



(12) 发明专利申请

(10) 申请公布号 CN 103473377 A

(43) 申请公布日 2013. 12. 25

(21) 申请号 201310457653. 8

(51) Int. Cl.

(22) 申请日 2013. 09. 30

G06F 17/30 (2006. 01)

(71) 申请人 国家电网公司

G06F 9/44 (2006. 01)

地址 100053 北京市西城区长安街 86 号

申请人 国网冀北电力有限公司信通分公司
北京博望华科科技有限公司

(72) 发明人 苏丹 吴佳 吴舜 李坤 高崧

聂正璞 许大卫 杜剑雯 刘昀

来骥 王黎 李贤 李晓东 李超

刘志伟 杨帆 李雪梅 那琼澜

(74) 专利代理机构 北京五月天专利商标代理有限公司 11294

代理人 李永联

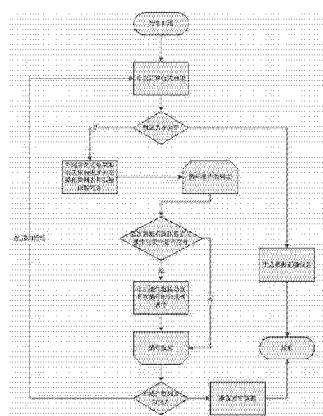
权利要求书1页 说明书2页 附图2页

(54) 发明名称

一种计算公式回圈点快速检测方法

(57) 摘要

本发明公开了一种计算公式回圈点快速检测方法,具体为:1)读取所有计算公式的列表;2)判断列表不为空,则进行递归分析;3)将所有计算公式等号左侧都标识为左操作数,右侧标识为右操作数;4)将所有的左操作数放到一个列表中,递归循环到一个计算公式时,依次判断其所有的右操作数是否在左操作数列表中,如果在,则抽取左操作数代表的公式到新的计算公式列表中,以备下次递归使用;如果不,则不作处理,继续下一个计算公式;5)当本次递归循环结束时,确认回圈点。本发明可有效解决计算机进行数据计算时进入死循环或将计算机资源耗尽的问题,能够提高计算机运行速度。本发明经过简单配置即可植入相关应用系统中,提高计算正确性。



1. 一种计算公式回圈点快速检测方法,其特征在于:

- 1) 读取所有计算公式的列表;
- 2) 判断列表不为空,则进行递归分析;
- 3) 将所有计算公式等号左侧都标识为左操作数,右侧标识为右操作数;

4) 将所有的左操作数放到一个列表中,递归循环到一个计算公式时,依次判断其所有的右操作数是否在左操作数列表中,如果在,则抽取左操作数代表的公式到新的计算公式列表中,以备下次递归使用;如果不,则不作处理,继续下一个计算公式;

5) 当本次递归循环结束时,如果当前的左操作数列表为空,则表示有回圈,将组织好的计算公式列表分析,收集所有左操作数集合和右操作数集合,把左操作数在右操作数集合中出现的公式提取出,即可标识为回圈点。

2. 根据权利要求 1 所述的计算公式回圈点快速检测方法,其特征在于,当检测到有依
赖回圈时,收集引起回圈的节点信息,整理后以异常形式抛出,应用系统检测到异常形式,
捕获后打开异常信息,便可获知所需要的信息。

3. 根据权利要求 1 所述的计算公式回圈点快速检测方法,其特征在于,当检查到有依
赖回圈时,将回圈信息收集整理,最后以参数形式返回,应用系统通过返回值来确定回圈位
置。

4. 根据权利要求 1-3 任一项所述的计算公式回圈点快速检测方法,其特征在于,所述
步骤 2) 的中断的条件有两个,一是所有公式检测完成,二是公式中出现异常回圈点。

5. 根据权利要求 1-4 任一项所述的计算公式回圈点快速检测方法,其特征在于,所述
步骤 3) 中,所有右操作数可以为一个计算公式,且如不受计算机存储容量限制,理论上可无
限嵌套。

一种计算公式回圈点快速检测方法

技术领域

[0001] 本发明属于计算机领域,涉及计算机数据结构中的单向链表、双向链表和数学公式计算等领域所涉及的问题,具体涉及复杂计算公式回圈点快速检测方法。

背景技术

[0002] 在实际业务系统中,经常遇到数据结果需要进行计算得出的情况。在计算公式中,并非所有的计算都为常量,有的为变量。而变量可能是其它公式的计算结果,但在定义其它计算公式时,就需要确定不引用到当前的计算公式,否则就会引起回圈,计算时会进入死循环或栈溢出等错误。为避免这种情况出现,需要在计算前就找到引起回圈点,如图 1 中粗线部分。

发明内容

[0003] 本发明公开了一种计算公式回圈点快速检测方法,具体为:

- 1) 读取所有计算公式的列表;
- 2) 判断列表不为空,则进行递归分析;
- 3) 将所有计算公式等号左侧都标识为左操作数,右侧标识为右操作数;
- 4) 将所有的左操作数放到一个列表中,递归循环到一个计算公式时,依次判断其所有的右操作数是否在左操作数列表中,如果在,则抽取左操作数代表的公式到新的计算公式列表中,以备下次递归使用;如果不作处理,继续下一个计算公式;
- 5) 当本次递归循环结束时,如果当前的左操作数列表为空,则表示有回圈,将组织好的计算公式列表分析,收集所有左操作数集合和右操作数集合,把左操作数在右操作数集合中出现的公式提取出,即可标识为回圈点。

[0004] 进一步,当检测到有依赖回圈时,收集引起回圈的节点信息,整理后以异常形式抛出,应用系统检测到异常形式,捕获后打开异常信息,便可获知所需要的信息。

[0005] 进一步,当检查到有依赖回圈时,将回圈信息收集整理,最后以参数形式返回,应用系统通过返回值来确定回圈位置。

[0006] 其中,所述步骤 2)的中断的条件有两个,一是所有公式检测完成,二是公式中出现异常回圈点。

[0007] 其中,所述步骤 3)中,所有右操作数可以为一个计算公式,且如不受计算机存储容量限制,理论上可无限嵌套。

[0008] 本发明可有效解决计算机进行数据计算时进入死循环或将计算机资源耗尽的问题,能够提高计算机运行速度。本发明经过简单配置即可植入相关应用系统中,提高计算正确性。

附图说明

[0009] 图 1 为存在回圈的数据结构示意图;

图 2 为正常的数据结构示意图；

图 3 为分析流程示意图。

具体实施方式

[0010] 下面结合附图对本发明进行具体说明。

[0011] 本发明的分析流程如图 3 所示。

[0012] 1. 读取所有计算公式的列表。

[0013] 2. 如果列表不为空，则进行递归分析。中断的条件有两个，一是所有公式检测完成，二是公式中出现异常回圈点。

[0014] 3. 将所有计算公式等号左侧都标识为左操作数，右侧标识为右操作数，右操作数也可以为一个计算公式。

[0015] 4. 首先将所有的左操作数放到一个列表中，递归循环到一个计算公式时，依次判断其所有的右操作数是否在左操作数列表中，如果在，则抽取左操作数代表的公式到新的计算公式列表中，以备下次递归使用。如果不作处理，则继续下一个公式。

[0016] 5. 当本次递归循环结束时，如果当前的左操作数列表为空，则表示有回圈。需将组织好的计算公式列表分析，收集所有左操作数集合和右操作数集合，把左操作数在右操作数集合中出现的公式提取出，即可标识为回圈点。

[0017] 在实际应用中，对于检测到的回圈点可以通过多种机制来处理。

[0018] 1. 通过抛出异常形式：当检测到有依赖回圈时，收集引起回圈的节点信息，整理后以异常形式抛出，应用系统检测到异常形式，捕获后打开异常信息，便可获知所需要的信息。

[0019] 2. 通过返回值：当检查到有依赖回圈时，将回圈信息收集整理，最后以参数形式返回。应用系统通过返回值来确定回圈位置。

[0020] 举例说明：如果有 4 个公式 $a=b+c$; $b=d+e$; $d=f+g$; $g=a+c$ ；左操作数集合分别为 $\{a, b, d, g\}$ ，右操作数集合为 $\{a, b, c, d, e, f, g\}$ 。经过上述图 3 所示的流程分析，则会得出公式 $a=b+c$ 和 $g=a+c$ 之间存在回圈依赖的结论，以异常形式抛出。

