Optimal Verification of a Minimum-Weight Basis in an Uncertainty Matroid

Haya Diwan
Department of Computer Science and Engineering, New York University, NY, USA

Lisa Hellerstein
Department of Computer Science and Engineering, New York University, NY, USA

Nicole Megow
Faculty of Mathematics and Computer Science, University of Bremen, Germany

Jens Schlöter
Centrum Wiskunde & Informatica (CWI), Amsterdam, The Netherlands

Abstract

Research in explorable uncertainty addresses combinatorial optimization problems where there is partial information about the values of numeric input parameters, and exact values of these parameters can be determined by performing costly queries. The goal is to design an adaptive query strategy that minimizes the query cost incurred in computing an optimal solution. Solving such problems generally requires that we be able to solve the associated verification problem: given the answers to all queries in advance, find a minimum-cost set of queries that certifies an optimal solution to the combinatorial optimization problem. We present a polynomial-time algorithm for verifying a minimum-weight basis of a matroid, where each weight lies in a given uncertainty area. These areas may be finite sets, real intervals, or unions of open and closed intervals, strictly generalizing previous work by Erlebach and Hoffman which only handled the special case of open intervals. Our algorithm introduces new techniques to address the resulting challenges.

Verification problems are of particular importance in the area of explorable uncertainty, as the structural insights and techniques used to solve the verification problem often heavily influence work on the corresponding online problem and its stochastic variant. In our case, we use structural results from the verification problem to give a best-possible algorithm for a promise variant of the corresponding adaptive online problem. Finally, we show that our algorithms can be applied to two learning-augmented variants of the minimum-weight basis problem under explorable uncertainty.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases Matroid verification, minimum-weight basis, query strategy, uncertainty matroid, explorable uncertainty

Digital Object Identifier 10.4230/LIPIcs.STACS.2026.32

Related Version *Full Version*: https://arxiv.org/abs/2512.17116 [4]

Funding *Haya Diwan*: Partially supported by National Science Foundation (NSF) Grant no. 1909335.
Lisa Hellerstein: Partially supported by National Science Foundation (NSF) Grant no. 1909335.
Nicole Megow: Supported by the Deutsche Forschungsgemeinschaft (DFG) Grant no. 547924951.
Jens Schlöter: Supported by the research project *Optimization for and with Machine Learning (OPTIMAL)*, funded by the Dutch Research Council (NWO), Grant no. OCENW.GROOT.2019.015.

1 Introduction

Consider the problem of finding a minimum-weight basis (MWB) of a matroid, under uncertainty. We consider a weighted matroid $M = (E, \mathcal{I}, w)$, where each element $e \in E$ has a weight w_e that is known to be contained within a given range, called an *uncertainty area* A_e. The exact value of w_e can be obtained by performing a query on element e that incurs a non-negative cost of c_e. The problem is to determine an MWB while minimizing the total query cost. In the special case where M is the graphic matroid of a connected graph, so the bases of M correspond to spanning trees, this is the minimum spanning tree (MST) problem under uncertain edge weights. This was one of the earliest studied problems in explorable uncertainty, motivated by network design: spanning trees capture fundamental connectivity tasks, and uncertainty areas naturally model edge weights known only within ranges, such as costs, latencies, or capacities.

The verification version of the MWB problem under uncertainty is an offline problem that, given access to both the uncertainty areas A_e and the exact weights w_e, seeks a minimum-cost set of queries for verifying some minimum-weight basis of M. More particularly, these queries and their answers, together with the uncertainty areas for all elements $e \in E$ not queried, would be sufficient to determine an MWB of M.

To illustrate the MWB verification problem, consider the special case of a graphic matroid of a connected graph G. The elements of this matroid are the edges in G, the independent sets are the cycle-free edge sets, and an MWB is an MST. Figure 1 shows three uncertainty graphs with different types of intervals as their uncertainty areas, and their corresponding minimum-cost query sets, assuming unit query cost per edge.

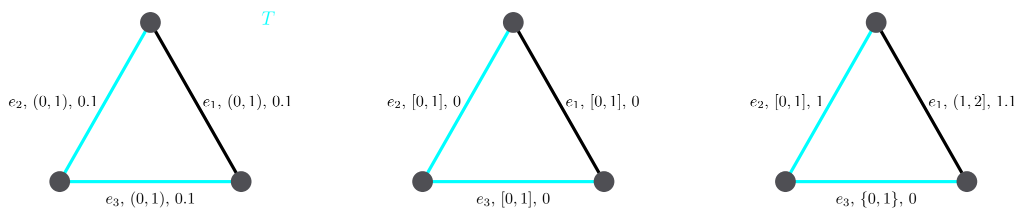

Figure 1 Let T (blue) be an MST of the given graph. The figure shows the minimum-cost query sets for instances with different types of uncertainty intervals: open $(0, 1)$, closed $[0, 1]$, or mixed. Edge labels indicate (name, interval, weight), and each query has unit cost. The corresponding minimum-cost query sets are $Q = \{e_1, e_2, e_3\}$ (open), $Q = \{e_2, e_3\}$ (closed), and $Q = \emptyset$ (mixed).

Our main result is a polynomial-time algorithm for this verification problem, producing both a minimum-cost verification set (we also say *certificate*), and an associated MWB B, for uncertainty areas that are a finite union of intervals. Each of these intervals can be open or closed. Note that this includes the case where each uncertainty area is a finite set of discrete values. For example, $\{0, 1\}$ is equivalent to $[0, 0] \cup [1, 1]$.

In addition to being an interesting combinatorial problem in its own right, the verification problem is fundamental to solving variants of the online MWB problem in settings beyond the worst-case, such as stochastic settings and adversarial learning-augmented variants. In the stochastic setting, each weight w_e is assumed to be drawn from a known probability distribution, and the goal is to minimize the *expected* query cost. The verification problem can be viewed as either an analog or a special case of such a stochastic formulation, where each distribution has support of size 1, but an element e still must be queried in order to "use" the value w_e. At the same time, our results have direct consequences for variants of the adversarial adaptive online problem: in Section 5 we use insights from the verification problem to obtain new results for a variant of the online adaptive MWB problem, and show how it provides a foundation for learning-augmented algorithms. Thus, verification serves both as a crucial first step toward stochastic formulations and as a direct tool for online problems under uncertainty.

A special case (in two senses) of our verification problem was studied before by Erlebach and Hoffman [8]. They considered finding the MST of a graph under weight uncertainty, with uncertainty areas that are either open intervals, i.e., $A_e = (L_e, U_e)$, or trivial, i.e., $A_e = \{w_e\}$ (meaning the weight of e is given). They gave an efficient algorithm that computes an optimal solution.

From a technical standpoint, handling uncertainty areas that are closed intervals and finite sets, as we do in our work, posed new challenges and required new techniques as well as structural insights. We discuss differences and challenges in more detail below.

1.1 Related Work

Verification problems. In the context of verifying optimal structures under uncertainty, the above-mentioned polynomial-time algorithm of Erlebach and Hoffman [8], for MST verification with open and trivial uncertainty areas, stands out as a notable exception in terms of computational complexity. In contrast, other verification problems are known to be NP-hard and even inapproximable. For instance, verification is NP-hard for identifying the set of maximal points under geometric uncertainty [3], and for selecting the cheapest set from a collection of sets with elements of uncertain weight [9]. The latter is even NP-hard to approximate within a factor of $o(\log m)$, where m is the number of sets [19]. Recently, verification for knapsack under explorable uncertainty was shown to be Σ_2^p-complete [23].

Online algorithms. The line of research on exploring uncertain numerical values under query costs was initiated by Kahan [14], who studied problems such as identifying the minimum or k-th smallest value from a set of uncertain values. Since then, many problems have been investigated [5], mainly in an *adaptive* online setting, where queries can be performed sequentially based on prior outcomes. The goal is to minimize the worst-case competitive ratio, comparing the total cost of an online algorithm to the minimum-cost of a verification set. Favorable constant (and often matching) upper and lower bounds are known for selection-type problems [14, 10], MST and matroid bases [13, 18, 21] and sorting [12]. However, strong lower bounds have been shown for more complex problems such as cheapest set [9], knapsack, matchings, linear programming [20], which rule out any constant competitive ratio.

The online adaptive MWB problem under uncertainty was studied by Erlebach et al. [9] for open and trivial intervals. They gave a deterministic algorithm with a best-possible competitive ratio of 2. This result extended earlier work [13] on finding the MST in a graph with the same types of uncertainty areas. Subsequently, Megow et al. gave a randomized algorithm with an improved bound of 1.707 for the online versions of both MST and MWB problems, again with open and trivial intervals [18]. In [17], Mathwieser and Çela gave improved randomized algorithms for special cases of the problem. Note that it is crucial to omit closed intervals in this online setting as otherwise no constant competitive ratio is possible [13].

In contrast to the above, *non-adaptive* online variants have received little attention, with the notable exception of the work of Merino and Soto [21]. They gave an algorithm that computes a minimum-cost "universal" set of queries that can be used to find an MWB of an uncertainty matroid, assuming no prior knowledge of any of the actual element weights. Their algorithm, like ours, allows for uncertainty areas that are finite unions of bounded real intervals that can be either open or closed. However, they require a query set to be a certificate for *every* weight vector w' such that $w'_e \in A_e$ for all elements e, while the verification problem requires that it be a certificate *only* for the given weights w_e. Because of the stronger requirements, the minimum-cost certificate for their problem can have significantly higher cost. In fact, using their algorithm for the verification problem (ignoring access to the weights w_e) only yields an n-approximation for uniform query costs and an unbounded approximation factor for general query costs (cf. discussion in full version [4]). Thus using the given weights w_e is crucial to optimally solving the verification problem.

