

Convex Stochastic Bounds and Stochastic Optimisation on Graphs

J. Cohen

LRI, CNRS, Université de Paris Sud, Université Paris Saclay, Orsay, France

A. Fauquette

LRI-CentraleSupélec, CNRS, Université de Paris Sud, Université Paris Saclay, Orsay, France

J.M. Fourneau

DAVID, UVSQ, Université Paris Saclay, Versailles, France

G.C. Noulkela

*LACL, Univ. Paris Est Créteil, France et
LRI, CNRS, Université de Paris Sud, Université Paris Saclay, Orsay, France*

N. Pekergin

LACL, Univ. Paris Est, Créteil, France

Abstract

This paper presents an approach to provide stochastic bounds for a large class of optimisation problems on graphs when the parameters (i.e. costs, weights or delays) for links are random variables. We consider the class of problems which are based on convex operators and whose complexity is polynomial, when the parameters are deterministic. Here, the parameters (for instance the delay of a link) are discrete random variables. Such an assumption drastically changes the complexity of the problem (typically, the problems turn out unfortunately to be NP-complete). We propose to give stochastic bounds (both upper and lower bounds) based on convex order. First, we prove how we can simplify a discrete distribution to obtain bounding distributions which are easier to deal with, leading to a tradeoff between the computation complexity and the accuracy of the bounds. Second, we design a polynomial time algorithm to compute an upper bound. The approach is illustrated by the computation of the execution time of a task graph.

Keywords: Stochastic Convex Ordering, Discrete Distributions, Optimisation on Graphs, Stochastic PERT.

1 Introduction

In most optimisation problems on graphs, it is assumed that the edge or node costs or weights are deterministic, but this is not realistic in many cases. In a communication network, links or networking elements may malfunction due to congestion, accidents, weather conditions, DDOS attacks, ... Thus, one may associate with each edge and node a probability that this element will be fully or only partially available. Typically, we assume that each element (node or edge) has several capacity levels (that we can denote as states) with associated probabilities. Our aim is to analyze the distribution of the results of the optimisation problem knowing the distributions of probabilities on the edges and the nodes. We consider a large class of optimisation problems : Max-Flow, Shortest Path, Reliability, Minimum Spanning Tree, Completion Time. These problems are built upon convex or concave operators. These problems have received a lot of attention due to their practical interest and their rich theoretical context (see for instance [5] for the complexity issue).

Many methods only consider approximations or bounds for the expectation and are based on the graph properties which are relevant for the problem. For instance, in [14], the authors study the properties of the trees and the cuts associated with them to bound the distribution of the total weights in a spanning tree where the edges' weights are random.

A completely different approach was used in [18] to bound the execution time of a task graph where the delays are independent and exponentially distributed. The authors obtain stochastic upper and lower bounds of the distributions of the completion time. The stochastic order they used is the strong stochastic bound and the comparison takes advantage of the association (i.e. positive correlation) of the random variables describing the delays experienced on a path between the source and the destination.

Unlike most traditional approaches for this problem, our approach is based on the distributions of probability on the edges or the nodes and the properties of the operators rather than the structure of the graph. As the operators are convex, we advocate that we can use stochastic convex order on the distributions of probability for the inputs to bound the distribution of the result.

The distribution of probability for the edges is a discrete distribution, for instance obtained from some measurements. We do not assume that we obtain a density from these measurements through a fitting algorithm such as [7]. We directly deal with the measurements and we obtain bounds with a tradeoff between computation complexity and accuracy. The main transformation (or actions) is the fusion of atoms (like in [8]) in the input distribution of the problems.

The technical part of the paper is as follows. In Section 2 we give a brief introduction to strong stochastic order and convex order and we define the family of optimisation problems we want to study. We show that these problems are based on the same set of operators which are increasing and convex (or concave). Section 3 is devoted to the elementary actions to design upper and lower bounds. In Section 4, we present an optimal algorithm for the upper bound and in Section 5

an algorithm for the lower bound. Finally in Section 6, we illustrate the approach with an example: the computation of the distribution of the completion time for a task graph.

2 Optimisation on Graphs and Stochastic Convex Ordering

2.1 Optimisation on graphs

Let us consider the following simple optimisation problem for a directed graph to illustrate our approach. As our approach is not based on the graph properties but on the operators used in the optimisation process, and on stochastic ordering, the approach can deal with many optimisation problems which looks distinct at first glance. However, these problems are all built with the same operators. Thus, our approach can be used to provide stochastic bounds for all of them.

To illustrate our approach, we consider a general Directed Acyclic Graph (V, E) with number of edges M . We assume that the DAG is not a serie-parallel directed graph as this family of graphs allows for a recursive computation. Thus we exclude such an algorithm based on recursion. We assume that the nodes are labelled using a topological ordering. Thus, if $(i, j) \in E$ then $i < j$.

The nodes are labelled with positive random delays $(w_i)_{i \in V}$. A directed edge from i to j means that j needs the results of i to be executed and must wait until all the preceding nodes have completed their own work.

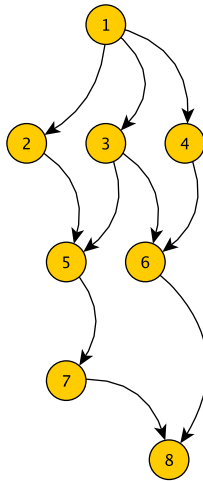


Fig. 1. Task graph.

Let b_i (resp. e_i) be the time when node i begins (resp. completes) its work. We have $e_i - b_i = w_i$ where w_i is the execution time of task i . It is well-known (see for instance [18]) that the time to complete the last node of the graph e_M can be

recursively computed by the following sequence:

$$e_i = w_i + \max_{j \in \Gamma^-(i)}(e_j)$$

where $\Gamma^-(i)$ is the set of predecessors of i and the sequence is initialized with $e_1 = b_1 + w_1$, and $b_1 = 0$. And the Completion-time of the task graph is e_M .

The operators "Max", "Min" and "+", are used to define many optimization problems on graphs. Thus, the methodology we develop here is useful to address many practical discrete problems. Let us illustrate the generality of the approach with a small list of well-known problems (see the reference book [2] for more precise statements):

- Shortest Path: the length of a path is the sum of the length of the edges and we need the path with the minimal length.
- Maximum Flow : as "Max Flow = Min Cut", we have to compute the minimum of the capacities of the cuts in the networks. A cut is a subset of edges which disconnect the source and the destination. The capacity of a cut is the sum of the capacity of the edges inside the cut. Again the problem is defined with operators "Min" and "+" on structures which only depend on the graph.
- "s-t" Reliability: the probability that there is an operating path from "s" to "t" in the network when each link may fail (in that case, the distributions have two atoms 0 and 1 to represent DOWN and UP links) is also the expectation for the maximum flow with edge capacity equal to 0 and 1.
- Minimum Spanning Tree: the cost of a spanning tree is the sum of the costs of the edges of the tree. And we take the minimum over all these costs. Again we use the "Min" and "+" operators.
- Completion Time (or Critical Path Method) makes use of the "Max" and "+" operators as seen previously.

We assume independence of the various random variables acting as inputs in the model. We denote by Ω the state space. Under the independence assumption, the probability of $(d1, \dots, dM)$ is given by:

$$Pr(d1, \dots, dM) = \prod_i Pr(X_i = di),$$

and Ω is the Cartesian product of the support of the input distributions. All these problems are polynomial when the inputs are deterministic. Unfortunately, it is not true anymore when the nodes are associated with random variables. Ball et al. have surveyed in [5] various methods to evaluate network reliability and some performability problems. They stated that calculating the distribution of the results is NP-hard for various shortest path and flow problems for networks or graphs with random costs. Note that it is still possible to solve the problem for small instances where the sizes of the supports are small (i.e. in a naive point of view, the distributions are almost deterministic). It is sufficient to use the Total Probability Theorem after conditioning on the states of all the random variables. We will

illustrate this approach on the "Completion Time" problem.

