Computing The Analytic Connectivity of A Uniform Hypergraph

Chunfeng Cui* Ziyan Luo† Liqun Qi‡ Hong Yan§

November 7, 2016

Abstract

The analytic connectivity, proposed as a substitute of the algebraic connectivity in the setting of hypergraphs, is an important quantity in spectral hypergraph theory. The definition of the analytic connectivity for a uniform hypergraph involves a series of polynomial optimization problems (POPs) associated with the Laplacian tensor of the hypergraph with nonnegativity constraints and a sphere constraint, which poses difficulties in computation. To reduce the involved computation, properties on the algebraic connectivity are further exploited, and several important structured uniform hypergraphs are shown to attain their analytic connectivities at vertices of the minimum degrees, hence admit a relatively less computation by solving a small number of POPs. To efficiently solve each involved POP, we propose a feasible trust region algorithm (`FTR`) by exploiting their special structures. The global convergence of `FTR` to the second-order necessary conditions points is established, and numerical results for both small and large size examples with comparison to other existing algorithms for POPs are reported to demonstrate the efficiency of our proposed algorithm.

Key words. Uniform hypergraph; Laplacian tensor; Analytic connectivity; Feasible trust region algorithm

AMS subject classifications. 05C65, 15A18, 90C55

1 Introductions

Spectral graph theory is a well-studied and highly applicable subject, which focuses on the connection between properties of a graph and the eigenvalues of matrices associated with the graph. Such matrices include the adjacency matrix, the Laplacian matrix and the signless Lpalacian matrix of the graph [3, 10, 15, 16, 39]. However, the study of graphs cannot fully meet the developments of modern science and technology, especially in big data analysis and complex networks. This motivates the study of hypergraphs, where an edge may connect more than two vertices [1, 2], comparing to two-vertices edges in ordinary graphs. Spectral hypergraph theory correspondingly emerged which was based upon matrix spectral analysis in its early stage.

In 2005, Lim [41] and Qi [49] independently introduced the concept of eigenvalues for tensors, which initiated the study of tensor spectral theory and paved a way for the development of spectral hypergraph

*Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong (`chunfcui@cityu.edu.hk`).

†State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, P.R. China (`starkeynature@hotmail.com`).

‡Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (`maqilq@polyu.edu.hk`).

§Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong (`h.yan@cityu.edu.hk`).

theory via tensors. The related research include spectral hypergraph theory [12, 13, 23, 35, 36, 38, 45, 47, 53, 54, 59, 65], eigenvalues [29, 37, 46, 55, 56, 57, 58, 61, 63], connectivity [28, 40], Laplacian tensor [5, 30, 32, 48, 50, 64], structured tensors related [9, 14], special hypergraphs [6, 31, 34, 51, 62], hypergraph properties [7, 20, 22, 42, 43]. The tensors studied in these papers include adjacency tensors, Laplacian tensors and signless Laplacian tensors of hypergraphs. Benefitting from the high sparsity of these tensors, Chang, Chen and Qi [8] recently proposed a CEST algorithms for computing extremal eigenvalues of large-scale adjacency tensors, Laplacian tensors and signless Laplacian tensors of uniform hypergraphs, which provides a useful computational tool for spectral hypergraph theory via tensors.

It is well-known that in spectral graph theory, the algebraic connectivity [19], defined as the second smallest eigenvalue of the Laplacian matrix of a graph, is an important quantity. However, as Laplacian tensors of uniform hypergraphs may have complex eigenvalues, a different approach for generalizing this concept to hypergraphs was introduced by Qi in [50], where the analytic connectivity for a uniform hypergraph was defined via an optimization formulation

$$\alpha(G) = \min_{j=1,\cdots,n} \min_{x\in\mathbb{R}^n} \{\mathcal{L}x^k \,:\, x \in \mathbb{R}^n_+,\ \sum_{i=1}^n x[i]^k = 1,\ x[j] = 0\}.$$

This is shown to be linked with the edge connectivity of the hypergraph. It was further studied by Li, Cooper and Chang in [40] where the analytic connectivity was shown to be connected with other important invariants of hypergraphs, such as the degree, the vertex connectivity, the diameter and the isoperimetric number.

To our best knowledge, no efficient algorithm has been proposed for computing the analytic connectivity of a uniform hypergraph in the literature. The definition of analytic connectivity involves a series of polynomial optimization problems. For dimension n large enough, this is very costly. To fix this issue, we firstly explore some specific hypergraphs, and shown some properties on the vertices the analytic connectivity will possibly attained.

For each specific j, we propose a feasible trust region algorithm for the computation of this quantity. Note that the analytic connectivity involves a series of optimization problems, each of which possesses nonnegativity constraints and a sphere constraint. Thus, they are special cases of the following general constrained optimization problem

$$\min_{x\in\mathbb{R}^n} \quad f(x) \quad \text{s.t.} \quad c(x) = 0, \quad x \geq 0, \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a nonconvex polynomial function and $c : \mathbb{R}^n \to \mathbb{R}$ is a nonlinear smooth function. Existing optimization algorithms for (1.1) can be roughly classified into three types. The first type includes the penalty methods, which incorporates the equality constrains $c(x) = 0$ into the objective function as a penalty term, and attempt to solve (1.1) by a sequential minimization problems of the form

$$\min_{x\in\mathbb{R}^n} \quad \mathcal{F}_k^{(1)}(x, \lambda_k) \quad \text{s.t.} \quad x \geq 0,$$

where the objective function $\mathcal{F}_k^{(1)}$ could be any penalty function such that the subproblem can be easily solved. For instance, the L-BFGS method [4], the gradient projection method [17, 18], and the active set method [26], etc. The solver `MINOS` belongs to this type. However, as the hard constraint $c(x) = 0$ has been relaxed as a penalty term in the objective, this type of methods usually result in an infeasible point for our sphere constraint. The second type of methods involves solving

$$\min_{x\in\mathbb{R}^n} \quad \mathcal{F}_k^{(2)}(x, \mu_k) \quad \text{s.t.} \quad c(x) = 0,$$

where the objective function $\mathcal{F}_k^{(2)}$ is always with some interior-point penalty of the nonnegativity constraint and the solver `IPOPT` belongs to this type. With the equality constraint in the above subproblem, this type of methods is always time consuming. The third type includes the sequential quadratic programming methods, which solves the subproblem

$$\min_{x\in\mathbb{R}^n} \quad \mathcal{F}_k^{(3)}(x,\lambda_k) \quad \text{s.t.} \quad c_k + A_k(x-x_k) = 0, \quad x \geq 0,$$

where the objective function $\mathcal{F}_k^{(3)}$ is a quadratic function using the information of the Lagrangian function or its variants [21]. This type of methods show their strength when the constraints have significant nonlinearity, and the solver `SNOPT` belongs to this type.

Note that the constraint $c(x) = 0$ in this paper is actually the k-norm sphere constraint. By exploring this special structure, we propose a feasible trust region method (`FTR`), the mixture of trust region method and the projection method, in which the projection step ensures the feasibility of each iteration and the trust region technique enhances the convergence. `FTR` was also used in [27] for computing Z-eigenvalues of symmetric tensors. While the main difference is that here we adopted the ∞-norm trust region instead of the Euclidean norm, which remarkably facilitates the computation as at each iteration only a linear constrained quadratic subproblem needs to be handled. Infinity norm was also used in [24] for bound constrained problems, where advantages in terms of computational costs were demonstrated.

This paper is organized as follows. In Section 2, several related basic concepts and properties on hypergraphs and the analytic connectivity are reviewed. Further properties on the vertices attainable for the analytic connectivity is discussed in Section 3 to reduce the computation by cutting down the number of the involved POPs. For each POP, an `FTR` algorithm for computing the analytic connectivity of a uniform hypergraph is proposed in Section 4. The global convergence to the second order stationary points is established in Section 5. Numerical results are reported in Section 6, which demonstrates the efficiency of our proposed algorithm, and indicates that the analytic connectivity is a good choice to characterize the connectivity of the involved hypergraph as well. Conclusions are drawn in Section 7.

Notations throughout the paper are listed here. Let k and n be any two positive integers. We use $T_{k,n}$ to denote the space of all k-th order n-dimensional tensors. $\mathbb{R}^n_+$ is used to stand for the set of all nonnegative vectors in $\mathbb{R}^n$. For any $x \in \mathbb{R}^n$ and any integer $i \in [n]$, $x[i]$ denotes the i-th component of x, $x^{[k]} := (x[i]^k) \in \mathbb{R}^n$ with any given positive integer k, and $\operatorname{diag}(x) \in \mathbb{R}^{n\times n}$ is the diagonal matrix generated by x. For any set C, $|C|$ denotes the cardinality of C. The index set $\{1, 2, \dots, n\}$ is simply denoted as $[n]$. The notation $\binom{n}{m}$ denotes the combinatorial number of choosing m from n.

2 Preliminaries

As a natural extension of a graph, a **hypergraph** $G = (V, E)$ with the vertex set $V = [n]$ and the edge set $E = \{e_1, \cdots, e_m\}$ allows each of its edge e_j joins any number of vertices. If each edge e_j connects exactly k vertices, this hypergraph is called a **k-uniform hypergraph**, or simply called as a **k-graph**. For more details on hypergraphs, refer to [1, 2, 12]. Obviously, G is reduced to an ordinary graph when $k = 2$. Thus, we assume $k \geq 3$ throughout the paper.

Many important structured hypergraphs have been introduced in the literature. Let $G = (V, E)$ be a uniform hypergraph. G is called a **sunflower** if there is a disjoint partition of the vertex set V as $V = V_0 \cup V_1 \cup \cdots \cup V_d$ such that $|V_0| = 1$ and $|V_1| = \cdots = |V_d| = k-1$, and $E = \{V_0 \cup V_i \,|\, i \in [d]\}$ ([31]); G is called a **hypercycle** if there are s subsets $V_1, \dots, V_s$ of the vertex set V such that $|V_1| = \cdots = |V_s| = k$,

$|V_1 \cap V_2| = \cdots = |V_{s-1} \cap V_s| = |V_s \cap V_1| = 1$ and $V_i \cap V_j = \emptyset$ for the other cases, the intersections $V_1 \cap V_2$, $\ldots$, $V_s \cap V_1$ are mutually different, and $E = \{V_i \mid i \in [s]\}$ ([32]); G is called a **squid** if we can number the vertex set V as $V = \{i_{1,1}, \cdots, i_{1,k}, \cdots, i_{k-1,1}, \cdots, i_{k-1,k}, i_{k,1}\}$ such that the edge set $E = \{\{i_{1,1}, \cdots, i_{1,k}\}$, $\cdots$,$\{i_{k-1,1}, \cdots, i_{k-1,k}\}$, $\{i_{1,1}, \cdots, i_{k-1,1}, i_{k,1}\}\}$ ([31]); More generally, G is called a **s-path** of length l if $V = \{v_1, v_2, \ldots, v_{s+l(k-s)}\}$ and $E = \{\{v_{1+i(k-s)}, v_{1+i(k-s)+1}, \ldots, v_{s+(i+1)(k-s)}\} \mid 0 \le i \le l-1\}$. Particularly, we call a 1-path hypergraph G as a **loose path**; G is called a **complete k-graph** if $E = \{e \mid e \subset V, |e| = k\}$.

