Flat XOR-based erasure codes in storage systems: Constructions, efficient recovery, and tradeoffs

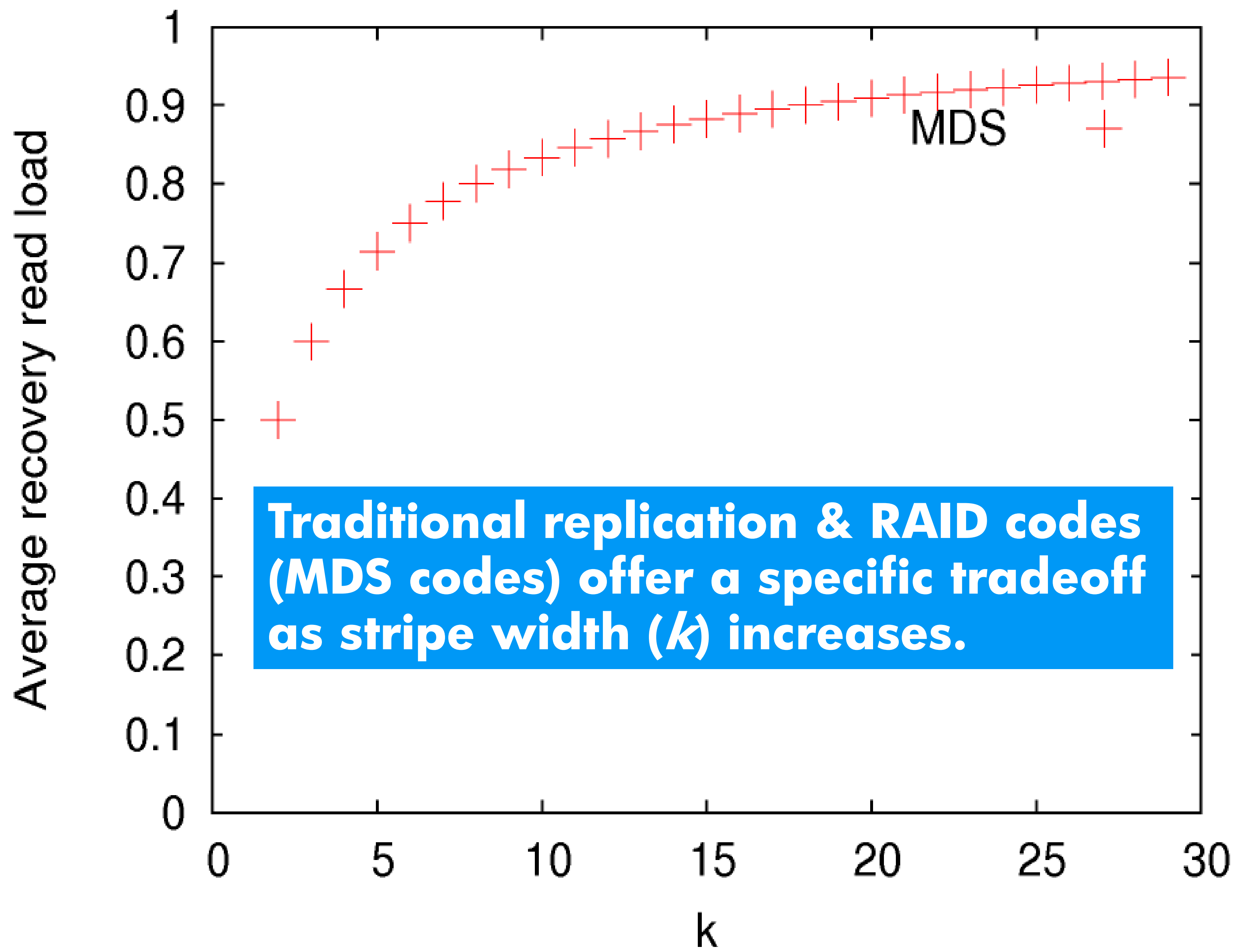
Kevin M. Greenan
ParaScale

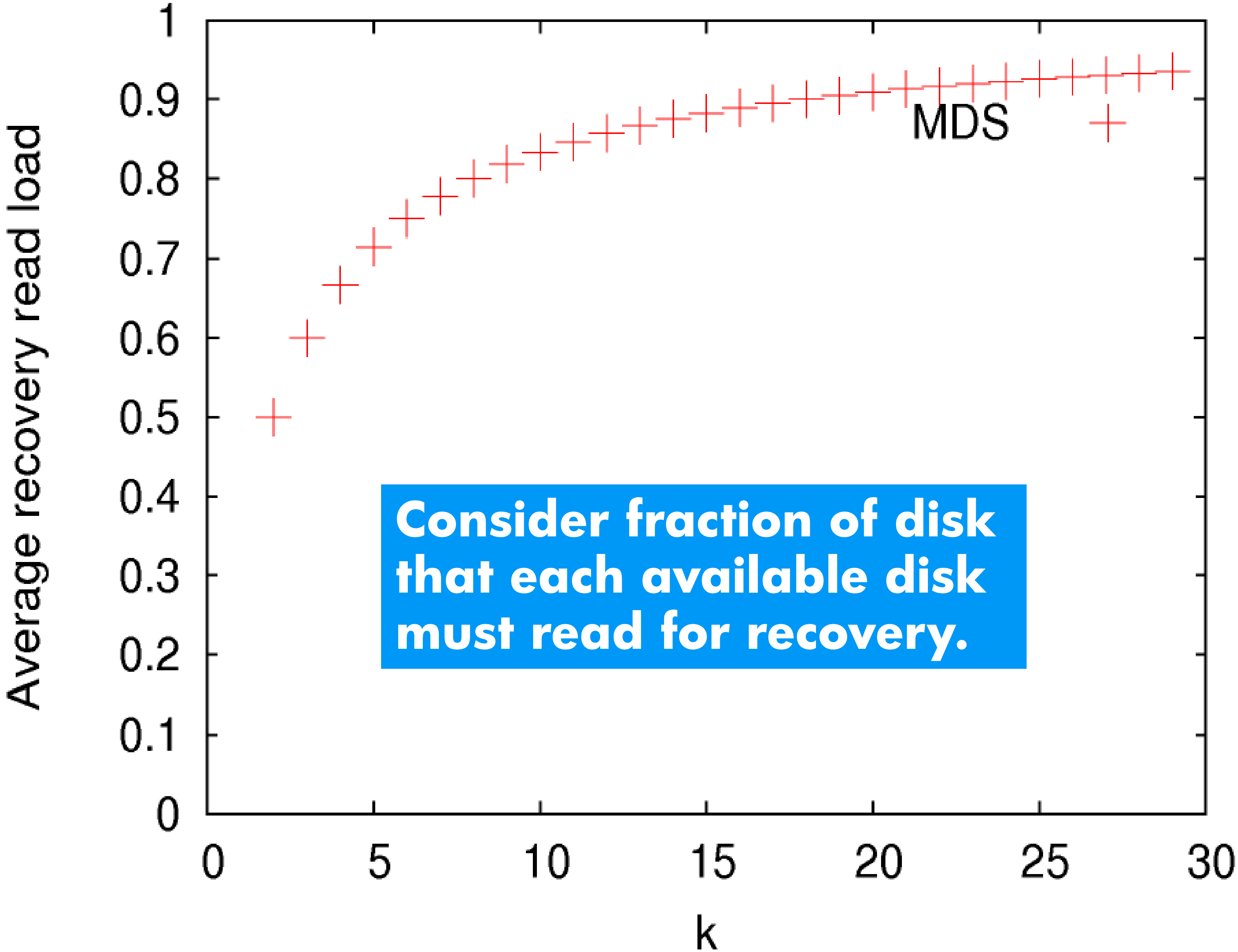
Xiaozhou Li
HP Labs

Jay J. Wylie
HP Labs

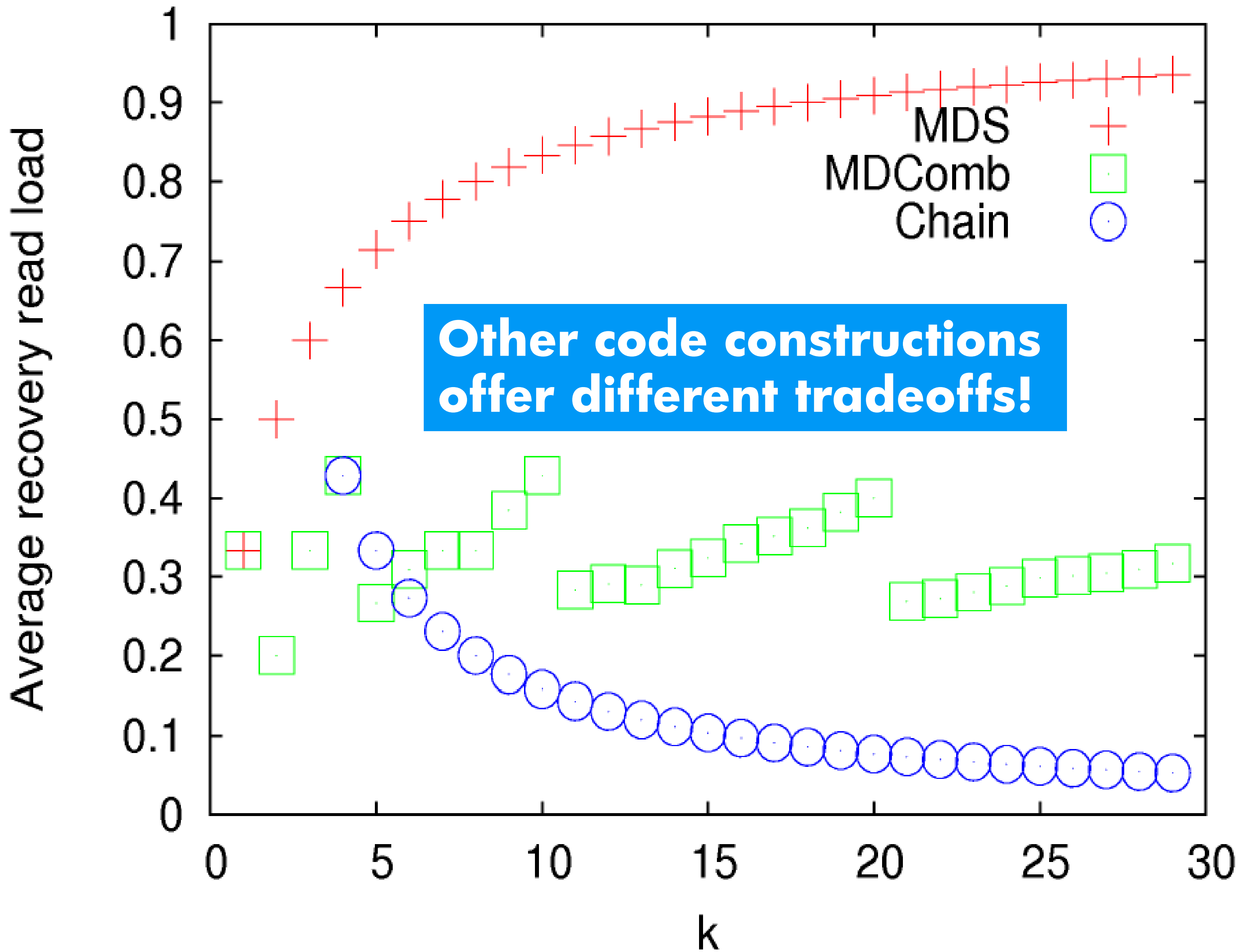
MSST Research Track
May 7, 2010

Contributions


Average recovery read load
k
MDS
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30
Traditional replication & RAID codes (MDS codes) offer a specific tradeoff as stripe width (k) increases.



Average recovery read load
k
MDS
Consider fraction of disk that each available disk must read for recovery.
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30


Other code constructions offer different tradeoffs!
MDS
MDComb
Chain
Average recovery read load
k
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30

Contributions

- **Efficient recovery of erasure-coded data**
