Distribution Sorting with Multiple Disks

Jeff Vitter

Department of Computer Science
Center for Geometric & Biological Computing
Duke University

EEF Summer School on Massive Data Sets

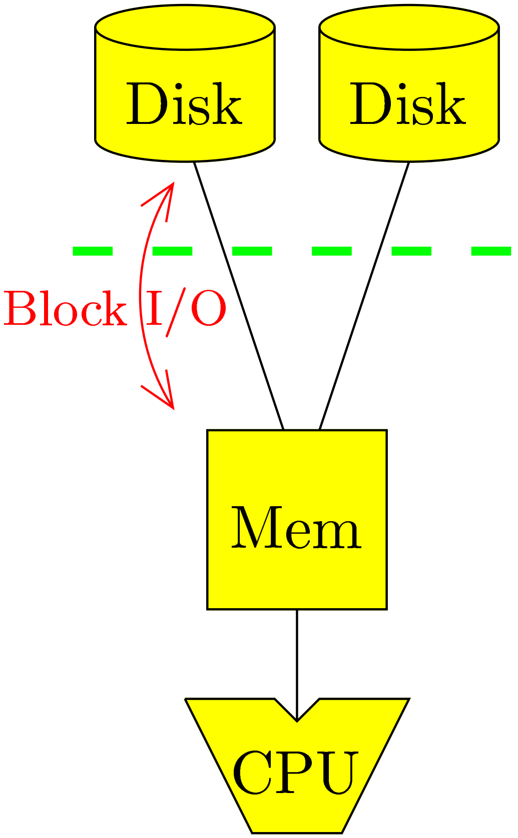
Review of Distribution Sort?

S-way Distribution Sort:

- ☆ 1. If the input stream (bucket) fits into memory, sort it and quit;
- ☆ 2. Otherwise
  - [Splitter Selection Phase] Choose $S - 1$ splitters.
  - [Distribution Phase] Read the input and distribute data into buckets as determined by the splitters.
  - Sort each bucket recursively.

Parallel Disk Model

[Vitter & Shriver 90, 94]



N = problem data size.

M = size of internal memory.

B = size of disk block.

D = number of independent disks.

Distribution sort requires a (double) buffer in internal memory for each bucket.

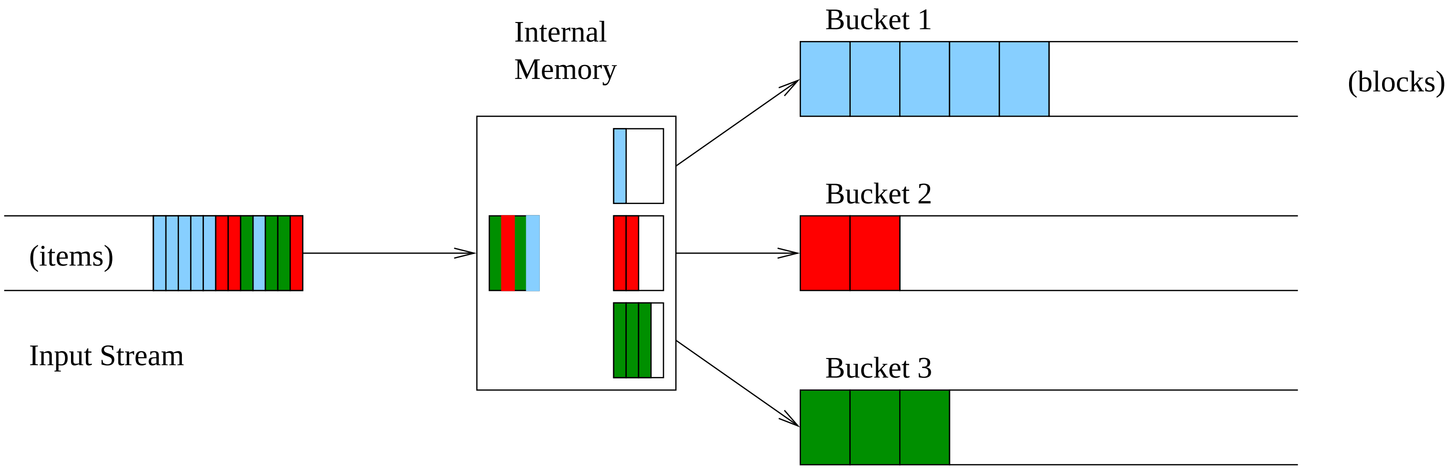
$\Longrightarrow$ Optimal choice of S is $(M/B)^{\Theta(1)}$

If each pass can be done in $O(N/DB)$ I/Os

$$\Longrightarrow \quad \Theta\left(\frac{N}{DB}\log_{M/B}\frac{N}{B}\right) \text{ I/Os total.}$$

Distribution Paradigm

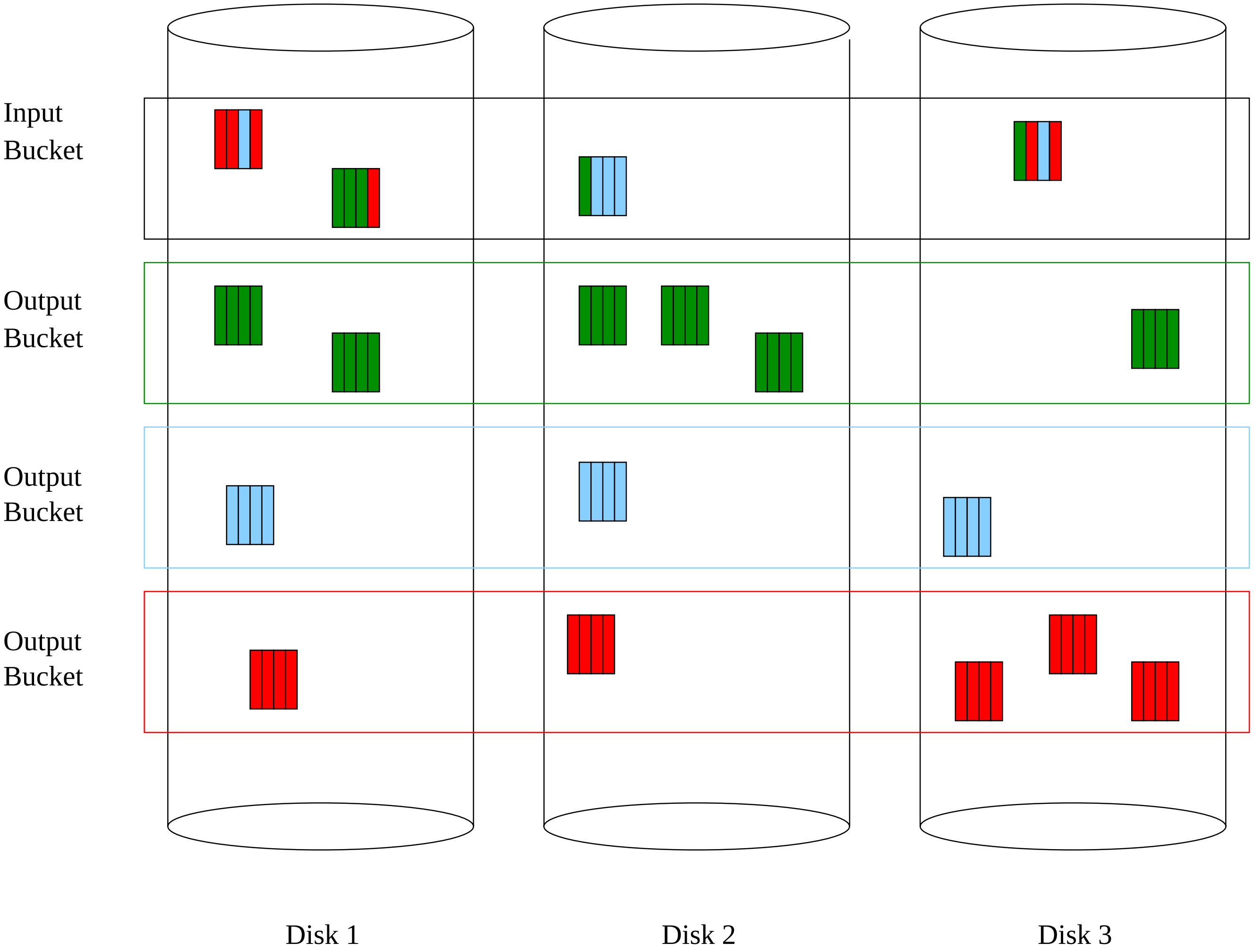
Example: $D = 3$ disks, $S = 3$ buckets:



Data streams through internal memory and is partitioned online.

Challenge: Each bucket must be distributed among the disks in an online manner. How can we prevent write bottlenecks at the disks? That is, how should we lay out each bucket on the disks?

What is the Challenge ?



- ⭐ Read D blocks (one block per disk) in each input operation.
  Write D blocks (one block per disk) in each write operation.
- ⭐ Buckets fill at different rates (no problem if only one disk).

