

PhyTune 使用说明

V1.9 (2025-03-29)

版权所有 © 飞腾信息技术有限公司 2022。保留所有权利。

未经本公司同意，任何单位、公司或个人不得擅自复制、翻译、摘抄本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

商标声明

Phytium和其他飞腾商标均为飞腾信息技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

本文档的内容视为飞腾的保密信息，您应当严格遵守保密义务；未经飞腾事先书面同意，您不得向任何第三方披露本文档内容或提供给任何第三方使用。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，飞腾在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但飞腾在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

本文档中所有内容，包括但不限于图片、架构设计、页面布局、文字描述，均由飞腾和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权等。非经飞腾和/或其关联公司书面同意，任何人不得擅自使用、修改、复制上述内容。
如有技术问题，可联系 support@phytium.com.cn 获取支持。

飞腾信息技术有限公司

地址：天津市滨海新区海缘路1号信安创业广场5号楼

网址：www.phytium.com.cn

版本历史

文档版本	发布日期	更新说明
V1.9	2025-03-29	根据软件版本 v1.9 更新。
V1.8	2024-12-31	根据软件版本 v1.8 更新。
V1.7	2024-09-30	根据软件版本 v1.7 更新。
V1.6	2024-06-14	根据软件版本 v1.6 更新。
V1.5	2023-09-04	根据软件版本 v1.5 更新。
V1.3	2023-06-04	根据软件版本 v1.3 更新。
V1.2	2023-04-28	根据软件版本 v1.2 更新。
V1.1	2023-04-08	根据软件版本 v1.1 更新。
V1.0	2023-03-20	初始发布。

目 录

1 概述	1
1.1 简介	1
1.2 功能特性	1
2 安装部署	2
2.1 环境要求	2
2.2 安装	3
2.3 运行	3
2.4 卸载	4
2.5 技术约束	4
3 功能说明	5
3.1 目标端配置	5
3.1.1 添加目标端	5
3.1.2 删除目标端	6
3.2 环境分析	7
3.3 处理器分析	11
3.3.1 计算处理分析	12
3.3.2 内存访问分析	17
3.3.3 设备访问分析	19
3.3.4 TopDown 分析	21
3.4 访存分析	27
3.4.1 系统级分析	27
3.4.2 应用级分析	35
3.5 NUMA 分析	38
3.6 I/O 分析	42
3.6.1 系统级分析	43

3.6.2 应用级分析	49
3.7 远程采样.....	52
4 常见问题	55
4.1 参数错误	55
4.1.1 问题现象	55
4.1.2 处理方法.....	55
4.2 无权访问内核关键数据.....	56
4.2.1 问题现象.....	56
4.2.2 处理方法	56

图 目 录

图 2-1 命令行安装	3
图 2-2 提示服务启动完成.....	3
图 2-3 桌面快捷方式.....	3
图 2-4 启动 PhyTune	4
图 2-5 卸载 PhyTune	4
图 3-1 添加目标端	5
图 3-2 填写目标端配置.....	6
图 3-3 目标端添加成功.....	6
图 3-4 删除目标端机器.....	7
图 3-5 目标端被删除.....	7
图 3-6 环境分析 - 配置	8
图 3-7 环境分析 - 系统设置信息.....	9
图 3-8 环境分析 - CPU 使用统计.....	10
图 3-9 环境分析 - 内存使用统计.....	10
图 3-10 环境分析 - 存储 I/O	11
图 3-11 环境分析 - 网络 I/O	11
图 3-12 选择目标机器	12
图 3-13 输入目标程序	12
图 3-14 使用测试程序	13
图 3-15 选择目标部件	13
图 3-16 手动设置分析选项.....	14
图 3-17 分析运行状态显示.....	14
图 3-18 停止分析任务	14
图 3-19 查看结果.....	15
图 3-20 计算处理性能分析结果表.....	16

图 3-21 计算处理性能分析结果图（条状图）	16
图 3-22 计算处理性能分析结果图（饼状图）	17
图 3-23 内存访问分析配置	18
图 3-24 内存访问分析结果表	19
图 3-25 设备访问分析配置	20
图 3-26 设备访问分析结果表	21
图 3-27 Topdown 分析配置界面	22
图 3-28 Topdown stage2 分析配置界面	23
图 3-29 结果表展示界面	24
图 3-30 stage1 指标分布饼状图	26
图 3-31 指标组各指标柱状图	27
图 3-32 各时间间隔下各指标的分簇柱状图	27
图 3-33 系统级访存分析 - Hit 分析	28
图 3-34 系统级访存分析 - Miss 分析	28
图 3-35 系统级访存分析 - Hit 分析记录模式状态显示	29
图 3-36 系统级访存分析 - Hit 分析实时模式状态显示	29
图 3-37 系统级访存分析 - Miss 分析记录模式状态显示	30
图 3-38 Hit 分析记录模式结果表	30
图 3-39 Hit 分析记录模式结果图	31
图 3-40 Miss 分析记录模式结果表	32
图 3-41 Miss 分析记录模式结果图	33
图 3-42 Hit 分析实时模式结果图	34
图 3-43 Miss 分析实时模式结果图	34
图 3-44 选择目标程序	35
图 3-45 选择目标进程	36
图 3-46 应用级访存分析 - Hit 分析记录模式状态显示	37
图 3-47 应用级访存分析 - Hit 分析记录模式状态显示	37
图 3-48 应用级访存分析 - Hit 分析记录模式状态显示	38
图 3-49 NUMA 分析 - 启动进程	39
图 3-50 NUMA 分析 - 绑核配置	39
图 3-51 NUMA 分析 - 手动绑核	40
图 3-52 NUMA 分析 - 分析进程	40
图 3-53 NUMA 分析 - 系统设置结果	41
图 3-54 NUMA 分析 - CPU/内存使用统计	42

图 3-55 NUMA 分析 - 线程使用统计	42
图 3-56 系统级 I/O 分析 - 记录模式	43
图 3-57 系统级 I/O 分析 - 实时模式	43
图 3-58 系统级 I/O 分析 - 记录模式状态显示	44
图 3-59 系统级 I/O 分析 - 记录模式状态显示	44
图 3-60 系统级 I/O 分析 - 各阶段时延信息	45
图 3-61 系统级 I/O 分析 - 读写请求信息	45
图 3-62 系统级 I/O 分析 - 设备寻道信息	45
图 3-63 系统级 I/O 分析 - I/O 流量信息	45
图 3-64 系统级 I/O 分析 - 请求合并信息	46
图 3-65 系统级 I/O 分析 - 活跃请求信息	46
图 3-66 系统级 I/O 分析 - I/O 请求流程总览	46
图 3-67 系统级 I/O 分析 - I/O 请求流程总览	47
图 3-68 系统级 I/O 分析 - 寻道距离变化	47
图 3-69 系统级 I/O 分析 - 队列深度变化	48
图 3-70 系统级 I/O 分析 - 流量信息	48
图 3-71 系统级 I/O 分析 - 操作及片段记录	48
图 3-72 系统级 I/O 分析 - 实时模式	49
图 3-73 应用级 I/O 分析 - 选择待测程序	50
图 3-74 应用级 I/O 分析 - 调用栈	51
图 3-75 应用级 I/O 分析 - 函数占比饼图	51
图 3-76 应用级 I/O 分析 - 函数调用时间轴	52
图 3-77 远程采样步骤 I	53
图 3-78 远程采样步骤 II	53
图 3-79 远程采样结果展示	54
图 4-1 参数错误	55
图 4-2 无权访问内核数据	56
图 4-3 执行提权	56

表 目 录

表 2-1	操作端环境要求.....	2
表 2-2	目标端环境要求.....	2
表 3-1	TopDown 指标	24

1 概述

1.1 简介

PhyTune 是用于飞腾处理器的性能分析工具，通过监视硬件事件并与软件运行情况相结合，可以评估负载是否高效运行，帮助识别性能瓶颈并发现优化机会。

1.2 功能特性

- 支持运行环境状态分析，包括 CPU、内存、硬盘、网络及系统配置等，帮助用户了解系统状态、识别潜在瓶颈。
- 支持计算处理性能分析，包括流水线性能、分支预测性能、TLB 性能、cache 性能、指令类型分析等，并支持 TopDown 分析。
- 支持内存访问性能分析，包括 DDR 访问带宽、高速缓存 Hit 及 Miss 情况等。
- 支持进程/线程的 NUMA 资源使用状态分析，帮助识别分布问题并改善 NUMA 分配策略。
- 支持设备访问性能分析，包括 PCIe 访问带宽和延迟等，并支持存储块设备访问性能分析。
- 支持 I/O 分析，支持跟踪指定进程记录关键数据，并周期性的实时统计设备请求的大小和时延的分布信息。
- 支持远程和本地两种部署方式、系统级和应用级两类分析对象、实时和记录两种分析模式。
- 支持图形化交互界面和可视化分析结果呈现。

2 安装部署

2.1 环境要求

表 2-1 操作端环境要求

处理器	飞腾腾锐 D2000、FT-2000/4
磁盘	> 300MB
网络	SSH 连接
操作系统	麒麟 v10 桌面系统
外部依赖	perf, glibc-2.28+

表 2-2 目标端环境要求

处理器	飞腾腾云 S5000C/S2500、飞腾腾锐 D3000/D2000
网络	SSH 服务
操作系统	麒麟 v10/openEuler2203/openCloud 8.6 服务器系统
外部依赖	perf, python3.7+, Phytium PMU drivers

说明:操作端指 PhyTune 软件安装及运行操作所使用的带有图形桌面环境的计算机,目标端指 PhyTune 软件分析的目标软件运行所在计算机。

2.2 安装

获取安装包后，命令行窗口进入到安装包所在目录，执行如下命令进行安装（自动安装依赖项）：

```
sudo apt install -f ./phytune_1.9.0.deb
```

```
nt@nt:~$ sudo apt install -f ./phytune_1.9.0.deb
正在读取软件包列表... 完成
正在分析软件包的依赖关系树
正在读取状态信息... 完成
注意，选中 'phytune' 而非 './phytune_1.9.0.deb'
下列软件包是自动安装的并且现在不需要了：
  libfltk-images1.3 libfltk1.3 libfreerdp-client2-2 libfreerdp2-2 libmediainfo0v5 libvncclient1 libwinpr2-2 libzen0v5
  localechooser-data nvme-cli python3-reportlab python3-reportlab-accel user-setup
使用 'sudo apt autoremove' 来卸载它(它们)。
下列【新】软件包将被安装：
  phytune
升级了 0 个软件包，新安装了 1 个软件包，要卸载 0 个软件包，有 329 个软件包未被升级。
需要下载 0 B/58.9 MB 的归档。
解压后会消耗 0 B 的额外空间。
获取:1 /home/nt/phytune_1.9.0.deb phytune arm64 1.9.0 [58.9 MB]
正在选中未选择的软件包 phytune。
(正在读取数据库 ... 系统当前共安装有 215349 个文件和目录。)
准备解压 /home/nt/phytune_1.9.0.deb ...
正在解压 phytune (1.9.0) ...
正在设置 phytune (1.9.0) ...
```

图 2-1 命令行安装

成功安装后，还将提示服务进程启动完成。

```
请确保密码输入正确否则会导致phytune组件运行错误！
请输入 "nt" 用户密码：
请再次输入 "nt" 用户的密码：
两次密码输入一致。
Server 设置完毕。
请双击 phytune 桌面图标进行性能检测。
```

图 2-2 提示服务启动完成

PhyTune 软件部署在/opt/phytune 目录，并在桌面生成快捷方式。

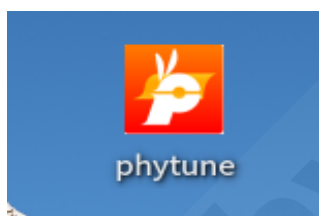


图 2-3 桌面快捷方式

2.3 运行

双击桌面图标即可启动 PhyTune。



图 2-4 启动 PhyTune

2.4 卸载

执行下列命令可以卸载 PhyTune 软件。

```
sudo dpkg -r phytune
```

```
nt@nt:~$ sudo dpkg -r phytune
输入密码
(正在读取数据库 ... 系统当前共安装有 215663 个文件和目录。)
正在卸载 phytune (1.9.0) ...
```

图 2-5 卸载 PhyTune

2.5 技术约束

- PhyTune 软件可以部署在一台操作端上，对多台目标端服务器执行性能分析任务；但不支持多台操作端对同一台目标端执行性能分析任务。
- 目标端网络配置发生变更后需要重新进行目标端配置操作。

3 功能说明

3.1 目标端配置

如前所述，目标端指 PhyTune 软件用于分析的软件运行所在的计算机。PhyTune 软件可以部署在一台操作端上，对多台目标端服务器执行性能分析任务。目标端目前支持飞腾腾云 S2500 服务器/S5000C 服务器；此外，也支持对 PhyTune 软件所在的操作端（飞腾 D3000/D2000、FT-2000/4 等桌面处理器）执行性能分析任务。

