

# Quantum Motion Segmentation

## Supplementary Material

Federica Arrigoni<sup>1</sup>, Willi Menapace<sup>2</sup>, Marcel Seelbach Benkner<sup>3</sup>, Elisa Ricci<sup>2,4</sup>,  
and Vladislav Golyanik<sup>5</sup>

<sup>1</sup> Politecnico di Milano, Italy

<sup>2</sup> University of Trento, Italy

<sup>3</sup> University of Siegen, Germany

<sup>4</sup> Bruno Kessler Foundation, Italy

<sup>5</sup> MPI for Informatics, Germany

This supplementary document provides further details on our experiments.

**Competing methods.** MODE [3], SYNCH [2] and Xu et al. [7] are evaluated in Matlab on a 2020 MacBook Pro with 1.4 GHz processor and 8 GB RAM.

**Q-MSEG Dataset.** We generate a new dataset for motion segmentation with ground-truth annotations, which comprises six images depicting three planar objects captured from diverse viewpoints, shown in Fig. 1. Each object is manually annotated with respectively 10, 11 and 12 keypoints, selected on highly-textured locations. Several motion segmentation problems are derived from the dataset by randomly sampling a subset of points/images/motions. In particular, we consider 14 different choices for the number of points/images/motions (see Table 1), and we sample 20 problem instances for each configuration, resulting in 280 problem instances in total. See Tab. 2 and the main paper for results.

Table 1: Statistics of the Q-MSEG Dataset. Each configuration has  $n$  images and  $d$  motions. In each image, there are  $m_1$  points in the first motion,  $m_2$  in the second motion and  $m_3$  in the third motion (if available). The total number of logical qubits is reported for each configuration, which is given by  $dn(m_1 + m_2 + m_3)$ .

# Qubits/Bin. Var.:	96	102	120	126	128	136	160	168	180	190	200	216	220	243
$n$	3	3	3	3	4	4	4	4	5	5	5	3	5	3
$d$	2	2	2	2	2	2	2	2	2	2	2	3	2	3
$m_1$	8	8	10	10	8	8	10	10	9	9	10	8	11	8
$m_2$	8	9	10	11	8	9	10	11	9	10	10	8	11	9
$m_3$	-	-	-	-	-	-	-	-	-	-	-	8	-	10

**Parameter Selection.** In the tests with Q-MSEG and Hopkins, we use  $\lambda_1 = 10.0$  (v1) and  $\lambda_2 = 10.0$ ,  $\lambda_3 = 4.0$  (v2). For the synthetic experiments, we set the parameters as follows:  $\lambda_1 = 14.0$  (v1) and  $\lambda_2 = 27.5$ ,  $\lambda_3 = 3.2$  (v2). To select parameters for our method, we perform a systematic (exhaustive) grid search on small problem instances (with 24 bits). Talking in terms of the energy



Fig. 1: Images from the Q-MSEG dataset depicting three planar motions.

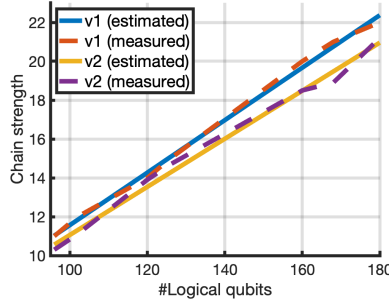


Fig. 2: Chain strength versus number of logical qubits in Q-MSEG dataset.

landscape, the purpose of this step is to find ranges of  $\lambda$  which push the lowest-energy solution of the target QUBO as close as possible to the desired ground-truth solution. We observe that the obtained parameter ranges generalise well to larger problems when solved on a QPU (*i.e.*, all tested problem sizes ranging from 96 qubits to 243 qubits). A similar observation was made in QSync [4]. Note that parameters can be selected in comparably large ranges, which allows safely using the same parameters across many—and much larger—problems. In particular, the parameters selected for our Q-MSEG dataset generalize well to Hopkins data without any tuning. The chain strength required to keep the chains of physical qubits intact, is chosen as a linear function  $c(k) = ak + b$  in the number of the logical qubits. We set  $a = 0.13508$ ,  $b = -1.94207$  and  $a = 0.1238$ ,  $b = -1.3180$  for v1 and v2, respectively. The coefficients  $a$  and  $b$  of function  $c$  are calculated by a linear regression applied to the measured average maximum chain lengths observed in exemplary problems of different sizes (see Fig. 2).