Beyond-worst-case models and the relevance of verification methods. The strong lower bounds for more complex problems suggest that the adversarial online model may be overly pessimistic. This has motivated the study of beyond-worst-case approaches in the context of explorable uncertainty, such as stochastic models [1, 2, 19], where query outcomes are drawn from (possibly unknown) probability distributions, and learning-augmented frameworks [6, 7], where algorithms have access to imperfect predictions of the actual weights. In all of these works, a solid understanding of the corresponding verification problem was fundamental to the design and analysis of algorithms.

1.2 Our Results

We give a polynomial-time algorithm that solves the MWB verification problem, producing both a minimum-cost verification set (a certificate) and an associated MWB, for uncertainty areas that are a finite union of bounded intervals. Each interval can be either open or closed.

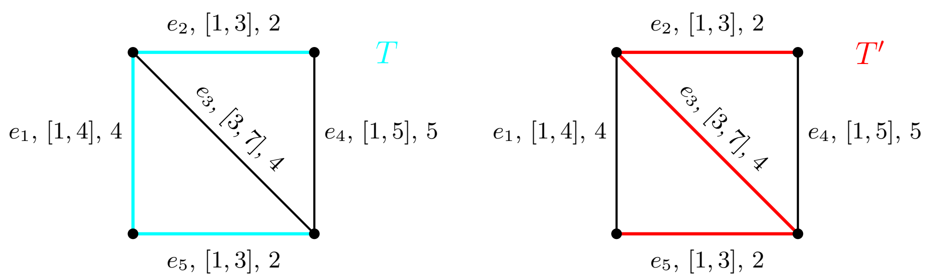

Figure 2 Let T (blue) and T' (red) be MSTs of the same uncertainty graph. Edge labels show (name, interval, weight), and all queries have unit cost. While both are MSTs, T has a cheaper minimum-cost query set, $Q = \{e_3, e_4\}$, compared to T' with $Q = \{e_1, e_3, e_4\}$.

Our algorithm has two phases, motivated by a key difference from the verification problem for open intervals: If uncertainty areas contain closed intervals, the cost of verification may vary across MWBs; see Figure 2. In contrast, for open intervals, all MWBs have the same minimum verification cost[1]. This raises a natural question for general uncertainty areas: Can we efficiently identify an MWB whose minimum verification cost is as small as possible? (We make the standard assumption that the matroid is given by an independence oracle, and that oracle queries are answered in time polynomial in the number of matroid elements.)

Our first main result answers this question in the affirmative. The first phase of our algorithm is a polynomial-time procedure that identifies an MWB which has minimum possible verification cost (Section 3). Our algorithm is based on carefully designed contraction and deletion rules that allow us to delete and contract *extreme case* elements, i.e., elements e with $w_e = \inf A_e$ or $w_e = \sup A_e$, while maintaining the invariant that there exists an MWB consistent with the deletions and contractions having minimum possible verification cost. As a corollary, we show that, for uniform query costs and uniform areas $A_e = \{L, U\}$ with $L < U$ (i.e., each w_e is either equal to L or U), all MWBs have the same minimum verification cost.

[1] This fact is only implicit in previous works [8, 7]. However, it is also a corollary of our results in Section 3.

In the second phase (Section 4), our algorithm efficiently computes a minimum-cost certificate to verify a given MWB. Together, the two phases of our algorithm yield a polynomial-time algorithm for our MWB verification problem.

Motivating the second phase, we give a full structural characterization of certificates for verifying an MWB. In contrast to previous work, we have to carefully handle extreme case elements. Intuitively, the presence or absence of such elements can make a huge difference for the certificates. For example, if we want to find an element of maximum weight in a set of elements with identical uncertainty areas, then querying a single element e with $w_e = \sup A_e$ is sufficient to identify an element of maximum weight, whereas the absence of extreme case elements forces us to query all elements. Based on our certificate characterization, we compute a minimum-cost certificate by solving a minimum-weight vertex cover problem in a bipartite auxiliary graph; this technique has also been used in previous work [8, 18, 7, 6, 1]. We remark that our results imply an alternative, arguably simpler, algorithm for the verification problem with open intervals (and other special cases where all MWBs have the same verification cost): Fix any MWB and run the algorithm's second phase.

Finally, in Section 5, we use insights from the second phase of our verification algorithm to give new results for the online adaptive MWB problem with general uncertainty areas. For open uncertainty intervals, the competitive ratio of the online adaptive MWB problem is 2 [13]. If closed uncertainty areas are allowed, the competitive ratio increases to n [11]. We show that the competitive ratio of 2 can be recovered in a promise variant of the online MWB problem with general uncertainty areas, where the algorithm is given an MWB and only has to verify this MWB. This shows that the increase in the competitive ratio from *open* to *general* uncertainty areas stems from the algorithm's task of *finding* rather than merely *verifying* an MWB B. Based on this insight, we design a learning-augmented algorithm for a setting where algorithms have access to an untrusted prediction of an MWB.

2 Preliminaries

We assume familiarity with matroids and only briefly define basic concepts and notation. For a comprehensive introduction, see Schrijver's book [24]. Readers might find it helpful to keep the graphic matroid and the MST problem on a connected graph in mind. For a set X and an element e, we use the short notation $X + e \coloneqq X \cup \{e\}$ and $X - e \coloneqq X \setminus \{e\}$.

2.1 Matroid Basics

A *matroid* is a non-empty, downward-closed set system $(E, \mathcal{I})$ with element set E and a family of subsets $\mathcal{I} \subseteq 2^E$, which satisfies the *augmentation property*: if $I, J \in \mathcal{I}$ and $|I| < |J|$, then $I + e \in \mathcal{I}$ for some $e \in J \setminus I$.

Given a matroid $M = (E, \mathcal{I})$, a set $I \subseteq E$ is called *independent* if $I \in \mathcal{I}$, and *dependent* otherwise. We refer to the elements of M by $E(M)$. An inclusion-wise maximal independent subset is called a *basis* of M. With a matroid M, we associate a rank function $r : 2^E \to \mathbb{Z}_{\geq 0}$, where $r(X)$ describes the maximal cardinality of an independent subset of X.

The *span* of some $X \subseteq E$ contains all elements that do not increase the rank when added to X, i.e., $\text{span}_M(X) = \{e \in E \mid r(X) = r(X + e)\}$. If X is a basis and $e' \notin \text{span}_M(X - e)$ for some $e \in X$, then the definition of the span implies that $X - e + e'$ is a basis.

A *circuit* C in a matroid $M = (E, \mathcal{I})$ is a minimally dependent set, that is, $C \notin I$ whereas $C \setminus \{e\} \in I$ for each $e \in C$. Any independent set $I \in \mathcal{I}$ of M and an $e \in E$ such that $I + e$ is dependent, form a unique circuit $C_e^I \subseteq I + e$. This circuit is called the *fundamental circuit* of e with respect to I. If B is a basis and $e \in C_{e'}^B$ for some $e' \notin B$, then the definition of fundamental circuits implies that $B' = B - e + e'$ is also a basis of M.

Given a matroid $M = (E, \mathcal{I})$, we obtain a matroid $M' = (E \setminus X, \mathcal{I}')$ from M by applying the operations "contraction" and "deletion" for subsets $X \subseteq E$. The *deletion* of X from M yields the matroid M' with $\mathcal{I}' := \{I \mid I \subseteq E \setminus X, I \in \mathcal{I}\}$. In this paper, we only consider deletions that do not decrease the rank of the matroid, i.e., satisfy $r(M) = r(M')$. *Contracting* an independent set X in M yields a matroid M' with independent sets $\mathcal{I}'$ defined as subsets $I \subseteq E \setminus X$ with $I \cup X \in \mathcal{I}$. A matroid M' obtained from M by a series of contractions and deletions is called a *minor of M*.

We consider *weighted* matroids $M = (E, \mathcal{I}, w)$, in which each element $e \in E$ has an associated weight $w_e \in \mathbb{R}$. The weight of a basis B of M is $w(B) := \sum_{e \in B} w_e$. We seek a minimum-weight basis (MWB) of M. Note that an MWB of M may not be unique.

2.2 Properties of Bases

Throughout the paper, we use well-known properties to argue that a basis B is an MWB. The propositions follow, e.g., from [15, Thm. 6.1], and the observation is proved in the full version [4].

▶ **Proposition 1.** *Let B be a basis of the matroid $M = (E, \mathcal{I}, w)$. If $w_e \leq w_{e'}$ holds for all $e \in B$ and all $e' \in E \setminus span_M(B - e)$, then B is an MWB of M.*

▶ **Proposition 2.** *Let B be a basis of the matroid $M = (E, \mathcal{I}, w)$. If $w_e \geq w_{e'}$ holds for all $e \in E \setminus B$ and all $e' \in C_e^B$, then B is an MWB of M.*

▶ **Observation 3.** Let B be a basis of the matroid $M = (E, \mathcal{I}, w)$. If C is a circuit of M with $e \in B \cap C$, then there is an $e' \in C - e$ such that $e' \notin \mathrm{span}_M(B - e)$.

2.3 Uncertainty Matroids and Certificates

In an uncertainty matroid, weights w_e are not known, but we are given uncertainty areas guaranteed to contain them.

