



(21)申请号 201710179607.4

(22)申请日 2017.03.23

(65)同一申请的已公布的文献号

申请公布号 CN 107066552 A

(43)申请公布日 2017.08.18

(73)专利权人 福建天晴在线互动科技有限公司

地址 350001 福建省福州市长乐市湖南镇
大鹤村

(72)发明人 刘德建 陈铭 钟良德 魏仁海
郭玉湖

(74)专利代理机构 福州市博深专利事务所(普
通合伙) 35214

代理人 林志峥

(51)Int.Cl.

G06F 16/22(2019.01)

(56)对比文件

CN 105512194 A, 2016.04.20,

CN 102460476 B, 2016.02.17,

US 2011313550 A1, 2011.12.22,

US 2014372554 A1, 2014.12.18,

审查员 陈巧艳

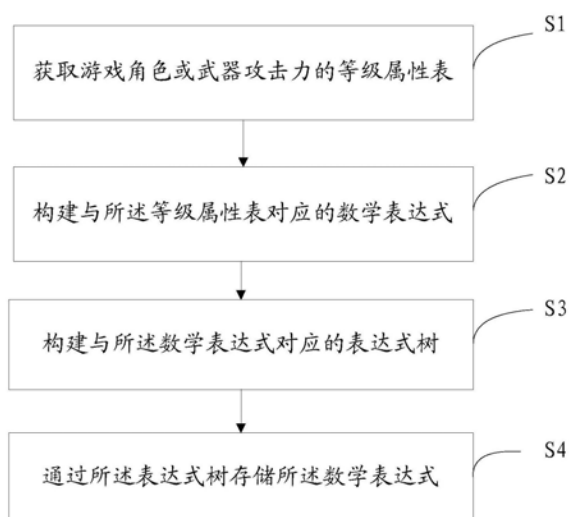
权利要求书2页 说明书6页 附图4页

(54)发明名称

一种游戏用户数据的存储方法及系统

(57)摘要

本发明涉及计算机技术领域,尤其涉及一种游戏用户数据的存储方法及系统,通过以下步骤实现了用户数据的存储:获取游戏角色或武器攻击力的等级属性表;构建与所述等级属性表对应的数学表达式;构建与所述数学表达式对应的表达式树;通过所述表达式树存储所述数学表达式。本发明解决了现有技术中通过数据表存储数据时,需要占用的内存空间过大,导致移动终端性能降低的问题;减小了游戏安装包的大小,同时也降低了游戏运行过程中对CPU占用率,提高了用户在游戏时数据加载的速率,对移动端网络游戏和网页游戏这类对通信和安装包大小要求比较严格的游戏领域具有重要的意义。



1. 一种游戏用户数据的存储方法,其特征在于,包括以下步骤:

S1: 获取游戏角色或武器攻击力的等级属性表;

S2: 构建与所述等级属性表对应的数学表达式;

S3: 构建与所述数学表达式对应的表达式树;

S4: 通过所述表达式树存储所述数学表达式;

其中,所述S2具体为:

S21: 根据所述等级属性表中所有的等级和攻击力,构建坐标点集合;所述坐标点集合中坐标点的横坐标为一所述等级,纵坐标为一所述攻击力;所述攻击力和所述等级一一对应;

S22: 根据坐标点集合的所有坐标点,得到拟合曲线;

S23: 根据所述拟合曲线,得到所述数学表达式。

2. 根据权利要求1所述一种游戏用户数据的存储方法,其特征在于,所述S23具体为:

根据所述拟合曲线上各点的曲率值,将所述拟合曲线分割为第一增长区曲线、平缓区曲线和第二增长区曲线;

根据所述第一增长区曲线和预设第一数学表达式模型,得到第一攻击力表达式;

根据所述平缓区曲线和预设第二数学表达式模型,得到第二攻击力表达式;

根据所述第二增长区曲线和预设第三数学表达式模型,得到第三攻击力表达式;

根据所述第一攻击力表达式、所述第二攻击力表达式和所述第三攻击力表达式,得到所述数学表达式。

3. 根据权利要求1所述一种游戏用户数据的存储方法,其特征在于,还包括:

将所述数学表达式中的除法运算转化为相应的乘法运算。

4. 根据权利要求1所述一种游戏用户数据的存储方法,其特征在于,所述S3具体为:

根据所述数学表达式,构建二叉树的树状结构。

5. 根据权利要求4所述一种游戏用户数据的存储方法,其特征在于,所述S4之后还包括:

加密所述二叉树的数据信息;

根据所述二叉树,得到与输入角色或武器等级相应的攻击力。

6. 一种游戏用户数据的存储系统,其特征在于,包括:

