THE UNIVERSITY OF SYDNEY

RANDOM DATA LAYOUTS

Cameron Silvestrini
Julián Mestre
School of Information Technologies
FACULTY OF ENGINEERING & INFORMATION TECHNOLOGIES

AIMS OF PROJECT

- Design robust and demand assignment-agnostic data layout
- Analyse the data layout and prove performance bounds
- Investigate performance on a variety of real-world-inspired demand distributions

MOTIVATION

Consider a large data centre, which streams movies or other multimedia to customers via the internet all over the world. The storage needs are considerable, and the bandwidth requirements are immense: the large customer-base is in constant demand of high-quality videos, music, etc., and the load on the servers is taxing. The items in the multimedia collection vastly outnumber the servers in the data centre, so a means of allocating those items among the storage disks is needed—one which spreads the customers' demand out over the servers as evenly as possible.

A data layout provides a way of allocating data items to storage disks, so that the demand on any one server is never too high, ensuring each customer has a good connection.

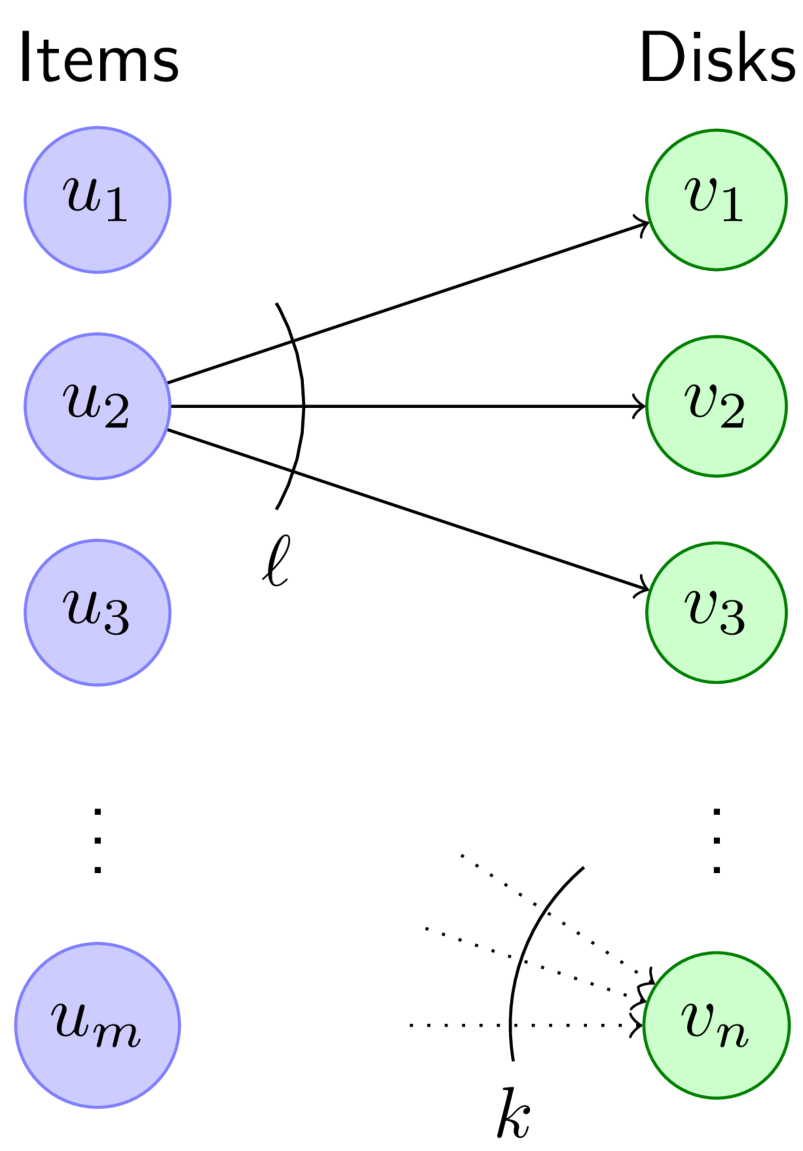
We consider the data layout problem with the condition of *uncertain demands*—where the demand for each item is unknown to the data layout algorithm. We aim to find a robust algorithm that achieves good performance for any distribution of demands.

