An Efficient and Secure Nonlinear Programming Outsourcing in Cloud Computing

M.Madhura
PG Scholar
Department of Computer Science and Engineering
Karpagam University,INDIA

R.Santosh
Assistant Professor
Department of Computer Science and Engineering
Karpagam University,INDIA

ABSTRACT
Cloud Computing provides a appropriate on-demand network access to a shared pool of configurable computing resources which could be rapidly deployed with much more great efficiency and with minimal overhead to management. This paper deals with the secure outsourcing of nonlinear programming. It provides a practical mechanism design which fulfils input/output privacy, cheating resilience, and efficiency. In the proposed approach practical efficiency is achieved by explicit decomposition of NLP into NLP solvers running on the cloud and private NLP parameters owned by the customer. When compared to the general circuit representation the resulting flexibility allows exploring appropriate security/efficiency trade-off via higher-level abstraction of NLP computations. It is possible to construct a set of effective privacy-preserving transformation techniques for any problem, by framing a private data possessed by the client for NLP problem as a combination of matrices and vectors, which allow customers to transform original NLP problem into some arbitrary value while defending sensitive input or output information. To confirm the computational result, the fundamental duality theorem of NLP computation should be explored and then derive the essential and adequate constraints that a accurate result must satisfy. Such a result verification mechanism is very competent and suffers close-to-zero extra cost on both cloud server and customers..

General Terms
Security.

Keywords
Non-linear Programming, Cloud Computing, Privacy Preservation, Information security, Outsourcing.

1. INTRODUCTION
Cloud Computing helps clients with inadequate computational resources to outsource their huge complex computational works to the cloud and economically adore the computational efficiency, storage, bandwidth, and even suitable software which can be used in a pay-on-use method. It has superior potentiality of exhibiting robust computational power to the society at a reduced price[8].

The term Cloud Computing is derived considering the Internet as a cloud which provides various services. Cloud computing may be defined as the dynamic provisioning of Information Technology resources such as hardware, software, or services from third parties over a network. Clouds may also be considered as hardware-based services offering compute, network and storage capacity where hardware management is highly hidden from the end user. The end users gain infrastructure costs as variable Operating Expenditures and infrastructure capacity is highly flexible. The cloud model differs from traditional techniques in which customers do not provide their own IT assets to be managed. Instead they login into the cloud, which serves as an internal data centre or system providing the same functions. In Software-as-a-Service (SaaS) the software is developed and hosted by the SaaS vendor, which is accessible by the end user over the Internet.

Unlike traditional applications that users install on their computers or servers, SaaS software is owned by the vendor and runs on computers in the vendor's data center (or a collocation facility). Broadly speaking, all customers of a SaaS vendor use the same software: these are one-size-fits-all solutions. Well known examples are Salesforce.com, Google's Gmail and Apps, instant messaging from AOL, Yahoo and Google, and Voice-over Internet Protocol (VoIP) from Vonage and Skype. The great advantage of cloud computing is "elasticity": the ability to add capacity or applications almost at a moment's notice. Companies buy exactly the amount of storage, computing power, security and other IT functions that they need from specialists in data-center computing. They get sophisticated data center services on demand, in only the amount they need and can pay for, at service levels set with the vendor, with capabilities that can be added or subtracted at will. Several trends are opening up the era of Cloud Computing, which is an Internet-based development and use of computer technology.

The ever cheaper and more powerful processors, together with the software as a service (SaaS) computing architecture, are transforming data centers into pools of computing service on a huge scale. Cloud Computing has been envisioned as the next generation architecture of IT enterprise, due to its long list of unprecedented advantages in the IT history. On-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk

As a disruptive technology with profound implications, Cloud Computing is transforming the very nature of how businesses use information technology. One fundamental aspect of this paradigm shifting is that data is being centralized or outsourced into the Cloud. In Cloud Computing, the remotely stored electronic data might not only be accessed but also updated by the clients, e.g., through block modification, deletion, insertion, etc. Unfortunately, the state of the art in the context of remote data storage mainly focus on static data files and the importance of this dynamic data updates has received limited attention so far.

