LightSolver: New Platform Challenges the Vehicle Routing Problem

Ilan Karpas[1], Avigail Kaner[1], Dov Furman[1], Talya Vaknin[1], Idan Meirzada[1], Harel Primack[1], and Ruti Ben Shlomi[1]

[1]LightSolver LTD., Tel Aviv, Israel

Keywords: vehicle routing problem, VRP, quantum annealing, combinatorial optimization, QUBO, QAOA, optimization

1 Introduction

From the local taxi station dispatcher to Amazon.com, the need for efficient transportation of people, goods or services, arises in virtually every sector of the modern economy. Providing quick, reliable service while keeping down expenses (on fuel, labor, fleets, etc.), is an essential component of the business model for airliners, shipping companies, retail and wholesale sellers, to name just a few. A large airliner, for instance, can have hundreds of airplanes, going to hundreds of different destinations. Planning the route of each airplane, considering factors like demand, carrying capacity, times, weather, fuel and staff costs, is clearly beyond the capabilities of a single dispatcher using intuition, pen and paper.

Instead, mathematical precision, large computing power and clever algorithms are necessary. The practical problem is formulated as a constrained mathematical optimization problem. Destinations and distances between them are represented as a graph, the routing plan as a set of paths on the graph, and each practical consideration mentioned above either as a hard constraint (an airplane can't take off with more people than seats), or preferences which go into the objective function (using 50 fully loaded planes instead of 100 half-empty ones might be a good idea, all other things being equal; See [1] for an attempt to model and solve airplane routing problems). Unfortunately, exact solutions for even the most simplified logistics problems are intractable.

In this paper, we focus on the *Vehicle Routing Problem* (VRP), originally known as the *Truck Dispatcher Problem* [2]. It is an old problem, dating back at least to the 1950's. It is simple to define, but not easy to solve. VRP is a problem interesting both to academia and industry, and has become a standard benchmark in Operations Research. More specifically, the problem we benchmark in this paper is known as the Equal Demand Capacitated Vehicle Routing Problem (EDCVRP). EDCVRP is a special case of the Capacitated Vehicle Routing Problem (CVRP). Both CVRP and EDCVRP are defined in section 2.

This research examines LightSolver (https://lightsolver.com/), a quantum-inspired platform that consists of LightSolver's Laser Processing Unit (LPU) and an algorithmic pre-processing layer. The LPU serves at the core of LightSolver's new laser computing paradigm. It is composed of a spatially coupled array of lasers, and reaches solutions to complex optimization problems more accurately and orders of magnitude faster than other platforms[3, 4, 5].

LightSolver's platform is benchmarked against Gurobi [6], a widely used commercial optimization solver. For Gurobi, we formulate the problem as an Integer Linear Programming (ILP) problem. In fact, we try two different formulations on Gurobi, the first is taken from [7], and the second from [8]. For LightSolver, we formulate the same problem as a Quadratic Unconstrained Binary Optimization (QUBO) problem. The popularity of QUBO formulations for optimization problems has surged over the past couple of decades, primarily due to the emergence of Quantum computers [9]. This formulation is used by Quantum Annealers [10], like D-Wave's Quantum Processing unit (QPU) as well as by gate-based quantum computers, via the Quantum Approximate Optimization Algorithm (QAOA) [11]. This same formulation can be adapted to LightSolver's platform.

The paper is structured as follows: In section 2 we define CVRP and EDCVRP, and expand on its applications, as well as its challenges. In section 3 we show how to formulate EDCVRP as a QUBO problem, in order to solve it on LightSolver' platform. In section 4, we present two formulations of the same problem as ILP instances, which are used for Gurobi. Our experimental setup and results are outlined in section 5. Finally, section 6 includes a discussion of the reults, and future research directions.

2 Vehicle Routing Problem

The Vehicle Routing Problem (VRP) stands as a quintessential challenge in the realm of optimization and logistics, tasked with finding the most efficient way to deliver goods or services from a central depot to a set of customers using a fleet of vehicles. The roots of VRP can be traced back to the pioneering work of George Dantzig and John Ramser in the 1950s. Their initial formulation laid the foundation for addressing distribution problems that abound in transportation and logistics domains. VRP and its variants have since captured the attention of researchers, inspiring the development of numerous solution approaches ranging from exact algorithms to heuristics and metaheuristics.

