



(12)发明专利申请

(10)申请公布号 CN 109614571 A

(43)申请公布日 2019.04.12

(21)申请号 201811173796.5

(22)申请日 2018.10.09

(71)申请人 阿里巴巴集团控股有限公司
地址 英属开曼群岛大开曼资本大厦一座四
层847号邮箱

(72)发明人 钱瑜

(74)专利代理机构 北京国昊天诚知识产权代理
有限公司 11315
代理人 许振新 朱文杰

(51)Int.Cl.
G06F 17/10(2006.01)

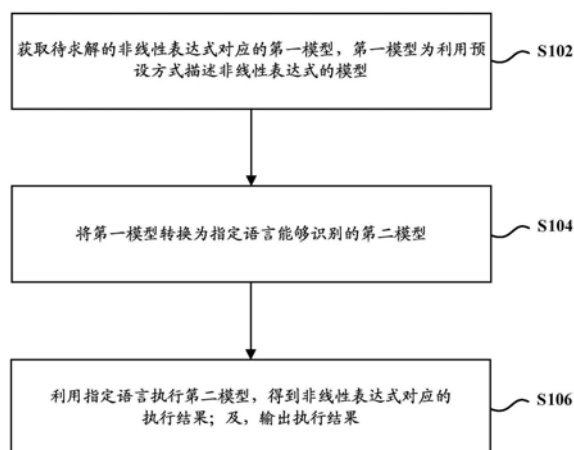
权利要求书3页 说明书13页 附图4页

(54)发明名称

非线性规划问题的处理方法及装置

(57)摘要

本说明书一个或多个实施例公开了一种非线性规划问题的处理方法及装置,用以有效求解分段函数的非线性规划问题。所述方法包括:获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;将所述第一模型转换为指定语言能够识别的第二模型;利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。



1. 一种非线性规划问题的处理方法,包括:
 - 获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;
 - 将所述第一模型转换为指定语言能够识别的第二模型;
 - 利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。
2. 根据权利要求1所述的方法,所述获取待求解的非线性表达式对应的第一模型,包括:
 - 获取所述非线性表达式;
 - 利用所述预设方式描述所述非线性表达式,得到所述非线性表达式对应的第一模型;其中,所述第一模型包含输入表达式、输出表达式、演绎表达式、约束表达式、目标表达式中的至少一项表达式。
3. 根据权利要求2所述的方法,所述将所述第一模型转换为指定语言能够识别的第二模型,包括以下至少一项:
 - 在所述第一模型中的所述输入表达式和/或所述演绎表达式的结尾位置添加预设符号;其中,所述预设符号为所述指定语言能够识别的符号;
 - 将所述第一模型中的所述约束表达式转换为布尔表达式;及,将所述布尔表达式赋值给所述约束表达式对应的赋值变量;
 - 将所述第一模型中的第一参数转换为所述指定语言能够识别的第二参数。
4. 根据权利要求3所述的方法,所述将所述布尔表达式赋值给所述约束表达式对应的赋值变量,包括:
 - 计算所述约束表达式对应的哈希值;
 - 根据所述哈希值和/或预设参数,生成所述约束表达式对应的赋值变量;
 - 将所述布尔表达式赋值给所述生成的赋值变量。
5. 根据权利要求3或4所述的方法,所述利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果,包括:
 - 多次迭代以下步骤,直至获得所述非线性表达式的最优解和/或达到预设迭代次数:
 - 生成一组输出变量值;
 - 将所述输出变量值作为所述第二模型中的各所述表达式的输入值进行计算,得到所述非线性表达式的第一解;
 - 判断所述第一解是否满足所述约束条件;
 - 若满足所述约束条件,则进一步判断所述第一解是否为所述最优解;若为所述最优解,则输出所述最优解;
 - 若不满足所述约束条件和/或不为所述最优解,则重新生成一组输出变量值。
6. 根据权利要求5所述的方法,所述判断所述第一解是否满足所述约束条件,包括:
 - 根据所述第一解确定所述布尔表达式的布尔值;及,判断所述布尔值是否为真;
 - 若是,则确定所述第一解满足所述约束条件;若否,则确定所述第一解不满足所述约束条件。
7. 根据权利要求2所述的方法,所述执行结果为最优解或无解;

相应的,所述方法还包括:

若所述执行结果为无解,则按照预设规则更新所述目标表达式;其中,所述更新包括降低所述目标表达式对应的目标值。

8. 一种非线性规划问题的处理装置,包括:

获取模块,用于获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;

转换模块,用于将所述第一模型转换为指定语言能够识别的第二模型;

执行模块,用于利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。

9. 根据权利要求8所述的装置,所述获取模块包括:

获取单元,用于获取所述非线性表达式;

描述单元,用于利用所述预设方式描述所述非线性表达式,得到所述非线性表达式对应的第一模型;其中,所述第一模型包含输入表达式、输出表达式、演绎表达式、约束表达式、目标表达式中的至少一项表达式。

10. 根据权利要求9所述的装置,所述转换模块包括以下至少一项:

添加单元,用于在所述第一模型中的所述输入表达式和/或所述演绎表达式的结尾位置添加预设符号;其中,所述预设符号为所述指定语言能够识别的符号;

转换及赋值单元,用于将所述第一模型中的所述约束表达式转换为布尔表达式;及,将所述布尔表达式赋值给所述约束表达式对应的赋值变量;

转换单元,用于将所述第一模型中的第一参数转换为所述指定语言能够识别的第二参数。

11. 根据权利要求10所述的装置,所述转换及赋值单元还用于:

计算所述约束表达式对应的哈希值;

根据所述哈希值和/或预设参数,生成所述约束表达式对应的赋值变量;

将所述布尔表达式赋值给所述生成的赋值变量。

12. 根据权利要求10或11所述的装置,所述执行模块包括:

迭代单元,用于多次迭代以下步骤,直至获得所述非线性表达式的最优解和/或达到预设迭代次数:

生成一组输出变量值;

将所述输出变量值作为所述第二模型中的各所述表达式的输入值进行计算,得到所述非线性表达式的第一解;

判断所述第一解是否满足所述约束条件;

若满足所述约束条件,则进一步判断所述第一解是否为所述最优解;若为所述最优解,则输出所述最优解;

若不满足所述约束条件和/或不为所述最优解,则重新生成一组输出变量值。

13. 根据权利要求12所述的装置,所述迭代单元还用于:

根据所述第一解确定所述布尔表达式的布尔值;及,判断所述布尔值是否为真;

若是,则确定所述第一解满足所述约束条件;若否,则确定所述第一解不满足所述约束条件。

14. 根据权利要求9所述的装置,所述执行结果为最优解或无解;

