2012 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 7-12, 2012. Vilamoura, Algarve, Portugal

Learning Disturbances in Autonomous Excavation

Guilherme J. Maeda and David C. Rye

Abstract— **Disturbances that arise in material removal by repeated attempts to track the same path have the particular characteristics of non-repetitive magnitudes, but nearly-repetitive or gradual gradient transitions. This paper proposes and validates Iterative Learning Control (ILC) with a PD-type learning function for this class of disturbance as a predictive controller for autonomous excavation. However, parameters of the PD learning function may require different tunings for different excavation conditions, and convergence can be slow when compared to changes in excavation dynamics. In order to improve convergence, a plant inversion learning function is reinterpreted as a disturbance observer in the iteration domain, effectively rendering a disturbance learning controller (DLC). A hydraulic mini-excavator was used to evaluate experimentally the performance of the conventional ILC and the DLC against a robust controller. ILC achieved a desired cut profile with non-monotonic transients and DLC converged faster by learning disturbances directly from command discrepancies.**

I. Introduction

Heavy equipment automation is one of the key enabling technologies of the mining industry as demand increases and mine locations move to remote areas [1]. Autonomous haul trucks [2] and drills [3] have recently become part of the production cycle in some mines, due principally to technological progress in areas related to perception, localisation and path planning. The deployment of autonomous excavators is yet to be seen, however, as in addition to the aforementioned technologies autonomous excavation requires encoding, predicting and counteracting large forces that approximate the machine's capability.

Control methods for autonomous excavation usually focus on single-pass position or force control based on prediction and generation of actions that are feasible under the manipulator constraints. These approaches tend to rely on soil-tool interaction models and their performance is usually limited by the quality of the predictions. Moreover the majority of approaches do not take in consideration that excavation is by its very nature an iterative process. Iterative learning methods can provide low-level controllers with the ability to learn and adapt to varying soil conditions while relaxing the dependence on explicit soil-tool interaction models.

The problem in iterative autonomous excavation is that of maintaining convergence towards a desired cut profile [4]. In this view, not every pass (here also termed an iteration) needs to be feasible with respect to actuator force limits. In fact, only the last pass needs to be feasible so that the final cut profile can be tracked without error. Individual passes with

G.J. Maeda and D.C. Rye are with the Australian Centre for Field Robotics, The University of Sydney 2006 NSW, Australia. Correspondence should be addressed to g.maeda@acfr.usyd.edu.au

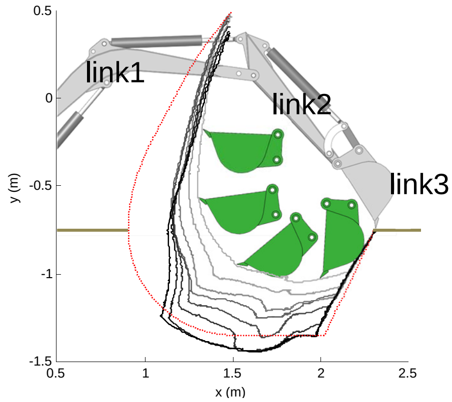


Fig. 1. A sequence of eight excavation passes executed by a conventional computed-torque feedforward controller, iteratively tracking the same dotted path. As soil compacts due to consecutive passes the convergence rate decreases and—in this case—eventually vanishes due to insufficient control authority.

potentially large position deviations caused by actuator force saturation are not considered as failures but as intermediate stages in reaching the goal. The control problem is then to generate and to adapt actuator commands to achieve the final cut profile with a minimum number of iterations.

In an ideal world, high-gain control strategies would be the easiest way to maintain a desired rate of dig convergence. The reality is that the open-loop bandwidth of heavy hydraulic machinery is severely limited by the inherent compliance of the flexible hydraulic lines and deterioration of the effective bulk modulus of the hydraulic fluid through entrapment of air or water in the circuit [5]. High gains can excite the low resonant modes of an hydraulic arm, which can have catastrophic consequences due to the large inertias involved. Fig. 1 shows eight iterative excavation passes with feedback controller gains bounded to avoid exciting the resonant modes of the arm. Note that the convergence decreases since the bounded gains effectively generates a low–stiffness controller, evidencing the need for control improvement.

This paper proposes maintaining convergence and improving the controller actions by learning better feedforward commands between iterations using Iterative Learning Control (ILC) [6]. If the first feedforward iteration is initialised reasonably (e.g. from the free-motion inverse dynamics of the arm), useful removal of material starts immediately without the need for explorative or learning passes. One expected issue is that ILC typically needs several iterations to converge, which can be too slow in comparison to the rapid changes that can occur in soil-tool dynamics.

