Second-Order Quantifier Elimination and Uniform Interpolation for Basic Path Logic and the Ordered Fragment (Extended Abstract)*

Renate A. Schmidt[1,*], Hongkai Yin[2]

[1]The University of Manchester, Oxford Road, Manchester M13 9PL, UK
[2]Central European University, Quellenstraße 51, 1100 Wien, Austria

Abstract

We consider and extend results on basic path logic and the ordered fragment of first-order logic, both of which originate from the functional translation of modal logic. Basic path logic is a subclass of the clausal class for the $\exists^*\forall^*$-fragment and has the remarkable property that binary resolution decides it. This decidability result and the consequence finding completeness of binary resolution allows us to observe that binary resolution also decides uniform interpolation and computes uniform interpolants for basic path logic. By introducing constant Skolemisation, we show that sentences of the ordered fragment can be transformed into basic path logic, and this transformation preserves logical consequences in the ordered fragment. We characterise the search space of the SCAN algorithm on the clausal form of the ordered fragment by a variation of basic path logic and prove that SCAN terminates on this class, and therefore it decides second-order quantifier elimination for this class. It remains unclear whether uniform interpolants in the ordered fragment can be extracted from the output of SCAN.

Keywords

First-order logic, Second-order quantifier elimination, Uniform interpolation, SCAN, Resolution

In this paper we investigate the application of resolution and the SCAN algorithm [1] in computing uniform interpolants and solving second-order quantifier elimination for basic path logic [2, 3].

We say that a clause is *prefix-stable for variables* if for any *variable* in the clause, its prefix in any argument sequence (in which it occurs) is the same [4]. For example, the first clause below is prefix stable for variables and the second is not (where y has two different prefixes):

$$P(x, y) \lor Q(x, y, a)$$
$$P(x, y) \lor Q(a, y, z).$$

By *basic path logic* (BPL) we understand the clausal class in which: (i) all argument sequences contain only variables and constants, and (ii) all clauses have prefix stability for variables. This implies that BPL is a subclass of the clausal class for the $\exists^*\forall^*$-fragment of first-order logic.

Basic path logic has been introduced as the clausal class associated with modal logic KD based on the world path semantics and the functional translation to first-order logic [2, 3]. Basic path logic is computationally well-behaved: it is decidable by binary resolution without special refinements (no ordering- or selection-based restriction of inferences is needed to force termination) [3]. This implies that any sound and refutationally complete ordering- and/or selection-based refinement provides a decision procedure for this class. (On the side we remark that these observations do not extend

CI-BD-SOQE 2026: Workshop on Craig Interpolation, Beth Definability, and Second-Order Quantifier Elimination, at FLoC 2026: The 9th Federated Logic Conference, July 24–25, 2026, Lisbon, Portugal

*This is an abstract of a paper with the same title published as arXiv Preprint `arXiv:2607.03645 [cs.LO]` which is under review for publication in a journal. Earlier versions of this work were presented in "First-Order Reasoning, Below and Beyond", Workshop in Honor of Christoph Weidenbach's 60th Birthday, held in August 2025, and Dagstuhl Seminar 26091 on "Revisiting the Foundations of Deduction in a New World", held in February 2026.

*Corresponding author.
Renate.Schmidt@manchester.ac.uk (R. A. Schmidt)
0000-0002-6673-3333 (R. A. Schmidt); 0009-0006-0082-6541 (H. Yin)

to the $\exists^*\forall^*$ clausal class, which is NEXPTIME-complete and can be decided by a range-restriction transformation and hyper-resolution [5].) Further, the performance of resolution theorem provers is superior for modal logic problems when they are mapped to basic path logic than when they are mapped to the guarded fragment using the standard relational translation [6, 7].

In the *ordered fragment* of first-order logic, the argument sequence of each predicate is of the form $(x_1, x_2, \ldots, x_n)$, and if $i < j$, no quantifier binding x_j out-scopes any quantifier binding x_i. For example, the sentence below belongs to the fragment:

$$\forall x_1 \exists x_2 (R(x_1, x_2) \wedge \forall x_3 \neg S(x_1, x_2, x_3)).$$

The ordered fragment first appeared in the "fluted schema" introduced by W. V. Quine [8, 9], but the formal definition we follow was given, independently of Quine's work, by A. Herzig [10], where the term "ordered formula" was coined. Herzig proposed a linear reduction from satisfiability of ordered formulas to satisfiability in KD, which implies that the former is decidable in PSPACE. Also, it is proved in [2] that the quantifier prefix in the prenex normal form of any ordered formula can be freely permuted without affecting satisfiability; thus, by this permutability, every ordered formula can be transformed into BPL [11, 3].

