SURVEYS IN OPERATIONS RESEARCH AND MANAGEMENT SCIENCE

Publication information: Surveys in Operations Research and Management Science (ISSN 1876-7354). For 2012, volume 17 is scheduled for publication. Subscription prices are available upon request from the Publisher or from the Elsevier Customer Service Department nearest you or from this journal's website (http://www.elsevier.com/locate/sorms). Further information is available on this journal and other Elsevier products through Elsevier's website: (http://www.elsevier.com). Subscriptions are accepted on a prepaid basis only and are entered on a calendar year basis. Issues are sent by standard mail (surface within Europe, air delivery outside Europe). Priority rates are available upon request. Claims for missing issues should be made within six months of the date of dispatch.

Orders, claims, and journal enquiries: Please contact the Elsevier Customer Service Department nearest you:
St. Louis: Elsevier Customer Service Department, 3251 Riverport Lane, Maryland Heights, MO 63043, USA; phone: (877) 8397126 [toll free within the USA]; (+1) (314) 4478878 [outside the USA]; fax: (+1) (314) 4478077; e-mail: JournalCustomerService-usa@elsevier.com
Oxford: Elsevier Customer Service Department, The Boulevard, Langford Lane, Kidlington, OX5 1GB, UK; phone: (+44) (1865) 843434; fax: (+44) (1865) 843970; e-mail: JournalsCustomerServiceEMEA@elsevier.com
Tokyo: Elsevier Customer Service Department, 4F Higashi-Azabu, 1-Chome Bldg, 1-9-15 Higashi-Azabu, Minato-ku, Tokyo 106-0044, Japan; phone: (+81) (3) 5561 5037; fax: (+81) (3) 5561 5047; e-mail: JournalsCustomerServiceJapan@elsevier.com
Singapore: Elsevier Customer Service Department, 3 Killiney Road, #08-01 Winsland House I, Singapore 239519; phone: (+65) 6349 0222; fax: (+65) 6733 1510; e-mail: JournalsCustomerServiceAPAC@elsevier.com

Surveys in Operations Research and Management Science 17 (2012) 69–84

Contents lists available at SciVerse ScienceDirect

Surveys in Operations Research and Management Science

journal homepage: www.elsevier.com/locate/sorms

Review

Copula-based multivariate input modeling

Bahar Biller [a,*], Canan Gunes Corlu [b]

[a] Tepper School of Business, Carnegie Mellon University, United States
[b] Department of Industrial Engineering, Bilkent University, Turkey

ARTICLE INFO

Article history:
Received 7 April 2010
Received in revised form
13 February 2012
Accepted 3 April 2012

ABSTRACT

In this survey, we review the copula-based input models that are well suited to provide multivariate input-modeling support for stochastic simulations with dependent inputs. Specifically, we consider the situation in which the dependence between pairs of simulation input random variables is measured by *tail dependence* (i.e., the amount of dependence in the tails of a bivariate distribution) and review the techniques to construct copula-based input models representing positive tail dependencies. We complement the review with the parameter estimation from multivariate input data and the random-vector generation from the estimated input model with the purpose of driving the simulation.

Contents

1. Introduction

An important step in the design of a stochastic simulation is input modeling; i.e., choosing a probability distribution to represent the inputs of the system being studied. Input modeling is easily performed when the system inputs can be represented as a sequence of independent and identically distributed random variables. Reviews of such input models are available in [1–3].

The focus of this survey is on stochastic simulations with dependent inputs that require the use of flexible multivariate input models to capture their joint distributional properties. Examples

* Corresponding author.
E-mail address: billerb@andrew.cmu.edu (B. Biller).

