A Relaxation Scheme for TSP-based 3D Printing Path Optimizer

Kai-Yin Fok†, Chi-Tsun Cheng, Chi K. Tse, and Nuwan Ganganath
Department of Electronic and Information Engineering
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
Email: †zerofky@gmail.com

***Abstract*—Additive-layered manufacturing has gained attention in recent years as it has many advantages over conventional injection moulding methods. The optimization of printing trajectories can be formulated as a travelling salesman problem (TSP) and solved accordingly. However, computational complexities of ordinary TSP solvers can increase tremendously with the scale of the problem, which make them impractical. In this work, a relaxation scheme for TSP-based 3D printing path optimizer is proposed. Simulation results show that the proposed scheme can significantly shorten the computational time of the optimization process with insignificant impact on solution quality.**

***Index Terms*—Additive manufacturing, 3D printing, TSP, optimisation, relaxation.**

I. Introduction

The popularity of additive-layered manufacturing is continuously rising and has gained more and more attentions in recent years. 3D printing technology can be applied in many different fields include prototyping and customized product manufacturing. Customized prostheses can be built using 3D printers and their costs can be less than one-tenth to those using injection moulding [1]. In addition, carbon-reinforced thermoplastic is used in building vehicle parts [2].

In order to build a model, a computer-aided design (CAD) file is required. The CAD file is usually fed into a slicing software to generate control instructions, which indicate the movement of the mechanical components in a 3D printer. Molten printing filament is dispensed onto the printing platform to construct the model in a layer-by-layer manner. The build time of a 3D model is directly affected by the motions of three mechanical parts in a 3D printer, know as the printing nozzle, printing platform, and extruder wheel. In a typical printer, the printing nozzle and platform move horizontally and vertically throughout the whole printing process, respectively. The extruder wheel controls the flow rate of filament.

Movements of the aforementioned mechanical parts can significantly affect the build time of a printed object. In general, in order to print a single segment, the printing nozzle will first move to the start point of such segment, it will then traverse the segment while the extruder wheel will apply pressure to the nozzle, such that desired amount of filament is dispensed along the path. Before reaching the end point, the extruder wheel will retrace and reduce the pressure to the printing nozzle and stop further disposition of filament beyond the end point. The process repeats until all segments on the same layer are printed. Then the elevation of the printing platform is suitably adjusted before starting the printing of the next layer.

In [3], [4], Thompson and Yoon tried to reduce filament waste by maintaining the printing nozzle at a constant desirable velocity while printing and adopting a fast motion profile for transitions. By predicting velocity errors, fluctuations in speed can be greatly reduced. In [5], Ganganath *et al.* formulated the printing nozzle optimization problem as a traveling salesman problem (TSP). Based on their formulation, several modified TSP solving algorithms were proposed and evaluated using computer simulations. In their simulations, a single-layer of print segments was considered. A significant improvement is observed when comparing their proposed methods with greedy-based solvers. Fok *et al.* in [6] further fine-tuned a Christofides-based solver in [5] and applied it to real-life 3D models. Simulation results show that their proposed algorithm can outperform Cura [7], a common open-source model slicer for 3D printers.

Despite nozzle trajectories, printing orientation of a 3D model can seriously affect its stability and the amount of support materials needed. Son and Choi studied the effects of different 3D model printing orientations to three printing performance indices [8], namely a) amount of support material used, b) resolution error on the Z dimension, and c) visual attention of support residue. Based on these performance indices, they proposed an automated orientation selection method, which can yield higher efficiency and accuracy in 3D printing processes. Zarbakhsh et al. in [9] considered about the physical properties of printed objects. They tried to analyze regions of interest on printed objects by adopting a technique called sub-modeling. Their results show that object filling patterns could introduce extra stress between printed layers.