VRP involves optimizing the routes of a fleet of vehicles to serve a set of customer demands while adhering to certain constraints. Each customer must be visited exactly once, and vehicles must travel from the central depot to customers and eventually return to the depot. The primary goal is to minimize total transportation costs, encompassing factors such as vehicle distances, time, or costs. If the fleet consists of a single vehicle, the problem is equivalent to the famous Travelling Salesperson Problem(TSP).

Despite decades of research, VRP remains a challenging problem. It is known to be computationally hard (NP-hard), rendering exact solutions infeasible for moderately large practical problems. Developing effective approximation algorithms, robust heuristics, and hybrid methods that balance solution quality and computational efficiency remains a forefront challenge.

We define a widely studied variant of VRP, called the Capacitated Vehicle Routing Problem (CVRP),

Definition 1. *In the* ***Capacitated Vehicle Routing Problem (CVRP)****, a fleet of k Trucks, located at a depot v_0, needs to deliver goods to a set of sites, $V = \{v_1, \ldots, v_n\}$. Each site has a demand for quantity D_i of goods, while all trucks have the same carrying capacity, C. between any two locations (two sites or the depot and a site) v_i, v_j, the distance is d_{ij}. The goal is to plan the routes of all trucks, so that every site is supplied with it's demand D_i, and each truck finishes its route at the depot, while minimizing the total distance driven by all trucks. Each site must be supplied by exactly one truck.*

Definition 2. *The* ***Equal Demand CVRP (EDCVRP)*** *[12], is a special case of CVRP, where all customers have unit demand. That is, $D_i = 1$ for all $i \in V$.*

3 QUBO Formulation of EDCVRP

Quadratic Unconstrained Binary Optimization, or QUBO, is an optimization problem of the form [13]:

$$\min_{\boldsymbol{x} \in \mathbb{B}^D} \boldsymbol{x}^\top \boldsymbol{Q} \boldsymbol{x} \tag{1}$$

where $\mathbb{B} = \{0, 1\}$ is a binary set and $\boldsymbol{W} \in \mathbb{R}^{D \times D}$ is a real-valued matrix. The ability to efficiently minimize (1) using quantum computers paves the way to the design of powerful solvers for fundamental optimization. Importantly, QUBO optimization becomes appealing when the run-time of quantum-based algorithms is smaller than that of classic computers. We solve instances of EDCVRP by formulating it as a QUBO problem. Our formulation proceeds as follows:

M trucks, each of capacity K, serve $N = MK$ customers, all with demand 1. All trucks start and end at the depot. The distance between locations v_i and v_j is d_{ij}, for $i, j \in [N]$. Because all demands are equal to 1, in any feasible solution each truck must visit exactly K customers. In the QUBO formulation, we define decision variables x_{mkn}, for $m \in [1, M], k \in [1, K], n \in [1, N]$. These decision variables encode a solution as follows:

- $x_{mkn} = 1$ if truck m is at its kth site, while serving customer n.
- $x_{mkn} = 0$ if truck m at its kth site did not serve customer n.

Denote $D := MKN$. Following standard practice when transforming constrained problems to QUBO, we encode the problem as:

$$\min_{\boldsymbol{x} \in \mathbb{B}^D} \lambda \boldsymbol{x}^\top \boldsymbol{Q}_{constraints} \boldsymbol{x} + \boldsymbol{x}^\top \boldsymbol{Q}_{cost} \boldsymbol{x} \tag{2}$$

$\boldsymbol{Q}_{constraints}, \boldsymbol{Q}_{cost} \in \mathbb{R}^{D \times D}$, $\lambda \in \mathbb{R}^+$. We choose the matrices and λ in such a way, that $\boldsymbol{x}^\top \boldsymbol{Q}_{constraints} \boldsymbol{x} \geq 0$ for any $x \in \mathbb{B}^D$, with equality iff x encodes a feasible solution to the problem. $\boldsymbol{Q}_{cost}$ corresponds to the objective function in the original, constrained problem. Observe that by taking sufficiently large λ, we can ensure that any solution x minimizing (2), must satisfy $\boldsymbol{x}^\top \boldsymbol{Q}_{constraints} \boldsymbol{x} = 0$, thus being a feasible solution.

