# On risk-averse weighted $k$-club problems

Maciej RYSZ [a], Foad PAJOUH [b], Pavlo KROKHMAL [a,1], and Eduardo PASILIAO [c]

[a] *Department of Mechanical and Industrial Engineering,*
*The University of Iowa, 3131 Seamans Center, Iowa City, IA, 52242*
[b] *Department of Industrial & Systems Engineering,*
*Research and Engineering Education Facility (UF-REEF),*
*University of Florida, 1350 N. Poquito Road, Shalimar, FL 32579*
[c] *Air Force Research Lab,*
*101 West Eglin Blvd, Eglin AFB, FL 32542*

**Abstract.** In this work, we consider a risk-averse maximum weighted $k$-club problems. It is assumed that vertices of the graph have stochastic weights whose joint distribution is known. The goal is to find the $k$-club of minimum risk contained in the graph. A stochastic programming framework that is based on the formalism of coherent risk measures is used to find the corresponding subgraphs. The selected representation of risk of a subgraph ensures that the optimal solutions are maximal $k$-clubs. A combinatorial branch-and-bound solution algorithm is proposed and solution performances are compared with an equivalent mathematical programming counterpart problem for instances with $k = 2$.

**Keywords.** $k$-club, clique relaxation, risk-averse subgraph problem, stochastic weights, coherent risk measures.

## 1. Introduction

A principal class of graph theoretical problems involves the identification of embodied subgraphs corresponding to some structural property. One particular setting of fundamental importance entails finding the largest "perfectly" cohesive group within a network such that the confined members are all interconnected, i.e., the largest *clique* (*complete subgraph*). Several prominent studies founded the basis for exact combinatorial solution algorithms for the maximum clique problem [1, 2, 3]. In particular, Carraghan and Pardalos [2] introduced a recursive branch-and-bound method for efficient finding maximum cliques by exploiting the heredity property [4] of complete subgraphs. Subsequent extensions of their work enhanced the process of eliminating solution space via vertex coloring schemes for *branching* and upper-bounds estimation on the maximal achievable subgraph sizes during the algorithmic processing (e.g. [5, 6, 7]). In many practical applications, the requirement that the desired subgraph must be complete may, however, impose excessive restrictions, and warrant some structural relaxation in terms of member connectivity. As a consequence, several clique relaxation models have been proposed in graph theory literature. A comprehensive review on clique relaxation models is provided

[1] Corresponding Author, E-mail: krokhmal@engineering.uiowa.edu.

in [4]. In this work we focus on a specific model, the *k-club* [8], where subgraph members may also be indirectly connected via at most $k$ intermediary members.

A popular extension of the described above class of problems involves the imposition topologically exogenous information in the form of deterministic vertex weights, and correspondingly finding a subset of maximum weight that conforms to a defined structural property. Similar exact weight-based branch-and-bound solution techniques have been developed for determining the maximum-weight subgraphs [9, 10, 11].

Particular circumstances may further justify the imposition of uncertain exogenous information over the graph's edges that influences network flow distribution, robustness, and costs [12, 13, 14, 15, 16, 17]. However, far fewer endeavors concern decision making regarding optimal resource allocation over defined subgraph topologies when uncertainties are induced by stochastic factors associated with network vertices. In this study, we adopt this setting and extend the techniques introduced in [18] to address problems seeking subgraphs of minimum risk that represent a $k$-club. A statistical framework utilizing the distributional information of stochastic vertex weights by means of *coherent risk measures* [19, 20] is employed to define a *risk-averse maximum weighted k-club* (R-MWK) problem as finding the lowest risk $k$-club in a network. As an illustrative example, we focus on instances when $k = 2$ and utilize a mathematical programming formulation for the maximum 2-club problem introduced in [21]. A branch-and-bound method for finding maximum $k$-clubs [22] is modified to accommodate the conditions of R-MWK problems by bounding solutions in a coherent risk measure context. We compare the solution performance of the proposed algorithm relative to an equivalent mathematical programming counterpart problem for R-MWK problems when $k = 2$.

The remainder of the paper is organized as follows. In Section 2 we examine the general representation of R-MWK problems and consider their properties. Section 3 presents a mathematical programming formulation and a combinatorial branch-and-bound method for R-MWK problems with $k = 2$. Finally, Section 4 furnishes numerical studies demonstrating the computational performance of the developed branch-and-bound method on problems where risk is quantified using higher-moment coherent risk measures [23].

## 2. Risk-averse stochastic maximum $k$-club problem

Given an undirected graph $G = (V,E)$ and any subset of its vertices $S \subseteq V$, let $G[S]$ represent the subgraph of $G$ induced by $S$ such that any pair of vertices $(i,j)$ share an edge in $S$ only if $(i,j)$ is an edge in $G$. To ease notation, define $\mathscr{Q}$ as a desired property which the induced graph $G[S]$ must satisfy. The present work considers the case when $\mathscr{Q}$ represents a certain relaxation of the *completeness* property, such that a subgraph with property $\mathscr{Q}$ represents a *clique relaxation*.

