Check for updates

Online Conversion with Switching Costs: *Robust and Learning-Augmented Algorithms*

Adam Lechowicz
University of Massachusetts Amherst
alechowicz@cs.umass.edu

Nicolas Christianson
California Institute of Technology
nchristianson@caltech.edu

Bo Sun
University of Waterloo
bo.sun@uwaterloo.ca

Noman Bashir
Massachusetts Institute of Technology
nbashir@mit.edu

Mohammad Hajiesmaili
University of Massachusetts Amherst
hajiesmaili@cs.umass.edu

Adam Wierman
California Institute of Technology
adamw@caltech.edu

Prashant Shenoy
University of Massachusetts Amherst
shenoy@cs.umass.edu

ABSTRACT

We introduce and study online conversion with switching costs, a family of online problems that capture emerging problems at the intersection of energy and sustainability. In this problem, an online player attempts to purchase (alternatively, sell) fractional shares of an asset during a fixed time horizon with length T. At each time step, a cost function (alternatively, price function) is revealed, and the player must irrevocably decide an amount of asset to convert. The player also incurs a *switching cost* whenever their decision changes in consecutive time steps, i.e., when they increase or decrease their purchasing amount. We introduce competitive (robust) threshold-based algorithms for both the minimization and maximization variants of this problem, and show they are optimal among deterministic online algorithms. We then propose learning-augmented algorithms that take advantage of untrusted black-box advice (such as predictions from a machine learning model) to achieve significantly better average-case performance without sacrificing worst-case competitive guarantees. Finally, we empirically evaluate our proposed algorithms using a carbon-aware EV charging case study, showing that our algorithms substantially improve on baseline methods for this problem.

ACM Reference Format:

Adam Lechowicz, Nicolas Christianson, Bo Sun, Noman Bashir, Mohammad Hajiesmaili, Adam Wierman, and Prashant Shenoy. 2024. Online Conversion with Switching Costs: *Robust and Learning-Augmented Algorithms.* In *Abstracts of the 2024 ACM SIGMETRICS/IFIP PERFORMANCE Joint International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS/PERFORMANCE Abstracts '24), June 10–14, 2024, Venice, Italy.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3652963.3655074

The full version of the paper is available at https://arxiv.org/abs/2310.20598.

SIGMETRICS/PERFORMANCE Abstracts '24, June 10–14, 2024, Venice, Italy

ACM ISBN 979-8-4007-0624-0/24/06.
https://doi.org/10.1145/3652963.3655074

1 PROBLEM FORMULATION

We present the online conversion with switching costs (OCS) problem, focusing on the minimization variant (OCS-min) in this abstract. In OCS-min, an online player must buy an asset with total size C, while minimizing their total cost. Without loss of generality, let $C = 1$. At each time step $t \in [T]$, a *convex cost function* $g_t(\cdot)$ arrives online. The player can buy $x_t \in [0, d_t]$ amount of the asset at a cost of $g_t(x_t)$, where $d_t \leq 1$ is a *rate constraint* that limits the purchase amount. Following convention, $g_t(0) = 0$; i.e., if the player purchases nothing, they pay no cost, and $g_t(x_t) \geq 0$ for any valid x_t. Whenever the player's decision changes in consecutive time steps, they incur a *switching cost* that is formalized as $\beta|x_t - x_{t-1}|$, where β is a coefficient charging the online player proportionally to their absolute movement. We let $x_0 = 0$ and $x_{T+1} = 0$, forcing any player to incur *some* switching costs to "turn on" and "off", respectively.

The online player must purchase the entire asset before the end of the sequence (the "deadline"). If the player has bought $w^{(t)} \in [0, 1]$ fraction of the asset at time t, a *compulsory trade* begins when $\sum_{\tau=t+1}^{T} d_\tau < 1 - w^{(t)}$ (i.e., when the future purchase opportunities will not be enough). During this compulsory trade, a cost-agnostic algorithm takes over and purchases maximally to satisfy the constraint. The offline version of OCS-min can be formalized as follows:

$$\min_{\{x_t \in [0,d_t] : t \in [T]\}} \underbrace{\sum_{t=1}^{T} g_t(x_t)}_{\text{purchasing}} + \underbrace{\sum_{t=1}^{T+1} \beta|x_t - x_{t-1}|}_{\text{switching}}, \ \text{s.t.,} \ \underbrace{\sum_{t=1}^{T} x_t = 1}_{\text{deadline}}, \quad (1)$$

Our focus is on the online version of OCS, where the player must make irrevocable decisions x_t at each time step without the knowledge of future inputs. The most important unknowns are the cost functions $g_t(\cdot)$, which are revealed online.

