A mixed-integer stochastic nonlinear optimization problem with joint probabilistic constraints*

T. Arnold† R. Henrion† A. Möller† S.Vigerske ‡

May 7, 2013

Abstract

We illustrate the solution of a mixed-integer stochastic nonlinear optimization problem in an application of power management. In this application, a coupled system consisting of a hydro power station and a wind farm is considered. The objective is to satisfy the local energy demand and sell any surplus energy on a spot market for a short time horizon. Generation of wind energy is assumed to be random, so that demand satisfaction is modeled by a joint probabilistic constraint taking into account the multivariate distribution. The turbine is forced to either operate between given positive limits or to be shut down. This introduces additional binary decisions. The numerical solution procedure is presented and results are illustrated.

Keywords:
Stochastic optimization, probabilistic constraints, mixed-integer nonlinear programming, power management
Mathematics Subject Classification: 90B05, 90C15, 90C10

1 Introduction

A conventional optimization problem under probabilistic constraints is given by

$$\min \{ f(x) \,|\mathbb{P}(g(x,\xi(\omega)) \geq 0) \geq p \} \,. \tag{1}$$

Here, $f : \mathbb{R}^n \to \mathbb{R}$ is some objective function, $g : \mathbb{R}^n \times \mathbb{R}^s \to \mathbb{R}^m$ is a constraint mapping, ξ is some s-dimensional random vector on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$ and $p \in [0,1]$ is some specified safety level. The meaning of the probabilistic constraint is as follows: a decision x is declared to be feasible, whenever the probability of satisfying the random inequality system

$$g_i(x, \xi(\omega)) \geq 0 \quad (i = 1, \ldots, m)$$

is at least p. Such constraints have importance in many engineering problems affected by random parameters the realization of which can be observed only after a (an optimal) decision has been taken. As basic references to theory, algorithms and applications of optimization problems with probabilistic constraints we refer to [19], [20] and [22].

The purpose of this paper is to illustrate the solution of a mixed-integer nonlinear optimization problem with joint probabilistic constraints in the context of a problem in power management involving a hydro reservoir and a wind farm. The importance of chance constrained programming in the context of water reservoir management has been recognized a long time ago (see, e.g., the basic monograph [15] or [9], [10]). We emphasize that, in contrast to most related papers, we consider the more appropriate and more difficult case of joint probabilistic constraints rather than individual ones which would allow

*This work was supported by the DFG Research Center Matheon "Mathematics for key technologies" in Berlin
†Weierstrass Institute Berlin, Germany
‡Humboldt University Berlin, Germany

for simple quantile-based reformulations of the chance constraints via linear programming (see Section 3).

One option to deal with mixed-integer problems under probabilistic constraints would be to discretize the random vector (e.g., sample average approximation as in [16]), which itself leads to a mixed-integer problem and thus does not suffer from additional binary decision variables. However, it is not evident how large the sample size for discretization has to be chosen in order to guarantee that the solution found recovers the theoretical solution (relating to an assumed continuous distribution) with a given precision. An example in [12] shows that even in dimension two a prohibitively large sample size may be required. That is why we follow in this paper the classical approach of treating probabilistic constraints under continuous distributions in the framework of nonlinear (possibly convex) optimization as pioneered by Prékopa. Corresponding models for the control of water reservoirs are found, for instance, in the early papers [17] and [18].

Progress in the efficient computation of multivariate distribution functions (e.g., [11]) offers the perspective of solving similar problems for dimensions of the random vector which are of interest in real life applications such as power management (e.g., [3]). A key issue here is the possibility to analytically reduce gradients of probability functions to function values themselves as it was demonstrated for different models (separated random vector under possibly singular linear transformation or bilinear model) under Gaussian distribution ([13],[4]). This approach has the potential to be extended to alternative distributions (e.g., multivariate log-normal or t-distribution) as well as to nonlinear models. So far, however, the focus in this context was directed on purely continuous problems. In this paper we add the consideration of binary decisions .

2 A coupled hydro-wind power management model

We consider a power management model consisting of a hydro plant coupled with a wind farm. Electricity produced by both components serves first to meet the local power demand of some area of interest and second to sell any surplus electricity on the market. In principle, there are several sources of uncertainty present in such model: uncertain inflow to the hydro plant, uncertain market prices, uncertain demand and uncertain wind force. We will apply the model for a short time planning period (2 days) which justifies to assume a constant (known) inflow of water to the hydro plant. We will also assume that the time profiles for the market price and for the demand are known (though not restricted to be constant) for this short period. In contrast, we do not neglect the randomness of the wind force which may be imagined to be much stronger than that of the previously mentioned sources. The wind farm supported by a part of the hydro power generation is supposed to meet the local demand of electricity. The remaining part of the hydro power generation is sold at the market with the aim of maximizing profit according to the given price signal. The hydro reservoir may be used to store water and thus to better adapt the water release strategy to the time profiles of price and demand. In order to exclude production strategies which are optimal for the given time horizon but at the expense of future ones (e.g., maximum production within capacity limits), a so-called *end level constraint* is imposed for the final water level in the hydro reservoir.

The decision variables of our problem are the profiles for hydro power generation over the considered time horizon used to support demand satisfaction or to sell electricity. The objective function is profit maximization. The constraints are simple bounds on the total water release (given by operational limits of the turbine), lower and upper bounds for the filling level of the hydro reservoir and demand satisfaction. The latter is a random constraint because the demand is met by the sum of a deterministic component of hydro energy and a stochastic component of wind energy. Now, the planning decision on optimal hydro power generation has to be taken before the beginning of the considered time horizon and without knowing future realizations of the random parameter (wind force). As mentioned in the introduction, random constraints in which a decision has to be taken prior to the observation of the random variable are not well-defined in the context of an optimization problem. This motivates the formulation of a corresponding probabilistic constraint in which a decision is defined to be feasible if the underlying random constraint is satisfied under this decision at least with a certain specified probability.

For longer time horizons, dynamic (closed loop) decisions could be set up as functions of past observations of the random parameter while time is running. This would lead to so-called dynamic probabilistic constraints as presented, for instance, in [6]. Such constraints are, however, very difficult to deal with numerically. In our application, the sale of energy at a spot market is part of the decision. As this is usually realized by a day-ahead bidding, decisions can not react on observations of the random parameter during the short time horizon we are considering. Therefore, we will assume a static (open loop) strategy for our decisions.

At this point one may wonder about the use of probabilistic (and not guaranteed) demand satisfaction. Indeed, in power management the customer may apply for so-called *interruptible tariffs* which allow him to pay a much lower price if he is willing to accept a well-defined (small) portion of non-delivered energy at certain unannounced periods of time. For a treatment of such models in the context of stochastic optimization (but different from the one considered in our paper), we refer to [7].

