Vol. 14 No. 4 Aug. 2 0 1 2

JOURNAL OF LIAONING PROVINCIAL COLLEGE OF COMMUNICATIONS

文章编号: 1008-3812(2012)04-023-03

# 一种低成本的 C 语言协程实现

#### 李中跃

(辽宁省交通高等专科学校,辽宁 沈阳 110122)

摘要本文介绍了C语言中一种新的协程编程模型,即利用Timer驱动有限状态机方式,有效地解决了传统C语言协程实现的兼容性和跨平台等问题,提供了C语言下并行编程除线程之外的另一种选择。

关键词 C语言; 协程; Timer 驱动; 有限状态机

中图分类号: TP311

文献标识码: A

#### 1 引言

在移动互联网时代,人们越来越多地追求软件的用户体验,iPhone与iPad的火爆也充分说明了这一点。传统的阻塞模式的编程方式虽然简单且符合人们的思维方式,但是也越来越不适应软件响应速度的要求。多线程看起来似乎是解决问题的办法,但是多线程程序难编写,难维护,难调试是众所周知的。在笔者看来多线程绝对不是未来并发时代的发展趋势,这也是为什么众多计算机学者要寻找其他的编程模型的原因。

#### 2 什么是协程

协程(coroutine)是一种不基于线程的并行编程模型,它并不是一个新鲜事物,而是早在 u-nix 时代就出现了,近来又重新引起人们的重视。和线程相比,协程有以下优点:

- (1) 由用户控制"协程"的调度,不需要任何的线程同步操作,不存在多线程死锁、同步, 资源竞争等问题。
- (2) 和线程相比,调度的开销非常小,可以同时创建大量的协程。
- (3) 编程模型简单,容易写出结构清晰的程序。

协程的缺点就是不能充分利用多核 CPU,但 是可以借助多进程来实现对多 CPU 核心的利用。

#### 3 现有语言对协程的支持

(1) Lua, 目前最火爆的脚本语言, 提供对 coroutine 语言级的支持, 值得一提的是 Lua 的作

者拒绝在 Lua 上加入对多线程的支持。

- (2) Java, 支持的不好, 需要借助 JNI 和操作系统的支持。
  - (3) c#, 专门增加关键字 yield 来支持协程。
  - (4) python, 提供语言级别的支持。
  - (5) C++, boost 库等第三方库。
- (6) C, 本文讨论的重点, 通过第三方库或 者本文介绍的新方法。

#### 4 目前 C 语言中的 coroutine 的实现。

C语言本身并不支持协程,而且由于标准 C中定义全局只能有一个栈,所以要完全通过标准 C来实现协程并不太容易,一般是通过第三方库或者系统调用来实现。

- (1) windows 系统下的协程被叫做 fiber, 可以通过专门的系统函数创建。fiber 只能在 windows 系统中, 不能跨平台。
- (2) posix 系统下可以通过 getcontext 和 setcontex 等系列函数来实现协程。
- (3) 通过 setjmp 和 longjmp 函数,通过跳转来实现协程。setjmp 和 longjmp 对 c++和有些编译器的支持不是很好,比如在 VC 系列就完全不能使用利用 setjmp 和 longjmp 实现的协程。
- (4) 通过宏定义来模拟。使用宏的方式带有太多的技巧性和副作用,写出的代码也很难看懂。
- (5) 通过 libtask 等第三方库来实现。libtask 也存在不少兼容性问题。

收稿日期: 2012-06-06

虽然有很多方式来实现 C 语言中的协程,但是这些方法都有自己的局限性和适用范围,尤其是用在复杂的 UI 编程环境中的时候问题就更多。

#### 5 本文的方法

鉴于目前 C 语言协程实现方式的兼容性等问题,笔者通过转变思路实现了一种新的 coroutine 编程模式。这种模式完全利用标准 c 来实现,实现简单,不借助任何第三方库和系统调用,可以完美兼容任何系统。其基本思想就是利用 Timer来驱动一系列的 Task 来运行,而每一个 Task 都是一个有限状态机。需要说明的是这个 Timer 不是仅指由硬件或操作系统提供的 Timer,还可以是一段循环代码,如图 1 所示。

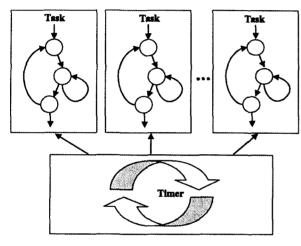


图 1 Timer 的实现

#### 6 Task 的实现

由于协程不是真正的线程,因此不能像线程一样可以在内部阻塞,协程要求内部运行的每一部分代码都不能长时间阻塞,否则就会造成整个系统的阻塞。比如不能使用阻塞方式访问网络。现在的操作系统对几乎每一种耗时的 I/O 操作都提供了非阻塞的访问方式,因此传统的阻塞模式的程序只要使用非阻塞的 API 重新改写就能方便地移植到协程代码中。

```
Task 类的关键代码
class CTask {
    public
        int32_t id;
        ...
    public
        virtual status_t OnTimer (int32_t interval) =0;
```

