Graph Partitioning Based on Link Distributions

Bo Long and **Mark (Zhongfei) Zhang**
Computer Science Dept., SUNY Binghamton
Binghamton, NY 13902
{blong1, zzhang}@binghamton.edu

Philip S. Yu
IBM Watson Research Center
19 skyline Drive, Hawthorne, NY 10532
psyu@us.ibm.com

Abstract

Existing graph partitioning approaches are mainly based on optimizing edge cuts and do not take the distribution of edge weights (link distribution) into consideration. In this paper, we propose a general model to partition graphs based on link distributions. This model formulates graph partitioning under a certain distribution assumption as approximating the graph affinity matrix under the corresponding distortion measure. Under this model, we derive a novel graph partitioning algorithm to approximate a graph affinity matrix under various Bregman divergences, which correspond to a large exponential family of distributions. We also establish the connections between edge cut objectives and the proposed model to provide a unified view to graph partitioning.

Introduction

Graph partitioning is an important problem in many machine learning applications, such as circuit partitioning, VLSI design, task scheduling, bioinformatics, and social network analysis. Existing graph partitioning approaches are mainly based on edge cut objectives, such as Kernighan-Lin objective (Kernighan & Lin 1970), normalized cut (Shi & Malik 2000), ratio cut (Chan, Schlag, & Zien 1993), ratio association(Shi & Malik 2000), and min-max cut (Ding *et al.* 2001).

The main motivation of this study comes from the fact that graphs from different applications may have very different statistical characteristics for their edge weights. Specifically, the graphs may have very different link distributions, where the link distribution refers to the *distribution of edge weights* in a graph. For example, in a graph with binary weight edges, the link distribution can be modeled as a Bernoulli distribution; in a graph with edges of real value weights, the link distribution may be modeled as an exponential distribution or a normal distribution. This fact naturally raises the following questions: is it appropriate to use edge cut objectives for all kinds of graphs with different link distributions? If not, what kinds of graphs the edge cut objectives work well for? How to make use of link distributions to partition different types of graphs? This paper attempts to answer these questions.

Another motivation of this study is to derive an effective algorithm to improve the existing graph partitioning algorithms on some aspects. For example, the popular spectral approaches involve expensive eigenvector computation and extra post-processing on eigenvectors to obtain the partitioning; the multi-level approaches such as METIS (Karypis & Kumar 1998) restrict partitions to have an equal size.

In this paper, we propose a general model to partition graphs based on link distributions. The key idea is that by viewing the link distribution of a graph as a mixture of link distributions within and between different partitions, we can learn the mixture components to find the partitioning of the graph. The model formulates partitioning a graph under a certain distribution assumption as approximating the graph affinity matrix under the corresponding distortion measure. Second, under this model, we derive a novel graph partitioning algorithm to approximate a graph affinity matrix under various Bregman divergences, which correspond to a large exponential family distributions. Our theoretic analysis and experiments demonstrate the the potential and effectiveness of the proposed model and algorithm. Third, we also establish the connections between the proposed model and the edge cut objectives to provide a unified view to graph partitioning.

We use the following notations in this paper. Capital letters such as A, B and C denote matrices; A_{ij} or $[A]_{ij}$ denote the (i, j)th element in A; small boldface letters such as $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ denote column vectors. A graph is denoted by $G = (\mathcal{V}, \mathcal{E}, A)$, which is made up of a set of vertices $\mathcal{V}$ and a set of edges $\mathcal{E}$, and the affinity matrix A of dimension $|\mathcal{V}| \times |\mathcal{V}|$, whose entries represent the weights of the edges.

Related Work

Graph partitioning divides a graph into subgraphs by finding the best edge cuts of the graph. Several edge cut objectives, such as the average cut (Chan, Schlag, & Zien 1993), average association (Shi & Malik 2000), normalized cut (Shi & Malik 2000), and min-max cut (Ding *et al.* 2001), have been proposed. Various spectral algorithms have been developed for these objective functions (Chan, Schlag, & Zien 1993; Shi & Malik 2000; Ding *et al.* 2001). These algorithms use the eigenvectors of a graph affinity matrix, or a matrix derived from the affinity matrix, to partition the graph.

Multilevel methods have been used extensively for graph partitioning with the Kernighan-Lin objective, which attempts to minimize the cut in the graph while maintaining equal-sized clusters (Bui & Jones 1993; Hendrickson & Leland ; Karypis & Kumar 1998). In multilevel algorithms, the graph is repeatedly coarsened level by level until only a small number of nodes are left. Then, an initial partitioning on this small graph is performed. Finally, the graph is uncoarsened level by level, and at each level, the partitioning from the previous level is refined using a refinement algorithm.

