A Fast MBO Scheme for Multiclass Data Classification

Matt Jacobs[1]

University of Michigan

Abstract. We describe a new variant of the MBO scheme for solving the semi-supervised data classification problem on a weighted graph. The scheme is based on the minimization of the graph heat content energy. The resulting algorithms guarantee dissipation of the graph heat content energy for an extremely wide class of weight matrices. As a result our method is both flexible and unconditionally stable. Experimental results on benchmark machine learning datasets shows that our approach matches or exceeds the performance of current state of the art variational methods while being considerably faster.

1 Introduction

Classifying high dimensional data is one of the central problems in machine learning and computer vision. The graphical approach to these problems builds a weighted graph from the data set and searches for an optimal partitioning of the vertices into distinct classes. The search is driven by the goal of minimizing the total weight of cut edges between adjacent vertices in different classes. To avoid trivial solutions, it is necessary to impose certain constraints or penalties on the segmentations. For example, one may penalize solutions that do not give a reasonably uniform distribution of vertices among the different classes. In general, solving graph partitioning problems with combinatorial penalties, such as the normalized cut [12] or Cheeger cut [3], is known to be NP-hard. The essential difficulty stems from the fact that one is attempting to minimize a non-convex objective function. Nonetheless, approximate solutions have been calculated using spectral clustering (for example [12], [21]), and more recently fast implementations of gradient descent [10], [2].

In this paper we consider the semi-supervised learning (SSL) data classification problem. In the SSL setting, the number of classes is known and a certain training subset of the data is provided with the correct classification given. The objective is to then classify the remaining points using the training data. The SSL problem is highly amenable to variational methods. The training data can be incorporated into norm or linear type penalty terms that are much easier to solve than the combinatorial penalties of the unsupervised methods mentioned above. Recent results in SSL have shown that variational methods are competitive with artificial neural networks, while requiring far less training data and computation time to obtain high quality solutions [9].

We approach the SSL problem using a variational model based on the weighted graph cut. We then solve the model using a scheme closely related to the MBO algorithm. The MBO algorithm was introduced by Merriman, Bence and Osher in [14] as an efficient algorithm for generating mean curvature flow of an interface. The algorithm alternates between solving a linear diffusion equation and pointwise thresholding. In Euclidean space, mean curvature flow arises as gradient descent for minimal partition problems. Thus, it is naturally connected to segmentation e.g. via the Mumford-Shah functional [16] and the many other models it inspired (see chapter 25 in [19] for an exhaustive reference). As a result, MBO type schemes have been used to solve a number of segmentation problems. The authors of [7] derived an MBO scheme from the Ginzburg-Landau energy to solve the piecewise constant Mumford-Shah functional. Building on the approach of [7], the authors of [9] introduced a multiclass version of the Ginzburg-Landau energy on graphs, and derived an MBO scheme for solving the SSL problem.

Recent theoretical developments in threshold dynamics [6], [5], [8] have led to vast generalizations of the original MBO algorithm. The key to these new developments is the heat content energy, which gives a non-local approximation to the perimeter of a set [1], [15]. Generalizations of the heat content form a family of energies, essentially indexed by diffusion kernels, that are Lyapunov functionals for MBO type algorithms [6]. These energies give a natural and principled way to extend MBO schemes to a wide variety of situations, including segmentation problems on graphs.

This work represents the first exploration and extension of the theory developed in [6], [5], [8] to problems in machine learning and graph partitioning. Our main contribution is two new MBO schemes for the SSL problem, GHCMBO and GHCMBOS, based on the graph heat content energy (GHC) introduced in [8]. Our resulting schemes are novel in several ways. They generalize and simplify previous graph MBO schemes [9], [13], allowing virtually any graph diffusion process. GHC is a Lyapunov functional for our algorithms, thus we can guarantee unconditional stability and convergence to a local minimum. We find that our methods match or exceed the accuracy of other state of the art variational methods for SSL, while being much faster, more flexible, and easier to code.

2 Background and Notation

2.1 The Graphical Model

We consider the SSL data classification problem over the structure of an undirected weighted graph $G = (\mathcal{V}, W)$. $\mathcal{V}$ is the set of data points, and the weight matrix $W : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ is a symmetric matrix that describes the connection strength between any two points.