}; Ta

Task 是通过 ID 来管理的,因为 Task 支持自己退出,因此使用 ID 管理相比较于指针更安全。关键的函数只有一个 OnTimer,是一个纯虚函数,也就是说这个函数必须被其子类实现。OnTimer 函数就是状态机所在的位置,每次 Task 被调度的时候这个函数就会被调用。

#### 7 状态机的实现

有限状态机(Finite State Machine) 简称FSM,是软件领域的重要工具,在人工智能、编译原理等领域都能看到它的身影。一般来说 C 语言实现 FSM 的方式有两种,一种是利用 switch、if 语句,一种是使用状态跳转表。一般都推荐使用状态跳转表来实现 FSM,这样的程序更加清晰易懂。但是本文推荐的方法是使用 if—else 语句来实现 FSM, 因为 Task 的状态一般比较少,而且都写在一个函数中看起来更加方便。

本文中的 FSM 就是 OnTimer 函数中的代码, 形式如下:

FSM 实现的关键在于对传统任务的拆解,比如一般的程序结构都可以和状态机——对应,如图 2 所示:

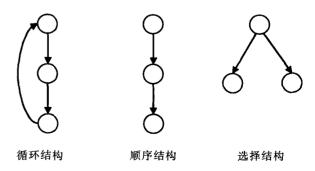


图 2 程序结构与状态机对应关系

图 2 中,每一个圆圈代表一种状态,箭头表示状态跳转。只要稍加训练,由传统的面向过程

的编程方式转向状态机的编程方式并不是很困 难。

#### 8 API 的实现

有一个全局的 TaskManager 变量来管理所有的 Task, 它提供了一套 API 来操作 Task 对象, 常用的 API 有:

ADD\_TASK (task);向全局列表中添加一个Task 对象。

QUIT\_TASK (id);根据ID退出一个Task。 SUSPEND\_TASK (id);根据ID来把一个Task置为休眠状态。

RESUME\_TASK (id); 根据 ID 来唤醒 Task 的运行。

GET\_TASK (id); 根据 ID 来获取 Task 对象的指针。

IS\_TASK (id); 判断一个 task 的 id 是否有效。

#### 9 Task 的通知机制

Task 运行完成之后要通知调用者,一种简单有效的办法就是使用回调函数,本例中回调函数是纯 C 类型的,其声明是 status\_t () (void p),其中的 p 是指向一个含有 32 个 void 类型的指针数组,这样可以给回调函数传入任意类型的值。在一个类中嵌套一个 Task 的时候往往希望把类的成员函数作为回调函数,但是普通的成员函数是不能作为回调函数的,必须是静态成员函数才可以,使用静态成员的关键就是把 this 指针正确地传入函数中,可以通过以下的方式:

// 调用回调函数的代码

this->param [0] =this; //通过第一个参数 传入 this 指针

```
this->param [1] = ....
this-callback (this-param);
// 在回调函数中
status_t XXX::on_callback (void p)
{
```

XXX self= (XXX) p [0]; //XXX 代表任意的类, 通过 p [0] 取得 this 指针

// self 指针就和 this 指针一样了,可以像 this 一样操作成员变量和函数了

#### 10 Task 的嵌套使用

Task 经常嵌套使用来完成复杂的功能, Task China Cloud Publishing Alliance. All rights reserved. 是基于 ID 管理的,因此如果在一个 Task 中嵌套 另一个 Task, 只要加上一个整型的变量表示被加 人 Task 的 ID 就可以了。

Task 是可以自动退出的,但是一旦一个Task 引用了另外一个对象问题就复杂起来。因为必须保证被引用对象的生命周期大于 Task 的生命周期。对于子 Task 也是一样,解决办法很简单,就是在 Task 的析构函数中对每一个子调用QUIT\_TASK 来结束子 Task 的运行。

还有一个重要的问题就是等待子 Task 的完成,有两种方法,一种就是把父 Task 的状态机置为一种等待状态,然后一直轮询子 Task 是否完成,这种方式固然可以工作,但是却浪费大量的 CPU 来轮询,因此不是一个好的选择。另一种就是通过 callback 的通知机制,启动子 Task 后把父 Task 置为 suspend 状态,然后在子 Task 的回调函数中把父 Task 重新唤醒,代码看起来会是如下的样子:

```
if (status=STATUS_1) {
    ...
```

启动子 Task

status=STATUS\_N;将在子 Task 结束后 跳转到 STATUS\_N 状态

```
Suspend ();
}
if (status=STATUS_N)
{
```

这里的内容只有在子 Task 结束后才能运行。

在子 Task 的 callback 中要调用父 Task 的 Resume 函数。

#### 11 结束语

本文介绍协程实现简单高效,容易实现,现已被笔者成功地用在 windows IOS android NDK 等多种环境中,有效地验证了本方法的实用性,读者可尝试自己实现下。

#### 参考文献

- [1] 李慧霸, 彭宇行, 卢锡城. 一种面向分布式系统的程序设计模式 [J], 计算机工程与科学, 2008 (12).
- [2] (英) 马修, (英) 斯通斯编著, 陈健, 宋健建翻译. Linux 程序设计 [M]. 北京: 人民邮电出版社, 2010.