The remainder of this paper is organised as follows. Sec. II discusses the suitability of ILC for overcoming disturbances that occur during material removal and Sec. III proposes an iterative method for achieving faster convergence. Validation is carried out by experiments with an hydraulic excavator in Sec. IV and the results are discussed in Sec. V.

II. Background

Improving a low-stiffness excavation controller requires adapting actuator commands to compensate for soil-cutting reactions without relying on the low gain feedback loop. Obtaining such adaptive actions through a model-based approach requires predictions from complicated soil-tool interaction models. These models are unavailable except for simple cases that can be approximated using flat blade theory, or are too complex for real-time compensation purposes [7]. Despite comprehensive work analysing and comparing several models available for resistive force prediction [8], [9] there is a lack of consensus and practical validation of such models for control purposes.

Robust and impedance control methods have been applied in excavation providing an alternative to detailed modelling [10], [11]. The assumption of linear mass–spring–damper dynamics as a soil-tool interaction model can be restrictive since there are no guarantees that parameter identification methods can properly fit the linear structure for all possible excavation conditions [8].

A. Disturbances in Excavation

Material removal processes are very often characterised by repetition of similar movements where each pass (iteration) causes partial removal of material, thus decreasing the tracking error at the next iteration. This strategy occurs in tasks as diverse as excavation, CNC machining and scooping ice-cream.

The stability of learning algorithms for processes similar to excavation, that is large disturbances caused by removal of material, has been investigated with 2-D systems theory in [12]. In the refereed work the theoretical stability analysis for multipass processes (motivated by long-wall coal cutting and metal rolling) was presented. The stability issue seems to be related to increasing oscillations from pass-to-pass, as if the damping of the systems decreases iteratively. A common practice in ILC is to low-pass filter the learned signal in order to eliminate those possible exciting inputs [13].

Forces required to cut and drag the material may vary from pass to pass due to the different ways that a material can fail, friction between the tool and medium and within the medium, compaction caused by pushing the tool through the medium and rocks. In spite of these confounding effects, our experimental results show that attempting to track the same path so that excavation occurs over several passes leads to near-repetition of the tool velocity profiles, as disclosed by the servo valve commands shown in Fig. 2. The figure shows the evolution of the disturbances at each of the three servo-valves of an excavator arm for the eight passes shown in Fig. 1, where the closed-loop controller is given only the final desired path.

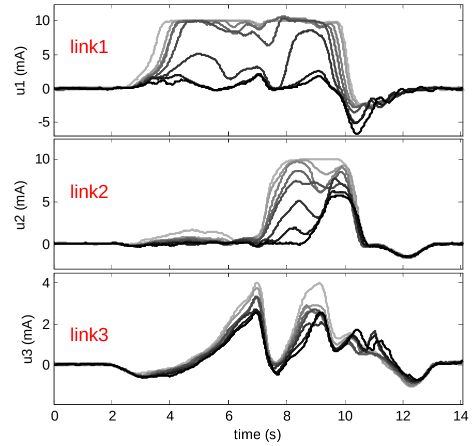


Fig. 2. Disturbances at the three servo-valves of the excavator arm during field experiments. The light gray curve shows the first pass in undisturbed soil. The same controller iterates eight times towards a 60 cm deep cut. From pass to pass, the direction of the disturbance is roughly consistent, offering a structure that can be learned.

B. Iterative Learning Control and Disturbances

Iterative Learning Control takes advantage of the fact that past experience is encoded in the form of tracking error due to disturbances. Disturbances are mapped into control actions by some learning rule that can completely eliminate the need for models. Despite the simplicity and effectiveness of ILC, strict assumptions on repeatability give little hope of disturbance compensation under non-repetitive exogenous signals.

Repeatability has been assumed since the inception of ILC [6] for proving convergence. ILC methods proposed to deal directly with non-repetitive exogenous signals include identifying and learning only the segments that are repeatable [14] and using disturbance observers or selecting the proper compensation signal through categorising the pattern of the disturbance [15].

The following analysis is based on the work in [16], except that here feedforward input is added as actuator commands rather than by adjusting the position reference (Fig. 3). The arm is considered to be composed of three independent SISO systems. The SISO linear treatment holds if one assumes that the controller starts from the first iteration with the compensating feedforward commands obtained from an inverse arm dynamics. The goal is not to learn the arm dynamics but the external disturbance forces.