The datasets we consider in this work are collections of real vectors embedded in a high dimensional Euclidean space. A key assumption of machine learning is that the data is concentrated near a low dimensional manifold. Our goal is to reflect this manifold structure in our choice of weight matrix. Ideally, we would

like to weight points based on the geodesic distances between them, however this information is not readily available to us and would lead to a very dense weight matrix. Instead, we assume that the manifold is locally Euclidean, and only compute the k nearest neighbors of each point in the Euclidean metric. Computing just a small fraction of the distances ensures that W will be a sparse matrix, which will be essential for the fast performance of our algorithms.

Under these assumptions a popular choice for the weights are the Zelnick-Manor and Perona (ZMP) weight functions [22]:

$$W(x,y) = \exp\big(\frac{-d_E(x,y)^2}{\sigma(x)\sigma(y)}\big) \tag{1}$$

where d_E is the Euclidean distance and $\sigma(x), \sigma(y)$ are local scaling parameters for x, y respectively. We will construct our weight matrices using the ZMP weights, where we set $\sigma(x) = d_E(x, x_r)$ where x_r is the r^{th} nearest neighbor of x. To recover a symmetric matrix we simply set $W(x,y) \leftarrow \max(W(x,y), W(y,x))$.

It will be useful for us to have a notion of an approximate geodesic distance between points in the graph that are not nearest neighbors. With the structure of the weight matrix, we may compute approximations to the geodesic distance by traversing through paths in the graph. Let a path p in the graph be a sequence of vertices $\{x_1, \ldots, x_s\}$ such that $W(x_i, x_{i+1}) \neq 0$ for every $1 \leq i \leq s-1$. Let the length $\ell_q(p)$ of a path be

$$\ell_q(p) = \Big(\sum_{1\leq i\leq s-1} (-\log(W(x_i,x_{i+1})))^{q/2}\Big)^{1/q} = \Big(\sum_{1\leq i\leq s-1} \big(\frac{d_E(x_i,x_{i+1})}{\sqrt{\sigma(x)\sigma(y)}}\big)^q\Big)^{1/q} \tag{2}$$

Let $\pi(x,y)$ be the set of all paths from x to y. Then the q-geodesic distance between x and y, denoted $d_{G,q}(x,y)$, may be defined as

$$d_{G,q}(x,y) = \min_{p\in\pi(x,y)} \ell_q(p) \tag{3}$$

Given any subset $S \subset \mathcal{V}$ the distances $d_{G,q}(x,S) = \min_{z\in S} d(x,z)$ may be efficiently computed using pathfinding algorithms.

2.2 Semi-Supervised Data Classification

Given a set of data points $\mathcal{V}$, a fixed collection of labels $\{1, \ldots, N\}$, and a fidelity subset $F \subset \mathcal{V}$ of points whose labels are known, the semi-supervised data classification problem asks to correctly label the remaining points in $\mathcal{V} \setminus F$. Any solution of the problem is a partition $\mathbf{\Sigma} = (\Sigma_1, \ldots, \Sigma_N)$ of $\mathcal{V}$ where Σ_i is the set of points that are assigned label i. An N-phase partition of $\mathcal{V}$ may be represented as a function $u : \mathcal{V} \to \{\mathbf{e}_1, \ldots, \mathbf{e}_N\}$ where $\mathbf{e}_i \in \mathbb{R}^N$ is the i^{th} standard basis vector. The convex relaxation of this space is the set of functions $u : \mathcal{V} \to \mathcal{S}_N$, where $\mathcal{S}_N$ is the simplex

$$\mathcal{S}_N = \{\mathbf{p} \in [0,1]^N : \sum_{i=1}^N p_i = 1\} \tag{4}$$

A point $\mathbf{p} \in \mathcal{S}_N$ can be interpreted as a vector of probabilities, where p_i gives the confidence that a point should be assigned label i. We will denote the correct segmentation of the points as the function $\mathbf{u}^*$.

