

# Levitron: Combining Ground and Lifted Planning

Augusto B. Corrêa<sup>1</sup>, Guillem Francès<sup>2</sup>, Markus Hecher<sup>3</sup>, Davide Mario Longo<sup>4</sup>, Jendrik Seipp<sup>5</sup>

<sup>1</sup>University of Basel, Switzerland

<sup>2</sup>Independent Researcher

<sup>3</sup>Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, USA

<sup>4</sup>TU Wien, Institute of Logic and Computation, Austria

<sup>5</sup>Linköping University, Sweden

augusto.blaascorrea@unibas.ch, guillem.frances@gmail.com, hecher@mit.edu,

davidem.longo@gmail.com, jendrik.seipp@liu.se

In this work, we describe the *Levitron* planner.<sup>1</sup> *Levitron* is essentially a wrapper around a lifted and a ground planner. It combines the lifted planner *Powerlifted* (Corrêa et al. 2023a) with the ground planner *Scorpion Maidu* (Corrêa et al. 2023b). Both are sequential portfolio planners but they have complementary strengths: *Scorpion Maidu* is efficient in tasks of moderate size; *Powerlifted* works well on larger tasks that are challenging to ground.

*Levitron* uses *Scorpion Maidu* as a default component, and *Powerlifted* as a fallback when the translator of *Scorpion Maidu* fails. It participated in the satisficing and the agile tracks, and *Scorpion Maidu*’s translator is given a different time limit depending on the track. For the satisficing track, this limit is 15 minutes. For the agile track, the limit is 3 minutes. If the translator reaches the time limit or surpasses the memory limit (of 8 GiB for both tracks), *Levitron* aborts *Scorpion Maidu* and calls *Powerlifted*. If the translator finishes correctly, *Powerlifted* is never used.

We do not describe the details of *Scorpion Maidu* and *Powerlifted* here, and instead refer to their planner abstracts for a complete description (Corrêa et al. 2023a; 2023b).

## Results

*Levitron* was the joint winner of the satisficing track together with *Scorpion Maidu*. The similar performance of both planners is not surprising, as *Scorpion Maidu* is the main component of *Levitron*. Nevertheless, the planners differed in a few domains. Our hypothesis was that *Levitron* would perform better than *Scorpion Maidu* in domains where *Powerlifted* also did so. In the best case scenario, *Levitron*’s score and coverage would be the maximum between *Maidu* and *Powerlifted* scores and coverage. In the domains used in the competition, our hypothesis seems correct. But things are not so simple.

Table 1 shows the coverage and score comparisons between *Levitron* and its two component planners, *Scorpion Maidu* and *Powerlifted*. These are the official results from

<sup>1</sup>A *Levitron* is a toy that demonstrates the principles of magnetic levitation, in which a spinning top is *lifted* and suspended above a magnetic base. The spinning top contains a magnet with its north pole facing outward, while the magnetic base has a north pole facing upward. The repelling forces between these two north poles generate the lift required for the top to levitate.

	Levitron		Maidu		PWL	
	S	C	S	C	S	C
Folding (20)	<b>9</b>	<b>8.66</b>	7	6.80	8	7.69
Folding-norm (20)	8	7.53	7	6.37	<b>10</b>	<b>9.69</b>
Labyrinth (20)	0	0.00	0	0.00	–	–
Quantum-Layout (20)	<b>20</b>	<b>19.63</b>	<b>20</b>	<b>19.63</b>	<b>20</b>	16.73
Recharging-Robots (20)	<b>14</b>	<b>13.78</b>	<b>14</b>	<b>13.78</b>	0	0.00
Recharging-Robots-norm (20)	<b>14</b>	<b>13.94</b>	<b>14</b>	<b>13.94</b>	0	0.00
Ricochet-Robots (20)	<b>17</b>	<b>11.44</b>	<b>17</b>	11.36	–	–
Rubiks-Cube (20)	<b>20</b>	<b>14.16</b>	<b>20</b>	13.08	0	0.00
Rubiks-Cube-norm (20)	<b>20</b>	13.63	<b>20</b>	<b>14.16</b>	0	0.00
Slitherlink (20)	<b>2</b>	<b>2.00</b>	0	0.00	<b>2</b>	<b>2.00</b>
Slitherlink-norm (20)	4	4.00	<b>6</b>	<b>6.00</b>	2	2.00
Sum (220)	<b>128</b>	<b>108.77</b>	125	105.13	42	38.12

Table 1: Coverage (C) and quality score (S) comparison between *Levitron*, *Scorpion Maidu* (“*Maidu*”), and *Powerlifted* (“*PWL*”). Best results marked in bold. Entries with a dash use PDDL fragments not supported by *Powerlifted*.

the competition. For a few domains, the organizers provided two versions: one using a more expressive fragment of PDDL (e.g., conditional effects, axioms), and another version where these features were compiled away (called “normalized” versions). The final results only considered the best version of each domain. We show results of both for completeness.

In the domains where *Powerlifted* performed better than *Maidu* (Folding, Folding-norm, Slitherlink), *Levitron* also did. For example, in the Folding domain *Levitron* capitalized on the strengths of both planners. However, in Slitherlink-norm, *Levitron* landed in between *Powerlifted* and *Maidu*.

Randomness plays an important role here as well. For example, *Levitron* scored better than *Maidu* in the Ricochet-Robots domain. But *Powerlifted* does not even support the PDDL fragment used in Ricochet-Robots, so it could not possibly have helped *Levitron*. At closer inspection, we see that the portfolio configuration of *Levitron* that computed the best plan, only found this plan in the very last seconds. Due to noise, the same configuration of *Maidu* did not have enough time to find this plan. The same behavior happened in the other direction (e.g., in Rubiks-Cube-norm).

## References

Corrêa, A. B.; Francès, G.; Hecher, M.; Longo, D. M.; and Seipp, J. 2023a. The Powerlifted Planning System in the IPC 2023. In *Tenth International Planning Competition (IPC-10): Planner Abstracts*.

Corrêa, A. B.; Francès, G.; Hecher, M.; Longo, D. M.; and Seipp, J. 2023b. Scorpion Maidu: Width Search in the Scorpion Planning System. In *Tenth International Planning Competition (IPC-10): Planner Abstracts*.