Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence

Optimization of Heterogeneous Computing Resources for Robotic Mapping

Adrian Ratter
ARC Centre of Excellence in Autonomous Systems
School of Computer Science and Engineering, UNSW
Sydney, NSW, 2052, Australia.
adrianr@cse.unsw.edu.au

Research Area

Robotic platforms have large computational demands, with many algorithms requiring extensive computing resources to run. This is at odds with the real time, fast response and low power usage requirements on mobile platforms that often have limited computing power. As a consequence, achieving optimal usage of these limited resources is important. With the increasing ability of Graphical Processing Units (GPUs) and Field Programmable Gate Arrays (FPGAs) to deliver a high performance by using parallel approaches, effectively utilizing this hardware in combination with a CPU has enormous potential to achieve these conflicting requirements.

The efficient use of computing resources on a heterogeneous robotics platform, both in terms of run time performance and power usage, presents an interesting research problem, and is the focus of my research. It is envisaged that this will be achieved by both finding parallel approaches to algorithms commonly used in robotics, and investigating the use of a scheduler to efficiently allocate resources across a heterogeneous hardware platform. In particular, while there has been much research on using specialized hardware for image and video processing algorithms, work on areas specific to robotics, such as position tracking, mapping and sensor fusion, is not as common.

Position tracking is an important problem in robotics, as many algorithms, such as Simultaneous Localization and Mapping (SLAM) require accurate position update information. However, in many situations, such as in environments with loose terrain, motor encoder odometry can be highly unreliable. As an alternative, Iterative Closest Point (ICP) algorithms, which work by aligning point clouds from range sensors, are commonly used, but they can be computationally expensive. Prominent examples from literature are (Milstein et al. 2011b) and (Kohlbrecher et al. 2011), both of which use occupancy grids to accelerate the closest point search, but still consume a large amount of computational resources. Taking position tracking a step further into a SLAM solution with loop closing presents many additional difficulties, with the problem sometimes avoided by moving much of the computation off board (Milstein et al. 2011a). However, having a full on board SLAM solution, such as graph SLAM (Grisetti et al. 2007), would be highly desirable.

One of the main related examples in literature, Kinect Fusion (Newcombe et al. 2011), uses ICP on a GPU to create a surface map of an area. While this does generate position tracking information, the computational overheads of the surface reconstruction are not needed for robotic position tracking. This work does however highlight the difficulties of adapting algorithms for use on devices such as GPUs, with their performance greatly reducing with branching threads, random memory access and data access synchronisation (Shams and Barnes 2007).

Once multiple algorithms have been adapted to run efficiently on parallel hardware, their scheduling of tasks across heterogeneous hardware, consisting of a multi-core CPU, GPU and FPGA, becomes an interesting research problem. Initial research in this area has focused on scheduling across a GPU and CPU, with (Gregg et al. 2011) using historical runtime data, and (Grewe and OBoyle 2011) using machine learning of code features as input into a static scheduler. A difficulty with scheduling algorithms across heterogeneous hardware is the vastly different requirements for efficient performance on each type of hardware, with the Twin Peaks platform (Gummaraju et al. 2010) being one of the first to investigate efficient mapping of GPU code to a CPU. Recent developments in allowing OpenCL to run on FPGAs (Czajkowski et al. 2012) has meant the one programming environment can be used on all devices, raising the possibility of using a similar idea to Twin Peaks across all three types of devices. In addition to algorithm efficiency, power usage is an important consideration for mobile robots. (Luk, Hong, and Kim 2009) showed power usage can be reduced by splitting work across GPU and CPU.

Research Questions and Anticipated Contributions

The major research questions and potential contributions of my thesis are:

- New parallel approaches to robotics algorithms, particularly in the areas of position tracking and SLAM. This involves designing a highly parallel algorithm to achieve encoder free, real-time, position tracking and graph based SLAM loop closing.

- A heterogeneous scheduler designed for mobile robotic platforms. The scheduler should be adapted to different hardware configurations, and should allocate algorithms across heterogeneous hardware in such a way to maximize the efficiency of the system, and to minimize power usage as required.
- A greater understanding of best compromise in hardware to use in heavily constrained environment of a mobile robot, taking into account considerations such as power usage, processing performance and weight.

Completed Work

The major section of work towards my thesis that has been completed so far is a highly parallel occupancy grid iterative closest point position tracking algorithm designed for use on a GPU. The high level of interaction between parallel calculations and the need for hundreds of threads to add scans to the occupancy grid in parallel without causing data corruption make this a complex task. In particular, a major component of this work was in designing a hybrid occupancy grid/active cells list data structure that can be rapidly shifted as the robot moves and be safely modified in parallel by hundreds of threads, while retaining the advantages of occupancy grids in being able to quickly find the closest points required by the ICP algorithm. This work also involved finding efficient parallel algorithms for a synchronisation free local search to find the closest points and for estimating the map gradient around an ICP match to be used as an input into a Kalman Filter.

