

Contents

List of Figures	v
1 Introduction	1
2 Background	4
2.1 Type Systems	4
2.1.1 Syntax and Static Semantics	5
2.1.2 Dynamic Semantics	6
2.1.3 Soundness	7
2.1.4 Using a Simple Type System for <code>saveName</code>	8
2.2 Type and Effect Systems	9
2.2.1 Syntax and Static Semantics	9
2.2.2 Dynamic semantics	11
2.2.3 Soundness	11
2.2.4 Using a Type and Effect System for <code>saveName</code>	12
2.3 Summary	13
3 Lambda May-Must	14
3.1 Static Semantics	14
3.1.1 Syntax	14
3.1.2 Effect Set Operations	15
3.1.3 Well Formedness	16
3.1.4 Subtyping	17
3.1.5 Typing Rules	18
3.1.6 Minimum Effect Set	21
3.2 Dynamic Semantics	23
3.3 Properties	25
3.4 Summary	26
4 Lambda May-Must With Effect Polymorphism	27
4.1 Static Semantics	27
4.1.1 Syntax	28
4.1.2 Effect Sets	29
4.1.3 Well formedness	30
4.1.4 Subtyping	31
4.1.5 Typing Rules	31
4.1.6 Minimum Effect Set	33

4.2	Dynamic Semantics	34
4.3	Properties	34
4.4	Summary	37
5	Related Work	38
5.1	Linear Haskell	38
5.2	Idris	39
5.3	First-Class Effects	40
5.4	Summary	41
6	Conclusions	42
6.1	Contributions	42
6.2	Limitations and Future Work	43
	Bibliography	44
	List of Lemmas, Propositions, Theorems, and Definitions	47
	A Proof	48