Acceleration of a Pseudo-Bacterial Potential Field Algorithm for Path Planning

Ulises Orozco-Rosas[1], Oscar Montiel[1], Roberto Sepúlveda[1], Antonio S. Montemayor[2], and Juan J. Pantrigo[2]
uorozco@citedi.mx, {oross, rsepulvedac}@ipn.mx, {antonio.sanz, juanjose.pantrigo}@urjc.es
[1]Instituto Politécnico Nacional (CITEDI-IPN), [2]Universidad Rey Juan Carlos (NVIDIA-GPU Education Center)

Motivation

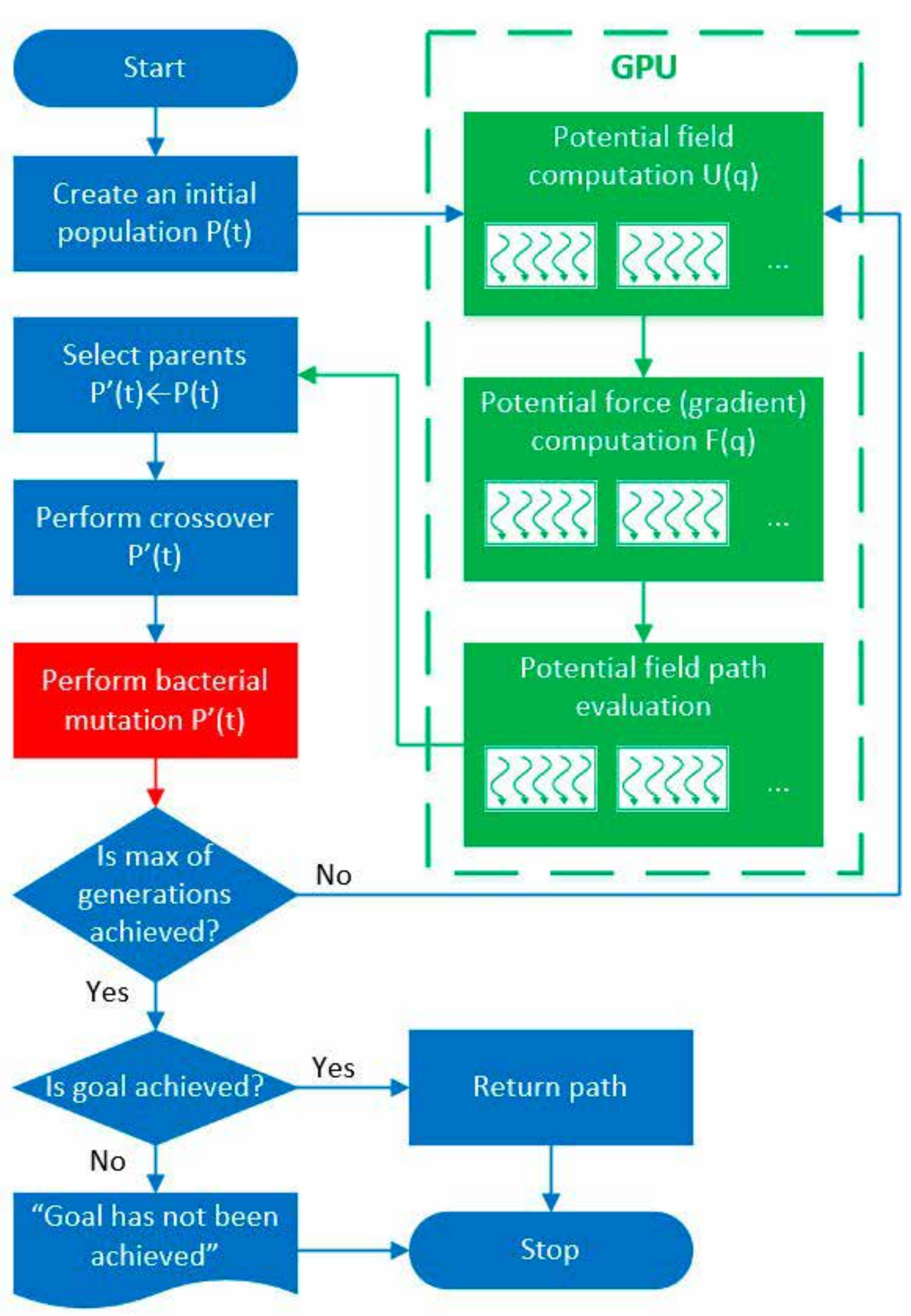
Robotics is one of the most important technologies since it is a fundamental part in automation and manufacturing process. In particular, there is an increasing demand of autonomous mobile robots in various field of application.

