

本次测试使用两个测试账号，最后一次测试用时约 20 小时。使用两个 IP（本地 IP 及 GoAgent 代理）测试新浪微博高级搜索接口能接受的频率。

测试过程的伪代码如下：

模拟登陆获得所有候选用户的 Cookie

Interval = 初始时间间隔

开启计时

While 1:

    若被发现次数 > 最大容许次数:

        Interval += 某个值

        被发现次数 = 0

        重新计时

    若计时大于某个值:

        Interval -= 某个值

        重新计时

    从 Cookie 库中随机抽取一个加入报头

    随机选择一种 IP 方式（代理 or 无代理）

    For i 从一到最大尝试次数:

        尝试访问页面

        若成功

            Break //结束尝试

        若抛出错误

            若 i = 最大尝试次数

                被发现次数++

        否则

            等待随机时间后继续尝试 //防止客观原因影响判断

    停留时间（P(Interval)）

由于最近校园网并不稳定，因此会出现实际能访问的页面可能无法访问的状况，我引入了多次尝试的方法，也就是一次访问失败不意味机器人被服务器发现。在测试中，一轮访问尝试最多三次，若三次均访问失败则视为非客观原因而结束当轮访问，并记录被发现次数。

当被发现次数大于三次时，当前访问的时频将进行调整，等待时间增加。当有一段较长时间（在这里是两小时）没有出现被发现的情况，等待时间将会缩短以逼近临界值。

这里的每次等待时间是一个在一定范围内的一致分布。

最后一次实验开始时设定等待时间均值为 15 秒（区间大小为 10，也即等待时间范围在 10-20 秒）。每一次对等待均值的调整大小为 5 秒。试验结束时的时间间隔均值经过程序的两次自发调整，稳定在 25 秒。

```
Sat, 28 Mar 2015 15:40:56 crawler.py[line:121] INFO An account may be spotted
Sat, 28 Mar 2015 15:41:16 crawler.py[line:96] DEBUG failed more than three times, stay a
longer interval
Sat, 28 Mar 2015 15:41:16 crawler.py[line:97] DEBUG Average Wait Time now: 25
Sat, 28 Mar 2015 15:41:17 crawler.py[line:112] DEBUG get a legal search page
Sat, 28 Mar 2015 15:41:46 crawler.py[line:112] DEBUG get a legal search page
Sat, 28 Mar 2015 15:42:20 crawler.py[line:112] DEBUG get a legal search page
```

因此根据初步测试，在两个用户以及两个 IP 随机交替的情况下，可接受的搜索平均间隔时间大约为 25 秒，服从一定区间的均匀分布。更详细的结果有待调整参数优化。