# Performance Evaluation of Tape Storage Systems

## Context

Initially designed for media recording, the usage domain of magnetic tapes has broadened over the decades and remains a real competitor to disk storage even for scientific data. The main advantages of this storage medium are a large storage capacity for a reasonable price, better data preservation, better security and energy efficiency. Indeed, it has been estimated that total costs are reduced by an average factor of 6 when archiving data on tape rather than disks [1].

Recent tape cartridges can store up to 20 terabytes of data on a one-kilometer-long physical storage, longitudinally divided into few bands which are each also longitudinally divided into dozens of wraps. Wraps are in turn divided into dozens of tracks. All tracks in a given wrap are read or written simultaneously. A tape is then composed of hundreds of parallel wraps which are logically linked together in a *linear serpentine*.

Thousands of such cartridges are usually stored on the shelves of robotic libraries (as in Figure 1), as books would be stored in an actual library. Then, when data on a given cartridge is not needed, its storage does not induce any power consumption, and it cannot be accessed by intruders. All these advantages of tape storage made it an unavoidable candidate for the storage of the exabytes of data produced at CERN by the Large Hadron Collider experiments [2] or data related to European weather forecast [3].

The huge amount of data stored in such tape libraries, typically hundreds of petabytes, is usually managed by a Mass Storage Management System (*e.g.,* IBM HPSS or HPE DMF) which keeps track of the exact location of the files stored on tapes and answers to users' requests. When a particular file is needed, the tape it is on will be fetched by a robotic arm, brought to a tape drive, and loaded. Then, the reading head of the tape drive is positioned to the beginning of the file to read, or to the first available space to write new data, and the I/O operation eventually occurs.

The main drawback of tape storage is the high latency to access a given file. Mounting a tape into a tape reader requires a delay of about a minute [4].When accesses to multiple files are requested, finding the right ordering of these accesses is thus key to performance. Moving the reading head back and forth along a kilometer long tape has a non-negligible cost and unnecessary movements thus have to be avoided. However, the optimization of tape request

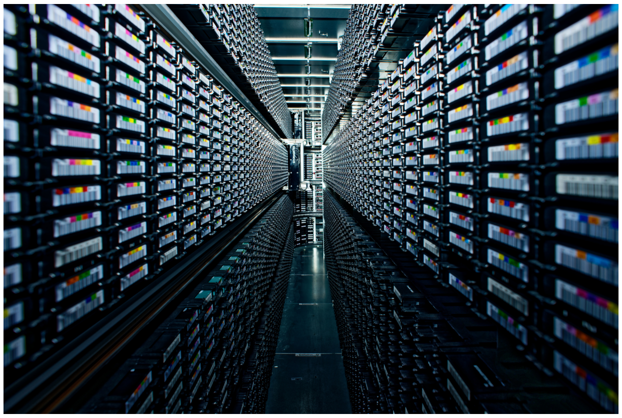

Figure 1: Photography of a tape storage system.

ordering has rarely been studied in the scheduling literature, much less than I/O scheduling on disks.

For instance, minimizing the average service time for several read requests, *i.e.,* the average time at which each request is read, on a single wrap is still an active research question.

## Goal of the project

In a recent work [5], we studied this problem from a theoretical perspective. We restricted to a tape model where a tape is considered as a linear sequence of files which all have to be read from the left to the right, modeling for instance a single wrap.

Using this model, we studied the Linear Tape Scheduling Problem (LTSP), introduced by [6]. In this problem, we wish to design a schedule (i.e., a trajectory of the reading head on the linear tape) to read all the requested files when the reading head is initially positioned on the right of the tape. The input of the problem we considered is thus a list of files that are requested on a given tape, associated with a number of requests for each file. We considered the average service time as a metric, to ensure a fair service among all requests.

In [5], we solved the LTSP problem with an exact algorithm (named **DP**), using a sophisticated dynamic program, thus disproving the NP-hardness conjecture from [6]. We also provided faster suboptimal algorithms and compared the performance of these original algorithms to that of existing algorithms on a dataset built from the recent history of the tape library of the IN2P3 Computing Center. For this evaluation, we developed a Python simulator that implements the tape model and algorithms (both ours and ones from literature), and run them on dataset entries.

The goals of this project are:

1. Read and reproduce the results in [5] as a first step.
2. Propose a new performance-oriented implementation of our simulator [7] in [5]. We expect to use a close-to-system language such as C or C++ to be able to study performance of the algorithmic solutions. Thus, we want to be able to add different LTSP algorithmic solutions in the simulation, and eventually enhanced tape model.
3. Explore algorithmic enhancements of **DP** algorithm. The performance of the dynamic program could be enhanced, for instance using memoization or pruning techniques.
4. Continue the analysis of the logs extracted from the IN2P3 Computing Center's tape storage system. One direction would be to focus on temporal data on tape (reading and moving speed of reading head etc).

## Contact

Valentin Honoré valentin.honore@ensiie.fr
Associate Professor ENSIIE, Evry
Member of the Parallel & Disitributed Systems group, Samovar

The project will be co-supervised with Bertrand Simon, CNRS Researcher at the IN2P3 Computing Center, Villeurbanne, FRANCE

## References

[1] D. Reine and M. Kahn, "Continuing the Search for the Right Mix of Long-term Storage Infrastructure – a TCO Analysis of Disk and Tape Solutions," The Clipper Group, Inc., Tech. Rep. TCG2015006, 2015, [Online, Dec. 2021]www.clipper.com/research/TCG2015006.pdf.

[2] M. C. Davis, V. Bahyl, G. Cancio, E. Cano, J. Leduc, and S. Murray, "CERN Tape Archive – from Development to Production Deployment," in *Proceedings of the 23rd International Conference on Computing in High Energy and Nuclear Physics*, ser. EPJ Web of Conferences, vol. 214. EDP Sciences, 2019, p. 04015.

[3] M. Mäsker, L. Nagel, T. Süß, A. Brinkmann, and L. Sorth, "Simulation and Performance Analysis of the ECMWF Tape Library System," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, Salt Lake city, UT, Nov. 2016, pp. 252–263.

[4] E. Cano, V. Bahyl, C. Caffy, G. Cancio, M. Davis, O. Keeble, V. Kotlyar, J. Leduc, and S. Murray, "Cern tape archive: a distributed, reliable and

scalable scheduling system," in *EPJ Web of Conferences*, vol. 251. EDP Sciences, 2021, p. 02037.

[5] V. Honoré, B. Simon, and F. Suter, "An Exact Algorithm for the Linear Tape Scheduling Problem," in *The 32nd International Conference on Automated Planning and Scheduling (ICAPS 2022)*, vol. Proceedings of the Thirty-Second International Conference on Automated Planning and Scheduling, no. 32, Singapore, Singapore, Jun. 2022, pp. 151–159. [Online]. Available: https://cnrs.hal.science/hal-03482022

[6] C. Cardonha and L. C. V. Real, "Theoretical and practical aspects of the linear tape scheduling problem," *CoRR*, vol. abs/1810.09005v1, 2018. [Online]. Available: http://arxiv.org/abs/1810.09005v1

[7] V. HONORE, B. Simon, and F. Suter, "Experiment Artifact for the "Algorithmic Studies for the Linear Tape Scheduling Problem" paper," 12 2022. [Online]. Available: https://figshare.com/articles/online_resource/Experiment_Artifact_for_the_Algorithmic_Studies_for_the_Linear_Tape_Scheduling_Problem_paper/21758225