Systemy Operacyjne (wtorek 15.15)

Zadanie 2 - Symulacja algorytmów planowania dostępu do dysku

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Imię, nazwisko,  nr. albumu | Kryterium: | FCFS (PNL) | FCFS (PNG) | SSTF (PNL) | SSTF (PNG) | SCAN (PNL) | SCAN (PNG) | C-  SCAN (PNL) | C-  SCAN (PNG) | EDF (PNL) | EDF (PNG) | FD-  SCAN (PNL) | FD-  SCAN (PNG) |
| Fryderyk Rott 238075 | Droga głowicy | 17d 1724 | 17 71526 | 175331 902 | 3282 1501 | 1852 1702 | 1497 1509 | 1 1898 | 1061 1350 | 4025 | 4305 | 1502 | 1870 |
| Zmiany kierunku | 2 | 79 | 3 | 62 | 1 | 2 | 2 | 2 | 51 | 66 | 1 | 2 |
| “Taczka” | 0 | 0 | 0 | 0 | 62 | 43 | 75 | 42 | 0 | 0 | 78 | 63 |

Wnioski Fryderyk Rott: Ze względu na ładowność do tak zwanej “taczki”, drogę jaką pokonuje głowica oraz zmiany kierunku, najgorszy jest algorytm FCFS. Po mimo dobrego wyniku w przebytej drodze przez głowice algorytm SSTF wypada słabo ze względu na zbyt częstą zmianę kierunku co przerzuca się na niszczenie dysku. Najskuteczniejsze okazują się algorytmu typu SCAN. Potrafią “załadować na taczkę” najwięcej, ponieważ uwzględniają spotkane po drodze zgłoszenia bez priorytetów. Także zmiana kierunku głowicy jest mała co przekłada się na małe zużycie dysku.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Jakub Majchrzak 237978 | Droga głowicy | 1753 | 17940 | 901 | 1452 | 1852 | 1497 | 1753 | 1061 | 3368 | 3871 | 1165 | 1604 |
| Zmiany kierunku | 2 | 80 | 2 | 56 | 2 | 2 | 2 | 2 | 49 | 66 | 1 | 2 |
| “Taczka” | 0 | 0 | 0 | 0 | 49 | 43 | 56 | 40 | 0 | 0 | 72 | 63 |

Wnioski: Na podstawie przeprowadzonych symulacji można stwierdzić, że algorytmy typu “SCAN” dają najlepsze efekty jeśli chodzi o zużywanie głowicy (najrzadziej zmieniają kierunek ruchu). Są również najefektywniejsze, jeśli chodzi o “ładowność”. Mimo że w pewnych przypadkach algorytmów nie-SCANowych głowica pokonuje krótszą drogę, to zważywszy na zużycie sprzętu są one mało wydajne. Najlepszym z algorytmów jest FD-SCAN. Odczytuje on zgłoszenia priorytetowe, uwzględniając przy tym spotkane po drodze zgłoszenia bez priorytetów. Występuje w nim (jak również w pozostałych algorytmach windowych) zjawisko “taczki”. Najmniej wydajnym algorytmem jest FCFS, w którym głowica ma do pokonania największe odległości, a do tego, w gorszych przypadkach, bardzo silnie eksploatuje sprzęt (dużo zmian kierunku głowicy). Nie występuje w nim zjawisko “taczki”.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bartłomiej Olejniczak  238338 | Droga głowicy | 1820 | 71931 | 901 | 1491 | 1000 | 1503 | 1951 | 1598 | 24103 | 43597 | 1753 | 4393 |
| Zmiany kierunku | 1 | 79 | 0 | 1 | 0 | 1 | 1 | 2 | 53 | 58 | 1 | 4 |
| “Taczka” | 0 | 0 | 0 | 0 | 80 | 41 | 80 | 41 | 0 | 0 | 79 | 60 |

