TOWARDS SHARED CONTROL OF HYDRAULIC EXCAVATORS

Aaron ENES[1] and Wayne BOOK[2]
Georgia Institute of Technology
Woodruff School of Mechanical Engineering
801 Ferst Drive
Atlanta, GA 30332-0405
USA
Email: aaron.enes@gatech.edu

Presently, excavator operators manually control all functions of the machine. Decreases to a task's time or energy cost are historically achieved by designing better systems and components or through operator training. In contrast, the approach discussed here allows the human and machine to share control of the end effector to improve machine performance. The paper will introduce the idea of blended shared control in the context of excavation, and will show methods of describing the operator task. A filter that increases the robustness of the task identification is presented. Examples using experimental data are presented to illustrate the results. This approach can potentially decrease task completion time without requiring modification of the machine hardware.

Keywords: Shared control, Minimum-time trajectories, Excavation

Nomenclature

u, u^*, $\overline{u}$	Actual, optimal, and operator-commanded inputs	ψ_i	Volumetric efficiency of function i
q	Generalized coordinate for the actuators	σ_k	The standard deviation of x_k
x_k	Relative displacement of the actuators during the k^{th} occurrence of a motion primitive	Ω_j	category of the j^{th} primitive in a sequence
$\hat{x}_k$	Expected value of x_k	K_{fg}	flow conductance between f and g
Q_i	Net flow entering function i from the system pump	A_a, A_b	Head and rod side area, respectively
C	Maximum flow rate of the system pump	D_i	Max flow rate through function i.

1 INTRODUCTION

Increasing the productivity and energy efficiency of earthmoving machines has garnered the attention of industry and academia researchers. Some researchers focus on components; others on new system architectures like displacement controlled actuators. This paper presents an approach to decrease the cycle time of hydraulically actuated excavators; this approach does not modify the machine hardware but rather modifies the electronic control input to a machine.

Several authors have proposed schemes to automate all (Stentz, Bares et al. 1999) or portions (Rocke 1995; Haga, Hiroshi et al. 2001) of a given excavation cycle. Other researchers (Kontz and Book 2007) showed that coordinated control may improve task efficiency while leaving the

[1] Ph.D. Candidate
[2] HUSCO/Ramirez Distinguished Professor of Fluid Power and Motion Control

operator in full manual control of the machine. These methods require extra components including position sensors, workspace situational awareness, and well defined working environments and tasks. The necessity of extra hardware components precludes wide-scale industry adoption of these otherwise beneficial technologies.

Some excavators, specifically compact excavators and backhoes, utilize a single pump to supply pressurized oil to the working functions. Often the pump is undersized so it impossible to simultaneously actuate all functions at full speed. This constraint on net consumed flow manifests itself in sometimes unintuitive multi-function interactions while digging. E.g., identical joystick displacements may result in different speeds of the boom depending on the *direction* the arm is moving. This coupling may result in larger task completion times than necessary.

Our paper presents a discussion on a promising method termed *blended shared control* (SC) to decrease cycle time of multi degree of freedom (DOF), flow-constrained hydraulic manipulators. We previously showed that blended SC decreased task completion time of a simple single DOF system (Enes and Book 2010). The focus here is on extending this to a multi-DOF system such as an excavator.

For SC, the electronic machine controller has authority to *modify* the original operator commands $\bar{u}$, as shown in Fig. 1a. The command perturbation is applied with consideration of both the intended operator task (defined by the expected displacement x_k discussed in Section 2) and the optimal input u^* for that particular task. Our core hypothesis is that the task completion time can be reduced with these control perturbations.

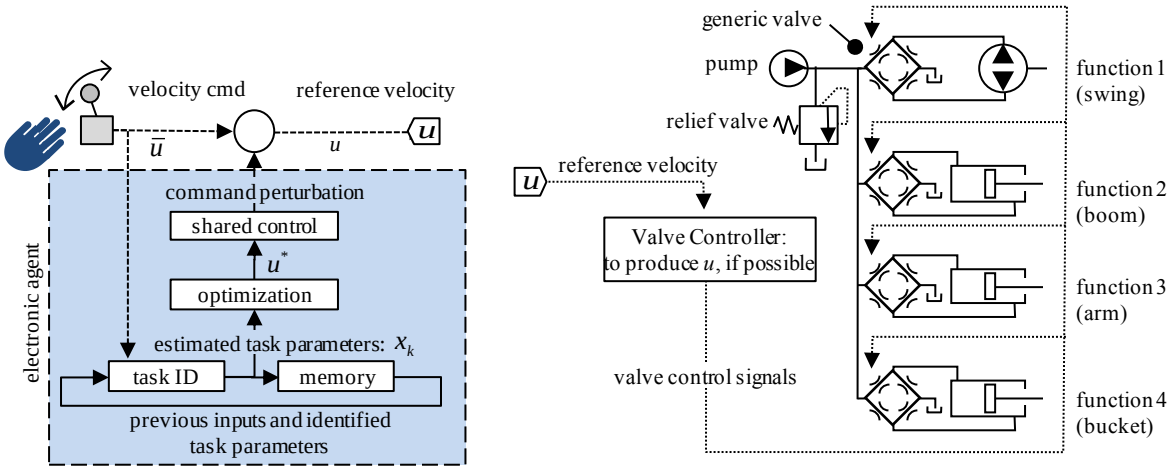

Fig. 1 (a): The shared control architecture. **(b):** simplified hydraulic circuit with a single-source.

