Towards Qualitative Constraint-Based HTN Planning: A Research Agenda

Tobias Schwartz, Christopher Walther, Diedrich Wolter

University of Luebeck, Lübeck, Germany
{tobias.schwartz, c.walther, diedrich.wolter}@uni-luebeck.de

Abstract

Hierarchical task networks offer an intuitive way to model robot tasks, yet bringing HTN planning to real environments exposes aspects – spatial resource constraints, uncertainty, the need to repair plans – that are not naturally captured by its classical formulation. Qualitative constraint networks are a well-established tool for reasoning about time and space, but remain underused in the planning community. Two recent results suggest that this is worth revisiting: HTN plan verification and grounded STRIPS planning can both be encoded in the same qualitative constraint language, one capturing hierarchical refinement and the other action executability. Taking these as a starting point, this paper outlines a research agenda on qualitative constraint-based HTN planning and sketches five directions in which the approach can be developed: combining structural and state verification, extending the network to spatial reasoning, geometric safety constraints, plan and domain repair, and structural robustness of verified plans. The paper is meant as a position paper that opens these directions to discussion rather than settling them.

1 Introduction and Motivation

Intelligent robotics is one of the most prominent application domains for planning, having fascinated researchers since the seminal Shakey project in the late 1960s. Planning with hierarchical task networks (HTN planning) has gained particular popularity in this area: tasks are organised hierarchically, with *compound tasks* decomposing into ordered networks of subtasks via *decomposition methods* until only directly executable *primitive tasks* (actions) remain. This task-centric view offers intuitive means to model robot behaviour and allows expert knowledge about the domain to be encoded in the task descriptions themselves. While HTN planning already offers many challenging research questions in its own right, a look into robotic applications reveals further under-explored aspects. In particular, every mobile robot faces spatial resource constraints imposed by its working environment. At the same time, the rigidity of pure planning may not suit real-world environments: the abstraction from an uncertain environment (for which not even stochastic models may be available) to a planning domain can introduce incongruence between plans and reality, for example when unforeseen changes occur during execution. While all of the topics mentioned so far have already been addressed in individual research, general approaches are sought after.

In this paper, we argue for a research agenda on using qualitative reasoning as an underlying framework for HTN planning, as it can provide a principled approach to several aspects relevant to bringing HTN planning to applications. Qualitative spatial and temporal reasoning is an established field of research that adopts a constraint-based approach. Two recent results (Schwartz and Wolter 2026; Wolter and Schwartz 2026) encode HTN plan verification and grounded STRIPS planning in the same qualitative constraint language. They do not yet amount to a unified planning framework, but they suggest that hierarchical structure and action executability can be studied in the same representation. This paper takes that observation as a starting point and sketches a research agenda built around it.

As a running example, consider a service robot that delivers a cup of coffee from a kitchen to an office. The compound task *deliver-coffee* decomposes hierarchically into picking up the coffee (*pick-up*), navigating to the destination (*navigate*), and placing it there (*place*); it admits alternative routes, comes with action preconditions and effects such as *robot-at* and *holding*, and may need to be repaired if part of the route becomes unavailable. Each subsequent section refers back to this example.

The paper is organised as follows. Section 2 recalls two encodings that motivate the agenda, and Section 3 develops five directions in which a qualitative constraint-based view of HTN planning can be taken further. Directions are open problems since the paper is meant to invite discussion on which of them are most promising, and where connections to existing constraint-based work should be drawn.

2 Qualitative Reasoning for Planning

Qualitative constraint networks (QCNs) are a well-established framework for reasoning about relations over infinite domains such as time intervals or spatial regions (Dylla et al. 2017; Sioutis and Wolter 2021). Formally, QCNs are constraint networks (X, C, D) consisting of a finite set of variables X that take values from a domain D. Unlike classic CSPs, there is only one joint domain D. Constraints C are (mostly binary) relational expressions $(x\ R\ y)$ where variables x, y are related by relation R from a so-called qualitative calculus (Dylla et al. 2017). Calculi are built from a finite set of jointly exhaustive and pairwise disjoint base relations and constitute a relational algebra that

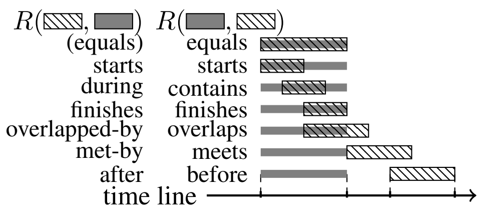

Figure 1: Illustration of base relations in the Allen calculus.