Some related fundamental concepts of uniform hypergraphs are reviewed as follows.

Definition 2.1 ([12, 50])**.** Let $G = (V, E)$ be a k-graph. The **adjacency tensor** of G is defined as the k-th order n-dimensional tensor $\mathcal{A}$ whose $(i_1, \cdots, i_k)$-entry is:

$$a_{i_1,\cdots,i_k} := \begin{cases} \frac{1}{(k-1)!}, & \text{if } \{i_1 \cdots, i_k\} \in E; \\ 0, & \text{otherwise.} \end{cases}$$

Let $\mathcal{D}$ be a k-th order n-dimensional diagonal tensor with its diagonal element $d_{i,\cdots,i}$ being d_i, the degree of vertex i, for all $i \in [n]$. $\mathcal{D}$ is called the **degree tensor** corresponding to G. Then **Laplacian tensor** of G is defined as $\mathcal{L} := \mathcal{D} - \mathcal{A}$, and the **signless Laplacian tensor** of G as $\mathcal{Q} := \mathcal{D} + \mathcal{A}$.

Definition 2.2 ([50])**.** Let G be a k-graph with n vertices. The **analytic connectivity** of G is defined as

$$\alpha(G) = \min_{j=1,\cdots,n} \alpha_j(G), \tag{2.1}$$

where

$$\alpha_j(G) = \min_{x \in \mathbb{R}^n} \{\mathcal{L}x^k \ : \ x \in \mathbb{R}^n_+, \ \sum_{i=1}^n x[i]^k = 1, \ x[j] = 0\}, \tag{2.2}$$

with $\mathcal{L}$ the Laplacian tensor of G.

Let $G = (V, E)$ be a k-graph with n vertices. For each vertex $i \in V$, denote by $E(i)$ the set of edges containing the vertex i, i.e., $E(i) := \{e \in E \,|\, i \in e\}$. The degree d_i of the vertex i is the cardinality $|E(i)|$ of the set $E(i)$. Denote by Δ, δ and $\bar{d}$ the maximum, minimum and average degree of G, respectively. Existing results on analytic connectivity of a uniform hypergraph include the following:

- [50] $\alpha(G) \ge 0$; $\alpha(G) > 0$ if and only if G is connected;
- [50] $e(G) \ge \frac{n}{k}\alpha(G)$, where $e(G)$ is the **edge connectivity** of G, defined as the minimum cardinality of an edge cut of G;
- [50] $\alpha(G) \le \delta$;
- [40] $\alpha(K_n^{(k)}) = \binom{n-2}{k-2}$, where $K_n^{(k)}$ is the complete k-graph;
- [40] denote $v(G)$ as the **vertex connectivity** of G, defined as the minimum cardinality of a vertex cut of G,

$$\alpha(G) \le \bar{\alpha} := \binom{n-2}{k-2} - \left[\binom{n - v(G) - 1}{k-1} - \binom{\lfloor \frac{(n-v(G))}{2} \rfloor - 1}{k-1}\right] \frac{k-1}{n-1}; \tag{2.3}$$

- [40] $\frac{k}{2}i(G) \geq \alpha(G) \geq \Delta - \sqrt{\Delta^2 - i^2(G)}$, where $i(G)$ is the **isoperimetric number**, or the **Cheeger constant** of G, defined by $i(G) = \min\left\{\frac{|E(S,\bar{S})|}{|S|} : S \subset V, 0 < |S| \leq \frac{n}{2}\right\}$, $\bar{S} = V \setminus S$, and $E(S, \bar{S})$ is an edge cut of G;

- [40] $\alpha(G) \geq \frac{4}{n^2(k-1)\mathrm{diam}(G)}$, where $\mathrm{diam}(G)$ is the **diameter** of G, defined as the maximum distance between any pair of vertices of G;

- [40] $\alpha(G) \leq \min\left\{\frac{1}{k}(d(v_{i_1}) + d(v_{i_2}) + \cdots + d(v_{i_k}) - k) : v_{i_1}, \cdots, v_{i_k} \in E(G)\right\}$.

It is worth pointing out that the isoperimetric number or the Cheeger constant of an ordinary graph provides a numerical measure of whether or not a graph has a "bottleneck", which has wide applications such as in constructing well-connected networks of computers and card shuffling. However, the computation of such an invariant is very difficult and the algebraic connectivity provides a reasonable good bound in terms of the well-known "**Cheeger inequality**" in the ordinary graph case. This result is in a certain sense theoretically extended to the uniform hypergraphs as stated above by Li, Cooper and Chang [40] where the analytical connectivity was adopted instead of the algebraic connectivity. In this regard, the computational algorithm presented in this paper makes the theoretical result of [40] practically feasible to efficiently bound the isoperimetric number of a k-graph.

3 Properties on the analytic connectivity

In this section, we will discuss the properties on finding which vertices of a uniform hypergraph the analytic connectivity will possibly be attained at. This will henceforth play an essential role in reducing the required computation for the analytic connectivity by cutting down the number of POPs involved in Definition 2.2. We begin with the following important lemma.

Lemma 3.1. *Let $G = (V, E)$ be a k-graph with $V = [n]$, and $i, j \in [n]$ be any two vertices with edge sets $E(i)$ and $E(j)$. If $E(i) \subset E(j)$, then $\alpha_i(G) \leq \alpha_j(G)$, where $\alpha_i(G)$ and $\alpha_j(G)$ are defined as in (2.2).*

Proof: Let $E(i) = \{e_1(i), \ldots, e_{d_i}(i)\}$ and $E(j) = E(i) \cup \{e_{d_i+1}(j), \ldots, e_{d_j}(j)\}$, where d_i and d_j are the degrees of vertices i and j, respectively. For any $x \in \mathbb{R}^n$, denote $\mathcal{L}(e)x^k = \sum_{i \in e} x[i]^k - k\Pi_{i \in e} x[i]$ as the Laplacian function corresponding to any given edge $e \in E$. For any $x_1 \in \mathbb{R}^n$ satisfying $x_1[i] = 0$, we have

$$\mathcal{L}x_1^k = \sum_{e_l \in E(i)} \sum_{l_t \in e_l, l_t \neq i} x_1[l_t]^k + \sum_{e \in E \setminus E(i)} \mathcal{L}(e)x_1^k.$$

For any $x_2 \in \mathbb{R}^n$ satisfying $x_2[j] = 0$, we have

$$\mathcal{L}x_2^k = \sum_{e_l \in E(i)} \sum_{l_t \in e_l, l_t \neq j} x_2[l_t]^k + \sum_{e_l \in E(j) \setminus E(i)} \sum_{l_t \in e_l, l_t \neq j} x_2[l_t]^k + \sum_{e \in E \setminus E(j)} \mathcal{L}(e)x_2^k.$$

Note that the vertex i is only contained in the edges of $E(i)$ and hence $x_2[i]$ only exists in the first term of the right hand side of the above expression. To achieve the minimum value $\alpha_j(G)$ in (2.2), it is evident from the nonnegativity constraint that for any optimal solution $\bar{x}$ of the problem (2.2) with $x[j] = 0$, it holds that $\bar{x}[i] = 0$. Therefore, $\bar{x}$ is also a feasible solution of the problem (2.2) with $x[i] = 0$. This immediately shows the desired inequality. **Q.E.D.**

With the help of Lemma 3.1, we can show that for several important uniform hypergraphs, such as sunflowers, hypercycles, squids and loose path, the computation of their analytic connectivities can be significantly reduced by the following theorem.

Theorem 3.2. *Let G be a k-graph with the vertex set $[n]$. If G is a sunflower, or a hypercycle, or a squid, or a loose path, then $\alpha(G) = \alpha_j(G)$, where $j \in [n]$ is a vertex with the minimum degree.*

Proof: Let $G = (V, E)$ be a k-graph with $V = [n]$. (i) If G is a sunflower, then we can find a disjoint partition of the vertex set V, says $V = V_0 \cup V_1 \cup \cdots \cup V_d$, such that $|V_0| = 1$ and $|V_1| = \cdots = |V_d| = k - 1$, and $E = \{V_0 \cup V_i \,|\, i \in [d]\}$, where $1 + d(k-1) = n$. Let $V_0 = \{v_0\}$. Obviously, v_0 has degree d and other vertices all have degree 1. Moreover, for any $v \in V \setminus V_0$, $E(v) \subset E(v_0)$. Invoking of Lemma 3.1, the desired result follows readily in this case. (ii) If G is a hypercycle, then there exist s subsets V_1, ..., V_s of the vertex set V such that $|V_1| = \cdots = |V_s| = k$, $|V_1 \cap V_2| = \cdots = |V_{s-1} \cap V_s| = |V_s \cap V_1| = 1$ and $V_i \cap V_j = \emptyset$ for the other cases. From the definition of hypercycles, we know that each intersected vertex has degree two and others has degree one. And for any $v \in V$ of degree two, there exists a vertex $v' \in V$ such that $E(v') \subset E(v)$. Thus, by applying Lemma 3.1, the desired result is obtained in this case. (iii) If G is a squid, then we can number the vertex set V as $V = \{i_{1,1}, \cdots, i_{1,k}, \cdots, i_{k-1,1}, \cdots, i_{k-1,k}, i_{k,1}\}$ such that the edge set $E = \{\{i_{1,1}, \cdots, i_{1,k}\}, \cdots, \{i_{k-1,1}, \cdots, i_{k-1,k}\}, \{i_{1,1}, \cdots, i_{k-1,1}, i_{k,1}\}\}$. Note that the vertices $i_{1,1}, \cdots, i_{k-1,1}$ all have degree two and others all have degree one, and for every vertex $i_{j,1}$ with degree two, there exist vertex $i_{j,2}$ such that $E(i_{j,2}) \subset E(i_{j,1})$. Thus, from Lemma 3.1, we have $\alpha_{j,2}(G) \le \alpha_{j,1}(G)$. (iv) Similar to case (ii), we can prove the case when G is a loose path by definition and Lemma 3.1. This completes the proof. **Q.E.D.**

Two more specific uniform hypergraphs are discussed whose analytic connectivities can be computed via solving (2.2) with special choices of j. The first one is the 2-path with n vertices which is plotted as follows.