Apart from the constraints discussed before, we impose an additional so-called *end level constraint* for the hydro reservoir. Without such constraint, optimization - in our case: profit maximization - over the given time period could be carried out at the expense of future time periods. A trivial solution of profit maximization would be to release as much water from the reservoir as technically possible. Then, however, the reservoir might run empty and thus result in initial conditions for future time intervals which are worse than the ones we were starting with. Therefore, a minimum end level is required for the reservoir. The choice of this end level is up to the decision maker, it could be defined as some average level or as the initial level or any other level justified by anticipation of future events (increasing prices etc.).

A further characteristic of the model we want to consider is the incorporation of binary decision variables. These are necessary because turbines cannot be operated at an arbitrarily small level: either they are in off state or they have to work at some positive minimum level. Such on/off constraints are easily modeled by binary variables.

Discretizing the time horizon into T intervals, the resulting optimization problem reads as follows:

$$\max \sum_{t=1}^{T} \pi_t y_t \tag{2}$$

subject to

$$\mathbb{P}(x_t + \xi_t \geq d_t \quad \forall t = 1, \dots, T) \geq p \tag{3}$$

$$z_t \underline{v} \leq x_t + y_t \leq z_t \bar{v} \qquad \forall t = 1, \dots, T \tag{4}$$

$$x_t, y_t \geq 0 \qquad \forall t = 1, \dots, T \tag{5}$$

$$z_t \in \{0, 1\} \qquad \forall t = 1, \dots, T \tag{6}$$

$$\underline{l} \leq l_0 + tw - \frac{1}{\varkappa} \sum_{\tau=1}^{t} (x_\tau + y_\tau) \leq \bar{l} \quad \forall t = 1, \dots, T \tag{7}$$

$$l_0 + Tw - \frac{1}{\varkappa} \sum_{\tau=1}^{T} (x_\tau + y_\tau) \geq l^* \tag{8}$$

Here, y_t is the amount of hydro energy produced in time interval t and sold at the market. With π_t referring to the time dependent price signal, the profit to be maximized over the given time horizon equals the objective function (2).

Next, x_t is the amount of hydro energy produced in time interval t and used to satisfy the local energy demand d_t in the same interval. In addition to hydro energy, demand satisfaction is supported by a random amount ξ_t of energy produced by the wind farm in time interval t. Hence, demand satisfaction can be described by the random inequality system

$$x_t + \xi_t \geq d_t \quad \forall t = 1, \dots, T. \tag{9}$$

As discussed above, we decide on the complete profiles $(x_1, \dots, x_T)$ and $(y_1, \dots, y_T)$ at the beginning of the time horizon when the random values ξ_t have not been observed yet. This makes the random inequality system (9) meaningless in the context of our optimization problem and thus leads us to set up the probabilistic constraint (3). Here, it is required that, given the entire strategy $(x_1, \dots, x_T)$, the probability of satisfying the demand over the whole future time horizon is at least some specified level $p \in (0, 1)$.

The constraints (4) take care of a minimum operation level for the turbine. Indeed, given the binary variables z_t in (6), there exist exactly two possibilities: either $z_t = 0$ in which case (4) along with the nonnegativity constraints (5) yields that $x_t = y_t = 0$, i.e., no water is released at all; or $z_t = 1$ in which case (4) enforces the total amount $x_t + y_t$ of released water to stay between the lower and upper operation limits $\underline{v}$ and $\bar{v}$ of the turbine.

Next, (7) represents the level constraints for the hydro reservoir: here, l_0 is the initial water level at the beginning of the horizon, $\underline{l}$ and $\bar{l}$ are the lower and upper water levels in the reservoir to be respected at any time, w denotes the constant amount of water inflow to the reservoir in each time interval t and $\varkappa$ represents a conversion factor between released water and turbined energy: 1 unit of water released corresponds to $\varkappa$ units of hydro power generated. Consequently, the term between inequality signs in (7) represents exactly the filling level of the reservoir at time interval t.

Finally, taking into account that the filling level of the reservoir at time interval T equals the left-hand side of (8), we recognize this last constraint as an end level constraint imposing a minimum end level l^* for the reservoir.

3 Simplifying approaches

If in our optimization problem (2)-(8) the probabilistic constraint (3) was not present then we would deal with a conventional mixed-integer linear program the numerical solution of which could be easily determined by standard methods even in comparatively large dimension. The challenging part of the problem is the probabilistic constraint (3) which is not only nonlinear but even lacks an explicit formula. Before discussing its numerical treatment, we briefly digress with the presentation of two simplified approaches avoiding these difficulties.

The first approach consists in simply replacing the random vector ξ in the inequality system (9) by its expectation $\bar{\xi}$. In this way one obtains a deterministic inequality system

$$x_t + \bar{\xi}_t \geq d_t \quad \forall t = 1, \ldots, T \tag{10}$$

which upon replacing (3) perfectly fits to the linearity of the remaining constraints. However, solving the corresponding mixed-integer linear program yields solutions that are not at all robust as will be demonstrated in Section 5.

The second approach consists in formulating so-called individual probabilistic constraints which differ from (3) by extracting the '$\forall t$'- quantifier from the probability term:

$$\mathbb{P}(x_t + \xi_t \geq d_t) \geq p \quad \forall t = 1, \ldots, T. \tag{11}$$

At first glance, (11) might look more difficult than (3) because now, instead of one single probabilistic constraint, one deals with a system of T probabilistic constraints. The interpretation of (11) is significantly different from that of (3): it is required here that, for each time interval t individually, the probability of demand satisfaction is at least p. In contrast, in (3) one insists on the fact that the probability of demand satisfaction over the whole time horizon is at least p. The latter is clearly a much stronger requirement. That is why (11) is also referred to as individual probabilistic constraints whereas (3) is called a joint probabilistic constraint. Thanks to the ξ_t being one-dimensional random variables, one may invert their distribution function (which is no longer possible in the multivariate case) in order to establish the equivalence

$$\mathbb{P}(x_t + \xi_t \geq d_t) \geq p \Longleftrightarrow x_t \geq d_t + q_t^p \quad \forall t = 1, \ldots, T, \tag{12}$$

where for $t = 1, \ldots, T$

$$q_t^p := \inf \{\tau | \mathbb{P}(-\xi_t \leq \tau) \geq p\}$$

denote the p-quantiles of the one-dimensional random variables $-\xi_t$. The latter are easily determined numerically or tabulated for most prominent one-dimensional distributions. Clearly, the right-hand side of (12) is a system of linear inequalities again which is very similar to but more stringent than the expectation constraints (10). Hence, the same standard mixed-integer linear program (with partially different data) can be solved as in the case of expectation constraints. However, while guaranteeing demand satisfaction at the chosen probability level p at each time interval individually, the corresponding solutions may lead with high probability to demand violations at some times in the entire horizon. Again, this will be demonstrated in Section 5.