1876-7354/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.sorms.2012.04.001

of dependent inputs in need of multivariate input-modeling support include the processing times of workpieces across work centers [4], the inter-arrival times of file accesses in computer systems [5–7], the medical characteristics of organ-transplant donors and recipients [8], arrival and service processes of Web servers [9], the inter-arrival times of customers in call centers [10], and the product demands and exchange rates of global supply chains [11]. Choosing correlation as the dependence measure, Ghosh and Squillante [9] show that ignoring the correlation between the inter-arrival and service times of Web server queues leads to a 25% overestimation of the waiting times, while assuming independent and identically distributed inter-arrival times leads to the underestimation of the expected waiting times by a factor of four. Clearly, the independence assumption can lead to very poor estimates of the performance measures when there is actually correlation present, and the consequences of ignoring correlation can be severe. A comprehensive review of the multivariate input models measuring dependence by correlation is available in [12]. Patton [13], on the other hand, demonstrates the need for an input model that captures not only the correlation but also the dependence in the tails of the exchange rate processes. What distinguishes our survey from others is its focus on this measure of dependence, which is known as the *tail dependence* and defined as the amount of dependence in the tails of a joint distribution. Other applications with focus on tail dependence include Corbett and Rajaram [14], Wagner et al. [15], and Tehrani et al. [16], and they demonstrate that it is imperative to develop multivariate input models that can capture stochastic tail dependencies among the input random variables of stochastic systems.

A close look at the existing literature reveals that multivariate input models can be classified into two types: random vectors and multivariate time series. Specifically, a random vector $\mathbf{X} = (X_1, X_2, \ldots, X_k)'$ denotes a collection of k random components, each of which is a real-valued random variable, and it is described by its joint distribution function. A k-dimensional time series $\mathbf{X}_t = \{(X_{1,t}, X_{2,t}, \ldots, X_{k,t})';\ t = 1, 2, \ldots\}$, on the other hand, denotes a sequence of random vectors observed at times $t = 1, 2, \ldots$. In this survey, we focus on random vectors and refer the reader to Biller and Ghosh [12] for a comprehensive review of the time-series processes for stochastic simulations. More specifically, we consider multi-dimensional copula-based input models that have the ability to capture a wide variety of dependence structures by describing dependence in a more general manner than correlation. The use of correlation as the measure of dependence in the simulation input-modeling research has been justified by the fact that making it possible for simulation users to incorporate dependence via correlation, while limited, is substantially better than the practice of ignoring dependence. However, when the simulation inputs are not jointly elliptically distributed and require the use of a joint probability distribution with positive dependence in the tails, correlation is no longer sufficient to describe the dependence structure of these simulation inputs. Therefore, in this survey we go beyond the use of correlation as a dependence measure and present the application of copula theory to multivariate input modeling with the purpose of constructing flexible density models that represent a wide variety of dependence structures. Additionally, we consider the problems of estimating the parameters of the copula-based input models from multivariate data and generating random vectors with the pre-specified marginal distributions and dependence structures to drive stochastic simulations. We refer the reader to Craney and White [17] for input modeling techniques when no data are available.

We organize the remainder of the paper as follows. In Section 2, we introduce correlation and tail dependence as the two measures of dependence that are used for multivariate input modeling with focus on the limitations of correlation as the dependence measure. In Section 3, we present copula-based input models with the ability to measure tail dependence; Section 3.1 reviews copula theory, Section 3.2 focuses on two-dimensional input models, and Section 3.3 extends the discussion to multivariate input models with three or more component random variables. In Section 4, we describe how to estimate the parameters of the copula-based input models from multivariate data via automated algorithms. We present the goodness-of-fit tests specifically designed for copula-based input models in Section 5. In Section 6, we provide the sampling algorithms that generate random vectors from the copula-based input models quickly and accurately to drive stochastic simulations. We conclude our review with a discussion of promising research areas in Section 7.

2. Two measures of dependence: correlation and tail dependence

Dependent random vectors are often specified partially in terms of the marginal distributions of their component random variables and pair-wise measures of dependence summarizing how these components interact with each other. Although this may not uniquely or even correctly specify the joint distribution of the random vector, the hope is to find a useful specification for the dependence structure among the components, while sparing the simulation practitioner the task of trying to estimate the full joint distribution. In Section 2.1, we review product-moment correlation and rank correlation as the dependence measures that are often used for this purpose in simulation input modeling. In Section 2.2, we introduce tail dependence and motivate its consideration in this survey as a measure of the dependence captured by neither product-moment correlation nor rank correlation. The dependence measures we consider in each of these sections are pair-wise measures, in that they are used to quantify the dependence between the pairs of random variables. It is important to note that, despite the focus of this survey, correlation and tail dependence are not the only means to measure dependence; we refer the reader to Nelsen [18] for a discussion of alternative measures.