Recently, graph partitioning with an edge cut objective has been shown to be mathematically equivalent to an appropriately weighted kernel k-means objective function (Dhillon, Guan, & Kulis 2004; 2005). Based on this equivalence, the weighted kernel k-means algorithm has been proposed for graph partitioning (Dhillon, Guan, & Kulis 2004; 2005). Yu, Yu, & Tresp (2005) propose graph-factorization clustering for the graph partitioning, which seeks to construct a bipartite graph to approximate a given graph. Long *et al.* (2006) propose a framework of relation summary network to cluster K-partite graphs.

Another related field is unsupervised learning with Bregman divergences (S.D.Pietra 2001; Wang & Schuurmans 2003). Banerjee *et al.* (2004b) generalizes the classic k-means to Bregman divergences. A generalized co-clustering framework is presented by Banerjee *et al.* (2004a) wherein any Bregman divergence can be used in the objective function.

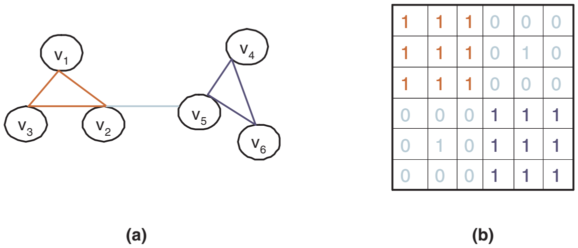
Model Formulation

We first define the link distribution as the follows.

Definition 1. *Given a graph $G = (\mathcal{V}, \mathcal{E}, A)$, the link distribution $f_{\mathcal{V}_1\mathcal{V}_2}$ is the probability density of edge weights between nodes in $\mathcal{V}_1$ and $\mathcal{V}_2$, where $\mathcal{V}_1, \mathcal{V}_2 \subseteq \mathcal{V}$.*

Based on Definition 1, the link distribution for the whole graph G is $f_{\mathcal{V}\mathcal{V}}$. The model assumption is that if G has k disjoint partitions $\mathcal{V}_1, \ldots, \mathcal{V}_k$, then $f_{\mathcal{V}\mathcal{V}} = \sum_{1 \le i \le j \le k} \pi_{ij} f_{\mathcal{V}_i\mathcal{V}_j}$, where π_{ij} is the mixing probability such that $\sum_{1 \le i \le j \le k} \pi_{ij} = 1$. Basically, the assumption states that the link distribution of a graph is a mixture of the link distributions within and between partitions. The intuition behind the assumption is that the vertices within the same partition are related in a (statistically) similar way to each other and the vertices from different partitions are related in different ways to each other from those within the same partition. In Section 5, we show that the traditional edge cut objectives also implicitly make this assumption under a normal distribution with extra constraints.

Let us have an illustrative example. Figure 1(a) shows a graph of six vertices and seven unit weight edges. It is natural to partition the graph into two components, $\mathcal{V}_1 = \{v_1, v_2, v_3\}$ and $\mathcal{V}_2 = \{v_4, v_5, v_6\}$. The link distribution of the whole graph can be modeled as a Bernoulli distribution $f_{\mathcal{V}\mathcal{V}}(x; \theta_{\mathcal{V}\mathcal{V}})$ with the parameter $\theta_{\mathcal{V}\mathcal{V}} = \frac{7}{15}$ (the number of edges in the graph is 7 and the number of possible edges is 15). Similarly, the link distributions for edges within and between $\mathcal{V}_1$ and $\mathcal{V}_2$ are Bernoulli distributions, $f_{\mathcal{V}_1\mathcal{V}_1}(x; \theta_{\mathcal{V}_1\mathcal{V}_1})$ with $\theta_{\mathcal{V}_1\mathcal{V}_1} = 1$, $f_{\mathcal{V}_2\mathcal{V}_2}(x; \theta_{\mathcal{V}_2\mathcal{V}_2})$ with $\theta_{\mathcal{V}_2\mathcal{V}_2} = 1$, and $f_{\mathcal{V}_1\mathcal{V}_2}(x; \theta_{\mathcal{V}_1\mathcal{V}_2})$ with $\theta_{\mathcal{V}_1\mathcal{V}_2} = \frac{1}{9}$. Note that $f_{\mathcal{V}\mathcal{V}}$ is a mixture of $f_{\mathcal{V}_1\mathcal{V}_1}$, $f_{\mathcal{V}_2\mathcal{V}_2}$ and $f_{\mathcal{V}_1\mathcal{V}_2}$, which can be verified by $\theta_{\mathcal{V}\mathcal{V}} = \frac{3}{15}\theta_{\mathcal{V}_1\mathcal{V}_1} + \frac{9}{15}\theta_{\mathcal{V}_1\mathcal{V}_2} + \frac{3}{15}\theta_{\mathcal{V}_2\mathcal{V}_2}$ (the mixing probability for $f_{\mathcal{V}_1\mathcal{V}_2}$, $\frac{9}{15}$, follows the fact that the number of possible edges between $\mathcal{V}_1$ and $\mathcal{V}_2$ is 9; similarly for other proportion probabilities).