serves as constraint language. Unions of base relations express indefinite knowledge, so that disjunction and uncertainty can be stated directly in the language. Returning to the coffee-delivery example, Allen's interval algebra (Allen 1983) lets us state that the subtasks *pick-up*, *navigate* and *place* must all occur *during* the compound task *deliver-coffee*, and that *pick-up meets navigate* and *navigate meets place*, without committing to specific times. We illustrate the set of base relations for these calculi in Figure 1. The $\mathcal{INDU}$ calculus (Pujari, Kumari, and Sattar 1999) extends Allen's 13 relations with a 3-way comparison of interval durations (shorter, equal length, longer), letting us add qualitative information such as "*navigate* is longer than *pick-up*" at the same symbolic level. The calculus has 25 rather than $3 \times 13 = 39$ base relations because duration and temporal arrangement are partially interdependent. For example, an interval needs to be shorter to occur during another interval.

Constraint propagation then narrows disjunctive relations until either a consistent arrangement or a contradiction emerges. For most qualitative calculi, deciding consistency of a given QCN is NP-complete but several tractable subclasses are known (Dylla et al. 2017).

What makes QCNs a natural fit for planning is that the structural notions used in planning (sequencing of actions, containment of subtasks within a parent task, overlap of concurrent activities) are themselves qualitative temporal relations. While QCN-based reasoning is a form of constraint satisfaction, the explicit qualitative vocabulary keeps the encoding close to how one thinks about a plan; numerical constraints can still be expressed when needed, but they do not dominate the representation. In the following, we briefly summarise two recent results that motivate this paper: one on HTN plan verification and one on action-state validity in STRIPS planning.

HTN Plan Verification via $\mathcal{INDU}$ QCNs

HTN planning extends classical planning by organising tasks hierarchically (Ghallab, Nau, and Traverso 2016; Bercher, Alford, and Höller 2019). An HTN domain consists of a set of facts describing the world, *primitive tasks* corresponding to directly executable actions, *compound tasks* representing abstract activities that must be refined, and *decomposition methods* that specify how compound tasks can be broken down into subtasks with associated ordering constraints. Planning means recursively decomposing an initial compound task until only primitive tasks remain, forming a *task network* whose linearisations are executable plans.

Given an HTN domain and a candidate primitive task network, *plan verification* asks whether that network can be derived from the initial compound task through valid decompositions and ordering refinements; in the grounded setting, the problem is NP-complete (Behnke, Höller, and Biundo 2015). Prior approaches have largely relied on SAT encodings (Behnke, Höller, and Biundo 2017; Lin, Behnke, and Bercher 2023) or parsing-based techniques (Barták et al. 2020; Pantůčková, Ondrčková, and Barták 2024). A recent line of work (Schwartz and Wolter 2026) instead takes the view that HTN decomposition is inherently temporal and makes this structure explicit as a QCN.

Taking up the coffee-delivery example once more, the decomposition of *deliver-coffee* into *pick-up*, *navigate* and *place* requires that all three subtasks lie within the execution of the parent, and that each meets the next. Every decomposition step induces such containment and ordering constraints, and every primitive task contributes a concrete interval that has to fit into this scaffold. Capturing these relations in the $\mathcal{INDU}$ calculus yields a QCN whose solutions are exactly the valid temporal arrangements of the plan under the given HTN. Because an HTN domain typically offers several decomposition methods for the same compound task, the encoding introduces a dedicated separator interval whose position in the network distinguishes task occurrences belonging to the chosen decomposition from those of alternative methods that were not selected. The construction is polynomial in the size of the input and the encoding is sound and complete for verification.

What matters for the rest of this paper is that solving the QCN does not only yield a yes/no answer. It also tightens the QCN, with the remaining relations describing the space of admissible temporal arrangements of the verified plan. This observation suggests natural connections to questions of robustness and repair, which we return to in Section 3.

Action-State Validity via $\mathcal{INDU}$ QCNs

The structural verification view above is complemented by recent work on planning states and action executability (Wolter and Schwartz 2026). In that setting, grounded STRIPS planning, including optional duration constraints, is reduced to an $\mathcal{INDU}$ QCN whose solutions correspond to valid plans. Actions are represented by timestamped interval variables, propositional facts by proposition intervals whose arrangement with respect to action boundaries encodes validity, and action preconditions and effects appear as $\mathcal{INDU}$ constraints between them.