Second-order quantifier elimination (SOQE) is the problem of reducing a formula $\exists \overline{X} \varphi$, where $\overline{X}$ are predicate variables and φ is a first-order formula, to an equivalent first-order formula ψ (possibly with equality) that does not contain any of the predicate symbols $\overline{X}$ [1, 12]. For example,

$$\exists X (Xa \wedge \forall x (Xx \to Bx)) \equiv Ba \quad \text{and} \quad \exists X (Xa \wedge \neg Xb) \equiv a \not\approx b.$$

SOQE is an important topic of artificial intelligence that has real-world applications in diverse areas, from knowledge representation, knowledge sharing and communication of agents, to answer set programming and logic, including description logics and modal correspondence theory. SOQE has been studied in the guise of forgetting [13], strongest necessary and weakest sufficient conditions in knowledge representation [14, 15, 16, 17, 18, 19], symbol elimination and projection [20]. SOQE has applications in knowledge representation for ontology extraction, ontology comparison, and abductive reasoning [15, 16, 21, 22, 23, 24, 25], and in artificial intelligence for common-sense reasoning and circumscription [1], databases [12] and answer set programming [26, 27].

SOQE is tightly connected to the problem of computing *uniform interpolants*. Given a first-order formula φ and a subset Σ of $\operatorname{sig}(\varphi)$, the signature of φ, a uniform interpolant of φ with regard to Σ is a first-order formula ψ such that $\operatorname{sig}(\psi) = \Sigma$ and

$$\models \varphi \to \chi \quad \text{iff} \quad \models \psi \to \chi \text{ for every } \chi \text{ with } \operatorname{sig}(\varphi) \cap \operatorname{sig}(\chi) \subseteq \Sigma.$$

Note that if $\overline{X} = \operatorname{sig}(\varphi) \setminus \Sigma$ and ψ is the result of SOQE for φ, i.e., $\psi \equiv \exists \overline{X} \varphi$, then ψ is a uniform interpolant of φ with regard to Σ. For example, Ba is a uniform interpolant of $Xa \wedge \forall x(Xx \to Bx)$ with regard to the signature containing only B and a, and $a \not\approx b$ is a uniform interpolant of $Xa \wedge \neg Xb$ with regard to the signature containing only a and b. Also, when we are interested in uniform interpolation in a fragment of first-order logic, the aforementioned φ, ψ and χ are assumed to be formulas of the fragment in question. In other words, the construction of uniform interpolants is dependent on the target language. For example, a uniform interpolant of $Xa \wedge \neg Xb$ is $\top$ in first-order logic *without* equality (instead of $a \not\approx b$). However, the target language of SOQE is always assumed to be first-order logic *with* equality.

There are few settings in which SOQE and uniform interpolation are known to be decidable. For propositional logic SOQE and uniform interpolation are decidable, but in more general settings, e.g., for many modal logics, description logics and first-order logic, they are not even semi-decidable.

Efforts to automate modal correspondence theory led to the development of the SOQE algorithm SCAN for first-order clause logic [1, 28]. Given a clause set N and predicate variables $\overline{X}$, the SCAN algorithm

Table 1
The C-inference rules.

C-**Resolution** $\dfrac{C \vee X(\overline{s}) \qquad D \vee \neg X(\overline{t})}{C \vee D \vee \overline{s} \not\approx \overline{t}}$

provided X is predicate variable, the two premises have no individual variables in common, and the premises are distinct (to prevent self-resolution).

