

# 分布式计算系统中的算法集成方法<sup>①</sup>



窦 帅, 李子扬, 朱家佳, 李晓辉, 米 琳, 李传荣

(中国科学院 空天信息创新研究院 中国科学院定量遥感信息技术重点实验室, 北京 100094)

通讯作者: 李传荣, E-mail: crli@aoc.ac.cn

**摘 要:** 算法集成是研制建设综合性数据处理分析系统的一项重要内容. 本文基于临近空间科学实验支持系统研制实践, 针对面向科学研究需求的数据处理分析系统的算法集成问题, 设计了一种大型分布式系统中集成多种类、多版本、多功能算法的方法, 通过构建算法插件, 满足了此类科学实验支持系统并行开发、插件化调用、灵活扩展、更新迭代的要求. 采用该方法集成的系统部署在 ZStack 私有云平台上, 通过对临近空间科学实验中多种大气参数、原位探测参数、生态参数的计算分析实验验证了该方法的有效性.

**关键词:** 分布式计算; 算法集成; 插件; 迭代

引用格式: 窦帅, 李子扬, 朱家佳, 李晓辉, 米琳, 李传荣. 分布式计算系统中的算法集成方法. 计算机系统应用, 2021, 30(3): 110-116. <http://www.c-s-a.org.cn/1003-3254/7804.html>

## Algorithm Integration in Distributed Computing System

DOU Shuai, LI Zi-Yang, ZHU Jia-Jia, LI Xiao-Hui, MI Lin, LI Chuan-Rong

(CAS Key Laboratory of Quantitative Remote Sensing Information Technology, Aerospace Information Innovation Research Institute, Chinese Academy of Sciences, Beijing 100094, China)

**Abstract:** Algorithm integration is an important content in developing a comprehensive data processing and analysis system. As the development of a near-space science experiment system, a method to integrate algorithms of multiple types, versions, and functions in a large distributed system is designed to facilitate the algorithm integration in a data processing and analysis system. Parallel development, plug-in calling, flexible expansion, and update iteration of the scientific experiment system are enabled by constructing algorithm plug-ins. The system improved with this method is applied to the ZStack private cloud platform, and the method is verified by calculation and analysis of atmospheric parameters, in-situ detection parameters, and ecological parameters in near-space science experiments.

**Key words:** distributed computing; algorithm integration; plug-in; iteration

综合性的数据处理系统往往需要实施多种类型的数据处理业务, 数据处理算法集成是研制建设此类系统的一项重要内容, 集成框架设计或集成方法的好坏将在很大程度上影响系统的数据处理效率、系统功能的可扩展性. 同时, 此类综合性数据处理系统提供计算分析服务常常采用分布式的架构实现, 在分布式系统中采用松耦合方式进行算法集成, 有利于提高系统运

行效率、降低系统的运行维护成本.

软件插件技术作为组件复用技术的一种, 能在不修改程序主体的情况下对软件的功能进行加强<sup>[1]</sup>, 增强代码的可重用性, 提高软件的开发生产率<sup>[2,3]</sup>, 可应用于数据处理算法集成. 近年来软件插件技术应用广泛, 多种常用应用软件甚至操作系统均采用了插件技术<sup>[4]</sup>, 但是在分布式计算系统中集成、调用插件程序时, 由于

① 基金项目: 中国科学院战略性先导科技专项 (A 类)(XDA17040303)

Foundation item: Category A Strategic Priority Research Program of Chinese Academy of Sciences (XDA17040303)

收稿时间: 2020-06-27; 修改时间: 2020-07-27; 采用时间: 2020-08-13; csa 在线出版时间: 2021-03-03

不同插件程序的开发者选用的开发语言、部署平台方面存在差异且各插件程序产生的数据格式不统一<sup>[5]</sup>,需要解决插件程序间的远程消息通信、异构数据共享等问题.目前很多应用系统未考虑插件程序间的信息交互,容易产生信息孤岛<sup>[6]</sup>.此外,在分布式计算系统中调度算法插件程序时,需建立动态反馈机制,根据各计算节点的资源使用情况动态调度节点中的插件程序,保证各计算节点资源使用不超载.