Gilbreath Principle

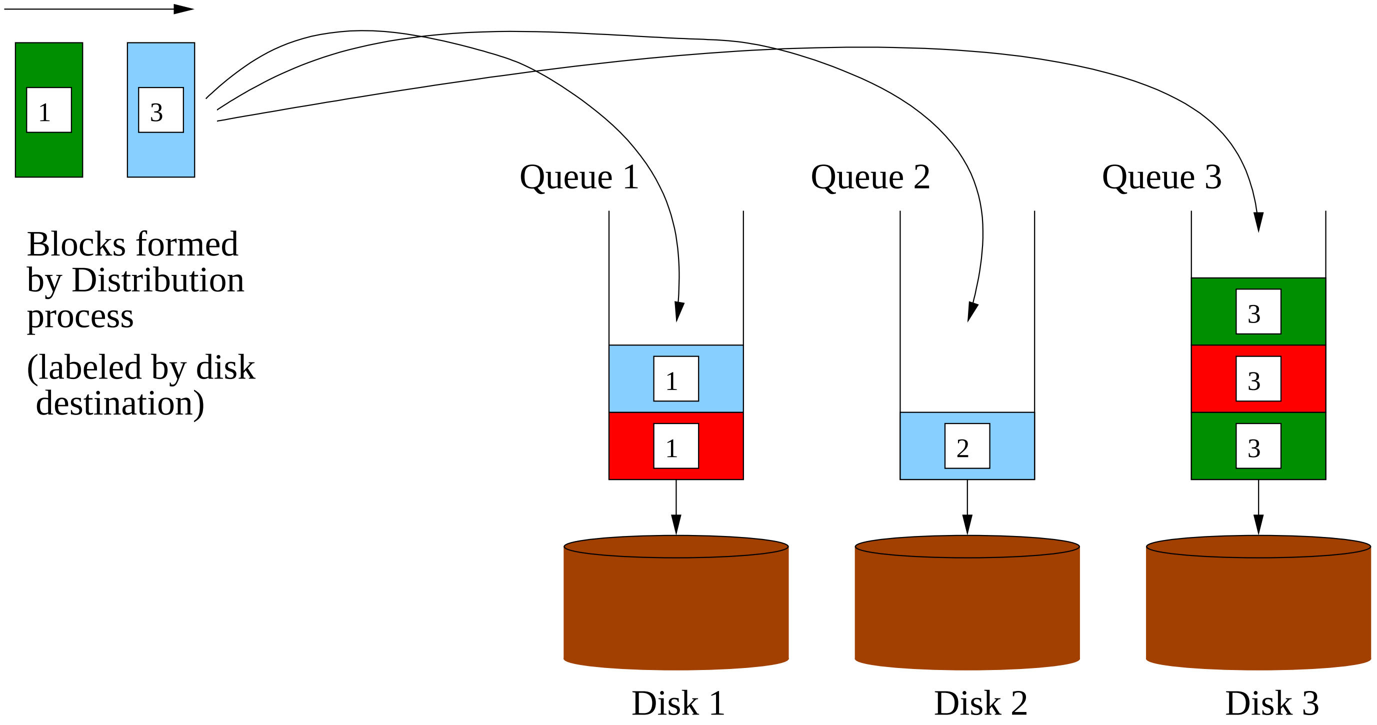
Writing is also no problem if we have only two buckets (streams).

- ☆ We can achieve perfect balance for writing two buckets:

	Disk 1	Disk 2	Disk 3	Disk 4
Bucket 1:	A	B	C	D
	E	F	G	H
	I	J	K	L
			…	
Bucket 2: (striped in reverse order)				
	D	C	B	A
	H	G	F	E
	L	K	J	I
			…	

- ☆ Each write of four blocks from the two blocks is guaranteed to be perfectly striped across the disks!
- ☆ Reduces necessary buffer space by half.
- ☆ Cannot be generalized to $R > 2$.

The Power of Queueing the Writes



- ☆ Need pool of D queue buffers (one per disk) in internal memory.
- ☆ Write cycle: For each nonempty queue, write a block to its disk.
- ☆ After each write cycle, bring in $(1 - \epsilon) \cdot D$ block arrivals.

Problem: If the queues fill up memory, we need to flush them, which takes many I/Os.

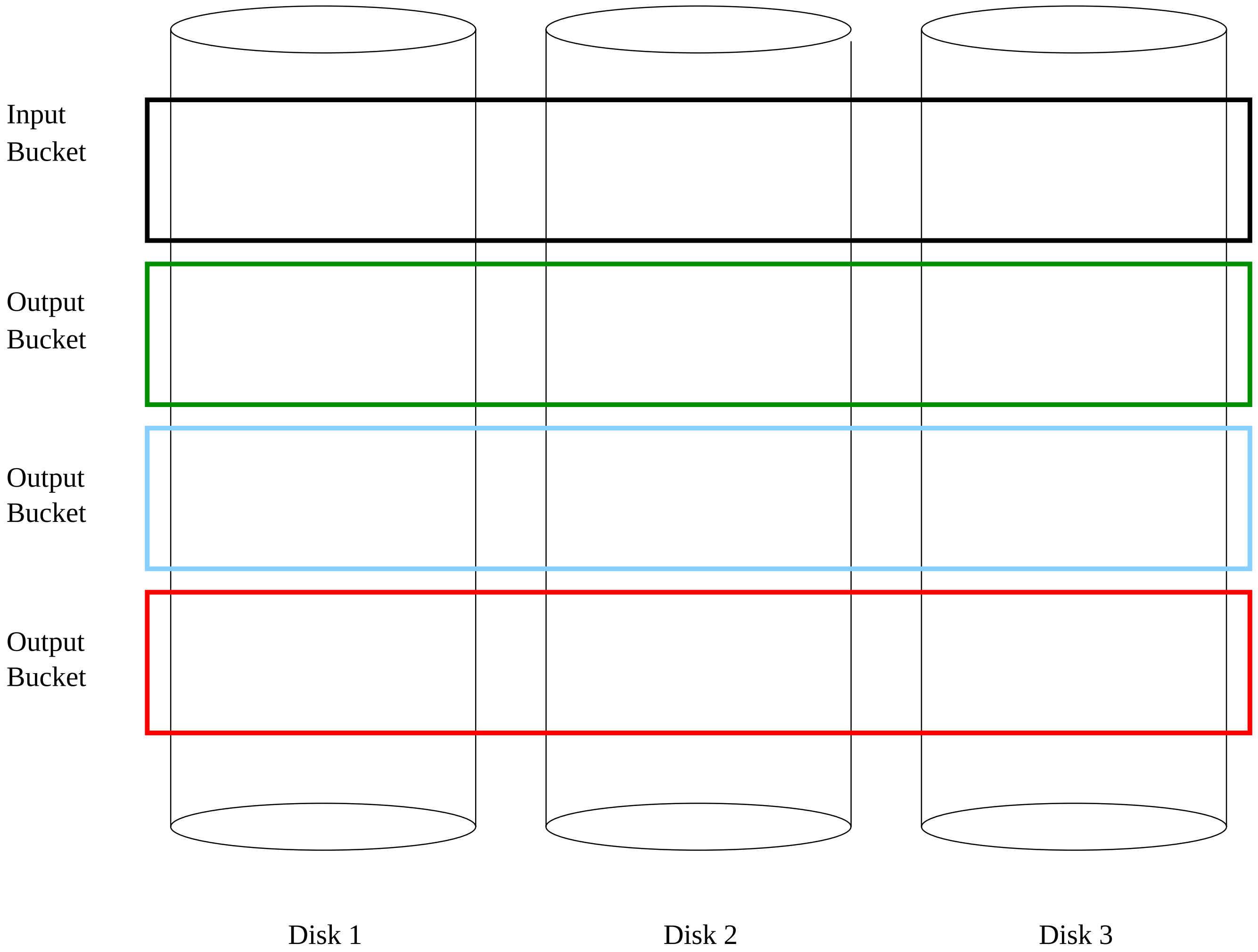
The challenge is to show that the total queue space stays small.

Randomized Distribution on Parallel Disks

1. **FRD: Fully Randomized Distribution:** *For each block, randomly select a destination disk.*
2. SRD: Simple Randomized Distribution: For each bucket, randomly select a starting disk then allocate the bucket's blocks to the disks in round-robin order.
3. RSD: Randomized Striping Distribution: For each bucket, for each successive set of D blocks allocated to that bucket, choose a new random starting disk and allocate the D blocks to disks in round-robin order.
4. **RCD: Randomized Cycling Distribution:** *For each bucket, use a different random permutation of the disk numbers.*

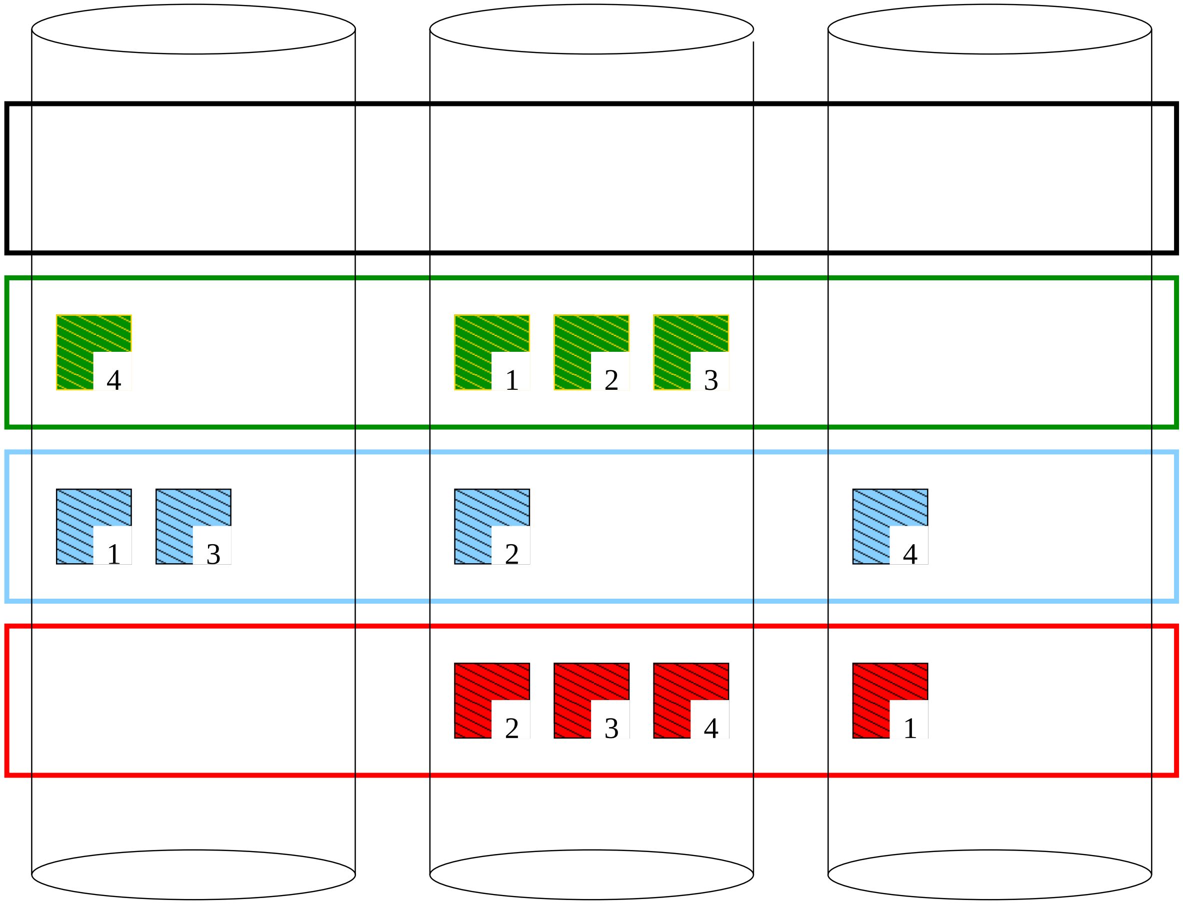
The analyses are complicated by dependencies among the sizes of the individual queues.