The discussion is organized as follows: First we describe how to describe an excavation task as a sequence of motion primitives; second, we demonstrate how the parameters of a motion primitive can be learned online; third, we optimize an example trajectory and compare it to what a human operator achieves.

2 EXCAVATION TASKS

An excavator operator uses joysticks to command the slew rate and the boom, arm, and bucket cylinder speeds so that the end effector (bucket) follows a desired path through the workspace.

Typical tasks include trenching, bench digging, and craning; the tasks are often repetitive cycles. We assume a set of *waypoints* in space adequately describe the motion. The end effector sequentially passes through each of the waypoints to complete the intended task. Waypoints may represent regions in space (e.g., the bucket position before unloading) or obstacles (e.g., the trench boundaries). The parameters defining the waypoints may vary slowly or abruptly between cycles.

To illustrate, consider a contrived task that emulates the motion that the swing and boom functions undergo while trenching. The operator taps the bucket on two targets representing the waypoints (shown in the cartoon in Fig. 2): one placed in a trench of depth 2.5 *m* and one placed on a soil pile of height 2 *m*. The targets are separated by 45 degrees of cab rotation and are placed about 5 *m* radially from the base. The arm and bucket cylinders are fixed at 2.66 *m* and 2.68 *m* respectively. Only the boom and swing are actuated. An expert operator performs the task with a 20-ton excavator, resulting in swing angle and boom cylinder length trajectories shown in Fig. 3.

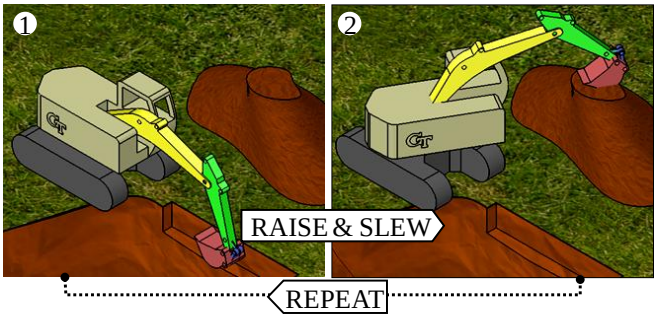

Fig. 2: The excavator pose while tapping the targets in the trench (1) and on the spoil pile (2).

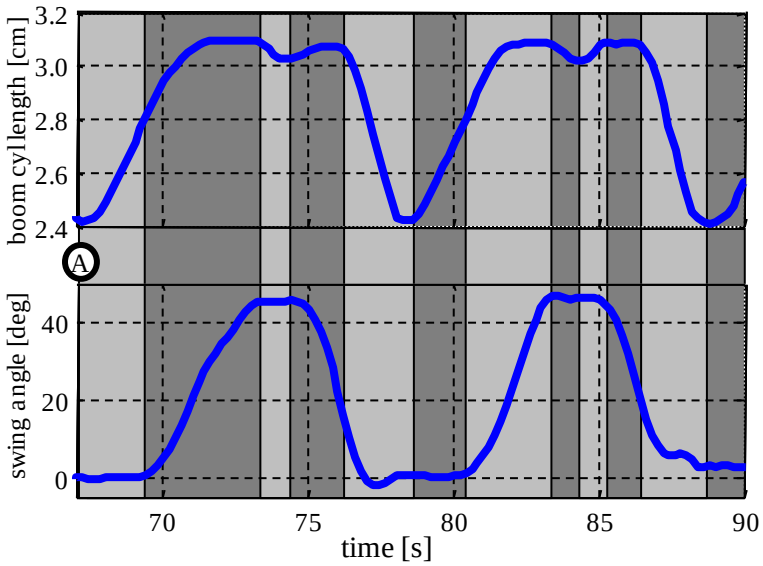

Fig. 3: Length of boom cylinder and angle of cab rotation for one cycle during the two-waypoint task. Shaded regions designate a divisions of piecewise monotonic segments.

2.1 Using Motion Primitives to Describe a Task

Any path of the end effector through the workspace can be expressed as a sequence of piecewise monotonic functions of the generalized coordinates $q = [q_{SW}, q_{BM}]^T$, with q_{sw} and q_{BM} representing the swing angle and boom cylinder length. The alternating shaded regions in Fig. 3 show one possible grouping into piecewise monotonic segments. Each of the highlighted regions is

termed a *motion primitive*. All possible motion primitives belong to one of eight categories (given numerical labels $\Omega = 1, 2, \cdots$) according to the function being actuated and its direction, as in Fig. 4. The motion primitive is the fundamental unit to describe the waypoints which define the bucket path. A motion primitive is denoted by $P^{(\Omega)}(x)$ and indicates two things: the magnitude of the change, *x*, undergone by each function; and the direction Ω in which the change occurs. The *relative* displacement x of the actuated functions over the course of the primitive are $x = q^i - q^f = \left[q^i_{SW} - q^f_{SW},\quad q^i_{BM} - q^f_{BM}\right]^T$.