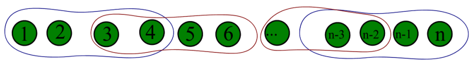

Figure 3.1: A 2-path 4-graph with length $\frac{n-2}{2}$

Proposition 3.3. *Let G be a 2-path 4-graph with $n \ge 4$ vertices, defined as in Figure 3.1. Then $\alpha(G) = \alpha_j(G)$, where j could be any element in $\{1, 2, n-1, n\}$. Moreover, $\alpha(G)$ is monotonically decreasing with n.*

Proof: First we consider the first part of the proposition. It is trivial when $n = 4$. For $n = 6$, the desired result can be obtained immediately from the symmetric structure of G and Lemma 3.1. Before proceeding for general cases of $n > 6$, we will introduce the following useful function for any given even integer $l \ge 4$,

$$\beta_\gamma^l = \min_y \{g_\gamma^l(y) := y_1^4 + \cdots + y_{l-2}^4 - 4y_1y_2y_3y_4 - \cdots - 4y_{l-3}y_{l-2}y_{l-1}y_l \quad \text{s.t.} \quad \sum_{i=1}^{l} y_i^4 = \gamma\}.$$

It is easy to see that $\beta_\gamma^l = \gamma\beta_1^l$ from the homogeneous structure of the above minimization problem. Moreover, we claim that β_γ^l is decreasing with l. Let l_1, l_2 be any two even integers satisfying $l_1 > l_2 \ge 4$. For any optimal solution $\hat{y}$ of problem with dimension l_2, $\bar{y} = [zeros(l_1 - l_2, 1), \hat{y}]$ is a feasible solution of dimension l_1. Hence

$$\beta_\gamma^{l_1} \le g_\gamma^{l_1}(\bar{y}) = g_\gamma^{l_2}(\hat{y}) = \beta_\gamma^{l_2},$$

where the first equality comes from the fact that the formulation of $g_\gamma^{l_2}(\hat{y})$ is the same with $g_\gamma^{l_1}(\bar{y})$. Furthermore, for any even integer $l \ge 4$, β_γ^l is negative. This comes from the claim above and the observation that given $\bar{y} = (\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$, $\beta_\gamma^4 \le g_\gamma^4(\bar{y}) = \bar{y}_1^4 + \bar{y}_2^4 - 4\bar{y}_1\bar{y}_2\bar{y}_3\bar{y}_4 = -\frac{1}{2}$.

For any even integer $n > 6$, it holds that $\mathcal{L}x^4 = 1 + \sum_{i=3}^{n-2} x_i^4 - \mathcal{A}x^4$. Suppose that for some j, $x_j = 0$, then the index set $[n] \setminus \{j\}$ can be partitioned into $\{1, \cdots, L_1\}$, $\{j - (-1)^j\}$, and $\{L_1 + 3, \cdots, n\}$. Hence $\mathcal{L}x^4$ can be rewritten as

$$\mathcal{L}x^4 = 1 + g_\gamma^{L_1}(x_{[1:L_1]}) + \delta + g_{1-\gamma-\delta}^{n-L_1-2}(x_{[L_1+3:n]}), \tag{3.1}$$

where $L_1 = j - \frac{3}{2} - \frac{1}{2}(-1)^j$, $\delta = x_{j-(-1)^j}^4$, and $g_\gamma^2 = 0$. Note that the variable x in (3.1) are partitioned into three subvectors, thus

$$\min\ \mathcal{L}x^4 \text{ s.t. } \sum_{i=1, i\neq j}^{n} x_i^4 = 1 \Longleftrightarrow \min_{\gamma,\delta\geq 0}\ 1 + \gamma\beta_1^{L_1} + \delta + (1-\delta-\gamma)\beta_1^{n-L_1-2} \text{ s.t. } \gamma + \delta \leq 1.$$

It follows from t β_1^l is negative that $\delta = 0$, and the objective function is reduced to $1 + \gamma\beta_1^{L_1} + (1 - \gamma)\beta_1^{n-L_1-2}$, as β_1^l decreasing with l, hence

$$\alpha_j(G) = 1 + \beta_1^{l_j},$$

where $l_j = \max(L_1, n - L_1 - 2)$. Hence $j^* = \arg\min_j\ \alpha_j = \arg\max_j\ l_j$. By direct computation we have $j^* \in \{1, 2, n-1, n\}$ and $l_j = n - 2$. When $l_j = L_1$, it holds that $\gamma = 1$; otherwise, $\gamma = 0$. Thus,

$$\alpha(G) = 1 + \beta_1^{n-2}. \tag{3.2}$$

As β_1^{n-2} is monotonically decreasing with n, so is the analytic connectivity α from (3.2). This completes the proof. **Q.E.D.**

The second specific one, termed as K_n^-, is the k-graph obtained by deleting an arbitrary edge from a complete k-graph $K_n^{(k)}$. For example, when $k = 3$, $n = 4$, the edge set of K_4^- are $\{\{1, 2, 4\}, \{1, 3, 4\}, \{2, 3, 4\}\}$, as shown in Figure 3.2.

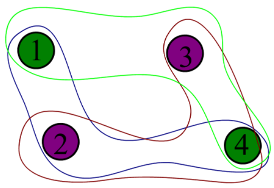

Figure 3.2: K_4^- generated by deleting the edge $\{1, 2, 3\}$ from $K_4^{(3)}$

Proposition 3.4. *Suppose K_n^- is the hypergraph generated by deleting an edge $\hat{e}$ from $K_n^{(k)}$. Then $\alpha(K_n^-) = \alpha_j(K_n^-)$, where j is some vertex in $\hat{e}$, i.e.,*

$$\alpha_{j_1}(K_n^-) < \alpha_{j_2}(K_n^-), \quad \forall j_1 \in e,\ \forall j_2 \in V \setminus \hat{e}. \tag{3.3}$$

Proof: Without loss of generality, suppose that the edge $\hat{e} = \{1, \cdots, k\}$ is deleted. By the symmetric property of this hypergraph, to show (3.3), we only need to prove $\alpha_1(K_n^-) < \alpha_n(K_n^-)$. For $j \in [n]$ satisfying $x[j] = 0$ and $\|x\|_k = 1$, we have

$$\mathcal{L}(K_n^-)x^k = \sum_{e\notin E(j)} \mathcal{L}(e)x^k + \sum_{e\in E(j)} \mathcal{L}(e)x^k.$$

If $j \notin \hat{e}$,

$$\sum_{e\in E(j)} \mathcal{L}(e)x^k = \sum_{e\in E(j)} \sum_{l\in e, l\neq j} x[l]^k = \binom{n-2}{k-2} \sum_{l\in[n]\setminus j} x[l]^k = \binom{n-2}{k-2}, \tag{3.4}$$

otherwise,

$$\sum_{e\in E(j)} \mathcal{L}(e)x^k = \sum_{e\in E(j)} \sum_{l\in e, l\neq j} x[l]^k - \mathcal{L}(\hat{e})x^k = \binom{n-2}{k-2} - \mathcal{L}(\hat{e})x^k. \tag{3.5}$$

For the case $x[1] = 0$, set $\bar{x}$ as

$$\bar{x}[i] = \begin{cases} (\frac{1}{n-1})^{\frac{1}{k}}, & \text{if} \quad i = 2, \cdots, n; \\ 0, & \text{if} \quad i = 1. \end{cases}$$

Then $j \in \hat{e}$. For all $e \notin E(j)$ it holds that $\mathcal{L}(e)\bar{x}^k = 0$. It follows from (3.5) that

$$\alpha_1(K_n^-) \leq \mathcal{L}(K_n^-)\bar{x}^k = \sum_{e\notin E(j)} \mathcal{L}(e)\bar{x}^k + \sum_{e\in E(j)} \mathcal{L}(e)\bar{x}^k - (\bar{x}[2]^k + \cdots + \bar{x}[k]^k) = \binom{n-2}{k-2} - \frac{k-1}{n-1}.$$

For the case $x[n] = 0$, it holds that $j \notin \hat{e}$. It follows from (3.4) that

$$\alpha_n(K_n^-) = \sum_{e\notin E(j)} \mathcal{L}(e)x^k + \sum_{e\in E(j)} \mathcal{L}(e)x^k \geq \binom{n-2}{k-2}.$$

where the last inequality follows from the arithmetic-geometric mean inequality that $\mathcal{L}(e)x^k \geq 0$ for all $e \notin E(j)$. In fact, the lower bound can be achieved by set $\tilde{x}$ as

$$\tilde{x}[i] = \begin{cases} (\frac{1}{n-1})^{\frac{1}{k}}, & \text{if} \quad i = 1, \cdots, n-1; \\ 0, & \text{if} \quad i = n. \end{cases}$$

Hence, $\alpha_n(K_n^-) = \binom{n-2}{k-2}$.

Hencefore,

$$\alpha_1(K_n^-) < \binom{n-2}{k-2} = \alpha_n(K_n^-).$$

This complete the proof of (3.3). **Q.E.D.**

As discussed above, those vertices of the smallest degree are highly possible to help attain the analytic connectivity of a uniform hypergraph. A conjecture comes as follows.

Conjecture 3.1 Let $G = ([n], E)$ be a k-graph. $\alpha(G) = \alpha_j(G)$ for some $j \in [n]$ of the smallest degree.

4 A feasible trust region algorithm

In this section, we propose the feasible trust region method (`FTR`) for solving (2.1). Noting that the projection to the k-norm sphere and nonnegative space are easy. Hence, we manage to project the iterate points to the feasible set, while maintaining the convergence.

The problem (2.2) can be rewritten as follows

$$\begin{aligned} \alpha_j = \min_{x\in\mathbb{R}^n} \quad & \frac{1}{k}\mathcal{L}x^k, \\ \text{s.t.} \quad & \frac{1}{k}\left(\sum_{i=1}^n x[i]^k - 1\right) = 0, \\ & x \geq 0, \quad x[j] = 0, \end{aligned} \tag{4.1}$$

which is equivalent to

$$
\begin{aligned}
\min_{x\in\mathbb{R}^{n-1}} \quad & f(x) = \frac{1}{k}\tilde{\mathcal{L}}x^k, \\
\text{s.t.} \quad & c(x) := \frac{1}{k}\left(\sum_{i=1}^{n-1} x[i]^k - 1\right) = 0, \\
& x \geq 0,
\end{aligned} \tag{4.2}
$$

where $\tilde{\mathcal{L}} \in T_{k,n-1}$ is the subtensor of $\mathcal{L}$ indexed by $[n] \setminus \{j\}$.

