

说明书

一种销账方法、存储介质及计算机系统

技术领域

本说明书涉及计算机技术领域，尤其涉及一种销账方法、存储介质及计算机系统。

背景技术

随着互联网技术的快速发展，线上支付和在线交易得到了快速的发展和广泛的应用。用户通过业务平台执行在线交易时，业务平台和业务执行机构（如银行）会分别对在线交易的交易信息进行记录。为了保证在线交易的安全性和稳定性，通常会基于业务平台记录的交易信息和业务执行机构记录的交易信息进行核对。如果业务平台记录的交易信息和业务执行机构记录的交易信息存在差异，则说明业务平台记录的交易信息中存在应收或应付的资金，与实际交易的资金流动不一致的情况。通常，会将交易信息存在差异的金额暂时进行挂账，由操作人员对多笔挂账统一进行核销处理后实现销账。

但是，随着互联网线上支付技术的广泛应用，线上支付和在线交易的规模越来越大，其对应的资金清算数据呈海量级增长的趋势，从而在资金对账过程中产生了大量的挂账。目前通常采用人工销账的方式对挂账进行核销，但这对操作人员的专业能力要求较高，且工作量较大，降低了销账的执行效率。

发明内容

本说明书实施例提供一种销账方法、存储介质及计算机系统，以部分解决上述现有技术存在的问题。

本说明书提供了一种销账方法，所述方法包括：获取各挂账凭证；所述挂账凭证是业务平台记录的交易金额与业务执行机构记录的交易金额核对不匹配时生成的；根据所述各挂账凭证，确定各挂账收入金额和各挂账支出金额；从所述各挂账收入金额和所述各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额，并根据相互匹配的挂账收入金额和挂账支出金额，确定待

销账金额组；根据所述待销账金额组对应的各挂账凭证，生成销账指令；响应于所述销账指令，对所述销账指令对应的各挂账凭证进行核销。

本说明书提供了一种计算机系统，包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序，所述处理器执行所述程序时实现如下步骤：获取各挂账凭证；所述挂账凭证是业务平台记录的交易金额与业务执行机构记录的交易金额核对不匹配时生成的；根据所述各挂账凭证，确定各挂账收入金额和各挂账支出金额；从所述各挂账收入金额和所述各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额，并根据相互匹配的挂账收入金额和挂账支出金额，确定待销账金额组；根据所述待销账金额组对应的各挂账凭证，生成销账指令；响应于所述销账指令，对所述销账指令对应的各挂账凭证进行核销。

本说明书实施例采用的上述至少一个技术方案能够达到以下有益效果中的至少一项：根据业务平台和业务执行机构分别记录的交易金额核对时生成的各挂账凭证，确定各挂账金额和各挂账支出金额，并从中确定相互匹配的挂账收入金额与挂账支出金额，从而得到待销账金额组，进而基于待销账金额组对应的各挂账凭证生成销账指令，以便基于销账指令对挂账凭证进行核销。通过自动化的金额匹配方案减少了销账的人工成本，并提升了销账的效率，从而降低了存在的风险金额。

附图说明

此处所说明的附图用来提供对本说明书的进一步理解，构成本说明书的一部分，本说明书的示意性实施例及其说明用于解释本说明书，并不构成对本说明书的不当限定。在附图中：

图 1 为本说明书中一种销账方法的流程示意图；

图 2 为本说明书中一种对账流程的示意图；

图 3 为本说明书中一种销账方法的流程示意图；

图 4 为本说明书中一种销账方法的流程示意图；

图 5 为本说明书中一种销账方法的流程示意图；

图 6 为本说明书中一种销账方法的流程示意图；

图 7 为本说明书实施例提供的计算机系统的结构示意图。

具体实施方式

为使本说明书的目的、技术方案和优点更加清楚，下面将结合本说明书具体实施例及相应的附图对本说明书技术方案进行清楚、完整地描述。显然，所描述的实施例仅是本说明书一部分实施例，而不是全部的实施例。基于本说明书中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本说明书保护的范围。

另外，需要说明的是，本说明书中所有获取信号、信息或数据的动作都是在遵照所在地相应的数据保护法规政策的前提下，并获得由相应装置所有者给予授权的情况下进行的。

需要说明的是，在不冲突的情况下，下述的实施例及实施方式中的特征可以相互组合。

在实际应用中，提供支付、资金管理等服务的业务平台，需要根据自身的账务系统内各类账户的资金与其在银行的存款资金一一对应，从而保证真实的资金账户与业务平台的虚拟账户的资金转换正确。

而在实际执行对账的过程中，由于国际业务渠道的多样性、机构日切、交易打包合并规则、机构账单解析（单笔/汇总）规则的差异等原因，业务平台与业务执行机构进行对账时，常常会出现两边记录的交易信息不一致的问题，尤其是交易金额可能会存在差异。

以业务平台和业务执行机构的交易打包合并规则不同为例，假设业务平台是按照 7 天的时间周期进行打包，业务执行机构是按照每周的工作日（5 天）进行打包，则在业务平台和业务执行机构进行对账时，就会因打包周期不一致，出现双方交易金额校对不一致的问题，这样就会出现挂账的问题，需要统一进行核销。

目前，在以国际业务为主的资金清算领域中，对于对账产生的挂账的核销主要依赖操作人员手动核销。但是，随着在线交易的规模越来越大，产生的挂账也随之增多。人工手动销账不仅要求操作人员业务水平较高，还承担了巨大的工作量，极有可能存在核销错误的问题，降低了资金安全性以及核销效率。

基于此，本说明书提供一种销账方法，通过自动化的金额匹配方案减少了销账的人工成本，并提升了销账的效率，从而降低了存在的风险金额。

以下结合附图，详细说明本说明书各实施例提供的技术方案。

图 1 为本说明书提供的一种销账方法的流程示意图。

S100：获取各挂账凭证；所述挂账凭证是业务平台记录的交易金额与业务执行机构记录的交易金额核对不匹配时生成的。

本说明书实施例中提供的销账方法的执行过程可由用于对挂账进行销账的服务器等电子设备执行。执行该方法的电子设备可以是单个服务器，也可以是由多个服务器组成的服务器集群，本说明书对此不做限定。

实际应用中，为了保证在线上支付交易中，业务平台在业务执行机构开设的备付金账户和业务执行机构的账户，两方的每日账户余额以及每日发生额核对一致，以及进一步保证备付金账户资金明确且资金账务健康，业务平台需及时和业务执行机构进行资金三层对账核对，以确保业务平台记录的交易信息中存在应收或应付的资金，与实际交易的资金核对一致。