**Wnioski:**  
Najsłabszym algorytmem zdecydowanie jest FCFS, gdyż w tym algorytmie głowica pokonuje najdłuższą drogę, do tego głowica musi zmienić aż 79 razy kierunek dla danych z gorszego przypadku, co bardzo obciąża i niszczy silnik. No i oczywiście ten algorytm wykonuje tylko jeden proces naraz. Następnie można zauważyć, że w algorytmie SSTF droga głowicy jest najkrótsza jednak to nie znaczy, że ten algorytm jest najlepszy, gdyż jego dużym minusem jest to, że też wykonuje tylko jeden proces naraz. A do tego, dla jakiś specyficznych danych, może okazac się, że wystąpi duza ilosc zmian kierunków, gdyz ten algorytm zawsze szuka najbliższego znajdującego od jego obecnego położenia, następnego procesu. Dlatego zdecydowanie najlepszymi algorytmami nie priorytetowymi sa algorytmy typu Scan, które cechuje bardzo niska droga głowicy, małe obciążenie silnika, gdyż kierunek w przypadku Scana głowica zmienia tylko maksymalnie raz, a w przypadku C-scana maksymalnie dwa razy. A co najważniejsze, wykonuje on wszystkie procesy, które ma po drodze, a że głowica dzięki temu algorytmowi jest zaprogramowana tak by ruszała sie od lewej do prawej, i prawej do lewej to, gdy mamy głowicę ustawiona na starcie na 0 to rusza sie ona do maksymalnego wychylenia, po drodze wykonując wszystkie procesy. Jeśli chodzi o algorytmy priorytetowe, to zdecydowanie jest gorszy EDF, który zawsze idzie wpierw do procesu, którego deadline jest najbliżej zakończenia przez co droga jaką głowica musi pokonać jest ogromna, a do tego musi często zmieniać kierunek. Dlatego algorytm FD-SCAN jest o wiele bardziej optymalny, gdyż on w drodze do najwcześniej kończącego się deadlina bierze na “taczke” wszystkie po drodze napotkane procesy, co bardzo mocno skraca drogę jaką musi pokonać głowice, a także zmiany jej kierunku.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Imię, nazwisko,  nr. albumu | Kryterium: | FCFS (PNL) | FCFS (PNG) | SSTF (PNL) | SSTF (PNG) | SCAN (PNL) | SCAN (PNG) | C-  SCAN (PNL) | C-  SCAN (PNG) | EDF (PNL) | EDF (PNG) | FD-  SCAN (PNL) | FD-  SCAN (PNG) |
| Mateusz Zimoch 222316 | Droga głowicy | 111111 1753 | 7103  8 | 907 | 1447 | 852 1621 | 1497 1969 | 1753 | 10612 1310 | 5637 | 8825 | 1206 | 1543 |
| Zmiany kierunku | 1 | 80 | 2 | 58 | 1 | 1 | 2 | 2 | 42 | 46 | 1 | 2 |
| “Taczka” | 0 | 0 | 0 | 0 | 50 | 42 | 58 | 42 | 0 | 0 | 73 | 63 |

Wnioski: Jak widzimy, jest kilka podstawowych kryteriów doboru algorytmów. Jednym z najważniejszych, jest droga przebyta przez głowicę podczas operacji na dysku. Musi być ona jak najmniejsza, ze względu na mechaniczne zużycie się części głowicy, na przykład łożyska. Z podobnej przyczyny chcielibyśmy, aby liczba zmian kierunku głowicy była jak najmniejsza. Z racji obecnego sposobu pobierania danych (Direct Memory Access), ilość obsługiwanych procesów nie ma dla nas większego znaczenia, gdyż jednorazowo możemy obsługiwać ich praktycznie nieskończenie wiele. Z praktycznego punktu widzenia algorytm FD-SCAN jest najbardziej optymalny, gdyż obsługuje wszystkie napotkane po drodze procesy, co znacząco zmniejsza drogę głowicy i zmiany jej kierunku. Nieco gorszymi, choć podobnymi właściwościami charakteryzują się pozostałe algorytmy SCAN, których główną zaletą jest tak zwane zjawisko taczki (równoległa obsługa kilku procesów). Dość korzystnie dla pewnych danych wypada też algorytm SSTF, jednak dla “gorszego” zestawu danych wypada już bardzo blado. Dodatkowo wykonuje się tu tylko jeden proces naraz. Do najgorszych algorytmów należą algorytmy ETF i FCFS, które cechuje wiele zmian kierunku głowicy, jak i najdłuższa droga przez nią pokonywana.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Piotr  Grobelny  238076 | Droga głowicy | **1802** | 70466 | **901** | 1398 | **1402** | 1653 | **2000** | 1678 | **1932** | 69566 | **1089** | 1457 |
| Zmiany kierunku | **15** | 58 | **62** | 74 | **1** | 1 | **0** | 0 | **19** | 62 | **0** | 1 |
| “Taczka” | **0** | 0 | **0** | 0 | **51** | 44 | **58** | 45 | **0** | 0 | **72** | 63 |