Before describing the details of `FTR` algorithm, the following functions are given. The Lagrangian function of (4.2) is

$$
L(x, \lambda) = f(x) - \lambda c(x), \tag{4.3}
$$

and its gradient vector and Hessian matrix are

$$
g(x) = \nabla_x L(x, \lambda) = \nabla f(x) - \lambda \nabla c(x), \tag{4.4}
$$

$$
W(x) = \nabla^2_{xx} L(x, \lambda) = \nabla^2 f(x) - \lambda \nabla^2 c(x), \tag{4.5}
$$

where $\nabla f(x) = \tilde{\mathcal{L}}x^{k-1}$, $\nabla^2 f(x) = (k-1)\tilde{\mathcal{L}}x^{k-2}$, $\nabla c(x) = x^{[k-1]}$, $\nabla^2 c(x) = (k-1)\mathrm{diag}(x^{[k-2]})$. Here, $\tilde{\mathcal{L}}x^{k-1} \in \mathbb{R}^{n-1}$ is a vector with the i-th element being

$$
(\tilde{\mathcal{L}}x^{k-1})[i] = \sum_{i_2,\cdots,i_k=1}^{n-1} \tilde{\mathcal{L}}_{i,i_2,\cdots,i_k} x_{i_2} \cdots, x_{i_k},
$$

and $\tilde{\mathcal{L}}x^{k-2} \in \mathbb{R}^{(n-1)\times(n-1)}$ with the (i,j)-th element denoted as

$$
(\tilde{\mathcal{L}}x^{k-2})[i,j] = \sum_{i_3,\cdots,i_k=1}^{n-1} \tilde{\mathcal{L}}_{i,j,i_3,\cdots,i_k} x_{i_3} \cdots, x_{i_k}.
$$

The function vector $\tilde{\mathcal{L}}x^{k-1}$ is the subvetcor of $\mathcal{L}x^{k-1}$, indexed by $[n] \setminus \{j\}$, and the matrix $\tilde{\mathcal{L}}x^{k-2}$ is $[n] \setminus \{j\}$ submatrix of $\mathcal{L}x^{k-1}$.

4.1 The feasible trust region algorithm

Given the current point x_t, the trust region subproblem of (4.2) can be reformulated as follows,

$$
\begin{aligned}
\min_{d\in\mathbb{R}^{n-1}} \quad & m_t(d) = f_t + g_t^{\mathrm{T}} d + \frac{1}{2} d^{\mathrm{T}} W_t\, d, \\
\text{s.t.} \quad & c(x_t) + \nabla c(x_t)^{\mathrm{T}} d = 0, \\
& \|d\| \leq \Delta_t, \\
& x_t + d \geq 0.
\end{aligned} \tag{4.6}
$$

where $f_t = f(x_t)$, $g_t = \nabla_x L(x_t, \lambda_t)$, $W_t = \nabla^2_{xx} L(x_t, \lambda_t)$, Δ_t is the trust region radius updated in (4.11).

In order to facilitate the computation of (4.6), we utilize the following strategies. Firstly, we adopt the ∞-norm in (4.6), and hence all the constrains will be linear. Secondly, at each iteration, the feasibility of x_t implies that $c(x_t) = 0$, which ensures the feasibility of the resulting trust region subproblem.

Consequently, each subproblem is formulated as

$$
\begin{aligned}
\min_{d\in\mathbb{R}^{n-1}} \quad & m_t(d) = f_t + g_t^{\mathrm{T}} d + \frac{1}{2} d^{\mathrm{T}} W_t\, d, \\
\text{s.t.} \quad & \nabla c(x_t)^{\mathrm{T}} d = 0, \\
& \|d\|_\infty \le \Delta_t, \\
& x_t + d \ge 0.
\end{aligned} \tag{4.7}
$$

Specifically, at each iteration, if the trial step d_t is accepted, the iterate $x_t + d_t$ is projected to be feasible by setting $x_{t+1} = P(x_t + d_t)$, where

$$
P(x) = \frac{x}{\|x\|_k} \tag{4.8}
$$

is a projection operator to the k-norm sphere and $\|x\|_k = (\sum_{i=1}^n x_i^k)^{1/k}$ is the k-norm of x. Set

$$
\lambda_t = \nabla f(x_t)^{\mathrm{T}} x_t = \mathcal{A} x_t^m. \tag{4.9}
$$

which is actually the Lagrange multiplier as will be clarified in (5.2).

The following definitions are commonly used in trust region methods. Denote the ratio of actual decrease and predicted decrease as

$$
\rho_t = \frac{f(x_t) - f\left(P(x_t + d_t)\right)}{m_t(0) - m_t(d_t)}. \tag{4.10}
$$

This is an important value for evaluating the error between $m_t(d)$ and $f(x)$ at x_t. If ρ_t is large enough, we are confident to increase the trust region radius Δ_t; but if ρ_t is less than a threshold, we have to decrease the radius. Specifically, Δ_{t+1} is updated as follows

$$
\Delta_{t+1} = \begin{cases} \frac{1}{2}\Delta_t, & \text{if } \rho_t \le \sigma_1; \\ \min\left(\Delta_{\max}, 2\Delta_t\right), & \text{if } \rho_t > \sigma_2; \\ \Delta_t, & \text{else,} \end{cases} \tag{4.11}
$$

where σ_1, σ_2 are constants with $0 < \sigma_1 < \sigma_2$ and $\sigma_1 < 1$. We only update x_t in the next iteration when ρ_t is greater than or equal to some threshold,

$$
x_{t+1} = \begin{cases} P(x_t + d_t), & \text{if } \rho_t \ge \sigma_0; \\ x_t, & \text{else,} \end{cases} \tag{4.12}
$$

where $\sigma_0 \in (0, \sigma_1)$ is a constant. It should be noted that when updated, x_{t+1} is defined as the projection $P(x_t + d_t)$ instead of $x_t + d_t$.

The detailed descriptions of the `FTR` method for computing the analytic connectivity (2.1) of symmetric tensors is as follows. The algorithm includes two steps: the outer step and the inner step. In the outer step, given an index j, let $x[j] = 0$, and compute $\alpha(G) = \min_j \alpha_j(G)$. In the inner step, the problem (4.7) is solved by the feasible trust region algorithm to compute $\alpha_j(G)$.

Algorithm 1: The feasible trust region method for the problem (2.1)

Step 0. Given an initial point x_0, set the parameters $\sigma_0, \sigma_1, \sigma_2, \epsilon$, Δ_0, $\Delta_{\max}$. Let $j = 1$, *iter* =0.

Step 1. For $j = 1, \cdots n$, do

s0. $\lambda_0 = \mathcal{A}x_0^m$ and $t := 0$.

s1. Solve the quadratic problem (4.7) to determine d_t.

s2. If $\|d_t\| \leq \epsilon$, stop and output $(\alpha_j(G) = \lambda_t,\ x^j = x_t)$. Let $iter = iter + t$, and go to **Step 1**.

s3. Calculate ρ_t by (4.10).

s4. Update the trust region radius Δ_t by (4.11).

s5. If $\rho_t \geq \sigma_0$, set $x_{t+1} = P(x_t + d_t)$ and $\lambda_{t+1} = \mathcal{A}x_{t+1}^k$; else $x_{t+1} = x_t$ and $\lambda_{t+1} = \lambda_t$. Set t $:= t + 1$ and go to s0.

Step 2. Let $j^* = \arg\min_{j=1}^n \alpha_j(G)$. Output $(\alpha_{j^*}(G), x^{j^*})$ and *iter*.

It is worth pointing out that if the involved uniform hypergraph has some special structure, such as those discussed in Section 3, then the computation in Algorithm 1 can be significantly reduced since the number of the outer loop can be cut down by merely considering those j of the minimum degree.

5 Convergence analysis

The first-order and the second-order optimality conditions of (4.2) are stated, and the global convergence of Algorithm 1 is established in this section.

5.1 Optimality conditions

For any local minimizer x^* of (4.2), the fact $\nabla c(x^*) = (x^*)^{[k-1]}$ implies that the set $\{\nabla c(x^*)\} \cup \{e_i : i \in \mathcal{I}(x^*)\}$ is linearly independent, where $e_i \in \mathbb{R}^n$ is the identity vector with the i-th element being one while the other elements are zero, and $\mathcal{I}(x^*)$ is the active set of x^*. Thus, the linear independence constraint qualification (LICQ) holds automatically. This observation immediately leads to the following first-order and second-order necessary conditions for (4.2) by invoking Theorems 12.1 and 12.5 in [44].

Lemma 5.1. *(First-order necessary conditions) Suppose that x^* is a local solution of (4.2). Then there is a Lagrange multiplier λ^* such that*

$$\min(x^*, g^*) = 0, \quad c(x^*) = 0, \tag{5.1}$$

where $g^* = \nabla_x L(x^*, \lambda^*) = \nabla f(x)^* - \lambda^* \nabla c(x^*)$. Further, we have

$$\lambda^* = (\nabla f(x)^*)^T x^*. \tag{5.2}$$

Lemma 5.2. *(Second-order necessary condition) Suppose that x^* is a local solution of (4.7). Let λ^* be the Lagrange multiplier satisfying (5.1). Then*

$$d^T W^* d \geq 0, \quad \forall\, d \in \mathcal{C}(x^*, \lambda^*), \tag{5.3}$$

where

$$\mathcal{C}(x^*, \lambda^*) = \{d \mid \nabla c(x^*)^T d = 0;\ d[i] = 0, \forall i \in \mathcal{I}(x^*) \text{ with } g^*[i] > 0;\ d[i] \geq 0, \forall i \in \mathcal{I}(x^*) \text{ with } g^*[i] = 0\}, \tag{5.4}$$

and $W^* = \nabla^2_{xx} L(x^*, \lambda^*)$.

5.2 Global convergence

In this subsection, we establish the global convergence of the inner problem of Algorithm 1; i.e., using feasible trust region algorithm to solve the problem (4.2). We shall employ the techniques in traditional trust region methods to derive the results. However, there are two key difficulties. Firstly, x_{t+1} is updated by $P(x_t + d_t)$ instead of $x_t + d_t$ in order to keep the feasibility. We should estimate the error between $f(P(x_t+d)) - f(x_t)$ with its second order approximation, instead of $f(x_t+d) - f(x_t)$. Secondly, ∞-norm is applied, hence the outline of proof is different from Euclidean-norm cases.