In our running example, the primitive task *pick-up* becomes an action interval with preconditions such as *robot-at(kitchen)* and effects such as *holding(cup)*; each of these facts is itself an interval whose validity must cover the moment at which *pick-up* consults or asserts it. The $\mathcal{INDU}$ constraints between these intervals say, for instance, that *robot-at(kitchen)* must hold *during pick-up*'s precondition check, and that *holding(cup)* must start no later than *pick-up* ends. What would be a separate fluent-timeline layer elsewhere thus lives directly in the same constraint language as the temporal arrangement of actions. This also leads to a shift of focus from hierarchical refinement to the flow of facts along a sequence of actions.

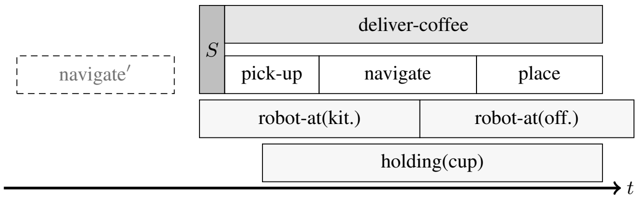

Figure 2: $\mathcal{INDU}$ network combining hierarchical refinement, action selection and fact validity for the coffee-delivery example; vertical displacement for readability only.

3 Proposed Research Agenda

Taking the two recent results described above as a starting point, we identify five directions in which a qualitative constraint-based view of HTN planning can be developed. The first brings hierarchical refinement and action executability into a single network, the second extends that network to spatial relations, the third considers spatio-temporal constraints on plans, and the fourth and fifth ask what the network can tell us about repair and robustness once verification has been carried out. Each subsection sketches the direction and lists the open questions we see; none of them is a result we already have.

Combining Structural and State Verification

The two encodings discussed in the previous section live in the same constraint language and differ only in what they constrain: one the hierarchical refinement of tasks, the other the flow of facts along a sequence of actions.

A natural next step is to unify both views into a single $\mathcal{INDU}$ QCN in which tasks, actions and propositional facts are all interval variables of the same network. Figure 2 sketches this for the coffee-delivery example: the compound task *deliver-coffee contains* the primitive tasks *pick-up*, *navigate* and *place*, which *meet* in sequence; proposition intervals for *robot-at(kitchen)*, *holding(cup)* and *robot-at(office)* are single spans that *meet* the action boundaries that flip their validity; and an alternative refinement of *navigate* via a second corridor is parked left of the separator S, as part of the network but outside the selected plan. Primitive tasks of the HTN encoding naturally correspond to the action intervals of the planning encoding, and the separator interval that isolates rejected decompositions suggests a connection to the start interval that separates selected from unselected actions. A successful combination would yield one encoding that simultaneously describes which hierarchical refinements are admissible, which actions are selected at which position, and how facts flow between them.

If everything can be brought into the same constraint network, the boundary between planning and verification may become a matter of *how much* is left open rather than a difference in kind. A fully ground input task network yields the verification problem of the previous section; leaving parts of the primitive task sequence unconstrained opens a path from verification toward planning starting from an initial compound task; and anything in between corresponds to a partially committed plan that still admits several refinements. This is useful in practice: the coffee-delivery robot may not need to commit up front to which of two corridors it will take from the kitchen to the office, since both reach the same door and the remainder of the plan does not depend on the choice. Qualitative constraint reasoning handles this situation directly, since disjunctive relations can be kept in the network and propagated together with the committed parts; constraint propagation tightens what can be decided and leaves open what does not need to be. This perspective appears well suited to the long-standing idea of least-commitment planning (Weld 1994), and more generally to partial-order and lifted planners that defer ordering and binding decisions until forced: qualitative constraint networks offer a uniform representation in which such commitments (whether over task ordering, action selection or fact validity) are all disjunctive relations over the same set of interval variables. A promising research question is how to exploit this flexibility algorithmically, for instance by interleaving decomposition, action selection and constraint propagation so that verification effort is spent where it reduces uncertainty, and planning effort is spent only where the network has not yet committed. Thus, our approach inhabits a continuum between verification and planning, establishing relations to CSP- and SAT-based approaches for planning, such as FAPE (Dvorak et al. 2014; Bit-Monnot et al. 2020), ARIES (Bit-Monnot 2023), or Lilotane (Schreiber 2021).

