Widening Operators for Weakly-Relational Numeric Abstractions*
(Extended Abstract)

Roberto Bagnara[1], Patricia M. Hill[2], Elena Mazzi[1], and Enea Zaffanella[1]

[1] Department of Mathematics, University of Parma, Italy
{bagnara,mazzi,zaffanella}@cs.unipr.it
[2] School of Computing, University of Leeds, UK
hill@comp.leeds.ac.uk

1 Introduction

In recent years there has been a lot of interest in the definition of so-called *weakly-relational* numeric domains, whose complexity and precision are in between the (non-relational) abstract domain of intervals [9] and the (relational) abstract domain of convex polyhedra [10]. The first weakly-relational domain proposed in the literature is based on systems of constraints of the form $x - y \leq c$ and $\pm x \leq c$, typically represented by Difference-Bound Matrices (DBMs). Even though DBMs have a long tradition in Computer Science, their use in the Abstract Interpretation field is quite recent. The idea of defining an abstract domain of DBMs was put forward in [1], where these constraints were called *bounded differences*. An independent application can be found in [19], where an abstract domain of transitively closed DBMs is defined. In this case, the transitive closure requirement was meant as a simple and well understood way to obtain a *canonical form* for the domain elements, so as to abstract away from merely syntactic differences. In [19] the specification of all the required abstract semantics operators is provided, including an operator that is meant to match the *standard widening* operator defined on the domain of convex polyhedra [10]. Unfortunately, as pointed out in [14, 15], this operator is not a widening since it does not provide a convergence guarantee for the abstract iteration sequence.

The abstract domain of (not necessarily transitively closed) DBMs is considered in [14]. In this more concrete, syntactic domain the transitive closure operator behaves as a kernel operator (monotonic, idempotent and reductive) mapping each DBM into the smallest DBM (with respect to the component-wise ordering) encoding the same geometric shape. As done in [19], a widening operator is also defined in [14] and it is observed that this widening "has some intriguing interactions" with transitive closure, therefore identifying the divergence issue faced in [19]. This observation has led to the conclusion that

* This work has been partly supported by projects "Constraint Based Verification of Reactive Systems" and "AIDA — Abstract Interpretation Design and Applications" and by the Royal Society International Joint Project — 2004/R1 Europe (ESEP).

"fixpoint computations *must* be performed" in the lattice of DBMs, without enforcing transitive closure [14].

2 Difference-Bound Shapes

While the analysis of the divergence problem is absolutely correct, the solution identified in [14] is sub-optimal since, as is usually the case, resorting to a syntactic domain (such as the one of DBMs) has a number of negative consequences. To identify a simpler, more natural solution, we first have to acknowledge that an element of this abstract domain should be a geometric shape, rather than (any) one of its matrix representations. To stress this concept, such an element will be called a *Difference-Bound Shape* (DBS). A DBS corresponds to the equivalence class of all the DBMs representing it. The implementation of the abstract domain can freely choose between these possible representations, switching at will from one to the other, as long as the semantic operators are implemented as expected. The other step towards the solution of the divergence problem is the simple observation that a DBS is a convex polyhedron and the set of all DBSs is closed under the application of the standard widening on polyhedra. Thus, no divergence problem can be incurred when applying the standard widening to an increasing sequence of DBSs.

On the other hand, the domain of DBSs is isomorphic to the domain of transitively closed DBMs considered in [19], which suffers from an actual divergence problem. A closer inspection reveals that these two observations are not in contradiction, because the widening operator defined in [19] is not equivalent to the standard widening for convex polyhedra. In fact, a key requirement in the specification of the standard widening is that the first argument is described by a non-redundant system of constraints.[3] Thus, using transitively closed DBMs does not work because they typically contain redundant constraints. What is needed for a correct implementation of the standard widening is a minimization procedure mapping a DBM representation into (any) one of the maximal elements in the corresponding equivalence class: such a procedure was defined in [13] and called *transitive reduction*.

In summary, the solution to the divergence problem for DBSs is to apply the standard widening of [10] to a transitively reduced DBM representation of the first argument. It is worth stressing that, from the point of view of the user, this is a transparent implementation detail: on the domain of DBSs, transitive reduction is the identity function, as was the case for transitive closure.

