

2020 Operating System Project 1 - Process Scheduling

by R07922148 鄭嘉賡

Design

使用兩顆 CPU, CPU 0 用作排程, CPU 1 用來跑子程式.

main

讀取 input data,將每個 process 按照 ready time 排列

建立 child processes,並為其分配 CPU 和設定 priority

當有 process 在跑的時候,check 是否已經完成

檢查是否有 process 已經 ready 準備要運行

將 CPU 交給下一個要運行的 Process,如果換人就要 content switch

process 運行完一個 Unit time 之後 process 剩下的時間減去 Unit time

重複步驟直到跑完所有 Processes

Process

讀取 input data,將每個 process 按照 ready time 排列

建立 child processes,並為其分配 CPU 和設定 priority

Queue

為 Round-Robin 規則下,run 完的 process 如果還要運行就要重新排隊,每一個重新排隊的都需要排在 Queue 最後.

Scheduling

按照四種不同的 scheduling policy 進行排程,從而得知下一個要運行的 child

process

System Calls

兩個自己的 syscalls，需要放入 linux/kernel 資料夾與 kernel 一同編譯。

334 my_gettime

335 my_dprint

用 getnstimeofday 來讀取當前時間戳 asmlinkage long sys_my_gettime(void) { static

```
const long giga = 1000000000; struct timespec _t; getnstimeofday(&_t); return  
(_t.tv_sec*giga + _t.tv_nsec); }
```

用來 printkl 來 print dmesg asmlinkage void sys_my_dprint(int pid,long start_time,

```
long end_time) { static const long giga = 1000000000; printk("[Project1] %d  
%ld.%09ld %ld.%09ld\n", pid, start_time / giga, start_time % giga, end_time / giga,  
end_time % giga); return; }
```

Experiment and Results

Environment Platform OS Kernel Virtualbox 6.1 on win10

Ubuntu 16.04 Linux 4.14.25

First In First Out (FIFO)

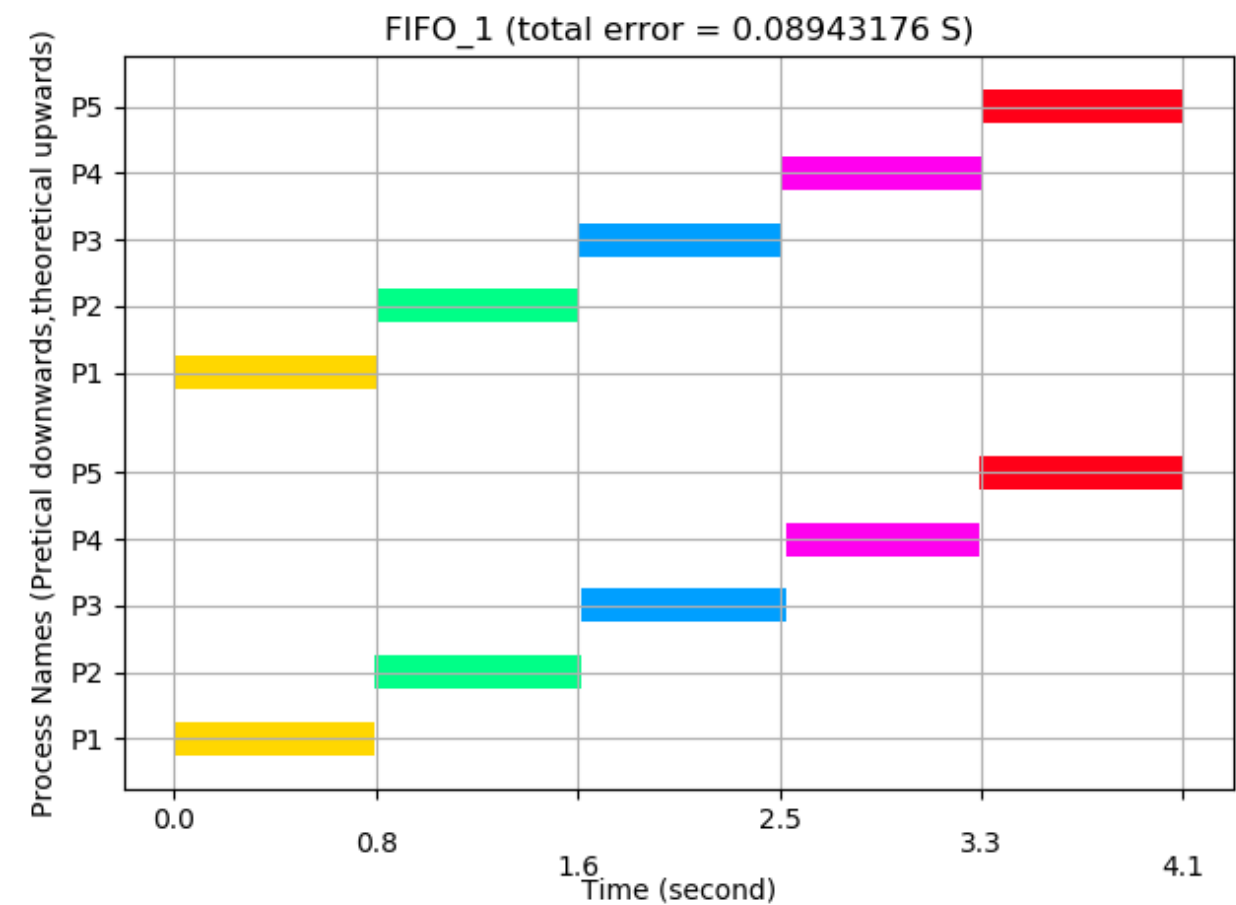
Test Data 1

FIFO 5 P1 0 500 P2 0 500 P3 0 500 P4 0 500 P5 0 500

Exclusion Result

```
[ 3999.001639] [Project1] 2647 1588168462.496226788 1588168463.313589004  
[ 3999.839055] [Project1] 2648 1588168463.313712051 1588168464.150971127  
[ 4000.675117] [Project1] 2649 1588168464.151094442 1588168464.986999186  
[ 4001.525969] [Project1] 2650 1588168464.987171502 1588168465.837815910  
[ 4002.342739] [Project1] 2651 1588168465.838098292 1588168466.654552204
```

Comparison of Theoretical and Pretical Results



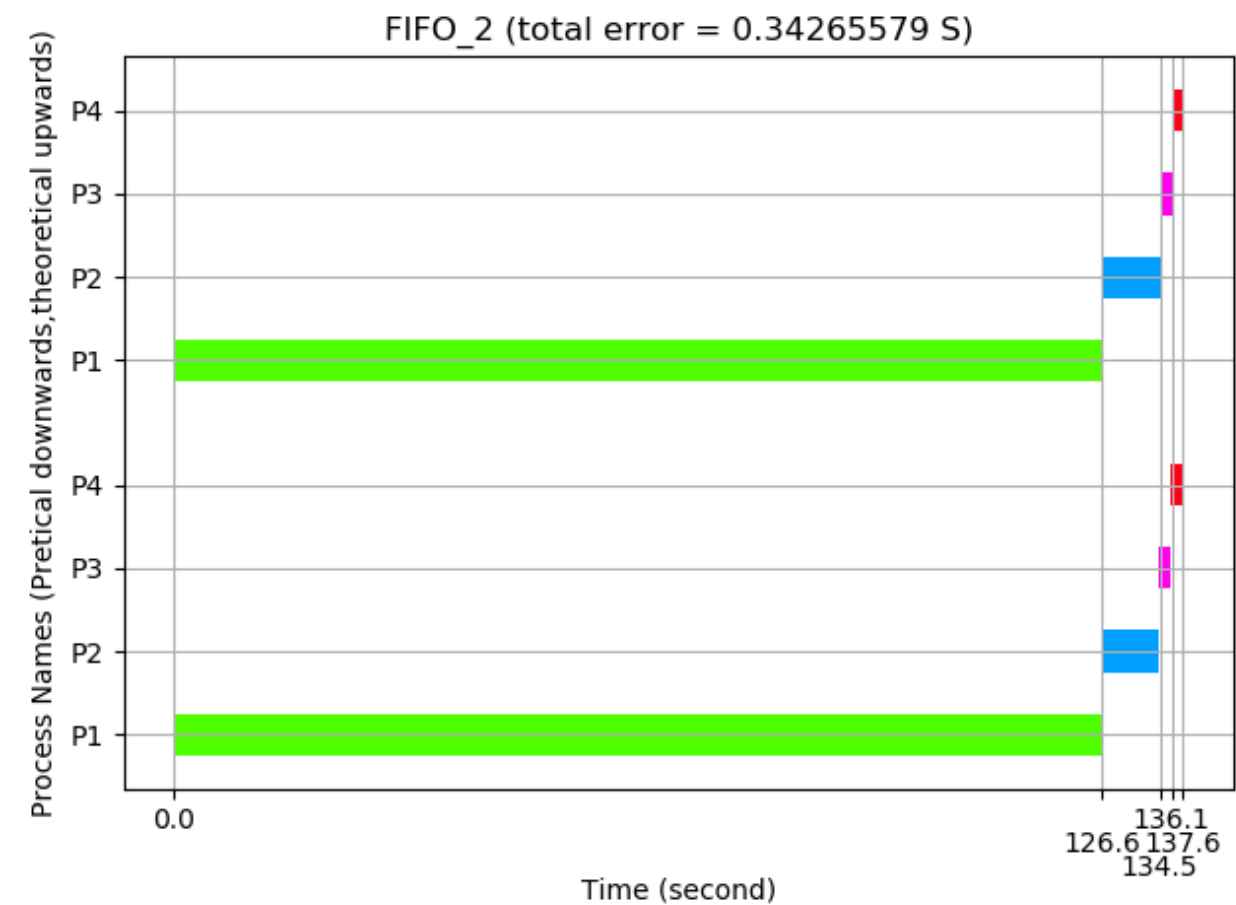
Test Data 2

FIFO 4 P1 0 80000 P2 100 5000 P3 200 1000 P4 300 1000

Exclusion Result

[4137.248056] [Project1] 2659 1588168466.665424981 1588168601.554405701
[4145.349248] [Project1] 2660 1588168601.554547696 1588168609.655599126
[4146.967657] [Project1] 2661 1588168609.655724017 1588168611.274006429
[4148.611852] [Project1] 2662 1588168611.274133536 1588168612.918200852

