

Storage Systems

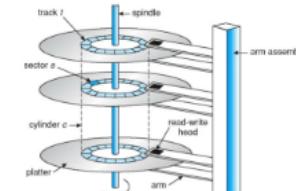
Chapter 12: Mass-Storage Systems

- Overview of Mass Storage Structure
- Disk Structure
- Disk Scheduling

Overview of Mass Storage Structure

- Magnetic disks provide bulk of secondary storage of modern computers
 - Drives rotate at 60 to 200 times per second
 - Transfer rate is rate at which data flow between drive and computer
 - Positioning time (random-access time) is time to move disk arm to desired cylinder (seek time) and time for desired sector to rotate under the disk head (latency)
- Drives can be removable
- Drive attached to computer via I/O bus
 - Buses vary, including EIDE, ATA, SATA, USB, Fibre Channel, SCSI
 - Host controller in computer uses bus to talk to disk controller built into drive or storage array

Moving-head Disk Mechanism



Disk Structure

- Disk drives are addressed as large 1-dimensional arrays of logical blocks, where the logical block is the smallest unit of transfer
- The 1-dimensional array of logical blocks is mapped into the sectors of the disk sequentially
 - Sector 0 is the first sector of the first track on the outermost cylinder
 - Mapping continues in order through that track, then the rest of the cylinders from outermost to innermost

Disk Scheduling

- The operating system is responsible for using hardware efficiently — for the disk drives, this means having a fast access time and disk bandwidth
- Access time has two major components
 - **Seek time** is the time for the disk arm to move the heads to the cylinder containing the desired sector
 - **Rotational latency** is the additional time waiting for the disk to rotate the desired sector to the disk head
- Minimize seek time
- Seek time \square seek distance
- Disk bandwidth is the total number of bytes transferred, divided by the total time between the first request for service and the completion of the last transfer

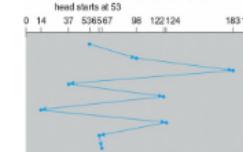
Disk Scheduling (Cont)

- Several algorithms exist to schedule the servicing of disk I/O requests
- We illustrate them with a request queue (0-189)
 - queue = 98, 183, 37, 122, 14, 124, 65, 67

Head pointer 53

FCFS

- Illustration shows total head movement of 640 cylinders
queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53



SSTF

- Selects the request with the minimum seek time from the current head position
- SSTF scheduling is a form of SJF scheduling; may cause starvation of some requests
- Illustration shows total head movement of 236 cylinders

SSTF (Cont)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53

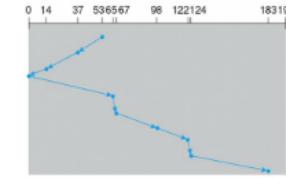


SCAN

- The disk arm starts at one end of the disk, and moves toward the other end, servicing requests until it gets to the other end of the disk, where the head movement is reversed and servicing continues.
- **SCAN algorithm**: Sometimes called the **elevator algorithm**
- Illustration shows total head movement of 236 cylinders

SCAN (Cont.)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53



C-SCAN

- Provides a more uniform wait time than SCAN
- The head moves from one end of the disk to the other, servicing requests as it goes
 - When it reaches the other end, however, it immediately returns to the beginning of the disk, without servicing any requests on the return trip
- Treats the cylinders as a circular list that wraps around from the last cylinder to the first one

C-SCAN (Cont)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53

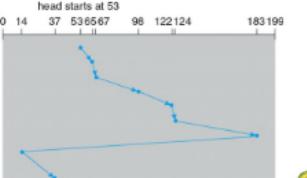


C-LOOK

- Version of C-SCAN
- Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk

C-LOOK (Cont.)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53



Selecting a Disk-Scheduling Algorithm

- SSTF is common and has a natural appeal
- SCAN and C-SCAN perform better for systems that place a heavy load on the disk
- Performance depends on the number and types of requests
- Requests for disk service can be influenced by the file-allocation method
- The disk-scheduling algorithm should be written as a separate module of the operating system, allowing it to be replaced with a different algorithm if necessary