This work addresses the problem of autonomous navigation of a mobile robot, to take it from position A to position B without the assistance of a human operator. In particular, **planning** a reachable set of mobile robot configurations (path) to accomplish its mission.

Path planning of a mobile robot is one of the most computationally intensive tasks and a challenge in dynamically changing environments. In particular, path planning is to determine an **optimal path** from a universe of possible solutions (paths). By means of GPUs it is possible to process data-intensive tasks (path planning) efficiently.

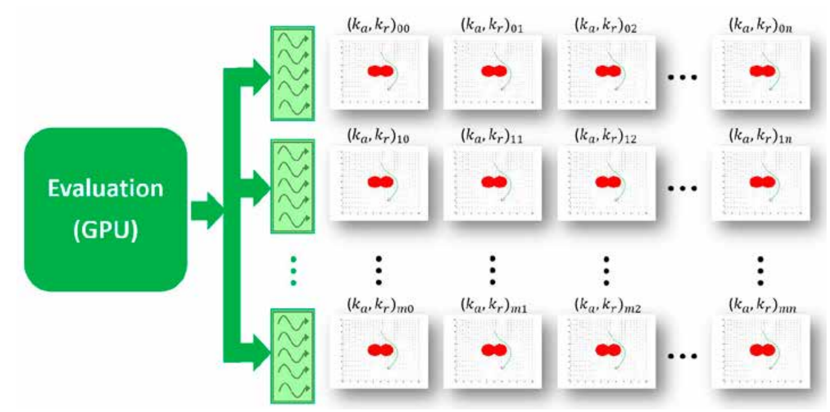
Proposal

In this work we propose the pseudo-bacterial potential field (PBPF) algorithm on a Matlab-CUDA implementation to find an optimal collision free path for a mobile robot and speed up the path planning computation through the use of GPUs.



The PBPF algorithm makes use of the artificial potential field [1] (APF) approach and mathematical programming, using a metaheuristic based on a pseudo-bacterial genetic algorithm [2] (PBGA) as the global optimization method, and parallel computing techniques, to solve efficiently a robot motion problem, in this case the path planning.

The flowchart (left) shows a simplified version of the PBPF [3] algorithm. This PBPF algorithm derives optimal potential field functions using the PBGA (on blue in the flowchart) to perform the path planning. The PBGA introduces the bacterial mutation operator (on red in the flowchart) as an advantage over other evolutionary algorithms. The bacterial mutation operator can spread the characteristics of a single individual to the rest of the population, this process is useful where there is a weak relationship between the parameters of a system [2].

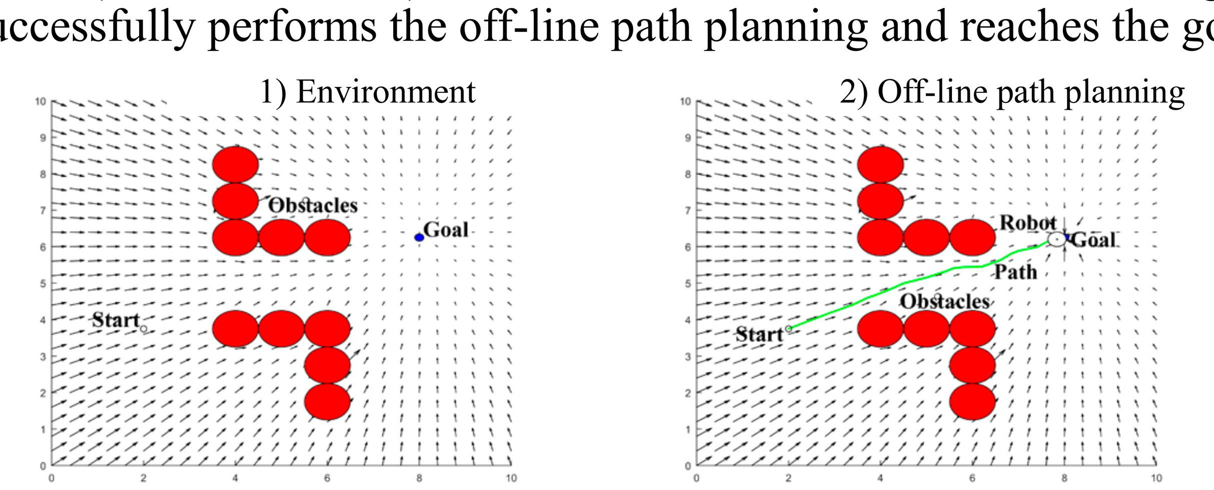


The backbone of this proposal is the use of PBPF algorithm and the parallel evaluation process on GPU (on green in the flowchart and the above figure) to find dynamically the optimal potential field functions to perform the path planning. This allows that the mobile robot navigates without being trapped in local minima, making the PBPF algorithm suitable for off-line and on-line path planning.