Competitive analysis. Our goal is to design an online algorithm that maintains a small *competitive ratio*. For an online algorithm ALG and an offline optimal solution OPT, ALG is b-competitive if $\texttt{ALG}(\mathcal{I}) \leq b\texttt{OPT}(\mathcal{I}) \ \forall \mathcal{I} \in \Omega$, where $\mathcal{I}$ denotes a valid input sequence for the problem and Ω is the set of all feasible inputs.

In the emerging literature on learning-augmented algorithms, competitive analysis is interpreted through *consistency* and *robustness*. Let $\texttt{LALG}(\mathcal{I}, \varepsilon)$ denote the cost of learning-augmented algorithm LALG on input $\mathcal{I}$ when provided predictions with error ε.

LALG is c-consistent when predictions are correct if $\texttt{LALG}(\mathcal{I}, 0) \le c\texttt{OPT}(\mathcal{I})\ \forall \mathcal{I} \in \Omega$, and r-robust if $\texttt{LALG}(\mathcal{I}, \mathbf{E}) \le r\texttt{OPT}(\mathcal{I})\ \forall \mathcal{I} \in \Omega$, where $\mathbf{E}$ is a maximum error (or ∞).

Assumptions and additional notation. We assume that cost functions $\{g_t(\cdot)\}_{t\in[T]}$ have a bounded derivative, i.e. $L \le dg_t/dx_t \le U$, where L and U are known. We assume that all $g_t(\cdot)$ are *convex* – this models *diminishing returns*, and is empirically valid for the applications of interest. The switching cost coefficient β is known to the player, and is bounded within an interval $\beta \in (0, {U-L}/{2})$.

2 ALGORITHMS AND MAIN RESULTS

Algorithm 1 Online Ramp-On, Ramp-Off (RORO) framework

input: $\textsc{RampOn}(\cdot)$ problem, $\textsc{RampOff}(\cdot)$ problem ,
pseudo-cost function $\textsc{PCost}(\cdot)$
initialization: initial decision $x_0 = 0$, initial utilization $w^{(0)} = 0$;
while cost/price function $g_t(\cdot)$ is revealed and $w^{(t-1)} < 1$ **do**
  solve the **(ramping-on problem)** to obtain decision x_t^+ and its pseudo cost r_t^+,

$$x_t^+ = \textsc{RampOn}(g_t(\cdot), x_{t-1}), \tag{2}$$

$$r_t^+ = \textsc{PCost}(g_t(\cdot), x_t^+, x_{t-1}). \tag{3}$$

  solve the **(ramping-off problem)** to obtain decision x_t^- and its pseudo cost r_t^-,

$$x_t^- = \textsc{RampOff}(g_t(\cdot), x_{t-1}), \tag{4}$$

$$r_t^- = \textsc{PCost}(g_t(\cdot), x_t^-, x_{t-1}). \tag{5}$$

  if $r_t^+ \le r_t^-$ **then** set $x_t = x_t^+$ **else** set $x_t = x_t^-$;
  update the utilization $w^{(t)} = w^{(t-1)} + x_t$;

Competitive algorithms. We present an online optimization framework called Ramp-On, Ramp-Off (RORO). At each time step RORO solves two *pseudo-cost minimization problems*, with a restricted decision space in each (the **ramping-on** and **ramping-off** problems). Pseudo-cost minimization is an online search technique that generalizes *threshold-based design* for continuous decision spaces.

In the full paper [1], we provide more context about how the RORO framework's *dynamic threshold* approach simultaneously generalizes prior work [2] on pseudo-cost minimization for one-way trading and our prior work on an online search problem with switching costs and a binary decision space (online pause and resume).

Definition 1 (Dynamic threshold ϕ for OCS-min). *For any utilization $w \in [0, 1]$, $\phi(w) = U - \beta + ({U}/{\alpha} - U + 2\beta)\exp({w}/{\alpha})$, where α is the competitive ratio and is defined in (9).*

Definition 2 (RORO instantiation for OCS-min (RORO-min)). *RORO solves OCS-min when instantiated with the following pseudo-cost, ramping-on problem, and ramping-off problem:*

$$\textsc{PCost}(g_t(\cdot), x_t, x_{t-1}) = g_t(x_t) + \beta|x_t - x_{t-1}| - \int_{w^{(t-1)}}^{w^{(t-1)}+x_t} \phi(u)du, \tag{6}$$

$$\textsc{RampOn}(g_t(\cdot), x_{t-1}) = \underset{x\in[x_{t-1}, \min(1-w^{(t-1)}, d_t)]}{\arg\min} \textsc{PCost}(g_t(\cdot), x, x_{t-1}), \tag{7}$$

$$\textsc{RampOff}(g_t(\cdot), x_{t-1}) = \underset{x\in[0, \min(x_{t-1}, d_t)]}{\arg\min} \textsc{PCost}(g_t(\cdot), x, x_{t-1}). \tag{8}$$

In the following, we state our main theoretical results for OCS-min.