4 Numerical Solution

4.1 Dealing with the probabilistic constraint

Problem (2)-(8) is a mixed-integer stochastic nonlinear optimization problem. Without the binary constraint (6) one would deal with a nonlinear optimization problem where the only nonlinearity arises from the probabilistic constraint $\alpha(x) \geq p$ where

$$\alpha(x) := \mathbb{P}(x_t + \xi_t \geq d_t \quad \forall t = 1, \dots, T). \tag{13}$$

Thanks to the convexity theory of probabilistic constraints developed by Prékopa [19, Theorem 2.1] it is well-known that one may rewrite the original probabilistic constraint $\alpha(x) \geq p$ in the equivalent form $\varphi(x) \leq 0$ with

$$\varphi(x) := \log p - \log \alpha(x) \tag{14}$$

such that φ is a convex function whenever the random vector $\xi := (\xi_1, \dots, \xi_T)$ obeys a so-called log-concave distribution. The latter is true for many prominent multivariate distributions including the multivariate normal distribution [19, Chapter 4]. Hence, problem (2)-(8) without (6) is a nonlinear convex optimization problem which in principle can be solved by any favorite method of this area. One has to take into account, however, that the function α and, hence, the convex function φ is not given by an explicit formula because the probability involved is defined by improper multivariate integrals.

On the other hand, there exist efficient codes to approximate distribution functions, for instance, of the multivariate normal, t- or Gamma distributions sufficiently well even in interesting dimension ([11],[8],[24],[23]). Observe, that the probabilistic function defined in (13) can be written in terms of the distribution function of ξ which is defined as

$$F_\xi(z) := \mathbb{P}(\xi_t \leq z_t \quad \forall t = 1, \dots, T). \tag{15}$$

Indeed, one gets that

$$\alpha(x) = F_{-\xi}(x - d). \tag{16}$$

Hence, if one is able to approximate the distribution function of $-\xi$, then one gets an approximation for α and thus for the convex function φ.

Usually, function values alone do not provide sufficient information to apply nonlinear optimization methods. One also has to be able to calculate their gradients. According to the previous remarks, the computation of the gradient of φ can be reduced to that of the gradient $\nabla F_{-\xi}$ of the distribution function. However, given that there is no explicit formula for function values $F_{-\xi}$, much less this is true for the gradients. Approximating $\nabla F_{-\xi}$ by finite differences isn't a good idea because the inaccuracy of function values $F_{-\xi}$ will lead to highly unreliable estimations of partial derivatives when driving the step size of the finite differences towards zero. Fortunately, for the case of the multivariate normal distribution, there exists an analytical relation between function values and gradients of the distribution function [19, p. 204]. This means that no additional inaccuracy - beyond the one already present in function values - is introduced when it comes to calculate gradients.

Postponing the discussion of the inaccuracy aspect to Section 4.3, function values and gradients of φ may be used in order to set up, for instance, a supporting hyperplane method as introduced by Veinott for convex optimization problems. This approach, which is classical in probabilistic programming (see, e.g., [19]) may not be the most efficient one but it fits well into the scheme of incorporating binary decisions as it will be presented in Section 4.2. To briefly present the idea of the supporting hyperplane method, we write our continuous optimization problem (2)-(8) without (6) in the following compact form:

$$\min \left\{ c^T u | u \in U, \, \varphi(u) \leq 0 \right\}. \tag{17}$$

Here u encompasses the original continuous decision variables (x, y), $\varphi(u) \leq 0$ represents the convex probabilistic constraint according to the discussion above and U represents a polyhedron defined by the linear constraints (4),(5),(7),(8). Then, the supporting hyperplane method is defined as follows:

1. Find a point $\bar{u} \in U$ such that $\varphi(\bar{u}) < 0$ (Slater point). Determine a polyhedron $\tilde{U}$ such that

   $$\{u | \, \varphi(u) \leq 0\} \subseteq U^0 := \tilde{U} \cap U$$

   and the linear objective of (17) is bounded below on U^0. Put $k := 0$.

2. Let u^k be a solution of the linear program $\min \left\{c^T u | u \in U^k\right\}$. If $\varphi\left(u^k\right) \leq 0$, then u^k is a solution of (17) and the algorithm is terminated.

3. Bisect the function φ on the line segment $\left[u^k, \bar{u}\right]$ in order to find a point v^k such that $\varphi\left(v^k\right) = 0$ (recall that $\varphi\left(u^k\right) > 0$ and $\varphi\left(\bar{u}\right) < 0$).

4. Add a linear inequality in order to define a new polyhedron

$$U^{k+1} := U^k \cap \left\{u | \left\langle \nabla\varphi\left(v^k\right), u\right\rangle \leq \left\langle \nabla\varphi\left(v^k\right), v^k\right\rangle\right\}$$

and put $k := k+1$. Go to 2.

If this algorithm generates an infinite number of iterations (which is usually the case) then each cluster point u^* of the sequence u^k is a solution to problem (17). The same holds true for each cluster point of the sequence v^k. By convexity of φ the cuts defined in step 4 generate a decreasing sequence

$$\{u \in U, \varphi(u) \leq 0\} \subseteq \cdots \subseteq U^{k+1} \subseteq U^k \subseteq \cdots \subseteq U^0 \tag{18}$$

of polyhedra all of them containing the feasible set of (17). In particular, u^k generated in step 2 provides a lower bound of the optimal value c^* of (17), whereas v^k being feasible for (17) provides an upper bound for c^*:

$$c^T u^k \leq c^* \leq c^T v^k \tag{19}$$

Moreover, the gap between upper and lower bound converges to zero and may be used as a termination criterion for the algorithm.

4.2 Taking into account binary decision variables

In order to take binary conditions (6) into account, the supporting hyperplane method from the previous section is embedded into a branch-and-bound algorithm [14]. This algorithm creates a tree of optimization problems (2)-(8) with additional conditions on the binary variables z_t.