■ Either SSTF or LOOK is a reasonable choice for the default algorithm

Storage Systems

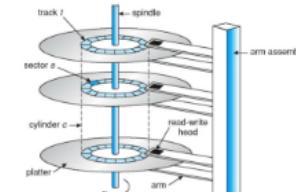
Chapter 12: Mass-Storage Systems

- Overview of Mass Storage Structure
- Disk Structure
- Disk Scheduling

Overview of Mass Storage Structure

- Magnetic disks provide bulk of secondary storage of modern computers
 - Drives rotate at 60 to 200 times per second
 - Transfer rate is rate at which data flow between drive and computer
 - Positioning time (random-access time) is time to move disk arm to desired cylinder (seek time) and time for desired sector to rotate under the disk head (latency)
- Drives can be removable
- Drive attached to computer via I/O bus
 - Buses vary, including EIDE, ATA, SATA, USB, Fibre Channel, SCSI
 - Host controller in computer uses bus to talk to disk controller built into drive or storage array

Moving-head Disk Mechanism



Disk Structure

- Disk drives are addressed as large 1-dimensional arrays of logical blocks, where the logical block is the smallest unit of transfer
- The 1-dimensional array of logical blocks is mapped into the sectors of the disk sequentially
 - Sector 0 is the first sector of the first track on the outermost cylinder
 - Mapping continues in order through that track, then the rest of the cylinders from outermost to innermost

Disk Scheduling

- The operating system is responsible for using hardware efficiently — for the disk drives, this means having a fast access time and disk bandwidth
- Access time has two major components
 - **Seek time** is the time for the disk arm to move the heads to the cylinder containing the desired sector
 - **Rotational latency** is the additional time waiting for the disk to rotate the desired sector to the disk head
- Minimize seek time
- Seek time \square seek distance
- Disk bandwidth is the total number of bytes transferred, divided by the total time between the first request for service and the completion of the last transfer

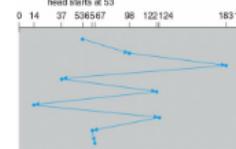
Disk Scheduling (Cont)

- Several algorithms exist to schedule the servicing of disk I/O requests
- We illustrate them with a request queue (0-189)
 - queue = 98, 183, 37, 122, 14, 124, 65, 67

Head pointer 53

FCFS

- Illustration shows total head movement of 640 cylinders
queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53



SSTF

- Selects the request with the minimum seek time from the current head position
- SSTF scheduling is a form of SJF scheduling; may cause starvation of some requests
- Illustration shows total head movement of 236 cylinders

SSTF (Cont)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53

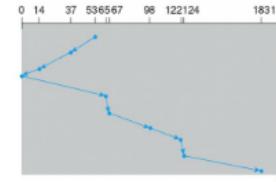


SCAN

- The disk arm starts at one end of the disk, and moves toward the other end, servicing requests until it gets to the other end of the disk, where the head movement is reversed and servicing continues.
- **SCAN algorithm**: Sometimes called the **elevator algorithm**
- Illustration shows total head movement of 236 cylinders

SCAN (Cont.)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53

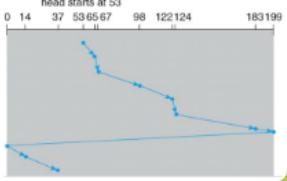


C-SCAN

- Provides a more uniform wait time than SCAN
- The head moves from one end of the disk to the other, servicing requests as it goes
 - When it reaches the other end, however, it immediately returns to the beginning of the disk, without servicing any requests on the return trip
- Treats the cylinders as a circular list that wraps around from the last cylinder to the first one

C-SCAN (Cont)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53

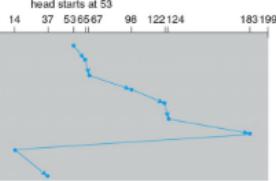


C-LOOK

- Version of C-SCAN
- Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk

C-LOOK (Cont.)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53



Selecting a Disk-Scheduling Algorithm

- SSTF is common and has a natural appeal
- SCAN and C-SCAN perform better for systems that place a heavy load on the disk
- Performance depends on the number and types of requests
- Requests for disk service can be influenced by the file-allocation method
- The disk-scheduling algorithm should be written as a separate module of the operating system, allowing it to be replaced with a different algorithm if necessary