1	1	1	0	0	0
1	1	1	0	1	0
1	1	1	0	0	0
0	0	0	1	1	1
0	1	0	1	1	1
0	0	0	1	1	1

Figure 1: A graph with two partitions (a) and its graph affinity matrix (b).

Learning mixture components of the link distribution of a graph is much more difficult than learning a traditional mixture model, since the graph structure needs to be considered, i.e., our goal is to find the mixture components associated with subgraphs and not just to simply draw the similar edges from anywhere in the graph to form a component. For example, in Figure 1(a), without considering the graph structure, the edge weights from two partitions $\mathcal{V}_1$ and $\mathcal{V}_2$ cannot be separated. To tackle this difficulty, we model the problem based on the graph affinity matrix, which contains all the information for a graph.

Figure 1(b) shows the graph affinity matrix for the graph in Figure 1(a). We observe that if the vertices within the same partition are arranged together, the edge weights within and between partitions form the diagonal blocks and off-diagonal blocks, respectively. Hence, learning the link distribution in a graph is equivalent to learning different distributions for non-overlapping blocks in the graph affinity matrix. To estimate the sufficient statistic for each block, we need to solve the problem of likelihood maximization. It is shown that maximizing likelihood under a certain distribution corresponds to minimizing distance under the corresponding distortion measure (Collins, Dasgupta, & Reina 2001). For example, the normal distribution, Bernoulli distribution, multinomial distribution and exponential distribution correspond to Euclidean distance, logistic loss, KL-divergence and Itakura-Satio distance, respectively. Therefore, learning the distributions of the blocks in a graph affinity matrix can be formulated as approximating the affinity matrix under a certain distortion measure. Formally, we define graph partitioning as the following optimization problem of matrix approximation.

Definition 2. *Given a graph $G = (\mathcal{V}, \mathcal{E}, A)$ where $A \in \mathbb{R}^{n \times n}$, a distance function $\mathfrak{D}$, and a positive integer k, the optimized partitioning is given by the minimization,*

$$\min_{C \in \{0,1\}^{n \times k}, B \in \mathbb{R}^{k \times k}} \mathfrak{D}(A, CBC^T), \tag{1}$$

where $C \in \{0,1\}^{n \times k}$ is an indicator matrix such that $\sum_j C_{ij} = 1$, i.e., $C_{ij} = 1$ indicates that the ith vertex belongs to the jth partition, and $\mathfrak{D}$ is a separable distance function such that $\mathfrak{D}(X, Y) = \sum_{i,j} \mathfrak{D}(X_{ij}, Y_{ij})$.

We call the model in Definition 2 as the Graph Partitioning with Link Distribution (GPLD). GPLD provides not only the partitioning of the given graph, which is denoted by the *partition indicator matrix* C, but also the *partition representative matrix* B, which consists of the sufficient statistics for edge weights within and between partitions. For example, $B = \left[\begin{smallmatrix} 1 & 1/9 \\ 1/9 & 1 \end{smallmatrix}\right]$ for the example in Fig 1(b). B also provides an intuition about the quality of the partitioning, since the larger the difference between the diagonal and the off-diagonal elements, the better the partitions are separated. Note that GPLD does not restrict A to be symmetric or non-negative. Hence, it is possible to apply GPLD to directed graphs or graphs with negative weights, though in this paper our main focus is undirected graphs with non-negative weights.

Algorithm Derivation

First we derive an algorithm for GPLD model based on the most popular distance function, Euclidean distance function. Under Euclidean distance function, our task is

$$\min_{C\in\{0,1\}^{n\times k}, B\in\mathbb{R}^{k\times k}} ||A - CBC^T||^2. \tag{2}$$

We prove the following theorem which is the basis of our algorithm.

