Supplemental: Quantum Ray Marching for Reformulating Light Transport Simulation

Logan Mosier
University of Waterloo

Morgan McGuire
Roblox, University of Waterloo

Toshiya Hachisuka
University of Waterloo

ACM Reference Format:
Logan Mosier, Morgan McGuire, and Toshiya Hachisuka. 2023. Supplemental: Quantum Ray Marching for Reformulating Light Transport Simulation. In *SIGGRAPH Asia 2023 Conference Papers (SA Conference Papers '23), December 12–15, 2023, Sydney, NSW, Australia.* ACM, New York, NY, USA, 1 page. https://doi.org/10.1145/3610548.3618151

1 PROOF OF EXPONENTIAL PATHS

To illustrate how our method can evaluate an exponential number of paths relative to the number of steps in the walk, we apply our method to evaluate the integral along some function $f(x)$ on a $1D$ line. We choose to evaluate our method this way as this setting allows for analysis of the paths traced by the walk compared to a more complicated domain of light paths in a $3D$ scene.

To analyze the paths of the walk we will be following the work of Joshi et al. [2018]. Beginning with a initial state

$$|\psi_0\rangle = |x_0\rangle \otimes (\alpha |\uparrow\rangle + e^{i\phi}\beta |\downarrow\rangle) \bigotimes^{s} |0\rangle \tag{1}$$

Where x_0 is the initial position of the walker. The direction register is set to $(\alpha |\uparrow\rangle + e^{i\phi}\beta |\downarrow\rangle)$, with α and β being real values, where $|\uparrow\rangle$ implies heading to the right and $|\downarrow\rangle$ is to the left. The remaining qubits will be used to store the samples along the walk. In the full rendering approach, $2s$ qubits would be needed as we sample 2 values at each step, but only 1 sample is needed for this task. Evolving the system using a coin of $C = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ results in

$$|\psi_1\rangle = SCM |\psi_0\rangle = (\alpha |x_1\rangle |\uparrow\rangle |s_0\rangle - e^{i\phi}\beta |x_{-1}\rangle |\downarrow\rangle |s_0\rangle) \bigotimes^{s-1} |0\rangle \tag{2}$$

Joshi et al. [2018] show that for a $1D$ walk, the number of potential paths doubles at each step. This means the total number of paths after n steps will be equal to 2^n giving us an exponential number of paths that could be traced. Unlike most quantum walk works, our method is interested in the result of the paths of the walk and not the vertices that are reached at the end of the walk. As the walk progresses and the state evolves, we sample the $f(x)$ at each position visited by the walker using the M circuit described in the main paper. This captures the path that was taken by the walker. The circuit M does not depend on the walk, depending solely on the current position of the walker, so it is able to capture the paths in the more complicated domain of rendering. Similarly to the $1D$ walk there will also be a similar exponential branching as the light is scattered at each diffuse surface.

SA Conference Papers '23, December 12–15, 2023, Sydney, NSW, Australia

ACM ISBN 979-8-4007-0315-7/23/12…$15.00
https://doi.org/10.1145/3610548.3618151

2 IMPLEMENTATION DETAILS

As quantum computers are still at the level of circuits, implementing complex algorithms is difficult as operations that would normally be trivial, such as adding, require the construction of specialized circuits. To help facilitate the construction of the circuits presented in this paper some approximations were made to the standard ray marching that would be typically implemented on a classical computer. The main approximation that is made is that a look-up table of directions must be defined before circuit construction. This lookup table greatly simplifies the implementation of both the surface coins and the shift operator.

Our approach does not rely on any particular implementation for defining this lookup table, we choose uniformly sampling the unit sphere for the results presented in the paper though empirically random sampling can provide better results when the number of directions used is small. It is important to note that the resolution of the lookup table effects the scaling of the circuit. The position of the walker must be represented with a high enough resolution such that any a step using any vector results in the walker arriving at a unique final location compared when starting from the same initial location. Failing to do this, will result in the direction of the lookup table not being accurately applied to the position of the walker. Also, the look-up of the appropriate vector to apply at each step is linear in the number of directions as there is no efficient memory lookups on quantum computers. This means that the depth of the circuit grows linearly as the number of directions increases.

Once the lookup table is defined we can then construct the circuit based on the scene and set of directions The shift operator is implemented by controlled increment and decrement gates that add the constant values to the position register, controlled by the direction register. The coin operator is built out of the many controlled coin gates. Each coin gate is controlled from the position of the walker based on what voxels the walker is currently in. The coins function as a mapping from the incoming direction ID to a set of outgoing IDS. For the air, each ID is mapped to its self. Since this operation has no effect, these gates are skipped in the circuit construction, but it can be helpful to think of them conceptually being the identity gate as then it may be treated the same as any other coin that is present in the scene. The specular coin is a 1-to-1 mapping of directions whereas the Lambertian coin is a 1 to many mapping. The exact directions that are elected for the mappings may differ depending on how the set of directions but the directions and weights must be selected such that they respect the properties of BSDF that the coin is emulating.

REFERENCES

Karthik S. Joshi, S. K. Srivatsa, and R. Srikanth. 2018. Path Integral Approach to One-dimensional Discrete-time Quantum Walk. arXiv:1803.00448 [quant-ph]