For the root of this tree, which corresponds to the original problem, the continuous relaxation (17) is solved by the previously outlined supporting hyperplane method, thereby constructing an equivalent linear relaxation U^k. If the solution of the relaxation U^k satisfies the binary conditions on the variables, an optimal solution for the original problem has been found. Otherwise, the algorithm selects a binary variable z_{t^*}, $t^* \in \{1, \ldots, T\}$, that takes a fractional value in the solution of U^k and creates two subproblems (branching) by adding the constraints $z_{t^*} = 0$ and $z_{t^*} = 1$, respectively. For both subproblems, a very similar algorithm is applied again. That is, the relaxation U^k, inherited from the parent problem and augmented by the subproblem specific fixations of binary variables, is resolved. If the relaxation solution violates the probabilistic constraint (3), a supporting hyperplane can be constructed as specified above, added to the relaxation, and the relaxation can be resolved. If the relaxation solution violates (6) for some $t^* \in \{1, \ldots, T\}$ two new subproblems are created by branching on z_{t^*}. If both (3) and (6) are violated, either a supporting hyperplane can be constructed or a branching can be performed. In our implementation, we do up to five rounds of the supporting hyperplane method before we branch on a binary variable.

When the solution of a subproblem relaxation satisfies both (3) and (6), a feasible solution for the original problem (2)-(8) has been found. The objective function value of this solution yields a lower bound on the optimal value of (2)-(8). Further, since the feasible space of subproblems associated to the child nodes in the branching tree yield a partition of the feasible space of the original problem, the highest optimal value of the linear relaxations among all these subproblems yields an upper bound on the optimal value of the original problem (bounding). This upper bound allows to estimate the quality of the best known feasible solution. Further, if the optimal value of a linear relaxation in a subproblem falls below the current lower bound, it is proven that this subproblem cannot contain an improving feasible solution, thus it does not need to be considered further (fathoming).

4.3 Safe cuts

An important step in the supporting hyperplane algorithm described in Section 4.1 is the bisection of the function φ on a line segment $[x, y]$ in order to find a point z such that $\varphi(z) = 0$. Here $y := \bar{u}$ denotes the Slater point satisfying $\varphi(y) < 0$ and $x := u^k$ is the current iterate with $\varphi(x) > 0$. By virtue of (14) this may also be understood as a bisection of the function α defined in (13) on the line segment $[x, y]$ in order to find a point z such that $\alpha(z) = p$. Here, $\alpha(y) > p$ and $\alpha(x) < p$. When realizing the bisection, one has to take into account that the distribution function α can be calculated with a given precision only. Accordingly, we denote by α^c the function assigning to each argument z the calculated probability $\alpha^c(z)$. Usually, α^c is a random function (it may be obtained by Monte Carlo simulation or more sophisticated methods like randomized Quasi Monte Carlo). Often, a confidence interval for the true value can be provided:

$$\mathbb{P}\left(|\alpha(z) - \alpha^c(z)| \leq \varepsilon\right) > \gamma \tag{20}$$

(in Genz' code [11], for instance, the user may select a precision $\varepsilon > 0$ for $\gamma = 0.99$). This implies, that an ideal bisection of α^c may result in a point z such that $\alpha^c(z) = p$ whereas, for instance, the true probability amounts to $\alpha(z) = p + \varepsilon$ or to $\alpha(z) = p - \varepsilon$. In order to maintain the character of the sandwiching sequence (19) yielding lower and upper bounds for the true optimal value, one may relax the definition of a bisection point as follows: instead of insisting in the equality $\alpha(z) = p$ we are looking for a couple z_1, z_2 of points such that:

1. $\alpha(z_1) < p$ (with large probability)
2. $\alpha(z_2) > p$ (with large probability)
3. z_1, z_2 are as close as possible.

The first property guarantees that the cut generated in step 4 of the supporting hyperplane algorithm with $v^k := z_1$ still provides an outer approximation of the feasible set in (17) as stated in (18). As a consequence, the sequence u^k will continue to yield a lower bound of the optimal value (left-hand side of (19)). The second property guarantees that with $v^k := z_2$ the point v^k remains feasible in (17). Hence, the sequence v^k will continue to yield an upper bound of the optimal value (right-hand side of (19)). The third property guarantees that the gap between lower and upper bound in (19) converges to a value which is small, though not zero as in the case of precise computations. How small this value is, depends on the precision ε for the computation of α^c in (20). This precision depends on the computational effort we are willing to spend. In the following we propose a bisection algorithm which yields points z_1, z_2 such that with large probability

$$p - 5\varepsilon < \alpha(z_1) < p < \alpha(z_2) < p + 5\varepsilon. \tag{21}$$

Evidently, these points satisfy the three requirements above with closeness between z_1, z_2 controlled by the term $|\alpha(z_1) - \alpha(z_2)| < 10\varepsilon$ which is a function of the chosen precision and probability in (20). To this aim, we set up the following bisection algorithm on the line segment $[x, y]$:

1. $a^0 := x, b^0 := y, k := 0$
2. If $\alpha^c\left(\frac{a^k+b^k}{2}\right) < p - 4\varepsilon$ then $a^{k+1} := \frac{a^k+b^k}{2}, b^{k+1} := b^k$. If $\alpha^c\left(\frac{a^k+b^k}{2}\right) > p - \varepsilon$ then $a^{k+1} := a^k, b^{k+1} := \frac{a^k+b^k}{2}$.
3. If $p - 4\varepsilon \leq \alpha^c\left(\frac{a^k+b^k}{2}\right) \leq p - \varepsilon$ then stop else $k := k + 1$, go to step 2.
4. Define $z_1 := \frac{a^k+b^k}{2}$.

In the next Lemma it will be shown that this algorithm yields the desired z_1 in (21) after an explicitly determinable finite number of iterations. A completely symmetric bisection algorithm can be formulated to determine the point z_2.

Lemma 1 *Let ξ have a multivariate normal distribution according to $\xi \sim \mathcal{N}(\mu, \Sigma)$, where μ denotes the vector of expected values of ξ and Σ is the covariance matrix. Let x, y be such that $\alpha(x) < p < \alpha(y)$ for α defined in (13) (this is the situation of step 3 in the supporting hyperplane algorithm). In (20) fix a probability γ and a precision $\varepsilon \in \left(0, \frac{p-\alpha(x)}{5}\right)$. Then, the bisection algorithm introduced above terminates after at most*

$$k_0 := \left\lceil -\log_2 \left(\varepsilon \sqrt{2\pi \min_{i=1,\ldots,s} \Sigma_{ii}} \right) \|x - y\|_1 \right\rceil$$

steps with a point z_1 satisfying the relation $p - 5\varepsilon < \alpha(z_1) < p$ (left part of (21)) with probability at least $6\gamma - 5$.