Theorem 3. *If $C \in \{0,1\}^{n\times k}$ and $B \in \mathbb{R}_+^{k\times k}$ is the optimal solution to the minimization in* (2)*, then*

$$B = (C^TC)^{-1}C^TAC(C^TC)^{-1}. \tag{3}$$

Proof. The objective function in Definition 2 can be expanded as follows.

$$\begin{aligned} L &= ||A - CBC^T||^2 \\ &= \mathrm{tr}((A - CBC^T)^T(A - CBC^T)) \\ &= \mathrm{tr}(A^TA) - 2\mathrm{tr}(CBC^TA) + \mathrm{tr}(CBC^TCBC^T) \end{aligned}$$

Take the derivative with respect to B, we obtain

$$\frac{\partial L}{\partial B} = -2C^TBC + 2C^TCBC^TC. \tag{4}$$

Solve $\frac{\partial L}{\partial B} = 0$ to obtain

$$B = (C^TC)^{-1}C^TAC(C^TC)^{-1}; \tag{5}$$

This completes the proof of the theorem. □

Based on Theorem 3, we propose an alternative optimization algorithm, which alternatively updates B and C until convergence. We first fix C and update B. Eq (3) in Theorem 3 provides an updating rule for B,

$$B = (C^TC)^{-1}C^TAC(C^TC)^{-1}. \tag{6}$$

This updating rule can be implemented more efficiently than it appears. First, it does not really involve computing inverse matrices, since C^TC is a special diagonal matrix with the size of each cluster on its diagonal such that $[C^TC]_{pp} = |\pi_p|$, where $|\pi_p|$ denotes the size of the pth partitioning; second, the product of C^TAC can be calculated without normal matrix multiplication, since C is an indicator matrix.

Then, we fix B and update C. Since each row of C is an indicator vector with only one element equal to 1, we adopt the re-assignment procedure to update C row by row. To determine which element of the hth row of C is equal to 1, for $p = 1, \ldots, k$, each time we let $C_{hp} = 1$ and compute the objective function $L = ||A - CBC^T||^2$, which is denoted as L_p, then

$$C_{hp^*} = 1 \text{ for } p^* = \arg\min_p L_p \tag{7}$$

Note that when we update the hth row of C, the necessary computation involves only the hth row or column of A and CBC^T.

Therefore, updating rules (6) and (7) provide a new graph partitioning algorithm, GPLD under Euclidean distance.

Presumably for a specific distance function used in Definition 2, we need to derive a specific algorithm. However, a large number of useful distance functions, such as Euclidean distance, generalized I-divergence, and KL divergence, can be generalized as the Bregman divergences (S.D.Pietra 2001; Banerjee *et al.* 2004b), which correspond to a large number of exponential family distributions. Moreover, the nice properties of Bregman divergences make it easy to generalize updating rules (6) and (7) to all Bregman divergences. The definition of a Bregman divergence is given as follows.

Definition 4. *Given a strictly convex function, $\phi : S \mapsto \mathbb{R}$, defined on a convex set $S \subseteq \mathbb{R}^d$ and differentiable on the interior of S, int(S), the Bregman divergence $D_\phi : S \times$ int(S) $\mapsto [0, \infty)$ is defined as*

$$D_\phi(x, y) = \phi(x) - \phi(y) - (x - y)^T\nabla\phi(y), \tag{8}$$

where $\nabla\phi$ is the gradient of ϕ.

Table 1 shows a list of popular Bregman divergences and their corresponding Bregman convex functions. The following Theorem provide an important property of Bregman divergence.

Theorem 5. *Let X be a random variable taking values in $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^n \subset S \subseteq \mathbb{R}^d$ following v. Given a Bregman divergence $D_\phi : S \times$ int(S) $\mapsto [0, \infty)$, the problem*

$$\min_{s\in S} E_v[D_\phi(X, s)] \tag{9}$$

has a unique minimizer given by $s^* = E_v[X]$.

The proof of Theorem 5 is omitted (please refer (S.D.Pietra 2001; Banerjee *et al.* 2004b)). Theorem 5 states that the Bregman representative of a random variable is always the expectation of the variable. Hence, when given a sample of a random variable, the optimal estimation of the Bregman representative is always the mean of the sample. Under the GPLD model, B_{pq} is the Bregman representative of each block of an affinity matrix. When C is given, i.e., the membership of each block is known, according to Theorem 5, B_{pq} is obtained as the mean of each block,

$$B_{pq} = \frac{1}{|\pi_p||\pi_q|}\sum_{i\in\pi_p, j\in\pi_q} A_{ij}, \tag{10}$$

Table 1: A list of Bregman divergences and the corresponding convex functions.