■ Either SSTF or LOOK is a reasonable choice for the default algorithm



Chapter 12: Mass-Storage Systems

- Overview of Mass Storage Structure
- Disk Structure
- Disk Scheduling





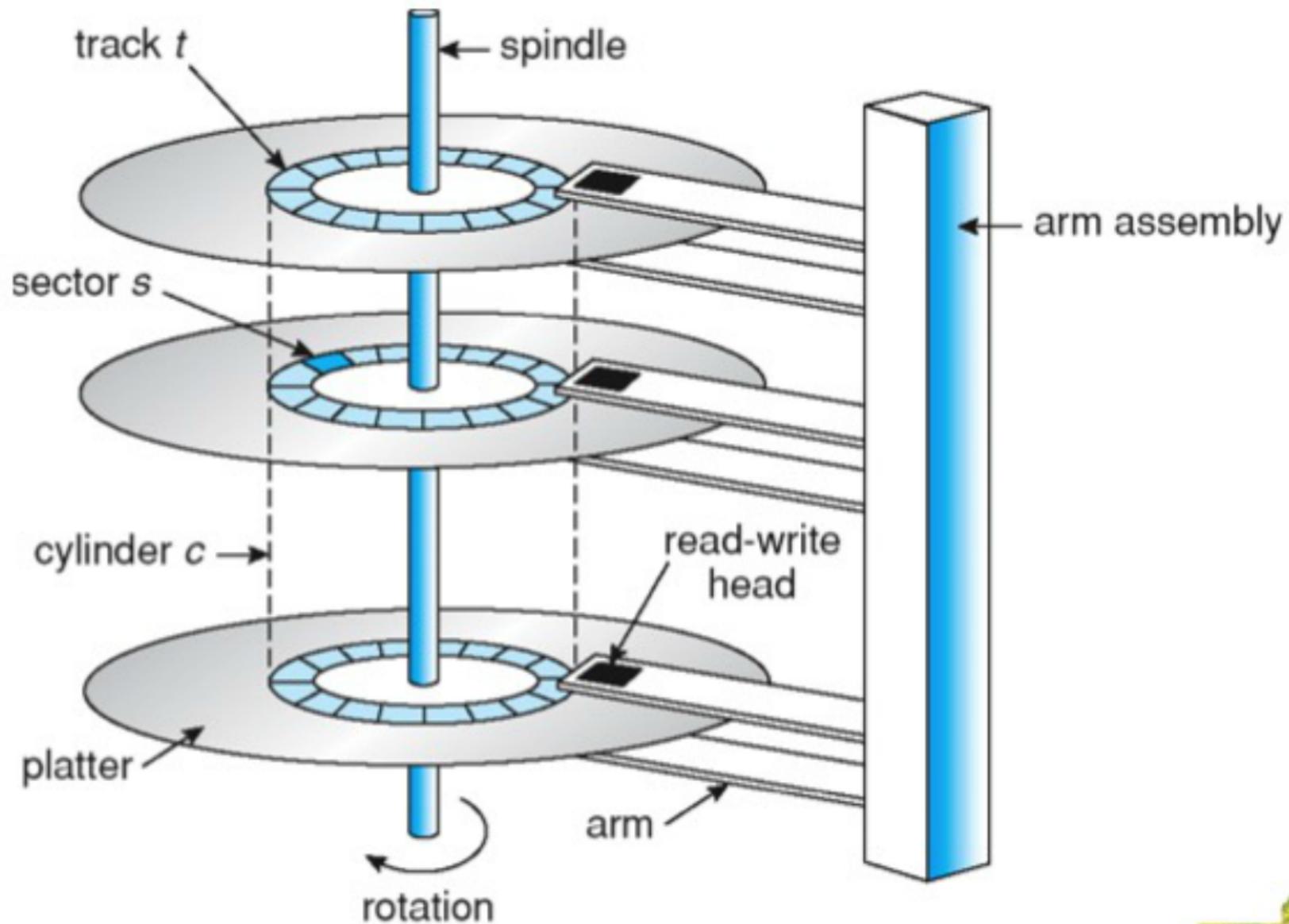
Overview of Mass Storage Structure

- Magnetic disks provide bulk of secondary storage of modern computers
 - Drives rotate at 60 to 200 times per second
 - Transfer rate is rate at which data flow between drive and computer
 - Positioning time (random-access time) is time to move disk arm to desired cylinder (seek time) and time for desired sector to rotate under the disk head (rotational latency)
- Disks can be removable
- Drive attached to computer via I/O bus
 - Buses vary, including EIDE, ATA, SATA, USB, Fibre Channel, SCSI
 - Host controller in computer uses bus to talk to disk controller built into drive or storage array