Let us first write out what are the constraints on variables x_{mkn}, and then proceed to examine how we convert these constraints to QUBO form. The constraints are:

1. $\sum_{n=1}^{N} x_{mkn} = 1$, for any $m \in [1, M]$, $k \in [1, K]$. When truck m is at its kth site, it serves exactly one customer.
2. $\sum_{m=1}^{M} \sum_{k=1}^{K} x_{mkn} = 1$, for every $N \in [1, N]$. Every customer n is served one time by one truck.

These constraints can be converted to QUBO form in a fairly straightforward way. Indeed, an assignment the decision variables x_{mkn} satisfy the first constraint iff

$$\sum_{m=1}^{M} \sum_{k=1}^{K} (\sum_{n=1}^{N} x_{mkn} - 1)^2 = 0,$$

and it satisfies the second constraint iff

$$\sum_{n=1}^{N} (\sum_{m=1}^{M} \sum_{k=1}^{K} x_{mkn} - 1)^2 = 0.$$

Combining the two, we obtain:

$$H_{constraints}(x) = \boldsymbol{x}^\top \boldsymbol{Q}_{constraints} \boldsymbol{x} = \sum_{m=1}^{M} \sum_{k=1}^{K} (\sum_{n=1}^{N} x_{mkn} - 1)^2 + \sum_{n=1}^{N} (\sum_{m=1}^{M} \sum_{k=1}^{K} x_{mkn} - 1)^2 = 0$$

Next, we turn to the QUBO formulation corresponding to the objective function to be minimized. Recall that this should correspond to the total distance traversed by all trucks. This can be expressed as:

$$H_{cost}(x) = \boldsymbol{x}^\top \boldsymbol{Q}_{cost} \boldsymbol{x} = \sum_{m=1}^{M} \sum_{n=1}^{N} d_{0n} x_{m1n} + \sum_{m=1}^{M} \sum_{n=1}^{N} d_{n0} x_{mKn} + \sum_{m=1}^{M} \sum_{k=1}^{K-1} \sum_{n_1, n_2 = 1}^{N} d_{n1n2} x_{mkn_1} x_{m(k+1)n_2}$$

.

The first summand corresponds to the first leg of each route, $k = 1$, where the trucks leave the depot 0. Similarly, the second summand corresponds to the last leg of each route, $k = K$, where the trucks return to the depot 0. The last summand corresponds to all the intermediate legs of each route. The distance $d_{n_1 n_2}$ between sites n_1 and n_2 will appear in the sum iff some truck m served client n_1 as its kth site, and then immediately traveled to serve client n_2 as its $(k + 1)$th site. Our final QUBO problem, therefore, is:

$$\min_{\boldsymbol{x} \in \mathbb{B}^{MNK}} H_{cost}(x) + \lambda H_{constraints}(x)$$

for sufficiently large λ.

4 ILP formulations of Equal Demand CVRP

We formulate the EDCVRP problem also as an Integer Linear Programming problem. We use two different formulations, and test them both. Let 0 be the depot, $\mathbb{V} = [1, N]$ customer locations, and $\{d_{ij}\}_{i \neq j}$ the distances between any two sites in $\mathbb{V} \cup \{0\}$. We assume M trucks, capacity K per truck, where $N = MK$, and demand 1 for all customers.

Formulation 1 [7]:

x_{ij} - $N(N+1)$ decision variables, for each edge (i, j), $0 \leq i \neq j \leq N$.

f_{ij} - $N(N+1)$ integer variables, for each edge (i, j), $0 \leq i \neq j \leq N$.

