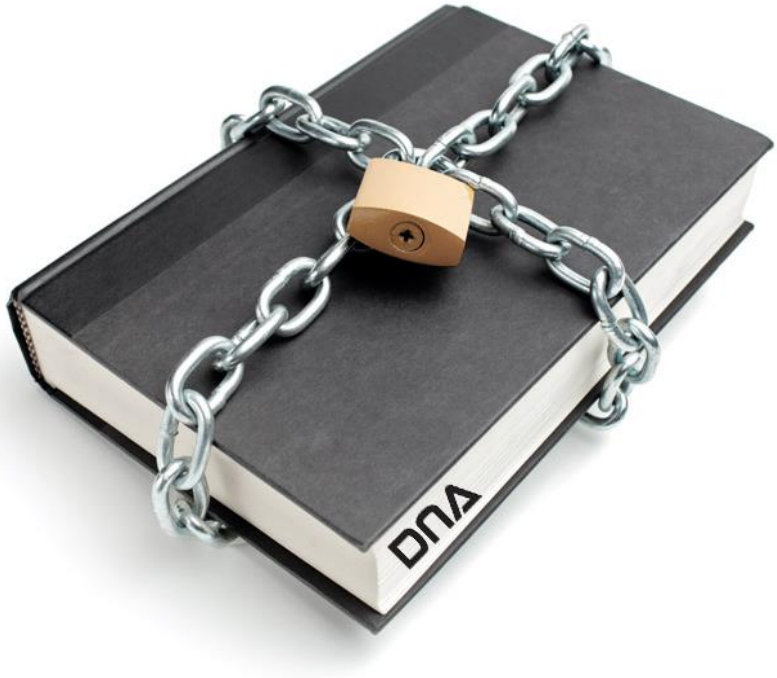
2018 年全国大学生信息安全竞赛

作品报告



作品名称： 基于区块链的隐私保护基因数据分析系统

电子邮箱： 572682252@qq.com

提交日期： 2018 年 6 月 5 日

填写说明

1. 所有参赛项目必须为一个基本完整的设计。作品报告书旨在能够清晰准确地阐述（或图示）该参赛队的参赛项目（或方案）。

2. 作品报告采用A4纸撰写。除标题外，所有内容必需为宋体、小四号字、1.5倍行距。

3. 作品报告中各项目说明文字部分仅供参考，作品报告书撰写完毕后，请删除所有说明文字。(本页不删除)

4. 作品报告模板里已经列的内容仅供参考,作者可以在此基础上增加内容或对文档结构进行微调。

5. 为保证网评的公平、公正，作品报告中应避免出现作者所在学校、院系和指导教师等泄露身份的信息。

参赛作品声明

本团队郑重声明：所呈交的“第十一届全国大学生信息安全竞赛信息安全作品赛”参赛作品是本团队在指导教师的指导下，独立进行研究取得的真实成果，本作品的创意及实现均参赛团队成员原创。

除参赛作品报告中已经注明引用的内容外，本作品不含任何其他个人或集体已经发表或撰写过的作品或成果，本团队对该参赛作品拥有完整、合法的著作权及其他相关权益。因本作品引起的法律结果完全由本团队承担。

本团队及作品严格遵守 2018 年全国大学生信息安全竞赛组委会颁布的《第十一届全国大学生信息安全竞赛参赛指南(信息安全作品赛)》相关规定，并且无侵害他人合法权益、违反国家有关法律、法规以及大赛章程的行为。

特此声明

目录

摘要

随着生物技术的快速发展，基因测序在疾病监测、医疗诊断、生物治疗等领域扮演着越来越重要的角色。同时，随着基因测序成本的降低，基因数据的获取更加便捷，针对基因数据的分析与应用更加普及化，为人类生活带来了诸多益处。

然而，基因序列中蕴含着大量重要的个人信息，一旦基因数据遭到泄露，个人的隐私就不复存在。因此，如何在基因序列保密的前提下，依然能够进行医学分析与疾病诊断，成为学术界以及工业界关注的热点问题。

由于人类全基因数据量庞大，本地基因数据分析只适用于对小规模基因数据。针对海量的基因数据，学者们提出了多种云辅助的数据计算模型。但是，云辅助机制由于引入第三方云端，因此存在着云端与参与方合谋这一重要隐患，不能很好地应对基因分析的场景。

针对上述问题，本项目提出了基于区块链的隐私保护基因数据分析系统。相较于现有模型，该系统利用区块链的抗合谋性解决服务端与参与的一方进行合谋隐患；利用区块链的匿名性实现无法通过致病基因进行身份追踪；利用区块链分布式数据库实现并行处理，提高了系统的效率和扩展性。

本文对比分析了目前典型的隐私保护交集协议，总结了这些工作的优缺点，并创新性地提出了基于区块链的隐私保护交集系统，同时给出了该协议在恶意模型下的安全性机制、有效性机制以及仲裁机制。最后，基于区块链实现了系统的客户端与服务端，完成了系统的构建与测试。

通过系统功能与性能测试，说明了本文提出系统的可靠性，高效性和可拓展性。该基因分析系统给出了海量基因数据时代下集隐私保护、基因分析、疾病诊断于一体的一套较为完整的解决方案。

关键词：隐私保护交集、区块链、基因测序、恶意模型、数据安全

第一章 作品概述

1.1 背景分析及意义

近年来，随着互联网技术的快速发展及云计算、分布式计算等广泛应用，虚拟网络与现实生活变的更加紧密，对互联网大数据进行有效的分析也为我们的生活带来了便捷，但同时随着大量有价值的个人隐私数据不断被挖掘，个人隐私安全也因此受到威胁，如何对大数据进行有效的隐私保护问题成为各行各业的热点问题，基因数据的隐私保护正是主要的研究方向。

基因数据由生物体的基因组数据构成，用以真实地反映基因组 DNA 上的遗传信息，进而较为全面地揭示基因组的复杂性和多样性。当下随着基因测序技术的快速发展，基因组序列数据的检测成本大幅下降，获取更加便捷，更能应用到有意义的领域中，如遗传亲子鉴定、遗传性疾病等等。但同时，由于基因数据包含着敏感的个人隐私，如何让基因数据在有效的保护下依然进行数据分析，成为当下学术界及工业界关注的热点。一方面，隐私保护下基因数据分析的研究可以促进用户对基因数据的使用，只有实现安全性，用户才会上传自身基因数据进行分析；另一方面，用户对基因数据的广泛使用也会加快未来基因的研究进展，为生物遗传学的发展起到不可磨灭的推动作用。

正常人体约有 30 亿个碱基对的基因组，利用二进制编码的方式进行存储占用约 3G 的空间，但全基因测序时，需要 30-40 倍的覆盖度，所以每个人的全基因数据的内存都是较大的。针对海量的基因数据，传统的基因分析技术不足以支撑，利用云服务器进行数据分析虽可以解决大规模数据的处理，但云服务器存在着与参与方合谋这一安全隐患。针对上述问题，本项目利用区块链的去中心化带来的抗合谋性解决。

区块链是一种新型的分布式多方共识协议，无需第三方可信机构的参与，通过分布式实现节点之间的安全共识。并利用加密算法、共识机制等协议，实现了无需信任节点的分布式网络中节点间的交易，解决了目前中心化模式下的可靠性差、安全性低、高成本、低效率等问题，推动了去中心化存储与记录数据的发展。区块链技术不仅可用于电子货币系统，还可用于需要分布式数据共享的应用场景。同时区块链可提供数据分散存储、不可撤销、不可篡改等特性，并兼顾效率、数据安全、隐私保护等安全

特性，同时分布式结构可以通过外包实现实现并行化，提升基因数据分析的效率。

基因数据分析中最常见的是与致病基因的比对，通过用户上传自己要检测的基因片段，与医疗检测机构的致病基因片段比对，判断用户是否患病。因此基因的数据分析可以转化成两个集合计算交集，如果致病基因在交集中说明用户患病。同时由区块链匿名性带来的不可追踪性，交集的基因数据是可以让医疗检测机构一方获知，因为其无法获知上传用户的具体身份。

基于上述原因，本项目希望设计基于区块链的隐私保护基因数据分析系统，该系统解决现有模型低效率、不抗合谋等问题，实现基因数据加密下的交集计算功能，满足用户和医疗检测双方的需求，为基因数据分析提供高强度的隐私保护机制。

1.2 国内外研究现状

1.2.1 基因数据隐私保护的研究与进展

基因数据中蕴藏着巨大的个人隐私，一旦发生泄漏，将会造成极为严重的后果。在法律领域内，基因数据的隐私保护问题一直引起着高度的关注。美国联邦健康和人类服务部在 2012 年 10 月发布了《全基因序列中的隐私与发展》，详尽规划了基因序列的隐私保护法则，我国也在 1998 年 6 月，发布并实施了《人类遗传资源管理暂行办法》。

然而，上述法规的颁布只是从法律角度规范了基因隐私保护问题，并未从技术角度设计如何进行隐私保护。在当下互联网大数据的时代，如何通过安全技术采集、存储、管理、使用基因组数据，已经成为一个不可逾越的必要问题。

近年来在学术界，研究者利用密码学、安全多方计算等隐私保护技术，针对不同类型的基因数据，对 DNA 链的搜索和比较、全基因组序列联合研究等方面进行了研究，对基因组数据如何存储与如何计算也有着诸多的研究。

针对基因组数据如何存储，当下主要有两种方式，一种是利用个人移动设备等存储介质进行存储；另一种是将基因组数据以加密的形式存储于公共云服务器上。Baldi 等[1]在 ACM CCS'11 的论文中，给出的方案是将基因组数据存储在个人设备，保证数据比较安全且费用也较为低廉；De Cristofaro 等[2]也提出了将数据存储在本地设备上，并在移动设备上实现应用的隐私保护方案。Ayday 等[3]利用云服务器的海量数据存储

能力，提出了云辅助的基因数据加密存储方案。

针对基因组数据如何计算，Baldi 等[1]在 ACM CCS'11 的论文中，给出了一个对亲子鉴定等基因易感性疾病进行检测在隐私保护下的安全协议，通过设计对应的 PSI 协议，在保证隐私数据的安全性情况下，给出了 DNA 链相同点位上碱基对的数目的计算。但当 DNA 链的某些点位进行删除、偏移等操作后，协议不再适用，所以灵活性不高。De Cristofaro 等[2]给出了几种基因计算在智能终端实现的方案，但有着只能少量数据计算的局限性。De Cristofaro 等[4]又针对检测特殊的 DNA 子链序列段是否存在于病人的 DNA 链中这一问题，设计了基于同态加密的检测协议，并保证了位置和内容均不泄露。Ayday 等[5,6,7,8]利用云服务器的计算能力，设计了基于同态加密及代理重加密的云辅助基因数据存储运算系统，实现了效率的提高，但现有的云大多都是私有云，存在发生云端与参与方进行合谋的风险，从而对用户造成大量的隐私泄露。

1.2.2 区块链技术的研究与进展

区块链由区块和链构成。以比特币为例，所有交易信息在数字签名后存储在区块上，每个节点在本地维护一份账本副本。区块链通过数字签名等密码学知识实现数据的防篡改、防抵赖等安全保护，解决了比特币系统的拜占庭将军问题与双花问题。Aggelos Kiayias[9]等人于 2015 年证明了区块链理论的安全性。区块链作为一种新型的分布式多方记账系统，正逐渐由单纯记录数字货币的交易信息发展至承载产权、合约等价值存储，并在各大金融机构的努力下衍生出了联盟链与私有链。