Depending on the characteristic of a complete graph that is relaxed, the clique relaxations can be categorized into *density-based*, *degree-based*, and *diameter-based* relaxations. The density of a graph $G = (V,E)$ is defined as a ratio $D(G) = |E|/\binom{|V|}{2}$, where the denominator represents the number of edges in a complete graph with $|V|$ vertices. Evidently, a complete graph (clique) has a density of 1. Then, for a fixed $\gamma \in (0,1)$, graph $G$ is called a *$\gamma$-quasi-clique* [24], if its density is at least $\gamma$:

$$D(G) \geq \gamma, \quad \text{or, equivalently,} \quad |E| \geq \gamma \binom{|V|}{2}.$$

The $\gamma$-quasi-clique is, therefore, a density-based relaxation of the clique concept, and as such is different from the *k-clique*, which is one of the diameter-based clique relaxations. Namely, let $d_G(i,j)$ be the distance between nodes $i,j \in V$, measured as the number of edges in the shortest path between $i$ and $j$ in $G$. Then, the subgraph $G[S]$ induced by a subset of nodes $S \subset V$ of the graph $G$ is called a $k$-clique if

$$\max_{i,j \in S} d_G(i,j) = k.$$

Note that the definition of the $k$-clique does not require that the shortest path between $i,j \in S$ belong to $G[S]$. If one requires that the shortest path between any two vertices $i,j$ in $S$ belong to the induced subgraph $G[S]$, then the subset $S$ such that

$$\max_{i,j \in S} d_{G[S]}(i,j) = k, \tag{1}$$

is called a *k-club*. Note that a $k$-club is also a $k$-clique, while the inverse is not true in general. The shortest path connecting two vertices in a clique is 1, thus 1-clique and 1-club are cliques. For a vertex $i \in V$, its *degree* $\deg_G(i)$ is defined as the number of adjacent vertices: $\deg_G(i) = |\{j \in V : (i,j) \in E\}|$. A degree-based clique relaxation, known as *k-plex,* is defined as a subset $S$ of $V$ such that the degree of each vertex in the induced subgraph $G[S]$ is at least $|S| - k$ [25]:

$$\deg_{G[S]}(i) \geq |S| - k \quad \text{for all} \quad i \in S,$$

(observe that the degree of each vertex in a clique of size $n$ is equal to $n-1$).

The present work considers the case when $\mathscr{Q}$ represents a distance-based relaxation of the clique model in the sense of $k$-club definition (1) when $k \geq 2$. Throughout the remainder of this study we let property $\mathscr{Q}_{G[S]}$ define a $k$-club as

$$\mathscr{Q}_{G[S]} = \{S \subseteq V \mid \forall i,j \in S : d_{G[S]}(i,j) \leq k\}. \tag{2}$$

A popular instance of graph-theoretic problems arises when seeking a subgraph $S$ with the maximum additive vertex weights, $w_i > 0$, that satisfies property $\mathscr{Q}_{G[S]}$. When $\mathscr{Q}_{G[S]}$ is defined by (2) a *maximum weight k-club problem* can take the form

$$\max_{S \subseteq V} \Big\{ \sum_{i \in S} w_i : G[S] \text{ satisfies } \mathscr{Q}_{G[S]} \Big\}. \tag{3}$$

Clearly, the optimal subgraph $G[S]$ in problem (3) will be *maximal*, but not necessarily the *maximum* (of the largest order) subgraph with property $\mathscr{Q}_{G[S]}$.

In this work, we consider an extension of problem (3) that assumes stochastic vertex weights. In this case, a direct translation into a stochastic framework is not trivial due to the fact that the maximization of random weights would be ill-posed in context of stochastic programming resulting from the absence of a deterministic optimal solution. Likewise, maximization of the expected weight of the sought subgraph is not interesting

in the sense that it reduces to the deterministic version of the problem presented above. A more suitable approach, thus, involves computing the subgraph's weight via a statistical functional that utilizes the distributional information about the weights' uncertainties, rather than as a simple sum of its (random) weights. To this end, we pursue a *risk-averse* approach so as to find the subgraph of $G$ that has the *lowest risk* and satisfies the property $\mathscr{Q}$. Let $X_i$ denote random variables that represent costs of losses associated with vertices $i \in V$, such that the joint distribution of vector $\mathbf{X}_G = (X_1, \ldots, X_{|V|})$ is known. The problem of finding the *minimum-risk* subgraph in $G$ with property $\mathscr{Q}$, or the risk-averse maximum weighted $\mathscr{Q}$ problem take the form:

$$\min_{S \subseteq V} \left\{ \mathscr{R}(S; \mathbf{X}_G) : G[S] \text{ satisfies } \mathscr{Q} \right\}, \tag{4}$$

where $\mathscr{R}(S; \mathbf{X}_G)$ is the risk of the induced subgraph $G[S]$ given the distributional information $\mathbf{X}_G$.