▶ **Definition 4** (Uncertainty Matroid). *An uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A)$ is a matroid $M = (E, \mathcal{I})$ and a function $A : E \to \mathbb{R}$ such that for each $e \in E$, $A(e)$ is a non-empty finite union of bounded real intervals (either open or closed). We call $\mathcal{M} = (E, \mathcal{I}, A, w)$ a* weighted uncertainty matroid *if $\mathcal{M} = (E, \mathcal{I}, A)$ is an uncertainty matroid and $w : E \to \mathbb{R}$ is such that $w_e := w(e) \in A_e$ for all $e \in E$. We define $A_e := A(e)$, $L_e := \inf A_e$, and $U(e) := \sup A_e$. We call A_e the uncertainty area of e and say e is* trivial *if A_e consists of the single value w_e.*

Consider the problem of computing an MWB in a weighted uncertainty matroid, where the exact weights are initially unknown but the true weight of an element e can be revealed through a *query*, which incurs a cost $c_e \geq 0$. We are now faced with the problem of determining an MWB at the lowest possible total query cost. We use the notion of *certificates*, where a certificate is a set $Q \subseteq E$ such that querying Q reveals sufficient information about the uncertain weights in A to identify a basis B that is an MWB, irrespective of the exact weights of the elements not in Q. We make this precise in the definitions below. Note that the formal definition of an uncertainty matroid only specifies uncertainty areas for the matroid elements e, not specific weights w_e. In what follows, we sometimes identify a weighted uncertainty matroid $\mathcal{M}(E, \mathcal{I}, A, w)$ with its associated weighted matroid $M = (E, I, w)$. For example, when we refer to a basis B of $\mathcal{M}(E, \mathcal{I}, A, w)$, we mean a basis of $M = (E, I, w)$.

▶ **Definition 5** (Consistent weight assignment). *Consider a weighted uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A, w)$. A* weight assignment *$w^*$ for $\mathcal{M}$ is a function $w^* \colon E \to \mathbb{R}$ such that $w_e^* \in A_e$ holds for each $e \in E$. We say a weight assignment w^* for $\mathcal{M}$ is* consistent *with Q if it additionally satisfies $w_e = w_e^*$ for each $e \in Q$.*

▶ **Definition 6** (Verification and certificates). *Consider a weighted uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A, w)$. Let B be an MWB of $\mathcal{M}$. We say that a certificate Q* verifies *B if for every weight assignment w^* for $\mathcal{M}$ that is consistent with Q (Definition 5), B is an MWB of the weighted matroid $M' = (E, \mathcal{I}, w^*)$.*

▶ **Definition 7** (Minimum-cost certificate). *Consider a weighted uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A, w)$. Let $\mathcal{Q}$ be the set of all certificates that verify a basis of $\mathcal{M}$. Then, $Q \in \mathcal{Q}$ is a* minimum-cost certificate *for $\mathcal{M}$, if $c(Q) := \sum_{e \in Q} c_e$ is minimal among all certificates in $\mathcal{Q}$.*

2.3.1 Characterization of Certificates

To further characterize certificates, define $L_e(Q) = w_e$ if $e \in Q$ and $L_e(Q) = L_e$ if $e \notin Q$. Similarly, define $U_e(Q) = w_e$ if $e \in Q$ and $U_e(Q) = U_e$ if $e \notin Q$. Intuitively, $L_e(Q)$ and $U_e(Q)$ denote the upper and lower limits of e *after* querying the certificate Q.

▶ **Lemma 8.** *Let B be an MWB of $\mathcal{M} = (E, \mathcal{I}, A, w)$. A set $Q \subseteq E$ is a certificate that verifies B if and only if $U_e(Q) \leq L_f(Q)$ for all $e \in B$ and $f \in (E \setminus span_M(B - e)) - e$.*

Proof. First, assume $U_e(Q) \leq L_f(Q)$ for all $e \in B$ and $f \in (E \setminus \mathrm{span}_M(B - e)) - e$. Consider any weight assignment w^* consistent with Q. Then, for each $e \in B$, we have $w^*_e \leq U_e(Q) \leq L_f(Q) \leq w^*_f$ for all $f \in (E \setminus \mathrm{span}_M(B - e)) - e$. This implies B is an MWB for w^*, by Proposition 1. Thus Q verifies B by Definition 6.

Next, assume it does not hold that $U_e(Q) \leq L_f(Q)$ for all $e \in B$ and $f \in (E \setminus \mathrm{span}_M(B - e)) - e$. Let $e \in B$ and $f \in (E \setminus \mathrm{span}_M(B - e)) - e$ be such that $L_f(Q) < U_e(Q)$. Then, there exists a weight assignment w^*, consistent with Q, such that $w^*_e > w^*_f$. Thus $B' = B - e + f$ is independent and satisfies $w^*(B') < w^*(B)$, and so Q does not verify B by Definition 6. ◀

The lemma implies the following alternative characterization, which we prove in the full version [4].

▶ **Corollary 9.** Let B be an MWB of $\mathcal{M} = (E, \mathcal{I}, A, w)$. A set $Q \subseteq E$ is a certificate that verifies B if and only if the following property holds for every $f \notin B$, and the fundamental circuit C^B_f: for every $e \in C_f - f$, $U_e(Q) \leq L_f(Q)$.

2.3.2 Exchange Properties of Certificates

We continue by showing some exchange properties of certificates.

▶ **Lemma 10.** *Let B be an MWB of $\mathcal{M} = (E, \mathcal{I}, A, w)$ and let Q be a certificate that verifies B. Let $e \in B$ and $e' \notin B$ with $U_e = U_{e'} = w_e = w_{e'}$ be such that $e \in C_{e'}$ for the fundamental circuit $C_{e'}$ of e' with respect to B. Then $Q' = Q - e' + e$ is a certificate for $B' = B - e + e'$.*

Proof. During this proof, we use C'_f to refer to the fundamental circuit of f with respect to B' for an $f \notin B'$. To show that Q' is a certificate for B', we show for each $f \notin B'$ that $L_f(Q') \geq U_{f'}(Q')$ for each $f' \in C'_f - f$. By Corollary 9, this implies Q' is a certificate for B'.

We distinguish between the two cases (1) $f = e$ and (2) $f \neq e$.

Case (1): Assume $f = e$ and consider circuit C'_e. Since $B' = B - e + e'$, we have $C'_e = C_{e'}$. Since Q is a certificate for B with $e' \notin B$, we have $w_{e'} \geq L_{e'}(Q) \geq U_{f'}(Q)$ for all $f' \in C_{e'} - e'$ by Corollary 9. For every $f' \in C_{e'} - \{e, e'\}$, the membership of f and f' in Q and Q' is identical, so $U_{f'}(Q') = U_{f'}(Q)$. Since $C_{e'} = C'_e$, $e \in Q'$ and by definition $w_e = w_{e'} = U_{e'}$, this implies $L_e(Q') = w_e = w_{e'} \geq L_{e'}(Q) \geq U_{f'}(Q) = U_{f'}(Q')$ for all $f' \in C'_e - \{e, e'\}$. Further, $L_e(Q') = w_{e'} = U_{e'} \geq U_{e'}(Q')$.

Case (2): Assume $f \neq e$ and consider the circuit C'_f. We distinguish two subcases.

- If $(C'_f - f) \subseteq B$, then $C'_f = C_f$ and $e, e' \notin C'_f$. In this case, $L_f(Q') = L_f(Q) \geq U_{f'}(Q) = U_{f'}(Q')$ holds for all $f' \in C'_f - f$ by definition of Q' and by our assumption that Q is a certificate for B with $f \notin B$ (cf. Corollary 9).
- If $(C'_f - f) \not\subseteq B$, then we must have $e' \in C'_f - f$. Consequently, $f \in E \setminus \mathrm{span}_M(B' - e') = E \setminus \mathrm{span}_M(B - e)$. Note that $w_{e'} = U_{e'}(Q') = U_e(Q)$ holds by definition because of $w_e = w_{e'} = U_e = U_{e'}$. Since Q is a certificate for B with $e \in B$, we must have $w_{e'} = U_{e'}(Q') = U_e(Q) \leq L_f(Q) = L_f(Q')$ by Lemma 8.
  If $C'_f = \{f, e'\}$, then the above argument already shows $L_f(Q') \geq U_{f'}(Q')$ for each $f' \in C'_f - f$, and we are done with the proof for this case. Thus, assume $C'_f \setminus \{f, e'\} \neq \emptyset$ and consider an arbitrary $f' \in C'_f - f - e'$. Then $f' \in B \cap C'_f$ and, by Observation 3, there must be an element $g \in C'_f - f'$ with $g \notin \mathrm{span}_M(B - f')$. The only two elements of $C'_f - f'$ that are not in $B - f' \subseteq \mathrm{span}_M(B - f')$ are e' and f.
  If $f \notin \mathrm{span}_M(B - f')$, then our assumption that Q is a certificate for B implies $U_{f'}(Q') = U_{f'}(Q) \leq L_f(Q) = L_f(Q')$. If $e' \notin \mathrm{span}_M(B - f')$, then our assumption that Q is a certificate for B implies $U_{f'}(Q') = U_{f'}(Q) \leq L_{e'}(Q) \leq w_{e'} = U_e(Q) \leq L_f(Q')$, where the last inequality uses $U_e(Q) \leq L_f(Q')$ as argued above. In all cases, $U_{f'}(Q') \leq L_f(Q')$. ◀

The next lemma is a dual version of Lemma 10. The proof is in the full version [4].

▶ **Lemma 11.** *Let B be an MWB of $\mathcal{M} = (E, \mathcal{I}, A, w)$ and let Q be a certificate that verifies B. Let $e \notin B$ and $e' \in B$ with $L_e = L_{e'} = w_e = w_{e'}$ be such that $e' \in C_e$ for the fundamental circuit C_e of e with respect to B. Then $Q' = Q - e' + e$ is a certificate for $B' = B + e - e'$.*

It is easy to show the following exchange property for trivial elements.

▶ **Lemma 12.** *Let B be an MWB of $\mathcal{M} = (E, \mathcal{I}, A, w)$ and let Q be a certificate that verifies B. Let $e \in B$ and $e' \notin B$ with $w_e = w_{e'}$ be such that $B' = B - e + e'$ is independent. If each $f \in \{e, e'\}$ is either trivial or contained in Q, then Q also verifies B'.*

3 Computing an MWB that has a Minimum-Cost Certificate

Given a weighted uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A, w)$, we compute an MWB B that admits a minimum-cost certificate, though we do not yet compute the certificate itself. To this end, we introduce a set of contraction and deletion rules. Starting from $B = \emptyset$, these rules allow us to safely add elements to B (contraction) and ban elements from ever entering B (deletion) while maintaining the invariant that there is a minimum-cost certificate which verifies a basis B' that contains B and does not contain any deleted elements. We formalize this invariant by introducing the notion of a "compatible minor" which intuitively is the remaining (uncertainty) matroid after applying contractions and deletions. We refer to sets of contracted and deleted elements as K and D, respectively.