Proof. The assumptions $\alpha(x) < p < \alpha(y)$ and $\varepsilon \in \left(0, \frac{p-\alpha(x)}{5}\right)$ yield by virtue of (20) that

$$\alpha^c(x) < p - 4\varepsilon < p - \varepsilon < \alpha^c(y) \tag{22}$$

with probability at least $2\gamma - 1$ (note that we used (20) twice, where each single estimate is guaranteed with probability γ, so that the probability of satisfying both estimates simultaneously is at least $2\gamma - 1$). Now, given (22), it follows from step 2 in the bisection algorithm above that for all iterates k one has

$$\alpha^c\left(a^k\right) < p - 4\varepsilon < p - \varepsilon < \alpha^c\left(b^k\right). \tag{23}$$

Moreover, evidently $\left\|a^k - b^k\right\| = 2^{-k}\left\|a^0 - b^0\right\| = 2^{-k}\|x - y\|$.

Our assumption $\xi \sim \mathcal{N}(\mu, \Sigma)$ entails that $-\xi \sim \mathcal{N}(-\mu, \Sigma)$. According to Corollary 4 to Theorem 3 proved in the Appendix, the distribution function $F_{-\xi}$ (for the definition see (15)) is globally Lipschitz continuous with respect to the 1-norm and with modulus

$$M := \frac{1}{\sqrt{2\pi \min_{i=1,\ldots,s} \Sigma_{ii}}}.$$

From (16) it follows that α is globally Lipschitz continuous too with respect to the 1-norm and with the same modulus M. It follows that

$$\left|\alpha\left(a^k\right) - \alpha\left(b^k\right)\right| \leq M\left\|a^k - b^k\right\|_1 = 2^{-k} M \|x - y\|_1 \tag{24}$$

for all iterates k of the bisection algorithm. Assume that the number of these iterates reaches the value k_0 defined in the statement of this Lemma. Then, by (24),

$$\left|\alpha\left(a^{k_0}\right) - \alpha\left(b^{k_0}\right)\right| \leq \varepsilon.$$

Now, invoking (20) again twice, we see that

$$\left|\alpha^c\left(a^{k_0}\right) - \alpha^c\left(b^{k_0}\right)\right| \leq \left|\alpha\left(a^{k_0}\right) - \alpha\left(b^{k_0}\right)\right| + 2\varepsilon \leq 3\varepsilon \tag{25}$$

with probability at least $2\gamma - 1$. This, however, contradicts (23). Therefore, the bisection algorithms stops after at most k_0 steps with probability at least $4\gamma - 3$. Here we have taken into account that the correctness of this statement relies on the correctness of (22) and of (25) both of which were given with probability of at least $2\gamma - 1$. Hence the probability that both statements are correct simultaneously is at least $4\gamma - 3$. By step 3 of the bisection algorithm, the final point z_1 obtained after at most k_0 steps satisfies the relation

$$p - 4\varepsilon \leq \alpha^c(z_1) \leq p - \varepsilon.$$

Invoking (20) a third time (for both of the two inequalities above) we infer that

$$p - 5\varepsilon \leq \alpha(z_1) \leq p \tag{26}$$

with probability at least $2\gamma - 1$. Now the statement of the Lemma follows upon taking into account that we need two partial statements to be satisfied, one of them being the termination of the algorithm after at most k_0 steps - which was ensured with probability at least $4\gamma - 3$ and the second one being (26) which is guaranteed with probability at least $2\gamma - 1$. Hence the overall probability for the statement of the Lemma is at least $6\gamma - 5$. ■

Example 2 *In the setting of Lemma 1 let* $\gamma = 0.99, \varepsilon = 10^{-4}, \mu = 0, \Sigma = I$ *(identity matrix) and* x, y *such that* $\|x - y\| = 1$. *Then,* $k_0 := \lceil -\log_2 (10^{-4}\sqrt{2\pi}) \rceil = \lceil 11.962 \rceil = 12$ *and* $6\gamma - 5 = 0.94$. *Hence, with probability of at least* 0.94, *the point* z_1 *satisfying the relation* $p - 5 \cdot 10^{-4} < \alpha(z_1) < p$ *is found in at most 12 iterations.*

4.4 Implementation

The algorithm from Section 4.2 has been implemented in the branch-cut-and-price framework SCIP[1] [1, 2]. SCIP includes a full-scale solver for mixed-integer linear programs, but can be extended to other types of problems via plugins. One of the most powerful plugin types is the *constraint handler*, which defines the semantics and the algorithms to process constraints of a certain class. A single constraint handler is responsible for all the constraints belonging to its constraint class. Each constraint handler has to implement an enforcement method. In enforcement, the handler has to decide whether a given solution, e.g., the optimum of a linear relaxation satisfies all of its constraints. If the solution violates one or more constraints, the handler may resolve the infeasibility by adding another constraint, performing a domain reduction, or a branching.

For our purposes, we extended SCIP by a plugin to handle the probabilistic constraint (3). Whenever SCIP has solved the linear relaxation of a current subproblem, it either branches on a binary variable which takes a fractional value or asks our plugin to construct a linear inequality that cuts off the current relaxation solution. If neither happens, SCIP knows that it found a new feasible solution and thus updates the lower bound on the optimal value.

The algorithm is extended by primal heuristics to find feasible solutions early in the search, cutting plane separators that cut off fractional solution from the relaxation without branching, and domain propagation routines that try to derive tighter variable bounds from current variable bounds and the constraints. For details, we refer to [1, 2].

5 Results

We consider optimization problem (2)-(8) with the following data: the time horizon equals 2 days subdivided in $T = 48$ hourly intervals; the probability level for demand satisfaction was chosen as $p = 0.9$; the profiles $\pi = (\pi_1, \ldots, \pi_T)$ and $d = (d_1, \ldots, d_T)$ for price and demand were adapted from real life data of a power spot market and an electricity provider, respectively, found on the internet; the minimum and maximum operation limits of the turbines were chosen such that $\underline{v} = 0.25\bar{v}$; the final filling level l^* was defined as the mean of lower and upper levels: $l_0 = l^* = 0.5\left(\underline{l} + \bar{l}\right)$. In contrast, the initial filling level l_0 was assumed to be slightly inferior to that average. Hence, one additional purpose of the optimization problem was to slightly increase the filling level in the reservoir at the end of the time horizon. This objective might reflect some strategic considerations by the decision maker.