Variational approaches solve the problem by finding minimizers of energies of the form

$$E(\mathbf{u}) = R(\mathbf{u}) + \mathrm{Fid}(\mathbf{u}). \tag{5}$$

Here R is a regularizing term that is typically some relaxation of the weighted graph cut (6), and Fid is a term that incorporates the fidelity data F.

$$\mathrm{Cut}(\mathbf{\Sigma}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{x\in\Sigma_i}\sum_{y\notin\Sigma_i} W(x,y). \tag{6}$$

Given some constants $f_i(x)$, we will assume throughout that $\mathrm{Fid}(\mathbf{u})$ has the structure

$$\mathrm{Fid}(\mathbf{u}) = \sum_{i=1}^{N}\sum_{x\in\mathcal{V}} f_i(x)u_i(x) \tag{7}$$

3 The MBO Scheme

There are many possible relaxations of the weighted graph cut (6). Our approach is to model the graph cut with the graph heat content energy (GHC) introduced in [8]. The graph heat content is a family of energies indexed by the class of affinity matrices, symmetric non-negative matrices $A : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$. Every affinity matrix induces a (potentially unnormalized) diffusion process on the graph. Given an affinity matrix A, the graph heat content of a function $\mathbf{u} : \mathcal{V} \to \mathcal{S}_N$ is

$$\mathrm{GHC}(\mathbf{u}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{x,y\in\mathcal{V}} A(x,y)u_i(x)(1-u_i(y)). \tag{8}$$

If the affinity matrix A is the weight matrix W then GHC is a relaxation of the graph cut.

GHC is based on the continuum heat content energy (HC) defined in [6]. Given a translationally invariant m-dimensional domain D and a nonnegative convolution kernel K the heat content of a function $\mathbf{u} : D \to \mathcal{S}_N$ is

$$\mathrm{HC}_\varepsilon(\mathbf{u}) = \frac{1}{\varepsilon}\sum_{i=1}^{N}\int_D\int_{\mathbb{R}^m} u_i(x)(1-u_i(x+\varepsilon z))K(z)dzdx. \tag{9}$$

In the special case that $\mathbf{u}$ is a partition, the term inside the sum measures the amount of heat that diffuses out of phase i in time ε under the diffusion generated by K. For small values of ε, the amount of heat that escapes is proportional to the perimeter of phase i. Thus, the heat content gives a non-local approximation to total variation on partitions.

The authors of [6] showed that the approximation of the heat content to total variation becomes exact in the limit $\varepsilon \to 0$. In fact, as $\varepsilon \to 0$, the energy $\mathrm{HC}_\varepsilon(\mathbf{u})$ Gamma converges in $L^1(D)$ to $\sum_{i=1}^N \|\nabla u_i\|$ when $\mathbf{u}$ is a partition and to ∞ otherwise. Therefore, as $\varepsilon \to 0$, minimizers of the heat content energy approach partitions of minimal perimeter. This makes the heat content energy a natural choice for segmentation problems.

3.1 MBO via Linearization of the Heat Content

We now derive an MBO scheme for minimizing energies of the form

$$E(\mathbf{u}) = \mathrm{GHC}(\mathbf{u}) + \mathrm{Fid}(\mathbf{u}) \tag{10}$$

following the approach developed in [6] for the continuum heat content. The connection to the MBO algorithm can be seen by considering the variations of GHC at a configuration $\mathbf{u}$ in the direction of $\boldsymbol{\varphi}$.

$$\mathrm{GHC}(\mathbf{u}+\boldsymbol{\varphi}) = \mathrm{GHC}(\mathbf{u}) + \sum_{i=1}^{N} \sum_{x\in\mathcal{V}} \varphi_i(x) \sum_{y\in\mathcal{V}} A(x,y)(1-u_i(y)) + Q(\boldsymbol{\varphi}). \tag{11}$$

where

$$Q(\boldsymbol{\varphi}) = -\sum_{i=1}^{N} \sum_{x,y\in\mathcal{V}} A(x,y)\varphi_i(x)\varphi_i(y) \tag{12}$$

When A is PSD the quadratic term Q is always negative and GHC is concave. Fid is linear, so E is also concave. Thus,

$$E(\mathbf{u}+\boldsymbol{\varphi}) - E(\mathbf{u}) \leq \sum_{i=1}^{N} \sum_{x\in\mathcal{V}} \varphi_i(x)\Big(f_i(x) + \sum_{y\in\mathcal{V}} A(x,y)(1-u_i(y))\Big) \tag{13}$$