如图 2 所示为业务平台和业务执行机构进行资金三层对账的基本架构，其中，业务平台会在业务执行机构开设账户，业务执行机构可记录该账户的资金变动并形成账户流水，业务平台则可以通过记录交易数据的方式获得交易流水，从而在第一层明细对账中，实际上是业务平台记录的交易流水与业务执行机构记录的账户流水之间的核对。进一步的，在明细对账之后，业务平台可基于业务平台记录的账单，与业务执行机构记录的账单进行核对，这是第二层账单对账。最后，业务平台可基于自身记录的金额变动，与账户的历史余额确定自身核对得到的当前余额，然后与业务执行机构处账户的当前余额进行对账，即可完成第三层的余额对账。

执行上述过程有利于确保业务平台内部记录与业务执行的记录保持一致，及时发现并纠正转账错误、重复记账、遗漏交易等错误，同时也能够有效识别可能的欺诈行为，有助于提升财务管理的规范化水平。

但是，当业务平台记录的交易金额信息和业务执行机构记录的交易金额信息存在差异时，会将差异金额部分暂时进行挂账，对应生成挂账凭证。

挂账凭证主要是围绕资金三层对账中的资金账单对账产生的。对账的双方

是业务平台与业务执行机构。其中，业务平台可以是为用户提供在线交易服务的第三方支付平台，也可以是第三方支付平台的资金清算对账后端业务平台。业务执行机构可以是为用户提供资金存储、管理、支付结算等金融业务的机构，如银行、信用社等。

在资金对账过程中，业务平台的资金清算平台和业务执行机构之间交易打包合并规则不一致等原因，对于这类场景的资金核对，会对同日期下两方金额校对不一致进行挂账，其生成的挂账凭证金额为其差额。挂账凭证中包括挂账凭证的标识、挂账时间、挂账业务日期、挂账金额、挂账摘要等信息。其中，挂账时间一般是在两方核对时，执行挂账操作，生成挂账凭证的时间。挂账业务日期一般是以业务执行机构的账单日为准，账单日即为业务平台在业务执行机构开设的账户余额发生变动的日期。一般情况下，由于挂账数量规模较大，且人工销账流程复杂，生成的挂账凭证会长期滞留，短期内无法做到精准快速销账，因此，在本说明书中，获取的挂账凭证可以是历史累积的所有挂账凭证，也可能是预设的挂账时间范围内的各挂账凭证，本说明书对此不做限定。

另外，业务平台可对接多个业务执行机构分别进行对账，具体的，业务平台在资金对账流程能获取到自身交易账单数据，并和获取到的业务执行机构的机构账单数据进行资金核对。若存在交易金额的差异，则确定业务平台记录的交易金额与该业务执行机构记录的交易金额核对不一致，并根据不一致的交易金额生成挂账凭证。

S102：根据所述各挂账凭证，确定各挂账收入金额和各挂账支出金额。

具体的，如果业务平台记录的交易金额大于业务执行机构的账单记录的交易金额，则业务平台一侧会生成待确认收入的挂账凭证，用于指示流入业务平台的金额比流出业务执行机构的金额多，业务平台可能存在多收资金或未付资金等问题。如果业务平台记录的交易金额小于业务执行机构记录的交易，则业务平台一侧会生成待确认支出的挂账凭证，用于指示流出业务平台的金额比流入业务执行机构的金额多，业务平台可能存在多付资金等问题。

因此，通过解析各挂账凭证对应的挂账金额，即可确定出各挂账收入金额和各挂账支出金额，一般的，业务平台一侧生成的挂账凭证中，待确认收入的挂账凭证对应的挂账金额为正数，也即挂账收入金额。待确认支出的挂账凭证

对应的挂账金额为负数，也即挂账支出金额。

S104：从所述各挂账收入金额和所述各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额，并根据相互匹配的挂账收入金额和挂账支出金额，确定待销账金额组。

具体的，要对存在交易金额不匹配的挂账进行核销，需要查找到其他挂账的挂账金额是否与该挂账的挂账金额匹配。本说明书中，相互匹配的多个挂账金额实际上可以是多个挂账金额的总和为 0。

例如，业务平台的 X 日账单与业务执行机构的 X 日账单进行对账时，存在多流入业务平台 100 元的差异，此时，业务平台会生成 X 日的待确认收入的挂账凭证，挂账金额为 100 元，也即挂账收入金额为 100 元。业务平台 Y 日账单与业务执行机构的 Y 日账单进行对账时，存在多流出业务平台 100 元的差异，此时，业务平台会生成 Y 日的待确认支出的挂账凭证，挂账金额为-100 元，也即挂账支出金额为-100 元。由于 X 日的挂账凭证对应的挂账金额与 Y 日的挂账凭证对应的挂账金额相加和为 0，因此，确定 X 日的挂账凭证对应的挂账收入金额与 Y 日的挂账凭证对应的挂账支出金额相互匹配。

由此可见，可基于各挂账凭证对应的挂账金额，从其他挂账凭证中查找是否存在挂账金额与该挂账凭证对应的挂账金额相加和为 0。另外，本说明书对相互匹配的挂账收入金额的数量，以及相互匹配的挂账支出金额的数量不做限定，可以是一个挂账收入金额与一个挂账支出金额相互匹配，也可以是一个挂账收入金额与多个挂账支出金额相互匹配，或是多个挂账收入金额与一个挂账支出金额相互匹配，当然，也可以是多个挂账收入金额与多个挂账支出金额相互匹配。

进一步的，将相互匹配的挂账收入金额和挂账支出金额放入同一个待销账金额组中。本说明书对一个待销账金额组中包含的挂账收入金额的数量和挂账支出金额的数量不做限定。

在进行销账金额匹配时，可以采用双指针或多指针（三指针及三指针以上）的方式，从排序后的各挂账收入金额和各挂账支出金额中查找相互匹配的挂账收入金额和挂账支出金额。

例如，各挂账凭证对应的销账金额经过排序后可以为[-2,-1,-0.5,1,2,3.5]。采

用双指针的方式可以确定相互匹配的一个挂账收入金额和一个挂账支出金额，可得 $[-2,2]$ 这一待销账金额组。如果采用四指针的方式，则待销账金额组中包含有四个挂账金额，如 $[-2,-1,-0.5,3.5]$ 。

可选地，采用指针的方式确定相互匹配的挂账收入金额和挂账支出金额可能存在计算时间过长的的问题。如果服务器的内存空间足够，为了进一步提升销账效率，减少匹配时间，还可以采用键值对查询的方式确定相互匹配的挂账收入金额和挂账支出金额，具体实施方式如下：