To test my approach, I used this algorithm to adapt OG-MBICP (Milstein et al. 2011b) for use on a commodity GPU. It was successful in significantly reducing the runtime from an average of 33ms to just 3ms. The accuracy of the algorithm also increased due to decreasing the number of skipped frames, and it scaled well with the number of data points and map size. This work has been submitted to the 2013 Robotics Science and Systems conference.

Current and Future Work

The focus of my current research is extended my parallel position tracking algorithm to consist of a full loop closing SLAM solution. This is designed to provide a totally encoder free method of position tracking and mapping in real time on a small resource limited robot. My current approach involves taking local snapshots from the occupancy grid position tracking algorithm, and using them as vertices in a graph, with the edges consisting of the translational and rotational distances between the local snapshots. Most of these edges are gained from the position tracking algorithm, with potential loop closing edges found by examing the correlation of the histograms of the local scans. The resulting graph can then be optimized in a similar manner to (Grisetti et al. 2007). It is anticipated that this work will be finished by the 2013 AAAI Doctoral Consortium.

My future work consists of investigating the use of a heterogeneous scheduler on the Robot Operating System (ROS) framework. This will involves studying the power usage of candidate FPGAs, CPUs and GPUs, investigating efficiency requirements for algorithms running using OpenCL on FPGAs, and researching ways for a scheduler to consider the characteristics of an algorithm against the characteristics of potential hardware to allow it to efficiently allocate algorithms across heterogeneous hardware. Another important consideration for the scheduler is data locality, as moving data structures between devices is potentially expensive. It is intended that this work will take the rest of the research time available for my thesis, and will hopefully provide an insight into a good compromise of hardware for use in the heavily constrained environment of a mobile robot.

References

Czajkowski, T.; Aydonat, U.; Denisenko, D.; et al. 2012. From opencl to high-performance hardware on fpgas. In *Field Programmable Logic and Applications (FPL)*, 531–534. IEEE.

Gregg, C.; Boyer, M.; Hazelwood, K.; and Skadron, K. 2011. Dynamic heterogeneous scheduling decisions using historical runtime data. In *Applications for multi-and many-core processors*.

Grewe, D., and OBoyle, M. 2011. A static task partitioning approach for heterogeneous systems using opencl. In *Compiler Construction*, 286–305. Springer.

Grisetti, G.; Stachniss, C.; Grzonka, S.; and Burgard, W. 2007. A tree parameterization for efficiently computing maximum likelihood maps using gradient descent. In *Proc. of robotics: science and systems (RSS)*.

Gummaraju, J.; Morichetti, L.; Houston, M.; Sander, B.; Gaster, B.; and Zheng, B. 2010. Twin peaks: a software platform for heterogeneous computing on general-purpose and graphics processors. In *Parallel architectures and compilation techniques*, 205–216. ACM.

Kohlbrecher, S.; von Stryk, O.; Meyer, J.; and Klingauf, U. 2011. A flexible and scalable slam system with full 3d motion estimation. In *International Symposium on Safety, Security, and Rescue Robotics*, 155–160. IEEE.

Luk, C.; Hong, S.; and Kim, H. 2009. Qilin: exploiting parallelism on heterogeneous multiprocessors with adaptive mapping. In *Proceedings of the 42nd Annual IEEE/ACM International Symposium on Microarchitecture*, 45–55. ACM.

Milstein, A.; McGill, M.; Wiley, T.; Salleh, R.; and Sammut, C. 2011a. A method for fast encoder-free mapping in unstructured environments. *Journal of Field Robotics* 28(6):817–831.

Milstein, A.; McGill, M.; Wiley, T.; Salleh, R.; and Sammut, C. 2011b. Occupancy voxel metric based iterative closest point for position tracking in 3d environments. In *ICRA*, 4048–4053. IEEE.

Newcombe, R.; Davison, A.; Izadi, S.; Kohli, P.; Hilliges, O.; et al. 2011. Kinectfusion: Real-time dense surface mapping and tracking. In *Symposium on Mixed and Augmented Reality*, 127–136. IEEE.

Shams, R., and Barnes, N. 2007. Speeding up mutual information computation using nvidia cuda hardware. In *Digital Image Computing Techniques and Applications*, 555–560. IEEE.