The errors $\mathbf{e}_k = \mathbf{r} - \mathbf{y}_k$ to a constant reference input $\mathbf{r}$ over two consecutive iterations are related to the outputs by

$$\mathbf{e}_{k+1} = \mathbf{e}_k + \mathbf{y}_k - \mathbf{y}_{k+1} \, , \quad (1)$$

where the bold notation is used to indicate that, for example,

$$\mathbf{y}_k = [y_k(1) \quad y_k(2) \quad y_k(3) \quad \ldots \quad y_k(N)]^T$$

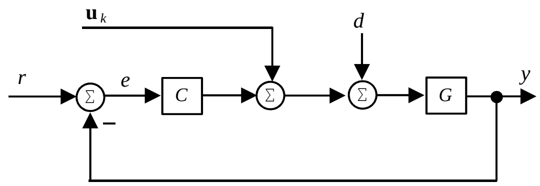


Fig. 3. ILC with feedforward actuator input; C is the feedback gain, G is the plant transfer function, r is the reference input, y is the output, $\mathbf{u}_k$ is the feedforward input updated by ILC and d is the exogenous disturbance.

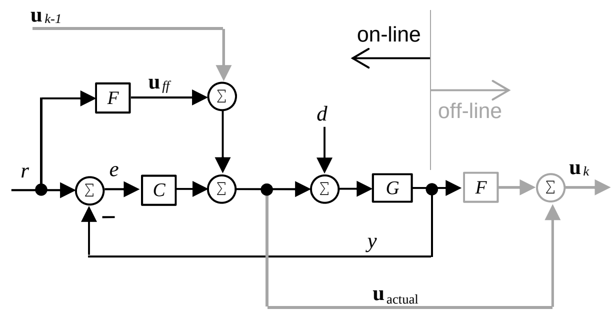


Fig. 4. DLC: an iterative version of a disturbance observer where C is a feedback controller, G is the plant and F is a feedforward compensator. The feedforward command is composed of a free motion inverse dynamics component (when $F = G^{-1}$) plus a component to overcome the disturbance estimated from the previous iteration.

is a vector of N equi-spaced sampled output values at iteration k; this notation is used similarly for other variables.

At iteration $(k+1)$ the controller in Fig. 3 has outputs

$$\mathbf{y}_{k+1} = T\left(\mathbf{d}_{k+1} + \mathbf{u}_{k+1}\right) + T_r \mathbf{r}\,, \tag{2}$$

where

$$T = \frac{G}{1+CG} \tag{3}$$

$$T_r = \frac{CG}{1+CG}. \tag{4}$$

Consider the learning rule

$$\mathbf{u}_{k+1} = \mathbf{u}_k + L\mathbf{e}_k, \tag{5}$$

and substituting (5) into (2)

$$\mathbf{y}_{k+1} = T\mathbf{d}_{k+1} + T\left(\mathbf{u}_k + L\mathbf{e}_k\right) + T_r \mathbf{r}\,, \tag{6}$$

where L is the learning function. Substituting (6) into (1) yields, after some manipulation,

$$\mathbf{e}_{k+1} = (1 - TL)\,\mathbf{e}_k - T\left(\mathbf{d}_{k+1} - \mathbf{d}_k\right)\,. \tag{7}$$

Equation (7) shows that the error at the next iteration is a function of the difference between the current disturbance $\mathbf{d}_k$ and the next disturbance $\mathbf{d}_{k+1}$. Thus, when two consecutive disturbances are identical their contribution to the error is zero. The disturbance fluctuation $\mathbf{d}_{k+1} - \mathbf{d}_k$ dictates the "baseline error" [15] and the error does not vanish if disturbances change between iterations.

An empirical observation during excavation experiments is that if the robot is allowed to repeat passes indefinitely, disturbance reactions stabilise either because the required final profile is achieved or because convergence vanishes. The latter may occur when soil reactions iteratively increases, equalizing the controller stiffness (e.g. due to compaction caused by previous passes). In either case these effects minimize the baseline error caused by the difference $\mathbf{d}_{k+1} - \mathbf{d}_k$

As discussed in subsection II-A, disturbances in material removal have structure insofar as the direction of the disturbances is constant: the tool-soil interaction is dissipative. In ILC, learning excavation reactions from one pass and applying them to the next pass is unlikely to predict the exact compensation required. Since the direction of the correction does not change, however, subsequent actions can be expected to tend towards the correct compensation. A feedforward update for this type of non-repetitive disturbance will lead to over- or under-shoot, depending on whether the disturbance increases or decreases relative to the previous pass. Although the tool-soil interaction is dissipative, monotonic transients can not be expected since disturbances are non-repetitive, in contradistinction to classical ILC.