为探索临近空间环境和生态系统,国内多家科研院所联合参与了“临近空间科学实验系统”科技专项,旨在建成我国首个临近空间科学实验系统,在多个典型区域分阶段、多次开展覆盖参量全、高度范围完整的临近空间综合探测科学实验和相关科学研究.为了能够对面向多种临近空间科学研究主题(如大气交换、电磁辐射环境探测、生物适应性、气体与同位素探测等)的大型综合性科学探测实验任务提供支撑与保障,需要设计和建设兼具任务规划、数据存储、数据计算、运行管理、数据分析、数据展示等多种功能的科学实验支持系统.快速高效完成各类科学载荷探测数据的处理、分析与展示是该科学实验支持系统的核心功能之一,数据处理算法集成也成为系统设计建设的一个重要工作.与其他综合性数据处理系统相比,临近空间科学实验支持系统的数据处理分析算法集成具有以下特点:1)专业领域跨越大.临近空间科学探测涉及多种科学研究领域,不同专业领域的探测数据在形式、体量和处理方法上存在很大差异.2)探测载荷多类且多变.临近空间科学探测载荷类别多样(成像/非成像、探空/地基……),即使是相同的探测载荷,也可能由于不断的改进优化,使得各次实验数据的记录格式、探测内容或处理算法随之变化.3)处理算法多样且不固定.作为一个科学研究支持系统,数据处理需要满足研究需求:系统需要集成研究过程中由于研究方法和思路的变化产生的不同处理算法;需要满足对比分析的需求,系统需要针对某一探测数据集成多种版本的处理算法(具有不同参数、不同处理过程、不同处理方法的算法).以上数据特点与功能需求,采用分布式计算+算法集成的方法构建处理数据处理系统,将大幅降低系统升级维护、算法迭代的复杂度,且有利于提升数据处理效率.

针对以上系统需求和集成难点,本文借鉴软件插件设计的思想,探讨了一种在大型分布式科学分析系统中集成多种、多版本、多功能算法的集成方法:综

合考虑了算法运行依赖、算法注册规范等方面因素,设计了算法管理器、算法调度器,实现了插件的加载与管控,将算法模块设计为拥有独立目录结构、相互隔离、可以“即插即用”的算法插件.采用该方法集成的系统部署运行在 ZStack 私有云平台上,对临近空间科学实验多种大气参数、原位探测参数、生态参数的计算分析实验验证了该方法的有效性.

## 1 用插件框架实现算法集成

### 1.1 插件框架

插件是应用系统的一部分,但是是相对独立的自治域.插件可被软件主体所引用,与之发生信息交互,但不能独立的脱离软件主体运行<sup>[7]</sup>.与之相对的是软件主体可在无插件的情况下独立运行,不需依赖插件.

基于插件框架的设计,是将部分功能从软件主体中剥离出来,以松耦合的方式和软件主体进行连接.二者遵守共同的约定的接口协议,可以独立的发生变化、升级、发布,互相不发生影响.这种松耦合的方式将整个软件的复杂度分布在软件主体和各插件中,降低了各自的开发难度,在软件研发初期若出现需求不确定的情况,或预期在软件运行期业务逻辑易发生变化,都可以采取这种方式应对.

一个基本的插件框架由3部分组成,应用程序主体、插件接口和插件组成.应用程序主体负责实现软件的主体功能,插件接口负责插件的接口协议和通讯,插件负责实现剥离出来的功能.其基本结构如图1所示.

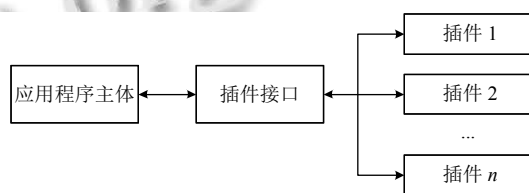


图1 插件框架基本结构

### 1.2 分布式系统中的插件

在大型分布式系统中,由于插件的松耦合特性,非常适合用来实现动态变化的某些核心业务逻辑,使用插件方式,可有效降低整个系统的维护特性.

但在分布式系统中插件与单机软件中的实现方式有所区别.

首先,由于不同的插件可能运行于不同的物理设备上,因此插件的接口更为复杂,往往包含了多机通讯的内容,一个插件有可能与多个物理设备发生交互.

