Karlsruher Institut für Technologie

ProvideQ: A Quantum Optimization Toolbox

Domenik Eichhorn[1], Nick Poser[1], Maximilian Schweikart[1,2], Ina Schaefer[1]
[1]Karlsruhe Institute of Technology, Germany, {domenik.eichhorn, nick.poser, ina.schaefer}@kit.edu
[2]University of Oxford, United Kingdom, maximilian.schweikart@cs.ox.ac.uk

ProvideQ:
Short Overview of a Hybrid Optimization Toolbox

Domenik Eichhorn, Nick Poser, Maximilian Schweikart, Ina Schaefer

Speaker Portfolio

Domenik Eichhorn
PhD Student at the Chair for
„Test, Validation and Analysis of Software-Intensive Systems“

Research Topics:
Quantum Software and Hybrid Algorithms

Contact:
domenik.eichhorn@kit.edu

Motivation – Quantum Supremancy

- **In theory,** quantum computers can utilize *quantum mechanical effects* to compute specific tasks faster than classical computers.

Use Cases:

- Cryptography
- Material Simulation
- Combinatorial Optimization

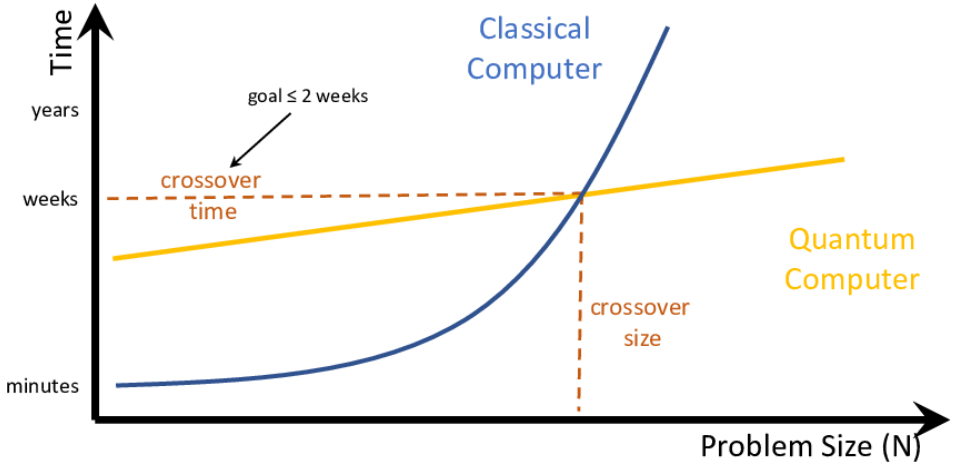


Image Src: T. Hoefler et al., Disentangling Hype from Practicality: On Realistically Achieving Quantum Advantage

Motivation – Quantum Supremancy

- **In theory,** quantum computers can utilize *quantum mechanical effects* to compute specific tasks better than classical computers.

Use Cases:

- Cryptography
- Material Simulation
- **Combinatorial Optimization**

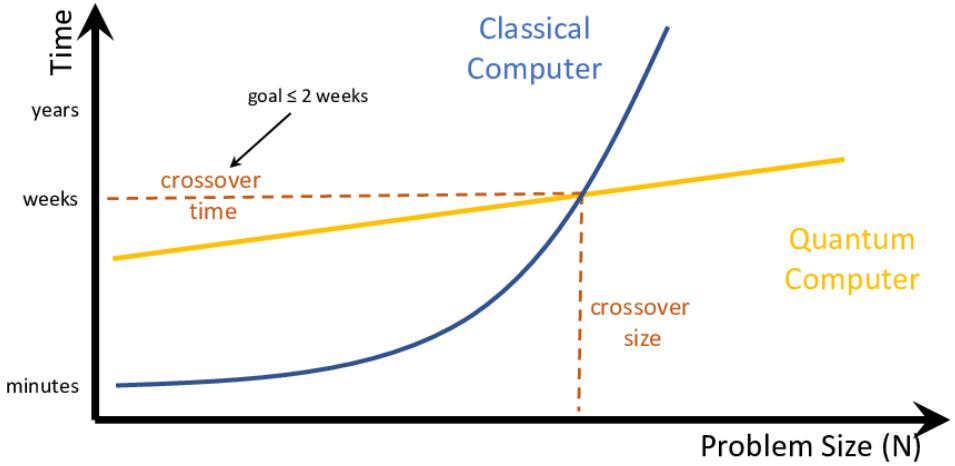
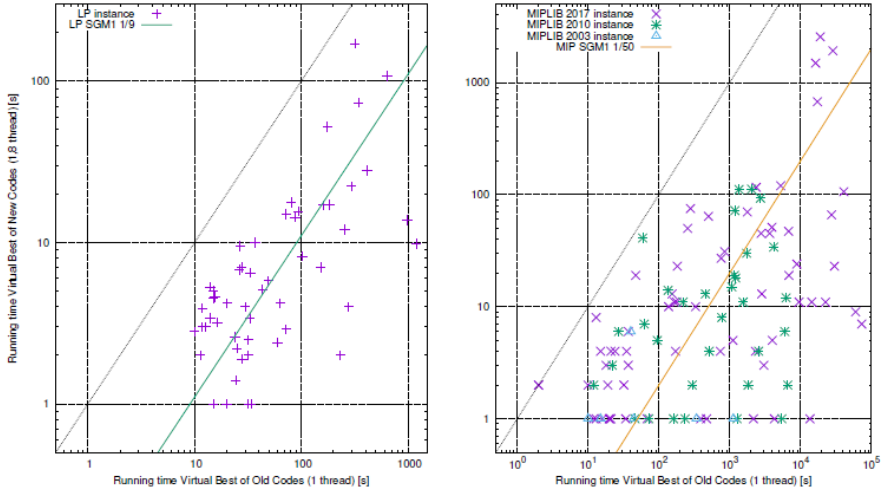


Image Src: T. Hoefler et al., Disentangling Hype from Practicality: On Realistically Achieving Quantum Advantage

What do we compete against?

- Classical Solvers have improved drastically in the recent years:



Example:
Speedup for (Mixed-Integer) Linear Programming Problems:

- LP: 9x Speedup
- MIP: 50x Speedup
- Hardware: 20x Speedup
- Combined LP: 180x
- Combined MIP: 1000x

Source: Koch, T., Berthold, T., Pedersen, J., Vanaret, C.: Progress in mathematical programming solvers from 2001 to 2020. EURO Journal on Computational Optimization 10, 100031 (2022)

Classical Optimization Frameworks

SotA: Traveling Sales Person

Example – World TSP:

- 1,907,711 cities
- Lower Bound: 7,512,218,268
- **Current best Solution: 7,515,755,956** (only 0.0471% greater than LB)
- Achieved with the „LKH-3“ Solver: http://webhotel4.ruc.dk/~keld/research/LKH-3/

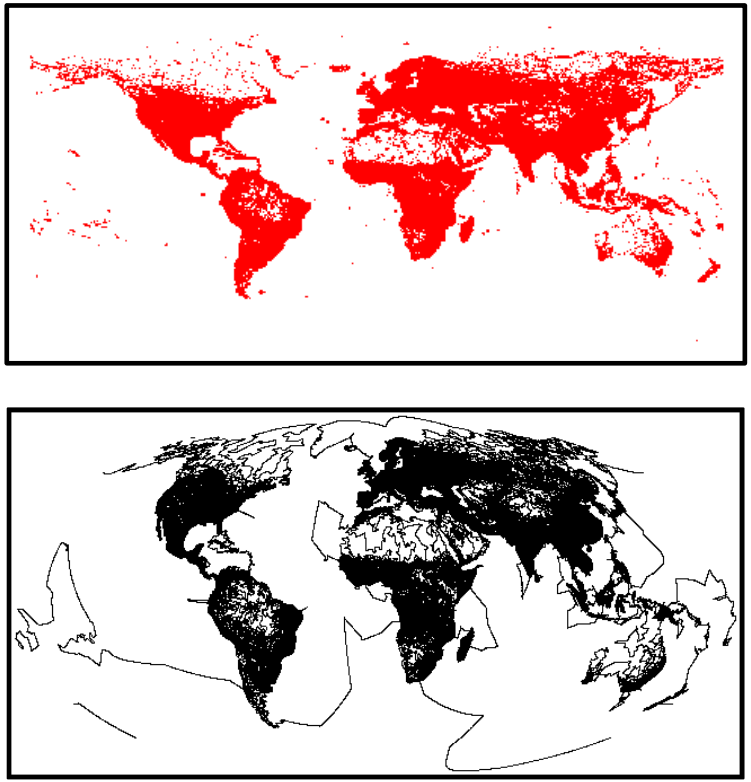


Image Source:
https://www.math.uwaterloo.ca/tsp/world/

Low Autocorrelation Binary Sequences (LABS)

- Also an Optimization problems, however, it „breaks“ much earlier.
- Even problems with only 60 variables are very hard to solve.