Without loss of generality we now assume that the random variables are associated with the nodes. Assume that the state of the network (or the graph) is a vector with size M called $X = X_1, \dots, X_M$ where X_i is the state of the random variable associated with node i (i.e. it is an atom as the distributions are discrete).

First let us formally define $CompletionTime(D_1, \dots, D_M)$ as the random variable equal to the completion time of the graph where the delays on node i is distributed according to D_i . $CT(d_1, \dots, d_M)$ will be the value (in R^+) of the completion time when the input random variables are equal to d_1, \dots, d_M . $CT(\cdot)$ can be computed in a linear time as the variables are now deterministic. Consider again the digraph (V, E) . Let D_i be the distribution of the delay in node i . Conditioning on the value of the random variables, we obtain:

$$Pr(CompletionTime(D_1, \dots, D_M) = T) = \sum_{(d_1, \dots, d_M) \in \Omega} Pr(d_1, \dots, d_M) 1_{CT(d_1, \dots, d_M) = T}$$

Thus, a naive approach consists in conditioning on all the random variables to obtain a problem with deterministic inputs which is solved in a linear time in the size of the graph. However, we have to deal with all the possible values of the random variables leading to a number of deterministic problems to solve, which is equal to the product of the number of atoms (i.e. the size of Ω). Assume that all the distributions have the same size (equal to N), the complexity of this naive approach is:

$$N^M B(M),$$

where $B(M)$ is the complexity in the deterministic case for a graph with size M . All the problems mentioned previously have a polynomial complexity and $CT(\cdot)$ has a linear complexity. Note that we still have to group these elements in a distribution (several conditionings may give the same value for $CT(\cdot)$). This may add an extra logarithmic term if we need to sort the data to merge all the configurations of random variables with the same value for $CT(\cdot)$.

Also note that we do not claim that such an approach is optimal. However, as the complexity of this family of problems with random variables as inputs is NP, there is no polynomial time algorithm (unless $NP = P$) for general instances of these problems.

2.2 Stochastic ordering and its application to reduce the complexity

Here, we propose to reduce the number of atoms while keeping some qualitative information on the results. This is obtained through the use of various stochastic orders. We do not establish new complexity results. We just prove that we can reduce the size of the problems when it is too large and keep some bounds according to some stochastic orders [17].

Definition 2.1 [increasing convex ordering] Let X and Y be two random variables, $X \preceq_{icx} Y$ if for all increasing convex function ϕ , $\mathbf{E}[\phi(X)] \leq \mathbf{E}[\phi(Y)]$ if the expectations exist.

Definition 2.2 [stochastic convex ordering] Let X and Y be two random variables, $X \preceq_{cx} Y$ if $\mathbf{E}[X] = \mathbf{E}[Y]$ and $X \preceq_{icx} Y$ [13,17].

Definition 2.3 [strong stochastic ordering] Let X and Y be two random variables, $X \preceq_{st} Y$ if for all increasing function ϕ , $\mathbf{E}[\phi(X)] \leq \mathbf{E}[\phi(Y)]$ if the expectations exist.

Corollary 2.4 *Thus, we have:*

- $X \preceq_{cx} X$ and $X \preceq_{st} X$.
- If $X \preceq_{st} Y$, then for all k , $\mathbf{E}[X^k] \leq \mathbf{E}[Y^k]$.
- If $X \preceq_{cx} Y$, then for all $k > 1$, $\mathbf{E}[X^k] \leq \mathbf{E}[Y^k]$. As $\mathbf{E}[X] = \mathbf{E}[Y]$, we get $\text{Var}[X] \leq \text{Var}[Y]$.

These orders differ considerably when we consider the expectations of the random variables.

Proposition 2.5 *Let X and Y be two random variables, such that $X \preceq_{st} Y$. If $\mathbf{E}[X] = \mathbf{E}[Y]$ then $X = Y$.*

Proposition 2.6 (Expectation) *Let X be a random variable with finite expectation, then $E[X] \preceq_{cx} X$ and $E[X] + X \preceq_{cx} 2X$.*

Finally we add some well-known properties which will be useful to prove our algorithms. Their proofs and more results on these stochastic orderings can be found in the literature [15,16].

Proposition 2.7 (Stop Loss) *Let X and Y be two random variables, $X \preceq_{cx} Y$ if and only if $\mathbf{E}[X] = \mathbf{E}[Y]$ and, for all d we have, $\mathbf{E}[(X - d)^+] \leq \mathbf{E}[(Y - d)^+]$.*

It is sometimes more convenient to use the following characterization.

Corollary 2.8 *Let X and Y be two random variables, $X \preceq_{cx} Y$ if and only if $\mathbf{E}[X] = \mathbf{E}[Y]$ and, for all d we have, $\mathbf{E}[(d - X)^+] \leq \mathbf{E}[(d - Y)^+]$.*

Theorem 2.9 (Theorem 3.A.44 of Shantikumar [16] page 133) *Let X and Y be two random variables with equal means ($\mathbf{E}[X] = \mathbf{E}[Y]$) and with respective cdf F_X and F_Y . $X \preceq_{cx} Y$, if and only if the number of sign changes for the function $F_Y - F_X$ is equal to 1 and the sign change sequence is $+$, $-$. The cases $F_Y - F_X = 0$ are discarded (not considered as sign changes).*

When we deal with the comparison of discrete random variables, distributions are step-functions. Both methods require to decompose the support into intervals where the expectation or the cumulative distribution function are constant. Thus, both methods of proof have roughly the same complexity. We use both in Section 3 and Section 4.

Proposition 2.10 (Mixing) *Let X , Y and Θ be three random variables such that $[X|\Theta = a] \preceq_{\psi} [Y|\Theta = a]$ for all a in the support of Θ , then $X \preceq_{\psi} Y$ with \preceq_{ψ} being \preceq_{icx} , \preceq_{st} or \preceq_{cx} .*

Thus, using the convex ordering and the increasing convex ordering instead of

the strong stochastic ordering we keep constant the expectation and we hope that we only introduce a small bias when we deal with bounds instead of the measurements. The main characterization of the optimisation problems we consider in this paper is an extension of monotony related to various stochastic orderings. Let us define first the generic Ψ –monotony.

Definition 2.11 [Ψ –Monotony] A function f is Ψ –monotone if for all X and Y random variables such that $X \preceq_{\psi} Y$, then $f(X) \preceq_{\psi} f(Y)$, with \preceq_{ψ} being \preceq_{icx} , \preceq_{st} or \preceq_{cx} .

In this paper, we will prove that the problems we consider are monotone for the strong ordering or monotone for the increasing convex ordering. These properties are now illustrated for some well-known optimisation problems on graphs.

Proposition 2.12 *The CompletionTime is monotone for the increasing convex ordering.*

Proof: Let us consider three random variables X , Y and Θ such that $X \preceq_{cx} Y$. Thus $X \preceq_{icx} Y$. Now consider an arbitrary increasing convex function f whose arguments are X and Θ (or Y and Θ) and taking value in R . For any real a , we have $f(X, a) \preceq_{icx} f(Y, a)$. Thus, for all a , $f(X, \Theta)|_{\Theta = a} \preceq_{icx} f(Y, \Theta)|_{\Theta = a}$. Due to the mixing property 2.10, we have $f(X, \Theta) \preceq_{icx} f(Y, \Theta)$. Thus one can replace each input random variables by a bounding distribution. Finally, we note that "CompletionTime" problem is defined with the "max" and "+" operators which are increasing and convex. Clearly, the assumptions on function f hold for "CompletionTime".

Proposition 2.13 *The CompletionTime is monotone for the strong stochastic ordering.*

The proof is similar.