目前已有多个区块链底层系统开源，同时，基于区块链的基本安全属性并结合其他密码算法也实现了较多更为复杂的应用。其中较为知名的 Hyperledger Fabric(超级账本)底层框架于 2015 年由 IBM 开源，并在 2017 年推出 1.0 preview 版本，为广大区块链开发者提供了一个完善、稳定的区块链底层框架并提供多种 API 供上层调用。它是一种联盟区块链系统，因此可避免使用消耗物理资源的工作量证明技术。本系统中我们正是对这一框架进行了改进，得到了本系统的框架。

区块链目前主要应用于金融领域，记账实现公开透明，方便监管。但是链上同时记载着许多私人数据，这些数据需要实现隐私保护避免所有人可以从中获取信息。然而区块链目前在隐私保护上只支持匿名账户的机制，隐私保护强度太低。近两年来，不少应用提出了各种区块链上的隐私保护方案，这些方案在强隐私与中心化两者之间

进行折中。

目前区块链主要利用零知识证明、同态加密等密码学知识来解决数据隐私保护问题。2016 年 10 月上线的匿名电子货币系统 Zcash[10]使用非交互式的零知识证明进行隐私保护。零知识证明允许双方在不泄露任何信息的情况下证明某个提议的真实性[11]。交易信息在 Zcash 中是保密的，无关人员无法获得交易中发送方、接收方地址与转账数额，但是他们通过零知识证明可以验证支付是否有效。而以太坊使用同态加密进行隐私保护，同态加密无需对加密数据提前解密就可以进行运算[12]，因此链上的数据可以进行加密，使公有区块链实现私有链的隐私效果。

1.1.3 隐私保护交集(Private Set Intersection，PSI)的研究与进展

隐私保护交集计算是安全多方计算领域内的一个重要方面，在各个领域内的数据都可以表示成集合的形式，并利用集合间的隐私保护计算比对来实现数据间的计算比对，其具体定义是在不泄露各自参与方输入信息的前提下，协同计算输入集合的交集。

PSI 计算是由 Freedom 等[13]在 2004 年提出，借助不经意多项式求值和同态加密得以实现，但这种实现受限于基础密码协议的计算代价。传统的应用系统还是采用不安全的基础 Hash 协议实现 PSI 技术，即对参与双方的集合分别进行 hash 映射，在 hash 值的基础上进行集合交集计算，但显然，这种加密方法很容易受到敌手的碰撞攻击。

近年来，随着对 PSI 问题的深入研究。在 NDSS、CCS 等国际信息安全著名会议上发表了大量的相关研究成果。针对 PSI 技术的研究不仅仅推动安全多方计算的理论基础发展，也推动着 PSI 技术相关的实际问题的发展。

PSI 计算技术根据是否有第三方参与可以分为两大类，第一类是传统的 PSI 计算技术，参与方直接交互执行真实的协议，从而实现对隐私集合的交集计算，这类技术又可以根据实现的原理分为基于公钥加密机制的 PSI、基于混乱电路的 PSI 以及基于不经意多项式的 PSI。在这里我们着重关注基于公钥加密体制的 PSI 协议。

基于公钥加密体制的 PSI 协议主要是对集合进行复杂的公钥加密操作，并通过协议的设计使之能在密文上进行计算，根据协议的设计思想又可以分为基于不经意多项式的 PSI、基于不经意伪随机函数的 PSI 以及基于盲签名的 PSI。

对于基于不经意多项式的 PSI，自从 Freedom 等[13]在 2004 年最先提出后，16 年 Freedom 等[15]又给出了随机 Hash、负载均衡 Hash、布谷鸟 Hash 的实验比对，证明了

负载均衡 Hash 和布谷鸟 Hash 的实验效果相对更好；同时在 2005 年 Kissner 等[17]给出了利用基于多项式环的裴蜀定理的协议设计方法；，2010 年 Hazay 等[21] Freedom 等[13]提出的同态加密的 PSI 协议，给出了恶意模型下利用 cut and choose 的解决办法；2017 年陈振华等[20]给出了给予离散对数且不依赖加密算法的 PSI 协议。

对于基于不经意伪随机函数的 PSI 协议，2008 年 Hazay 等[22]给出了根据 OT 协议涉及的不经意伪随机函数的 PSI 协议；2009 年 Jarecki 等[23]基于复合剩余假设，利用 Dodis-Yampolskiy 伪随机函数[24]和 Camenisch-Shoup 加法同态[25]以及零知识证明提出了另一种 PSI；2010 年 Jarecki 等[26]又利用不可预测函数，提出了速度提升 20 倍的 PSI 协议。

对于基于盲签名的 PSI 协议，2010 年 De Cristofaro 等[27]提出了基于 RSA 的 PSI 协议，但是该协议仅针对半诚实模型是安全的，为了解决恶意模型下的安全性问题，De Cristofaro 等[28]在 2010 年提出了基于零知识证明的 PSI 协议，同时也在 2012 年给出了传统 PSI 的效率比对[29]，证明了基于零知识证明的 PSI 协议是在几个经典的传统 PSI 模型中效率是最高的。

第二类是云辅助的 PSI 计算技术，主要利用云服务器的计算资源进行隐私交集的安全计算，云服务器承担着交集计算的作用，但是不会得到任何的明文信息。云辅助的 PSI 技术大大的提升了隐私计算的效率，但是在模型中，我们要假设云端是可信的，从而解决云端与某一参与方的合谋性，否则的话将退化成两个计算能力相差悬殊的安全多方计算协议。

云辅助的 PSI 协议近几年才有了相关细致的研究，是由 Kerschbaum 等[30]在 2012 年提出的，分别是基于 hash 函数和 RSA 公钥加密算法的 PSI 协议，但两个方案一个是牺牲了安全性换来了高效性，另一个是牺牲了高效性换来了安全性；2014 年 Liu 等[31]提出了基于对称加密和非对称加密的 PSI 协议，实现相对简单但泄露了交集基数；2014 年 Kamara 等[32]提出了基于伪随机函数的 PSI 协议，解决了恶意模型且有着较高的计算效率，但存在这不抗合谋缺陷；2015 年 Abadi[33]提出了基于同态加密和多项式插值的 PSI 协议，但是存在着效率较低的问题。

1.2 项目核心工作

本项目中，我们调研了现有模型存在的问题，传统的模型无法处理大规模数据，以及云辅助的模型存在云服务器与参与的一方进行合谋的隐患。基于上述问题，设计了基于区块链的隐私保护基因数据分析系统，利用区块链的抗合谋性解决服务端与参与的一方进行合谋隐患；利用区块链的匿名性实现无法通过致病基因进行身份追踪；利用区块链分布式数据库实现并行处理，提高了系统的效率和扩展性。

本项目也对比分析了目前典型的隐私保护交集(PSI)协议，总结了这些工作的优缺点，并设计了基于区块链的隐私保护交集(Blockchain Private Set Intersection，BPSI)协议，同时给出了该协议在恶意模型下的安全性机制、有效性机制以及仲裁机制，利用智能合约在改进的区块链上最后应答，并将每次基因分析进行了可视化处理。

最后，基于区块链实现了系统的客户端与服务端，并完成了系统的构建与测试，说明了本文提出系统的可靠性，高效性和可拓展性，给出了海量基因数据时代下集隐私保护、基因分析、疾病诊断于一体的一套较为完整的解决方案。

综上所述，本项目的核心工作主要有以下几点：

1. 设计了利用伪随机函数加密的基于区块链的隐私保护交集(BPSI)协议，实现在链上进行基因比对与患病比对分析，并给出了该协议在恶意模型下的安全性机制、有效性机制与仲裁机制
2. 改进了现有的区块链框架，设计了适用于本项目的改进框架，利用智能合约在链上的计算节点上执行 BPSI 协议，从而实现了本系统的服务端
3. 使用 Express 框架设计了客户端，实现了客户端与服务端功能的接口调用，并完成了基因数据分析时链的可视化
4. 对系统进行了性能测试与功能测试，证明了项目的可靠性，高效性和可拓展性

1.3 特色描述

1. 设计了满足抗合谋性、匿名性与高效性的基于区块链的隐私保护数据分析系统
   - 利用区块链的抗合谋性，解决了传统服务端不可信问题
   - 利用区块链的匿名性，解决了致病基因的不可追踪性
   - 利用区块链的分布式结构，并行化处理提升了系统的效率和扩展性

- 设计并实现了集客户端、服务端以及对应接口的完整系统

2. 改进了区块链的框架，重新设计了区块中的存储结构及存储内容
3. 实现了基因数据分析过程的可视化
4. 设计了基于区块链的隐私保护交集计算(BPSI)协议，实现基因数据的比对分析
   - 给定了恶意模型下协议的安全性机制、有效性机制以及仲裁机制
5. 利用智能合约及时进行双方的费用的自动结算，通过经济手段促进用户的大范围使用，增强系统的时效型
6. 可以将基因切断处理，通过智能合约放在链上，医疗检测机构去响应，实现功能上的外包，提升系统的性能

1.4 应用前景分析

本项目的主要应用人群为医疗基因分析与疾病诊断机构以及需要进行基因分析的个人用户。随着基因测序成本的大幅降低，基因数据的获取更加便捷，针对基因数据的分析与应用更加普及化。然而由于人类基因具有高度机密性，同时涉及到伦理、隐私等问题，因此对于基因隐私的保护成为必然。而当下应用的隐私保护基因数据分析系统，往往存在着效率不高、无法应对海量数据以及消除恶意参与方/第三方下的安全性隐患。另外，现有模型未引入先验知识，也无法验证服务提供商返回数据的真实性。

本项目创新性地提出了利用区块链的抗合谋性、匿名性、分布式特性，解决了现有方案中重要基因隐私数据泄露，服务端/参与方不可信，大规模数据处理速度较慢，通过交集数据可以进行追踪等一系列的缺陷。本项目所设计的基于区块链的隐私保护交集(BPSI)协议，以及该协议在恶意模型下的安全性机制、公平性机制以及仲裁机制，完整地给出了一套安全、高效、可拓展的大数据时代下 PSI 的解决方案，具有较高的理论价值和实用价值。

第二章 作品设计与实现

本章分为三个部分，第一部分为系统概述，说明了本项目的设计思路并对客户端和服务端进行简单的概述；第二部分为技术原理，介绍了我们设计的基于区块链的隐私保护交集(BPSI)协议，并给出了在恶意模型下该协议的安全性机制、有效性机制以及仲裁机制；第三部分为系统实现，给出了系统客户端与服务端实现的详细介绍。

2.1 系统概述

2.1.1 设计思路

本项目为基于区块链的隐私保护基因数据分析系统，本项目中实现的基因数据分析的主要功能是致病基因的比对。对于医疗检测机构，其存储着大量的致病基因片段供用户进行检测；对于用户，若检测是否患病，需固定首尾特定片段寻找待检测基因片段。本项目即可以转化为两个基因数据集合的比对，通过判断致病基因是否在两个基因数据集合交集中确定用户是否患病，所以我们可以转化为隐私保护交集的计算。

隐私保护交集计算是在不泄露各自参与方输入信息的前提下，协同计算输入集合的交集。在本项目中，系统需要在不影响交集计算的结果下，比对用户数据与医疗检测机构数据的交集，且双方不知道对方与己方不同的数据。

在本系统中，交集部分的信息即比对后得到的致病基因，是可以被医疗检测机构一方获得的，但是这个并不存在隐私的泄露，因为区块链的匿名性，医疗检测机构仅能知道有用户换了某种病，但是无法知道是谁患了这种病，所以我们认为交集部分的基因数据泄露是不存在安全性风险的，同时由于基因追踪是通过个人或家族的特有基因部分进行追踪，而这部分是无法比对出，也就是加密在链上的，所以利用区块链的匿名性，我们也解决了对用户的基因追踪这一问题。