目标端配置的主要功能是为 PhyTune 软件配置执行分析任务的目标端计算机的 IP 地址、访问账号等内容。

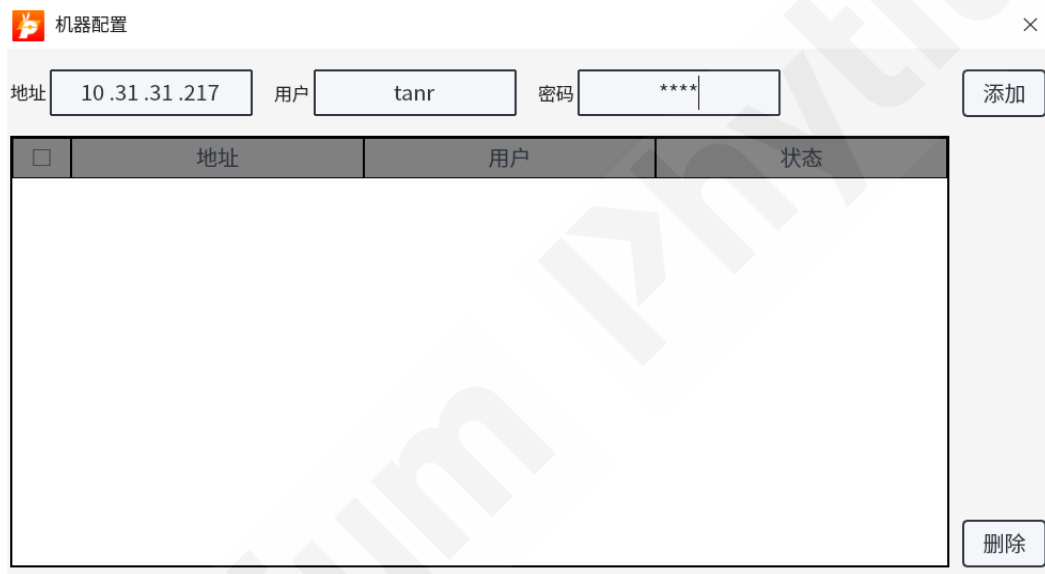
3.1.1 添加目标端

在任一子模块中都可以添加目标机器，下图以"处理器分析"为例，点击"目标配置"按钮。



图 3-1 添加目标端

填写目标机器配置信息包括地址、用户名、密码：



机器配置

地址 10.31.31.217 用户 tanr 密码 **** 添加

	地址	用户	状态
--	----	----	----

删除

图 3-2 填写目标端配置

点击"添加"按钮进行添加，成功后显示如下：



机器配置

地址 10.31.31.217 用户 tanr 密码 **** 添加

	地址	用户	状态
☐	10.31.31.217	tanr	available

图 3-3 目标端添加成功

3.1.2 删除目标端

选中需要删除的目标端点击"删除"按钮进行删除：

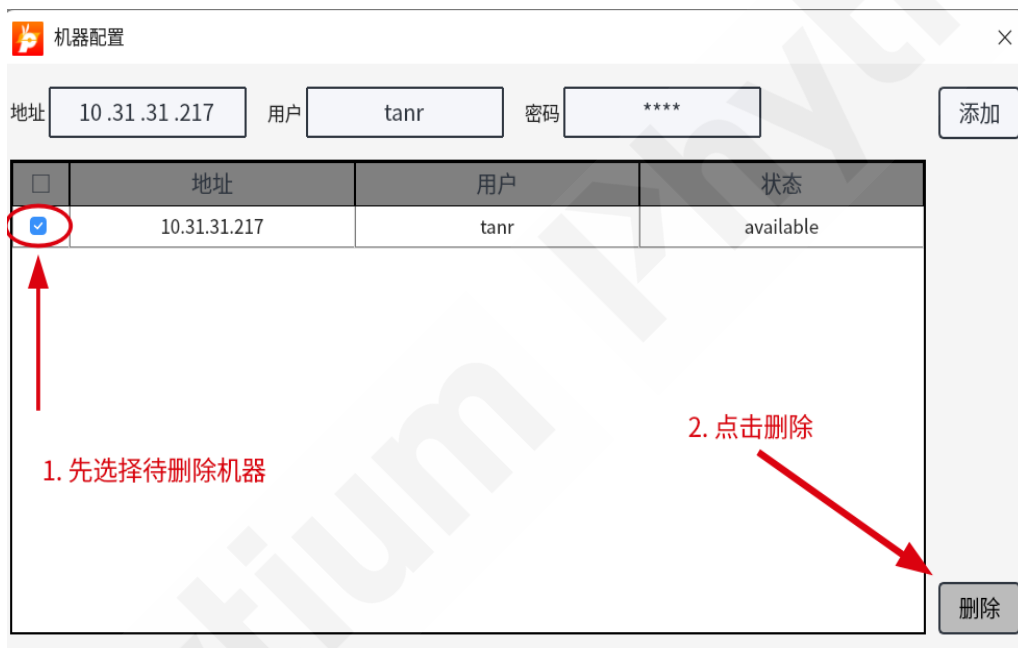


图 3-4 删除目标端机器

成功删除后，列表中不再显示该目标端机器信息：

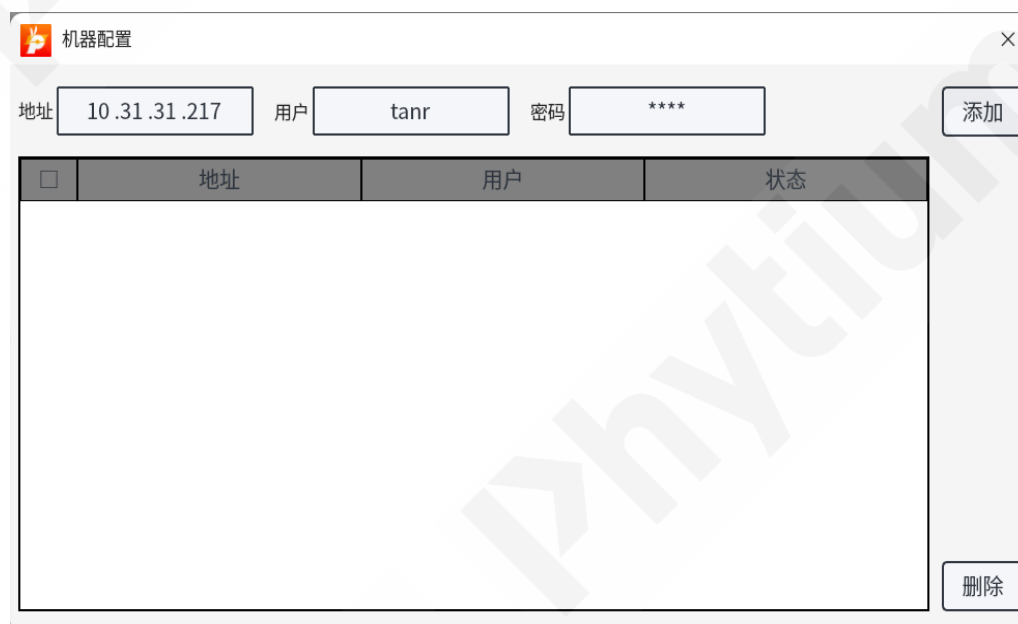


图 3-5 目标端被删除

3.2 环境分析

环境分析功能提供全面的系统资源监控和性能评估。用户通过记录和实时

两种操作模式，收集进程运行环境的 CPU、内存、硬盘、网络及系统配置等各方面的信息，帮助用户了解系统状态、识别潜在瓶颈、优化资源配置。

3.2.1.1 分析配置

环境分析配置页面如下图，其中：

系统：对系统环境的 CPU、内存、硬盘、网络及系统配置等信息分析。

进程：环境分析起止的同时关联应用程序的启停。适用于对程序运行时的系统环境的 CPU、内存、硬盘、网络及系统配置等信息分析。

实时模式：根据采样间隔(s)持续采样，实时更新分析结果，直到手动停止，适用于长期监控与性能跟踪的场景。

记录模式：根据采样间隔(s)和采样次数采样结束后停止，提供一次采样时间段内的分析结果，适用于短期性能分析和瞬时状态诊断的场景

目标程序：由文件浏览窗口获取待测程序绝对路径。

应用参数：待测应用程序参数，可选填。

采样间隔：以秒为采样间隔单位。

采样次数：满足采样次数后结束分析。

注意：系统/进程必选其一，实时模式/记录模式必选其一，目标程序必填，应用参数选填，采样间隔/采样次数必填。



图 3-6 环境分析 - 配置

3.2.1.2 分析执行

点击分析配置页面“开始”按钮，等待连接目标机器、启动环境分析工具（如勾选进程选项，则同步启动目标程序）。

3.2.1.3 结果查看

环境分析结果包括五个部分，分别为系统设置、CPU 使用统计、内存使用统计、存储 I/O 和网络 I/O。

系统设置具体功能及指标如下图，其中：

CPU 信息：socket 序号、型号、架构、核数、最大频率、当前频率。

缓存：L1d、L1i、L2、L3 大小，括号内表示缓存实体个数。

NUMA 相关信息：各 NUMA 节点的 CPU 核分布、内存容量、拓扑距离，numa-balancing 开关状态。

内存条信息：插槽位置、容量、最大速率、配置速率、类型。

硬盘设备信息：设备名称、型号、容量、类型。

文件系统信息：设备名称、类型、容量、使用率、挂载点。

网卡信息：设备地址、延迟、所在 NUMA 节点、内核驱动程序、内核模块、子系统。

网口信息：网口名称、MAC 地址、IPv4、IPv6。

系统信息：主机名、OS 版本、内核版本、glibc 及 BI/OS 版本。



图 3-7 环境分析 – 系统设置信息

CPU 使用统计具体功能及指标如下图，其中：

使用率：%user、%nice、%system、%I/Owait、%hirq、%sirq、%steal、%guest、%gnice、%idle。

负载：runq-sz、plist-sz、ldavg-1、ldavg-5、ldavg-15。

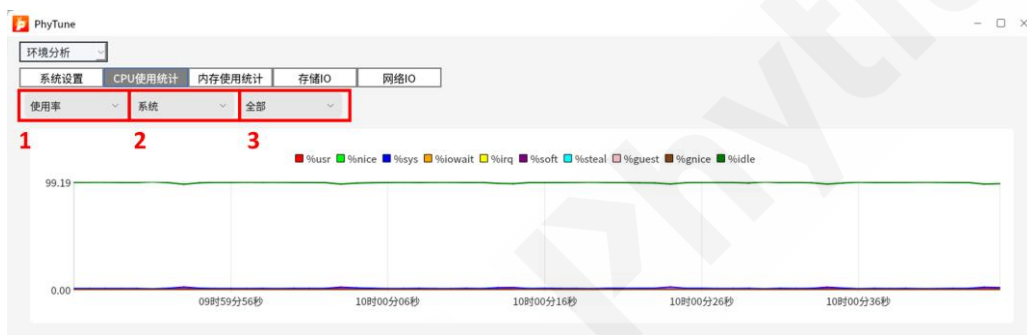


图 3-8 环境分析 - CPU 使用统计

内存使用统计具体功能及指标如下图，其中：

内存使用率：memtotal、memfree、memavail、memused、buffers、cached、active、inactive、dirty。

内存负载：pgpgin/s、pgpgout/s、fault/s、majflt/s、pgfree/s、pgscank/s、pgscand/s、pgsteal/s、%vmeff。

交换内存使用率：swaptotal、swapfree、swapused。

交换内存负载：pswpin/s、pswpout/s。

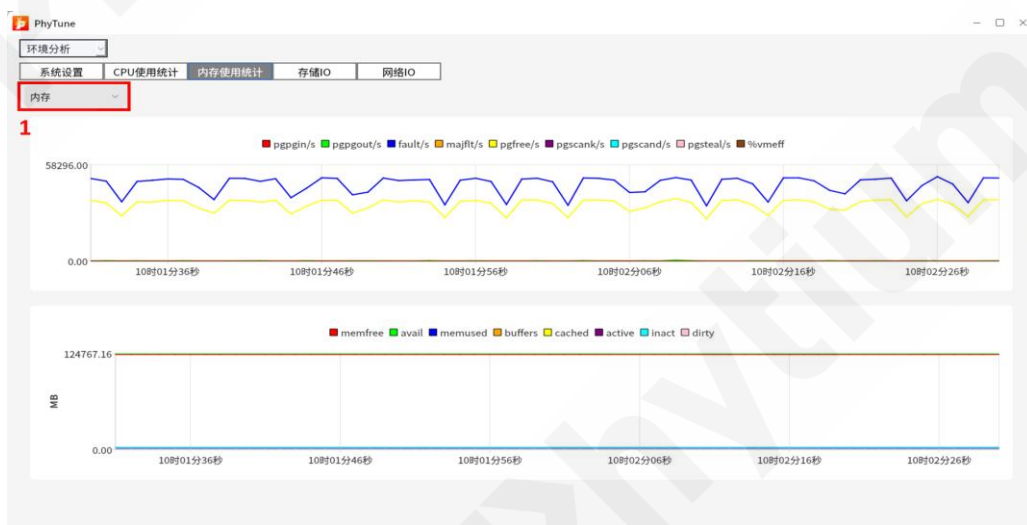


图 3-9 环境分析 - 内存使用统计

存储 I/O 具体功能及指标如下图，其中：

硬盘存储 I/O：rkB/s、wkB/s、areq-sz、aqu-sz、await、tps、%util。

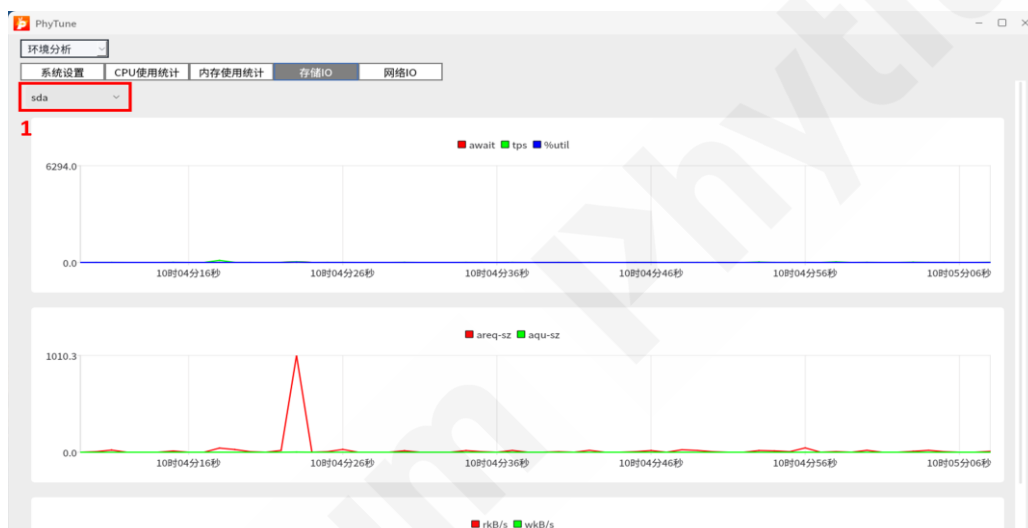


图 3-10 环境分析 - 存储 I/O

网络 I/O 具体功能及指标如下图，其中：

网口传输速率：rxkb/s、txkb/s、rxpck/s、txpck/s。

网口负载：rxdrop/s、txdrop/s、rxfifo/s、txfifo/s。

错误：rxerr/s、txerr/s。

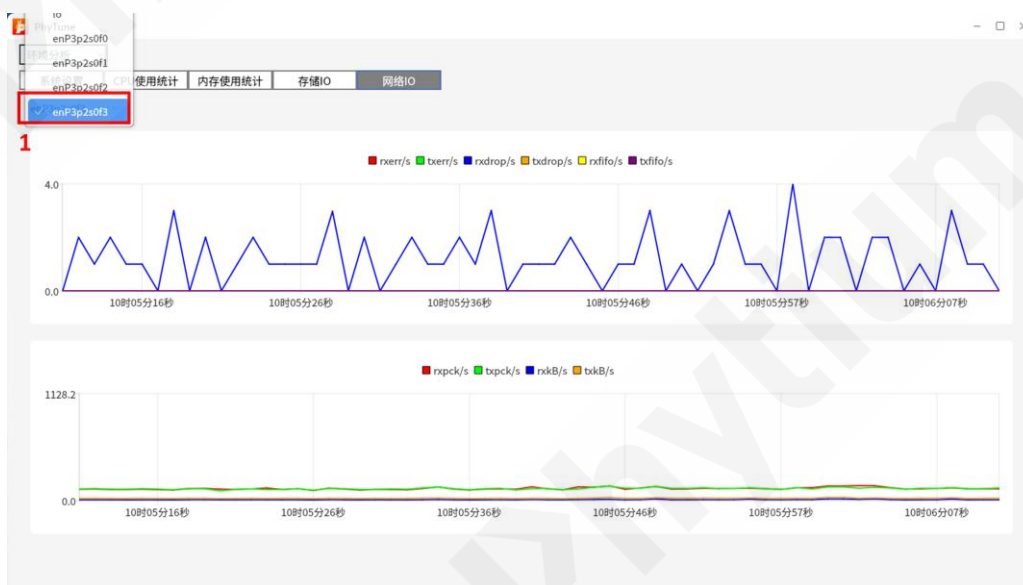


图 3-11 环境分析 - 网络 I/O

3.3 处理器分析

处理器分析的主要功能是从微架构层面分析程序运行期间的处理器性能，包括计算处理性能、内存访问性能、设备访问性能，以及对计算处理性能的 TopDown 分析等。

3.3.1 计算处理分析

3.3.1.1 分析配置

首先，选择待分析程序所在的目标机器（即之前配置的目标端 IP 地址）。



图 3-12 选择目标机器

然后输入目标机器上待分析的"目标程序"（全路径）：

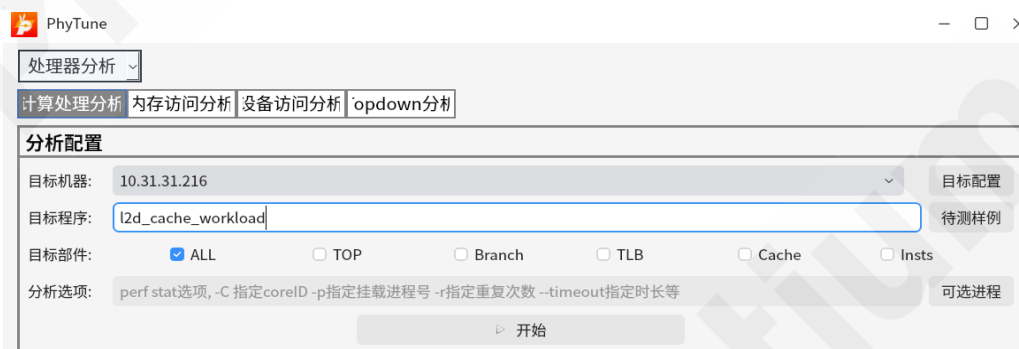


图 3-13 输入目标程序

提示：PhyTune 自带了部分用于测试目的的待分析程序，可通过“待测样例”按钮获取程序名称并复制到“目标程序”输入框中。

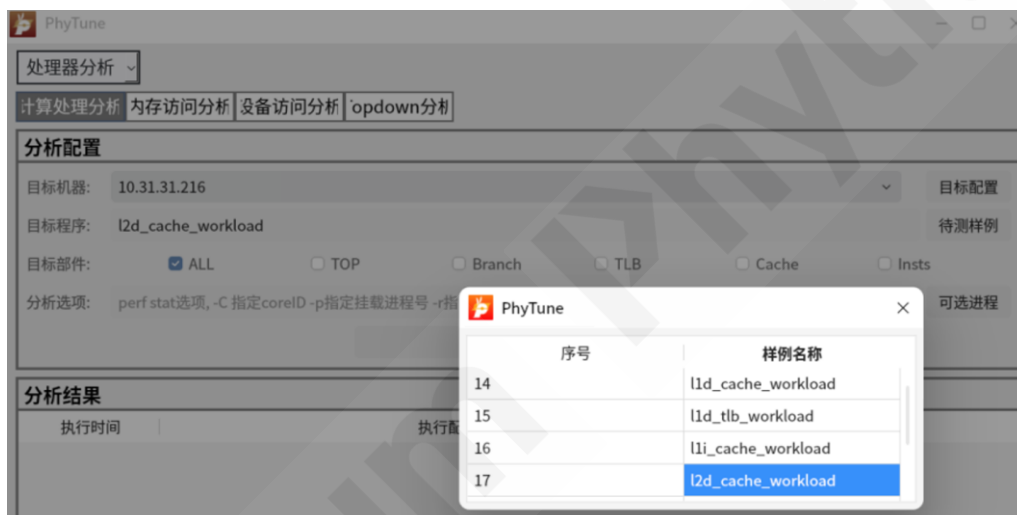


图 3-14 使用测试程序

接着,可以选择关注的计算处理目标部件性能,包括流水线整体性能 TOP、分支处理性能 Branch、高速缓存性能 TLB/Cache、指令执行类型 Insts 等,也可以直接勾选 ALL 选择所有目标部件。

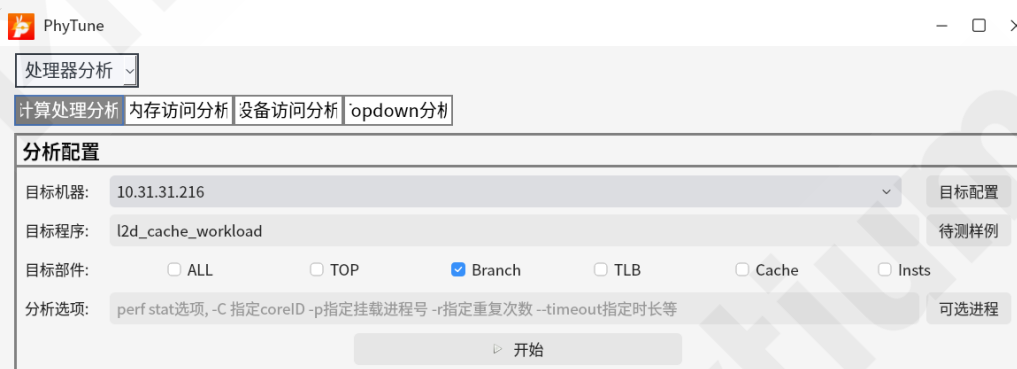


图 3-15 选择目标部件

说明: 除了以上常规配置操作外,还可以在“分析选项”中手动输入配置参数。目前支持-p 参数指定目标机器上当前已在运行的进程号(此时不需指定“目标程序”),--timeout 参数执行分析时长(而不是等待进程结束退出),-C 参数执行目标机器上的处理器运行核心编号,-r 执行分析任务重复次数。