Bucket Distribution Variants

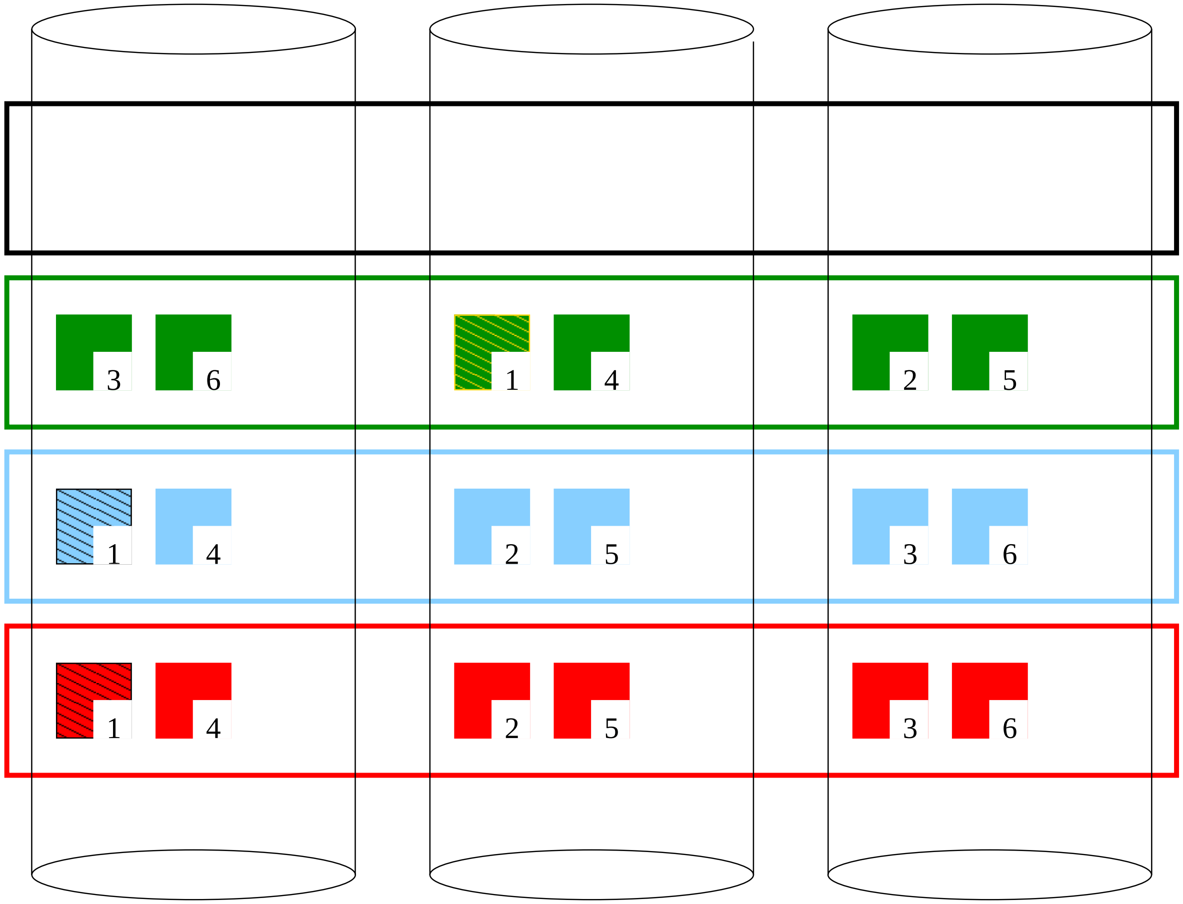


Each crosshatched disk block involves a random placement decision.

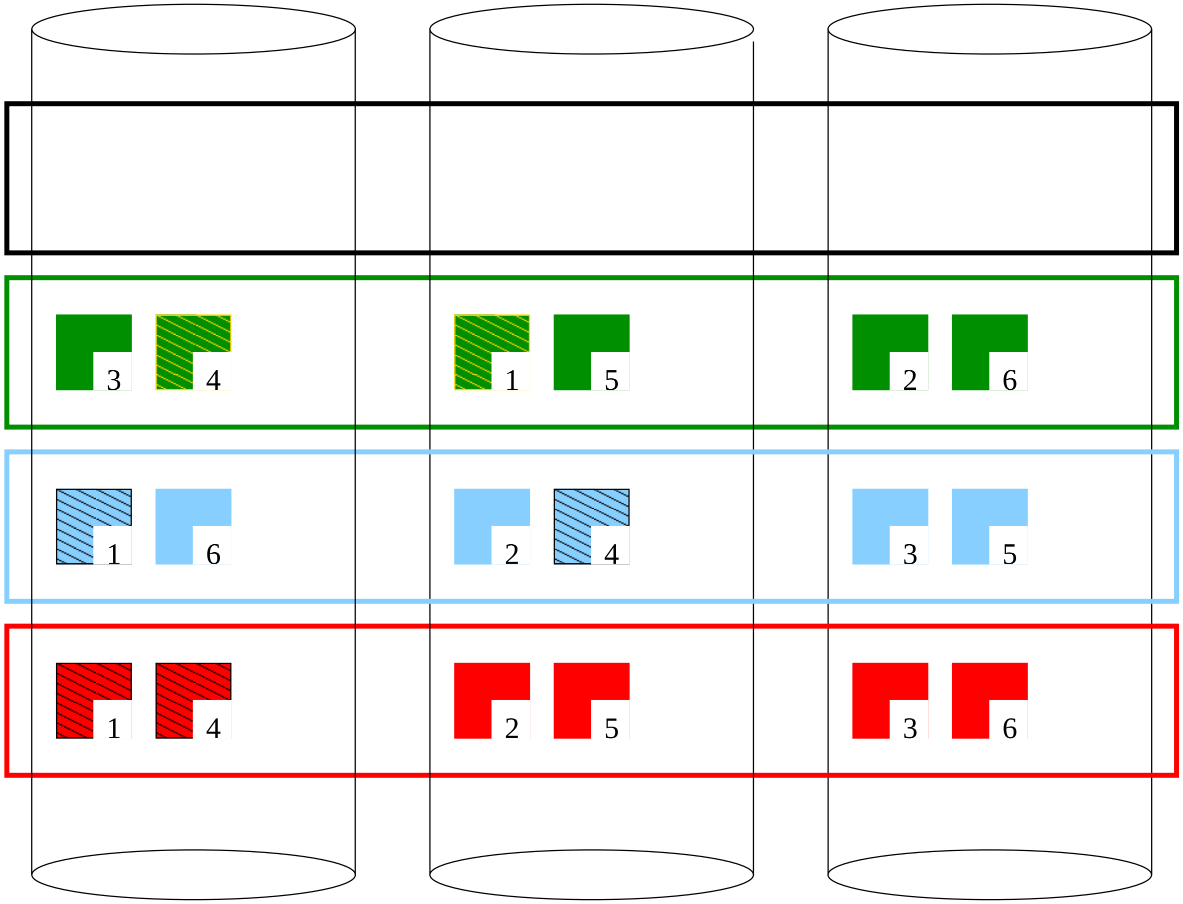
FRD (Fully Randomized Distribution)



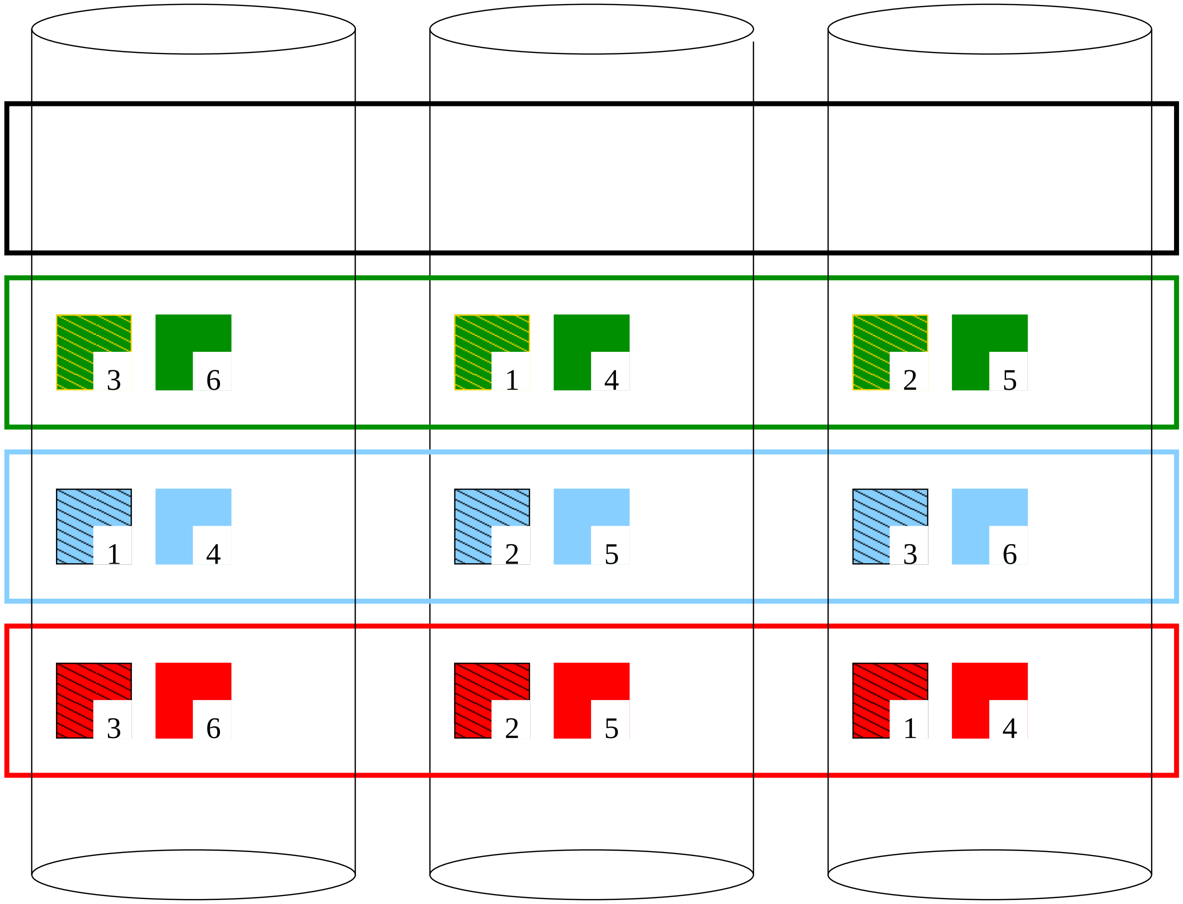
SRD (Simple Randomized Distribution)