其次, 由于有调度问题, 分布式系统中的插件, 需配合调度方法或工作流, 以实现不同业务逻辑。

再次, 插件的接口有可能不一定直接和软件主体发生联系, 而是以中间件的方式, 通过第三方软件与系统主体进行交互。

最后, 由于大型分布式系统的特性, 插件的数量和规模, 有可能比单机软件大一到两个数量级。

图2显示了一种使用中间件的插件实现方式。

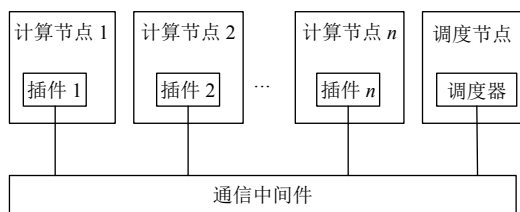


图2 一种利用中间件通讯的插件架构

### 1.3 分布式系统中的 SOA 和插件

面向服务的架构 SOA 是一种有效的分布式系统松耦合集成方式, 它将应用系统不同的功能单元包装成独立的服务, 通过服务间定义的接口协议进行通讯, 协同完成系统功能。相对本文的插件来说, SOA 的服务是一种粗粒度的, 而计算内核是相对内敛的, 非完全功能的模块, 而是针对某特定计算过程的程序, 其更新速度要求开发及更新难度小, 因此本文没有采用 SOA 方式, 而是采用了插件架构进行数据处理算法的集成, 将数据计算视为一种服务, 该服务由算法管理器、算法调度器和多种算法插件组成。因此在本系统中采用的是如图3所示架构。

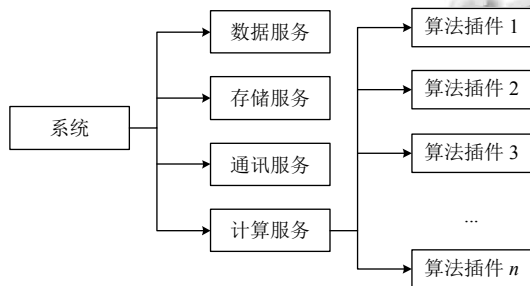


图3 系统架构图

## 2 系统设计

为满足某大型科学实验系统中的数据处理需求, 设计了用于探测数据处理的算法集成插件框架。框架主要包括了系统主体、算法管理器、算法调度器。系

统主体实现实验数据的采集、存储、管理、检索、任务控制等主体功能, 由多种数据处理算法实现具体的科学数据计算功能。算法管理器实现了算法插件注册、算法注册信息管理、算法运行环境配置。算法调度器实现了计算节点的计算资源监视、任务分配、算法调度功能。算法管理器、算法调度器共同实现了多种数据处理算法的集成。在系统实现过程中还专门设计了算法注册规范, 方便科研人员规范算法实现方式, 规范化上注数据处理算法。同时还考虑了算法间的依赖问题, 以算法依赖图的方式辅助算法的调度过程。

### 2.1 算法管理器

算法管理器负责算法插件的注册, 对系统中注册的算法进行管理, 同时负责系统中算法插件运行环境的配置。算法管理器提供数据处理算法插件上传服务, 科研人员根据统一的处理算法集成接口上传各自算法, 由算法管理器实现多计算节点的部署。算法注册信息管理包括了算法信息的录入、查询、浏览、编辑与删除等功能。算法插件运行环境配置实现算法插件的输入路径、输出路径、数据库环境配置, 构建支持算法插件正常运行的环境。算法管理器接口如图4所示。

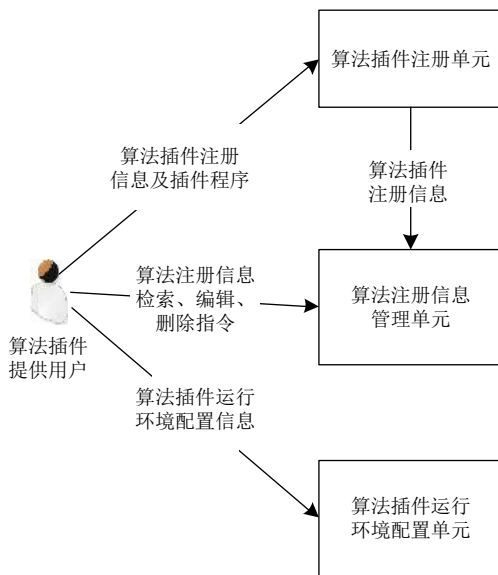


图4 算法管理器接口图

算法管理器通过算法插件注册单元接收用户上传的算法插件注册信息及插件程序, 向算法注册信息管理单元发送算法插件注册信息。算法注册信息管理单元接收算法插件注册单元提供的算法插件注册信息; 接收算法插件提供用户输入的编辑、删除、检索指令,