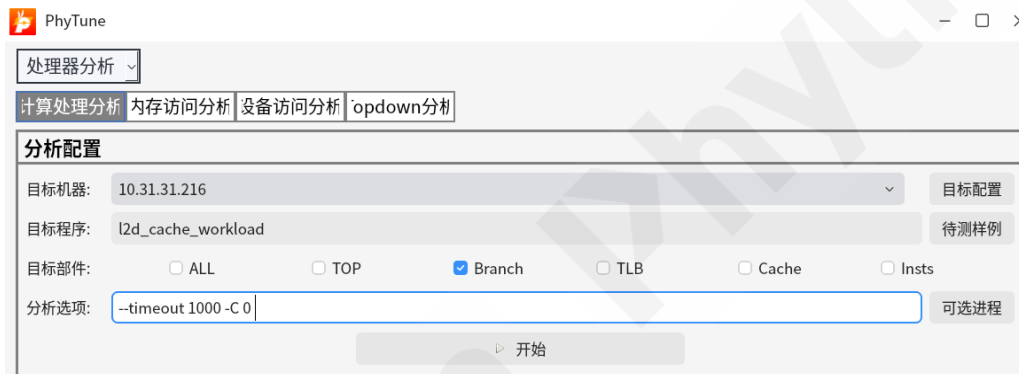


图 3-16 手动设置分析选项

3.3.1.2 分析执行

完成上述配置操作后，点击“开始”按钮，即可启动目标机器上关于目标程序的目标部件相关的计算处理性能分析，并实时显示分析运行状态。



图 3-17 分析运行状态显示

说明：在启动分析后，也可以在点击“停止”按钮中止正在运行中的分析任务。该情况下将没有分析结果输出。

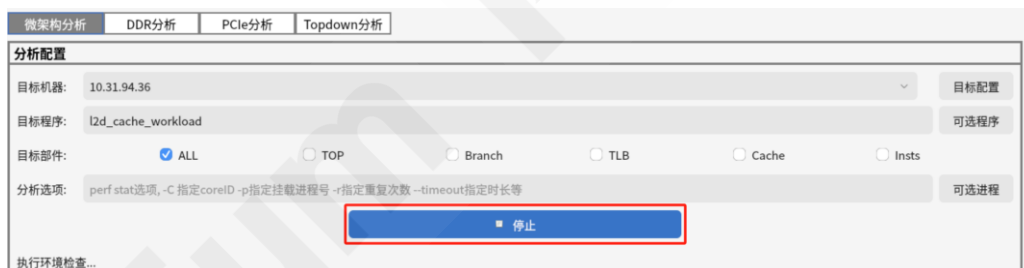


图 3-18 停止分析任务

3.3.1.3 结果查看

分析任务正常结束后，在分析结果列表中将生成最新结果项，包括执行时间、执行配置以及“查看”按钮。点击该按钮则可以查看本次性能分析的结果，包括表格和图形两种呈现形式。

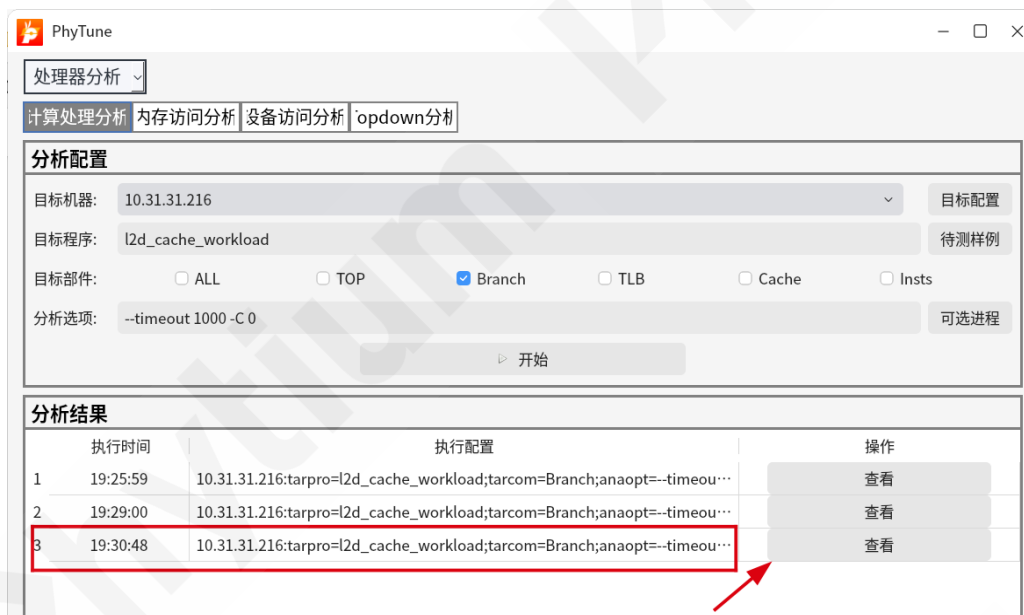


图 3-19 查看结果

下图是计算处理性能分析结果表示例。图中显示的 IPC 为 1.19，表示处理器平均每个时钟周期执行了 1.19 条指令。前端停滞率 STALL_FRNT_RT 为 24.97%，后端停滞率 STALL_BKND_RT 为 38.45%。前端停滞率高说明处理器在等待指令获取或解码，可能是由于指令缓存未命中、分支预测错误或其他前端问题导致的。后端停滞率高通常表示处理器在等待数据或资源，可能是由于缓存未命中、内存访问延迟或执行单元忙碌等原因。

此外，BR_MP_PKI 是每千条指令发生的分支预测错误次数，BR_MP_RT 是分支预测错误的百分比。

PhyTune

处理器分析

表格图形

一级指标名称	二级指标名称	统计结果值
▼ Cycle Effectiveness:	inst_retired	9526242
	cpu_cycles	7968084
	stall_frontend	1990351
	stall_backend	3064409
	[IPC]	1.19
	[STALL_FTND_RT]	24.97%
	[STALL_BKND_RT]	38.45%
▼ Branch Effectiveness:	inst_retired	10425852
	inst_spec	10864136
	br_mis_pred	6906
	br_pred	2503037
	br_mis_pred_retired	6141
	br_retired	2473394
	[BR_MP_PKI_SPEC]	0.63
	[BR_MP_RT_SPEC]	0.27%
	[SPEC_FLUSH_RT]	4.03%
	[BR_MP_PKI]	0.58
	[BR_MP_RT]	0.24%
> TLB Effectiveness:		
> CACHE & MEM Effectiveness:		
> Instructions Effectiveness:		

图 3-20 计算处理性能分析结果表

下图是计算处理性能分析结果图示例：

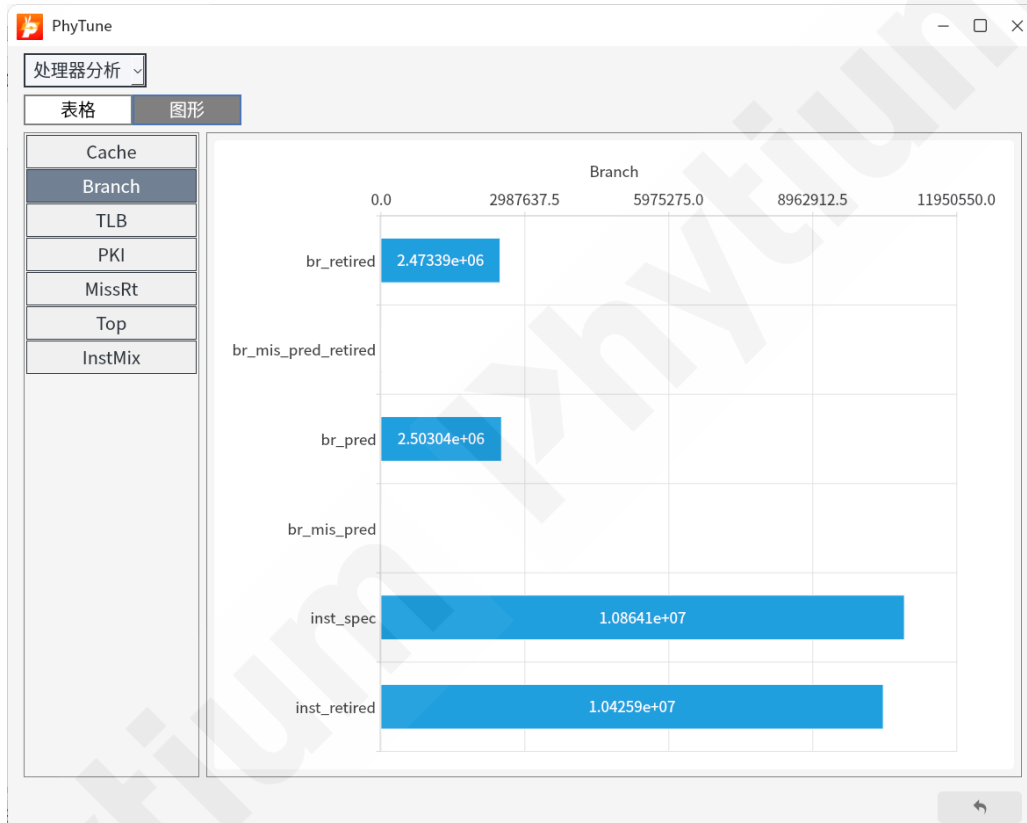


图 3-21 计算处理性能分析结果图（条状图）

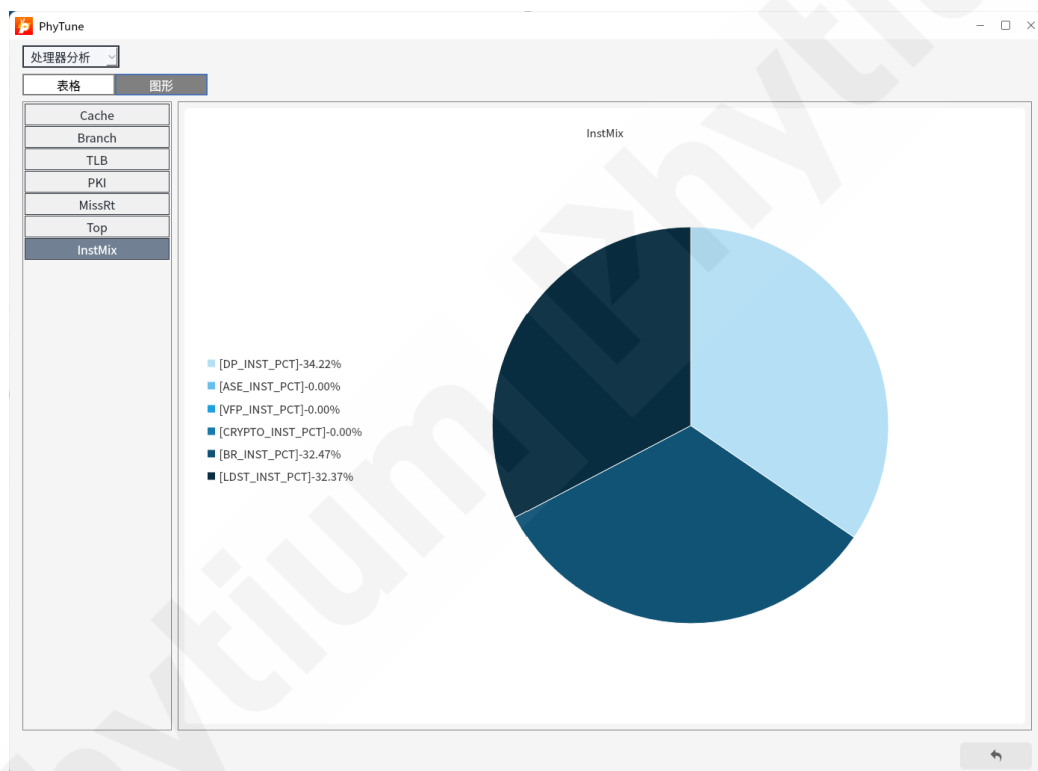


图 3-22 计算处理性能分析结果图（饼状图）

3.3.2 内存访问分析

3.3.2.1 分析配置

“目标机器”和“目标程序”的配置同前。

“NUMA 节点”指定所关注的内存访问相关的 NUMA 节点编号，例如对于双路 S5000C（128 核）服务器，取值范围是 0-7。

“HM 设备”指定相关的访存控制器编号，取值范围是 0 或 1。

“通道”指定相关的访存通道编号，取值范围是 0 或 1。

“分析选项”同前。



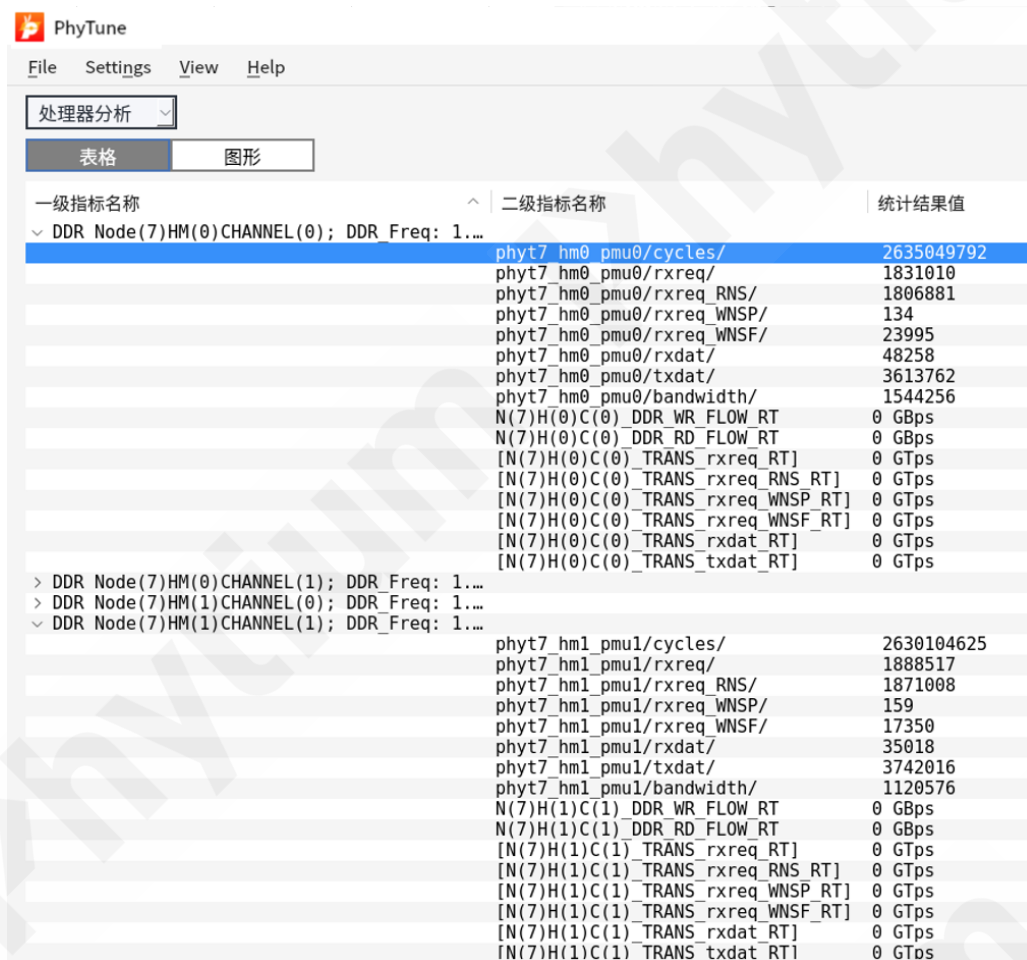
图 3-23 内存访问分析配置

3.3.2.2 分析执行

同前。

3.3.2.3 结果查看

下图是访存访问分析结果表示例。图中显示的 N(7)H(0)C(0)_DDR_WR_FLOW_RT 表示 7 号 NUMA 节点 0 号存控 0 号通道的写访存速率，N(7)H(0)C(0)_DDR_RD_FLOW_RT 表示 7 号 NUMA 节点 0 号存控 0 号通道的读访存速率。