The right hand side of equation (13) is the linearization of E at the function $\mathbf{u}$. The concavity of E implies that we may obtain a configuration of lower energy $\mathbf{u}+\boldsymbol{\varphi}$ by minimizing the linearization over valid directions $\boldsymbol{\varphi}$. The only restriction on $\boldsymbol{\varphi}$ is that $\mathbf{u}+\boldsymbol{\varphi}$ must be an element of the domain of E, i.e. $\mathbf{u}(x)+\boldsymbol{\varphi}(x) \in \mathcal{S}_N$ for all x. It will always be possible to find a valid direction $\boldsymbol{\varphi}$ such that $E(\mathbf{u}+\boldsymbol{\varphi}) < E(\mathbf{u})$, as long as $\mathbf{u}$ is not a local minimum of E. This allows us to take extremely large steps through the configuration space, while still dissipating the energy E.

Iterating the minimization procedure leads to the following algorithm, GHCMBO, which is a graph analogue of the MBO scheme of alternating diffusion with pointwise thresholding. Since the configuration space is compact, the above argument implies that the iteration must converge to a local minimum of E. Our scheme's guarantee of energy dissipation and convergence represents a significant theoretical advancement over previous graph MBO schemes for the SSL problem [9], [13].

Algorithm 1: GHCMBO $\mathbf{u}^{k+1}$ is obtained from $\mathbf{u}^k$ as follows:

1. Diffusion by A:

$$\psi_i^{k+1}(x) = \sum_{y \in \mathcal{V}} A(x,y) u_i^k(y) \quad \text{for} \quad 1 \leq i \leq N \tag{14}$$

2. Thresholding:

$$i^{k+1}(x) = \arg\min_j f_j(x) - \psi_j^{k+1}(x). \tag{15}$$

$$\mathbf{u}(x) = \mathbf{e}_{i^{k+1}(x)} \tag{16}$$

GHCMBO has several appealing properties that are apparent from the structure of the algorithm. The thresholding step (16) implies that the algorithm produces a partition $\mathbf{u} : \mathcal{V} \to \{\mathbf{e}_1, \ldots, \mathbf{e}_N\}$ at every iteration. Therefore, our variational method always produces a combinatorial solution. Furthermore, since $\mathbf{u}^k$ is always a partition, computing the vector $\boldsymbol{\psi}^{k+1}(x) = \left(\psi_1^{k+1}(x), \ldots, \psi_N^{k+1}(x)\right)$ requires just $\deg_0(x)$ additions, where $\deg_0(x)$ counts the number of nonzero entries of A in row x. When A is sparse, this implies that each iteration has low computational complexity. The combination of simple computations and large step sizes makes GHCMBO an extremely fast algorithm.

To adapt GHCMBO to the problem at hand, we need to construct a PSD affinity matrix A that is related to the weighted graph structure $G = (\mathcal{V}, W)$. The simplest choice is to take $A = W^2$. Another possible choice is the graph heat kernel $H_t = e^{-tL}$, where L is the symmetric normalized graph Laplacian and $t > 0$. However, this adds a parameter t, and the heat kernel is typically not sparse. Previous graph MBO schemes [9], [13] have been restricted to diffusion by the heat equation and associated kernels. In addition to energy dissipation, one of the chief advantages of our approach is the ability to more freely choose a diffusion generated by a sparse matrix.

A natural question to ask is when can W itself be chosen for A. W is a desirable choice, as W is the sparsest matrix that still retains the full structure of the graph. Furthermore, when $A = W$ the graph heat content is a relaxation of the weighted graph cut. In general, one cannot expect that W as constructed in (1) will be positive semi-definite. An example in [8] (with no fidelity term) shows that for a given binary partition and a very natural nonnegative but not positive semi-definite weight matrix, GHCMBO gets trapped in a 2-periodic loop between two different configurations. One of the configurations has a higher energy than the other, thus there are cases where A is not PSD and GHCMBO both increases the energy and gets stuck in a non-productive loop.