Comparison of Theoretical and Pretical Results



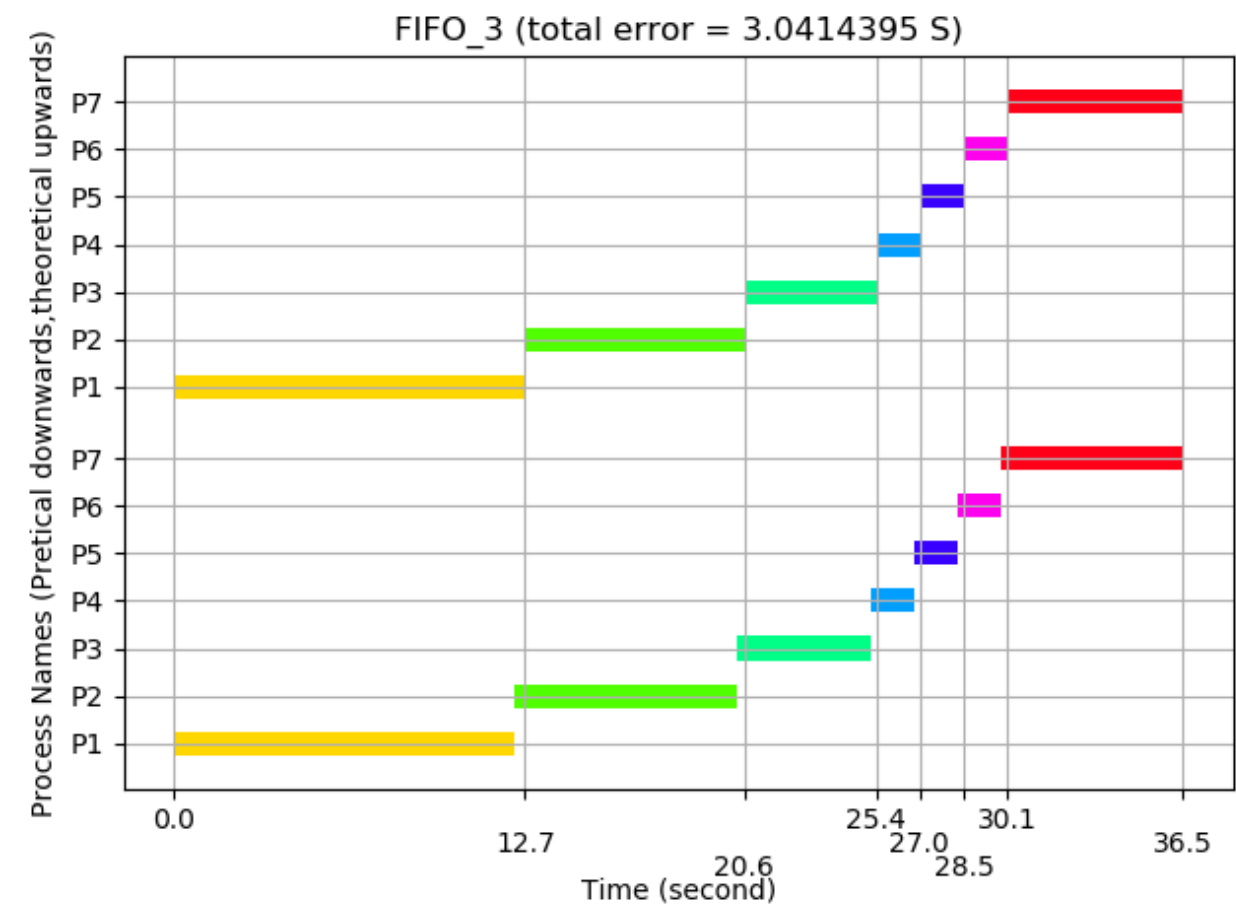
Test Data 3

FIFO 7 P1 0 8000 P2 200 5000 P3 300 3000 P4 400 1000 P5 500 1000 P6 500 1000 P7 600 4000

Exclusion Result

[4162.329934] [Project1] 2676 1588168612.929549667 1588168626.636273411
[4170.374733] [Project1] 2677 1588168626.636408431 1588168634.681060230
[4175.380045] [Project1] 2678 1588168634.681185265 1588168639.686362470
[4177.032926] [Project1] 2679 1588168639.686498069 1588168641.339239132
[4178.705258] [Project1] 2680 1588168641.339363096 1588168643.011568446
[4180.369589] [Project1] 2681 1588168643.011691793 1588168644.675895937
[4187.096760] [Project1] 2682 1588168644.676043260 1588168651.403048890

Comparison of Theoretical and Pretical Results



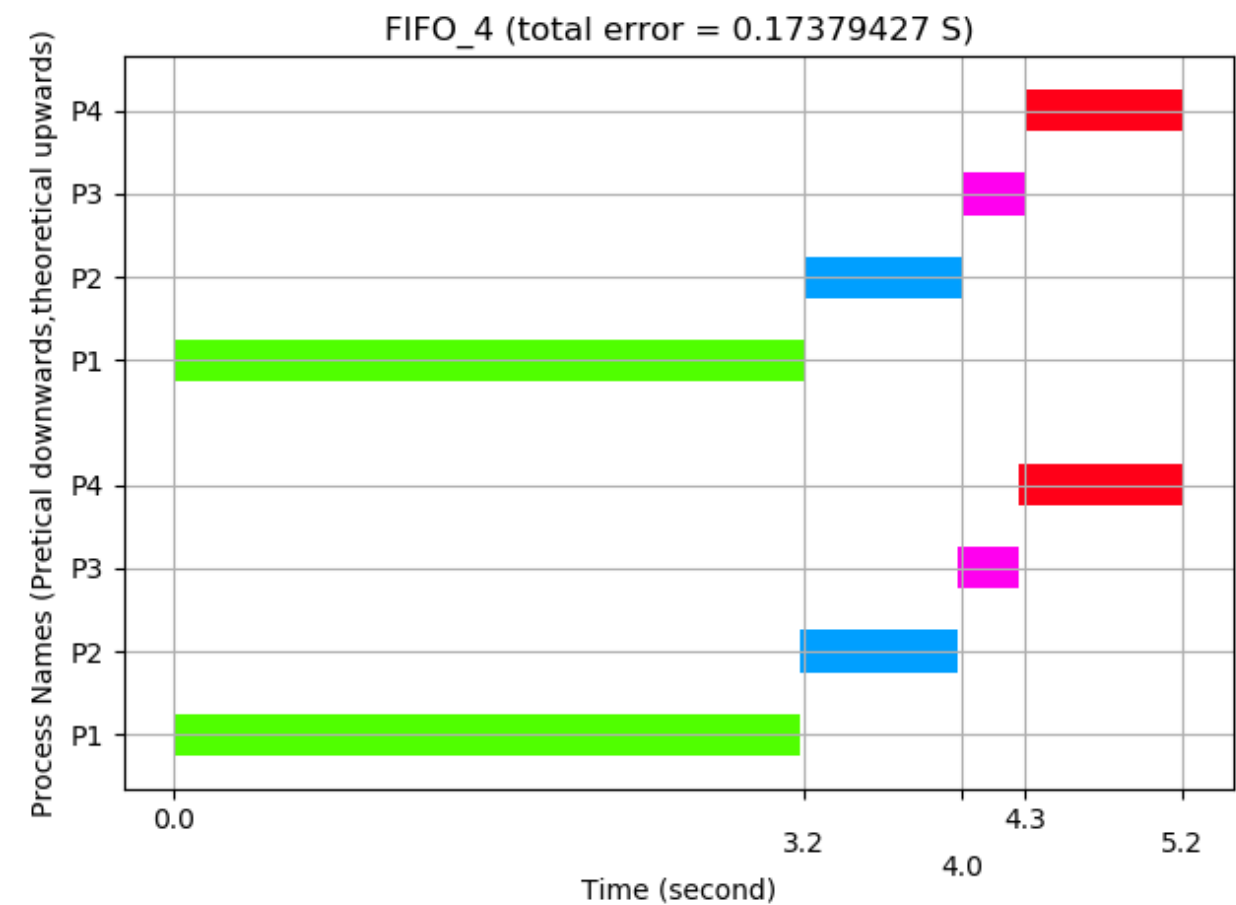
Test Data 4

FIFO 4 P1 0 2000 P2 500 500 P3 500 200 P4 1500 500

Exclusion Result

[4190.297946] [Project1] 2690 1588168651.414808157 1588168654.604226191
[4191.091461] [Project1] 2691 1588168654.604343397 1588168655.397738641
[4191.412277] [Project1] 2692 1588168655.397894474 1588168655.718553744
[4192.220176] [Project1] 2693 1588168655.718673694 1588168656.526449525

Comparison of Theoretical and Pretical Results



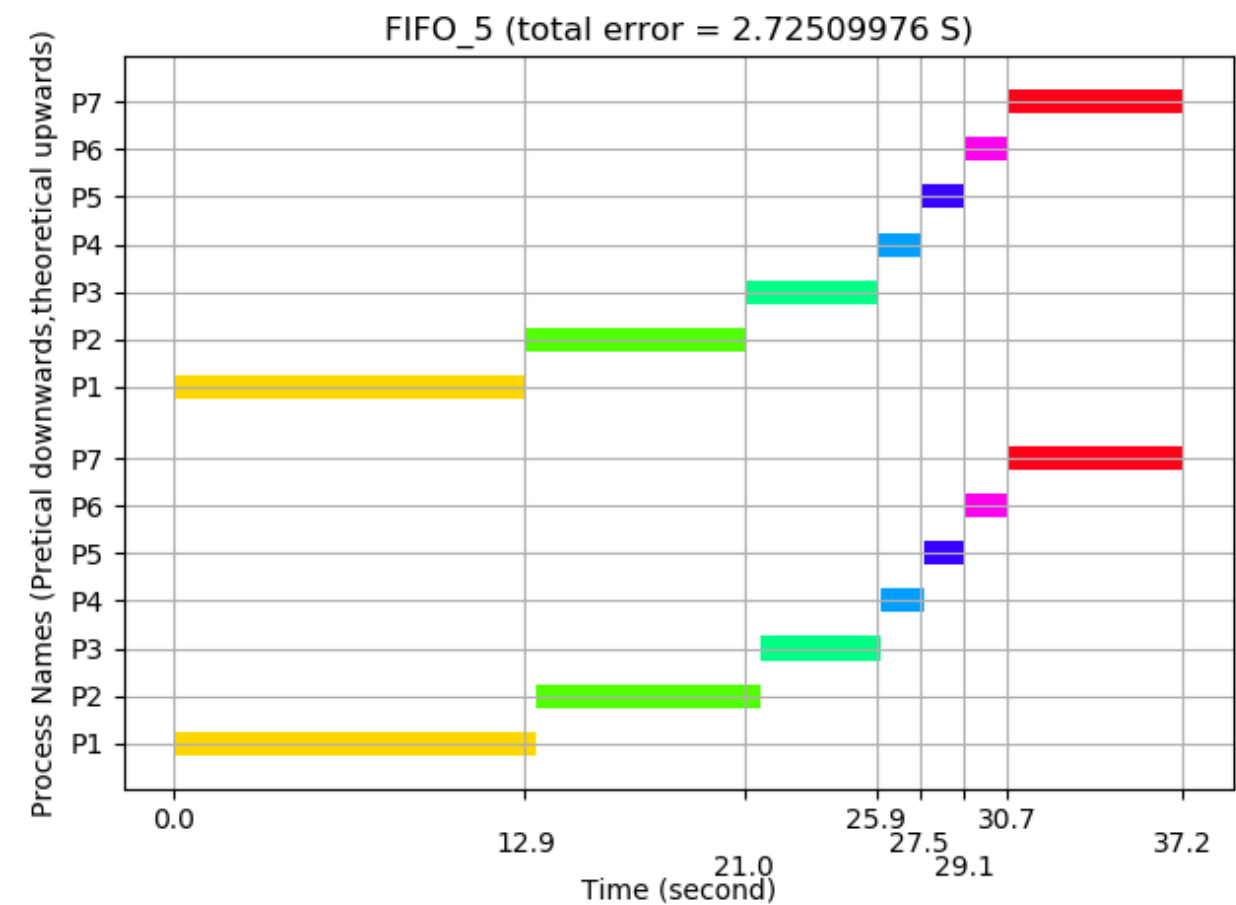
Test Data 5

FIFO 7 P1 0 8000 P2 200 5000 P3 200 3000 P4 400 1000 P5 400 1000 P6 600 1000 P7 600 4000

Exclusion Result

[4205.588340] [Project1] 2701 1588168656.538213131 1588168669.894567641
[4213.891166] [Project1] 2702 1588168669.894703158 1588168678.197359122
[4218.449547] [Project1] 2703 1588168678.197513418 1588168682.755719731
[4220.153260] [Project1] 2704 1588168682.755861868 1588168684.459423697
[4221.889144] [Project1] 2705 1588168684.459548296 1588168686.195299602
[4223.546254] [Project1] 2706 1588168686.195427609 1588168687.852402136
[4229.800881] [Project1] 2707 1588168687.852524843 1588168694.106996663