Moving-head Disk Mechanism





Disk Structure

- Disk drives are addressed as large 1-dimensional arrays of [logical blocks](#), where the logical block is the smallest unit of transfer
- The 1-dimensional array of logical blocks is mapped into the sectors of the disk sequentially
 - Sector 0 is the first sector of the first track on the outermost cylinder
 - Mapping proceeds in order through that track, then the rest of the tracks in that cylinder, and then through the rest of the cylinders from outermost to innermost





Disk Scheduling

- The operating system is responsible for using hardware efficiently — for the disk drives, this means having a fast access time and disk bandwidth
- Access time has two major components
 - **Seek time** is the time for the disk arm to move the heads to the cylinder containing the desired sector
 - **Rotational latency** is the additional time waiting for the disk to rotate the desired sector to the disk head
- Minimize seek time
- Seek time \propto seek distance
- Disk **bandwidth** is the total number of bytes transferred, divided by the total time between the first request for service and the completion of the last transfer





Disk Scheduling (Cont)

- Several algorithms exist to schedule the servicing of disk I/O requests
- We illustrate them with a request queue (0-199)

98, 183, 37, 122, 14, 124, 65, 67

Head pointer 53



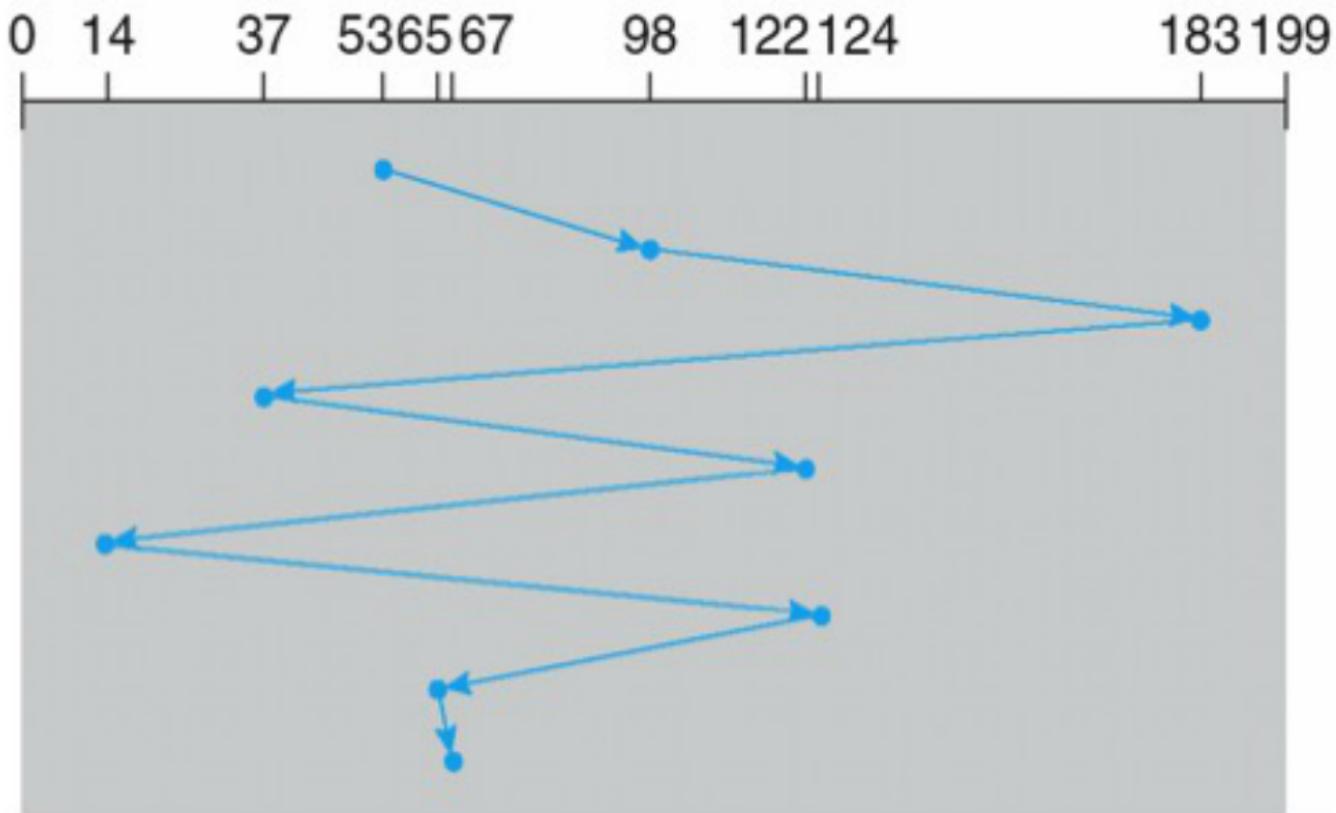


FCFS

Illustration shows total head movement of 640 cylinders

queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53





SSTF

- Selects the request with the minimum seek time from the current head position
- SSTF scheduling is a form of SJF scheduling; may cause starvation of some requests
- Illustration shows total head movement of 236 cylinders