▶ **Definition 13.** *Let $\mathcal{M} = (E, \mathcal{I}, A, w)$ be a weighted uncertainty matroid with a minimum-cost certificate of cost c^* and let $D, K \subseteq E$. Let $M[D, K]$ be the matroid obtained from $M = (E, \mathcal{I})$ by deleting D and contracting K. We say $M[D, K]$ is a* compatible minor *of $\mathcal{M}$ if there is a query set for $\mathcal{M}$ with cost c^* that verifies an MWB B of $\mathcal{M}$ with $K \subseteq B$ and $D \cap B = \emptyset$. Furthermore, we say that a certificate Q for $\mathcal{M}$ is* compatible with *$M[D, K]$ if Q verifies a basis B of $\mathcal{M}$ with $K \subseteq B$ and $D \cap B = \emptyset$.*

Our goal is to iteratively create sets D and K until K becomes an MWB while maintaining the invariant that there is a minimum-cost certificate compatible with D and K. To this end, we first introduce a series of observations and lemmas that, given a compatible minor

$M[D, K]$, allow us to compute a *larger* compatible minor $M[D', K']$ (Section 3.1). Formally, larger means $D' \supseteq D$, $K' \supseteq K$ and $D' \supset D$ or $K' \supset K$. We also refer to these observations and lemmas as *contraction and deletion rules*. We use the phrase *applying a rule* to refer to the process of computing the larger compatible minor $M[D', K']$ and replacing the given one, i.e. we set $K := K'$ and $D := D'$. We will show that exhaustively applying these rules in a certain order until K is a basis and $D = E \setminus K$ yields an MWB that can be verified with minimum-cost. Afterwards, we show that the rules can be implemented with a polynomial-time algorithm (Section 3.2), which will imply the following theorem.

▶ **Theorem 14.** *There is a polynomial-time algorithm that computes, for any given weighted uncertainty matroid $\mathcal{M}$, an MWB B that can be verified by a minimum-cost certificate.*

3.1 Contraction and Deletion Rules

Consider a weighted uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A, w)$. Elements that appear in every MWB of $\mathcal{M}$ or in none can be safely contracted (added to K) or deleted (added to D). The next observations follow from standard arguments (proofs are in the full version [4]).

▶ **Observation 15.** Let $M' := M[D, K]$ be a compatible minor of $\mathcal{M}$. If there is a circuit C in $M[D, K]$ with an element $e \in C$ that has the unique maximum weight in C, then $M[D + e, K]$ is a compatible minor of $\mathcal{M}$.

▶ **Observation 16.** Let $M' := M[D, K]$ be a compatible minor of $\mathcal{M}$. If there is a basis B of M' with an element $e \in B$ that has unique minimum weight in $E(M') \setminus \mathrm{span}_{M'}(B - e)$, then $M[D, K + e]$ is a compatible minor of $\mathcal{M}$.

We continue by giving contraction and deletion rules for *extreme case elements* e satisfying $w_e = U_e$ or $w_e = L_e$. The existence of these elements separates our work from [8].

3.1.1 Rules for Non-Trivial Extreme Case Elements

We start by considering non-trivial extreme case elements. For each $w \in \{w_e \mid e \in E\}$, we define $E^L_w = \{e \in E \mid L_e = w \wedge e \text{ is non-trivial}\}$ and $E^U_w = \{e \in E \mid U_e = w \wedge e \text{ is non-trivial}\}$.

▶ **Lemma 17.** *Let $M' := M[D, K]$ be a compatible minor of $\mathcal{M}$. Let e be a* non-trivial *element such that (i) $e \in B$ for some MWB B of M', (ii) $w_e = U_e$, (iii) e has minimum weight in $E(M') \setminus span_{M'}(B - e)$ and (iv) e has maximum query cost in $(E(M') \setminus span_{M'}(B - e)) \cap E^U_{w_e}$. Then $M[D, K + e]$ is a compatible minor of $\mathcal{M}$.*

Proof. Let Q^* denote a minimum-cost certificate for $\mathcal{M}$ that is compatible with M'. Then, this certificate must also verify some MWB B' for M'.

To prove the lemma, we have to show that there exists a minimum-cost certificate that verifies an MWB B'' of $\mathcal{M}$ with $K + e \subseteq B''$ and $D \cap B'' = \emptyset$. If $e \in B'$, then this directly follows for Q^* and the MWB $B'' = K \cup B'$ of $\mathcal{M}$. Thus, assume $e \notin B'$ and let C denote the fundamental circuit of e with respect to B'. By Observation 3, there exists an $e' \in C - e$ such that $e' \notin \mathrm{span}_{M'}(B - e)$. Also, $e' \in B' - e$ because $C' - e \subseteq B'$. By assumption that e has minimum weight in $E(M') \setminus \mathrm{span}_{M'}(B - e)$ and that B' is an MWB of M', we have $U_e = w_e = w_{e'}$. We distinguish the following cases:

1. If $e, e' \in Q^*$, then Q^* also verifies that $B'' = K \cup (B' - e' + e)$ is an MWB by Lemma 12. Note that $(B' - e' + e)$ is an MWB for M' and B'' is an MWB for $\mathcal{M}$.

2. If $e \in Q^*$ and $e' \notin Q^*$, then we must have $w_e \geq U_{e'}$. Otherwise, Q^* would not verify that e has maximum weight in C. Since $U_{e'} \geq w_{e'} = w_e = U_e$, we get $U_e = U_{e'}$. By Lemma 10, this implies that $Q = (Q^* - e + e')$ verifies the MWB $B'' = K \cup (B' - e' + e)$. Note that e' can be trivial, in which case we can just use $Q = (Q^* - e)$ instead.
   If e' is trivial, then we clearly have $c(Q) \leq c(Q^*)$. Otherwise, i.e., if e' is non-trivial, we have $e' \in E^U_{w_e}$ as we already argued that $U_{e'} = w_{e'} = w_e$. By assumption (iv) of the lemma, this implies $c_e \geq c_{e'}$ and, thus, $c(Q) \leq c(Q^*)$.
3. If $e \notin Q^*$, then we must have $w_{e'} \leq L_e$. Otherwise Q^* cannot verify that e has maximum weight in C. However, this implies $w_{e'} = U_e \leq L_e$, which can only be the case if e is trivial; a contradiction to the requirements of the lemma. ◀

The next lemma is a dual of Lemma 17 and can be shown analogously (cf. the full version [4]).

▶ **Lemma 18.** *Let $M' := M[D, K]$ be a compatible minor of $\mathcal{M}$. Let e be a* non-trivial *element such that (i) there exists a circuit C in $M[D, K]$ with $e \in C$, (ii) $w_e = L_e$, (iii) e has maximum weight in C and (iv) e has maximum query cost in $C \cap E^L_{w_e}$. Then $M[D + e, K]$ is a compatible minor of $\mathcal{M}$.*

Iteratively and exhaustively applying the deletion rules in Lemma 18 and Observation 15, and the contraction rules in Lemma 17 and Observation 16, yields D and K such that $M[D, K]$ satisfies the following assumption, enabling further rules depending on this assumption.

▶ **Assumption 19.** We may assume that a compatible minor $M' := M[D, K]$ of $\mathcal{M}$ does not contain non-trivial elements e with any of the following properties:
- **(i)** $w_e = L_e$ and e is a maximum weight element in a circuit C of M'.
- **(ii)** $w_e = U_e$ and e is a minimum weight element in $E(M') \setminus \text{span}_{M'}(B - e)$ for a basis B of M' that contains e.

3.1.2 Rules for Trivial Extreme Case Elements

We continue by introducing deletion and contraction rules for trivial extreme case elements in instances that satisfy Assumption 19. We defer the following two proofs to the full version [4], but remark that they follow a similar proof strategy as the proof of Lemma 17.

▶ **Lemma 20.** *Let $M' := M[D, K]$ be a compatible minor of $\mathcal{M}$ and assume that Assumption 19 is satisfied. Let e be a* trivial *element such that (i) there exists basis B of M' with $e \in B$ and (ii) e has minimum weight among elements not in span$_{M'}(B - e)$. Then $M[D, K + e]$ is a compatible minor of $\mathcal{M}$.*

▶ **Lemma 21.** *Let $M[D, K]$ be a compatible minor of $\mathcal{M}$ and assume that Assumption 19 is satisfied. Let e be a* trivial *element such that (i) there exists a circuit C in $M[D, K]$ with $e \in C$ and (ii) e has maximum weight in C. Then $M[D + e, K]$ is a compatible minor of $\mathcal{M}$.*

Given a compatible minor $M[D, K]$ that satisfies Assumption 19, we check for an element e that meets the requirements of Lemma 20 or Lemma 21. If such an element exists, we apply the corresponding rule to extend D or K, while maintaining our invariant that $M[D, K]$ is a compatible minor. We then return to applying the rules of the previous sections to restore Assumption 19, and repeat this argument until no such element e remains. The resulting compatible minor $M[D, K]$ satisfies the following properties:

1. $M[D,K]$ does not contain elements e with $w_e = L_e$ such that e is a maximum-weight element in some circuit of $M[D,K]$.
2. $M[D,K] =: M'$ does not contain elements e with $w_e = U_e$ such that there is a basis B of M' with $e \in B$ and e has minimum weight among elements not in $\text{span}_{M'}(B-e)$.
3. There is a minimum-cost certificate that is compatible with D and K.