Comparison of Theoretical and Pretical Results



RR (Round-Robin)

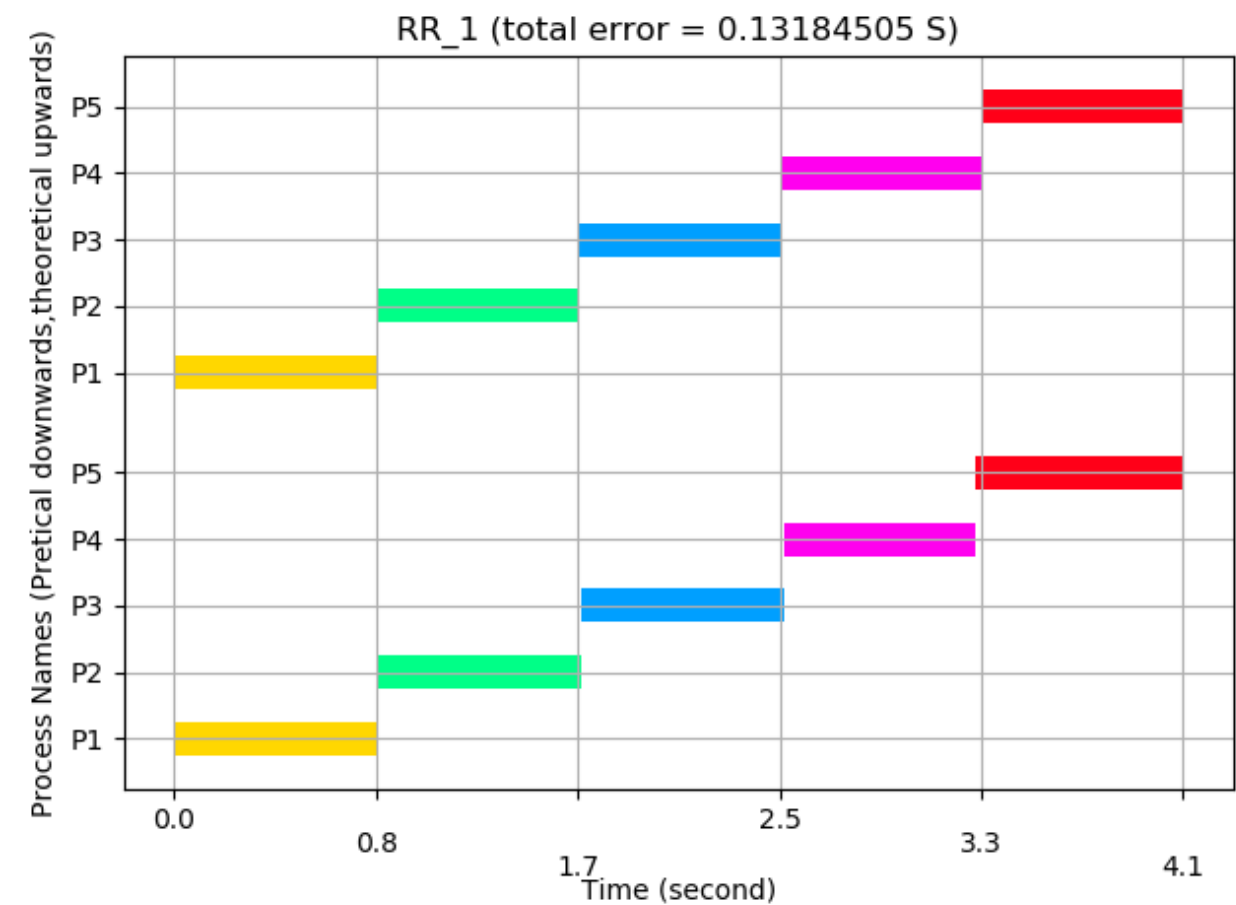
Test Data 1

RR 5 P1 0 500 P2 0 500 P3 0 500 P4 0 500 P5 0 500

Exclusion Result

[3617.034446] [Project1] 2454 1588168080.535052288 1588168081.361916858
[3617.834567] [Project1] 2455 1588168081.362056980 1588168082.162005850
[3618.644825] [Project1] 2456 1588168082.162430872 1588168082.972230385
[3619.396872] [Project1] 2457 1588168082.972363093 1588168083.724246657
[3620.206140] [Project1] 2458 1588168083.724532296 1588168084.533482284

Comparison of Theoretical and Pretical Results



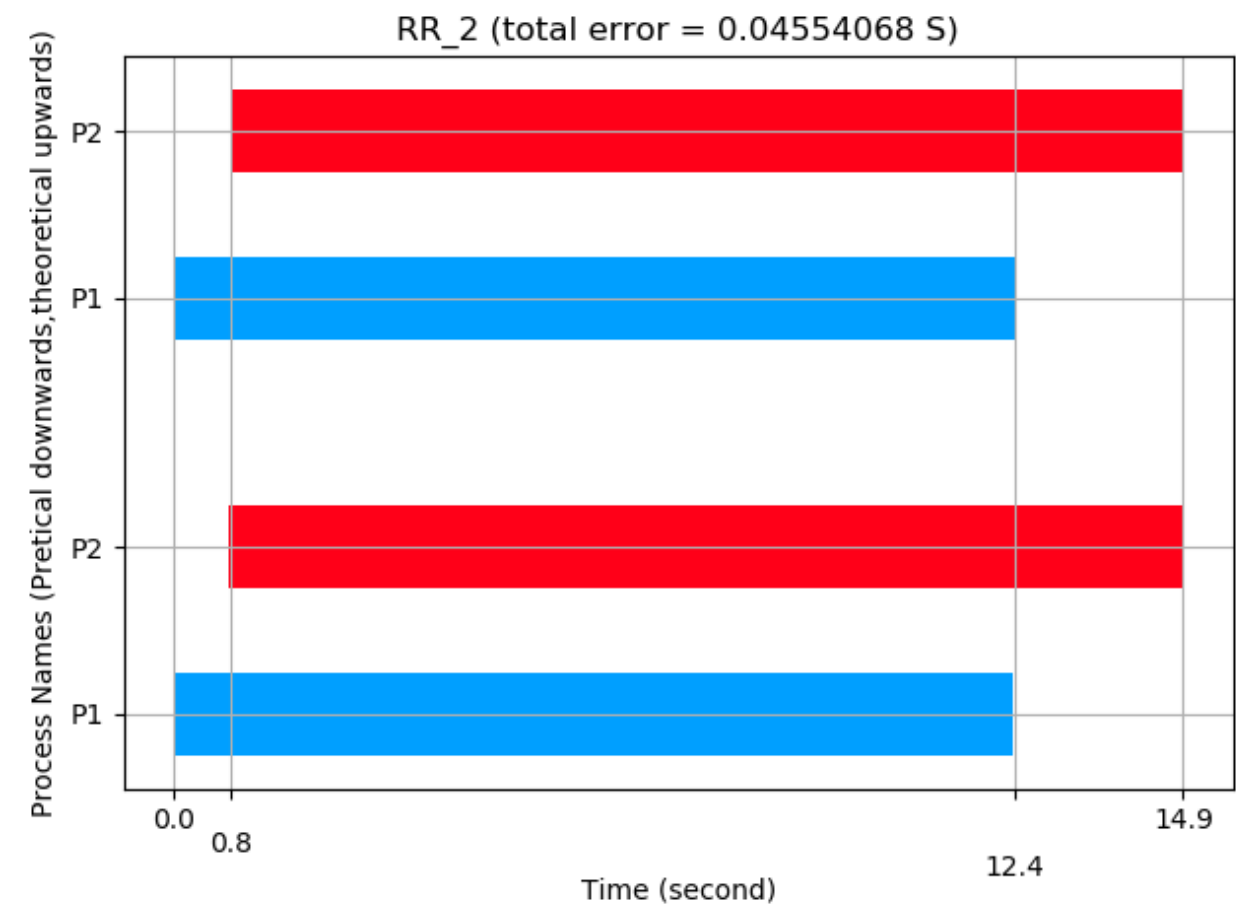
Test Data 2

RR 2 P1 600 4000 P2 800 5000

Exclusion Result

[3633.079858] [Project1] 2467 1588168085.417022320 1588168097.406678115
[3635.527349] [Project1] 2468 1588168086.158480668 1588168099.854069875

