

# ffmpeg 性能调优报告

任务目的  
对 ffmpeg 进行性能调优。  
测试环境

- 软件：ffmpeg
- 参数：encoder;threads;b:v;bufsize;c:v;preset
- 测试工具：time,top,ssim,psnr（Linux 命令行工具）,Phoronix Test Suite,FFmpeg Benchmark
- 操作系统：openEuler 20.03-LTS-SP2

以下为过程及调优结果,详细日志见文件夹 ffmpeg/log

1. phoroni-test-suite 测试调优

先确定编码器(ffmpeg 中参数为 c:v)

方案 Video On Demand,变量 encoder:

Encoder	Seconds (多次测量取平均值)	FPS (多次测量取平均值)
Libx265	359.83	21.05
Libx264	205.07	31.42

应选用 libx264 作为编码器

选用 Encoder:libx264 后,不同方案的时间和帧数值如下:

Description	Scale	Proportion	DisplayFormat	Identifier	Value
Encoder: libx264 - Scenario: Live	Seconds	LIB	BAR_GRAPH	ffmpeg_x264	31.13
Encoder: libx264 - Scenario: Live	FPS	HIB	BAR_GRAPH	ffmpeg_x264	162.23
Encoder: libx264 - Scenario: Upload	Seconds	LIB	BAR_GRAPH	ffmpeg_x264	316.3937578
Encoder: libx264 - Scenario: Upload	FPS	HIB	BAR_GRAPH	ffmpeg_x264	8.12
Encoder: libx264 - Scenario: Platform	Seconds	LIB	BAR_GRAPH	ffmpeg_x264	211.3
Encoder: libx264 - Scenario: Platform	FPS	HIB	BAR_GRAPH	ffmpeg_x264	35.85

Encoder: libx264 - Scenario: Video On Demand	Seconds	LIB	BAR_GRAPH	ffmpeg_x264	245.17
Encoder: libx264 - Scenario: Video On Demand	FPS	HIB	BAR_GRAPH	ffmpeg_x264	31.49

取 c:v=libx264,与 libx265 相比,用时减少了 43%,帧数提高了 49%,性能约提高了 46%

2.进一步手动调优

参数表:

threads	b:v	bufsize	c:v	preset
1	200k	2000k	libx264	ultrafast
2	400k	4000k	libx265	superfast
3	600k	6000k		veryfast
4	800k	8000k		faster
5	1000k	10000k		fast
6				medium
7				slow
8				slower

得到如下结果:

判断最优线程数

threads	时钟时间 (程序从开始到结束时间)	SSIM	PSNR	%CPU	%MEM
1	5m27.574s	0.950891	26.505340	106.7	1.3
2	3m8.219s	0.950892	26.505257	200.0	1.2
3	2m20.982s	0.950892	26.505270	293.3	1.3
4	2m6.220s	0.950892	26.505341	340.0	1.3
5	2m2.610s	0.950892	26.505341	306.7	1.3
6	1m56.563s	0.950891	26.505340	325.0	1.4
7	1m57.164s	0.950891	26.505340	246.3	1.4
8	1m58.682s	0.950891	26.505340	348.0	1.5

取 threads=6

在此基础上判断 b:v

b:v	时钟时间	SSIM	PSNR	%CPU	%MEM
200k	1m39.020s	0.949486	26.480318	380.0	1.5
400k	1m57.356s	0.952673	26.527522	340.0	1.4
600k	2m13.374s	0.953917	26.538236	320.0	1.4

800k	2m10.469s	0.954626	26.543122	366.7	1.4
1000k	2m20.848s	0.955077	26.546117	353.3	1.4

取 b:v=400k  
在此基础上判断 bufsize:

bufsize	时钟时间	SSIM	PSNR	%CPU	%MEM
2000k	1m59.177s	0.952786	26.529040	346.7	1.5
4000k	2m3.697s	0.952809	26.529336	339.2	1.4
6000k	2m2.913s	0.952788	26.529167	348.0	1.4
8000k	1m58.659s	0.952782	26.529091	328.2	1.4
10000k	1m59.886s	0.952776	26.529040	340.0	1.4

取 bufsize=8000k,  
在此基础上更改 preset  
直接选择 utlrafast  
得到

preset	时钟时间	SSIM	PSNR	%CPU	%MEM
utlrafast	0m49.296s	0.938144	26.371736	186.7	0.7

故经一系列测试后,最终取参数为 c:v=libx264 threads=6 b:v=400k  
bufsize=8000k preset=utlrafast

#### 转码速度

转码速度是指 ffmpeg 在单位时间内转码的视频时长(总时长除以时钟时间),反映了 ffmpeg 的处理能力。本次测试中,ffmpeg 的平均转码速度 50.5923。与基准测试相比,提高了 ffmpeg 在进行视频转码时的处理速度 148%。

#### 转码质量

转码质量是指 ffmpeg 在进行视频转码时对视频画面和音频的保真度,反映了 ffmpeg 的压缩效率。本次测试中,ffmpeg 在指定参数进行视频转码时,根据 [SSIM](#) (结构相似性) 指标,输出视频与输入视频之间的画面相似度为 0.9381,接近于 1,表示几乎没有可见的画面损失。根据 [PSNR](#) (峰值信噪比) 指标,输出视频与输入视频之间的画面差异为 26.37 dB,低于 30 dB,表示画面有一定噪声。与基准测试相比分别降低了 1.3%,0.4%,

#### 资源消耗

资源消耗是指 ffmpeg 在进行视频转码时所占用的系统资源,如 CPU、内存等,反映了 ffmpeg 的资源利用率。本次测试中,使用 top 命令监控 ffmpeg 的资源消耗,ffmpeg 的进程占用了 186.7%的 CPU 和 0.7%的内存,资源占用降低了 46%,56%。

测试数据表:

转码速度	SSIM	PSNR	CPU	内存
50.5923	0.9381	26.37	186.7%	0.7%

可以发现调整 c:v, threads, b:v, bufsize 几个参数可以较大幅度改变转码质量,有效改变转码速度和资源消耗,改变 c:v, preset 参数可以大幅度改变转码速度和资源消耗,

综合以上两步调优可得,性能提升为

$$43%*0.05+46%*0.05+148%*0.3+(-1.3%)*0.15+(-0.2%)*0.15+46%*0.15+56%*0.15=63.925\%$$