向用户提供算法注册信息检索、编辑、删除等功能。算法插件运行环境配置单元接收算法插件提供用户提交的算法插件运行环境配置信息,配置运行环境。

算法插件注册与集成过程如图5所示。

- 1) 用户向系统上传插件程序,由系统推荐或手工指定部署至某数据处理节点。
- 2) 系统解析用户提交的算法注册信息并存储,包括算法名称、适用范围、输入输出等。
- 3) 配置算法运行环境,例如依赖包、文件交互路径、数据库信息等。
- 4) 算法入库完成。

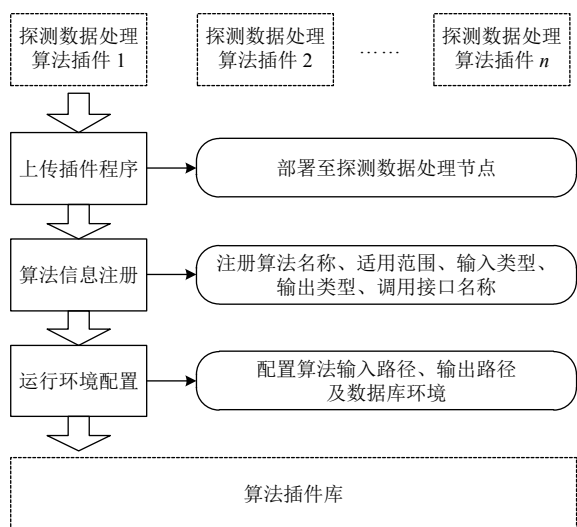


图5 算法插件注册与集成流程图

## 2.2 算法调度器

在科学探测实验中,获取的探测数据种类多,数据处理任务量大,处理过程消耗时间长,并且对系统硬件资源有较高需求。为提高数据处理任务的执行效率,采用了集群式并行数据处理技术,在多台数据计算节点间合理地分配探测数据处理任务。探测数据并行处理过程涉及资源调度节点以及多个数据计算节点。资源调度节点接收到数据处理任务后,采集各数据计算节点的系统资源状态,根据任务分配算法将任务分配至相应的数据计算节点。数据计算节点按照执行流程,逐步对探测数据进行处理,并上报处理进度,直至任务执行完毕,最后将产品数据存储至文件存储系统。

要使并行集群有较好的处理效果,需要考虑数据计算节点自身的处理性能和当前的实时负载情况,因此,引入动态反馈机制以实时获取数据计算节点的当前状态,各个计算节点需要定期的将自身的状态信息

反馈给调度节点,更新资源调度节点上记录的信息。

动态反馈机制模型如图6所示。调度主节点会周期地收集信息, $W_0$ 代表数据计算节点前一个周期的权重值,数据计算节点周期性地向调度主节点反馈其当前的负载信息,调度主节点根据算法 $F$ 和这些信息对当前数据计算节点的权重值进行调整,计算出其当前权重值 $New\_W(i)$ 。当数据处理任务 $r$ 到来后,调度主节点根据算法把任务分配到负载权值比例较小(即相对空闲)的数据计算节点上<sup>[8]</sup>。