Comparison of Theoretical and Pretical Results



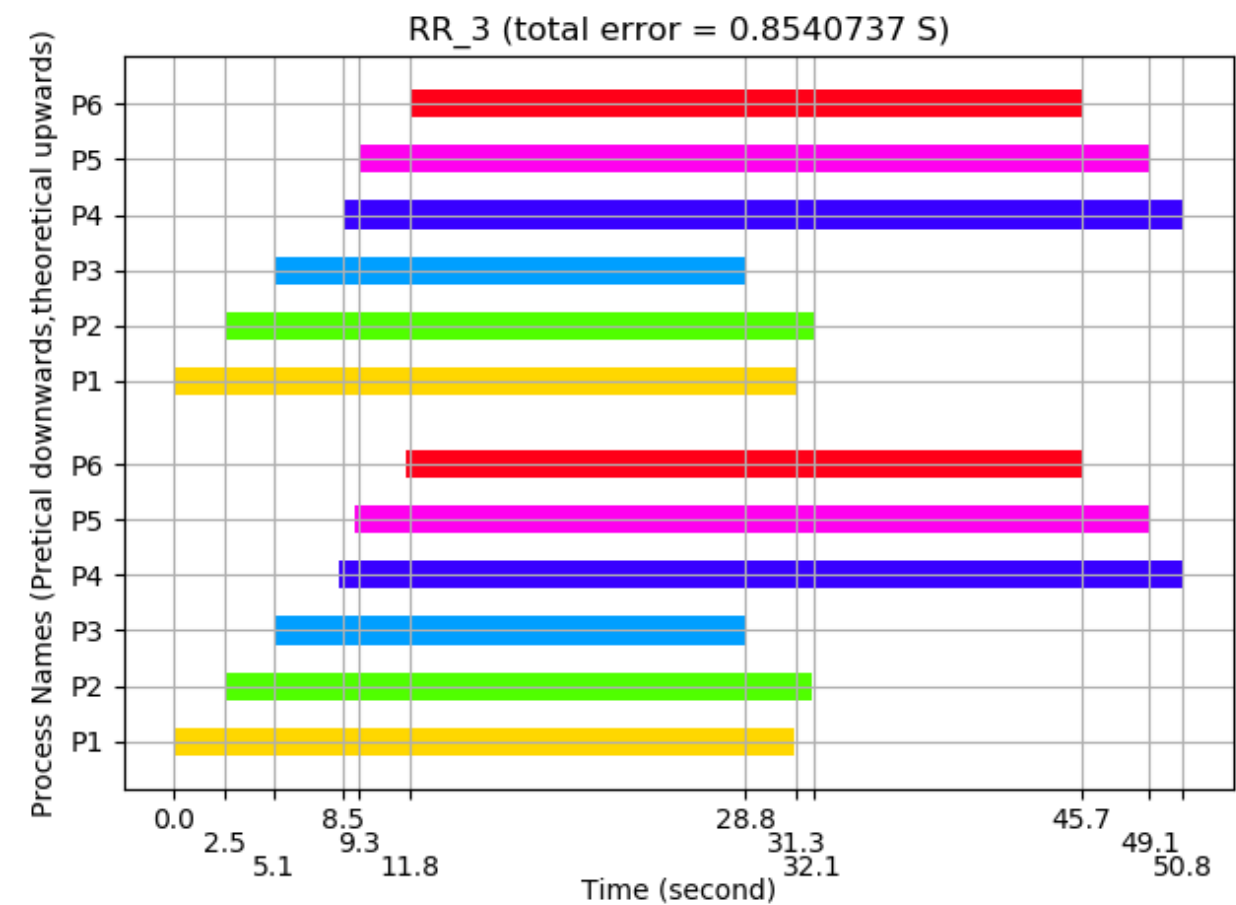
Test Data 3

RR 6 P1 1200 5000 P2 2400 4000 P3 3600 3000 P4 4800 7000 P5 5200 6000 P6 5800 5000

Exclusion Result

[3665.174424] [Project1] 2478 1588168106.260601535 1588168129.499940848
[3667.705237] [Project1] 2476 1588168101.571097830 1588168132.030650763
[3668.504261] [Project1] 2477 1588168104.009259327 1588168132.829643312
[3681.991124] [Project1] 2481 1588168112.868398091 1588168146.315958496
[3685.355679] [Project1] 2480 1588168110.379141254 1588168149.680375937
[3687.021035] [Project1] 2479 1588168109.550226157 1588168151.345665143

Comparison of Theoretical and Pretical Results



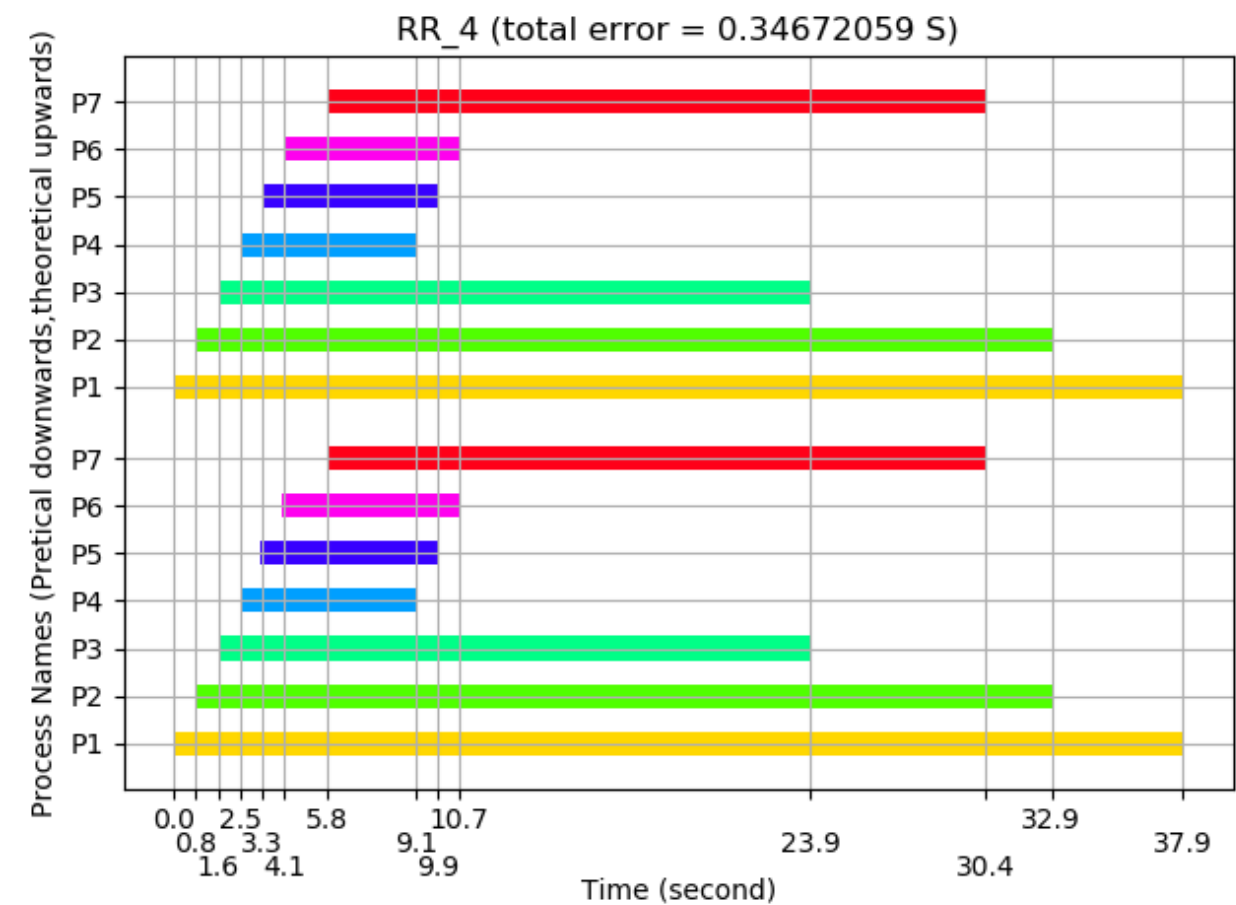
Test Data 4

RR 7 P1 0 8000 P2 200 5000 P3 300 3000 P4 400 1000 P5 500 1000 P6 500 1000 P7 600 4000

Exclusion Result

[3696.117990] [Project1] 2494 1588168153.843135725 1588168160.442250363
[3696.981309] [Project1] 2495 1588168154.644135471 1588168161.305535358
[3697.858220] [Project1] 2496 1588168155.474344742 1588168162.182409687
[3711.103706] [Project1] 2493 1588168152.990798487 1588168175.427358179
[3717.654807] [Project1] 2497 1588168157.153814645 1588168181.978193134
[3720.102524] [Project1] 2492 1588168152.180365431 1588168184.425810889
[3725.063099] [Project1] 2491 1588168151.356064047 1588168189.386184088

Comparison of Theoretical and Pretical Results



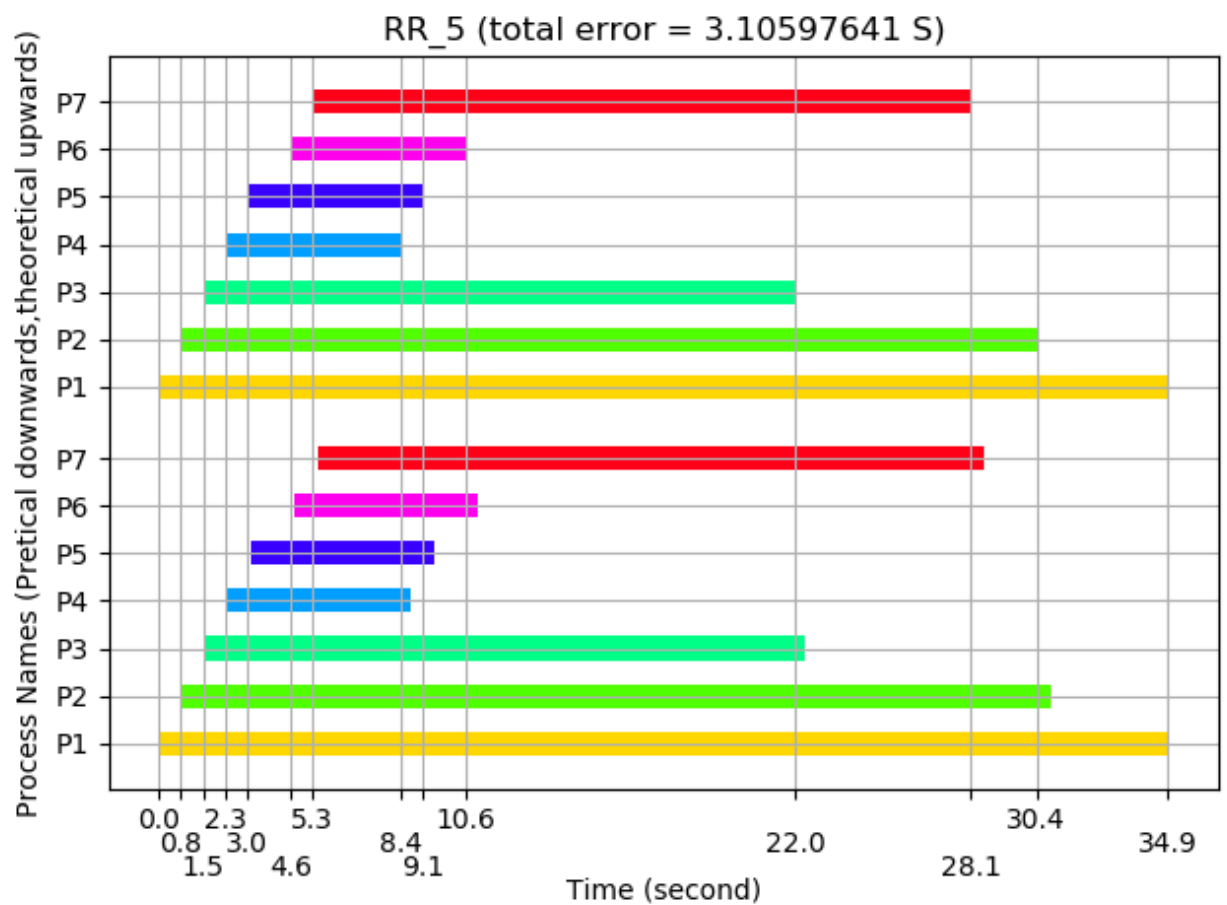
Test Data 5

RR 7 P1 0 8000 P2 200 5000 P3 200 3000 P4 400 1000 P5 400 1000 P6 600 1000 P7 600 4000

Exclusion Result

[3733.968150] [Project1] 2510 1588168191.856440697 1588168198.290873352
[3734.798315] [Project1] 2511 1588168192.671516466 1588168199.121004969
[3736.347727] [Project1] 2512 1588168194.287407109 1588168200.670353572
[3748.674285] [Project1] 2509 1588168191.068122937 1588168212.996410915
[3755.269664] [Project1] 2513 1588168195.031447116 1588168219.591518906
[3757.784351] [Project1] 2508 1588168190.219940103 1588168222.106106857
[3762.622342] [Project1] 2507 1588168189.397001909 1588168226.943902059