Corollary 2.14 *Consider the CompletionTime problem. Assuming that the durations of the task are random. Replacing the random durations by their expectations introduces a systematic bias. The deterministic result is a lower bound for the increasing convex ordering for the distribution and a lower bound for the expectation.*

Proof: Let \mathbf{D}_i be the distribution of the duration of task i . According to Prop. 2.6, we have for all node index i

$$\mathbf{E}[\mathbf{D}_i] \preceq_{cx} \mathbf{D}_i.$$

As CompletionTime is monotone for the increasing convex ordering, we get:

$$CT(\mathbf{E}[\mathbf{D}_1], \mathbf{E}[\mathbf{D}_2], \dots, \mathbf{E}[\mathbf{D}_M]) \preceq_{icx} \text{CompletionTime}(\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_M).$$

Taking the Expectation, we obtain:

$$\mathbf{E}[CT(\mathbf{E}[\mathbf{D}_1], \mathbf{E}[\mathbf{D}_2], \dots, \mathbf{E}[\mathbf{D}_M])] \leq \mathbf{E}[\text{CompletionTime}(\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_M)].$$

As $CT(.)$ is the result of the deterministic problem, it is equal to its expectation.

Thus:

$$CT(\mathbf{E}[\mathbf{D}_1], \mathbf{E}[\mathbf{D}_2], \dots, \mathbf{E}[\mathbf{D}_M]) \leq \mathbf{E}[CompletionTime(\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_M)].$$

Thus, one must be aware that replacing the input distributions by their expectations introduce a systematic bias (in fact we proved that it is a lower bound). Using stochastic bounds instead of this crude approximation will help to balance between the complexity and the accuracy of the results to derive a better approach for practical problems.

Proposition 2.15 *Similarly, the Shortest-Path problem, the max-flow problem and the s-t reliability problem which are defined with the "min" and "+" operators are monotone for the strong stochastic ordering as these operators are increasing.*

Combining the Ψ -monotone property and the approach based on conditioning suggest to algorithmically simplify the distributions to get bounds on the inputs and obtain bounds of the outputs due to the monotone property. Some of us had already studied this approach for the strong stochastic ordering [4]. More precisely, for an arbitrary distribution D with size N and any positive increasing reward function r , we proved in [4] an algorithm to find the distributions $D1$ and $D2$ with size $K < N$ such that

- $D1 \preceq_{st} D \preceq_{st} D2$
- $D1$ and $D2$ are optimal bounds according to the expectation of an arbitrary increasing function r .

The optimality of $D1$ means that if we found a distribution $D3$ such that $D3 \preceq_{st} D$ and $\sum_i r(i)D1(i) \leq \sum_i r(i)D3(i) \leq \sum_i r(i)D(i)$, then $D3 = D1$ or $D3 = D$. The optimality of $D2$ is defined in a similar manner. Note that, as function r is increasing, $D1 \preceq_{st} D$ implies that $\sum_i r(i)D1(i) \leq \sum_i r(i)D(i)$. Such an approach was shown to be valuable for network performance modeling [3], operation research [4], reliability modeling [11,12].

In this paper, we investigate a similar problem for the convex order. More precisely, for an arbitrary distribution D with N positive atoms, we prove an algorithm to find $D2$ such that

- $D \preceq_{cx} D2$
- $D2$ has size $K < N$.
- $D2$ is an optimal bound according to the expectation of an arbitrary convex function $\phi(x)$ (i.e. for instance any moment).

Furthermore, we propose an algorithm to find a lower bound $D1$ with size K but we do not prove the optimality of our method.

3 Basic Operations on Atoms

The following lemma will help to deal with large distributions by allowing to extract a part of them (by conditioning) and proving the algorithms on these smaller

distributions. We begin with a very simple technical lemma.

Lemma 3.1 *Let x_1, x_2, x_3, x_4 be arbitrary positive values such that $x_1 < x_2 < x_3 < x_4$. Then we have:*

$$(x_3 - x_2)(x_4 - x_1) < (x_3 - x_1)(x_4 - x_2).$$

Proof: $x_1 < x_2 < x_3 < x_4$. Thus, $(x_2 - x_1)(x_4 - x_3) > 0$. Thus,

$$(x_3 - x_2)(x_4 - x_1) < (x_4 - x_2)(x_3 - x_1).$$

Indeed,

$$(x_3 - x_2)(x_4 - x_1) = x_3x_4 + x_1x_2 - x_4x_2 - x_1x_3$$

and

$$(x_4 - x_2)(x_3 - x_1) = x_3x_4 + x_1x_2 - x_2x_3 - x_1x_4$$

Finally, $(x_3 - x_2)(x_4 - x_1) + (x_2 - x_1)(x_4 - x_3) = (x_4 - x_2)(x_3 - x_1)$ and the result holds.

Lemma 3.2 *We consider distribution \mathbf{D} on support \mathcal{S} . We define a true partition of \mathcal{S} into \mathcal{S}_1 and \mathcal{S}_2 . We now consider X an arbitrary random variable with distribution \mathbf{D} and the boolean random variable $\Theta = 1_{X \in \mathcal{S}_1}$. We will bound the two random variables $[X|\Theta = \text{True}]$ and $[X|\Theta = \text{False}]$.*

Assume that one builds a new random variable (say Y) on \mathcal{S} as follows:

$$[X|\Theta = \text{False}] = [Y|\Theta = \text{False}],$$

and

$$[Y|\Theta = \text{True}] \preceq_{cx} [X|\Theta = \text{True}],$$

then $Y \preceq_{cx} X$.

Proof: it is a simple application of property 2.10.

Corollary 3.3 *Similarly, if Y satisfies*

$$[X|\Theta = \text{False}] = [Y|\Theta = \text{False}],$$

and,

$$[X|\Theta = \text{True}] \preceq_{cx} [Y|\Theta = \text{True}],$$

then $X \preceq_{cx} Y$. We obtain an upper stochastic bound for the convex ordering.

These last two properties are very important to simplify the proofs of the following results on the comparison of distributions. They allow to decompose the distributions into two parts: the one which is not impacted by the action, and the one which contains the atoms which are modified. Therefore in the following lemmas, we only consider the atoms which are modified. We begin with a new formulation of the basic actions needed to design a strong stochastic bound.

Lemma 3.4 We consider an arbitrary discrete distribution (say **D1**) on two atoms a and b (without loss of generality we assume that $a < b$) defined by the following positive probabilities p_a and p_b . Let us define by **D2** and **D3** the distributions with a single atom in a for **D2** and in b for **D3**, with probability equal to 1 in both cases. Then, $\mathbf{D2} \preceq_{\text{st}} \mathbf{D1} \preceq_{\text{st}} \mathbf{D3}$.

We now prove some lemmas on the basic operations of fusion of atoms to obtain lower bounds (Lemma 3.5) and upper bounds (Lemma 3.6) for the convex ordering.

Lemma 3.5 We consider an arbitrary discrete distribution (say **D1**) on two atoms a and b (without loss of generality we assume that $a < b$) defined by the following positive probabilities p_a and p_b . Let us define by **D2** the distribution with a single atom M equal to $(a p_a + b p_b)$. Then, **D2** is a lower bound for the convex stochastic ordering of **D1**: $\mathbf{D2} \preceq_{\text{cx}} \mathbf{D1}$.

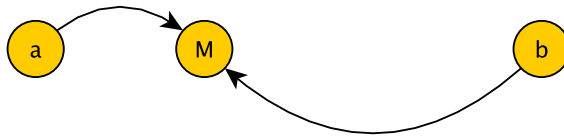


Fig. 2. Fusion of two atoms for a lower bound

Proof: it is a simple application of property 2.6.

Lemma 3.6 We consider an arbitrary discrete distribution (say **D3**) on three atoms a, b, c (without loss of generality we assume that $a < b < c$) defined by the positive probabilities p_a, p_b and p_c . Let us define by **D4** the distribution of atoms a and c the probabilities of which (denoted as q_a and q_c) are defined by

$$q_a + q_c = 1$$

and

$$aq_a + cq_c = ap_a + bp_b + cp_c.$$

Then, **D4** is an upper bound for the convex stochastic ordering of **D3**: $\mathbf{D3} \preceq_{\text{cx}} \mathbf{D4}$.