C-**Factoring** $\dfrac{C \vee (\neg)X(\overline{s}) \vee (\neg)X(\overline{t})}{C \vee (\neg)X(\overline{s}) \vee \overline{s} \not\approx \overline{t}}$

provided X is a predicate variable.

attempts to eliminate the symbols $\overline{X}$ in an iterative process performing all necessary inferences upon $\overline{X}$-literals and deleting clauses once they are no longer needed. On successful termination, the returned clause set M is equivalent to $\exists\overline{X}N$. The soundness of SCAN for the SOQE problem has been proved in [1], but because of the inherent undecidability of SOQE it cannot generally be complete. It was however shown that SCAN is complete for solving the correspondence problem of the class of Sahlqvist axioms in modal logic [29]. It is also known to be sound and complete for propositional problems. Moreover, if SCAN does not terminate then the conjunction of the infinite set of persisting clauses (closed under first-order universal quantification and) free of predicate variables from $\overline{X}$ is a solution to the problem [1].

When SCAN is applied to a first-order formula φ, the formula is first transformed into clausal form as implemented in the Otter theorem prover. This involves four steps: (i) transforming the formula into prenex normal form $\overline{Qx}\varphi'$, which moves the quantifiers to the outside of the formula so that φ' is free of quantifiers. (ii) Skolemisation: the replacement of existentially quantified variables by Skolem terms and dropping the existential quantifiers, which results in a formula $\forall\overline{y}\varphi''$ where φ'' is quantifier-free and $\overline{y} \subseteq \overline{x}$. In the next step, (iii), φ'' is transformed into conjunctive normal form. The clausal form $N = \mathrm{Cls}(\varphi)$ is then obtained by (iv) writing φ'' as a set of clauses.

The idea of the SCAN algorithm is to generate sufficiently many logical consequences of N and keep only those non-redundant clauses in which no symbols from $\overline{X}$ occur. This is realised by a saturation process which interleaves (i) inference steps, (ii) normalisation and redundancy elimination steps, and (iii) purifying deletion steps. These are based on the inference system C defined by the rules given in Tables 1 and 2. Also, we assume that *clauses are eagerly normalised using:*

Constraint elimination: $C \vee s \not\approx t \;\Rightarrow_{\mathrm{CEl}}\; C\sigma$

if σ is a most general unifier of $s \approx t$, i.e., $s\sigma = t\sigma$ and σ is the most general substitution with this property.

Because of the mentioned nice properties of basic path logic in resolution, the question arises whether the SCAN algorithm can be applied to solve the SOQE problem and compute uniform interpolants in this case, and what can be concluded for the ordered fragment. The main research questions and contributions of this paper are the following:

1. Uniform Interpolation in Basic Path Logic. Since unrefined binary resolution is in general complete for consequence finding [30], and, with subsumption deletion, it is guaranteed to terminate on clauses of basic path logic, unrefined binary resolution with subsumption deletion is a decision procedure for consequence finding (up to subsumption) in basic path logic. Also, as a uniform interpolant of a formula (with regard to a signature) is the strongest logical consequence of the formula (over the signature), this in turn implies that binary resolution with subsumption deletion computes uniform interpolants in basic path logic.

Table 2
The C-resolution system.

Inference $\dfrac{N}{N \cup \{\mathrm{norm}(C)\}}$
where C is a C-resolvent or C-factor of clauses in N, and $C \Rightarrow^*_{\mathrm{CEl}} \mathrm{norm}(C)$.

Redundancy elimination $\dfrac{N}{M}$
if M is obtained from N by redundancy elimination.

Purified clause deletion $\dfrac{N \cup \{C \vee (\neg)X(\overline{s})\}}{N}$
provided X is a predicate variable and $C \vee (\neg)X(\overline{s})$ is *purified in* N *with respect to the* X*-literal* $(\neg)X(\overline{s})$, i.e., no non-redundant inferences are possible with the clause $C \vee (\neg)X(\overline{s})$ upon the literal $(\neg)X(\overline{s})$ and clauses in N.

Extended purity deletion$^{+(-)}$ $\dfrac{N \cup M}{N}$
if N and M are disjoint, N is free of predicate variable X, and every clause in M contains X positively (negatively).

2. Characterisation of the Search Space of SCAN on Basic Path Logic with Fixed Constant Positions. The SCAN algorithm based on constraint resolution replaces unification in the resolution and factoring rules by adding inequalities (constraints) to conclusions. These constraints fall outside the scope of the language of basic path logic, which means the latter is insufficient to characterise the search space of SCAN. For that reason, and because BPL is slightly more general than it needs to be for deciding the ordered fragment and modal logic KD, we introduce the clausal class $BPL^{\not\approx}_{fcp}$, in which inequalities are allowed and constants have fixed argument positions.