相应的,所述装置还包括:

更新模块,用于若所述执行结果为无解,则按照预设规则更新所述目标表达式;其中,所述更新包括降低所述目标表达式对应的目标值。

15. 一种非线性规划问题的处理设备,包括:

处理器;以及

被安排成存储计算机可执行指令的存储器,所述可执行指令在被执行时使所述处理器:

获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;

将所述第一模型转换为指定语言能够识别的第二模型;

利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。

16. 一种存储介质,用于存储计算机可执行指令,所述可执行指令在被执行时实现以下流程:

获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;

将所述第一模型转换为指定语言能够识别的第二模型;

利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。

非线性规划问题的处理方法及装置

技术领域

[0001] 本说明书涉及非线性规划技术领域,尤其涉及一种非线性规划问题的处理方法及装置。

背景技术

[0002] 非线性规划问题的求解有很多方法,现有技术中,常用的求解非线性规划问题的方法为拉格朗日乘子法、罚函数法、增广拉格朗日乘子法等。当非线性规划问题的目标函数为连续函数、且约束条件较少的情况下,上述方法的求解效率会比较高并且有效。

[0003] 但是,如果目标函数中含有的复合子函数为分段函数,那么上述几种方法也就不适用了。原因在于,上述几种方法在求解过程中,都需要先将有约束的非线性规划问题转变为无约束的非线性规划问题,然后求解目标函数的一阶导数。而分段函数意味着函数不连续,存在不可导的地方,即无法求解目标函数的导数,从而导致无法采用上述几种方法来求解非线性规划问题。

发明内容

[0004] 本说明书一个或多个实施例的目的是提供一种非线性规划问题的处理方法及装置,用以有效求解分段函数的非线性规划问题。

[0005] 为解决上述技术问题,本说明书一个或多个实施例是这样实现的:

[0006] 一方面,本说明书一个或多个实施例提供一种非线性规划问题的处理方法,包括:

[0007] 获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;

[0008] 将所述第一模型转换为指定语言能够识别的第二模型;

[0009] 利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。

[0010] 在一个实施例中,所述获取待求解的非线性表达式对应的第一模型,包括:

[0011] 获取所述非线性表达式;

[0012] 利用所述预设方式描述所述非线性表达式,得到所述非线性表达式对应的第一模型;其中,所述第一模型包含输入表达式、输出表达式、演绎表达式、约束表达式、目标表达式中的至少一项表达式。

[0013] 在一个实施例中,所述将所述第一模型转换为指定语言能够识别的第二模型,包括以下至少一项:

[0014] 在所述第一模型中的所述输入表达式和/或所述演绎表达式的结尾位置添加预设符号;其中,所述预设符号为所述指定语言能够识别的符号;

[0015] 将所述第一模型中的所述约束表达式转换为布尔表达式;及,将所述布尔表达式赋值给所述约束表达式对应的赋值变量;

[0016] 将所述第一模型中的第一参数转换为所述指定语言能够识别的第二参数。

- [0017] 在一个实施例中,所述将所述布尔表达式赋值给所述约束表达式对应的赋值变量,包括:
- [0018] 计算所述约束表达式对应的哈希值;
- [0019] 根据所述哈希值和/或预设参数,生成所述约束表达式对应的赋值变量;
- [0020] 将所述布尔表达式赋值给所述生成的赋值变量。
- [0021] 在一个实施例中,所述利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果,包括:
- [0022] 多次迭代以下步骤,直至获得所述非线性表达式的最优解和/或达到预设迭代次数:
- [0023] 生成一组输出变量值;
- [0024] 将所述输出变量值作为所述第二模型中的各所述表达式的输入值进行计算,得到所述非线性表达式的第一解;
- [0025] 判断所述第一解是否满足所述约束条件;
- [0026] 若满足所述约束条件,则进一步判断所述第一解是否为所述最优解;若为所述最优解,则输出所述最优解;
- [0027] 若不满足所述约束条件和/或不作为所述最优解,则重新生成一组输出变量值。
- [0028] 在一个实施例中,所述判断所述第一解是否满足所述约束条件,包括:
- [0029] 根据所述第一解确定所述布尔表达式的布尔值;及,判断所述布尔值是否为真;
- [0030] 若是,则确定所述第一解满足所述约束条件;若否,则确定所述第一解不满足所述约束条件。
- [0031] 在一个实施例中,所述执行结果为最优解或无解;
- [0032] 相应的,所述方法还包括:
- [0033] 若所述执行结果为无解,则按照预设规则更新所述目标表达式;其中,所述更新包括降低所述目标表达式对应的目标值。
- [0034] 另一方面,本说明书一个或多个实施例提供一种非线性规划问题的处理装置,包括:
- [0035] 获取模块,用于获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;
- [0036] 转换模块,用于将所述第一模型转换为指定语言能够识别的第二模型;
- [0037] 执行模块,用于利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。
- [0038] 在一个实施例中,所述获取模块包括:
- [0039] 获取单元,用于获取所述非线性表达式;
- [0040] 描述单元,用于利用所述预设方式描述所述非线性表达式,得到所述非线性表达式对应的第一模型;其中,所述第一模型包含输入表达式、输出表达式、演绎表达式、约束表达式、目标表达式中的至少一项表达式。
- [0041] 在一个实施例中,所述转换模块包括以下至少一项:
- [0042] 添加单元,用于在所述第一模型中的所述输入表达式和/或所述演绎表达式的结尾位置添加预设符号;其中,所述预设符号为所述指定语言能够识别的符号;