Consider a sequence $S = (s_1, \ldots, s_N)$ with $s_i = \pm 1$. The autocorrelations of S are defined as

$$C_k(S) = \sum_{i=1}^{N-k} s_i s_{i+k} \tag{1}$$

for $k = 0, 1, \ldots, N-1$, and the "energy" of S is defined as the sum of the squares of all off-peak correlations,

$$E(S) = \sum_{k=1}^{N-1} C_k^2(S)\,. \tag{2}$$

The *low-autocorrelation binary sequence* (LABS) problem is to find a sequence S of given length N that minimizes $E(S)$ or, equivalently, maximizes the *merit factor*

$$F(S) = \frac{N^2}{2E(S)}\,. \tag{3}$$

Source: Tom Packebusch & Stephan Mertens
Low Autocorrelation Binary Sequences
https://arxiv.org/abs/1512.02475

PHYSICS

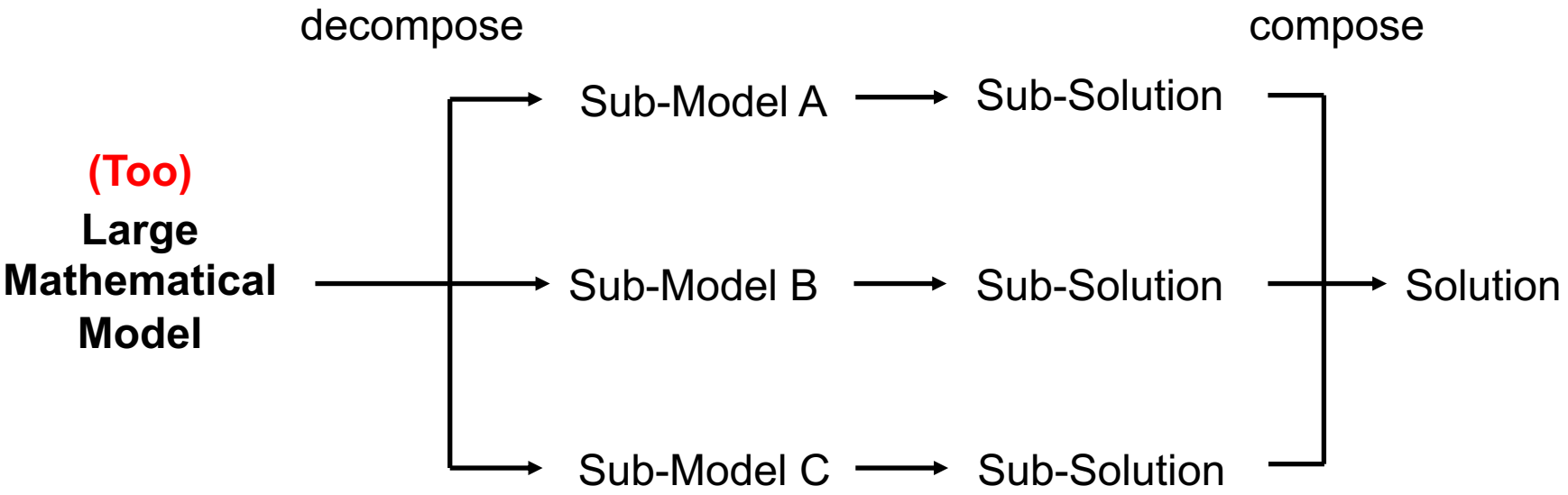
Evidence of scaling advantage for the quantum approximate optimization algorithm on a classically intractable problem

Ruslan Shaydulin[1]*, Changhao Li[1], Shouvanik Chakrabarti[1], Matthew DeCross[2], Dylan Herman[1], Niraj Kumar[1], Jeffrey Larson[3], Danylo Lykov[1,4], Pierre Minssen[1], Yue Sun[1], Yuri Alexeev[4], Joan M. Dreiling[2], John P. Gaebler[2], Thomas M. Gatterman[2], Justin A. Gerber[2], Kevin Gilmore[2], Dan Gresh[2], Nathan Hewitt[2], Chandler V. Horst[2], Shaohan Hu[1], Jacob Johansen[2], Mitchell Matheny[2], Tanner Mengle[2], Michael Mills[2], Steven A. Moses[2], Brian Neyenhuis[2], Peter Siegfried[2], Romina Yalovetzky[1], Marco Pistoia[1]

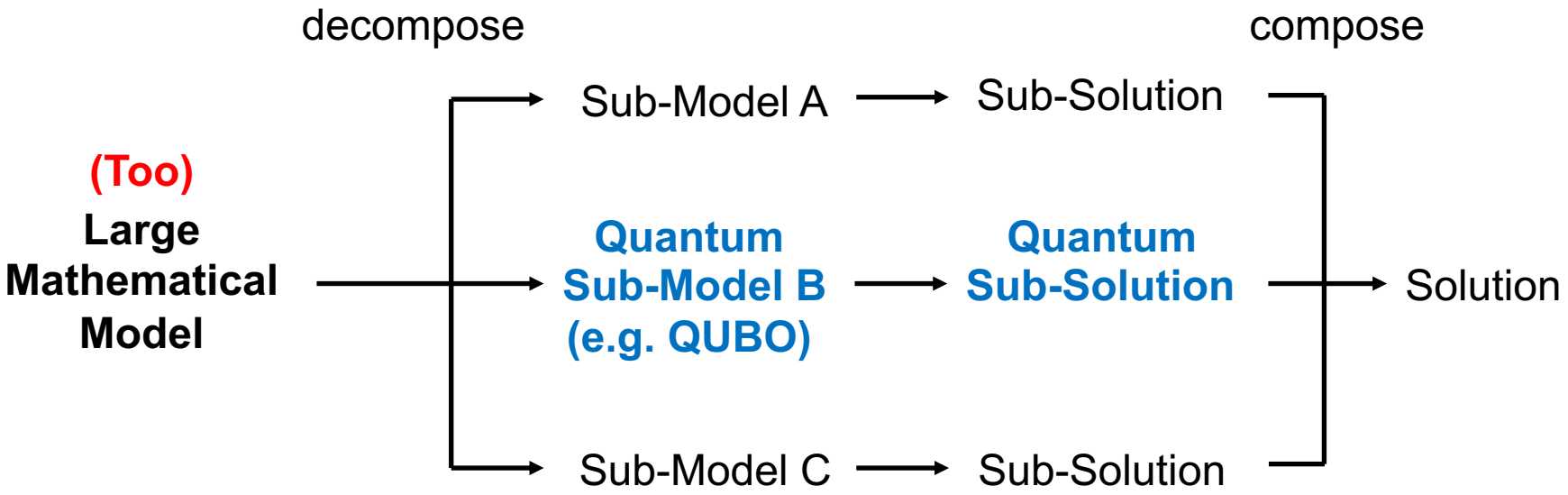
The quantum approximate optimization algorithm (QAOA) is a leading candidate algorithm for solving optimization problems on quantum computers. However, the potential of QAOA to tackle classically intractable problems remains unclear. Here, we perform an extensive numerical investigation of QAOA on the low autocorrelation binary sequences (LABS) problem, which is classically intractable even for moderately sized instances. We perform noiseless simulations with up to 40 qubits and observe that the runtime of QAOA with fixed parameters scales better than branch-and-bound solvers, which are the state-of-the-art exact solvers for LABS. The combination of QAOA with quantum minimum finding gives the best empirical scaling of any algorithm for the LABS problem. We demonstrate experimental progress in executing QAOA for the LABS problem using an algorithm-specific error detection scheme on Quantinuum trapped-ion processors. Our results provide evidence for the utility of QAOA as an algorithmic component that enables quantum speedups.