- New erasure codes (flat XOR-codes)
  - MD Combination codes
  - Stepped Combination codes
  - Flattened parity-check array codes
- Recovery equations & schedules for XOR-codes
- Analytic comparison
  - Apples-to-apples analysis of many codes
  - For key properties of erasure-coded storage

Background

Replication

Two-fold replication

Three-fold replication

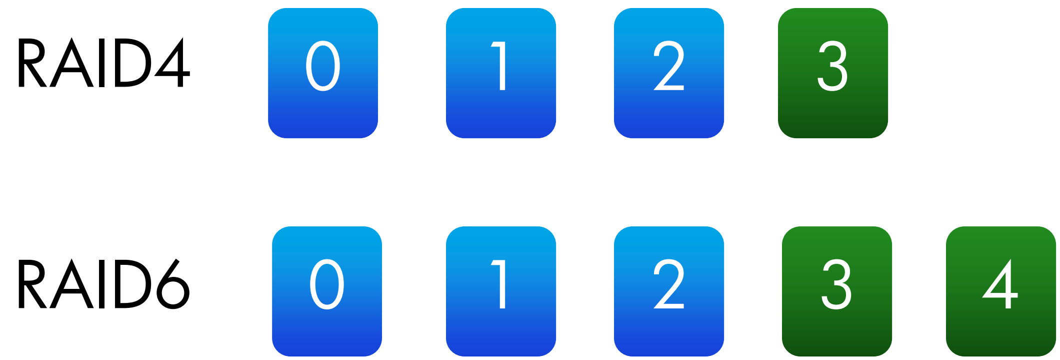
Four-fold replication

– Blue fragments are “data”

– Green fragments are “parity”

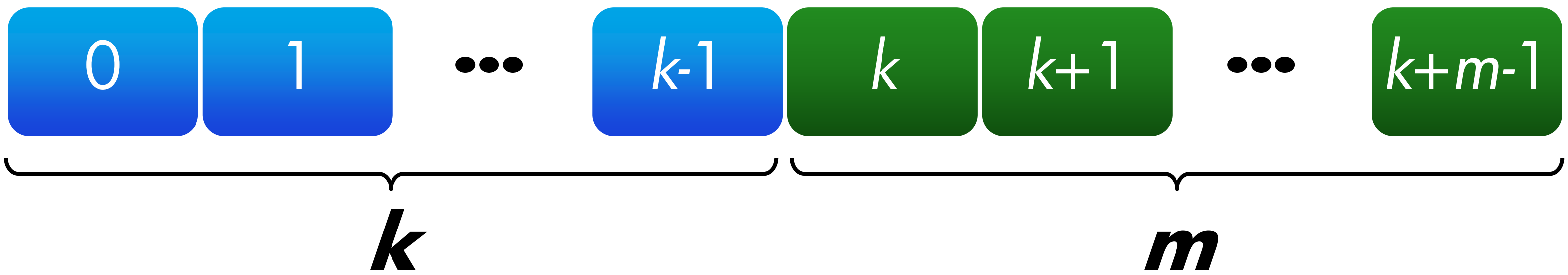
– For replication, “parity” and “data” are the same…

RAID



- Ignore rotation (e.g., RAID5)
- Ignore details of how “parity” is calculated

MDS (Maximally Distance Separable) codes



– Replication, RAID4, and RAID6 are all MDS

– MDS codes are optimally space-efficient

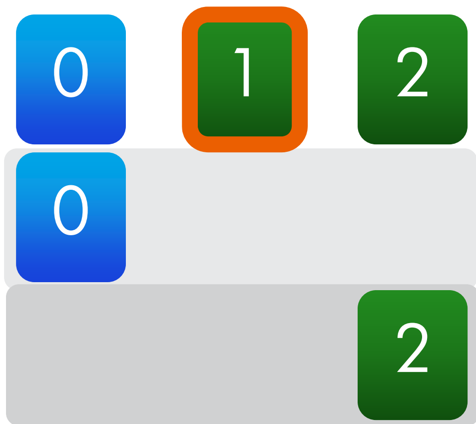
– I.e., each parity disk increases fault tolerance