The model for the discrete random process $\xi = (\xi_1, \ldots, \xi_T)$ of wind speed (scaled to wind energy produced) was assumed to be multivariate normal according to $\xi \sim \mathcal{N}(\mu, \Sigma)$, where μ denotes the vector of expected values of ξ and Σ is the covariance matrix associated with the components of ξ. As observed in [5], raw data for wind speed are not normal but can be transformed into normal by raising them to a certain power. The main purpose of this paper being an illustration of how binary decisions can be integrated into probabilistic constraints, we kept for simplicity the normality assumption in our example. A constant mean wind speed with relative standard deviation of 1/3 was assumed along with correlation coefficients $\rho^{t_1,t_2} := 0.85^{|t_1 - t_2|}$ between components ξ_{t_1} and ξ_{t_2}. In this way, dependencies between components are taken into account which is an essential issue in modeling wind speed [5, p. 2114]. Of course, assuming independent components would allow for a much simpler computation of probabilities and their gradients in the constraint (3).

For the numerical solution we applied the methodology presented in Section 4. For the sake of comparison, we provide not only the results for the case of a joint probabilistic constraint (3) but also for the two simplifying approaches (expected value constraint (10) and individual probabilistic constraints

[1] `http://scip.zib.de`

(11) with same probability level $p = 0.9$ as for the joint case) discussed in Section 3 which are easily solved as mixed-integer linear programs.

Figure 1 illustrates the optimal turbining profile (i.e., the sum $x_t + y_t$) for the three models. The plots show connected parts in which turbines operate within their positive technical limits $0 < \underline{v} \leq \bar{v}$ as well as disrupted parts due to shut down or switch on decisions implying zero energy production at certain times.

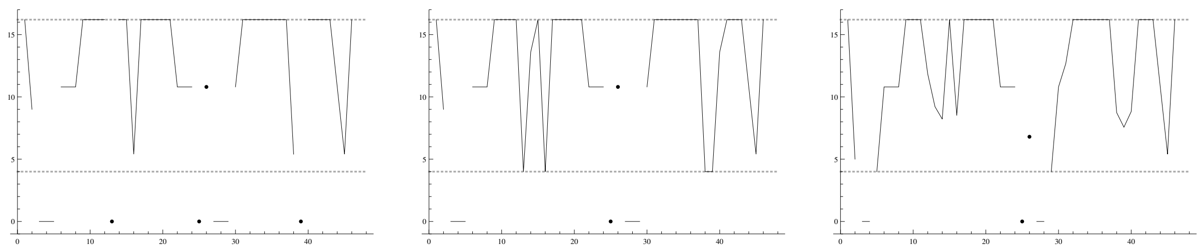

Figure 1: Optimal turbining profiles for the hydro reservoir in case of using expected values (left), individual probabilistic constraints (middle) or a joint probabilistic constraint (right). Turbines are either switched off (zero level) or work within positive operation limits (dotted lines).

Figure 2 shows the price signal π_t and the part y_t of the total hydro energy production from Figure 1 which is sold at the market. It can be seen that the expected value solution follows best the price signal, whereas the solution based on the joint probabilistic constraint deviates most in shape from the price signal. Accordingly, the optimal values (profits) of the three different models are: 25.698 (expected values), 21.143 (individual probabilistic constraints) and 10.473 (joint probabilistic constraint). However, we will see next that the better profits obtained by the simplified approaches come at the price of lacking robustness.

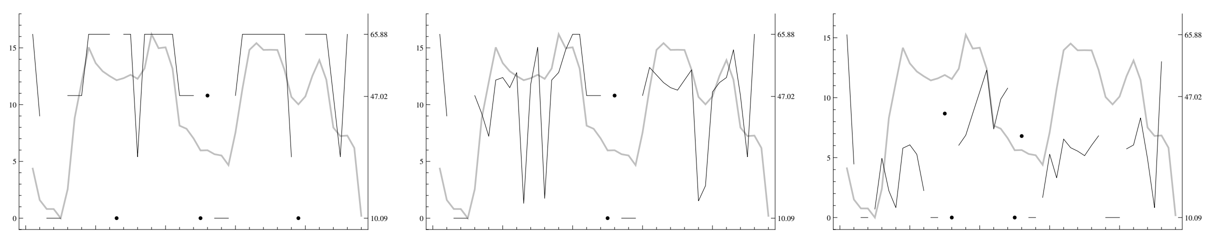

Figure 2: Part of hydro energy sold at the market in case of using expected values (left), individual probabilistic constraints (middle) or a joint probabilistic constraint (right). The (common) price signal is plotted in gray color.

In Figure 3, the satisfaction of local energy demand by the sum of wind energy ξ_t and the remaining (unsold) part of hydro energy x_t is represented. The demand profile exhibits the typical two-days shape with a low during night time. In order to visualize demand satisfaction, given the optimal solutions x_t, y_t, z_t of the corresponding problems, a number of one hundred scenarios ξ_t for wind energy were simulated according to the assumed distribution parameters $\xi \sim \mathcal{N}(\mu, \Sigma)$. We emphasize that these scenarios were not used to solve the optimization problems but just serve the purpose of an à posteriori check of the previously calculated solutions. Each figure shows the plots of supplied energy $\xi_t + x_t$ for the different scenarios ξ_t. The generated wind energy scenarios ξ_t are the same for all three models but, of course, the visualized scenarios of supplied energy $\xi_t + x_t$ differ by their hydro energy component x_t. It can be seen that the expected value solution frequently violates demand satisfaction. Indeed, it turns out that only nine out of one hundred scenarios satisfy the demand through the whole time horizon. This empirical estimate corresponds very well with the theoretical probability of 8.9% calculated according to

$$\mathbb{P}(x_t + \xi_t \geq d_t \quad \forall t = 1, \dots, T)$$

as in the constraint(3). This total lack of robustness demonstrates why the expected value solution is meaningless despite its attractive profit. Inspection of the solution based on individual probabilistic constraints reveals that for each point in time separately, only approximately 10 scenarios (or less) fall below the demand line. This is coherent with the chosen probability level $p = 0.9$ in the model (11). However, this does not tell anything about the probability of meeting the demand uniformly because different scenarios may violate the demand at different times. Indeed, an enumeration of the generated scenarios yields that only 41 out of 100 scenarios satisfy the demand through the whole time horizon (theoretical probability: 35.2%). In contrast, 93 scenarios pass through the whole time horizon without demand violation in case of the joint probabilistic constraint which fits well to the chosen probability level of $p = 0.9$ (of course, with another set of 100 generated scenarios, the empirical number of success can differ from 93 but is likely to stay around 90). Evidently, the demand is not only satisfied in a robust sense but due to the randomness of wind speed, even a considerable surplus in the energy supply is observed in general. This surplus may be thought of being sold as well or used for an additional pumped storage plant. Anyhow, the surplus is not affected by our decisions, so it is purely random and can be ignored in the optimization problem.