In this work, a relaxation scheme for TSP-based 3D printing path optimizer is proposed to shorten the processing time of the optimization process. Section II defines the problem formulation including the motion models of the printing nozzle and the objective function used in the printing trajectories optimization process. The relaxation scheme is introduced in Section III. The proposed scheme is evaluated using computer simulations. Simulation settings are described in Section IV followed by results and analysis. Concluding remarks are given in Section V.

II. Problem Formulation

This section focuses on the formulation of the 3D print path optimization problem. The objective is to find a quick path which traverses through all print segments. For typical 3D printers, in order to build an object, 3D model is first sliced into many thin layers. Object fragments in each layers are then constructed by massive volume of segments. A segment is a straight line defined by its two end points. Curves are represented by multiple segments. In fact, there are two types of segments. The first type is print segments, which represent the print part of the model. The printing nozzle will deposit material when traversing through a print segment. In order to maintain consistent material deposition, the rate of depositing is adjusted jointly by the speed of extruder wheel and the motion of the printing nozzle. The second type of segments is transition segments, which represent the path for the nozzle to traverse between print segments. In order to minimize the transition time, the printing nozzle should move at its highest possible velocity while it is moving along a transition segment.

A. Motion Model

The motion model used in this work has three major properties. First, the velocity of the printing nozzle can be changed when traversing both print and transition segments, given that the extruder wheel can control the rate of depositing material accordingly. Second, it is assumed that the printing nozzle can stop precisely at the exact defined point after traversed through a segment. Third, the printing nozzle takes time to accelerate before reaching its maximum velocity. Triangular and trapezoidal velocity motion profiles are used to describe the motion of the printing nozzle.

Let us denote a_1 and a_2 as the acceleration and deceleration of the printing nozzle respectively, and $v_{\max}$ be the maximum velocity for traversing a transition segment. Minimum distance required to accelerate the printing nozzle to its maximum velocity and decelerate till stop is expressed as

$$d_{\min} = \frac{v_{\max}^2}{2}\left(\frac{1}{a_1} - \frac{1}{a_2}\right). \tag{1}$$

Let d be the length of a transition segment. The time required to traverse such a segment is calculated as

$$t_d = \begin{cases} \sqrt{2d\left(\frac{1}{a_1} - \frac{1}{a_2}\right)} & d \leq d_{\min} \\ \frac{v_{\max}}{a_1} - \frac{v_{\max}}{a_2} + \frac{d - d_{\min}}{v_{\max}} & d > d_{\min} \end{cases}. \tag{2}$$

B. Retraction

Excess filament drippings being deposited outside the printed model are known as *strings*. 3D printers can perform retractions on their extruder wheels to mitigate this string problem. According to [10], a retraction is performed by pulling filament back from the nozzle while it moves between two separated print segments. Some printers also descend the printing platform to further reduce the chance of having strings on the final product [10]. However, an extra time is required for performing a retraction. In this work, a retraction is always preformed when the printing nozzle traverses across separated build parts. The time required to perform a retraction is denoted as t_r.

C. Objective Function

The goal of the printing path optimizer considered in this work is to minimize the total print time of a model. The objective function utilized in the optimizer consists of two components, namely the total time required to traverse all the required transition segments and the total time spent on all the necessary retractions. The time required to traverse a transition segment is calculated based on the motion model introduced in Section II-A and the length of the transition segment. The time required to perform a retraction is assumed to be a constant which can be obtained empirically.

D. Travelling Salesman Problem

An intuitive approach for finding a fast path is to conduct exhaustive searches. However, it is not practical for medium to large-scale models due to its computational complexity. The printing path optimization problem in 3D printing can be formulated as a TSP by regarding all the print segments as the cities in TSP which are required to be visited. The objective is to find a fast tour which can visit all print segments at least once. Via such formulation, efficient TSP solvers can be applied with minor modifications. In our work [6], a path optimizer based on Christofides algorithm was proposed. However, the computational complexity of such optimizer is still high for large-scale 3D models. Furthermore, it was using distance instead of time as the criterion in its optimization which may omitted the overhead due to retractions.

