Preprints, 10th IFAC International Symposium on
Advances in Automotive Control
Columbus, Ohio, USA, August 28-30, 2022

Secure Eco-Routing with private function evaluations

Bharatkumar Hegde, Chen-Fang Chang
General Motors Company, Detroit, MI
USA (Tel: 586-596-0334; e-mail: bharatkumar.hegde@gm.com).

Abstract: Ubiquity of connected devices ranging from cellphones to in-vehicle navigation systems has enabled information rich routing and navigation services. Eco-routing utilizes these infrastructure and data about the routes to ascertain and inform the driver the energy cost of traversing a route to their destination. An accurate energy consumption model of the vehicle traversing the route is essential to perform eco-routing effectively. Unfortunately, very accurate energy consumption models also contain operating strategies that their owners are disinclined to disclose publicly. We propose the use of partially homomorphic cryptosystem for private function evaluations to enable secure eco-routing. A novel way to encrypt the energy consumption model to enable secure eco-routing, methods for private evaluation of the encrypted energy consumption model, and the associated protocol are described. Practical considerations for implementing such a system are explored through software implementation.

Keywords: Private function evaluation, eco-routing, Paillier cryptosystem, partially homomorphic cryptosystems

1. INTRODUCTION

Energy consumption for transportation in the US is reported to be 26% total energy usage in 2020; 55% of which is accounted for by light-duty vehicles (EIA 2022). It remains one of the significant sources of greenhouse emissions and reduction in energy usage through improvement in efficiency of vehicles remains a priority. One of the efficiency improvement strategies is energy conscious routing of vehicles, commonly termed eco-routing (Barth et al. 2007), where a driver traverses the route with least energy consumption to get to their destination. Eco-routing is shown to be very effective in reducing trip energy cost; on average 5% and up to 15% energy reduction (Brown et al. 2014). This potential makes it a worthy candidate to pursue for widespread implementation and use. The computation of an eco-route requires the knowledge of the relationship between the road conditions on a route and the energy consumption of the vehicle being driven. The energy consumption model of the vehicle quantifies the energy efficiency of each route between two places.

Routing is widely used in the form of GPS-based navigation systems on vehicles and cell phones. Historically, the navigation system was implemented on-board with mapping and routing components built into a single product. Ubiquity of connected devices such as cell phones and data up link in automobiles has enabled decoupling of mapping and routing from step-by-step navigation; exemplified by Google Maps, Waze etc. In this new paradigm, a device such as a cell phone sends a routing request, specifying its position and a destination, to a server. The server then uses a rich map database to find a route between the specified origin-destination (OD) pair and responds to the request with a route. Eco-routing is an extension of this process with a key difference; the request for eco-route needs to also include description of the vehicle or energy consumption model of the vehicle. The energy consumption model is used to compare candidate routes. These are unique for a vehicle model and may also be customized for individual driving styles. The candidate routes and corresponding trade-offs are then presented to the driver for selection. An example interface used to present the eco-route and to communicate the trade-off is shown in Fig 1.

Fig. 1. Eco-routing

The challenge in deploying eco-routing, compared to widely available routing and navigation, stems from disinclination of automakers to disclose very accurate and high-fidelity energy consumption model of vehicles. This reluctance is attributed to maintaining a competitive advantage because some control strategies may be discerned from such models. Since accuracy of the energy consumption model directly affects the decision metric for eco-routing, they are a necessary enabler.

There are two well-known potential solutions to this dilemma. First solution is to use vehicle agnostic energy consumption models for eco-routing. This obviates the need for disclosure of energy consumption model from automakers but are reported to have error accuracy of about 9% over the trip (Holden et al. 2018). This level of error

makes them harder to use for eco-routing reliably. The second solution is to use the accurate energy consumption models but rely on the eco-routing service provider, cloud service, and all parties in between to maintain the confidentiality of the energy consumption model. This solution may work based on the level of risk tolerance and trade-off with cost for some. The instances of data leak and misuse by employees of the cloud service providers (Cox 2021) is a cause for concern.