In any solution, we would like the decision variable x_{ij} to be equal to 1 iff edge (i, j) was traversed by some truck. Thus, the total distance travelled by all trucks, is the sum of $x_{ij} f_{ij}$ over all edges (i, j).

Objective function:

$$\min_{x_{ij} \in \mathbb{B}^{|E|}} \sum_{(i,j) \in E} d_{ij} x_{ij}$$

The integer variables f_{ij}, represent the remaining capacity of the truck that traversed the edge (i, j), when it was at location i. If no truck traversed edge (i, j), then $f_{ij} = 0$.

Constraints:

1. $\sum_{i=0}^{N} x_{ij} = 1$, $j \in [1, N]$. (For each non-depot site j, exactly one of its incoming edges is traversed).
2. $\sum_{j=0}^{N} x_{ij} = 1$. $i \in [1, N]$. (For each non-depot site i, exactly one of its outgoing edges is traversed).
3. $\sum_{j=1}^{N} x_{0j} = M$. (Exactly M edges that go out of the depot, 0, are traversed).
4. $\sum_{i=1}^{N} x_{i0} = M$. (Exactly M edges that go into the depot, 0, are traversed).
5. $0 \leq f_{ij} \leq K x_{ij}$. (Capacity of any truck at any point is at most K. If edge (i, j) not traversed, $f_{ij} = 0$).
6. $\sum_{i=0}^{n} f_{ik} - \sum_{j=0}^{n} f_{kj} = 1$, $k \in [1, N]$. (Demand 1 for all customers).

Formulation 2 [8]:

In the second formulation, the role of x_{ij} is the same as in the first. However, instead of using $N(N+1)$ integer variables f_{ij}, one for each edge, we only use N integer variables, u_i, $1 \leq i \leq N$. In any solution, u_i represents the remaining capacity of the truck that visited site i, at the time of its visit.

x_{ij} - $N(N+1)$ decision variables, for each edge (i, j), $0 \leq i \neq j \leq N$.

u_i - N integer variables, for each non-depot location $i \in [1, N]$.

The Objective function and constraints 1-4 are as in the first formulation. The last constraint compactly says that in any edge (i, j) that is traversed, the capacity of the truck that traversed it decreased by one when it moved from i to j. In any case, the capacity at each non-depot site is between 1 and K.

Objective function:

$$\min_{x_{ij}\in\mathbb{B}^{|E|}} \sum_{(i,j)\in E} d_{ij}x_{ij}$$

Constraints:

1. $\sum_{i=0}^{N} x_{ij} = 1$, for all $j \in [1, N]$.
2. $\sum_{j=0}^{N} x_{ij} = 1$, for all $i \in [1, N]$.
3. $\sum_{j=1}^{N} x_{0j} = M$.
4. $\sum_{i=1}^{N} x_{i0} = M$.
5. $u_i - u_j + Kx_{ij} \leq K - 1$, for all $1 \leq i \neq j \leq N$. (Demand 1 for all customers).

5 Experiments and Results

5.1 Data

We used the OSMnx [14] to collect the locations data. The distances are driving distances, in an area of Dallas, Texas. Using the distances obtained with OSMnx, we created 15 instances of EDCVRP. Each instance is defined by the number of trucks M, with $M \in \{5, 10, 15, 20\}$, and the number of customers per truck K, where $K \in \{2, 4, 8, 10\}$ (20X10 was not tested). The number of customers is always taken to be $N = KM$.

5.2 Experiments

Each instance was solved with the LightSolver's platform (LS), after it was formulated as a QUBO problem, as described in section 3. For comparison, we also used Formulation 1 and Formulation 2 described in section 4, running both of them on Gurobi (GUR1, GUR2, respectively). Each run was set with a 10-second time limit. We report, in each case, the best solution value (minimum total distance) found by each solver, after 0.1, 1 and 10 seconds. Note that in many instances Gurobi does not find any feasible solution within a given time range.