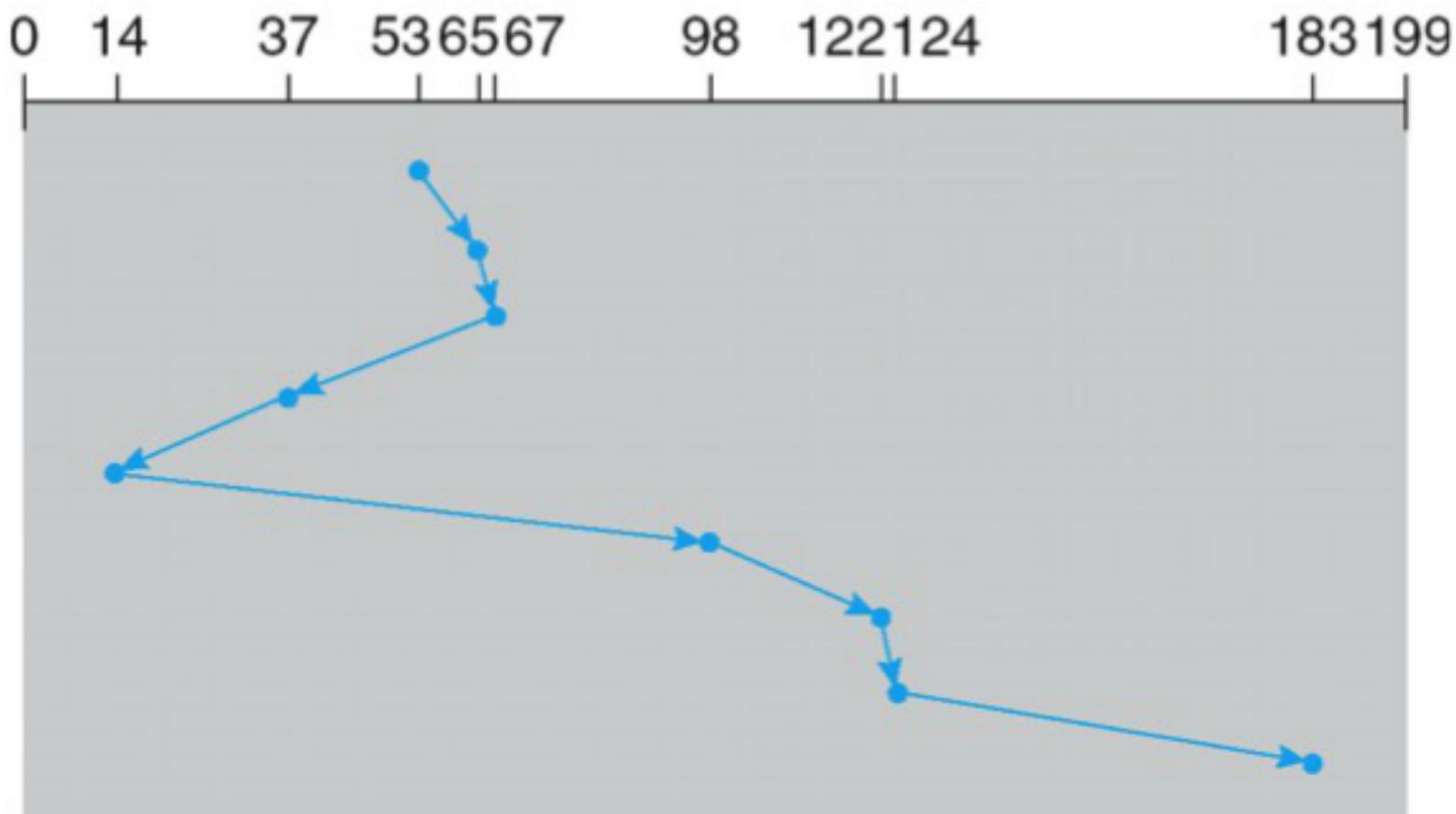




SSTF (Cont)

queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53





SCAN

- The disk arm starts at one end of the disk, and moves toward the other end, servicing requests until it gets to the other end of the disk, where the head movement is reversed and servicing continues.
- **SCAN algorithm** Sometimes called the **elevator algorithm**
- Illustration shows total head movement of 236 cylinders