Security is the major problem that prevents the extensive usage of this promising computational model, despite its great

benefits. Mainly the client faces the security threat that their private data are consumed and created during the computation .i.e. when the data are outsourced for the computation over cloud. So, to overcome this security constraint the cloud is treated as an intrinsically insecure computing platform from the viewpoint of the cloud customers, a mechanism is planned which not only protects critical information by allowing calculations with encoded data, but also protect clients from malevolent actions by enabling them with the validation of the computation result. To design mechanisms that are almost effective remains a challenging problem, though general protected computation outsourcing was recently shown to be possible theoretically.

There have been several existing approaches provided in order to overcome the secure outsourcing of the programming in cloud computing. Based on Yao's garbled circuits and Gentry's breakthrough work on Fully Homomorphic Encryption (FHE)[2][3], a general result of secure computation outsourcing has been shown viable in theory, where the computation is represented by an encrypted combinational Boolean circuit that allows to be evaluated with encrypted private inputs[7].

However, its far from practical to apply this general mechanism to our daily computations, because the FHE operation is highly complex and also the pessimistic circuit sizes cannot be handled in practice when constructing original and encrypted circuits. In general this overhead motivates us to seek efficient solutions at higher abstraction levels when compared to the circuit representations for specific computation outsourcing problems. Even though some elegant designs have been proposed in the literature, on secure outsourcing of sequence comparisons, scientific computations and matrix multiplication[6] etc. still it is hardly possible to apply them directly in a practically efficient manner, especially for large problems.

1.1 Disadvantages:

In those approaches, either heavy cloud-side cryptographic computations, or multi-round interactive protocol executions, or huge communication complexities, are involved. In short, practically efficient mechanisms with immediate practices for secure computation outsourcing in cloud are still missing.

In our proposed approach we are dealing with the Sequential Quadratic programming method of NLP [1] [5] [9]. Initially a framework is designed for the approach. After presenting the basic techniques we are extending the existing approach for the security strength of NLP outsourcing, we must be able to change the feasible region of original NLP and at the same time hide output vector x during the problem input encryption. Finally the proposed approach is evaluated for the security with the existing..

2. PAGE SIZE

There have been several existing approaches provided in order to overcome the secure outsourcing of the programming in cloud computing. Based on Yao's garbled circuits and Gentry's breakthrough work on fully homomorphic encryption (FHE), in theory a general result of secure computation outsourcing has been shown viable, where the computation is represented by an encrypted combinational Boolean circuit that allows to be evaluated with encrypted private inputs. However, practical application of this general mechanism to our daily computations would be far from practical, due to the extremely high complexity of FHE operation as well as the pessimistic circuit sizes that cannot be handled in practice when constructing original and encrypted circuits. This overhead in general solutions motivates us to seek efficient solutions at higher abstraction levels than the circuit representations for specific computation outsourcing problems. Although some elegant designs on secure outsourcing of scientific computations, sequence comparisons, and matrix multiplication etc. have been proposed in the literature, it is still hardly possible to apply them directly in a practically efficient manner, especially for large problems.

This mechanism brings cloud customer great computation savings from secure LP outsourcing as it only incurs $O(n^{\rho})$ for some $2 < \rho \leq 3$ local computation overhead on the customer, while solving a normal LP problem usually requires more than $O(n^3)$ time.

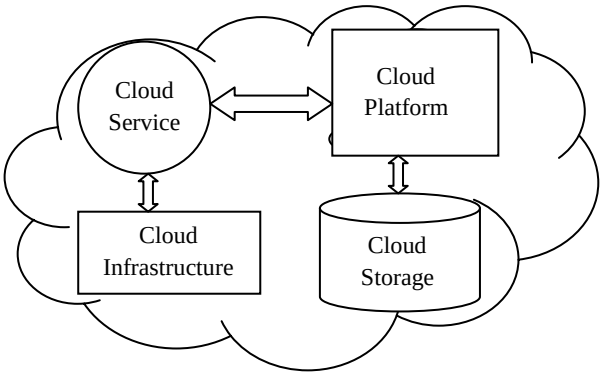

Fig. 1 Architecture of Secure outsource linear programming problems in cloud computing

3. LINEAR PROGRAMMING

Usually an optimization problem is formulated as a mathematical programming problem that needs the values for a set of decision variables to minimize or maximize an objective function to represent the cost subjected to a set of constraints. The objective function is an affine function of the decision variables and the constraints are a system of linear equations and inequalities for linear programming. A non-negative slack variable can be introduced to express a constraint in the form of linear equation, which is a linear inequality and the difference of two non-negative auxiliary variables can be expressed as a free decision variable.