首先，以键值对的方式存储所述各挂账凭证的标识和所述各挂账凭证对应的挂账金额，其中，所述挂账凭证的标识为键，所述挂账凭证对应的挂账金额为值。

之后，针对每个挂账凭证，根据该挂账凭证的挂账金额，从所述键值对中查找与该挂账凭证的挂账金额匹配的至少一个其他挂账凭证的挂账金额。

S106：根据所述待销账金额组对应的各挂账凭证，生成销账指令。

具体的，针对生成的每个销账金额组，根据该销账金额组中的一个或多个挂账收入金额对应的挂账凭证，与该销账金额组中的一个或多个挂账支出金额对应的挂账凭证，生成销账指令。一般的，一个销账金额组对应于生成一个销账指令。

S108：响应于所述销账指令，对所述销账指令对应的各挂账凭证进行核销。

进一步的，服务器可基于销账指令中携带的各挂账凭证进行核销，并生成销账凭证，销账凭证中包括核销处理的各挂账凭证的标识，核销处理的各挂账凭证对应的挂账金额，挂账凭证对应的业务执行机构的标识等信息。

其中，销账指令可以离线生成，之后统一流回到线上交由执行销账指令的服务器进行销账，销账指令也可以是在线生成的，一旦销账指令生成，就触发服务器响应销账指令，本说明书对此不做限定。

在第三方支付机构的账务层面，要保证财务健康，有挂账必然对应应有销账。例如，假设有：挂账收入凭证 A_1 ， A_1 对应的挂账收入金额是+20，挂账收入凭证 A_2 ， A_2 对应的挂账收入金额是+30，挂账支出凭证 B_1 ， B_1 对应的挂账支出金额是-50，通过本说明书提供的销账方法， $20+30-50=0$ ，正好这几个凭证可以构成待销账金额组，那么基于该待销账金额组就可以生成一条销账指令 S ， S

包含 (A_1, A_2, B_1) 。执行业务平台销账指令 S 时，挂账凭证 A_1+A_2 对应会生成一个销账凭证 C_1 ，挂账凭证 B_1 对应有一个销账凭证 C_2 ，这样满足会计规则的“有挂必有销”。

在图1所示本说明书提供的销账方法中，根据业务平台和业务执行机构分别记录的交易金额核对时生成的各挂账凭证，确定各挂账金额和各挂账支出金额，并从中确定相互匹配的挂账收入金额与挂账支出金额，从而得到待销账金额组，进而基于待销账金额组对应的各挂账凭证生成销账指令，以便基于销账指令对挂账凭证进行核销。

可见，通过上述自动化的金额匹配方案减少了销账的人工成本，并提升了销账的效率，从而降低了存在的风险金额。

在本说明书一个或多个实施例中，图1所示步骤S104中，可以通过限制相互匹配的挂账收入金额的数量和挂账支出金额的数量，避免统计过多挂账收入金额和过多挂账支出金额是否匹配，降低计算复杂度，从而提高金额匹配的效率，具体实施方式如下，如图3所示：

S200：获取预先优化的数量阈值。

S202：从所述各挂账收入金额和所述各挂账支出金额中，确定第一目标数量的挂账收入金额与第二目标数量的挂账支出金额匹配。其中，所述第一目标数量与所述第二目标数量之和不大于所述预先优化的数量阈值。

具体的，在本说明书中，相互匹配的挂账收入金额的数量和挂账支出金额的数量均可以为一个或多个。因此，可以将挂账收入金额与挂账支出金额的匹配抽象为多数之和（ $nsum$ ）的确定问题，其中多数之和的目标结果为0。

由于多数之和（ $nsum$ ）的确定问题中， n 的具体取值会影响计算的时间复杂度，从而增加从各挂账收入金额和各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额的匹配时长。

由此，为了平衡匹配效率（也即匹配时长）以及销账效率（一笔销账指令包含的相互匹配的挂账收入金额和挂账支出金额的数量），可以获取预先优化的数量阈值，以预先优化的数量阈值为上限，限制相互匹配的挂账收入金额的数量和挂账支出金额的数量综合不超过预先优化的数量阈值，从而同时保证销账效率以及匹配效率。

进一步的，以相互匹配的挂账收入金额和挂账支出金额中，挂账收入金额的数量为第一目标数量，挂账支出金额的数量为第二目标数量，限制第一目标数量与第二目标数量之和不大于预先优化的数量阈值。这样，在从各挂账收入金额和各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额时，计算复杂度不会太高，从而提升匹配的效率，缩短匹配的时长。

其中，预先优化的数量阈值可以是基于专家经验得到，可以是预先基于多个人工设置的数量阈值分别进行测试后得到的最优的数量阈值作为预先优化的数量阈值，还可以是基于匹配时长经多轮迭代优化得到。

在本说明书一个可选的实施例中，S200 中获取到的预先优化的数量阈值可通过下述实施方式迭代优化得到，如图 4 所示：

S300：确定当前数量阈值。

具体的，在迭代优化数量阈值时，可在第一轮优化时初始化当前数量阈值，初始化的方式可以是基于经验的设置，也可以是随机设置，本说明书对此不作限定。

当然，在第二轮优化以及第二轮之后的若干轮优化中，当前数量阈值可以是根据上一轮优化过程中得到的匹配时长，以及上一轮优化过程中采用的当前数量阈值确定。

S302：获取各第一参考挂账凭证，并根据所述各第一参考挂账凭证，确定各参考收入金额和各参考支出金额。

之后，获取用于优化数量阈值的各第一参考挂账凭证。第一参考挂账凭证可以是业务平台与业务执行结构进行对账时，双方记录的交易金额存在差异，而产生的挂账的凭证。当然，第一参考挂账凭证可以是历史上实际对账产生的挂账凭证，也可以是模拟对账产生的挂账凭证，本说明书对此不作限定。

本说明书在优化数量阈值的过程中采用的第一参考挂账凭证与 S100 所述的挂账凭证类似，此处不再赘述。另外，在每一轮迭代优化过程中所使用的第一参考挂账凭证可以是相同的也可以是不同，本说明书对此同样不作限定。

S304：从所述各参考收入金额和所述各参考支出金额中，确定第一参考数量的参考收入金额和第二参考数量的参考支出金额匹配，并记录当前匹配时长。其中，所述第一参考数量和所述第二参考数量之和与所述当前数量阈值相同。

在此步骤中，在从各参考收入金额和各参考支出金额中确定相互匹配的参考金额时，以当前数量阈值为限制，即从各参考收入金额中选中的参考收入金额的第一目标数量，与从各参考支出金额中选中的参考支出金额的第二目标数量之和不大于当前数量阈值。