Name	$D_\phi(x,y)$	$\phi(x)$	Domain
Euclidean distance	$\|\|\mathbf{x}-\mathbf{y}\|\|^2$	$\|\|\mathbf{x}\|\|^2$	$\mathbb{R}^d$
Generalized I-divergence	$\sum_{i=1}^d x_i \log(\frac{x_i}{y_i}) - \sum_{i=1}^d (x_i - y_i)$	$\sum_{i=1}^d x_i \log(x_i)$	$\mathbb{R}_+^d$
Logistic loss	$x\log(\frac{x}{y}) + (1-x)\log(\frac{1-x}{1-y})$	$x\log(x) + (1-x)\log(1-x)$	$\{0,1\}$
Itakura-Saito distance	$\frac{x}{y} - \log xy - 1$	$-\log x$	$(0,\infty)$
Hinge loss	$\max\{0, -2\text{sign}(-y)x\}$	$\|x\|$	$\mathbb{R}\setminus\{0\}$
KL-divergence	$\sum_{i=1}^d x_i \log(\frac{x_i}{y_i})$	$\sum_{i=1}^d x_i \log(x_i)$	d-Simplex
Mahalanobis distance	$(\mathbf{x}-\mathbf{y})^T\mathbf{A}(\mathbf{x}-\mathbf{y})$	$\mathbf{x}^T\mathbf{A}\mathbf{x}$	$\mathbb{R}^d$

Algorithm 1 Graph Partitioning with Bregman Divergences

Input: A graph affinity matrix A, a Bregman divergence D_ϕ, and a positive integer k.

Output: A partition indicator matrix C and a partition representative matrix B.

Method:

```
Initialize $B$.
repeat
    for $h = 1$ to $n$ do
        $C_{hp^*} = 1$ for $p^* = \arg\min_p L_p$ where $L_p$ denotes
            $D_\phi(A, CBC^T)$ for $C_{hp} = 1$.
    end for
    $B = (C^TC)^{-1}C^TAC(C^TC)^{-1}$.
until convergence
```

where π_p and π_q denote the pth and the qth cluster, respectively, and $1 \le p \le k, 1 \le q \le k, 1 \le i \le n$ and $1 \le j \le n$. If we write Eq (10) in a matrix form, we obtain Eq. (3), i.e., Theorem 3 is true for all Bregman divergences. Hence, updating rule (6) is applicable to GPLD with any Bregamen divergneces. For updating rule (7), there is only a minor change for a given Bregman divergence, i.e., we calculate the object function L based on this given bregman divergence.

Therefore, we obtain a general graph partitioning algorithm, Graph Partitioning with Bregman Divergences (GPBD), which is summarized in Algorithm 1. Unlike the traditional graph partitioning approaches, this simple algorithm is capable of partitioning graphs under different link distribution assumptions by adopting different Bregman divergences. The computational complexity of GPBD can be shown to be $O(tn^2k)$ for t iterations. For a sparse graph, it is reduced to $O(t|\mathcal{E}|k)$. GPBD is faster than the popular spectral approaches, which involve expensive eigenvector computation (typically $O(n^3)$) and extra post-processing on eigenvectors to obtain the partitioning. Comparing with the multi-level approaches such as METIS (Karypis & Kumar 1998), GPBD does not restrict partitions to have an equal size.

The convergence of Algorithm 1 is guaranteed based on the following facts. First, based on Theorem 3 and Theorem 5, the objective function is non-increasing under updating rule (6); second, by the criteria for reassignment in updating rule (7), it is trivial to show that the objective function is non-increasing under updating rule (7).

A Unified View to Graph Partitioning

In this section, we establish the connections between the GPLD model and the edge cut objectives to provide a unified view for graph partitioning.

In general, the edge cut objectives, such as ratio association (Shi & Malik 2000), ratio cut(Chan, Schlag, & Zien 1993), Kernighan-Lin objective (Kernighan & Lin 1970), and normalized cut (Shi & Malik 2000), can be formulated as the following trace maximization (Zha *et al.* 2002; Dhillon, Guan, & Kulis 2004; 2005),

$$\max \text{tr}(\tilde{C}^T A \tilde{C}). \tag{11}$$

In (11), typically $\tilde{C}$ is a weighted indicator matrix such that

$$\tilde{C}_{ij} = \begin{cases} \frac{1}{|\pi_j|^{\frac{1}{2}}} & \text{if } v_i \in \pi_j \\ 0 & \text{otherwise} \end{cases}$$

where $|\pi_j|$ denotes the number of nodes in the jth partition. In other words, $\tilde{C}$ satisfies the constraints $\tilde{C} \in \mathbb{R}_+^{n\times k}$ and $\tilde{C}^T\tilde{C} = I_k$, where I_k is the $k \times k$ identity matrix.