获取模块,用于获取游戏角色或武器攻击力的等级属性表;

第一构建模块,用于构建与所述等级属性表对应的数学表达式;

第二构建模块,用于构建与所述数学表达式对应的表达式树;

存储模块,用于通过所述表达式树存储所述数学表达式;

其中,所述第一构建模块包括:

第一构建单元,用于根据所述等级属性表中所有的等级和攻击力,构建坐标点集合;所述坐标点集合中坐标点的横坐标为一所述等级,纵坐标为一所述攻击力;所述攻击力和所述等级一一对应;

拟合单元,用于根据坐标点集合的所有坐标点,得到拟合曲线;

获取单元,用于根据所述拟合曲线,得到所述数学表达式。

7. 根据权利要求6所述一种游戏用户数据的存储系统,其特征在于,所述获取单元包

括：

分割子单元，用于根据所述拟合曲线上各点的曲率值，将所述拟合曲线分割为第一增长区曲线、平缓区曲线和第二增长区曲线；

第一获取子单元，用于根据所述第一增长区曲线和预设第一数学表达式模型，得到第一攻击力表达式；

第二获取子单元，用于根据所述平缓区曲线和预设第二数学表达式模型，得到第二攻击力表达式；

第三获取子单元，用于根据所述第二增长区曲线和预设第三数学表达式模型，得到第三攻击力表达式；

第四获取子单元，用于根据所述第一攻击力表达式、所述第二攻击力表达式和所述第三攻击力表达式，得到所述数学表达式。

8. 根据权利要求6所述一种游戏用户数据的存储系统，其特征在于，所述第二构建模块包括：

第二构建单元，用于根据所述数学表达式，构建二叉树的树状结构。

一种游戏用户数据的存储方法及系统

技术领域

[0001] 本发明涉及计算机技术领域,尤其涉及一种游戏用户数据的存储方法及系统。

背景技术

[0002] 现有技术中,游戏属性多基于遍历数据表而得到,如:某种武器的攻击力属性,根据武器等级不同拥有不同的攻击力数值,所以每个等级都相应的需要存储一条数据记录。程序在获取该武器攻击力时再从数据表中查询对应的武器名称以及等级的攻击力数值。

[0003] 而上述方法存在以下缺点:

[0004] 缺点1:数据维护困难,初始化时需要对大量数据初始化,维护过程中如果出现需要修改相关属性值,需要维护的数据庞大容易出错。

[0005] 缺点2:数据量大,假设游戏中有50种武器,武器级别有200个等级,则数据表中需存储10000条记录,而这仅仅是该武器一种属性的数据。这些配置文件增加了客户端安装包的大小。

[0006] 缺点3:内存占用高,一个完整的游戏客户端包含了大量的属性配置数据(例如:角色属性、技能属性、战斗属性、装备属性、交易属性、系统属性、社交属性等等),这些配置在游戏启动后需要长时间驻留在客户端的内存中。

[0007] 缺点4:CPU占用高,属性配置数据一般都是以JSON、XML、CSV等格式存储在客户端,这些格式的数据在检索方面有着先天的劣势,导致客户端在搜索配置数据时需要额外占用很高的CPU。对于移动终端来说这意味着更大的耗电量。

发明内容

[0008] 本发明所要解决的技术问题是:提供了一种游戏用户数据的存储方法,通过表达式树对角色或武器等级对应的攻击力进行存储,减小了内存占用空间,提高了内存利用率。

[0009] 为了解决上述技术问题,本发明提供了一种游戏用户数据的存储方法,包括以下步骤:

[0010] S1:获取游戏角色或武器攻击力的等级属性表;

[0011] S2:构建与所述等级属性表对应的数学表达式;

[0012] S3:构建与所述数学表达式对应的表达式树;

[0013] S4:通过所述表达式树存储所述数学表达式。

[0014] 本发明还提供了一种游戏用户数据的存储系统,包括:

[0015] 获取模块,用于获取游戏角色或武器攻击力的等级属性表;

[0016] 第一构建模块,用于构建与所述等级属性表对应的数学表达式;

[0017] 第二构建模块,用于构建与所述数学表达式对应的表达式树;

[0018] 存储模块,用于通过所述表达式树存储所述数学表达式。

[0019] 本发明的有益效果为:

[0020] 相对于现有技术中通过数据表存储游戏角色或武器不同等级对应不同攻击力,存

在占用内存空间过大的问题;本发明通过获取游戏角色或武器攻击力的等级属性表,通过等级属性表中不同等级对应不同的攻击力,得到对应的数学表达式,构建与所述数学表达式对应的表达式树;本发明通过表达式树存储所述角色或武器攻击力与等级之间的对应关系,解决了现有技术中通过庞大的数据表存储数据时,需要占用的内存空间过大,导致移动终端性能降低的问题;通过一个表达式树就能存储一种武器或一种角色的等级与攻击力相关的数据,减小了游戏安装包的大小,同时也降低了游戏运行过程中对CPU占用率,提高了用户在游戏时数据加载的速率,对移动端网络游戏和网页游戏这类对通信和安装包大小要求比较严格的游戏领域具有重要的意义。

附图说明

- [0021] 图1为本发明实施例的一种游戏用户数据的存储方法的主要步骤流程图;
- [0022] 图2为本发明实施例一的一种游戏用户数据的存储方法的步骤流程图;
- [0023] 图3为本发明实施例的一种游戏用户数据的存储系统的结构示意图;
- [0024] 图4为本发明实施例的第一构建模块和第二构建模块的结构示意图;
- [0025] 图5为本发明实施例的获取单元的结构示意图;
- [0026] 标号说明:
- [0027] 1、获取模块;2、第一构建模块;3、第二构建模块;4、存储模块;5、第一构建单元;6、拟合单元;7、获取单元;8、第二构建单元;9、分割子单元;10、第一获取子单元;11、第二获取子单元;12、第三获取子单元;13、第四获取子单元。

具体实施方式

[0028] 为详细说明本发明的技术内容、所实现目的及效果,以下结合实施方式并配合附图详予说明。

[0029] 本发明最关键的构思在于:构建表达式树,通过表达式树对角色或武器攻击力与等级之间的对应关系进行存储,减小了内存占用空间,提高了内存利用率。

[0030] 请参照图1至图2,本发明提供的一种游戏用户数据的存储方法,包括以下步骤:

[0031] S1:获取游戏角色或武器攻击力的等级属性表;

[0032] S2:构建与所述等级属性表对应的数学表达式;

[0033] S3:构建与所述数学表达式对应的表达式树;

[0034] S4:通过所述表达式树存储所述数学表达式。

[0035] 从上述描述可知,从上述描述可知,本发明通过获取游戏角色或武器攻击力的等级属性表,通过等级属性表中不同等级对应不同的攻击力,得到对应的数学表达式,构建与所述数学表达式对应的表达式树;本发明通过表达式树存储所述角色或武器攻击力与等级之间的对应关系,解决了现有技术中通过庞大的数据表存储数据时,需要占用的内存空间过大,导致移动终端性能降低的问题;通过一个表达式树就能存储一种武器或一种角色的等级与攻击力相关的数据,减小了游戏安装包的大小,同时也降低了游戏运行过程中对CPU占用率,提高了用户在游戏时数据加载的速率,对移动端网络游戏和网页游戏这类对通信和安装包大小要求比较严格的游戏领域具有重要的意义。

[0036] 进一步的,所述S2具体为:

[0037] S21:根据所述等级属性表中所有的等级和攻击力,构建坐标点集合;所述坐标点集合中坐标点的横坐标为一所述等级,纵坐标为一所述攻击力;所述攻击力和所述等级一一对应;

[0038] S22:根据坐标点集合的所有坐标点,得到拟合曲线;

[0039] S23:根据所述拟合曲线,得到所述数学表达式。

[0040] 从上述描述可知,通过坐标点,得到拟合曲线,通过拟合曲线,能够更准确得到所述数学表达式,在降低数据存储的同时,提高了数据加载的准确性。

[0041] 进一步的,所述S23具体为:

[0042] 根据所述拟合曲线上各点的曲率值,将所述拟合曲线分割为第一增长区曲线、平缓区曲线和第二增长区曲线;

[0043] 根据所述第一增长区曲线和预设第一数学表达式模型,得到第一攻击力表达式;

[0044] 根据所述平缓区曲线和预设第二数学表达式模型,得到第二攻击力表达式;

[0045] 根据所述第二增长区曲线和预设第三数学表达式模型,得到第三攻击力表达式;

[0046] 根据所述第一攻击力表达式、所述第二攻击力表达式和所述第三攻击力表达式,得到所述数学表达式。

[0047] 从上述描述可知,由于攻击力与等级之间的曲线主要包括三部分:第一增长区曲线、平缓区曲线和第二增长区曲线;第一增长区曲线,为游戏的新手阶段或武器的初级阶段,为了提高用户吸引力,随着等级的增加,攻击力快速增长;平缓区曲线,此阶段随着等级的增加,攻击力的增长平缓;第二增长区曲线,此阶段随着等级的增加,攻击力的增长相对变快;对于三种不同的曲线,建立不同的数学模型,进一步提高了所述数学表达式体现攻击力与等级之间关系的准确性。