- [0043] 转换及赋值单元,用于将所述第一模型中的所述约束表达式转换为布尔表达式;及,将所述布尔表达式赋值给所述约束表达式对应的赋值变量;
- [0044] 转换单元,用于将所述第一模型中的第一参数转换为所述指定语言能够识别的第二参数。
- [0045] 在一个实施例中,所述转换及赋值单元还用于:
- [0046] 计算所述约束表达式对应的哈希值;
- [0047] 根据所述哈希值和/或预设参数,生成所述约束表达式对应的赋值变量;
- [0048] 将所述布尔表达式赋值给所述生成的赋值变量。
- [0049] 在一个实施例中,所述执行模块包括:
- [0050] 迭代单元,用于多次迭代以下步骤,直至获得所述非线性表达式的最优解和/或达到预设迭代次数:
- [0051] 生成一组输出变量值;
- [0052] 将所述输出变量值作为所述第二模型中的各所述表达式的输入值进行计算,得到所述非线性表达式的第一解;
- [0053] 判断所述第一解是否满足所述约束条件;
- [0054] 若满足所述约束条件,则进一步判断所述第一解是否为所述最优解;若为所述最优解,则输出所述最优解;
- [0055] 若不满足所述约束条件和/或不作为所述最优解,则重新生成一组输出变量值。
- [0056] 在一个实施例中,所述迭代单元还用于:
- [0057] 根据所述第一解确定所述布尔表达式的布尔值;及,判断所述布尔值是否为真;
- [0058] 若是,则确定所述第一解满足所述约束条件;若否,则确定所述第一解不满足所述约束条件。
- [0059] 在一个实施例中,所述执行结果为最优解或无解;
- [0060] 相应的,所述装置还包括:
- [0061] 更新模块,用于若所述执行结果为无解,则按照预设规则更新所述目标表达式;其中,所述更新包括降低所述目标表达式对应的目标值。
- [0062] 再一方面,本说明书一个或多个实施例提供一种非线性规划问题的处理设备,包括:
- [0063] 处理器;以及
- [0064] 被安排成存储计算机可执行指令的存储器,所述可执行指令在被执行时使所述处理器:
- [0065] 获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;
- [0066] 将所述第一模型转换为指定语言能够识别的第二模型;
- [0067] 利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。
- [0068] 再一方面,本申请实施例提供一种存储介质,用于存储计算机可执行指令,所述可执行指令在被执行时实现以下流程:
- [0069] 获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述

所述非线性表达式的模型;所述非线性表达式包含约束条件;

[0070] 将所述第一模型转换为指定语言能够识别的第二模型;

[0071] 利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。

[0072] 采用本说明书一个或多个实施例的技术方案,通过获取待求解的非线性表达式对应的第一模型,该第一模型为利用预设方式描述非线性表达式的模型,并将第一模型转换为指定语言能够识别的第二模型,进而利用指定语言执行第二模型,得到非线性表达式对应的执行结果,并输出该执行结果。可见,该技术方案在求解非线性表达式时,无需对非线性表达式求导,只需利用指定语言执行非线性表达式对应的第二模型即可,因此相较于现有技术中仅能通过对非线性表达式求导的方式而言,该技术方案能够有效求解出分段函数的非线性规划问题。

附图说明

[0073] 为了更清楚地说明本说明书一个或多个实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本说明书一个或多个实施例中记载的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0074] 图1是根据本说明书一实施例的一种非线性规划问题的处理方法的示意性流程图;

[0075] 图2是根据本说明书另一实施例的一种非线性规划问题的处理方法的示意性流程图

[0076] 图3是根据本说明书一实施例的一种非线性规划问题的处理装置的示意性框图;

[0077] 图4是根据本说明书一实施例的一种非线性规划问题的处理设备的示意性框图。

具体实施方式

[0078] 本说明书一个或多个实施例提供一种非线性规划问题的处理方法及装置,用以有效求解分段函数的非线性规划问题。

[0079] 为了使本技术领域的人员更好地理解本说明书一个或多个实施例中的技术方案,下面将结合本说明书一个或多个实施例中的附图,对本说明书一个或多个实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本说明书一部分实施例,而不是全部的实施例。基于本说明书一个或多个实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都应当属于本说明书一个或多个实施例保护的范围内。

[0080] 图1是根据本说明书一实施例的一种非线性规划问题的处理方法的示意性流程图,如图1所示,该方法包括:

[0081] S102,获取待求解的非线性表达式对应的第一模型,第一模型为利用预设方式描述非线性表达式的模型。

[0082] 其中,非线性表达式包含约束条件。

[0083] S104,将第一模型转换为指定语言能够识别的第二模型。

[0084] S106,利用指定语言执行第二模型,得到非线性表达式对应的执行结果;及,输出执行结果。

[0085] 本实施例中,指定语言可以是Groovy语言,其支持Java函数。

[0086] 采用本说明书一个或多个实施例的技术方案,通过获取待求解的非线性表达式对应的第一模型,该第一模型为利用预设方式描述非线性表达式的模型,并将第一模型转换为指定语言能够识别的第二模型,进而利用指定语言执行第二模型,得到非线性表达式对应的执行结果,并输出该执行结果。可见,该技术方案在求解非线性表达式时,无需对非线性表达式求导,只需利用指定语言执行非线性表达式对应的第二模型即可,因此相较于现有技术中仅能通过对非线性表达式求导的方式而言,该技术方案能够有效求解出分段函数的非线性规划问题。

[0087] 以下详细介绍上述实施例提供的非线性规划问题的处理方法。

[0088] 首先获取待求解的非线性表达式对应的第一模型。在一个实施例中,可按照下述方法获取非线性表达式对应的第一模型:首先获取非线性表达式,然后利用预设方式描述非线性表达式,即得到非线性表达式对应的第一模型;其中,第一模型包含输入表达式、输出表达式、演绎表达式、约束表达式、目标表达式中的至少一项表达式。

[0089] 其中,预设方式包括预设字符,如字符“input”表示输入,字符“@out_n”表示输出,字符“indirect”表示演绎,字符“@f”表示函数调用,“@target()”表示后续的表达式为目标表达式;等等。

[0090] 以下列举几种第一模型中的表达式:

[0091] (1) 输入表达式:

[0092] `input_ICBC_flowability_limit=4500000000`

[0093] (2) 输出表达式:

[0094] `@out_n((0.0,1.0),0.01):output_indirect_ICBC_antbank1202_ebpp10006_distribution_ratio`

[0095] 其中,@out_n表示输出变量是一个普通的输出变量,((0.0,1.0),0.01)表示值域范围为(0.0,1.0),步长为0.01。

[0096] (3) 演绎表达式:

[0097] `indirect_ICBC_ebpp10006_interval_cnt=indirect_ICBC_ebpp10006_cnt_ratio*input_ICBC_ebpp10006_cnt`

[0098] 演绎表达式表示各中间变量换算演绎的过程。如上述演绎表达式的含义为:“indirect_ICBC_ebpp10006_interval_cnt”等于“indirect_ICBC_ebpp10006_cnt_ratio”与“input_ICBC_ebpp10006_cnt”相乘的结果。

[0099] 此外,由于本实施例中的指定语言——Groovy语言支持Java函数,因此上述演绎表达式还可采用如下方式表述:

[0100] `indirect_ICBC_ebpp10006_cnt_ratio=@f:get_cnt_ratio("ICBC","ebpp10006",indirect_ICBC_ebpp10006_amount_interval)`

[0101] 其中,“@f”表示此处是一个函数调用。

[0102] (4) 约束表达式:

[0103] `input_ICBC_fnccpay20003_system_success_ratio_min<=ICBC_fnccpay20003_`

system_success_ratio

[0104] (5) 目标表达式:

[0105] @target(1):indirect_objective=input_ICBC_ebpp10006_cnt*ICBC_ebpp10006_system_success_ratio+input_ICBC_fnccpay20003_cnt*ICBC_fnccpay20003_system_success_ratio

[0106] 其中,“@target(1)”表示此处的表达式为目标表达式,括号“()”中的“1”表示目标为最大化。此外,若“()”内为“2”,则表示目标为最小化。

[0107] 获取到待求解的非线性表达式对应的第一模型后,将该第一模型转换为指定语言能够识别的第二模型。在一个实施例中,将第一模型转换为Groovy语言能够识别的第二模型,转换方法可包括对表达式中变量的优化及对表达式本身的优化,其中,变量可分为常量、演绎变量、输出变量和目标变量。具体的转换方法可包括以下a、b、c中的至少一项:

[0108] a、在第一模型中的输入表达式和/或演绎表达式的结尾位置添加预设符号;其中,预设符号为指定语言能够识别的符号。

[0109] 其中,预设符号包括分号“;”在输入表达式和/或演绎表达式的结尾位置添加分号“;”之后,输入表达式和/或演绎表达式表达式可被Groovy语言识别。

[0110] b、将第一模型中的约束表达式转换为布尔表达式;及,将布尔表达式赋值给约束表达式对应的赋值变量。

[0111] 对于本身即为布尔表达式的约束表达式,可直接将约束表达式作为转换后的布尔表达式,进而赋值给对应的赋值变量;对于本身不为布尔表达式的约束表达式,则需将约束表达式转换为布尔表达式,进而赋值给对应的赋值变量。

[0112] 在将约束表达式转换为布尔表达式时,需采用布尔表达式的表示方式描述约束表达式。例如,将约束表达式中的文字“大于”转换为符号“>”。

[0113] 在一个实施例中,将布尔表达式赋值给约束表达式对应的赋值变量时,可先计算约束表达式对应的哈希值,然后根据哈希值和/或预设参数,生成约束表达式对应的赋值变量,最后将布尔表达式赋值给生成的赋值变量。

[0114] 其中,对哈希值的计算属于现有技术,此处不再赘述。

[0115] 预设参数可以是预设的可被指定语言识别的任意参数,如预设参数为“constrainId”,那么赋值变量可表示为“constrainId+md5”。

[0116] c、将第一模型中的第一参数转换为指定语言能够识别的第二参数。

[0117] 其中,第一参数可包括函数调用参数,例如,将第一参数“@f函数”转换为第二参数“函数名.execute(args)”,这两种方式均表示调用函数,且均支持变量的回调,即在演绎表达式的执行过程中,可根据需要从外部(如预先指定的数据库)中调取变量。

[0118] 例如,约束表达式为:

[0119] input_ICBC_fnccpay20003_system_success_ratio_min<=ICBC_fnccpay20003_system_success_ratio

[0120] 按照上述a、b、c方式,将上述约束表达式转换为如下:

[0121] constraintIdle111xb123123123123=input_ICBC_fnccpay20003_system_success_ratio_min<=ICBC_fnccpay20003_system_success_ratio;

[0122] 由上述转换可看出,在转换过程中,由于原始的约束表达式本身即为布尔表达式,

因此可直接将原始的约束表达式确定为转换后的布尔表达式,并将其赋值给赋值变量,其中,赋值变量为“constraintIdle111xb123123123123”。在该赋值变量中,“constraintId”为预设参数,“1e111xb123123123123”为约束表达式的md5值。此外,还在约束表达式的结尾位置添加了分号“;”。至此,转换后的表达式可被指定语言所识别。

[0123] 将第一模型转换为指定语言能够识别的第二模型之后,利用指定语言执行第二模型,以得到非线性表达式对应的执行结果。具体的,在利用指定语言执行第二模型时,可多次迭代以下步骤,直至获得非线性表达式的最优解和/或达到预设迭代次数:

[0124] A1、生成一组输出变量值。

[0125] 在执行A1时,可随机生成一组输出变量值,输出变量值的变量数与非线性表达式的目标解的个数相同。

[0126] 在一个实施例中,可在预设的值域范围内生成输出变量值,具体包括两种生成方式:一种是在预设的值域范围内随机生成输出变量值,另一种是按照预设的步长依次生成输出变量值。当按照预设的步长依次生成输出变量值时,步长可设定为固定步长,也可设定为可变步长。其中,可变步长可按照从大到小的规则进行调整。例如,对于前10组输出变量值,以步长 x 生成每组输出变量值;对于后10组输出变量值,以步长 y 生成每组输出变量值;其中, $x > y$ 。

[0127] A2、将输出变量值作为第二模型中的各表达式的输入值进行计算,得到非线性表达式的第一解。

[0128] A3、判断第一解是否满足约束条件;若满足约束条件,则进一步判断第一解是否为最优解;若为最优解,则输出最优解;若不满足约束条件和/或不满足最优解,则重新生成一组输出变量值。

[0129] 在一个实施例中,判断第一解是否满足约束条件时,可根据第一解确定约束表达式对应的布尔表达式的布尔值,并判断布尔值是否为真。若布尔值为真,则确定第一解满足约束条件;若布尔值不为真,则确定第一解不满足约束条件。其中,约束表达式对应的布尔表达式指由约束表达式转换成的布尔表达式。

[0130] 例如,对于如下转换后的约束表达式:

[0131] `constraintIdle111xb123123123123=input_ICBC_fnccpay20003_system_success_ratio_min<=ICBC_fnccpay20003_system_success_ratio;`

[0132] 在判断第一解是否满足约束条件时,可将第一解代入上述约束表达式中的布尔表达式“`input_ICBC_fnccpay20003_system_success_ratio_min<=ICBC_fnccpay20003_system_success_ratio`”中,并判断结果是否为真,即判断代入第一解后的布尔表达式是为真,若为真,则确定第一解满足约束条件;若不为真,则确定第一解不满足约束条件。

[0133] 在执行A3时,若第一解满足约束条件,则进一步判断第一解是否为最优解。判断第一解是否为最优解时,可将第一解与当前留存的最新解进行比较,进而从第一解与最新解中选择出更优的解,并将更优的解替代当前留存的最新解。然后,继续执行上述A1-A3,得到的第一解继续与当前留存的最新解进行比较,直至获得最优解。

[0134] 在一个实施例中,执行结果为最优解或无解。当执行结果为最优解时,输出最优解。当执行结果为无解时,可按照预设规则更新目标表达式,如降低目标表达式对应的目标值,其中,目标值指的是目标表达式中用于约束非线性规划问题的目标的因素。