2.1. Correlation

The product-moment correlation and the rank correlation are the two widely used measures of dependence in applications of stochastic simulations. Specifically, the rank correlation $r(i, j)$ between random variables X_i and X_j is defined by

$$r(i,j) = \frac{\mathrm{E}\left(F_i(X_i)F_j(X_j)\right) - \mathrm{E}\left(F_i(X_i)\right)\mathrm{E}\left(F_j(X_j)\right)}{\sqrt{\mathrm{Var}\left(F_i(X_i)\right)\mathrm{Var}\left(F_j(X_j)\right)}},$$

where F_i is the cumulative distribution function (cdf) of X_i [19]. The product-moment correlation $\rho_{\mathbf{X}}(i, j)$ between X_i and X_j with finite variances $\mathrm{Var}(X_i)$ and $\mathrm{Var}(X_j)$ is, on the other hand, given by

$$\rho_{\mathbf{X}}(i,j) = \frac{\mathrm{Cov}\left(X_i, X_j\right)}{\sqrt{\mathrm{Var}\left(X_i\right)\mathrm{Var}\left(X_j\right)}} = \frac{\mathrm{E}\left(X_iX_j\right) - \mathrm{E}\left(X_i\right)\mathrm{E}\left(X_j\right)}{\sqrt{\mathrm{Var}\left(X_i\right)\mathrm{Var}\left(X_j\right)}},$$

where $\mathrm{Cov}(X_i, X_j)$ is the product-moment covariance between X_i and X_j [20]. Thus, a correlation of 1 is the maximum possible for bivariate normal random variables $\Phi^{-1}(F_i(X_i))$ ($\equiv Z_i$) and $\Phi^{-1}(F_j(X_j))$ ($\equiv Z_j$), where Φ is the cdf of a standard normal random variable. Therefore, taking $\mathrm{Cov}(Z_i, Z_j) = 1$ is equivalent (in distribution) to setting $Z_i \leftarrow \Phi^{-1}(U)$ and $Z_j \leftarrow \Phi^{-1}(U)$, where U is a uniform random variable in the interval (0, 1) [21]. This definition of Z_i and Z_j implies that $X_i \leftarrow F_i^{-1}(U)$ and $X_j \leftarrow F_j^{-1}(U)$,

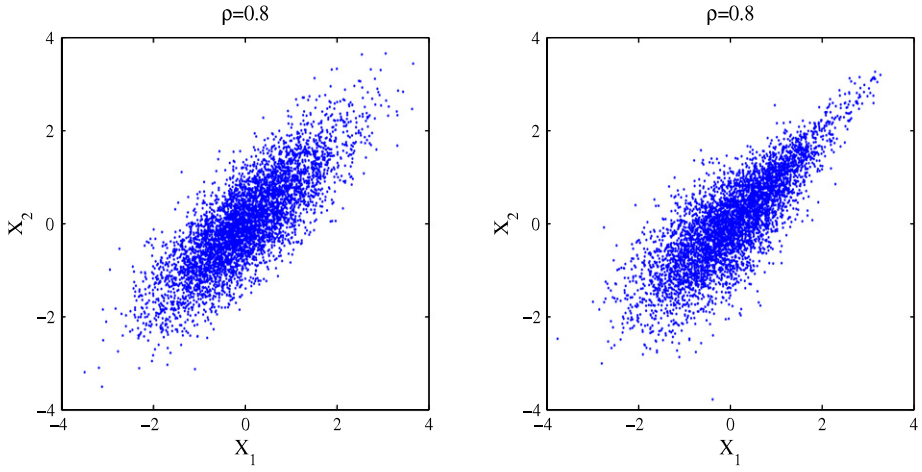


Fig. 1. Input models with standard normal marginals, but with different dependence structures.