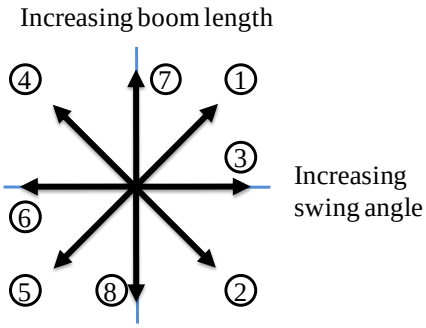

Fig. 4: There are eight categories—or classes—of motion primitives for the case of two functions

with q^i_{SW} and q^f_{SW} denoting the absolute position of the swing function at the start and end of the primitive. These data can be measured directly or estimated based on sensed velocity input and simple machine models. To illustrate, the task in Fig. 3, starting from point (A), is described as the sequence

$$\left[P_1^{(7)}(0.4, 0\,),\ P_2^{(1)}(0.3, 45), P_3^{(8)}(0.1, 0),\ P_4^{(7)}(0.1, 0), P_5^{(6)}(0, 30), P_6^{(5)}(0.7, 15)\right] \qquad (1)$$

Using the motion primitive construct has several practical benefits. First, the displacements of a primitive are measured relative to the prior one. Thus, lower error due to compounding modeling errors can be expected compared to models that require absolute position estimates. Further, tasks can be defined with a reduced parameter set. The onerous undertaking of describing (and then optimizing) a complete digging cycle is replaced by a simpler job that is only concerned with the endpoints of the piecewise monotonic displacements which define the dig cycle.

A disadvantage is that since only the *endpoints* of the motion are constrained, there are no restrictions—apart from monotonicity—which exist on the actual trajectory *within* the primitive. Consequently, an identical sequence of motion primitives describes two quite different function paths. E.g., the paths shown in Fig. 5 are each described by the primitive in (1).

3 IDENTIFYING MOTION PRIMITIVES IN PRACTICE

Fig. 6 shows the boom and swing trajectory and the classification of the primitives for the two-function task performed by an expert operator. An initial classification round (Fig. 6c) based solely on instantaneous velocity inputs exhibits considerable noise. Noise may be introduced if the operator inadvertently overshoots and compensates by reversing course. This "glitch" may disrupt the estimation of the primitive parameters. One approach to filter unintended motions is the backward merge algorithm.

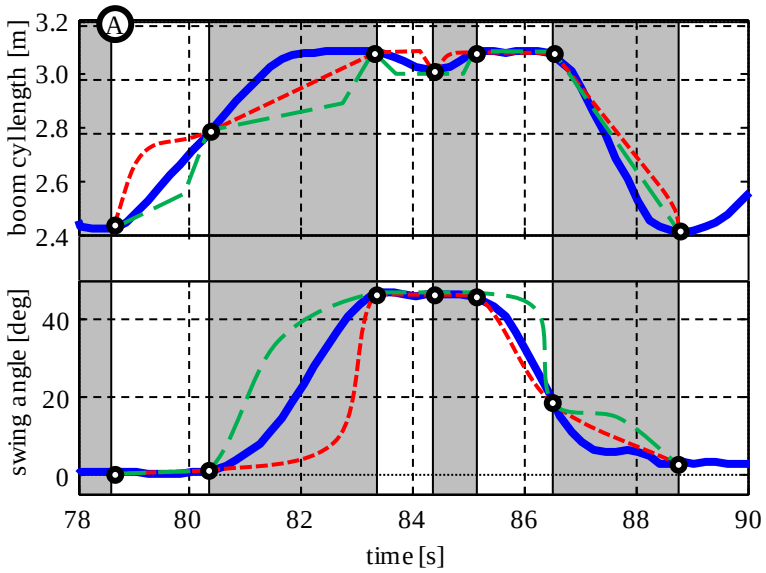

Fig. 5: The actual motion (solid line) is described by the primitives of (1). Two quite different paths (dashed lines) are also described by the same sequence of primitives.

3.1 Backward merge algorithm

The backward merge algorithm described next combines two primitives from different categories when only small (and, presumably, unintentional) relative displacements in cylinder position occur between them. Let $P_j^{(\Omega_j)}(x_j)$ be the primitive for the present motion. If $|x_{j-1}|_\infty < K$ (where K is a tuned constant), then $x_{j-2} = x_{j-2} + x_{j-1}$ and $\Omega_{j-1} = \Omega_{j-2}$. Fig. 7 illustrates this backward merge algorithm for an arbitrary process, showing the merging of two prior primitives if the norm of the displacement is small. Fig. 6d is the resulting backward-merged filtered classification of the motion. Observe that the short-duration classifications around 172 to 174 *s* are filtered.

3.2 Estimating the parameters, an example case with actual excavator data

Recall that the goal is to estimate the displacement, x_k, of the k^{th} occurrence of the present primitive Ω_j—given the displacement $x_{k-1}, x_{k-2}, \cdots$, of the previous times this primitive was encountered—so that the motion can be optimized. In the previous section we detected the present primitive and extracted the displacement associated with it. These data are stored for each primitive. By processing this past data, a good estimate of the expected duration can be achieved.