Another solution to preserving the confidentiality of the energy consumption model while enabling computations required for eco-routing is to use homomorphic cryptosystems. Homomorphic encryption schemes allow for mathematical operations on numbers in their encrypted form (Yi et al. 2014). The energy consumption model would be encrypted and sent to the eco-routing service, which then uses homomorphic operations to calculate eco-route without ever decrypting the energy consumption model. In this work we propose an application of homomorphic encryption, private function evaluation, for enabling eco-routing. The remainder of this paper introduces concepts pertaining to eco-routing and encryption schemes, describes the proposed eco-routing protocol, a method to encrypt the energy consumption database that enables private function evaluation, and brief note on practical considerations.

2. BACKGROUND

2.1 Energy consumption model for eco-routing

Energy consumption of vehicles have two components: the characteristics of the powertrain that converts stored energy into tractive force; and the reaction of the driver to road conditions that transpires into an energy demand for the powertrain (Hegde et al. 2020).

$$P_{traction} = V_{veh}.\left(M_{total}\dot{V}_{veh} + M_{curb}\, g\, sin(\theta_{grade}) + M_{curb}C_{rr}\, g\, V_{veh}\cos(\theta_{grade}) + \frac{1}{2}\rho_{air}V_{veh}^2\, C_d\, A_f\right) \quad (1)$$

A simplified representation of power required for a vehicle to travel at speed V_{veh} and acceleration $\dot{V}_{veh}$ over a road segment with road grade θ_{grade} is given by (1), where, the total mass of the vehicle and its curb weight is M_{total} and M_{curb} respectively, C_{rr} is the rolling resistance, ρ_{air} is the air density, C_d is the drag coefficient, and A_f is the effective frontal area. The traction power $P_{traction}$, along with intermediate power transfer component efficiencies, is used to compute the total power consumption of the vehicle. It is to be noted that accurate estimation of power consumption over a road segment requires the knowledge of the powertrain parameters and efficiencies. The velocity of the vehicle and the acceleration are determined by the driver's response to the road conditions comprising traffic signals, stop signs, pedestrians, vehicular traffic, road surface, statutory speed limits etc. The velocity of the vehicle over a route can be predicted with a-priori knowledge of these road conditions via a driver behaviour model, extracted from historical data (Chen et al. 2013), or from data aggregation entities such as google maps, HERE maps etc.

The energy consumption model for eco-routing application is simplified by two important steps: 1) by assuming that the velocity and acceleration of the vehicle resemble a typical vehicle, available through recorded historical data or through real-time data streams on the traffic state; and 2) by modelling energy consumption as a function of both powertrain parameters and road conditions. A great example of such a model is RouteE (Holden et al 2020) from National Renewable Energy Laboratory (NREL). They represent the energy consumption of a vehicle traveling over a road segment as a function of the attributes of the road such as average speed, length, elevation, road grade, number of lanes etc. The function may be realized by a look-up table or a regression.

2.2 Eco-routing as a service

Eco-routing as a service off-loads computations of routing and energy consumptions onto a server and thereby reduces workload of the vehicle's onboard computing resources or of a smartphone. A schematic for eco-routing service is shown in Fig 2. A client device such as vehicle or a smartphone is used by the driver to send eco-routing requests to the server. The request typically includes origin, destination and the vehicle's energy consumption model or parameters to inform a predetermined energy consumption model. The eco-routing server uses the energy consumption model to quantify the advantage of traversing a route. To do so, the server queries map databases for features of the route such as speed, stops, road grade etc. to inform the energy consumption model. The source of these road features may reside on multiple servers across many data vendors. The energy consumption model is then evaluated to compute the total energy consumption of the vehicle on a particular road-segment and hence over a route.