Comparison of Theoretical and Pretical Results



SJF (Shortest Job First)

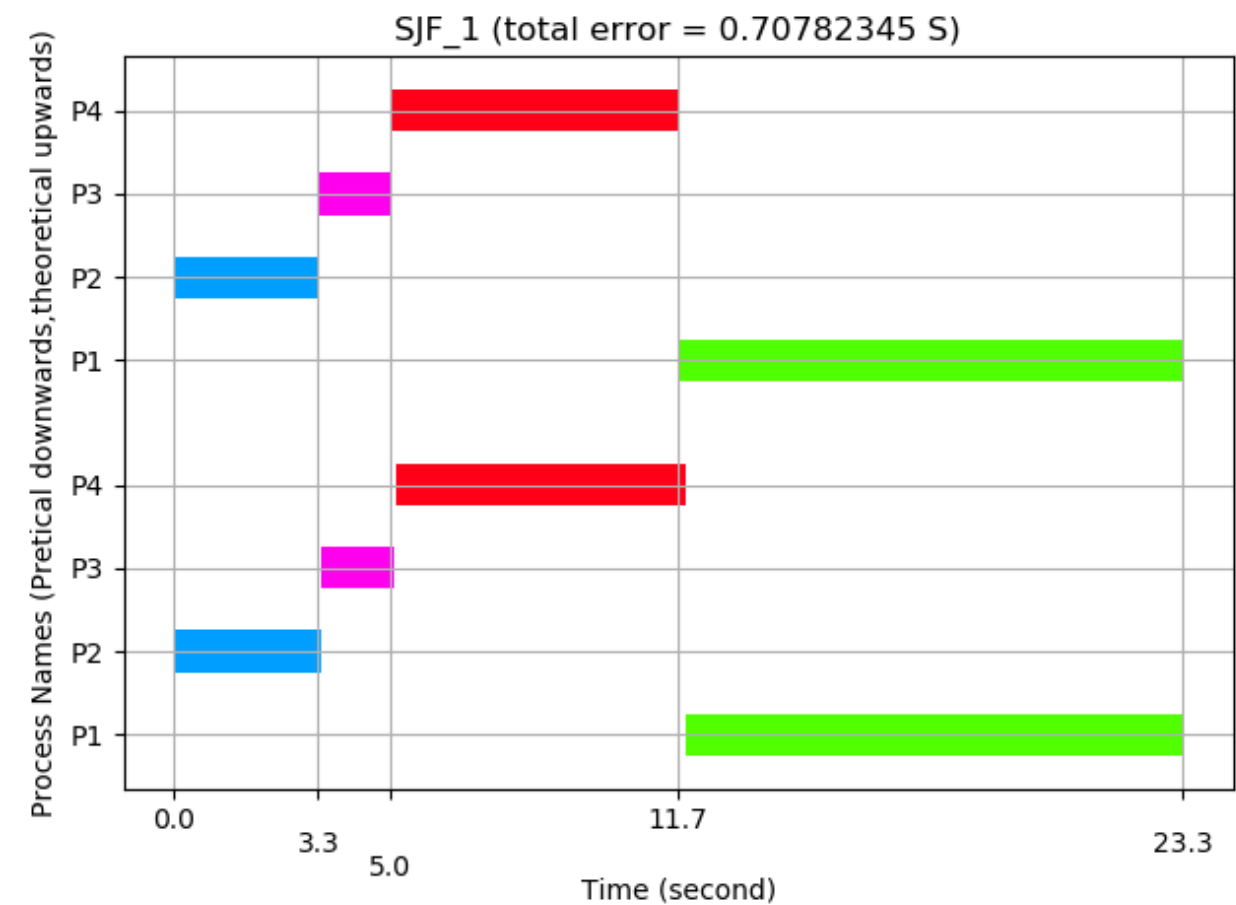
Test Data 1

SJF 4 P1 0 7000 P2 0 2000 P3 100 1000 P4 200 4000

Exclusion Result

[3765.983625] [Project1] 2522 1588168226.954558352 1588168230.305048211
 [3767.652945] [Project1] 2523 1588168230.305194086 1588168231.974300075
 [3774.514104] [Project1] 2524 1588168231.974428459 1588168238.835180271
 [3786.270682] [Project1] 2521 1588168238.835308592 1588168250.591280946

Comparison of Theoretical and Pretical Results



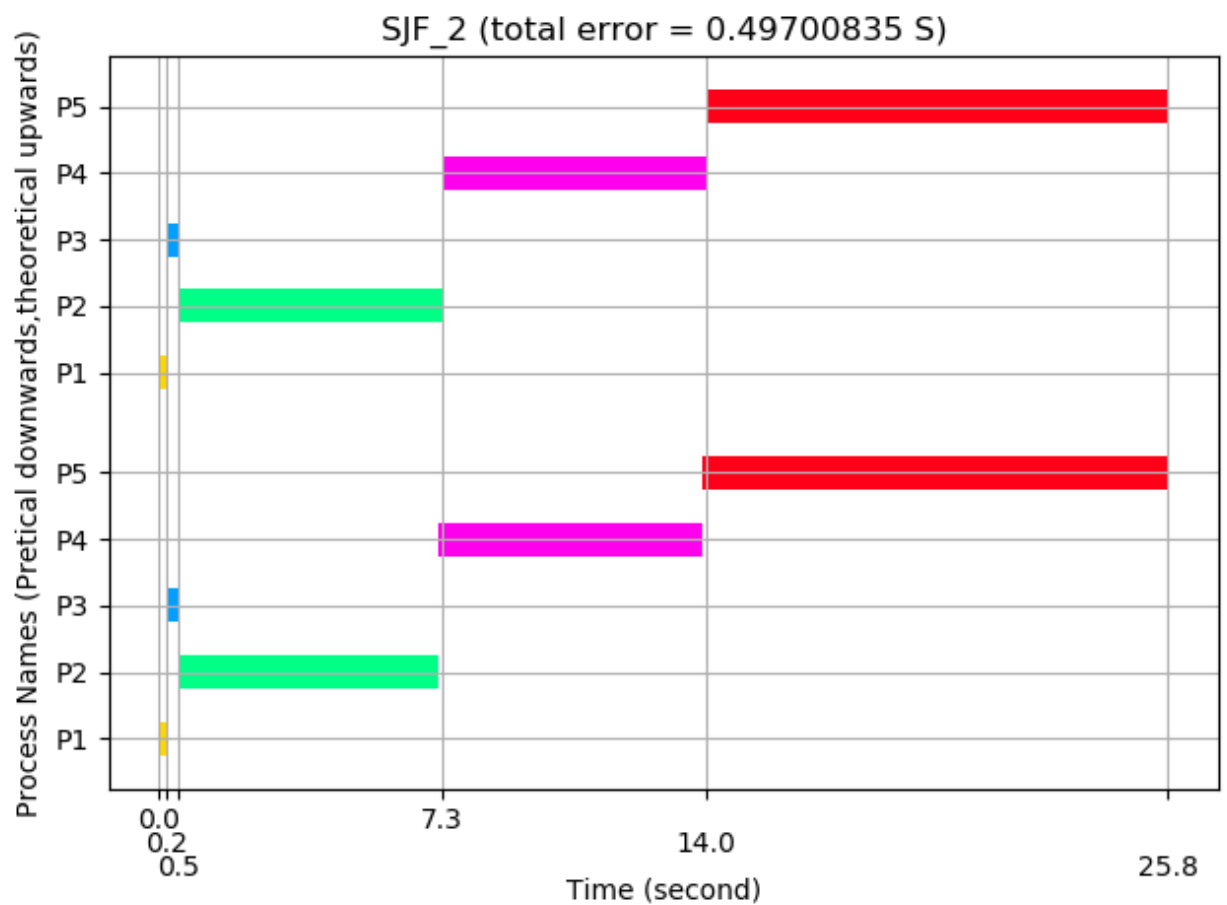
Test Data 2

SJF 5 P1 100 100 P2 100 4000 P3 200 200 P4 200 4000 P5 200 7000

Exclusion Result

[3786.561881] [Project1] 2534 1588168250.744927722 1588168250.882468040
[3786.845468] [Project1] 2536 1588168250.884163185 1588168251.166043474
[3792.494621] [Project1] 2535 1588168251.166166475 1588168256.814967352
[3798.297399] [Project1] 2537 1588168256.815149377 1588168262.617509196
[3809.386146] [Project1] 2538 1588168262.617713757 1588168273.705805114