III. Learning Disturbances from Inverse Models

Inverse models are useful not only to compute expected torques (as in computed-torque control), but also to assess disturbances without requiring a model of the interaction dynamics through examining the difference between ideal and actual commands; this is the essential feature of all disturbance observers (DOB) [17].

The ideal DOB implementation requires the inversion of a plant model whose output is compared with the actual actuator command. Assuming that the plant model is perfect, the cause of the difference is attributed to unmodelled disturbances that act on the plant.

Thus, the immediate difficulty in implementing a disturbance observer is on obtaining an inverse plant model that is sufficiently accurate to allow decoupling of external disturbances from internal dynamics. Lack of accuracy can cause compensation commands from the observer to interfere with the feedforward/feedback actions. In the worst case the plant can be destabilised, for example by generating negative damping or by exciting resonant modes. A second difficulty is that plant inversion requires acceleration inputs which are not adequately recovered by differentiation and causal filtering of position encoder signals, especially at the low range of excavation velocities.

Based on plant inversion for disturbance estimation and with the insight of the semi-structured disturbances found in excavation an ILC with inverse dynamics learning function is presented as a disturbance observer in the iteration domain. The implementation block diagram in Fig. 4 explicitly separates the commands from the inverse dynamics of the arm $\mathbf{u}_{ff}$ from the learned disturbance commands $\mathbf{u}_{k+1}$ in order to make clear that the learning accounts only for external disturbances.

The second inverse dynamics block F calculates, off-line, the free motion commands of the resulting disturbed motion. In the case of modelling error, mismatches between the model and the real plant are compensated iteratively since they are not different from exogenous signals as seeing from the learning perspective.

The DLC update rule is

$$\begin{aligned}\mathbf{u}_{k+1} &= (\mathbf{u}_{actual})_k - F\mathbf{y}_k \\ \mathbf{u}_{k+1} &= C\mathbf{e}_k + F\mathbf{r} + \mathbf{u}_k - F\mathbf{y}_k \\ \mathbf{u}_{k+1} &= (F+C)\mathbf{e}_k + \mathbf{u}_k \,, \end{aligned} \quad (8)$$

and, following a similar derivation as for the ILC, the evolution of the error becomes

$$\begin{aligned}\mathbf{e}_{k+1} &= \left[1 - T\left(F+C\right)\right]\mathbf{e}_k - T\left(\mathbf{d}_{k+1} - \mathbf{d}_k\right) \quad (9) \\ \mathbf{e}_{k+1} &= \left[1 - \frac{G}{1+CG}\left(F+C\right)\right]\mathbf{e}_k - T\left(\mathbf{d}_{k+1} - \mathbf{d}_k\right) \,. \end{aligned}$$

If the feedforward compensator F is set to be the plant inverse dynamics G^{-1}, the error is simply

$$\mathbf{e}_{k+1} = -T\left(\mathbf{d}_{k+1} - \mathbf{d}_k\right) \,. \quad (10)$$

Note that the feedforward term is the sum of torques to drive the arm free dynamics (the first F block in Fig. 4) and to counteract terrain reactions (estimated from (8)).

While conventional PD-type learning rules make the ILC update dependent on the tracking error, the use of the inverse dynamics makes the learning rule dependent on commands only. This difference has an important consequence on the evolution of the error. By comparing the conventional ILC error (7) with the DLC error (10) it is clear that DLC error is independent of the previous tracking error, while ILC needs extra steps to decrease the residual error that is propagated to future iterations.

In summary, assuming that disturbances are repetitive and are independent of the trajectory, the DLC update theoretically achieves convergence in one step. The price to pay is that an inverse model must be available. In relation to an on-line disturbance observer, DLC can be seen as a compromise between implementing a stable inverse dynamics model off-line and being able to accept partial correction in the feedforward action.

IV. Experiments

A. Platform

The experimental platform is a 1.5 tonne Komatsu PC05-7 mini-excavator. The arm links and cylinders weigh a total of 110 kg and the arm reaches 3 m from the boom base. The hydraulic cylinders are flow controlled by servo-valves. All cylinders are supplied from the same accumulator, which is charged to 70 bar by a hydraulic pump driven by a diesel engine. Command signals sent to the servo-valves are spool position references; these are controlled by high-bandwidth analog feedback loops internal to the servo-valves. More details on the platform can be found in [11]; issues related to hydraulic compliance and friction are described in [18].

B. Procedure

Quantitative validation and comparison of controllers in excavation is difficult to achieve due to the high variability of the soil conditions. Variations in moisture content, compaction and inclusions such as rocks and vegetation roots can cause significant disturbances.