– Notation: ***k*** data and ***m*** parity fragments

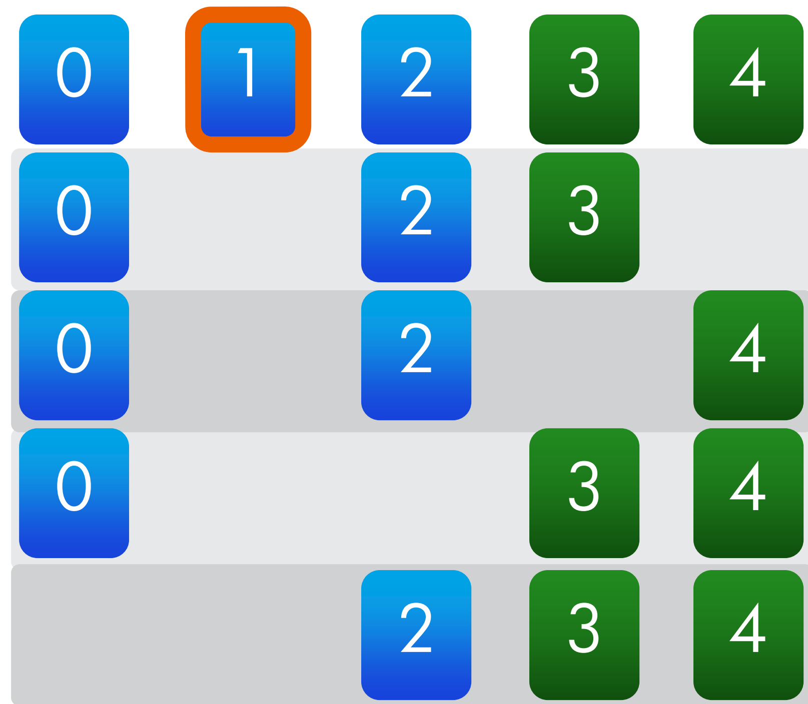
– An MDS code is ***m*** disk fault tolerant (DFT)