We assume that each constant is associated with an index i, which stands for the argument position in which the constant is allowed to occur in non-constraint literals. $BPL^{\not\approx}_{fcp}$ is the class of $\exists^*\forall^*$ clauses in which the following conditions hold. For any two non-constraint literals $(\neg)P(\underline{u})$ and $(\neg)Q(\underline{v})$ in a clause C,

L1 if u_i and v_j are the same variable then $i = j$, and

L2 if u_k, v_k, u_i and v_i are all variables, u_i is the same as v_i, and $k < i$, then u_k and v_k are the same.

L3 For any constant a, if the index of a is i then a occurs only at argument position i in non-constraint literals of any clause.

L4 In each constraint $u \not\approx v$ of a clause C, u and v are *position compatible*, i.e., if u is a variable occurring at argument position i in a non-constraint literal of C or u is a constant with index i, then either v is a variable occurring at argument position i in a non-constraint literal of C, v is singular (does not occur anywhere else in C) or v is a constant with index i. We say that the *position associated with* $u \not\approx v$ is i.

We show that $BPL^{\not\approx}_{fcp}$ is closed under the rules of SCAN, and thus each clause in the search space of SCAN belongs to $BPL^{\not\approx}_{fcp}$ when the input is a set of BPL clauses where constants occur in fixed argument positions (i.e., **L3** is safisfied).

3. Termination and Completeness of SCAN on $BPL^{\not\approx}_{fcp}$. We further show that the set of $BPL^{\not\approx}_{fcp}$-clauses over a finite relational signature is finitely bounded modulo condensing (a form of subsumption deletion). This implies that SCAN is a decision procedure for SOQE and is SOQE-complete for $BPL^{\not\approx}_{fcp}$. The result can be seen to follow from the characterisation and properties of BPL in [11, 3], noting that normalised conclusions of SCAN inferences are either standard resolvents or conditionals enumerating the coincidence of pairs of constants with the same indices.

4. Consequence-Preserving Clausification for the Ordered Fragment. The previous transformation of ordered formulas to clauses in BPL or $BPL^{\not\approx}_{fcp}$ involves an unusual quantifier exchange operation [2]. Although it is sufficient for solving the satisfiability problem, for uniform interpolation we need a clausification that preserves *logical consequences* in the ordered fragment. To address this problem, we define *constant Skolemisation* (CSk), in which new *constants* are introduced for all eliminated quantifiers and no functional terms are added. (The definition below is adapted from that of *inner Skolemisation* [31, 32] presented in [12, §3.3.5].) Let ϕ be a first-order formula in which no occurrence of subformulas of ϕ is of zero polarity. The *constant Skolemisation of* ϕ, written $\mathrm{CSk}(\phi)$, is the result of exhaustive application of the following operation.

1. If δ is an occurrence of a subformula of the form $\exists x\gamma$ and is of positive polarity, pick a fresh constant c and replace δ by $\gamma(x/c)$, the result of substituting c for x in γ.
2. If δ is an occurrence of a subformula of the form $\forall x\gamma$ and is of negative polarity, pick a fresh constant c and replace δ by $\gamma(x/c)$, the result of substituting c for x in γ.

For example,

$$\mathrm{CSk}(\forall x\exists y\forall z P(x,y,z)) = \forall x\forall z P(x,c,z)$$
$$\mathrm{CSk}(\exists x\forall y(R(x,y) \vee \neg\forall z S(x,y,z))) = \forall y(R(c_1,y) \vee \neg S(c_1,y,c_2)).$$

Based on [33], where the bisimulation for the ordered fragment is proposed and a characterisation of its expressivity is provided, we show that, for any sentences ϕ and ψ of the ordered fragment,

$$\models \phi \to \psi \text{ iff } \models \mathrm{CSk}(\phi) \to \psi.$$

Moreover, it is clear that the clausification of ordered sentences using CSk also results in BPL and $BPL^{\not\approx}_{fcp}$ clauses.