从而，通过匹配过程，可以记录从开始匹配，到得到相互匹配的参考收入金额和参考支出金额为止的当前匹配时长。

S306：根据所述当前匹配时长和预设的匹配时长阈值之间的差异，重新确定当前数量阈值，并根据重新确定的当前数量阈值，重新匹配参考收入金额和参考支出金额，直至所述当前匹配时长满足预设条件。

如果当前匹配时长大于预设的匹配时长阈值，则说明当前匹配时长较长，可能是因为当前数量阈值设置的过高，提升了匹配的时间复杂度，从而导致的当前匹配时长较长。此时，可以通过以当前数量阈值为基准，调低当前数量阈值，得到重新确定的当前数量阈值，以降低匹配的时间复杂度。

如果当前匹配时长小于预设的匹配时长阈值，则说明当前匹配时长较短，可能是因为当前数量阈值设置的过低，匹配的时间复杂度较小。此时，可以通过以当前数量阈值为基准，调低当前数量阈值，得到重新确定的当前数量阈值，以提升匹配的时间复杂度。但同时，由于提升了当前数量阈值，则在下一轮匹配过程中，会有更多的挂账可以进行核销，实际上提升了核销的效率。

例如，当前数量阈值为 m ，基于当前数量阈值 m 进行匹配，记录的当前匹配时长是 t_1 。预设的匹配时长阈值是 T 。如果 t_1 大于 T ，则下一轮采用的当前数量阈值应当小于 m ，因此，重新确定当前数量阈值为 n ，其中 $n < m$ 。如果 t_1 小于 T ，则下一轮采用的当前数量阈值应当大于 m ，因此，重新确定当前数量阈值为 k ，其中 $k > m$ 。

如果当前匹配时长等于预设的匹配时长阈值，说明当前匹配时长的长短较为合适，可以停止当前数量阈值的优化。

由此，本说明书中，停止优化所需当前匹配时长满足的预设条件可以是：当前匹配时长等于预设的匹配时长阈值，或者是迭代优化的次数超过预设的迭代次数阈值。

另外，本说明书中，预设的匹配时长阈值可以是预先基于经验得到的，或

者是人工设置的，本说明书对此不做限定。

在本说明书一个或多个实施例中，图 1 所示步骤 S104 中，可以通过限制相互匹配的挂账收入金额和挂账支出金额对应的挂账凭证所在的挂账时间周期的长度，避免统计大量的挂账收入金额和大量的挂账支出金额是否匹配，从而提高金额匹配的效率，具体实施方式如下，如图 5 所示：

S400：获取预先优化的挂账时间周期。

与图 3 所示基于预先优化的数量阈值提升金额匹配的效率的方案类似。实际应用中，如果需要进行匹配的挂账收入金额和挂账支出金额过多，同样会消耗较多的匹配时长。为了平和匹配效率和销账效率，可以限制获取到的各挂账凭证都属于在预先优化的挂账时间周期内生成的，也即，获取到的各挂账凭证对应的挂账时间都落入预先优化的挂账时间周期。这样，获取到的挂账凭证的数量规模就会受到限制，从而提升匹配的效率。

S402：根据挂账凭证对应的挂账时间，获取挂账时间落入所述预先优化的挂账时间周期的各挂账凭证。

由此，在图 1 所示 S100 获取挂账凭证时，首先确定历史上生成的所有挂账凭证对应的挂账时间，将挂账时间落入预先优化的挂账时间周期的各挂账凭证，作为本说明书执营销账方法所获取的各挂账凭证。

例如，挂账凭证 A₂ 的挂账时间是 50 天前，挂账凭证 B₂ 的挂账时间是 100 天前，如果预先优化的挂账时间周期为 75 天，则挂账凭证 A₂ 落入预先优化的挂账时间周期，但挂账凭证 B₂ 并未落入预先优化的挂账时间周期，从而，基于预先优化的挂账时间周期获取到的挂账凭证即为挂账凭证 A₂。

同样的，与前述预先优化的数量阈值类似，该预先优化的挂账时间周期可以是基于专家经验得到，可以是预先基于多个人工设置的挂账时间周期分别进行测试后得到的最优的挂账时间周期作为预先优化的挂账时间周期，还可以是基于匹配时长经多轮迭代优化得到。

具体的，S400 中获取到的预先优化的挂账时间周期可通过下述实施方式迭代优化得到，如图 6 所示：

S500：确定当前挂账时间周期。

具体提的，在迭代优化挂账时间周期时，可在第一轮优化时优化当前挂账

时间周期，初始化的方式可以是基于经验的设置，也可以是随机设置，本说明书对此不作限定。

当然，在第二轮优化以及第二轮之后的若干轮优化中，当前挂账时间周期可以根据上一轮优化过程中得到的匹配时长，以及上一轮优化过程中采用的当前挂账时间周期确定。

S502：根据挂账凭证对应的挂账时间，获取挂账时间落入所述当前挂账时间周期的各第二参考挂账凭证。

具体的，当前挂账时间周期用于限制匹配所采用的挂账凭证的数量。因此，在此步骤中，获取的各第二参考挂账凭证分别对应的挂账时间实际上均是落入当前挂账时间周期的挂账凭证。

一般情况下，当前挂账时间周期越长，获取到的各第二参考挂账凭证的数量越多，从而匹配时间越长。反之，当前挂账时间周期越短，获取到的各第二参考挂账凭证的数量越少，从而匹配时间越短。

S504：根据所述各第二参考挂账凭证，确定各参考收入金额和各参考支出金额。

此步骤与前述 S302 类似，此处不做赘述。

S506：从所述各参考收入金额和所述各参考支出金额中，确定匹配的参考收入金额和匹配的参考支出金额，并记录当前匹配时长。

此步骤与前述 S304 类似，此处不做赘述。

S508：根据所述当前匹配时长和预设的匹配时长阈值之间的差异，重新确定当前挂账时间周期，并根据重新确定的当前挂账时间周期，重新匹配参考收入金额和参考支出金额，直至所述当前匹配时长满足预设条件。

如果当前匹配时间大于预设的匹配时长阈值，则说明当前匹配时长较长，可能是因为当前挂账时间周期设置的过长，第二参考挂账凭证的数量较多，提升了匹配的时间复杂度，从而导致的当前匹配时长较长。此时，可以通过以当前挂账时间周期为基准，调短当前挂账时间周期，得到重新确定的当前挂账时间周期，减少第二参考挂账凭证的数量，以降低匹配的时间复杂度。