图 2-1 展现了本系统的设计架构。本系统使用区块链作为隐私保护交集计算的服务端，医疗检测机构和用户各自上传待检测致病基因数据库和用户个人的待检测基因片段序列。通过智能合约执行 2.2 部分设计的基于区块链的隐私交集(BPSI)协议,由区块链中的计算节点完成计算后，将结果返回给客户。

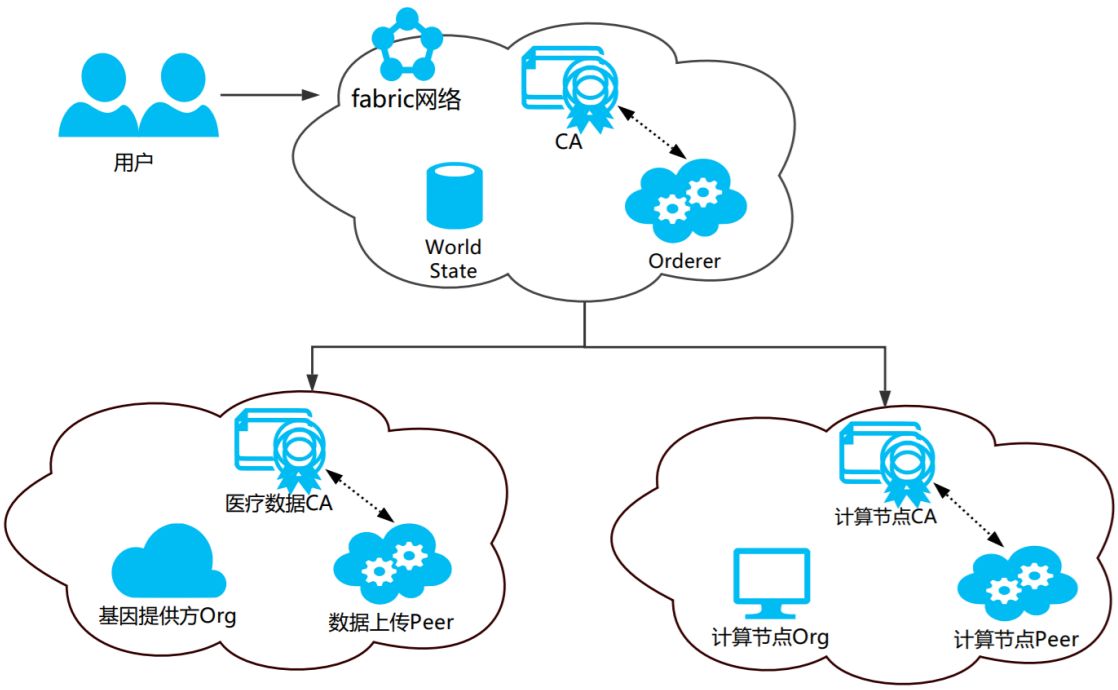


图 2-1 系统架构

2.1.2 系统功能

表 2-1 系统功能描述

序号	功能描述
1	医疗机构上传致病基因数据
2	用户发起基因比对
3	区块链返回基因比对结果
4	用户分析疾病风险

2.1.3 系统设计

1) 客户端

客户端网站由前后端构成，后端负责中间层的路由、异步的发起网络请求和校验，主要用 Express 框架实现，前端的交互界面使用 React 开发。

客户端实现的主要功能有医疗节点上传致病基因、客户匿名的比对基因和区块链的查看。用户在客户端上传加密的基因序列片段，会在片段中加入随机扰动，以判断扰动是否被识别出，来判断计算节点的计算是否正确。

2) 服务端

服务端是一个私有链，是由 Hyperledger Fabric 开发。私有链由排序节点、医疗中心 CA、医疗中心节点、计算节点 CA、计算节点组成。计算节点负责计算用户上传的待检测基因片段和医疗检测中心上传的致病基因片段的交集，并将结果返回给用户。根据计算量，计算节点可以收到相应的报酬，交易的可靠性由区块链来保证。有资质的医疗检测机构上传已发现的致病基因片段，并根据最新的研究，动态的更新致病基因库。

计算节点和医疗数据上传方的规模都是动态的，分别从计算中心 CA 和医疗数据 CA 得到认证的节点都可以加入区块链网络，完成功能，体现了系统的灵活性和可拓展性。

2.2 技术原理

2.2.1 基于区块链的隐私保护交集(BPSI)计算技术原理

传统的隐私保护交集(PSI)技术是不适用于大规模场景的，这是因为基于公钥加密机制协议的开销很大，不适合大规模信息的传输比对；基于混乱电路的协议需要将电路提前构建并全部加载到内存中，当集合很大时会造成内存的限制；基于布隆过滤器的协议需要加载布隆过滤器结构至内存，所以上述的办法都不适合大规模数据的应用问题。

云辅助的 PSI 技术中解决了大规模数据的应用问题，但由于现在的云服务器大多都是私有云，所以云服务器存在与参与方进行合谋的隐患。区块链相比云服务端，其发生合谋性取决于计算所有节点的矿工中是否有超过 2/3 的矿工是恶意的，这个可能性是极为微小的，所以采用区块链进行协议的运行可以解决抗合谋性问题。

相比传统的 PSI 协议和云辅助的 PSI 协议，区块链由于链上信息公开透明，所以任何人都可以看到链上的信息，所以对于密钥协商、加密处理等步骤是完全不同的，所以我们需要根据区块链的特性设计新的 PSI 协议，在这里我们称之为基于区块链的隐私保护交集(BPSI)协议。

我们比对了当下的云辅助 PSI 协议，综合算法复杂度、计算复杂度、以及在不同模型（半诚实模型、恶意模型）下的安全性，基于 Kamara 等[19]提出的云辅助的基于

伪随机函数加密的 PSI 技术的思想，设计出了适用于本系统的 BPSI 协议。在给出具体的算法之前，我们先定义一下我们在协议设计中使用的参数的定义。

表 2-2 协议符号与参数

k	伪随机置换密钥的长度
T	补充参量集合长度
λ	信息复制处理份数
S_1	用户P_1拥有的信息集合
S_2	用户P_2拥有的信息集合
K	伪随机置换密钥，$K=\{0,1\}^k$
S^λ	信息复制处理，$S^\lambda=\{x\|\|1,\dots,x\|\|\lambda: x\in S\}$，$\lambda\geq 1$，其中$(S^\lambda)^{-\lambda}=S$,$\lambda$换成二进制序列
F	伪随机置换函数，F：$\{0,1\}^k\times U\to\{0,1\}^{\geq k}$，$F(S)=\{F(s)$：$s\in S\}$
π	随机扰动处理，$\pi(S)=\{x_{\pi(i)}$：$x_i\in S\}$
D_i	补充参量集合，$\|D_i\|=t, D_i\in D\neq U$,i = 0,1,2

下面我们给出我们设计的 BPSI 协议的流程图以及具体协议设计。

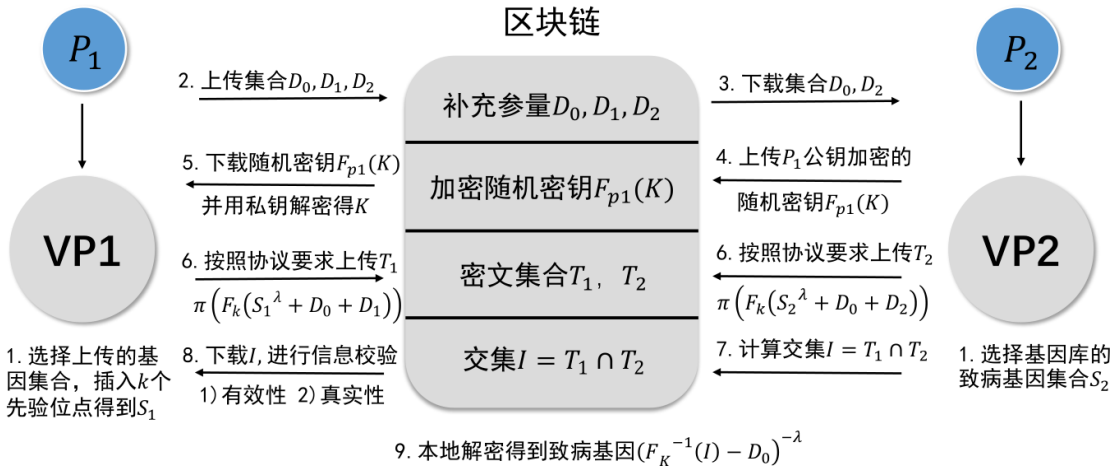


图 2-2 BPSI 协议流程图

表 2-3：BPSI 协议

定义与输出：

$$F\colon \{0,1\}^k \times U \to \{0,1\}^{\geq k}$$

协议流程：

//交互补充参量集合

1.用户P_1选择$D_i \in D \neq U, |D_i| = t, i = 0,1,2$，通过智能合约上传到区块链上，并发起任务分包请求。

2.某个医疗服务提供商P_2应答智能合约，下载到本地得到D_i $(i = 0,1,2)$。并发起新的智能合约，将P_2公钥上传到区块链上。

//伪随机函数密钥协商

3.用户P_1通过智能合约将自身公钥上传到区块链上，同时应答链上P_2的智能合约，下载得到该医疗服务提供商P_2的公钥。

4.用户P_1随机生成安全参数为k的密钥K，利用P_2的公钥进行加密，通过智能合约上传到区块上。

5.医疗服务提供商P_2利用自身私钥解密，得到随机密钥K。并计算$T_2 = \pi_2\left(F_k\left({S_2}^{\lambda} + D_0 + D_2\right)\right)$,上传到区块链上。

6.用户P_1计算$T_1 = \pi_1\left(F_k\left({S_1}^{\lambda} + D_0 + D_1\right)\right)$，响应$P_2$智能合约，将$T_1$上传到区块上。

//计算交集

7.医疗服务提供商P_2获得T_1, T_2，计算交集$I = T_1 \cap T_2$，并将结果公布在链上。

//判断机制

8.用户P_1响应智能合约，从区块链上下载交集。如果出现下述情况，信息校验无效，协议终止：

(1)$D_0 \notin {F_K}^{-1}(S_i)$或$D_0 \cap {F_K}^{-1}(S_i) \neq \phi$

(2)存在$x \in S_i, \alpha, \beta \in \{1,2,\dots,\lambda\}$，使得$x||\alpha \in {F_K}^{-1}(S_i)$且$x||\beta \notin {F_K}^{-1}(S_i)$

//返回结果

9.P_1, P_2在链上获得交集信息，下载到本地将交集进行解密$({F_K}^{-1}(I) - D_0)^{-\lambda}$

以上即为我们设计的 BPSI 协议，关于本协议的安全性机制、有效性机制以及仲裁机制，我们会在下面的章节详细介绍。

2.2.2 安全性机制

安全性机制是安全多方计算（SMC）和隐私保护交集（PSI）领域的核心问题。目前有很多学者针对不同的安全性机制提出了多种实用协议。为了讨论提出的基于区块链的隐私保护系统的安全级别，首先给出敌手模型的基本概念，根据其参与的行为方式可分为以下三类：

1）半诚实模型（semi-honest），即协议参与的各方均遵循协议的执行过程，不恶意篡改协议规范。但参与方可能根据协议执行过程中推断其他参与者的信息，或泄露相关信息。

2）恶意模型（malicious adversary），即参与方不遵守协议执行过程，对协议进行一定程度的破坏：如拒绝参与协议、修改隐私的输入集合信息、提前终止协议的执行等。

3）隐蔽敌手模型（covert adversary），即介于半诚实模型和恶意模型之间，由于其担心暴露或被检测，则将恶意行为混淆在正常行为中，从而只有一定的概率被检测出来。

我们提出的模型，解决了恶意模型下的安全性问题，具体原因如下:

首先，本系统的参与方共有两方，分别为用户方P_1、医疗服务提供商P_2(自身充当链计算节点S)。其中，用户P_1和区块链计算节点S是半诚实的。这是由于用户P_1是服务购买者，故在计算过程中由于需要获取有用信息，故需要提供自己真实信息并按照协议要求执行，故为半诚实模型。在支付过程的安全性有成熟的区块链技术保证，防止用户P_1存在二次购买等恶意行为。而区块链节点由于其自身的行者，保证了在参与节点较多时，其被收买的难度很大，即是恶意模型的概率很小。因此我们假设S是半诚实模型（由于区块链上信息公开，故存在被第三方利用的可能）。

其次，医疗服务提供商P_2在模型被视为恶意模型（不遵守协议）。只要该系统能够在医疗服务提供商恶意的前提下仍能保证安全，则在半诚实模型和隐蔽敌手模型下同样适用。容易得到，当医疗服务提供商发生不遵守协议的行为时，用户可以在本地进行安全性核查。这是由于在伪随机置换协议中引入了信息复制操作S^{λ}、补充参量操作$\tau_i = D_0 \cup D_i$和随机扰动操作Π，从而客户端当不履行协议（如：不参与协议、修改隐私的输入集合信息、提前终止协议的执行）时，会导致最后上传集合的不完整或错

误，从而患者在进行安全性校验时会发现该错误，如算法 1 所示。此外，由于引入冗余保护机制，使得医疗服务提供商端的破解空间明显增大（λ倍以上），使得暴力破解方式（如彩虹表攻击等）的难度更大，破解成功概率更低。综上所示，本系统算法实现了在恶意模型下的安全机制。

另外，我们考虑到由于医疗机构提供商可能提供错误的信息，来达到获取他人信息的恶意目的。在算法外引入了有效性机制检验（见章节 2.2.3）。在区块链可信任的前提下，用户端通过引入先验性知识对医疗机构数据进行概率性校验，避免了医疗机构恶意构造输入，但遵守协议产生的信息泄露等问题。

最后，由于区块链的天然抗合谋性，使得我们系统计算方和医疗服务提供方无法合谋获取用户的隐私数据。值得注意的是，在区块链可信任的前提下，本系统的安全级别超过了多种传统云隐私保护交集技术的安全级别[30,31,32,33]，并从根本上解决了多方合谋的安全隐患。

2.2.3 有效性机制

有效性机制是考虑如果恶意医疗检测机构P_2并没有拿出真正的致病基因，供用户进行基因数据分析，即信息源存在错误时如何检测出问题。

对于用户P_1，其上传自己的致病基因前，应该首先进行测试，在这里，我们认为用户P_1可以通过查阅等方式对基因序列知识有着一定的了解。同时需要一个权威的基因更新发布机构，以供医疗检测机构更新数据库、用户了解新的基因序列知识，本系统中选择 Cosmic[34]数据库作为权威基因更新发布机构。测试方法如下：

表 2-4 BPSI 协议的有效性机制检测

1.用户P_1选择一段正常人的致病基因段，从中更改 a 个引起致病的基因点，上传到区块上，并进行广播
2.医疗检测机构P_2更新链，拿到用户P_1上传的待检测数据进行检测
3.检测后，医疗检测机构P_2将进行基因比对的致病点序列段交集上传到区块上，并进行广播
4.用户P_1更新链，比对医疗检测机构P_2是否检测出 a 个致病基因点
5.如果 a 个致病基因点全部检测出来，且并未检测出其他致病基因点，我们认为医疗检测机构P_2是诚实的；否则，医疗检测机构P_2是恶意的

下面我们对有效性机制进行证明：

不妨设这段基因序列一共有$n+a$个碱基对。如果医疗检测机构P_2是恶意的，第一种可能是P_2故意上传了一段错误的基因端，由有效性机制可直接判定；第二种可能是这段基因是医疗检测机构P_2随机生成的，所以医疗检测机构P_2为恶意的敌手且无法检测出来的概率为$P=\frac{2^n}{2^{n+a}}=\frac{1}{2^a}$ 。

例如，在恶意医疗检查机构随机返回交集，安全参数$a \geq 10$时，计算得到无法检测出敌手的概率$P<\frac{1}{1000}$，是一个比较小的数值。因此，当安全参数 a 足够大时，医疗检测机构P_2成功欺骗用户的概率可以忽略不计。

2.2.4 仲裁机制

本节给出该系统的仲裁机制算法设计和说明。值得注意的是，仲裁机制依赖于安全性与有效性的分析。仲裁节点由可信第三方负责，若用户P_1与医疗检测机构P_2双方在整个过程中完全遵守协议运行，则无需仲裁节点参与，但在数据传输及结果返回过程中，若有任何一方提出异议，则需要将此次共享过程提交给仲裁方进行判定。仲裁法进行审判后，失败一方将受到惩罚。具体惩罚机制可由联盟自行约定，不在本系统中做具体规定。本系统设计的仲裁机制需要在联盟中的各个节点配合的情况下才能成功解决争议。该仲裁机制与 CCP 机制中的中央对手方作用类似，增加了系统的中心化色彩，但区别在于 CCP 需要对每一笔交易进行校验，但本系统的仲裁只在参与双方提出异议时工作。

对于本系统，由于用户P_1与医疗检测机构P_2双方资源差距过大，所以我们只考虑用户P_1申请仲裁带来的影响。

- 用户P_1控诉医疗检测机构P_2，当用户P_1对交集部分的基因数据解密后，安全校验出现错误，及出现$D_0 \notin {F_K}^{-1}(S_i)$或$D_0 \cap {F_K}^{-1}(S_i) \neq \phi$或$x||\alpha \in {F_K}^{-1}(S_i)$且$x||\beta \notin {F_K}^{-1}(S_i)$时，说明安全校验出现问题，此时仲裁介入，由于区块链的不可篡改性，D_0, D_1, D_2均有着记录，用户P_1将安全校验的错误通过智能合约发布在链上，仲裁通过判定安全校验的错误性判定用户P_1与医疗检测机构P_2谁将胜诉。
- 用户P_1控诉医疗检测机构P_2上传的数据不正确，当进行有效性检测时，用

户P_1上传的基因序列段中 k 个致病基因点存在未检测出的，此时仲裁介入，用户P_1将进行有效性检测的加密密钥，以及上传的带有 k 个致病基因点的基因序列段通过智能合约发布在链上，仲裁通过将这 k 个致病基因点的基因序列端利用密钥加密与之前得到的交集进行比较，仲裁通过是否存在某加密后的序列段不在之前的交集内判定用户P_1与医疗检测机构P_2谁将胜诉。

2.3 系统实现

本项目中我们设计了基于区块链的隐私保护基因数据分析系统，该系统解决现有模型低效率、不抗合谋等问题。本系统的客户端实现医疗检测机构方的基因库上传与更新的功能，以及用户方选择疾病类型、上传个人基因、缴费、得到返回检测结果等功能；本系统的服务端实现了返回数据以及和客户端互相认证的功能。图 2-3 位本系统的系统设计构架图，本章节我们将通过系统客户端与系统服务端介绍系统的实现。

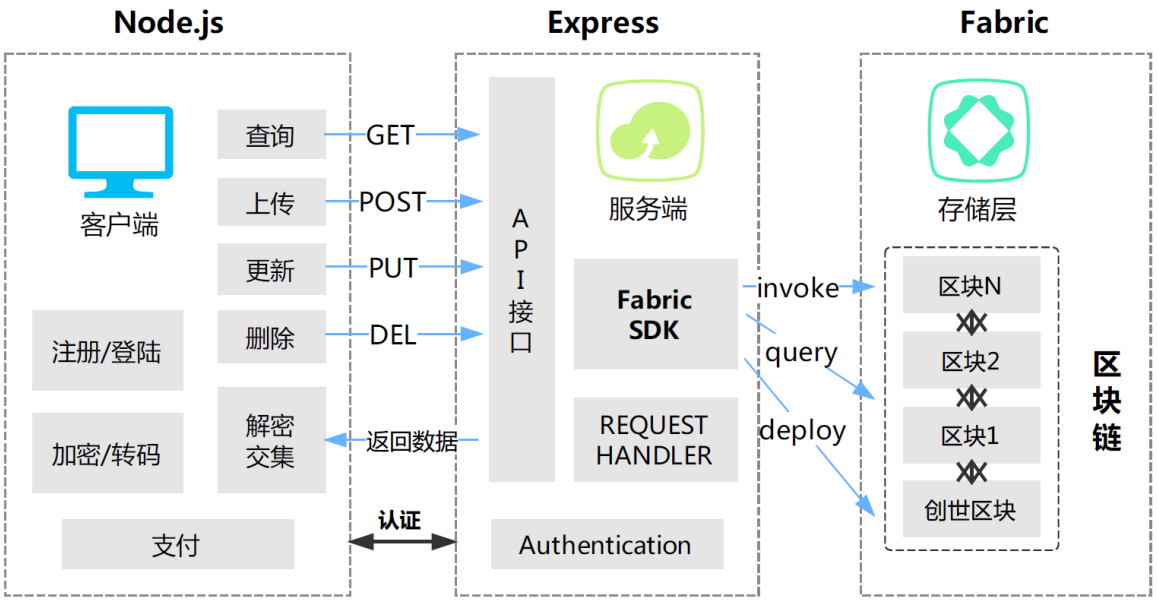


图 2-3 系统设计构架图

2.3.1 系统客户端

本系统的客户端网站后端使用 Express 4.15.4 框架，前端使用 React15.6.1 框架。客户端与区块链之间开发 Fabric-SDK-node 进行交互，交互方式是 gRPC 点对点通信。计算节点会把由共识节点共识的交易发送给排序节点，获取实际的收益；排序节点根

据交易的先后顺序生成区块。

表 2-5 是 express 中间层实现的接口及调用方法，客户端需要访问表中的接口实现预期功能。

表 2-5 express 中间层实现的接口及调用方法

中间层地址	方法	说明
/gene/api/compare	POST	用户上传比较基因
/gene/api/retrieve	POST	用户查询结果
/disease/api/upload	POST	上传致病基因
/disease/api/ls	GET	检索可供检测的疾病
/block/api/getall	GET	获取最新的区块链

本系统客户端共分为两个部分，包括医疗检测机构使用的基因数据节点界面以及用户使用的用户界面。对于基因数据节点界面，实现了指定疾病的类型、向区块链上传新发现的致病基因的功能；对于用户界面，实现了选择比对疾病类型、利用计算节点上传个人基因、根据计算量使用区块链的代币支付、得到返回检测结果、判断基因比对结果合法性等功能。图 2-4 为进入本系统客户端的界面，之后选择登陆。