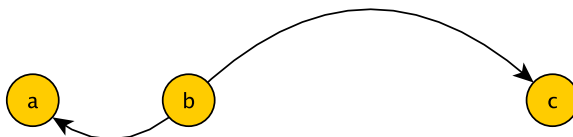


Fig. 3. Upper bounding distribution.

Proof: Two steps. First, we check that these two equations define a distribution of probability. Second, we prove that it is an upper bound for the convex ordering.

As $a \neq c$, determinant $\begin{vmatrix} 1 & 1 \\ a & c \end{vmatrix}$ is not equal to 0 and there exists a unique solution

to the system of equations. This solution is:

$$q_a = p_a + p_b \frac{c-b}{c-a} \quad q_c = p_c + p_b \frac{b-a}{c-a}$$

We now prove that **D3** \preceq_{cx} **D4** using the Stop Loss property (i.e. Property 2.7). We have to check that for all d , $\mathbf{E}[(X-d)^+] \leq \mathbf{E}[(Y-d)^+]$. We decompose the proof according to the interval which contains d :

- $d \geq c$. Thus $(X-d)^+$ and $(Y-d)^+$ are equal to 0. Thus the equality holds.
- $b \leq d < c$. $(X-d)^+$ is positive when $d = c$. And $\mathbf{E}[(X-d)^+] = (c-d)p_c$. Similarly, $\mathbf{E}[(Y-d)^+] = (c-d)q_c$. Clearly $p_c \leq q_c$, and $(c-d)$ is positive, therefore the relation holds.
- $a \leq d < b$. $(X-d)^+$ is positive when $d = c$ and $d = b$. Thus,

$$\mathbf{E}[(X-d)^+] = (c-d)p_c + (b-d)p_b$$

$(Y-d)^+$ is positive when $d = c$. Thus,

$$\mathbf{E}[(Y-d)^+] = (c-d)q_c$$

As $q_c = p_c + p_b \frac{b-a}{c-a}$, after simplifications we have to compare $(b-d)$ and $\frac{(b-a)(c-d)}{c-a}$. And we apply Lemma 3.1 to prove that $\mathbf{E}[(X-d)^+] \leq \mathbf{E}[(Y-d)^+]$ for all d between a and b .

- $d < a$. $(X-d)^+$ and $(Y-d)^+$ are positive. Thus $\mathbf{E}[(X-d)^+] = \mathbf{E}[X-d] = \mathbf{E}[X] - d$. Similarly, $\mathbf{E}[(Y-d)^+] = \mathbf{E}[Y] - d$. As $\mathbf{E}[X] = \mathbf{E}[Y]$ the equality holds.

We have also proved another method to obtain a lower bound with less atoms. We have to consider two cases to remove one atom out of three. We just illustrate one of them in the following lemma.

Lemma 3.7 *We consider an arbitrary discrete distribution (say **D1**) on three atoms a, b, c (without loss of generality we assume that $a < b < c$) defined by the positive probabilities p_a, p_b and p_c . Assume that $b > \frac{a p_a + c p_c}{p_a + p_c} = M$. Let us define by **D2** the distribution of atoms a and b the probabilities of which (denoted as q_a and q_b) are defined by*

$$q_a + q_b = 1,$$

and

$$a q_a + b q_b = a p_a + b p_b + c p_c.$$

Then, **D2** \preceq_{cx} **D1**.

The Lemma is not proved as we do not use this construction to obtain a lower bound in the following of the paper. It is just given here to show that we can obtain a lower bound according to the convex ordering based on a subset of atoms. We have a similar construction when $b < \frac{a p_a + c p_c}{p_a + p_c} = M$ as illustrated by the following example.

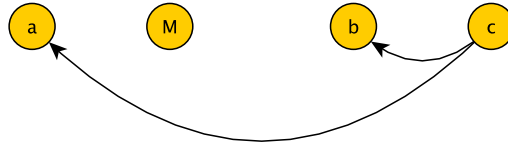


Fig. 4. Lower bound, first case.

Example 3.8 Let $\mathbf{D1}$ be a discrete distribution defined on $\mathcal{H1} = \{1, 2, 4\}$ ($a=1, b=2, c=4$) with following probabilities $[0.2, 0.4, 0.4]$. Thus $\mathbf{E}[\mathbf{D1}] = \mathbf{2.6}$, and the barycenter between 1 and 4 is $M = 3$. The lower bounding distribution $\mathbf{D2}$ is defined on $\mathcal{H2} = \{2, 4\}$ with probabilities $q_2 = 0.7$ et $q_4 = 0.3$, thus one can check that $\mathbf{E}[\mathbf{D2}] = \mathbf{2.6} = \mathbf{E}[\mathbf{D1}]$. One can check easily that there is 1 sign changes and the sequence of signs is $+, -$, it follows from Theorem 2.9 that

$$\mathbf{D2} \preceq_{\text{cx}} \mathbf{D1}$$

Note that when $b = \frac{a p_a + c p_c}{p_a + p_c}$, the construction is equivalent to the one studied in Lemma 3.5.

4 Algorithm for an optimal upper bound

Before proceeding with the algorithms, we prove a characterization of the atoms of the upper bound for the convex order.

4.1 Structure of an optimal solution for the convex ordering on a subset of atoms

Lemma 4.1 Let $\mathbf{D1}$ a distribution on N atoms and $\mathbf{D2}$ a distribution on K atoms, such that $\mathbf{D1} \preceq_{\text{cx}} \mathbf{D2}$. Let x_1, x_2, \dots, x_N be the atoms of $\mathbf{D1}$ sorted in increasing order, each value x_i is associated a probability p_i . Similarly, let y_1, y_2, \dots, y_K be the atoms of $\mathbf{D2}$ sorted in increasing order, associated to probabilities q_i . Then $y_1 = x_1$ and $y_K = x_N$.

Proof: it has two steps. First, we prove that $x_1 \geq y_1$ and then, we show that $x_1 \neq y_1$ does not hold. Let X (resp. Y) be a random variable distributed following $\mathbf{D1}$ (resp. $\mathbf{D2}$).

Assume that $x_1 < y_1$. Then there exists some d between: $x_1 < d < y_1$. We compute $\mathbf{E}[(d - X)^+]$ and $\mathbf{E}[(d - Y)^+]$.

$$\mathbf{E}[(d - X)^+] = (d - x_1)p_1 > 0$$

$$\mathbf{E}[(d - Y)^+] = 0$$

Thus there is a contradiction with $\mathbf{D1} \preceq_{\text{cx}} \mathbf{D2}$ and Corollary 2.8.

Now assume that $y_1 < x_1$. Consider d such that $y_1 < d < x_1$. $\mathbf{E}[(X - d)^+] = \mathbf{E}[X] - d$ as $d < x_1$ and x_1 is the smallest atom of $\mathbf{D1}$. And $\mathbf{E}[(Y - d)^+] < \mathbf{E}[Y] - d$,

as $d > y_1$. Taking into account that $\mathbf{E}[X] = \mathbf{E}[Y]$, we get $\mathbf{E}[(X-d)^+] > \mathbf{E}[(Y-d)^+]$, again a contradiction. The proof that $y_K = x_N$ is similar and is omitted for the sake of conciseness.

We now combine the basic actions we have studied in the previous section to obtain algorithms which reduce the complexity of the distributions while providing upper and lower bounds for the stochastic convex ordering. We first try to compare actions on distributions which share atoms to obtain an optimal action for the design of an upper bound.

Lemma 4.2 *We consider distribution \mathbf{D} with 4 atoms a, b, c, f such that $(a < b < c < f)$. Assume that p_a, p_b, p_c, p_f are positive. We apply Lemma 3.6 to atoms a, b et c to obtain an upper bound (i.e. say $\mathbf{D2}$) on the support $\{a, c, f\}$. We also apply Lemma 3.6 to atoms a, b et f to obtain $\mathbf{D3}$. This last distribution has also $\{a, c, f\}$ as a support and $\mathbf{D1} \preceq_{\text{cx}} \mathbf{D2} \preceq_{\text{cx}} \mathbf{D3}$.*

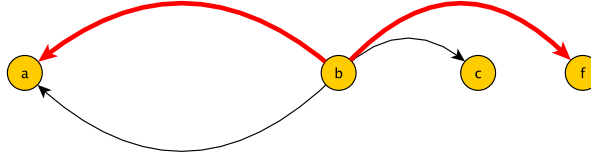


Fig. 5. Comparing upper bounds.