from which it follows that $\rho_{\mathbf{X}}(i,j)$ takes on its maximum magnitude. Similarly, taking $\text{Cov}(Z_i, Z_j) = -1$ is equivalent (in distribution) to setting $X_i \leftarrow F_i^{-1}(U)$ and $X_j \leftarrow F_j^{-1}(1-U)$, in which case the correlation $\rho_{\mathbf{X}}(i,j)$ assumes the minimum possible value for the random variables X_i and X_j. Furthermore, in the special case of jointly normal input random variables, the product-moment correlation $\rho_{\mathbf{X}}(i,j)$ relates to the rank correlation $r(i,j)$ by $\rho_{\mathbf{X}}(i,j) = 2\sin(\pi r(i,j)/6)$ [22].

Despite its wide use, the product-moment correlation suffers from several limitations that have motivated simulation practitioners to look for alternative measures of dependence (e.g., [23,12]):

(1) The product-moment correlation cannot capture the nonlinear dependence between random variables. Consequently, it fails to model the non-zero dependence in the tails of a bivariate distribution. As an example, Fig. 1 shows 10 000 bivariate realizations sampled from two different input models constructed for the random vector $\mathbf{X} = (X_i, X_j)'$. In both of these models, X_i and X_j have standard normal marginal distributions and the product-moment correlation $\rho_{\mathbf{X}}(i,j)$ is 0.8, but with different dependence structures: The first model has the bivariate normal distribution, while the second model has the Gumbel distribution with parameter θ that takes the value of 3.8 (Section 3.2). More specifically, the dependence in the tails of the joint distribution is zero in the first model, while extreme positive realizations have a tendency to occur together in the second model. Thus, the structure of dependence in the two models cannot be distinguished on the grounds of product-moment correlation alone.

(2) A product-moment correlation of zero between two random variables does not guarantee their independence. For example, the correlation $\rho_{\mathbf{X}}(i,j)$ is zero between X_i and X_j that are uniformly distributed on the unit circle, but X_i and X_j are dependent as $X_i^2 + X_j^2 = 1$.

(3) A weak product-moment correlation does not imply low dependence. For example, the minimum product-moment correlation between two lognormal random variables with zero means and standard deviations of 1 and σ^2 is the correlation between e^Z and $e^{-\sigma Z}$, while the maximum product-moment correlation between these two random variables is the correlation between e^Z and $e^{\sigma Z}$, where Z is a standard normal random variable [23]. Although both of these correlations tend to zero with increasing values of σ, they are highly dependent.

(4) It follows from the definition of the product-moment correlation that $\rho_{\mathbf{X}}(i,j)$ takes values between -1 and 1, but the actual values $\rho_{\mathbf{X}}(i,j)$ can assume depends on the marginal distributions of the input random variables X_i and X_j [24]. For example, the attainable interval for the product-moment correlation of two lognormal random variables with zero means and standard deviations of 1 and 2 is $[-0.090, 0.666]$; i.e., it is not possible to find a bivariate distribution with these marginals and a product-moment correlation of 0.7.

(5) Product-moment correlation is not invariant under transformations of the input random variables. For example, the product-moment correlation between $\log(X_i)$ and $\log(X_j)$ is not the same as the product-moment correlation between X_i and X_j unless they are independent.

(6) Product-moment correlation is only defined when the variances of the random variables are finite. Therefore, it is not an appropriate dependence measure for heavy-tailed inputs with infinite variances.

The use of the rank correlation as the dependence measure avoids the theoretical deficiencies in 3, 4, 5, and 6. It further provides a natural way to separate the characterization of the component distribution functions $F_i(X_i)$ and $F_j(X_j)$ from that of the correlation between X_i and X_j. Danaher and Smith [25] use the rank correlation to study the interaction between the length of customer visit to an online store and the purchase amount. A bivariate plot of the visit duration of a customer against the total amount spent by this customer shows that the marginal distributions are far from being normal and the product-moment correlation between the visit duration and the purchase amount is 0.08, indicating a weak relationship. However, Danaher and Smith compute a stronger dependence via rank correlation with a value of 0.26. A comprehensive review of similar monotone and transformation-invariant measures of dependence like rank correlation can be found in [18]. However, as in the case of product-moment correlation, the dependence structures of the input models in Fig. 1 cannot be distinguished on the grounds of rank correlation alone, and deficiencies in 1 and 2 remain.

