Contents

Abstract **xi**

Resumen **xiii**

Resum **xv**

List of Figures **xvii**

List of Tables **xxiii**

1 Introduction **1**
1.1 Preliminary Considerations and Background . . . 2
1.2 Remote GPU Virtualization . . . 3
1.3 Objectives and Contributions of the Thesis . . . 6
1.4 Technology Transfer: An Industry-driven Thesis . . . 7
1.5 Thesis Outline . . . 9

2 Related Work **11**
2.1 Existing Remote GPU Virtualization Solutions . . . 12
2.2 rCUDA: remote CUDA . . . 14
2.3 Bandwidth Comparison . . . 15
2.4 Summary . . . 19

3 Improving the user experience of rCUDA **21**
3.1 Background . . . 22
3.2 CU2rCU: A CUDA-to-rCUDA Converter . . . 24
3.2.1 Need for a CUDA-to-rCUDA Converter . . . 24
3.2.2 Implementing the `CU2rCU` Converter . . . 27
3.2.2.1 Kernel Calls . . . 28
3.2.2.2 Kernel Names . . . 28
3.2.2.3 CUDA Symbols . . . 29
3.2.2.4 Textures and Surfaces . . . 29
3.2.3 Evaluation of the `CU2rCU` converter . . . 30
3.3 Performance evaluation . . . 33
3.4 Support for multithreaded applications and CUDA libraries . . . 38
3.4.1 Support for Multithreaded Applications . . . 38
3.4.2 Support for CUDA Libraries . . . 43
3.5 Summary . . . 44

4 Tuning rCUDA for InfiniBand Networks 45
4.1 Introduction . . . 46
4.2 Influence of FDR InfiniBand on the Performance of rCUDA . . . 47
4.2.1 Basic Performance Comparison of the Networks . . . 48
4.2.1.1 Testbed System . . . 48
4.2.1.2 Influence on Bandwidth . . . 49
4.2.1.3 Influence on Latency . . . 50
4.2.2 NVIDIA CUDA Samples . . . 51
4.2.3 Influence of FDR InfiniBand on Production Applications . . . 54
4.2.3.1 CUDASW++ . . . 54
4.2.3.2 GPU-BLAST . . . 56
4.2.3.3 LAMMPS . . . 58
4.2.4 Summary . . . 60
4.3 Enhancing rCUDA with Support for InfiniBand Dual-port Adapters . . . 61
4.3.1 Adding Dual-port Support to rCUDA . . . 62
4.3.2 Impact of Connect-IB on Remote GPU Usage . . . 62
4.3.2.1 From Theoretical to Real Bandwidth and Latency Figures 63
4.3.2.2 Influence on the Bandwidth of rCUDA . . . 67
4.3.2.3 Analyzing the Bandwidth of rCUDA . . . 69
4.3.3 Summary . . . 71
4.4 InfiniBand Verbs Optimizations for Remote GPU Virtualization . . . 71
4.4.1 Related Work . . . 72
4.4.2 Bandwidth Optimizations . . . 73
4.4.2.1 Number of Queue Pairs per Port . . . 74
4.4.2.2 Capacity of Send/Receive Queues . . . 76
4.4.2.3 Combining both Optimizations . . . 78
4.4.3 Experiments . . . 80
4.4.3.1 Impact of the Optimizations on rCUDA . . . 80
4.4.3.2 Impact of the Optimizations on Applications using rCUDA with Single-port Adapters . . . 82
4.4.3.3 Impact of the Optimizations on Applications using rCUDA with Dual-port Adapters . . . 84
4.4.4 Summary . . . 89
4.5 Influence of EDR InfiniBand on the Bandwidth of rCUDA . . . 90
4.5.1 Background . . . 90
4.5.2 Motivation . . . 91
4.5.3 Experiments . . . 95
4.5.3.1 Bandwidth analysis . . . 95
4.5.3.2 The Rodinia Benchmark Suite . . . 95
4.5.3.3 Production Applications . . . 99
4.5.4 Summary . . . 101
4.6 Conclusions . . . 101

5 Peer to Peer Memory Copies between Remote GPUs 103
5.1 Introduction . . . 104
5.2 Related Work on P2P Memory Copies . . . 106
5.3 Implementing Efficient P2P Memory Copies within rCUDA . . . 109

5.3.1 Version 1: Using GPUDirect RDMA . . . 110
5.3.2 Version 2: Pre-allocating Intermediate Buffers . . . 111
5.3.3 Version 3: Using Multiple Intermediate Buffers . . . 114
5.3.4 Version 4: Adaptive Intermediate Buffer Size . . . 115
5.3.5 Latency Analysis . . . 119
5.3.6 Final Version: Hybrid Approach . . . 121
5.4 Experiments with a Real Application . . . 122
5.5 Conclusions . . . 129

6 schedGPU: Fine-Grain Dynamic and Adaptive Scheduling for GPUs 131
6.1 Introduction . . . 132
6.2 Related Work . . . 134
6.3 GPU Scheduling Framework . . . 136
6.4 Implementation Approaches . . . 138
6.4.1 Client-Server . . . 138
6.4.2 Shared Memory . . . 139
6.4.2.1 Shared Memory Data . . . 140
6.4.2.2 Synchronizing Access to Shared Memory . . . 141
6.5 The Life Cycle . . . 142
6.6 Notification Policies . . . 145
6.7 Experimental Setup and Use-Cases . . . 147
6.7.1 Hardware Platform . . . 147
6.7.2 Use-cases . . . 148
6.8 Evaluation . . . 150
6.8.1 Overhead of the approaches . . . 150
6.8.2 Performance Gain . . . 151
6.8.2.1 Concurrent Execution of Individual Applications . . . 151
6.8.2.2 Workloads Comprising Multiple Applications . . . 156
6.8.3 Evaluation Summary . . . 160
6.9 Conclusions . . . 161

7 Conclusions 163
7.1 Contributions . . . 164
7.2 Publications . . . 164
7.3 Future Directions . . . 171

References 173