To simplify our analysis, define

$$h(x) = f(P(x)).$$

Then the gradient and the Hessian of $h(x)$ are

$$\nabla h(x) = \nabla P(x) \nabla f(P(x)), \tag{5.5}$$

$$\begin{aligned}\nabla^2 h(x) = & \frac{\nabla P(x) \nabla^2 f(P(x))}{\|x\|_k} - \frac{\nabla c(x) \nabla f(P(x))^{\mathrm{T}}}{\|x\|_k^{k+1}} + \frac{(k+1) \nabla x^{\mathrm{T}} f(P(x)) \nabla c(x) \nabla c(x)^{\mathrm{T}}}{\|x\|_k^{2k+1}} \\ & - \frac{x^{\mathrm{T}} \nabla f(P(x)) \nabla^2 c(x) + \nabla P(x) \nabla^2 f(P(x)) x \nabla c(x)^{\mathrm{T}} + \nabla f(P(x)) \nabla c(x)^{\mathrm{T}}}{\|x\|_k^{k+1}},\end{aligned}$$

where $\nabla P(x) = \left(\frac{I}{\|x\|_k} - \frac{\nabla c(x) x^{\mathrm{T}}}{\|x\|_k^{k+1}} \right)$. A key property is that when $\|x_t\|_k = 1$ and $\nabla c(x_t)^{\mathrm{T}} d = 0$, we have

$$\nabla h(x_t)^{\mathrm{T}} d = \nabla f(x_t)^{\mathrm{T}} d = g(x_t)^{\mathrm{T}} d \tag{5.6}$$

and

$$d^{\mathrm{T}} \nabla^2 h(x_t) d = d^{\mathrm{T}} \nabla^2 f(x_t) d - \lambda d^{\mathrm{T}} \nabla^2 c(x_t) d = d^{\mathrm{T}} W(x_t) x. \tag{5.7}$$

That is, the feasible direction d satisfying $\nabla c(x)^T d = 0$, the second order approximations of $h(x)$ and $L(x, \lambda)$ are the same. Several technical lemmas are presented for the convergence analysis.

Lemma 5.3. *(i) Let $g(x)$ and $W(x)$ are defined in (4.4) and (4.5), respectively. When λ is fixed, for all $x \geq 0$, $y \geq 0$ satisfying $\|x\|_k = 1$ and $\|y\|_k = 1$, we have*

$$\|W(x)\| \leq M, \tag{5.8}$$

$$\|g(x) - g(y)\| \leq L_0 \|x - y\|, \tag{5.9}$$

$$\|W(x) - W(y)\| \leq L_1 \|x - y\|, \tag{5.10}$$

where M, L_0 and L_1 are positive constants.

(ii) Suppose $\|x\|_k \geq \eta_1$, $\|y\|_k \geq \eta_2$, where η_1 and η_2 are positive constants. We have

$$\|\nabla^2 h(x) - \nabla^2 h(y)\| \leq L_2 \|x - y\|, \tag{5.11}$$

where L_2 is a positive constant.

Proof. They are obvious since $g(x)$, $W(x)$ and $\nabla^2 h(x)$ are smooth and bounded on the closed sets. **Q.E.D.**

Lemma 5.4. *Suppose x_t is feasible solution of model (4.2), and d_t is feasible solution of model (4.7). For the error between the models $m_t(d_t)$ and $h(x_t + d_t)$, we have*

$$|m_t(d_t) - h(x_t + d_t)| \le \beta \|d_t\|^3, \tag{5.12}$$

where β is some positive constant.

Proof. By the mean value theorem for integration, we have

$$h(x_t + d_t) = h(x_t) + \nabla h(x_t)^{\mathrm{T}} d_t + \frac{1}{2} d_t^{\mathrm{T}} \nabla^2 h(x_t + \theta_t d_t) d_t$$

for some $\theta_t \in (0, 1)$. It follows from $h(x_t) = f(x_t)$, (5.6) and (5.7) that

$$\begin{aligned} |m_t(d_t) - h(x_t + d_t)| &= \left| \frac{1}{2} d_t^{\mathrm{T}} W_t d_t - \frac{1}{2} d_t^{\mathrm{T}} \nabla^2 h(x_t + \theta_t d_t) d_t \right| \\ &= \left| \frac{1}{2} d_t^{\mathrm{T}} \nabla^2 h(x_t) d_t - \frac{1}{2} d_t^{\mathrm{T}} \nabla^2 h(x_t + \theta_t d_t) d_t \right| \\ &\le \frac{1}{2} L_2 \|d_t\|^3. \end{aligned}$$

To show the above inequality by Lemma 5.3 (*ii*), we still need to prove $\|x_t\|_k$ and $\|x_t + \theta d_t\|_k$ are positive. The feasible point x_t satisfies $\|x_t\|_k = 1$. As two norms are equivalent, i.e., for $x \in \mathbb{R}^n$ if $r_1 > r_2 > 0$, then

$$\|x\|_{r_1} \le \|x\|_{r_2} \le n^{\frac{1}{r_2} - \frac{1}{r_1}} \|x\|_{r_1}.$$

Hence, it follows from $\|x_t\|_k = 1$ that for $k \ge 3$, $\|\nabla c(x_t)\| = \|x_t^{[k-1]}\| = \|x_t\|_{2k-2}^{k-1} \le 1$. Furthermore, it follows from $\nabla c(x_t)^{\mathrm{T}} d_t = 0$, $\nabla c(x_t) = x_t^{[k-1]}$ and $\nabla c(x_t)^{\mathrm{T}} x_t = \|x_t\|_k^k = 1$ that $\nabla c(x_t)^{\mathrm{T}}(x_t + \theta_t d_t) = 1$. Therefore,

$$\|\nabla c(x_t)\| \cdot \|x_t + \theta_t d_t\| \ge 1.$$

As a result, both x_t and $x_t + \theta_t d_t$ are lower bounded. **Q.E.D.**

Lemma 5.5. *Consider the sequence $\{x_t\}$ generated by Algorithm 1. Then sequence $\{f(x_t)\}$ of the objective value is nondecreasing. Furthermore, at least one of the cluster points of $\{x_t\}$ is a KKT points of the problem (4.2), i.e.,*

$$\liminf_{t \to \infty} \| \min(x_t, \nabla f(x_t) - \lambda_t \nabla c(x_t)) \| = 0. \tag{5.13}$$

Proof. Suppose the theorem is false, we assume that

$$\lim_{k \to \infty} \Delta_t = 0. \tag{5.14}$$

If (5.14) fails, there exists a const $\delta > 0$, such that for infinite many t, it holds that

$$\Delta_t \ge \delta \quad \text{and} \quad \rho_t \ge \sigma_1. \tag{5.15}$$

Denote the set of k satisfying (5.15) as K_0. Without loss of generality, suppose

$$\lim_{t \in K_0, t \to \infty} x_t = \bar{x}.$$

According to our assumption, $\bar{x}$ is not a stationary point of (4.2), hence $d = 0$ is not the optimal solution of the following system

$$\begin{aligned} \min_{d \in \mathbb{R}^{n-1}} \quad & \bar{m}(d) = f(\bar{x}) + g(\bar{x})^{\mathrm{T}} d + \frac{1}{2} d^{\mathrm{T}} W(\bar{x})\, d, \\ \text{s.t.} \quad & \nabla c(\bar{x})^{\mathrm{T}} d = 0, \\ & \|d\|_\infty \le \delta, \\ & \bar{x} + d \ge 0. \end{aligned} \tag{5.16}$$

Denote $\bar{d}$ as its solution, then

$$\gamma = \bar{m}(0) - \bar{m}(\bar{d}) = -g(\bar{x})^{\mathrm{T}}\bar{d} - \frac{1}{2}\bar{d}^{\mathrm{T}}W(\bar{x})\,\bar{d} > 0.$$

It follows from Lemma 5.6 that

$$m_t(0) - m_t(d_t) \geq \frac{1}{2}(\bar{m}(0) - \bar{m}(\bar{d})) \geq \frac{1}{2}\gamma$$

for all $t \in K_0$ large enough. As a result, $f(x_t) - f(x_{t+1}) \geq \frac{1}{2}\sigma_1\gamma > 0$ for all large enough $t \in K_0$. This contradicts to $\lim_{t\to\infty} f(x_t) = f(\bar{x})$. The contradiction indicates that (5.14) holds.

If (5.14) holds, there exists a subsequence such that

$$\rho_t \leq \sigma_1, \quad \forall\, t \in K_1. \tag{5.17}$$

Without loss of generality, suppose

$$\lim_{t\in K_1, t\to\infty} x_t = \hat{x}, \tag{5.18}$$

According to our assumption, $\hat{x}$ is not a stationary point of (4.2), hence $d = 0$ is not the optimal solution of the following system

$$\begin{aligned} \min_{d\in\mathbb{R}^{n-1}} \quad & \hat{m}(d) = f(\hat{x}) + g(\hat{x})^{\mathrm{T}}d + \frac{1}{2}d^{\mathrm{T}}W(\hat{x})\,d \\ \text{s.t.} \quad & \nabla c(\hat{x})^{\mathrm{T}}d = 0, \\ & \|d\|_\infty \leq 1, \\ & \hat{x} + d \geq 0. \end{aligned} \tag{5.19}$$

Denote $\hat{d}$ as its solution, then

$$\hat{\gamma} = \hat{m}(0) - \hat{m}(\hat{d}) = -g(\hat{x})^{\mathrm{T}}\hat{d} - \frac{1}{2}\hat{d}^{\mathrm{T}}W(\hat{x})\,\hat{d} > 0.$$

As $\hat{d}_t = \Delta_t\hat{d}$ is the solution of (5.19) with the trust region radius replaced by Δ_t. Then $\hat{m}(0) - \hat{m}(\hat{d}_t) \geq \frac{1}{2}\Delta_t\hat{\gamma}$. It follows from Lemma 5.6 that

$$m_t(0) - m_t(d_t) \geq \frac{1}{2}(\hat{m}(0) - \hat{m}(\hat{d}_t)) \geq \frac{1}{4}\Delta_t\hat{\gamma} \tag{5.20}$$

for all $t \in K_1$ large enough, where the last inequality comes from $\Delta_t \to 0$. Further,

$$\begin{aligned} \rho_t &\geq 1 - |1 - \rho_t| \\ &= 1 - \frac{|m_t(0) - m_t(d_t) + h(x_t + d_t) - h(x_t)|}{|m_t(0) - m_t(d_t)|} \\ &= 1 - \frac{|h(x_t + d_t) - m_t(d_t)|}{|m_t(0) - m_t(d_t)|} \\ &\geq 1 - \frac{\beta\|d_t\|^3}{|m_t(0) - m_t(d_t)|}. \end{aligned} \tag{5.21}$$