一级指标名称	二级指标名称	统计结果值
DDR Node(7)HM(0)CHANNEL(0); DDR Freq: 1...	phyt7_hm0_pmu0/cycles/	2635049792
	phyt7_hm0_pmu0/rxreq/	1831010
	phyt7_hm0_pmu0/rxreq_RNS/	1806881
	phyt7_hm0_pmu0/rxreq_WNSP/	134
	phyt7_hm0_pmu0/rxreq_WNSF/	23995
	phyt7_hm0_pmu0/rxdat/	48258
	phyt7_hm0_pmu0/txdat/	3613762
	phyt7_hm0_pmu0/bandwidth/	1544256
	N(7)H(0)C(0)_DDR_WR_FLOW_RT	0 GBps
	N(7)H(0)C(0)_DDR_RD_FLOW_RT	0 GBps
	[N(7)H(0)C(0)_TRANS_rxreq_RT]	0 GTps
	[N(7)H(0)C(0)_TRANS_rxreq_RNS_RT]	0 GTps
	[N(7)H(0)C(0)_TRANS_rxreq_WNSP_RT]	0 GTps
	[N(7)H(0)C(0)_TRANS_rxreq_WNSF_RT]	0 GTps
	[N(7)H(0)C(0)_TRANS_rxdat_RT]	0 GTps
	[N(7)H(0)C(0)_TRANS_txdat_RT]	0 GTps
DDR Node(7)HM(0)CHANNEL(1); DDR Freq: 1...		
DDR Node(7)HM(1)CHANNEL(0); DDR Freq: 1...		
DDR Node(7)HM(1)CHANNEL(1); DDR Freq: 1...	phyt7_hm1_pmu1/cycles/	2630104625
	phyt7_hm1_pmu1/rxreq/	1888517
	phyt7_hm1_pmu1/rxreq_RNS/	1871008
	phyt7_hm1_pmu1/rxreq_WNSP/	159
	phyt7_hm1_pmu1/rxreq_WNSF/	17350
	phyt7_hm1_pmu1/rxdat/	35018
	phyt7_hm1_pmu1/txdat/	3742016
	phyt7_hm1_pmu1/bandwidth/	1120576
	N(7)H(1)C(1)_DDR_WR_FLOW_RT	0 GBps
	N(7)H(1)C(1)_DDR_RD_FLOW_RT	0 GBps
	[N(7)H(1)C(1)_TRANS_rxreq_RT]	0 GTps
	[N(7)H(1)C(1)_TRANS_rxreq_RNS_RT]	0 GTps
	[N(7)H(1)C(1)_TRANS_rxreq_WNSP_RT]	0 GTps
	[N(7)H(1)C(1)_TRANS_rxreq_WNSF_RT]	0 GTps
	[N(7)H(1)C(1)_TRANS_rxdat_RT]	0 GTps
	[N(7)H(1)C(1)_TRANS_txdat_RT]	0 GTps

图 3-24 内存访问分析结果表

3.3.3 设备访问分析

3.3.3.1 分析配置

“目标机器”和“目标程序”的配置同前。

“NUMA 节点”的配置同前。

“PCIe 设备”指定相关的 PCIe 根编号，例如对于 S5000C 服务器，取值范围是 0-2。

“控制器”指定相关的控制器编号，因 PCIe 根而不同。例如对于 S5000C 服务器，0 号 PCIe 设备对应的控制器取值范围是 0，1 号 PCIe 设备对应的控制器取值范围是 1-3，2 号 PCIe 设备对应的控制器取值范围是 4-7。

“遍历”选项勾选情况下，将忽略“NUMA 节点”、“PCIe 设备”、“控制器”等内容，直接遍历所有设备。

“分析选项”同前。

分析配置			
目标机器:	10.31.94.247		目标配置
目标程序:	l2d_cache_workload		可选程序
NUMA节点:	{0,1,...,8}, 缺省为0	PCIe设备:	{0,1,2}, 缺省为0
控制器:	{0/1-3/4-7}		☒ 遍历
分析选项:	perf stat选项, -C 指定coreID -p 指定挂进程号 -r 指定重复次数 -t timeout指定时长等		可选进程
开始 perf一键提权			

分析结果		
执行时间	执行配置	操作
1 14:56:01	10.31.94.247:tarpro=l2d_cache_workload;node_id=pmu_id=ctrlr_id=tra_switch=0;anaopt=	查看
2 14:56:47	10.31.94.247:tarpro=l2d_cache_workload;node_id=pmu_id=ctrlr_id=tra_switch=1;anaopt=	查看
3 15:14:06	10.31.94.247:tarpro=l2d_cache_workload;node_id=pmu_id=ctrlr_id=tra_switch=1;anaopt=	查看
4 15:17:03	10.31.94.245:tarpro=l2d_cache_workload;node_id=pmu_id=ctrlr_id=tra_switch=1;anaopt=	查看
5 16:23:22	10.31.94.247:tarpro=l2d_cache_workload;node_id=pmu_id=ctrlr_id=tra_switch=1;anaopt=	查看

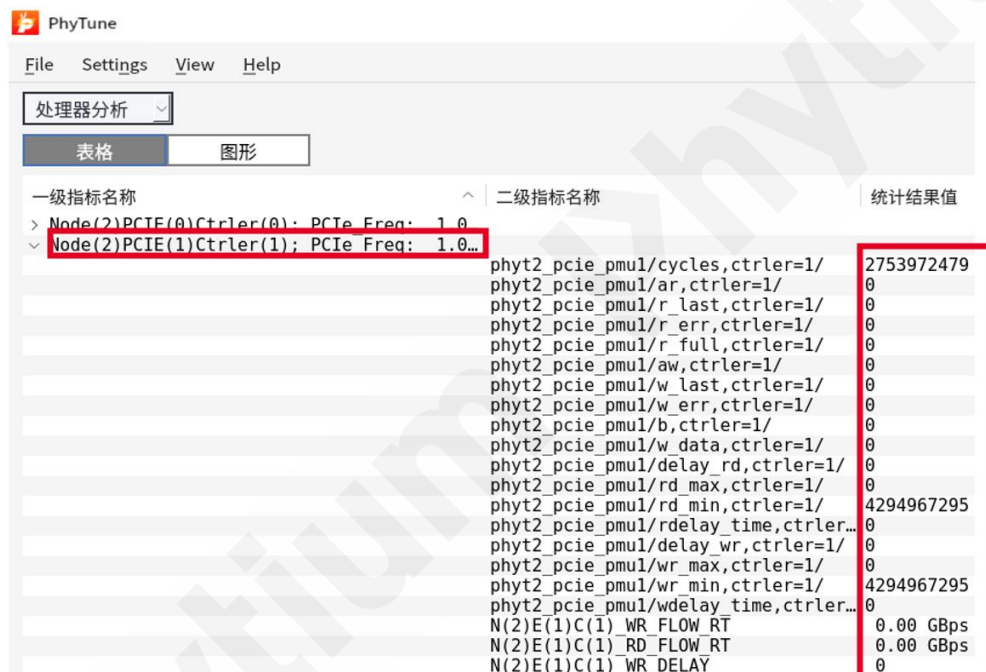
图 3-25 设备访问分析配置

3.3.3.2 分析执行

同前。

3.3.3.3 结果查看

下图是设备访问分析结果表示例。图中显示的 N(2)E(1)C(1)_WR_FLOW_RT 表示 2 号 NUMA 节点 1 号 PCIe 根 1 号控制器的写访问速率, N(2)E(1)C(1)_RD_FLOW_RT 表示 2 号 NUMA 节点 1 号 PCIe 根 1 号控制器的读访问速率, N(2)E(1)C(1)_WR_DELAY 表示 2 号 NUMA 节点 1 号 PCIe 根 1 号控制器的写访问延迟, 等。



一级指标名称	二级指标名称	统计结果值
> Node(2)PCIE(0)Ctrlr(0); PCIe Freq: 1.0		
Node(2)PCIE(1)Ctrlr(1); PCIe Freq: 1.0...		
	phyt2_pcie_pm1/cycles,ctrlr=1/	2753972479
	phyt2_pcie_pm1/ar,ctrlr=1/	0
	phyt2_pcie_pm1/r_last,ctrlr=1/	0
	phyt2_pcie_pm1/r_err,ctrlr=1/	0
	phyt2_pcie_pm1/r_full,ctrlr=1/	0
	phyt2_pcie_pm1/aw,ctrlr=1/	0
	phyt2_pcie_pm1/w_last,ctrlr=1/	0
	phyt2_pcie_pm1/w_err,ctrlr=1/	0
	phyt2_pcie_pm1/b,ctrlr=1/	0
	phyt2_pcie_pm1/w_data,ctrlr=1/	0
	phyt2_pcie_pm1/delay_rd,ctrlr=1/	0
	phyt2_pcie_pm1/rd_max,ctrlr=1/	0
	phyt2_pcie_pm1/rd_min,ctrlr=1/	4294967295
	phyt2_pcie_pm1/rdelay_time,ctrlr...	0
	phyt2_pcie_pm1/delay_wr,ctrlr=1/	0
	phyt2_pcie_pm1/wr_max,ctrlr=1/	0
	phyt2_pcie_pm1/wr_min,ctrlr=1/	4294967295
	phyt2_pcie_pm1/wdelay_time,ctrlr...	0
	N(2)E(1)C(1) WR FLOW RT	0.00 GBps
	N(2)E(1)C(1) RD FLOW RT	0.00 GBps
	N(2)E(1)C(1) WR DELAY	0

图 3-26 设备访问分析结果表

3.3.4 TopDown 分析

3.3.4.1 分析配置

“目标机器”和“目标程序”的配置同前。

“检测策略”指定 TopDown 的不同分析阶段，分为 stage1 和 stage2，stage1 阶段会分析流水线相关的 4 个指标，可帮助用户确定流水线的哪个阶段存在性能问题；stage2 阶段可以选择分析缓存、分支、页表等操作相关的指标，以帮助用户确定具体的性能瓶颈。

“分析指标”部分提供了三个层级的选择框供用户选择具体的分 TopDown 析指标或指标组。第一级的 TMA 表示总选择框，勾选后可选择所有指标/指标组；第二级有 4 个流水线相关的一级指标进行选择，勾选后可选择分析选中的流水线指标和其相关的二级指标组；第三级有 4 类勾选框，分别表示与相应流水线指标对应的二级指标组。

“间隔模式”是 TopDown 分析功能向用户提供的一种可设置间隔时长的周期性检测模式，以便观察不同周期间隔内各项指标的数值。



图 3-27 Topdown 分析配置界面

3.3.4.2 分析执行

使用 TopDown 分析目标程序时，“检测策略”首先建议选择 stage1，在 stage1 检测策略下，获取目标程序运行期间的流水线相关指标，明确目标程序运行期间流水线性能的分布情况。

完成 stage1 阶段的分析后，在“检测策略”中选择 stage2，此时分析指标变为可选状态，可以根据 stage1 阶段的分析结果合理选择需要关注的指标/指标组，“分析指标”的三级勾选框中，第二级勾选框可选择 4 个流水线相关的指标及其相关联的指标组。

若通过 stage1 明确目标程序的性能瓶颈与后端停顿占比较高有关，在 stage2 阶段可选择 backend_stall_pct 指标相关的所有指标组进行下一步。如下图所示。



图 3-28 Topdown stage2 分析配置界面

然而，用户在使用的时候也可以直接勾选 TMA，一次性分析所有的微架构相关的性能指标，然后再通过结果展示界面查看所有微架构相关的指标值，但 TopDown 是通过 perf 调用不同的 PMU 核心事件构建相关的性能指标进行分析，在 S5000C 平台下，仅支持同时调度 6 个非周期性 CPU 核心 PMU 事件，在相同程序运行时间内，统计的事件越多，经过 perf 的分时调度机制，统计相应指标所需的事件所分配的统计时长就越少。因此，建议用户遵循 TopDown 分析方法论使用本章节对应的功能，首先选择 stage1 阶段的指标组进行分析，确定流水线性能瓶颈，再根据 stage1 阶段的分析结果，选择 stage2 阶段的指标组进行下一步分析，确定目标程序出现性能问题的具体区域。

另外，若有间隔检测的需求，以便观察不同的时间间隔内各项指标的数值情况，用户可以选择勾选“间隔模式”，然后再输入间隔时长（单位：ms）。选择间隔时长时，间隔时长不宜过短，否则没有足够的时间获取所选的性能指标数据；同时，间隔时长也不宜超过目标程序/进程的运行时间，否则视为非间隔模式。

其他操作，同处理器分析。

3.3.4.3 结果查看

Topdown 分析的结果有表与图两种方式展示形式，选择“表格”标签，可以看到不同的指标及其结果，并通过绿、蓝、橙、浅红、深红五种颜色反映某项指标的性能优劣程度，由绿到深红分别代表性能由好到差，如下图所示。



图 3-29 结果表展示界面

结果表界面将用户在分析配置阶段选择的指标组/指标按照指标组的类别进行了区分，用户可以通过点击指标组查看组内指标的结果值、单位。Cycle_Accounting 指标组下存放了 stage1 阶段相关的指标组，表示流水线相关的指标；Branch_Effectiveness 指标组存放了评估分支指令有效性的相关指标，可用于综合评估分支操作的性能；DTLB_Effectiveness 指标组存放了数据页表缓存相关的指标；ITLB_Effectiveness 指标组存放了指令页表缓存相关的指标；L1I_Cache_Effectiveness 指标组存放了 L1 指令缓存相关的性能指标；L1D_Cache_Effectiveness 指标组存放了 L1 数据缓存相关的指标；L2_Cache_Effectiveness 指标组存放了 L2 缓存相关的指标；Operatl/On_Mix 指标组存放了微操作相关的性能指标，用于观察不同类型的微操作执行的分布情况。Miss_Ratl/O 指标组存放了不同处理器资源失误差率相关的指标。MPKI 指标存放了不同的处理器资源的在每千条指令中的未命中次数的相关指标。各指标组下的指标含义如下表所示。

表 3-1 TopDown 指标

指标	描述
frontend_stall_pct	每个周期内流水线前端停顿的百分比。

backend_stall_pct	每个周期内流水线后端停顿的百分比。
retired_pct	每个周期内流水线指令正常退役的百分比。
bad_speculatI/On_pct	每个周期内因流水线刷新未进行架构性执行的前瞻性执行指令的百分比，表示分支预测错误的占比情况。
ipc	每个周期内退役指令的数量。
branch_mpki	每千条执行的指令中分支预测错误的数量。
br_inst_pct	分支指令在前瞻性执行的指令中所占的百分比。
itlb_mpki	每千条执行的指令中 TLB 遍历次数。
l1i_tlb_mpki	每千条执行的指令中一级指令 TLB 访问丢失的次数。
l2_tlb_mpki	每千条执行的指令中二级 TLB 访问丢失的次数。
itlb_walk_rt	指令 TLB tabel page walk 与指令 TLB 访问的比率。
l1i_tlb_miss_rt	L1 指令 TLB 访问丢失与 L1 指令 TLB 访问的比率。
l2_tlb_miss_rt	L2 TLB 访问丢失与 L2 TLB 访问总数的比率。
dtlb_mpki	每千条执行的指令中数据 TLB tabel page walk 次数。
l1d_tlb_mpki	每千条执行的指令中数据 TLB tabel page walk 次数。
dtlb_walk_rt	数据 TLB tabel page walk 与数据 TLB 访问总数的比率。
l1d_tlb_miss_rt	L1 数据 TLB 访问丢失与 L1 数据 TLB 访问总数的比率。
l1i_cache_mpki	每千条执行的指令中 L1 指令缓存未命中的数量。
l1i_cache_miss_rt	L1 指令缓存未命中的数量与 L1 指令缓存访问的比率。
l1d_cache_mpki	每千条执行的指令中 L1 数据缓存未命中的数量。
l1d_cache_miss_rt	L1 数据缓存未命中的数量与 L1 数据缓存访问的比率。
mem_err_rt	访存错误率，是本地内存错误与内存访问的比率。
load_pct	前瞻性指令的读指令占前瞻性执行指令中的百分比。
store_pct	前瞻性指令的写指令占前瞻性执行指令中的百分比。
mem_inst_pct	访存指令在前瞻性执行指令中的百分比。
integer_dp_pct	逻辑或算数指令在前瞻性执行指令中的百分比。
simd_pct	前瞻执行的 ASIMD 指令在前瞻性执行指令中的百分比。
scalar_fp_pct	前瞻执行的浮点指令在前瞻性执行指令中的百分比。
branch_pct	分支操作指令在前瞻性执行指令中的百分比。
crypto_pct	前瞻执行加密指令在前瞻性执行指令中的百分比。
pcw_inst_pct	导致程序计数器 PC 更改的前瞻性执行指令的百分比,包括分支、以 PC 为目标的读指令、SMC、HVC、BKPT 等指令。
l2_cache_mpki	每千条执行的指令中 L2 缓存未命中的数量。
l1i_cache_miss_rt	L1 数据缓存未命中的数量与 L1 数据缓存访问的比率。