MODEL

Data Layout

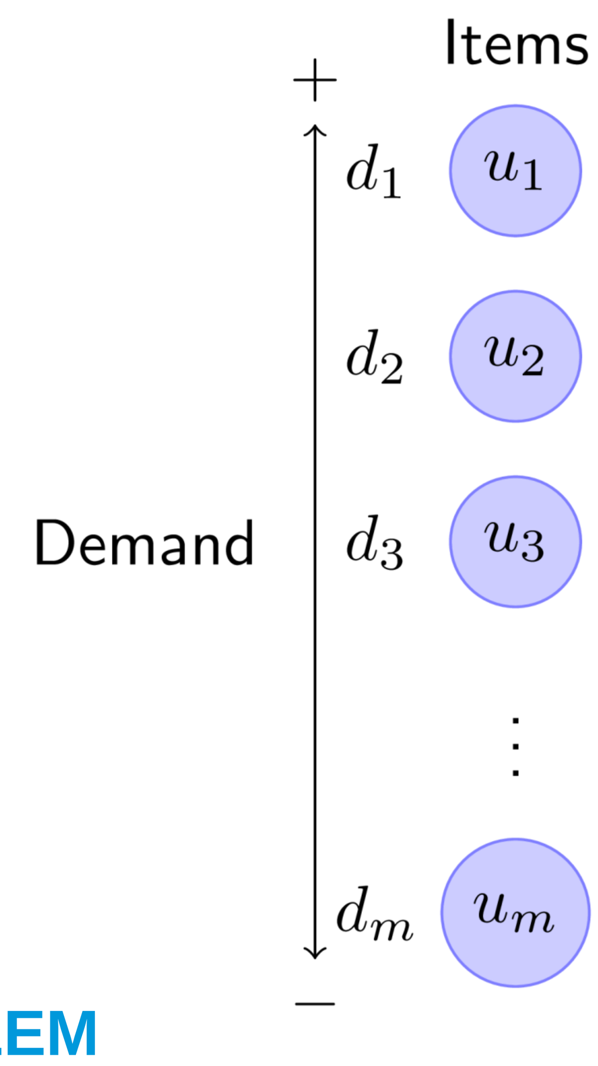
Let $U = \{u_1, \ldots, u_m\}$ be the set of m data items that are to be copied onto the servers. The corresponding demands for the items are labelled $d_1, \ldots, d_m$. Let $V = \{v_1, \ldots, v_n\}$ be the set of n storage disks, each having capacity k. We require that there are to be ℓ copies made of each item, which together totally fill the servers, so $\ell m = kn$.

We can regard (U, V) as a bipartite graph, with an edge from an item u to a disk v occurring when u is copied into v (shown below).



Demands

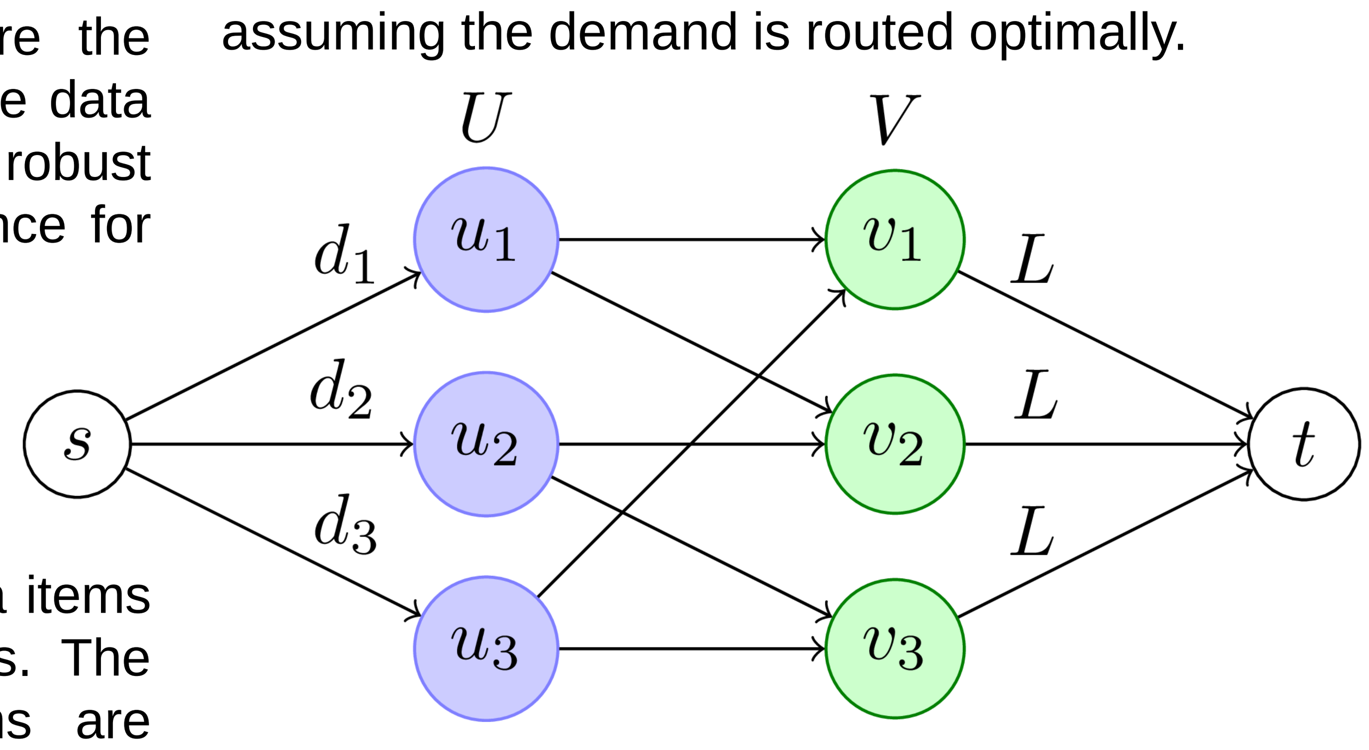
The items are listed in descending order by demand, so that $d_1 \geq \cdots \geq d_m$, but these demands and their ordering are not known ahead of time by the data layout.