Classical Optimization Frameworks

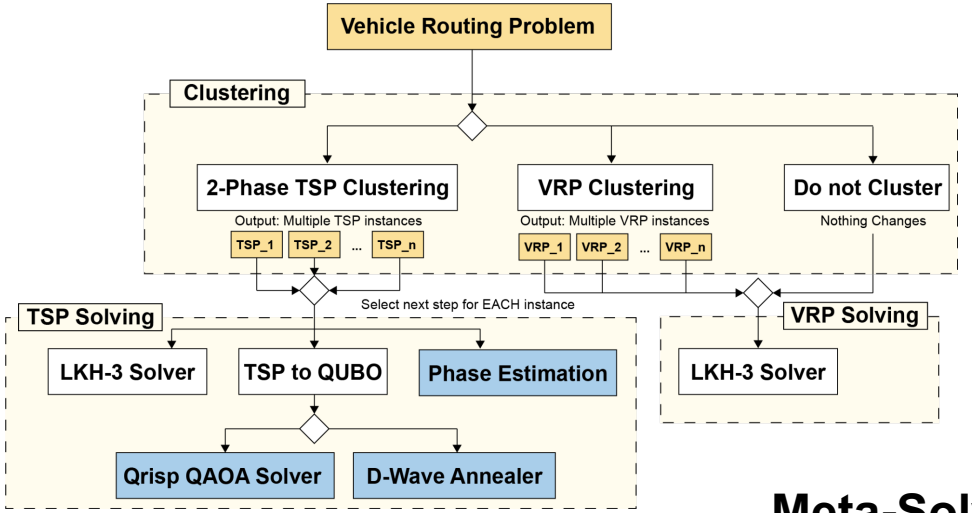
Polylithic Modeling



Polylithic Modeling

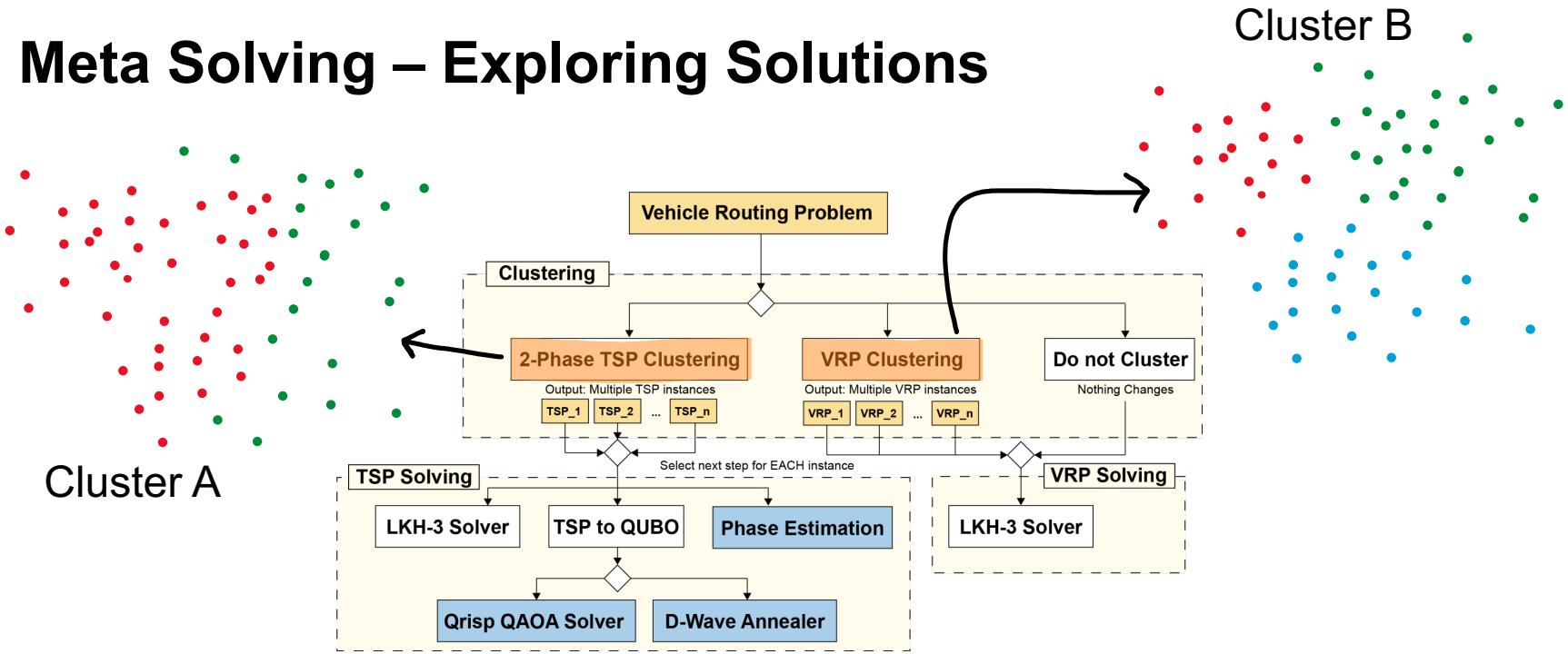


How to add Quantum? → Decompose Problems!



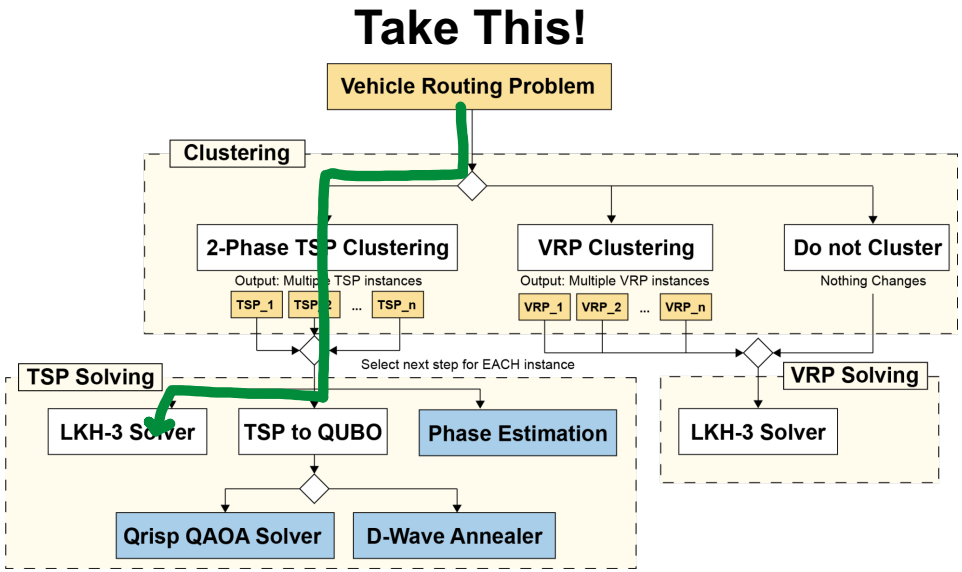
„Meta-Solving“

Meta Solving – Exploring Solutions

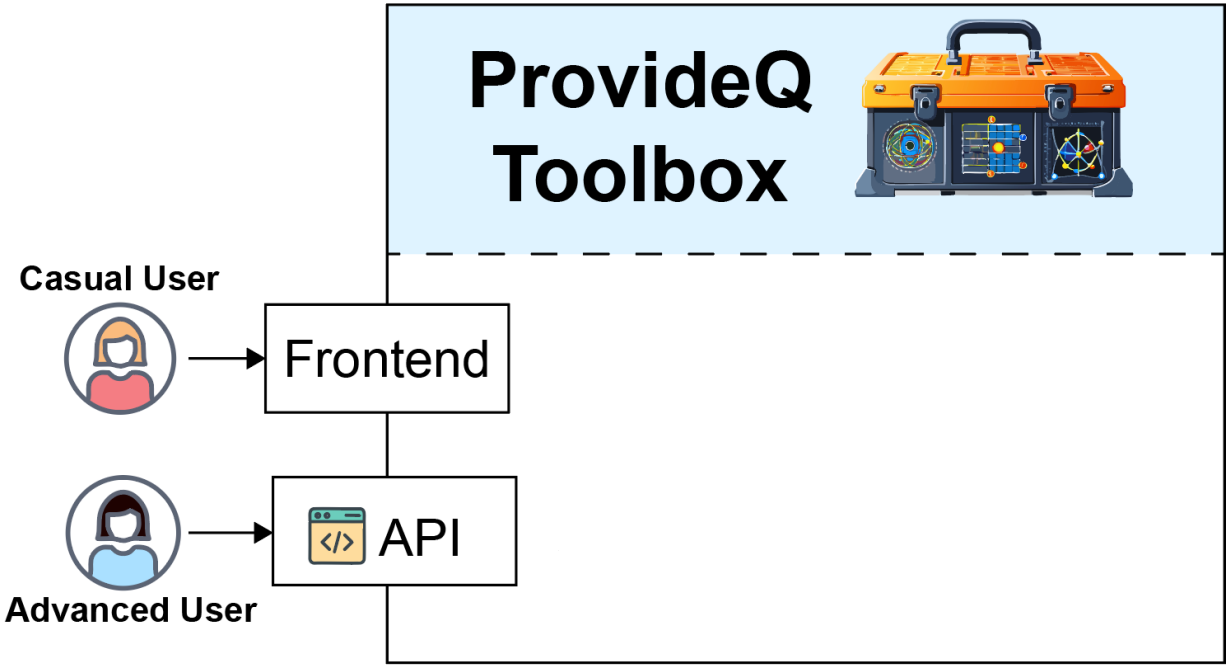


If user is unsure about the next step,
they can try out multiple solution at once.

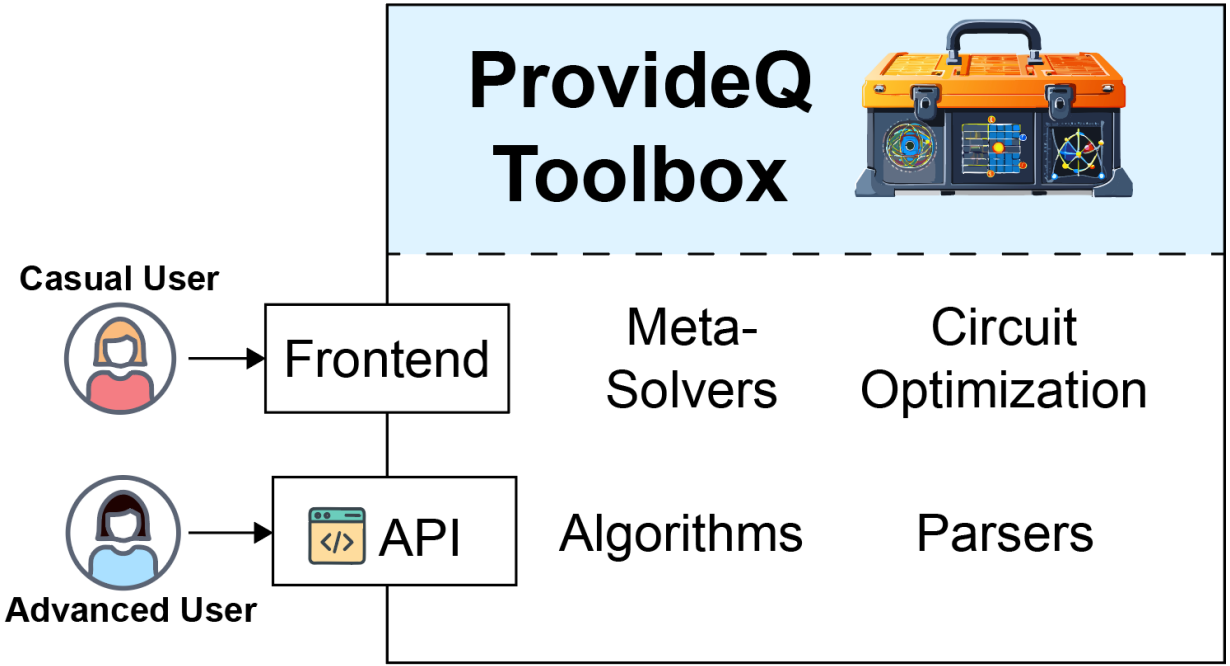
Meta Solving – Proposing Optimal Solvers