2.1 On the Precision of the Standard Widening

The standard widening on DBSs could result, if used without any precaution, in poorer precision with respect to its counterpart defined on the syntactic domain

[3] This requirement was sometimes neglected in recent papers describing the standard widening; it was recently recalled and exemplified in [2, 3].

of DBMs. The specification of [14] prescribes, for maximum precision, two constraints on the abstract iteration sequence: the first one restricts the application of the standard widening to a transitively closed representation for the second argument (note that, in this case, no divergence problem can arise); the second one demands that the first DBM of the iteration sequence $M_0, M_1, \ldots, M_i, \ldots$ is transitively closed. The effects of both improvements can be obtained also with the semantic domain of DBSs. As for the first one, this can be applied as is (since transitive closure is just an implementation detail). The other improvement can be achieved by applying the well-known 'widening up to' technique defined in [11, 12] or its variation called 'staged widening with thresholds' [6, 7, 17]: in practice, it is sufficient to add to the set of 'up to' thresholds all the constraints of M_0 that are redundant for the representation of the corresponding DBS (i.e., those constraints that are removed by the transitive reduction algorithm).

Further precision improvements can be obtained by applying any delay strategy and/or the framework defined in [2, 3]. In particular, by providing the widening on DBSs with a finite convergence certificate, it is possible to lift it to a corresponding widening on the *finite powerset* of DBSs [4]. It should be stressed that, in this case, using the syntactic domain of DBMs may have drawbacks: since different DBMs may represent the same DBS, the presence of these "duplicates" in a finite powerset element may have a negative effect on both efficiency and precision (e.g., when considering a *cardinality-based* widening operator). Also note that, in general, the systematic removal of these duplicates would interfere with widenings, possibly compromising the convergence guarantee.

3 Octagonal Shapes and Beyond

The abstract domain of DBMs has been generalized in [15] so as to allow for the manipulation of constraints of the form $ax + by \leq c$, where $a, b \in \{-1, 0, +1\}$, leading to the definition of the *octagon* abstract domain (octagons were called *simple sections* in [5]). Each octagon is represented by using a *coherent* DBM and the transitive closure algorithm is specialized into a *strong closure* procedure. All the previous reasoning can be repeated, leading to the definition of the semantic abstract domain of *octagonal shapes* together with a correct implementation of the standard widening. In this case, the transitive reduction algorithm defined in [13] does not eliminate all redundancies: we will describe a new minimization procedure that takes into account all the constraint inferences performed by the strong closure algorithm.

Other examples of weakly-relational numeric domains include the 'two variables per inequality' abstract domain [20], the octahedron abstract domain [8], and the abstract domain of template constraint matrices [18], as well as the abstract domain of bounded quotients [1] and the zone congruence abstract domain [16]. As long as their implementation is based on (extensions of) the transitive closure algorithm, it is possible to define the corresponding syntactic and semantic versions. The choice between the two versions mainly depends on the availability of a reasonably efficient minimization procedure: in our opinion, all the rest being equal, the semantic versions should be preferred for their greater elegance and the more natural integration with domain constructions such as the finite powerset operator.

References

1. R. Bagnara. *Data-Flow Analysis for Constraint Logic-Based Languages*. PhD thesis, Dipartimento di Informatica, Università di Pisa, Pisa, Italy, March 1997. Printed as Report TD-1/97.
2. R. Bagnara, P. M. Hill, E. Ricci, and E. Zaffanella. Precise widening operators for convex polyhedra. In R. Cousot, editor, *Static Analysis: Proceedings of the 10th International Symposium*, volume 2694 of *Lecture Notes in Computer Science*, pages 337–354, San Diego, California, USA, 2003. Springer-Verlag, Berlin.
3. R. Bagnara, P. M. Hill, E. Ricci, and E. Zaffanella. Precise widening operators for convex polyhedra. *Science of Computer Programming*, 2005. To appear.
4. R. Bagnara, P. M. Hill, and E. Zaffanella. Widening operators for powerset domains. In B. Steffen and G. Levi, editors, *Proceedings of the Fifth International Conference on Verification, Model Checking and Abstract Interpretation (VMCAI 2004)*, volume 2937 of *Lecture Notes in Computer Science*, pages 135–148, Venice, Italy, 2003. Springer-Verlag, Berlin.
5. V. Balasundaram and K. Kennedy. A technique for summarizing data access and its use in parallelism enhancing transformations. In B. Knobe, editor, *Proceedings of the ACM SIGPLAN'89 Conference on Programming Language Design and Implementation (PLDI)*, volume 24(7) of *ACM SIGPLAN Notices*, pages 41–53, Portland, Oregon, USA, 1989. ACM Press.
6. B. Blanchet, P. Cousot, R. Cousot, J. Feret, L. Mauborgne, A. Miné, D. Monniaux, and X. Rival. Design and implementation of a special-purpose static program analyzer for safety-critical real-time embedded software. In T. Æ. Mogensen, D. A. Schmidt, and I. Hal Sudborough, editors, *The Essence of Computation*, volume 2566 of *Lecture Notes in Computer Science*, pages 85–108. Springer-Verlag, Berlin, 2002.
7. B. Blanchet, P. Cousot, R. Cousot, J. Feret, L. Mauborgne, A. Miné, D. Monniaux, and X. Rival. A static analyzer for large safety-critical software. In *Proceedings of the ACM SIGPLAN 2003 Conference on Programming Language Design and Implementation (PLDI'03)*, pages 196–207, San Diego, California, USA, 2003. ACM Press.
8. R. Clarisó and J. Cortadella. The octahedron abstract domain. In R. Giacobazzi, editor, *Static Analysis: Proceedings of the 11th International Symposium*, volume 3148 of *Lecture Notes in Computer Science*, pages 312–327, Verona, Italy, 2004. Springer-Verlag, Berlin.
9. P. Cousot and R. Cousot. Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *Proceedings of the Fourth Annual ACM Symposium on Principles of Programming Languages*, pages 238–252, New York, 1977. ACM Press.
10. P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. In *Conference Record of the Fifth Annual ACM Symposium on Principles of Programming Languages*, pages 84–96, Tucson, Arizona, 1978. ACM Press.