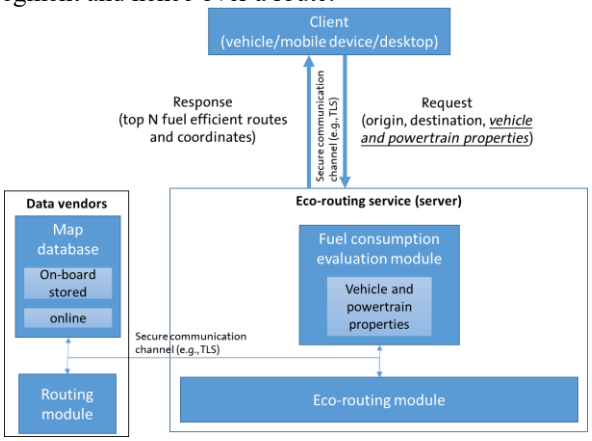

Fig 2. Eco-routing as a service

Several routes are identified from origin to destination and their respective energy consumption, trip time, traffic congestion etc. are computed. Among the routes, a predetermined number of routes, including ones with lowest fuel consumption and fastest trip times, are collected. Finally, the trade-off between energy consumption and travel time is sent to the requester along with the routes as illustrated in Fig 1. The response from the eco-routing service may also include metrics such as CO_2 emission, monetary cost etc. The requester, in this case a driver, then chooses a route from the response to drive on to their destination. In the case of a fleet, this decision may be made with heuristic metrics.

2.3 Cryptosystems and homomorphism

A cryptosystem, loosely defined, are algorithms that define the mapping between a plaintext and ciphertext. The ciphertext is the entity intended to be protected. An encryption operation converts a plaintext into ciphertext using a key, and a decryption operation converts ciphertext back to the correct plaintext. In this paper, the notation in (2) is used for encryption and decryption operation. Encrypted numbers (ciphertext) are represented in boldface in expressions. The relationship between ciphertext and plaintext are also described by (2)

$$E(m) = \boldsymbol{m};\ D(\boldsymbol{m}) = m;\ D\big(E(m)\big) = m \qquad (2)$$

A class of cryptosystems known as public-key cryptosystems is used in this work and is illustrated in Fig 3. This cryptosystem relies on two sets of keys: a public key (k_{pb}); and a private key (k_{pv}). The keys may be a set of numbers depending on algorithms used in a particular cryptosystem. The public key is necessary for the encryption operation: $\boldsymbol{m} = E(m, k_{pb})$. The encrypted number may be decrypted correctly only with the corresponding private key: $m = D(\boldsymbol{m}, k_{pv})$. Decryption operation with any other key yields incorrect plaintext. This type of cryptosystem is used to securely communicate sensitive data. As an example, Alice generates their public and private keys, and broadcasts their public key. Bob uses Alice's public key to encrypt a message and sends it to Alice. Since only Alice's private key can decrypt the message, anyone else with access to the communication cannot read the message. (Yi et al. 2014)

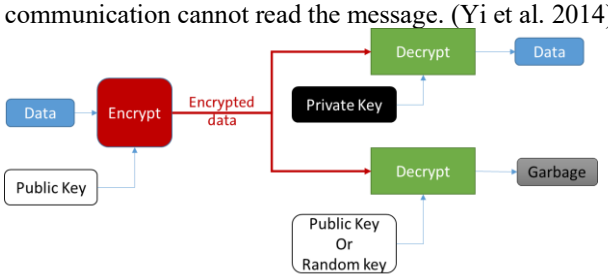

Fig. 3. Public key cryptosystem

A subset of the cryptosystems, homomorphic cryptosystem, has a unique property which allows mathematical operations on the encrypted numbers such that the result, when decrypted, corresponds to a mathematical operation on the corresponding plaintexts. A cryptosystem with additive homomorphic property satisfies (3) where f_{HM} is an operation on the encrypted numbers.

$$D\left(f_{HM}\big(E(m_1), E(m_2)\big)\right) = m_1 + m_2 \qquad (3)$$

The homomorphic property enables private function evaluations where an encrypted mathematical function is evaluated by an entity without the need for decryption. The function evaluation results are also encrypted and hence this operation can be securely done by a cloud service provider or a server without leaking the contents of the mathematical model. In this paper, we utilize this private function evaluation concept for secure eco-routing.

A cryptosystem that is homomorphic under any arbitrary operation qualifies as Fully Homomorphic Encryption (FHE). This is an active research field and a breakthrough in 2009 enabled FHE realization (Gentry 2009). While FHE systems are capable of arbitrary operations enabling any private function evaluation, they also are computationally extremely expensive. Partially Homomorphic Encryption (PHE) schemes have limited homomorphic properties but provide an alternative to FHEs. Since computation time is important for a real-time operation of eco-routing system, we propose the use of Paillier cryptosystem, a PHE, for private evaluation of energy consumption models.