**Evaluation on Adv1.1.** Table 2 reports our evaluation on D-Wave Advantage1.1 (Adv1.1). For reference, we also report results on Adv4.1 and those of competing methods, which are copied from the main paper. Results show that the accuracy on Adv1.1 is slightly lower than on Adv4.1, and the latter can solve larger problems. This is also reflected in the number of found ground-truth solutions and the probability of finding a ground-truth solution. Moreover, larger problems can be minor-embedded to Adv4.1 compared to Adv1.1. We also observe that for small problems, the accuracy of our method on Adv1.1 and Adv4.1 is similar, whereas for larger problems, Adv4.1 performs consistently better. This observation agrees with the statement done in Appendix A.1 *Performance on Native Inputs* of the technical report by McGeoch and Farré [5].

Table 2: The accuracy (1.0 is the best) and its standard deviation for several methods on Q-MSEG dataset. For QUMoSEG, we also report the number of found ground-truth solutions (“#sol”) and the probability (“prob”) of measuring them as lowest-energy samples, and we perform evaluation both on Adv1.1 and Adv4.1.

# Qubits/Bin. Var.:		96	102	120	126	128	136	160	168	180	190	200	216	220	243
Mode [3]	acc	0.93	0.93	0.96	0.93	<b>0.97</b>	0.97	<b>0.98</b>	<b>0.99</b>	<b>0.98</b>	<b>0.99</b>	<b>0.99</b>	<b>0.93</b>	<b>1</b>	<b>0.94</b>
	std	0.07	0.06	0.05	0.06	0.04	0.05	0.02	0.01	0.02	0.007	0.01	0.04	0	0.04
Synch [2]	acc	0.93	0.94	0.95	0.95	0.84	0.92	0.97	1	0.89	0.95	0.90	0.94	0.99	0.92
	std	0.14	0.15	0.08	0.14	0.21	0.15	0.03	0	0.15	0.15	0.20	0.06	0.003	0.11
Xu et al. [7]	acc	0.89	0.89	0.94	0.75	0.96	0.97	0.86	0.86	0.97	0.88	0.96	0.77	0.83	0.74
	std	0.18	0.12	0.08	0.14	0.05	0.04	0.18	0.16	0.04	0.10	0.06	0.09	0.15	0.05
QUMoSEG-v1 ( Adv4.1 )	acc	<b>0.97</b>	<b>0.97</b>	<b>0.97</b>	<b>0.96</b>	0.95	<b>0.98</b>	<b>0.98</b>	<b>0.99</b>	<b>0.98</b>	<b>0.99</b>	<b>0.99</b>	0.64	–	–
	std	0.04	0.03	0.03	0.05	0.11	0.03	0.02	0.02	0.02	0.01	0.01	0.10	–	–
	#sol	9	5	6	11	6	11	8	9	7	6	9	0	–	–
	prob	0.16	0.07	0.03	0.03	0.04	0.05	0.008	0.0095	0.0097	0.0007	0.006	0	–	–
QUMoSEG-v2 ( Adv4.1 )	acc	0.96	<b>0.97</b>	0.95	0.94	0.89	0.89	0.88	0.85	0.74	0.75	0.79	0.59	0.75	0.58
	std	0.03	0.02	0.03	0.02	0.09	0.04	0.03	0.10	0.12	0.07	0.04	0.10	0.06	0.07
	#sol	2	2	0	0	0	0	0	0	0	0	0	0	0	0
	prob	0.0001	0.0001	0	0	0	0	0	0	0	0	0	0	0	0
QUMoSEG-v1 ( Adv1.1 )	acc	<b>0.97</b>	<b>0.97</b>	<b>0.97</b>	<b>0.96</b>	0.94	0.97	0.97	0.97	0.97	–	–	–	–	–
	std	0.04	0.02	0.03	0.05	0.11	0.04	0.03	0.03	0.03	–	–	–	–	–
	#sol	9	5	6	11	6	9	5	3	5	–	–	–	–	–
	prob	0.13	0.05	0.03	0.01	0.03	0.02	0.0003	0.0002	0.0007	–	–	–	–	–
QUMoSEG-v2 ( Adv1.1 )	acc	0.95	0.96	0.93	0.92	0.86	0.86	0.97	0.81	0.69	0.67	0.71	0.59	0.66	–
	std	0.05	0.02	0.03	0.03	0.09	0.06	0.03	0.07	0.07	0.08	0.05	0.08	0.05	–
	#sol	1	0	0	0	0	0	5	0	0	0	0	0	0	–
	prob	0.0001	0	0	0	0	0	0.0003	0	0	0	0	0	0	–

