Abstract

As the need for data explodes with the passage of time and the increase of computing power, data storage becomes more and more important. Distributed storage, as distributed computing before it, is coming of age as a good solution to make systems highly available, i.e., highly scalable, reliable and efficient. The focus of this thesis is how to achieve data reliability and efficiency in distributed storage systems.

This thesis consists of two parts. The first part deals with the *reliability* of distributed storage systems. Reliability is achieved by computationally efficient MDS array codes that eliminate single points of failure in the systems, thus providing more reliability and flexibility to the systems. Such codes can be used as general MDS error-correcting codes. They are particularly suitable for use in distributed storage systems. The second part deals with the *efficiency* of distributed storage systems. Methods are proposed to improve the performance of data server and storage systems significantly through the proper use of data redundancy. These methods are based on error-correcting codes, particularly the MDS array codes developed in the first part.

Two new classes of MDS array codes are presented: the X-Code and the B-Code. The encoding operations of both codes are optimal, i.e., their update complexity achieves the theoretical lower bound. They distribute parity bits over all columns rather than concentrating them on some parity columns. As with other array codes, the error model for both codes is that errors or erasures are columns of the array, i.e., if at least one bit of a column is an error or erasure, then the whole column is considered to be an error or erasure. Both codes are of distance 3, i.e., they can either: correct two erasures, detect two errors or correct one error. In addition to encoding algorithms, efficient decoding algorithms are proposed, both for erasure-correcting and for error-correcting. In fact, the erasure-correcting algorithms are also optimal in terms of computation complexity.

The X-Code has a very simple geometrical structure: the parity bits are constructed along two groups of parallel *parity lines* of slopes 1 and -1. This is the origin of the name X-Code. This simple geometrical structure allows simple erasure-decoding and error-decoding algorithms, using only XORs and vector cyclic-shift operations.

The significance of the B-Code not only includes all its optimality properties: MDS, optimal encoding and optimal decoding, but also its relation with a 3-decade old graph theory problem. It is proven in this thesis that constructing a B-Code of *odd* length is exactly equivalent to constructing a *perfect one-factorization* (or P1F) of a complete graph. Constructing a P1F of an *arbitrary* complete graph has remained a conjecture since the early 1960's. Though the P1F conjecture remains unsolved, the B-Code as the first real application of the P1F problem will hopefully spur more research on it. It is also conjectured in this thesis that constructing a B-Code of *any* length, even or odd, is equivalent to constructing a P1F of a complete graph. An efficient error-correcting algorithm for the B-Code is also presented, which is based on the relations between the B-Code and its dual. The algorithm might give a hint of how to develop efficient decoding algorithms for other codes.

While it is intuitive that redundancy can bring reliability to a system, this thesis gives another direction: using redundancy actively to improve performance (efficiency) of distributed data systems. The results in this direction are both theoretical and experimental. System models are extracted from experiments in real practical systems; analytical results are derived using these and are then fed back to experiments for verification.

In this thesis, a novel *deterministic* voting scheme that uses error-correcting codes is proposed. The voting scheme generalizes all known simple deterministic voting algorithms. It can be tuned to various application environments with different error rates to drastically reduce average *communication complexity*, i.e., the amount of information that must be transmitted in order to get correct voting results.

Two problems are identified to improve the performance of general data server systems, namely the *data distribution* problem and the *data acquisition* problem. Solutions to these are proposed, as are general analytical results on performance of (n, k) systems. A simple service time model of a practical disk-based distributed server system is given. This model, which is based on experimental results, is a starting point for data distribution and data acquisition schemes. These results, both experimental and analytical, can be further used for more sophisticated scheduling schemes to optimize or improve the performance of data server systems that serve multiple clients simultaneously.

Finally, some research problems related to storage systems are proposed as future directions.