Comparison of Theoretical and Pretical Results



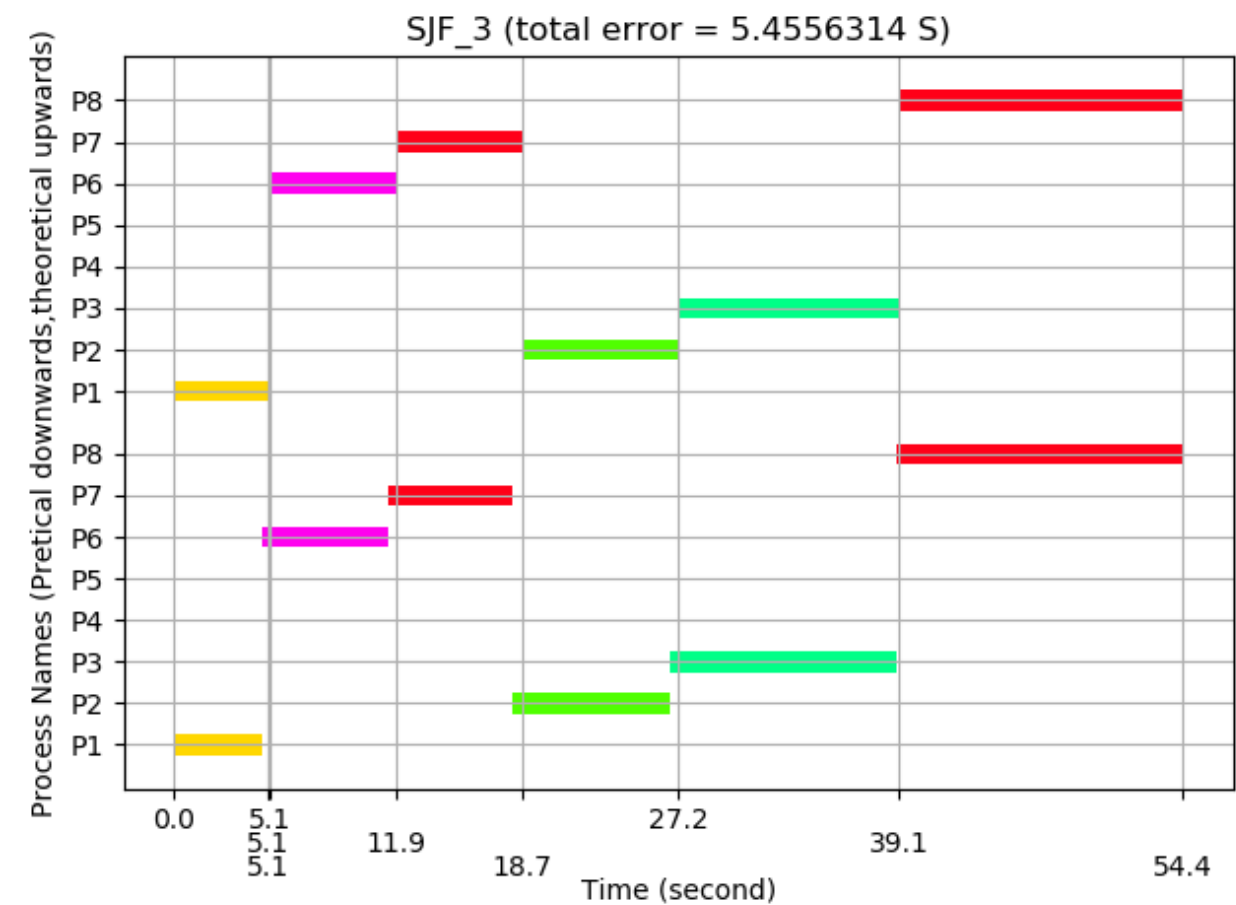
Test Data 3

SJF 8 P1 100 3000 P2 100 5000 P3 100 7000 P4 200 10 P5 200 10 P6 300 4000 P7 400 4000 P8 500 9000

Exclusion Result

[3814.503513] [Project1] 2546 1588168273.863863245 1588168278.822964384
[3814.520351] [Project1] 2549 1588168278.823126113 1588168278.839801302
[3814.535997] [Project1] 2550 1588168278.839986493 1588168278.855448418
[3821.031143] [Project1] 2551 1588168278.855562824 1588168285.350329673
[3827.749584] [Project1] 2552 1588168285.350816721 1588168292.068497834
[3835.887565] [Project1] 2547 1588168292.068630251 1588168300.206147578
[3847.365435] [Project1] 2548 1588168300.206290881 1588168311.683551576
[3862.149434] [Project1] 2553 1588168311.683682669 1588168326.466949704

Comparison of Theoretical and Pretical Results



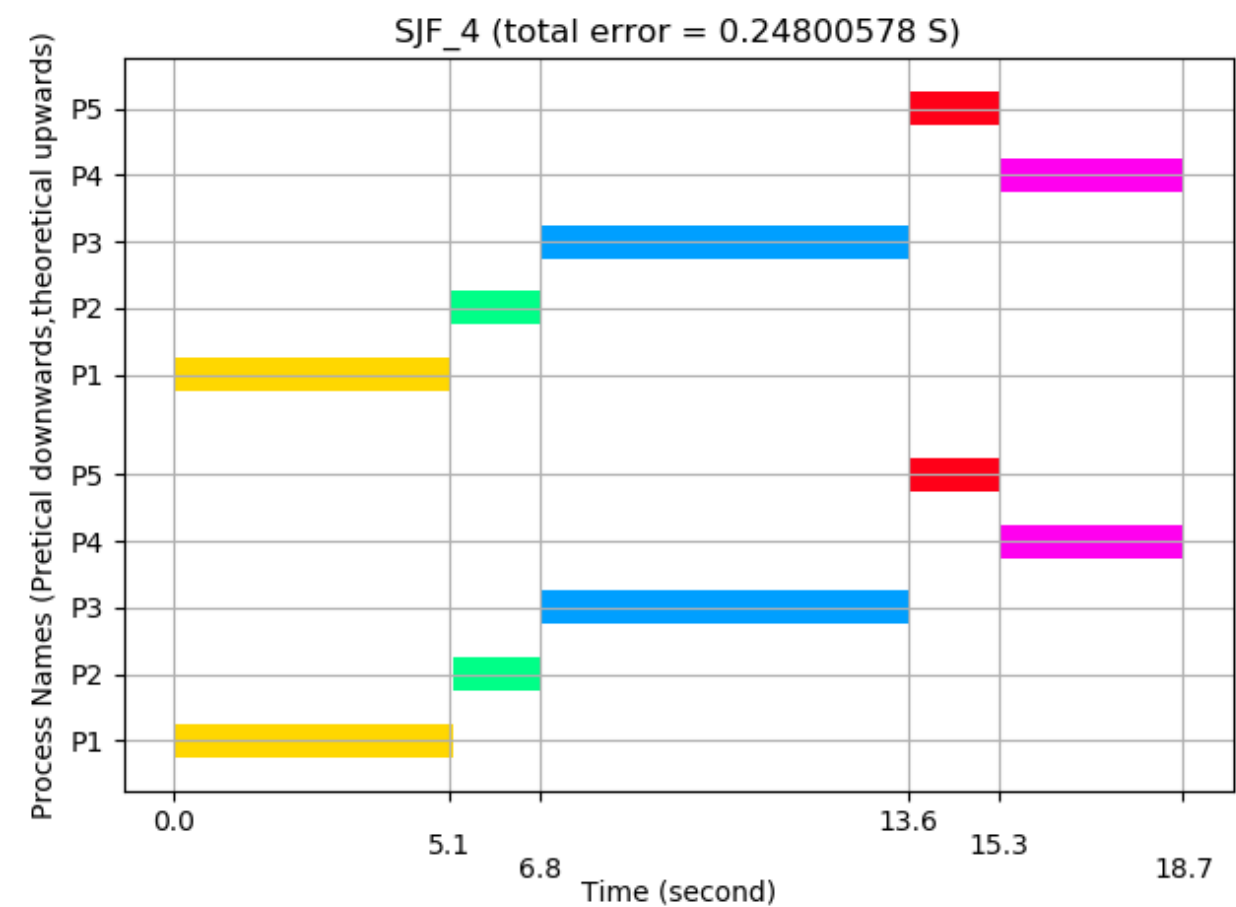
Test Data 4

SJF 5 P1 0 3000 P2 1000 1000 P3 2000 4000 P4 5000 2000 P5 7000 1000

Exclusion Result

[3867.236918] [Project1] 2563 1588168326.481374039 1588168331.554226504
[3868.807832] [Project1] 2564 1588168331.554368176 1588168333.125076231
[3874.867014] [Project1] 2565 1588168333.125712900 1588168339.184011566
[3876.406792] [Project1] 2567 1588168339.184429197 1588168340.723727941
[3879.764819] [Project1] 2566 1588168340.723941473 1588168344.081617662

Comparison of Theoretical and Pretical Results



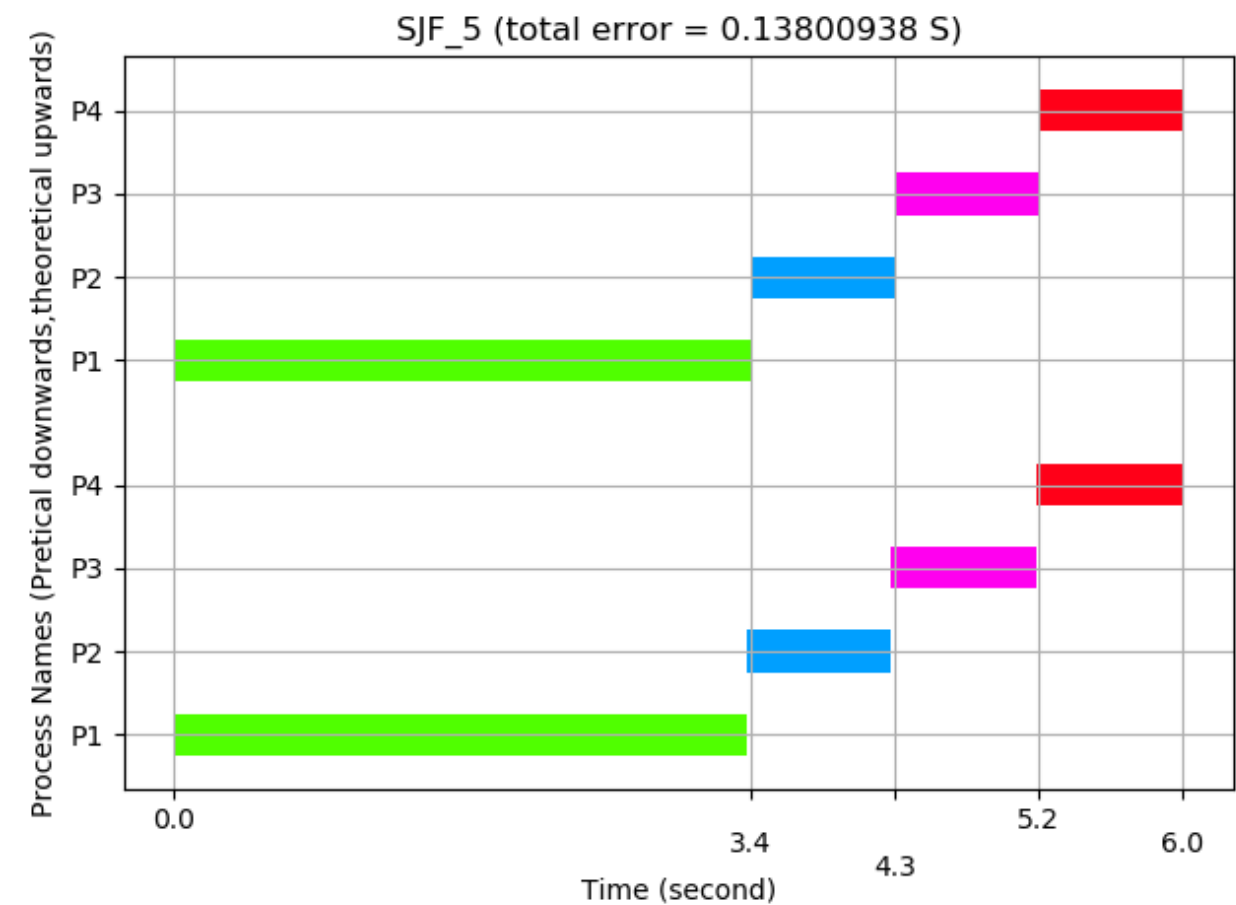
Test Data 5

SJF 4 P1 0 2000 P2 500 500 P3 1000 500 P4 1500 500

Exclusion Result

[3882.954626] [Project1] 2575 1588168344.092670505 1588168347.271295043
[3883.779132] [Project1] 2576 1588168347.271431682 1588168348.095767885
[3884.601899] [Project1] 2577 1588168348.095896804 1588168348.918501367
[3885.462978] [Project1] 2578 1588168348.918630081 1588168349.779545250