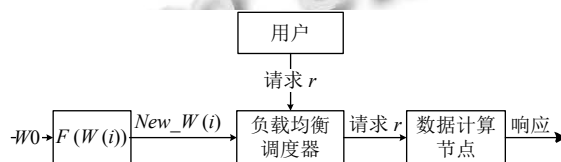


图6 动态反馈机制模型

## 2.3 算法注册规范

采用插件框架,要求算法开发人员要按照约定的规范对算法进行描述,才能在系统中进行算法注册。数据处理算法以程序插件形式独立封装,为确保算法插件可被规范化集成调用,需从内容、格式、方法等层面制定数据处理算法集成规范,明确数据处理算法插件集成过程涉及到的算法配置要求、算法编译要求、算法集成形式、算法接口要求、算法运行要求及算法注册要求。算法配置要求包括算法的依赖库、配置文件的存放目录要求及命名要求;算法编译要求中规定通过相对路径加载依赖库和配置文件的方法;算法封装形式要求中规定算法实体插件必须封装为在Linux操作系统或Windows操作系统下的可执行程序;算法接口要求分为输入接口和输出接口,算法输入接口包括探测数据文件、探测数据名称、探测数据类型、探测数据描述信息等内容,算法输出接口包括探测产品数据、数据处理产品存储路径、数据处理进度上报信号、任务完成报告格式要求;算法运行要求包括算法插件的调用方式、算法插件的调用指令、算法插件执行过程中的文件操作权限设计、算法插件各类结束状态的返回值;算法注册要求包括算法名称、算法描述信息、算法版本、算法部署路径、算法插件运行最长允许时间、算法插件启动的进程数、算法插件占用的内存等注册信息的格式规范。

算法插件注册要求包括算法插件名称、算法插件

描述信息、算法插件版本、算法插件部署路径、算法插件启动的进程数、算法插件占用的内存、算法插件研制单位、算法插件联系人等注册信息的格式规范。

1) 算法插件名称: 算法名称由英文或数字构成, 不包含中文及特殊符号。

2) 算法插件描述信息: 算法描述信息由中英文字符、符号及数字组成, 不超过 500 字。

3) 算法插件版本: 由“V”+版本号构成, 版本号范围为 0-99。

4) 算法插件部署路径: 要求算法插件可在任意路径下正常运行。

5) 算法插件启动进程数: 描述信息由数字组成。

6) 算法插件占用内存: 描述信息由数字+“MB”组成。算法插件注册信息需填入 XML 文件后上传至系统, 文件名为算法插件名称+“.xml”, 文件内容如下所示:

```
<算法插件注册信息>
<注册信息>
<算法插件名称>.....</算法插件名称>
<算法插件用途>.....</算法插件用途>
<算法插件描述信息>.....<算法插件描述信息>
<算法插件版本>.....<算法插件版本>
<算法插件启动进程数>.....</算法插件启动进程数>
<算法插件占用内存>.....</算法插件占用内存>
<算法插件研制单位>.....</算法插件研制单位>
<算法插件联系人>.....</算法插件联系人>
</注册信息>
</算法插件注册信息>
```

### 2.4 算法依赖

由于科学计算的数据, 处理过程多种多样, 算法之间有可能有依赖关系。这种依赖关系并非调用关系, 而是数据处理过程与级别的关系。在本文中定义算法  $A_1$  依赖于算法  $A_2$ , 表示为  $D(A_1, A_2)$ 。是指如果要用算法  $A_1$  对某次实验的某个载荷获取数据进行处理, 必须要由算法  $A_2$  先对数据进行处理。

系统依赖图是程序基于图形的中间表示, 能清晰第表示出程序中的各种控制依赖和数据依赖关系及程序的语法与语义信息<sup>[9]</sup>。通过构建算法依赖图的方式进行算法间的依赖描述, 借用系统依赖图的概念, 但是做相应简化, 将语句节点、调用节点、参数节点等简化为单一算法节点, 将控制依赖、流依赖、声明依

赖、调用及传参等关系, 简化为单一的调用关系。因此算法依赖图可用简单的有向无环图来表示, 图中节点表示算法, 带方向的连接表示依赖关系。节点可以单独存在, 即不需依赖其他算法可独立进行数据处理。因算法不可循环依赖, 因此图为无环图。图中还有一类特殊的节点, 为底层库, 多为数据格式读写库、通用计算库等, 这些底层库并非自行开发。

算法注册时表示的  $D(A_1, A_2), D(A_2, A_4), \dots, D(A_m, A_n)$ , 最终在系统中合并为整个计算系统的算法依赖图。