RSD (Randomized Striping Distribution)



RCD (Randomized Cycling Distribution)



Previous Work and Our Results

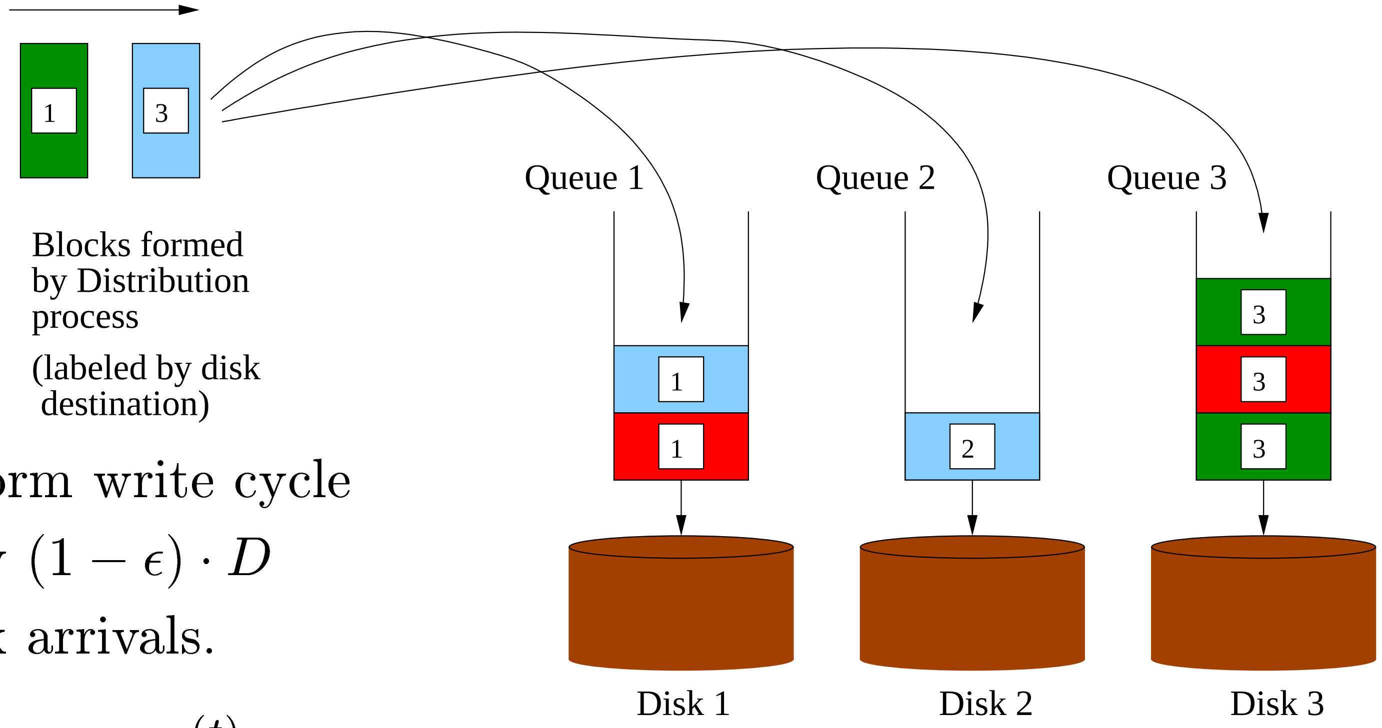
- ☆ SRM: Simple Randomized Mergesort [Barve and Vitter].
- ☆ Analysis of FRD recently given by [Sanders,Egner,Korst SODA'00] using negative dependence property.
- ☆ In this talk we reduce RCD (practical) to FRD (not practical) and thus bound the write I/Os of RCD by that of FRD.
  - (Expected) I/O complexity is optimal.
  - only a constant number of queued blocks per disk are required (on average).
- ☆ RCD read complexity is optimal; but *not* FRD's.
- ☆ RCD is simple to implement.
- ☆ Experiments confirm theoretical indications.

Outline

1. Analysis of FRD, RCD
   - ⋆ FRD guarantees and drawbacks
   - ⋆ RCD reduction to FRD
   - ⋆ RCD I/O bounds
2. Experiments
   - ⋆ FRD, RCD, SRD, RSD
3. Non-sorting Applications
4. Future Work

Analysis of Queue Space Needed

Example: $D = 3$ disks, $S = 3$ buckets



Perform write cycle every $(1 - \epsilon) \cdot D$ block arrivals.

$$Q_i^{(t)} = \text{size of queue } i \text{ (in blocks) at time } t$$

$$Q^{(t)} = \text{total queue space} = \sum_i Q_i^{(t)}$$

We use $\widehat{Q}_i^{(t)}$ and $\widehat{Q}^{(t)}$ as corresponding terms for FRD.

Total Queue Size is $W = O(D/\epsilon)$ blocks

At each read step,

Each nonempty queue writes one block to its disk.

A total of $(1-\epsilon)D$ blocks arrive in queues.

If ever the total queue size is $> W$,
flush the queues (expensive operation).

Theorem 1

Theorem 1:

Let total queue size be $W = (\ln(2) + \delta)D/\epsilon$, for some $\delta > 0$.
Let $n^{(t)}$ be the number of write steps for the t^{th} read step.
Then $\mathrm{E}(n^{(t)}) \leq 1 + e^{-\Omega(D)}$.

Lemma 2:

Let $\widehat{Q}_i^{(t)}$ be the length of $\widehat{Q}_i$ at the t^{th} read step.

Then $\mathrm{E}(\widehat{Q}_i^{(t)}) \leq \frac{1}{2\epsilon}$ and $\mathrm{Prob}\{\widehat{Q}_i^{(t)} > q\} < 2e^{-\epsilon q}$, for all $q > 0$.

Arrival rate of $(1 - \epsilon)D$ represents a fraction of the peak bandwidth D for writing. *It allows the total queue size to stay bounded as $t \to \infty$.*

Flushing the queues at very end may be nonoptimal for FRD, if $N \approx D$.

Binomial Distribution

Let $X_i^{(t)}$ be the number of blocks arriving in $\widehat{Q}_i$ in the t^{th} read step.

$X_i^{(1)}, X_i^{(2)}, X_i^{(3)} \ldots$ are independent binomially distributed random variables $B\big((1-\epsilon)D, 1/D\big)$
with $(1-\epsilon)D$ trials and probability $1/D$ of occurrence per trial.

$$\mathrm{Prob}\{B(n,p)=k\} = \left(\begin{array}{c} n \\ p \end{array}\right) p^k (1-p)^{n-k}$$

One block can leave per time unit. The number of blocks that arrive at each time unit is distributed as a $B\big((1-\epsilon)D, 1/D\big)$ random variable.

Probability Generating Functions

$$G_X(z) = \sum_{k \ge 0} \mathrm{Prob}\{X = k\} z^k$$

encodes complete information about the distribution of random variable X.

Properties:

1. $G'_X(1) = \sum_{k \ge 0} \mathrm{Prob}\{X = k\} k z^{k-1} \Big|_{z=1} = \sum_{k \ge 0} k \, \mathrm{Prob}\{X = k\} = \mathrm{E}(X)$
2. $G_X(1) = 1$
3. $G_{X+Y}(z) = G_X(z) G_Y(z)$ if the RVs X and Y are independent.

$$\begin{aligned} G_X(z) &= \sum_{k \ge 0} p_k z^k \\ G_Y(z) &= \sum_{k \ge 0} q_k z^k \\ G_{X+Y}(z) &= \sum_{k \ge 0} (p_0 q_k + p_1 q_{k-1} + \ldots + p_k q_0) z^k \end{aligned}$$

Probability Generating Functions

Let $Y_i^{(t+1)} = \begin{cases} \widehat{Q}_i^{(t)} - 1 & \text{if } \widehat{Q}_i^{(t)} \geq 1 \\ 0 & \text{otherwise} \end{cases}$

The recurrence for queue size is

$$\widehat{Q}_i{}^{(t+1)} = Y_i^{(t+1)} + X_i^{(t+1)}.$$

$Y_i^{(t+1)}$ = # blocks still in queue from previous time step, and
$X_i^{(t+1)}$ = # newly arriving blocks for queue i.

By independence of $X_i^{(t)}$ and $Y_i^{(t)}$,

$$G_{\widehat{Q}_i{}^{(t+1)}}(z) = G_{Y_i^{(t+1)}}(z) \times G_{X_i^{(t+1)}}(z)$$

Newly Arriving Blocks

$$\begin{aligned}
X_i^{(t+1)}(z) &= \text{the number of newly arriving blocks for queue i} \\
&= T_1(z) + T_2(z) + \ldots + T_{(1-\epsilon)D}(z) \\
&\quad \text{(sum of independent 0-1 RVs)}
\end{aligned}$$

$$\begin{aligned}
G_{T_j}(z) &= \left(1 - \frac{1}{D}\right) z^0 + \frac{1}{D} z^1 \\
&= \frac{D-1}{D} + \frac{z}{D} \\
&= \frac{1}{D}(z + D - 1)
\end{aligned}$$

$$\Longrightarrow \quad G_{X_i^{(t+1)}}(z) = \left(\frac{z + D - 1}{D}\right)^{(1-\epsilon)D}$$

Blocks still in queue

$$\text{Let } Y_i^{(t+1)} = \begin{cases} \widehat{Q}_i^{(t)} - 1 & \text{if } \widehat{Q}_i^{(t)} \geq 1 \\ 0 & \text{otherwise} \end{cases}$$

$$= \widehat{Q}_i^{(t)} - 1 + [\widehat{Q}_i^{(t)} = 0]$$

$$\begin{aligned} \text{If } G_A(z) &= p_0 z^0 + p_1 z^1 + p_2 z^2 + \ldots \\ \text{then } G_{A-1}(z) &= p_0 z^{-1} + p_1 z^0 + p_2 z^1 + \ldots = \frac{1}{z} G_A(z) \\ G_{A-1+[A=0]}(z) &= p_0 z^0 + p_1 z^0 + p_2 z^1 + \ldots = \frac{1}{z} G_A(z) + A(0) - \frac{1}{z} A(0) \end{aligned}$$

$$\Longrightarrow \quad G_{Y_i^{(t+1)}}(z) = \frac{1}{z} G_{\widehat{Q}_i^{(t)}}(z) + G_{\widehat{Q}_i^{(t)}}(0) - \frac{1}{z} G_{\widehat{Q}_i^{(t)}}(0)$$

$$= \frac{1}{z} G_{\widehat{Q}_i^{(t)}}(z) + G_{\widehat{Q}_i^{(t)}}(0) \left(1 - \frac{1}{z}\right)$$