11. N. Halbwachs, Y.-E. Proy, and P. Raymond. Verification of linear hybrid systems by means of convex approximations. In B. Le Charlier, editor, *Static Analysis: Proceedings of the 1st International Symposium*, volume 864 of *Lecture Notes in Computer Science*, pages 223–237, Namur, Belgium, 1994. Springer-Verlag, Berlin.
12. N. Halbwachs, Y.-E. Proy, and P. Roumanoff. Verification of real-time systems using linear relation analysis. *Formal Methods in System Design*, 11(2):157–185, 1997.
13. K. Larsen, F. Larsson, P. Pettersson, and W. Yi. Efficient verification of real-time systems: Compact data structure and state-space reduction. In *Proceedings of the 18th IEEE Real-Time Systems Symposium (RTSS'97)*, pages 14–24, San Francisco, CA, 1997. IEEE Computer Society Press.
14. A. Miné. A new numerical abstract domain based on difference-bound matrices. In O. Danvy and A. Filinski, editors, *Proceedings of the 2nd Symposium on Programs as Data Objects (PADO 2001)*, volume 2053 of *Lecture Notes in Computer Science*, pages 155–172, Aarhus, Denmark, 2001. Springer-Verlag, Berlin.
15. A. Miné. The octagon abstract domain. In *Proceedings of the Eighth Working Conference on Reverse Engineering (WCRE'01)*, pages 310–319, Stuttgart, Germany, 2001. IEEE Computer Society Press.
16. A. Miné. A few graph-based relational numerical abstract domains. In M. V. Hermenegildo and G. Puebla, editors, *Static Analysis: Proceedings of the 9th International Symposium*, volume 2477 of *Lecture Notes in Computer Science*, pages 117–132, Madrid, Spain, 2002. Springer-Verlag, Berlin.
17. A. Miné. Relational abstract domains for the detection of floating-point run-time errors. In D. Schmidt, editor, *Programming Languages and Systema: Proceedings of the 13th European Symposium on Programming*, volume 2986 of *Lecture Notes in Computer Science*, pages 3–17, Barcelona, Spain, 2004. Springer-Verlag, Berlin.
18. S. Sankaranarayanan, H. Sipma, and Z. Manna. Scalable analysis of linear systems using mathematical programming. In R. Cousot, editor, *Proceedings of the Sixth International Conference on Verification, Model Checking and Abstract Interpretation (VMCAI 2005)*, volume 3385 of *Lecture Notes in Computer Science*, Paris, France, 2004. Springer-Verlag, Berlin. To appear.
19. R. Shaham, E. K. Kolodner, and S. Sagiv. Automatic removal of array memory leaks in Java. In D. A. Watt, editor, *Proceedings of the 9th International Conference on Compiler Construction (CC 2000)*, volume 1781 of *Lecture Notes in Computer Science*, pages 50–66. Springer-Verlag, Berlin, 2000.
20. A. Simon, A. King, and J. M. Howe. Two variables per linear inequality as an abstract domain. In M. Leuschel, editor, *Logic Based Program Synthesis and Tranformation, 12th International Workshop*, volume 2664 of *Lecture Notes in Computer Science*, pages 71–89, Madrid, Spain, 2002. Springer-Verlag, Berlin.