Extending the QCN to Space

The encodings considered so far are temporal and relational, yet robot tasks are almost always spatial as well. The world has rooms, corridors, regions that cannot be entered simultaneously, and objects whose arrangement determines whether an action is meaningful at all. In the coffee-delivery example, the robot must reason about which rooms contain the kitchen and the office, which corridors connect them, and whether a door along the way is currently traversable.

A natural way to add this to the encoding is to express spatial aspects in the same qualitative constraint language already used for time. Qualitative spatial calculi such as RCC-8 (Randell, Cui, and Cohn 1992) describe region containment, overlap and disconnection symbolically, and they integrate with temporal calculi through constraint propagation in much the same way $\mathcal{INDU}$ does for intervals and durations. In the unified view sketched in the previous subsection, task intervals, action intervals, proposition intervals and spatial regions would all be variables of a single QCN, and hierarchical, temporal and spatial constraints would share the same closure reasoning. For the coffee-delivery robot this means that the containment of the kitchen in a room, the connection between corridors, and the temporal arrangement of *navigate* and *place* are all relations in one network, and constraint propagation couples them whenever an action or decomposition refers to both.

Earlier work on integrating spatial reasoning into HTN planning, most notably the CHIMP planner (Stock et al. 2015), couples the planner to a Meta-CSP (Mansouri and Pecora 2014, 2016) in which temporal and spatial reasoning are held in separate networks linked through meta-constraints, and search proceeds over world states. Our direction differs in where the spatial layer lives: not in a separate network consulted during state-based search, but as further relations in the same plan-space QCN that already carries hierarchy and actions. Whether this yields a practical advantage, and which spatial calculus strikes a useful balance between expressiveness and tractability at plan length, remains open.

From Symbolic Relations to Geometric Safety

Adopting a more concrete perspective of the underlying geometric space, the running example can be extended to impose safety constraints on plans that disallow the coffee from coming within some safety margin of the space above humans or a valuable rug at any point during its continuous movement. These requirements could be expressed as dense-time temporal logic formulas over a spatial domain in which objects are modelled as mesh-bounded regions. Consequently, we want the coffee and the unsafe zones to remain non-intersecting until the coffee has been delivered and similarly, whether the robot can fit through a door while holding the coffee may restrict which plans are safe and thus permitted. Specific geometry that requires far-ahead planning or specific motion to complete an action is akin to the difficult piano mover's problem (Schwartz and Sharir 1983).

While the entailed geometric problems themselves are beyond the scope of a QCN, a degree of integration could be beneficial as some sequences of actions may become unsafe under such plan constraints depending on the spatial context. It is a relevant direction to explore whether the logical aspects of the temporal constraints, which could be formulated in a fragment of Signal Temporal Logic (STL) (Maler and Nickovic 2004), can be efficiently encoded in a QCN, or whether the completion of the spatio-temporal model should remain entirely external.

Such a formula specifying the safety requirement may be sketched as $((cup \cap expand(unsafe, m)) = \emptyset)\ \mathcal{U}\ (cup \subset target)$ where $\mathcal{U}$ is the STL until operator, $expand(unsafe, m)$ is the unsafe region expanded by the safety margin m, and cup and $target$ are spatial regions corresponding to the coffee and the delivery location, respectively. The fit requirement could be expressed as $cup \cap robot = \emptyset$ alongside a goal and room geometry specification. Both formulas could be monitored during execution to trigger replanning if they are violated or a violation is imminent.

Plan and Domain Repair via Constraint Relaxation

When verification fails, the qualitative constraint view suggests not only rejecting the plan, but also asking how it may be repaired. In QCN terms, this amounts to identifying the smallest or otherwise best set of constraints whose relaxation restores consistency. Existing work on over-constrained qualitative networks and MaxQCN-style reasoning (Condotta et al. 2015; Condotta, Nouaouri, and Sioutis 2016; Sioutis 2023; Bayerkuhnlein, Schwartz, and Wolter 2024), already provides a natural technical starting point for this perspective.

For HTN planning, such repairs could concern both the current plan and the domain description itself. A failed verification network may indicate that a temporal relation should be weakened, that an alternative decomposition method should be selected, or that the domain lacks a suitable refinement for a situation that occurs in practice. Recent work on HTN repair via task insertion and deletion (Pantůčková and Barták 2025) points in a related direction, but the qualitative setting raises the more general question of what should count as a minimal repair when structural, temporal and eventually spatial constraints interact in the same model.