Result of Lemma 2

In steady state, as $t \rightarrow \infty$, $G_{\widehat{Q}_i{}^{(t+1)}}(z) = G_{\widehat{Q}_i^{(t)}}(z) = G(z)$.

$$\begin{aligned} G(z) &= G_{Y_i^{(\infty)}}(z) \times G_{X_i^{(\infty)}}(z) \\ &= \left(\frac{1}{z}G(z) + G(0) - \frac{1}{z}G(0)\right) \times \left(\frac{z+D-1}{D}\right)^{(1-\epsilon)D} \\ \Longrightarrow \quad G(z) &= \frac{G(0)(1-\frac{1}{z})}{(\frac{z+D-1}{D})^{-(1-\epsilon)D} - \frac{1}{z}}. \end{aligned}$$

Result of Lemma 2

To solve for $G(0)$, we know that $G(1) = 1$. By L'Hôpital's rule,

$$\begin{aligned} 1 = G(1) &= \lim_{z \to 1} \left(\frac{G(0)(1 - \frac{1}{z})}{(\frac{z+D-1}{D})^{-(1-\epsilon)D} - \frac{1}{z}} \right) = \frac{G(0)}{\epsilon} \\ \Longrightarrow \quad G(0) &= \epsilon \end{aligned}$$

$$\begin{aligned} \Longrightarrow \quad G(z) &= \frac{(1-z)\epsilon}{1 - G_{X_i}(z)^{-1} z}, \text{ where } G_{X_i}(z) = \left(\tfrac{z+D-1}{D}\right)^{(1-\epsilon)D} \\ \mathrm{E}(\widehat{Q}_i^{(t)}) &\leq \mathrm{E}(\widehat{Q}_i{}^{(\infty)}) = G'(1) \leq \frac{1}{2\epsilon} \quad \text{(by L'Hôpital's rule)} \end{aligned}$$

Result of Lemma 2

We now show Prob$\{\widehat{Q}_i^{(t)} > q\} \le$ Prob$\{\widehat{Q}_i{}^{(\infty)} > q\}$ for all $q > 0$: .

Consider two queues processing identical input but with different initial lengths.

In any step the difference in length remains the same or gets reduced by one. This continues until lengths become equal at which point they remain the same forever.

The queues are initially empty at time $t = 0$ (i.e., $\widehat{Q}_i^{(t)} = 0$),
but in steady state the queues are not empty.
Therefore, the tail probability Prob$\{\widehat{Q}_i^{(t)} > q\}$ is
$\le$ the steady-state tail probability Prob$\{\widehat{Q}_i{}^{(\infty)} > q\}$.

Result of Lemma 2

$$G(e^\epsilon) = \frac{\epsilon(1-e^\epsilon)}{1 - \frac{z}{G_{X_i}(e^\epsilon)}} < \frac{\epsilon(1-e^\epsilon)}{1-\exp\bigl(\epsilon-(1-\epsilon)(e^\epsilon-1)\bigr)} < 2$$

using the bound $\ln(1+x) = x - \dfrac{x^2}{2} + \dfrac{x^3}{3} - \ldots < x$, for $|x| < 1$.

General Tail Inequality:

$$\begin{aligned} \mathrm{Prob}\{X \ge r\} &= p_r + p_{r+1} + \ldots \\ &\le z^{-r} G_X(z), \qquad \text{for all } z > 1 \\ &= p_0 z^{-r} + p_1 z^{-r+1} + \ldots + p_r z^0 + p_{r+1} z^1 + \ldots \end{aligned}$$

Substituting $z = e^\epsilon > 1$ and $r = q$

$$\Longrightarrow \quad \mathrm{Prob}\{\widehat{Q}_i{}^{(\infty)} > q\} < G(e^\epsilon) e^{-\epsilon q} = 2e^{-\epsilon q}$$

Lemma 3

At each read step,

Each nonempty queue writes one block to its disk.

$(1-\epsilon)D$ blocks arrive in queues.

Let $\widehat{Q}^{(t)} = Q_1^{(t)} + \ldots + Q_D^{(t)}$ with $\widehat{Q}_i^{(t)}$, as in Lemma 2.
Then if the total queue capacity is $W = (\ln 2 + \delta)D/\epsilon$, we have

$$\begin{aligned} \mathrm{E}(\widehat{Q}^{(t)}) &\leq \frac{D}{2\epsilon}; \\ \mathrm{Prob}\{\widehat{Q}^{(t)} > qD\} &< e^{-\delta D}, \end{aligned}$$

where $\delta = \epsilon \dfrac{W}{D} - \ln 2$ is a parameter that can be set.

$\Longrightarrow$ Buffer overflow is exponentially improbable.

δ gives a tradoff between queue space and likelihood of overflow.

Proof of Lemma 3

Negative Association (NA)

If an item is placed in a queue then it cannot be placed in any of the other queues. The sizes of the other queues will then be shorter. This is, in a sense, better than independence. Placing an item in one queue affects the other queues negatively.

Proof of Lemma 3

Let $W = \Theta(\delta D/s)$ be the allowable total memory space (in blocks):

$$\begin{aligned} \mathrm{Prob}\{\widehat{Q}^{(t)} > W\} &= \mathrm{Prob}\{e^{s\widehat{Q}^{(t)}} > e^{sW}\} \\ &< e^{-sW}\,\mathrm{E}(e^{s\widehat{Q}^{(t)}}) \quad \text{by tail inequality} \end{aligned}$$

If the queue sizee $\{\widehat{Q}_i^{(t)}\}$ were independent, we would get a Chernoff bound on total queue space:

$$\begin{aligned} \mathrm{E}(e^{s\widehat{Q}^{(t)}}) &= \mathrm{E}(e^{\sum_{0\le i<D} s\widehat{Q}_i^{(t)}}) = \mathrm{E}\Bigg(\prod_{0\le i<D} e^{s\widehat{Q}_i^{(t)}}\Bigg) \\ &= \prod_{0\le i<D} \mathrm{E}(e^{s\widehat{Q}_i^{(t)}}) = \big(\mathrm{E}(e^{s\widehat{Q}_1^{(t)}})\big)^D \end{aligned}$$

W/o independence, negative association gives Chernoff-like bound:

$$\mathrm{E}(e^{s\widehat{Q}^{(t)}}) \le \prod_{0\le i<D} \mathrm{E}(e^{s\widehat{Q}_i^{(t)}}) = \big(\mathrm{E}(e^{s\widehat{Q}_1^{(t)}})\big)^D$$

Proof of Lemma 3

$\mathrm{E}(e^{\epsilon \widehat{Q}_1^{(t)}})$ is the moment generating function

$$\mathrm{Prob}\{\widehat{Q}_1 = k\} e^{\epsilon k} = G(e^z).$$

From Lemma 2 , we know $\mathrm{E}(e^{\epsilon Q_1^{(t)}}) = G(e^{\epsilon}) < 2$

Choose $s = \epsilon$:

$$\begin{aligned}
\mathrm{Prob}\{\widehat{Q}^{(t)} > W\} &< e^{-\epsilon W} \mathrm{E}(e^{s \widehat{Q}^{(t)}}) \\
&< e^{-\epsilon W} \left(\mathrm{E}(e^{\epsilon \widehat{Q}_1^{(t)}})\right)^D \\
&< e^{-\epsilon W} (2^D) \\
&= e^{-(\epsilon \frac{W}{D} - \ln 2) D} \\
&= e^{-\delta D}
\end{aligned}$$

$\mathrm{E}(\widehat{Q}^{(t)}) \leq \dfrac{D}{2\epsilon}$ since $\mathrm{E}(\widehat{Q}_i^{(t)}) \leq \dfrac{1}{2\epsilon}$ (linearity of expected value)

Theorem 1 Result

Lemma 3 gives the probability that the queues are flushed.

$$p = \mathrm{Prob}\{\widehat{Q}^{(t)} > W\} \leq e^{-\delta D}$$

In the worst case it takes $W + D$ write steps to flush the queues. The expected number $n^{(t)}$ of write steps at time t is

$$\begin{aligned} \mathrm{E}(n^{(t)}) &\leq 1 + p(W + D) \\ &= 1 + O\left(\frac{D}{\epsilon}\right) e^{-\delta D} \\ &\leq 1 + e^{-\Omega(D)} \end{aligned}$$

Main Theorem

- ☆ The Main Theorem states that RCD has the same performance guarantees as does FRD. (In fact, they're better, because of the final emptying of the queues is guaranteed to be balanced.)

- ☆ The challenge is to show that the expected exponential of the total queue space $\mathrm{E}(e^{sQ^{(t)}})$ in RCD is at most that of FRD:

$$\text{that is, } \mathrm{E}(e^{sQ^{(t)}}) \leq \mathrm{E}(e^{s\widehat{Q}^{(t)}})$$

- ☆ We would then inherit the desired tail bound on the total queue size of RCD:

$$\begin{aligned}
\mathrm{Prob}\{Q^{(t)} > W\} &= \mathrm{Prob}\{e^{sQ^{(t)}} > e^{sW}\} \\
&< e^{-sW}\mathrm{E}(e^{sQ^{(t)}}) \quad \text{by tail inequality} \\
&< e^{-sW}\mathrm{E}(e^{-s\widehat{Q}^{(t)}}) \\
&= e^{-\delta D}
\end{aligned}$$

Reduction of RCD to FRD

singleton bucket $\equiv$ bucket that contains a total of one block

FRD $\equiv$ RCD in which all buckets are singletons (each block is randomly assigned to a disk)

```
for $r := 1$ to $t$ do
    while there is a nonsingleton bucket $b$ that
          issues at least one block at time step $r$
        do the following transformation step
            Remove one block from bucket $b$ at time step $r$;
            Create a new singleton bucket with its block at time step $r$
        enddo
enddo
```

Key Lemma: *Each transformation step causes the quantity* $\mathrm{E}(e^{sQ^{(t)}})$ *to increase or stay the same.* At the end, it is $\mathrm{E}(e^{s\widehat{Q}^{(t)}})$.

Layout of Bucket b on the Disks

What is the effect of removing the first block issued by bucket b at time step r?
(Assume WLOG that the other blocks can stay where they are and that the disks are arranged in cycle order.)

