

6X-01

## グラフベース点群符号化におけるパラメータ最適化に関する基礎検討

桐原 日向<sup>†</sup>伊吹 将一<sup>‡</sup>藤橋 卓也<sup>‡</sup>秋濃 (小池) 俊昭<sup>‡†</sup>渡辺 尚<sup>‡</sup><sup>†</sup> 大阪大学工学部<sup>‡</sup> 大阪大学大学院情報科学研究科<sup>‡†</sup> Mitsubishi Electric Research Laboratories

## 1 はじめに

点群は3次元空間上の点を多数用いて3次元情報を表現する。点群をネットワーク伝送する場合、その多大なトラフィックが大きな課題となる。トラフィックの抑制を図るためにグラフ信号処理を用いた情報源符号化技術が提案されている。本研究ではグラフ信号処理を用いた符号化技術において低トラフィックを達成するパラメータ最適化手法について提案する。

## 2 課題

限られた帯域を有するネットワークを介して点群を伝送する場合、点群に含まれる冗長な情報を効率的に削減するトラフィック削減手法の実現が重要な課題となる。H.264/Advanced Video Coding や H.265/High Efficiency Video Coding に挙げられる画像情報や映像情報を対象としたトラフィック削減手法は平面上に等間隔でサンプリングされた画素信号を前提として設計されている。一方で、点群においては3次元上で不規則にサンプリングされた信号を対象とする必要がある。点群を圧縮するための符号化技術として Draco [1], Point Cloud Library (PCL) [2], グラフ信号処理 (Graph Signal Processing:GSP) [3] を用いたトラフィック削減手法がそれぞれ提案されている。例えば、Draco においては Kd 木、PCL においては Octree を用いて各点の座標成分をビット列に圧縮する。グラフ信号処理を用いたトラフィック削減手法においては、点群を構成する各点を頂点、点間の距離を辺の重みとみなしたグラフ信号を構築する。その後、グラフ信号から得られたグラフラプラシアン演算子に対する固有値分解を経てグラフ信号に対する基底関数を得ることで対象となる点群に対するグラフフーリエ変換 (Graph Fourier Transform: GFT) [4, 5] を定義することができる。グラフフーリエ変換を通して点群を周波数変換することで点間に含まれる冗長な情報を削減できることがわかっている。

一方で、グラフ信号処理を用いたトラフィック削減手法においては基底関数の導出に起因する計算量を抑制する必要がある。基底関数の元となるグラフラプラシアン演算子はグラフ信号から得られた隣接行列、次数行列ならびに実数で構成される7パラメータを入力として構成される。その後、得られたグラフラプラシアン演算子に対して固有値分解を用いることで基底関数に相当する固有ベクトルを得ることができる。グラフラプラシアン演算子に応じて得られる基底関数およびグラフ信号の周波数成分は変化するため、トラフィック削減効果が高いパラメータを適切に設定する必要がある。しかしながら、点群に含まれる点数が  $N$ 、各パラメータに対する探索回数が  $M$  であるとき、 $M^7$  個のパラメータそれぞれに対して固有値分解、すなわち  $O(N^3)$  の計算量が必要となる。本研究では、1) トラフィック削減にもたらす影響を考慮したパラメータ数の削減、2) 固有値分解を必要としないパラメータ探索の実現、3) 訓練用点群から得られたパラメータの適用を通して計算量を抑えつつ効率的にトラフィック削減を達成するグラフベース点群符号化手法を提案する。

## 3 点群におけるグラフ信号処理

グラフ信号処理では、3次元点群を重み付き無向グラフ信号  $G = (V, \mathcal{E}, W)$  としてみなす。ここで  $V$  と  $\mathcal{E}$  はそれぞれ  $G$  の頂点と辺の集合である。また、 $W$  は正の重みをもつ隣接行列であり、隣接行列の  $(i, j)$  要素  $W_{i,j}$  は頂点  $i$  と  $j$  の間の辺の重みを表す。本稿では、頂点数  $N$  の3次元点群  $\mathbf{p} = [x, y, z]^T \in \mathbb{R}^{3 \times N}$  をグラフ  $G$  の頂点集合とみなす。隣接行列の各重み  $W_{i,j}$  を次式のとおり定義する。

$$W_{i,j} = \exp\left(-\frac{\|\mathbf{p}_i - \mathbf{p}_j\|_2^2}{\kappa_p}\right) \quad (1)$$

ここで、 $\kappa_p$  は座標成分に対するパラメータであり、本稿では座標成分の分散値を用いた。次数行列  $D$  は隣接行列にしたがって次式のとおり導