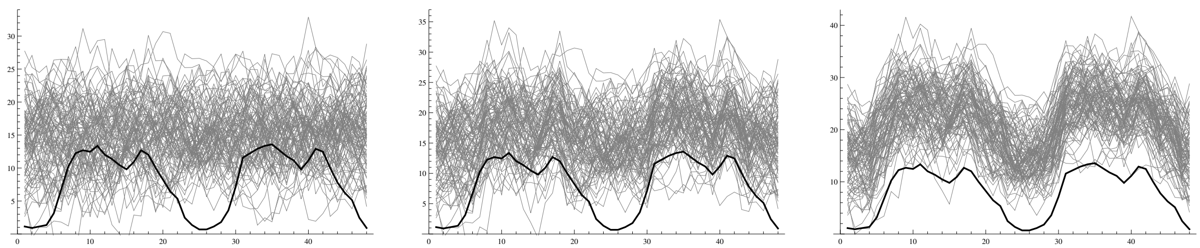

Figure 3: Simulated energy supply (wind plus unsold hydro energy) for 100 simulated wind energy scenarios in case of using expected values (left), individual probabilistic constraints (middle) or a joint probabilistic constraint (right). The (common) demand profile is plotted as a thick black curve.

Finally, Figure 4 proves that all three solutions satisfy the level constraints (7) and (8): in all cases, the filling levels stay between the critical values $\underline{l}$ and $\bar{l}$. Moreover, in all cases the required end level l^* is reached.

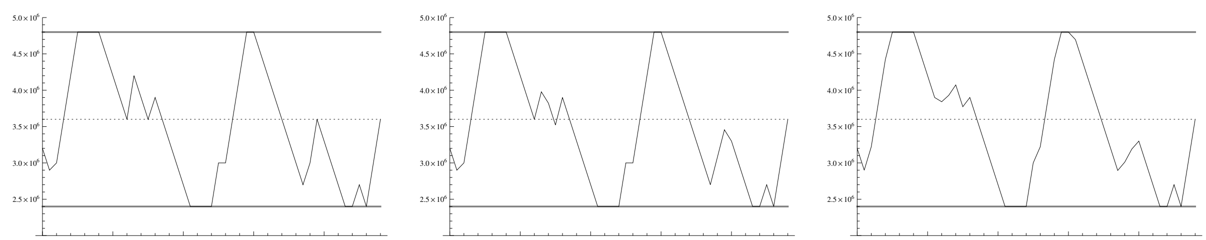

Figure 4: Water level for the hydro reservoir in case of using expected values (left), individual probabilistic constraints (middle) or a joint probabilistic constraint (right). The critical lower and upper level of the reservoir are represented by a solid line and the end level to be reached by a dotted line, respectively.

References

[1] T. Achterberg, Constraint Integer Programming, PhD Thesis, TU Berlin, 2007

[2] T. Achterberg, SCIP: Solving Constraint Integer Programs, Mathematical Programming Computation 1 (2009), 1-41.

[3] W. Van Ackooij, R. Henrion, A. Möller and R. Zorgati, On probabilistic constraints induced by rectangular sets and multivariate normal distributions, Mathematical Methods of Operations Research 71 (2010) 535-549.

[4] W. Van Ackooij, R. Henrion, A. Möller and R. Zorgati, On joint probabilistic constraints with Gaussian coefficient matrix, Operations Research Letters 39 (2011), 99-102.

[5] H. Aksoy , Z. Fuat Toprak, A. Aytek and N. Erdem Ünal, Stochastic generation of hourly mean wind speed data, Renewable Energy 29 (2004), 2111–2131.

[6] L. Andrieu, R. Henrion and W. Römisch, A model for dynamic chance constraints in hydro power reservoir management, European Journal of Operations Research 207 (2010), 579-589.

[7] R. Baldick, S. Kolos and S. Tompaidis, Interruptible Electricity Contracts from an Electricity Retailer's Point of View: Valuation and Optimal Interruption, Operations Research 54 (2006), 627-642.

[8] I. Deák, Subroutines for Computing Normal Probabilities of Sets - Computer Experiences, Annals of Operations Research 100 (2000), 103-122.

[9] Duranyildiz, I., Önöz, B. and Bayazit, M., A chance-constrained LP model for short term reservoir operation optimization, Turkish Journal of Engineering 23 (1999), 181-186.

[10] Edirisinghe, N.C.P., Patterson, E.I. and Saadouli, N., Capacity planning model for a multipurpose water reservoir with target-priority operation, Annals of Operations Research 100 (2000), 273-303.

[11] A. Genz and F. Bretz, *Computation of Multivariate Normal and t Probabilities*, Lecture Notes in Statistics, vol. 195, Springer, Heidelberg, 2009.

[12] R. Henrion, A critical note on empirical (sample average, Monte Carlo) approximation of solutions to chance constrained programs, in: D. Hömberg and F. Tröltzsch (eds.): CSMO 2011, IFIP AICT 391, Springer, Berlin, 2013, pp. 25-37.

[13] R. Henrion u. A. Möller, A gradient formula for linear chance constraints under Gaussian distribution, Mathematics of Operations Research 37 (2012) 475-488.

[14] A. H. Land and A. G. Doig, An automatic method for solving discrete programming problems, Econometrica 38 (1960), 497-520.

[15] Loucks, D.P., Stedinger, J.R. and Haith, D.A., Water Resource Systems Planning and Analysis, Prentice Hall, New Jersey, 1981.

[16] Pagnoncelli, B., Ahmed, S. and Shapiro, A., Sample Average Approximation Method for Chance Constrained Programming: Theory and Applications, Journal of Optimization Theory and Applications 142 (2009), 399-416.

[17] Prékopa, A. and Szántai, T., Flood control reservoir system design using stochastic programming, Mathematical Programming Study 9 (1978), 138-151.

[18] Prékopa, A. and Szántai, T., On optimal regulation of a storage level with application to the water level regulation of a lake, European Journal of Operations Research 3 (1979), 175-189.

[19] A. Prékopa, Stochastic Programming. Kluwer, Dordrecht, 1995.

[20] Prékopa, A., Probabilistic Programming, Chapter V in: A. Ruszczyński and A. Shapiro (eds.): Stochastic Programming, Handbooks in Operations Research and Management Science, Vol. 10, Elsevier, Amsterdam, 2003.

[21] W. Römisch and R. Schultz: Stability of solutions for stochastic programs with complete recourse, Mathematics of Operations Research 18 (1993), 590-609.

[22] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on Stochastic Programming*, MPS-SIAM series on optimization vol. 9, 2009.