		Disks							
		1	2	3	4	5	6	7	8
	r		●	●					
	$r+1$				●	●	●		
	$r+2$								
Time									
								●	
									●
	$t-1$	●	●	●	●				
	t								

Analogy with a Lake

- ☆ Suppose each day the sun removes a gallon of water from the lake.
- ☆ Then, later in the day, it may or may not rain.
  If it rains, the lake gets some added water.
- ☆ If the lake always has at least two gallons at the start of each day, then if we remove a gallon of water in April, it will have a gallon less in September.
- ☆ If on the other hand, the lake has only one gallon at the start of June 28, then the sun will empty the lake. Therefore, if we remove a gallon in April, there will be no change in September.

A Critical Queue (a Sufficiently Full Lake)

t'	r	$r+1$	$\ldots$			$t-1$	t
$Q_i^{(t')}$	≥ 2	≥ 2	$\ldots$	≥ 2	≥ 2	≥ 2	$Q_i^{(t)}$
Item Arrivals	●		$\ldots$	●	●		

☆ The size of the ith queue $Q_i^{(t')}$ is at least 2 for $r \leq t' < t$.

☆ $Q_i^{(t')}$ will remain at least 1 even without the arrival at time step r, and a block will continue to be consumed at each time step.

☆ Hence, if there is no arrival of a block into the ith queue at time r (keeping all other block arrivals the same),
the final size $Q_i^{(t)}$ of the queue will be one less than before.

Proof

- ⋆ $Q^{(t)}$ is the sum of queues at time t.
- ⋆ $Q'^{(t)}$ is the sum of queues at time t after the block is removed.
- ⋆ $Q''^{(t)}$ is the sum of queues at time t after bucket b has been transformed.

Then

$$Q''^{(t)} = Q'^{(t)} + [\text{new bucket increases queue size}]$$

We want to show

$$\mathrm{E}\big(f(Q''^{(t)})\big) \geq \mathrm{E}\big(f(Q^{(t)})\big),$$

where $f(x) = e^{sx}$.

Proof

Suppose that c of the D possible starting points for bucket b are critical with respect to time step t.

- ☆ Case 1: Starting Point is Critical

$$Q''^{(t)} \text{ is either } Q^{(t)} - 1 \text{ or } Q^{(t)}$$

$$\begin{aligned}
&\mathrm{E}\big(\, f(Q''^{(t)}) \;\big|\; \text{the starting point is critical}\big) \\
&\qquad \geq \quad \left(1 - \frac{c}{D}\right) \mathrm{E}\big(\, f(Q^{(t)} - 1) \;\big|\; \text{starting point is critical}\big) \\
&\qquad\qquad\quad + \frac{c}{D}\, \mathrm{E}\big(\, f(Q^{(t)}) \;\big|\; \text{starting point is critical}\big) \\
&\qquad = \quad \left(\left(1 - \frac{c}{D}\right) \frac{1}{f(1)} + \frac{c}{D}\right) \mathrm{E}\big(\, f(Q^{(t)}) \;\big|\; \text{starting point is critical}\big),
\end{aligned}$$

since $f(x) = e^{sx}$ and thus $f(Q^{(t)} - 1) = \dfrac{1}{f(1)} f(Q^{(t)})$.

Proof

☆ Case 2: Starting Point is Non-critical

$$Q''^{(t)} \text{ is either } Q^{(t)} \text{ or } Q^{(t)} + 1$$

$$\begin{aligned}
&\mathrm{E}\big(\, f(Q''^{(t)}) \;\big|\; \text{starting point is non-critical}\big) \\
&\quad \geq \quad \left(1 - \frac{c}{D}\right) \mathrm{E}\big(\, f(Q^{(t)}) \;\big|\; \text{starting point is non-critical}\big) \\
&\qquad\quad + \frac{c}{D}\, \mathrm{E}\big(\, f(Q^{(t)} + 1) \;\big|\; \text{starting point is non-critical}\big) \\
&\quad = \quad \left(\left(1 - \frac{c}{D}\right) + \frac{c}{D}\, f(1)\right) \mathrm{E}\big(\, f(Q^{(t)}) \;\big|\; \text{starting point is non-critical}\big),
\end{aligned}$$

since $f(x) = e^{sx}$ and thus $f(Q^{(t)} + 1) = \dfrac{1}{f(1)} f(Q^{(t)})$.

Proof

Before Transformation

$$
\begin{aligned}
\mathrm{E}\big(f(Q^{(t)})\big) \\
&= \frac{c}{D}\,\mathrm{E}\big(\,f(Q^{(t)}) \mid \text{starting point is critical}\,\big) \\
&\qquad + \left(1 - \frac{c}{D}\right)\mathrm{E}\big(\,f(Q^{(t)}) \mid \text{starting point is non-critical}\,\big) \qquad (1)
\end{aligned}
$$

After Transformation

$$
\begin{aligned}
\mathrm{E}\big(f(Q''^{(t)})\big) \\
&= \frac{c}{D}\mathrm{E}(f(Q''^{(t)})|\text{starting point is critical}) \\
&\qquad + \left(1 - \frac{c}{D}\right)\mathrm{E}(f(Q''^{(t)})|\text{starting point is non-critical}) \\
&\geq \frac{c}{D}\left(\left(1 - \frac{c}{D}\right)\frac{1}{f(1)} + \frac{c}{D}\right)\mathrm{E}(\,f(Q^{(t)}) \mid \text{starting point is critical}\,) \\
&\qquad + \left(1 - \frac{c}{D}\right)\left(\left(1 - \frac{c}{D}\right) + \frac{c}{D}\,f(1)\right)\mathrm{E}(\,f(Q^{(t)}) \mid \text{starting point is non-critical}\,) \qquad (2)
\end{aligned}
$$

Proof

Combining (1) and (2) from the previous slide we get

$$
\begin{aligned}
&\mathrm{E}(f(Q''^{(t)})) - \mathrm{E}(f(Q^{(t)})) \\
&\quad \geq \quad -\frac{1}{f(1)}\left(\frac{c}{D}(f(1)-1) - \frac{c^2}{D^2}(f(1)-1)\right) \mathrm{E}(\, f(Q^{(t)}) \mid \text{starting point is critical}) \\
&\qquad\qquad + \left(\frac{c}{D}(f(1)-1) - \frac{c^2}{D^2}(f(1)-1)\right) \mathrm{E}(\, f(Q^{(t)}) \mid \text{starting point is non-critical}). \\
&\quad \geq \quad 0 \text{ by Lemma below}
\end{aligned}
$$

Lemma 5:

$$
\begin{aligned}
&\mathrm{E}(\, f(Q^{(t)}) \mid \text{starting point is non-critical}) \\
&\quad \geq \quad \mathrm{E}(\, f(Q^{(t)} - 1) \mid \text{starting point is critical}) \\
&\quad = \quad \frac{1}{f(1)} \mathrm{E}(\, f(Q^{(t)}) \mid \text{starting point is critical})
\end{aligned}
$$

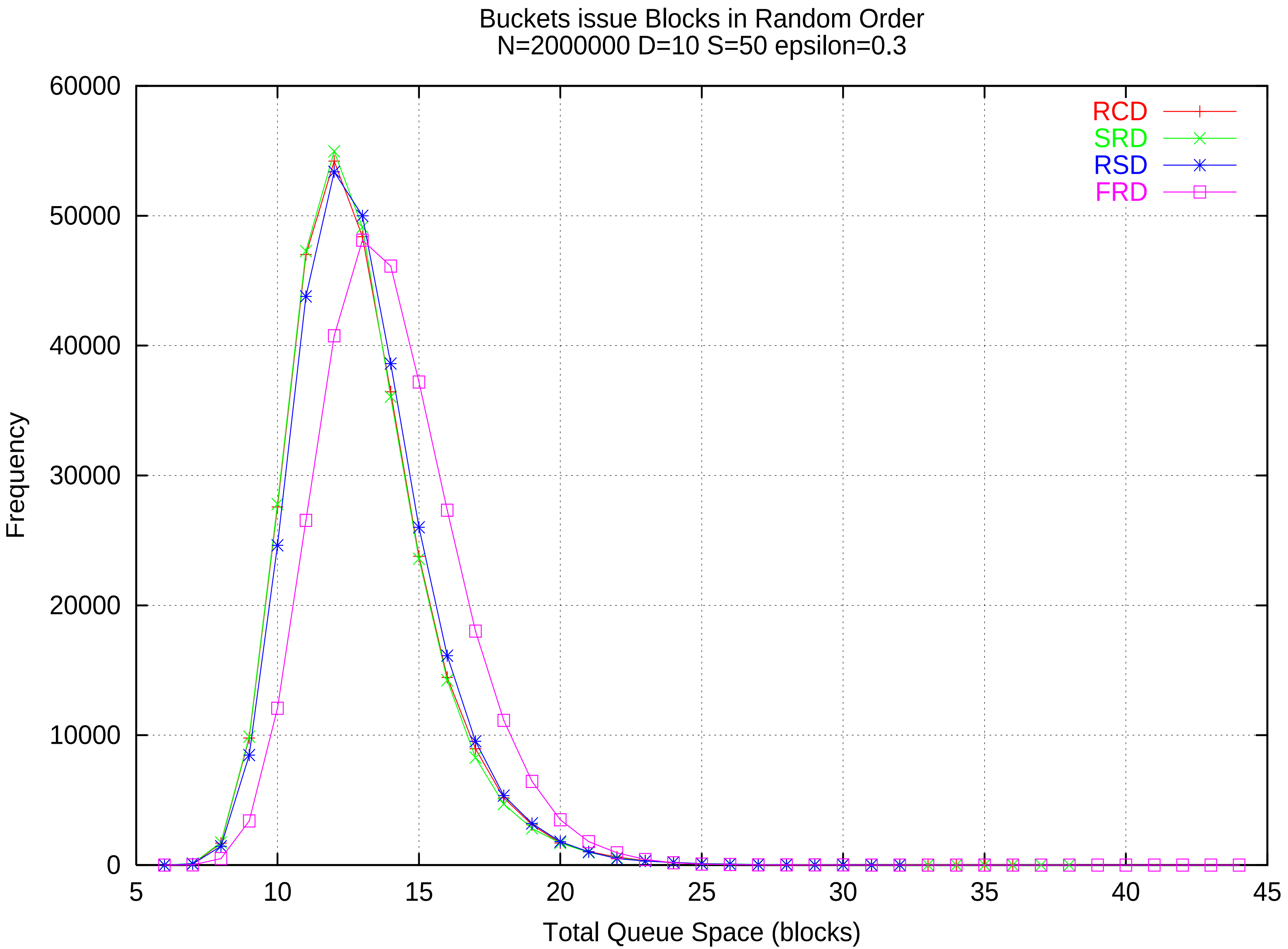
Intuition: Let $1 \leq c \leq D$ be the number of critical starting points. The proof uses a mapping between the $c(D-1)!$ cycle orders with a critical starting point and the $(D-c)(D-1)!$ cycle orders with a non-critical starting point.