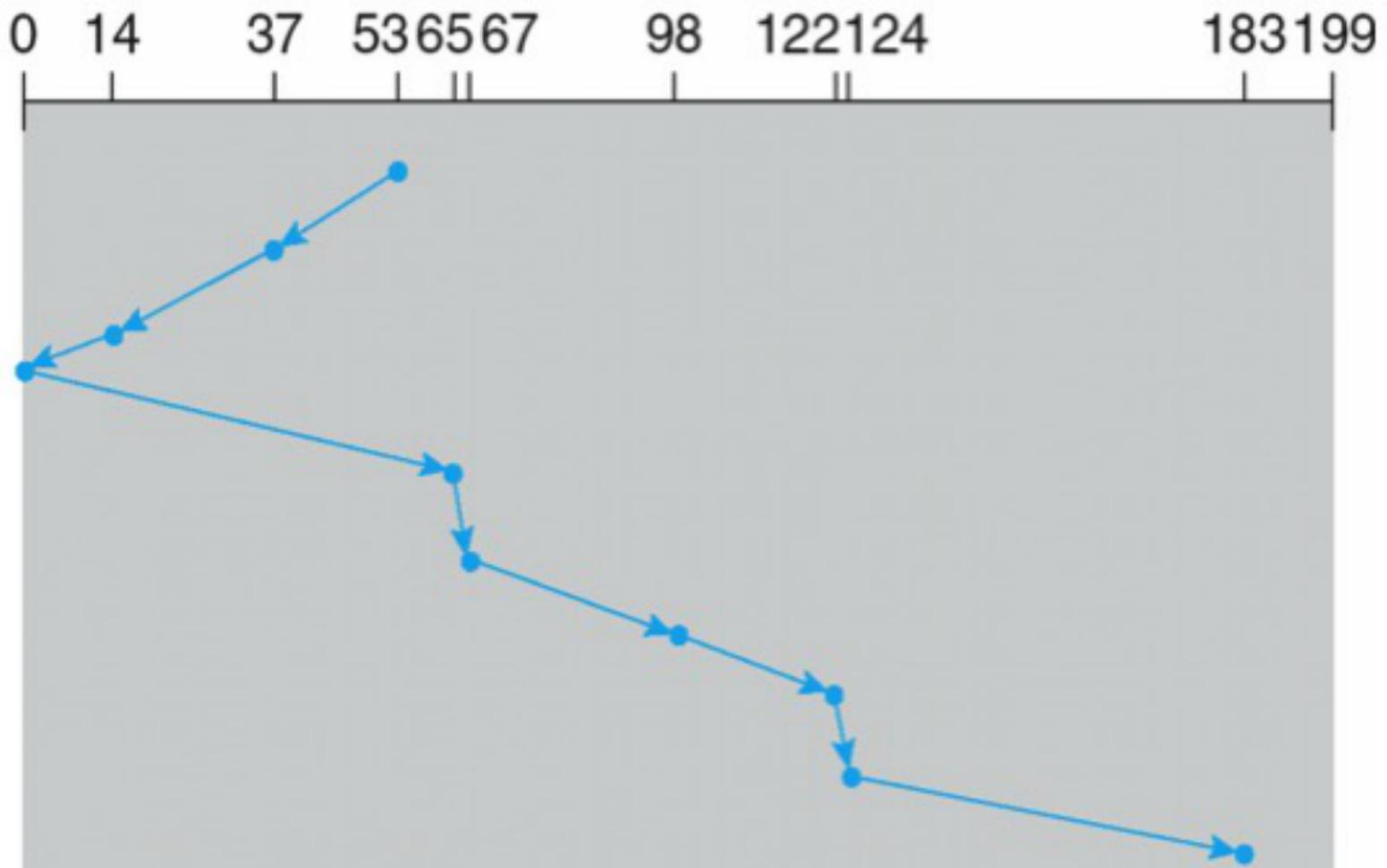




SCAN (Cont.)

queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53





C-SCAN

- Provides a more uniform wait time than SCAN
- The head moves from one end of the disk to the other, servicing requests as it goes
 - When it reaches the other end, however, it immediately returns to the beginning of the disk, without servicing any requests on the return trip
- Treats the cylinders as a circular list that wraps around from the last cylinder to the first one

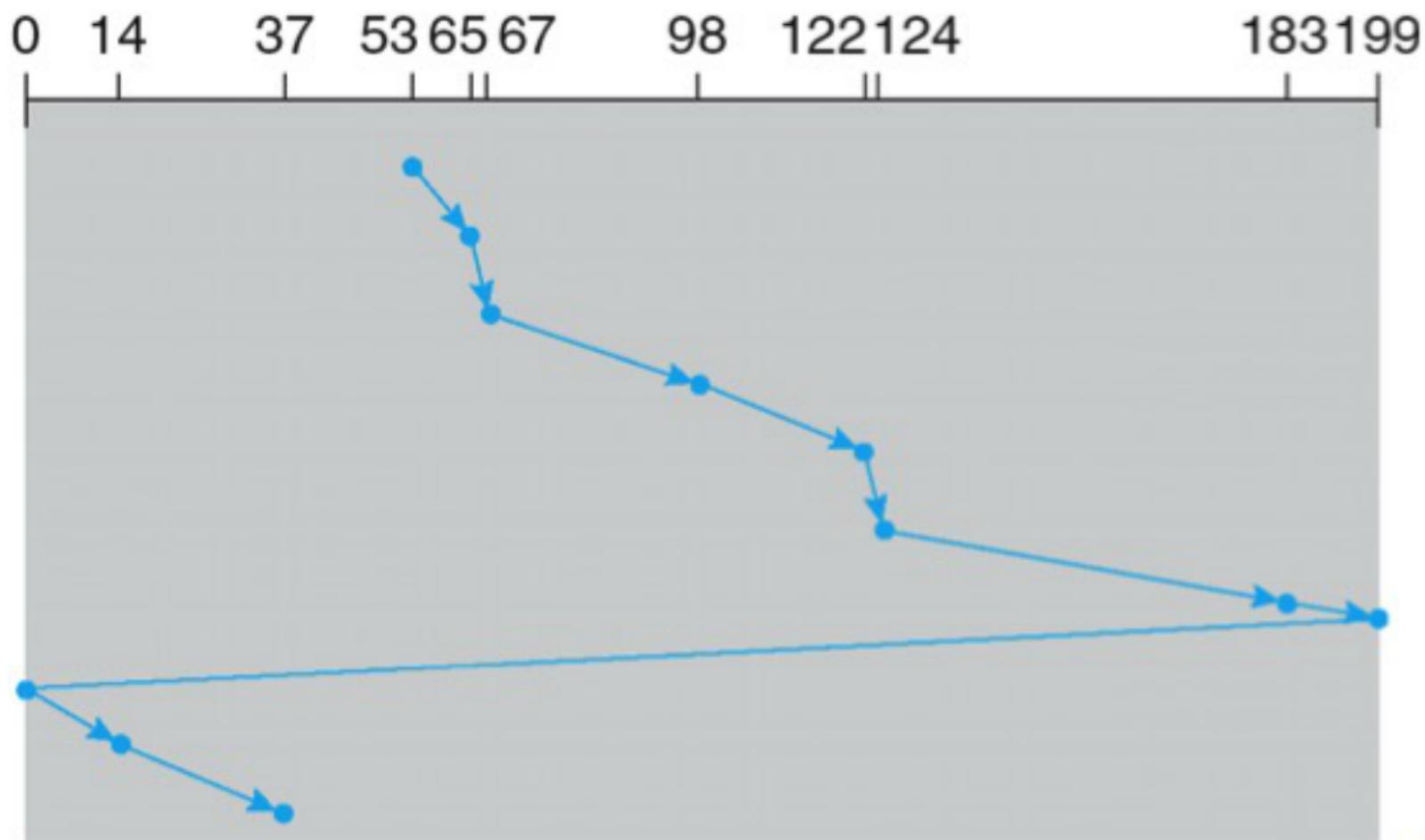




C-SCAN (Cont)

queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53





C-LOOK

- Version of C-SCAN
- Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk

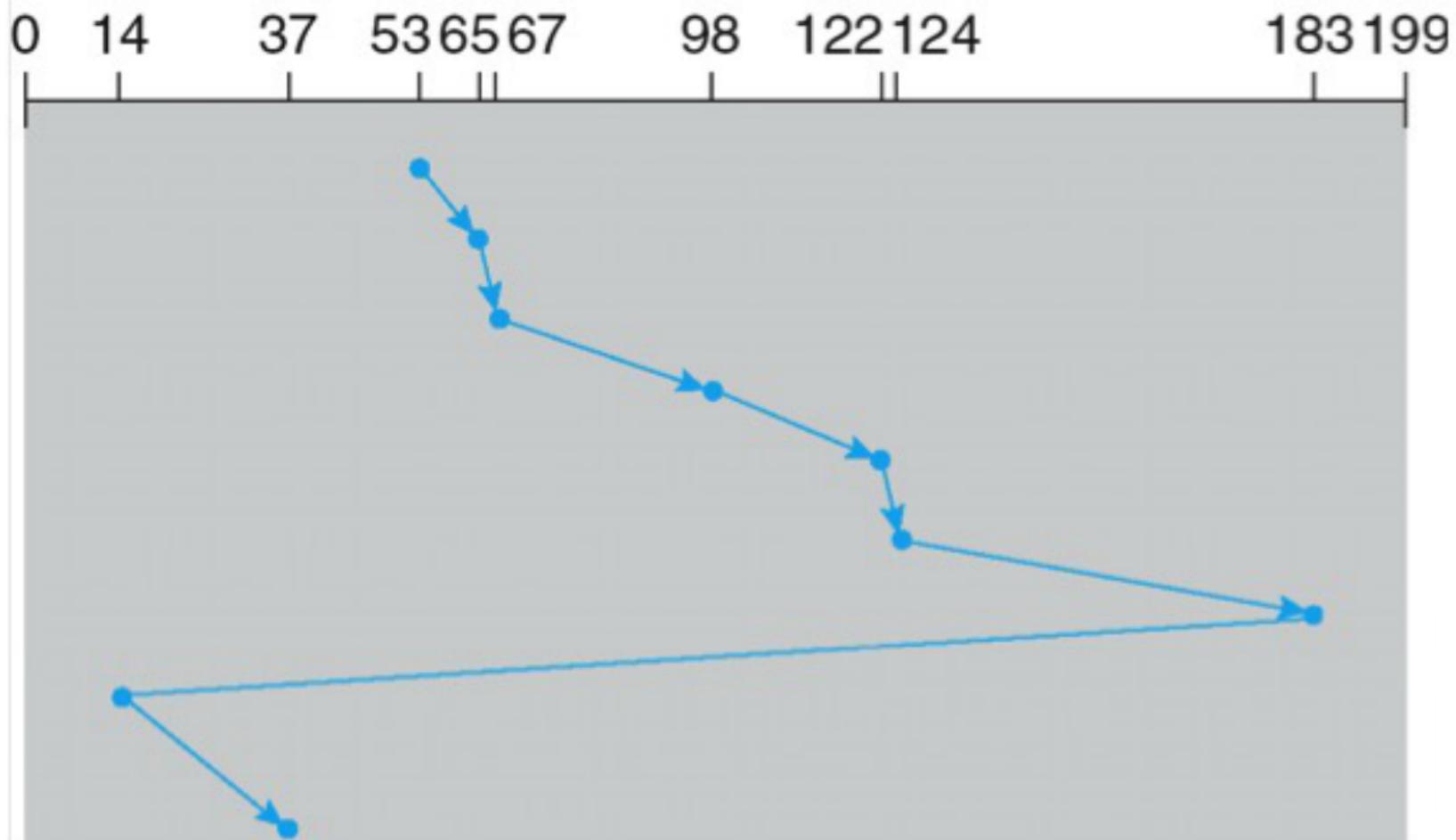




C-LOOK (Cont)

queue 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53





Selecting a Disk-Scheduling Algorithm

- SSTF is common and has a natural appeal
- SCAN and C-SCAN perform better for systems that place a heavy load on the disk
- Performance depends on the number and types of requests
- Requests for disk service can be influenced by the file-allocation method
- The disk-scheduling algorithm should be written as a separate module of the operating system, allowing it to be replaced with a different algorithm if necessary
- Either SSTF or LOOK is a reasonable choice for the default algorithm