如果当前匹配时长小于预设的匹配时长阈值，则说明当前匹配时长较短，可能是因为当前挂账时间周期设置的过短，第二参考挂账凭证的数量较少，匹

配的时间复杂度较小。此时，可以通过以当前挂账时间周期为基准，调长当前挂账时间周期，得到重新确定的当前挂账时间周期，增加第二参考挂账凭证的数量，以提升匹配的时间复杂度。

例如，当前挂账时间周期为 τ_1 ，基于当前挂账时间周期 m 进行匹配，记录的当前匹配时长是 t_2 。预设的匹配时长阈值是 T 。如果 t_2 大于 T ，则下一轮采用的当前挂账时间周期应当短于 τ_1 ，因此，重新确定当前挂账时间周期为 τ_2 ，其中 $\tau_2 < \tau_1$ 。如果 t_2 小于 T ，则下一轮采用的当前挂账时间周期应当大于 τ_1 ，因此，重新确定挂账时间周期为 τ_3 ，其中 $\tau_3 > \tau_1$ 。

如果当前匹配时长等于预设的匹配时长阈值，说明当前匹配时长的长短较为合适，可以停止当前挂账时间周期的优化。

由此，本说明书中，停止优化过程所需要当前匹配时长满足的预设条件可以是：当前匹配时长等于预设的匹配时长阈值，或者是迭代优化的次数超过预设的迭代次数阈值。

另外，本说明书中，预设的匹配时长阈值可以是预先基于经验得到的，或者是人工设置的，可以与前述优化数量阈值采用的匹配时长阈值相同也可以不同，本说明书对此不做限定。

在本说明书一个可选的实施例中，图 1 所示步骤 S106 中，可能会出现多个不同的待销账金额组分别包含一个或多个挂账收入金额的总和相同的情况，此时，可以确定这多个待销账金额组的销账优先级，按照销账优先级依次生成销账指令，具体实施方式如下：

第一步：如果存在多个待销账金额组包含的挂账收入金额相同，则针对每个待销账金额组，获取该待销账金额组对应的各挂账凭证分别对应的挂账时间。

一般，一个待销账金额组对应于生成一个销账指令。由于执行销账指令对挂账凭证进行核销需要耗费一定的处理时长，因此，不同的销账指令具有不同的优先级。一般的，销账指令对应的待销账金额组所对应的需要进行销账处理的挂账收入金额越高，销账指令的优先级越高。

但是，实际中，可能存在多个待销账金额组对应的需要进行销账处理的挂账收入金额的总数是相同的。此时，可以根据挂账时间确定挂账指令的优先级。

第二步：从该待销账金额组对应的各挂账凭证分别对应的挂账时间中选择

该待销账金额组对应的参考挂账时间。

由于待销账金额组对应的挂账凭证的数量最少有两个，因此，需要从待销账金额组对应的各挂账凭证分别对应的挂账时间中选择一个挂账时间，作为待销账金额组对应的参考挂账时间。

选择该待销账金额组对应的参考挂账时间的方式，可以是将该待销账金额组对应的各挂账凭证分别对应的挂账时间中时间最靠前的挂账时间作为参考挂账时间，也可以是随机选择，还可以是将时间最靠后的挂账时间作为参考挂账时间，本说明书对此不作限定。

第三步：根据所述多个待销账金额组分别对应的参考挂账时间，确定所述多个待销账金额对分别对应的销账优先级。

具体的，比较各个待销账金额组对应的参考挂账时间的先后关系，参考挂账时间越靠前，待销账金额组对应的销账优先级越高。参考挂账时间越靠后，待销账金额组对应的销账优先级越低。

第四步：按照所述销账优先级，依次根据所述待销账金额组对应的各挂账凭证，生成各销账指令。

具体的，在线生成销账指令的场景中，可按照销账优先级依次生成销账指令，从而，服务器可按照销账优先级依次相应销账指令。在离线生成销账指令的场景中，可以根据销账优先级和待销账金额组对应的各挂账凭证生成各销账指令。当各销账指令回流线上执行时，服务器可从各销账指令中解析得到各销账指令的销账优先级，从而按照销账优先级依次执行各销账指令。

在本说明书一个可选的实施例中，S108 之前，还可以对生成的销账指令进行校验，以确保销账指令的正确性和完整性，具体实施方式如下：

首先，获取预先设置的销账规则。

其次，确定所述销账指令包含的挂账收入金额对应的挂账凭证与挂账支出金额对应的挂账凭证，符合所述预先设置的销账规则。

具体的，在执行销账指令之前，还可以对销账指令包含的各挂账凭证再次进行校验。校验过程中所采用的预先设置的销账规则可包括校验销账指令包含的各挂账凭证中，挂账收入金额与挂账支出金额是相互匹配的（即加和为零）。销账规则还可包括销账指令包含的各挂账凭证中的挂账信心是否完整、挂账凭

证对应的交易是否执行完毕等。当然，根据不同的应用场景，还可以适应性的设置销账规则，本说明书对此不作限定。

当确定所述销账指令包含的挂账收入金额对应的挂账凭证与挂账支出金额对应的挂账凭证，符合所述预先设置的销账规则时，才可以由服务器响应于销账指令进行销账。如果销账指令包含的挂账凭证不符合销账规则，则该丢弃该销账指令。

在本说明书一个可选的实施例中，S108 之后，还可以对销账指令的执行情况进行监控，以便在销账失败时及时发现销账失败结果和定位销账失败原因，具体实施方式如下：

第一步：获取所述销账指令的执行结果。

具体的，可实时监控销账指令的执行情况，并按照业务执行机构的统计账户、清算通道的整体情况，将销账指令的执行情况汇总成实时报表，并展示给用户。

第二步：如果所述销账指令的执行结果指示销账失败，则根据所述销账指令对应的各挂账凭证生成销账失败消息。

第三步：将所述销账失败消息返回给用户，以使所述用户根据所述销账失败消息确定销账失败原因。

针对执行失败的销账指令，可根据该销账指令对应的各挂账凭证的标识生成销账失败消息。并将销账失败消息返回给用户（操作人员），以使用户根据销账失败消息中携带的各挂账凭证的标识确定销账失败原因。

实际中，可能存在自动销账与人工销账共同处理销账的场景。如果人工销账，没有及时更新挂账凭证的状态，就有可能出现服务器自动销账时错位的根据已经核销的挂账凭证生成销账指令的问题，这样，在执行该销账指令时，销账指令指示的挂账凭证已经经过核销，无法再次核销，因此，该销账指令执行失败。

The specification also provides a computer-readable non-transitory storage medium, wherein the storage medium stores a computer program, which, when executed by a processor, can be used to perform one or more steps of one or more methods described or illustrated herein or provides functionality described or

illustrated herein. Herein, the computer-readable non-transitory storage medium or media may include one or more semiconductor-based or other integrated circuits (ICs) (such, as for example, field-programmable gate arrays (FPGAs) or application-specific ICs (ASICs)), hard disk drives (HDDs), hybrid hard drives (HHDs), optical discs, optical disc drives (ODDs), magneto-optical discs, magneto-optical drives, floppy diskettes, floppy disk drives (FDDs), magnetic tapes, solid-state drives (SSDs), RAM-drives, SECURE DIGITAL cards or drives, any other suitable computer-readable non-transitory storage media, or any suitable combination of two or more of these, where appropriate. A computer-readable non-transitory storage medium may be volatile, non-volatile, or a combination of volatile and non-volatile, where appropriate.