It is possible however, to modify the algorithm so that the energy is guaranteed to decrease for a much wider class of matrices. In [8] it was shown that one can guarantee dissipation of GHC for any affinity matrix A by computing convolutions slightly more often. In particular, this implies that we may take $A = W$. The key feature of this new scheme, GHCMBOS, is that only one phase is al-

lowed to shrink at a time. Although GHCMBOS has a more restrictive update rule, arguments in [8] show that the algorithm does not terminate prematurely. As long as the diagonal entries $A(x,x)$ are strictly positive, GHCMBOS halts on a configuration $\mathbf{u}$ if and only if $\mathbf{u}$ is a local minimum. The modified minimization scheme GHCMBOS is given in Algorithm 2 below.

Algorithm 2: GHCMBOS $\mathbf{u}^{k+1}$ is obtained from a sequence of substeps $\mathbf{u}^{k,\ell}$ indexed by the labels $1 \leq \ell \leq N$, with $\mathbf{u}^{k,0} = \mathbf{u}^k$ and $\mathbf{u}^{k+1} = \mathbf{u}^{k,N}$. Then, $\mathbf{u}^{k,\ell}$ is obtained from $\mathbf{u}^{k,\ell-1}$ as follows:

1. Diffusion by A:

$$\psi_i^{k,\ell}(x) = \sum_{y \in \mathcal{V}} A(x,y) u_i^{k,\ell-1}(y) \quad \text{for} \quad 1 \leq i \leq N \tag{17}$$

2. Restricted Thresholding:

$$i^{k,\ell}(x) = \arg\min_j f_j(x) - \psi_j^{k,\ell}(x). \tag{18}$$

$$\mathbf{u}^{\mathbf{k},\ell}(x) = \mathbf{e}_{i^{k,\ell}(x)} \quad \text{if} \quad \mathbf{u}^{\mathbf{k},\ell-\mathbf{1}}(x) = \mathbf{e}_\ell \tag{19}$$

$$\mathbf{u}^{\mathbf{k},\ell}(x) = \mathbf{u}^{\mathbf{k},\ell-\mathbf{1}}(x) \quad \text{otherwise} \tag{20}$$

At the ℓ^{th} substep the quantities (17-18) only need to be computed for x in the ℓ^{th} phase. Thus, the complexity of a full step of GHCMBOS is comparable to the complexity of a step of GHCMBO. In our experiments GHCMBOS runs faster than GHCMBO (see Section 4). The sparsity of W as compared to W^2 offsets any increase in the computational complexity of GHCMBOS.

3.2 A Fidelity Term Based on Graph Geodesics

Thus far, we have not described how to construct the fidelity term $\text{Fid}(\mathbf{u}) = \sum_{i=1}^N \sum_{x \in \mathcal{V}} f_i(x) u_i(x)$ from the fidelity data F. The simplest way is to impose a penalty on points whose labeling differs from the correct labeling, $\mathbf{u}^*$, on F. Thus, we may take $f_i(x) = \lambda(1 - u_i^*(x))$ for $x \in F$ and zero for all other x. When λ is taken to infinity, the fidelity term becomes a hard constraint. We can easily incorporate the hard constraint into the minimization algorithms GHCMBO and GHCMBOS by simply not updating the points in the fidelity set.

If $\text{Fid}(\mathbf{u})$ is only active on fidelity nodes, the correct labeling $\mathbf{u}^*$ may be difficult to find in the energy landscape, especially when the size of F is very small compared to $\mathcal{V}$. For example, if F is small, then the global minimum of the energy will be near a partition that assigns all points outside of the fidelity set to the same label. For this reason, we introduce a fidelity term that is active on all of the nodes. Our approach is inspired by the region force in [20]. There the authors introduce a linear penalty term where $f_i(x)$ is based on the diffusion distance [4] between x and elements of the fidelity set with labeling i.

Our fidelity term instead uses the graph geodesic distance defined in equation (3). For nodes in the fidelity set we use the hard constraint described above. For $x \notin F$, and some positive constant τ we take,

$$f_i(x) = -\tau \exp\big(-d_{G,2}(x, F_i)^2\big). \tag{21}$$

where F_i is the set of fidelity points labeled i. We find that our fidelity term outperforms the diffusion distance fidelity term of [20]. On the MNIST data set, the initialization produced by labeling $x \in \mathcal{V} \setminus F$ according to $i(x) = \arg\min_j f_j(x)$ is much closer to the correct labeling, when using (21) instead of the term in [20] (see Table 4).

