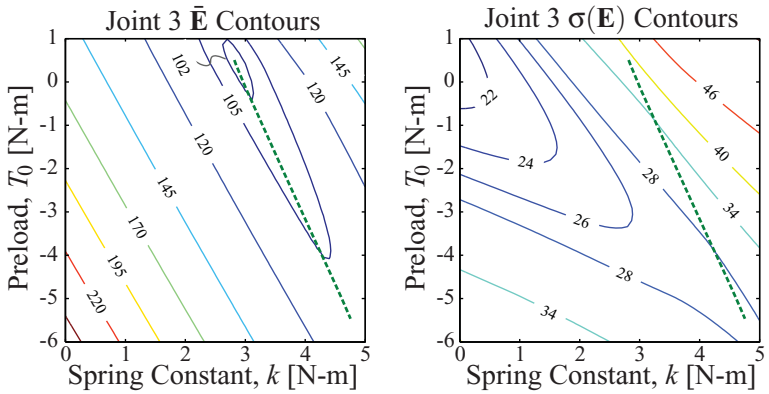

Fig. 9. Required average energy (left) and standard deviation in energy (right) contours as functions of the two spring variables: k, T_0. The set of optimal robust designs form a Pareto curve (dashed green) with weights ranging from 0 (minimize $\bar{\mathbf{E}}$) to 1 (maximize $C(\mathbf{S})$).

(robots, machining systems, material handling, etc.) the proposed approach can have a major impact by increasing energy efficiency. Some machines execute maneuvers that vary so much that the addition of a PAD designed by this approach will have negligible impact on performance. However, while many automated systems perform a variety of tasks, such tasks are often similar, that is, the range of trajectories and loads is limited. Therefore, PADs optimized for a particular maneuver can be effectively deployed on these systems so long as the particular maneuver is typical of the trajectories and loads of the augmented system.[2] Furthermore, we have presented a robust PAD design approach for systems where the maneuver can be described by a known distribution.[18]

In Section 2, we extended the maneuver-based, passive assist design approach from parallel PADs to series and dual systems. We illustrated how the design processes for the series and parallel systems were analogous using simple mass–spring–actuator systems. To make the optimization converge more quickly, we introduced a new initial design using a weighted force displacement curve fit FDCF+ for a parallel system or a weighted derivative of force velocity curve fit dFVCF+ for a series system. We provide engineering insight into why different types of PADs perform differently depending on the maneuver and offer guidelines on how to select a specific type based on the application. Specifically, parallel designs in general have greater potential when external forces dominate, whereas series

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

springs are often superior for maneuvers with large inertial forces. We also discuss how the addition of different PAD types can affect a system from a non-energy standpoint.

In Section 3, we showed that our design approach can effectively be applied to more complicated machines performing known periodic motions by simulating a 3-link UGV manipulator arm executing a representative sample of typical missions. The torques and energy required of the motor at each joint can be decoupled from the rest of the manipulator arm prior to optimization. This simplifies the problem from an n-DOF problem to n 1-DOF problems. We then found that by adding energy minimizing PADs (torsion springs) we achieved joint energy savings of 39–79% and a total system energy savings of 70.9%. This energy savings is substantial; a UGV that may have only had enough battery energy to perform one arm maneuver could now perform three, greatly increasing utility. Most importantly, we showed that PADs can be designed to be robust against variations in the maneuver—substantially increasing the guaranteed energy savings at a 90% confidence level.

Acknowledgment

This research was supported in part by the Ground Robotics Reliability Center at the University of Michigan, with funding from government contract DoD-DoA W56H2V-04-2-0001 through the US Army Tank Automotive Research, Development, and Engineering Center.