l2_cache_miss_rt

L1 数据缓存未命中的数量与 L1 数据缓存访问的比率。

另外，还支持饼状图、柱状图、分簇柱状图等形式展示不同的指标结果，如下图所示。

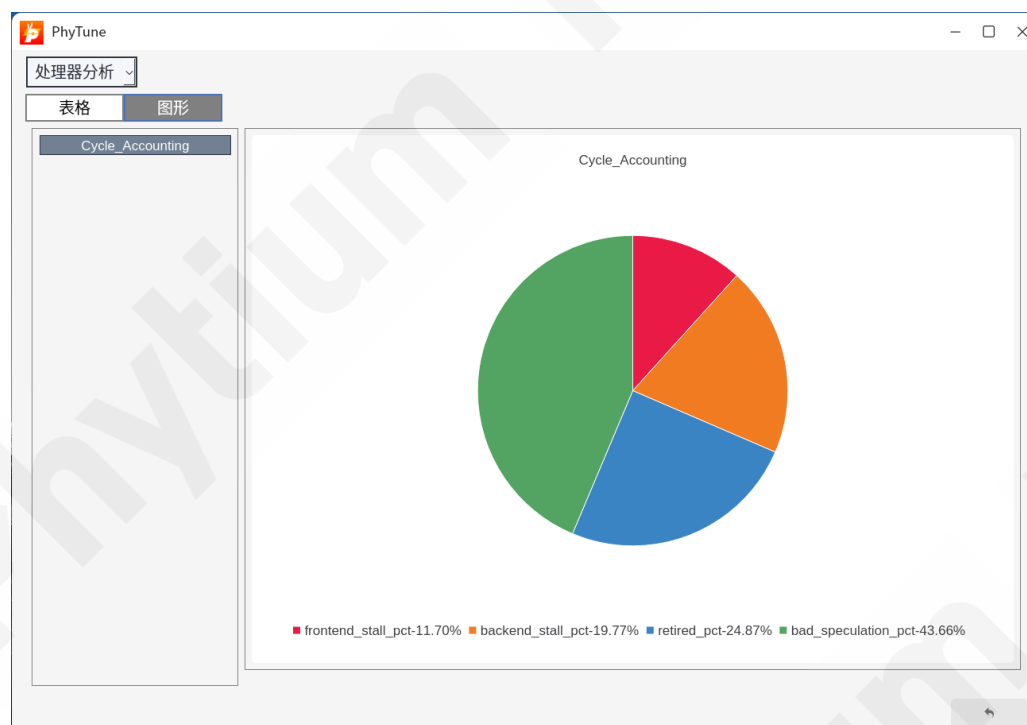


图 3-30 stage1 指标分布饼状图

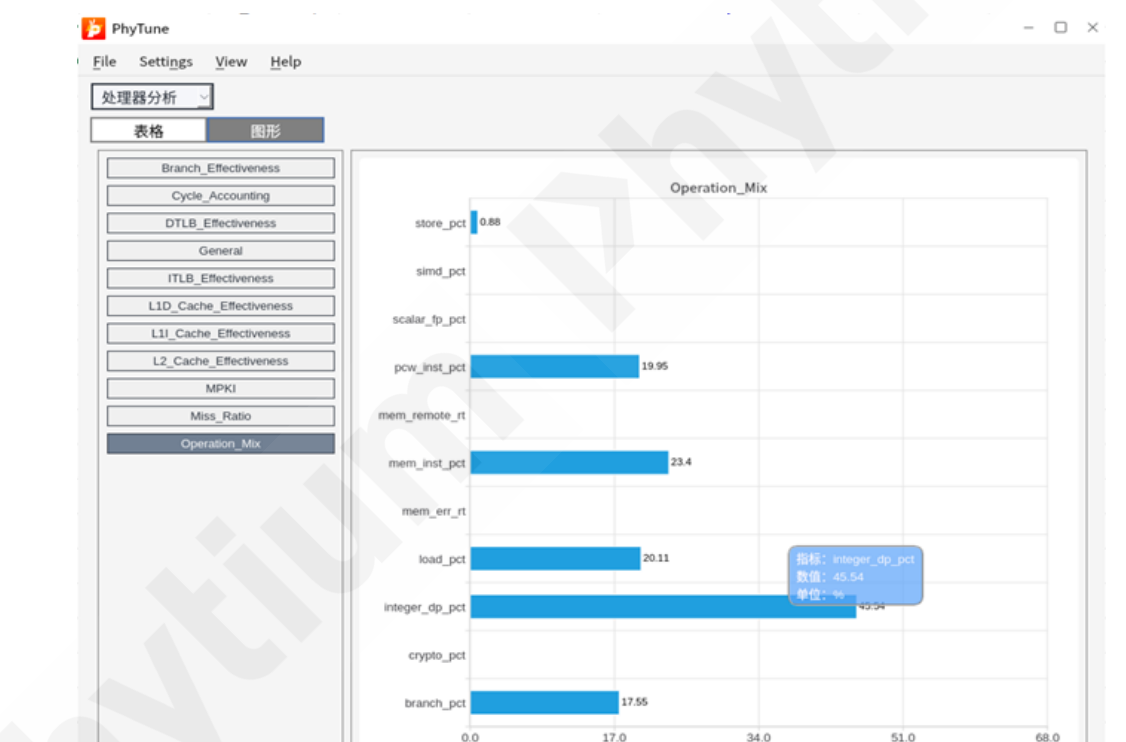


图 3-31 指标组各指标柱状图

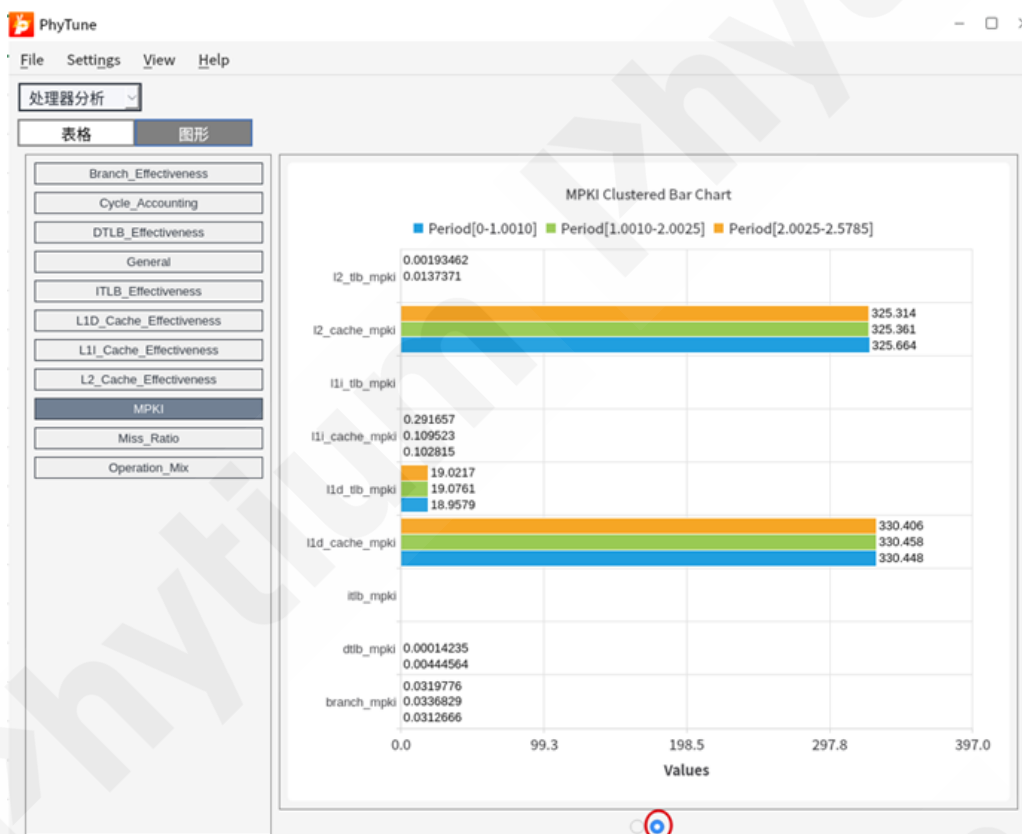


图 3-32 各时间间隔下各指标的分簇柱状图

其中，分簇柱状图只有在间隔模式下才会进行展示，间隔模式用于观察程序在不同的周期间隔内的性能指标情况，可以分析目标程序在不同周期内的性能指标结果。

3.4 访存分析

访存分析的主要功能是分析系统或应用程序对各级高速缓存的访问性能，并基于函数调用栈采集函数调用相关的高速缓存访问不命中状态信息。

3.4.1 系统级分析

系统级分析是对整个系统对象进行访存分析，包括“Hit 分析”和“Miss 分析”两项功能，每项分析功能又支持实时和记录两种分析模式。

3.4.1.1 分析配置

首先，选择待分析程序所在的目标机器。

然后选择分析功能，不同的分析功能支持不同的采样指标。

如果选择“Hit 分析”，将触发对目标机器的状态检查，左下角显示该机

器支持的采样指标，并且进行自动勾选，不支持的采样指标则自动关闭。“Hit 分析”记录模式需要输入采样间隔和采样时长，实时模式仅需输入采样间隔。



图 3-33 系统级访存分析 – Hit 分析

如果选择“Miss 分析”，需要点击 cpu 列表的下拉框手动勾选待分析的 CPU 编号，然后点击确认按钮将选好的 cpu id 填写到文本框中。“Miss 分析”记录模式需要输入采样时长，实时模式需输入采样间隔。

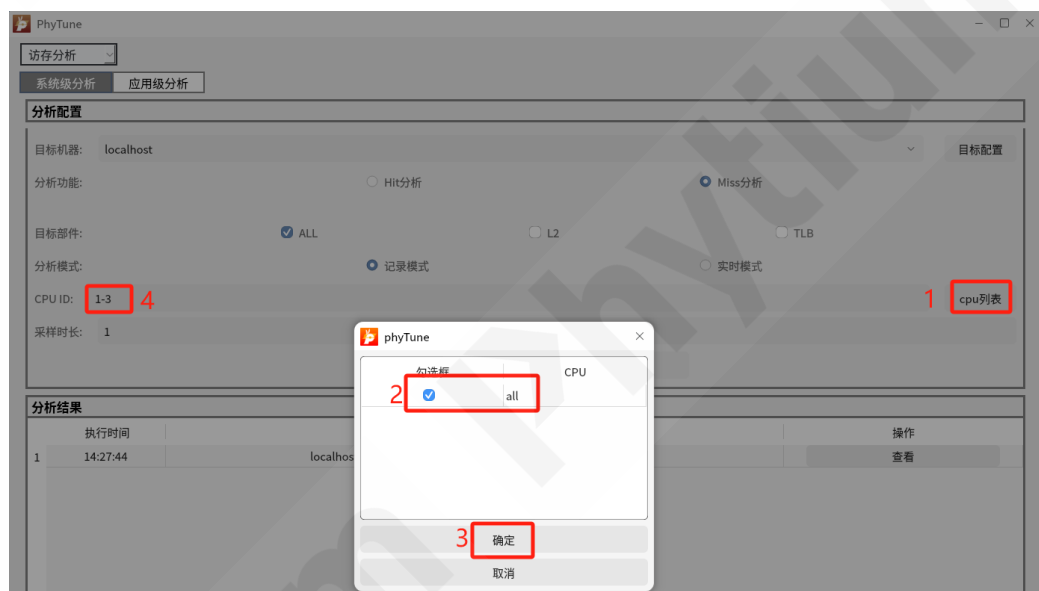


图 3-34 系统级访存分析 – Miss 分析

3.4.1.2 分析执行

记录模式完成上述配置操作后，点击“开始”按钮，即可启动目标机器上的系统级分析的记录模式，并实时显示记录模式的分析运行状态。



图 3-35 系统级访存分析 - Hit 分析记录模式状态显示

实时模式完成上述配置操作后，点击“开始”按钮，即可启动目标机器上的系统级分析的实时模式，并等待实时进程开启，将跳转到实时页面。



图 3-36 系统级访存分析 - Hit 分析实时模式状态显示

说明：在启动分析后，也可以在点击“停止”按钮中止正在运行中的分析任务。该情况下将没有分析结果输出。

3.4.1.3 结果查看

(1) 记录模式

记录模式的分析任务正常结束后，在分析结果列表中将生成最新结果项，包括执行时间、执行配置以及“查看”按钮。点击该按钮则可以查看本次性能分析的结果。



图 3-37 系统级访存分析 - Miss 分析记录模式状态显示

①Hit 分析

“Hit 分析”基于 CPU 访问缓存和内存的 PMU 事件，分析存储的访问次数、命中率、带宽等情况。下图是“Hit 分析”的表格形式结果示例。

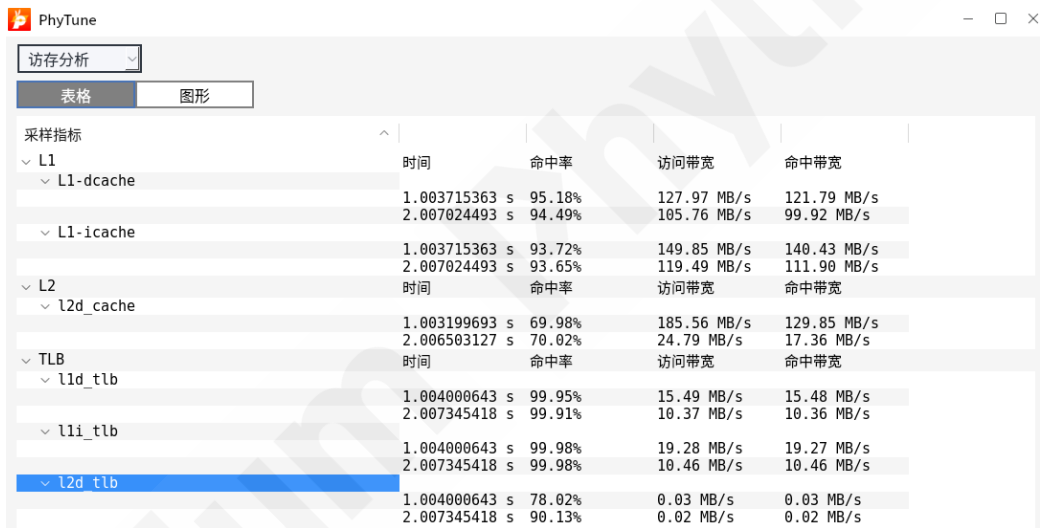


图 3-38 Hit 分析记录模式结果表

表格展示界面中显示缓存访问的相关采样指标在不同时间节点的命中率、

访问带宽和命中带宽，以及内存访问的采样指标在不同时间节点的缓存访问带宽、缓存命中带宽以及读/写带宽。其中，缓存的命中率是一个重要指标，它表示了程序访问缓存的效率。一个高的缓存命中率表示访问了大量的缓存数据而不是从较慢的主存中读取数据，从而提高了程序的性能。

下图以折线图展示各时间节点对应的指标结果。当鼠标悬停到折线图上的点时，将弹出提示框显示该折线图的名称以及坐标点。

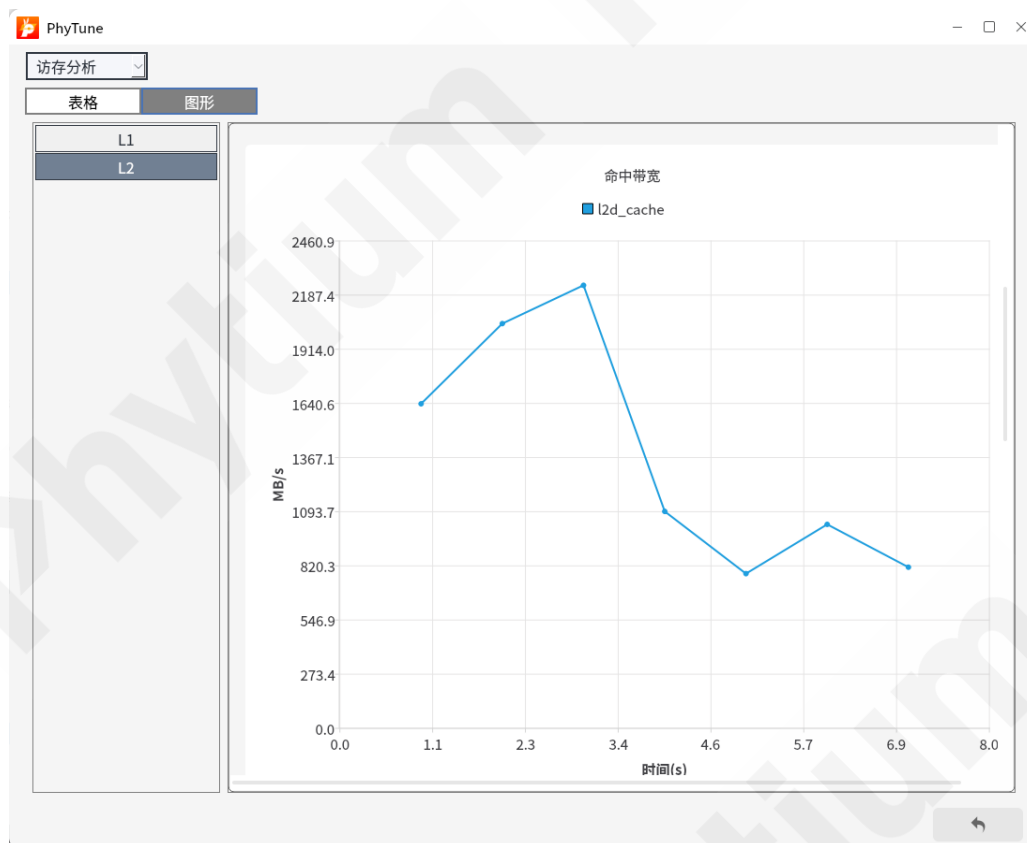


图 3-39 Hit 分析记录模式结果图

②Miss 分析

“Miss”分析基于对 miss 事件的采样，并根据函数调用栈采集函数调用期间引发的高速缓存访问不命中信息。下图是“Miss 分析”的表格形式结果示例。

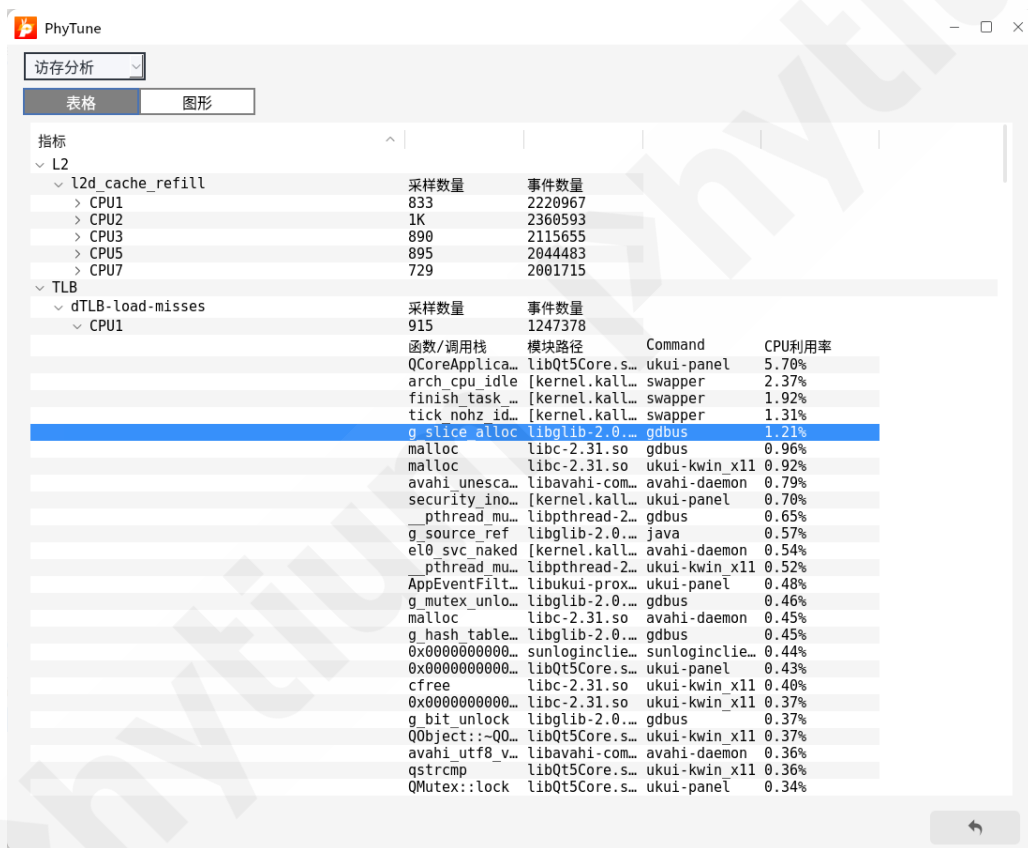


图 3-40 Miss 分析记录模式结果表

L2 Miss：内存请求在第二级 cache 中未命中次数的比率。

TLB Miss：CPU 在内存访问或取指过程中，在 TLB 中没有找到虚拟地址到物理地址映射次数的比率。

在表格展示界面中，显示 Miss 事件在不同 CPU 上的函数调用栈表格，结果展示按照 Miss 次数由大到小进行排序。

在函数/调用栈中的位置和模块路径能够揭示哪些代码路径导致了缓存未命中。定位这些未命中的位置可以帮助开发人员优化访问模式，减少对缓存的依赖，提高数据局部性。还可以揭示哪些代码部分频繁引起 TLB 缺失，从而优化内存访问模式。