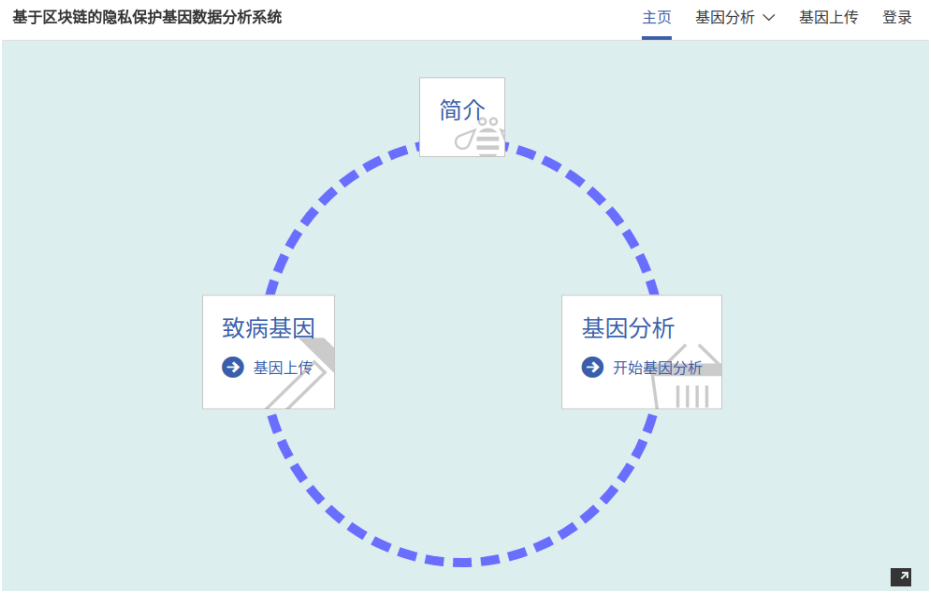


图 2-4 系统客户端初始界面

当医疗检测机构登陆时，进入医疗检测机构使用的基因数据节点界面，如图 2-5 所示。在此界面，医疗机构可以选择上传致病基因并选择类型。

图 2-5 医疗检测机构使用的基因数据节点界面

当用户登陆时，进入用户使用的用户界面，如图 2-6 所示。用户选择待检测的疾病类型。在本章，我们选用糖尿病、心脏病和肺癌作为示例进行系统功能介绍。

图 2-6 用户选择待检测疾病类型

用户选择了待检测疾病类型后，需要选择恰当的计算节点，如图 2-7 所示；再根据评价进行缴费，如图 2-8 所示。本系统中，用户可以看到计算节点的历史工作频率、评价和价格，根据分析选择性价比高而且可靠的计算节点。其中计算节点的使用频率和评价通过计算节点的工作历史获得，每次交易后，自动触发智能合约进行统计。

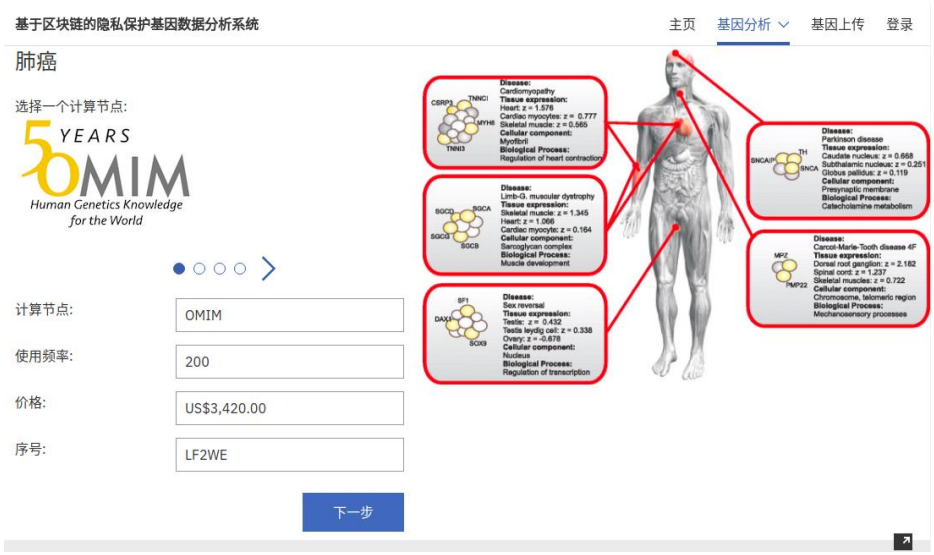


图 2-7 用户选择计算节点界面

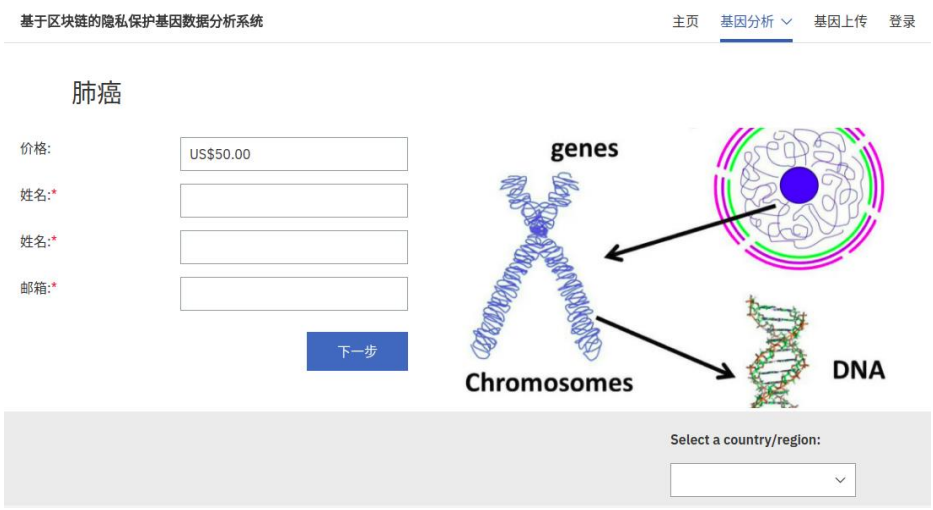


图 2-8 用户缴费界面

选择计算节点并缴费后，用户上传自己带检测疾病的基因片段，便可以在计算节点上与从医疗检测中心获得的致病基因序列进行在加密的模式下进行比对。在本章，我们选择上传与癌相关的基因片段进行基因数据分析，如图 2-9 所示。

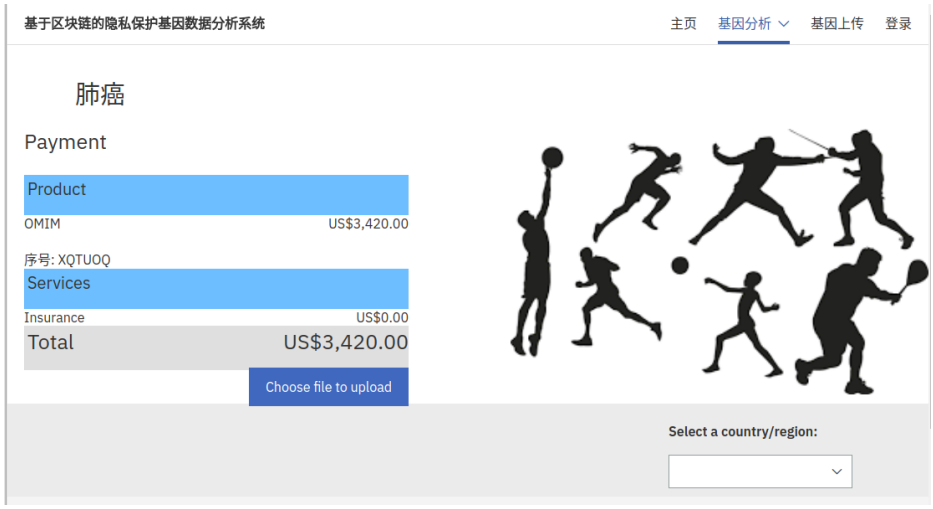


图 2-9 用户上传基因界面

用户上传基因后，首先进行 2.2.4 部分的有效性测试，及需要检测出所有用户主动修改的致病位点，证明计算节点是诚实工作的。

之后进行待检测基因与致病基因的检测，在这里，计算节点检测出用户存在一位与肺癌相关的异常基因位点，因此分析得到用户未来患肺癌的几率要高于平均值，同时表明肺癌基因致病与环境因素有关，戒烟能大幅降级用户你患病可能性，最后将上述结论返回给用户。

如图 2-10 所示，界面显示了返回给用户的信息内容，同时在界面下方将区块链的动态变化可视化处理，可以看到区块的数量随着交易的进行开始增长，增强用户的使用观感以及使用效果。

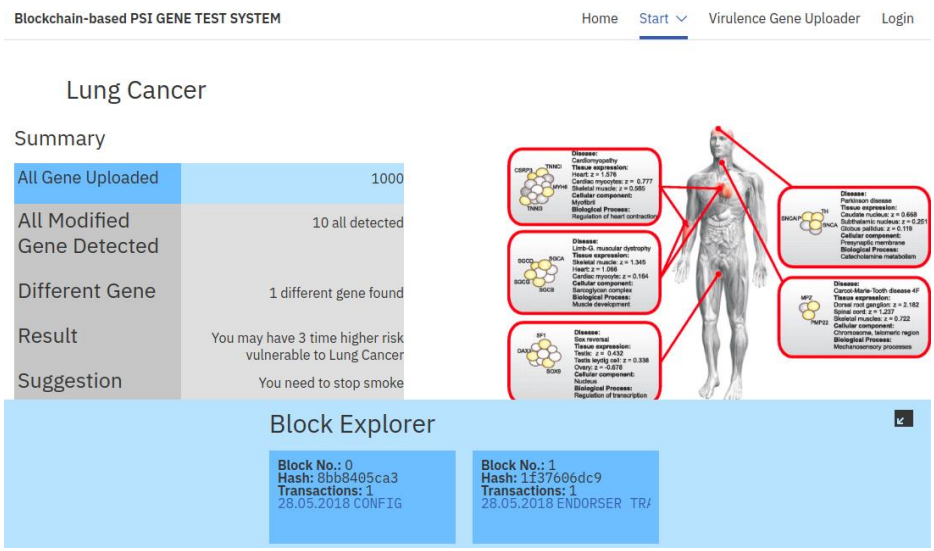


图 2-10 基因数据分析结果与分析

2.3.2 系统服务端

2.3.2.1 系统架构实现

系统的服务端通过部署私有链实现，本系统中，我们使用 Hyperleger Fabric1.1.0 作为本系统的开发框架。Hyperledger Fabric(超级账本)底层框架于 2015 年由 IBM 开源，并在 2017 年推出 1.0 preview 版本，为广大区块链开发者提供了一个完善、稳定的区块链底层框架并提供多种 API 供上层调用。它是一种联盟区块链系统，因此可避免使用消耗物理资源的工作量证明技术。

Hyperleger Fabric 支持多通道。每一个通道均包含由背书节点、排序节点、CA 的节点结构和账本、智能合约、区块链等组成的逻辑结构。每个通道只维护自己通道上的区块链和账本，不同的通道之间将参与者的数据进行隔离，满足了不同业务场景下的“不同的人访问不同数据”的基本要求。一个节点也可以参与到多个通道中。Hypeledger 中账本是一系列有序的、不可篡改的状态转移记录日志。状态转移是智能合约执行的结果，每个交易都是通过增删改操作提交一系列键值对到账本。每个通道都有其账本，每个 peer 节点都保存着其加入的通道的账本，包含着账本数据库、状态数据库以及历史数据库。账本状态数据库实际上存储的是所有曾经在交易中出现的键值对的最新值。调用链码执行交易可以改变状态数据，为了高效的执行链码调用，所有数据的最新值都被存放在状态数据库中。图 2-11 为 Hyperleger Fabric1.1.0 的系统架构；图 2-12 为 Hyperleger Fabric1.1.0 的交易流程图。