Piotr Grobelny - Wnioski:

Porównując najlepsze przypadki wszystkich algorytmów z powyższego zestawienia od razu można zauważyć, że pod względem pokonanej przez głowicę drogi na pierwszym miejscu ustawia się algorytm SSTF, ale uwzględniając pozostałe kategorie natychmiast można stwierdzić, że ten algorytm nie jest najlepszym rozwiązaniem, a nawet jest jednym z gorszych, ponieważ częste zmiany kierunku doprowadzają do bardzo intensywnego eksploatowania sprzętu(mimo bardzo krótkiej drogi, którą przebywa głowica). Analizując całą tabelę widać, że algorytmy, które swoim zachowaniem podczas planowania dostępu do dysku przypominają zachowanie windy(SCAN) są w kategorii najlepszych do użycia w praktyce, ponieważ cechuje je przyzwoity czas w jakim obsługują dany proces(droga przebyta przez głowicę), są najlepszym rozwiązaniem pod kątem aspektów eksploatacyjnych, ponieważ najrzadziej zmieniają kierunek oraz występuje u nich "efekt taczki", czyli kierując się do danego procesu algorytm wykonuje wszystkie inne napotkane na swej drodze procesy, nie opóźniając wykonania procesu, do którego kieruje się głowica. Podsumowując, symulowane algorytmy można podzielić na optymalne i nieoptymalne. Do optymalnych zaliczają się algorytmy "windowe" (SCAN'owe), gdzie najlepszym rozwiązaniem jest algorytm FD-SCAN, a do nieoptymalnych zaliczają się pozostałe algorytmy, czyli: FCFS, SSTF oraz EDF, gdzie najgorszy jest FCFS.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Jarosław Skowron  238434 | Droga głowicy | 1752 | 71457 | 1562 | 2874 | 1998 | 1998 | 1998 | 3991 | 5960 | 8459 | 1246 | 1396 |
| Zmiany kierunku | 1 | 79 | 6 | 18 | 2 | 2 | 2 | 4 | 10 | 8 | 1 | 12 |
| “Taczka” | 0 | 0 | 0 | 0 | 48 | 40 | 52 | 40 | 0 | 0 | 70 | 61 |