The proposed filter is a recursive first-order resettable filter (Fig. 8). We consider the process to be stationary; although, in reality the parameters describing the excavation process are time-varying (e.g., the trench deepens after each cycle). We use a recursive exponential filter to "forget" old data, making it amenable to reasonably estimating the mean of slowly-varying processes. The filter is made sensitive to sharp process aberrations by a reset condition that triggers if a sample exceeds a number of standard deviations from the mean of previous samples. The standard deviation is estimated recursively.

The two-point, two-function task described in Fig. 2 was performed twice on an actual machine. In the first configuration, one target was located in the trench and the other atop a soil pile of height 2.0 *m*. In the second configuration the height of the soil pile was decreased to about 1.8 *m* to simulate an abrupt change in waypoint location. Fig. 9 shows the performance of the resetting recursive exponential filter compared to more simple averaging filters.

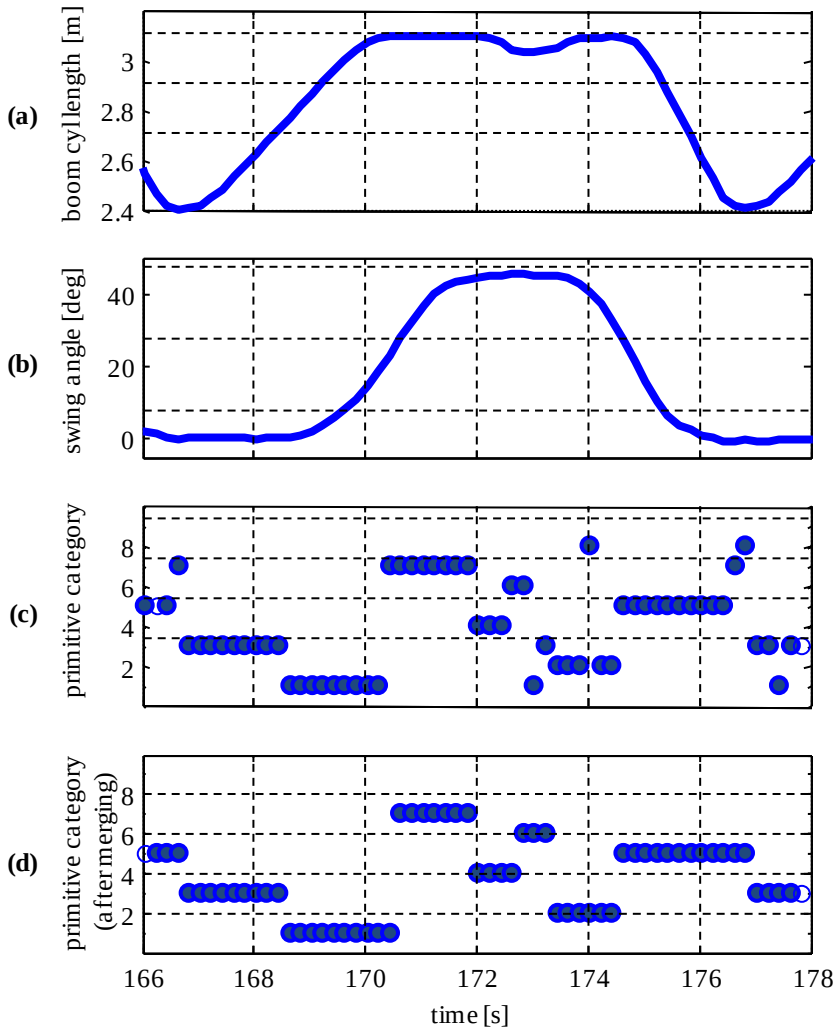

Fig. 6 (a): Boom cylinder length. **(b):** Swing angle. **(c):** Initial classification of motion primitives. **(d):** Classifications after *backward merge* operation. The two-point, two-function cycle was performed by an expert operator on a 20-ton excavator.

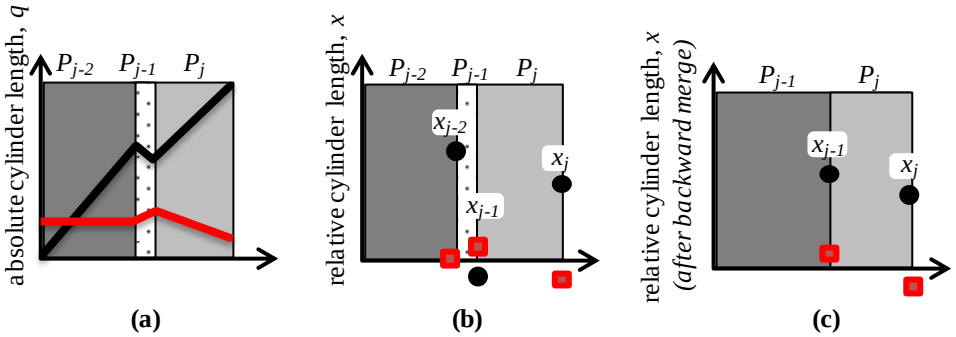

Fig. 7: Illustration of backward merge process to clean-up unintentional motion primitive classifications. **(a):** The current motion is a P_j primitive. **(b):** The previous primitive P_{j-1} produced small changes x_{j-1} relative to its prior primitive. **(c):** The category of P_{j-1} is changed to P_{j-2}, and the length of this primitive is shifted by x_{j-1}.