2.2. Tail dependence

Motivated by the pitfalls of correlation, focus on the recent multivariate input-modeling research has been finding alternative ways to understand and model dependence by moving away from simple measures of dependence. An alternative measure of dependence, which has been of interest in recent years, is tail dependence; i.e., the amount of dependence in the tails of a joint

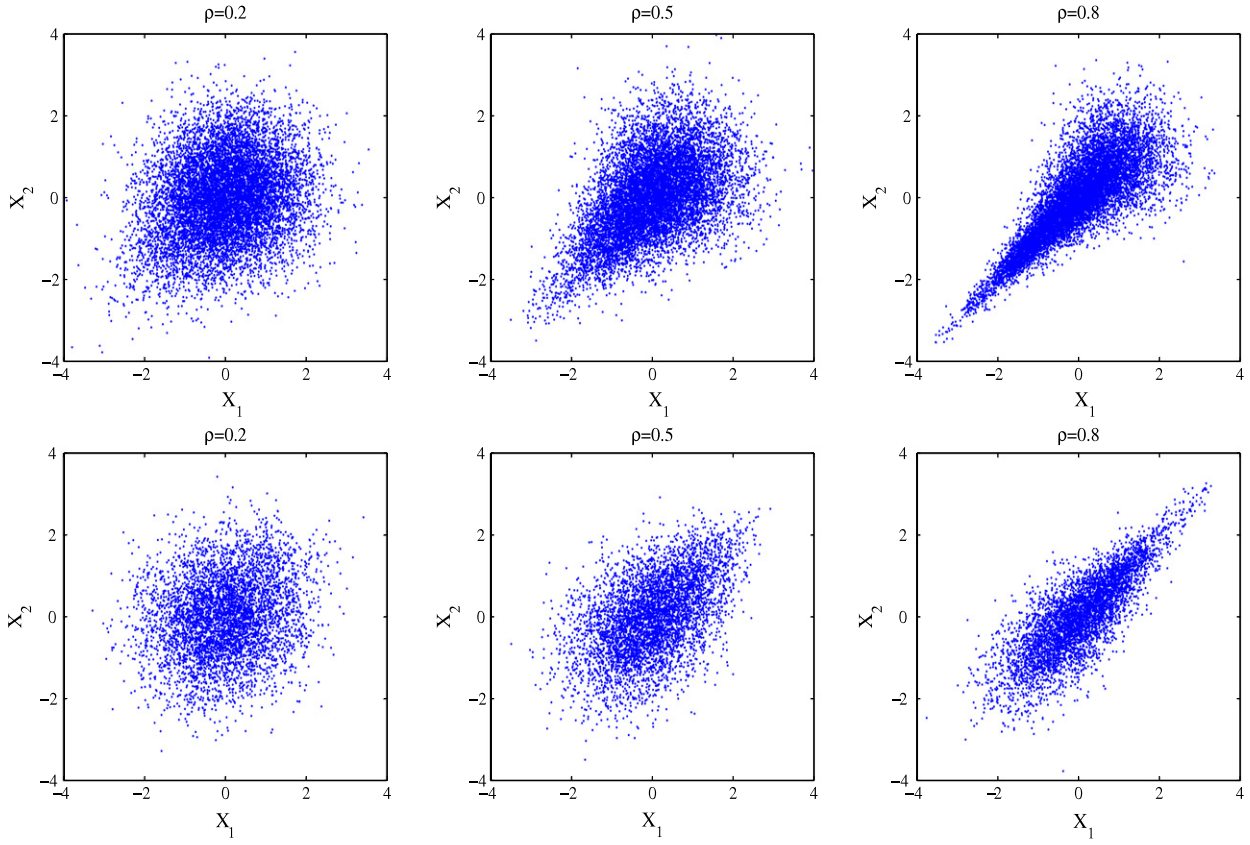


Fig. 2. Examples of bivariate input models with standard normal marginals and positive tail dependencies.