In any basis B of M' with these properties, no element $e \in B$ with $w_e = U_e$ has minimum-weight in $E(M')\setminus\text{span}_{M'}(B-e)$. Hence, such elements $e \in B$ with $w_e = U_e$ cannot be part of any MWB B of M' as we can swap e with an element of $E(M')\setminus\text{span}_{M'}(B-e)$ that has strictly smaller weight. Thus, we can delete all these elements $e \in B$ with $w_e = U_e$ and add them to D while maintaining the property that there is a minimum-cost certificate that is compatible with D and K. Note that this corresponds to applying Observation 15. Similarly, we can argue that all remaining elements e with $w_e = L_e$ can be contracted by applying Observation 16. This way we compute sets D and K that satisfy the following assumption.

▶ **Assumption 22.** We may assume that a compatible minor $M[D,K]$ of a weighted uncertainty matroid $\mathcal{M}$ does not contain elements e with $w_e = U_e$ or $w_e = L_e$.

The rules for extreme case elements imply the following (cf. the full version [4] for a proof).

▶ **Corollary 23.** Let $M[D,K]$ be a compatible minor of $\mathcal{M}$ with $A_e = \{L_e, U_e\}$ for all $e \in E(M)$ that satisfies Assumption 19, then K is an MWB that can be verified with a minimum-cost certificate. If in addition the query costs are uniform and all A_e are uniform and non-trivial, then every MWB B of M has the same verification cost.

3.1.3 Rules for Instances without Extreme Case Elements

Instances $M' := M[D,K]$ that satisfy Assumption 22 essentially behave like instances with open intervals, since all elements satisfy $w_e \in (L_e, U_e)$ (although we still might have $A_e \neq (L_e, U_e)$). For the special case of graphic matroids with open intervals, the results in [7, 8] imply that *all* MWB can be verified with minimum-cost certificates. In the full version [4], we generalize these insights to arbitrary matroids and prove the following lemma.

▶ **Lemma 24.** *Let $M' := M[D,K]$ be a compatible minor of $\mathcal{M}$ such that Assumption 22 is satisfied, and let B be any MWB of M'. Then, $M[D',K']$ with $K' = K \cup B$ and $D' = D \cup (E(M') \setminus B)$ is a compatible minor of $\mathcal{M}$.*

Once we create a compatible minor $M' := M[D,K]$ that satisfies Assumption 22 by applying the contraction and deletion rules of the previous sections, Lemma 24 allows us to just fix any MWB B of M', contract (delete) the elements of B (of $E(M') \setminus B$) and add them to K (to D). Afterwards, K is a basis that can be verified with a minimum-cost certificate.

3.2 A Polynomial-time Algorithm

The results of Section 3.1 imply a sequence of contraction and deletion rules applied to an uncertainty matroid, leading to a compatible minor that has a unique MWB with a minimum-cost certificate. Algorithm 1 exhaustively applies these rules in the right order to compute this MWB, and runs in polynomial time, as we show with the following lemma. This proves Theorem 14.

▶ **Lemma 25.** *Algorithm 1 runs in polynomial time.*

Algorithm 1 Computing an MWB B that can be verified with a minimum-cost certificate.

Input: A weighted uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A, w)$
$K, D \leftarrow \emptyset$
while $K \cup D \neq E$ **do**
    $M' \leftarrow M[D, K]$;
    Select the element $e \in E(M')$ that satisfies the first condition listed, in the following order, and add to D or K as specified:
    if e satisfies Observation 15 **then** add e to D;
    if e satisfies Observation 16 **then** add e to K;
    if e satisfies Lemma 17 **then** add e to K;
    if e satisfies Lemma 18 **then** add e to D;
    if e satisfies Lemma 20 **then** add e to K;
    if e satisfies Lemma 21 **then** add e to D;
end while
Compute an MWB B of $M' = M[D, K]$. Add B to K and $E(M') \setminus B$ to D;
return K;

Proof. The values of L_e and U_e can be easily computed at the beginning for each $e \in E$, using the list of intervals for each A_e. Subsequently, no further information is needed about A_e.

Since the number of iterations of the while-loop is bounded by the number of elements, it only remains to argue that each iteration can be executed in polynomial time. To this end, we mainly have to argue that, for an element e, we can check in polynomial time whether the prerequisites of Observation 15 and 16 or Lemmas 17 to 21 are satisfied.

We separately argue about the different lemmas and observations. Assume there is given a compatible minor $M' := M[D, K]$ of an uncertainty matroid $\mathcal{M}$.

(a) Observation 16: Checking whether an element e satisfies the condition of Observation 16 is equivalent to checking whether e is part of every MWB of M'. To check whether an element e is part of every MWB w.r.t. the weights w, we can just compute some minimum-weight basis B using the standard greedy algorithm. If $e \notin B$, then Observation 16 clearly does not apply. Otherwise, i.e., $e \in B$, we can check whether e can be exchanged with an element $e' \in E \setminus B$ with $w_{e'} \leq w_e$. If this is possible, then e is not part of every minimum-weight basis. If not, then e is part of every minimum-weight basis. The running time of this test is dominated by the running time of the greedy algorithm and, thus, polynomial in the input size.

(b) Observation 15: Using matroid duality arguments allows us to argue in the same way as in the previous case.

(c) Lemma 17: Checking whether there exists an element e that is *non-trivial* and (i) there exists a basis B of M' with $e \in B$, (ii) $w_e = U_e$, (iii) e has minimum weight in $E(M') \setminus \mathrm{span}_{M'}(B - e)$ and (iv) e has maximum query cost in $E(M') \setminus \mathrm{span}_{M'}(B - e) \cap E^U_{w_e}$, can also be done in polynomial time.
Note that a non-trivial element e that satisfies (i)-(iii) is a non-trivial element e with $w_e = U_e$ that is in at least one MWB B. To decide whether a non-trivial element e with $w_e = U_e$ is part of at least one MWB B, we can just compute some MWB B' and check whether $e \in B'$ or, if not, whether e can swapped into B' by exchanging it with an equal-weight element in the fundamental circuit of B' w.r.t. e. Note that this check can be performed in polynomial time, as the running time is dominated by the running time for computing the MWB B'. If the element e is not in B' and cannot be swapped into B', then e is not part of any MWB and, therefore, does not satisfy the properties (i)-(iii). Otherwise, it satisfies the properties (i)-(iii).

Executing this check for each non-trivial element e with $w_e = U_e$, we can either find such an element e that satisfies (i)-(iii) or determine that Lemma 17 cannot be applied. In the latter case, we are done. If we find an element e with $w_e = U_e$ that satisfies (i)-(iii), then the check procedure described above also gives us an MWB B with $e \in B$ (this is either B' or the MWB that is created by swapping e into B'). Given such a pair of e and MWB $B \ni e$, we can easily check whether e has maximum query cost in $(E(M') \setminus \text{span}_{M'}(B - e)) \cap E^U_{w_e}$, i.e., satisfies (iv). If this is the case, then we are done. If not, then some other element of $(E(M') \setminus \text{span}_{M'}(B - e)) \cap E^U_{w_e}$ satisfies the condition of Lemma 17 and we are done anyway.

(d) Lemma 18: Using matroid duality arguments allows us to argue in the same way as in the previous case.

(e) The conditions of Lemma 20 and 21 can be checked similarly to Observation 15 and 16 plus checking Assumption 19, which is trivial.

(f) The final step of the algorithm consists of computing an MWB of the minor $M[D, K]$ which can be done by the greedy algorithm in polynomial time. ◀

4 Computing a Minimum-Cost Certificate to Verify a Given Basis

In this section, we provide an algorithm for computing a minimum-cost certificate that verifies a given MWB. In particular, the algorithm can be applied to the MWB produced by Algorithm 1, thereby yielding a minimum-cost certificate for $\mathcal{M}$.

Lemma 26 is a key result that dictates specifically what elements must be in Q in order for it to adhere to Corollary 9.

▶ **Lemma 26.** *Let B be an MWB of a weighted uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A, w)$. For a non-basis element e of B, let C_e be the fundamental circuit of e with respect to B, let F_e be the set of elements $f \in C_e - e$ with $U_f > L_e$, and let $\hat{F}_e$ be the set of elements $f \in C_e - e$ with $U_f > w_e$. A subset $Q \subseteq E$ is a certificate of B if and only if for each non-basis element e of B, the following holds:*

1. *If $w_e \geq U_f$ for all $f \in C_e - e$, and there exists $f' \in C_e - e$ such that $w_{f'} > L_e$, then $e \in Q$.*
2. *If $w_e \geq U_f$ for all $f \in C_e - e$, and $w_f \leq L_e$ for all $f \in C_e - e$, then $e \in Q$ or $F_e \subseteq Q$.*
3. *If there exists $f \in C_e - e$ such that $w_e < U_f$, and there exists $f' \in C_e - e$ (could have $f' = f$) such that $w_{f'} > L_e$, then $\hat{F}_e + e \subseteq Q$.*
4. *If there exists $f \in C_e - e$ such that $w_e < U_f$, and for all $f' \in C_e - e$ it holds that $w_{f'} \leq L_e$, then $F_e \subseteq Q$ or $\hat{F}_e + e \subseteq Q$.*

Proof. Let Q be a certificate that verifies B. Consider a non-basis element e, and the fundamental circuit C_e with respect to B. Let w^* be some weight assignment consistent with Q (Definition 5). Given w^*, if any element $f \in C_e - \{e\}$ satisfies $U_f \leq L_e$, it immediately follows that $w^*_e \geq w^*_f$, without including either e or f in Q. Define F_e as the set of elements $f \in C_e - \{e\}$ where $U_f > L_e$. Additionally, define $\hat{F}_e$ as the set of elements $f \in C_e - \{e\}$ with $U_f > w_e$. Clearly, $\hat{F}_e \subseteq F$.