出される。

$$D = (D_1, D_2, \dots, D_N), D_i = \sum_{n=1}^N W_{i,n} \quad (2)$$

GFT に用いる基底関数は隣接行列と次数行列によって得られるグラフラプラシアン演算子  $L$  によって定義される。ここで、グラフラプラシアン演算子は次式のとおり求められる。

$$L = m_1 D_a^{e_1} + m_2 D_a^{e_2} W_a D_a^{e_3} + m_3 I \quad (3)$$

ここで、隣接行列  $W_a = W + aI$ 、 $D_a$  は隣接行列  $W_a$  から得られた次数行列、 $I$  は  $N \times N$  単位行列である。また、 $\mathbf{m} = (m_1, m_2, m_3, e_1, e_2, e_3, a) \in \mathbb{R}^7$  はグラフラプラシアン演算子を構成するハイパーパラメータ集合である。一般のグラフラプラシアン演算子は実対称行列と相似な行列であるため、互いに直交する固有ベクトルと各固有ベクトルに対応する非負の固有値を持つ。グラフラプラシアン演算子に対する固有値分解は次式のとおり定義できる。

$$L = \Phi \Delta \Phi^{-1} \quad (4)$$

ここで、 $\Phi$  は固有ベクトル行列、 $\Delta$  は対角成分を固有値とする対角行列である。各属性の GFT 係数  $\mathbf{f}$  は基底関数  $\Phi$  と各属性値  $\mathbf{s}$  を乗じることで得られる。

$$\mathbf{f} = \mathbf{s} \Phi \quad (5)$$

逆グラフフーリエ変換 (IGFT) は GFT 係数ならびに基底関数から各属性値を得ることができる。IGFT は次式のとおり定義できる。

$$\mathbf{s} = \mathbf{f} \Phi^{-1} \quad (6)$$

## 4 提案手法

## 4.1 全体像

図1に提案手法の全体像を示す。提案手法は GFT を用いた点群符号化器および点群復号化器からなる。点群符号化器は  $N$  個の点から構成される3次元点群を入力としてグラフ構造を構築するとともにグラフ構造から得られた基底関数を用いて各点の座標情報を属性値として GFT を適用することで GFT 係数  $\mathbf{f}$  を得る。得られた GFT 係数は量子化ステップサイズ  $q$  にしたがって  $\mathbf{c}$  に量子化される。このとき、量子化ステップサイズは次式のとおり量子化後の最大値と最小値の差が  $r$  となる最小のステップサイズを用いる。

$$\mathbf{c} = \text{round}\left(\frac{\mathbf{f}}{q}\right) \quad (7)$$

$$q = \min q', \text{ s.t. } \max\left(\text{round}\left(\frac{\mathbf{f}}{q'}\right)\right) - \min\left(\text{round}\left(\frac{\mathbf{f}}{q'}\right)\right) = r$$

量子化後の GFT 係数はハフマン符号を用いてビット列に符号化して伝送する。

復号化器では、受信したビット列を量子化後の GFT 係数  $\hat{\mathbf{c}}$  に復号化する。得られた係数に対して点群符号化器で用いた量子化ステップサイズ  $q$  を次式のように適用することで GFT 係数を復元する。

$$\hat{\mathbf{f}} = q \hat{\mathbf{c}} \quad (8)$$

最後に復元した GFT 係数に対して式 (6) で示した IGFT を導入することで各点の座標情報の推定値  $\hat{\mathbf{s}}$  を得ることができる。

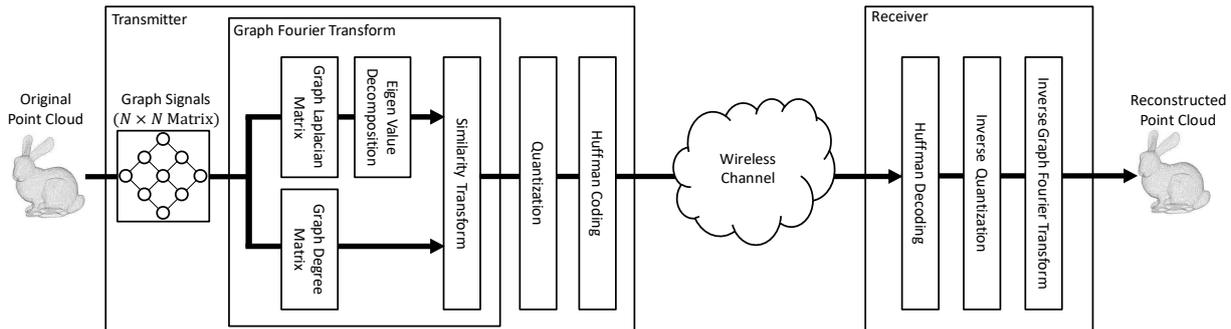


図1: 提案手法の全体像

## 4.2 パラメータ削減

トラヒックを効率的に削減できる基底関数  $\Phi$  を得るためには、グラフラプラシアン演算子に対する適切なパラメータ集合  $m$  ならびに量子化パラメータ  $q$  を定める必要がある。一方で、各パラメータに対する探索回数が  $M$  であるとき、 $M^8$  回のパラメータ探索を通して点群の復元品質を維持したままトラヒックを削減可能なパラメータ対を決定する必要がある。