Recovery equations for MDS codes

$k = 1,\ m = 2$



$k = 3,\ m = 2$

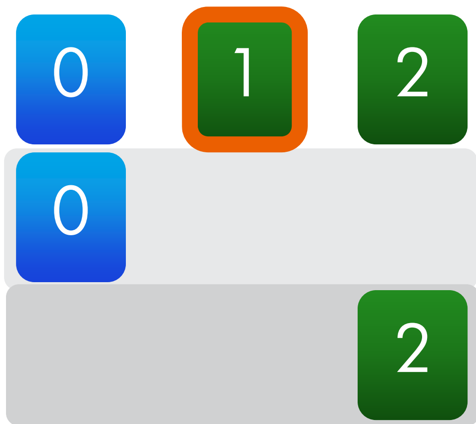


– Any k fragments can recover a failed fragment

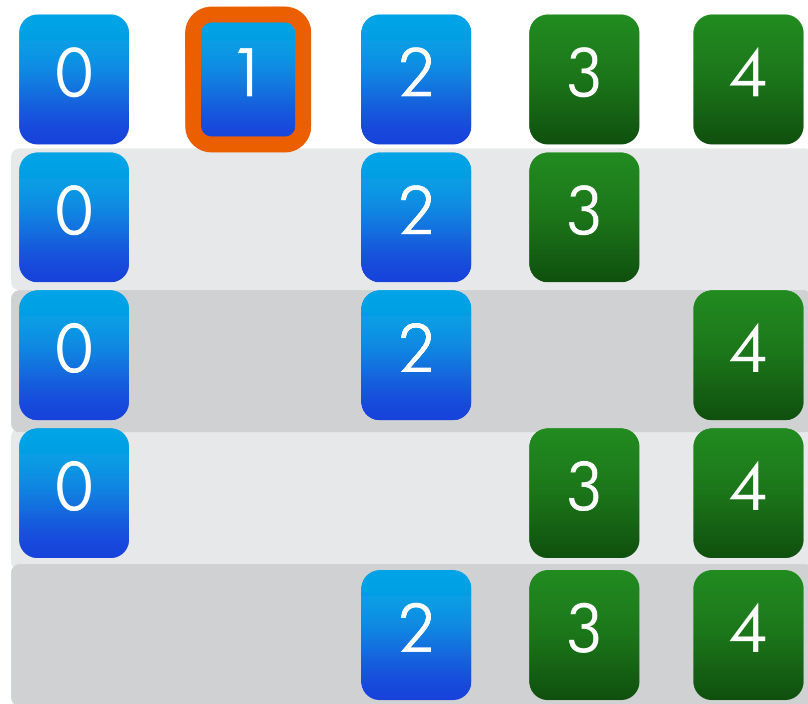
– E.g., consider if fragment 1 fails

Recovery equations for MDS codes

3-fold replication



RAID6

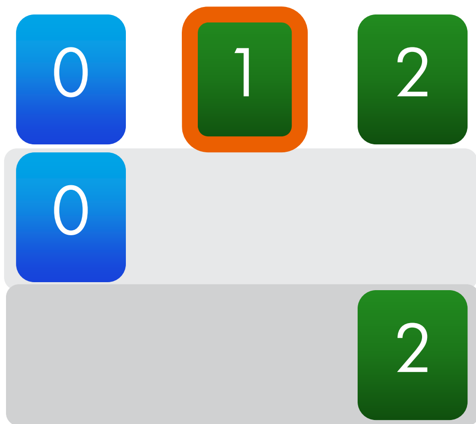


– Any *k* fragments can recover a failed fragment

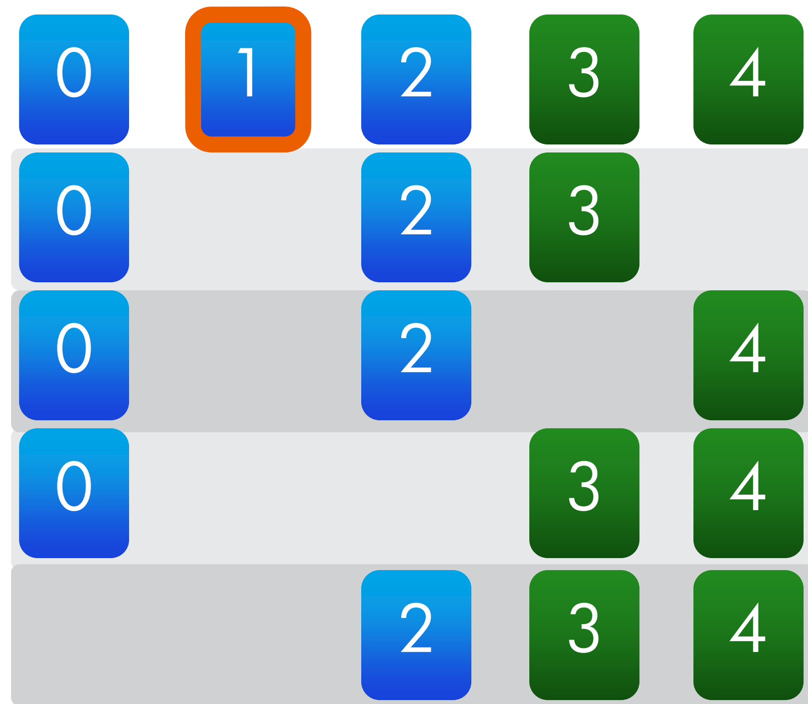
– E.g., consider if fragment 1 fails

Recovery schedules for MDS codes

3-fold replication



RAID6

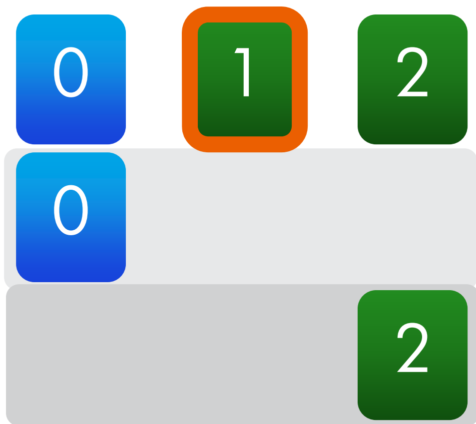


– Use multiple recovery equations simultaneously

– Reduces read recovery load on available disks

Recovery schedules for MDS codes

3-fold replication

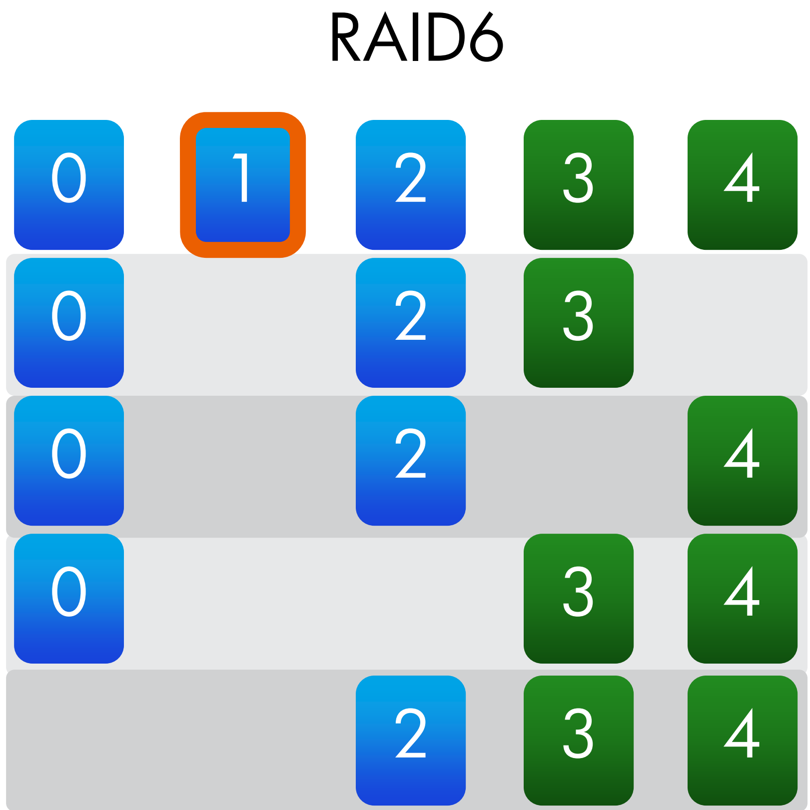


If disk one fails, then each of disk zero and disk two only need to read half the stripes.

– Use multiple recovery equations simultaneously

– Reduces read recovery load on available disks

Recovery schedules for MDS codes



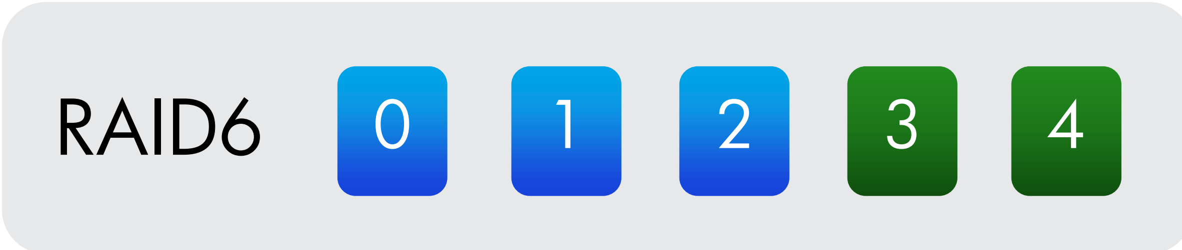
For this RAID6, each available disk must read ¾ of the stripes.

– Use multiple recovery equations simultaneously

– Reduces read recovery load on available disks

Flat XOR-codes

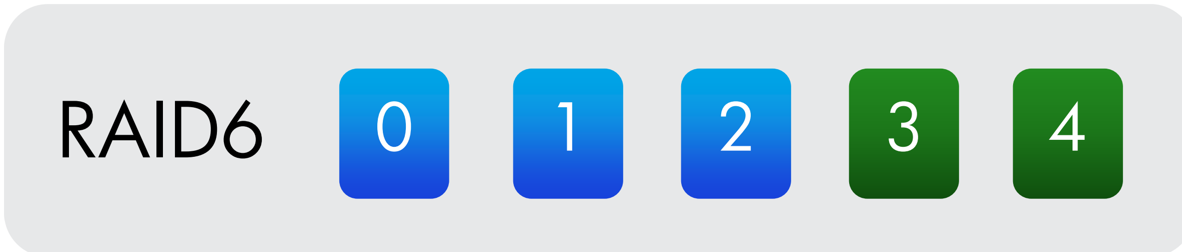
Flat code vs Array code



Flat code

Parity check array code

Flat code vs Array code

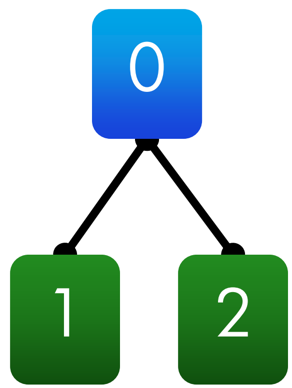


Flat code

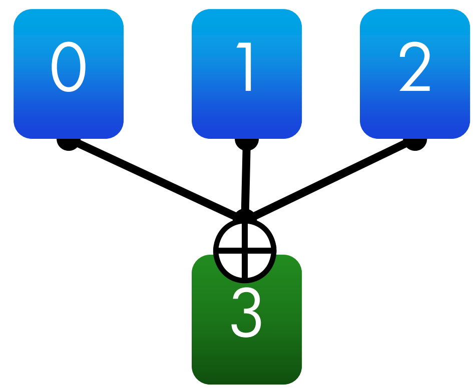
Parity check array code

Flat XOR-based erasure codes

Three-fold replication

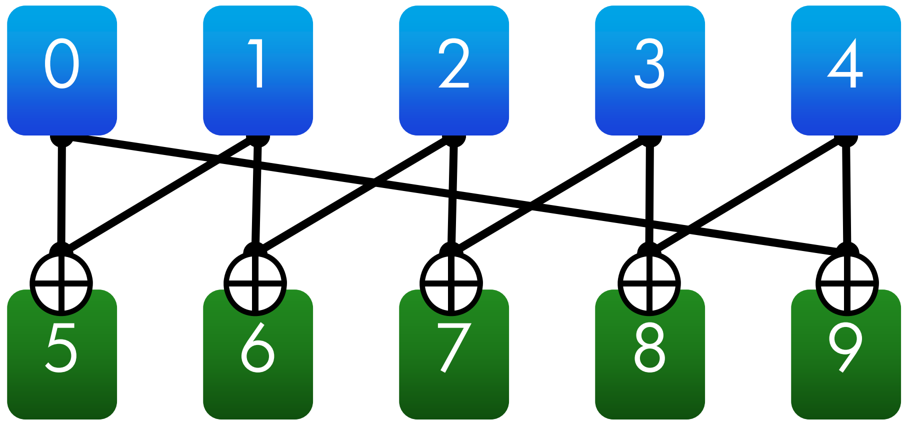


RAID4



- Each parity is XOR of a subset of data fragments
- Can be illustrated with a *Tanner graph*
- Replication and RAID4 are MDS flat XOR-codes
- Other flat XOR-code constructions **not** MDS

