The Online Pause and Resume Problem: *Optimal Algorithms and An Application to Carbon-Aware Load Shifting*

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

Nicolas Christianson
California Institute of Technology
nchristianson@caltech.edu

Jinhang Zuo
Caltech & UMass Amherst
jhzuo@cs.umass.edu

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 the online pause and resume problem. In this problem, a player attempts to find the k lowest (alternatively, highest) prices in a sequence of fixed length T, which is revealed sequentially. At each time step, the player is presented with a price and decides whether to accept or reject it. The player incurs a *switching cost* whenever their decision changes in consecutive time steps, i.e., whenever they pause or resume purchasing. This online problem is motivated by the goal of carbon-aware load shifting, where a workload may be paused during periods of high carbon intensity and resumed during periods of low carbon intensity and incurs a cost when saving or restoring its state. It has strong connections to existing problems studied in the literature on online optimization, though it introduces unique technical challenges that prevent the direct application of existing algorithms. Extending prior work on threshold-based algorithms, we introduce *double-threshold* algorithms for both variants of this problem. We further show that the competitive ratios achieved by these algorithms are the best achievable by any deterministic online algorithm. Finally, we empirically validate our proposed algorithm through case studies on the application of carbon-aware load shifting using real carbon trace data and existing baseline algorithms.

ACM Reference Format:
Adam Lechowicz, Nicolas Christianson, Jinhang Zuo, Noman Bashir, Mohammad Hajiesmaili, Adam Wierman, and Prashant Shenoy. 2024. The Online Pause and Resume Problem: *Optimal Algorithms and An Application to Carbon-Aware Load Shifting*. 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/10.1145/3652963.3655086

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

ACM ISBN 979-8-4007-0624-0/24/06.
https://doi.org/10.1145/3652963.3655086

1 PROBLEM FORMULATION

We present the online pause and resume problem (OPR), focusing on the minimization version (OPR-min) in this abstract, and deferring the maximization version to the full paper. In OPR-min. a player must buy $k \geq 1$ units of some asset (one unit at each time step) with the goal of minimizing their total cost. At each time step $1 \leq t \leq T$, the player is presented with a price c_t, and must immediately decide whether to accept this price ($x_t = 1$) or reject it ($x_t = 0$). The player is required to complete this transaction for all k units at or before time T. Both k and T are known in advance. The requirement of k transactions is a *deadline* constraint, i.e., $\sum_{t=1}^{T} x_t = k$, and if at time $T - i$ the player still has i units remaining to buy/sell, they must accept the prices in the subsequent i slots to accomplish k transactions. Additionally, the player incurs a *fixed switching cost* $\beta > 0$ whenever they decide to change decisions between two adjacent time steps (i.e., $|x_{t-1} - x_t| = 1$). We assume $x_0 = x_{T+1} = 0$, implying that any player must incur a minimum switching cost of 2β, once for switching "on" and once for switching "off". We also note that the total switching cost incurred by the player is bounded by the size of the asset k, since the switching cost cannot be larger than $k2\beta$. The offline version of OPR-min is summarized as follows:

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

Our focus is the online version of the above, where the player must make irrevocable decisions at each time step without the knowledge of future inputs. More specifically, the sequence of prices $\{c_t\}_{t \in [T]}$ is revealed sequentially – future prices are *unknown* to an online algorithm, and each decision x_t is irrevocable.

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 c-competitive if $\text{ALG}(\mathcal{I}) \leq c\text{OPT}(\mathcal{I})\ \forall \mathcal{I} \in \Omega$, where $\mathcal{I}$ denotes a valid input sequence for the problem and Ω is the set of all feasible inputs.

Assumptions and additional notations. We make no assumptions on the underlying price distribution other than assuming that the set of prices arriving online $\{c_t\}_{t \in [T]}$ has bounded support, i.e.,

$c_t \in [L, U]\ \forall t \in [T]$, where $L \geq 0$ and $U > 0$ are known to the player. We also define $\theta = U/L$ as the *price fluctuation*. These are standard assumptions in the literature for many online problems, including one-way trading, online search, and online knapsack; and without them the competitive ratio of any algorithm is unbounded.

Relation to k-search. The OPR problem is a generalization of the k-search problem [2], which belongs to a broader class of online conversion problems. OPR generalizes k-search by adding the switching cost, which poses a significant additional challenge in algorithm design. We note that OPR exactly reduces to k-search as $\beta \longrightarrow 0$.