[0048] 进一步的,所述一种游戏用户数据的存储方法,还包括:

[0049] 将所述数学表达式中的除法运算转化为相应的乘法运算。

[0050] 从上述描述可知,乘法运算相对于除法运算,能够提高计算机的计算速度,提高了数据加载的速率。

[0051] 进一步的,所述S3具体为:

[0052] 根据所述数学表达式,构建二叉树的树状结构。

[0053] 从上述描述可知,通过所述二叉树对数据进行存储,能进一步减小数据存储的数据量。

[0054] 进一步的,所述S4之后还包括:

[0055] 加密所述二叉树的数据信息;

[0056] 根据所述二叉树,得到与输入角色或武器等级相应的攻击力。

[0057] 从上述描述可知,通过所述二叉树,输入角色或武器等级,就能输出相应的攻击力,相对于数据表的存储方法,极大的减小了内存的占用空间;同时对数据进行加密,提高了数据存储的安全性。

[0058] 请参照图3至图5,本发明还提供了一种游戏用户数据的存储系统,包括:

[0059] 获取模块1,用于获取游戏角色或武器攻击力的等级属性表;

[0060] 第一构建模块2,用于构建与所述等级属性表对应的数学表达式;

[0061] 第二构建模块3,用于构建与所述数学表达式对应的表达式树;

[0062] 存储模块4,用于通过所述表达式树存储所述数学表达式。

[0063] 从上述描述可知,本发明通过获取游戏角色或武器攻击力的等级属性表,通过等级属性表中不同等级对应不同的攻击力,得到对应的数学表达式,构建与所述数学表达式对应的表达式树;本发明通过表达式树存储所述角色或武器攻击力与等级之间的对应关系,解决了现有技术中通过庞大的数据表存储数据时,需要占用的内存空间过大,导致移动终端性能降低的问题;通过一个表达式树就能存储一种武器或一种角色的等级与攻击力相关的数据,减小了游戏安装包的大小,同时也降低了游戏运行过程中对CPU占用率,提高了用户在游戏时数据加载的速率,对移动端网络游戏和网页游戏这类对通信和安装包大小要求比较严格的游戏领域具有重要的意义。

[0064] 进一步的,所述第一构建模块2包括:

[0065] 第一构建单元5,用于根据所述等级属性表中所有的等级和攻击力,构建坐标点集合;所述坐标点集合中坐标点的横坐标为一所述等级,纵坐标为一所述攻击力;所述攻击力和所述等级一一对应;

[0066] 拟合单元6,用于根据坐标点集合的所有坐标点,得到拟合曲线;

[0067] 获取单元7,用于根据所述拟合曲线,得到所述数学表达式。

[0068] 从上述描述可知,通过坐标点,得到拟合曲线,通过拟合曲线,能够更准确得到所述数学表达式,在降低数据存储的同时,提高了数据加载的准确性。

[0069] 进一步的,所述获取单元7包括:

[0070] 分割子单元9,用于根据所述拟合曲线上各点的曲率值,将所述拟合曲线分割为第一增长区曲线、平缓区曲线和第二增长区曲线;

[0071] 第一获取子单元10,用于根据所述第一增长区曲线和预设第一数学表达式模型,得到第一攻击力表达式;

[0072] 第二获取子单元11,用于根据所述平缓区曲线和预设第二数学表达式模型,得到第二攻击力表达式;

[0073] 第三获取子单元12,用于根据所述第二增长区曲线和预设第三数学表达式模型,得到第三攻击力表达式;

[0074] 第四获取子单元13,用于根据所述第一攻击力表达式、所述第二攻击力表达式和所述第三攻击力表达式,得到所述数学表达式。

[0075] 从上述描述可知,由于攻击力与等级之间的曲线主要包括三部分:第一增长区曲线、平缓区曲线和第二增长区曲线;第一增长区曲线,为游戏的新手阶段或武器的初级阶段,为了提高用户吸引力,随着等级的增加,攻击力快速增长;平缓区曲线,此阶段随着等级的增加,攻击力的增长平缓;第二增长区曲线,此阶段随着等级的增加,攻击力的增长相对变快;对于三种不同的曲线,建立不同的数学模型,进一步提高了所述数学表达式体现攻击力与等级之间关系的准确性。

[0076] 进一步的,所述第二构建模块3包括:

[0077] 第二构建单元8,用于根据所述数学表达式,构建二叉树的树状结构。

[0078] 从上述描述可知,通过所述二叉树对数据进行存储,能进一步减小数据存储的数据量。

[0079] 请参照图2,本发明的实施例一为:

[0080] 本发明提供一种游戏用户数据的存储方法,包括以下步骤:

[0081] S1:获取游戏角色或武器攻击力的等级属性表;

[0082] S2:根据所述等级属性表中包括的等级和与所述等级对应的攻击力,构建坐标点集合;所述坐标点集合中坐标点的横坐标为一所述等级,纵坐标为一所述攻击力;所述攻击力和所述等级一一对应;

[0083] S3:根据坐标点集合的所有坐标点,得到拟合曲线;

[0084] S4:根据所述拟合曲线,得到所述数学表达式;

[0085] S5:根据所述数学表达式,构建二叉树的树状结构;

[0086] S6:通过所述二叉树存储所述数学表达式。

[0087] 从上述描述可知,通过表达式树存储武器或角色的等级与攻击力之间的对应关系,无需通过数据表对数据进行存储,减小了游戏安装包的大小,同时也降低了游戏运行过程中对CPU占用率,提高了用户在游戏时数据加载的速率,对移动端网络游戏和网页游戏这类对通信和安装包大小要求比较严格的游戏领域具有重要的意义。

[0088] 本发明的实施例二为:

[0089] 本实施例二与实施例一的区别在于,一种游戏用户数据的存储方法,还包括:

[0090] 将所述数学表达式中的除法运算转化为相应的乘法运算;

[0091] 加密所述二叉树的数据信息;

[0092] 根据所述二叉树,得到与输入角色或武器等级相应的攻击力。

[0093] 从上述描述可知,通过上述方法提高了数据加载的速率,并提高了数据存储的安全性。

[0094] 请参照图3至图5,本发明的实施例三为:

[0095] 通过获取模块获取游戏角色或武器攻击力的等级属性表,并将获取结果发送至第一构建单元;第一构建单元根据所述等级属性表中所有的等级和攻击力,构建坐标点集合,并将坐标点集合发送至拟合单元;所述坐标点集合中坐标点的横坐标为一所述等级,纵坐标为一所述攻击力;所述攻击力和所述等级一一对应;拟合单元根据坐标点集合的所有坐标点,得到拟合曲线,并将拟合曲线发送至分割单元;分割子单元根据所述拟合曲线上各点的曲率值,将所述拟合曲线分割为第一增长区曲线、平缓区曲线和第二增长区曲线,并将第一增长区曲线发送至第一获取单元,将平缓区曲线发送至第二获取单元,将第二增长区曲线发送至第三获取单元;第一获取子单元根据所述第一增长区曲线和预设第一数学表达式模型,得到第一攻击力表达式,并将第一攻击力表达式发送至第四获取单元;第二获取子单元根据所述平缓区曲线和预设第二数学表达式模型,得到第二攻击力表达式,并将第二攻击力表达式发送至第四获取单元;第三获取子单元根据所述第二增长区曲线和预设第三数学表达式模型,得到第三攻击力表达式,并将第三攻击力表达式发送至第四获取单元;第四获取子单元根据所述第一攻击力表达式、所述第二攻击力表达式和所述第三攻击力表达式,得到所述数学表达式,并将所述数学表达式发送至第二构建模块;第二构建模块构建与所述数学表达式对应的表达式树,并将构建结果发送至存储模块;存储模块通过所述表达式树存储所述数学表达式。

[0096] 本发明的实施例四为:

[0097] 获取游戏角色或武器攻击力的等级属性表;根据所述等级属性表中包括的等级和

与所述等级对应的攻击力,构建坐标点集合;所述坐标点集合中坐标点的横坐标为一所述等级,纵坐标为一所述攻击力;将坐标点集合中的所有坐标点进行曲线拟合,通过设置斜率阈值,将得到的曲线根据曲线各点的斜率值,分为第一增长区曲线、平缓区曲线和第二增长区曲线;根据第一增长区曲线和预设的第一数学表达式模型,计算得到第一攻击力表达式,由于第一增长曲线中,随着等级增加,攻击力快速增长,故预设的第一数学表达式模型为 $Y = \text{ATAN}(\text{PLV}) * A$,其中Y表示攻击力的数值,PLV表示等级,A为预设第一系数;根据所述平缓区曲线和预设的第二数学表达式模型,计算得到第二攻击力表达式,由于平缓区曲线,随着等级增加,攻击力趋于平缓,故预设的第二攻击力表达式模型为 $Y = \text{LOG}(\text{PLV} + B) * C$,其中Y表示攻击力的数值,PLV表示等级,B为预设第二系数,C为预设第三系数;通过第二增长区曲线和预设的第三数学表达式模型,计算得到第三攻击力表达式,由于第二增长区曲线,随着等级增加,攻击力增长加快,故预设的第三攻击力表达式模型为 $Y = \text{PLV} * \text{PLV} * \text{PLV} / D$,其中Y表示攻击力的数值,PLV表示等级,D为预设第四系数;根据计算得到的第一攻击力表达式、第二攻击力表达式和第三攻击力表达式,能够得到表示攻击力与等级之间的数学表达式,其中数学表达式中各个变量及常量的数据类型为浮点型,将数学表达式中2的整数次幂的乘法运算或除法运算,转换为相应的位运算,并将数学表达式中的其他的除法运算转化为相应的乘法运算,通过上述方法对数学表达式进行化简,能够提高计算机的运算性能;并将化简后的数学表达式,进行构建相应的表达式树,通过表达式树存储数学表达式。