In order to compare controllers under deterministic and reproducible experimental conditions a method similar to vehicle suspension track replay was used. The initial experiments were carried out in the field under different excavation conditions while recording controller commands and excavator responses. The disturbances at each pass were then estimated as

$$(\mathbf{u}_{dist}(t))_k = (\mathbf{u}_{actual}(t))_k - F\mathbf{y}_k(t) \,, \quad (11)$$

where F is the inverse arm dynamics. The first term on the right side of the equation is the sequence of recorded control inputs during iteration k and the second term represents what should be the free motion commands of the arm for the recorded motion $\mathbf{y}_k(t)$. The estimated disturbances $(\mathbf{u}_{dist}(t))_k$ are vectors at 100 Hz. Fig. 5a) shows the sequence of extracted disturbances for four different cuts; Fig. 5b) shows one of those cuts.

The values $(\mathbf{u}_{dist}(t))_k$ obtained from (11) are repeated in the same order of the field experiments for each cut: that is, $k = 1, 2 ... M$ where M is the last pass of each cut. Those values contaminate the controller signal as an additional input to the servos (input d in Figs. 3 and 4). The controllers under evaluation do not have direct access to the disturbance signal $(\mathbf{u}_{dist}(t))_k$ but have to learn a better compensation command for the next iteration by observing the current input commands and the arm response. Using this procedure a side-by-side quantitative comparison is possible since all controllers are subject to exactly the same sequence of disturbances values.

C. Controller Settings

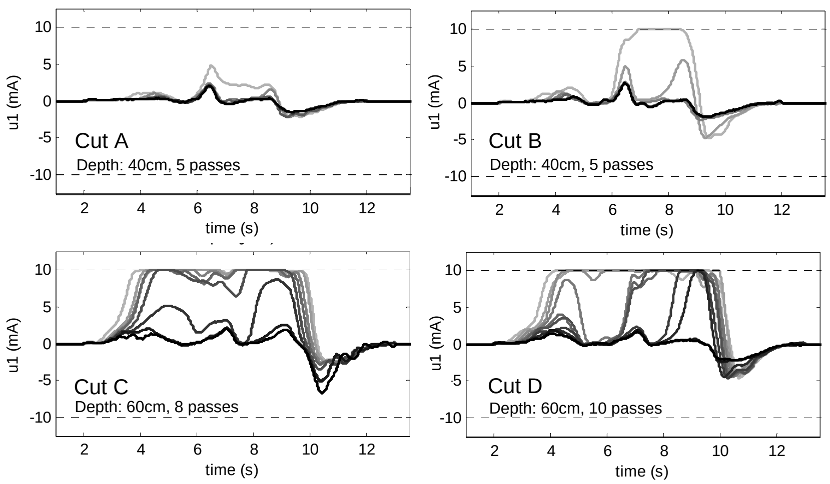
The development of the DLC assumes that an accurate inverse dynamics model of the arm is available, so it is sensible to initialise the first feedforward commands of all evaluated controllers with

$$\mathbf{u}_0 = \boldsymbol{M}(\mathbf{q}_r)\ddot{\mathbf{q}}_r + \mathbf{v}(\mathbf{q}_r, \dot{\mathbf{q}}_r) + \mathbf{g}(\mathbf{q}_r) \,, \quad (12)$$

where $\mathbf{u}_0$ is the vector of required torques for the free motion of the arm, $\boldsymbol{M}$ is the inertia matrix, $\mathbf{v}$ is the vector of centrifugal and Coriolis forces, and $\mathbf{g}$ is the gravity vector. The desired joint positions $\mathbf{q}_r$ are obtained from the inverse kinematics of the path that defines the desired shape of the cut to be excavated.

This work adopts a PD learning rule for the ILC update

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \widetilde{\mathbf{q}}_k P + \dot{\widetilde{\mathbf{q}}}_k D \,,$$



a) Disturbances from experiments

b) Example of a cut

Fig. 5. a) Disturbances recovered from four cuts are used in playback mode to evaluate the controllers (showing only the values for the first joint). The first pass is shown in light grey and progresses gradually until the last pass in black. b) One of the cuts opened during field experiments showed that the soil consisted of clay and scattered pieces of bricks and roots.

where $\widetilde{\mathbf{q}}_k$ is the joint tracking error, P and D are the proportional and derivative gains and $\dot{\widetilde{\mathbf{q}}}_k$ is the joint velocity error, filtered by a moving-average smoothing method.

D. Benchmark

A conventional disturbance observer shown in Fig. 6 is implemented as a benchmark. Amongst several implementations of disturbance observers, a variable structure observer (VSO) has been experimentally validated [19], [20] as providing superior performance with different types of hydraulic machinery. The suitability for hydraulic systems arises in part because VSO induces a sliding mode in the estimated variable, making the compensator robust to error and variations in its internal parameters. These are usually difficult to identify in heavy hydraulic equipment.