Paillier cryptosystem (Paillier 1999) is a public-key encryption scheme with partial homomorphic properties. A brief review of the Paillier encryption scheme and its properties described in (Yi et al. 2014) are presented here. As is the case with public-key encryption schemes, there are three major steps in utilizing them: key generation; encryption; and decryption.

Key generation: Two large prime numbers p, q are chosen randomly such that $\gcd\big(p.q, (p-1).(q-1)\big) = 1$. Compute $n = p.q$ and $\lambda = lcm(p-1, q-1)$, where *gcd* is greatest common denominator, and *lcm* is least common multiple. Select a random integer $g \in \mathbb{Z}^*_{n^2}$ i.e. $g^n = 1\ mod\ n^2$. The public encryption key is then $k_{pb} = (n, g)$; and the private decryption key is $k_{pv} = (\lambda, \mu)$ where $\mu = (L(g^\lambda (mod\ n^2)))^{-1} (mod\ n)$. Note that the negative exponent signifies modular multiplicative inverse. L is a utility function defined as $L(u) = (u-1)/n$.

Encryption: to encrypt a message $m \in \mathbb{Z}_n$, select a random $r \in \mathbb{Z}^*_n$ and compute the encrypted message (4):

$$E\big(m, k_{pb}\big) = \boldsymbol{m} = g^m . r^n (mod\ n^2) \qquad (4)$$

Decryption: to decrypt a ciphertext $\boldsymbol{m}$, use the private decryption key k_{pv} and compute (5):

$$D\big(\boldsymbol{m}, k_{pv}\big) = m = L\left(c^\lambda (mod\ n^2)\right).\mu (mod\ n) \qquad (5)$$

Note that the encryption operation can be performed with the public key which is generally not a secret. Essentially, once the public key is published, anyone can encrypt new messages using the public key. Decryption can only be performed with the private key which is kept a secret and all messages encrypted with the corresponding public key can be decrypted with and only with the private key.

Homomorphic properties: The Paillier encryption scheme has two interesting homomorphic properties: homomorphic addition; and limited homomorphic multiplication (Yi et al. 2014). Consider two encrypted numbers $\boldsymbol{m_1} = g^{m_1} r_1^n (mod\ n^2)$ and $\boldsymbol{m_2} = g^{m_2} r_2^n (mod\ n^2)$. Homomorphic addition of the two numbers can be achieved by multiplication of the ciphertexts (6):

$$E(m_1 + m_2) = \boldsymbol{m_1}.\boldsymbol{m_2} = g^{m_1+m_2} (r_1 r_2)^n (mod\ n^2) \qquad (6)$$

Upon decryption, product of two encrypted numbers results in their summation. This additive homomorphic property extends to addition of an encrypted number (m_3) to an unencrypted number given that the public key is known (7).

$$E(m_1 + m_3) = \boldsymbol{m_1}.g^{m_3} = g^{m_1+m_3} (r_1)^n (mod\ n^2) \qquad (7)$$

Paillier encryption scheme also has a limited multiplicative homomorphism which enables multiplication of encrypted number with an unencrypted number (8):

$$E(m_1.m_3) = \boldsymbol{m_1}^{m_3} = g^{m_1.m_3} (r^{m_3})^n (mod\ n^2) \qquad (8)$$

An encrypted number raised to the power of an unencrypted constant m_3, results in encrypted product of the encrypted and unencrypted numbers. Note that homomorphic

multiplication, an operation that results in the encrypted product of two numbers, is not feasible in this encryption scheme. In this paper, these partial homomorphic properties are used to construct a protocol for secure eco-routing that allows for private evaluation of the energy consumption models.