Chain codes

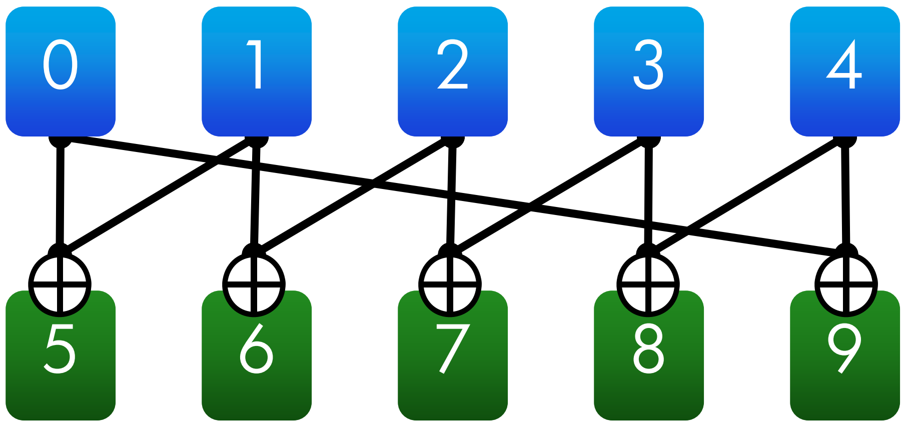


– Two- and three-disk fault tolerant constructions

– Example two-disk fault tolerant Chain code

- Each parity XOR of two subsequent data fragments
- Non-MDS: $k = m = 5$

Chain codes

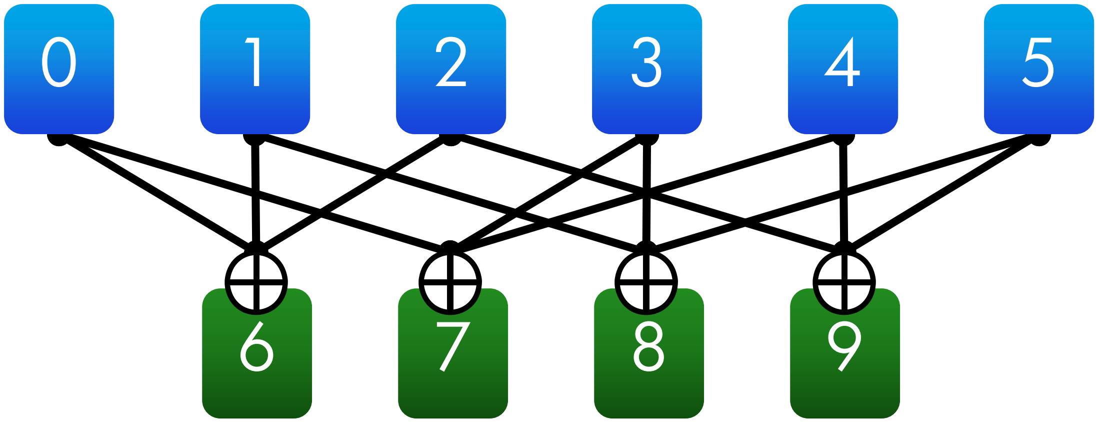


– *Chain code* is variant of prior constructions

– Related constructions

- Wilner/LSI codes [patent 6,327,627, 2001]
- Weaver(n,2,2) codes [Hafner FAST, 2005]
- SSPiRAL codes [Amer et al. SNAPI, 2007]

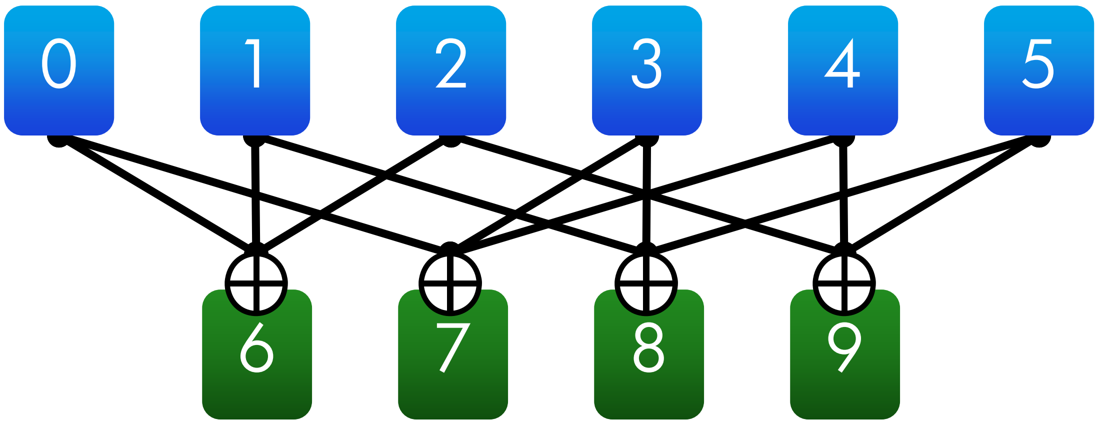
Minimum Distance (MD) Combination codes



– Lets construct a 2 DFT MD Combination code
  - Each data must connect to 2 parities
  - Every data must connect to **distinct** set of parities

– How large a code can we construct with 4 parities?
  - If $m = 4$, then there are 6 combinations of 2 parity
  - I.e., $k \leq$ (4 choose 2) = 6

Minimum Distance (MD) Combination codes



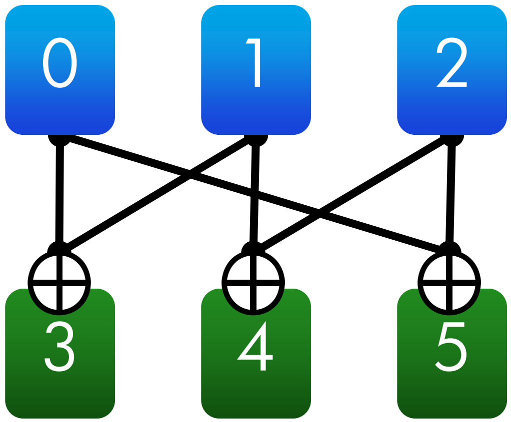
– More details in the paper
  - 2 & 3 DFT constructions
  - Bounds on *k* relative to *m*
  - Proof that constructions achieve desired DFT

Even more details in the paper…

– Stepped Combination code
  - Extension of MD Combination code
  - 2 & 3 DFT variants, bounds on *k* & *m*, proof
– Flattening
  - Converts parity-check array codes into flat XOR-codes
  - E.g., SPC, RDP, EVENODD, STAR
– Related work
  - Other non-MDS code constructions
  - Other recovery techniques