A formal representation of risk $\mathscr{R}(S; \mathbf{X}_G)$ is invoked via the well-known concept of *risk measure* in stochastic optimization literature [26]. Namely, given a probability space $(\Omega, \mathscr{F}, \mathbb{P})$, where $\Omega$ is the set of random events, $\mathscr{F}$ is the $\sigma$-algebra, and $\mathbb{P}$ is a probability measure, a risk measure is defined as a mapping $\rho : \mathscr{X} \mapsto \mathbb{R}$, where $\mathscr{X}$ is a linear space of $\mathscr{F}$-measurable functions $X : \Omega \mapsto \mathbb{R}$. Further, assuming that risk measure $\rho$ is lower semi-continuous (l.s.c.), the risk $\mathscr{R}(S; \mathbf{X}_G)$ of subgraph of $G[S]$ with uncertain vertex weights $X_i$ can be defined as an optimal value of the following stochastic programming problem:

$$\mathscr{R}(S; \mathbf{X}_G) = \min \left\{ \rho \left( \sum_{i \in S} u_i X_i \right) : \sum_{i \in S} u_i = 1,\ u_i \geq 0,\ i \in S \right\}. \tag{5}$$

Notice that this definition of the subgraph risk function $\mathscr{R}(\cdot)$ admits risk reduction through diversification as illustrated by the following proposition:

**Proposition 1 ([18])** *Given a graph $G = (V, E)$ with stochastic weights $X_i$, $i \in V$, and a l.s.c. risk measure $\rho$, the subgraph risk function $\mathscr{R}$ defined by* (5) *satisfies*

$$\mathscr{R}(S_2; \mathbf{X}_G) \leq \mathscr{R}(S_1; \mathbf{X}_G) \quad \textit{for all} \quad S_1 \subseteq S_2. \tag{6}$$

The following observation regarding the optimal solution of the risk-averse maximum weighted $\mathscr{Q}$ problem (4) stems directly from property (6):

**Corollary 1** *There exists an optimal solution of the risk-averse maximum weighted $\mathscr{Q}$ problem* (4) *with $\mathscr{R}(S; \mathbf{X}_G)$ defined by* (5) *that is a maximal $\mathscr{Q}$-subgraph in $G$.*

Additional properties of $\mathscr{R}(S; \mathbf{X}_G)$ ensue from the assumption that risk measure $\rho$ belongs to the family of coherent measures of risk. Namely, the definition of $\rho$ is augmented with the properties of monotonicity, subadditivity, transitional invariance, and positive homogeneity (see [19]). Assuming that risk measure $\rho$ in (5) is coherent, or satisfies the first three properties and is l.s.c, then the corresponding subgraph risk function $\mathscr{R}(S; \mathbf{X}_G)$ satisfies analogous properties with respect to the stochastic weights vector $\mathbf{X}_G$,

(G1) *monotonicity*: $\mathscr{R}(S;\mathbf{X}_G) \leq \mathscr{R}(S;\mathbf{Y}_G)$ for all $\mathbf{X}_G \leq \mathbf{Y}_G$;
(G2) *positive homogeneity*: $\mathscr{R}(S;\lambda\mathbf{X}_G) = \lambda\mathscr{R}(S;\mathbf{X}_G)$ for all $\mathbf{X}_G$ and $\lambda > 0$;
(G3) *transitional invariance*: $\mathscr{R}(S;\mathbf{X}_G + a\mathbf{1}) = \mathscr{R}(S;\mathbf{X}_G) + a$ for all $a \in \mathbb{R}$;

where $\mathbf{1}$ is the vector of ones, and the vector inequality $\mathbf{X}_G \leq \mathbf{Y}_G$ is interpreted component-wise.

Observe that $\mathscr{R}(S;\mathbf{X}_G)$ violates the sub-additivity requirements with respect to the stochastic weights. However, risk reduction via diversification is guaranteed by (6), which ensures that the inclusion of additional vertices to the existing feasible solution is always beneficial. Further, under an assumption of non-negative stochastic vertex weights, $\mathbf{X}_G \geq \mathbf{0}$, the subgraph risk $\mathscr{R}(S;\mathbf{X}_G)$ can be shown to be subadditive in relative to induced subgraphs in $G$,

$$\mathscr{R}(S_1 \cup S_2;\mathbf{X}_G) \leq \mathscr{R}(S_1;\mathbf{X}_G) + \mathscr{R}(S_2;\mathbf{X}_G), \quad S_1, S_2 \subseteq V. \tag{7}$$

Clearly, it is required that $S_1$, $S_2$, and $S_1 \cup S_2$ satisfy property $\mathscr{Q}$ in conformance to the context of risk-averse maximum weighted $\mathscr{Q}$ problems.

## 3. Solution approaches for risk-averse maximum weighted 2-club problems

In this section we consider a mathematical programming formulation for the R-MWK problem when $k = 2$, and where the risk $\mathscr{R}(S)$ of induced subgraph $G[S]$ is defined by (5). Also, we propose a combinatorial branch-and-bound algorithm utilizing the solution space processing principals for finding maximum $k$-clubs introduced by Pajouh and Balasundaram [22].