Recall that if $e \in Q$, then $w_e = w^*_e$, and if $e \notin Q$, $w^*_e \in A_e$. We aim to establish the necessary and sufficient conditions for Q to satisfy Corollary 9 for all elements $f \in F$, (i.e., for Q to satisfy $L_e(Q) \geq U_f(Q)$ for all $f \in F$). Observe the following cases:

Case 1. The first case is if $w_e \geq U_f$ for all $f \in F_e$. Two sub-cases arise in this scenario.

- **Subcase 1.1:** There exists some $f' \in F$ where $w_{f'} > L_e$. In this case, we claim it is necessary and sufficient to have $e \in Q$. Suppose $e \notin Q$. Then by Definition 5, we have $w_e^* \in A_e$. Since $w_{f'} > L_e$, Corollary 9 is not satisfied. Thus, e must be in Q. Conversely, if $e \in Q$, we have that for all $f \in F_e$, $w_e^* = w_e = L_e(Q) \geq U_f \geq U_f(Q) \geq w_f$ and thus $w_e^* \geq w_f$ and Corollary 9 is satisfied.
- **Subcase 1.2:** For all $f \in F_e$, $w_f \leq L_e$. We show that in this case, it is necessary and sufficient to have $e \in Q$ or $F_e \subseteq Q$. Suppose $e \notin Q$ and $F_e \not\subseteq Q$. Then there exists some element $f' \in F_e \backslash Q$. By Definition 5, we have $w_e^* \in A_e$ and $w_{f'}^* \in [L_{f'}, U_{f'}]$. Since $U_{f'} > L_e$, Corollary 9 is not satisfied. Thus, it's clear that $e \in Q$ or $F_e \subseteq Q$. Conversely, suppose $e \in Q$ or $F_e \subseteq Q$. If $e \in Q$ we have that $w_e^* = w_e = L_e(Q) \geq U_f \geq U_f(Q) \geq w_f$ and so $w_e^* \geq w_f$. If $F_e \subseteq Q$, we have that for all $f \in F_e$, $w_e \geq L_e(Q) \geq L_e \geq U_f(Q) = w_f = w_f^*$ and thus $w_e \geq w_f^*$. Either way, Corollary 9 is satisfied.

Case 2. The second case is if $w_e < U_f$ for some $f \in F_e$. Again two sub-cases arise in this scenario.

- **Subcase 2.1:** There exists some $f' \in C_e - e$ where $w_{f'} > L_e$. In this case, we show that it is necessary and sufficient to have $\hat{F}_e + e \subseteq Q$. Suppose $\hat{F}_e + e \not\subseteq Q$. Then either $e \notin Q$ or there exists some $f'' \in \hat{F}_e \backslash Q$. Suppose $e \notin Q$. By Definition 5, we have $w_e^* \in A_e$. Since there exists some $w_{f'} > L_e$, Corollary 9 is not satisfied. Suppose $f'' \notin Q$. By Definition 5, we have $w_{f''}^* \in [L_{f''}, U_{f''}]$. Since $w_e < U_{f''}$, Corollary 9 is not satisfied. Thus, $\hat{F}_e + e \subseteq Q$.

  Conversely, if $\hat{F}_e + e \subseteq Q$ we have for all $f \in \hat{F}_e$ that $w_e^* = w_e = L_e(Q) \geq U_f \geq U_f(Q) = w_f = w_f^*$. For all $f \in F_e \setminus \hat{F}_e$, given that $L_e < U_f \leq w_e$, the inclusion of $e \in Q$ ensures that $w_e^* = w_e = L_e(Q) \geq U_f \geq U_f(Q) \geq w_f$. Therefore, Corollary 9 is satisfied.
- **Subcase 2.2:** For all $f' \in C_e - \{e\}$, $w_{f'} \leq L_e$. In this case, we show it is necessary and sufficient to have $F_e \subseteq Q$ or $\hat{F}_e + e \subseteq Q$. Suppose neither F_e nor $\hat{F}_e + e$ is contained in Q. Then there exists some $f'' \in \hat{F}_e \backslash Q$. By Definition 5, we have that $w_{f''}^* \in [L_{f''}, U_{f''}]$. Since $w_e < U_{f''}$, Corollary 9 is not satisfied. Thus, it's clear that all such f'' must be in Q. Now suppose some $f'' \in F_e \backslash \hat{F}_e$ is not in Q. This element has the property that $w_e \geq U_{f''} > L_e \geq w_{f''}$. By Definition 5 we have that $w_{f''}^* \in [L_{f''}, U_{f''}]$. Since $L_e < U_{f''}$, Corollary 9 is not satisfied. However, having e or all such $f'' \in Q$ will show that $w_e^* > w_{f''}^*$.

  By ensuring that $F_e \subseteq Q$ or $\hat{F}_e + e \subseteq Q$ we have that either $w_e \geq L_e(Q) \geq L_e \geq U_f(Q) = w_f = w_f^*$ and so $w_e \geq w_f^*$ for all $f \in F$ or $w_e^* = w_e = L_e(Q) \geq U_f \geq U_f(Q) = w_f = w_f^*$ so $w_e^* \geq w_f^*$ for all $f'' \in \hat{F}_e$. For all $f'' \in F_e \setminus \hat{F}_e$, given that $L_e < U_f \leq w_e$, the inclusion of $e \in Q$ ensures that $w_e^* = w_e = L_e(Q) \geq U_f \geq U_f(Q) \geq w_f$. Therefore, Corollary 9 is satisfied. ◀

In all cases, the construction of Q verifies that each element $e \notin B$ is a maximum-weight element on the unique circuit it forms in B, which completes the proof by Corollary 9. Intuitively, Lemma 26 exhaustively formulates the "choices" that a certificate Q has to make to ensure that Corollary 9 is satisfied for a fundamental cycle C_e with respect to B of an $e \notin B$. For example, if C_e falls into the second case of the lemma, then we must have $F_e \subseteq Q$ or $e \in Q$ (or both). The "choices" for different $e, e' \notin B$ however, are not necessarily disjoint, e.g., if we have $F_e \cap F_{e'} \neq \emptyset$. Similar to verification algorithms in the literature, we define an auxiliary graph G^B such that every vertex cover of G^B corresponds to a combination of "choices" of Lemma 26, and vice versa. The following definition makes this more precise.

Algorithm 2 Identifying a Minimum-Cost Certificate of B.

Input: $\mathcal{M} = (E, \mathcal{I}, A, w)$ is a weighted uncertainty matroid.
Let B be an MWB of $\mathcal{M}$ as computed by Algorithm 1
Let $G^B = (V^B, E^B)$ be the auxiliary graph as defined in Definition 27
Let S^* be a minimum-weight vertex cover of G^B for the weights $w(e) = c_e$, $e \in V^B = E$.
return S^* and B

▶ **Definition 27.** *Given a weighted uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A, w)$ and an MWB B, we define the bipartite (with the exception of self-loops) auxiliary graph $G^B = (V^B, E^B)$ with $V^B = E$ and $E^B = \bigcup_{e \in E \setminus B} E^B_e$, where the sets E^B_e for $e \in E \setminus B$ are defined as follows:*

1. *If $w_e \geq U_f$ for all $f \in C_e - e$, and there is a $f' \in C_e - e$ with $w_{f'} > L_e$, then $E^B_e = \{\{e, e\}\}$.*
2. *If $w_e \geq U_f$ for all $f \in C_e - e$, and $w_f \leq L_e$ for all $f \in C_e - e$, then $E^B_e = \{\{e, f\} \mid f \in F_e\}$.*
3. *If there exists $f \in C_e - e$ such that $w_e < U_f$, and there exists $f' \in C_e - e$ (could have $f' = f$) such that $w_{f'} > L_e$, then $E^B_e = \{\{f, f\} \mid f \in \hat{F}_e + e\}$.*
4. *If there exists $f \in C_e - e$ such that $w_e < U_f$, and for all $f' \in C_e - e$ it holds that $w_{f'} \leq L_e$, then $E^B_e = \{\{e, f\} \mid f \in F_e \setminus \hat{F}_e\} \cup \{\{f, f\} \mid f \in \hat{F}_e\}$.*

Each case of Definition 27 corresponds to a case of Lemma 26 and exactly models the corresponding "choice" of the lemma. The following corollary follows from Lemma 26 and formally proves (cf. the full version [4]) the connection between Definition 27 and Lemma 26.

▶ **Corollary 28.** Given a weighted uncertainty matroid $\mathcal{M} = (E, \mathcal{I}, A, w)$ with MWB B, a set Q is a certificate of B if and only if Q is a vertex cover of G^B.

Since a set $Q \subseteq E$ is a certificate for B if and only if Q is a vertex cover of G^B, we can compute a minimum-cost certificate for B by computing a minimum-cost vertex cover for G^B, using the query costs c_e as weights for the vertices $V^B = E$. Note that G^B is bipartite except for the self-loop edges. To employ polynomial-time vertex-cover algorithms for bipartite graphs (see, e.g., [24]), we modify G^B by replacing each loop-edge $\{e, e\} \in E^B$ with an edge $\{e, v_e\}$ to a distinct new dummy vertex v_e with weight $w(v_e) = w_e + \epsilon$ for some $\epsilon > 0$. Executing the algorithm, formalized as Algorithm 2, for the MWB B computed by Algorithm 1 yields a polynomial-time algorithm that computes a minimum-cost certificate.

▶ **Theorem 29.** *Let $\mathcal{M} = (E, \mathcal{I}, A, w)$ be a weighted uncertainty matroid. Algorithm 2 is a polynomial-time algorithm that computes a minimum-cost certificate Q and an MWB B of $\mathcal{M}$ that is verified by Q.*

Proof. The correctness of the algorithm immediately follows Corollary 28. Thus, it only remains to argue about the running-time. The MWB B can be computed in polynomial time by Lemma 25. Based on B, the graph G^B can also be computed in polynomial time.