**Hopkins Dataset.** Starting from the well-known Hopkins155 dataset [6], we create small problems (with 120-240 qubits) by sampling a subset of images/points from the *cars2-06.g23* sequence, which represents two moving objects in an outdoor environment. See Tab. 3 for the details. For each configuration, 20 instances were created, resulting in 400 examples in total. See the main paper for results on this dataset.

Table 3: Statistics of sub-problems sampled from the Hopkins dataset [6]. Each configuration has 3 images and 2 motions. In each image, there are  $m_1$  points in the first motion and  $m_2$  in the second motion. The total number of logical qubits is reported for each configuration, which is given by  $6(m_1 + m_2)$ .

# Qubits/Bin. Var.:	120	126	132	138	144	156	162	168	174	180	186	192	198	204	210	216	222	228	234	240
$m_1$	10	10	11	11	12	13	13	14	14	15	15	16	16	17	17	18	18	19	19	20
$m_2$	10	11	11	12	12	13	14	14	15	15	16	16	17	17	18	18	19	19	20	20

**Minor Embeddings.** Fig. 3 shows an exemplary minor embedding for a problem with 96 qubits. Fig. 3-left is a photograph of the Adv4.1 processor. Fig. 3-(middle) and -(right) show the qubits graph of the logical problem (96 nodes) and its embedding into the processor (803 nodes), respectively. Each node in the logical problem graph is colored according to whether the qubit was

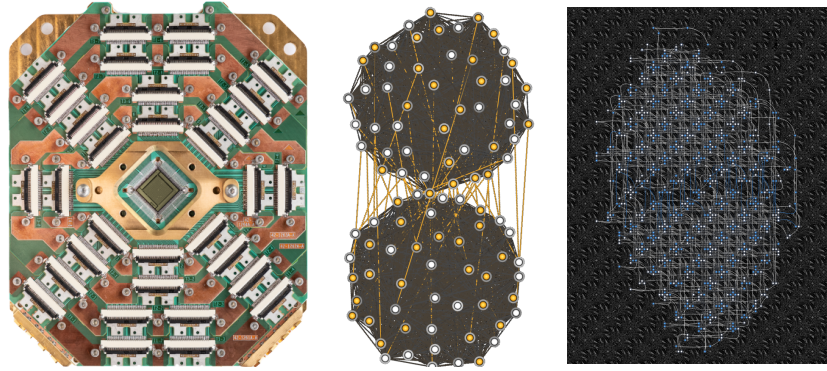


Fig. 3: D-Wave Adv4.1 quantum annealer (left; reproduced with permission from D-Wave Systems), graph of 96 logical problem qubits (middle) and its minor embedding with 803 physical nodes (right).

measured as one (yellow) or zero (white). The figure on the right shows a section of the processor centered around the region with active qubits. The colors denote qubits measured—according to the Ising model—as +1 (blue) or −1 (white).