### *3.1. A mathematical programming formulation*

Let binary decision variables $x_i$ indicate whether node $i \in V$ belongs to a subset $S$:

$$x_i = \begin{cases} 1, & i \in S \text{ such that } G[S] \text{ satisfies } \mathscr{Q} \\ 0, & \text{otherwise.} \end{cases}$$

When the property $\mathscr{Q}$ denotes a 2-club, one can choose the *edge formulation* of the maximum 2-club problem proposed by Balasundaram et al. [21], whereby the mathematical programming formulation of the R-MWK problem with $k = 2$ takes the form

$$\begin{aligned} \min \quad & \rho\Big(\sum_{i\in V} u_i X_i\Big) \\ \text{s. t.} \quad & \sum_{i\in V} u_i = 1, \\ & u_i \leq x_i, \quad i \in V, \\ & x_i + x_j - \sum_{l \in \mathscr{N}^{\cap}(i,j)} x_l \leq 1, \quad (i,j) \in \overline{E}, \\ & x_i \in \{0,1\}, \ u_i \geq 0, \quad i \in V, \end{aligned} \tag{8}$$

where $\overline{E}$ represents the complement edges of graph $G$, and $\mathcal{N}^{\cap}(i,j)$ denotes the vertices that are both adjacent to vertex $i$ and vertex $j$. Appropriate (nonlinear) mixed integer programming solvers can be used to solve formulation (8) with risk measures $\rho$ whose representations admits some form of mathematical programming problems. A combinatorial branch-and-bound algorithm for solving R-MWK problems is described next.

### *3.2. A combinatorial branch-and-bound algorithm*

The following branch-and-bound (BnB) algorithm for solving R-MWK problems entails efficient processing of solution space by traversing "levels" of the BnB tree until a subgraph $G[S]$ that represents a maximal 2-club of minimum risk in $G$ as measured by (5) is found. The algorithm begins at level $\ell = 0$ with a partial solution $Q := \emptyset$, incumbent solution $Q^* := \emptyset$, and an upper bound on risk $L^* := +\infty$ (risk induced by $Q^*$), where $Q$ consists of the vertices of the induced subgraph with property $\mathcal{Q}$, and $Q^*$ contains vertices corresponding to a maximal $\mathcal{Q}$-subgraph whose risk equals $L^*$ in $G$. A set of "candidate" vertices $C_\ell$ is maintained at each level $\ell$, from which a certain *branching* vertex $q$ is selected and added to the partial solution $Q$, or simply deleted from set $C_\ell$ without being added to $Q$. In order to ensure that the proper vertices are removed from $Q$ when the algorithm backtracks between levels of the BnB tree, we introduce set $F := \emptyset$ to account for the levels at which nodes were created to delete a vertex $q$ from $C_\ell$.

Due to the distance-based properties of $k$-clubs, considerations are warranted upon transferring or deleting a vertex $q$ from candidate set $C_\ell$, as the structural integrity of corresponding to the graph induced by $Q$ and the candidate set at the subsequent level $\cup C_{\ell+1}$ may be affected. Thus, the removal of $q$ from $C_\ell$ to add to $Q$, and the deletion of $q$ from $C_\ell$ without adding it to $Q$ are considered independently via the construction of two BnB tree nodes for any given current node at level $\ell$. The first node is created to include $q$ in $Q$, while the other to delete $q$ from $C_\ell$. The necessary structural properties of $Q$ and $C_{\ell+1}$ at each node are described next.

Consider a $k$-clique in graph $G$ as a subset $S$ that satisfies

$$\{S \subseteq V \mid \forall i,j \in S : d_G(i,j) \leq k\},$$

and observe that any $k$-club in $G$ also satisfies the properties of a $k$-clique, while a $k$-clique is not necessarily a $k$-club for $k \geq 2$. Further, both reduces to a complete graph in the case of $k = 1$. By this notion, an incumbent solution $Q^*$ defines a $k$-club if the following conditions are maintained for all graphs $G[Q \cup C_{\ell+1}]$:

(C1) $Q$ is a $k$-clique in $G[Q \cup C_{\ell+1}]$
(C2) $d_{G[Q \cup C_{\ell+1}]}(i,j) \leq k, \ \forall i \in Q, \ \forall j \in C_{\ell+1}$

The algorithm is then initialized with $C_0 := V$. Whenever a vertex $q$ is selected from $C_\ell$ and added to $Q$, the candidate set at level $\ell + 1$ must be accordingly constructed by removing all vertices from $C_\ell$ whose distances to vertex in $q$ are larger than $k$,

$$C_{\ell+1} := \{j \in C_\ell : d_{G[Q \cup C_\ell]}(q,j) \leq k\}.$$

In situations when the deleted vertices serve as intermediaries, their removal from $C_\ell$ may, however, impose pairwise distance violations among the vertices in $Q \cup q$ with respect to condition (C2). In other words, after removing vertex $q$ from $C_\ell$, the distance

between a pair of vertices $(i,j) \in Q$ follows $d_{G[Q \cup C_{\ell+1}]}(i,j) > k$. In such cases, the corresponding node of the BnB tree is fathomed and the algorithm backtracks to level $\ell$. If a BnB tree node is created to delete vertex $q$, the candidate set $C_{\ell+1}$ is likewise constructed by eliminating vertices that violate (C2). If the removal of vertices from the candidate sets in either of the above cases results in a violation of (C1), then the corresponding BnB node is fathomed.