2 ALGORITHMS AND MAIN RESULTS

Algorithm 1 Double Threshold Pause and Resume for OPR-min (DTPR-min)

Input: threshold values $\{\ell_i\}_{i\in[k]}$ and $\{u_i\}_{i\in[k]}$ (2), deadline T
Output: online decisions $\{x_t\}_{t\in[T]}$
initialize: $i = 1$
while price c_t arrives **and** $i \leq k$ **do**
  if $(k - i) \geq (T - t)$ **then** ▷ must accept remaining prices
    price c_t is accepted, set $x_t = 1$
  else if $x_{t-1} = 0$ **then** ▷ If previous price was not accepted
    if $c_t \leq \ell_i$ **then** price c_t is accepted, set $x_t = 1$
    else price c_t is rejected, set $x_t = 0$
  else if $x_{t-1} = 1$ **then** ▷ If previous price was accepted
    if $c_t \leq u_i$ **then** price c_t is accepted, set $x_t = 1$
    else price c_t is rejected, set $x_t = 0$
  update $i = i + x_t$

We propose *double threshold* algorithms for both variants of this problem, abbreviated by DTPR and summarized in Algorithm 1. Prior to any prices arriving online, DTPR-min computes two families of threshold values, $\{\ell_i\}_{i\in[k]}$ and $\{u_i\}_{i\in[k]}$, where $\ell_i \leq u_i\ \forall i \in [k]$, defined below. The DTPR algorithm then chooses a family of thresholds to use based on *previous online decision*, i.e., $x_{t-1} \in \{0, 1\}$.

Definition 1 (DTPR-min Threshold Values). *For each $i \in [k]$, the following expressions give the corresponding threshold values of u_i and ℓ_i for* DTPR-min.

$$u_i = U - \left(U - \frac{U}{\alpha}\right)\left(1 + \frac{1}{k\alpha}\right)^{i-1} + \left(\frac{2\beta}{k\alpha} - \frac{2\beta}{k} + 2\beta\right)\left(1 + \frac{1}{k\alpha}\right)^{i-1} \quad (2)$$

where $\ell_i = u_i - 2\beta$ and α is the competitive ratio of DTPR-min *from* (3).

The key idea of DTPR is to design the thresholds in Equation (2) by incorporating the switching cost as a hedge against possible worst-case scenarios. We discuss the design of these thresholds in detail in the full paper [1]. The intuition behind this double threshold technique is to address a shortcoming in threshold-based algorithm design, which is oblivious to the switching cost present in OPR. By adding some "resistance to change", the double thresholds allow DTPR to exhibit desirable behavior: (1) when DTPR is in "trading mode," it will not impulsively switch off in response to a price that is only slightly worse, since this would result in a switching penalty; and (2) DTPR will not switch to "trading mode" unless prices are sufficiently good to justify the switching cost. In the following theorems, we state our main theoretical results for DTPR when applied to the minimization version of OPR.

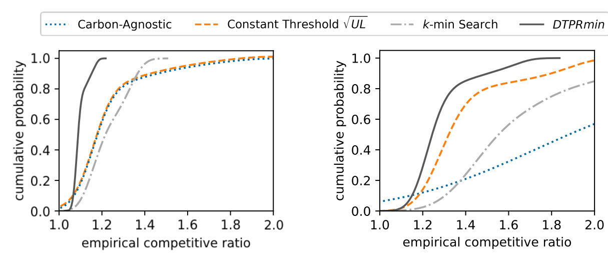
Theorem 2. DTPR-min *is an α-competitive deterministic algorithm for* OPR-min, *where α is the unique positive solution of*

$$\frac{U - L - 2\beta}{U(1 - 1/\alpha) - \left(2\beta - \frac{2\beta}{k} + \frac{2\beta}{k\alpha}\right)} = \left(1 + \frac{1}{k\alpha}\right)^k. \quad (3)$$

To investigate the tightness of Theorem 2, we first consider special cases that correspond to prior work. DTPR exactly recovers the optimal single-threshold based k-search algorithms [2] when $\beta \longrightarrow 0$ (i.e., when OPR degenerates to k-search). Outside of this special case, one can ask if the competitive ratios of DTPR can be improved upon. The next result highlights that no improvement is possible, i.e., that DTPR-min achieves the optimal competitive ratio for any deterministic algorithm solving OPR-min.

Theorem 3. *Let $k \geq 1$, $\theta \geq 1$, and $\beta \in (0, \frac{U-L}{2})$. Then α given by Equation* (3) *is the best competitive ratio that a deterministic online algorithm for* OPR-min *can achieve.*

In the full paper [1], we prove and discuss our results for both the minimization and maximization settings in detail. We also provide an array of empirical experiments for the motivating application of carbon-aware load shifting using real carbon data. In Fig. 1, we plot example plots of DTPR's performance versus three baseline methods from the literature, showing that DTPR performs well.



(a) New Zealand **(b) Ontario, Canada**

Figure 1: Example CDF plots of the empirical competitive ratios for DTPR and three baseline methods, using carbon traces from two electric grids for the carbon-aware load shifting task. Experiment details can be found in the full paper [1].

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, Jinhang Zuo, Noman Bashir, Mohammad Hajiesmaili, Adam Wierman, and Prashant Shenoy. 2023. The Online Pause and Resume Problem: Optimal Algorithms and An Application to Carbon-Aware Load Shifting. *Proc. of the ACM on Measurement and Analysis of Computing Systems* 7, 3, Article 53 (Dec 2023), 36 pages. arXiv:2303.17551 [cs.DS]

[2] Julian Lorenz, Konstantinos Panagiotou, and Angelika Steger. 2008. Optimal Algorithms for k-Search with Application in Option Pricing. *Algorithmica* 55, 2 (Aug. 2008), 311–328. https://doi.org/10.1007/s00453-008-9217-8