Efficient recovery

Efficient recovery example

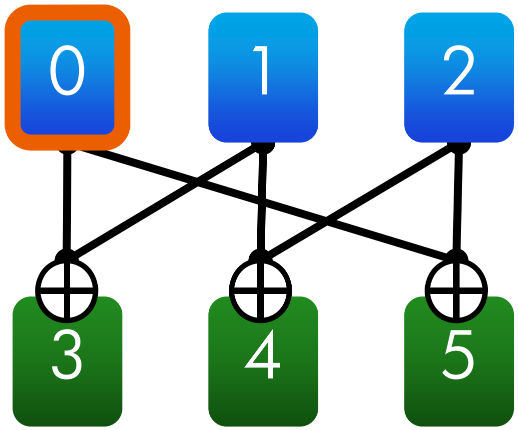


– 2 DFT flat XOR-code

– $k = m = 3$

– Chain and MD Combination codes equivalent

Recovery equation example

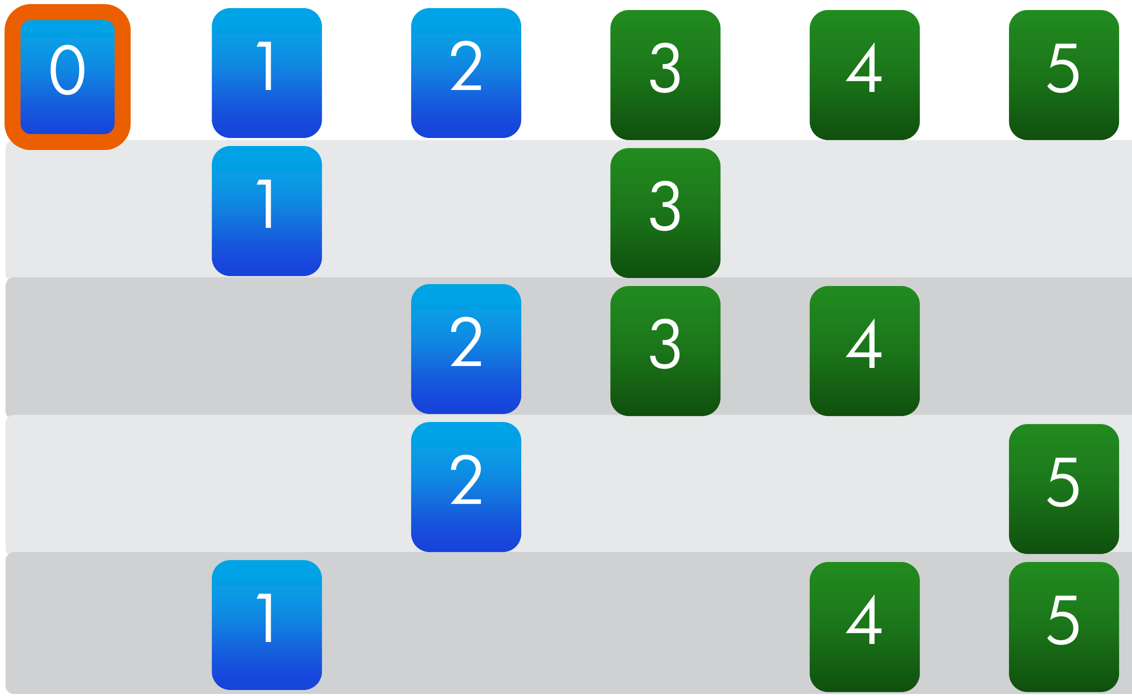


$$
\begin{aligned}
0 = \; & 1 \oplus 3 \\
& 2 \oplus 3 \oplus 4 \\
& 2 \oplus 5 \\
& 1 \oplus 4 \oplus 5
\end{aligned}
$$

– Recovery equations for fragment zero?

– Some recovery equations less than *k* in size!

Chain code recovery schedule example I



– Use all four recovery equations simultaneously
– Each available disk reads 0.5 disk's data
– A total of 2.5 disk's data is read to recover

Chain code recovery schedule example II

– Use two shortest recovery equations simultaneously
– Four of the five available disks read 0.5 disk's data
– A total of 2.0 disk's data is read to recover

Efficient recovery of flat XOR-codes

– Short recovery equations
  - Recovery equations smaller than *k*
  - Read less total data to recover than MDS
– Recovery schedules distribute read load
  - Each available disk reads less data to recover than MDS

More details in paper…

- Recovery equations algorithm for flat XOR-codes
- Algorithms to determine recovery schedules
- Discuss rotated codes (e.g., RAID5)
- Complements prior techniques
  - Parity declustering & chained declustering
  - Distributed sparing

Analytic comparison

Analytic comparison

– Focus on 3-DFT codes

– Analyze following codes

- MDS
- MD-Combination (MDComb)
- Chain

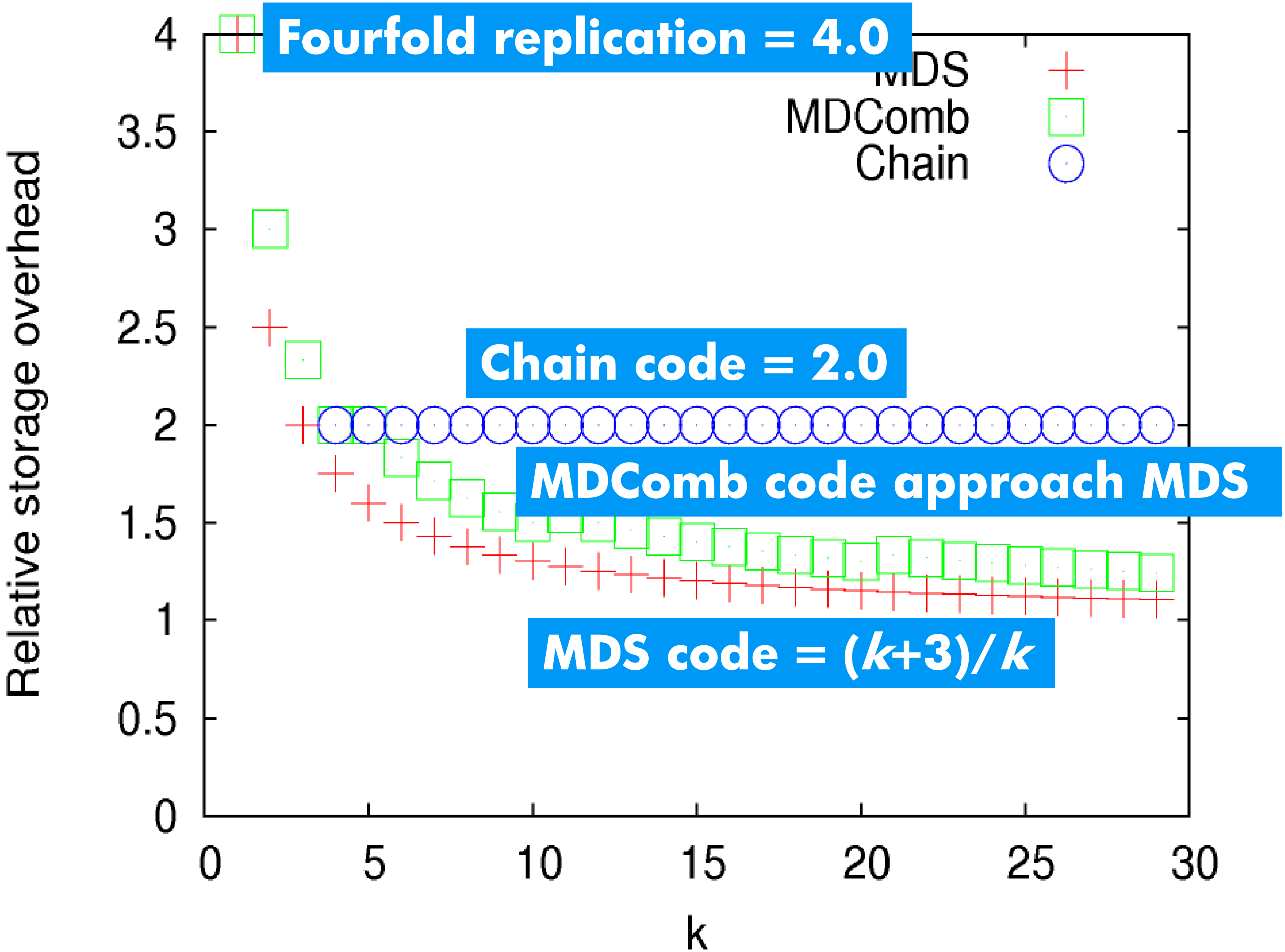
– Consider stripes with *k* from 1 to 30