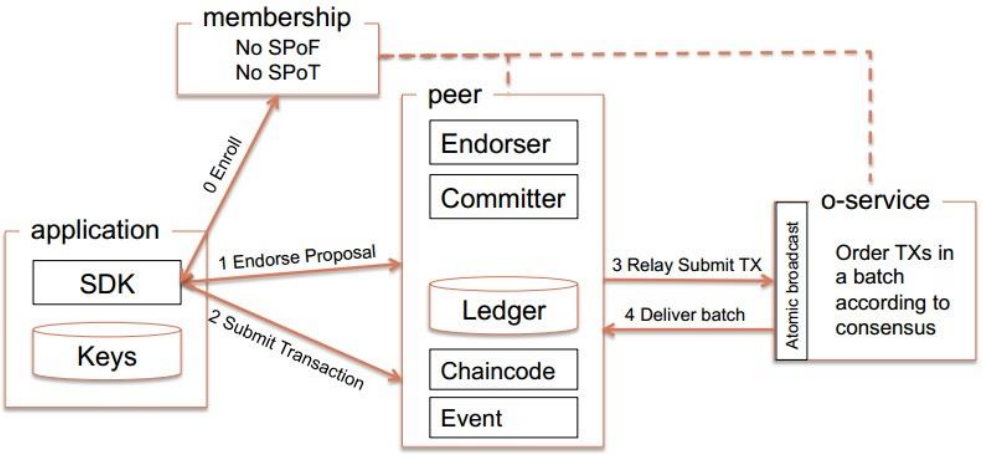


图 2-11 Hyperleger Fabric1.1.0 的系统架构

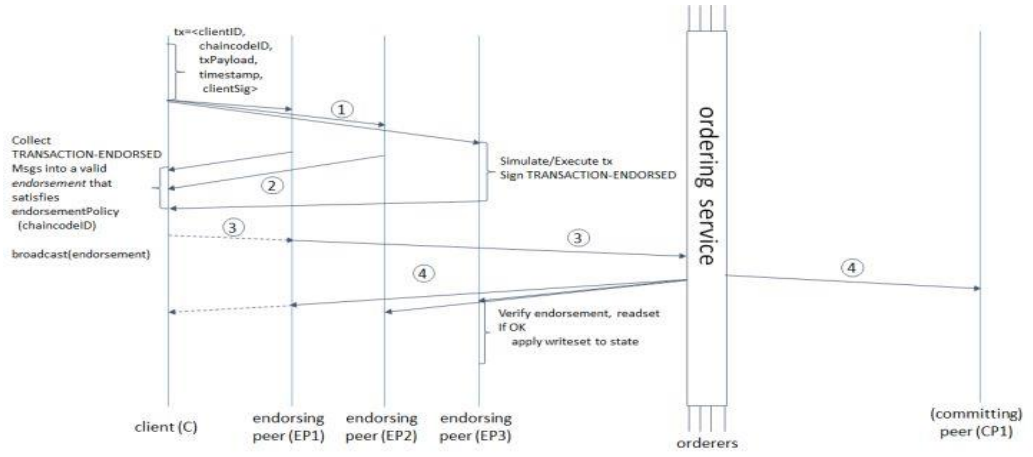


图 2-12 Hyperleger Fabric1.1.0 的交易流程图

在本项目中，我们根据我们系统的需求，设计了区块链的几种不同类型节点的名称以及它们对应的功能，如表 2-6 所示。

表 2-6 Hyperleger Fabric 网络组成

名字	功能
排序节点（Orderer）	先到先得的方式为网络上所有的 channel 作交易排序，根据需要，可以扩充
医疗中心 CA	医疗中心默认的证书管理组件，它向网络成员及其用户颁发基于 PKI 的证书。CA 为每个成员颁发一个根证书（rootCert），为每个授权用户颁发一个注册证书（eCert），为每个注册证书颁发大量交易证书（tCerts）。
医疗中心节点	上传新发现的致病基因，需要得到认证
计算节点 CA	同医疗中心 CA
计算节点	负责使用智能合约，计算医疗数据提供的加密致病基因和普通用户的基因的交际，并把结果返回给用户
背书节点	负责模拟交易，并提供背书。当一个排序节点接收一个交易后，就会调用与该交易的智能合约相关的背书规则验证，来确定交易的有效性。为此，一个交易包含一个或多个来自背书节点的背书。

普通用户	客户端（网页）使用 SDK 加入区块链网络，不负责任何计算任务。使用自己的基因数据，通过计算节点的计算，得到自己的基因和标准基因库的差异

2.3.2.2 智能合约实现

Hyperleger 支持开发者使用 Golang 语言或者 Java 语言实现智能合约。本系统中，智能合约的开发是基因数据分析在区块链上实现的核心工作。合约首先会对合约触发方的身份进行验证。不允许的触发方将不能触发合约。下面给出智能合约的主要数据结构以及智能合约的设计。

我们首先给出智能合约中用到的 3 个数据结构：

表 2-7 Disease 数据结构

```
type Disease struct {
    Name []byte
    ID []byte
    Description []byte
    Suggestion []byte
    GeneList []Gene
}
```

Name 表示疾病名字；ID表示疾病的标示；Description表示疾病的详细描述；Suggestion表示对于有高风险的患者，如何降低发病可能性，GeneList表示一个基因列表，内容是所有关于此疾病的基因片段。

表 2-8 Gene 数据结构

```
type Gene struct {
    Name []byte
    Location []byte
    Value []byte
    Uploader []byte
    Reference [][]byte
}
```

Name表示基因名字；Location小时基因在人体染色体的具体位置，如1:69091-70008 (+)；Value表示该基因片段的标准值；Uploader表示上传的医疗机构；Reference表示有关该基因的论文集合。

表 2-9 CompareSession 数据结构

```
type CompareUnit struct {
    SessionId []byte
    D [3][]byte
    Ready bool
    Accomplished bool
    T1    []byte
    T2    []byte
}
```

SessionId表示基因比较的id，每次比较均生成一个新的Sessionid；D表示长度为3的数组，内容是用户上传的3个补充参量集合；Ready表示医疗检测机构是否提交加密集合T_2；Accoompanished表示计算节点是否完成了交集计算；T1表示用户上传的加密集合T_1，T2表示医疗检测机构上传加密集合T_2。基因数据的交集比对完成后，Accomplished 参量为true，代表交易计算已经完成，计算节点不会再参与计算。

利用上述的数据结构，在本系统中设计了3个智能合约实现本系统的功能，智能合约及其对应功能如表2-10所示。

表2-10 智能合约设计

智能合约	功能
Payment	实现代币发放
Gene	实现基因数据计算
Calculation	关于计算节点的操作

对于上述的智能合约，每个智能合约的主要函数及其函数对应功能如表2-11—表2-12所示。

表2-11 Gene智能合约的主要函数及其功能

函数	功能	允许的使用方
psi_gene_compare	使用BPSI算法,进行基因比较	普通用户
psi_retrieve	查询之前的psi比对结果	普通用户
disease_search	查询的与该疾病相关联的基因库（加密）	普通用户
disease_gene_upload	上传新发现的致病基因	医疗数据节点
disease_info_upload	上传疾病的详细信息	医疗数据节点
disease_ls	数据库中所有支持比对的疾病	普通用户
disease_info	查询疾病的详细信息，如基因的上传来自于的医疗机构，相关论文的索引等	普通用户，医疗数据节点
disease_suggestion	查询疾病的相关建议	普通用户，医疗数据节点
pk_upload	医疗机构公开或更新其公钥	医疗数据节点

表2-12 Calculation智能合约中的主要函数

函数	功能	允许的使用方
calculation_claim	用户在基因比对完后，对比对结果的正确性进行验证，根据验证结果，对计算节点时候诚实做出评价	使用该计算节点进行比对的普通用户
calculation_ls	查询所有的计算节点	普通用户
calculation_history	查询计算节点的使用频率和评价历史	普通用户

特别的，在这里我们给出 psi_gene_compare 函数的参量说明。psi_gene_compare 函数作为几个智能合约的核心函数，有必要进行详细的说明。

表 2-13 psi_gene_compare 函数的参数说明

```
psi_gene_compare {
    sessionId []byte
    T1 []byte
    K []byte
    D0 []byte
    D1 []byte
    D2 []byte
}
```

用户触发智能合约中的 psi_gene_compare 函数后，需要上传 sessionId 参量。sessionId 表示交易唯一标识；T1 是用户上传的加密集合T_1；K 表示用医疗节点的公钥P_1加密的伪随机加密的密钥$F_{P_1}(K)$；D0、D1、D2 表示用户生成的补充参量集合。

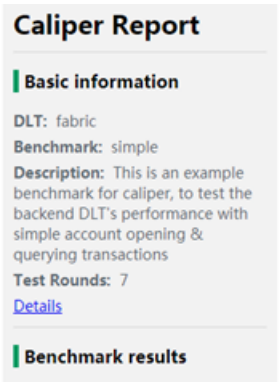
第三章 作品测试与分析

对于本项目中设计的基于区块链的隐私保护基因数据分析系统，本章将通过医疗检测机构上传大量数据的性能测试、致病基因交集计算的区块链承压节点测试以及传统的隐私保护交集(PSI)协议与基于区块链的隐私保护交集协议(BPSI)的性能比较测试这 3 个方面测试。测试结果表明，医疗检测机构上传海量数据不超过上限时，区块链执行智能合约效率相差无几、区块链的分布式结构提升了系统的效率、以及在协议模型安全性更高的情况下，运行时间相比最快的协议相差不大。

本此测试采用 Hyperleger fabric 实现区块链网络，并部署在服务器上，所有节点运行在 docker 容器中，通过服务器的端口映射使客户端进行访问。客户端使用 ubuntu 17.10 主机，用以对区块链发起高频次的请求。

本此测试采用的共识协议为 POS 机制，在联盟链上实现。现实中的区块链应用大多使用 POW 机制，如比特币、以太坊等，每个区块被确认有效性的时间约为 3-4 个区块生成的时间，按照一个区块生成的平均时间 15 分钟进行计算，需近一个小时，耗时巨大，无法进行测试。所以我们选择 POS 机制作为测试环境，区块的生成速度由交易数量决定。需要测试大量请求下，区块链共识机制和计算节点在进行大量基因比对时的交易成功率、延迟以及流量统计。

本次测试采用 Caliper 框架进行压力测试。Caliper 是 Hyperledger 开发的区块链性能基准框架，它允许用户使用预定义的用例，测试不同的区块链解决方案，并获得性能测试结果，目前支持 Fabric，sawtooth，Iroha 等区块链框架的性能测试。其测试结果界面如图 3-1 所示。



Summary

Test	Name	Succ	Fail	Send Rate	Max Latency	Min Latency	Avg Latency	75%ile Latency	Throughput
1	uploadGene	1000	0	49 tps	2.31 s	0.75 s	1.51 s	1.76 s	47 tps
2	uploadGene	1000	0	100 tps	3.45 s	0.91 s	2.23 s	2.54 s	85 tps
3	uploadGene	1000	0	149 tps	7.11 s	1.68 s	4.62 s	5.70 s	91 tps
4	queryGene	5000	0	100 tps	2.72 s	0.60 s	1.51 s	1.77 s	97 tps
5	queryGene	3189	1811	198 tps	30.04 s	1.67 s	17.98 s	25.80 s	70 tps
6	compareGene	5000	0	100 tps	2.54 s	0.63 s	1.57 s	1.82 s	97 tps
7	compareGene	2922	2078	189 tps	30.33 s	2.38 s	20.20 s	28.04 s	56 tps

图 3-1 Caliper 测试结果界面

3.1 测试环境

3.1.1. 客户端

表 3-1 客户端测试设备参数

项目	配置参数值
操作系统	Ubuntu17.10
CPU	Intel(R) Core(TM) i5-4200H CPU @2.80GHz
内存	4.00GB DDR_3
磁盘	500G 7200r
网络连接	以太网

3.1.2. 服务器端

表 3-2 服务端测试设备参数

项目	配置参数值
操作系统	Ubuntu 16.04 服务器版
CPU	Intel(R) Core(TM) i7-4720HQ CPU @2.60GHz
内存	8.00GB DDR_3
磁盘	500G 7200r
网络连接	以太网