Experimental Results

Testing with smaller numbers of disks shows that even non-asymptotic behavior is attractive. Test parameters:

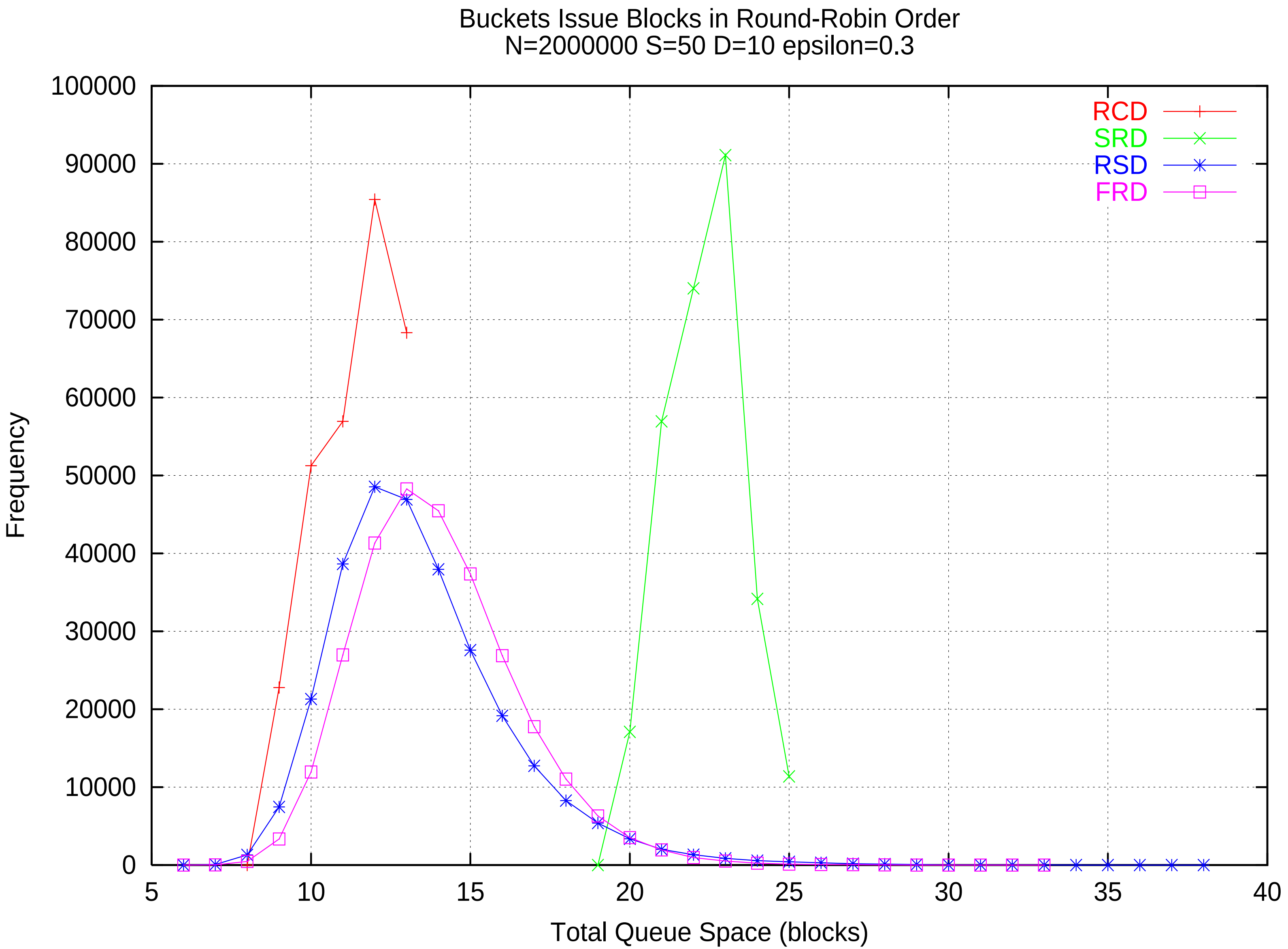
- ☆ Block arrival regimes:
  1. "Random input": next bucket to receive a block is chosen randomly.
  2. "Balanced input" : round-robin issue of blocks to buckets.
- ☆ Small and large ϵ. Can $\epsilon = 0$? (That is, can we write out a full $(1-\epsilon)D = D$ items in each write cycle?)
- ☆ Wait for steady state and then record the histogram of the total queue space (i.e., total memory space) used.

Random Issue, $\epsilon = 0.3$



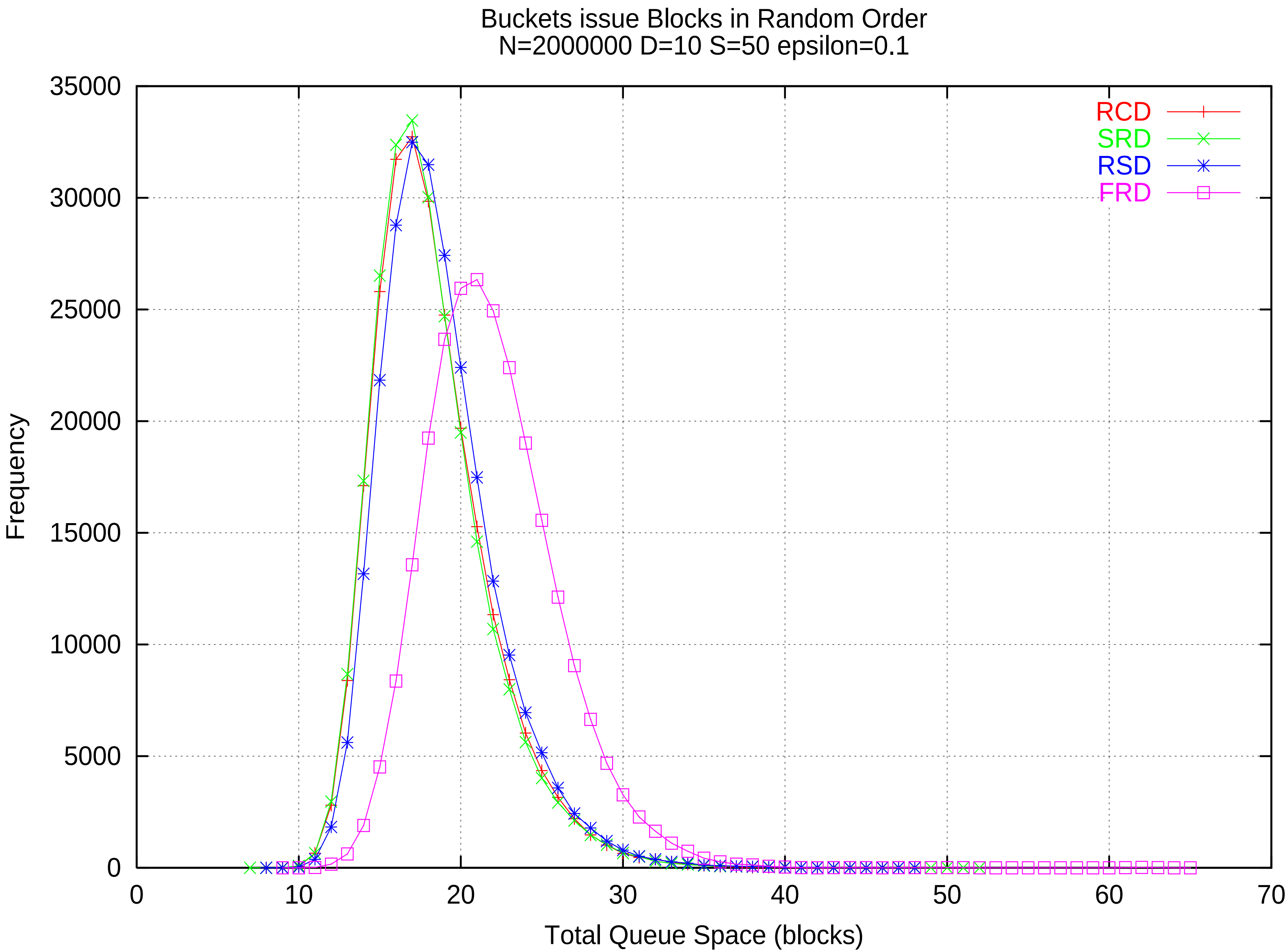
Center for Geometric & Biological Computing