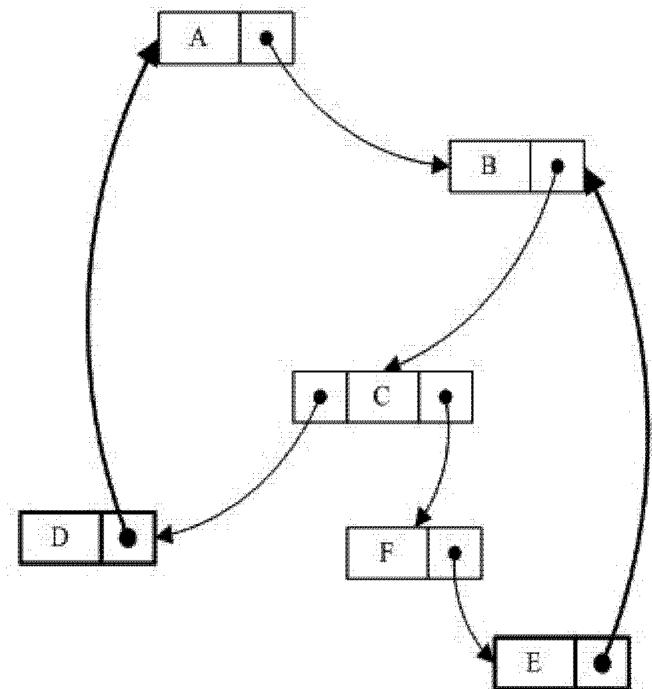


图 1

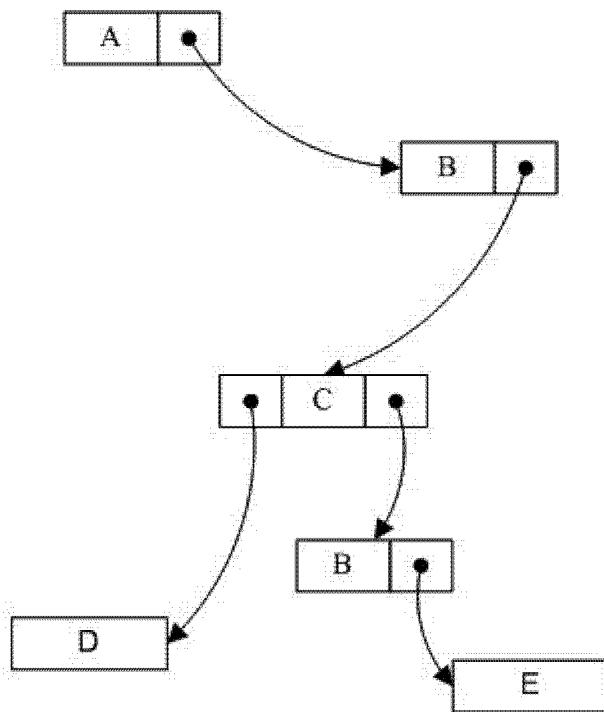


图 2

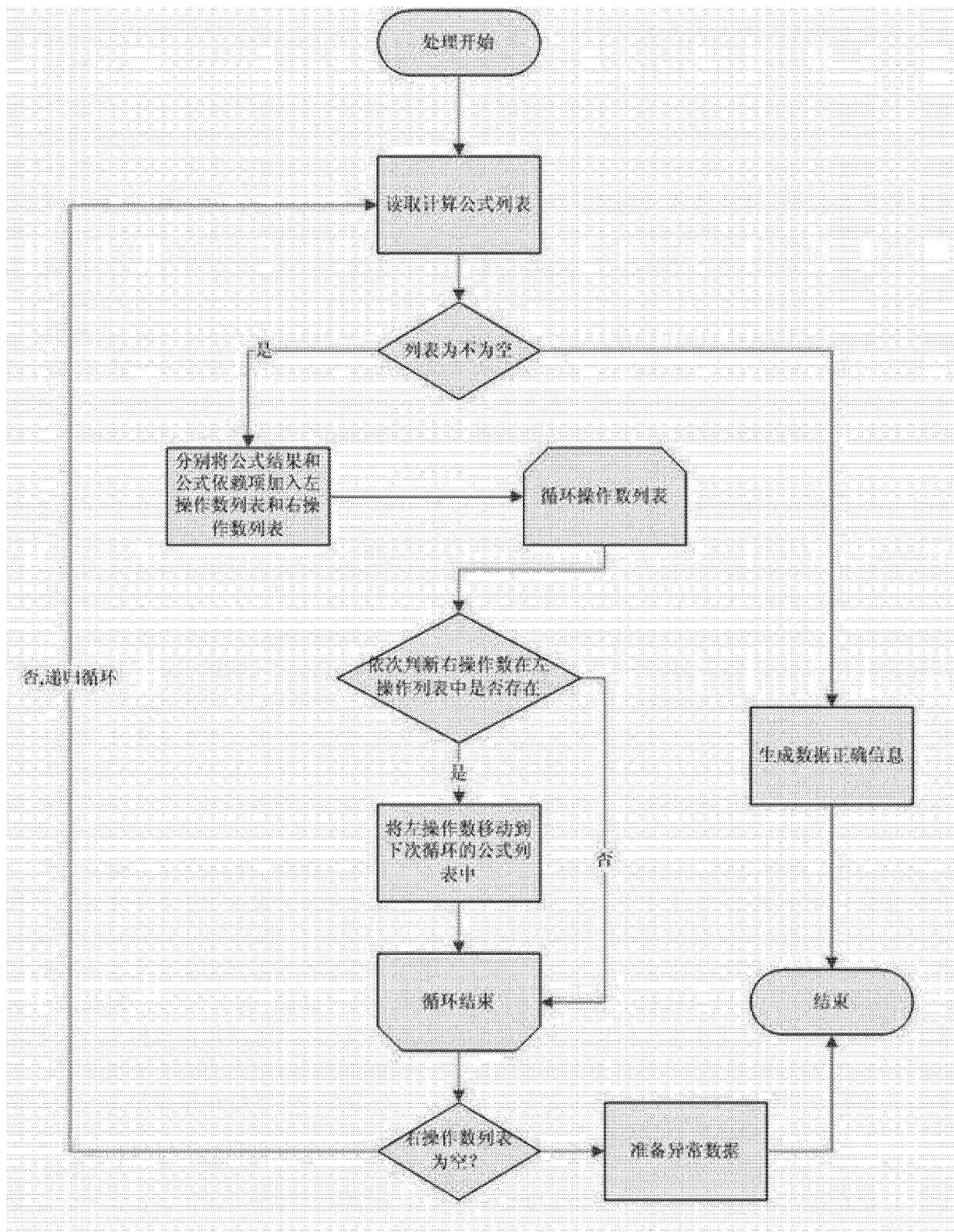


图 3