3.2 性能实验

本项目使用的基因数据来自癌症体细胞突变目录 Cosmic[34]，Cosmic 是目前最大的人类癌症相关的体细胞突变信息的来源。本测试中基因总量为 32000 个，全部来自医疗机构的大规模基因组筛选数据和其他数据库，如 TCGA、ICGC 等。本测试中选择部分基因数据作为性能测试的数据集。

```
>FAM138A ENST00000417324 1:35138-35736(-)
atgctgctgactatagagacaaagtctcactatgttgctcaggctggtcttgaactcctggcctcaagcgatcctcccac
ctcagcctcccaaagtgttgggattatagacatgagccactgcacctggccgaccttgggcaagttcttaaacccttcaa
agcctcatttttctccaatcacaaaagggaaagatggtaatattttccccaccaaattcttgtcggatgccctcacagaa
ttgagattatgtacgtaa


>ENSG00000197490 ENST00000359752 1:37397-54936(+)
atgttgctcaccttatgggcagggtctcactatgttgctgaggctggtctcaaactcctgacctcaagcaatctgtctgc
ttcagcctcccaagtagctgagaatacagggacaagccattgcacctga


>OR4F5 ENST00000326183 1:69091-70008(+)
atggtgactgaattcatttttctgggtctctctgattctcaggaactccagaccttcctatttatgttgtttttgtatt
ctatggaggaatcgtgtttggaaaccttcttattgtcataacagtggtatctgactcccaccttcactctcccatgtact
tcctgctagccaacctctcactcattgatctgtctctgtcttcagtcacagcccccaagatgattactgacttttcagc
cagcgcaaagtcatctctttcaagggctgccttgttcagatatttctccttcacttctttggtgggagtgagatggtgat
cctcatagccatgggctttgacagatatatagcaatatgcaagcccctacactacactacaattatgtgtggcaacgcat
gtgtcggcattatggctgtcacatggggaattggctttctccattcggtgagccagttggcgtttgccgtgcacttactc
ttctgtggtcccaatgaggtcgatagtttttattgtgaccttcctagggtaatcaaacttgcctgtacagatacctacag
gctagatattatggtcattgctaacagtggtgtgctcactgtgtgttcttttgttcttctaatcatctcatacactatca
tcctaatgaccatccagcatcgccctttagataagtcgtccaaagctctgtccactttgactgctcacattacagtagtt
cttttgttctttggaccatgtgtctttatttatgcctggccattccccatcaagtcattagataaattccttgctgtatt
ttattctgtgatcacccctctcttgaacccaattatatacacactgaggaacaaagacatgaagacggcaataagacagc
tgagaaaatgggatgcacattctagtgtaaagttttag
```

图3-2 Fasta格式的体细胞突变信息

3.2.1 上传致病基因数据的性能实验

上传致病基因是本系统进行基因数据分析时，用户和医疗检测机构双方的第一步骤。针对双方上传基因内容的安全性和有效性，在 2.2 章节中已经进行了详细的定义与分析，在此不再赘述。本部分是对上传大量基因数据库时，区块链的稳定性进行抗压测试，判断区块链是否能正常运行智能合约。本测试的对象是医疗检测机构。下面是本测试的固定参量：

表 3-3 医疗检测机构上传大量致病基因数据的性能实验的固定参量

并发的医疗检测机构数量：5
每秒请求数（固定速率）：50
总请求数量：5000

表 3-4 CPU 占用率和运行时间分析

基因条目	成功率	平均内存占用(M)	平均延迟(s)
1000	100%	27.5	1.74
10000	100%	71.6	8.00
100000	2%	100.6	27.38

根据表 3-4 的测试结果，在 10000 条基因的压力测试下，本系统的区块链系统都能工作正常，但是提交延迟会大幅增加。如果要导入大规模的基因数据库，可以进行分次上传。

表 3-4 中的最后一栏，上传 100000 条基因时仅仅出现 2%的成功率。这一情况出现的原因是 docker 容器中的一台共识节点容器的 CPU 占用率超过了 100%，导致程序奔溃。对于这一问题，解决方案包括：使用真实的服务器担任共识节点，突破 CPU 和内存的局限、使用多台排序节点和共识节点，增加网络的容错率和使计算量均分等。

接下来对多次上传基因的性能进行测试，模拟的场景是医疗机构上传新基因至区块链。下面是本次测试的固定参量。

表 3-5 多次上传基因的性能测试的固定参量

每秒请求数（固定速率）：50
每次上传基因片段长度：1 条基因
总请求数量：5000

表 3-6 多次上传基因的性能测试

医疗机构节点	成功率	平均内存占用(M)	平均延迟(s)
50	100%	47.1	5.26
70	82.5%	62.5	12.78
100	1.8%	77.6	25.53

根据表 3-6 的测试结果，可以看出当并发节点在 50 时，区块链的抗压性很好，随着并发节点的增加，抗压性逐渐减弱。值得说明的是，由于计算资源有限，我们只在单台服务器上进行了实验，因此节点的并发性受到性能限制。

3.2.2 致病基因交集计算的性能试验

致病基因交集计算是基因数据分析系统中的核心步骤，本测试主要是对进行致病基因交集计算的区块链承压节点测试，通过增大上传基因的维数对平均流量、CPU 占用率和运行时间等变化进行分析，判断系统能否正常运行。下面是本测试的固定参量：

表 3-7 致病基因交集计算的性能试验的固定参量

每秒请求数（固定速率）：50
计算节点数：4
总请求数量：5000
医疗检测机构的致病基因库维度：5000

表 3-8 区块链承压节点测试

基因维数	CPU 占用率	流量(M)	运行时间(s)	内存占用(M)
10	33.81%	2.8	20.20	11.50
50	36%	3.4	30.50	11.55
100	41%	4.6	40	12.10
200	53%	7.9	60	12.59
500	70%	12.5	80	14.67
800	112%	15.7	>120	15.67

通过表 3-8 可以看出，计算时间的增长是影响用户交易流畅性的首要影响因素，带宽和内存占用都还在合理范围内。

接下来考虑区块链网络中的计算节点扩充对网络性能的影响。当有空闲计算资源的主机出于商业利益目的加入区块链网络后，判断网络的性能是否进行提升。在这个测试场景下，我们不再关注单个计算节点的内存开销和计算开销，关注的是区块链的分布性能给系统容量带来的扩充容量，所以下一测试检验 PSI 交易是否全部完成。本测试中，依然指定每个客户端发起 200 次请求。下面是本测试的固定参量。

表 3-9 致病基因交集计算的性能试验的固定参量

每秒请求数（固定速率）：50
总请求数量：5000
用户的待检测致病基因维度：1000
医疗检测机构的致病基因库维度：10000

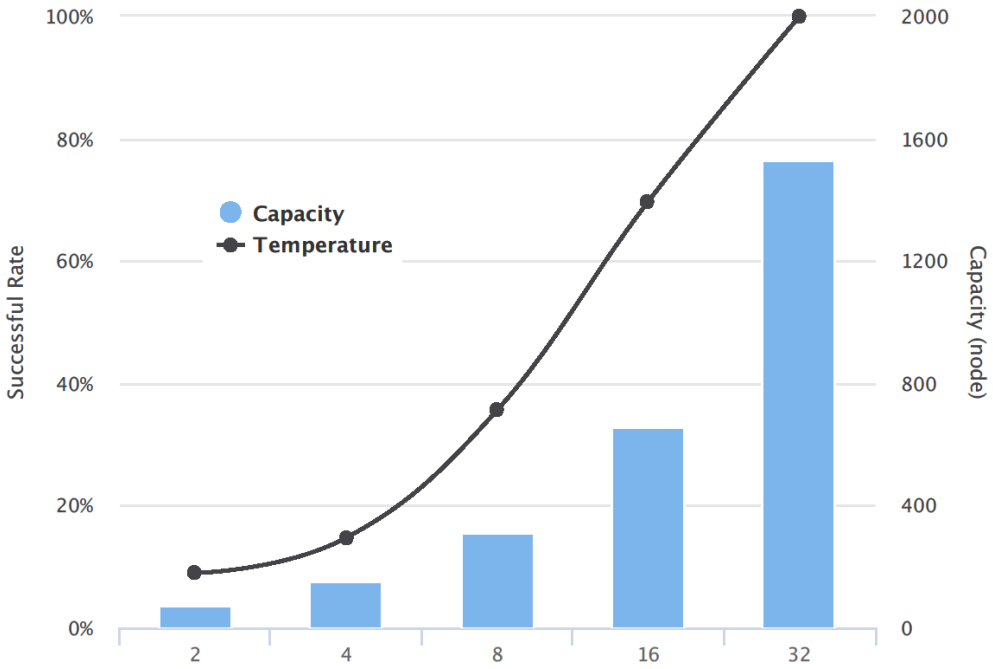


图 3-3 区块链的分布性能给系统容量扩充性测试

通过图 3-3 可以看出，随着计算节点数目的增加，系统的计算容量增长速率高于线性，证明了区块链的分布性将带来大量的计算资源。

3.2.3 使用隐私保护机制的性能试验

相比 3.2.2 章节考虑的是对进行致病基因交集计算的区块链承压节点测试，本测试考虑的是不同的算法协议的性能比较测试。由于隐私保护机制的引入，交集算法的性能必定会遇到一定的牺牲，但是我们希望算法的复杂性增加有限。本实验比对传统 PSI 算法协议包括基于 Hash 的 PSI 协议、基于布隆过滤器的 PSI 协议与我们本系统中应用的 BPSI 协议在区块链中的性能表现。

基于 Hash 的 PSI 协议是医疗检测机构在数据节点上传致病基因片段的单向散列值，与用户上传的待检测基因片段的单向散列值进行比对，计算节点在两个散列集合中寻找交集，实现基因数据分析。基于 Hash 的 PSI 协议从密码学的角度来看是安全性较低的算法协议，通过暴力破解或彩虹攻击均可造成一定的隐私泄露，但鉴于 Hash 算法的计算简洁性，本测试将基于 Hash 的 PSI 协议作为性能测试比对的基准值。

表 3-10 使用隐私保护机制的性能测试的固定参量

并发的医疗机构数量：5
每秒请求数（固定速率）：50
计算节点数：4
用户的待检测致病基因维度：1000
医疗检测机构的致病基因库维度：10000

表 3-11 协议算法性能测试

协议算法	运行总时间	内存占用	区块链平均流量	CPU 占用率
BPSI 协议	1.32s	45M	96M	25%
基于 hash 的 PSI	1.21s	23M	42M	22%
基于布隆过滤器的 PSI	1.65s	78M	126M	37%

通过表 3-11 的运行结果分析可得，本系统采用的 BPSI 协议算法在运行时间略高于基于 Hash 的 PSI 协议 10%，并不会对系统造成很大的延缓影响，同时由于 2.2.3 与 2.2.4 章节中证明了本协议在恶意模型下的安全性与有效性，综合上述分析，说明了本系统的安全性与高效性。

3.3 结果分析

本章对本系统进行了3个模块的测试实验，包括：医疗检测机构上传大量数据的性能测试、致病基因交集计算的区块链承压节点测试、以及传统的隐私保护交集(PSI)协议与基于区块链的隐私保护交集协议(BPSI)的性能比较测试。测试结果表明，医疗检测机构上传海量数据不超过上限时，区块链执行智能合约效率相差无几、区块链的分布式结构提升了系统的效率、以及在协议模型安全性更高的情况下，运行时间相比最快的协议相差不大。经过上述实验，说明了本系统的可靠性，高效性和可拓展性。

第四章 创新性说明

本作品创新性的提出了基于区块链的隐私保护数据分析系统，旨在实现海量基因数据时代下集隐私保护、基因分析、疾病诊断于一体的一套较为完整的基因数据分析系统。通过设计了基于区块链的隐私保护交集(BPSI)协议，并给出该协议在恶意模型下的安全性机制、公平性机制以及仲裁机制；利用区块链的抗合谋性、匿名性、分布式结构等特性，解决了现有方案中服务端不可信、通过交集数据可以进行追踪、大规模数据处理速度较慢等一系列的缺陷，同时我们改进了区块链的框架，修改了存储结构与内容，实现了基于区块链的隐私保护数据基因分析系统。最后我们基于区块链实现了系统的客户端与服务端，完成了系统的构建与测试，说明了本项目的可靠性，高效性和可拓展性。