Relative storage overhead

– Storage overhead relative to one replica

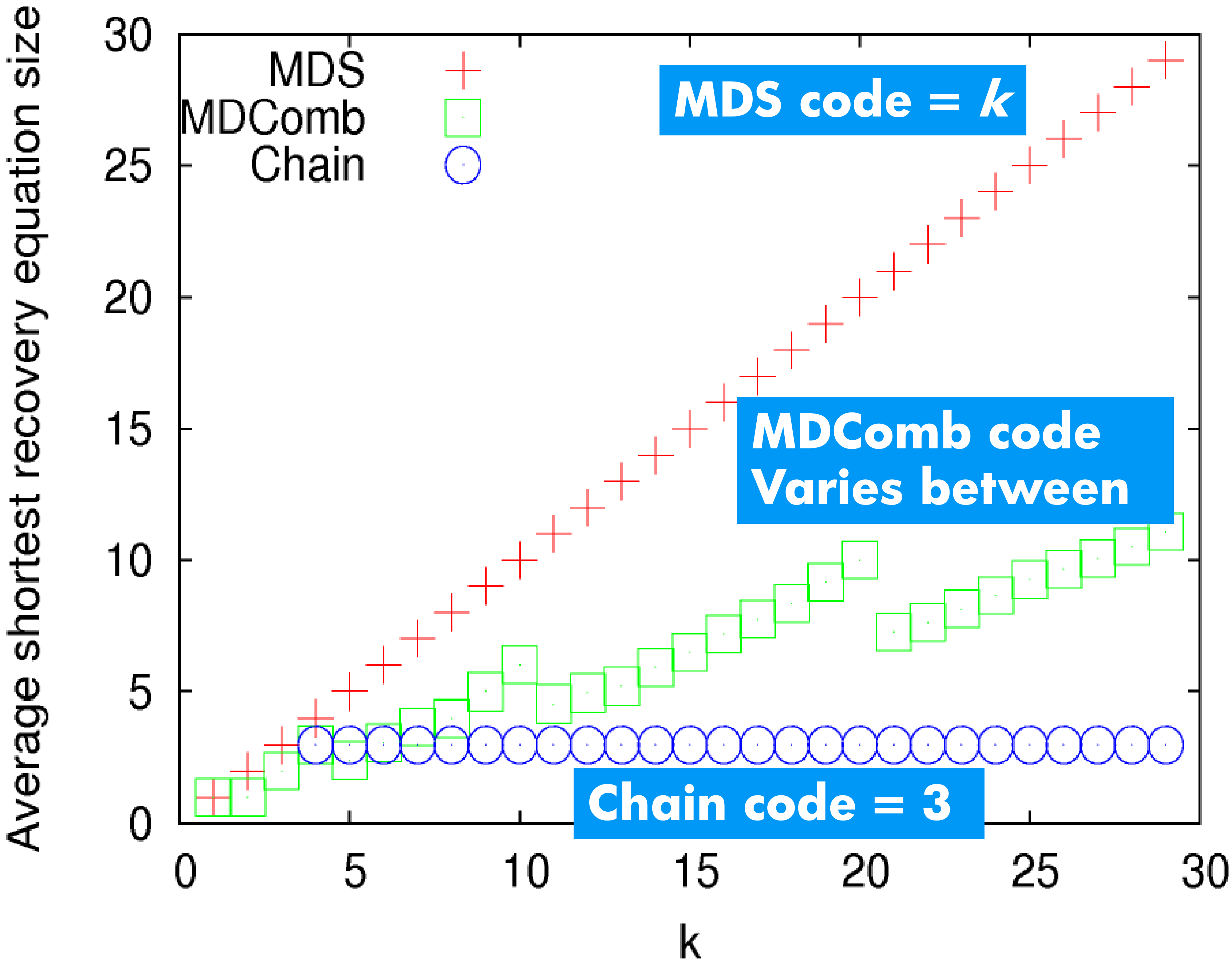
– MDS codes: $(k+m)/k$

– Non-MDS have greater overhead than MDS codes


Fourfold replication = 4.0
MDS
MDComb
Chain
Chain code = 2.0
MDComb code approach MDS
MDS code = (k+3)/k
Relative storage overhead
k
0
0.5
1
1.5
2
2.5
3
3.5
4
0
5
10
15
20
25
30

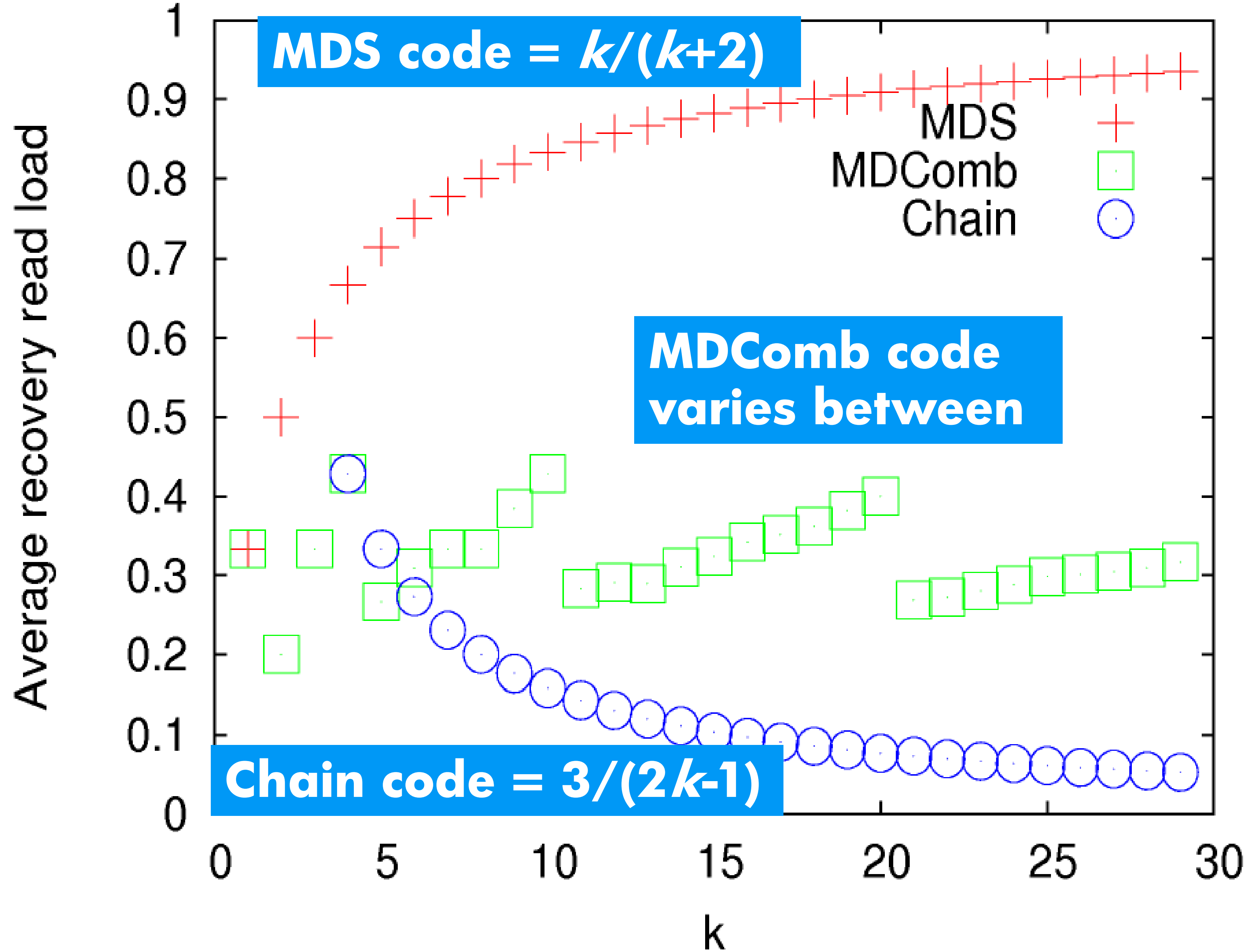
Average shortest recovery equation size

– Determine shortest recovery equation per fragment
– Average size over all fragments


MDS code = k
MDComb code
Varies between
Chain code = 3
MDS
MDComb
Chain
Average shortest recovery equation size
k
0
5
10
15
20
25
30