5. Uniform Interpolation in the Ordered Fragment. Unskolemisation of a clause set returned by SCAN is always possible since the clauses in question contain only variables and constants. Specifically, every such clause set can be expressed as a first-order formula in prenex normal form with a $\exists^*\forall^*$ quantifier prefix. However, such a formula is in general not expressible in the ordered fragment.

As a partial solution to the problem, we show that, in the output of SCAN, clauses with inequalities between constants can be deleted without any cost on logical consequences in equality-free first-order logic, and therefore we can obtain a "uniform interpolant" *expressed in basic path logic*. It remains open whether an additional procedure can be devised to construct a uniform interpolant *in* the ordered fragment.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] D. M. Gabbay, H. J. Ohlbach, Quantifier elimination in second-order predicate logic, South African Computer Journal 7 (1992) 35–43. And in *Proc. KR'92*, pp. 425–436. Morgan Kaufmann (1992).

[2] H. J. Ohlbach, R. A. Schmidt, Functional translation and second-order frame properties of modal logics, Journal of Logic and Computation 7 (1997) 581–603.
[3] R. A. Schmidt, Decidability by resolution for propositional modal logics, Journal of Automated Reasoning 22 (1999) 379–396.
[4] H. J. Ohlbach, A resolution calculus for modal logics, in: E. L. Lusk, R. A. Overbeek (Eds.), Proceedings of 9th International Conference on Automated Deduction (CADE-9), volume 310 of *Lecture Notes in Computer Science*, Springer, 1988, pp. 500–516.
[5] P. Baumgartner, R. A. Schmidt, Blocking and other enhancements for bottom-up model generation methods, Journal of Automated Reasoning 64 (2020) 197–251.
[6] U. Hustadt, R. A. Schmidt, On evaluating decision procedures for modal logics, in: M. Pollack (Ed.), Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI'97), Morgan Kaufmann, 1997, pp. 202–207.
[7] U. Hustadt, R. A. Schmidt, An empirical analysis of modal theorem provers, Journal of Applied Non-Classical Logics 9 (1999) 479–522.
[8] W. V. Quine, On the limits of decision, in: Akten des XIV. Internationalen Kongresses für Philosophie, volume 3, 1969, pp. 57–62.
[9] W. V. Quine, The variable, in: R. Parikh (Ed.), Logic Colloquium, Springer, 1975, pp. 155–168.
[10] A. Herzig, A new decidable fragment of first order logic, in: Abstracts of the 3rd Logical Biennial Summer School and Conference in Honour of S. C. Kleene, 1990.
[11] R. A. Schmidt, Optimised Modal Translation and Resolution, Ph.D. thesis, Universität des Saarlandes, Saarbrücken, Germany, 1997.
[12] D. M. Gabbay, R. A. Schmidt, A. Szałas, Second-Order Quantifier Elimination, College Publications, 2008.
[13] F. Lin, R. Reiter, Forget it!, in: Proceedings of the AAAI Fall Symposium on Relevance, New Orleans, 1994.
[14] B. Konev, D. Walther, F. Wolter, Forgetting and uniform interpolation in large-scale description logic terminologies, in: C. Boutilier (Ed.), Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009), 2009, pp. 830–835.
[15] P. Koopmann, R. A. Schmidt, Uniform interpolation of $\mathcal{ALC}$-ontologies using fixpoints, in: P. Fontaine, C. Ringeissen, R. A. Schmidt (Eds.), Proceedings of the 9th International Symposium on Frontiers of Combining Systems (FroCoS 2013), volume 8152 of *Lecture Notes in Artificial Intelligence*, Springer, 2013, pp. 87–102.
[16] P. Koopmann, R. A. Schmidt, Forgetting concept and role symbols in $\mathcal{ALCH}$-ontologies, in: K. McMillan, A. Middeldorp, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning (LPAR 2013), volume 8312 of *Lecture Notes in Computer Science*, Springer, 2013, pp. 552–567.
[17] J. P. Delgrande, A knowledge level account of forgetting, Journal of Artificial Intelligence Research 60 (2017) 1165–1213.
[18] H. Férée, I. van der Giessen, S. van Gool, I. Shillito, Mechanised uniform interpolation for modal logics k, gl, and isl, in: C. Benzmüller, M. J. H. Heule, R. A. Schmidt (Eds.), Automated Reasoning - 12th International Joint Conference, IJCAR 2024, Nancy, France, July 3-6, 2024, Proceedings, Part II, volume 14740 of *Lecture Notes in Computer Science*, Springer, 2024, pp. 43–60.
[19] P. Doherty, W. Lukaszewicz, A. Szalas, Computing strongest necessary and weakest sufficient conditions of first-order formulas, in: B. Nebel (Ed.), Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI 2001), Morgan Kaufmann, 2001, pp. 145–154.
[20] D. Peuter, V. Sofronie-Stokkermans, Symbol elimination and applications to parametric entailment problems, in: B. Konev, G. Reger (Eds.), Frontiers of Combining Systems (FroCoS 2021), volume 12941 of *Lecture Notes in Computer Science*, Springer, 2021, pp. 43–62.
[21] J. Chen, G. Alghamdi, R. A. Schmidt, D. Walther, Y. Gao, Ontology extraction for large ontologies via modularity and forgetting, in: M. Kejriwal, P. A. Szekely, R. Troncy (Eds.), Proceedings of the 10th International Conference on Knowledge Capture (K-CAP'19), ACM, 2019, pp. 45–52.