Proof: Both distributions have the same support and the same expectations as they are both equal to the expectations of \mathbf{D} . We already know due to Lemma 3.6 that $\mathbf{D1} \preceq_{\text{cx}} \mathbf{D2}$ and $\mathbf{D1} \preceq_{\text{cx}} \mathbf{D3}$. We now prove that $\mathbf{D2} \preceq_{\text{cx}} \mathbf{D3}$ using the Stop-Loss property. Using Lemma 3.6 we know that $\mathbf{D2}$ is:

$$q_a = p_a + p_b \frac{c-b}{c-a}, \quad q_c = p_c + p_b \frac{b-a}{c-a}, \quad q_f = p_f$$

as p_b has been distributed among a and c while f was not modified. Similarly, $\mathbf{D3}$ is:

$$r_a = p_a + p_b \frac{f-b}{f-a}, \quad r_f = p_f + p_b \frac{b-a}{f-a}, \quad r_c = p_c$$

because we distribute p_b between a and f while c was kept unchanged. We consider an arbitrary d and we compare $\mathbf{E}[(X-d)^+]$ and $\mathbf{E}[(Y-d)^+]$ where X (resp. Y) is distributed according to $\mathbf{D2}$ (resp. $\mathbf{D3}$). One must study 4 cases:

- $d \geq f$. We have, as usual, $\mathbf{E}[(X-d)^+] = 0 = \mathbf{E}[(Y-d)^+]$.
- $c \leq d < f$. $\mathbf{E}[(X-d)^+] = (f-d)p_f$ and $\mathbf{E}[(Y-d)^+] = (f-d)r_f$. As $f > d$ and $r_f > p_f$, the inequality holds.
- $a \leq d < c$. We compute the expectations:

$$\mathbf{E}[(X-d)^+] = (c-d)q_c + (f-d)q_f = (c-d)p_c + \frac{(c-d)(b-a)}{c-a}p_b + (f-d)p_f,$$

and

$$\mathbf{E}[(Y - d)^+] = (c - d)r_c + (f - d)r_f = (c - d)p_c + (f - d)p_f + \frac{(f - d)(b - a)}{f - a}p_b.$$

Lemma 3.1 implies that $\frac{c-d}{c-a} < \frac{f-d}{f-a}$. Therefore, $\mathbf{E}[(X - d)^+] < \mathbf{E}[(Y - d)^+]$

- $d < a$. As usual, $\mathbf{E}[(X - d)^+] = \mathbf{E}[X] - d = \mathbf{E}[Y] - d = \mathbf{E}[(Y - d)^+]$.

This concludes the proof.

Let us now assume that we have selected K atoms out of N and we prove the structure of the optimal distribution which is an upper bound for the convex ordering.

Lemma 4.3 *Let X be a random variable taking value on set \mathcal{S} which contains N atoms, following distribution \mathbf{D} . Consider an arbitrary subset \mathcal{S}_1 of \mathcal{S} with K atoms $x_{u_1}, x_{u_2} \dots x_{u_K}$.*

Let Y be a random variable taking value on set \mathcal{S}_1 , following distribution \mathbf{D}_1 which is an upper bounding distribution of \mathbf{D} for the \preceq_{icx} ordering (i.e. $\mathbf{D} \preceq_{\text{icx}} \mathbf{D}_1$). Moreover we assume that for all u in \mathcal{S}_1 , we have $\mathbf{E}[(X - u)^+] = \mathbf{E}[(Y - u)^+]$. Consider an arbitrary upper bound (say \mathbf{D}_2) of \mathbf{D} for this subset of K atoms. Then, we have: $\mathbf{D}_1 \preceq_{\text{cx}} \mathbf{D}_2$.

Proof: Consider a r.v. Z distributed following \mathbf{D}_2 . By construction we have:

$$\forall d \in \mathbf{R}, \quad \mathbf{E}[(X - d)^+] \leq \mathbf{E}[(Z - d)^+]$$

And by assumption we also have for all u in \mathcal{S}_1 :

$$\mathbf{E}[(X - u)^+] = \mathbf{E}[(Y - u)^+] \leq \mathbf{E}[(Z - u)^+]$$

We now have to prove that $\mathbf{E}[(Y - d)^+] \leq \mathbf{E}[(Z - d)^+]$ for all d in \mathbf{R} . We decompose the proof with respect to the values of d compared to the atoms in \mathcal{S}_1 .

- Consider two consecutive atoms x_{u_i} and $x_{u_{i+1}}$ in \mathcal{S}_1 . We know that

$$\mathbf{E}[(Y - x_{u_i})^+] \leq \mathbf{E}[(Z - x_{u_i})^+]$$

and

$$\mathbf{E}[(Y - x_{u_{i+1}})^+] \leq \mathbf{E}[(Z - x_{u_{i+1}})^+].$$

Moreover, functions $\mathbf{E}[(Z - d)^+]$ and $\mathbf{E}[(Y - d)^+]$ are linear and decreasing functions of d on $[x_{u_i}, x_{u_{i+1}}]$. Therefore, for all d in $[x_{u_i}, x_{u_{i+1}}]$,

$$\mathbf{E}[(Y - d)^+] \leq \mathbf{E}[(Z - d)^+].$$

- Now, assume that $d > u_K$. We have:

$$\mathbf{E}[(X - d)^+] = \mathbf{E}[(Y - d)^+] = \mathbf{E}[(Z - d)^+] = 0$$

- Similarly, assume that $d < u_1$.

$$\mathbf{E}[(X - d)^+] = \mathbf{E}[X] - d, \quad \mathbf{E}[(Y - d)^+] = \mathbf{E}[Y] - d, \quad \mathbf{E}[(Z - d)^+] = \mathbf{E}[Z] - d.$$

And $\mathbf{E}[X] = \mathbf{E}[Y] = \mathbf{E}[Z]$. Therefore $\mathbf{E}[(Z - d)^+] = \mathbf{E}[(Y - d)^+]$.

Finally

$$\forall d \in \mathbf{R}, \quad \mathbf{E}[(Z - d)^+] \geq \mathbf{E}[(Y - d)^+],$$

This concludes the proof of the Lemma.

We now prove that this optimal distribution already exists and we give an explicit solution.

Theorem 4.4 *Let \mathbf{D} be a distribution with support $\mathcal{S} = \{x_1, \dots, x_N\}$. The N atoms are sorted in increasing order, and probability p_i denotes the probability of atoms x_i . Consider an arbitrary subset \mathcal{S}_1 of \mathcal{S} with K atoms ($x_{u_1}, x_{u_2} \dots x_{u_K}$), sorted in increasing order, with $u_1 = 1$ and $u_K = N$. Let $\mathbf{D1}$ be a distribution on \mathcal{S}_1 associated with probabilities q_{u_i} , defined as follows for all $1 < i < K$:*

$$q_{u_i} = p_{u_i} + \sum_{\ell=u_{i-1}+1}^{u_i-1} p_\ell \frac{x_\ell - x_{u_{i-1}}}{x_{u_i} - x_{u_{i-1}}} + \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \frac{x_{u_{i+1}} - x_\ell}{x_{u_{i+1}} - x_{u_i}},$$

and for the first and last atom:

$$q_1 = p_1 + \sum_{\ell=2}^{u_2-1} p_\ell \frac{x_{u_2} - x_\ell}{x_{u_2} - x_1}, \quad q_K = p_N + \sum_{\ell=u_{K-1}+1}^{N-1} p_\ell \frac{x_\ell - x_{u_{K-1}}}{x_N - x_{u_{K-1}}}$$

Then, $\mathbf{D1}$ is a upper bounding distribution of \mathbf{D} for the \preceq_{cx} ordering, with support \mathcal{S}_1 , such that for all u in \mathcal{S}_1 , we have $\mathbf{E}[(X - u)^+] = \mathbf{E}[(Y - u)^+]$ where X (resp. Y) is a random variable distributed following \mathbf{D} (resp. $\mathbf{D1}$).