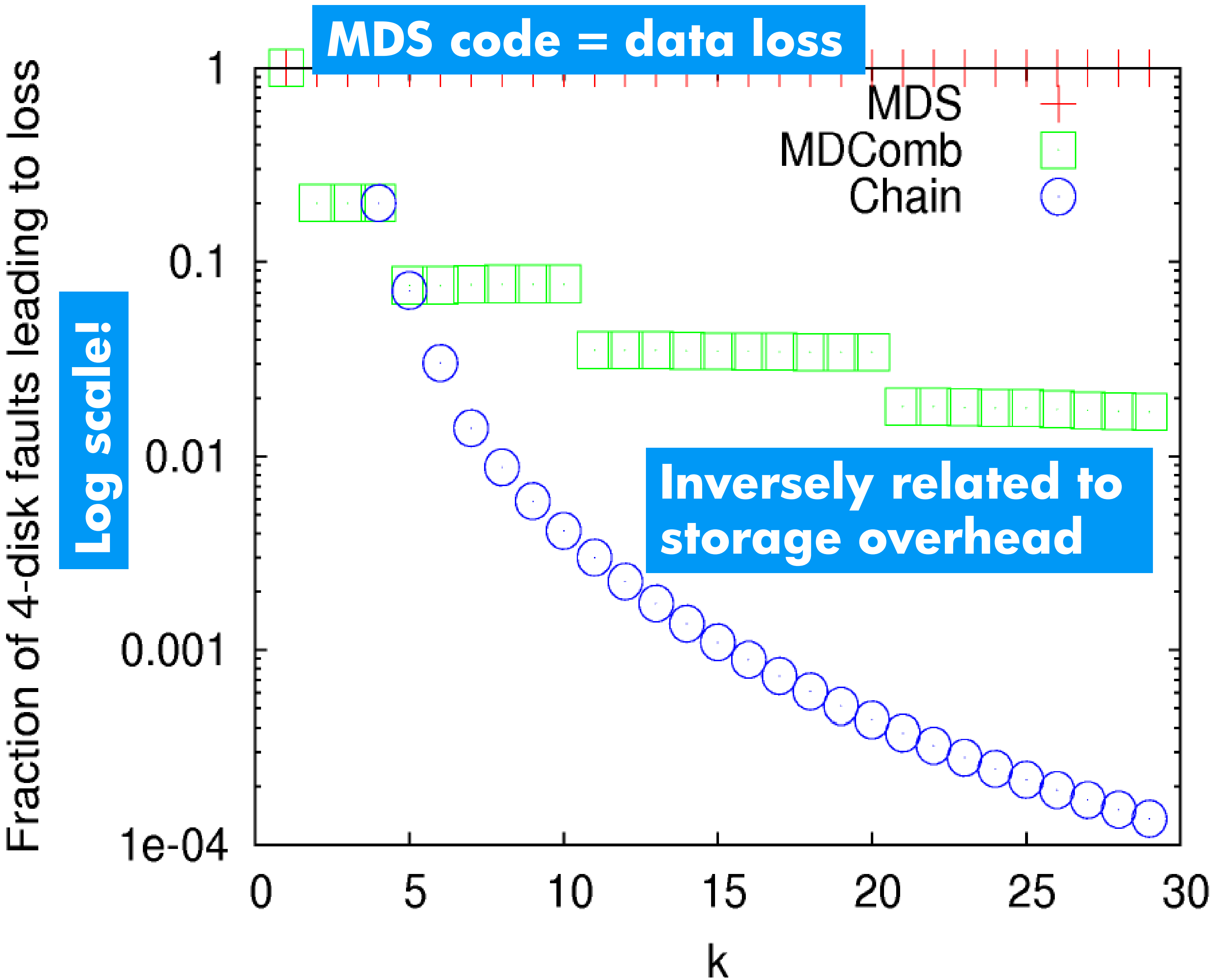
Average recovery read load

– Optimal recovery schedule per lost fragment
– Average over all fragments


MDS code = k/(k+2)
MDComb code
varies between
Chain code = 3/(2k-1)
MDS
MDComb
Chain
Average recovery read load
k
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30

Fraction of 4-disk faults leading to loss

– Since flat XOR-codes are non-MDS
– They may tolerate specific sets of 4 disk failures!
– (Or, even more than 4 disk failures.)


MDS code = data loss
Log scale!
Inversely related to storage overhead
Fraction of 4-disk faults leading to loss
1
0.1
0.01
0.001
1e-04
0
5
10
15
20
25
30
k
MDS
MDComb
Chain

Analytic comparison at $k=15$

	Storage overhead	Avg. short rec. eq. size	Avg. read rec. load	4-disk fault data loss
MDS	1.2	15.0	0.88	100.0%
MDComb	1.4	6.5	0.32	3.5%
Chain	2.0	3.0	0.10	1.1%

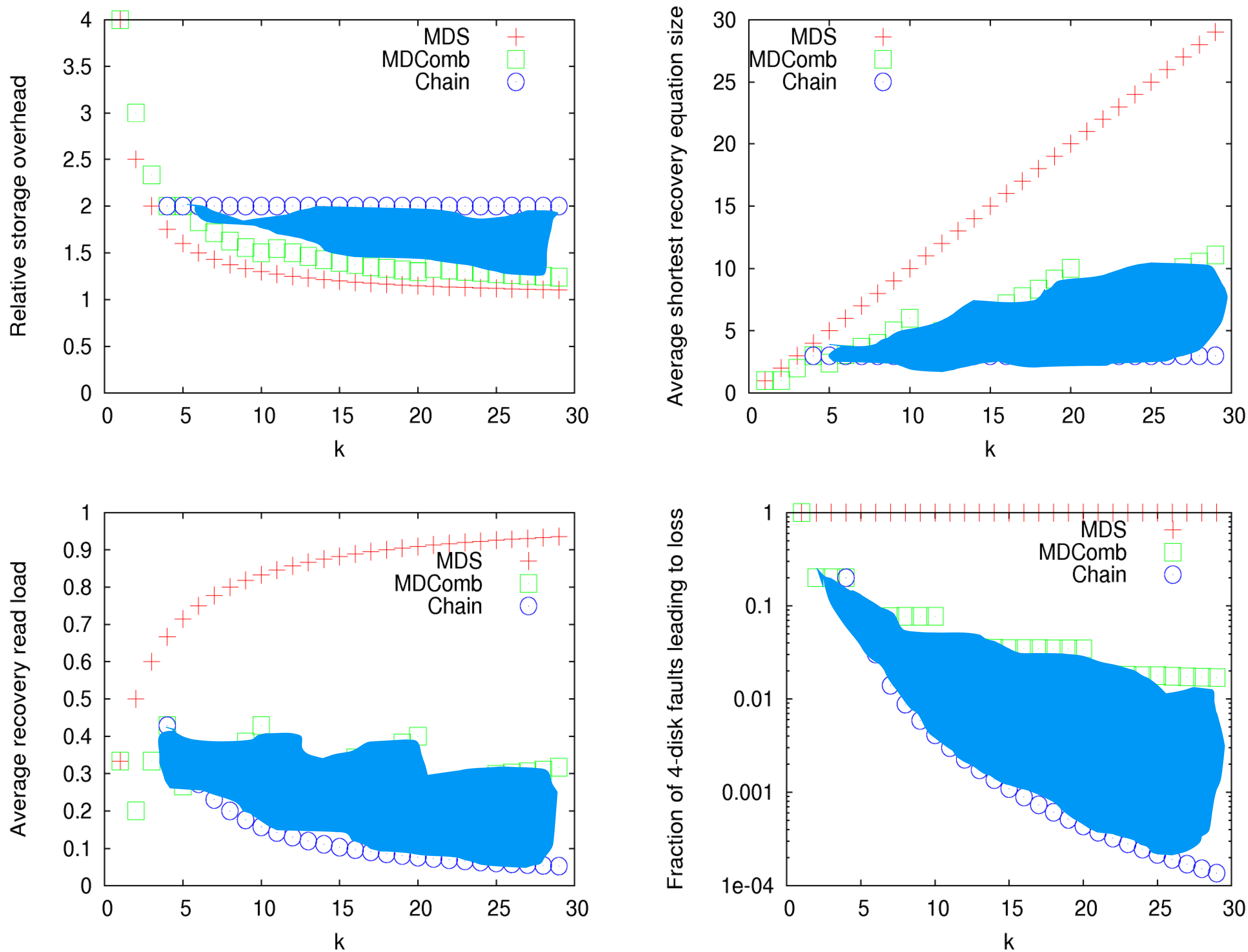
As storage overhead increases, other metrics improve

More analysis in the paper

- More codes
  - 2DFT codes
  - Stepped-Combination
  - Flattened parity-check array codes
- More metrics
  - Discussion of encode/decode performance
  - Analyze small write costs

Summary

- Novel flat XOR-code constructions
  - MD-Combination codes
  - Stepped Combination codes
- Efficient recovery
  - Recovery equations
  - Recovery schedules
- Analytic comparison
  - Storage overhead, small writes, read recovery load, fault tolerance
  - Believe Chain & Comb codes delimit XOR-code tradeoff space


Relative storage overhead
k
MDS
MDComb
Chain
Average shortest recovery equation size
k
MDS
MDComb
Chain
Average recovery read load
k
MDS
MDComb
Chain
Fraction of 4-disk faults leading to loss
k
MDS
MDComb
Chain

Q&A