The specification also provides a computer system. FIG. 7 illustrates an example computer system 700. In particular embodiments, one or more computer systems 700 perform one or more steps of one or more methods described or illustrated herein. In particular embodiments, one or more computer systems 700 provide functionality described or illustrated herein. In particular embodiments, software running on one or more computer systems 700 performs one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Particular embodiments include one or more portions of one or more computer systems 700. Herein, reference to a computer system may encompass a computing device, and vice versa, where appropriate. Moreover, reference to a computer system may encompass one or more computer systems, where appropriate.

This disclosure contemplates any suitable number of computer systems 700. This disclosure contemplates computer system 700 taking any suitable physical form. As example and not by way of limitation, computer system 700 may be an embedded computer system, a system-on-chip (SOC), a single-board computer system (SBC) (such as, for example, a computer-on-module (COM) or system-on-module (SOM)), a desktop computer system, a laptop or notebook computer system, an interactive kiosk, a mainframe, a mesh of computer systems, a mobile telephone, a personal

digital assistant (PDA), a server, a tablet computer system, or a combination of two or more of these. Where appropriate, computer system 700 may include one or more computer systems 700; be unitary or distributed; span multiple locations; span multiple machines; span multiple data centers; or reside in a cloud, which may include one or more cloud components in one or more networks. Where appropriate, one or more computer systems 700 may perform without substantial spatial or temporal limitation one or more steps of one or more methods described or illustrated herein. As an example and not by way of limitation, one or more computer systems 700 may perform in real time or in batch mode one or more steps of one or more methods described or illustrated herein. One or more computer systems 700 may perform at different times or at different locations one or more steps of one or more methods described or illustrated herein, where appropriate.

In particular embodiments, computer system 700 includes a processor 702, memory 704, storage 706, an input/output (I/O) interface 708, a communication interface 710, and a bus 712. Although this disclosure describes and illustrates a particular computer system having a particular number of particular components in a particular arrangement, this disclosure contemplates any suitable computer system having any suitable number of any suitable components in any suitable arrangement.

In particular embodiments, processor 702 includes hardware for executing instructions, such as those making up a computer program. As an example and not by way of limitation, to execute instructions, processor 702 may retrieve (or fetch) the instructions from an internal register, an internal cache, memory 704, or storage 706; decode and execute them; and then write one or more results to an internal register, an internal cache, memory 704, or storage 706. In particular embodiments, processor 702 may include one or more internal caches for data, instructions, or addresses. This disclosure contemplates processor 702 including any suitable number of any suitable internal caches, where appropriate. As an example and not by way of limitation, processor 702 may include one or more instruction caches, one or more data caches, and one or more translation lookaside buffers (TLB s). Instructions in the instruction

caches may be copies of instructions in memory 704 or storage 706, and the instruction caches may speed up retrieval of those instructions by processor 702. Data in the data caches may be copies of data of 70 in memory 704 or storage 706 for instructions executing at processor 702 to operate on; the results of previous instructions executed at processor 702 for access by subsequent instructions executing at processor 702 or for writing to memory 704 or storage 706; or other suitable data. The data caches may speed up read or write operations by processor 702. The TLBs may speed up virtual-address translation for processor 702. In particular embodiments, processor 702 may include one or more internal registers for data, instructions, or addresses. This disclosure contemplates processor 702 including any suitable number of any suitable internal registers, where appropriate. Where appropriate, processor 702 may include one or more arithmetic logic units (ALUs); be a multi-core processor; or include one or more processors 702. Although this disclosure describes and illustrates a particular processor, this disclosure contemplates any suitable processor.

In particular embodiments, memory 704 includes main memory for storing instructions for processor 702 to execute or data for processor 702 to operate on. As an example and not by way of limitation, computer system 700 may load instructions from storage 706 or another source (such as, for example, another computer system 700) to memory 704. Processor 702 may then load the instructions from memory 704 to an internal register or internal cache. To execute the instructions, processor 702 may retrieve the instructions from the internal register or internal cache and decode them. During or after execution of the instructions, processor 702 may write one or more results (which may be intermediate or final results) to the internal register or internal cache. Processor 702 may then write one or more of those results to memory 704. In particular embodiments, processor 702 executes only instructions in one or more internal registers or internal caches or in memory 704 (as opposed to storage 706 or elsewhere) and operates only on data in one or more internal registers or internal caches or in memory 704 (as opposed to storage 706 or elsewhere). One or

more memory buses (which may each include an address bus and a data bus) may couple processor 702 to memory 704. Bus 712 may include one or more memory buses, as described below. In particular embodiments, one or more memory management units (MMUs) reside between processor 702 and memory 704 and facilitate accesses to memory 704 requested by processor 702. In particular embodiments, memory 704 includes random access memory (RAM). This RAM may be volatile memory, where appropriate. Where appropriate, this RAM may be dynamic RAM (DRAM) or static RAM (SRAM). Moreover, where appropriate, this RAM may be single-ported or multi-ported RAM. This disclosure contemplates any suitable RAM. Memory 704 may include one or more memories 704, where appropriate. Although this disclosure describes and illustrates particular memory, this disclosure contemplates any suitable memory.

In particular embodiments, storage 706 includes mass storage for data or instructions. As an example and not by way of limitation, storage 706 may include a hard disk drive (HDD), a floppy disk drive, flash memory, an optical disc, a magneto-optical disc, magnetic tape, or a Universal Serial Bus (USB) drive or a combination of two or more of these. Storage 706 may include removable or non-removable (or fixed) media, where appropriate. Storage 706 may be internal or external to computer system 700, where appropriate. In particular embodiments, storage 706 is non-volatile, solid-state memory. In particular embodiments, storage 706 includes read-only memory (ROM). Where appropriate, this ROM may be mask-programmed ROM, programmable ROM (PROM), erasable PROM (EPROM), electrically erasable PROM (EEPROM), electrically alterable ROM (EAROM), or flash memory or a combination of two or more of these. This disclosure contemplates mass storage 706 taking any suitable physical form. Storage 706 may include one or more storage control units facilitating communication between processor 702 and storage 706, where appropriate. Where appropriate, storage 706 may include one or more storages 706. Although this disclosure describes and illustrates particular storage, this disclosure contemplates any suitable storage.