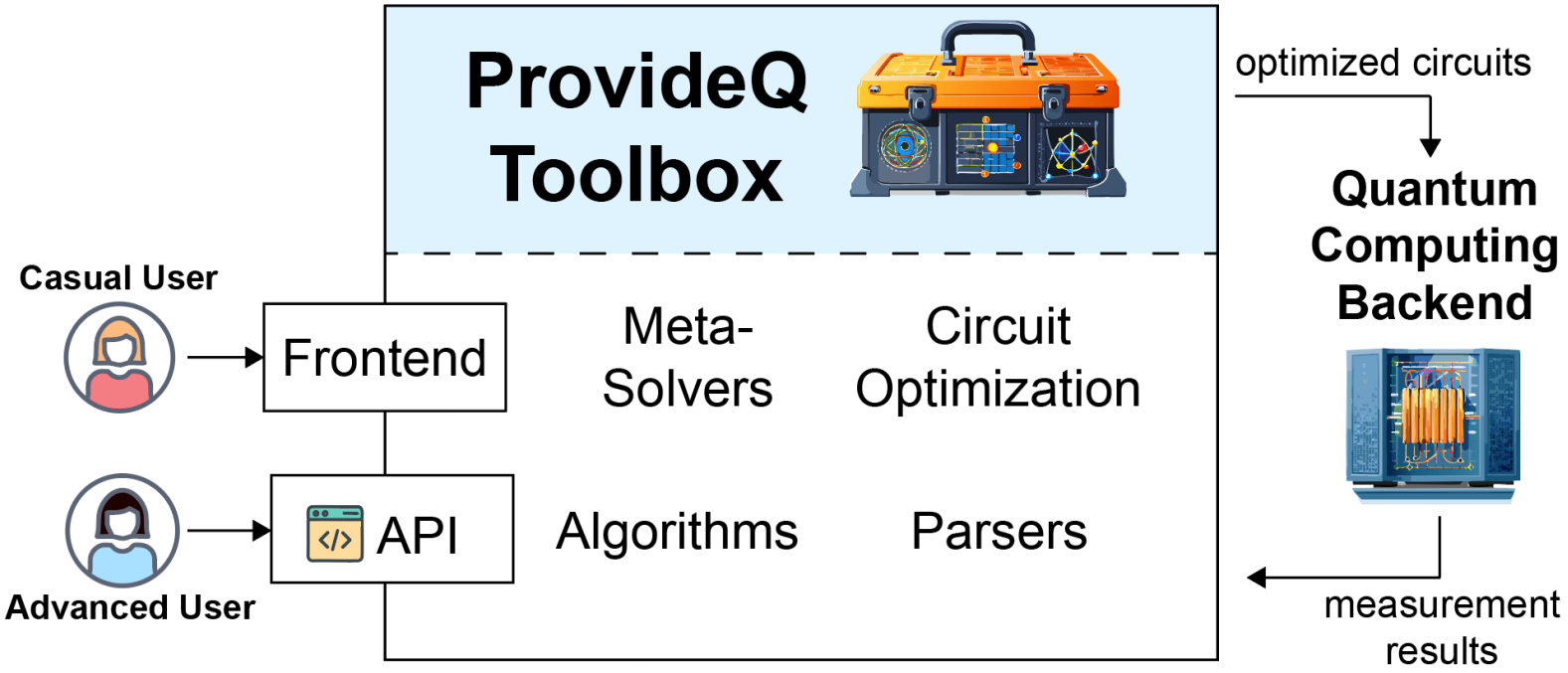
How to implement this?



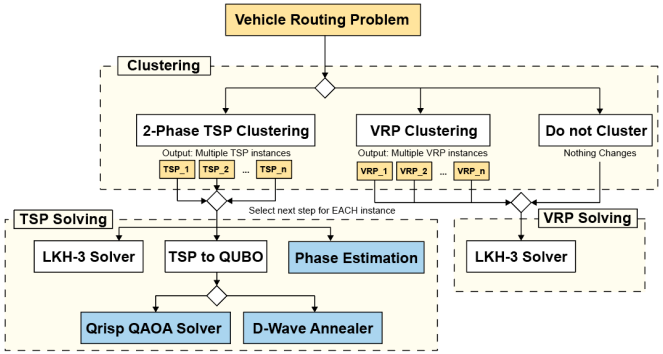
How to implement this?

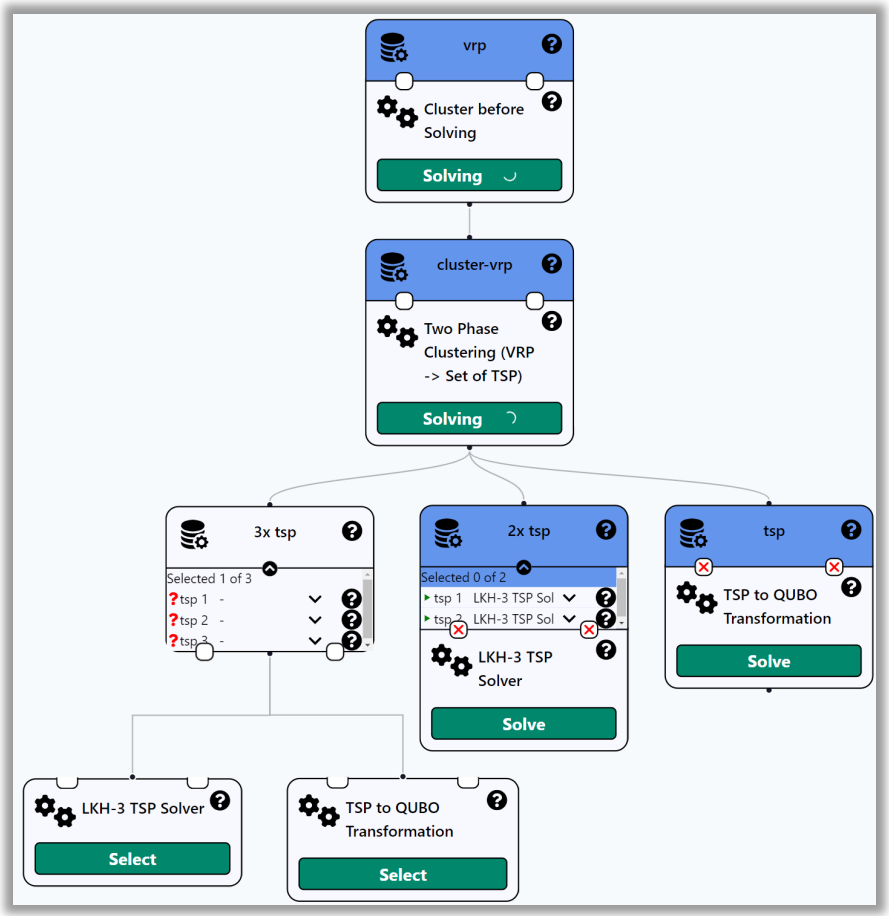


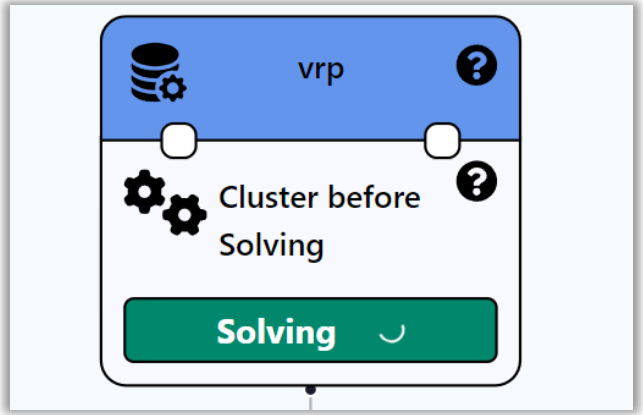
How to implement this?