PROBLEM

Maximum Load

Once a data layout is established, we can route the incoming demand for each item to the various servers on which it is stored. The *maximum load* is defined to be the total demand on the most overloaded server, assuming the demand is routed optimally.



Objective

The aim is to find a data layout algorithm that will minimise the expected maximum load across the servers.

A method of determining the maximum load computationally is to solve a network flow instance constructed from the bipartite graph of items and disks (shown above). Outgoing edges from vertices in V are given capacity L. If the maximum flow equals the total demand, then a maximum load of L is attainable.

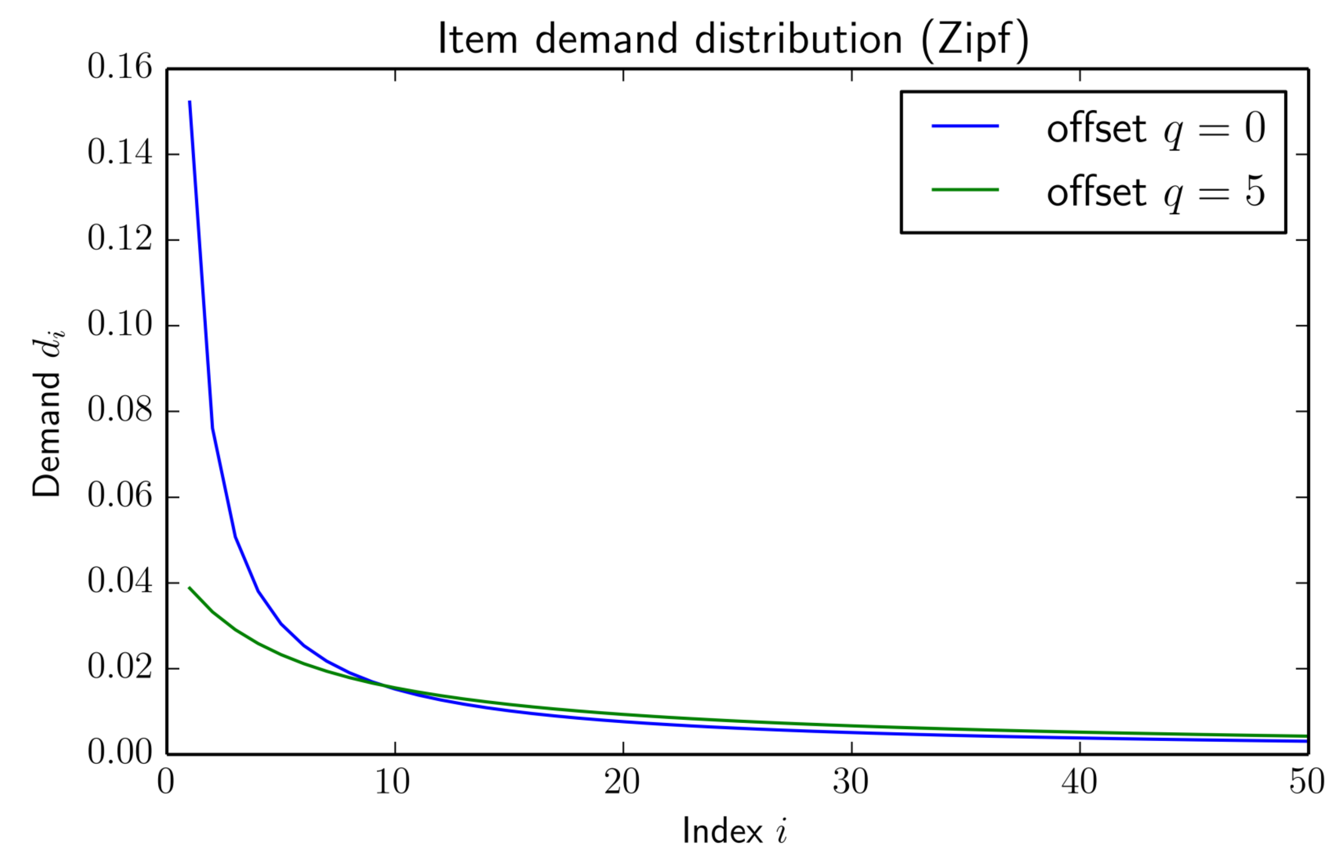
Uniform Random Data Layouts

The layout we primarily considered is where the item copies are allocated uniformly at random.

EXPERIMENTS

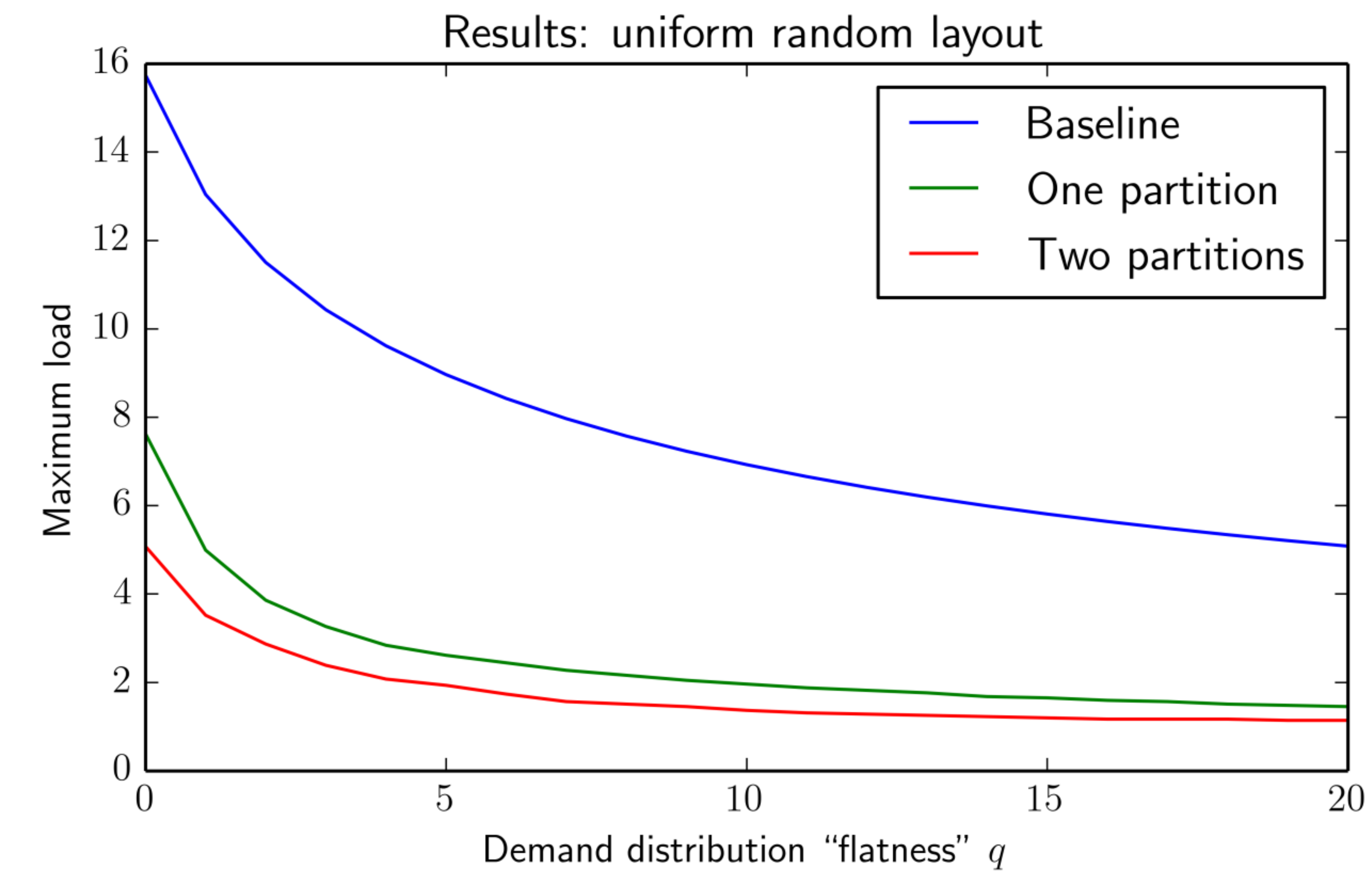
For the majority of our experiments, we modelled the client demand using a Zipf distribution, tuning the distribution parameters in various ways. The *offset* q is used to control the "flatness" of the distribution (shown on right).