Wnioski: Porównując algorytmy FCFS, SSTF, SCAN oraz C-SCAN, najgorzej wypada oczywiście algorytm FCFS, ponieważ obsługuje zgłoszenie w takiej kolejności w jakiej przyszły, nie patrząc na ważne kryteria, takie jak droga, czy eksploatacja głowicy, dlatego jest on bardzo mało efektywny. Jeśli chcemy osiągnąć najkrótszą przebytą przez głowicę drogę, to w większości przypadków najlepiej sprawdzi się w tym SSTF, ponieważ działa on w taki sposób, że będzie obsługiwał żądania znajdujące się najbliżej aktualnego położenia głowicy. Przy zaimplementowanych algorytmach FCFS oraz SSTF głowica przemieszczając się do kolejnego zgłoszenia nie obsługuje zgłoszeń, które pojawiają się na jej drodze i głównie z tego powodu algorytmy SCAN oraz C-SCAN będą sprawdzały się o wiele lepiej, ponieważ przeszukują cały dysk w poszukiwaniu zgłoszeń i zabierają je po kolei na “taczkę”. Ponadto w algorytmach SCAN i C-SCAN głowica będzie zmieniać kierunek tylko wtedy, gdy dojdzie do krawędzi dysku, czyli o wiele rzadziej, niż w FCFS oraz SSTF, przez co będzie ona znacznie mniej eksploatowana, dzięki czemu nie zużyje się tak szybko. Dobrze widać to w najgorszym przypadku ciągu żądań, gdzie zmiana kierunku głowicy dla SCAN oraz C-SCAN wynosi tylko 2, dla SSTF 18, a dla FCFS aż 79. Porównując algorytmy stosowane w systemach czasu rzeczywistego, FD-SCAN wypada znacznie lepiej niż EDF, który obsługuje najpierw żądania z najkrótszym deadlinem, przez co głowica musi często przebyć o wiele dłuższą drogę. Ponadto EDF, podobnie jak FCFS oraz SSTF, nie obsługuje żądań, które głowica spotyka na swojej drodze, poruszając się w kierunku wyznaczonego żądania. FD-SCAN je obsługuje, przez co jest o wiele bardziej efektywnym algorytmem.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Mateusz  Pakuła  238336 | Droga głowicy | 1753 | 71586 | 901 | 1460 | 1816 | 1486 | 2014 | 1356 | 2721 | 3205 | 1217 | 1445 |
| Zmiany kierunku | 1 | 79 | 2 | 54 | 1 | 2 | 2 | 4 | 53 | 57 | 0 | 1 |
| “Taczka” | 0 | 0 | 0 | 0 | 49 | 42 | 54 | 42 | 0 | 0 | 72 | 63 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Rafał Szewczyk  238486 | Droga głowicy | 1753 | 71560 | 901 | 1431 | 1807 | 1453 | 2014 | 1356 | 2721 | 3419 | 1198 | 1445 |
| Zmiany kierunku | 1 | 79 | 2 | 55 | 1 | 2 | 2 | 4 | 53 | 59 | 0 | 1 |
| “Taczka” | 0 | 0 | 0 | 0 | 49 | 43 | 54 | 42 | 0 | 0 | 72 | 63 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bartosz Bujak  238470 | Droga głowicy | 1753 | 71469 | 1581 | 1631 | 1753 | 1491 | 2080 | 2459 | 1753 | 71469 | 1541 | 1491 |
| Zmiany kierunku | 1 | 79 | 2 | 3 | 1 | 1 | 6 | 6 | 1 | 79 | 3 | 1 |
| “Taczka” | 0 | 0 | 0 | 0 | 49 | 41 | 54 | 41 | 0 | 0 | 72 | 63 |

Wnioski: Na podstawie przeprowadzonych symulacji można stwierdzić że najsłabszym algorytmem spośród testowanych jest algorytm FCFS. Wypada najgorzej we wszystkich kryteriach, zwłaszcza dla przypadków najgorszych. Algorytmy SSTF, SCAN, C-SCAN, dziś już nie wykorzystywane, oferują znaczną poprawę zarówno w czasie obsługi, czyli w drodze jaką musi pokonać igła by odczytać dane, ale również redukują liczbę zmian kierunku obsługi co przekłada się na wydłużenie żywotności dysku. Algorytm EDF działa niemal identycznie z algorytmem FCFS, dodając do tego obsługę zgłoszeń priorytetowych. Najlepszy jest algorytm FD-SCAN, szybko i w pierwszej kolejności obsługuje procesy priorytetowe, ale po drodze do nich odczytuje także inne dane. Algorytm ten najefektywniej wykorzystuje szybkie łącza współczesnych dysków twardych poprzez opisane zjawisko ładowania procesów na „taczke”. Jest także optymalny pod względem pokonywanej drogi i zmian kierunku. Te parametry nawet w najgorszych przypadkach są sporo niższe w porównaniu do pozostałych algorytmów.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Svitlana Babinska  220351 | Droga głowicy | 1842 | 7193 | 940 | 1825 | 1842 | 1423 | 1984 | 1583 | 3207 | 4004 | 1021 | 1572 |
| Zmiany kierunku | 50 | 57 | 59 | 69 | 1 | 1 | 0 | 0 | 61 | 71 | 0 | 1 |
| “Taczka” | 0 | 0 | 0 | 0 | 49 | 43 | 56 | 41 | 0 | 0 | 71 | 68 |