The disturbance observer has the following transfer function

$$X_1 = \frac{X_2}{s} + \frac{\sigma}{ms} \tag{13}$$

$$X_2 = (-U + U_{dist} + L_1\sigma)\,\frac{1}{ms + d} \tag{14}$$

$$U_{dist} = \frac{-L_2\sigma}{s} \tag{15}$$

$$\sigma = W tanh\left(\frac{y - X_1}{\gamma_e}\right), \tag{16}$$

where U is the control input, y is the position; X_1, X_2, and U_{dist} are estimates of position, velocity and disturbance torques; m, L_1, L_2, W and γ_e are design parameters, σ is the function that induces the sliding mode and d is the servo-valve gain. Implementation required only the position and control inputs since velocity is estimated internally.

Note that implementing the VSO for each joint required hand-tuning of five parameters plus system identification of the servo-valve gain; a total of 18 design parameters.

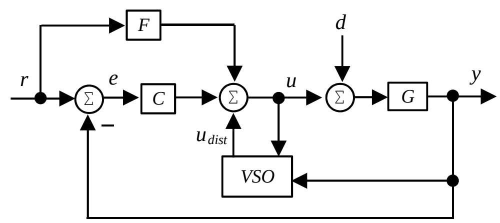


Fig. 6. Controller with VSO for disturbance compensation; C is the feedback gain, G is the plant transfer function, r is the reference, y is the output, u is the command input and u_{dist} is the estimated disturbance.

Although the robustness given by the sliding mode is worth the design effort[1], VSO has been notoriously hard to tune in practice [20].

V. Results

Fig. 7 shows the evolution of the ILC, DLC and the VSO as they are subjected to the same sequence of disturbances. During the initial passes disturbances show large changes in magnitude and the controller with VSO tracks more consistently than the other controllers. In some passes both the ILC and the DLC showed the expected overshoot/undershoot behaviour depending on the changes in disturbance magnitudes, however both controllers maintained convergence towards the desired reference path.

Fig. 8 shows a comparison of the three controllers under the four disturbance data sets in the form of root-mean-square (RMS) error of the distance between the bucket tip and the reference trajectory during each pass. The iterations marked with an asterisk are passes where the disturbance of the last pass without the asterisk is repeated. This is useful to observe how many extra iterations the DLC and the ILC

[1] At least for three joints, but probably not for seven or more.

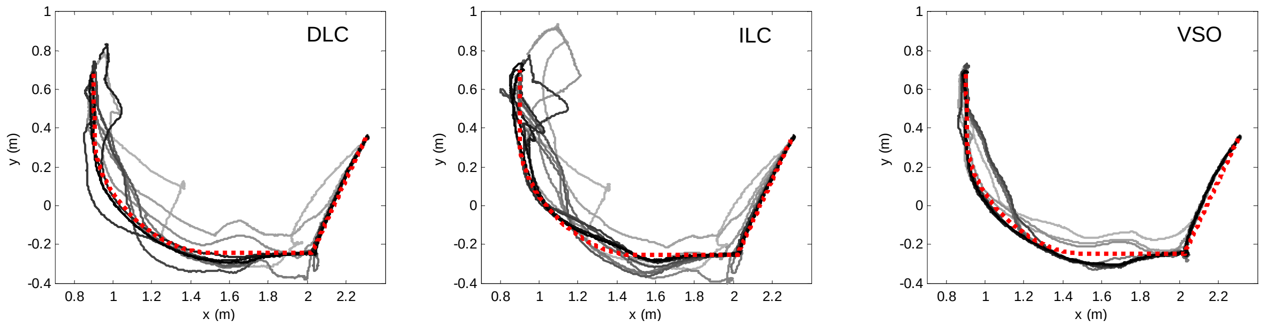


Fig. 7. Iterative passes under the influence of disturbances from cut D. The grey curve represents the trajectory of the bucket tip during the first pass; trajectories evolve to finish at the black curve. The dotted line is the common desired trajectory for all passes.