We propose the following theorem to show that the various edge cut objectives are mathematically equivalent to a special case of the GPLD model. To be consistent with the weighted indicator matrix used in edge cut objects, in the following theorem we modify the constraints on C as $C \in \mathbb{R}_+$ and $C^TC = I_k$ to make C to be a weighted indicator matrix.

Theorem 6. *The GPLD model under Euclidean distance function and $B = rI_k$ for $r > 0$, i.e.,*

$$\min_{\substack{C\in\mathbb{R}_+^{n\times k},\\ C^TC=I_k}} ||A - C(rI_k)C^T||^2 \tag{12}$$

is equivalent to the maximization

$$\max tr(C^TAC), \tag{13}$$

where tr denotes the trace of a matrix.

Proof. Let L denote the objective function in Eq. 12.

$$\begin{aligned} L &= ||A - rCC^T||^2 \\ &= \text{tr}((A - rCC^T)^T(A - rCC^T)) \\ &= \text{tr}(A^TA) - 2r\text{tr}(CC^TA) + r^2\text{tr}(CC^TCC^T) \\ &= \text{tr}(A^TA) - 2r\text{tr}(C^TAC) + r^2k \end{aligned}$$

The above deduction uses the property of trace $\text{tr}(XY) = \text{tr}(YX)$. Since $\text{tr}(A^TA)$, r and k are constants, the minimization of L is equivalent to the maximization of $\text{tr}(C^TAC)$. The proof is completed. □

Table 2: Summary of the synthetic graphs

Graph	Parameter	n	k	distribution
syn1	$\begin{bmatrix} 3 & 3 & 2.7 \\ 3 & 2.7 & 2.7 \\ 2.7 & 2.7 & 3 \end{bmatrix}$	300	3	Normal
syn2	$\begin{bmatrix} 6.9 & 7 & 6.3 \\ 7 & 6.3 & 6.3 \\ 6.3 & 6.3 & 7 \end{bmatrix}$	600	3	Poisson
syn3	$\mathbb{R}^{20\times 20}$	20000	20	Normal

Theorem 6 states that with the partition representative matrix B restricted to be of the form rI_k, the GPLD model under Euclidean distance is reduced to the trace maximization in (13). Since various edge cut objectives can be formulated as the trace maximization, Theorem 6 establishes the connection between the GPLD model and the existing edge cut objective functions.

Based on this connection, edge cut objectives make two implicit assumptions for a graph's link distribution. First, Euclidean distance in Theorem 6 implies normal distribution assumption for the edge weights in a graph. Second, since the off-diagonal entries in B represent the mean edge weights between partitions and the diagonal elements of B represent the the mean edge weights within partitions, restricting B to be of the form rI_k for $r > 0$ implies that the edges between partitions are very sparse (close to 0) and the edge weights within partitions have the same positive expectation r. However, these two assumptions are not appropriate for the graphs whose link distributions deviate from normal distribution or dense graphs. Therefore, compared with the edge cut based approaches, the GPBD algorithm is more flexible to deal with graphs with different statistic characteristics.

Experimental Results

Although GPBD actually provides a family of algorithms under various Bregman divergences, due to the space limit, in this paper we present the experimental evaluation of the effectiveness of the GPBD algorithm under two most popular divergences, GPBD under Euclidean Distance (GPBD-ED) corresponding to normal distribution, and GPBD under Generalized I-divergence (GPBD-GI) corresponding to Poisson distribution, in comparison with two representative graph partitioning algorithms, Normalized Cut (NC) (Shi & Malik 2000; Ng, Jordan, & Weiss 2001) and METIS (Karypis & Kumar 1998).

We use synthetic data to simulate graphs whose edge weights are under normal and poisson distributions. The distribution parameters to generate the graphs are listed in the second column of Table 2 as matrices. In a parameter matrix P, P_{ij} denotes the distribution parameter that generates the edge weights between the nodes in the ith partition and the nodes in the jth partition. Graph syn3 has twenty partitions of 20000 nodes and about 10 million edges. Due to the space limit, its distribution parameters are omitted here.