图 7 为部分算法的依赖关系图示例。

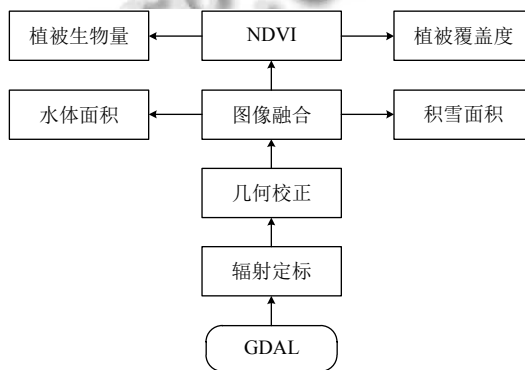


图 7 算法依赖图示例

系统在初始化计算任务时, 需根据算法依赖图辅助计算任务的制定, 以确保业务逻辑的顺利执行。

### 3 系统实现

在实际工作中, 用 Java 实现了临近空间探测数据处理算法集成调度系统, 布署于曙光天阔 I620-G20 机架式服务器中, 算法管理器与并行计算管理器相配合, 在 ZStack 私有云服务器中实现了多业务、多算法、多类数据的并行处理和业务逻辑的随时调整。系统运行硬件环境如图 8 所示。

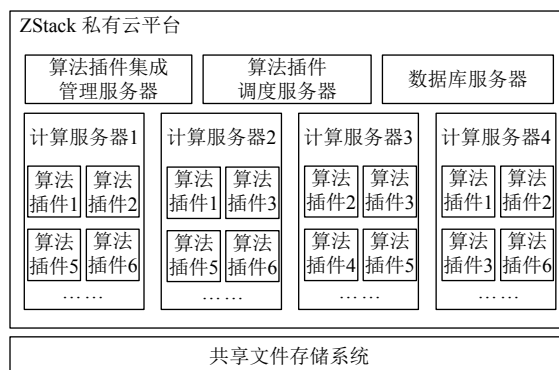


图 8 系统运行硬件环境图

临近空间探测数据处理算法集成调度系统包括:资源监控模块、算法集成管理模块、任务调度模块.资源监控模块收集各计算服务器的状态信息并向用户提供图形化展示;算法集成管理模块提供算法插件注册、注册信息管理、插件运行环境配置等功能;任务调度模块将队列中的数据任务合理分配并调用相关的算法插件执行数据处理任务.系统结构如图9.

系统可支持实现科学探测数据的处理,包括数据归档处理、辐射校正、几何校正、生态参数计算等算法共3类16个;探测数据处理任务分配用时少于0.2s;算法插件调度响应时间少于0.5s;可支持不少于32个探测数据处理任务并行执行.

系统具备良好的可扩展性,在实验期间,根据处理结果分析迭代的情况,对算法进行了多次版本升级,共计12次.版本升级过程中,大系统业务无影响、无宕机.且多个版本共存,可根据具体业务的不同,在工作

流中配置使用不同版本的算法,完成不同业务逻辑的实现.系统中的资源监控模块界面如图10所示.算法集成管理模块界面如图11所示.任务调度模块界面如图12所示.

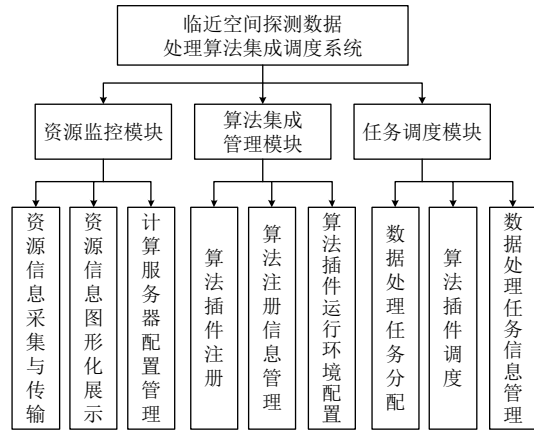


图9 系统结构图

任务列表	节点编号	IP地址	计算机名	CPU使用率	内存总量/内存剩余	操作系统
1. 2020-04-10 15:04:24 时序影像异常检测 处理中	1	192.168.0.152.80	Q-PC	CPU 33.8%	内存总量 8084 MB, 已用内存	操作系统 Windows 7
2. 2020-04-05 14:04:46 时序影像异常检测 处理中	2	192.168.20.6.80	DOUSHUAI-PC	CPU 0.7%	内存总量 8191 MB, 已用内存	操作系统 Windows 7
3. 2020-04-05 14:04:41 积雪面积 已完成	3	192.168.0.156.80	P-PC	CPU 28.5%	内存总量 8084 MB, 已用内存	操作系统 Windows 7