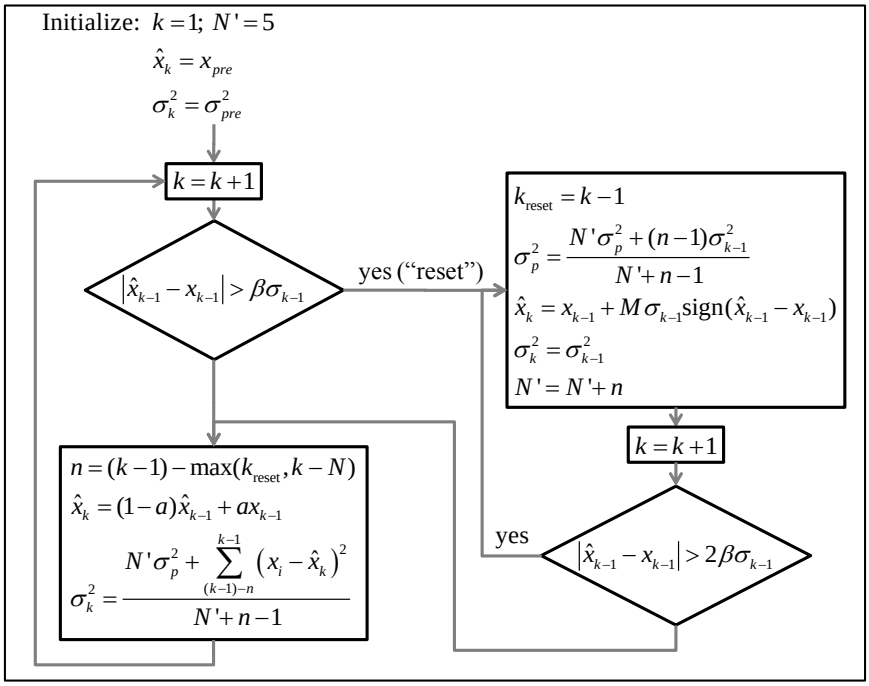

Fig. 8: Process diagram for the resetting recursive exponential filter. The objective is to estimate the duration x of the kth occurrence of primitive category Ω. The $\hat{x}_k$ are understood to be unique for *each* primitive class, Ω. The filter parameters used are: $a = 0.2, \beta = 3, M = 1$.

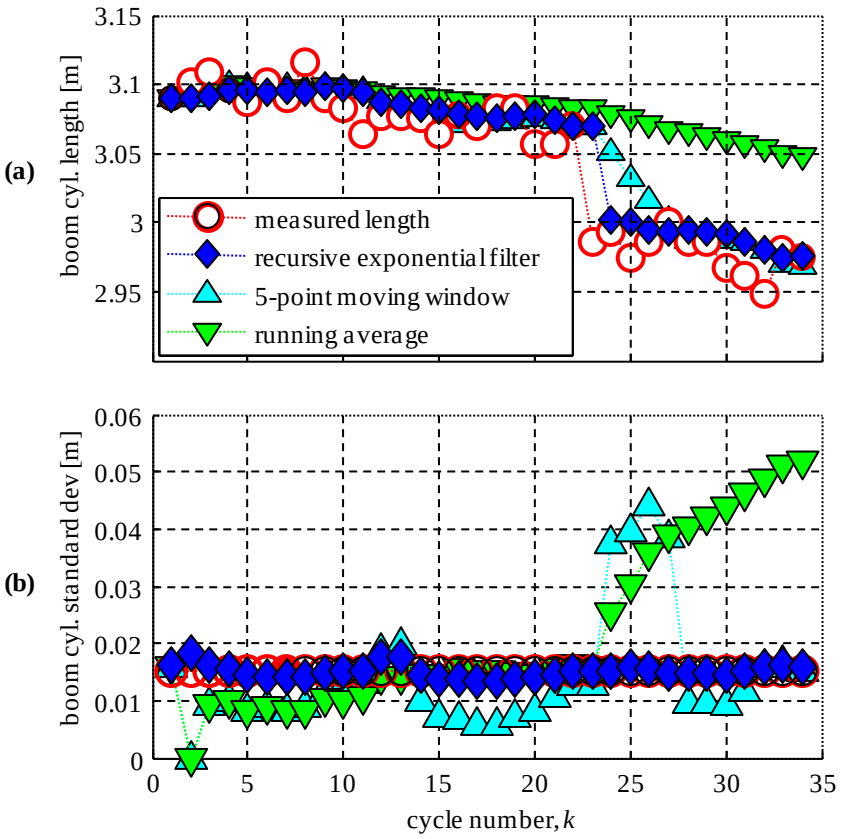

Fig. 9 (a): Mean and **(b):** Standard deviation of the resetting recursive exponential filter, compared to measured data and two other filters.