作品的几大创新点分别是：

1. 设计了满足抗合谋性、匿名性与高效性的基于区块链的隐私保护数据分析系统
   - 利用区块链的抗合谋性，解决了传统服务端不可信问题
   - 利用区块链的匿名性，解决了致病基因的不可追踪性
   - 利用区块链的分布式结构，并行化处理提升了系统的效率和扩展性
   - 设计并实现了集客户端、服务端以及对应接口的完整系统
2. 改进了区块链的框架，重新设计了区块中的存储结构及存储内容
3. 实现了基因数据分析过程的可视化
4. 设计了基于区块链的隐私保护交集计算(BPSI)协议，实现基因数据的比对分析
   - 给定了恶意模型下协议的安全性机制、有效性机制以及仲裁机制
5. 利用智能合约实现双方的费用自动结算，促进用户的使用，增强系统时效型
6. 可以将基因切断处理，通过智能合约放在链上，医疗检测机构去响应，实现功能上的外包，提升系统的性能

4.1 实现集抗合谋性、匿名性与高效性的隐私保护基因数据分析系统

随着基因测序技术的快速发展，获取自己的基因组序列数据变得更加便捷，如何在保护隐私的情况下进行基因数据的处理是我们关注的重要问题。

传统的 PSI 协议无法处理大规模数据；云辅助的 PSI 协议存在着云端与参与方合

谋的隐患，基于上述问题，我们设计了基于区块链的的隐私保护基因数据分析系统，利用区块链的抗合谋性，解决了传统服务端不可信问题；利用区块链的匿名性，解决了致病基因的不可追踪性；利用区块链的分布式结构，并行化处理提高了系统的效率和扩展性，解决了现有系统存在的问题，实现了集抗合谋性、匿名性和高效性一体化的隐私保护基因数据分析系统。

4.2 可视化基因数据分析过程

在本系统中，我们设计了基于区块链的隐私保护交集(BPSI)协议实现基因的比对分析。每次密钥的协商、基因数据的上传、基因数据的比对都会改变链的结构，基于此，我们使用 ECharts 模拟链的结构改变，从而实现了基因数据上传、基因数据更新、基因数据比对等的可视化实现。并且将可视化的分布式结构写入我们用 Express 框架设计的客户端的界面上，给使用者带来良好的用户体验。

4.3 设计了基于区块链的隐私保护交集(BPSI)协议

相比传统的隐私保护交集(PSI)协议和云辅助的 PSI 协议，区块链存在着所有链上信息公开透明这一特点，所以我们需要根据区块链的特性设计新的 PSI 协议，即基于区块链的隐私保护交集(BPSI)协议。我们选择了效率较高、安全性较高的伪随机加密作为加密方案，并通过增加随机扰动、补充参量、信息复制等改进，使得协议的安全性得到了进一步的增强。同时我们给出了恶意模型下的安全性机制、有效性机制以及仲裁机制，实现了在任意模型下的隐私保护基因数据交集计算。

4.4 具有项目时效型且实现功能外包

区块链上目前最为完善的就是关于电子货币的研究。在本系统中，用户进行基因数据分析，医疗检测机构会很快得到报酬；同时若医疗检测机构欺骗了用户，用户也可以短时间内得到补偿，这是现有系统中无法实现的。这一切均由智能合约进行自动结算，同时用户可将基因数据分析进行外包，通过智能合约放在链上，医疗检测机构去响应，通过经济手段的时效性推动项目的广泛传播。

第五章 总结

5.1 作品工作总结

本项目研究了隐私保护交集(PSI)协议的设计方法，分析了现有方案中传统 PSI 技术以及云辅助 PSI 技术的不足，提出了基于区块链的隐私保护交集(BPSI)协议，并给出了针对恶意模型下安全性、有效性的算法改进；同时根据系统的需求，改进了区块链的存储结构内容的框架，实现了基于区块链的隐私保护数据基因分析系统。最后基于区块链实现了客户端和服务端，并将基因数据分析后的分布式结构变化进行了可视化处理，设计了用户体验感良好的客户端界面；最后对系统进行了性能测试和功能测试，证明了本项目的可靠性，高效性和可拓展性，给出了海量基因数据时代下集隐私保护、基因分析、疾病诊断于一体的一套较为完整的解决方案。

本项目的主要贡献是创新性的提出并实现了基于区块链的隐私保护基因数据分析系统，解决了现有模型不抗合谋、效率较低、可基因追踪等缺陷，设计了符合系统的基于区块链的隐私保护交集(BPSI)协议，基于区块链实现了客户端与服务端，通过性能测试和功能测试证明了项目的可行性与高效性，为基因数据分析领域的发展做出巨大的贡献。

5.2 未来作品展望

(1)本项目由于采用区块链的抗合谋性、分布式结构的可拓展性以及匿名性，解决了现有模型存在的缺陷，但是使用区块链框架也会带来项目的开销略大，如何设计复杂度更低，效率更优的算法是我们接下来的主要问题。

(2)本项目目前对基因数据分析仅限于基因数据的比对，通过 BPSI 协议进行的交集计算，但基因数据分析不只限于比对功能，如何将功能进一步拓展也将是我们接下来考虑的主要问题。

(3)本系统目前在联盟链上实现，未来可以推广在公有链上实现系统。

参考文献

[1] Baldi P, Baronio R, De Cristofaro E, et al. Countering gattaca: efficient and secure testing of fully-sequenced human genomes[C]//Proceedings of the 18th ACM conference on Computer and communications security. ACM, 2011: 691-702.

[2] De Cristofaro E, Faber S, Gasti P, et al. Genodroid: are privacy-preserving genomic tests ready for prime time?[C]//Proceedings of the 2012 ACM workshop on Privacy in the electronic society. ACM, 2012: 97-108.

[3] Ayday E, Raisaro J L, Hengartner U, et al. Privacy-preserving processing of raw genomic data[M]//Data Privacy Management and Autonomous Spontaneous Security. Springer, Berlin, Heidelberg, 2014: 133-147.

[4] De Cristofaro E, Faber S, Tsudik G. Secure genomic testing with size-and position-hiding private substring matching[C]//Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society. ACM, 2013: 107-118.

[5] De Cristofaro E, Faber S, Tsudik G. Secure genomic testing with size-and position-hiding private substring matching[C]//Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society. ACM, 2013: 107-118.

[6] Johnson A, Shmatikov V. Privacy-preserving data exploration in genome-wide association studies[C]//Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013: 1079-1087.

[7] Ayday E, Raisaro J L, Hubaux J P. Personal use of the genomic data: privacy vs. storage cost[C]//Global Communications Conference (GLOBECOM), 2013 IEEE. IEEE, 2013: 2723-2729.

[8] Ayday E, Raisaro J L, Hubaux J P, et al. Protecting and evaluating genomic privacy in medical tests and personalized medicine[C]//Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society. ACM, 2013: 95-106.

[9] Garay J, Kiayias A, Leonardos N. The bitcoin backbone protocol: Analysis and applications[C]//Annual International Conference on the Theory and Applications of

Cryptographic Techniques. Springer, Berlin, Heidelberg, 2015: 281-310.

[10] Sasson E B, Chiesa A, Garman C, et al. Zerocash: Decentralized anonymous payments from bitcoin[C]//Security and Privacy (SP), 2014 IEEE Symposium on. IEEE, 2014: 459-474.

[11] Feige U, Fiat A, Shamir A. Zero-knowledge proofs of identity[J]. Journal of cryptology, 1988, 1(2): 77-94.

[12] Rivest R L, Adleman L, Dertouzos M L. On data banks and privacy homomorphisms[J]. Foundations of secure computation, 1978, 4(11): 169-180.

[13] Freedman M J, Nissim K, Pinkas B. Efficient private matching and set intersection[C]//International conference on the theory and applications of cryptographic techniques. Springer, Berlin, Heidelberg, 2004: 1-19

[14] Azar Y, Broder A Z, Karlin A R, et al. Balanced allocations[J]. SIAM journal on computing, 1999, 29(1): 180-200.

[15] Freedman M J, Hazay C, Nissim K, et al. Efficient set intersection with simulation-based security[J]. Journal of Cryptology, 2016, 29(1): 115-155.

[16] Pagh R, Rodler F F. Cuckoo hashing[C]//European Symposium on Algorithms. Springer, Berlin, Heidelberg, 2001: 121-133.

[17] Kissner L, Song D. Privacy-preserving set operations[C]//Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2005: 241-257.

[18] Dachman-Soled D, Malkin T, Raykova M, et al. Efficient robust private set intersection[C]//International Conference on Applied Cryptography and Network Security. Springer, Berlin, Heidelberg, 2009: 125-142.

[19] Kamara S, Mohassel P, Raykova M, et al. Scaling private set intersection to billion-element sets[C]//International Conference on Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, 2014: 195-215.

[20] 陈振华，李顺东，黄琼，等. 非加密方法安全计算两种集合关系[J]. 软件学报，2018, 29(2): 473-482.

[21] Hazay C, Nissim K. Efficient set operations in the presence of malicious adversaries[C]//International Workshop on Public Key Cryptography. Springer, Berlin, Heidelberg, 2010: 312-331.

[22] Hazay C, Lindell Y. Efficient protocols for set intersection and pattern matching with security against malicious and covert adversaries[C]//Theory of Cryptography Conference. Springer, Berlin, Heidelberg, 2008: 155-175.

[23] Jarecki S, Liu X. Efficient oblivious pseudorandom function with applications to adaptive OT and secure computation of set intersection[C]//Theory of Cryptography Conference. Springer, Berlin, Heidelberg, 2009: 577-594.

[24] Dodis Y, Yampolskiy A. A verifiable random function with short proofs and keys[C]//International Workshop on Public Key Cryptography. Springer, Berlin, Heidelberg, 2005: 416-431.

[25] Camenisch J, Shoup V. Practical verifiable encryption and decryption of discrete logarithms[C]//Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2003: 126-144.

[26] Jarecki S, Liu X. Fast secure computation of set intersection[C]//International Conference on Security and Cryptography for Networks. Springer, Berlin, Heidelberg, 2010: 418-435.

[27] De Cristofaro E, Tsudik G. Practical private set intersection protocols with linear complexity[C]//International Conference on Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, 2010: 143-159.

[28] De Cristofaro E, Kim J, Tsudik G. Linear-complexity private set intersection protocols secure in malicious model[C]//International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2010: 213-231.

[29] De Cristofaro E, Tsudik G. Experimenting with fast private set intersection[C] // International Conference on Trust and Trustworthy Computing. Springer, Berlin, Heidelberg, 2012: 55-73.

[30] Kerschbaum F. Collusion-resistant outsourcing of private set intersection[C] // Proceedings of the 27th Annual ACM Symposium on Applied Computing. ACM, 2012: 1451-1456.

[31] Liu F, Ng W K, Zhang W, et al. Encrypted set intersection protocol for outsourced datasets[C]//Cloud Engineering (IC2E), 2014 IEEE International Conference on. IEEE, 2014: 135-140.

[32] Kamara S, Mohassel P, Raykova M, et al. Scaling private set intersection to billion-element sets[C]//International Conference on Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, 2014: 195-215.

[33] Abadi A, Terzis S, Dong C. O-PSI: delegated private set intersection on outsourced datasets[C]//IFIP International Information Security Conference. Springer, Cham, 2015: 3-17.

[34] https://cancer.sanger.ac.uk/cosmic