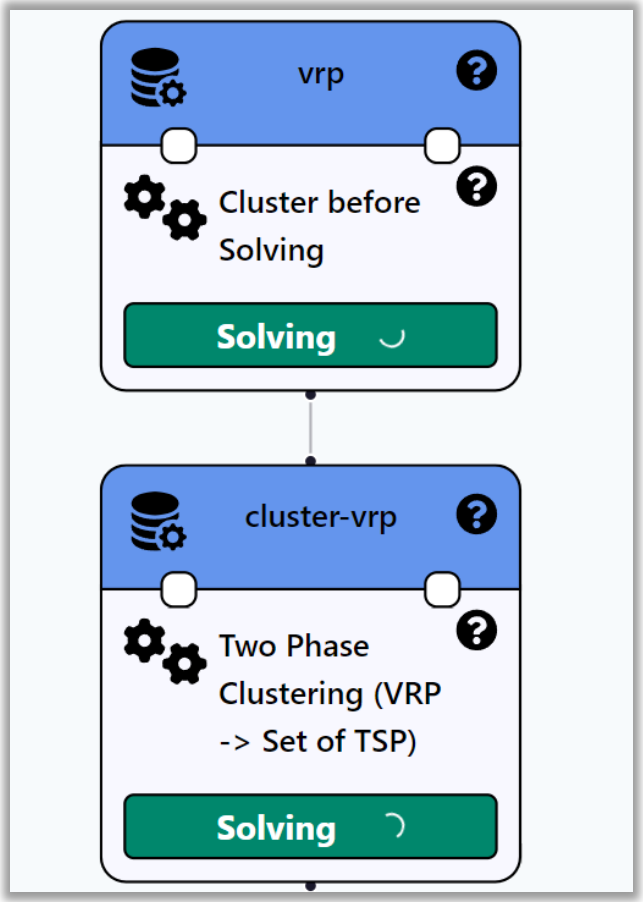


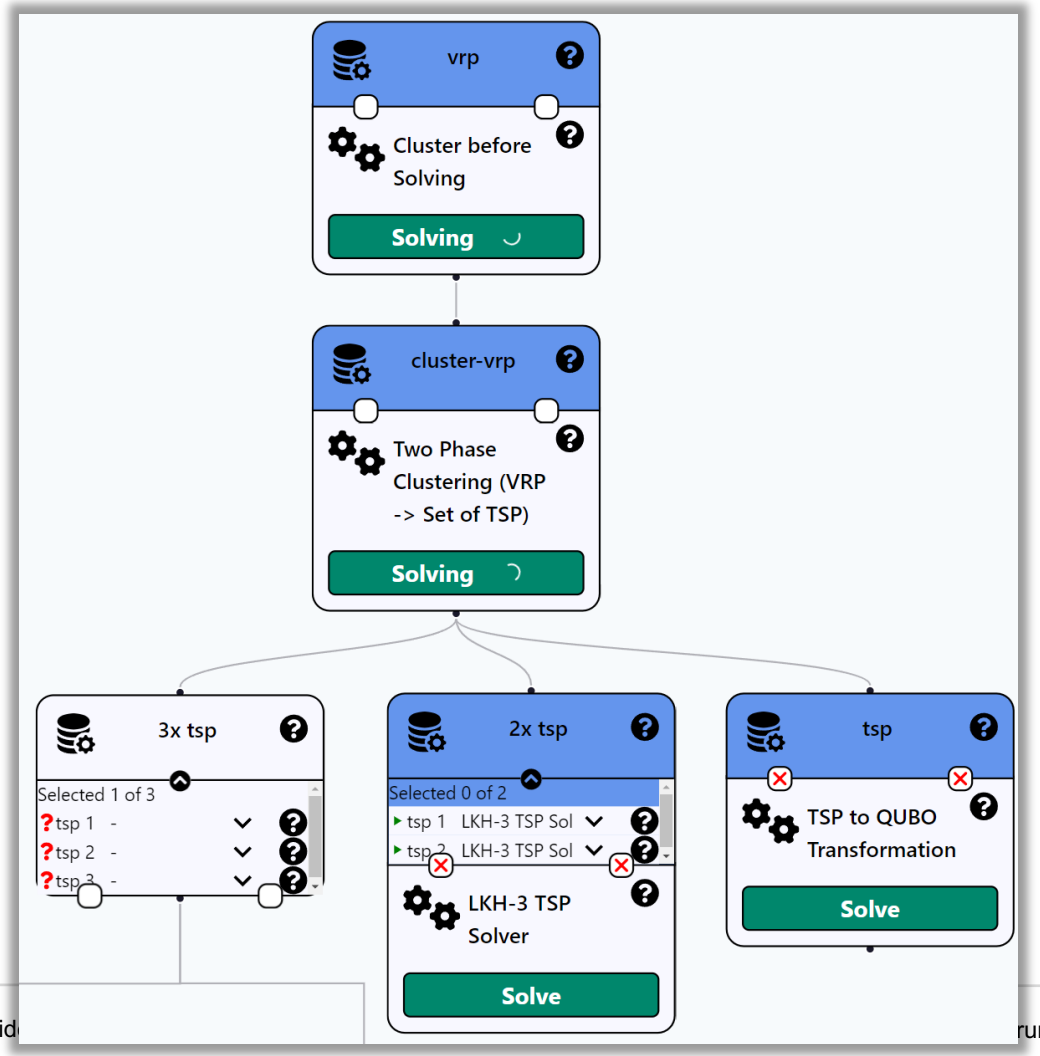
The ProvideQ Toolbox

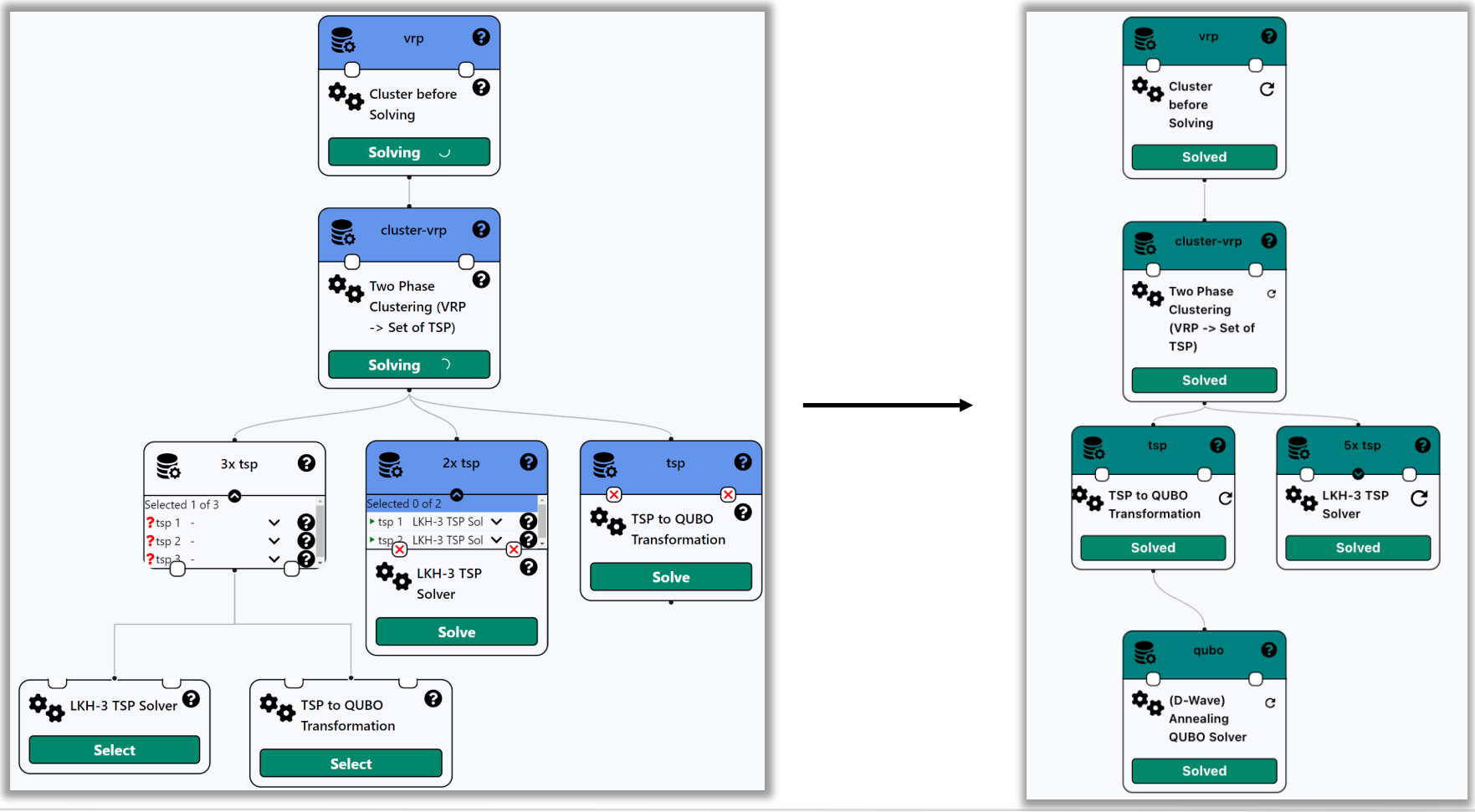





vrp
Cluster before Solving
Solving


vrp
Cluster before Solving
Solving
cluster-vrp
Two Phase Clustering (VRP -> Set of TSP)
Solving


vrp
Cluster before Solving
Solving
cluster-vrp
Two Phase Clustering (VRP -> Set of TSP)
Solving
3x tsp
Selected 1 of 3
tsp 1 -
tsp 2 -
tsp 3 -
2x tsp
Selected 0 of 2
tsp 1 LKH-3 TSP Sol
tsp 2 LKH-3 TSP Sol
LKH-3 TSP Solver
Solve
tsp
TSP to QUBO Transformation
Solve


vrp
Cluster before Solving
Solving
cluster-vrp
Two Phase Clustering (VRP -> Set of TSP)
Solving
3x tsp
Selected 1 of 3
tsp 1 -
tsp 2 -
tsp 3 -
2x tsp
Selected 0 of 2
tsp 1 LKH-3 TSP Sol
tsp 2 LKH-3 TSP Sol
LKH-3 TSP Solver
Solve
tsp
TSP to QUBO Transformation
Solve
LKH-3 TSP Solver
Select
TSP to QUBO Transformation
Select
vrp
Cluster before Solving
Solved
cluster-vrp
Two Phase Clustering (VRP -> Set of TSP)
Solved
tsp
TSP to QUBO Transformation
Solved
5x tsp
LKH-3 TSP Solver
Solved
qubo
(D-Wave) Annealing QUBO Solver
Solved

Solution by Two Phase Clustering (VRP -> Set of TSP)

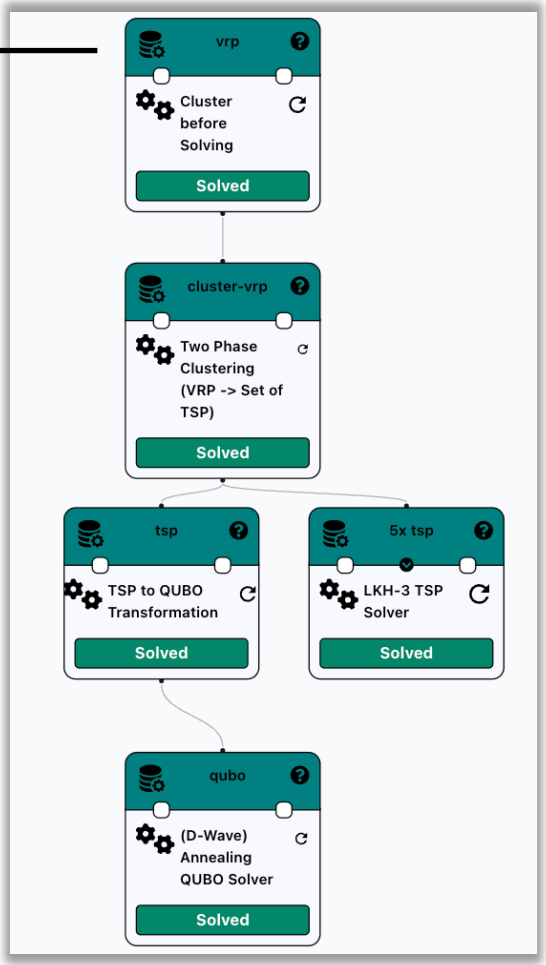
```
NAME : CMT1 solved with length 657.2501647960648
TYPE : TOUR
DIMENSION : 51
TOUR_SECTION
1 20 41 43 42 14 19 45 18 5 -1
1 7 15 26 25 44 8 24 49 28 -1
1 6 39 10 50 11 34 46 16 38 13 -1
1 33 2 23 3 30 22 51 17 12 47 48 -1
1 32 29 4 37 36 21 35 31 40 -1
1 9 27 -1
-1
EOF
```