III. Relaxation Scheme

In this paper, the path optimizer proposed in [6] is further modified to address the above issues. The objective function is replaced by the time-based objective function described in Section II-C. Solutions obtained from the optimizer will undergo a refinement process using the 2-opt algorithm [11] which showed to be highly effective in [5].

To reduce the computational complexity of the optimization process, a relaxation scheme is proposed in this work. The proposed relaxation scheme contains two major tasks includes Partitioning and Segments consolidation.

A. Partitioning

The first task of the relaxation scheme is to partition the whole print layer into sub-parts. By doing so, TSP-optimization processes can be performed on those smaller parts separately which are normally associated with much lower computational complexities. As a 3D object is printed layer-by-layer, although the object may not have separated components, after slicing, there can be discrete print parts on some of the layers. A print layer can therefore be partitioned according to its boundaries. Print segments located inside the same boundary are assigned with the same partition index. TSP optimizations will then be carried out at the inter-partitions level and the intra-partition level sequentially. At the inter-partitions level, centres of the partitions are regarded as the vertices to be visited. TSP-optimization processes at the intra-partition level will then be carried our according to the visiting schedule obtained in the inter-partitions level. Details will be described in the next section.

B. Segments Consolidation

As mentioned in Section II, curves on a print layer are represented by chains of connected short print segments. The computational complexity of the TSP-optimization process will be largely increased if we consider those segments as separated components. On the other hand, the complexity of the problem can be significantly reduced by consolidating multiple connected print segments into a single replacement segment during calculations. Note that such process needs to be carried out with cautions, as consolidating a very long chain of print segments may eliminate branch options at the middle of the chain which may yield better results in the TSP-optimization. A consolidation threshold ϕ is therefore introduced here, which controls the maximum length of print segments to be consolidated at a time. Effects on the selection of ϕ are further discussed in Section IV. At the end of the TSP-optimization process, replacement segments are reverted to the corresponding chains of print segments.

Given the print segments on the current layer, the optimization process will begin with the partitioning task. A TSP-optimization will be performed to find a fast tour to visit all the centers points of the identified partitions, which also determines their order of visit. Inside each partition, the Christofides-based TSP solver with 2-opt in [5] will be used to find a fast tour. To shorten the transition segments among partitions, the virtual segment required by the method in [5] will be placed between two end points that are nearest to the centers of the adjacent partitions. This will ensure the tour will always start and end at these two selected points. An extra transition segment will be added to guide the printing nozzle to the nearest selected end point mentioned above. Transition segments will then be added to join those selected points of adjacent partitions together and form a single tour.

IV. Simulations

Performances of the relaxation scheme together with the optimizer introduced in the last section were evaluated using computer simulations. Simulation parameters and results are presented in this section.

A. Simulations Settings

In the simulations, seven 3D models in [12] were chosen as sample models. The sizes of the models were adjusted such that they can fit in the print platform of a typical 3D printer, which is 250×250 mm^2. The consolidation threshold ϕ used in the proposed relaxation scheme is 4 mm which is arbitrarily selected. The print segments of each model were generated using Cura [7]. The order for visiting those print segments were further optimized using the proposed scheme and the optimizer mentioned in III. In Cura, the speed for traversing a print segment is set to 50 mms^{-1} while that for traversing a transition segment (i.e. $v_{\max}$) is set to 150 mms^{-1}. Retractions were enabled with a retraction length of 4.5 mm. Vertical hop during retracting was set to 0.075 mm. The filling density was 10% of the total volume. All optimized printing instructions were evaluated using the Code Print Simulator-1.32 [13]. The extra processing time required by the proposed method were the average values obtained from 10 individual simulations conducted on a computer with Windows 8.1, Intel Core i7 processor, and 16 GB RAM. All the results are presented in Table I.