除表格外，还支持水平堆叠柱状图图形展示采样期间的 miss 事件数量和采样数量，如下图所示。当鼠标悬停到柱体上时，将弹出提示框显示该柱体的名称以及数值。

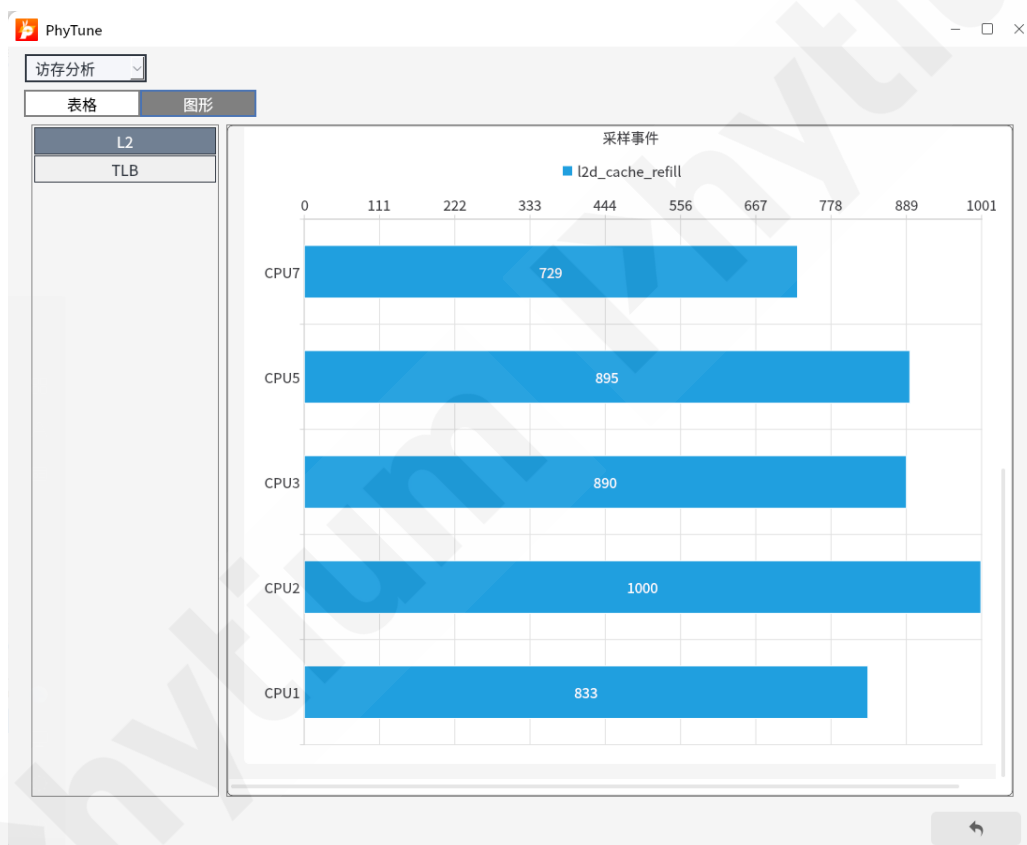


图 3-41 Miss 分析记录模式结果图

(2) 实时模式

①Hit 分析

实时进程开启后，将跳转到实时界面进行动态展示。下面是“Hit 分析”的实时结果图，动态展示缓存指标实时更新的命中率、访问带宽和命中带宽，和 DDR 的读/写带宽。



图 3-42 Hit 分析实时模式结果图

点击“停止”按钮之后，将关闭所有的实时进程，并清除实时结果，然后点击返回按钮返回主页面。

②Miss 分析

下面是“Miss 分析”的实时结果图，动态展示 miss 事件在不同 cpu 上实时更新的结果。

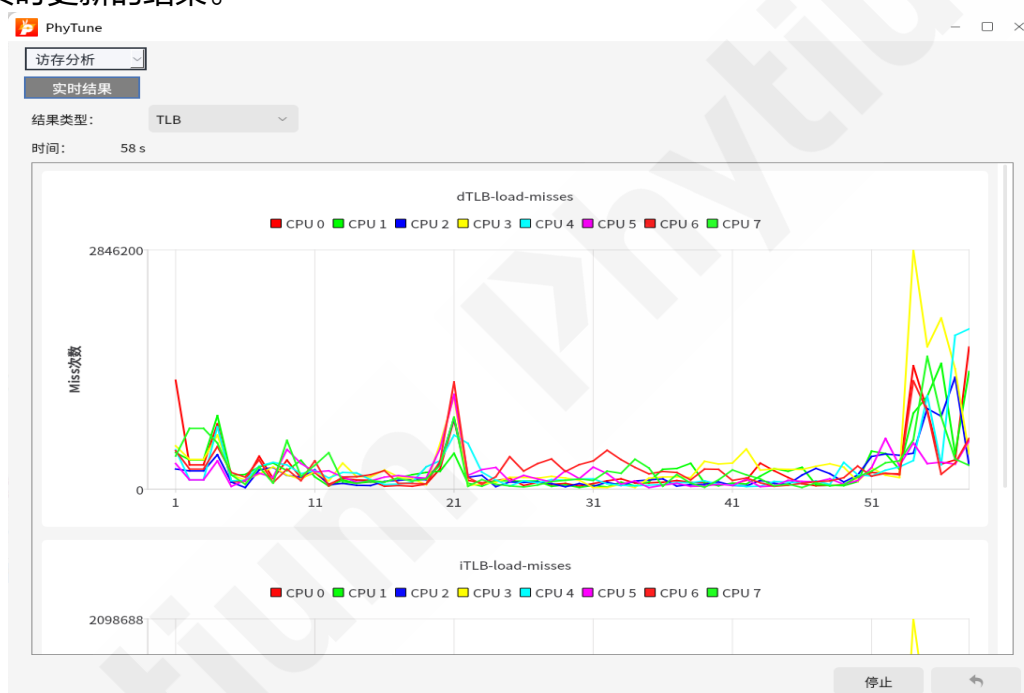


图 3-43 Miss 分析实时模式结果图

3.4.2 应用级分析

应用级分析是对“程序”或“进程”对象进行访存分析，包括“Hit 分析”和“Miss 分析”两项功能，每项功能又支持实时和记录两种分析模式。

3.4.2.1 分析配置

首先，选择待分析程序所在的目标机器。

然后选择分析对象。“程序”对象需要输入目标程序和程序参数。可通过点击“程序文件浏览”按钮获得待测程序的绝对路径。



图 3-44 选择目标程序

“进程”对象需要点击进程下拉框手动勾选待测进程，然后点击“确定”按钮获取进程号自动填写到“目标进程”输入框中。

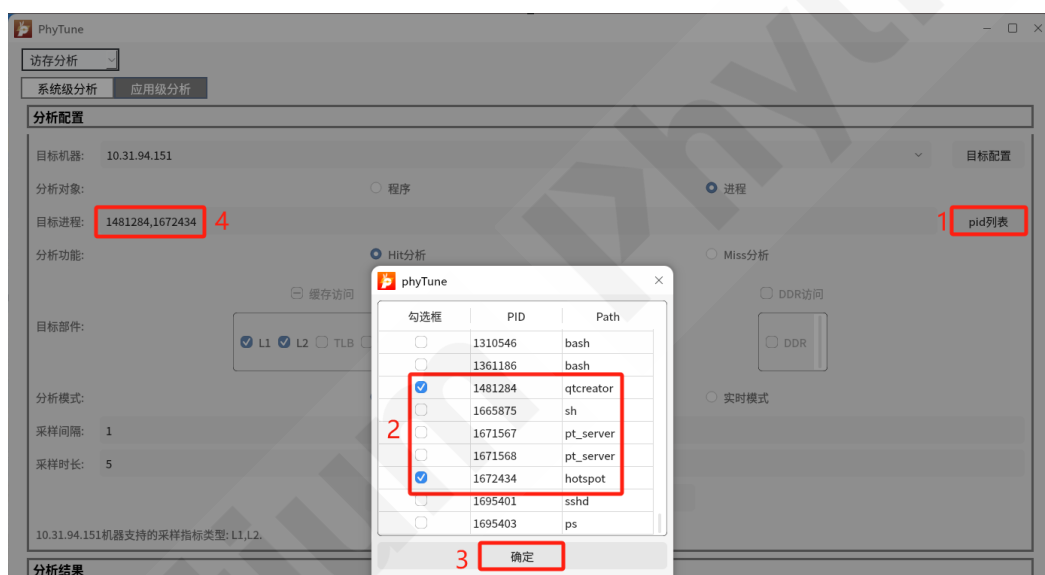


图 3-45 选择目标进程

继续选择分析功能，“Hit 分析”和“Miss 分析”功能选择同系统级分析一致。

最后选择分析模式，记录模式和分析模式的参数输入同系统级分析一致。

3.4.2.2 分析执行

记录模式完成上述配置操作后，点击“开始”按钮，即可启动目标机器上的系统级分析，并实时显示记录模式的分析运行状态。



图 3-46 应用级访存分析 – Hit 分析记录模式状态显示

实时模式完成上述配置操作后，点击“开始”按钮，即可启动目标机器上的系统级分析的实时模式，并等待实时进程开启，将跳转到实时页面。



图 3-47 应用级访存分析 – Hit 分析记录模式状态显示

说明：在启动分析后，也可以在点击“停止”按钮中止正在运行中的分析任务。该情况下将没有分析结果输出。

3.4.2.3 结果查看

分析任务正常结束后，在分析结果列表中将生成最新结果项，包括执行时间、执行配置以及“查看”按钮。点击该按钮则可以查看本次性能分析的结果。



图 3-48 应用级访存分析 – Hit 分析记录模式状态显示

访存分析的应用级分析结果和系统级分析结果的展示形式一致。

3.5 NUMA 分析

NUMA(非一致性内存访问)分析功能对特定进程进行深入的 NUMA 分析，以优化进程调度和内存分配策略。通过监控进程及其线程在 NUMA 节点间的资源使用，帮助程序提高性能和资源利用率。

3.5.1.1 分析配置

分析配置涉及启动进程、资源调度和分析进程三个功能模块。

启动进程页面自动绑核功能如下图，其中：

绑核配置：修改绑核配置列表。

目标应用：从绑核配置列表中选择其一。

应用参数：待测应用程序参数，可选填。

启动：在目标机器上启动目标应用。

停止：在目标机器上杀死目标应用。



图 3-49 NUMA 分析 – 启动进程

绑核配置页面如下图，其中：

应用程序：由文件浏览窗口获取目标可执行程序绝对路径。

内存节点：待绑定的 NUMA 节点。

CPU 核：待绑定的 CPU 节点。

添加：将新的应用程序绑核配置添加到列表中。

删除：将勾选的应用程序绑核配置从列表中删除。

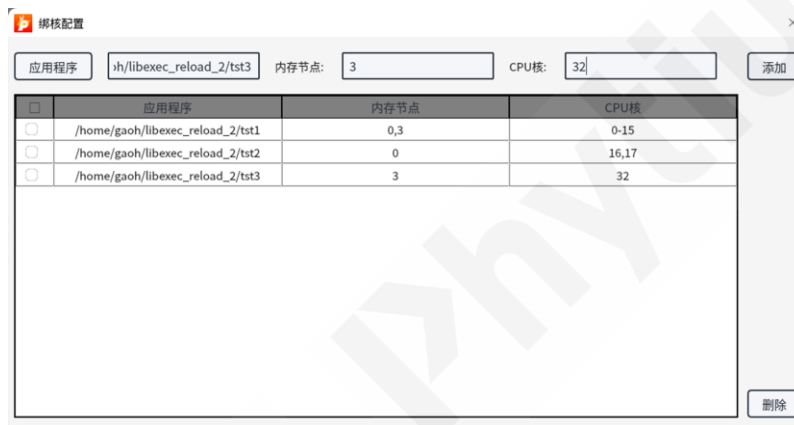


图 3-50 NUMA 分析 – 绑核配置

启动进程界面手动绑核功能如下图，其中：

目标应用：由文件浏览窗口获取目标可执行程序绝对路径，粘贴到输入框内。

应用参数：待测应用程序参数，可选填。

进程调度策略：节点表示只在指定节点上执行目标应用。CPU 核表示只在指定 CPU 核上执行目标应用。

内存分配策略：本地表示在进程本地节点上分配内存，输入框不填。交错表示只在指定节点上交错分配内存。优先表示在指定节点(单节点)上优先分配内存，如果指定节点内存剩余空间不足，再分配到其他节点上。绑定表示只在指定节点上分配内存。