3. SECURE ECO-ROUTING

Secure eco-routing proposed in this work relies on private function evaluation techniques. It is enabled by three well-coordinated components: a partially homomorphic encryption scheme that enables evaluation of encrypted energy consumption model; formulation of encryption of the energy consumption model such that they are secure and amenable to private evaluation; and an eco-routing protocol to facilitate the encryption, evaluation, and eco-routing. The novelty of our work relies in crafting these three components to enable eco-routing as an application of homomorphic cryptosystems (Hegde and Chang 2020). This section elaborates on the specific formulation of energy consumption models for encrypted evaluation, the secure eco-routing protocol, and private evaluation of encrypted energy consumption models.

3.1 Formulation of energy consumption models for secure eco-routing

The energy consumption model for eco-routing is represented in two forms in this paper: look-up table (LUT); and polynomial. The inputs to these models, which may include road grade, speed limits etc., are represented as $X = \{x, y, z \dots\}$. The look-up table form is described by a grid of independent variables $X_{i,j,k,\dots} = \{x_i, y_j, z_k, \dots\}$ which map to the value of the energy consumption $f_{i,j,k,\dots}(X_{i,j,k,\dots})$. An arbitrary value of the energy consumption is evaluated by interpolation using the energy consumption values at grid points that are closest to the inputs. Equation (9) shows the 1-dimensional case where the fuel consumption is evaluated at the input x; where x_a and x_b are the grid points in the look-up table with defined function values.

$$f(x) = f(x_a) + \frac{x - x_a}{x_b - x} * (f(x_b) - f(x_a)); x_a \le x \le x_b \quad (9)$$

We propose formulating LUT based energy consumption model as a tuple of plaintext input grid points and encrypted energy consumption values: $\langle E(f_i(X_i)),\ X_i \rangle$; illustrated in Fig 4. The encrypted value of the function at the grid points X_i is the corresponding $\boldsymbol{f_i}(X_i)$ in the tuple. This encrypted representation of the energy consumption model can be evaluated with partially homomorphic encryption as described in the following sections.

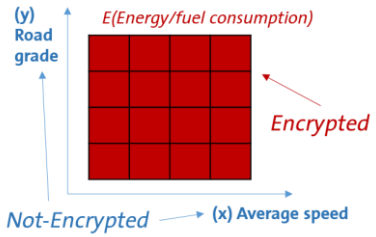

Fig 4. Energy consumption model

The polynomial form of the energy consumption model is described by the coefficients $W = \{w_1, w_2, \dots, w_N\}$ and their weighted sum of the inputs $X = \{x, y, z, \dots\}$ and their combinations as exemplified in (10).

$$f(X = \{x, y, z, \dots\}) = w_1 x + w_2 y + w_3 xy + w_4 z^2 y + \cdots \quad (10)$$

We propose that the polynomial form of the energy consumption model be represented by a tuple of encrypted coefficients and their corresponding relationship to the inputs: $\langle E(W), X \rangle = \langle (\boldsymbol{w_1}, x), (\boldsymbol{w_2}, y), (\boldsymbol{w_3}, xy) \dots \rangle$. Evaluation of this tuple yields an encrypted value of the energy consumption as described in the following sections. The advantage of these representations of the encrypted energy consumption model lies in their flexibility to adapt to various types of modelling paradigms and sensitivities to each available input feature.

3.2 Secure eco-routing protocol

The secure eco-routing protocol defines the relationship and dataflow between a client device that initiates a request for eco-routing, and a server (eco-routing service) that computes eco-route using private function evaluation. The following presents the eco-routing protocol.

Client:

1. Generate *public key* and *private key* for public key homomorphic encryption scheme
2. Encrypt energy consumption database with *public key* as described in Section 3.1
3. Send Origin, destination, and encrypted energy consumption database, and public key
4. Wait for response from eco-routing service
5. Receive N routes along with respective trip time and encrypted energy consumption
6. Decrypt encrypted energy consumption with *private key*
7. Find the route with least energy consumption that meets all the given criteria

Eco-routing service:

1. Receive origin-destination (OD), and encrypted energy consumption database, and public key
2. Generate N candidate routes for the OD pair
3. for N candidate routes do:
   a. for each segment of the route do:
      i. generate dynamic road data X (grade, traffic density, speed limits etc.)
      ii. evaluate encrypted energy consumption $\boldsymbol{f}(X)$ from the encrypted energy consumption database as described in Section 3.3
      iii. accumulate encrypted energy consumption
   b. Store encrypted total energy consumption $\sum f_i(X_i)$
4. Evaluate secondary trip criteria: trip time, via points etc.
5. Return all N routes with their respective trip time, and encrypted energy consumption

3.3 Private evaluation of energy consumption model

The encrypted energy consumption models are evaluated on the server using homomorphic properties of the encryption scheme. The private evaluation of the model is necessary to

ensure confidentiality of client's data encapsulated in the model.