B. Results and Discussion

According to the simulation results given in Table I, it can be observed that the proposed method can on average reduce object print time by 10.63% when comparing with Cura. Even by including the processing time of the proposed method, while assuming that of Cura is negligible, an average build time (print time + processing time) saving of 9.62% can still be achieved.

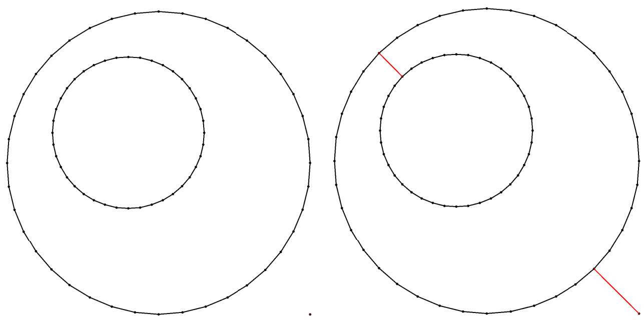
Note that for the results associated with the model "UltimakerRobot_support_2015" the proposed method can only deliver a saving of 2.46% on the total build time over Cura. One possible reason is due to the compact structure of the model, where most print segments are connected together and have left very limited rooms for optimization to take place. In contrast, the proposed method can achieve a saving of 11.06% on the total build time over Cura on the model "dragon_65_tilted large", which consists of a large number of separated components.

As mentioned in Section III, consecutive print segments are substituted by replacement segments in the proposed relaxation scheme based on the consolidation threshold. The selection of such threshold can be crucial to the solution quality. A small value of ϕ will not have significant effects on reducing processing time as very few print segments can be consolidated. A large value of ϕ can greatly reduce the complexity of the problem but may reduce branching options in the TSP-optimization process and end up with longer tours. Fig. 1 shows a simplified example illustrating how such threshold may affect the print time.

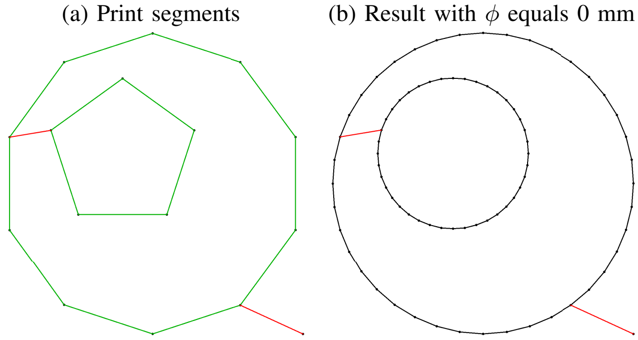
Consider a set of print segments as shown in Fig. 1a, which appear as two tetracontagons. Suppose the printing nozzle is located at the bottom right corner. When ϕ is smaller than the length of the shortest segment, no segment consolidation will be executed. A TSP-optimization will be performed by considering them as 80 separated segments and yield the result shown in Fig. 1b. For a larger value of ϕ, multiple consecutive segments can be substituted by replacement segments in the calculation process. In this example, the computational complexity can be greatly reduced as the number of segments is reduced from 80 to 15 as shown in Fig. 1d. However, it can also be observed that the length of transition segments in Fig. 1d is slightly longer than that in Fig. 1b. In general, for most scenarios, selecting a large value of ϕ is recommended as the negative impact due to longer transition paths is not

TABLE I: SIMULATION RESULTS

3D models	Built-in algorithm	Proposed optimizer	
	Build time (s)	Print time (s)	Average processing time (s)
UltimakerRobot_support_2015	1640.0454	1593.6926	5.9317
TortureTestV2	7556.5776	6775.9507	57.3606
testModel	1688.8782	1584.6810	1.3679
dragon_65_tilted_large	2499.2075	2093.8506	10.7101
Debailey_x10	3735.8640	3313.2075	9.2961
ctrlV_3D_test	3393.6820	2967.4090	83.8600
3DHackerTest	4566.2954	3881.9224	122.9981



(a) Print segments (b) Result with ϕ equals 0 mm



(c) Replacement segments (d) Result with ϕ equals 70 mm

Fig. 1: Illustrations of print instruction using different values of ϕ. Print, transition, and replacement segments are shown in black, red, and green colour, respectively.

significant to the total print time which is often eliminated by the huge reduction in processing time.