Proof: We have to prove that the distribution is well-defined. In the following, index ℓ describes the atoms of \mathbf{D} and index i the atoms of $\mathbf{D1}$. First, it is easy to notice that $q_{u_i} \geq 0$, for all i , $1 \leq i \leq K$. We now prove that $\sum_{i=1}^K q_{u_i} = 1$.

$$\begin{aligned} \sum_{i=1}^K q_{u_i} &= \sum_{i=2}^K \sum_{\ell=u_{i-1}+1}^{u_i-1} p_\ell \frac{x_\ell - x_{u_{i-1}}}{x_{u_i} - x_{u_{i-1}}} + \sum_{i=1}^{K-1} \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \frac{x_{u_{i+1}} - x_\ell}{x_{u_{i+1}} - x_{u_i}} + \sum_{i=1}^K p_{u_i} \\ &= \sum_{\ell=2, \ell \notin \mathcal{S}_1}^{N-1} p_\ell + \sum_{i=1}^K p_{u_i} = 1 \end{aligned}$$

Second, we prove that $\mathbf{E}[Y] = \mathbf{E}[X]$

$$\mathbf{E}[Y] = \sum_{i=1}^K q_{u_i} x_{u_i} = \sum_{i=1}^K p_{u_i} x_{u_i} + \sum_{\ell=2, \ell \notin \mathcal{S}_1}^{N-1} p_\ell x_\ell = \mathbf{E}[X]$$

Third, we prove by induction that $\mathbf{E}[(X - x_{u_i})^+] = \mathbf{E}[(Y - x_{u_i})^+]$, for all i , $1 \leq i \leq K$. Clearly the equality holds for u_K . Let us consider now u_{K-1} .

$$\begin{aligned} \mathbf{E}[(Y - x_{u_{K-1}})^+] &= q_{u_K}(x_{u_K} - x_{u_{K-1}}) \\ &= (x_{u_K} - x_{u_{K-1}}) \sum_{\ell=u_{K-1}+1}^{u_K-1} p_\ell \frac{x_\ell - x_{u_{K-1}}}{x_{u_K} - x_{u_{K-1}}} \\ &= \mathbf{E}[(X - x_{u_{K-1}})^+] \end{aligned}$$

Now, we will assume that $\mathbf{E}[(X - x_{u_j})^+] = \mathbf{E}[(Y - x_{u_j})^+]$ is true for all j , $i < j \leq K$. We prove that $\mathbf{E}[(X - x_{u_i})^+] = \mathbf{E}[(Y - x_{u_i})^+]$

$$\begin{aligned} \mathbf{E}[(Y - x_{u_i})^+] &= \sum_{j>i} q_{u_j}(x_{u_j} - x_{u_i}) \\ &= \sum_{j>i} q_{u_j}(x_{u_j} - x_{u_i} + x_{u_{i+1}} - x_{u_{i+1}}) \\ &= \mathbf{E}[(Y - x_{u_{i+1}})^+] + (x_{u_{i+1}} - x_{u_i}) \sum_{j>i} q_{u_j}. \end{aligned}$$

By induction $\mathbf{E}[(Y - x_{u_{i+1}})^+] = \mathbf{E}[(X - x_{u_{i+1}})^+]$ and after some algebraic manipulations, we get:

$$\begin{aligned} \mathbf{E}[(Y - x_{u_i})^+] &= \mathbf{E}[(X - x_{u_{i+1}})^+] + \sum_{u_i < \ell \leq u_{i+1}} (x_\ell - x_{u_i}) p_\ell + (x_{u_{i+1}} - x_{u_i}) \sum_{\ell \geq u_{i+1}} p_\ell \\ &= \mathbf{E}[(X - x_{u_i})^+]. \end{aligned}$$

This concludes the proof.

4.2 Building the graph

Now, we describe a polynomial time algorithm taking a distribution (say \mathbf{D}) on N atoms as input. It will return an upper bound distribution $\mathbf{D1}$ which has an included subset of size K (say x_{u_1}, \dots, x_{u_K}) as a support and which minimizes $\mathbf{E}[\phi(Y)] - \mathbf{E}[\phi(X)]$ where ϕ is an arbitrary convex function chosen by the modeler.

From Lemma 4.1, we know that the smallest and largest atom of \mathbf{D} must also be in $\mathbf{D1}$. We build the weighted oriented graph $G = (V, E)$ which contains all the paths associated to a final distribution. Its vertices are $V = \{1, 2, \dots, N\}$. The set of the arcs is $\{(i, j) \in V^2 | j > i\}$, and arc (i, j) is associated to weight w_{ij} . The weight of arc (i, j) corresponds to the increase of function ϕ when all atoms between x_i and x_j are deleted. Thus every distribution can be represent as a path passing by nodes $(x_{u_k})_{k \in [1, K]}$. Our algorithm is based on Bellman-Ford algorithm [6,9] to find the path having K edges with the minimum weight. Now, finding the upper bound distribution of K atoms with the minimal increase for function ϕ is equivalent to find the shortest path in G starting from vertex 1 to N with $K - 1$ arcs. This path

can be obtained by adapting the Bellman-Ford algorithm. Let $OPT(i, k)$ be the shortest path starting from the vertex 1 to i using **exactly** k vertices. Then the recursion formula became

$$OPT(i, k) = \min_{j < i} (OPT(j, k-1) + w_{ji})$$

$$\text{where } w_{i,j} = \phi(x_i) \sum_{\ell=i+1}^{j-1} p_\ell \frac{x_j - x_\ell}{x_j - x_i} + \phi(x_j) \sum_{\ell=i+1}^{j-1} p_\ell \frac{x_\ell - x_i}{x_j - x_i} - \sum_{\ell=i}^j \phi(x_\ell) p_\ell$$

One can easily check that the difference of expectations is the sum of the weights on the path.

$$\begin{aligned} \mathbf{E}[\phi(Y)] - \mathbf{E}[\phi(X)] &= \sum_{i=1}^K q_{u_i} \phi(x_{u_i}) - \sum_{\ell=1}^N p_\ell \phi(x_\ell) \\ &= \sum_{i=1}^K \phi(x_{u_i}) (q_{u_i} - p_{u_i}) - \sum_{i=1}^{K-1} \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \phi(x_\ell) \\ &= \sum_{i=1}^{K-1} \phi(x_{u_i}) \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \frac{x_{u_{i+1}} - x_\ell}{x_{u_{i+1}} - x_{u_i}} \\ &\quad + \sum_{i=1}^{K-1} \phi(x_{u_{i+1}}) \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \frac{x_\ell - x_{u_i}}{x_{u_{i+1}} - x_{u_i}} + \sum_{i=1}^{K-1} \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \phi(x_\ell) \\ &= \sum_{i=1}^{K-1} \left(\phi(x_{u_i}) \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \frac{x_{u_{i+1}} - x_\ell}{x_{u_{i+1}} - x_{u_i}} + \phi(x_{u_{i+1}}) \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \frac{x_\ell - x_{u_i}}{x_{u_{i+1}} - x_{u_i}} \right. \\ &\quad \left. - \sum_{\ell=u_i+1}^{u_{i+1}-1} \phi(x_\ell) p_\ell \right) \\ &= \sum_{i=1}^{K-1} w_{u_i, u_{i+1}} \end{aligned}$$

Complexity of the approach

Assume that the input distributions have size N and that the task graph has M nodes. Thus, the initial problem has a complexity of $N^M M$.

Our approach consists in two steps: first, bounding the inputs distributions and second, and then solving the problem with the new input distributions with size K . The second step (i.e. solving the completion time) has therefore a complexity equal to $K^M M$.