Round-Robin Issue, $\epsilon = 0.3$

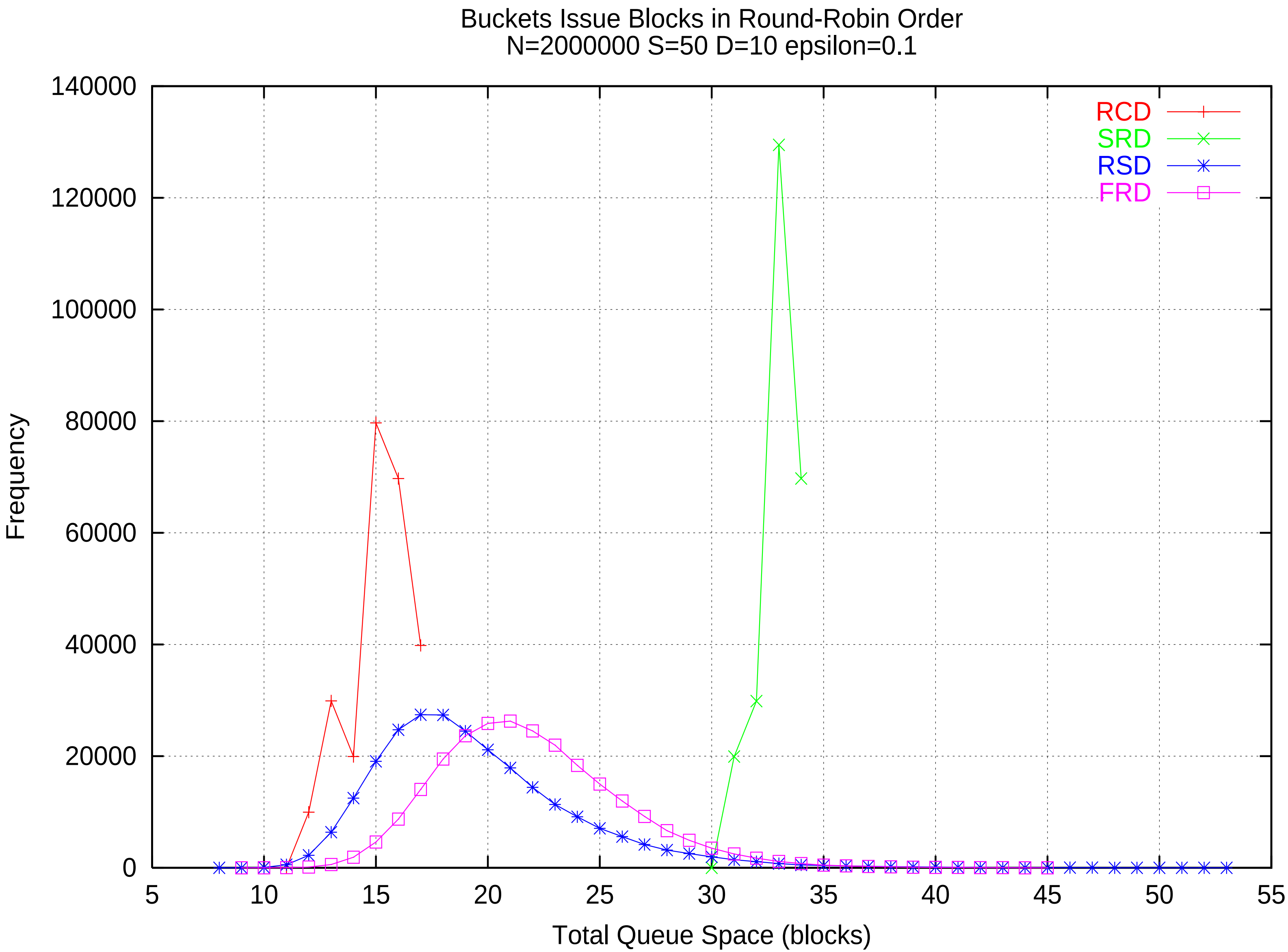


Center for Geometric & Biological Computing

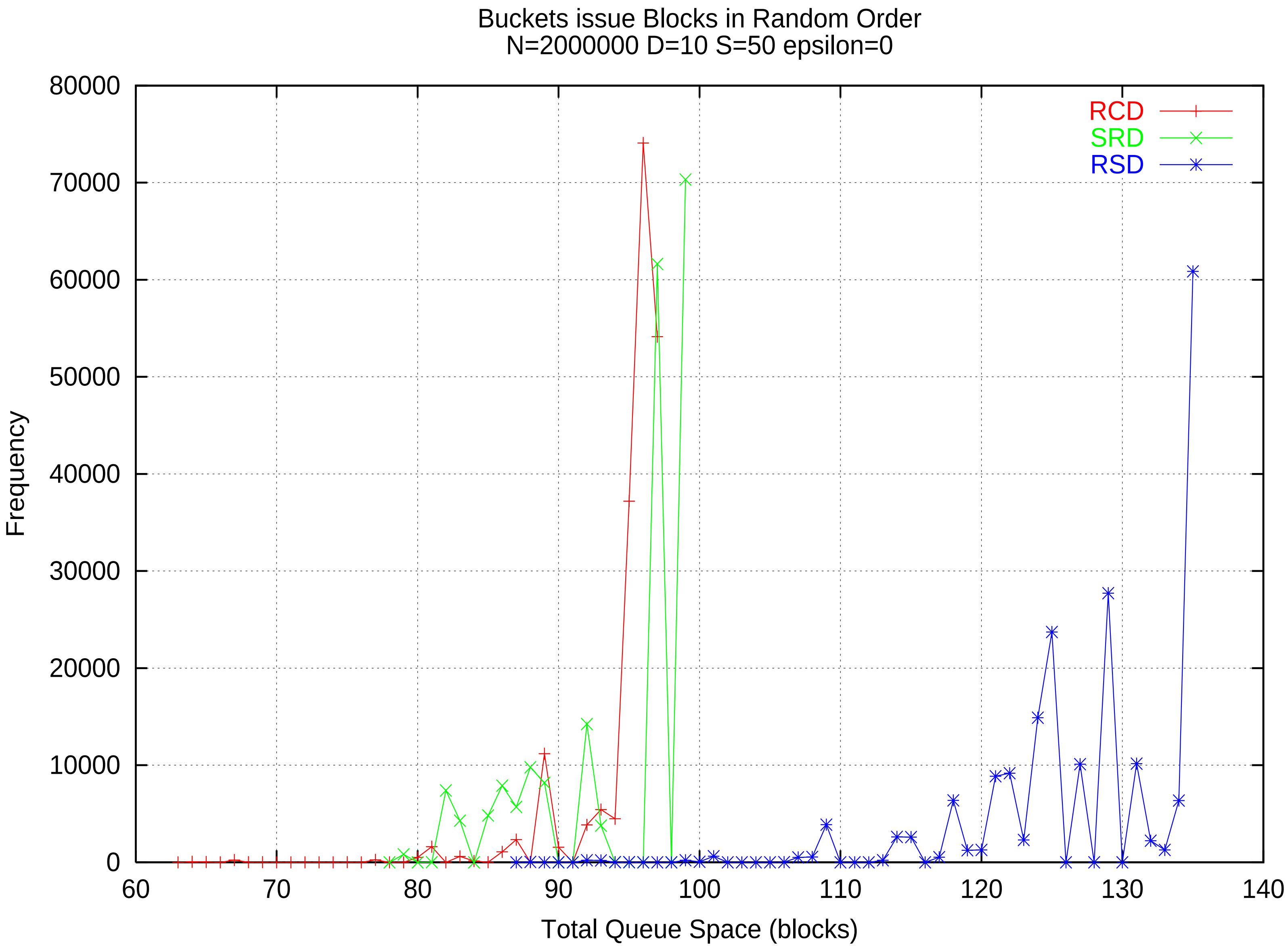
Random Issue, $\epsilon = 0.1$



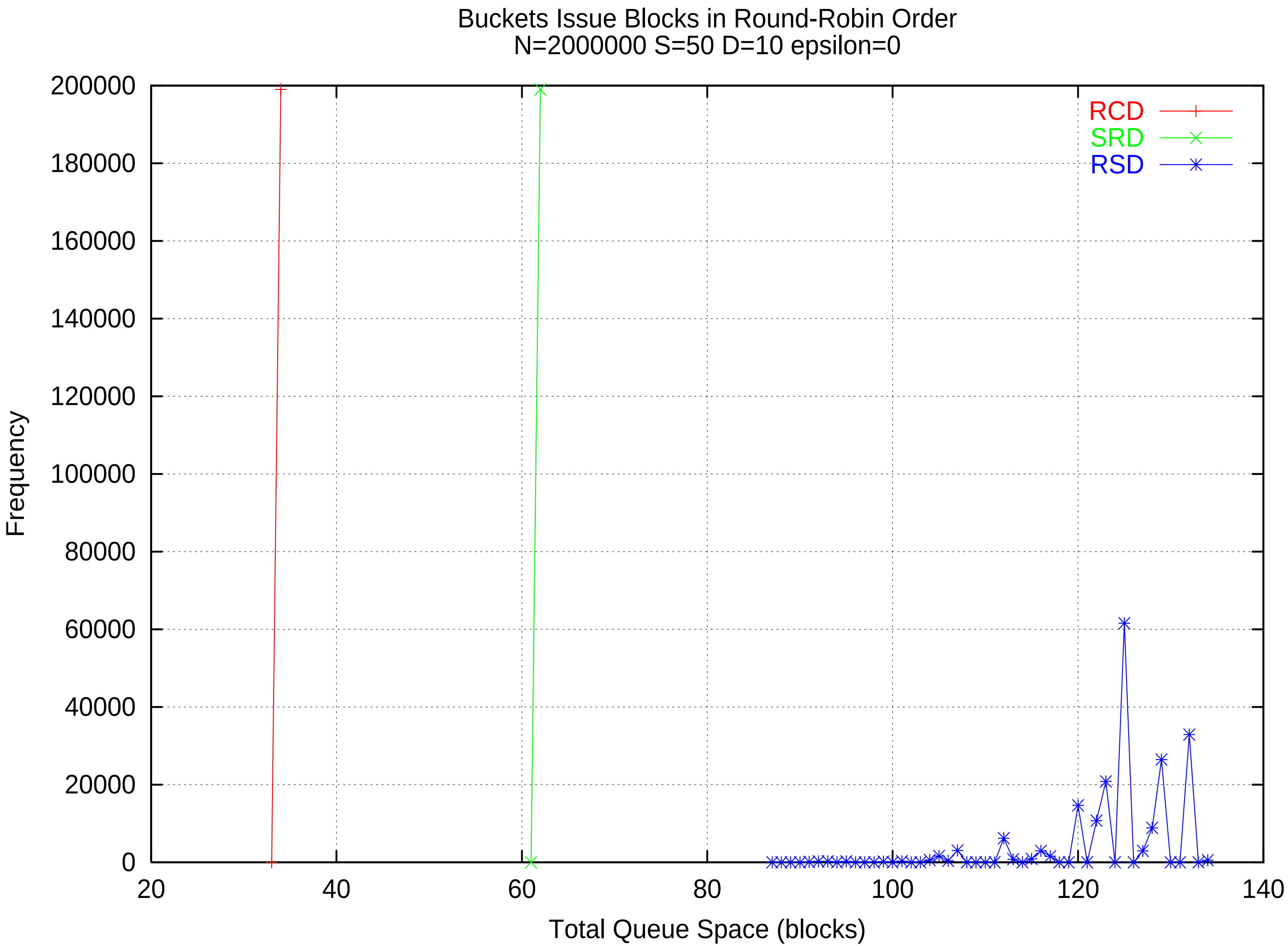
Round-Robin Issue, $\epsilon = 0.1$



Random Issue, $\epsilon = 0$



Round-Robin Issue, $\epsilon = 0$



Center for Geometric & Biological Computing

Conclusions and Future Work

- ☆ RCD is a simple, practical, and provably good method for sorting with parallel disks.
- ☆ We conjecture that SRD and RSD perform similarly to RCD.
- ☆ Randomized cycling can be applied to merge sort to get a practical and theoretically optimal sorting algorithm.
- ☆ RCD can be used in distribution sweeping applications.
- ☆ We are starting practical implementation/study.