Strategy analysis: from sequential to parallel strategies

(Position paper)

Maria Paola Bonacina ⋆

Dept. of Computer Science – University of Iowa – Iowa City, IA 52242-1419, USA
E-mail: bonacina@cs.uiowa.edu

The growth of methods for mechanical theorem proving poses the problem of evaluating their efficiency. The evaluation of an approach for the automated solution of a problem is made usually of at least two components: algorithm analysis (machine-independent), and performance evaluation (machine-dependent). In the tradition of theorem proving, only the second component has been available. Research in *strategy analysis* aims at reducing this hole, by providing mathematical tools for some machine-independent evaluation of theorem-proving strategies.

In earlier work [4] (also presented in [1]), an approach which begins to address issues in strategy evaluation was proposed. It comprises *a model for the representation of the search space and search process*, a notion of *complexity of search* in infinite spaces, and a measure of this complexity, termed *bounded search spaces*. These tools were applied to *contraction-based strategies*, that is, forward-reasoning strategies with eager contraction (those originated from the term-rewriting and Knuth-Bendix paradigm on one hand, the resolution-paramodulation paradigm on the other). Strategies of different contraction power were compared, showing that a strategy with higher contraction power causes a larger reduction of the bounded search spaces during the derivation. This was the first theoretical result to justify in analytical terms the advantage of contraction observed in experiments.

In recent work [2, 3], this methodology was extended from sequential search to *parallel search*. Parallel search means that deductive processes search in parallel the space of the problem: each executes a strategy, develops a derivation, builds a set of data, and communicates with the others; the parallel search succeeds as soon as one of the processes does. Approaches to parallel search differ in how they differentiate and combine the activities of the processes. In *distributed search*, the strategy *subdivides the search space* among the processes, with communication to preserve completeness: *distributed-search contraction-based strategies* are the object of the analysis.

In a strategy with distributed search the modelling problem becomes more complicated, because not only contraction, but also *subdivision* and *communication* modify the search space. Furthermore, many processes are active in parallel. In a *parallel marked search graph* [2], the structure of the graph represents the search space of all the possible inferences, while the *marking of vertices and arcs* (one per process) represents the search, including: selections by the search plan,

⋆ Supported in part by the National Science Foundation with grant CCR-97-01508.

deletions by contraction, communication (clauses may be received in addition to being generated), and subdivision (an arc may be *allowed* or *forbidden* to a process by the *subdivision function* of the search plan).

The next objective was to refine the complexity measure in such a way to capture both benefits (e.g., the subdivision) and costs (e.g., communication) of distributed search. A clause occurs in the bounded search space of bound j with multiplicity equal to the number of its *ancestor-graphs* whose *distance* is smaller or equal to j. Clauses made unreachable by contraction have infinite distance, and are excluded from all bounded search spaces. In [3], the notion of *ancestor-graph* is refined to distinguish between *allowed* and *forbidden* ancestor-graphs, in such a way to take both subdivision (which forbids ancestor-graphs) and communication (which may re-allow them) into account. The *bounded search spaces for process* p_k only count ancestor-graphs allowed to p_k, so that they measure both the advantage of subdivision and the cost of communication.

These tools were applied to identify and analyze issues in the parallelization of contraction-based strategies. It is not obvious that the parallelization of a contraction-based strategy is contraction-based, because of the interactions of eager contraction with subdivision and communication. Once sufficient conditions for a parallel strategy to be contraction-based had been identified, the impact of communication delays on eager contraction was analyzed: two patterns of behaviour – *late contraction* and *contraction undone* – where communication delays cause eager contraction to fail, were discovered. While in a sequential derivation the bounded search spaces decrease monotonically (expansion visits the search space, contraction prunes it), in a parallel derivation they may oscillate *non-monotonically*, because contraction and subdivision reduce them, whereas communication undoes some of the subdivision and causes failures of eager contraction. A strength of this analysis is to capture the cost of communication in terms of bounded search spaces, without the possibility to recur to the classical notion of time complexity. The next step was to consider all the parallel processes together, by introducing *parallel bounded search spaces*. If different processes are allowed to visit the same ancestor-graphs, there is an *overlap*, which may be caused by imperfect subdivision or communication. Sufficient conditions to minimize the overlap in the presence of local eager contraction were given.

The last phase was to compare a contraction-based strategy $\mathcal{C}$ with its parallelization $\mathcal{C}'$. The analysis of the evolution of their bounded search spaces shows that: all redundant clauses excluded by $\mathcal{C}$ are excluded also by $\mathcal{C}'$; if $\mathcal{C}'$ has instantaneous communication and minimizes the overlap, its parallel bounded search spaces are smaller than those of $\mathcal{C}$. This result is significant as a limit theorem, similar to other theoretical results obtained under an ideal assumption: it explains the nature of the problem, indicating in overlap and communication-contraction node its essential aspects, and it represents a limit that concrete strategies may approximate. Since instantaneous communication is necessary, it is also a negative result on the parallelizability of contraction-based strategies, which contributes to explain – analytically, not empirically – the difficulty with obtaining generalized improvements by parallelism in theorem proving.

References

1. M. P. Bonacina. Machine-independent evaluation of theorem-proving strategies (position paper). In B. Gramlich and H. Kirchner, editors, *Proc. of the Workshop on Strategies in Automated Deduction, CADE-14*, pages 37–39, 1997.
2. M. P. Bonacina. On the representation of parallel search in theorem proving (extended abstract). In *Proc. of FTP-97*, number 97-50 in Technical Report of RISC, pages 22–28. Johannes Kepler Universität, 1997.
3. M. P. Bonacina. Distributed contraction-based strategies: model and analysis. Technical report, Department of Computer Science, University of Iowa, April 1998.
4. M. P. Bonacina and J. Hsiang. On the modelling of search in theorem proving – towards a theory of strategy analysis. *Information and Computation*, forthcoming, 1998.