Each upper bounding distribution is obtained with our algorithm which needs to build the associated graph and compute the bounds after $K N^2$ operations for the bounds and N^2 operations for the construction of the associated graph (see the following remark). We must compute the bound for M distributions which are, in general, all distinct. Therefore this first step has a complexity of $N^2 M K$ operations and we obtain a bounding graph with M distributions with size K .

Remark 4.5 Computing the weights in the directed graph associated with a distribution of size N only requires $O(N^2)$ operations. Indeed,

$$\begin{aligned}
 w_{i,i+1} &= \phi(x_{u_i}) \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \frac{x_{u_{i+1}} - x_\ell}{x_{u_{i+1}} - x_{u_i}} + \phi(x_{u_{i+1}}) \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \frac{x_\ell - x_{u_i}}{x_{u_{i+1}} - x_{u_i}} - \sum_{\ell=u_i+1}^{u_{i+1}-1} \phi(x_\ell) p_\ell \\
 &= \frac{\phi(x_{u_i})}{x_{u_{i+1}} - x_{u_i}} \left(x_{u_{i+1}} \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell - \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell x_\ell \right) + \frac{\phi(x_{u_{i+1}})}{x_{u_{i+1}} - x_{u_i}} \left(\sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell x_\ell - x_{u_i} \sum_{\ell=u_i+1}^{u_{i+1}-1} p_\ell \right) \\
 &\quad - \sum_{\ell=u_i+1}^{u_{i+1}-1} \phi(x_\ell) p_\ell \\
 &= \frac{\phi(x_{u_i})}{x_{u_{i+1}} - x_{u_i}} (x_{u_{i+1}} P_{u_i, u_{i+1}} - E_{u_i, u_{i+1}}) + \frac{\phi(x_{u_{i+1}})}{x_{u_{i+1}} - x_{u_i}} (E_{u_i, u_{i+1}} - x_{u_i} P_{u_i, u_{i+1}}) - \Phi_{u_i, u_{i+1}}
 \end{aligned}$$

where

$$\Phi_{i,j} = \sum_{\ell=i+1}^{j-1} \phi(x_\ell) p_\ell, \quad E_{i,j} = \sum_{\ell=i+1}^{j-1} x_\ell p_\ell; \quad P_{i,j} = \sum_{\ell=i+1}^{j-1} p_\ell.$$

These three variables are computed in an iterative way with a quadratic complexity (i.e. $O(N^2)$). From $\Phi_{i,j}$, $E_{i,j}$, and $P_{i,j}$, we obtain $w_{i,i+1}$ within $O(1)$ operations. Therefore computing the graph requires $O(N^2)$ operations.

5 Lower bound discrete distributions

For the lower bound, we do not have a result equivalent to Lemma 4.2 to obtain an optimal lower bound and we cannot obtain an optimal bound. Thus we propose the following heuristic to compute a lower bound for the convex order based on Lemma 3.5. To simplify the presentation we present the heuristic for a function ϕ associated with the second moment. Remember that the lower bound has a smaller variance than the initial distribution. Thus at each step of the heuristic, we replace two atoms by one new atom. And by construction we keep the same expectation and the second moment decreases. Let a and b two arbitrary atoms, and Δ the difference for the second moment.

$$\Delta(a, b) = p_a a^2 + p_b b^2 - (p_a + p_b) \left(\frac{p_a a + p_b b}{p_a + p_b} \right)^2$$

After some cancellations, we get: $\Delta(a, b) = \left(\frac{p_a p_b}{p_a + p_b} \right) (a - b)^2$. Thus, at each stage, we search the couple of atoms a and b which minimizes this difference to keep the second moment and the variance as high as possible. There is no proof of optimality for the whole process as the heuristic is myopic. The complexity is N^2 to prepare matrix Δ . We then have $(N - K)$ fusions of atoms. The complexity of the fusion depends on the data-structure used to store Δ and to find the minimum and can be linear. Thus, the complexity of the heuristic is N^2 .

Remark 5.1 One can modify the value of Δ to use any convex function ϕ and use the same heuristic. Δ is still the difference between the rewards:

$$\Delta(a, b) = p_a \phi(a) + p_b \phi(b) - (p_a + p_b) \phi \left(\frac{p_a a + p_b b}{p_a + p_b} \right).$$

Algorithm 1 Heuristic for a lower bound.**Input:** input distribution **D1**, input size N , output size K **Output:** Output distribution **D2**

- 1: **D2** = **D1**.
- 2: **for** all atoms a **do**
- 3: **for** all atoms b **do**
- 4: Compute $\Delta(a, b)$ and store it in a data structure.
- 5: **end for**
- 6: **end for**
- 7: **for** $i = N$ down to $K + 1$ **do**
- 8: Search for the couple (a, b) which minimizes $\Delta(a, b)$.
- 9: Make the fusion of a and b in **D2**. Let c be the atom created by the fusion.
- 10: Update the matrix $\Delta(x, y)$ (remove the entries related to a and b , add the new entries related to c).
- 11: **end for**

6 Example

We consider the task graph decided in Fig. 1 associated with discrete distributions of the task durations to see how our algorithms perform. We have three generated instances for this graph. In the first instance, the processing time is a uniform distribution between 1 and 10. In the second (resp. third) instance, the processing time follows a Zipf's distribution [1] with exponent equal to 1.4267 (resp. 2) having $\{1, \dots, 10\}$ as support. The random variables for the tasks are independent. We consider relatively small instances to be able to solve exactly the problem and compare the solutions with the upper bounds. Note that due to the size of the graphs and the number of atoms, the number of graph analysis is 10^8 .

In Fig. 6, we have represented the distribution of the exact result for the completion time of the task graph (on the left) and the upper bounding distribution for the convex order for distributions having only 5 atoms. The input distributions for the nodes are the Zipf distribution with an exponent equal to 1.4267.

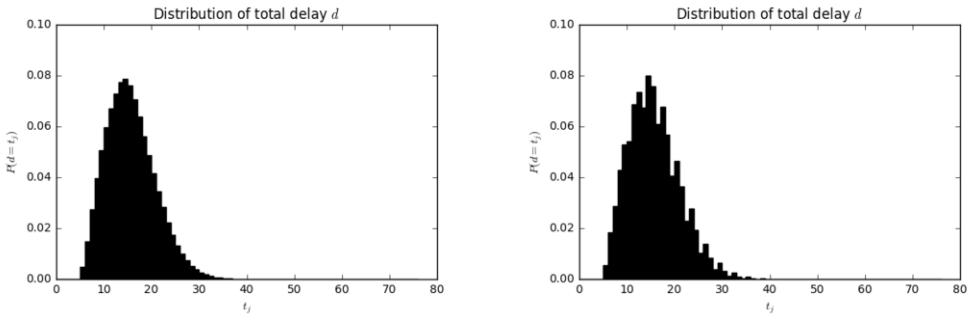


Fig. 6. Distribution of the total delay for the execution of the task graph (left: exact result, right: upper bound with 5 atoms for the input distributions).