In particular embodiments, I/O interface 708 includes hardware, software, or both, providing one or more interfaces for communication between computer system 700 and one or more I/O devices. Computer system 700 may include one or more of these I/O devices, where appropriate. One or more of these I/O devices may enable communication between a person and computer system 700. As an example and not by way of limitation, an I/O device may include a keyboard, keypad, microphone, monitor, mouse, printer, scanner, speaker, still camera, stylus, tablet, touch screen, trackball, video camera, another suitable I/O device or a combination of two or more of these. An I/O device may include one or more sensors. This disclosure contemplates any suitable I/O devices and any suitable I/O interfaces 708 for them. Where appropriate, I/O interface 708 may include one or more device or software drivers enabling processor 702 to drive one or more of these I/O devices. I/O interface 708 may include one or more I/O interfaces 708, where appropriate. Although this disclosure describes and illustrates a particular I/O interface, this disclosure contemplates any suitable I/O interface.

In particular embodiments, communication interface 710 includes hardware, software, or both providing one or more interfaces for communication (such as, for example, packet-based communication) between computer system 700 and one or more other computer systems 700 or one or more networks. As an example and not by way of limitation, communication interface 710 may include a network interface controller (NIC) or network adapter for communicating with an Ethernet or other wire-based network or a wireless NIC (WNIC) or wireless adapter for communicating with a wireless network, such as a WI-FI network. This disclosure contemplates any suitable network and any suitable communication interface 710 for it. As an example and not by way of limitation, computer system 700 may communicate with an ad hoc network, a personal area network (PAN), a local area network (LAN), a wide area network (WAN), a metropolitan area network (MAN), or one or more portions of the Internet or a combination of two or more of these. One or more portions of one or more of these networks may be wired or wireless. As an example, computer system

700 may communicate with a wireless PAN (WPAN) (such as, for example, a BLUETOOTH WPAN), a WI-FI network, a WI-MAX network, a cellular telephone network (such as, for example, a Global System for Mobile Communications (GSM) network), or other suitable wireless network or a combination of two or more of these. Computer system 700 may include any suitable communication interface 710 for any of these networks, where appropriate. Communication interface 710 may include one or more communication interfaces 710, where appropriate. Although this disclosure describes and illustrates a particular communication interface, this disclosure contemplates any suitable communication interface.

In particular embodiments, bus 712 includes hardware, software, or both coupling components of computer system 700 to each other. As an example and not by way of limitation, bus 712 may include an Accelerated Graphics Port (AGP) or other graphics bus, an Enhanced Industry Standard Architecture (EISA) bus, a front-side bus (FSB), a HYPERTRANSPORT (HT) interconnect, an Industry Standard Architecture (ISA) bus, an INFINIBAND interconnect, a low-pin-count (LPC) bus, a memory bus, a Micro Channel Architecture (MCA) bus, a Peripheral Component Interconnect (PCI) bus, a PCI-Express (PCIe) bus, a serial advanced technology attachment (SATA) bus, a Video Electronics Standards Association local (VLB) bus, or another suitable bus or a combination of two or more of these. Bus 712 may include one or more buses 712, where appropriate. Although this disclosure describes and illustrates a particular bus, this disclosure contemplates any suitable bus or interconnect.

以上所述仅为本说明书的实施例而已，并不用于限制本说明书。对于本领域技术人员来说，本说明书可以有各种更改和变化。凡在本说明书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的权利要求范围之内。

权利要求书

1、一种销账方法，所述方法包括：

获取各挂账凭证；所述挂账凭证是业务平台记录的交易金额与业务执行机构记录的交易金额核对不匹配时生成的；

根据所述各挂账凭证，确定各挂账收入金额和各挂账支出金额；

从所述各挂账收入金额和所述各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额，并根据相互匹配的挂账收入金额和挂账支出金额，确定待销账金额组；

根据所述待销账金额组对应的各挂账凭证，生成销账指令；

响应于所述销账指令，对所述销账指令对应的各挂账凭证进行核销。

2、如权利要求 1 所述的方法，从所述各挂账收入金额和所述各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额，具体包括：

获取预先优化的数量阈值；

从所述各挂账收入金额和所述各挂账支出金额中，确定出相互匹配的第一目标数量的挂账收入金额与第二目标数量的挂账支出金额；

其中，所述第一目标数量与所述第二目标数量之和不大于所述预先优化的数量阈值。

3、如权利要求 1 所述的方法，获取各挂账凭证，具体包括：

获取预先优化的挂账时间周期；

根据挂账凭证对应的挂账时间，获取挂账时间落入所述预先优化的挂账时间周期的各挂账凭证。

4、如权利要求 1 所述的方法，根据所述待销账金额组对应的各挂账凭证，生成销账指令，具体包括：

如果存在多个待销账金额组包含的挂账收入金额相同，则针对每个待销账金额组，获取该待销账金额组对应的各挂账凭证分别对应的挂账时间；

从该待销账金额组对应的各挂账凭证分别对应的挂账时间中选择该待销账金额组对应的参考挂账时间；

根据所述多个待销账金额组分别对应的参考挂账时间，确定所述多个待销账金额对分别对应的销账优先级；

按照所述销账优先级，依次根据所述待销账金额组对应的各挂账凭证，生成各销账指令。

5、如权利要求 2 所述的方法，预先优化数量阈值，具体包括：

确定当前数量阈值；

获取各第一参考挂账凭证，并根据所述各第一参考挂账凭证，确定各参考收入金额和各参考支出金额；

从所述各参考收入金额和所述各参考支出金额中，确定第一参考数量的参考收入金额和第二参考数量的参考支出金额匹配，并记录当前匹配时长；其中，所述第一参考数量和所述第二参考数量之和不大于所述当前数量阈值；

根据所述当前匹配时长和预设的匹配时长阈值之间的差异，重新确定当前数量阈值，并根据重新确定的当前数量阈值，重新匹配参考收入金额和参考支出金额，直至所述当前匹配时长满足预设条件。