This, together with (5.20), derives $\lim_{t\in K_1, t\to\infty} \rho_t = 1$, which contradicts with (5.17). This completes the proof. **Q.E.D.**

Lemma 5.6. *The optimal value of (5.16) is continuous for all feasible points $\bar{x}$ of (4.2). Namely, given two points x_{t_1} and x_{t_2} satisfying $\|x_{t_i} - \bar{x}\| \leq \epsilon_0,\ i = 1, 2$ with $x_{t_i} \geq 0,\ \|x_{t_i}\|_k = 1,\ i = 1, 2$, their optimal solution for (5.16) are d_{t_1} and d_{t_2}, respectively. Then, for all $\epsilon > 0$ small enough, it holds that*

$$|g(x_{t_1})^T d_{t_1} + \frac{1}{2}d_{t_1}^T W(x_{t_1})d_{t_1} - g(x_{t_2})^T d_{t_2} - \frac{1}{2}d_{t_2}^T W(x_{t_2})d_{t_2}| \leq \epsilon. \tag{5.22}$$

Proof. As $\bar{x}$ satisfies $\sum_i \bar{x}[i]^k = 1$, there exists at least an index p such that $\bar{x}[p] > 0$. For two points x_{t_1} and x_{t_2} near $\bar{x}$, there exists a positive value ϵ_1 such that

$$\|\nabla c(x_{t_2})^{\mathrm{T}} d_{t_1}\| = \|\nabla c(x_{t_2})^{\mathrm{T}} d_{t_1} - \nabla c(x_{t_1})^{\mathrm{T}} d_{t_1}\| \leq \|d_{t_1}\| \|\nabla c(x_{t_2}) - \nabla c(x_{t_1})\| \leq \epsilon_1,$$

where the last inequality follows from that d_{t_1} is bounded, and $\nabla c(x)$ is continuous. If $d_{t_1}[p] < \delta$ and $\nabla c(x_{t_2})^{\mathrm{T}} d_{t_1} > 0$ or $d_{t_1}[p] > -\delta$ and $\nabla c(x_{t_2})^{\mathrm{T}} d_{t_1} < 0$, then $\tilde{d}_{t_2} = d_{t_1} - \frac{\nabla c(x_{t_2})^{\mathrm{T}} d_{t_1}}{\nabla c(x_{t_2})^{\mathrm{T}} e_p} e_p$ is a feasible solution for

$$\begin{aligned} \chi_{t_2} = \min_{d \in \mathbb{R}^n} \quad & g(x_{t_2})^{\mathrm{T}} d + \frac{1}{2} d^{\mathrm{T}} W(x_{t_2})\, d, \\ \text{s.t.} \quad & \nabla c(x_{t_2})^{\mathrm{T}} d = 0, \\ & \|d\|_\infty \leq \delta, \\ & x_{t_2} + d \geq 0. \end{aligned} \tag{5.23}$$

Otherwise, suppose that $d_{t_1}[p] = \delta(-\delta)$, from $\nabla c(x_{t_1})^{\mathrm{T}} d_{t_1} = 0$ that there exists some positive index q such that $\bar{x}[q] > 0$ and $d_{t_1}[q] < (>) 0$, hence $\tilde{d}_{t_2} = d_{t_1} - \frac{\nabla c(x_{t_2})^{\mathrm{T}} d_{t_1}}{\nabla c(x_{t_2})^{\mathrm{T}} e_q} e_q$ is a feasible solution for the above problem. Therefore, from the fact that the objective function of (5.23) is continuous and that $\tilde{d}_{t_2}$ is only a feasible solution, we have

$$\chi_{t_2} \leq g(x_{t_2})^{\mathrm{T}} \tilde{d}_{t_2} + \frac{1}{2} \tilde{d}_{t_2}^{\mathrm{T}} W(x_{t_2})\, \tilde{d}_{t_2} \leq \chi_{t_1} + \epsilon.$$

On the other hand, we can show $\chi_{t_2} + \epsilon \geq \chi_{t_1}$. Therefore, (5.22) holds true. **Q.E.D.**

Theorem 5.7. *Suppose that the iterates $\{x_t\}$ generated by Algorithm 1 converge to x^*. Then the second-order necessary conditions (5.3) holds.*

Proof. We show this theorem by contradiction. Suppose that there exists a negative eigenvalue $-\eta_0$ satisfying

$$v^{\mathrm{T}} W^* v = -\eta_0 < 0, \quad \text{where } v \in \mathcal{C}(x^*, \lambda^*), \quad \|v\|_2 = 1. \tag{5.24}$$

It follows from the definition of (5.4) that v is a feasible solution of (4.7) with x_t replaced by x^*, and Δ_t replaced by 1. For all $i \in \mathcal{I}(x^*)$, either $g^*[i] = 0$ or $v[i] = 0$, and for all $i \notin \mathcal{I}(x^*)$, $g^*[i] = 0$, hence

$$(g*)^{\mathrm{T}} v = 0. \tag{5.25}$$

When x_t is close enough to x^*, it follows from the proof of Lemma 5.6 and that $\hat{d}_t = \Delta_t v + d_t^\epsilon$ is a feasible point for the problem (4.7), where $\|d_t^\epsilon\|$ is small enough to be bounded by $\|x_t - x^*\|$. Furthermore, it follows from (5.25) that $g_t^{\mathrm{T}} \hat{d}_t$ is small, $v^{\mathrm{T}} W^* v = -\eta_0 < 0$. Hence $\hat{d}_t$ is an decrease direction for the problem (4.7). Therefore,

$$\begin{aligned} m_t(0) - m_t(\hat{d}_t) &= -g_t^{\mathrm{T}} \hat{d}_t - \frac{1}{2} d^{\mathrm{T}} W_t \hat{d}_t \\ &= -\Delta_t g_t^{\mathrm{T}} v - \frac{1}{2} \Delta_t^2 v^{\mathrm{T}} W_t v + o(1) \\ &\geq -\frac{1}{4} \Delta_t^2 v^{\mathrm{T}} W_t v. \end{aligned}$$

Since $|v^{\mathrm{T}} W_t v - v^{\mathrm{T}} W^* v| \leq \|W_t - W^*\| \|v\|^2$, $\|v\| = 1$, then $v^{\mathrm{T}} W_t v \leq -\frac{1}{2}\eta_0$ and

$$m_t(0) - m_t(d_t) \geq m_t(0) - m_t(\hat{d}_t) \geq -\frac{1}{4} \Delta_t^2 v^{\mathrm{T}} W_t v \geq \frac{1}{8} \Delta_t^2 \eta_0. \tag{5.26}$$

It follows from (5.21) that $\rho_t \to 1$. Therefore, there exists K_2 large enough such that

$$f(x_t) - f(x_{t+1}) \geq \sigma_1 (m_t(0) - m_t(d_t)) \geq \frac{1}{8} \Delta_t^2 \sigma_1 \eta_0, \quad \forall\, k \geq K_2, \tag{5.27}$$

which derives that $\Delta_t \to 0$. This contradicts with $\rho_t \to 1$. Thus, (5.24) is false. **Q.E.D.**

6 Numerical experiments

In this section, we present several numerical results of computing the analytic connectivity. Our codes are implemented in MATLAB (R2014a). All the experiments are preformed on a Dell desktop with Intel dual core i7-4770 CPU at 3.40 GHz and 8GB of memory running Windows 7. The parameters are set as

$$\sigma_0 = 0.25, \quad \sigma_1 = 0.5, \quad \sigma_2 = 0.75, \quad \epsilon = 1.0^{-8}, \quad \Delta_0 = 2, \quad \Delta_{\max} = 10.$$

We execute the `FTR` algorithm 100 times with different initial points, and report the average results. The initial points are generated by the following `Matlab` commands

```
for rd = 1:100;    randn('seed', rd);    x0 = randn(n-1,1);    end;
```

which obey the Gaussian distribution. Afterwards, x_0 is restricted to the feasible set of (4.2) by doing the projection $P(|x_0|)$.

`FTR` is compared with an Sparse Nonlinear OPTimizer solver `SNOPT` [52], which is called by the free trial software `TOMLAB` [1]. The exact gradient and the Hessian are provided for `FTR` and `SNOPT`, and both the quadratic programming subproblems of `FTR` and `SNOPT` are computed by `SQOPT`. Furthermore, for small dimensional problems, we utilize the global optimization software `GloptiPoly 3` [33] [2] to solve (4.2), which can help us to judge whether our solution is the global optimal solution. `GloptiPoly 3` relaxes the polynomial problem into a hierarchy of semidefinite subproblems, which are solved by `SDPNAL+` [60].

Noting that the main computation of `FTR` includes calculating $\mathcal{L}x^k$, $\mathcal{L}x^{k-1}$ and $\mathcal{L}x^{k-2}$. To deal with this, we adopt the methods in Chang, Chen and Qi [8] to calculate $\mathcal{L}x^k$, $\mathcal{L}x^{k-1}$, where they store a uniform hypergraph by a compact matrix $G_r \in \mathbb{R}^{m\times k}$, where m is the number of edges, and k is the number of vertices in an edge; namely, the i-th edges of the hypergraph is the i-th row of G_r as

$$G(i,:) = (v_{i_1}, \cdots, v_{i_k}).$$

The computational method for $\mathcal{L}x^{k-2}$ follows the same strategy. Thus, the computation cost for $\mathcal{L}x^k$, $\mathcal{L}x^{k-1}$, $\mathcal{L}x^{k-2}$ are $O(mk)$, $O(mk^2 + mnk)$ and $O(mk^3 + mn^2k^2)$, respectively. It should also be noted that the sparsity ratio of $\mathcal{L}x^{k-2}$ is

$$nnz(\mathcal{L}x^{k-2}) = O\left(\frac{mk^2}{n^2}\right).$$

Thus our method enjoys fast computation when the sparsity property is utilized.

6.1 Comparison of `FTR` with `SNOPT` and `GloptiPoly 3` for small size hypergraphs

In this subsection, we show the numerical results of our `FTR` algorithm, compared with `SNOPT` and `GloptiPoly 3`. We will use the hypergraphs in Figure 6.1 which are found in [8, 31, 32, 50] as the testing instances.