V. Conclusion

In this paper, a relaxation scheme for TSP-based 3D printing path optimizers is proposed. The relaxation scheme reduces the computational complexity of TSP in 3D printing by partitioning each layer of print area into sub-parts and consolidating consecutive print segments in the calculations. Performance of the proposed scheme is evaluated using computer simulations. Simulation results show that the proposed scheme can greatly shorten the processing time of a TSP-based printing path optimizer without introducing significant impacts on the solution quality.

Acknowledgment

This work is supported by the Department of Electronic and Information Engineering, the Hong Kong Polytechnic University (Projects RU9D, RTKL and G-YBKH) and the Hong Kong PhD Fellowship Scheme.

References

[1] T. Hornyak, "This Exiii Hackberry 3D-printed hand costs $200 and could mean the world for amputees," (Accessed: 2016-07-01). [Online]. Available: http://www.pcworld.com/article/2932632

[2] C. Engelking, "Worldś first 3D printed car takes to the streets," (Accessed: 2016-07-01). [Online]. Available: http://blogs.discovermagazine.com/d-brief/2014/09/29/worlds-first-3-d-printed-car

[3] B. Thompson and H. S. Yoon, "Efficient path planning algorithm for additive manufacturing systems," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 4, no. 9, pp. 1555–1563, Sept 2014.

[4] B. Thompson and H.-S. Yoon, "Velocity-regulated path planning algorithm for aerosol printing systems," *Journal of Manufacturing Science and Engineering*, vol. 137, no. 3, pp. 031 020–031 020, Jun. 2015. [Online]. Available: http://dx.doi.org/10.1115/1.4029976

[5] N. Ganganath, C. T. Cheng, K. Y. Fok, and C. K. Tse, "Trajectory planning for 3D printing: A revisit to traveling salesman problem," in *2016 2nd International Conference on Control, Automation and Robotics (ICCAR)*, April 2016, pp. 287–290.

[6] K. Y. Fok, N. Ganganath, C. T. Cheng, and C. K. Tse, "A 3D printing path optimizer based on christofides algorithm," in *2016 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, May 2016, pp. 1–2.

[7] Ultimaker, "Cura 3D Printing Slicing Software," (Accessed: 2016-07-01). [Online]. Available: https://ultimaker.com/en/products/cura-software

[8] J. Son and S. Choi, "Orientation selection for printing 3D models," in *3D Imaging (IC3D), 2015 International Conference on.* IEEE, 2015, pp. 1–6.

[9] J. Zarbakhsh, A. Iravani, and Z. Amin-Akhlaghi, "Sub-modeling finite element analysis of 3D printed structures," in *Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE), 2015 16th International Conference on.* IEEE, 2015, pp. 1–4.

[10] " Material — Ultimaker: 3D Printers," (Accessed: 2016-07-01). [Online]. Available: https://ultimaker.com/en/settings/20232-material

[11] D. S. Johnson and L. A. McGeoch, "The traveling salesman problem: A case study in local optimization," *Local search in combinatorial optimization*, vol. 1, pp. 215–310, 1997.

[12] T. Kuipers, "CuraEngine/tests at 4c547b9a66433885d4d4128f5f416982878b7e56 Ultimaker/CuraEngine GitHub," (Accessed: 2016-07-01). [Online]. Available: https://github.com/Ultimaker/CuraEngine/tree/4c547b9a66433885d4d4128f5f416982878b7e56/tests

[13] M. Dietz, "GCode Print Simulator by mdietz - Thingiverse," (Accessed: 2016-01-7). [Online]. Available: http://www.thingiverse.com/thing:44286