**QuMoSeg Demo Script.** We provide a demo example of our QUMoSEG (v1 and v2)<sup>6</sup>. Its execution requires three steps: 1) Creating a Leap2 account [1], 2) Opening an IDE workspace and uploading the supplied folders *QuMoSeg\_Data* and *QuMoSeg\_Results* and the main script *QuMoSeg\_Demo.py* to it, and 3) Running the demo by executing in the workspace’s terminal the command “*python QuMoSeg\_Demo.py*”. After the successful execution, one will see the statistics in the terminal, as shown in Fig. 4. Note that the exact numbers can differ slightly from run to run. The script also allows visualising the connectivity (coupling) graphs of the logical problem (Fig. 5-(left)) and the histogram of the obtained sample energies (Fig. 5-(right)). See *QuMoSeg\_Demo.py* for further details.

```

MEAN_ERROR           = 0.9661764705882352
STD_DEV_ERROR        = 0.025933862182745408
MEAN_ERROR_ALL_SAMPLES = 0.9843137254901961
STD_DEV_ERROR_ALL_SAMPLES = 0.0181055003309145
MEAN_E_DIFF          = 6.15
STD_DEV_E_DIFF       = 5.769931131578425
MEAN_MAX_CHAIN_LENGTHS = 11.85
STD_DEV_MAX_CHAIN_LENGTHS = 1.3088765773505318
MEAN_PHYS_VAR        = 822.4
STD_DEV_PHYS_VAR     = 57.63167988455472
N_GT_SOLUTIONS       = 5
MEAN_P_OPTIMAL        = 0.0504

```

Fig. 4: Statistics printed in the terminal by the provided demo example after successful execution of the script.

<sup>6</sup> See the project page <https://4dqv.mpi-inf.mpg.de/QuMoSeg/>

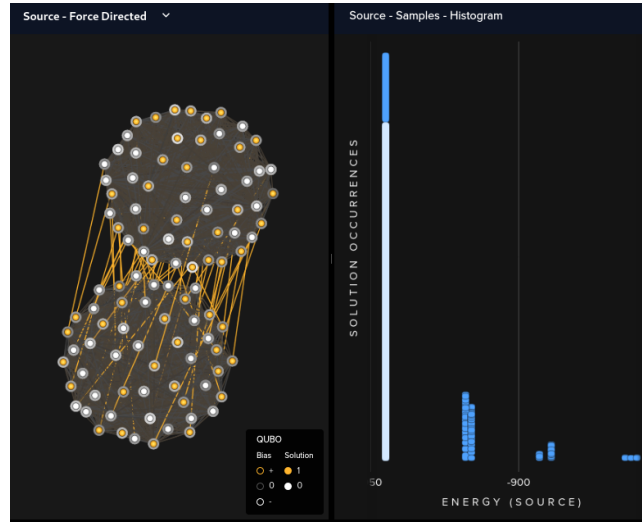


Fig. 5: Connectivity graph of an exemplary logical problem (left) and the histogram of the obtained sample energies (right).

## References

1. <https://cloud.dwavesys.com/leap/> 4
2. Arrigoni, F., Pajdla, T.: Motion segmentation via synchronization. In: IEEE International Conference on Computer Vision Workshops (ICCVW) (2019) 1, 3
3. Arrigoni, F., Pajdla, T.: Robust motion segmentation from pairwise matches. In: Proceedings of the International Conference on Computer Vision (2019) 1, 3
4. Birdal, T., Golyanik, V., Theobalt, C., Guibas, L.: Quantum permutation synchronization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2021) 2
5. McGeoch, C., Farré, P.: The advantage system: Performance update (2021-10-01). [https://www.dwavesys.com/media/qdmlgsu1/14-1054a-a\\_advantage\\_system\\_performance\\_update.pdf](https://www.dwavesys.com/media/qdmlgsu1/14-1054a-a_advantage_system_performance_update.pdf) (2021), online; accessed on the 20.02.2022 2
6. Tron, R., Vidal, R.: A benchmark for the comparison of 3-d motion segmentation algorithms. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2007) 3
7. Xu, X., Cheong, L.F., Li, Z.: 3d rigid motion segmentation with mixed and unknown number of models. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019) 1, 3