[0135] 例如,对于目标表达式“通过率>90%”,其中用于约束通过率的因素——90%即为该目标表达式对应的目标值。当该目标表达式对应的执行结果为无解时,可相应降低目标值90%,如将目标值90%降低为85%。

[0136] 在一个实施例中,可按照预设的步长降低目标值,其中,步长可以是固定步长,也可以是可变步长。例如,第一次降低目标值时,可按照5%的步长将目标值90%降低为85%,第二次降低目标值时,可按照3%的步长将目标值85%降低为82%。

[0137] 图2是根据本说明书另一实施例的一种非线性规划问题的处理方法的示意性流程图,如图2所示,该方法包括:

[0138] S201,获取非线性表达式,并利用预设方式描述非线性表达式,得到非线性表达式对应的第一模型。

[0139] 其中,非线性表达式包含约束条件。第一模型包含输入表达式、输出表达式、演绎表达式、约束表达式、目标表达式中的至少一项表达式。预设方式包括预设字符,如字符“input”表示输入,字符“@out_n”表示输出,字符“indirect”表示演绎,字符“@f”表示函数调用,“@target()”表示后续的表达式为目标表达式;等等。

[0140] S202,将第一模型转换为Groovy语言能够识别的第二模型。

[0141] 具体转换方法包括以下(1)-(3):

[0142] (1)在输入表达式及演绎表达式的结尾位置添加“;”。

[0143] (2)将约束表达式转换为布尔表达式,并将布尔表达式赋值给约束表达式对应的赋值变量。其中,赋值变量采用“constrainId+md5”的方式表示,“md5”为约束表达式的哈希值。

[0144] (3)将函数调用“@f函数”转换为“函数名.execute(args)”。其中,“@f函数”及转换后的“函数名.execute(args)”均表示函数调用,且均支持变量的回调,即在演绎表达式的执行过程中,可根据需要从外部(如预先指定的数据库)中调取变量。

[0145] S203,根据第二模型生成非线性规划问题的求解任务。

[0146] S204,在预设的值域范围内随机生成一组输出变量值。

[0147] S205,将输出变量值作为第二模型中的各表达式的输入值进行计算,得到非线性表达式的第一解。

[0148] S206,判断第一解是否满足约束条件;若是,则执行S207;若否,则执行S209。

[0149] 该步骤中,判断是否满足约束条件的方法是:将第一解代入约束表达式对应的布尔表达式中求解布尔值,若布尔值为真,则确定第一解满足约束条件;若布尔值不为真,则确定第一解不满足约束条件。其中,约束表达式对应的布尔表达式指由约束表达式转换成的布尔表达式。

[0150] S207,进一步判断第一解是否为最优解;若是,则执行S208;若否,则执行S209。

[0151] S208,输出非线性表达式的最优解。

[0152] S209,判断是否达到迭代次数;若是,则执行S210;若否,则返回S204,重新生成一组新的输出变量值进行迭代。

[0153] S210,输出非线性表达式的解为无解。

[0154] 由此可见,本实施例的技术方案在求解非线性表达式时,无需对非线性表达式求导,只需利用Groovy语言执行非线性表达式对应的第二模型即可,因此相较于现有技术中

仅能通过对非线性表达式求导的方式而言,该技术方案能够有效求解出分段函数的非线性规划问题。

[0155] 综上,已经对本主题的特定实施例进行了描述。其它实施例在所附权利要求书的范围内。在一些情况下,在权利要求书中记载的动作可以按照不同的顺序来执行并且仍然可以实现期望的结果。另外,在附图中描绘的过程不一定要求示出的特定顺序或者连续顺序,以实现期望的结果。在某些实施方式中,多任务处理和并行处理可以是有利的。

[0156] 以上为本说明书一个或多个实施例提供的非线性规划问题的处理方法,基于同样的思路,本说明书一个或多个实施例还提供一种非线性规划问题的处理装置。

[0157] 图3是根据本说明书一实施例的一种非线性规划问题的处理装置的示意性框图,如图3所示,非线性规划问题的处理装置300包括:

[0158] 获取模块310,用于获取待求解的非线性表达式对应的第一模型,第一模型为利用预设方式描述非线性表达式的模型;非线性表达式包含约束条件;

[0159] 转换模块320,用于将第一模型转换为指定语言能够识别的第二模型;

[0160] 执行模块330,用于利用指定语言执行第二模型,得到非线性表达式对应的执行结果;及,输出执行结果。

[0161] 在一个实施例中,获取模块310包括:

[0162] 获取单元,用于获取非线性表达式;

[0163] 描述单元,用于利用预设方式描述非线性表达式,得到非线性表达式对应的第一模型;其中,第一模型包含输入表达式、输出表达式、演绎表达式、约束表达式、目标表达式中的至少一项表达式。

[0164] 在一个实施例中,转换模块320包括以下至少一项:

[0165] 添加单元,用于在第一模型中的输入表达式和/或演绎表达式的结尾位置添加预设符号;其中,预设符号为指定语言能够识别的符号;

[0166] 转换及赋值单元,用于将第一模型中的约束表达式转换为布尔表达式;及,将布尔表达式赋值给约束表达式对应的赋值变量;

[0167] 转换单元,用于将第一模型中的第一参数转换为指定语言能够识别的第二参数。

[0168] 在一个实施例中,转换及赋值单元还用于:

[0169] 计算约束表达式对应的哈希值;

[0170] 根据哈希值和/或预设参数,生成约束表达式对应的赋值变量;

[0171] 将布尔表达式赋值给生成的赋值变量。

[0172] 在一个实施例中,执行模块330包括:

[0173] 迭代单元,用于多次迭代以下步骤,直至获得非线性表达式的最优解和/或达到预设迭代次数:

[0174] 生成一组输出变量值;

[0175] 将输出变量值作为第二模型中的各表达式的输入值进行计算,得到非线性表达式的第一解;

[0176] 判断第一解是否满足约束条件;

[0177] 若满足约束条件,则进一步判断第一解是否为最优解;若为最优解,则输出最优解;

[0178] 若不满足约束条件和/或不为最优解,则重新生成一组输出变量值。

[0179] 在一个实施例中,迭代单元还用于:

[0180] 根据第一解确定布尔表达式的布尔值;及,判断布尔值是否为真;

[0181] 若是,则确定第一解满足约束条件;若否,则确定第一解不满足约束条件。

[0182] 在一个实施例中,执行结果为最优解或无解;

[0183] 相应的,装置300还包括:

[0184] 更新模块,用于若执行结果为无解,则按照预设规则更新目标表达式;其中,更新包括降低目标表达式对应的目标值。

[0185] 采用本说明书一个或多个实施例的装置,通过获取待求解的非线性表达式对应的第一模型,该第一模型为利用预设方式描述非线性表达式的模型,并将第一模型转换为指定语言能够识别的第二模型,进而利用指定语言执行第二模型,得到非线性表达式对应的执行结果,并输出该执行结果。可见,该技术方案在求解非线性表达式时,无需对非线性表达式求导,只需利用指定语言执行非线性表达式对应的第二模型即可,因此相较于现有技术中仅能通过对非线性表达式求导的方式而言,该技术方案能够有效求解出分段函数的非线性规划问题。

[0186] 本领域的技术人员应可理解,上述非线性规划问题的处理装置能够用来实现前文所述的非线性规划问题的处理方法,其中的细节描述应与前文方法部分描述类似,为避免繁琐,此处不另赘述。

[0187] 基于同样的思路,本说明书一个或多个实施例还提供一种非线性规划问题的处理设备,如图4所示。非线性规划问题的处理设备可因配置或性能不同而产生比较大的差异,可以包括一个或一个以上的处理器401和存储器402,存储器402中可以存储有一个或一个以上存储应用程序或数据。其中,存储器402可以是短暂存储或持久存储。存储在存储器402的应用程序可以包括一个或一个以上模块(图示未示出),每个模块可以包括对非线性规划问题的处理设备中的一系列计算机可执行指令。更进一步地,处理器401可以设置为与存储器402通信,在非线形规划问题的处理设备上执行存储器402中的一系列计算机可执行指令。非线性规划问题的处理设备还可以包括一个或一个以上电源403,一个或一个以上有线或无线网络接口404,一个或一个以上输入输出接口405,一个或一个以上键盘406。

[0188] 具体在本实施例中,非线性规划问题的处理设备包括有存储器,以及一个或一个以上的程序,其中一个或者一个以上程序存储于存储器中,且一个或者一个以上程序可以包括一个或一个以上模块,且每个模块可以包括对非线性规划问题的处理设备中的一系列计算机可执行指令,且经配置以由一个或者一个以上处理器执行该一个或者一个以上程序包含用于进行以下计算机可执行指令:

[0189] 获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;

[0190] 将所述第一模型转换为指定语言能够识别的第二模型;

[0191] 利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;及,输出所述执行结果。

[0192] 可选地,计算机可执行指令在被执行时,还可以使所述处理器:

[0193] 获取所述非线性表达式;

[0194] 利用所述预设方式描述所述非线性表达式,得到所述非线性表达式对应的第一模型;其中,所述第一模型包含输入表达式、输出表达式、演绎表达式、约束表达式、目标表达式中的至少一项表达式。

[0195] 可选地,计算机可执行指令在被执行时,还可以使所述处理器执行以下至少一项:

[0196] 在所述第一模型中的所述输入表达式和/或所述演绎表达式的结尾位置添加预设符号;其中,所述预设符号为所述指定语言能够识别的符号;

[0197] 将所述第一模型中的所述约束表达式转换为布尔表达式;及,将所述布尔表达式赋值给所述约束表达式对应的赋值变量;

[0198] 将所述第一模型中的第一参数转换为所述指定语言能够识别的第二参数。

[0199] 可选地,计算机可执行指令在被执行时,还可以使所述处理器:

[0200] 计算所述约束表达式对应的哈希值;

[0201] 根据所述哈希值和/或预设参数,生成所述约束表达式对应的赋值变量;

[0202] 将所述布尔表达式赋值给所述生成的赋值变量。

[0203] 可选地,计算机可执行指令在被执行时,还可以使所述处理器:

[0204] 多次迭代以下步骤,直至获得所述非线性表达式的最优解和/或达到预设迭代次数:

[0205] 生成一组输出变量值;

[0206] 将所述输出变量值作为所述第二模型中的各所述表达式的输入值进行计算,得到所述非线性表达式的第一解;

[0207] 判断所述第一解是否满足所述约束条件;

[0208] 若满足所述约束条件,则进一步判断所述第一解是否为所述最优解;若为所述最优解,则输出所述最优解;

[0209] 若不满足所述约束条件和/或不作为所述最优解,则重新生成一组输出变量值。

[0210] 可选地,计算机可执行指令在被执行时,还可以使所述处理器:

[0211] 根据所述第一解确定所述布尔表达式的布尔值;及,判断所述布尔值是否为真;

[0212] 若是,则确定所述第一解满足所述约束条件;若否,则确定所述第一解不满足所述约束条件。

[0213] 可选地,所述执行结果为最优解或无解;

[0214] 相应的,计算机可执行指令在被执行时,还可以使所述处理器:

[0215] 若所述执行结果为无解,则按照预设规则更新所述目标表达式;其中,所述更新包括降低所述目标表达式对应的目标值。

[0216] 本说明书一个或多个实施例还提出了一种计算机可读存储介质,该计算机可读存储介质存储一个或多个程序,该一个或多个程序包括指令,该指令当被包括多个应用程序的电子设备执行时,能够使该电子设备执行上述非线性规划问题的处理方法,并具体用于执行:

[0217] 获取待求解的非线性表达式对应的第一模型,所述第一模型为利用预设方式描述所述非线性表达式的模型;所述非线性表达式包含约束条件;

[0218] 将所述第一模型转换为指定语言能够识别的第二模型;

[0219] 利用所述指定语言执行所述第二模型,得到所述非线性表达式对应的执行结果;

及,输出所述执行结果。

[0220] 上述实施例阐明的系统、装置、模块或单元,具体可以由计算机芯片或实体实现,或者由具有某种功能的产品来实现。一种典型的实现设备为计算机。具体的,计算机例如可以为个人计算机、膝上型计算机、蜂窝电话、相机电话、智能电话、个人数字助理、媒体播放器、导航设备、电子邮件设备、游戏控制台、平板计算机、可穿戴设备或者这些设备中的任何设备的组合。

[0221] 为了描述的方便,描述以上装置时以功能分为各种单元分别描述。当然,在实施本说明书一个或多个实施例时可以把各单元的功能在同一个或多个软件和/或硬件中实现。

[0222] 本领域内的技术人员应明白,本说明书一个或多个实施例可提供为方法、系统、或计算机程序产品。因此,本说明书一个或多个实施例可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且,本说明书一个或多个实施例可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。

[0223] 本说明书一个或多个实施例是参照根据本申请实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器,使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0224] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中,使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品,该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0225] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上,使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0226] 在一个典型的配置中,计算设备包括一个或多个处理器(CPU)、输入/输出接口、网络接口和内存。