3.3.1. Evaluation of encrypted energy consumption with LUT model

Evaluation of a 2-D encrypted look-up table using linear interpolation for two features of the road segment is illustrated in this section. Hence, the objective is to evaluate $\boldsymbol{f}(x,y) = E(f(x,y))$ given x, y. First, the grid points around (x, y) are obtained by finding grid points x_a, x_b, y_a, y_b such that $x_a \leq x \leq x_b$ and $y_a \leq y \leq y_b$. Then encrypted energy consumption at the grid points are looked up as shown in (11)

$$\boldsymbol{f_{aa}} = E(f(x_a,y_a)), \boldsymbol{f_{ab}} = E(f(x_a,y_b)), \boldsymbol{f_{ba}} = E(f(x_b,y_a)), \boldsymbol{f_{bb}} = E\big(f(x_b,y_b)\big); \quad (11)$$

The 4 coefficients for linear interpolation are then computed using the grid points and inputs (12)

$$C_{aa} = \left(\frac{(x_b-x)(y_b-y)}{(x_b-x_a)(y_b-y_a)}\right); C_{ba} = \left(\frac{(x-x_a)(y_b-y)}{(x_b-x_a)(y_b-y_a)}\right)$$
$$C_{ab} = \left(\frac{(x_b-x)(y-y_a)}{(x_b-x_a)(y_b-y_a)}\right); C_{bb} = \left(\frac{(x-x_a)(y-y_a)}{(x_b-x_a)(y_b-y_a)}\right) \quad (12)$$

Note that $C_{aa}, C_{ab}, C_{ba}, C_{bb}$ are unencrypted plaintexts while $\boldsymbol{f_{aa}}, \boldsymbol{f_{ab}}$ etc. are encrypted. Finally, using the properties of Paillier homomorphic encryption scheme the encrypted energy consumption at x,y is calculated (13)

$$E\big(f(x,y)\big) = \boldsymbol{f}(x,y) = (\boldsymbol{f_{aa}})^{C_{aa}} \cdot (\boldsymbol{f_{ab}})^{C_{ab}} \cdot (\boldsymbol{f_{ba}})^{C_{ba}} \cdot (\boldsymbol{f_{bb}})^{C_{bb}} \ (mod\ n^2) \quad (13)$$

The process for N-dimensional look-up table is similar with 2^N look-up operations and power operations, and $2^N - 1$ multiplications of encrypted energy consumption values.

3.3.2. Evaluation of encrypted energy consumption with polynomial model

Similarly, evaluation of a polynomial encrypted energy consumption for two road features is illustrated in this section. Assuming that the energy consumption of the vehicle at x, y is modeled as (14)

$$f(x,y) = w_1 x + w_2 x^2 + w_3 y + w_4 y^2 + w_5 y^2 x. \quad (14)$$

The encrypted energy consumption is obtained, using properties of Paillier homomorphic encryption scheme, as shown in (15):

$$E\big(f(x,y)\big) = \boldsymbol{w_1}^x \cdot \boldsymbol{w_2}^{x^2} \cdot \boldsymbol{w_3}^y \cdot \boldsymbol{w_4}^{y^2} \cdot \boldsymbol{w_5}^{y^2 x}\ (mod\ n^2) \quad (15)$$

Unlike the look-up table case, the number of operations required to evaluate the encrypted energy consumption depends on the number of terms in the polynomial model and not the number of input road features. A polynomial model with N terms requires N power operations and N-1 multiplications of encrypted numbers. Accumulation of the total energy consumption over a route is obtained by adding the encrypted energy consumption of each segment of the route: $E(\sum f_i) = \prod E(f_i)(mod\ n^2)$