To show that the minimum-weight vertex cover can be computed in polynomial time, note that G^B is bipartite except for the self-loop edges. Indeed, all non-loop edges have one endpoint in B and one endpoint in $E \setminus B$. To still compute a minimum-weight vertex cover for G^B using the algorithms for bipartite graphs (see, e.g., [24]), we define $\bar{G}^B$ as a copy of G^B that replaces each loop-edge $\{e, e\} \in E^B$ with an edge $\{e, v_e\}$ to a distinct new vertex v_e with weight $w(v_e) = w_e + \epsilon$ for some $\epsilon > 0$. Clearly, the size of $\bar{G}^B$ is polynomial in the size of G^B and a set Q is a minimum-weight vertex cover of G^B if and only if Q is a minimum-weight vertex cover for $\hat{G}^B$. Since $\hat{G}^B$ is bipartite, we can compute a minimum-weight vertex cover for $\hat{G}^B$, and thus for G^B, by using the classical algorithm for bipartite graphs. ◀

5 Applications

In this section, we show that our verification algorithm and our structural insights, in particular Lemma 26, can be used to obtain new results for the adaptive online variant and for two learning-augmented variants of the MWB problem under uncertainty.

5.1 A Best-Possible Adaptive Online Algorithm for a Given MWB

In the following, we give an optimal algorithm for the adaptive online problem under the assumption that we are given an MWB B that can be verified with a minimum-cost certificate.

In the adaptive online problem, the weights w_e are initially unknown, and an algorithm has access only to the uncertainty matroid $\mathcal{M}$. The goal of the algorithm is to adaptively query elements until the set of queried elements Q verifies some MWB B of $\mathcal{M}$ (cf. Definition 6). Algorithms are analyzed in terms of their *competitive ratio*. An algorithm is *ρ-competitive* if $c(Q(\mathcal{M})) \leq \rho \cdot c(Q^*(\mathcal{M}))$ for all uncertainty matroids $\mathcal{M}$, where $Q(\mathcal{M})$ is the set of elements queried by the algorithm on $\mathcal{M}$, and $Q^*(\mathcal{M})$ is a minimum-cost certificate for $\mathcal{M}$. The competitive ratio of an algorithm is the minimum ρ such that the algorithm is ρ-competitive.

The best-possible competitive ratio heavily depends on the type of uncertainty areas. If all uncertainty areas are either open, i.e., $A_e = (L_e, U_e)$, or trivial, i.e., $A_e = \{w_e\}$, then the best-possible competitive ratio is 2 [13]. Once the uncertainty areas can be closed intervals, the best-possible competitive ratio increases to n [11, Section 7]. Both results hold for uniform query costs ($c_e = 1$ for $e \in E$). Here, we consider the case of uniform query costs.

It is not hard to see that the lower bound of 2 for minimum spanning trees in the case of open uncertainty areas [13] holds even if the algorithm is given an MWB B with the promise that (i) B is indeed an MWB for the unknown weights and (ii) B can be verified with a minimum-cost certificate. Note that for open uncertainty areas, (ii) is an implication of (i).

▶ **Observation 30** (Follows from [13]). *No deterministic algorithm for the online adaptive MWB problem with uniform query costs is better than 2-competitive, even if the algorithm is given an MWB B w.r.t. to the unknown weights.*

In contrast, the lower bound of n for the case of general uncertainty areas [11, Section 7] does not hold if the algorithm receives the same promise as described above. This contrast highlights that the increase in the competitive ratio from *open* to *general* uncertainty areas stems from the algorithm's task of *finding* rather than merely *verifying* an MWB B. In the following, we formally prove this insight. The core of the proof is an application of Lemma 26.

▶ **Lemma 31.** *There exists an online adaptive algorithm that, given an uncertainty matroid $\mathcal{M}$ and an MWB B (w.r.t. the unknown weights w) that can be verified with a minimum-cardinality certificate Q^*, computes a certificate Q with $|Q| \leq 2|Q^*|$ that verifies B.*

Proof. To prove the lemma, we define the following algorithm:

1. Initialize $Q = \emptyset$.
2. For each $e \in E \setminus B$:
   a. Let C_e denote the fundamental circuit of e with respect to B.
   b. While there exists an $f \in C_e - e$ with $U_f(Q) > L_e(Q)$:
      i. If there is an element $g \in C_e \setminus (\{e\} \cup Q)$ with $U_g(Q) > L_e(Q)$, then let $\hat{g}$ denote such an element with maximum $U_{\hat{g}}(Q)$. Add $\hat{g}$ to Q and query $\hat{g}$.
      ii. If $e \notin Q$, then add e to Q and query e.
3. Return Q.

It is not hard to see that the set Q computed by the algorithm is a certificate that verifies B: Exploiting the assumption that B is indeed an MWB, the definition of Step 2b ensures that Q satisfies Corollary 9. Hence, Q is a certificate that verifies B.

It remains to show that $|Q| \leq 2 \cdot |Q^*|$. The proof will heavily exploit that Q^*, by assumption, is a certificate that verifies B. In particular, this means that Q^* satisfies Lemma 26 for B, which will help us to bound $|Q|$ in terms of $|Q^*|$.

Let $E \setminus B = \{e_1, \dots, e_k\}$ be indexed in the order in which the algorithm considers the elements of $E \setminus B$ in the loop of Step 2. For each $j \in \{1, \dots, k\}$ let Q_j denote the set of elements that the algorithm queries during iteration j of the loop of Step 2. We show that

$$\frac{|Q_j|}{|Q_j \cap Q^*|} \leq 2 \tag{1}$$

holds for each Q_j with $j \in \{1, \dots, k\}$. Since the sets Q_j form a partition of Q, this then implies $|Q| \leq 2|Q^*|$.

Fix an arbitrary Q_j. Define $P_j = \bigcup_{j'<j} Q_{j'}$. We show that Q_j satisfies (1) via case distinction over the cardinality of Q_j.

1. If $|Q_j| = 0$, then (1) holds trivially.
2. If $|Q_j| = 1$, then the definition of the loop of Step 2b implies that we must have $Q_j = \{e_j\}$ as $e_j \in Q_j$ holds by Step 2bii of the algorithm. The fact that the loop of Step 2b is executed but only queries e_j implies that all $g \in C_{e_j} \setminus (\{e_j\} \cup P_j)$ have $w_g \leq U_g(P_j) \leq L_{e_j}(P_j) = L_{e_j}$ (cf. Step 2bi). However, then the fact that the loop of Step 2b is executed implies that there is some $g \in C_{e_j} \cap P_j$ with $U_g(P_j) = w_g > L_{e_j}(P_j) = L_{e_j}$. This in turn implies that C_{e_j} either satisfies the first or third case of Lemma 26. Hence, $e_j \in Q^*$ and $\frac{|Q_j|}{|Q_j \cap Q^*|} = 1$.
3. If $|Q_j| = 2$, then Lemma 26 implies $|Q_j \cap Q^*| \geq 1$. Hence, $\frac{|Q_j|}{|Q_j \cap Q^*|} \leq 2$
4. If $|Q_j| > 2$, then the loop of Step 2b is executed at least twice. For the two elements that are queried in the first iteration, we can argue that Q^* has to contain at least one of them as in the previous case.

   Let Q'_j denote the subset of Q_j that is queried after the first iteration of this Step 2b loop. All $g \in Q'_j$ have to satisfy $U_g > w_{e_j}$, as otherwise the loop would terminate before g is queried. This implies that all $g \in Q'_j$ are in $\hat{F}_{e_j}$ (defined as in Lemma 26). Hence, C_{e_j} either satisfies Case 3 or 4 of Lemma 26 and, thus, $Q'_j \subseteq \hat{F}_{e_j} \subseteq Q^*$. We can conclude with $\frac{|Q_j|}{|Q_j \cap Q^*|} \leq \frac{2+|Q'_j|}{1+|Q'_j|} \leq 2$. ◀

5.2 Learning-Augmented Algorithms

Recently, online adaptive problems under explorable uncertainty have been studied in *learning-augmented* settings [6, 7], where the algorithm has access to imperfect predictions $\hat{w}_e$ on the unknown weights w_e. The goal is to design algorithms that leverage the access to the predictions to achieve an improved competitive ratio in case the predictions are accurate, and at the same time maintain worst-case guarantees even if the predictions are arbitrarily wrong. Such learning-augmented algorithms are analyzed w.r.t. their consistency, the competitive ratio if $w_e = \hat{w}_e$ for all $e \in E$, and robustness, the competitive ratio for arbitrary predictions [16, 22].

For the online adaptive MST problem with open uncertainty areas, Erlebach et al. [7] gave a 1.5-consistent and 2-robust algorithm. In the following we show that our verification algorithm and the insights from the previous section can be used to design learning-augmented algorithms for the online adaptive MWB problem with general uncertainty areas for two prediction models:

1. **Weight predictions:** As in [6, 7], the algorithm has access to predictions $\hat{w}_e$ on the unknown element weights w_e.
2. **Basis predictions:** The algorithm has access to a prediction $\hat{B}$ on the MWB B^* that can be verified with a minimum-cardinality certificate.

5.2.1 Weight Predictions

Our verification algorithm implies a 1-consistent and n-robust algorithm for the case of uniform query costs. Note that this consistency and robustness are optimal due to the aforementioned lower bound given in [11].

1. Use the verification algorithm to compute and query a minimum-cost certificate under the assumption that the predictions are accurate, i.e., $w_e = \hat{w}_e$ for all $e \in E$.
2. If the instance is not solved yet, then query all remaining elements.