[0227] 内存可能包括计算机可读介质中的非永久性存储器,随机存取存储器(RAM)和/或非易失性内存等形式,如只读存储器(ROM)或闪存(flash RAM)。内存是计算机可读介质的示例。

[0228] 计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括,但不限于相变内存(PRAM)、静态随机存取存储器(SRAM)、动态随机存取存储器(DRAM)、其他类型的随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器(CD-ROM)、数字多功能光盘(DVD)或其他光学存储、磁盒式磁带,磁带磁磁盘存储或其他磁性存储设备

或任何其他非传输介质,可用于存储可以被计算设备访问的信息。按照本文中的界定,计算机可读介质不包括暂存电脑可读媒体(transitory media),如调制的数据信号和载波。

[0229] 还需要说明的是,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、商品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

[0230] 本说明书一个或多个实施例可以在由计算机执行的计算机可执行指令的一般上下文中描述,例如程序模块。一般地,程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本申请,在这些分布式计算环境中,由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中,程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。

[0231] 本说明书中的各个实施例均采用递进的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于系统实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。

[0232] 以上所述仅为本说明书一个或多个实施例而已,并不用于限制本说明书。对于本领域技术人员来说,本说明书一个或多个实施例可以有各种更改和变化。凡在本说明书一个或多个实施例的精神和原理之内所作的任何修改、等同替换、改进等,均应包含在本说明书一个或多个实施例的权利要求范围之内。