The subsequent step entails evaluating the quality of the solution that can be obtained from the subgraph induced by vertices in $Q \cup C_{\ell+1}$. An exact approach of directly finding the 2-club with the lowest possible risk that is contained in $G[Q \cup C_{\ell+1}]$ would involve solving problems (8) where $x_i = 0, \quad i \in V \setminus (Q \cup C_{\ell+1})$. However, solving a mixed 0–1 problem at every node of the BnB tree is impractical, and a lower bound problem is obtained by eliminating variables $x_i$, $i \in V$, and the graph structural constraints,

$$\begin{aligned} \mathscr{R}(Q \cup C_{\ell+1}; \mathbf{X}_G) \geq \mathscr{L}(Q \cup C_{\ell+1}) := \min \quad & \rho\Big(\sum_{i \in V} u_i X_i\Big) \\ \text{s. t.} \quad & \sum_{i \in V} u_i = 1 \\ & u_i = 0, \quad i \in V \setminus (Q \cup C_{\ell+1}) \\ & u_i \geq 0, \quad i \in Q \cup C_{\ell+1}. \end{aligned} \tag{9}$$

This notion admits the assumption that $G[Q \cup C_{\ell+1}]$ is a 2-club, under which all the mentioned graph structural constraints would be satisfies and thus vanish. Therefore, by virtue of Proposition 1, the solution to (9) provides a lower bound on the risk achievable by any 2-club contained in the graph induced via the union of vertices in Q and any subset of vertices in $C_{\ell+1}$. As a result, the risk at any subsequent level $\ell'$ along the current branch of the BnB tree cannot deteriorate as the set $Q \cup C_{\ell'+1}$ is refined.

The computed values of $\mathscr{L}(Q \cup C_{\ell+n})$ determine whether the algorithm branches further or prunes/backtracks. If $\mathscr{L}(Q \cup C_{\ell+n}) \geq L^*$, then the corresponding branch of the BnB tree is fathomed due to the fact that sequential refinement can not achieve a further reduction in risk. If $C_\ell \neq \emptyset$, another branching vertex is selected and either removed from $C_\ell$ and added to $Q$, or deleted from $C_\ell$. Alternatively, if $C_\ell = \emptyset$, the algorithm backtracks to level $\ell - 1$.

In the case when $\mathscr{L}(Q \cup C_{\ell+1}) < L^*$ and $C_{\ell+1} \neq \emptyset$, the a branching vertex $q$ is selected at the next level $\ell + 1$. In the case of $\mathscr{L}(Q \cup C_{\ell+1}) < L^*$ and $C_{\ell+1} = \emptyset$, the $G[Q]$ represents a maximal 2-club in $G$ and is assigned as the new incumbent solution, $Q^* := Q$, and the global upper bound on risk is updated $L^* := \mathscr{L}(Q \cup C_{\ell+1})$. The algorithm then backtracks to level $\ell - 1$.

Empirical experimental observations suggest that branching on a vertex $q$ with the smallest value of $\rho(X_q)$ or $\mathbb{E}X_q$ can significantly enhance computational performance. To this end, the vertices in any candidate set $C_\ell$ are ordered in descending order with respect to their risks $\rho(X_i)$ or expected values $\mathbb{E}X_i$, and the last vertex in $C_\ell$ is always selected for branching.

The described branch-and-bound algorithm procedure for R-MWK problems is formalized in Algorithm 1. Notice that it is applicable to any positive integer value $k$.

**Algorithm 1** Graph-based branch-and-bound method for problem (8)

1. **Initialize:** $\ell := 0$; $C_0 := V$; $Q := \emptyset$; $Q^* := \emptyset$; $L^* := \infty$; $F := \emptyset$;
2. **While** (**not** STOP) **do**
3.  **if** $C_\ell \neq \emptyset$ **then**
4.   select a vertex $q \in C_\ell$;
5.   $C_\ell := C_\ell \setminus q$;
6.   $Q := Q \cup q$;
7.   $C_{\ell+1} := \{i \in C_\ell : d_{G[Q\cup C_\ell]}(q,i) \leq k \; \forall i \in C_\ell\}$;
8.   **if** $Q$ is a $k$-clique in $G[Q \cup C_{\ell+1}]$ **then**
9.    solve $\mathscr{L}(Q \cup C_{\ell+1})$;
10.    **if** $\mathscr{L}(Q \cup C_{\ell+1}) < L^*$ **then**
11.     **if** $C_{\ell+1} \neq \emptyset$ **then**
12.      $\ell := \ell + 1$;
13.     **else**
14.      $Q^* := Q$;
15.      $L^* := \mathscr{L}(Q \cup C_{\ell+1})$;
16.      $Q := Q \setminus q$;
17.      **if** $\ell \notin F$ **then**
18.       $Q := Q \setminus q$
19.       $C_{\ell+1} := \{j \in C_\ell : d_{G[Q\cup C_\ell]}(i,j) \leq k, \; \forall i \in Q, \}$;
20.       **if** $C_{\ell+1} \neq \emptyset$ **then**
21.        **if** $Q$ is a $k$-clique in $G[Q \cup C_{\ell+1}]$ **then**
22.         $F := F \cup \ell$;
23.         go to step 9;
24.        **else**
25.         go to step 3;
26.      **else**
27.       $F := F \setminus \ell$;
28.    **else**
29.     **if** $\ell \notin F$ **then**
30.      $Q := Q \setminus q$;
31.     **else**
32.      $F := F \setminus \ell$;
33.   **else**
34.    $Q := Q \setminus q$;
35.    $C_{\ell+1} := \{j \in C_\ell : d_{G[Q\cup C_\ell]}(i,j) \leq k, \; \forall i \in Q, \}$;
36.    **if** $Q$ is a $k$-clique in $G[Q \cup C_{\ell+1}]$ **then**
37.     $F := F \cup l$;
38.     go to step 9;
39.    **else**
40.     go to step 3;
41.  **else**
42.   $\ell := \ell - 1$;
43.   **if** $\ell = -1$ **then**
44.    STOP
45.   **if** $\ell \notin F$ **then**
46.    $Q := Q \setminus q$;
47.   **else**
48.    $F := F \setminus l$;
49. **return** $Q^*$