Structural Robustness of Plans

Once verification succeeds, the next question is no longer whether a plan is valid, but how gracefully it tolerates execution-time deviations. The tightened QCN already carries the material for such a judgement: for every pair of task intervals it records the set of $\mathcal{INDU}$ base relations that remain locally consistent with the decomposition. Atomic duration drift moves the actual relation between two intervals one conceptual neighbour at a time, so a feasible set that forms a connected region in the $\mathcal{INDU}$ Conceptual Neighborhood Graph (Freksa 1992) absorbs drift through smooth qualitative transitions, while a scattered set requires a discontinuous reconfiguration as soon as drift pushes a relation across a boundary.

This suggests a notion of *structural robustness* that falls out of the verification network without extra encoding, and that is sensitive to the topology of the feasible relation sets rather than only their size. Open questions we see here include how to turn this observation into a principled metric, how to aggregate across task pairs, and whether such a robustness signal can be used not only for post-hoc evaluation but also as a tie-breaker or search guidance inside a planner.

4 Discussion

The two encodings recalled in Section 2 already show that hierarchical refinement and action executability can be expressed in the same qualitative constraint language, and the directions outlined above suggest that the same language naturally extends to spatial reasoning, repair and robustness. What unifies them is not a single new encoding but a common vocabulary in which structural, temporal, spatial and executability aspects of a plan appear as relations over the same interval and region variables, and in which established constraint-reasoning techniques – closure, relaxation, conceptual neighbourhood – become available without leaving the planning layer. Seen this way, QCNs appear not merely as yet another constraint formalism for planning, but as a single representation in which several open planning questions can be formulated and related to one another.

This paper presents a research agenda rather than a finished framework. Each of the five directions is stated as an open problem, and whether the unified view ultimately simplifies matters or rediscovers known architectures by a different route is precisely what the agenda is meant to find out. Feedback on which directions are most promising, which are likely dead ends, and where the qualitative view should be connected to existing constraint-based work on planning is therefore particularly welcome.

References

Allen, J. F. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 832–843.

Barták, R.; Ondrčková, S.; Maillard, A.; Behnke, G.; and Bercher, P. 2020. A Novel Parsing-based Approach for Verification of Hierarchical Plans. In *Proc. of ICTAI 2020*, 118–125. IEEE.

Bayerkuhnlein, M.; Schwartz, T.; and Wolter, D. 2024. From Resolving Inconsistencies in Qualitative Constraints Networks to Identifying Robust Solutions: A Universal Encoding in ASP. In *Proc. of KI 2024*, volume 14992 of *LNCS*, 3–16. Springer.

Behnke, G.; Höller, D.; and Biundo, S. 2015. On the Complexity of HTN Plan Verification and Its Implications for Plan Recognition. In *Proc. of ICAPS 2015*, volume 25, 25–33. AAAI Press.

Behnke, G.; Höller, D.; and Biundo, S. 2017. This Is a Solution! (... But Is It Though?) - Verifying Solutions of Hierarchical Planning Problems. In *Proc. of ICAPS 2017*, 20–28. AAAI Press.

Bercher, P.; Alford, R.; and Höller, D. 2019. A Survey on Hierarchical Planning – One Abstract Idea, Many Concrete Realizations. In *Proc. of IJCAI 2019*, 6267–6275.

Bit-Monnot, A. 2023. Enhancing Hybrid CP-SAT Search for Disjunctive Scheduling. In *Proc. of ECAI 2023*, 255–262. IOS Press.

Bit-Monnot, A.; Ghallab, M.; Ingrand, F.; and Smith, D. E. 2020. FAPE: a Constraint-based Planner for Generative and Hierarchical Temporal Planning. *CoRR*, abs/2010.13121.

Condotta, J.; Nouaouri, I.; and Sioutis, M. 2016. A SAT Approach for Maximizing Satisfiability in Qualitative Spatial and Temporal Constraint Networks. In *Proc. of KR 2016*, 432–442. AAAI Press.

Condotta, J.-F.; Mensi, A.; Nouaouri, I.; Sioutis, M.; and Saïd, L. B. 2015. A Practical Approach for Maximizing Satisfiability in Qualitative Spatial and Temporal Constraint Networks. In *Proc. of ICTAI 2015*, 445–452.