本稿では、より少ない探索回数で適切なパラメータセットを決定するために、グラフラプラシアン演算子を構成するハイパーパラメータ集合  $m$  に含まれるパラメータのうち、トラヒック削減に寄与する度合いが小さいパラメータに着目する。まず、提案手法では基底関数として固有ベクトルのみを利用することから、その値によらずグラフラプラシアン演算子から同じ固有ベクトルを取得できるパラメータ  $m_3 = 0$  として次式のとおりグラフラプラシアン演算子を定義する。

$$L_{6\text{params}} = m_1 D_a^{e_1} + m_2 D_a^{e_2} W_a D_a^{e_3} \quad (9)$$

次に  $m'_1 = m_1/m_2$  を定義する。上述のグラフラプラシアン演算子は  $m'_1$  を用いて次式のとおり置き換えることができる。

$$L_{5\text{params}} = \frac{L_{6\text{params}}}{m_2} = m'_1 D_a^{e_1} + D_a^{e_2} W_a D_a^{e_3} \quad (10)$$

固有ベクトルの定義から  $L_{6\text{params}}$  の固有ベクトルと  $L$  の固有ベクトルは一致することから  $m_2$  の探索を省くことができる。

なお、 $L_{5\text{params}}$  から得られる固有ベクトルが基底行列、すなわち直交行列であるためには  $L_{5\text{params}}$  が実対称行列であることが求められる。 $L_{5\text{params}}$  において第1項は実対称行列、第2項は  $e_2 = e_3$  であるときに実対称行列となるため、次式のとおりグラフラプラシアン演算子  $L_{4\text{params}}$  を定義できる。

$$L_{4\text{params}} = m'_1 D_a^{e_1} + D_a^{e_2} W_a D_a^{e_2} \quad (11)$$

以上の点から4変数に削減したハイパーパラメータ集合  $m' = (m_1, e_1, e_2, a)$  ならびに量子化パラメータ  $q$  を適切に定めることでより少ない計算量でトラヒックを効率的に削減できる基底関数  $\Phi$  を得ることができる。

## 4.3 固有値分解を必要としないパラメータ探索

前節で定義したグラフラプラシアン演算子  $L_{4\text{params}}$  を利用することで、より少ない計算量で基底関数  $\Phi$  を定めることができる。一方で、それぞれのパラメータ対  $m' = (m_1, e_1, e_2, a)$  に対して  $O(N^3)$  の固有値分解を適用して基底関数を求める必要がある。

提案手法ではグラフラプラシアン演算子に対する相似変換を利用することで固有値分解後のパラメータ探索を実現する。本稿ではグラフラプラシアン演算子に対する相似変換を次数行列  $D_a$  を用いて次式のとおり定義する。

$$L' = D_a^e L_{4\text{params}} D_a^{-e} = m_1 D_a^{e_1} + D_a^{e_2+e} W_a D_a^{e_2-e} \quad (12)$$

このとき  $L_{4\text{params}}$  の固有ベクトル行列  $\Phi$  と  $L'$  の固有ベクトル行列  $\Phi'$  には次式の関係が成り立つ。

$$\Phi' = D_a^e \Phi, \quad \Phi'^{-1} = \Phi^{-1} D_a^{-e} \quad (13)$$

式(13)より、パラメータ対  $m' = (m_1, e_1, e_2, a)$  を固定して得られた固有ベクトル行列に対してパラメータ  $e$  に応じて固有ベクトル行列を変換することで固有値分解を必要とせず、異なる基底関数を求めることが可能となる。

## 5 性能評価

### 5.1 評価環境

本稿では提案手法によるトラヒック削減効果を評価するために、データセットとしてShapeNetを用いた。ShapeNetは“Airplane”, “Chair”, “Table”に挙げられる複数のカテゴリから構成される。本評価ではカテゴリ“Chair”を用いた。

