

[Home](#) > [Publications](#) >

Online Submodular Welfare Maximization: Greedy Beats 1/2 in Random Order

[Nitish Korula](#) · [Vahab S. Mirrokni](#) · [Morteza Zadimoghaddam](#) ·

STOC (2015), pp. 889-898

Download

Google Scholar

Copy Bibtex

Abstract

In the Submodular Welfare Maximization (SWM) problem, the input consists of a set of n items, each of which must be allocated to one of m agents. Each agent I has a valuation function v_I , where $v_I(S)$ denotes the welfare obtained by this agent if she receives the set of items S . The functions v_I are all submodular; as is standard, we assume that they are monotone and $v_I(\emptyset) = 0$. The goal is to partition the items into m disjoint subsets $S_1, S_2, \dots, S_m$ in order to maximize the social welfare, defined as $\sum_I v_I(S_I)$. A simple greedy algorithm gives a $1/2$ -approximation to SWM in the offline setting, and this was the best known until Vondrak's recent $(1-1/e)$ -approximation algorithm [34]. In this paper, we consider the online version of SWM. Here, items arrive one at a time in an online manner; when an item arrives, the algorithm must make an irrevocable decision about which agent to assign it to before seeing any subsequent items. This problem is motivated by applications to Internet advertising, where user ad impressions must be allocated to advertisers whose value is a submodular function of the set of users / impressions they receive. There are two natural models that differ in the order in which items arrive. In the fully adversarial setting, an adversary can construct an arbitrary / worst-case instance, as well as pick the order in which items arrive in order to minimize the

algorithm's performance. In this setting, the $1/2$ -competitive greedy algorithm is the best possible. **Research** To improve on this, one must weaken the adversary slightly: In the random order model, the adversary can construct a worst-case set of items and valuations, but does not control the order in which the items arrive; instead, they are assumed to arrive in a random order. The random order model has been well studied for online SWM and various special cases, but the best known competitive ratio (even for several special cases) is $1/2 + 1/n$ [9,10], barely better than the ratio for the adversarial order. Obtaining a competitive ratio of $1/2 + \Omega(1)$ for the random order model has been an important open problem for several years. We solve this open problem by demonstrating that the greedy algorithm has a competitive ratio of at least 0.505 for online SWM in the random order model. This is the first result showing a competitive ratio bounded above $1/2$ in the random order model, even for special cases such as the weighted matching or budgeted allocation problems (without the so-called 'large capacity' assumptions). For special cases of submodular functions including weighted matching, weighted coverage functions and a broader class of "second-order supermodular" functions, we provide a different analysis that gives a competitive ratio of 0.51. We analyze the greedy algorithm using a factor-revealing linear program, bounding how the assignment of one item can decrease potential welfare from assigning future items. We also formulate a natural conjecture which, if true, would improve the competitive ratio of the greedy algorithm to at least 0.567.

In addition to our new competitive ratios for online SWM, we make two further contributions: First, we define the classes of second-order modular, supermodular, and submodular functions, which are likely to be of independent interest in submodular optimization. Second, we obtain an improved competitive ratio via a technique we refer to as gain linearizing, which may be useful in other contexts (see [26]): Essentially, we linearize the submodular function by dividing the gain of an optimal solution into gain from individual elements, compare the gain when it assigns an element to the optimal solution's gain from the element, and, crucially, bound the extent to which assigning elements can affect the potential gain of other elements.

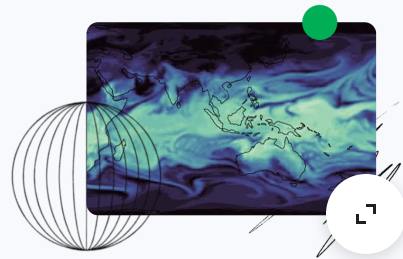
Research Areas



Meet the teams driving innovation

Our teams advance the state of the art through research, systems engineering, and collaboration across Google.

See our teams



Follow us



Explore our other initiatives

Google AI

Discover how Google AI is committed to enriching knowledge and solving complex challenges

Products

Build

Research

Responsibility

Google Cloud

High-performance infrastructure for cloud computing, data analytics & machine learning

Overview

Solutions

Products

Pricing

Research

Societal Impact

Resources

About

Google DeepMind

Our mission is to build AI responsibly to benefit humanity

Models

Research

Science

About

Google Labs

Explore the future of AI responsibly with Google Labs

About

Experiments

Stay connected

Google

About Google

Google Products

Privacy

Terms