

Mechanizing Metatheory Without Typing Contexts

Jonghyun Park · Jeongbong Seo ·
Sungwoo Park · Gyesik Lee

Received: 9 August 2011 / Accepted: 2 April 2013
© Springer Science+Business Media Dordrecht 2013

Abstract When mechanizing the metatheory of a programming language, one usually needs many lemmas proving structural properties of typing judgments, such as permutation and weakening. Such structural lemmas are sometimes unnecessary if we eliminate typing contexts by expanding typing judgments into their original hypothetical proofs. This technique of eliminating typing contexts, which has been around since Church (J Symb Log 5(2):56–68, 1940), is based on the view that entailment relations, such as typing judgments, are just syntactic tools for displaying only the hypotheses and conclusion of a hypothetical proof while hiding its internal structure. In this paper, we apply this technique to parts 1A/2A of the POPLMARK challenge (Aydemir et al. 2005) and experimentally evaluate its efficiency by formalizing System $F_{<}$ in the Coq proof assistant in a number of different ways. An analysis of our Coq developments shows that eliminating typing contexts produces a more significant reduction in both the number of lemmas and the count of tactics than the cofinite quantification, one of the most effective ways of simplifying the mechanization involving binders. Our experiment with System $F_{<}$ suggests three guidelines to follow when applying the technique of eliminating typing contexts.

Keywords Typing context · POPLMARK · System $F_{<}$

J. Park · J. Seo · S. Park (✉)
Pohang University of Science and Technology (POSTECH), Gyeongsangbuk, Korea
e-mail: gla@postech.ac.kr

J. Park
e-mail: parjong@postech.ac.kr

J. Seo
e-mail: baramseo@postech.ac.kr

G. Lee
Hankyong National University, Gyeonggi-do, Korea
e-mail: gslee@hknu.ac.kr