Jeśli chodzi o zużycie iglicy to najlepiej siebie pokazuje algorytm SCAN ponieważ najmniej zmienia kierunek ruchu głowicy. Także SCAN jest optymalnym jeśli chodzi o ładowanie bufora. Jeśli rozważać algorytmy dla aplikacji real-time to najlepiej siebie pokazuje algorytm FD-SCAN. Odczytuje on zgłoszenia priorytetowe, uwzględniając przy tym spotkane po drodze zgłoszenia bez priorytetów. Takie rozwiązanie pozwala na zmniejszenie do minimum ilość zmian kierunku głowicy. Najgorszym algorytmem według wyników jest FCFS.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Rafał Kosyl  238070 | Droga głowicy | 1893 | 73707 | 937 | 1517 | 1897 | 1511 | 2095 | 1424 | 2966 | 3487 | 1294 | 1532 |
| Zmiany kierunku | 1 | 88 | 2 | 62 | 1 | 2 | 2 | 4 | 58 | 64 | 0 | 1 |
| “Taczka” | 0 | 0 | 0 | 0 | 57 | 51 | 63 | 48 | 0 | 0 | 83 | 76 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Marcin Makuch  238063 | Droga  głowicy | 1755 | 71459 | 964 | 1521 | 1869 | 1869 | 2033 | 1327 | 2458 | 4587 | 1128 | 1189 |
| Zmiany  kierunku | 1 | 81 | 5 | 52 | 1 | 2 | 2 | 4 | 56 | 60 | 0 | 1 |
| “Taczka” | 0 | 0 | 0 | 0 | 50 | 41 | 57 | 45 | 0 | 0 | 84 | 73 |

Wnioski: Algorytmem wypadającym najgorzej pod względem wszystkich uwzględnionych kryteriów jest algorytm FCFS. Najbardziej efektywne pod względem małego wykorzystania głowicy są algorytmy typu “SCAN” (kierunek ruchu głowicy jest zmieniany najrzadziej). Są one także najbardziej efektywne co do ich “ładowności” danych. Pomimo tego, że w algorytmie SSTF droga pokonywana przez głowicę jest najkrótsza, zużycie głowicy jest większe, gdyż musi ona bardzo często zmieniać kierunek poruszania się. Z powodu wykorzystywania w dzisiejszych czasach techniki Direct Memory Access do pobierania danych, ilość obsługiwanych przez nas procesów nie ma znaczenia i jest pomijalna. Z tego właśnie powodu decydującym kryterium przy doborze optymalnego algorytmu jest zużywanie się głowicy i jej mechanicznych części. Biorąc to pod uwagę, najbardziej efektywnym algorytmem jest w tym wypadku FD-SCAN. Jest to algorytm priorytetowy, jednak obsługujący “po drodze” wrzucając na “taczkę” procesy bez zwiększonego priorytetu co pozwala nam na jak najrzadsze zmienianie kierunku głowicy i skrócenie jej drogi.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Konrad  Żołyński  222364 | Droga głowicy | 1943 | 72615 | 977 | 2099 | 1867 | 1920 | 2016 | 1701 | 2486 | 4967 | 1119 | 1172 |
| Zmiany  kierunku | 1 | 74 | 8 | 50 | 1 | 1 | 2 | 4 | 37 | 51 | 0 | 1 |
| “Taczka” | 0 | 0 | 0 | 0 | 52 | 43 | 52 | 39 | 0 | 0 | 81 | 68 |