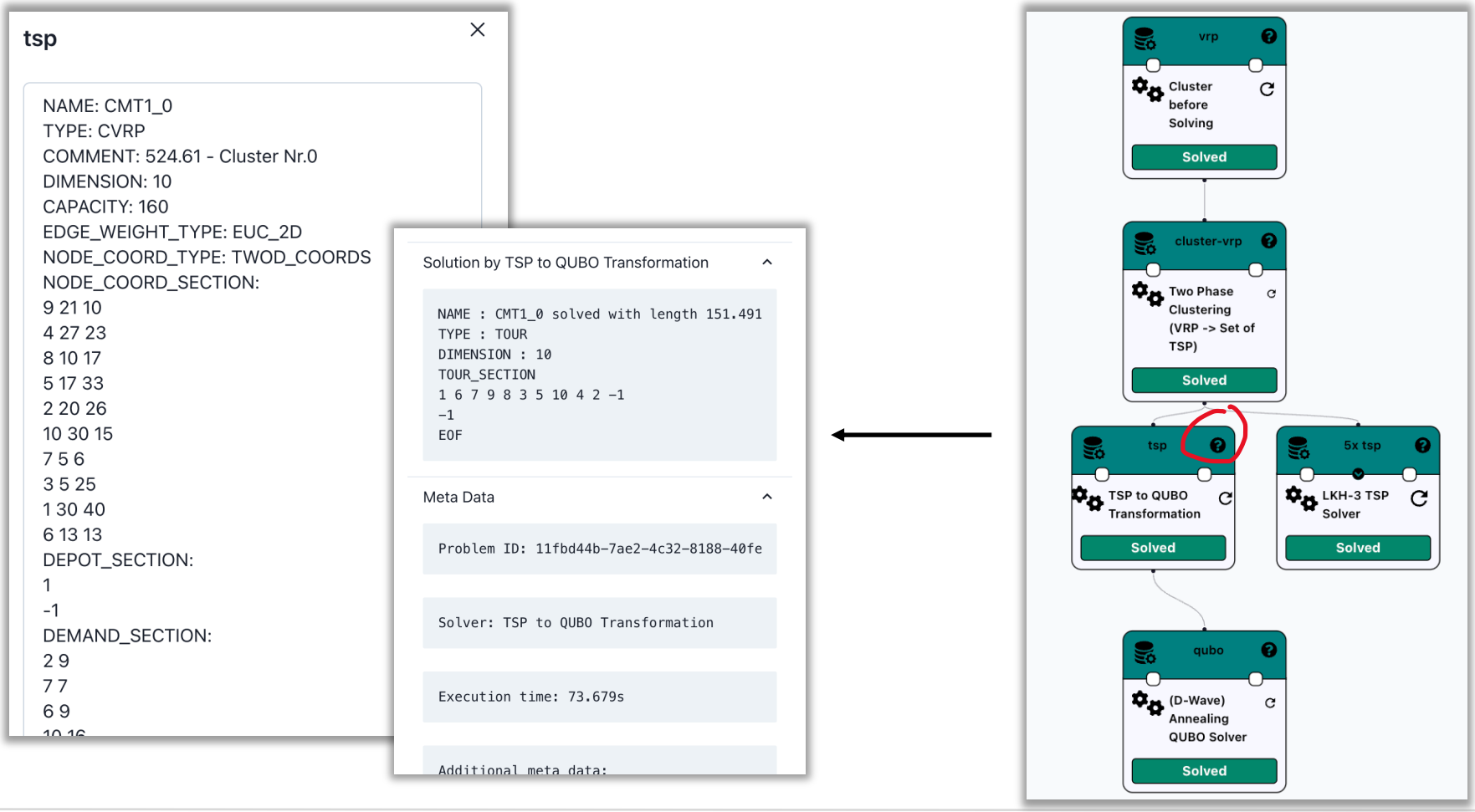
Meta Data

```
Problem ID: 23cbcb8d-e25f-4ca3-8282-a3d1a22097f6
```

```
Solver: Two Phase Clustering (VRP -> Set of TSP)
```

```
Execution time: 99.231s
```




tsp
NAME: CMT1_0
TYPE: CVRP
COMMENT: 524.61 - Cluster Nr.0
DIMENSION: 10
CAPACITY: 160
EDGE_WEIGHT_TYPE: EUC_2D
NODE_COORD_TYPE: TWOD_COORDS
NODE_COORD_SECTION:
9 21 10
4 27 23
8 10 17
5 17 33
2 20 26
10 30 15
7 5 6
3 5 25
1 30 40
6 13 13
DEPOT_SECTION:
1
-1
DEMAND_SECTION:
2 9
7 7
6 9
Solution by TSP to QUBO Transformation
NAME : CMT1_0 solved with length 151.491
TYPE : TOUR
DIMENSION : 10
TOUR_SECTION
1 6 7 9 8 3 5 10 4 2 -1
-1
EOF
Meta Data
Problem ID: 11fbd44b-7ae2-4c32-8188-40fe
Solver: TSP to QUBO Transformation
Execution time: 73.679s
Additional meta data:
vrp
Cluster before Solving
Solved
cluster-vrp
Two Phase Clustering (VRP -> Set of TSP)
Solved
tsp
TSP to QUBO Transformation
Solved
5x tsp
LKH-3 TSP Solver
Solved
qubo
(D-Wave) Annealing QUBO Solver
Solved

qubo

\ QUBO from TSP

Minimize
obj: 0 + [-1688.0758276807353 x1 * x1 +
1688.0758276807353 x1 * x2 +
1688.0758276807353 x1 * x3 +
1688.0758276807353 x1 * x4 +
1688.0758276807353 x1 * x5 +
1688.0758276807353 x1 * x6 +
1688.0758276807353 x1 * x7 +
1688.0758276807353 x1 * x8 +

0
0
0
0
1
0
0

Meta Data

Problem ID: 9070c437-534e-4e25-8347-00f1

Solver: (D-Wave) Annealing QUBO Solver

Execution time: 24.239s

Additional meta data:

Solver Settings:

D-Wave Token

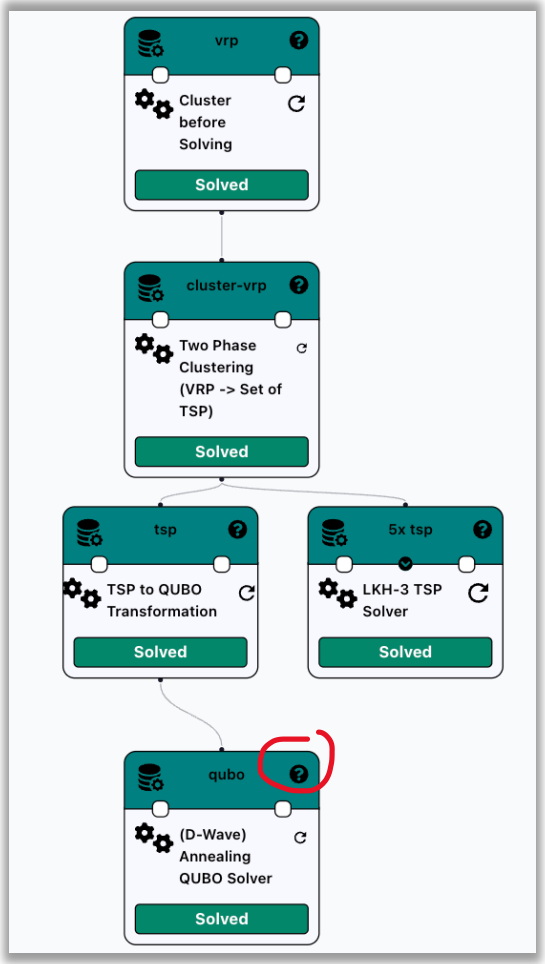
The D-Wave token to use, needed to access the D-Wave hardware

Annealing Method

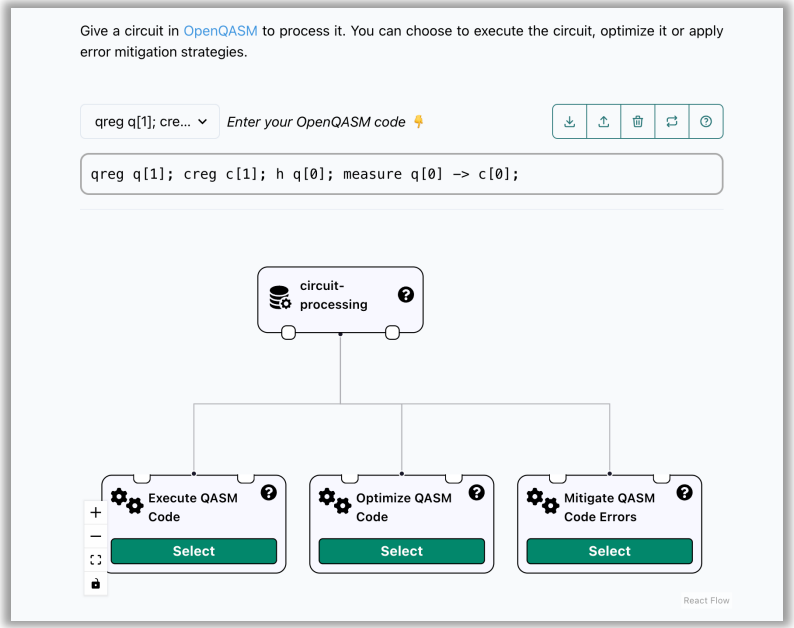
The annealing method to use, only relevant when a token is added