品質指標として式(14)に示したChamfer distanceと式(15)に示したEntropyを用いた。

$$d = \frac{1}{2} \left\{ \frac{1}{|S|} \sum_{p \in S} \min_{\hat{p} \in \hat{S}} \|p - \hat{p}\|_2 + \frac{1}{|\hat{S}|} \sum_{\hat{p} \in \hat{S}} \min_{p \in S} \|p - \hat{p}\|_2 \right\} \quad (14)$$

$$E = - \sum_{q \in c} P(q) \log_2 P(q) \quad (15)$$

ここで、 $S$  は元信号となる3次元点群の集合、 $\hat{S}$  は復号後の3次元点群の集合、 $c$  は量子化後のGFT係数、 $P(q)$  は  $c$  に占める  $q$  の割合を表す。

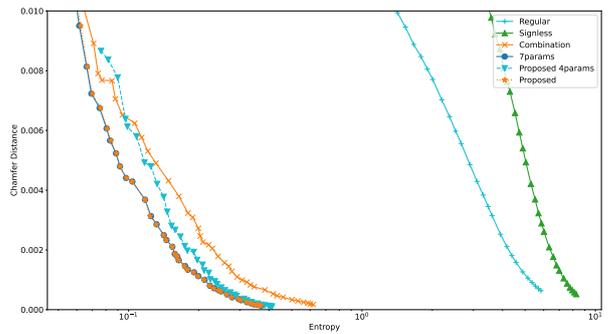


図2: エントロピーに対する復元点群品質

また、最適なパラメータ対  $(m_1, e_1, e_2, a)$  を決定するための指標として式(19)に示したBjontegaard delta (BD) rateを用いた。

$$\text{BD rate} = 10^{\frac{1}{D_H - D_L} \int_{D_L}^{D_H} (\log E_2(d_2) - \log E_1(d_1)) dd} \quad (16)$$

$$C = \{(d, E) \mid \{(d', E') \mid d' < d \wedge E' < E\} = \emptyset\} \quad (17)$$

$$D_H = \min \left( \max_{d_1 \in C_1} (d_1), \max_{d_2 \in C_2} (d_2) \right) \quad (18)$$

$$D_L = \max \left( \min_{d_1 \in C_1} (d_1), \min_{d_2 \in C_2} (d_2) \right) \quad (19)$$

比較手法としてグラフラプラシアン演算子として広く用いられているRegular, Signless, Combinationalを用いた。それぞれのグラフラプラシアン演算子は次式のとおり定まる。

$$\text{Regular} : L = D - W \quad (20)$$

$$\text{Signless} : L = D + W \quad (21)$$

$$\text{Combination} : L = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \quad (22)$$

また、最も計算量がかかる場合の性能として式(3)に示したグラフラプラシアン演算子を“7params”として採用した。提案手法として式(11)に示したグラフラプラシアン演算子を“proposed 4params”, 式(13)に示したグラフラプラシアン演算子を“proposed”として示す。

## 5.2 評価結果

図2にそれぞれの比較手法におけるエントロピーに対する復元点群品質を示す。評価結果から以下の2つのことが分かる。

- 提案手法は7paramsとほとんど同じ性能を達成できること
- 典型的なグラフラプラシアン演算子に対して同エントロピー下において、より高い復元点群品質を達成できること

## 6 おわりに

本稿ではグラフベース点群符号化におけるパラメータ最適化に関しての高速化手法を提案した。提案手法を導入することで探索空間を維持しながら設定する必要のあるパラメータ数の削減および演算子の変換を行うことができるため、高速な探索が可能となる。評価結果より、従来の手法に比べて劣化がほとんどないことがわかった。

## 謝辞

本研究はJSPS 科研費 (JP19H01101, JP22H03582) の支援の下で行った。

## 参考文献

- [1] “Draco 3D data compression,” <https://google.github.io/draco/>.
- [2] R.B. Rusu, and S. Cousins, “3D is here: Point cloud library (PCL),” IEEE International Conference on Robotics and Automation, pp.1-4, 2011.
- [3] T. Fujihashi, T. Koike-Akino, T. Watanabe, and P.V. Orlik, “Holocaust+: Hybrid digital-analog transmission for graceful point cloud delivery with graph fourier transform,” IEEE Transactions on Multimedia, vol.24, pp.2179-2191, 2022.
- [4] A. Ortega, P. Frossard, J. Kovačević, J.M.F. Moura, and P. Vandergheynst, “Graph signal processing: Overview, challenges, and applications,” Proceedings of the IEEE, vol.106, no.5, pp.808-828, 2018.
- [5] C. Zhang, D. Florêncio, and C. Loop, “Point cloud attribute compression with graph transform,” IEEE International Conference on Image Processing (ICIP), pp.2066-2070, 2014.