Comparison of Theoretical and Pretical Results



Preemptive Shortest Job First (PSJF)

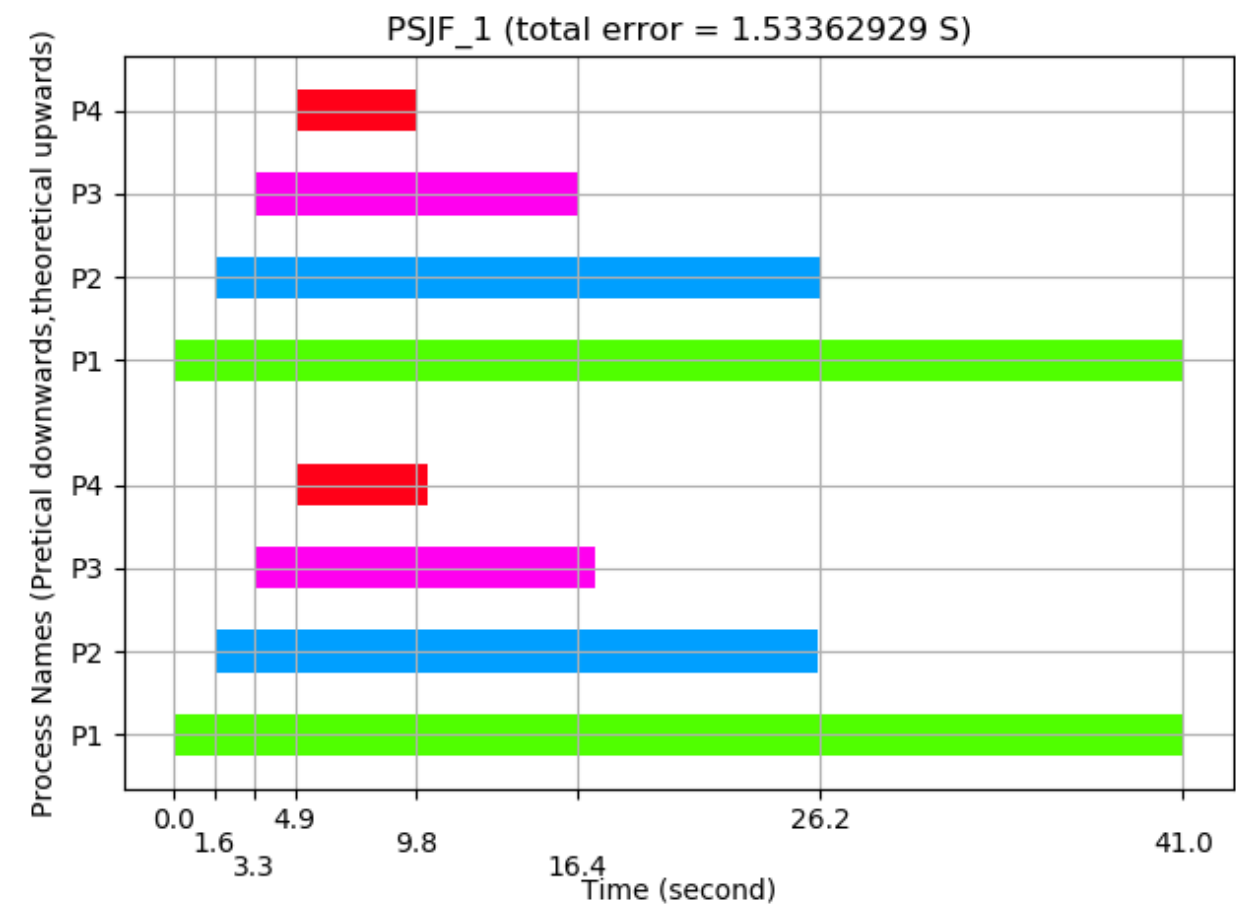
Test Data 1

PSJF 4 P1 0 10000 P2 1000 7000 P3 2000 5000 P4 3000 3000

Exclusion Result

[3895.237368] [Project1] 2589 1588168354.697952554 1588168359.553537881
[3901.503196] [Project1] 2588 1588168353.044345510 1588168365.819111444
[3911.030351] [Project1] 2587 1588168351.439450196 1588168375.345879032
[3925.549427] [Project1] 2586 1588168349.791059823 1588168389.864364762

Comparison of Theoretical and Pretical Results



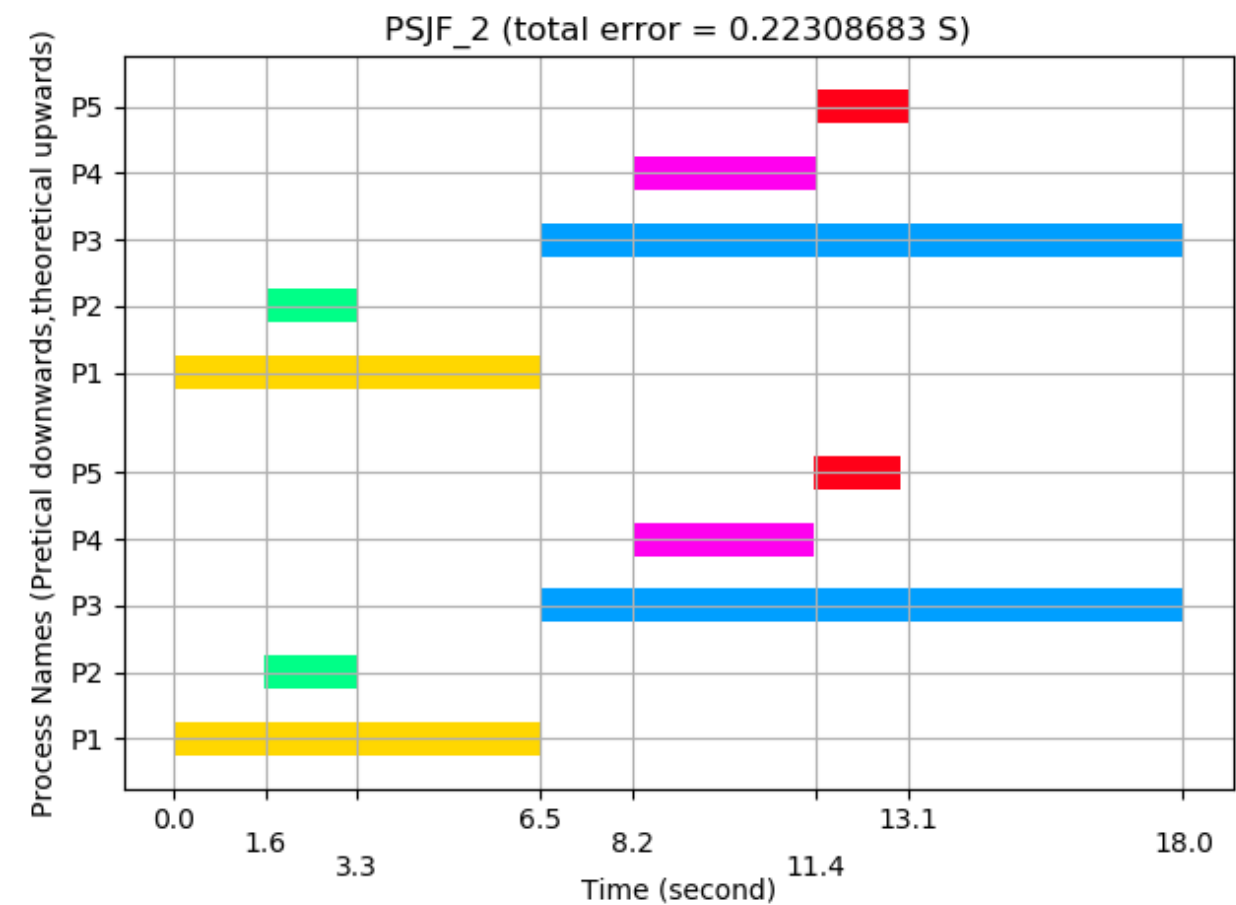
Test Data 2

PSJF 5 P1 0 3000 P2 1000 1000 P3 2000 4000 P4 5000 2000 P5 7000 1000

Exclusion Result

[3928.807554] [Project1] 2600 1588168391.507021802 1588168393.122358970
[3932.197447] [Project1] 2599 1588168389.874888729 1588168396.512113653
[3937.211036] [Project1] 2602 1588168398.196267289 1588168401.525498956
[3938.857241] [Project1] 2603 1588168401.526978091 1588168403.171636682
[3943.701520] [Project1] 2601 1588168396.512244847 1588168408.015718653