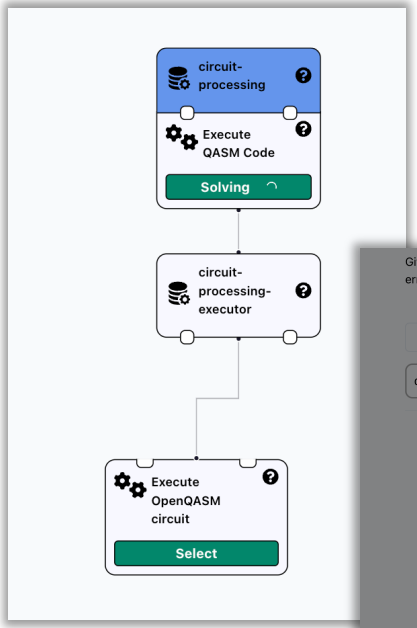
sim

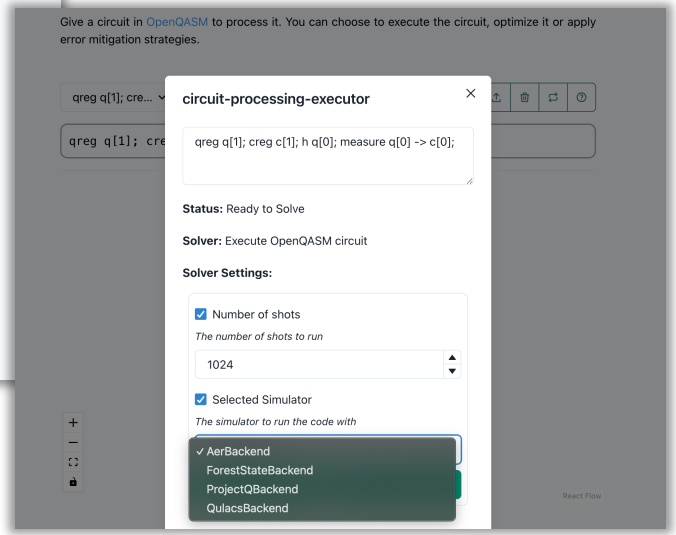
Save



Whats coming: Quantum Circuit Execution & Optimization

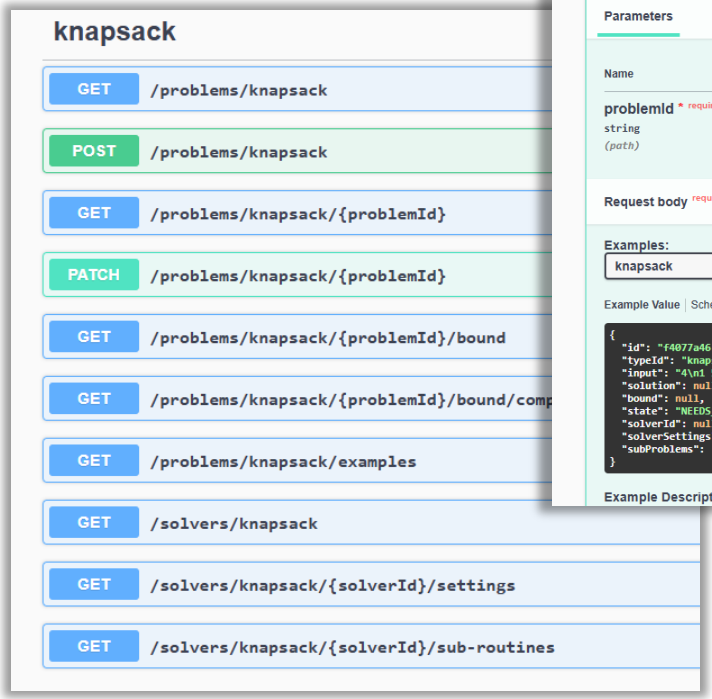


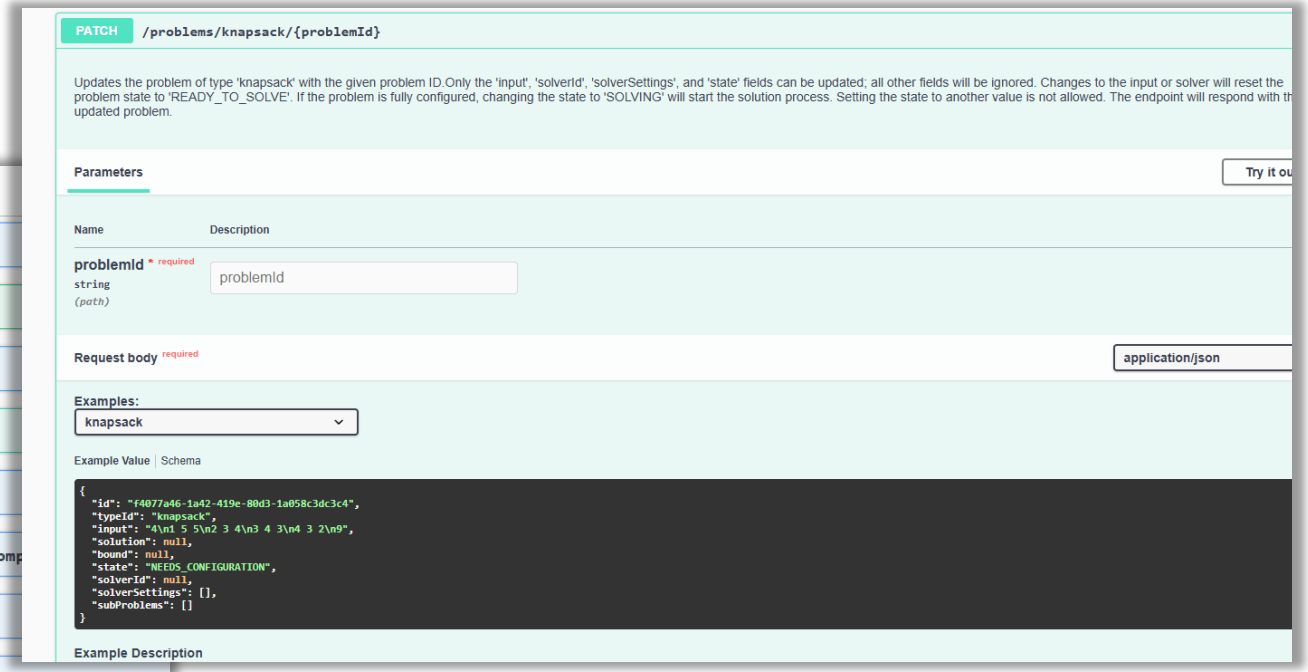




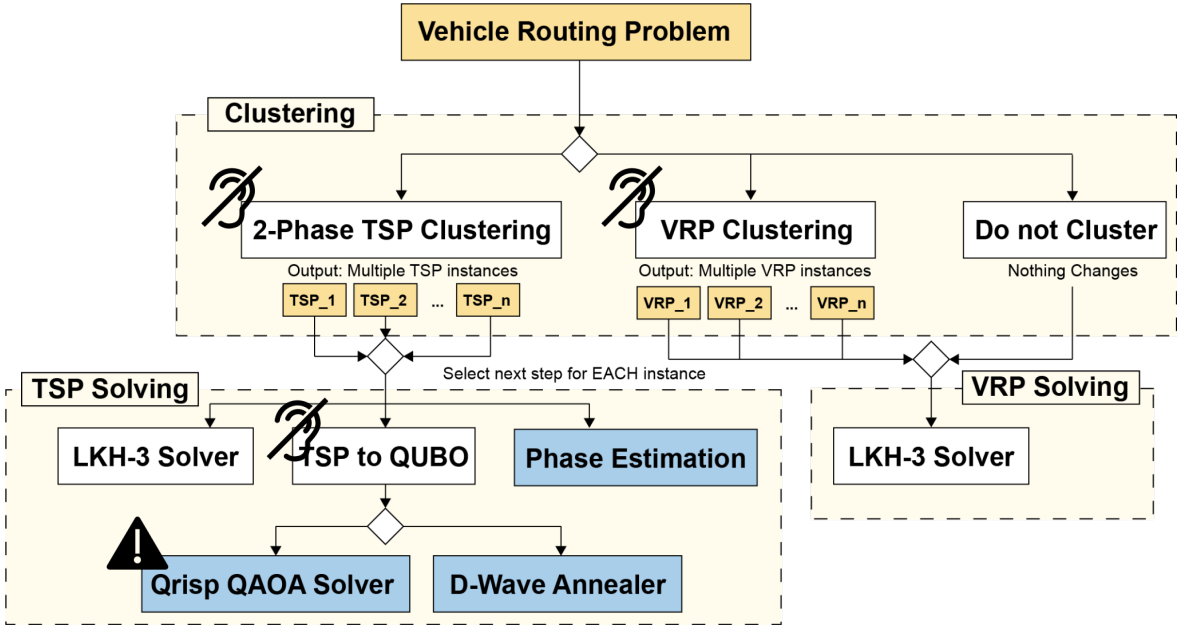
ProvideQ API

Try out the API:





FW: Formalizing Meta-Solver Strategies



Example Tags:

#invalid == algorithm might return an invalid result

#optimal == algorithms always finds an optimal solution

#unsound == transformation could alter the solution space

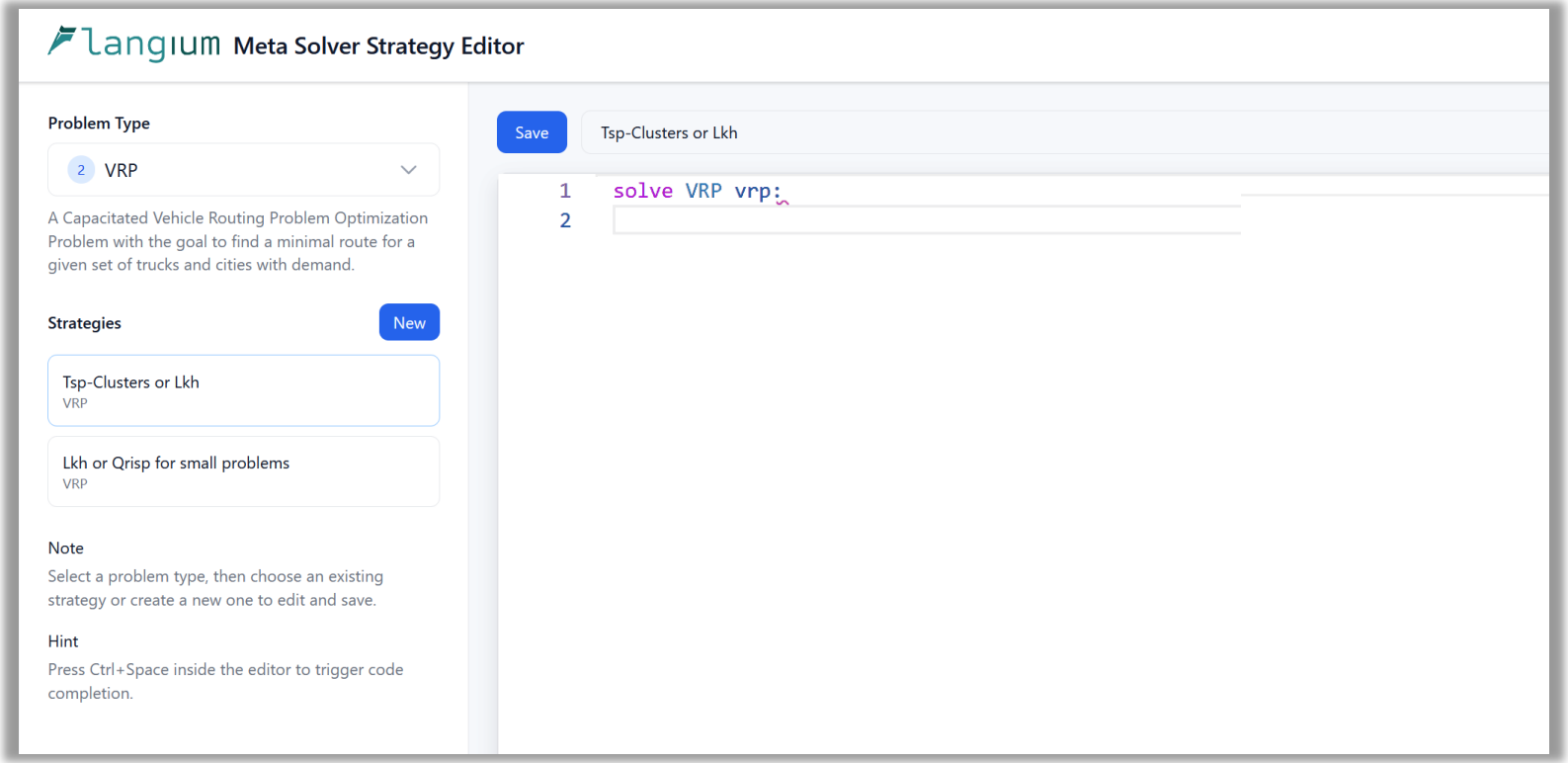
Rule 1 -- Invalid Paths:
A Path is invalid as soon as one subroutine is invalid.

Rule 2 -- Optimal Paths:
A Path is optimal when all subroutines are optimal.

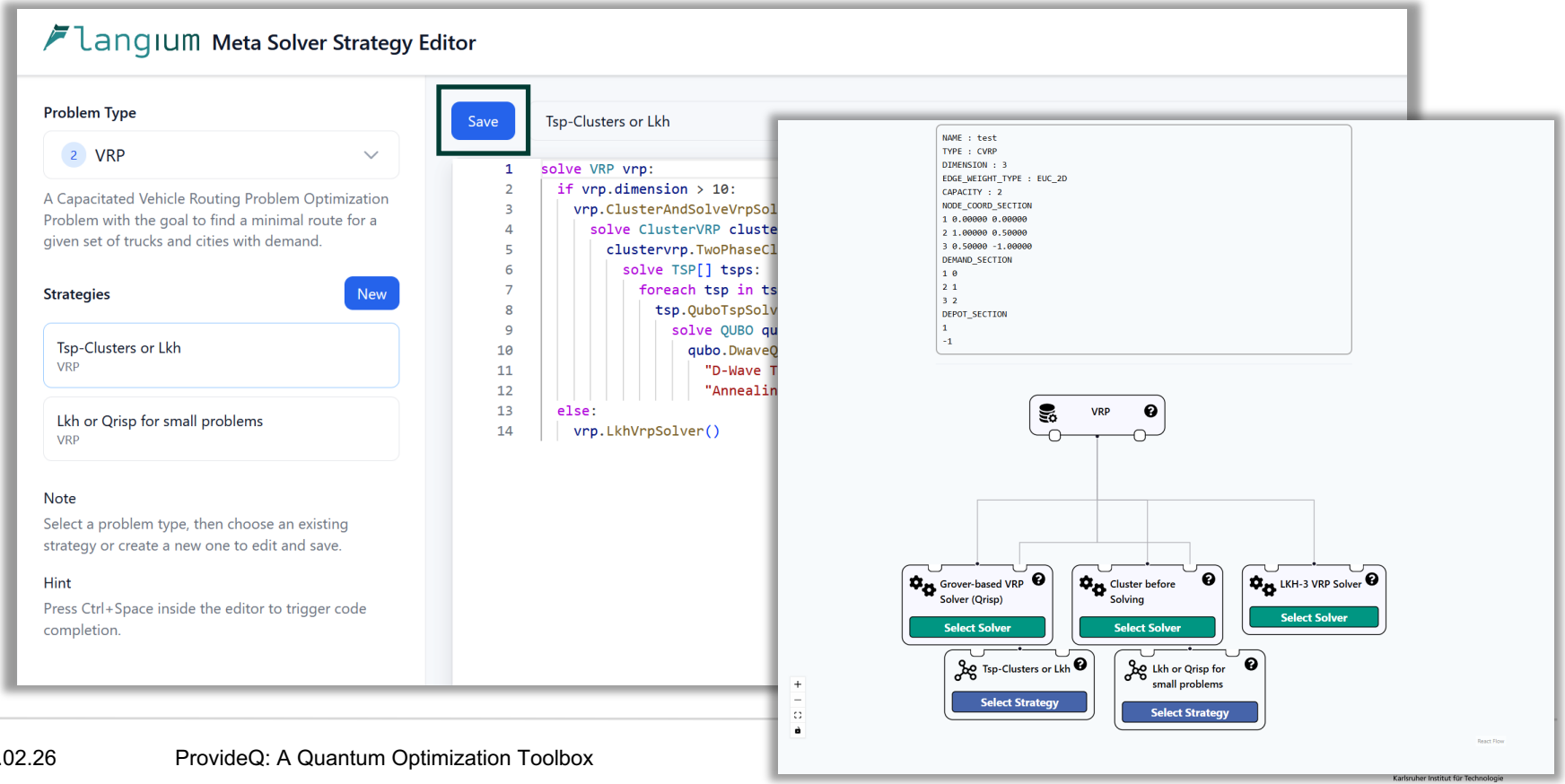
FW: Meta-Solver Language

```
1   solve VRP vrp:
2     if vrp.dimension > 10:
3       vrp.ClusterAndSolveVrpSolver():
4         solve ClusterVRP clustervrp:
5           clustervrp.TwoPhaseClusterer():
6             solve TSP[] tsps:
7               foreach tsp in tsps:
8                 tsp.QuboTspSolver():
9                   solve QUBO qubo:
10                    qubo.DwaveQuboSolver(
11                      "D-Wave Token" = "token",
12                      "Annealing Method" = "sim")
13    else:
14      vrp.LkhVrpSolver()
```

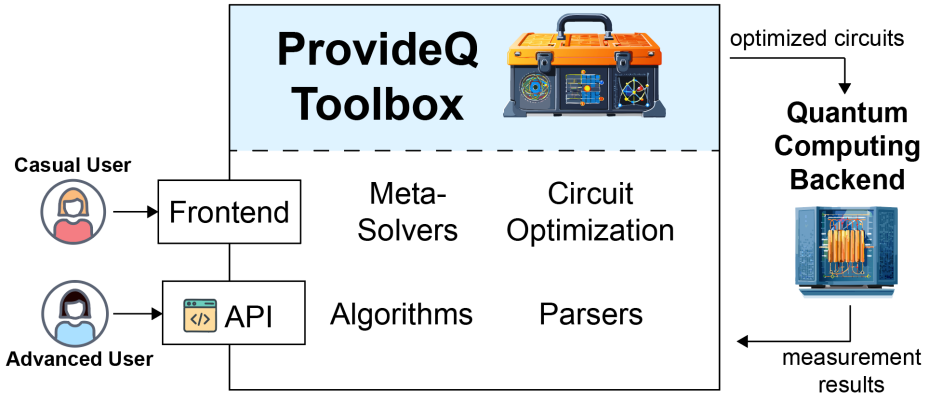
FW: Meta-Solver Language



FW: Meta-Solver Language



Conclusion



Find us at:
https://github.com/ProvideQ
https://provideq.kit.edu/