[23] T. Szántai, Evaluation of a special multivariate Gamma distribution, Mathematical Programming Study 27 (1996), 1-16.

[24] T. Szántai, Improved Bounds and Simulation Procedures on the Value of the Multivariate Normal Probability Distribution Function, Annals of Operations Research 100 (2000), 85-101.

[25] A.F. Veinott, The supporting hyperplane method for unimodal programming, Operations Research 15 (1967), 147-152.

6 Appendix

The following result is based on the idea of proof in [21, Prop. 3.8].

Theorem 3 *Let the s-dimensional random vector ξ have a density f_ξ. Then, the distribution function F_ξ of ξ is globally Lipschitz continuous if and only if all marginal densities $f_\xi^{(i)}$ $(i = 1, \dots, s)$ are essentially bounded. Moreover, the largest of these bounds is a Lipschitz modulus for F_ξ with respect to the 1-norm.*

Proof. Let $x, y \in \mathbb{R}^s$ be arbitrary. Put

$$z^i := (y_1, \dots, y_i, x_{i+1}, \dots, x_s) \quad \forall i \in \{0, \dots, s\}.$$

Then, $z^0 = x$ and $z^s = y$. It follows that

$$\begin{aligned}
|F_\xi(x) - F_\xi(y)| &\leq \sum_{i=1}^{s} \left|F_\xi(z^i) - F_\xi(z^{i-1})\right| \\
&= \sum_{i=1}^{s} |\mathbb{P}(\xi_1 \leq y_1, \dots, \xi_i \leq y_i, \xi_{i+1} \leq x_{i+1}, \dots, \xi_s \leq x_s) - \\
&\qquad \mathbb{P}(\xi_1 \leq y_1, \dots, \xi_{i-1} \leq y_{i-1}, \xi_i \leq x_i, \dots, \xi_s \leq x_s)| \\
&= \sum_{i=1}^{s} \mathbb{P}(\xi_1 \leq y_1, \dots, \xi_{i-1} \leq y_{i-1}, \xi_i \in (\min\{x_i, y_i\}, \max\{x_i, y_i\}], \\
&\qquad \xi_{i+1} \leq x_{i+1}, \dots, \xi_s \leq x_s) \\
&\leq \sum_{i=1}^{s} \mathbb{P}(\xi_i \in (\min\{x_i, y_i\}, \max\{x_i, y_i\}]) \\
&= \sum_{i=1}^{s} \left(F_\xi^{(i)}(\max\{x_i, y_i\}) - F_\xi^{(i)}(\min\{x_i, y_i\})\right).
\end{aligned}$$

Assume that there exist $M_i \in \mathbb{R}$ such that $f_\xi^{(i)}(\tau) \leq M_i$ for almost all $\tau \in \mathbb{R}$ and for $i = 1, \dots, s$. Then, for all $i = 1, \dots, s$,

$$F_\xi^{(i)}(\max\{x_i, y_i\}) - F_\xi^{(i)}(\min\{x_i, y_i\}) = \int_{\min\{x_i, y_i\}}^{\max\{x_i, y_i\}} f_\xi^{(i)}(\tau) d\tau \leq M_i |x_i - y_i|.$$

Along with the previous estimate, the global Lipschitz continuity of F with modulus $M := \max_i M_i$ with respect to the 1-norm results :

$$|F_\xi(x) - F_\xi(y)| \le \sum_{i=1}^{s} M_i |x_i - y_i| \le M \|x - y\|_1 .$$

By equivalence of all norms in $\mathbb{R}^s$ the global Lipschitz continuity of F with respect to any norm follows.

Let conversely F_ξ be globally Lipschitz continuous with modulus M. Then, the marginal distribution functions $F_\xi^{(i)}$ are Lipschitz continuous with the same modulus. To see this, choose arbitrary $r, v \in \mathbb{R}$, $i \in \{1, \ldots, s\}$ and $\varepsilon > 0$. Defining

$$A_t := \left\{ z \in \mathbb{R}^s \,\middle|\, z \le (t, \ldots, t, \underset{i}{r}, t, \ldots, t) \right\} \quad (t \in \mathbb{R}),$$

it holds that A_t forms an increasing sequence with respect to set inclusion. Therefore,

$$\begin{aligned} \lim_{t\to\infty} F_\xi(t, \ldots, t, r, t, \ldots, t) &= \lim_{t\to\infty} \mathbb{P}(\xi \in A_t) = \mathbb{P}\left(\xi \in \bigcup_{t\in\mathbb{R}} A_t\right) = \mathbb{P}(\xi \in \mathbb{R}^{i-1} \times (-\infty, r] \times \mathbb{R}^{s-i}) \\ &= \mathbb{P}(\xi_i \le r) = F_\xi^{(i)}(r). \end{aligned}$$

Consequently, for $\bar{t}$ sufficiently large, one has that

$$\left| F_\xi(\bar{t}, \ldots, \bar{t}, r, \bar{t}, \ldots, \bar{t}) - F_\xi^{(i)}(r) \right|, \left| F_\xi(\bar{t}, \ldots, \bar{t}, v, \bar{t}, \ldots, \bar{t}) - F_\xi^{(i)}(v) \right| < \varepsilon.$$

We infer that

$$\left| F_\xi^{(i)}(r) - F_\xi^{(i)}(v) \right| \le 2\varepsilon + M |r - v| .$$

As $\varepsilon > 0$ was arbitrary, one arrives at the asserted global Lipschitz continuity of $F_\xi^{(i)}$ with modulus M. Furthermore, it holds that

$$F_\xi^{(i)}(r) = \int_{-\infty}^{r} f_\xi^{(i)}(\tau) d\tau.$$

By the Fundamental Theorem of calculus for the Lebesgue Integral, we know that $\left[F_\xi^{(i)}\right]'(r) = f_\xi^{(i)}(r)$ for almost all $r \in \mathbb{R}$. Hence the Lipschitz continuity of $F_\xi^{(i)}$ with modulus M yields that

$$f_\xi^{(i)}(r) = \lim_{h\downarrow 0} \frac{F_\xi^{(i)}(r+h) - F_\xi^{(i)}(r)}{h} \le M \quad \text{for almost all } r \in \mathbb{R}.$$

It results that the $f_\xi^{(i)}$ are essentially bounded. ■

Corollary 4 *Let $\xi \sim \mathcal{N}(\mu, \Sigma)$. Then, F_ξ is globally Lipschitz continuous such that*

$$\frac{1}{\sqrt{2\pi \min\limits_{i=1,\ldots,s} \Sigma_{ii}}}$$

is a Lipschitz modulus with respect to the 1*-norm.*