启动：在目标机器上启动目标应用。

停止：在目标机器上杀死目标应用。



图 3-51 Numa 分析 - 手动绑核

分析进程界面如下图，其中：

目标进程 pid：获取目标机器上的用户级进程列表。

实时模式：根据采样间隔(s)持续采样，实时更新分析结果，直到手动停止，适用于长期监控与性能跟踪的场景。

记录模式：根据采样间隔(s)和采样次数采样结束后停止，提供一次采样时间段内的分析结果，适用于短期性能分析和瞬时状态诊断的场景。

采样间隔：以秒为采样间隔单位。

采样次数：满足采样次数后结束分析。



图 3-52 Numa 分析 - 分析进程

3.5.1.2 分析执行

点击分析配置页面“开始”按钮，等待连接目标机器、启动 NUMA 分析工具。

3.5.1.3 结果查看

分布情况如下图。其中：

CPU 亲和性：指定了进程可以在哪些 CPU 核上运行。

内存亲和性：指定进程的内存分布信息，包括每个内存区域所在的 NUMA 节点。

内存迁移：将进程的内存页从原始节点迁移到目标节点上。优化内存分布，减少跨节点的内存访问。

线程迁移：改变进程的 CPU 亲和性，将目标线程迁移到目标 CPU 核上。避免多线程任务之间的 CPU 资源竞争。

目标线程 tid：获取目标机器上待测进程的子线程列表。

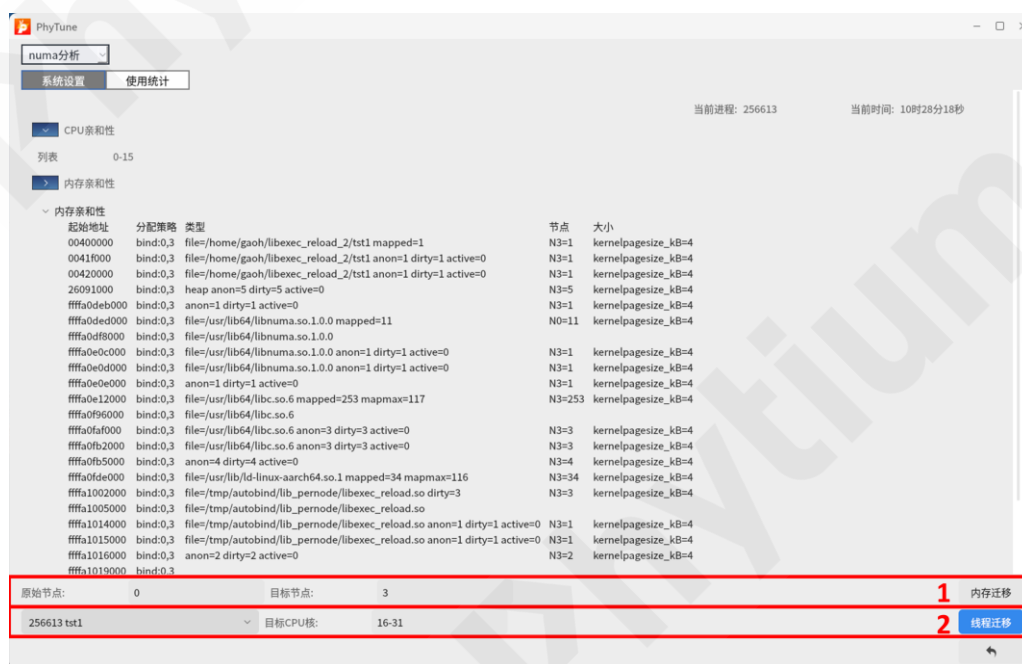


图 3-53 NUMA 分析 – 系统设置结果

进程使用统计如下列各图。其中：

CPU 使用统计：查看进程在各 numa 节点的平均使用率。

内存使用统计：



图 3-54 NUMA 分析 – CPU/内存使用统计

线程使用统计如下图。

线程 tid：获取目标机器上待测进程的子线程列表。

CPU 核：

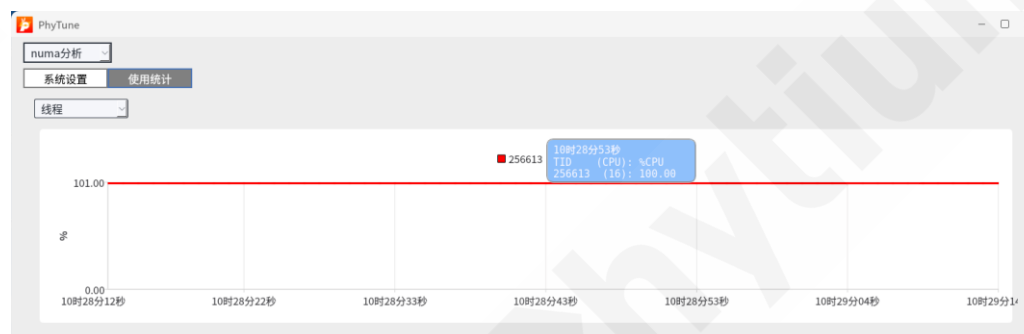


图 3-55 NUMA 分析 – 线程使用统计

3.6 I/O 分析

I/O 分析主要以存储块设备为分析对象，分析 I/O 访问生命周期得出块设备访问的性能数据，并周期性的实时统计设备请求的大小和时延的分布信息。

此外，支持跟踪指定的程序或进程，记录并解析相关的具体 I/O 操作事件、进程/线程 ID、调用栈、应用层 I/O 接口等信息。根据 I/O 性能数据分析可以

给出优化建议。

3.6.1 系统级分析

3.6.1.1 分析配置

I/O 系统级分析对象为存储块，根据分析模式可以分为实时模式和记录模式。

记录模式需要输入块设备名称(如 sda)和采样时长来监控 I/O 生命周期。



图 3-56 系统级 I/O 分析 - 记录模式

实时模式输入块设备名称和采样周期进行实时监控。



图 3-57 系统级 I/O 分析 - 实时模式

3.6.1.2 分析执行

记录模式完成上述配置操作后，点击“开始”按钮，即可启动目标机器上的系统级分析的记录模式，并实时显示记录模式的分析运行状态。



图 3-58 系统级 I/O 分析 - 记录模式状态显示

实时模式完成上述配置操作后，点击“开始”按钮，即可启动目标机器上的系统级分析的实时模式，并等待实时进程开启，将跳转到实时页面。

说明：在启动分析后，也可以在点击“停止”按钮中止正在运行中的分析任务。该情况下将没有分析结果输出。

3.6.1.3 结果查看

(1) 记录模式

分析任务正常结束后，在分析结果列表中将生成最新结果项，包括执行时间、执行配置以及“查看”按钮。点击该按钮则可以查看本次性能分析的结果，包括表格和图形和文本记录三种呈现形式。



图 3-59 系统级 I/O 分析 - 记录模式状态显示

下图是 I/O 分析的系统级分析的表格展示结果：

1) 各阶段时延信息。

- ① 整个设备和进程的 I/O 时延
- ② d2c、q2c、q2d 时延的时序信息

各阶段时延信息表

阶段	最小时延	平均时延	最大时延	时延次数
> Q2Q	0.000000646	0.148823300	4.208188157	99
> Q2G	0.000000411	0.000004468	0.000016185	44
> G2I	0.000000641	0.000037671	0.000207053	38
> Q2M	0.000000132	0.000000644	0.000003053	56
> I2D	0.000006986	0.000033047	0.000056599	44
> M2D	0.000015766	0.000029006	0.000161936	50
> D2C	0.000048554	0.000273288	0.000539646	100
> Q2C	0.000066823	0.000329272	0.000760663	100

图 3-60 系统级 I/O 分析 - 各阶段时延信息

2) 读写请求数信息。

读写请求数表							
8,0							
CPU	读-Q请求数	写-Q请求数	读-D请求数	写-D请求数	读-C请求数	写-C请求数	读-M请求数
CPU50	0	55	0	10	0	10	45
CPU51	0	6	0	3	0	3	3
CPU59	0	39	0	31	0	31	8
Total	0	100	0	44	0	44	56

图 3-61 系统级 I/O 分析 - 读写请求信息

- 3) 设备寻道信息：衡量连续的 Q2Q（接收-排队）和 D2D（提交-发送）I/O 之间扇区距离的变化性。

设备寻道信息表				
D2D				
设备	非连续请求次数	请求平均距离	中位数	众数
8,0	44	55882840.9	0	0(5)
Average	44	55882840.9	0	0(5)
Q2Q				
设备	非连续请求次数	请求平均距离	中位数	众数
8,0	100	27021041.8	0	0(61)
Average	100	27021041.8	0	0(61)

图 3-62 系统级 I/O 分析 - 设备寻道信息

- 4) I/O 流量信息：统计类似于 I/Ostat 的数据以及 Mbps 和 I/OPS 的时序数据。

I/O 流量信息表											
8,0											
时间(s)	读IOPS	写IOPS	读吞吐量	写吞吐量	每秒合并读	每秒合并写	每秒读扇区数	每秒写扇区数	请求平均大	I/O平均耗时	I/O平均扇区
2.17	0.00 次/s	1.38 次/s	0.00 KiB/s	29.50 KiB/s	0.00 次/s	4.61 次/s	0.00 扇区/s	59.00 扇区	42.67 扇区	0.21 ms	0.94 ms
5.05	0.00 次/s	4.17 次/s	0.00 KiB/s	45.83 KiB/s	0.00 次/s	1.39 次/s	0.00 扇区/s	91.67 扇区	22.00 扇区	0.38 ms	0.66 ms
7.77	0.00 次/s	1.84 次/s	0.00 KiB/s	36.76 KiB/s	0.00 次/s	7.35 次/s	0.00 扇区/s	73.53 扇区	40.00 扇区	0.24 ms	1.21 ms
8.93	0.00 次/s	8.61 次/s	0.00 KiB/s	103.35 KiB/s	0.00 次/s	2.58 次/s	0.00 扇区/s	206.71 扇区	24.00 扇区	0.34 ms	0.63 ms
13.14	0.00 次/s	0.71 次/s	0.00 KiB/s	15.21 KiB/s	0.00 次/s	2.85 次/s	0.00 扇区/s	30.41 扇区	42.67 扇区	0.23 ms	1.18 ms
14.73	0.00 次/s	5.65 次/s	0.00 KiB/s	27.61 KiB/s	0.00 次/s	0.63 次/s	0.00 扇区/s	55.22 扇区	9.78 扇区	0.41 ms	0.55 ms
TOTAL	0.00 次/s	2.99 次/s	0.00 KiB/s	37.74 KiB/s	0.00 次/s	3.80 次/s	0.00 扇区/s	75.47 扇区	25.27 扇区	0.28 ms	0.75 ms

图 3-63 系统级 I/O 分析 - I/O 流量信息

- 5) 请求合并信息。了解块 I/O 层将多少传入请求(Q)合并为更少但更大的 I/O(D)传递给底层驱动程序。

设备合并信息表							
设备	Q请求数	D请求数	合并比例	BLKmin	BLKavg	BLKmax	Total
8,0	100	44	2.3	8	25	136	1112

图 3-64 系统级 I/O 分析 - 请求合并信息

6) 活跃请求信息。若一个时间段内一直有处于 D 到 C 之间处理状态的请求，这个时间段可以作为一个活跃时间段。

活跃周期信息表				
设备	活跃时间段计数	平均活跃周期	平均不活跃周期	活跃时间占比
8,0	18	0.000198064	0.866471321	0.02
整个系统	18	0.000198064	0.866471321	0.02

图 3-65 系统级 I/O 分析 - 活跃请求信息

下图是 I/O 分析的系统级分析的图形展示结果：

1) I/O 请求流程总览。下图展示了 I/O 的生命周期，并标记了每个时延阶段的占比。时延占比最多的标记为红色，时延占比大于 0 的阶段标记为深绿色，时延占比为 0 的阶段标记为浅绿色。