Path Planning Results

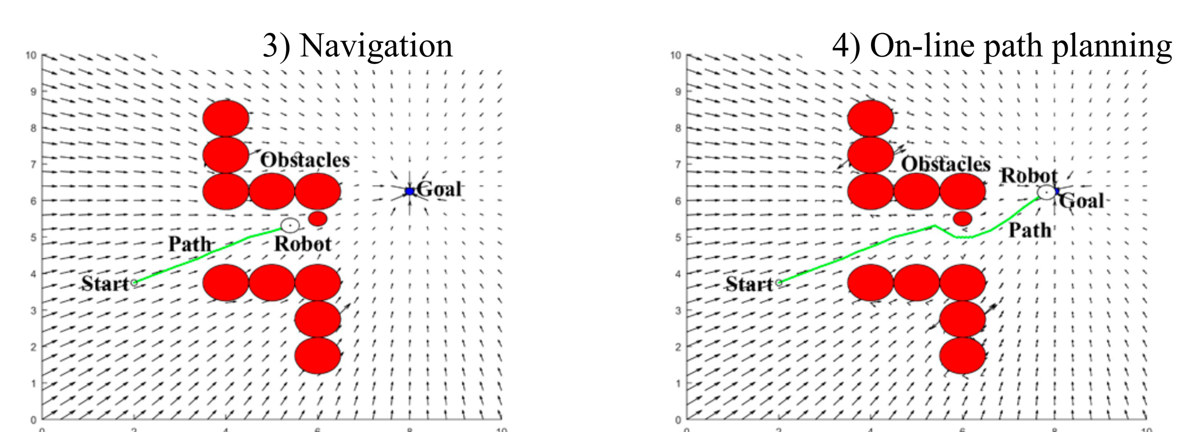
1) The experiment to test the PBPF algorithm consists on a two dimensions map, divided with a grid of 10x10 meters and containing two blocks of obstacles to simulate a real-world corridor (navigation environment).

2) We perform the off-line path planning test, where the environment is totally known (described above). Below we can observe how the PBPF algorithm successfully performs the off-line path planning and reaches the goal.



3) The mobile robot can perform the navigation following the previous path planned (off-line), but most of the times the environment is changing, i.e., new obstacles added or closed ways are presented.