Theorem 3. *RORO-min is α-competitive for OCS-min, where α is the solution to $\frac{U-L-2\beta}{U/\alpha - U - 2\beta} = \exp(1/\alpha)$ and is given by*

$$\alpha := \left[W\left(\left(\frac{2\beta}{U} + \frac{L}{U} - 1 \right) e^{2\beta/U - 1} \right) - \frac{2\beta}{U} + 1 \right]^{-1}. \tag{9}$$

In the above, $W(\cdot)$ is the Lambert W function.

Theorem 4. *No deterministic online algorithm for OCS-min can achieve a competitive ratio better than α, as defined in (9).*

Learning-augmentation. We consider how *untrusted advice* (e.g., from an ML model) can help break past pessimistic competitive bounds for OCS. We propose a meta-algorithm, RO-Advice, that integrates *black-box advice* to significantly improve performance.

Definition 5 (Black-box advice model for OCS). *A learning-augmented algorithm LALG receives advice of the form $\{\hat{x}_t\}_{t\in[T]}$ for some valid instance $\mathcal{I}$. If the advice is correct, a naïve algorithm ADV choosing $\hat{x}_t$ at each time step satisfies $\texttt{ADV}(\mathcal{I}) = \texttt{OPT}(\mathcal{I})$.*

RO-Advice combines the robust decision of RORO (denoted by $\tilde{x}_t$) at each time step with the predicted $\hat{x}_t$ obtained from the black-box advice. Let $\epsilon \in [0, \alpha - 1]$ parameterize a trade-off between consistency and robustness. RO-Advice-min sets a combination factor $\lambda := \frac{\alpha - 1 - \epsilon}{\alpha - 1} \in [0, 1]$ that determines the decision fraction from each subroutine (i.e., λ from the black-box advice and $(1 - \lambda)$ from RORO). At each time step, RO-Advice chooses the online decision $x_t = \lambda \hat{x}_t + (1 - \lambda)\tilde{x}_t$.

Theorem 6. *Given a parameter $\epsilon \in [0, \alpha - 1]$, RO-Advice-min is $(1 + \epsilon)$-consistent and $\left(\frac{(U+2\beta)/L(\alpha-1-\epsilon)+\alpha\epsilon}{(\alpha-1)} \right)$-robust for OCS-min.*

In the full paper [1], we prove and discuss all of the above results in detail. We also provide additional theoretical results about the *advice complexity* of OCS, showing that prior advice models used for e.g., one-way trading are insufficient for the OCS setting.

Experiments. In the full paper [1], we implement and evaluate RORO and RO-Advice for the motivating task of carbon-aware electric vehicle (EV) charging. We use real EV charging traces, carbon intensity data, and a pre-trained open-source ML model for carbon forecasts, showing that RORO and RO-Advice perform well.

ACKNOWLEDGMENTS

This research is supported by National Science Foundation grants CAREER-2045641, CNS-2102963, CNS-2106299, CNS-2146814, CNS-1518941, CPS-2136197, CPS-2136199, NGSDI-2105494, NGSDI-2105648, 1908298, 2020888, 2021693, 2045641, 2213636, 2211888, and an NSF Graduate Research Fellowship (DGE-1745301).

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Department of Energy Computational Science Graduate Fellowship under Award Number DE-SC0024386.

REFERENCES

[1] Adam Lechowicz, Nicolas Christianson, Bo Sun, Noman Bashir, Mohammad Hajiesmaili, Adam Wierman, and Prashant Shenoy. 2024. Online Conversion with Switching Costs: Robust and Learning-augmented Algorithms. arXiv:2310.20598

[2] Bo Sun, Ali Zeynali, Tongxin Li, Mohammad Hajiesmaili, Adam Wierman, and Danny H.K. Tsang. 2020. Competitive Algorithms for the Online Multiple Knapsack Problem with Application to Electric Vehicle Charging. *Proc. of the ACM on Measurement and Analysis of Computing Systems* 4, 3, Article 51 (Dec 2020).