4. PROPOSED SYSTEM

The main issue of the proposed approach is to handle the secure outsourcing of NLP computations in cloud computing, and provide such a practical mechanism design which fulfils input/output privacy, cheating resilience, and efficiency. In the proposed approach the cloud computer deals with the Sequential Quadratic programming method of NLP.

Usually, the Non-linear function is $f(x)$ under the non-linear inequality constraints

$$x \in \Re^n : \min\ f(x), g(x) \leq 0 \qquad (1)$$

where x is an n-dimensional parameter vector. The vector-valued function $g(x)$ defines m inequality constraints, $g(x) = (g_1(x), \ldots\ldots g_m(x))^T$. To simplify (1), the equality

constraints and upper or lower bounds of the variables are omitted. The problem is considered to be non-convex and non-linear in general. Sequential Quadratic Programming is considered a best method to solve smooth Non-Linear Optimization problems by using standard general purpose algorithms. The following assumptions are made

1. The problem is not too big.
2. Function values can be calculated within sufficient precision.
3. The problem is smooth and well scaled.

The sub-problems consist of strictly convex quadratic programming problems with inequality constraints obtained by linearizing the constraints and by approximating the Lagrangian function of (1) quadratically. The Sequential Quadratic Programming have the roots in unconstrained optimization. The main objective behind is to establish a quadratic approximation based on second order information with the goal to achieve a fast local convergence speed. The linearly constrained, strictly convex quadratic program must be solved in each iteration step by an available black box solver.

This is mostly used in structural optimization. The method is based on the observation that in some special cases, typical structural constraints become linear in the inverse variables. Although this special situation is rarely observed in practice, a suitable substitution of structural variables by inverse ones depending on the sign of the corresponding partial derivatives and subsequent linearization is expected to linearize constraints somehow.

To formulate the sub-problem, as said above, the process starts form the given iterates

$$x_k \in \Re^n$$

,an approximation of the solution,

$$u_k \in \Re^m$$

,an approximation of the vector of multipliers and

$$B_k \in \Re^{n \times n}$$

,an approximation of the Hessian of the Lagrange function. Then we obtain sub problem

$$y \in \Re^n : \min f^k(y), g^k(y) \le 0 \qquad (2)$$

by defining

$$f^k(y) = \frac{1}{2}(y - x_k)^T B_k (y - x_k) + \nabla f(x_k)^T (y - x_k) + f(x_k)$$

$$g_j^k = \nabla g_j(x_k)^T (y - x_k) + g_j(x_k),\ j = 1,\ldots m$$

It is clearly visible that the requirements of (2) are satisfied. The core idea is to approximate the second order information to get a fast final convergence speed. The most interesting feature of the Sequential Quadratic Programming method is the super linear convergence speed in the neighbourhood of the solution. That is

$$\| x_{k+1} - x^* \| < \gamma_k \| x_k - x^* \|$$

with $\gamma_k \to 0$.

In order to achieve practical efficiency, our mechanism design explicitly decomposes the NLP computation outsourcing into public NLP solvers running on the cloud and private NLP parameters owned by the customer. The resulting flexibility allows us to explore appropriate security efficiency tradeoff via higher-level abstraction of NLP computations than the general circuit representation. Initially a framework is designed for the approach. After presenting the basic techniques the existing approach of the cloud computer is being extended to the security strength of NLP outsourcing, it must be able to change the feasible region of original NLP and at the same time hide output vector x during the problem input encryption. Finally the proposed approach is evaluated for the security with the existing approaches.

4.1 Goals

To enable secure outsourcing of NLP under the aforementioned model, our mechanism design should achieve the following security and performance guarantees.