distribution [26]. Specifically, the positive lower-tail dependence $\nu_L(i,j)$ between random variables X_i and X_j is the amount of dependence in the lower-quadrant tail of the joint distribution of X_i and X_j and thus, it is given by $\lim_{\ell\downarrow 0} \Pr(X_i \leq F_i^{-1}(\ell)|X_j \leq F_j^{-1}(\ell))$. The positive upper-tail dependence $\nu_U(i,j)$ is, on the other hand, the amount of dependence in the upper-quadrant tail of the joint distribution of X_i and X_j and thus, it is given by $\lim_{\ell\uparrow 1} \Pr(X_i \geq F_i^{-1}(\ell)|X_j \geq F_j^{-1}(\ell))$. Fig. 2 provides examples of dependence structures for bivariate input models with standard normal marginals, but different positive tail dependencies [27]. The plots of the first, second, and third columns of this illustration are obtained for product-moment correlations of 0.2, 0.5, and 0.8, respectively. However, the first-row (second-row) plots exhibit greater dependence in the joint lower (upper) tail than in the joint upper (lower) tail; i.e., $\nu_L(i,j) > 0$ and $\nu_U(i,j) = 0$ ($\nu_U(i,j) > 0$ and $\nu_L(i,j) = 0$). More specifically, the first-row (second-row) plots are obtained from a Clayton (Gumbel) distribution with parameters 0.43 (1.18) for the first column, 1.00 (1.71) for the second column, and 3.11 (3.80) for the third column (Section 2). The plots of Fig. 2 also coincide with the plots of Burr–Pareto-Logistic family proposed by Cook and Johnson [28,29]. However, most multivariate input models, including the multivariate normal distribution [30], multivariate Johnson translation system [31], and the Normal-To-Anything (NORTA) distribution [32], measure the dependence between simulation inputs using correlation; therefore, they fail to capture the types of dependence structures illustrated in Fig. 2.

The need for input models with asymmetric dependence structures arises in situations where extreme positive realizations have a tendency to occur together. For example, Fortin and Kuzmics [33] show that the stock-return pairs of financial markets exhibit high dependence in the lower tail as well as low dependence in the upper tail of their joint distribution. Similar empirical evidence for the need to measure tail dependence is provided in [34–36]. Patton [13], on the other hand, studies the dependence between mark–dollar and Yen–dollar exchange rates and shows that they are more dependent when they are depreciating than when they are appreciating. Thus, the asymmetric dependence structure of these exchange-rate processes cannot be adequately modeled by the widely used multivariate input models of the simulation input-modeling literature.

Despite our focus on copula-based input models representing positive tail dependencies, it is important to note that the notion of negative tail dependence has been introduced by Zhang [37]. Specifically, the negative upper-tail dependence, which is called the lower–upper tail dependence in [37], is defined by $\lim_{\ell\downarrow 0} \Pr(X_i \leq F_i^{-1}(\ell)|X_j \geq F_j^{-1}(\ell))$, while the negative lower-tail dependence, which is also known as the upper–lower tail dependence, is given by $\lim_{\ell\uparrow 1} \Pr(X_i \geq F_i^{-1}(\ell)|X_j \leq F_j^{-1}(\ell))$. Copula theory reviewed in this paper for positive tail dependence readily extends to these definitions of negative tail dependence. However, most of the available empirical evidence has been for positive tail dependence. Although Sun and Wu [38] provide empirical evidence for the existence of negative tail dependence between the returns of the S&P 500 index and the returns of the Market Volatility Index, the negative tail dependence is rarely mentioned. Therefore, we restrict the focus of this survey to the representation of positive tail dependence. In the next section, we describe how to utilize copula theory to develop multivariate input models with the ability to capture asymmetric dependence structures, which are characterized by the positive tail dependencies among the simulation input random variables.