[0098] 综上所述,本发明提供一种游戏用户数据的存储方法及系统,本发明通过获取游戏角色或武器攻击力的等级属性表,通过等级属性表中不同等级对应不同的攻击力,得到对应的数学表达式,构建与所述数学表达式对应的表达式树;本发明通过表达式树存储所述角色或武器攻击力与等级之间的对应关系,解决了现有技术中通过庞大的数据表存储数据时,需要占用的内存空间过大,导致移动终端性能降低的问题;通过一个表达式树就能存储一种武器或一种角色的等级与攻击力相关的数据,减小了游戏安装包的大小,同时也降低了游戏运行过程中对CPU占用率,提高了用户在游戏时数据加载的速率,对移动端网络游戏和网页游戏这类对通信和安装包大小要求比较严格的游戏领域具有重要的意义。

[0099] 以上所述仅为本发明的实施例,并非因此限制本发明的专利范围,凡是利用本发明说明书及附图内容所作的等同变换,或直接或间接运用在其他相关的技术领域,均同理包括在本发明的专利保护范围内。

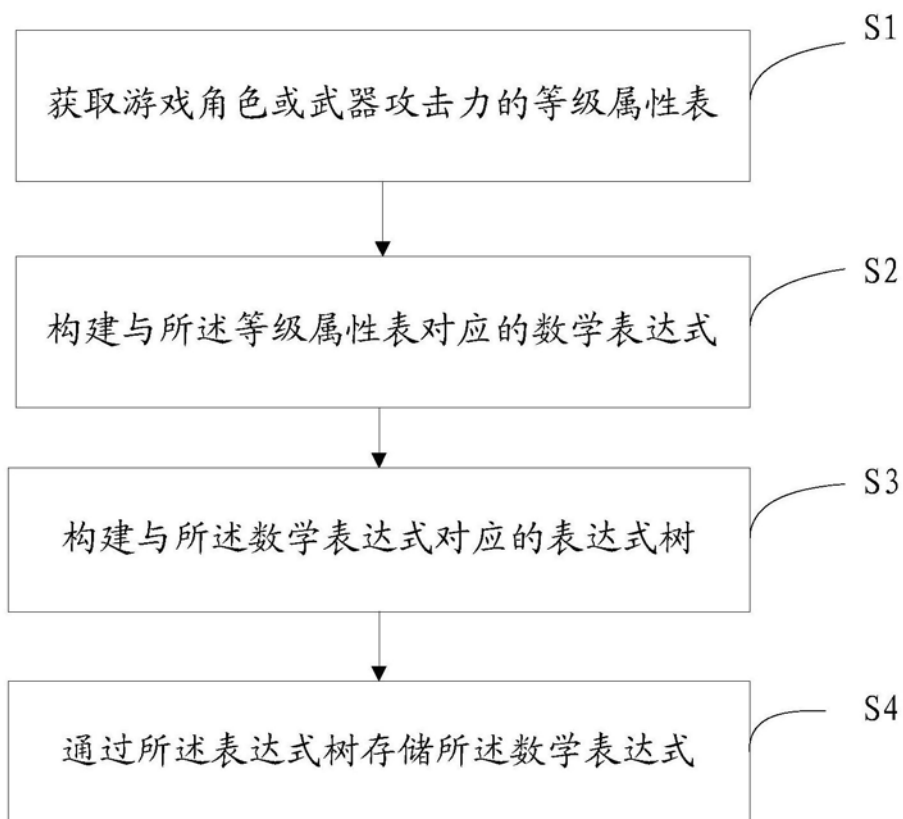


图1

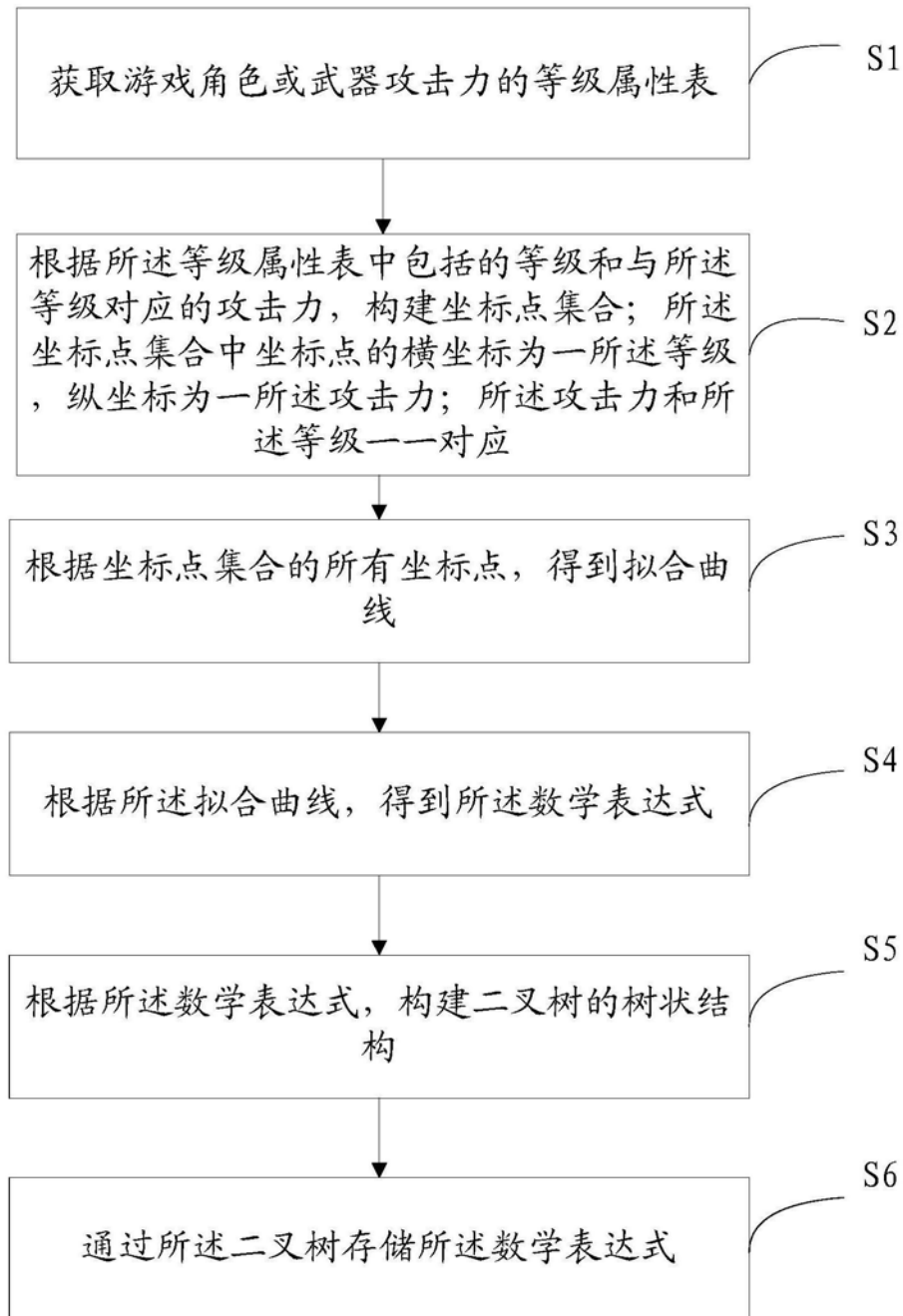


图2

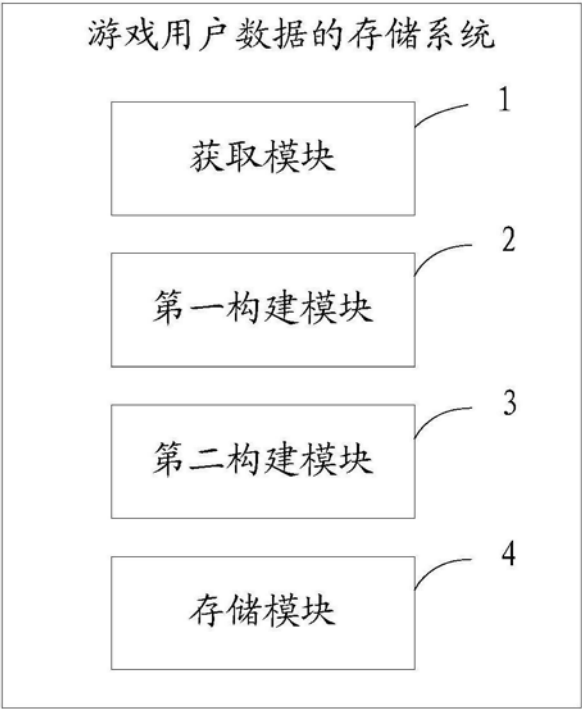


图3

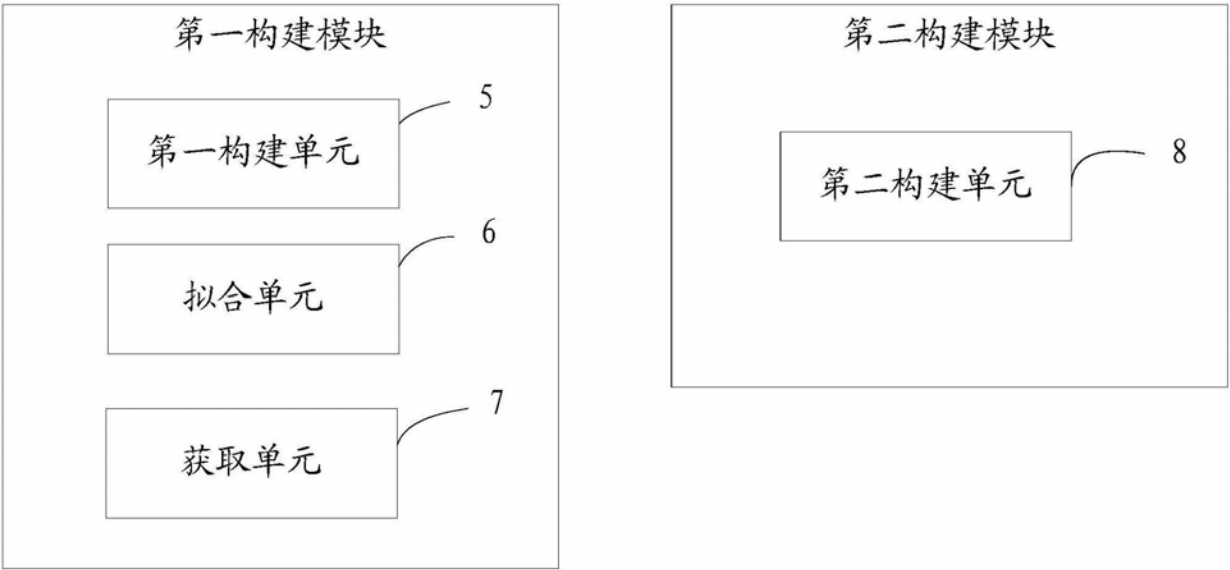


图4

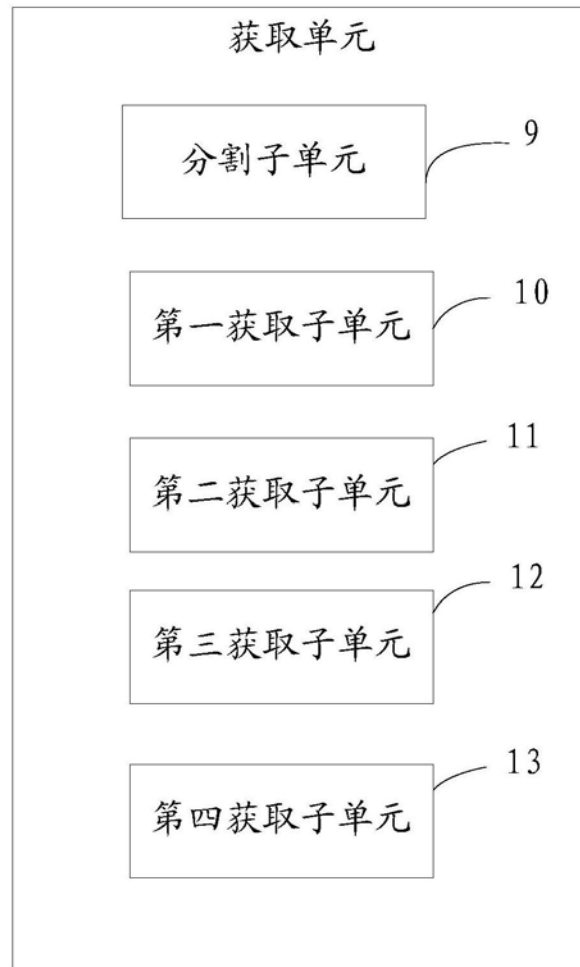


图5