5.3 Results

The results of our experiments are summarized in Figure 1. Observe that in the larger instances, Gurobi does not find **any** feasible solution within the time limit. Indeed, after 10 seconds,GUR1 does not reach a feasible solution in 8 out of 15 instances, and GUR2 in 5 instances. LightSolver, on the other hand, always finds a feasible solution even after

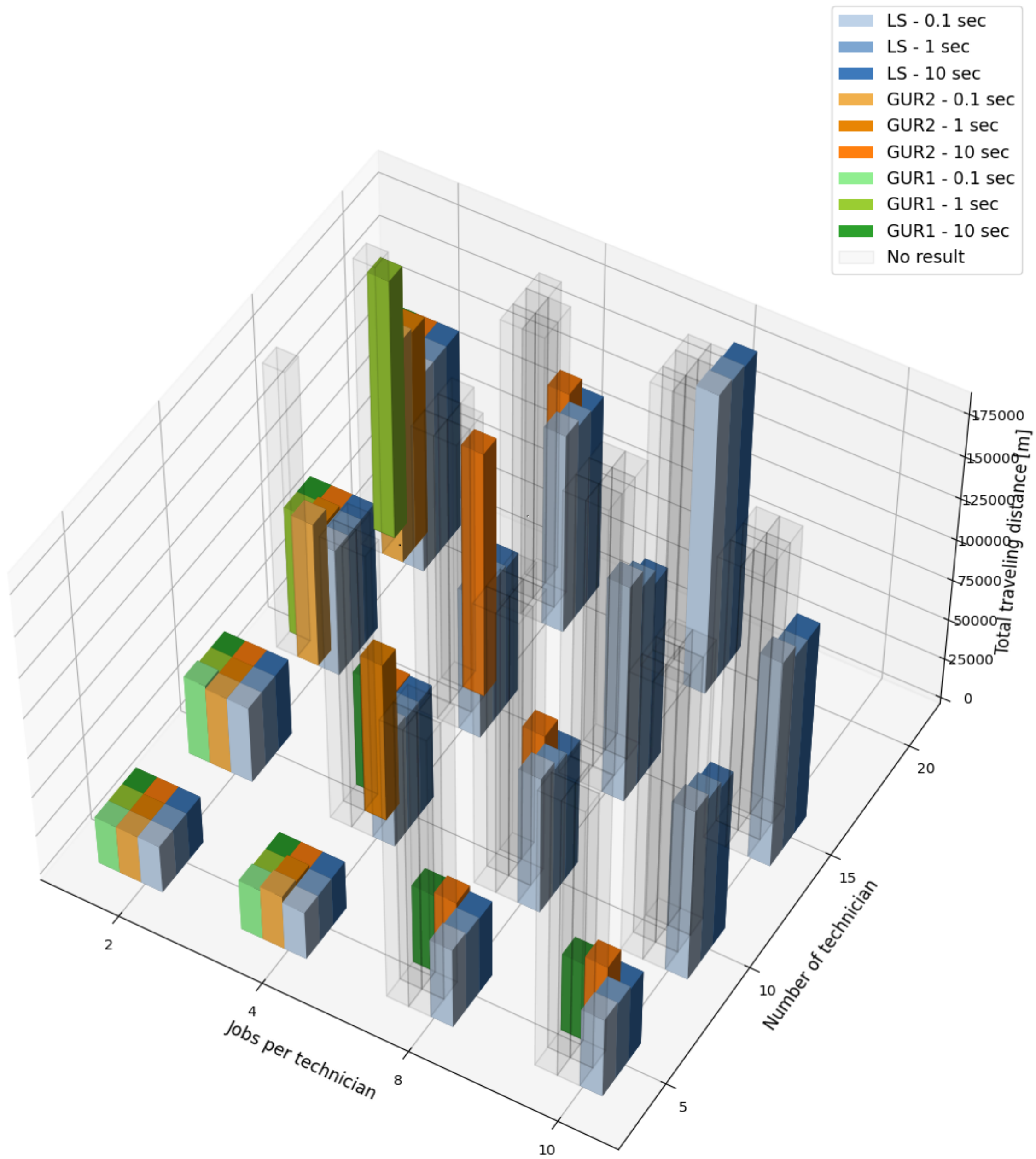

Figure 1: Comparison between the solutions obtained by LightSolver and those obtained by Gurobi, on both formulations. GUR1 and GUR2 are the first and second formulations, defined in section 4.