The graphs based on the text data have been widely used to test graph partitioning algorithms (Ding *et al.* 2001; Dhillon 2001; Zha *et al.* 2001). In this study, we construct real graphs based on various data sets from the 20-newsgroups (Lang 1995) data, which contains about $20,000$ articles from the 20 news groups and can be used to generate data sets of different sizes, balances and difficulty levels. We pre-process the data by removing stop words and file headers and selecting the top 2000 words by the mutual information. Each document is represented by a term-frequency vector using TF-IDF weights and the cosine similarity is adopted for the edge weight. Specific details of data sets are listed in Table 3. For example, the third row of Table 3 shows that three data sets NG5-1, NG5-2 and NG5-3 are generated by sampling from five newsgroups with size 900, 1200 and 1450, respectively, and with *balance* 1.5, 2.5, and 4, respectively. Here *balance* denotes the ratio of the largest partition size to the smallest partition size in a graph. Normalized Mutual Information (NMI) (Strehl & Ghosh 2002) is used for performance measure, which is a standard way to measure the cluster quality. The final performance score is the average of twenty runs.

Table 4 shows the NMI scores of the four algorithms. For the synthetic data syn1 and syn3 with normal link distribution, the GPBD-ED algorithm, which assumes normal distribution for the links, provides the best NMI score. Similarly, for data syn2 with poisson link distribution, the GPBD-GI algorithm, which assumes poisson distribution for the links, provides the best performance.

For real graphs, we observe that GPBD-GI provides best NMI scores for all the graphs and preforms significantly better than NC and METIS in most graphs . This implies that link distributions of the graphs are closer to Poisson distribution than normal distribution. How to determine appropriate link distribution assumption for a given graph is beyond the scope of this paper. However, the result shows that the appropriate link distribution assumption (appropriate distance function for GPBD) leads to a significant improvement on the partitioning quality. For example, for the graph NG2-3, even NC totally fails and other algorithms perform poorly, GPBD-IS still provides satisfactory performance. We observe that all the algorithms perform poorly for NG10. One possible reason for this is that in NG10 some partitions are heavily overlapped and very unbalanced. We also observe that the performance of the GPBD with the appropriate distribution is more robust to unbalanced graphs. For example, from NG2-1 to NG2-3, the performance of GPBD-IS decreases much less than those of NC and METIS. One possible reason for METIS's performance deterioration on unbalanced graphs is that it restricts partitions to have equal size.

Conclusion

In this paper, we propose a general model to partition graphs based on link distribution. This model formulates graph partitioning under a certain distribution assumption as approximating the graph affinity matrix under the corresponding distortion measure. Under this model, we derive a novel graph partitioning algorithm to approximate a graph affinity matrix under various Bregman divergences, which correspond to a large exponential family of distributions. Our theoretic analysis and experiments demonstrate the potential

Table 3: Subsets of Newsgroup Data for constructing graphs.

Name	**Newsgroups Included**	**# Documents**	**Balance**
NG2-1/2/3	alt.atheism, comp.graphics	330/525/750	1.2/2.5/4
NG3-1/2/3	comp.graphics, rec.sport.hockey,talk.religion.misc	480/675/900	1.2/2.5/4
NG5-1/2/3	comp.os.ms-windows.misc, comp.windows.x, rec.motorcycles,sci.crypt, sci.space	900/1200/1450	1.5/2.5/4
NG10	comp.graphics, comp.sys.ibm.pc.hardware, rec.autos, rec.sport.baseball,sci.crypt, sci.med,comp.windows.x, soc.religion.christian, talk.politics.mideast,talk.religion.misc	5600	7