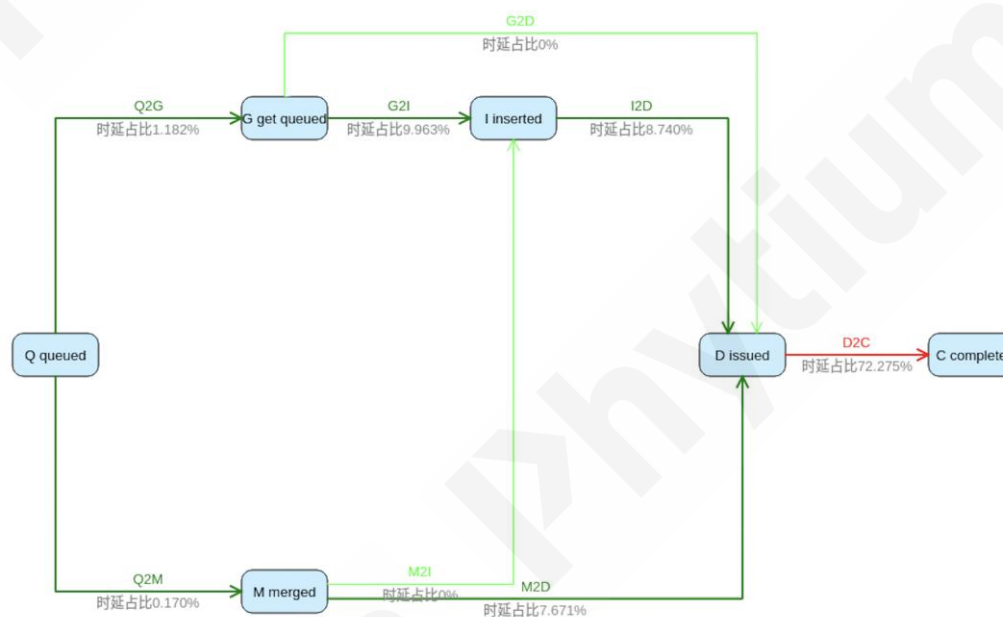


图 3-66 系统级 I/O 分析 - I/O 请求流程总览

2) 各阶段时延的分布和时延变化。

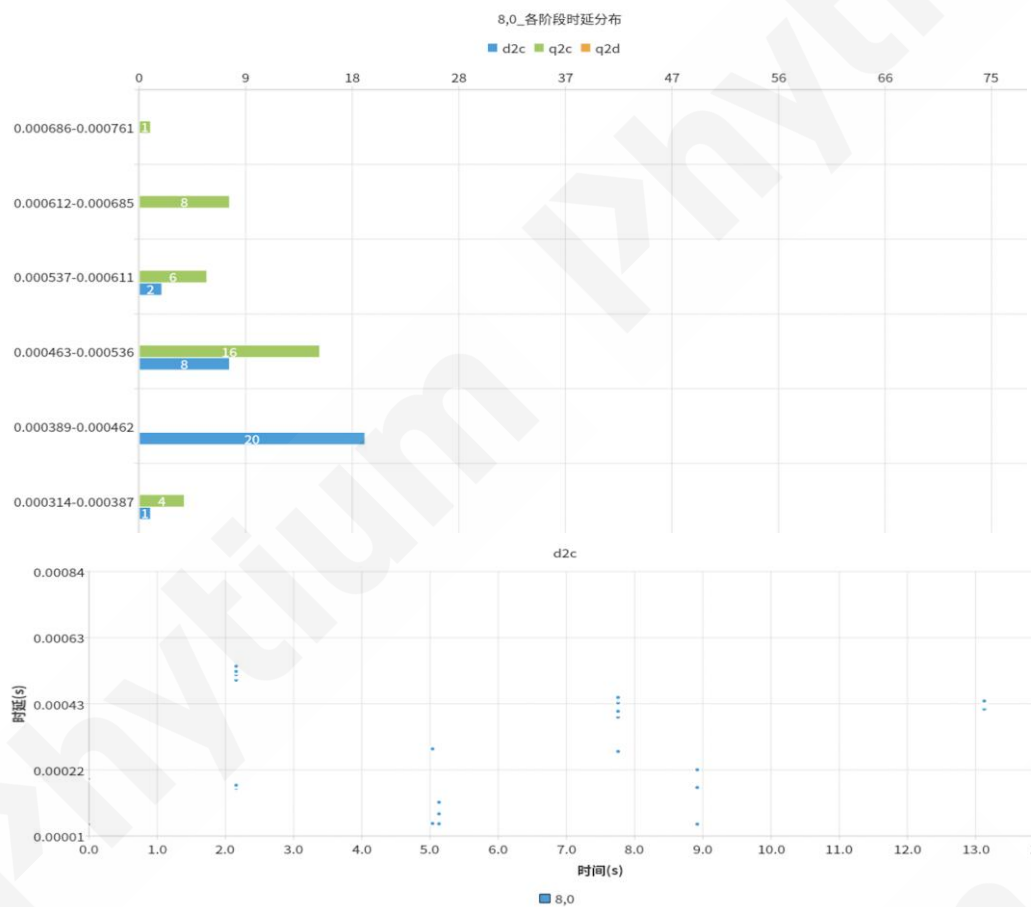


图 3-67 系统级 I/O 分析 - I/O 请求流程总览

3) 寻道距离变化。

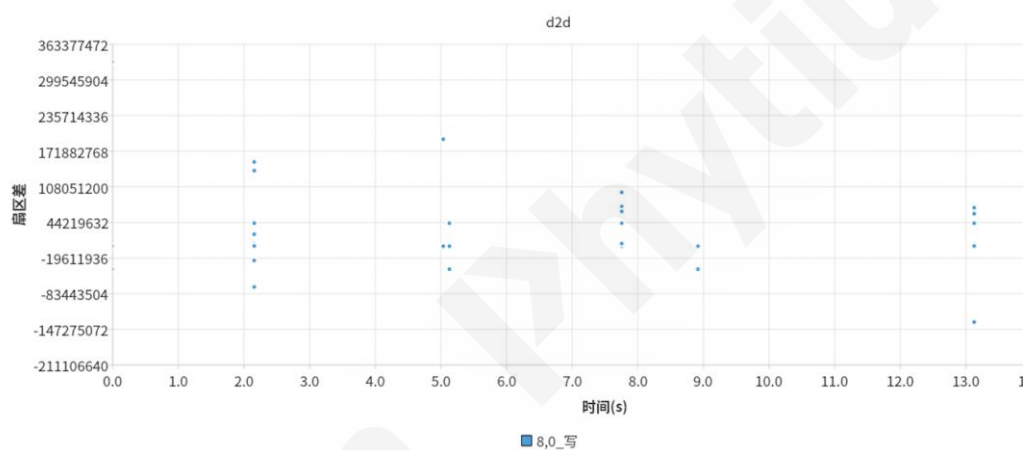


图 3-68 系统级 I/O 分析 - 寻道距离变化

4) I/O 生命周期的队列深度变化。

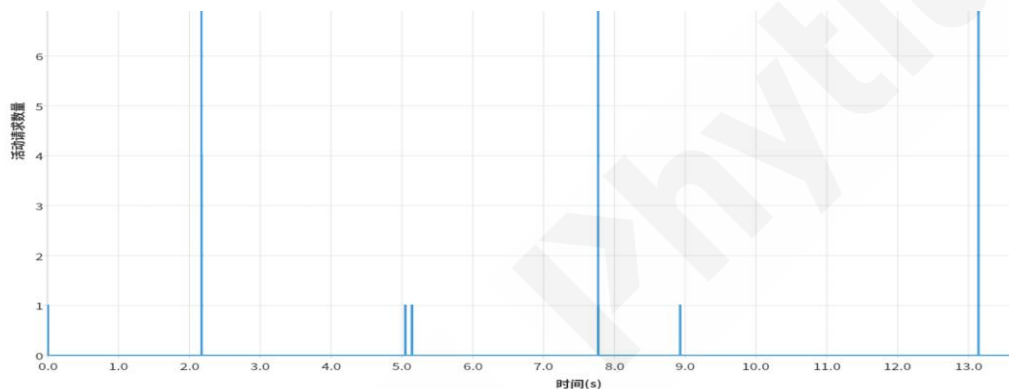


图 3-69 系统级 I/O 分析 - 队列深度变化

5) I/OPS 和 Mbps 的流量信息。



图 3-70 系统级 I/O 分析 - 流量信息

下图是运行期间产生的每个 I/O 请求的跟踪和 I/O 片段的记录，可以通过搜索框搜索关键字查找对应的 I/O 时刻。

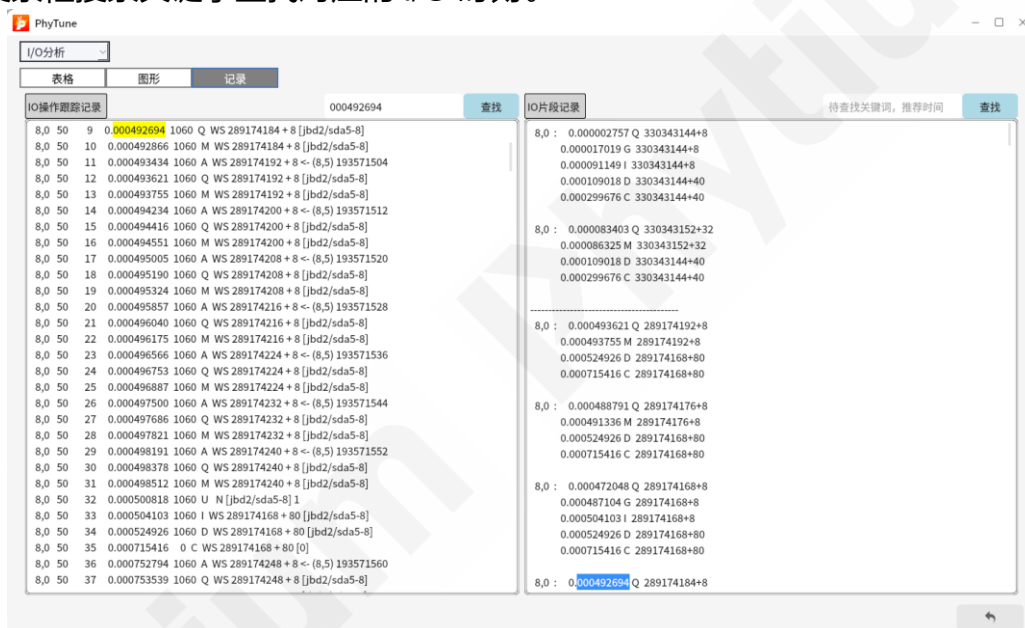


图 3-71 系统级 I/O 分析 - 操作及片段记录

(2) 实时模式

周期性的动态展示请求大小和请求延迟统计信息。

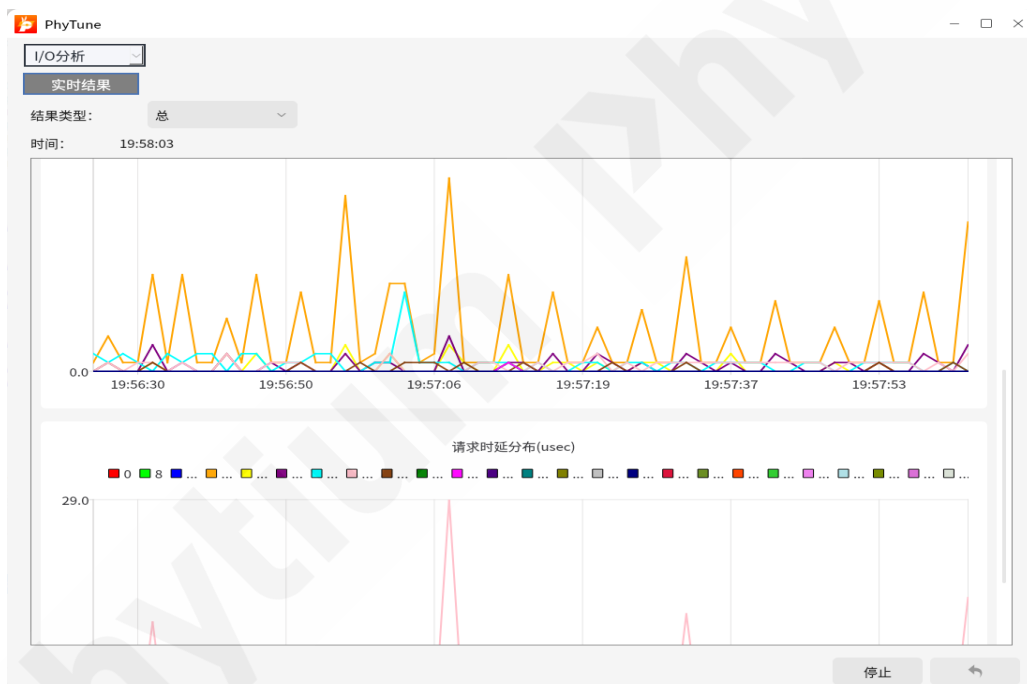


图 3-72 系统级 I/O 分析 - 实时模式

3.6.2 应用级分析

应用级分析是对“程序”或“进程”对象进行访存分析，仅支持记录模式。

3.6.2.1 分析配置

首先，选择待分析程序所在的目标机器。

然后选择分析对象。“程序”对象需要输入目标程序和程序参数。可通过点击“程序文件浏览”按钮获得待测程序的绝对路径。

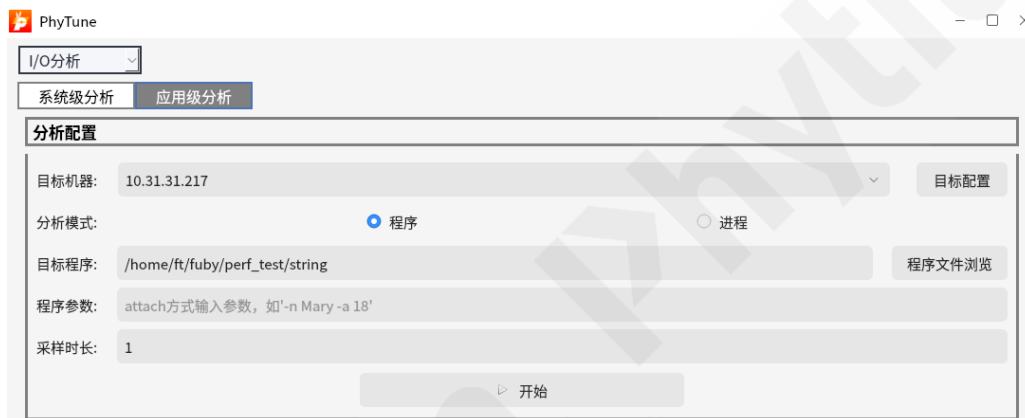


图 3-73 应用级 I/O 分析 – 选择待测程序

“进程”对象需要点击进程下拉框手动勾选待测进程，然后点击“确定”按钮获取进程号自动填写到“目标进程”输入框中。

3.6.2.2 分析执行

记录模式完成上述配置操作后，点击“开始”按钮，即可启动目标机器上的系统级分析的记录模式，并实时显示记录模式的分析运行状态。

说明：在启动分析后，也可以在点击“停止”按钮中止正在运行中的分析任务。该情况下将没有分析结果输出。

3.6.2.3 结果查看

下表统计了程序或者进程的 I/O 分析过程中发生的系统调用，包括函数/进程 ID、平均执行时间、总执行时间、正确调用次数、错误调用次数及执行时间占比。更进一步，展示了每一个函数的详细系统调用，包含调用函数，函数参数，函数返回值等。

PhyTune

I/O分析

表格 图形

函数/进程ID	平均执行时间	总执行时间	正确调用次数	错误调用次数	执行时间占比
> brk/408051	0.000010	0.000031	3	0	2.138%
> close/408051	0.000008	0.000023	3	0	1.586%
子进程ID	发生时间	参数列表	返回值	执行时间	调用类型
408051	20:13:10.127035		0	0.000009	有效调用
408051	20:13:10.127248		0	0.000007	有效调用
408051	20:13:10.127558		0	0.000007	有效调用
> execve/408051	0.000955	0.000955	1	0	65.862%
> faccessat/408051	0.000042	0.000042	1	0	2.897%
> fstat/408051	0.000010	0.000042	4	0	2.897%
> getrandom/408051	0.000009	0.000009	1	0	0.621%
> mmap/408051	0.000015	0.000092	6	0	6.345%
> mprotect/408051	0.000017	0.000067	4	0	4.621%
> munmap/408051	0.000036	0.000073	2	0	5.034%
> openat/408051	0.000019	0.000057	3	0	3.931%
子进程ID	发生时间	参数列表	返回值	执行时间	调用类型
408051	20:13:10.126673	AT_FDCWD, "/etc/ld.so.pre...	3	0.000024	有效调用
408051	20:13:10.127147	AT_FDCWD, "/etc/ld.so.cac...	3	0.000017	有效调用
408051	20:13:10.127282	AT_FDCWD, "/usr/lib64/lib...	3	0.000016	有效调用
> prlimit64/408051	0.000009	0.000009	1	0	0.621%
> read/408051	0.000014	0.000014	1	0	0.966%
> set_robust_list/40...	0.000007	0.000007	1	0	0.483%
> set_tid_address/40...	0.000007	0.000007	1	0	0.483%
> write/408051	0.000022	0.000022	1	0	1.517%

图 3-74 应用级 I/O 分析 - 调用栈

使用饼图展示不同函数的事件占比。

mmap/408051
close/408051
getrandom/408051
execve/408051
read/408051
mprotect/408051
set_tid_address/408051
set_robust_list/408051
write/408051
prlimit64/408051
faccessat/408051
fstat/408051
munmap/408051
brk/408051
openat/408051

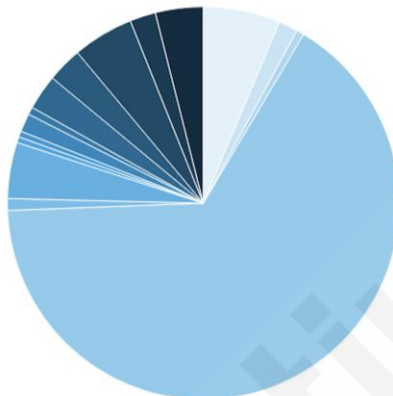


图 3-75 应用级 I/O 分析 - 函数占比饼图

下图展示了多个函数的时间轴图形，在时间轴上绘制了有效调用和错误调用的时间片，每个时间片的左下角坐标的 x 值为调用发生的时间，时间片的长为执行时间。



图 3-76 应用级 I/O 分析 - 函数调用时间轴

3.7 远程采样

远程采样分析的主要功能是支持对目标机器的软件运行执行性能采样并进行在操作端的本地分析，支持热点分析、调用栈分析、火焰图分析等。

步骤 1: ① 选中远程采样单选框，此时远程机器列表可用。② 点击远程文件浏览功能，随即弹出远程文件浏览窗口。③ ④ 在远程文件浏览窗口中选择待测可自行文件。⑤ 点击选中。⑥ 选中文件路径会填充到对应控件中。

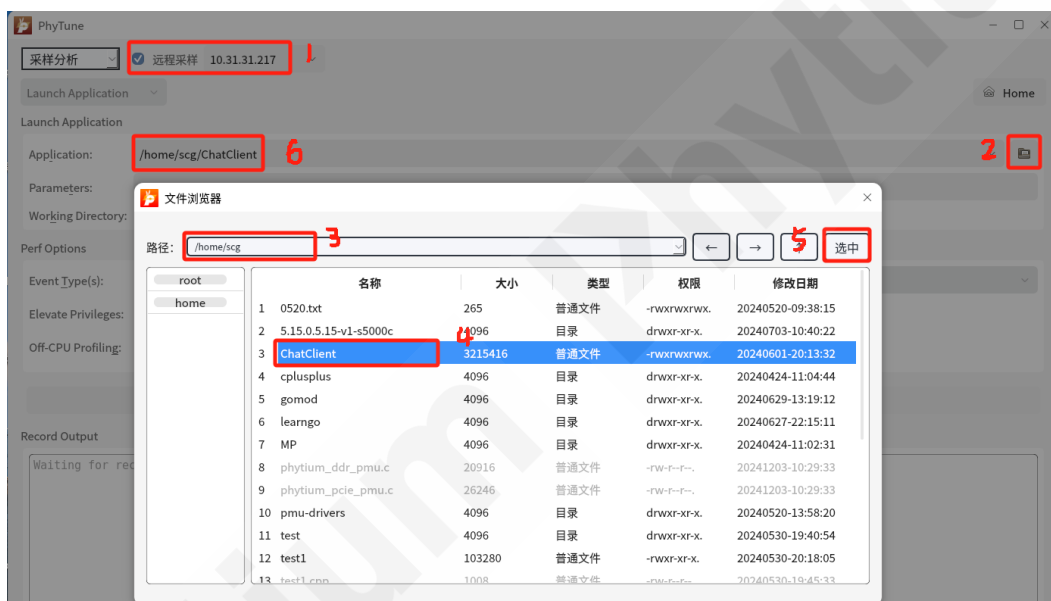


图 3-77 远程采样步骤 I

步骤 II：①点击开始。②在远程采样结束后结果查看按钮可用。

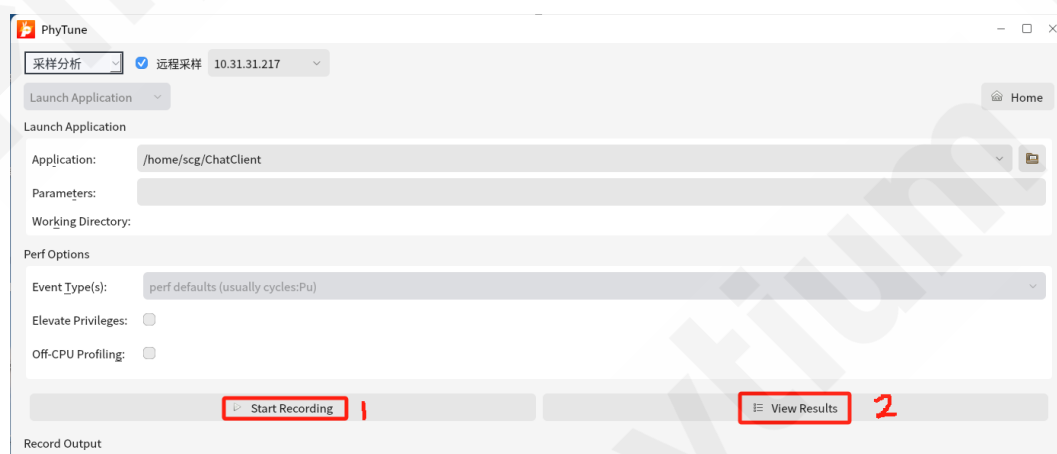


图 3-78 远程采样步骤 II

远程采样结果展示：

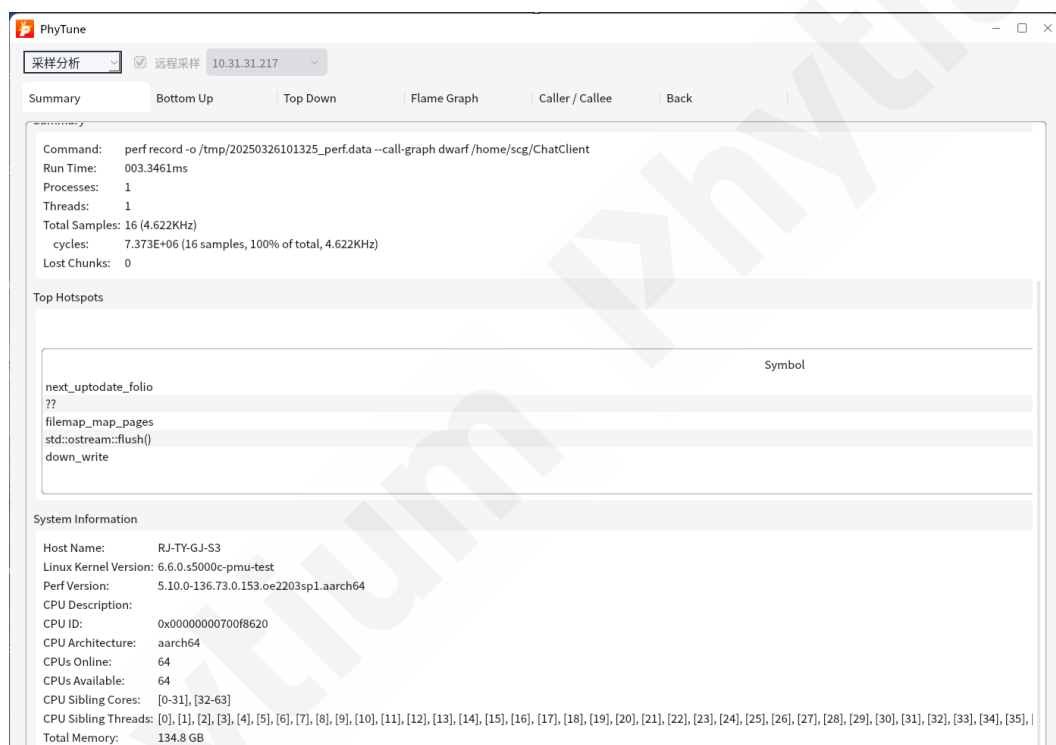


图 3-79 远程采样结果展示

4 常见问题

4.1 参数错误

4.1.1 问题现象



图 4-1 参数错误

通过“分析选项”以挂载进程号方式指定待分析程序时，报参数错误。

4.1.2 处理方法

原因是未限定时长，可通过在分析选项中增加类似“--timeout 1000”参数来解决。

4.2 无权访问内核关键数据

4.2.1 问题现象



图 4-2 无权访问内核数据

进入功能项时弹出对话框显示无权访问内核关键数据。

4.2.2 处理方法

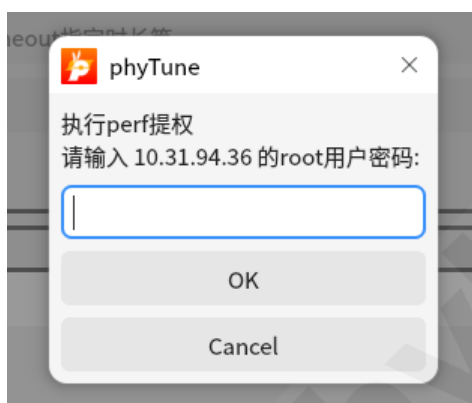


图 4-3 执行提权

执行提权操作，输入 root 用户密码后点击“OK”按钮。