4 OPTIMIZING THE TASK

The identified task parameters are inputs to an optimization algorithm that determines the optimal inputs for the task. Some research focus has been applied to optimizing the digging process (Koivo, Thoma et al. 1996; Budny, Chlosta et al. 2003). These methods rely on accurate models of the system dynamics and soil/bucket interactions. Often the models are very complicated and are untenable for real-time computation. We propose a low-fidelity kinematic model that may be suitable for optimizing the gross motion of the excavator.

4.1 Kinematic Velocity-Constrained Excavator model

A simplified hydraulic circuit to power the machine consists of a single pump connected through two valves to a fixed displacement motor and a cylinder (Fig. 10). The control valves and hydraulic system architecture discussed in the next section.

Excavators have dynamics that occur over very different time scales, ranging from the very fast pressure rise within a closed volume of fluid to the slower pump and rigid-body linkage dynamics (Merrit 1967). Fluid power researchers studying the gross motion of large hydraulic systems often assume the hydraulic components undergo "quasi-static" dynamics. This means the plant dynamics are slowly changing and the capacitance of the cylinder chambers and conduits is neglected. The quasi-static assumption is useful for energy analysis of high-level valve control (Shenouda and Book 2005) and for motion planning.

For the sake of deriving a tenable solution to the optimal control problem, we make the convenient though not entirely accurate assumption that all the hydraulic dynamics can be neglected. Thus, assuming the actuators follow a simple kinematic velocity-controlled law gives

$$\dot{x} = -u$$

with the assumption that $u \in U$, where u is the velocity control input to the cylinders, and U is the set of allowable controls, as defined next. We also assume the function is not stalled and that the pump is not power limited. Hence this model is generally only applicable to free-air motions with moderate loads.

4.2 Calculating Effective Volumetric Efficiency

A typical independent metering control valve is shown in Fig. 10. At steady state, the velocity of function i is $u_i = \frac{Q_a}{A_a} = -\frac{Q_b}{A_b}$.

The flow entering the actuator is related to the valve's operating mode (the particular combination of orifices are open). The valve controller determines the appropriate operating mode based on commanded velocity, direction of motion, system pressure, and actuator workport pressures. The operating mode determines the effective volumetric efficiency (EVE) of the function, which we denote ψ_i. This will be used to relate the velocity u_i of the function to the flow Q_i entering the function as

$$\psi_i \equiv \frac{Q_i}{u_i}$$

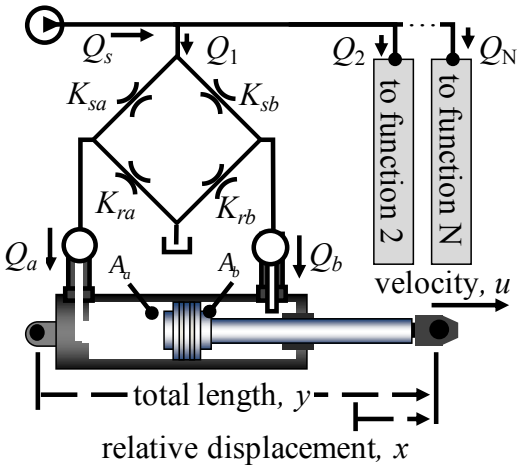

Fig. 10: Hydraulic independent metering valve.

To illustrate computation of the EVE for a typical hydraulic valve, consider two industry standard operating modes for extending a cylinder $\dot{q} > 0$: *standard extend* (SE) and *high-side regeneration extend* (HSRE) (Pfaff and Tabor 2005). In SE mode, orifices K_{sa} and K_{rb} are open, so $\mathrm{Q_i} = \mathrm{Q_a} = \mathrm{A_a u_k}$. Thus $\psi_k^{SE} = A_a$. In HSRE mode, orifices K_{sa} and K_{sb} are open, so $Q_i = Q_a + Q_b$. Thus $\psi_k^{HSRE} = A_a - A_b$.

We omit the EVE calculations for other operating modes, but summarize the results in Table 1. In the absence of cross-functional regeneration modes (which distribute high-pressure discharge flow to other functions rather than draining this flow to tank), $\psi \geq 0$; otherwise ψ may assume negative values. (And if a conduit is disconnected or leaking, ψ may be infinite!)

Table 1: Effective volumetric efficiency for various operating modes of a common valve.

Operating Mode	Orifices open	EVE, ψ
Standard extend (SE)	K_{sa}, K_{rb}	A_a
High-side regen extend (HSRE)	K_{sa}, K_{sb}	$A_a - A_b$
Low-side regen extend (LSRE)	K_{sa}, K_{ra}, K_{rb}	$A_a - A_b$ [1]
Standard retract (SR)	K_{sb}, K_{ra}	A_b
Low-side regen retract (LSRR)	K_{rb}, K_{ra}	0

[1] $\psi < A_a - A_b$ if circuit enables low-pressure flow from other functions to enter through return line (cross-functional regeneration)

The pump delivers a nonnegative volumetric flow rate up to a maximum rate *C*. There also is a maximum rate of flow D_i through each valve. This limit depends on the maximum supply pressure, the valve's maximum flow conductance, and the assumption that no cavitation occurs. Thus the allowable control region is

$$U = \{u: 0 \leq u_i \leq D_i,\ \psi^T u \leq C\}$$

where $u = [u_1, \cdots, u_N]^T$ is the control input, and $\psi = [\psi_1, \cdots, \psi_N]^T$ is a vector of the effective volumetric efficiencies for all functions.