4. IMPLEMENTATION AND RESULTS

The Paillier encryption scheme is defined for non-negative numbers and hence it is necessary to ensure the energy consumption model does not yield a negative number. Typically, energy consumption is always modelled to be zero or positive but special attention needs to be paid for intermediate values of interpolation as well to ensure they are positive. In Section 2.3 on key generation, it is stated that the encryption key n can be a product of any two large primes p, q. A consequence of Paillier encryption scheme is that any number greater than n changes to a corresponding modulus on n. Consequently, all numbers in evaluation of the energy consumption and accumulated energy consumption must be less than n and hence, in principle there needs to be a lower bound on selection of the primes p, q depending on the energy consumption model parameters. In practice however, recommended key-length for security, at least 1024 bits, far exceeds any fuel consumption value for a reasonable trip, even if measured in micrograms or millijoules per road segment.

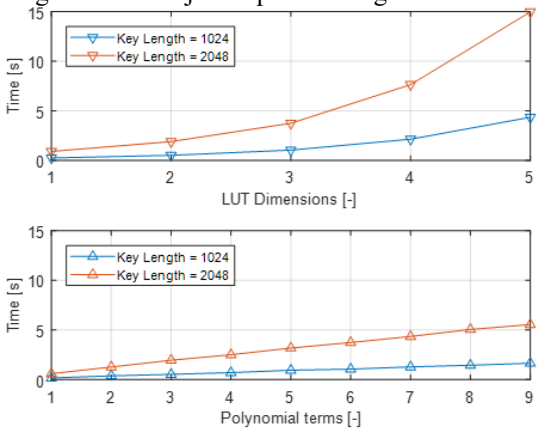

Fig 5. Time required for encrypted fuel consumption evaluation for 100 road segments.

The Paillier encryption scheme and the described private function evaluation of energy consumption model for eco-routing are implemented in Python using an open-source library *python-paillier* (CSIRO's Data61 2013). This library handles key generation, homomorphic operations, and managing floating point numbers by appropriate integer encodings. Encrypted look-up tables and encrypted polynomial models are implemented as described in section 3.1. Time required for evaluation of energy consumption for 100 road segments by private evaluation of the look-up tables and polynomial models are presented in Fig 5. The graph also presents impact of using two key-lengths for encryption: 1024 bits and 2048 bits. It should be noted that the private function evaluation code is not optimized for speed and runs sequentially on the computer. The number of operations for encrypted look-up table (LUT) evaluation increase exponentially with the number of dimensions, this is reflected in the time taken for the operations. It is feasible to parallelize operations in (11), (12), and (13), but their impact on computation time may depend on the number of dimensions of LUT. The example case of energy consumption model described by Holden et al. 2020 has 4 input features and can be represented by a 4-dimensional LUT. This evaluation costed about 2.15s for 100 road segments. Polynomial representation of the energy consumption model shows a linear trend with respect to the number of terms. The key-length for encryption has a proportional impact on computation time so it is important

to balance the trade-off between security and turn-around time for eco-routing.

The computation time trend for LUT and polynomial can be approximated as $\alpha \cdot 2^{N_l}$ and $\beta \cdot N_p$ respectively where α and β are constants, N_l is number of dimensions in LUT, and N_p is number of terms in the polynomial. Table 1 presents the best fit for the results presented in Fig. 5.

Table 1. Coefficients for computation time trends

Key-length (bits)	$\boldsymbol{\alpha}$	$\boldsymbol{\beta}$
1024	0.1354	0.1823
2048	0.4695	0.6251

Computational time parity can be calculated using this approximation to compare relative efficiency of LUT and polynomial representation of the energy consumption model. Fig. 6 shows the computation time parity curve $\frac{\alpha}{\beta}2^{N_l} - N_p = 0$. Region under the curve represents higher efficiency compared to the corresponding LUT dimension. For instance, if an energy consumption model represented as a 5D LUT can also be represented as a polynomial with fewer than 24 terms, then the polynomial representation is more computationally efficient.

