

编程语言PythonPyPy即时编译 (JIT)性能优化✎ 修改

## 优化 Python 性能：PyPy、Numba 与 Cython，谁才是目前最优秀的 Python 运算解决方案？✎ 修改

就算是使用了numpy等包，再做了大量的代码优化工作，python的运算速度仍然没令人满意。这样看来，python要摆脱“开发快、运行慢”这种身份还...显示全部

关注问题✎ 写回答+ 邀请回答1 条评论🚩 分享★ 邀请回答🚩 举报...

查看全部 18 个回答

 Felis sapiens  函数式编程、编程语言、编程 话题的优秀回答者

nameoverflow、hhhhhhhhh 等 81 人赞同了该回答

爪机占坑。看题主关心的应该是科学计算，那么问题的关键在并行化。如果能自动把基于numpy的科学计算代码编译然后并行化，那最好不过。用这个眼光来看的话，有几个解决方案就没啥意思了。

Numba可以pass掉。它还有一个商业版NumbaPro，能够自动做多核/GPU优化。反过来说，社区版的Numba短期内也只能编译到LLVM，不能自动并行化。看题主关心的应该是科学计算，不能并行化的解决方案没什么意思。

Cython可以pass掉，手动写并行，不如直接换C++，用那边的并行库。

pypy可以pass掉，这货拿来做web后端倒挺适合，拿来做科学计算就免了，numpy支持一堆坑。尤其是再做各种第三方numpy优化就更不行了，毕竟它的c api跟CPython都不一样。


说说容易做并行化的。

开源解决方案里，支持自动并行化的，有numexpr/theano，能把基于numpy的表达式编译成能在多核/gpu上跑的形式，之前大作业有用到，效果不错，推荐题主试试。

对于复杂一些的情形，不是几个numpy表达式能搞掂的计算，你可以手动拆分计算，写一个worker函数，然后用multiprocessing库或者iPython Parallel来把worker函数分派到本地/远程机器上的多个核心搞计算。multiprocessing库属于标准库，在py3k里还有一个标准库concurrent.futures，相当于一个wrapper，用里面的ProcessPoolExecutor十分方便。对了，multiprocessing库不怎么鲁棒，有一系列坑，包括不能在repl下用/不能处理类的成员函数等。按照文档里最简单的写法来就没问题了。

发布于 2014-08-04

▲ 赞同 81 ▼14 条评论🚩 分享★ 收藏❤️ 感谢...

 find goo MOV AX, BX

17 人赞同了该回答  
推荐pypy,pypy内置jit。

更多回答

关注者1,061被浏览72,914

他们也关注了该问题



关于作者



Felis sapiens

✎ 函数式编程、编程语言、编程 话题的优秀回答者

👤 电影旅行敲代码、Antokha Yuuki、暮无井见铃也关注了她

回答624文章40关注者14,871

已关注

发私信

被收藏 62 次

有用云中客 创建 11 人关注

quant陈立 创建 10 人关注

Python酿酿酿酿酿泉 创建 8 人关注

学tom pareto 创建 6 人关注

自己编制的不调用外部包的pure python遗传算法代码，我用python ga.py执行时间为47.6秒，用pypy ga.py执行时间为5.9秒。其实只要代码写得好（尽量按pypy官方写），性能直逼c语言。这个代码用c语言写的运行时间是12左右，我估计pypy的jit在代码编译优化时做了循环展开，而c语言是直接编译成二进制优化较少。我把这个代码转换写成c#的，在windows运行时间是20多秒，java没测试，估计和c#差不多。python写代码可读性强，比较简短，虽然没有java/c#/c++工程味强，但比写c语言要省指针内存的心，适合业务语言系统，如我用于量化交易程序，数据处理程序，算法原型实现程序，如果不加速的确比较慢，而免费的加速不用白不用，否则是地球二氧化碳排放的罪人。很多别的语言苦苦追求的性能优化，只在你的设计一瞬间，你只要python改成pypy，就可以加速几倍，体验一下换个引擎的快感。

...的地点是时很多通用...[展开阅读全文](#) ...

▲ 赞同 17 ▼


● 3 条评论

➦ 分享

★ 收藏

♥ 感谢

...



dontbeatmycat

heleifz.github.io/

我用 Cython，挺好用的。

[展开阅读全文](#) ▼

▲ 赞同 ▼

● 添加评论

➦ 分享

★ 收藏

♥ 感谢

...

[查看全部 18 个回答](#)

IT-DEVELOP  
张先生 创建

5 人关

相关问题

[CPython是什么？PyPy是什么？Python和这两个东西有什么关系呢？Python的底层使用什么语言实现？学习Python需要学习底层实现吗？](#) 6 个回答


[php7 更新后对于 python 会造成怎样的冲击？python+c 扩展能否 hold 住性能？](#) 20 个回答

[同为动态语言，Python 的性能为何只有 PHP 的五分之一？](#) 27 个回答

[为什么苹果新语言 Swift 的 RC4 运算效能是 Python 的 220 倍？](#) 17 个回答

[为什么这几种fib函数的性能差异如此之大？](#) 3 个回答


相关推荐



淼懂物理学：理解世界的极简指南

共 31 节课

▶ 试听



汽车研发：整车性能开发入门

★★★★★ 245 人参与



并行编程方法与优化实践

刘文志

0 人读过

📖 阅读

知乎

超级会员

戳此进入>>>

3000+ 场好课随意听

1000+ 本好书任意读

广告

刘看山 · 知乎指南 · 知乎协议 · 隐私政策  
应用 · 工作 · 申请开通知乎机构号  
侵权举报 · 网上有害信息举报专区  
违法和不良信息举报：010-82716601  
儿童色情信息举报专区  
电信与服务业务经营许可证  
网络文化经营许可证  
联系我们 © 2018 知乎

