

Contents

1	Automated Proof Construction in Type Theory using Resolution	1
1.1	Introduction	1
1.2	A Two-level Approach	3
1.3	Clausification and Correctness	6
1.3.1	Negation Normal Form	6
1.3.2	Skolemisation	7
1.3.3	Composing the Modules	9
1.4	Minimal Resolution Logic	10
1.5	Lifting to Predicate Logic	15
1.6	Examples	16
1.6.1	A small example	16
1.6.2	A medium scale example: Newman’s Lemma	17
2	Proof Reflection in Coq	23
2.1	Introduction	23
2.2	Objects	24
2.3	Recursive Patterns	27
2.4	Lifting and Substitution	29
2.5	Properties of De Bruijn Operations	32
2.6	Judgements	33
2.7	Admissible Rules	36
2.8	Translation to Coq’s Native Logic	37
2.9	Free Variables	39
2.9.1	Free Variables, Finitely versus Infinitely Many	39
2.9.2	Free Variables and Substitution	39
2.10	Thinning and Substitution Lemmas	40
2.11	Soundness with respect to the Native Logic	40
2.12	Type Checking Function	42
2.13	Correctness of Type Checking Function	42
2.14	Unique Types	43
2.15	Proof Reduction	43
2.16	Subject Reduction	45
2.17	Conclusion and Future Research	47

3	λ	49
3.1	Introduction	49
3.2	Preliminaries	54
3.3	α	55
	3.3.1 $\lambda\alpha$	55
	3.3.2 $\lambda\alpha$	58
3.4	β	64
	3.4.1 $\lambda\beta$	64
	3.4.2 $\lambda\beta$	65
	3.4.3 α and β	74
	3.4.4 Confluence of λ -calculus	76
3.5	Applications	81
3.6	Conclusion and Discussion	82