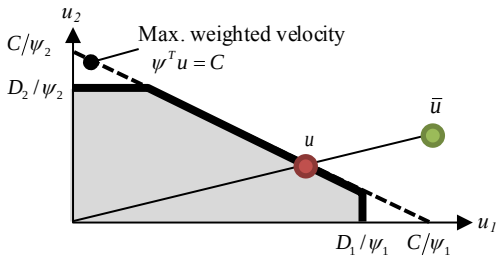

Fig. 11: Feasible function velocities given single- and multi-function flow constraints.

For convenience in notation we assume without loss of generality that all motions take place in the first quadrant of the u_1u_2-input plane. Motion in other quadrants is realized by noting the direction of motion and assigning the corresponding value and sign to ψ_i based on the change of variables in Section 2.1.

If the commanded velocity $\bar{u}_k$ is outside the feasible region then the function velocities must adjust such that the flow constraint is always satisfied. The flow will follow the path of least resistance first. The valves openings may be modified to proportionately reduce the speed of each function until feasible or by providing full preference for a particular function. We assume that an infeasible velocity command $\bar{u} = [\bar{u}_1, \bar{u}_2]^T$ is mapped to the intersection of the ray from the origin to $\bar{u}$ and the boundary of the feasible region *U*, as in Fig. 11.

4.3 Optimizing the Motion

We wish to find the control input required to trace the designated path in minimum time. This path is described by a sequence of motion primitives P_k of length x_k. The optimization problem is formulated such that minimizing the time to complete the sequence of primitives is equivalent to controlling a state $x(t)$ from the initial state $x(0)$ to the origin $x(T) = 0$ in minimum time (Enes and Book 2010). The optimal input can be shown to be

$$u^*(x) = \left[\frac{C/\psi_1}{1+\alpha(x)} \quad \alpha(x)\frac{C/\psi_2}{1+\alpha(x)}\right]^T, \text{ where } \alpha(x) \triangleq \frac{x_2}{x_1}\frac{\psi_2}{\psi_1} \tag{5}$$

4.4 Example: Comparison of Manual Control and Optimal Control

An experiment was performed with an operator using a standard PC gaming joystick to control the swing and boom of a simulated excavator displayed on a monitor. The operator is instructed to control the excavator so that the bucket follows a path designated by a sequence of waypoints, as shown in Fig. 12: the bucket must tap the top of target *A*, move over the obstacle *B*, and touch the ground at point *C*. The bucket must pass straight down through point *C'* on the way to point *C*. The cycle reverses, and repeats.

While the limited number of trials completed so far is not conclusive, they do suggest that there is appreciable opportunity to decrease the task time. The results summarized in Table 4 indicate that mean task time under manual control is 14 percent slower than the optimal time. The boom cylinder position, swing angle, and velocity commands are shown in Fig. 13 for the manual control case and the optimal control case.

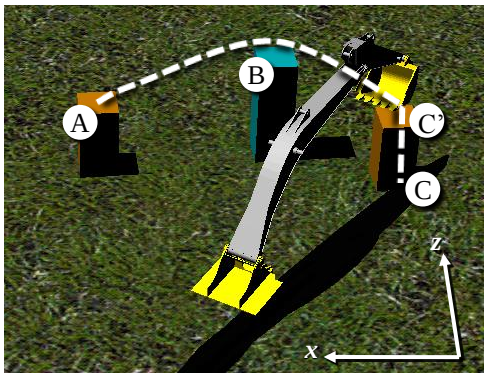

Fig. 12: Waypoints for two-function experiments. Waypoints *A*, *B*, and *C* are at points (2.7, 2.0, 7.2), (-5.0, 2.0, 5.9), and (-2.0, 3, 7.4) respectively.

Table 2: Primitive category and magnitudes for the two-function dig cycle task described above

	Magnitude for $P_j^{(\Omega)}$					
j	1	2	3	4	5	6
Ω	4	5	8	7	1	2
$x_j = \begin{bmatrix} x(SW) \\ x(BM) \end{bmatrix}$	$\begin{bmatrix} 41.7 \\ 16.1 \end{bmatrix}$	$\begin{bmatrix} 22.7 \\ 16.1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 18.9 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 18.9 \end{bmatrix}$	$\begin{bmatrix} 22.7 \\ 16.1 \end{bmatrix}$	$\begin{bmatrix} 41.7 \\ 16.1 \end{bmatrix}$

Table 3: Experiment parameters

Parameter	Value
C, multi-function flow limit	1244 *cc/s*
D, single-function flow limit	[933, 933] *cc/s*
ψ_1^+, boom SE volumetric efficiency	122.7 cm^2
ψ_1^-, boom SR volumetric efficiency	59.1 cm^2
$\psi_2^+ = \psi_2^{-1}$, swing volumetric efficiency	46.1 cm^2

Table 4: Cycle times. Experiment results during control of the virtual excavator, compared to optimal task time. Times in seconds. *N*=22 trials.