[22] M. Ludwig, B. Konev, Practical uniform interpolation and forgetting for ALC tboxes with applications to logical difference, in: C. Baral, G. De Giacomo, T. Eiter (Eds.), Principles of Knowledge Representation and Reasoning (KR 2014), AAAI Press, 2014.

[23] Z. Liu, C. Lu, G. Alghamdi, R. A. Schmidt, Y. Zhao, Tracking semantic evolutionary changes in large-scale ontological knowledge bases, in: G. Demartini, G. Zuccon, J. S. Culpepper, Z. Huang, H. Tong (Eds.), CIKM'21: The 30th ACM International Conference on Information and Knowledge Management, ACM, 2021, pp. 1130–1139.

[24] W. Del-Pinto, R. A. Schmidt, ABox abduction via forgetting in ALC, in: P. Van Hentenryck, Z.-H. Zhou (Eds.), Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-2019), AAAI Press, 2019, pp. 2768–2775.

[25] P. Koopmann, W. Del-Pinto, S. Tourret, R. A. Schmidt, Signature-based abduction for expressive description logics, in: D. Calvanese, E. Erdem, M. Thielscher (Eds.), Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning (KR 2020), AI Press, 2020, pp. 592–602.

[26] T. Eiter, G. Kern-Isberner, A brief survey on forgetting from a knowledge representation and reasoning perspective, Künstliche Intelligenz 33 (2019) 9–33.

[27] R. Gonçalves, M. Knorr, J. Leite, Forgetting in answer set programming: A survey, Theory and Practice of Logic Programming 23 (2023) 111–156.

[28] T. Engel, Quantifier Elimination in Second-Order Predicate Logic, Diplomarbeit, Fachbereich Informatik, Universität des Saarlandes, Saarbrücken, 1996.

[29] V. Goranko, U. Hustadt, R. A. Schmidt, D. Vakarelov, SCAN is complete for all Sahlqvist formulae, in: R. Berghammer, B. Möller, G. Struth (Eds.), Relational and Kleene-Algebraic Methods in Computer Science (RelMiCS 7), volume 3051 of *Lecture Notes in Computer Science*, Springer, 2004, pp. 149–162.

[30] R. C. T. Lee, A Completeness Theorem and a Computer Program for Finding Theorems from Given Axioms, Ph.D. thesis, University of California, Berkeley, 1967.

[31] A. Nonnengart, Strong Skolemization, Technical Report MPI-I-1996-2-010, Max-Planck-Institut für Informatik, Saarbrücken, Germany, 1996.

[32] A. Nonnengart, C. Weidenbach, Computing small clause normal forms, in: A. Robinson, A. Voronkov (Eds.), Handbook of Automated Reasoning, volume I, Elsevier Science, 2001, pp. 335–367.

[33] B. Bednarczyk, R. Jaakkola, Towards a model theory of ordered logics: Expressivity and interpolation, in: S. Szeider, R. Ganian, A. Silva (Eds.), 47th International Symposium on Mathematical Foundations of Computer Science (MFCS 2022), volume 241 of *Leibniz International Proceedings in Informatics*, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022, pp. 15:1–15:14.