In Table 6.1, 'm' is the number of edges of the hypergraph, 'n' is the number of vertices, k is the number of vertices in an edge. 'α' means the analytic connectivity returned by `FTR` and `SNOPT`, 'α^*' stands for the analytic connectivity computed from the global optimization software `GloptiPoly 3`, 'ratio' means the ratio `FTR` and `SNOPT` get the same result with `GloptiPoly 3`, and 'iter' is the average number of iterations of 100 runs with random initializations. 'time (s)' denotes the average CPU time of seconds consumed by `FTR` and `SNOPT`, or the total CPU time of `GloptiPoly 3`.

[1] http://tomopt.com/tomlab/

[2] http://homepages.laas.fr/henrion/software/gloptipoly/

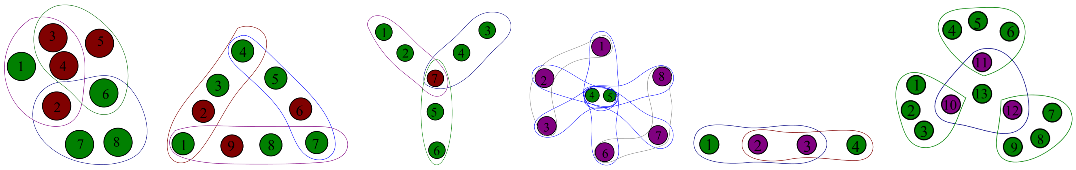

Figure 6.1: The uniform hypergraphs (a)–(f)

Table 6.1: Comparisons of `FTR` with `SNOPT` and `GloptiPoly 3`

		SNOPT				FTR				GloptiPoly 3	
Hypergraph	(m, n, k)	α	ratio	iter	time (s)	α	ratio	iter	time (s)	α^*	time (s)
(a)	(3, 8, 4)	0.2516	100%	323.61	0.3065	0.2516	100%	75.42	0.0332	0.2516	59.515
(b)	(3, 9, 4)	0.2100	100%	403.64	0.3619	0.2100	100%	83.65	0.0365	0.2100	110.14
(c)	(3, 7, 3)	0.1607	100%	142.15	0.1007	0.1607	97%	48.15	0.0185	0.1607	74.136
(d)	(8, 8, 3)	0.4300	100%	151.46	0.1216	0.4300	100%	67.03	0.0263	0.4300	110.10
(e)	(2, 4, 3)	0.5344	100%	42.28	0.0381	0.5344	100%	25.28	0.0080	0.5344	23.052
(f)	(4, 13, 4)	0.0592	100%	850.18	0.8496	0.0592	97%	131.77	0.0603	0.0592	18.877

Table 6.1 shows that both `SNOPT` and `FTR` produce the same results with `GloptiPoly3` for almost 100%. This is in accord with Theorem 5.7 that `FTR` converges to second order necessary points, which has a high possibility to converge to global optimal point. Besides, the average iteration number that `FTR` takes is relatively small comparing to that of `SNOPT`, since `FTR` has utilized the trust region technique. As the main computation costs in each iteration for both `FTR` and `SNOPT` are to solve the quadratic programming, this makes `FTR` take less CPU time than `SNOPT`, as one can see from Table 6.1. Additionally, it is known from Table 6.1 that, among the above six hypergraph instances, the hypergraph (f) has the smallest analytic connectivity, while (d) and (e) have relatively large ones. This, to some extent, reflects the connectivity of the corresponding hypergraphs as can be seen from Figure 6.1.

6.2 Larger dimensional problems

In this subsection, we are ready to compute relatively large dimensional problems by `FTR`, and compare its performance with that of `SNOPT`. As `GloptiPoly 3` will be too costly both in time and in space for large problems, we will not consider this algorithm here. Similar to the small dimensional cases, we also give 100 initial points, and show the overall and average results. We take the 2-path 4-graph as discussed in Proposition 3.3 and K_n- in Proposition 3.4 for testing instances with different values of n. The computational results are shown in Tables 6.2 and 6.3, where 'α' is the analytic connectivity in question, and 'ratio' stands for the percentage from 100 experiments to achieve that minimal value.

We can see from Table 6.2 that both `FTR` and `SNOPT` produce the same optimal value for each of the above instances, and the successful ratio is above 70%, while `FTR` is slightly better than `SNOPT`. Comparing to those small size problems as computed in Subsection 6.1, large dimensional problems here are relatively hard to achieve the global optimum with local optimal algorithms such as `FTR` and `SNOPT`. For the iteration number, we find that `FTR` scales well for dimension as large as 500, while `SNOPT` takes

Table 6.2: Results for the 2-path 4-graphs with different n by `FTR` and `SNOPT`

	`FTR`				`SNOPT`			
n	α	ratio	iter	time (s)	α	ratio	iter	time (s)
10	1.21e-01	100%	11.67	0.0058	1.21e-01	100%	48.66	0.0408
50	4.11e-03	92%	12.46	0.0095	4.11e-03	95%	186.16	0.1426
100	1.01e-03	82%	15.00	0.0233	1.01e-03	86%	268.11	0.2554
200	2.49e-04	98%	14.92	0.0872	2.49e-04	79%	534.81	1.3374
300	1.10e-04	95%	14.86	0.2274	1.10e-04	73%	816.92	4.6972
400	6.20e-05	96%	14.50	0.4935	6.20e-05	87%	1039.38	11.7781
500	3.96e-05	94%	14.71	0.9096	3.96e-05	89%	1329.87	26.4040

far more iteration steps for larger dimensional problems. This leads to overwhelming superiority of `FTR` in computation time comparing to `SNOPT`, as one can see from Table 6.2. Besides, it is worth pointing out that the sparse ratio of the Hessian matrix for this problem is about $O(\frac{1}{n})$, and both the quadratic subproblems of `FTR` and `SNOPT` have taken this advantage. Thus, the overall computation time is not long even when the iteration number as big as more than 1000. Additionally, we can see that as n increases, $\alpha(G)$ is monotonically decreasing, which fits the result in Proposition 3.3.

The numerical results for K_n^- with $k = 3$ and different values of n are shown in Table 6.3, with the comparison on performances of `FTR` and `SNOPT`, and the upper bounds $\bar{\alpha} = n - 2 - \frac{2}{n-1}$ given in (2.3). As already known from Proposition 3.4, $\alpha(K_n^-) = \min_{j=1,\cdots,k} \alpha_j(K_n^-)$. Combining with the inherited symmetric structure of K_n^-, we only need to compute $\alpha_1(K_n^-)$.

Table 6.3: Numerical results for K_n^- with different n by `FTR` and `SNOPT`

	`FTR`				`SNOPT`				upper bound
n	α	ratio	iter	time (s)	α	ratio	iter	time (s)	$\bar{\alpha}$
10	7.7736	100%	6.82	0.0031	7.7736	100%	30.01	0.0262	7.7778
20	17.8943	100%	7.27	0.0072	17.8943	100%	17.56	0.0226	17.8947
30	27.9309	100%	8.03	0.0242	27.9309	100%	14.48	0.0455	27.9310
40	37.9487	100%	8.67	0.0764	37.9487	100%	13.29	0.1578	37.9487
50	47.9592	100%	8.54	0.2082	47.9592	100%	14.72	0.5159	47.9592
60	57.9661	100%	8.38	0.4900	57.9661	100%	15.18	1.8829	57.9661
70	67.9710	100%	8.01	1.6986	67.9710	100%	15.85	7.1758	67.9710
80	77.9747	100%	8.00	3.2806	77.9747	100%	14.80	20.4195	77.9747
90	87.9775	100%	8.01	6.1458	87.9775	100%	15.09	45.8924	87.9775
100	97.9798	100%	8.00	13.7736	97.9798	100%	15.42	89.7867	97.9798

From Table 6.3, we can see that `FTR` takes less iterations and hence less CPU time than that of `SNOPT`, and the upper bound given in (2.3) is quite tight as it is pretty close to the value from computation. In addition, as the hypergraph K_n^- is well connected by definition, the analytic connectivity is relatively high comparing to all the others in this section, which again verify that the analytic connectivity is a good choice to measure the connectivity of hypergraphs. However, as one can see from Tables 6.2 and 6.3, big analytic connectivities of hypergraphs result in more CPU time for the corresponding hypergraphs with the same n.

7 Conclusions

In this paper, we have exploited properties on the analytic connectivity and have shown that several structured uniform hypergraphs attain their analytic connectivities at vertices of the minimum degrees. To efficiently compute the analytic connectivity of any general uniform hypergraph, we have proposed a feasible trust region algorithm with global convergence, and have conducted numerical experiments to shown the advantages of our algorithm in comparison of other existing ones. All the numerical results have verified that the analytic connectivity is a good choice to measure the connectivity of a hypergraph. Moreover, the efficiency of the proposed algorithm makes the extended version of "Cheeger inequality" in the setting of uniform hypergraphs practically feasible to efficient bound the Cheeger numbers of uniform hypergraphs.

Acknowledgements

The first author thank Professor Wei Li and Miss Lizhu Sun for their useful discussions on Conjecture 3.1. The first, third and forth authors were supported in part by the Research Grants Council (RGC) of Hong Kong (Project C1007-15G), the second author's work was supported by the National Natural Science Foundation of China (11301022,11431002), and the third author's work was supported by Grants No. PolyU 501212, 501913, 15302114 and 15300715.

References

[1] C. Berge, *Hypergraphs, Combinatorics of Finite Sets*, 3rd edn., North-Holland, 1989.

[2] A. Bretto, *Hypergraph Theory: An Introduction*, Springer, 2013.

[3] A.E. Brouwer and W.H. Haemers, *Spectra of Graphs*, Springer, 2011.

[4] R. H. Byrd, P. Lu, J. Nocedal, et al. "A limited memory algorithm for bound constrained optimization", *SIAM Journal on Scientific Computing* **16(5)** (1995) 1990-1208.

[5] C. Bu, Y. Fan and J. Zhou, "Laplacian and signless Laplacian Z-eigenvalues of uniform hypergraphs", *Frontiers of Mathematics in China* (2015) 1-10.

[6] C. Bu, J. Zhou and Y. Wei, "E-cospectral hypergraphs and some hypergraphs determined by their spectra", *Linear Algebra and Its Applications* **459** (2014) 397-403.

[7] S.R. Bulò and M. Pelillo, "New bounds on the clique number of graphs based on spectral hypergraph theory", *International Conference on Learning and Intelligent Optimization*, Springer Verlag, Berlin (2009) 45-58.