## 4. Case study: Risk-averse maximum weighted 2-club problem with higher moment coherent risk measures

In this section we present a computational framework for problem (8) and conduct numerical experiments demonstrating the computational performance enhancements associated with the proposed BnB algorithm. We adopt higher-moment coherent risk (HMCR) measure class that was introduced in [23] as optimal values to the following stochastic programming problem:

$$\mathrm{HMCR}_{\alpha,p}(X) = \min_{\eta \in \mathbb{R}} \; \eta + (1-\alpha)^{-1} \left\| (X-\eta)^+ \right\|_p, \quad \alpha \in (0,1), \quad p \geq 1, \tag{10}$$

where $X^+ = \max\{0, X\}$ and $\|X\|_p = \left(\mathbb{E}|X|^p\right)^{1/p}$. Mathematical programming problems that contain HMCR measures can be formulated using $p$-order cone constraints. Typically, in stochastic programming models, the set of random events $\Omega$ is assumed to be discrete, $\Omega = \{\omega_1, \ldots, \omega_N\}$, with the probabilities $\mathbb{P}(\omega_k) = \pi_k > 0$, and $\pi_1 + \cdots + \pi_N = 1$. The corresponding mathematical programming model (8) with $\rho(X) = \mathrm{HMCR}_{p,\alpha}(X)$ takes the following mixed integer $p$-order cone programming form:

$$
\begin{aligned}
\min \quad & \eta + (1-\alpha)^{-1} t \\
\text{s. t.} \quad & t \geq \|(y_1, \ldots, y_N)\|_p, \\
& \pi_k^{-1/p} y_k \geq \sum_{i \in V} u_i X_{ik} - \eta, \quad k = 1, \ldots, N, \\
& \sum_{i \in V} u_i = 1, \\
& u_i \leq x_i, \ \ i \in V, \\
& x_i + x_j - \sum_{l \in \mathscr{N}^{\cap}(i,j)} x_l \leq 1, \quad (i,j) \in \overline{E}, \\
& x_i \in \{0,1\}, \ u_i \geq 0, \ i \in V; \ \ y_k \geq 0, \ k = 1, \ldots, N,
\end{aligned}
\tag{11}
$$

where $X_{ik}$ represents the realization of the stochastic weight of vertex $i \in V$ under scenario $k \in \mathscr{N}$. Analogously, the lower bound problem (9) takes the form

$$
\begin{aligned}
\mathscr{L}(Q \cup C_{\ell+1}) = \min \quad & \eta + (1-\alpha)^{-1} t \\
\text{s. t.} \quad & t \geq \|y_1, \ldots, y_N\|_p, \\
& \pi_k^{-1/p} y_k \geq \sum_{i \in V} u_i X_{ik} - \eta, \quad k = 1, \ldots, N, \\
& \sum_{i \in V} u_i = 1, \\
& u_i \geq 0, \ \ i \in Q \cup C_{\ell+1}, \\
& u_i = 0, \ \ i \in V \setminus (Q \cup C_{\ell+1}), \\
& y_k \geq 0, \ k = 1, \ldots, N.
\end{aligned}
\tag{12}
$$

For instances when $p = 1$ or 2, problems (11) and (12) reduce to linear programming (LP) and second order cone programming (SOCP) models, respectively. However, in cases when when $p \in (1,2) \cup (2,\infty)$ the $p$-cone is not self-dual and there exist no efficient long-step self-dual interior point solution methods. Consequently, we employ the methods for representing $p$-order cones into a higher dimensional space [27] that are based on polyhedral approximations of $p$-order cones and representation of rational-order $p$-cones via second order cones.

### *4.1. Setup of the numerical experiments and results*

Numerical experiments of the risk-averse maximum weighted 2-club problem were conducted on randomly generated Erdös-Rényi graphs of orders $|V| = 25, 50, 100$ with av-

erage densities $d = 0.0125, 0.025, 0.05, 0.1, 0.15$. The specified edge probabilities were chosen due to empirical observations indicating that a graph of order $|V| \geq 50$ commonly reduces to a 2-club when the density is in the range [0.15,0.25]. The stochastic weights of graphs' vertices were generated as i.i.d. samples from the uniform $U(0,1)$ distribution. Scenario sets with $N = 100$ were generated for each combination of graph order and density. The HMCR risk measure (10) with $p = 1, 2, 3$, and $\alpha = 0.9$ was used.