3. Copula-based input modeling

Copulas have been used extensively for a variety of financial applications including Value-at-Risk calculations [39–41], option

pricing [42–46], credit risk modeling [47,48], and portfolio optimization [49,50]. For a comprehensive review of the applications of copulas in finance, we refer the reader to Patton [51]. However, copulas are not used only for solving financial problems; they are also used for decision and risk analysis [52], aggregation of expert opinions [53], estimation of joint crop yield distributions [54], disruptive event modeling in project management [55], analysis of accident precursors [56,57], and modeling travel behavior by accommodating self-selection effects [58]. Recently, copulas have also been used in marketing to model the purchase behavior of customers [25], and in operations management to model retailer demands [14], supplier defaults [15], and supply disruptions in supply chains [16]. We provide the basics of copula theory in Section 3.1, review bivariate copula models in Section 3.2, and present multivariate copula models with three or more component random variables in Section 3.3.

3.1. Fundamentals of copula theory

A way of modeling the dependence among the components of a k-dimensional random vector that avoids the pitfalls of correlation is to use a k-dimensional copula [18, Definition 2.10.6]:

Definition 1. A k-dimensional copula is a function $C_k : [0, 1]^k \rightarrow [0, 1]$ with the following properties: (1) For every $\mathbf{u} = (u_1, u_2, \ldots, u_k)$ in $[0, 1]^k$, $C_k(\mathbf{u}) = 0$ if at least one coordinate of $\mathbf{u}$ is 0; and if all coordinates of $\mathbf{u}$ are 1 except u_ℓ, then $C_k(\mathbf{u}) = u_\ell$ for $\ell = 1, 2, \ldots, k$. (2) For every $\mathbf{a} = (a_1, a_2, \ldots, a_k)$ and $\mathbf{b} = (b_1, b_2, \ldots, b_k)$ in $[0, 1]^k$ such that $\mathbf{a} \leq \mathbf{b}$; i.e., $a_\ell \leq b_\ell,\ \ell = 1, 2, \ldots, k$, and for every k-box $[\mathbf{a}, \mathbf{b}]$; i.e., $[a_1, b_1] \times [a_2, b_2] \times \cdots \times [a_k, b_k]$, the C_k-volume given by $\Delta_{\mathbf{a}}^{\mathbf{b}} C_k(\mathbf{t}) = \Delta_{a_k}^{b_k} \Delta_{a_{k-1}}^{b_{k-1}} \cdots \Delta_{a_2}^{b_2} \Delta_{a_1}^{b_1} C_k(\mathbf{t})$ with $\Delta_{a_\ell}^{b_\ell} C_k(\mathbf{t}) = C_k(t_1, \ldots, t_{\ell-1}, b_\ell, t_{\ell+1}, \ldots, t_k) - C_k(t_1, \ldots, t_{\ell-1}, a_\ell, t_{\ell+1}, \ldots, t_k)$ is ≥ 0.

The first condition of this definition provides the lower bound on the joint distribution function and insures that the marginal distributions of the copula are uniform. The second condition insures that the probability of observing a point in a k-box is nonnegative. Thus, a k-dimensional copula is simply a k-dimensional distribution function with uniform marginals.

It is important to note that copulas are not the only means to obtain joint distribution functions from uniform marginals. There exist useful families of multivariate uniform distributions that could be the basis for multivariate input modeling: Multivariate Burr distribution [59], multivariate Pareto distribution [60], and multivariate logistic distribution [61] are essentially obtained by transforming uniform marginal distributions to arbitrary marginal distributions. Similarly, Plackett's distribution [62] is obtained by transforming bivariate uniform marginal distributions to arbitrary marginal distributions, while Morgenstern's distribution [63] with uniform marginals is generalized to have arbitrary marginal distributions in [64]. We refer the reader to Chapter 9 and Chapter 10 of Johnson [65] for a detailed discussion on obtaining multivariate distributions with arbitrary marginals from multivariate uniform distributions. This survey, however, focuses on tail dependence and therefore, describes the use of copulas to obtain multivariate distributions with arbitrary marginals and positive tail dependencies.