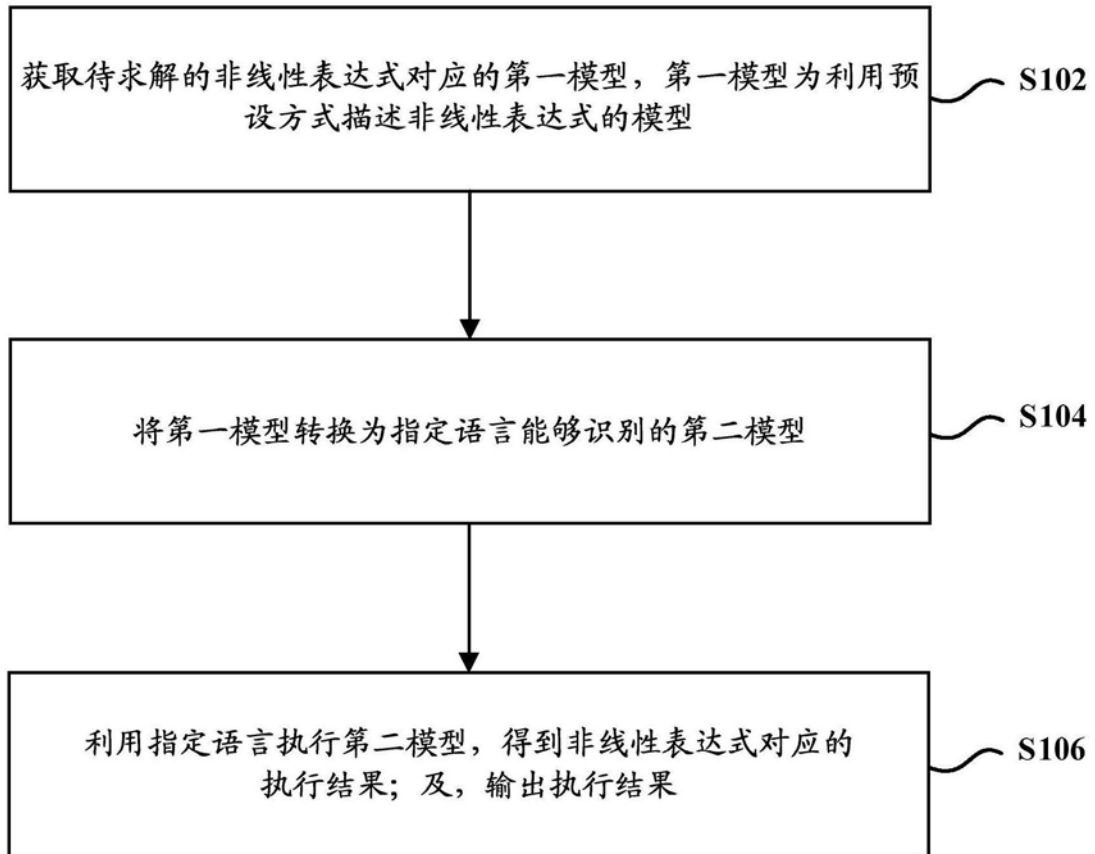


图1

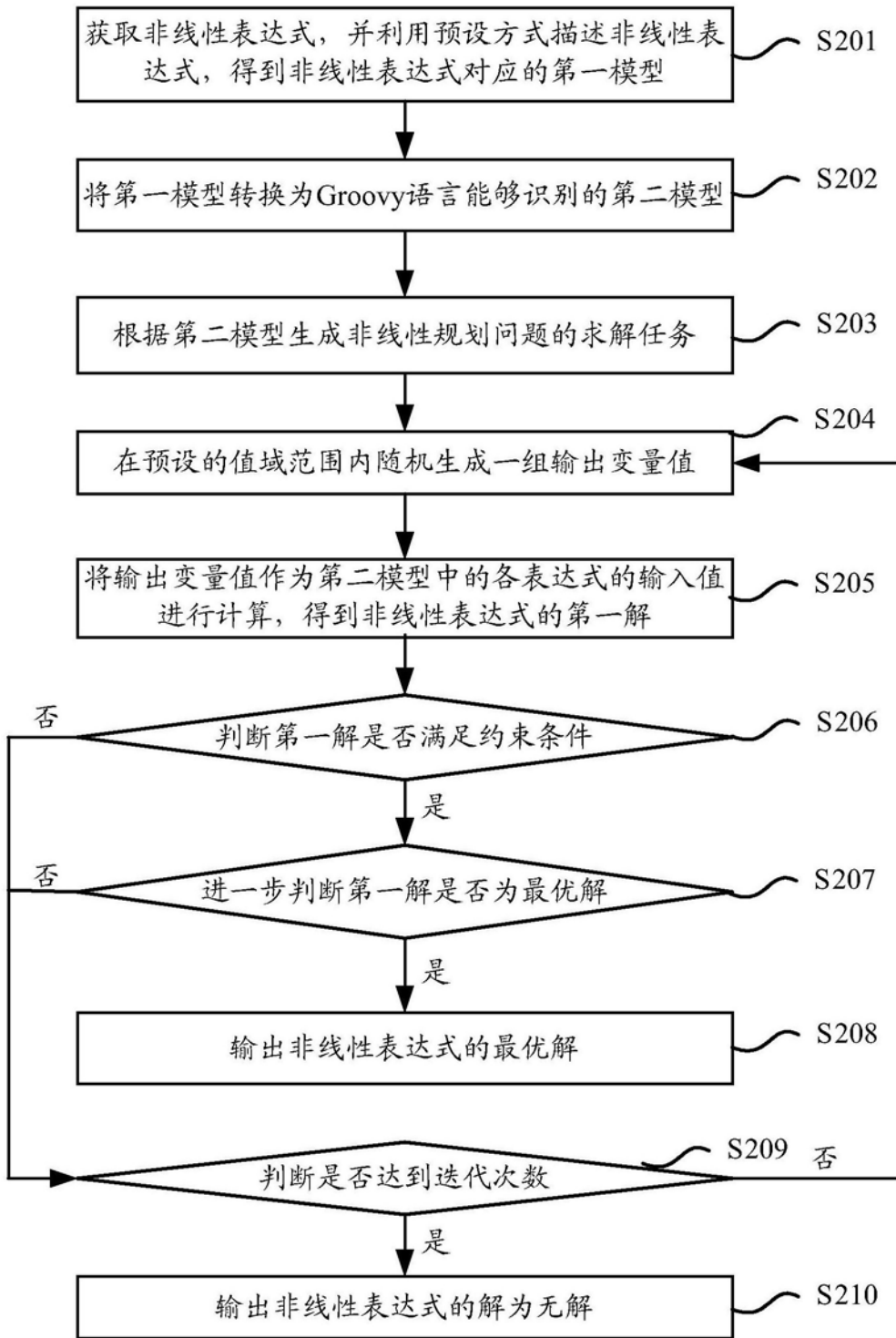


图2

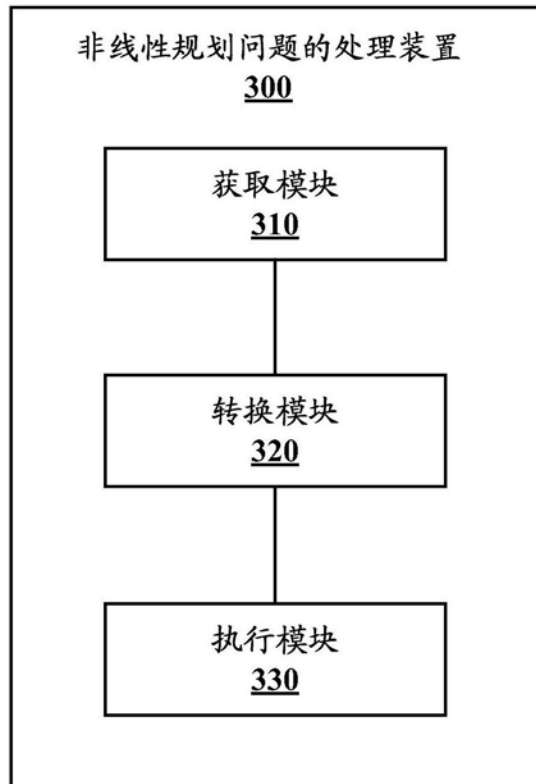


图3

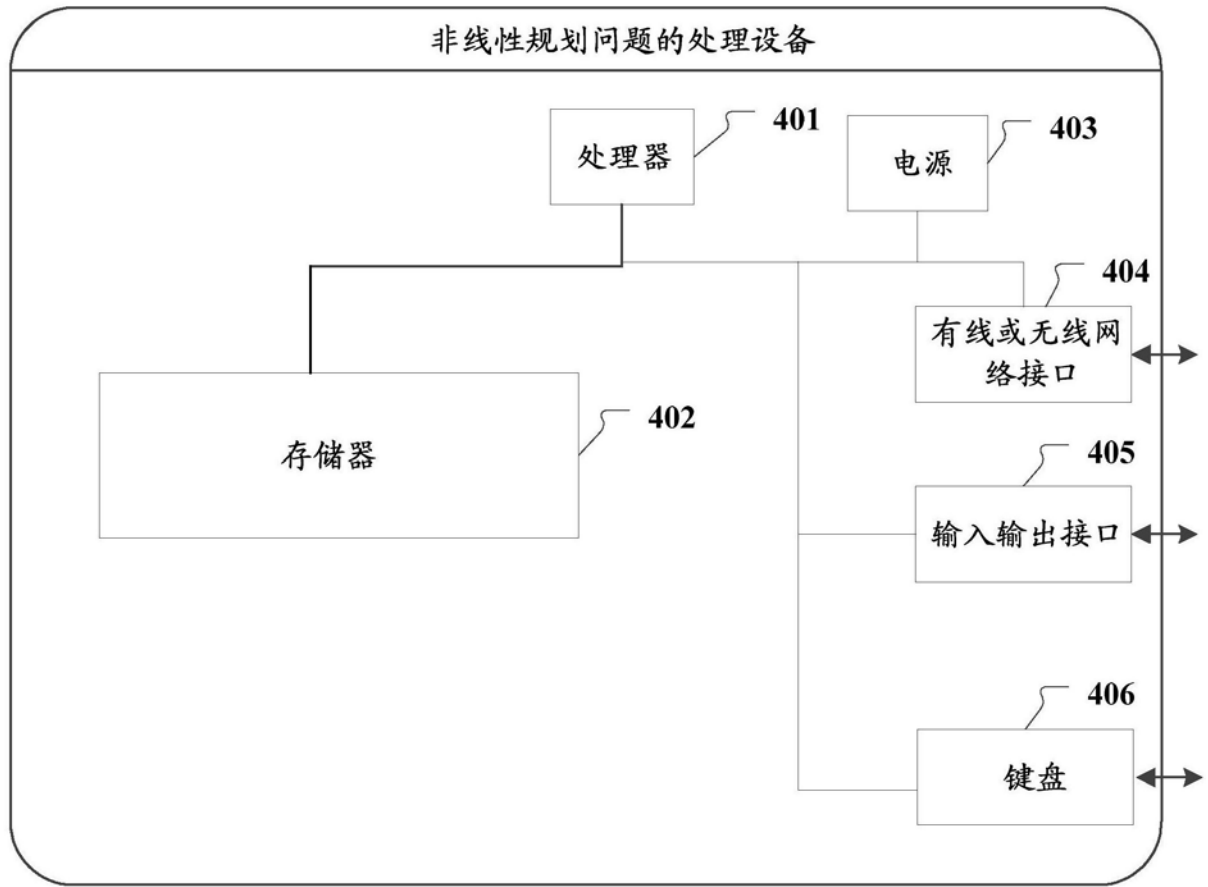


图4