Vol. 14 No. 4 Aug. 2 0 1 2

文章编号: 1008-3812(2012)04-026-03

# 新拌高强混凝土砂的作用机理与影响研究

冯乃辉

(鞍山市市政工程有限责任公司, 辽宁 鞍山 114001)

摘 要 高强混凝土的强度对砂的要求比较严格苛刻,通过实验室试验,研究砂率对新拌高强混凝土工作性的影响以及对水泥用量的影响。运用控制变量法,通过改变砂率,研究砂率与坍落度、碎石粒径、水灰比以及水泥用量等因素的关系。研究结果表明:存在最佳砂率,使坍落度、碎石粒径、水泥用量等因素达到最佳状态。

关键词 砂率;工作性;坍落度;水灰比

中图分类号: TU528

文献标识码:B

新拌高强混凝土是指在施工过程中尚未凝结 硬化的高强混凝土,是混凝土生产过程中的一种 过渡状态。混凝土的工作性(和易性)是一项综 合技术指标,包括流动性、捣实性、黏聚性、保 水性等方面,是混凝土拌合物易于施工操作、搅 拌、运输、捣实和表面处理,并获得质量均匀, 成型密实的性能。这些性能在很大程度上制约着 硬化混凝土的技术性能。因此研究新拌混凝土的 工作性具有十分重要的意义。本文通过试验重点 细致研究砂率对新拌高强混凝土工作性的影响。

#### 1 试验

#### 1.1 试验材料

所用水泥为冀东 42.5、52.5 级普通硅酸盐水泥; 鞍钢牌萘系高效减水剂,减水效率为 20%;

所用砂为河砂,细度模数 2.74,属于中砂,筛分结果如表 1 所示;所用碎石规格为 4.75~26.5 连续级配;水为自来水。

表 1 试验用砂筛分数据

筛孔尺寸 (mm)	9.5	4.75	2.36	1.18	0.6	0.3	0.15	0.075	筛底
筛余质量 (g)	0	15	63	99	105	115	75	22	6
分计筛余 (g)	0	3	12.6	19.8	21	23	15	4.4	1.2
累计筛余 (g)	0	3	15.6	35.4	56.4	79.4	94.4	98.8	100
通过率 (%)	100	97	84.4	64.6	43.6	20.6	5.6	1.2	0

#### 1.2 试验内容和试验依据及设备

本试验研究的内容如下:

保持基本水泥用量不变的情况下,逐步改变 砂率,探究其对新拌高强混凝土的工作性、强度

收稿日期: 2012-06-19

作者简介: 冯乃辉 (1972— ), 男, 辽宁海城人, 工程师。研究方向: 道路与桥梁。

### A low-cost C-Language Coroutine Implementation

#### LI Zhong-yue

(Abstract) This article describes a new coroutine programming model with C-Language. That is Timer-driven Finite State Machine, which can effectively solve the compatibility and cross-platform issue of traditional coroutine implementation with C-Language. The new method gives another option of parallel programming in C-Language outside of the thread.

(**Keywords**) C-Language, Coroutine, Timer-driven, Finite State Machine China Cloud Publishing Alliance. All rights reserved.

• 26 •

# 专业提供学术期刊、学位论文下载、外文文献检索下载服务

### ★资源介绍★

## 【中文资源】

中文文献,期刊论文,硕士论文,博士论文,会议论文,电子图书等等.

## 【英文资源】

IEEE、Wiley、SD、EBSCO、ProQuest、LexisNexis、Springer Link、Jstor、EI、OSA、sag、Acs 等上百种全英文资源.

### 【顶级医学】

ovid、pubmed、md、高权 sciencedirect、Emabse 万方医学、中国生物医药数据库、美国医学会等.

### 【经济资源】

中经、中宏、国泰安、搜数、resset 金融、知网统计等等.

# 【名校图书馆】

国内高校图书馆、地方图书馆、国外高校图书馆。授权进入,极致体验.

# 【论文发表】

提供专科、本科、研究生、MPA、EMBA、MBA各个专业毕业论文代写、代修改服务。企事业单位员工职称论文代写、代发表服务。

论文代写咨询电话 18118022153 陆老师 咨询 QQ 29338355