图10 资源监控模块界面截图

算法序号	算法名称	算法部署路径	算法版本	算法更新日期	实际语言	算法提供方	更新组件程序	编辑注册信息
1	紫外光谱仪原始数据处理	C:\SJ\Yexe\LKkygpy.exe	V1.1	2020-06-09	C++	张林	更新组件程序	编辑注册信息
2	臭氧探测仪原始数据	C:\SJ\Yexe\LKCY.exe	V1.0	2020-06-09	C++	刘维	更新组件程序	编辑注册信息
3	气体采样装置原始数据	C:\SJ\Yexe\LKYSCY.exe	V1.1	2020-06-09	C++	杨旭	更新组件程序	编辑注册信息
4	水体面积	C:\SJ\Yexe\sanjiangyuan\Debug\sanjiangyuan_4.exe	V1.0	2020-04-02	C#	陈霖霖	更新组件程序	编辑注册信息
5	积雪面积	C:\SJ\Yexe\sanjiangyuan\Debug\sanjiangyuan_5.exe	V1.0	2020-04-02	C#	陈霖霖	更新组件程序	编辑注册信息
6	时序影像异常检测	C:\SJ\Yexe\IRIR-3.6.3\bin\G4\IRscript.exe	V1.0	2020-04-02	R	周增光	更新组件程序	编辑注册信息
7	植被生物量	C:\SJ\Yexe\sanjiangyuan\Debug\sanjiangyuan_3.exe	V1.0	2020-04-02	C#	陈霖霖	更新组件程序	编辑注册信息
8	植被覆盖度	C:\SJ\Yexe\sanjiangyuan\Debug\sanjiangyuan_2.exe	V1.0	2020-04-02	C#	陈霖霖	更新组件程序	编辑注册信息
9	植被指数处理	C:\SJ\Yexe\sanjiangyuan\Debug\sanjiangyuan.exe	V1.0	2020-04-02	C#	陈霖霖	更新组件程序	编辑注册信息

图11 算法集成管理模块界面截图

任务编号	选用算法	执行状态	提交时间	启动参数	待处理数据	结果数据	下载产品	删除任务	开始处理
42	时序影像异常检测	accepted	2020-04-05 14:04:46	Z:\123456\F\Z\JQK067117.tif 编辑	F:\SJ\data\123456\F 选择 上传	暂无	下载	删除	开始处理
41	积雪面积	application	2020-04-05 14:04:41	Z:\123456\F\Z\JQK067116.tif 编辑	F:\SJ\data\123456\F 选择 上传	暂无	下载	删除	开始处理
40	水体面积	finished	2020-04-05 14:04:36	Z:\123456\F\Z\JQK067115.tif 编辑	F:\SJ\data\123456\F 选择 上传	Z\JQK067116.tif	下载	删除	开始处理
39	植被生物量	finished	2020-04-05 14:04:31	Z:\123456\F\Z\JQK067114.tif 编辑	F:\SJ\data\123456\F 选择 上传	Z\JQK067116.tif	下载	删除	开始处理
38	植被覆盖度	accepted	2020-04-05 14:04:25	Z:\123456\F\Z\JQK067113.tif 编辑	F:\SJ\data\123456\F 选择 上传	暂无	下载	删除	开始处理
37	植被指数处理	accepted	2020-04-05 14:04:18	Z:\123456\F\Z\JQK067111.tif 编辑	F:\SJ\data\123456\F 选择 上传	暂无	下载	删除	开始处理
36	植被指数处理	finished	2020-04-03 15:04:14	Z:\123456\F\Z\JQK067111.tif 编辑	F:\SJ\data\123456\F 选择 上传	Z\JQK067116.tif	下载	删除	开始处理
34	植被指数处理	accepted	2020-04-05 15:04:49	Z:\123456\F\Z\JQK0671212.tif 编辑	F:\SJ\data\123456\F 选择 上传	暂无	下载	删除	开始处理
33	植被指数处理	finished	2020-04-03 15:04:52	Z:\123456\F\Z\JQK067111.tif 编辑	F:\SJ\data\123456\F 选择 上传	Z\JQK067116.tif	下载	删除	开始处理
32	植被指数处理	finished	2020-04-03 15:04:09	Z:\123456\F\Z\JQK067111.tif 编辑	F:\SJ\data\123456\F 选择 上传	Z\JQK067116.tif	下载	删除	开始处理
31	植被指数处理	finished	2020-04-03 15:04:57	Z:\123456\F\Z\JQK067111.tif 编辑	F:\SJ\data\123456\F 选择 上传	Z\JQK067116.tif	下载	删除	开始处理
30	植被指数处理	finished	2020-04-03 15:04:08	Z:\123456\F\Z\JQK067111.tif 编辑	F:\SJ\data\123456\F 选择 上传	Z\JQK067116.tif	下载	删除	开始处理
29	时序影像异常检测	finished	2020-04-03 15:04:25	Z:\123456\F\Z\JQK067117.tif 编辑	F:\SJ\data\123456\F 选择 上传	Z\JQK067116.tif	下载	删除	开始处理
28	植被指数处理	finished	2020-04-03 15:04:20	Z:\123456\F\Z\JQK067111.tif 编辑	F:\SJ\data\123456\F 选择 上传	Z\JQK067116.tif	下载	删除	开始处理