Comparison of Theoretical and Pretical Results



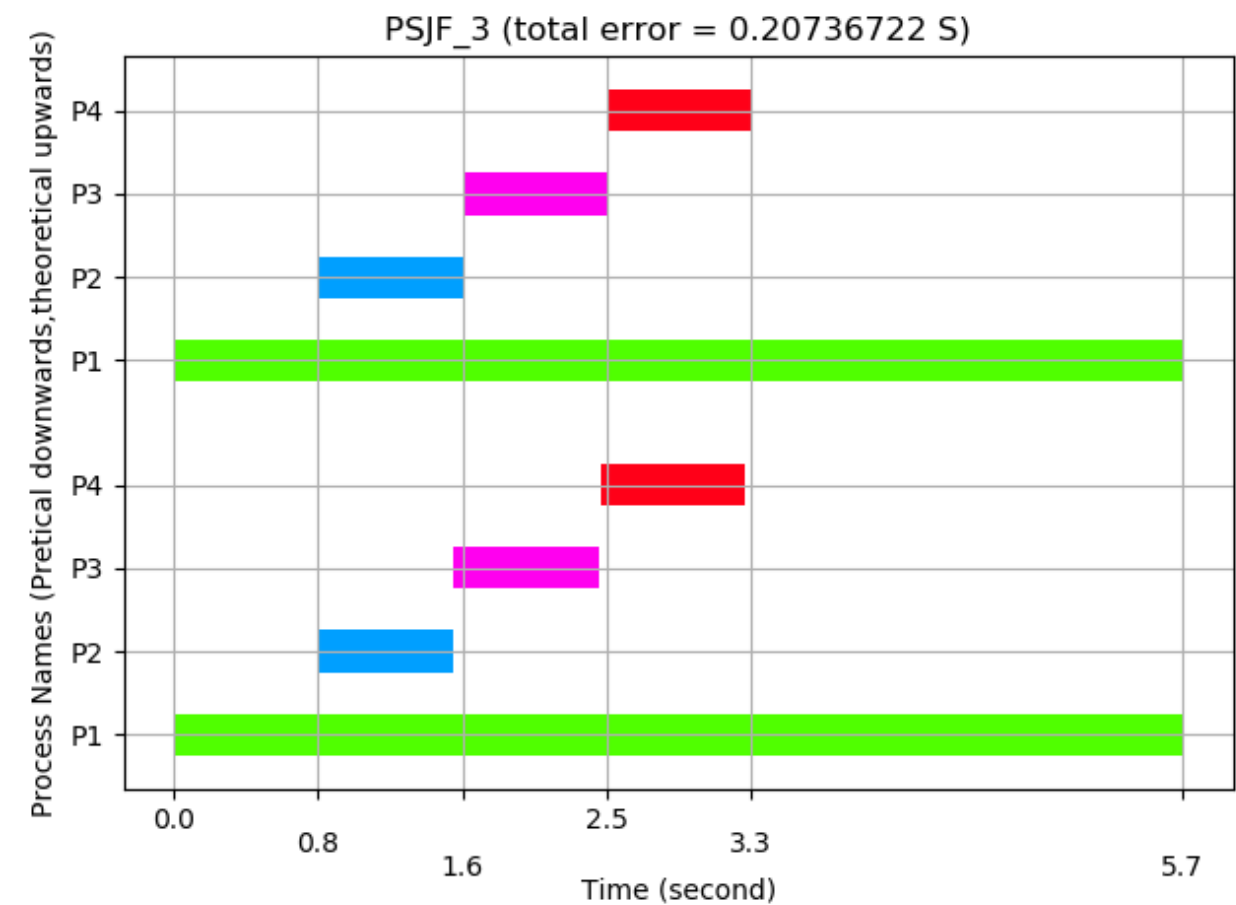
Test Data 3

PSJF 4 P1 0 2000 P2 500 500 P3 1000 500 P4 1500 500

Exclusion Result

[3945.321407] [Project1] 2612 1588168408.855796587 1588168409.635540611
[3946.072594] [Project1] 2613 1588168409.637421286 1588168410.386696315
[3946.893506] [Project1] 2614 1588168410.388225868 1588168411.207575397
[3949.356089] [Project1] 2611 1588168408.026120606 1588168413.670057886

Comparison of Theoretical and Pretical Results



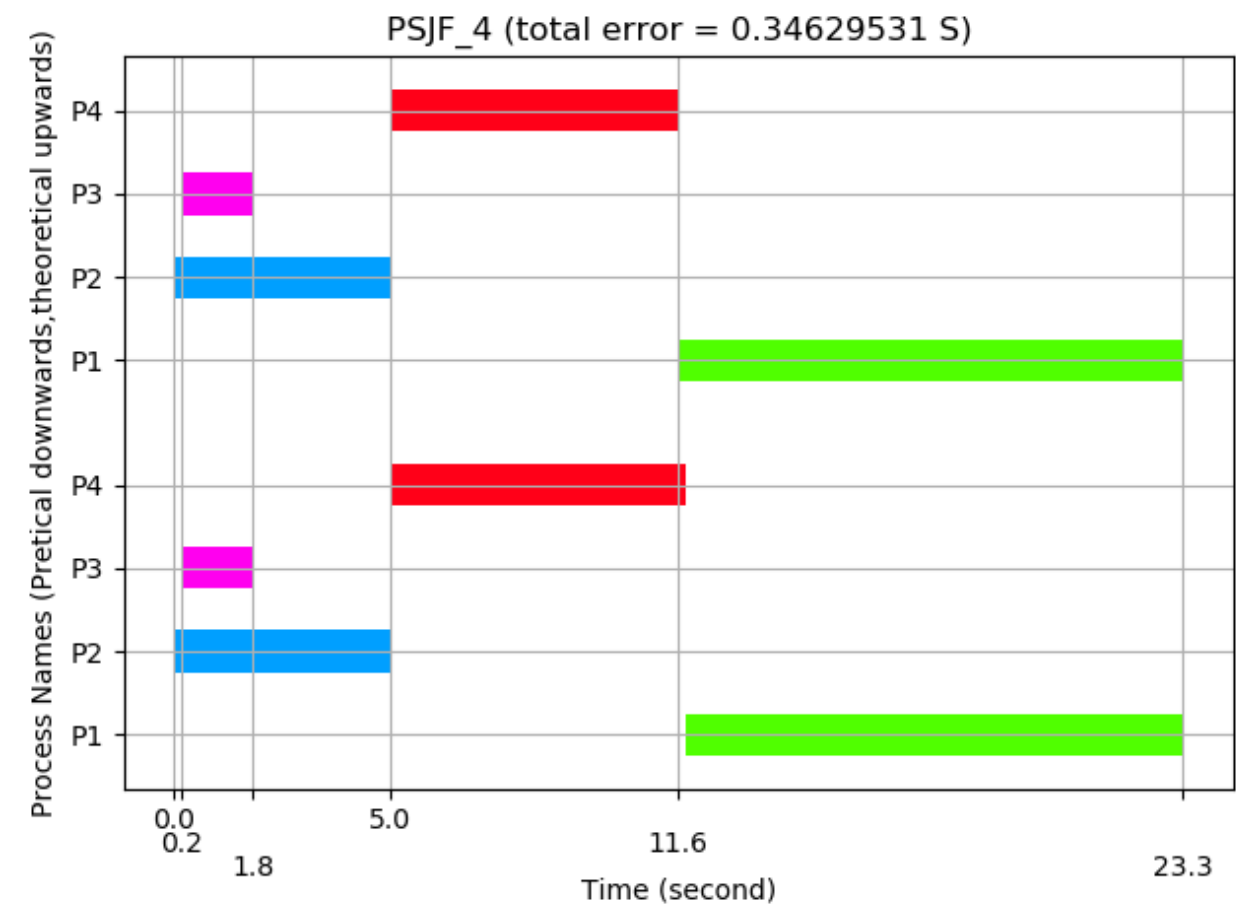
Test Data 4

PSJF 4 P1 0 7000 P2 0 2000 P3 100 1000 P4 200 4000

Exclusion Result

[3951.191765] [Project1] 2624 1588168413.839951428 1588168415.505659485
[3954.414490] [Project1] 2623 1588168413.680532190 1588168418.728253268
[3961.024168] [Project1] 2625 1588168418.728384321 1588168425.337662486
[3972.555337] [Project1] 2622 1588168425.337793976 1588168436.868362374

Comparison of Theoretical and Pretical Results



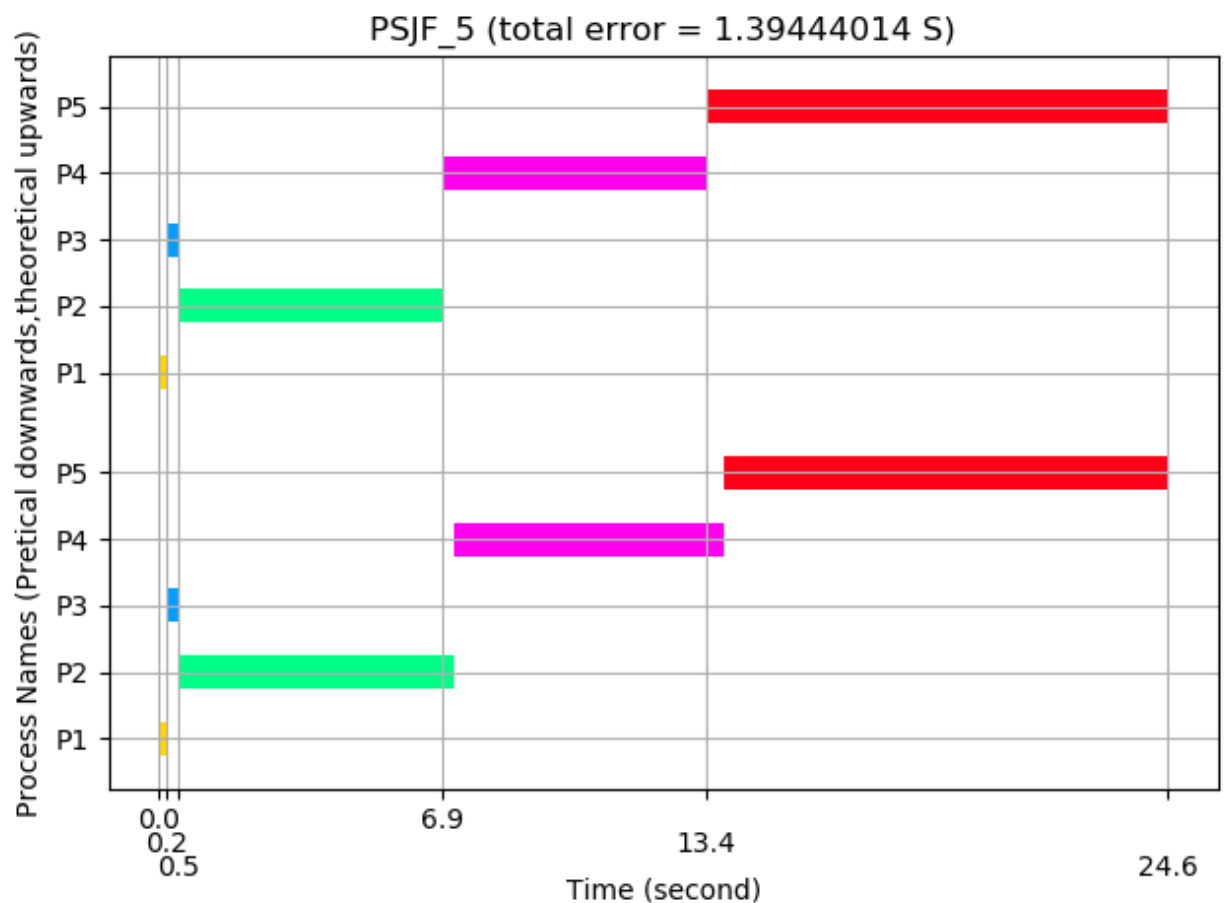
Test Data 5

PSJF 5 P1 100 100 P2 100 4000 P3 200 200 P4 200 4000 P5 200 7000

Exclusion Result

[3972.870730] [Project1] 2635 1588168437.014183349 1588168437.183743326
[3973.161501] [Project1] 2637 1588168437.185412355 1588168437.474502248
[3979.621266] [Project1] 2636 1588168437.474628456 1588168443.934003933
[3986.377239] [Project1] 2638 1588168443.934134972 1588168450.689702203
[3998.173417] [Project1] 2639 1588168450.689835431 1588168462.485400909

Comparison of Theoretical and Pretical Results



Conclusion

發現每一個 process 在 run 的時間都比理論值要長，即使是 FIFO 中第一個

process 的 running time，也會比其需要的時間長。推測可能的原因是：

Project1的運行環境於 virtual box 中，virtual box 為原生系統的一個應用程式，

其無法完全占用物理 CPU

與前一個原因相似，

在兩次用 timer 計時之間的 code,除了 run process 還會做一些別的事情. void

```
execute_unit_time(){ volatile unsigned long i; for(i=0;i<UNIT_TIME;i++); }
```

在理論值中沒有考慮 turn around 的時間,而實際中在 processes 切換的時候需要

短暫占用 CPU.

Running Process 的時候,只是一個 for loop,不知道 CPU 和 compiler 會不會偷偷做一些黑魔法.

Reference:

測量 Linux 內核中函數的執行時間: <https://www.codenong.com/4655711/>

繪圖程式參考自:

<https://github.com/wangyenjen/OS-Project-1/blob/master/report.md>