▶ **Corollary 32.** *The algorithm above runs in polynomial time and is* 1*-consistent and* n*-robust for the online adaptive MWB problem with weight predictions and uniform query costs.*

Proof. Let Q denote the query set computed by the algorithm and let Q^* denote the minimum-cardinality query set. If the predictions are correct, $w_e = \hat{w}_e$ for all $e \in E$, then the second step of the algorithm computes and queries a minimum-cardinality certificate for the instance. Hence, Step 2 is never executed and the algorithm is 1-consistent. If the predictions are incorrect and the algorithm queries at least one element, then also $|Q^*| \geq 1$. Since the algorithm queries at most n elements, n-robustness follows immediately. ◀

This naive algorithm is a first step towards *smooth* learning augmented-algorithms for the online adaptive MWB problem with weight predictions, i.e., algorithms with a competitive ratio that smoothly degrades from 1 to n depending on some error measure describing the quality of the predictions. Next, we give such a smooth algorithm (with a worse consistency) for basis predictions. Since weight predictions can be used to compute a basis prediction, the algorithm can also be interpreted as a smooth algorithm for weight predictions.

5.2.2 Basis Predictions

We consider the prediction model where algorithms have access to a prediction $\hat{B}$ on an MWB B^* that can be verified with a minimum cardinality query set Q^*. W.l.o.g. we may assume that $\hat{B}$ is indeed a basis of the given uncertainty matroid. This is easy to check and if $\hat{B}$ is not a basis, we remove a minimum number of elements to achieve independence and then augment it to a basis.

We say that the prediction $\hat{B}$ is *correct* if it is indeed an MWB that can be verified with some minimum cardinality query set. Otherwise, the prediction is *incorrect.* We now define error measures to quantify the quality of a prediction $\hat{B}$. Note that a prediction $\hat{B}$, can have two different types of error. First, $\hat{B}$ might not actually be an MWB. Second, even if $\hat{B}$ is an MWB, it might not be an MWB of minimum verification cost. Our error measure takes both of these error types into account. To this end, let C_e for $e \in E \setminus \hat{B}$ denote the fundamental circuit of e with respect to $\hat{B}$. Define such a circuit C_e to be *correct* if $w_e \geq w_f$ for all $f \in C_e - e$ and $w_f \leq w_g$ for all $f \in C_e - e$ and $g \in E \setminus \text{span}_M(\hat{B} - f)$. Otherwise, call C_e *incorrect.* Let $\hat{B}_s \subseteq \hat{B}$ denote the set of elements in $\hat{B}$ that are part of a correct fundamental circuit C_e, $e \in E \setminus \hat{B}$. We define two errors:

- Let $\hat{B}_s^*$ denote an MWB with $\hat{B}_s \subseteq \hat{B}_s^*$ such that $\hat{B}_s^*$ has minimum verification cost among all MWB's that contain $\hat{B}_s$ and let Q' denote a minimum-cardinality certificate for $\hat{B}_s^*$. In the full version [4], we argue that such an MWB $\hat{B}_s^*$ always exists. Define

$\eta_1 = |Q'| - |Q^*|$, where Q^* is the minimum-cardinality certificate. Note that if $\hat{B}$ is an MWB, then η_1 is just the difference between the verification cost of $\hat{B}$ and the minimum verification cost $|Q^*|$.

- Let η_2 denote the number of incorrect circuits C_e with $e \in E \setminus \hat{B}$.

Intuitively, η_1 measures how much larger the verification cost of $\hat{B}$ is compared to the minimum-cardinality query set. Error η_2 captures how far $\hat{B}$ is from actually being an MWB.

The following theorem can be achieved using a slight variation of the algorithm in Section 5.1, taking into account that elements $e \in E \setminus \hat{B}$ might not be maximal on circuit C_e. In such cases, the algorithm deviates from the procedure in Section 5.1 by querying all elements in circuit C_e. In the full version [4], we give the formal definition of this algorithm and the proof of the theorem, and argue that the factor $c_{\max}$ of the error η_2 is indeed necessary.

▶ **Theorem 33.** *Given an instance of the online adaptive MWB problem with a predicted basis $\hat{B}$ and uniform query costs, there is an algorithm that computes a certificate Q for some MWB B and satisfies $|Q| \leq \min\{2 \cdot (|Q^*| + \eta_1) + \eta_2 \cdot c_{\max}, n\}$, where Q^* is a minimum-cardinality certificate and $c_{\max}$ is the size of the largest circuit. In particular, the algorithm is 2-consistent and n-robust.*

References

1 Evripidis Bampis, Christoph Dürr, Thomas Erlebach, Murilo Santos de Lima, Nicole Megow, and Jens Schlöter. Orienting (hyper)graphs under explorable stochastic uncertainty. In *ESA*, volume 204 of *LIPIcs*, pages 10:1–10:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ESA.2021.10.

2 Steven Chaplick, Magnús M. Halldórsson, Murilo Santos de Lima, and Tigran Tonoyan. Query minimization under stochastic uncertainty. In *LATIN*, volume 12118 of *Lecture Notes in Computer Science*, pages 181–193. Springer, 2020. doi:10.1007/978-3-030-61792-9_15.

3 George Charalambous and Michael Hoffmann. Verification problem of maximal points under uncertainty. In *IWOCA 2013*, volume 8288 of *Lecture Notes in Computer Science*, pages 94–105. Springer, 2013. doi:10.1007/978-3-642-45278-9_9.

4 Haya Diwan, Lisa Hellerstein, Nicole Megow, and Jens Schlöter. Optimal verification of a minimum-weight basis in an uncertainty matroid, 2025. arXiv:2512.17116.

5 T. Erlebach and M. Hoffmann. Query-competitive algorithms for computing with uncertainty. *Bulletin of the EATCS*, 116:22–39, 2015. URL: http://bulletin.eatcs.org/index.php/beatcs/article/view/335.

6 Thomas Erlebach, Murilo S. de Lima, Nicole Megow, and Jens Schlöter. Sorting and hypergraph orientation under uncertainty with predictions. In *IJCAI*, pages 5577–5585. ijcai.org, 2023. doi:10.24963/IJCAI.2023/619.

7 Thomas Erlebach, Murilo Santos de Lima, Nicole Megow, and Jens Schlöter. Learning-augmented query policies for minimum spanning tree with uncertainty. In *ESA*, volume 244 of *LIPIcs*, pages 49:1–49:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.ESA.2022.49.

8 Thomas Erlebach and Michael Hoffmann. Minimum spanning tree verification under uncertainty. In Dieter Kratsch and Ioan Todinca, editors, *Graph-Theoretic Concepts in Computer Science*, volume 8747 of *Lecture Notes in Computer Science*, pages 164–175. Springer International Publishing, Cham, 2014. doi:10.1007/978-3-319-12340-0_14.

9 Thomas Erlebach, Michael Hoffmann, and Frank Kammer. Query-competitive algorithms for cheapest set problems under uncertainty. *Theor. Comput. Sci.*, 613:51–64, 2016. doi:10.1016/J.TCS.2015.11.025.

10 T. Feder, R. Motwani, R. Panigrahy, C. Olston, and J. Widom. Computing the median with uncertainty. *SIAM Journal on Computing*, 32(2):538–547, 2003. doi:10.1137/S0097539701395668.

11 Manoj Gupta, Yogish Sabharwal, and Sandeep Sen. The update complexity of selection and related problems. *Theory Comput. Syst.*, 59(1):112–132, 2016. doi:10.1007/S00224-015-9664-Y.

12 Magnús M. Halldórsson and Murilo Santos de Lima. Query-competitive sorting with uncertainty. *Theor. Comput. Sci.*, 867:50–67, 2021. doi:10.1016/J.TCS.2021.03.021.

13 Michael Hoffmann, Thomas Erlebach, Danny Krizanc, Matús Mihalák, and Rajeev Raman. Computing minimum spanning trees with uncertainty. In *STACS*, volume 1 of *LIPIcs*, pages 277–288. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Germany, 2008. doi:10.4230/LIPIcs.STACS.2008.1358.

14 Simon Kahan. A model for data in motion. In *Proceedings of the Twenty-Third Annual ACM Symposium on Theory of Computing*, pages 265–277, New York, NY, USA, 1991. Association for Computing Machinery. doi:10.1145/103418.103449.

15 Eugene L Lawler. *Combinatorial optimization: networks and matroids*. Courier Corporation, 2001.

16 Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. *J. ACM*, 68(4):24:1–24:25, 2021. doi:10.1145/3447579.

17 Corinna Mathwieser and Eranda Çela. Special cases of the minimum spanning tree problem under explorable edge and vertex uncertainty. *Networks*, 83(3):587–604, 2024. doi:10.1002/NET.22204.

18 Nicole Megow, Julie Meißner, and Martin Skutella. Randomization helps computing a minimum spanning tree under uncertainty. *SIAM J. Comput.*, 46(4):1217–1240, 2017. doi:10.1137/16M1088375.

19 Nicole Megow and Jens Schlöter. Set selection under explorable stochastic uncertainty via covering techniques. In *IPCO*, volume 13904 of *Lecture Notes in Computer Science*, pages 319–333. Springer, 2023. doi:10.1007/978-3-031-32726-1_23.

20 J. Meißner. *Uncertainty Exploration: Algorithms, Competitive Analysis, and Computational Experiments*. PhD thesis, Technischen Universität Berlin, 2018. doi:10.14279/depositonce-7327.

21 Arturo Merino and José A. Soto. The minimum cost query problem on matroids with uncertainty areas. In *ICALP*, volume 132 of *LIPIcs*, pages 83:1–83:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.ICALP.2019.83.

22 Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving online algorithms via ML predictions. In *NeurIPS*, pages 9684–9693, 2018. URL: https://proceedings.neurips.cc/paper/2018/hash/73a427badebe0e32caa2e1fc7530b7f3-Abstract.html.

23 Jens Schlöter. On the complexity of knapsack under explorable uncertainty: Hardness and algorithms. In *33rd Annual European Symposium on Algorithms*, volume 351 of *LIPIcs*, pages 6:1–6:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.ESA.2025.6.

24 Alexander Schrijver. *Combinatorial Optimization – Polyhedra and Efficiency*. Springer, 2003.