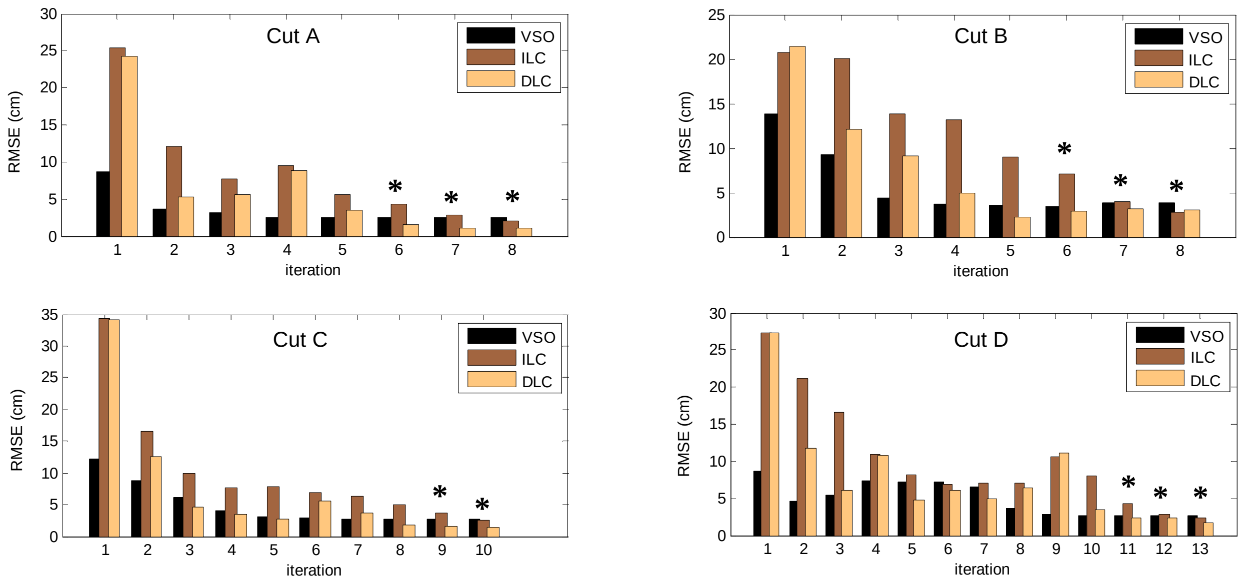


Fig. 8. RMS error distance between the bucket tip and the reference trajectory of the three controllers. The asterisks denote iterations where the previous disturbance vector is artificially repeated to observe convergence under repetitive conditions.

would require to achieve the same performance as the VSO. In all cases both the DLC and the ILC achieve better final accuracy than the controller with VSO since they do not suffer from estimation delays.

Table I summarises the results by comparing the number of iterations that the DLC and the ILC required to achieve an error less than or equal to the minimum error achieved by the VSO. For example, with disturbance data set A, the controller with VSO achieved the minimum RMS error of 2.5 cm at the 5th iteration. DLC took 6 iterations since at the 5th iteration the RMS error value was 3.4 cm. In general the convergence of DLC and VSO were very similar, with the DLC usually delayed by one pass; the ILC required approximately three extra passes when compared to the VSO.

Finally, the DLC controller was evaluated under real excavation conditions where it was deployed on undisturbed sandy-clay-loam soil. The DLC controller was not only used to cut single profiles but also to open long trenches of 4 meters in length and 0.8 meters deep (Fig. 9 a)). As shown in Fig. 9 b), the excavator dug under a variety of rocks distribution that increased in frequency according to the depth of the cut. Fig. 9 c) shows the average error per pass, where the DLC and VSO controllers dug seven different cuts each. The convergence of the DLC was comparable to the VSO controller.

VI. Conclusions

This paper proposed and analysed the use of ILC for a particular class of non-repetitive disturbances found in material removal processes. A parameter-free iterative Disturbance Learning Control (DLC) that approximates an ideal disturbance observer implementation was introduced. While a conventional PD-type ILC learns disturbances by iteratively decreasing the tracking error, DLC learns disturbances directly in the command space; independent of the tracking error. Both control methods were validated by disturbing servo commands of a mini-excavator with commands due to excavation reactions in field experiments. Experimental results showed that DLC achieves faster convergence than ILC during the non-repetitive disturbances passes. During

a) Excavating a long trench (depth: 0.8 m) b) The trench wall showing the heterogeneous conditions of the soil c) Average RMS error comparison between the DLC and VSO controllers

Fig. 9. Validating and comparing the proposed disturbance learning controller against the disturbance observer controller under real excavation conditions.

TABLE I

ILC AND DLC ITERATIONS REQUIRED TO ACHIEVE LESS THAN THE MINIMUM VSO ERROR

	cut	iteration	extra passes	RMS(cm)
VSO min. error	A	5		2.5
DLC		6	1	1.5
ILC		8	3	2.0
VSO min. error	B	6		3.5
DLC		5	-1	2.2
ILC		8	3	2.8
VSO min. error	C	7		2.8
DLC		8	1	1.8
ILC		10	3	2.6
VSO min. error	D	9		2.8
DLC		11	2	2.3
ILC		13	4	2.3

field trials, the DLC achieved the same cut accuracy and convergence when compared with a disturbance observer controller.