Dvorak, F.; Barták, R.; Bit-Monnot, A.; Ingrand, F.; and Ghallab, M. 2014. Planning and Acting with Temporal and Hierarchical Decomposition Models. In *Proc. of ICTAI 2014*, 115–121. IEEE Computer Society.

Dylla, F.; Lee, J. H.; Mossakowski, T.; Schneider, T.; van Delden, A.; van de Ven, J.; and Wolter, D. 2017. A Survey of Qualitative Spatial and Temporal Calculi: Algebraic and Computational Properties. *ACM Computing Surveys*, 50. Article 7.

Freksa, C. 1992. Temporal Reasoning Based on Semi-Intervals. *Artificial Intelligence*, 54(1-2): 199–227.

Ghallab, M.; Nau, D. S.; and Traverso, P. 2016. *Automated Planning and Acting*. Cambridge University Press. ISBN 978-1-107-03727-4.

Lin, S.; Behnke, G.; and Bercher, P. 2023. Accelerating SAT-Based HTN Plan Verification by Exploiting Data Structures from HTN Planning. In *Proc. of ECAI 2023*, 1489–1496. IOS Press.

Maler, O.; and Nickovic, D. 2004. Monitoring Temporal Properties of Continuous Signals. In *Proc. of the International Symposium on Formal Techniques in Real-Time and Fault-Tolerant Systems (FTRTFT 2004)*, 152–166. Springer.

Mansouri, M.; and Pecora, F. 2014. More Knowledge on the Table: Planning with Space, Time and Resources for Robots. In *Proc. of ICRA 2014*, 647–654. IEEE.

Mansouri, M.; and Pecora, F. 2016. A Robot Sets a Table: A Case for Hybrid Reasoning with Different Types of Knowledge. *Journal of Experimental & Theoretical Artificial Intelligence*, 28(5): 801–821.

Pantůčková, K.; and Barták, R. 2025. What to Do If a Plan Does Not Comply with an HTN Model? In *Proc. of ECAI 2025*. IOS Press.

Pantůčková, K.; Ondrčková, S.; and Barták, R. 2024. Using Earley Parser for Verification of Totally Ordered Hierarchical Plans. In *Proc. of FLAIRS 2024*, volume 37.

Pujari, A. K.; Kumari, G. V.; and Sattar, A. 1999. INDU: An Interval and Duration Network. In *Proc. of AI 1999*, volume 1747 of *LNCS*, 291–303. Springer.

Randell, D. A.; Cui, Z.; and Cohn, A. G. 1992. A Spatial Logic Based on Regions and Connection. In *Proc. of KR 1992*, 165–176. Morgan Kaufmann Publishers Inc.

Schreiber, D. 2021. Lilotane: A Lifted SAT-based Approach to Hierarchical Planning. *Journal of Artificial Intelligence Research*, 70: 1117–1181.

Schwartz, J. T.; and Sharir, M. 1983. On the "Piano Movers" Problem. II. General Techniques for Computing Topological Properties of Real Algebraic Manifolds. *Advances in Applied Mathematics*, 4(3): 298–351.

Schwartz, T.; and Wolter, D. 2026. HTN Plan Verification by Qualitative Temporal Reasoning. In *Proc. of AAAI 2026*, volume 40, 36378–36385.

Sioutis, M. 2023. Embarrassingly Greedy Inconsistency Resolution of Qualitative Constraint Networks. In *Proc. of TIME 2023*, volume 278, 12:1–12:12.

Sioutis, M.; and Wolter, D. 2021. Qualitative Spatial and Temporal Reasoning: Current Status and Future Challenges. In *Proc. of IJCAI 2021*, 4594–4601. Survey Track.

Stock, S.; Mansouri, M.; Pecora, F.; and Hertzberg, J. 2015. Online Task Merging with a Hierarchical Hybrid Task Planner for Mobile Service Robots. In *Proc. of IROS 2015*, 6459–6464. IEEE.

Weld, D. S. 1994. An Introduction to Least Commitment Planning. *AI Magazine*, 15(4): 27–61.

Wolter, D.; and Schwartz, T. 2026. Planning and Temporal Planning by Qualitative Temporal Reasoning. In *Proc. of FoIKS 2026*, volume 16475 of *LNCS*, 366–371. Springer.