The BnB algorithm has been coded in C++, and we used the CPLEX Simplex and Barrier solvers for the polyhedral approximations and SOCP reformulations of the $p$-order cone programming lower bound problem (12), respectively (see [27]). For instances when $p = 1$, the CPLEX Simplex solver was utilized to solve problem (12) directly. The computations were conducted on an Intel Xeon 3.30GHz PC with 128GB RAM, and the CPLEX 12.5 solver in Windows 7 64-bit environment was used.

The computational performance of the mathematical programming model (11) was compared with that of developed BnB algorithm. In the case of $p = 1$, problem (11) was solved with CPLEX MIP solver. The CPLEX MIP Barrier solver was used for the SOCP version in the case of $p = 2$, and using the SOCP reformulation in the case of $p = 3$.

Table 1 presents the computational times, averaged over five instances. Observe that the BnB algorithm outperforms the CPLEX MIP solver over all the listed graph configurations, and one to two orders of magnitude in performance improvements were witnesses for the majority of instances. Further, the relative differences in performance also become more pronounced with an increase in $p$. Also noteworthy is improvement in relative performance of the BnB method for problems with $p = 3$ in comparison to $p = 2$. This results from properties of the cutting-plane algorithm for solving polyhedral approximations of $p$-order cone programming problems, which becomes more effective as $p$ increases [27].

| | | $d = 0.0125$ | | $d = 0.025$ | | $d = 0.05$ | | $d = 0.1$ | | $d = 0.15$ | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $p$ | $\|V\|$ | CPLEX | BnB | CPLEX | BnB | CPLEX | BnB | CPLEX | BnB | CPLEX | BnB |
| 1 | 25 | 0.47 | 0.06 | 0.54 | 0.04 | 0.46 | 0.04 | 0.31 | 0.04 | 0.32 | 0.08 |
| | 50 | 1.32 | 0.13 | 0.74 | 0.14 | 0.79 | 0.18 | 1.29 | 0.33 | 2.47 | 1.91 |
| | 100 | 1.99 | 0.07 | 3.25 | 0.38 | 6.00 | 2.19 | 57.62 | 40.90 | — | — |
| 2 | 25 | 11.00 | 0.56 | 9.63 | 0.72 | 6.24 | 0.33 | 6.38 | 0.37 | 10.57 | 0.43 |
| | 50 | 16.20 | 0.69 | 14.89 | 0.52 | 19.01 | 0.46 | 46.19 | 1.10 | 167.51 | 4.91 |
| | 100 | 38.25 | 0.61 | 119.15 | 1.15 | 253.27 | 2.91 | 973.18 | 70.45 | — | — |
| 3 | 25 | 40.48 | 0.90 | 25.65 | 0.81 | 15.53 | 0.42 | 15.26 | 0.66 | 27.25 | 0.86 |
| | 50 | 35.89 | 1.11 | 31.80 | 1.21 | 42.39 | 1.09 | 90.74 | 1.55 | 232.49 | 5.36 |
| | 100 | 70.47 | 1.08 | 188.71 | 1.54 | 316.38 | 3.13 | 1455.73 | 62.73 | — | — |

**Table 1.** Average computation times (in seconds) obtained by solving problem (8) using the proposed BnB algorithm and CPLEX with risk measure (10) and scenarios $N = 100$. All running times are averaged over 5 instances and symbol "—" indicates that the time limit of 7200 seconds was exceeded.

## 5. Conclusions

We have considered a R-MWK problems which entail finding a $k$-club of minimum risk in a graph. HMCR risk measures were utilized for quantifying the distributional information of the stochastic factors associated with vertex weights. It was shown that the

optimal solutions to R-MWK problems are maximal $k$-clubs. A combinatorial BnB solution algorithm was developed and tested on a special case of the R-MWK problem when $k = 2$. Numerical experiments on randomly generated graphs of various configurations suggest that the proposed BnB algorithm significantly reduces solution times in comparison with the mathematical programming model solved using the CPLEX MIP solver.

## 6. Acknowledgements

This work was supported in part by the AFOSR grant FA9550-12-1-0142 and the U.S. Department of Air Force grant FA8651-12-2-0010. In addition, support by the AFRL Mathematical Modeling and Optimization Institute is gratefully acknowledged.

## References

[1] E. Balas and C. S. Yu, "Finding a maximum clique in an arbitrary graph," *SIAM Journal on Computing*, vol. 15, pp. 1054–1068, Nov. 1986.

[2] R. Carraghan and P. M. Pardalos, "An exact algorithm for the maximum clique problem," *Operations Research Letters*, vol. 9, no. 6, pp. 375 – 382, 1990.

[3] P. R. J. Östergård, "A fast algorithm for the maximum clique problem," *Discrete Applied Mathematics*, vol. 120, no. 1–3, pp. 197–207, 2002. Special Issue devoted to the 6th Twente Workshop on Graphs and Combinatorial Optimization.