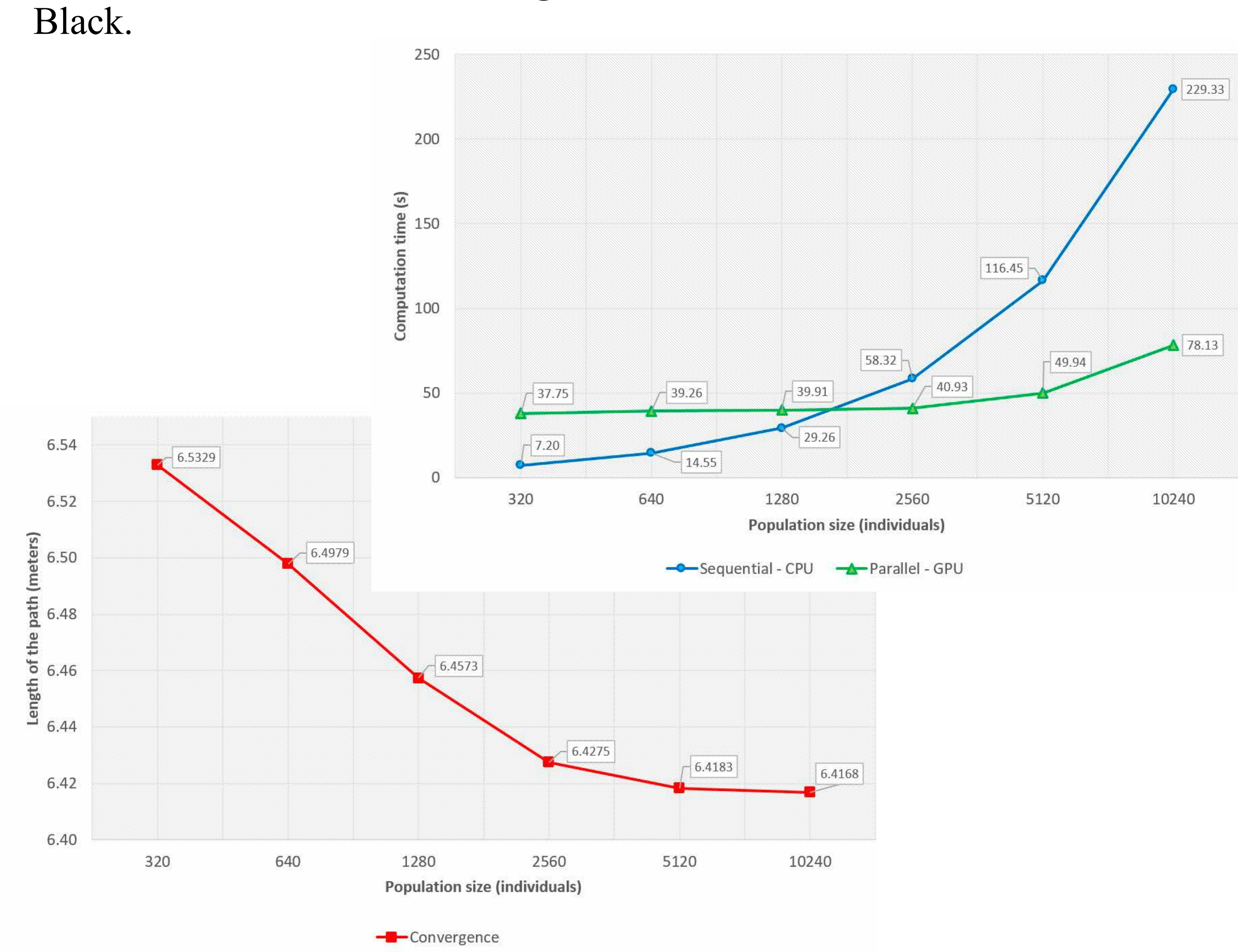
4) So the on-line path planning is performed to take action on the changes of the environment and the path is recalculated to reach the goal. Below we can observe how the PBPF successfully performs the on-line path planning.



Performance Results

To compare the CPU sequential implementation versus GPU parallel implementation we have carried out thirty tests for each population size with the aim to record the execution times. The convergence plot and the average times for each execution are shown in the next graphs. The implementation settings were:

- Software: NVIDIA CUDA 7.0; Matlab R2015A; Ubuntu 14.04.
- Hardware: Intel i7-4770 CPU @ 3.40GHz; NVIDIA GeForce GTX TITAN Black.



Conclusion

- In this work, we have presented the PBPF algorithm for path planning on a Matlab-CUDA implementation.
- The parallelization of the PBPF evaluation is one of the strongest contributions of this proposal. Other major contribution of this work is the capability of population sizing, in the performance results we observed that a "small" population size could guide the PBPF algorithm to "moderate" solutions and that a "large" population size could make the PBPF algorithm spends "more" computation time to find the "best" solution.
- The results demonstrate that the GPU implementation is very powerful to accelerate the evaluation of the PBPF solutions.

References & Acknowledgments

[1]O. Khatib.: Real-time obstacle avoidance for manipulators and mobile robots. The International Journal of Robotics Research, pages 90-98, 1986.
[2] N.E. Nawa, T. Hashiyama, T. Furuhashi, and Y. Uchikawa.: A study on fuzzy rules discovery using Pseudo-Bacterial Genetic Algorithm with adaptive operator. IEEE - Evolutionary Computation, pages 589-593, 1997.
[3] U. Orozco-Rosas, O. Montiel, and R. Sepúlveda.: Pseudo-bacterial Potential Field Based Path Planner for Autonomous Mobile Robot Navigation. International Journal of Advanced Robotic Systems, 12(81): 1-14, 2015.

We thank to the Instituto Politécnico Nacional (National Polytechnic Institute), to the Comisión de Fomento y Apoyo Académico del IPN (COFAA), and to the Mexican National Council of Science and Technology (CONACYT) for supporting our research activities.