[8] J. Chang, Y. Chen and L. Qi, "Computing eigenvalues of large scale sparse tensors arising from a hypergraph", to appear in: *SIAM Journal on Scientific Computing* (2016).

[9] Z. Chen and L. Qi, "Circulant tensors with applications to spectral hypergraph theory and stochastic process", *Journal of Industrial and Management Optimization* **12** (2016) 1227-1247.

[10] F.R.K. Chung, *Spectral Graph Theory*, American Mathematical Society, 1997.

[11] A.R. Conn, N.M. Gould and P.L. Toint, "A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds", *SIAM Journal on Numericl Analysis* **28** (1991) 545-572.

[12] J. Cooper and A. Dutle, "Spectra of uniform hypergraphs", *Linear Algebra and Its Applications* **436** (2012) 3268-3292.

[13] J. Cooper and A. Dutle, "Computing hypermatrix spectra with the Poisson product formula", *Linear and Multilinear Algebra* **63** (2015) 956-970.

[14] R. Cui, W. Li and M. Ng, "Primitive tensors and directed hypergraphs", *Linear Algebra and Its Applications* **471** (2015) 96-108.

[15] D.M. Cvetković, M. Doob, I. Gutman and A. Torgašev, *Recent Results in the Theory of Graph Spectra*, North Holland, Amsterdam, 1988.

[16] D.M. Cvetković, M. Doob and H. Sachs, *Spectra of Graphs, Theory and Application*, Academic Press, 1980.

[17] P. H. Calamai, J. J. Moré, "Projected gradient methods for linearly constrained problems", *Mathematical Programming* **39** (1987) 93-116.

[18] Y. Dai , R. Fletcher, "New algorithms for singly linearly constrained quadratic programs subject to lower and upper bounds", *Mathematical Programming* **106** (2006) 403-421.

[19] M. Fiedler, "Algebraic connectivity of graphs", *Czechoslovak mathematical journal* **23** (1973) 298-305.

[20] Y. Fan, Y. Tan, X. Peng and A. Liu, "Maximizing spectral radii of uniform hypergraphs with few edges", to appear in: *Discussiones Mathematicae Graph Theory* (2016).

[21] M. P. Friedlander and M. A. Saunders, "A globally convergent linearly constrained lagrangian method for nonlinear optimization", *SIAM Journal on Optimization* **15** (2005) 863-897.

[22] D. Ghoshdastidar and A. Dukkipati, "Consistency of spectral partitioning of uniform hypergraphs under planted partition model", *Advances in Neural Information Processing Systems* (2014) 397-405.

[23] D. Ghoshdastidar and A. Dukkipati, "A Provable Generalized Tensor Spectral Method for Uniform Hypergraph Partitioning", *Proceedings of the 32nd International Conference on Machine Learning*, Lille, France (2015) 400-409.

[24] S. Gratton, M. Mouffe, P. L. Toint and M. Weber-Mendonca, " A recursive-trust-region method for bound-constrained nonlinear optimization", *IMA Journal of Numerical Analysis* **28(4)** (2008), 827-861.

[25] W. Hager, B. Mair and H. Zhang, "An affine-scaling interior-point CBB method for box-constrained optimization", *Mathematical Programming* **119** (2009) 1-32.

[26] W. Hager and H. Zhang, "A New Active Set Algorithm for Box Constrained Optimization", *SIAM Journal on Optimization* **17** (2006) 526-557.

[27] C. Hao, C. Cui and Y. Dai, "A feasible trust region method for calculating extreme Z-eigenvalues of symmetric tensors", *Pacific Journal of Optimization* **11** (2015) 291-307.

[28] S. Hu and L. Qi, "Algebraic connectivity of an even uniform hypergraph", *Journal of Combinatorial Optimization* **24** (2012) 564-579.

[29] S. Hu and L. Qi, "The eigenvectors associated with the zero eigenvalues of the Laplacian and signless Laplacian tensors of a uniform hypergraph", *Discrete Applied Mathematics* **169** (2014) 140-151.

[30] S. Hu and L. Qi, "The Laplacian of a uniform hypergraph", *Journal of Combinatorial Optimization* **29** (2015) 331-366.

[31] S. Hu, L. Qi and J. Shao, "Cored hypergraphs, power hypergraphs and their Laplacian eigenvalues", *Linear Algebra and Its Applications* **439** (2013) 2980-2998.

[32] S. Hu, L. Qi and J. Xie, "The largest Laplacian and signless Laplacian H-eigenvalues of a uniform hypergraph", *Linear Algebra and Its Applications* **469** (2015) 1-27.

[33] D. Henrion, J. Lasserre and J. Löfberg, "GloptiPoly 3: moments, optimization and semidefinite programming", *Optimization Methods & Software* **24** (2009) 761-779.

[34] L. Kang, V. Nikiforov and X. Yuan, "The p-spectral radius of k-partite and k-chromatic uniform hypergraphs", *Linear Algebra and Its Applications* **478** (2015) 81-107.

[35] M. Khan and Y. Fan, "On the spectral radius of a class of non-odd-bipartite even uniform hypergraphs", *Linear Algebra and Its Applications* **480** (2015) 93-106.

[36] M. Khan, Y. Fan and Y. Tan, "The H-spectra of a class of generalized power hypergraphs", *Discrete Mathematics* **339** (2016) 1682-1689.

[37] G. Li, L. Qi and G. Yu, "The Z-eigenvalues of a symmetric tensor and its application to spectral hypergraph theory", *Numerical Linear Algebra with Applications* **20** (2013) 1001-1029.

[38] H. Li, J. Shao and L. Qi, "The extremal spectral radii of k-uniform supertrees", *Journal of Combinatorial Optimization* **32** (2016) 741-764.

[39] X.L. Li, Y.T. Shi and I. Gutman, *Graph Energy*, Springer, 2012.

[40] W. Li, J. Cooper and A. Chang, "Analytic connectivity of k-uniform hypergraphs", *Linear and Multilinear Algebra*, doi/abs/10.1080/03081087.2016.1234575 (2016).

[41] L-H. Lim, "Singular values and eigenvalues of tensors: A variational approach", *Proceedings of the 1st IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, December 13-15 (2005) 129-132.

[42] V. Nikiforov, "Analytic methods for uniform hypergraphs", *Linear Algebra and Its Applications* **457** (2014) 455-535.

[43] V. Nikiforov, "Some extremal problems for hereditary properties of graphs", *The Electronic Journal of Combinatorics* **21** (2014) 1-17.

[44] J. Nocedal and S. Wright, *Numerical optimization*, Springer Science & Business Media, 2006.

[45] K. Pearson, "Spectral hypergraph theory of the adjacency hypermatrix and matroids", *Linear Algebra and Its Applications* **465** (2015) 176-187.

[46] K. Pearson and T. Zhang, "Eigenvalues on the adjacency tensor of products of hypergraphs", *International Journal on Contemporary Mathematical Sciences* **8** (2013) 151-158.

[47] K. Pearson and T. Zhang, "On spectral hypergraph theory of the adjacency tensor", *Graphs and Combinatorics* **30** (2014) 1233-1248.

[48] K. Pearson and T. Zhang, "The Laplacian tensor of a multi-hypergraph", *Discrete Mathematics* **338** (2015) 972-982.

[49] L. Qi, "Eigenvalues of a real supersymmetric tensor", *Journal of Symbolic Computation* **40** (2005) 1302-1324,

[50] L. Qi, "H^+-Eigenvalues of Laplacian and signless Lapaclian tensors", *Communications in Mathematical Sciences* **12** (2014) 1045-1064.

[51] L. Qi, J. Shao and Q. Wang, "Regular uniform hypergraphs, s-cycles, s-paths and their largest Laplacian H-eigenvalues", *Linear Algebra and Its Applications* **443** (2014) 215-227.

[52] P. E. Gill, M. Walter, and M. A. Saunders. "SNOPT: An SQP algorithm for large-scale constrained optimization", *SIAM review* **47** (2005) 99-131.

[53] J. Shao, L. Qi and S. Hu, "Some new trace formulas of tensors with applications in spectral hypergraph theory", *Linear and Multilinear Algebra* **63** (2015) 971-992.

[54] J. Shao, H. Shan and B. Wu, "Some spectral properties and characterizations of connected odd-bipartite uniform hypergraphs", *Linear and Multilinear Algebra* **63** (2015) 2359-2372.

[55] J. Xie and A. Chang, "On the Z-eigenvalues of the signless Laplacian tensor for an even uniform hypergraph", *Numerical Linear Algebra with Applications* **20** (2013) 1030-1045.

[56] J. Xie and A. Chang, "On the Z-eigenvalues of the adjacency tensors for uniform hypergraphs", *Linear Algebra and Its Applications* **430** (2013) 2195-2204.

[57] J. Xie and A. Chang, "H-eigenvalues of the signless Laplacian tensor for an even uniform hypergraph", *Frontiers of Mathematics in China* **8** (2013) 107-128.

[58] J. Xie and L. Qi, "The clique and coclique numbers' bounds based on the H-eigenvalues of uniform hypergraphs", *International Journal of Numerical Analysis & Modeling* **12** (2015) 318-327.

[59] J. Xie and L. Qi, "Spectral directed hypergraph theoy via tensors", *Linear and Multilinear Algebra* **64** (2016) 780-794.

[60] L. Yang, D. Sun and K. Toh, "SDPNAL+: a majorized semismooth Newton-CG augmented Lagrangian method for semidefinite programming with nonnegative constraints", *Mathematical Programming Computation* **7** (2015) 331-366.

[61] X. Yuan, L. Qi and J. Shao, "The proof of a conjecture on largest Laplacian and signless Laplacian H-eigenvalues of uniform hypergraphs", *Linear Algebra and Its Applications* **490** (2016) 18-30.

[62] X. Yuan, J. Shao and H. Shan, "Ordering of some uniform supertrees with larger spectral radii", *Linear Algebra and Its Applications* **495** (2016) 206-222.

[63] X. Yuan, M. Zhang and M. Lu, "Some upper bounds on the eigenvalues of uniform hypergraphs", *Linear Algebra and Its Applications* **484** (2015) 540-549.

[64] J. Yue, L. Zhang and M. Lu, "The largest adjacency, signless Laplacian, and Laplacian H-eigenvalues of loose paths", *Frontiers of Mathematics in China* (2016) 1-23.

[65] J. Zhou, L. Sun, W. Wang and C. Bu, "Some spectral properties of uniform hypergraphs", *The Electronic Journal of Combinatorics* **21** (2014) 4-24.