0.1 seconds. Moreover, when Gurobi does find a solution, in either formulation, it is never better than that found by LightSolver.

Our results also indicate why we have decided to include both GUR1 and GUR2 in our experiments. GUR2 finds feasible solutions on more instances than GUR1. However, when both find feasible solutions, those of GUR1 are usually better.

6 Conclusion

In any practical application of a routing problem, it is unrealistic to expect that all prior estimates would be completely accurate. For instance, suppose we are faced with the problem of assigning routes and jobs to technicians. A technician might call in sick, a customer might cancel or postpone, a vehicle can break down, jobs might take more time than expected, traffic jams can block routes, etc. A plan that seemed optimal in the morning, might turn out to be inefficient, or even impossible, in real-time. Such a dynamic environment calls for fast solvers, that can adapt to the constant inflow of new information and generate new plans accordingly (see [15] for a survey on the Dynamic Vehicle Routing Problem).

In our research, we show that LightSolver's platform is a suitable solution for this challenge. We saw in section 5 that LightSolver generated good solutions in fractions of a second, even for instances of more than 100 clients. LightSolver extends an invitation for a business collaboration, to any enterprise in need of feasible, efficient, fast solutions for their logistics problems. Demos can be booked at `http://lightsolver.com`

References

[1] V. Ho-Huu, S. Hartjes, J.A. Pérez-Castán, H.G. Visser, and R. Curran. A multilevel optimization approach to route design and flight allocation taking aircraft sequence and separation constraints into account. *Transportation Research Part C: Emerging Technologies*, 117:102684, 2020.

[2] G.B. Dantzig and J.h. Ramser. The truck dispatching problem. *Management science*, 6:80–91, 1959.

[3] Idan Meirzada, Assaf Kalinski, Dov Furman, Tsafrir Armon, Talya Vaknin, Harel Primack, Chene Tradonsky, and Ruti Ben-Shlomi. Lightsolver – a new quantum-inspired solver cracks the 3-regular 3-xorsat challenge, 2022.

[4] Yaniv Romano, Harel Primack, Talya Vaknin, Idan Meirzada, Ilan Karpas, Dov Furman, Chene Tradonsky, and Ruti Ben Shlomi. Quantum sparse coding, 2022.

[5] Hod Wirzberger, Assaf Kalinski, Idan Meirzada, Harel Primack, Yaniv Romano, Chene Tradonsky, and Ruti Ben Shlomi. Lightsolver challenges a leading deep learning solver for max-2-sat problems, 2023.

[6] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2023.

[7] Godinho Maria Teresa et al. On the unit demand vehicle routing problem: Flow based inequalities implied by a time-dependent formulation. 2009.

[8] Zuzana Borcinova. Two models of the capacitated vehicle routing problem. *Croatian Operational Research Review*, 8:463–469, 2017.

[9] G. Kochenberger, JK. Hao, Fred Glover, et al. The unconstrained binary quadratic programming problem: a survey. *Journal of Combinatorial Optimization*, pages 58–81, 2014.

[10] Tadashi Kadowaki and Hidetoshi Nishimori. Quantum annealing in the transverse ising model. *Phys. Rev. E*, 58:5355–5363, 1998.

[11] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, pages 1–16, 2014.

[12] Simchi-Levi David, Chen Xin, and Bramel Julien. *The Capacitated VRP with Equal Demands*, pages 301–312. 2014.

[13] Robert Patton, Catherine Schuman, Thomas Potok, et al. Efficiently embedding QUBO problems on adiabatic quantum computers. *Quantum Information Processing*, 18(4):1–31, 2019.

[14] Geoff Boeing. Osmnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, 65:126–139, 2017.

[15] Brenner Humberto Ojeda Rios, Eduardo C. Xavier, Flávio K. Miyazawa, Pedro Amorim, Eduardo Curcio, and Maria João Santos. Recent dynamic vehicle routing problems: A survey. *Computers & Industrial Engineering*, 160:107604, 2021.