Table 4: NMI scores of the five algorithms

Data	NC	METIS	GPBD-ED	GPBD-GI
syn1	0.673 ± 0.081	0.538 ± 0.016	**0.915 ± 0.017**	0.893 ± 0.072
syn2	0.648 ± 0.052	0.533 ± 0.018	0.828 ± 0.139	**0.863 ± 0.111**
syn3	0.801 ± 0.029	0.799 ± 0.010	**0.933 ± 0.047**	0.811 ± 0.055
NG2-1	0.482 ± 0.299	0.759 ± 0.024	0.678 ± 0.155	**0.824 ± 0.045**
NG2-2	0.047 ± 0.041	0.400 ± 0.000	0.283 ± 0.029	**0.579 ± 0.073**
NG2-3	0.042 ± 0.023	0.278 ± 0.000	0.194 ± 0.008	**0.356 ± 0.027**
NG3-1	0.806 ± 0.108	0.810 ± 0.017	0.718 ± 0.128	**0.852 ± 0.081**
NG3-2	0.185 ± 0.116	0.501 ± 0.012	0.371 ± 0.131	**0.727 ± 0.070**
NG3-3	0.048 ± 0.013	0.546 ± 0.016	0.235 ± 0.091	**0.631 ± 0.179**
NG5-1	0.598 ± 0.077	0.616 ± 0.032	0.550 ± 0.043	**0.662 ± 0.025**
NG5-2	0.5612 ± 0.030	0.570 ± 0.020	0.546 ± 0.032	**0.670 ± 0.022**
NG5-3	0.426 ± 0.060	0.574 ± 0.018	0.515 ± 0.033	**0.668 ± 0.035**
NG10	0.281 ± 0.011	0.310 ± 0.017	0.308 ± 0.015	**0.335 ± 0.009**

and effectiveness of the proposed model and algorithm. We also show the connections between the traditional edge cut objectives and the proposed model to provide a unified view to graph partitioning.

Acknowledgement

This work is supported in part by NSF (IIS-0535162), AFRL (FA8750-05-2-0284), and AFOSR (FA9550-06-1-0327).

References

Banerjee, A.; Dhillon, I. S.; Ghosh, J.; Merugu, S.; and Modha, D. S. 2004a. A generalized maximum entropy approach to bregman co-clustering and matrix approximation. In *KDD*, 509–514.

Banerjee, A.; Merugu, S.; Dhillon, I. S.; and Ghosh, J. 2004b. Clustering with bregman divergences. In *SDM*.

Bui, T. N., and Jones, C. 1993. A heuristic for reducing fill-in in sparse matrix factorization. In *PPSC*, 445–452.

Chan, P. K.; Schlag, M. D. F.; and Zien, J. Y. 1993. Spectral k-way ratio-cut partitioning and clustering. In *DAC '93*, 749–754.

Collins, M.; Dasgupta, S.; and Reina, R. 2001. A generalizaionof principal component analysis to the exponential family. In *NIPS'01*.

Dhillon, I.; Guan, Y.; and Kulis, B. 2004. A unified view of kernel k-means, spectral clustering and graph cuts. Technical Report TR-04-25, University of Texas at Austin.

Dhillon, I.; Guan, Y.; and Kulis, B. 2005. A fast kernel-based multilevel algorithm for graph clustering. In *KDD '05*.

Dhillon, I. S. 2001. Co-clustering documents and words using bipartite spectral graph partitioning. In *KDD*, 269–274.

Ding, C. H. Q.; He, X.; Zha, H.; Gu, M.; and Simon, H. D. 2001. A min-max cut algorithm for graph partitioning and data clustering. In *Proceedings of ICDM 2001*, 107–114.

Hendrickson, B., and Leland, R. A multilevel algorithm for partitioning graphs. In *Supercomputing '95*.

Karypis, G., and Kumar, V. 1998. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.* 20(1):359–392.

Kernighan, B., and Lin, S. 1970. An efficient heuristic procedure for partitioning graphs. *The Bell System Technical Journal* 49(2):291–307.

Lang, K. 1995. News weeder: Learning to filter netnews. In *ICML*.

Long, B.; Wu, X.; Zhang, Z. M.; and Yu, P. S. 2006. Unsupervised learning on k-partite graphs. In *KDD-2006*.

Ng, A.; Jordan, M.; and Weiss, Y. 2001. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*.

S.D.Pietra, V.D.Pietera, J. 2001. Duality and auxiliary functions for bregman distances. Technical Report CMU-CS-01-109, Carnegie Mellon University.

Shi, J., and Malik, J. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(8):888–905.

Strehl, A., and Ghosh, J. 2002. Cluster ensembles – a knowledge reuse framework for combining partitionings. In *AAAI 2002*, 93–98.

Wang, S., and Schuurmans, D. 2003. Learning latent variable models with bregman divergences. In *IEEE International Symposium on Information Theory*.

Yu, K.; Yu, S.; and Tresp, V. 2005. Soft clustering on graphs. In *NIPS'05*.

Zha, H.; Ding, C.; Gu, M.; He, X.; and Simon, H. 2001. Bi-partite graph partitioning and data clustering. In *ACM CIKM'01*.

Zha, H.; Ding, C.; Gu, M.; He, X.; and Simon, H. 2002. Spectral relaxation for k-means clustering. *Advances in Neural Information Processing Systems* 14.