	Experiment			
Optimal	*mean*	*max*	*min*	σ, std. dev.
11.97	13.64	15.3	12.07	1.01

5 CONCLUSION

The framework shown in Fig. 1 for using SC to reduce excavation task completion times first requires identification of the operator's intended task. Describing the task as a sequence of motion primitives allows for convenient representation of the task. It was shown that the motion primitive category and expected magnitudes x_k can be identified using data from an actual excavator process. A task, once identified, can be optimized. There is appreciable difference between a typical operator's performance with manual control compared to the optimal solution. Future work should focus on combing the optimal command with the operator's original command to reduce operator's task completion time.

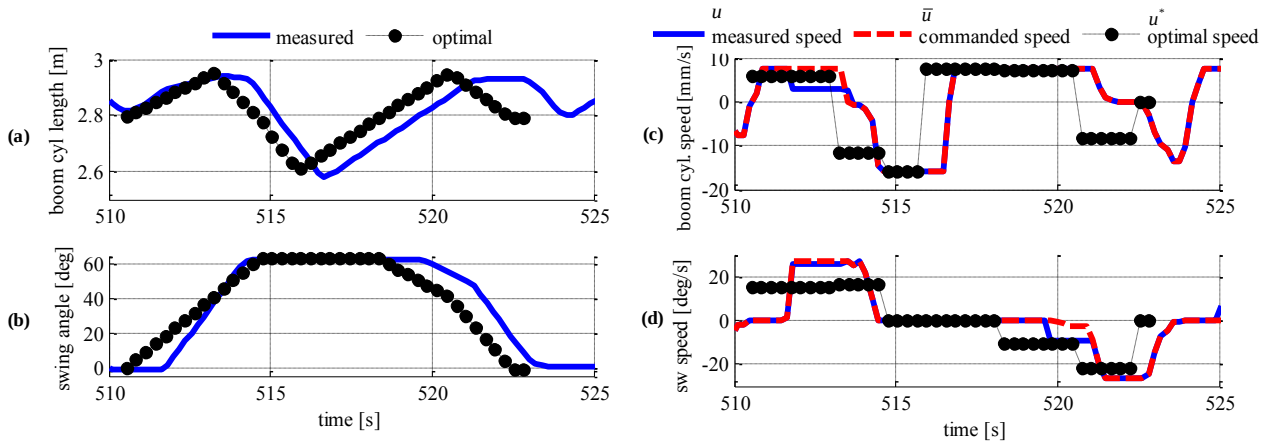

Fig. 13 (a): Solid line: The boom cylinder length and swing angle for a typical cycle under manual control; Circle line: The optimal cylinder length and swing angle. **(b):** Velocity commands for the boom and swing function. Solid line: Actual velocity u under manual control; Dashed line: original velocity $\bar{u}$ under manual control; Circle line: Optimal input u^*.

6 ACKNOWLEDGEMENTS

The experimental data taken on the excavator is courtesy of HUSCO International of Waukesha, WI. This research was funded in part by the National Science Foundation Center for Compact and Efficient Fluid Power (CCEFP), grant number EEC-0540834.

7 REFERENCES

Budny, E., M. Chlosta, et al. (2003). Load-independent control of a hydraulic excavator. *Automation in Construction 12*(3): 245-254.

Enes, A. and W. Book (2010). Optimizing Point to Point Motion of Net Velocity Constrained Manipulators. *CDC2010: IEEE Conference on Decision and Control, Atlanta, GA (submitted)*.

Enes, A. R. and W. J. Book (2010). Blended Shared Control of Zermelo's Navigation Problem. *Proc. of IEEE ACC2010: American Control Conference, Baltimore, MD*.

Haga, M., W. Hiroshi, et al. (2001). Digging control system for hydraulic excavator. *Mechatronics 11*(6): 665-676.

Koivo, A. J., M. Thoma, et al. (1996). Modeling and Control of Excavator Dynamics during Digging Operation. *Journal of Aerospace Engineering 9*(1): 10.

Kontz, M. E. and W. J. Book (2007). Flow control for coordinated motion and haptic feedback. *International Journal of Fluid Power 8*(3): 13-23.

Merrit, H. E. (1967). *Hydraulic Control Systems*. New York, Wiley.

Pfaff, J. L. and K. A. Tabor (2005). Method of Selecting a Hydraulic Metering Mode for a Function of a Velocity Based Control System. 6880332.

Rocke, D. J. (1995). Automatic excavation control system and method. 5446980.

Shenouda, A. and W. J. Book (2005). Selection of Operating Modes of a Multi-Functional Hydraulic Device. *ASME International Mechanical Engineering Congress and Exposition, Orlando, Florida*.

Stentz, A., J. Bares, et al. (1999). A Robotic Excavator for Autonomous Truck Loading. *Autonomous Robots 7*(2): 175-186.