Wnioski: Bezsprzecznie najlepszą funkcjonalnością wykazuje się algorytm SCAN, który sprawia, że otrzymujemy najmniejsze zużycie głowicy, najmniejszą ilość zwrotów a także największą żywotność łożysk głowicy. Bardzo dobrze wypada również pod kątem pakowności na tzw. "taczkę". Najgorzej wypada algorytm FCFS. Algorytm SSTF powoduje za to największe zużycie głowicy - zgodnie z tym co mówił Pan Doktor - głowica ma momenty w których lata z wysoką częstotliwością zwrotów po niewielkim obszarze pamięci, co powoduje zajeżdżenie głowicy. Jest to niedopuszczalne ze względu na charakterystykę pamięci, gdzie pojemność jakby bufora nie stanowi dla nas żadnej przeszkody, a liczy się tylko uciążliwa w niektórych przypadkach eksploatacja części mechanicznych jak łożyska głowicy. To sprawia, że najbardziej interesującym nas algorytmem pod względem optymalizacji tych parametrów jest algorytm priorytetowy FD-SCAN, który zbiera wszystko po drodze na "taczkę".

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Krzysztof  Dziubek  238347 | Droga głowicy | 1764 | 71614 | 988 | 2138 | 1936 | 1936 | 2053 | 1687 | 2465 | 4564 | 1148 | 1162 |
| Zmiany  kierunku | 1 | 75 | 10 | 47 | 1 | 1 | 2 | 4 | 36 | 54 | 0 | 1 |
| “Taczka” | 0 | 0 | 0 | 0 | 52 | 44 | 57 | 42 | 0 | 0 | 78 | 67 |

Krzysztof Dziubek wnioski: FCFS to najgorzej wypadający algorytm planowania dostępu do dysku, najdłuższa przebyta droga, ogromna ilość zmian kierunku i brak efektu "taczki", który występuje przy algorytmach typu SCAN. Był to pierwszy, najbardziej pierwotny algorytm, który trzeba było zmienić. SSTF jest algorytmem nieco lepszym, niestety droga przebyta przez głowicę nadal jest spora. Jednak wciąż największą wadą tych dwóch rozwiązań jest zbyt częsta zmiana kierunku spowodowana naturą tych koncepcji. Jest to duży minus ponieważ duża ilość zmian kierunku może uszkodzić mechanizmy poruszające głowicą. EDF jest to algorytm priorytetowy, co jak widać nie służy głowicy, ponieważ stwarza sytuacje ,w których głowica często zmienia swój kierunek. Ilość tych zmian jak i brak efektu "taczki" jest wykluczająca. Algorytmy typu "SCAN" są bardziej optymalnymi rozwiązaniami. Przebyta droga w obu sytuacjach (w najlepszym przypadku i najgorszym) jest porównywalna, różnice nie są zauważalne. Algorytmy typu "SCAN" muszą swoje przybyć, ale nie nadwyrężają głowicy zmianami kierunku. Ilość zmian w najgorszym przypadku waha się w granicach od 1 do 4, co w porównaniu do reszty algorytmów (nie typu "SCAN") jest wspaniałym wynikiem. Również w tego typu algorytmach występuje efekt "taczki", to pozwala zbierać i wysyłać dane w większej ilości naraz. Najlepiej wypadającym algorytmem jest FD-SCAN. Jest to algorytm, który zmierza do odwołania z największym priorytetem zbierając po drodze dane na "taczkę". Takie rozwiązanie pozwala na zmniejszenie do minimum ilość zmian kierunku głowicy. Droga przebyta jest najmniejszą ze wszystkich, a efekt "taczki" jest najbardziej widoczny, co pozwala przesyłanie danych dużymi porcjami wykorzystując chociaż część potencjału "buforów", która jest spora w porównaniu do wykorzystania.