[4] S. Trukhanov, C. Balasubramaniam, B. Balasundaram, and S. Butenko, "Algorithms for detecting optimal hereditary structures in graphs, with application to clique relaxations," *Computational Optimization and Applications*, vol. 56, no. 1, pp. 113–130, 2013.

[5] E. Tomita, Y. Sutani, T. Higashi, S. Takahashi, and M. Wakatsuki, "A simple and faster branch-and-bound algorithm for finding a maximum clique," in *WALCOM: Algorithms and Computation* (M. Rahman and S. Fujita, eds.), vol. 5942 of *Lecture Notes in Computer Science*, pp. 191–203, Springer Berlin Heidelberg, 2010.

[6] J. Konc and D. Janezic, "An improved branch and bound algorithm for the maximum clique problem," *proteins*, vol. 4, p. 5, 2007.

[7] R. Carmo and A. Zge, "Branch and bound algorithms for the maximum clique problem under a unified framework," *Journal of the Brazilian Computer Society*, vol. 18, no. 2, pp. 137–151, 2012.

[8] R. D. Alba, "A graph-theoretic definition of a sociometric clique," *Journal of Mathematical Sociology*, vol. 3, pp. 3–113, 1973.

[9] L. Babel, "A fast algorithm for the maximum weight clique problem," *Computing*, vol. 52, no. 1, pp. 31–38, 1994.

[10] P. R. J. Östergård, "A new algorithm for the maximum-weight clique problem," *Nordic Journal of Computing*, vol. 8, pp. 424–436, Dec. 2001.

[11] D. Kumlander, "A new exact algorithm for the maximum-weight clique problem based on a heuristic vertex-coloring and a backtrack search," in *Proceedings of The Fourth International Conference on Engineering Computational Technology*, pp. 137–138, Civil-Comp Press, 2004.

[12] G. D. Glockner and G. L. Nemhauser, "A dynamic network flow problem with uncertain arc capacities: Formulation and problem structure," *Operations Research*, vol. 48, no. 2, pp. 233–242, 2000.

[13] A. Atamtürk and M. Zhang, "Two-stage robust network flow and design under demand uncertainty," *Operations Research*, vol. 55, no. 4, pp. 662–673, 2007.

[14] A. M. Campbell and B. W. Thomas, "Probabilistic traveling salesman problem with deadlines," *Transportation Science*, vol. 42, no. 1, pp. 1–21, 2008.

[15] A. Gupta, V. Nagarajan, and R. Ravi, "Technical noteapproximation algorithms for vrp with stochastic demands," *Operations Research*, vol. 60, no. 1, pp. 123–127, 2012.

[16] Y. P. Aneja, R. Chandrasekaran, and K. P. K. Nair, "Maximizing residual flow under an arc destruction," *Networks*, vol. 38, no. 4, pp. 194–198, 2001.

[17] B. Verweij, S. Ahmed, A. Kleywegt, G. Nemhauser, and A. Shapiro, "The sample average approximation method applied to stochastic routing problems: A computational study," *Computational Optimization and Applications*, vol. 24, no. 2-3, pp. 289–333, 2003.

[18] M. Rysz, M. Mirghorbani, P. Krokhmal, and E. Pasiliao, "On risk-averse maximum weighted subgraph problems," *Submitted for publication*.

[19] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath, "Coherent measures of risk," *Mathematical Finance*, vol. 9, no. 3, pp. 203–228, 1999.

[20] F. Delbaen, "Coherent risk measures on general probability spaces," in *Advances in Finance and Stochastics* (K. Sandmann and P. Schnbucher, eds.), pp. 1–37, Springer Berlin Heidelberg, 2002.

[21] B. Balasundaram, S. Butenko, and S. Trukhanov, "Novel approaches for analyzing biological networks," *Journal of Combinatorial Optimization*, vol. 10, pp. 23–39, 2005.

[22] F. M. Pajouh and B. Balasundaram, "On inclusionwise maximal and maximum cardinality $k$-clubs in graphs," *Discrete Optimization*, vol. 9, no. 2, pp. 84 – 97, 2012.

[23] P. A. Krokhmal, "Higher moment coherent risk measures," *Quantitative Finance*, vol. 7, pp. 373–387, 2007.

[24] A. Veremyev, V. Boginski, P. Krokhmal, and D. Jeffcoat, "Dense percolation in large-scale mean-field random networks is provably "explosive"," *PLOS ONE*, 2012.

[25] B. Balasundaram, S. Butenko, and I. Hicks, "Clique relaxations in social network analysis: The maximum k-plex problem," *Operations Research*, vol. 59, pp. 133–142, 2011.

[26] P. Krokhmal, M. Zabarankin, and S. Uryasev, "Modeling and optimization of risk," *Surveys in Operations Researh and Management Science*, vol. 16, no. 2, pp. 49–66, 2011.

[27] P. A. Krokhmal and P. Soberanis, "Risk optimization with $p$-order conic constraints: A linear programming approach," *European Journal of Operational Research*, vol. 201, no. 3, pp. 653–671, 2010.