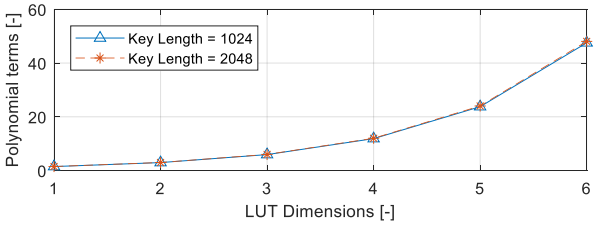

Fig 6. Computation time parity of LUT and polynomial models

A combination of LUT and polynomial can also be used to represent an encrypted energy consumption model. This hybrid version with 2-D LUT which contained encrypted coefficients for a polynomial with 6 terms is implemented as an example. The computation time for the hybrid model is 1.25s and 4.36s for key-lengths of 1024 and 2048 bits respectively. The error introduced by the secure eco-routing methods is negligible, at around 3×10^{-14}%, across all cases.

5. CONCLUSIONS

Secure eco-routing using partially homomorphic encryption scheme and private function evaluation is presented in this paper. The unique representation of the energy consumption model coupled with Paillier encryption scheme enables the secure eco-routing protocol described. The methods presented in this paper may be extended to other representations of energy consumption models. The challenge in doing so primarily resides in tailoring the representation of the model to fit the choice of encryption scheme. Private function evaluations can accommodate larger and more complex model representations such as neural networks but comes with added computational expense.

6. ACKNOWLEDGEMENT

This work was authored with funding provided by the Advanced Research Projects Agency-Energy (ARPA-E), U.S. Department of Energy, under Award Number DE-AR0000790. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. The U.S. Government retains, and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

REFERENCES

Barth, M., Boriboonsomsin, K. and Vu, A., 2007, September. Environmentally-friendly navigation. In 2007 IEEE Intelligent Transportation Systems Conference (pp. 684-689). IEEE.

Brown, A., Gonder, J. and Repac, B., 2014. An analysis of possible energy impacts of automated vehicles. In Road vehicle automation (pp. 137-153). Springer, Cham.

Chen, D., Chen, L. and Liu, J., 2013. Road link traffic speed pattern mining in probe vehicle data via soft computing techniques. Applied Soft Computing, 13(9), pp.3894-3902.

Cox, J., 2021. Leaked document says Google fired dozens of employees for Data Misuse. VICE. Available at: https://www.vice.com/en/article/g5gk73/google-fired-dozens-for-data-misuse [Accessed February 1, 2022].

CSIRO's Data61, 2013. Python Paillier Library. GitHub Repository. https://github.com/data61/python-paillier.

EIA, 2022. Use of energy for transportation in depth - U.S. Energy Information Administration (EIA). [online] Available at: https://www.eia.gov/energyexplained/use-of-energy/transportation-in-depth.php

Gentry, C., 2009, May. Fully homomorphic encryption using ideal lattices. In Proceedings of the forty-first annual ACM symposium on Theory of computing (pp. 169-178).

Hegde, B. and Chang, CF., 2020. Secure eco-routing with databases under homomorphic encryption. United States Patent and Trademark Office, application number 17/231,288.

Hegde, B., Ahmed, Q. and Rizzoni, G., 2020. Velocity and energy trajectory prediction of electrified powertrain for look ahead control. Applied Energy, 279, p.115903.

Holden, J., Reinicke, N. and Cappellucci, J., 2020. RouteE: A Vehicle Energy Consumption Prediction Engine. Society of Automotive Engineers Technical Paper Series, 2(NREL/JA-5400-78089).

Holden, J., Van Til, H., Wood, E., Zhu, L., Gonder, J. and Shirk, M., 2018. Trip Energy Estimation Methodology and Model Based on Real-World Driving Data for Green-Routing Applications. Transportation Research Record, 2672(24), pp.41-48.

Paillier, P., 1999, May. Public-key cryptosystems based on composite degree residuosity classes. In International conference on the theory and applications of cryptographic techniques (pp. 223-238). Springer, Berlin, Heidelberg.

Yi X., Paulet R., Bertino E. (2014) Homomorphic Encryption. In: Homomorphic Encryption and Applications. SpringerBriefs in Computer Science. Springer, Cham. https://doi.org/10.1007/978-3-319-12229-8_2