References

1. C. Bryan, M. Grenwalt and A. Stienecker, "Energy consumption reduction in industrial robots," *Proc. ASEE North Central Sectional Conference* (2010).
2. W. R. Brown and A. G. Ulsoy, "A Maneuver Based Design of a Passive-Assist Device for Augmenting Active Joints," *J. Mech. Robot.* **5**, 031003 (2013).
3. T. Yang, E. Westervelt, J. Schmiedeler and R. Bockbrader, "Design and control of a planar bipedal robot ERNIE with parallel knee compliance," *Auton. Robots* **25**, 317–330 (2008).
4. J. Herder, Energy-Free Systems: Theory, Conception, and Design of Statically Balanced Spring Mechanisms *PhD Thesis* (Delft University of Technology, Delft, Netherlands, 2001).
5. S. C. Li, J. X. Qiu and J. Y. Zhu, "CIRP," *Ann. Manuf. Technol.* **39**, 455–458 (1990).
6. H. A. Akeel, Robot with Counterbalance Mechanism having Multiple Attachment Locations. U. S. Patent 4653975 (1987).
7. R. H. Gorman and R. N. Aggarwal, Industrial robot having counterbalanced arms. U. S. Patent 4768918 (1988).
8. S. Krut, M. Benoit, E. Dombre and F. Pierrot, "MoonWalker, a lower limb exoskeleton able to sustain bodyweight using a passive force balancer," *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2215–2220 (2010).
9. Q. Lu, J. McAvoy and O. Ma, "A simulation study of a reduced-gravity simulator for simulating human jumping and walking in a reduced-gravity environment," *Proc. ASME Dynamic Systems and Control Conference (DSCC)*, 1675–1682 (2009).
10. U. Mettin, P. X. La Hera, L. B. Freidovich and A. S. Shiriaev, "Parallel Elastic Actuators as a Control Tool for Preplanned Trajectories of Underactuated Mechanical Systems," *Int. J. Robot. Res.* **29**, 1186–1198 (2010).
11. W. R. Brown and A. G. Ulsoy, "Maneuver based design of passive-assist devices: A comparison of parallel and serial systems," *Proc. ASME Dynamic Systems and Control Conference (DSCC)*, (2013).
12. F. C. Anderson and M. G. Pandy, "Storage and utilization of elastic strain energy during jumping," *J. Biomech.* **26**, 1413–1427 (1993).
13. S. Wolf *et al.*, "Soft Robotics with Variable Stiffness Actuators: Tough Robots for Soft Human Robot Interaction," **In**: *Soft Robotics* (Springer, Berlin Heidelberg, 2015) pp. 231–254.
14. M. Hutter, C. Remy, M. Hoepflinger and R. Siegwart, "High compliant series elastic actuation for the robotic leg ScarlETH," *Proc. International Conference on Climbing and Walking Robots (CLAWAR)*, (2011).
15. K. Harper, M. Berkemeier and S. Grace, "Decreasing the energy costs of swimming robots through passive elastic elements," *Proc. IEEE International Conference on Robotics and Automation (ICRA)* **3**, 1839–1844 (1997).
16. G. Pratt and M. Williamson, "Series elastic actuators," *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems.* **1**, 399–406 (1995).
17. Q. Zeng, K. Ehmann and J. Cao, "Tri-pyramid Robot: Stiffness modeling of a 3-DOF translational parallel manipulator," *Robotica* **72**, (2014).
18. W. R. Brown and A. G. Ulsoy, "Robust maneuver based design of a passive-assist device for augmenting active robotic joints," *Proc. ASME Dynamic Systems and Control Conference (DSCC)*, (2013).
19. D. Yoon and C. E. Okwudire, "CIRP," *Ann. Manuf. Technol.* **64**, 381–384 (2015).

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press

20. G. Hao and H. Li, "Extended Static Modeling and Analysis of Compliant Compound Parallelogram Mechanisms Considering the Initial Internal Axial Force," *J. Mech. Robot.* **8**, 041008 (2016).
21. T. A. McMahon *Muscles, Reflexes, and Locomotion* (Princeton University Press, 41 William Street, Princeton, New Jersey, 1984).
22. C. S. G. Lee and Chang, P. R. "Efficient Parallel Algorithm for Robot Inverse Dynamics Computation," *IEEE Trans. Syst. Man Cybern.* **16**, 532–542 (1986).
23. G. Rodriguez, "Kalman filtering, smoothing, and recursive robot arm forward and inverse dynamics," *IEEE J. Robot. Autom.* **3**, 624–639 (1987).
24. P. I. Corke, "A Robotics Toolbox for MATLAB," *IEEE Robot. Autom. Mag.* **3**, 24–32 (1996).
25. T. J. Yeh and F. K. Wu, "Modeling and robust control of worm-gear driven systems," *Simul. Modelling Pract. Theory* **17**, 767–777 (2009).
26. Faulhaber, DC-Micromotors, Graphite Commutation, 70 mNm, Series 3257 . . . CR. Dr. Fritz Faulhaber GMBH & Co. KG.
27. M. Phadke, *Quality Engineering Using Robust Design* (Prentice Hall, Englewood Cliffs, NJ, 1989).
28. I. Doltsinis and Z. Kang, "Robust design of structures using optimization methods," *Comput. Methods Appl. Mech. Eng.* **193**, 2221–2237 (2004).
29. T. Xing and X. Zhou, "Reformulation and Solution Algorithms for Absolute and Percentile Robust Shortest Path Problems," *IEEE Trans. Intell. Transp. Syst.* **99**, 1–12 (2013).

https://doi.org/10.1017/S0263574716000850 Published online by Cambridge University Press