4 Experimental Results

We test the two variants of our scheme GHCMBO and GHCMBOS with the fidelity term (21). In GHCMBO we take $A = W^2$, and in GHCMBOS we take $A = W$. For all experiments we set $\tau = 0.1$ in the fidelity term.

The algorithm stops whenever the relative energy change falls below some threshold. If we let E_k denote the energy at the k^{th} step then the algorithms end when $(E_k - E_{k+1})/E_{k+1} < \eta$ for some small positive η. In all of our experiments we have set $\eta = 10^{-6}$. The labeling on non-fidelity nodes $x \in \mathcal{V} \setminus F$ is initialized by setting $i(x) = \arg\min_j f_j(x)$.

We test our algorithm on several benchmark machine learning datasets: Three Moons, MNIST, Opt-Digits, and COIL. All experiments were run using C code on a single core of an Intel i5-4250U processor at 1.30 GHz with 4GB RAM. k-nearest neighbors were calculated using the kd-tree code in the VLFeat library. Table 1 shows the timing information for constructing the weight matrices using VLFeat. All of our subsequent timing results for GHCMBO and GHCMBOS include the time it takes to calculate the fidelity coefficients $f_i(x)$ and run the iterations (14-16) or (17-20). For every dataset we averaged our results over 100 trials at different fixed fidelity set sizes. In each trial, the points in the fidelity set were chosen at random and the number of points in each class was allowed to be random.

We compare our results to previous graph MBO schemes (MBO eigenvectors [9], HKPR1/2 MBO [13]) and the total variation based convex method (TVRF [20]). The results reported for the other methods are taken from their respective papers.

4.1 Three Moons

The Three Moons synthetic data set consists of three half circles embedded into $\mathbb{R}^{100}$ with Gaussian noise. The standard construction is built from circles centered at $(0,0), (3,0), (1.5, 0.4)$ with radii of 1,1, and 1.5 respectively. The first two half circles lie in the upper half plane, while the third circle lies in the lower half plane. The circles are then embedded into $\mathbb{R}^{100}$ by setting the remaining

98 coordinates to zero. Finally, Gaussian noise with mean zero and standard deviation 0.14 is added to each of the 100 coordinates.

We construct the dataset by sampling 500 points from each of the three circles, for a total of 1500 points. The weight matrix was built using the 15 nearest neighbors with local scaling by the 7^{th} nearest neighbor. We tested fidelity sets of size 25, 50 and 75. Results for this dataset are recorded in Table 2. GHCMBO and GHCMBOS outperform the methods of [20] and are comparable to the accuracy of [9]. Both of our methods are nearly two orders of magnitude faster than [9].

4.2 MNIST

MNIST is a database of 70,000 grayscale 28×28 pixel images of handwritten digits (0-9). Each of the digits is centered and size normalized. We combine them to create a single set of 70,000 images to test against. We perform no preprocessing on the images.

The weight matrix is calculated using the 15 nearest neighbors with local scaling based on the 7^{th} nearest neighbor. We tested fidelity sets of size 150, 300, 450 and 2500. Results for this dataset are recorded in Table 3. GHCMBO outperforms all of the other methods while being 1.8 to 4 orders of magnitude faster. GHCMBOS is even faster than GHCMBO, but is less accurate at the smaller fidelity set sizes.

In Table 4 we compare our fidelity term (21) with the diffusion distance fidelity term used in [20]. Each point is labeled according to $i(x) = \arg\min_j f_j(x)$ and then the accuracy is measured without running any further algorithms. Our fidelity term is significantly more accurate than the fidelity term in [20].

4.3 Opt-Digits

Opt-Digits is a database of 5620 handwritten digits [11]. The data is recorded as an 8×8 integer matrix, where each element is between 0 and 16.

We construct the weight matrix using the 15 nearest neighbors and local scaling by the 7^{th} nearest neighbor. We tested fidelity sets of size 50, 100, and 150. Results for this dataset are recorded in Table 5. Our methods are comparable or superior to the results of [20].