1) Correctness: Any cloud server must produce an output that can be decrypted and verified successfully by the customer, which faithfully follows the mechanism.

2) Soundness: An incorrect output should not be generated by any cloud server which could be decrypted and verified successfully by the customer with non-negligible probability.

5. EXPERIMENTAL RESULTS

An experimentation to compare the performance of Linear Programming over Non-Linear Programming in a Cloud Computing environment is performed. From the outcome it is visual that the NLP outsourcing consumes more computational time compared to that of the LP outsourcing. This is clearly visualized from the graph in Fig. 2.

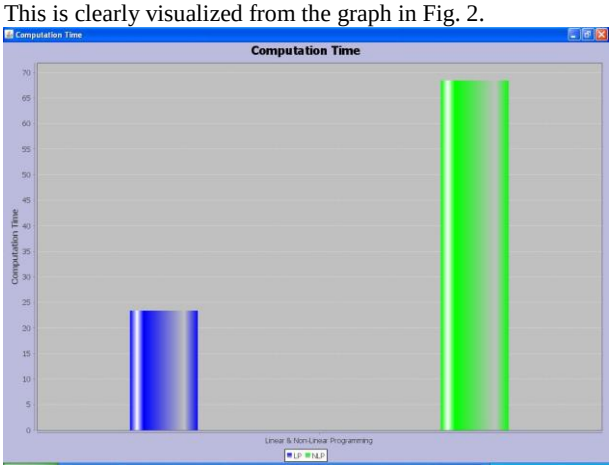

Fig 2. Comparison of Computation time –LP vs NLP

It is clearly visible that the NLP consumes more computation time due to the internal security computations. Though, the computations may be complex, it is evident from the results that the NLP computation outsourcing is more secure compared to that of the other existing models.

6. CONCLUSION

In our proposed approach they are dealing with the non-linear programming approach. In order to achieve practical efficiency, our mechanism design explicitly decomposes the NLP computation outsourcing into public NLP solvers

running on the cloud and private NLP parameters owned by the customer. The resulting flexibility allows us to explore appropriate security efficiency tradeoff via higher-level abstraction of NLP computations than the general circuit representation. Initially a framework is designed for the approach. After presenting the basic techniques they are proposing an enhanced technique for the security strength of LP outsourcing, they must be able to alter the feasibility region of the original LP and at the same time hide output vector x during the problem input encryption. Finally the proposed approach is evaluated for the security with the existing approaches. Here we present a method of Non Linear Programming named as Sequential Quadratic programming method. Sequential quadratic programming (SQP) methods are very well known and are considered as the standard general purpose algorithm for solving smooth nonlinear optimization problems at least under the following assumptions: (i) the problem is not too big, (ii) function and especially gradient values can be evaluated within sufficient precision, and (iii) the problem is smooth and well-scaled.

7. REFERENCES

[1] Cong Wang, Kui Ren, and Jia Wang, Secure and Practical Outsourcing of Linear Programming in Cloud Computing, Illinois Institute of Technology.

[2] Gentry C. 2009 Fully homomorphic encryption using ideal lattices. In STOC, 169-178.

[3] Smart N.P. and Vercauteren F. 2009 Fully homomorphic encryption with relatively small key and ciphertext sizes.

[4] Atallah M. and Frikken K. 2010 Securely outsourcing linear algebra computations. In Proceedings of ASIACCS, 48–59.

[5] Luenberger D. and Ye Y. 2008 Linear and Nonlinear Programming. 3rd ed.

[6] Coppersmith D. and Winograd S. 1987 Matrix multiplication via arithmetic progressions. In Proceedings of STOC'87, 1–6.

[7] Shamir A. 1979 How to share a secret. In Communications of ACM, vol. 22, no. 11, 612–613.

[8] Yu S., Wang C., Ren K., Lou W. 2010 Achieving secure, scalable, and fine-grained access control in cloud computing. In Proceedings of IEEE INFOCOM'10.

[9] Li J. and Atallah M.J. 2006 Secure and private collaborative linear programming. In Proceedings of Collaborate Communication.