VII. ACKNOWLEDGMENTS

The authors would like to thank Russ Tedrake and Surya Singh for brief but fruitful discussions on ILC, and Vitor Guizilini for his assistance during the experiments. This work was supported by the Rio Tinto Centre for Mine Automation and the Australian Centre for Field Robotics, funded in part by the New South Wales State Government.

REFERENCES

[1] D. Hellige, "Building the technologies for the mine sites of the future," *Viewpoint: perspectives on modern mining*, vol. 4, 2008.

[2] C. Latimer, "Australina mining: Rio expands automated truck fleet," 2 November 2011. [Online]. Available: http://www.miningaustralia.com.au/news/rio-expands-automated-truck-fleet

[3] X. Fan, S. Singh, F. Oppolzer, E. Nettleton, R. Hennessy, A. Lowe, and H. Durrant-Whyte, "Integrated planning and control of large tracked vehicles in open terrain," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 4424–4430.

[4] G. Maeda, D. Rye, and S. Singh, "Iterative autonomous excavation," in *Accepted: International Conference on Field and Service Robotics*, 2012.

[5] H. Merritt, *Hydraulic control systems*. John Wiley & Sons Inc, 1967.

[6] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *Journal of robotic systems*, vol. 1, no. 2, pp. 123–140, 1984.

[7] O. Luengo, S. Singh, and H. Cannon, "Modeling and identification of soil-tool interaction in automated excavation," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, vol. 3. IEEE, 1998, pp. 1900–1906.

[8] S. Blouin, A. Hemami, and M. Lipsett, "Review of resistive force models for earthmoving processes," *Journal of Aerospace Engineering*, vol. 14, p. 102, 2001.

[9] A. Wilkinson and A. DeGennaro, "Digging and pushing lunar regolith: Classical soil mechanics and the forces needed for excavation and traction," *Journal of Terramechanics*, vol. 44, no. 2, pp. 133–152, 2007.

[10] S. Salcudean, S. Tafazoli, P. Lawrence, and I. Chau, "Impedance control of a teleoperated mini excavator," in *Proc. of the 8th IEEE International Conference on Advanced Robotics*. Citeseer, 1997.

[11] Q. Ha, M. Santos, Q. Nguyen, D. Rye, and H. Durrant-Whyte, "Robotic excavation in construction automation," *IEEE Robotics & Automation Magazine*, vol. 9, no. 1, pp. 20–28, 2002.

[12] D. Owens and E. Rogers, "Stability analysis for a class of 2-D continuous–discrete linear systems with dynamic boundary conditions," *Systems & control letters*, vol. 37, no. 1, pp. 55–60, 1999.

[13] R. Longman, "Iterative learning control and repetitive control for engineering practice," *International Journal of Control*, vol. 73, no. 10, pp. 930–954, 2000.

[14] S. Mishra, J. Coaplen, and M. Tomizuka, "Precision positioning of wafer scanners segmented iterative learning control for nonrepetitive disturbances [applications of control]," *Control Systems Magazine*, vol. 27, no. 4, pp. 20–25, 2007.

[15] Y. Chen and K. Moore, "Harnessing the nonrepetitiveness in iterative learning control," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 3, 2002, pp. 3350–3355.

[16] M. Norrlöf and S. Gunnarsson, "Disturbance aspects of iterative learning control," *Engineering Applications of Artificial Intelligence*, vol. 14, no. 1, pp. 87–94, 2001.

[17] C. Kempf and S. Kobayashi, "Disturbance observer and feedforward design for a high-speed direct-drive positioning table," *IEEE Transactions on Control Systems Technology*, vol. 7, no. 5, pp. 513–526, 1999.

[18] G. Maeda, S. Singh, and D. Rye, "Improving operational space control of heavy manipulators via open-loop compensation," in *Proceedings of the IEEE/RSJ International Conferencen on Intelligent Robots and Systems (IROS)*, 2011, pp. 725–731.

[19] Q. Ha, A. Bonchis, D. Rye, and H. Durrant-Whyte, "Variable structure systems approach to friction estimation and compensation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, vol. 4, 2000, pp. 3543–3548.

[20] A. Bonchis, P. Corke, and D. Rye, "Experimental evaluation of position control methods for hydraulic systems," *Control Systems Technology, IEEE Transactions on*, vol. 10, no. 6, pp. 876–882, 2002.