图 12 任务调度模块界面截图

### 4 结论与展望

本文设计的分布式计算系统中的插件式算法集成方法, 经过两套科学实验计算系统的运行实验, 验证了该算法集成方法的有效性, 可有效的实现系统计算功能, 且灵活性高, 升级便利, 配置方便. 从目前的研究现状和发展趋势来看, 在云计算、雾计算等分布式计算架构下, 研究用于灵活科学计算的插件式算法集成算法, 对于科学实验系统所采集的数据开展灵活的处理与分析具有重要的实际意义. 由于目前的设计模式还无法实现插件自己解决依赖问题, 只能使用调度器, 在设立处理任务的时候, 用依赖图来人工辅助, 后续将持续研发, 实现任务初始化自动遍历依赖图, 规划计算逻辑和流程.

#### 参考文献

- 李延春. 软件插件技术的原理与实现. 计算机系统应用, 2003, 12(7): 24-26.
- 范海涛, 常克武, 金煌煌, 等. 基于插件的卫星集成平台软件开发技术. 信息化技术, 2014, (3): 45-48.

- 李俊娥, 周洞汝. “平台/插件”软件体系结构风格. 小型微型计算机系统, 2007, 28(5): 876-881. [doi: 10.3969/j.issn.1000-1220.2007.05.024]
- 史纪强, 何兴曙, 王志琼, 等. 基于插件技术的企业应用集成架构研究. 计算机与应用化学, 2012, 29(2): 191-194. [doi: 10.3969/j.issn.1001-4160.2012.02.015]
- 李冠宇, 刘军, 张俊. 分布式异构数据集成系统的研究与实现. 计算机应用研究, 2004, 21(3): 96-98. [doi: 10.3969/j.issn.1001-3695.2004.03.034]
- 戴佳男, 赵侃侃. 分布式对象在软件系统集成中的应用. 信息通信, 2020, (5): 159-160. [doi: 10.3969/j.issn.1673-1131.2020.05.072]
- 宋文婷, 赵建新, 艾冰, 等. 基于 OSGi 的军用指挥软件插件机制研究. 火力与指挥控制, 2019, 44(5): 172-176. [doi: 10.3969/j.issn.1002-0640.2019.05.037]
- 张小龙, 霍剑青, 袁泉, 等. 基于软件插件的虚拟实验资源库系统的构建. 中国科学技术大学学报, 2007, 37(3): 327-332. [doi: 10.3969/j.issn.0253-2778.2007.03.019]
- 王克朝, 王甜甜, 苏小红, 等. 面向程序理解的系统依赖图构建算法. 哈尔滨工业大学学报, 2013, 45(1): 78-84. [doi: 10.11918/hitxb20130115]