This distribution mimics the idea that only a few items in a collection are extremely popular, while the large bulk of items is comparatively ignored by clients—a realistic assumption.



Baseline

We compared the uniform random layout with a naïve deterministic baseline model, which allocates the copies of the items fairly systematically.



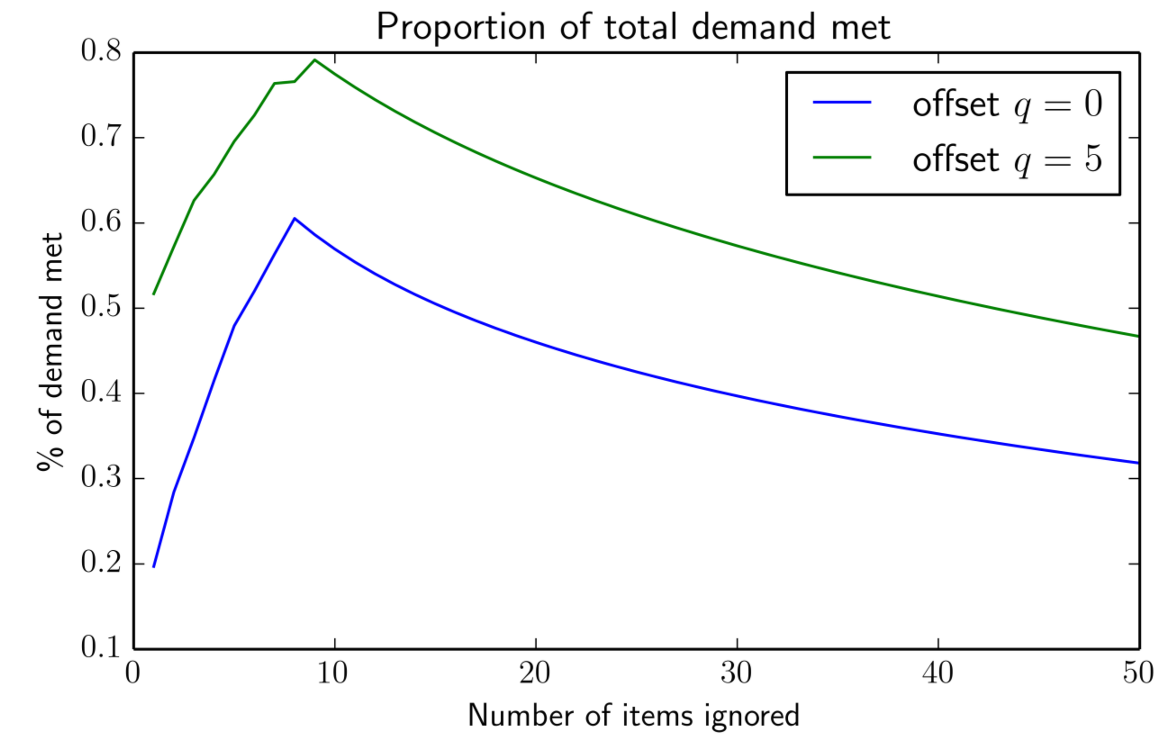
FINDINGS

Maximum Load

The uniform random layout performs very well on even very steep demand distributions. The figure above shows the results for two separate analyses of the uniform random layout (using one and two partitions) compared against the baseline. The expected maximum load is about one third of the baseline.

Addressable Demand

The figure below shows that—despite having a fairly large maximum load—by completely ignoring a few items, we can still serve a large portion of the demand.



FUTURE WORK

- Evaluate the layout on other theoretical and also real-world demand distributions
- Consider the case where approximate demand estimates are known, by extending the process to create a number of copies of each item proportionate to the demand.