We depict in Fig. 7, the optimal bounds on 4 atoms and 3 atoms when the initial distribution is an uniform distribution on 7 atoms. The distributions show that it is

not possible to derive the optimal distribution on 3 atoms with only one action on one atom for the optimal distribution on 4 atoms. Thus a greedy approach will not work here. In Fig. 8, we report some examples for the lower bounds. Remember

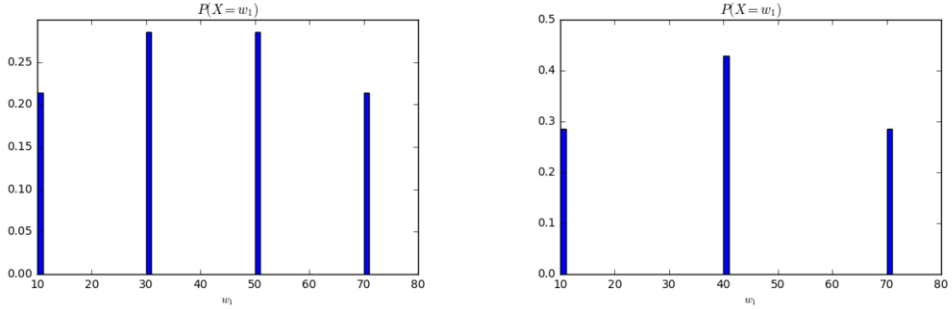


Fig. 7. Input distributions (left: optimal bound with 4 atoms, right: optimal bounds with 3 atoms). The initial distribution is an Uniform with 7 atoms.

that this bound is not optimal and we do not characterize the support as we made for optimal upper bounding distribution. We evaluate the distribution computed

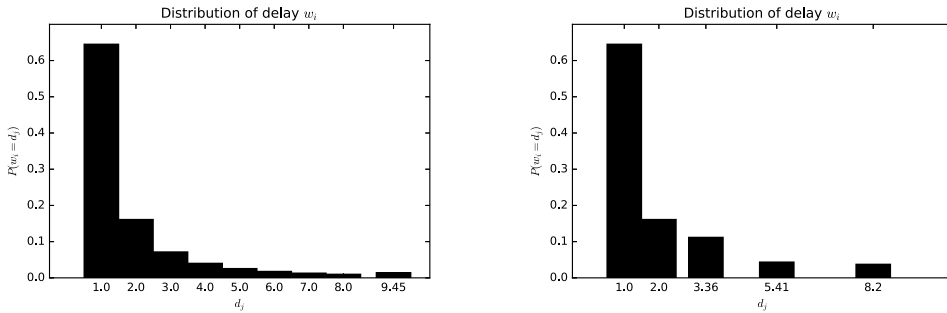


Fig. 8. Zipf with exponent 2 Input distributions (left) , lower bound with 5 atoms (right).

by the previous algorithm using the Kullback-Leibler divergence.

Definition 6.1 [Kullback-Leibler divergence] Let **D1** and **D2** two discrete probability distributions, the KullbackLeibler divergence from **D2** to **D1** is defined as:

$$D_{\text{KL}}(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}.$$

The divergence is a measure of how one probability distribution diverges from a second one. If the Kullback-Leibler divergence is equal to 0, then both distributions are equal. In Figure 9, we draw this divergence from the bound to the exact distribution, versus the number of atoms we keep in the optimal upper bounding distribution. The similarity of the two distributions remains relevant when one half of the atoms are deleted.

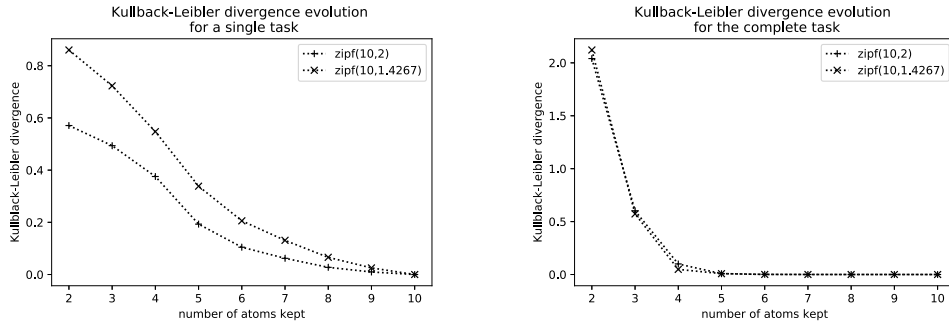


Fig. 9. Kullback-Leibler divergence from the optimal upper bound to the exact distribution. For input distribution (left). For the task graph completion time (right).

7 Conclusion

Some questions are still open. The upper bound is optimal for any convex function r . For the lower bound, we do not have proof of optimality and it is not clear that we can transform the minimization problem into a graph related algorithm such as the shortest path approach we use here like in [4]. But the theoretical results and the applications on the examples are promising and the approach proposes a bounding method for many practical optimization problems. Furthermore, replacing some measurements by their expectation in such an optimisation problem leads to systematic biases which must be understood. We have also proved such a relation for MaxFlow problem (in that case, we obtain an upper bound). We are also investigating how we can combine our approach and the various structure based approaches developed in the literature. The numerical methods will be implemented in a new version of XBorné [10], the software tool we develop to study stochastic bounds.

Acknowledgement

This work was partially supported by grant CNRS PEPS MASTODONS project ADOC 2017. The authors thank Y. Strobecki for many interesting remarks during this work.

References

- [1] L. A. Adamic and B. A. Huberman. Zipf's law and the internet. *Glottometrics*, 3(1):143–150, 2002.
- [2] A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *Data Structures and Algorithms*. Addison-Wesley, 1983.
- [3] F. Aït-Salaht, H. Castel-Taleb, J. Fourneau, and N. Pekergin. Performance analysis of a queue by combining stochastic bounds, real traffic traces and histograms. *Comput. J.*, 59(12):1817–1830, 2016.
- [4] F. At Salaht, J. Cohen, H. Castel Taleb, J. M. Fourneau, and N. Pekergin. Accuracy vs. complexity: the stochastic bound approach. In *11th International Workshop on Discrete Event Systems (WODES2012)*, number 8, 2012.
- [5] M. Ball, C. Colbourn, and J. Provan. Network reliability. *Handbooks in Operation Research and Management Science*, 7:673–762, 1995.
- [6] R. Bellman. On a routing problem. *Quarterly of applied mathematics*, 16(1):87–90, 1958.

- [7] A. Bobbio, A. Horvath, M. Scarpa, and M. Telek. Acyclic discrete phase type distributions: properties and a parameter estimation algorithm. *Performance Evaluation*, 54(1):1–32, 2003.
- [8] J. Elton and T. P. Hill. Fusions of a probability distribution. *The Annals of Probability*, 20(1):421–454, 1992.
- [9] L. R. Ford Jr. Network flow theory. Technical report, RAND CORP. SANTA MONICA CA, 1956.
- [10] J. Fourneau, Y. A. E. Mahjoub, F. Quessette, and D. Vekris. Xborne 2016: A brief introduction. In T. Czachórski, E. Gelenbe, K. Grochla, and R. Lent, editors, *Computer and Information Sciences - 31st International Symposium, ISCIS 2016, Kraków, Poland, October 27-28, 2016, Proceedings*, volume 659 of *Communications in Computer and Information Science*, pages 134–141. Springer, 2016.
- [11] J.-M. Fourneau and N. Pekergin. A numerical analysis of dynamical fault trees based on stochastic bounds. In *12th Int Conference on Quantitative Evaluation of Systems, QEST15*. IEEE, 2015.
- [12] J.-M. Fourneau and N. Pekergin. Dynamic fault trees with rejuvenation: Numerical analysis and stochastic bounds. *Electronic Notes in Theoretical Computer Science*, 327:27 – 47, 2016. The 8th International Workshop on Practical Application of Stochastic Modeling, {PASM} 2016.
- [13] A. K. Gupta and M. A. S. Aziz. Convex ordering of random variables and its applications in econometrics and actuarial science. *EUROPEAN JOURNAL OF PURE AND APPLIED MATHEMATICS*, 3(5):779–785, 2010.
- [14] K. Hutson and D. Shier. Bounding distributions for the weights of a minimum spanning tree in stochastic networks. *Operation Research*, 53(5):879–886, 2005.
- [15] A. Muller and D. Stoyan. *Comparison Methods for Stochastic Models and Risks*. Wiley, New York, NY, 2002.
- [16] M. Shaked and J. G. Shantikumar. *Stochastic Orders and their Applications*. Academic Press, San Diego, CA, 1994.
- [17] R. Szekli. *Stochastic Ordering and Dependence in Applied Probability*, volume 97. Springer Verlag, Lecture Notes in Statistics, 2012.
- [18] N. Yazici-Pekergin and J.-M. Vincent. Stochastic bounds on execution times of parallel programs. *IEEE Trans. Software Eng.*, 17(10):1005–1012, 1991.