6、如权利要求 3 所述的方法，预先优化挂账时间周期，具体包括：

确定当前挂账时间周期；

根据挂账凭证对应的挂账时间，获取挂账时间落入所述当前挂账时间周期的各第二参考挂账凭证；

根据所述各第二参考挂账凭证，确定各参考收入金额和各参考支出金额；

从所述各参考收入金额和所述各参考支出金额中，确定匹配的参考收入金额和匹配的参考支出金额，并记录当前匹配时长；

根据所述当前匹配时长和预设的匹配时长阈值之间的差异，重新确定当前挂账时间周期，并根据重新确定的当前挂账时间周期，重新匹配参考收入金额和参考支出金额，直至所述当前匹配时长满足预设条件。

7、如权利要求 1 所述的方法，响应于所述销账指令，对所述销账指令对应的各挂账凭证进行核销之前，所述方法还包括：

获取预先设置的销账规则；

确定所述销账指令包含的挂账收入金额对应的挂账凭证与挂账支出金额对应的挂账凭证，符合所述预先设置的销账规则。

8、如权利要求 1 所述的方法，响应于所述销账指令，对所述销账指令对应的各挂账凭证进行核销之后，所述方法还包括：

获取所述销账指令的执行结果；

如果所述销账指令的执行结果指示销账失败，则根据所述销账指令对应的各挂账凭证生成销账失败消息；

将所述销账失败消息返回给用户，以使所述用户根据所述销账失败消息确定销账失败原因。

9、如权利要求 1 所述的方法，从所述各挂账收入金额和所述各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额，具体包括：

以键值对的方式存储所述各挂账凭证的标识和所述各挂账凭证对应的挂账金额，其中，所述挂账凭证的标识为键，所述挂账凭证对应的挂账金额为值；

针对每个挂账凭证，根据该挂账凭证的挂账金额，从所述键值对中查找与该挂账凭证的挂账金额匹配的至少一个其他挂账凭证的挂账金额。

10、一种计算机系统，包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序，所述处理器执行所述程序时实现如下步骤：

获取各挂账凭证；所述挂账凭证是业务平台记录的交易金额与业务执行机构记录的交易金额核对不匹配时生成的；

根据所述各挂账凭证，确定各挂账收入金额和各挂账支出金额；

从所述各挂账收入金额和所述各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额，并根据相互匹配的挂账收入金额和挂账支出金额，确定待销账金额组；

根据所述待销账金额组对应的各挂账凭证，生成销账指令；

响应于所述销账指令，对所述销账指令对应的各挂账凭证进行核销。

11、如权利要求 10 所述的计算机系统，从所述各挂账收入金额和所述各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额，具体包括：

获取预先优化的数量阈值；

从所述各挂账收入金额和所述各挂账支出金额中，确定出相互匹配的第一目标数量的挂账收入金额与第二目标数量的挂账支出金额；

其中，所述第一目标数量与所述第二目标数量之和不大于所述预先优化的数量阈值。

12、如权利要求 10 所述的计算机系统，获取各挂账凭证，具体包括：

获取预先优化的挂账时间周期；

根据挂账凭证对应的挂账时间，获取挂账时间落入所述预先优化的挂账时间周期的各挂账凭证。

13、如权利要求 10 所述的计算机系统，根据所述待销账金额组对应的各挂账凭证，生成销账指令，具体包括：

如果存在多个待销账金额组包含的挂账收入金额相同，则针对每个待销账金额组，获取该待销账金额组对应的各挂账凭证分别对应的挂账时间；

从该待销账金额组对应的各挂账凭证分别对应的挂账时间中选择该待销账金额组对应的参考挂账时间；

根据所述多个待销账金额组分别对应的参考挂账时间，确定所述多个待销账金额对分别对应的销账优先级；

按照所述销账优先级，依次根据所述待销账金额组对应的各挂账凭证，生成各销账指令。

14、如权利要求 11 所述的计算机系统，预先优化数量阈值，具体包括：

确定当前数量阈值；

获取各第一参考挂账凭证，并根据所述各第一参考挂账凭证，确定各参考收入金额和各参考支出金额；

从所述各参考收入金额和所述各参考支出金额中，确定第一参考数量的参考收入金额和第二参考数量的参考支出金额匹配，并记录当前匹配时长；其中，所述第一参考数量和所述第二参考数量之和不大于所述当前数量阈值；

根据所述当前匹配时长和预设的匹配时长阈值之间的差异，重新确定当前数量阈值，并根据重新确定的当前数量阈值，重新匹配参考收入金额和参考支出金额，直至所述当前匹配时长满足预设条件。

15、如权利要求 12 所述的计算机系统，预先优化挂账时间周期，具体包括：

确定当前挂账时间周期；

根据挂账凭证对应的挂账时间，获取挂账时间落入所述当前挂账时间周期的各第二参考挂账凭证；

根据所述各第二参考挂账凭证，确定各参考收入金额和各参考支出金额；

从所述各参考收入金额和所述各参考支出金额中，确定匹配的参考收入金额和匹配的参考支出金额，并记录当前匹配时长；

根据所述当前匹配时长和预设的匹配时长阈值之间的差异，重新确定当前挂账时间周期，并根据重新确定的当前挂账时间周期，重新匹配参考收入金额和参考支出金额，直至所述当前匹配时长满足预设条件。

16、如权利要求 10 所述的计算机系统，响应于所述销账指令，对所述销账指令对应的各挂账凭证进行核销之前，所述计算机系统还包括：

获取预先设置的销账规则；

确定所述销账指令包含的挂账收入金额对应的挂账凭证与挂账支出金额对应的挂账凭证，符合所述预先设置的销账规则。

17、如权利要求 10 所述的计算机系统，响应于所述销账指令，对所述销账指令对应的各挂账凭证进行核销之后，所述计算机系统还包括：

获取所述销账指令的执行结果；

如果所述销账指令的执行结果指示销账失败，则根据所述销账指令对应的各挂账凭证生成销账失败消息；

将所述销账失败消息返回给用户，以使所述用户根据所述销账失败消息确定销账失败原因。

18、如权利要求 10 所述的计算机系统，从所述各挂账收入金额和所述各挂账支出金额中，确定相互匹配的挂账收入金额和挂账支出金额，具体包括：

以键值对的方式存储所述各挂账凭证的标识和所述各挂账凭证对应的挂账金额，其中，所述挂账凭证的标识为键，所述挂账凭证对应的挂账金额为值；

针对每个挂账凭证，根据该挂账凭证的挂账金额，从所述键值对中查找与该挂账凭证的挂账金额匹配的至少一个其他挂账凭证的挂账金额。

19、一种计算机可读非易失性存储介质，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述权利要求 1-9 任一项所述的方法。