4.4 COIL

The Columbia Object Image Library (COIL-100) is a database of 128×128 pixel color images of 100 different objects photographed at various different angles [17]. In [18] the authors processed the COIL images to create a more difficult benchmark set. The red channel of each image is downsampled to 16×16 pixels by averaging over blocks of 8×8 pixels. The images are then further distorted and downsampled to create 241 dimensional feature vectors. Then 24 of the objects are randomly selected and randomly partitioned into 6 different

classes. Discarding 38 images from each class leaves 250 images per class for a total of 1500 points.

We construct the weight matrix using the 4 nearest neighbors and local scaling by the 4^{th} nearest neighbor. We tested fidelity sets of size 50, 100, and 150. Results for this dataset are recorded in Table 6. Both GHCMBO and GHCMBOS considerably outperform all of the other methods. In addition, our approaches are anywhere from 200 to nearly 100,000 times faster than the other methods.

5 Conclusion

We have presented two MBO schemes, GHCMBO and GHCMBOS, for solving the SSL problem on a weighted graph. Our schemes are based on the graph heat content energy (GHC) and the theory developed in the series of papers [6], [5], [8]. We solve the SSL problem by minimizing an energy that combines GHC with a linear fidelity term based on graph geodesics, inspired by the region force in [20]. GHC depends on the choice of affinity matrix A, which induces a diffusion process on the graph. If A is PSD then GHCMBO decreases the energy at every step, while GHCMBOS minimizes the energy for all affinity matrices. Our approach considerably generalizes and simplifies previous SSL graph MBO schemes [9], [13]. The guarantee of energy dissipation and convergence to local minima is a new and important theoretical advance for SSL graph MBO schemes.

Experimental results on benchmark datasets shows that both GHCMBO and GHCMBOS produce results with comparable or superior accuracy to other state of the art methods [9], [13], [20]. In addition, our schemes were considerably faster. Our slower algorithm, GHCMBO, was nearly two orders of magnitude faster on every dataset. Our algorithms are so fast because we are free to choose diffusions generated by extremely sparse matrices, and take very large step sizes through the configuration space.

Unlike the basic MBO scheme, the new variants discussed in this paper extend to very general multiphase situations where the interaction between each phase pair may be treated differently. In a future work we plan to apply this idea to the SSL problem, using the fidelity data to learn the most favorable set of pairwise interactions.

Acknowledgments

The author is grateful to Selim Esedoḡlu for helpful comments and suggestions. The author was supported by NSF DMS-1317730.

References

1. G. Alberti and G. Bellettini. A non-local anisotropic model for phase transitions: asymptotic behavior of rescaled energies. *European J. Appl. Math.*, 9:261–284, 1998.

Table 1: Benchmark datasets

Dataset	Dimension	Points	Classes	W construction timing (s)
Three Moons	100	1,500	3	0.025
MNIST	784	70,000	10	149.04
Opt-Digits	64	5620	10	2.03
COIL	241	1500	6	0.33

Table 2: Three Moons

Method	$\|F\|$=25	$\|F\|$=50	$\|F\|$=75	Timing (ms)
TVRF [20]	96.4%	98.2%	98.6%	–
MBO eigenvectors [9]	–	–	**99.12**%	344
GHCMBO	97.45%	98.61%	98.94%	4.1
GHCMBOS	**97.81**%	**98.93**%	99.08%	3.1

Table 3: MNIST

Method	$\|F\|$=150	$\|F\|$=300	$\|F\|$=450	$\|F\|$=2500	Timing (s)
TVRF [20]	94.6%	96.6%	96.8%	–	61
HKPR1 MBO [13]	–	–	–	97.52%	22.3
HKPR2 MBO [13]	–	–	–	97.48%	4,428
MBO eigenvectors [9]	–	–	–	96.91%	1,699
GHCMBO	**95.97**%	**96.81**%	**97.09**%	**97.54**%	0.30
GHCMBOS	92.91%	95.33%	96.32%	97.27%	0.17

Table 4: Comparing Fidelity terms on MNIST

Method	$\|F\|$=150	$\|F\|$=300	$\|F\|$=450	Timing (s)
Fidelity only [20]	35.5%	52.3%	71.5%	0.4
Fidelity only (21)	**84.93**%	**88.61**%	**90.90**%	0.13

Table 5: Opt-Digits

Method	$\|F\|$=50	$\|F\|$=100	$\|F\|$=150	Timing (ms)
TVRF [20]	**95.9**%	97.2%	**98.3**%	–
GHCMBO	95.68%	**97.63**%	98.10%	15.4
GHCMBOS	94.20%	96.30%	97.28%	11.0

Table 6: COIL

Method	$\|F\|$=50	$\|F\|$=100	$\|F\|$=150	Timing (ms)
TVRF [20]	80.3%	90.0%	91.7%	–
MBO eigenvectors [9]	–	–	91.46%	220
HKPR1 MBO [13]	–	–	91.09%	1,000
HKPR2 MBO [13]	–	–	91.23%	92,000
GHCMBO	**83.01**%	92.24%	94.30%	1.00
GHCMBOS	82.96%	**92.30**%	**94.34**%	0.76

2. X. Bresson, T. Chan, X. Tai, and A. Szlam. Multi-class trans- ductive learning based on l1 relaxations of cheeger cut and mumford-shah-potts model. *Journal of Mathematical Imaging and Vision*, 49(1):191–201, August 2013.
3. J. Cheeger. A lower bound for the smallest eigenvalue of the Laplacian. In *Problems in Analysis*, pages 195–199. 1970.
4. R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *PNAS*, 2005.
5. M. Elsey and S. Esedoḡlu. Threshold dynamics for anisotropic surface energies. Technical report, UM, 2016. Under review.
6. S. Esedoḡlu and F. Otto. Threshold dynamics for networks with arbitrary surface tensions. *Communications on Pure and Applied Mathematics*, 68(5):808–864, 2015.
7. S. Esedoḡlu and Y.-H. Tsai. Threshold dynamics for the piecewise constant Mumford-Shah functional. *Journal of Computational Physics*, 211(1):367–384, 2006.
8. Selim Esedoḡlu and Matt Jacobs. Convolution kernels, and stability of threshold dynamics methods. Technical report, University of Michigan, 2016.
9. C. Garcia-Cardona, E. Merkurjev, A. L. Bertozzi, A. Flenner, and A. G. Percus. Multiclass data segmentation using diffuse interface methods on graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1600–1613, 2014.
10. M. Hein and S. Setzer. Beyond spectral clustering - tight relaxations of balanced graph cuts. In *Advances in Neural Information Processing Systems 24 (NIPS 2011)*, 2011.
11. C Kaynak. Methods of combining multiple classifiers and their applications to handwritten digit recognition. Master's thesis, Institute of Graduate Studies in Science and Engineering, Bogazici University, 1995.
12. J. Malik and J. Shi. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.
13. E. Merkurjev, A. Bertozzi, and F. Chung. A semi-supervised heat kernel pagerank mbo algorithm for data classification. *Submitted*, 2016.
14. B. Merriman, J. K. Bence, and S. J. Osher. Diffusion generated motion by mean curvature. In J. Taylor, editor, *Proceedings of the Computational Crystal Growers Workshop*, pages 73–83. AMS, 1992.
15. M. Miranda, D. Pallara, F. Paronetto, and M. Preunkert. Short-time heat flow and functions of bounded variation in $\mathbb{R}^N$. *Ann. Fac. Sci. Toulouse, Mathematiques*, 16(1):125–145, 2007.
16. D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42:577–685, 1989.
17. S.A. Nene, S.K. Nayar, and H. Murase. Columbia object image library (coil-100). Technical report, Columbia University, 1996.
18. Alexander Zien Olivier Chapelle, Bernhard Scholkopf. *Semi-Supervised Learning*. The MIT Press, 2006.
19. O. Scherzer, editor. *Handbook of Mathematical Methods in Imaging*. Springer, 2011.
20. K. Yin, Xue-Cheng Tai, and S. J. Osher. An effective region force for some variational models for learning and clustering. Technical report, UCLA, 2016.
21. S. X. Yu and J. Shi. Multiclass spectral clustering. In *Ninth IEEE International Coference on Computer Vision, 2003*, volume 1, pages 313–319, October 2003.
22. Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. *Advances in neural information processing systems*, 2004.