The use of a copula for representing the joint distribution of a random vector has been studied extensively for the last two decades [66,67,26,68,18]. The name "copula" emphasizes the manner in which a k-dimensional distribution function is "coupled" to its k (one-dimensional) marginal distributions; this property is formally stated in Sklar's theorem [18, Theorem 2.10.9]:

Theorem 1. *Let F be a k-dimensional distribution function with marginals $F_i,\ i = 1, 2, \ldots, k$. Then, there exists a k-dimensional copula C_k such that for all $x_i,\ i = 1, 2, \ldots, k$ in domain $\Re^k$,*

$$F(x_1, x_2, \ldots, x_k) = C_k(F_1(x_1), F_2(x_2), \ldots, F_k(x_k)).$$

If $F_i,\ i = 1, 2, \ldots, k$ are all continuous, then C_k is unique; otherwise, C_k is uniquely determined on $\text{Ran}F_1 \times \text{Ran}F_2 \times \cdots \times \text{Ran}F_k$. Conversely, if C_k is a k-dimensional copula and $F_i,\ i = 1, 2, \ldots, k$ are distribution functions, then the function F is a k-dimensional distribution function with marginals $F_i,\ i = 1, 2, \ldots, k$.

The major implication of this theorem is that copula C_k is the joint distribution function of $U_i \equiv F_i(X_i),\ i = 1, 2, \ldots, k$, where the random variables $U_i,\ i = 1, 2, \ldots, k$ are the probability integral transforms of $X_i,\ i = 1, 2, \ldots, k$. Thus, each of the random variables $U_i,\ i = 1, 2, \ldots, k$ follows a uniform distribution in [0, 1], regardless of the distributions of the component random variables $X_i,\ i = 1, 2, \ldots, k$. Moreover, C_k can be uniquely determined when the marginal cdfs $F_i,\ i = 1, 2, \ldots, k$ are all continuous. If the marginal cdfs $F_i,\ i = 1, 2, \ldots, k$ are all discrete, then C_k is uniquely determined on $\text{Ran}F_1 \times \text{Ran}F_2 \times \cdots \times \text{Ran}F_k$, where $\text{Ran}F_i$ is the range of the cdf F_i. In any case, the copula C_k captures the dependence structure of the joint cdf F and it can be written as $C_k(u_1, u_2, \ldots, u_k) = F(F_1^{-1}(u_1), F_2^{-1}(u_2), \ldots, F_k^{-1}(u_k))$, where F_i^{-1} is the generalized inverse of the marginal cdf F_i [18, Corollary 2.10.10].

The practical premise of Sklar's theorem in multivariate input modeling is that the joint distribution F can be constructed by choosing the marginal distributions $F_i,\ i = 1, 2, \ldots, k$ and the copula density function c_k separately. More specifically, any joint probability density function (pdf) f can be written as a product of its marginal pdfs $f_i,\ i = 1, 2, \ldots, k$ and copula density function c_k for differentiable cdfs $F_i,\ i = 1, 2, \ldots, k$ and differentiable copula C_k; i.e.,

$$f(x_1, x_2, \ldots, x_k) = c_k\Big(F_1(x_1), F_2(x_2), \ldots, F_k(x_k)\Big) \cdot \prod_{i=1}^{k} f_i(x_i).$$

The marginal pdf f_i is obtained from $\partial F_i(x_i)/\partial x_i$, while $\partial^k C_k(u_1, u_2, \ldots, u_k)/(\partial u_1 \partial u_2 \cdots \partial u_k)$ provides the k-dimensional copula density function c_k, encoding all of the information about the dependencies among the random variables $X_i,\ i = 1, 2, \ldots, k$. Thus, c_k takes the value of 1 when $X_i,\ i = 1, 2, \ldots, k$ are independent, and the joint density function reduces to the product of only the marginal pdfs.

There are numerous parametric families of copulas proposed in the literature, emphasizing different distributional properties. In this survey, we distinguish these copulas on the grounds of the tail dependence they capture; i.e., some copulas assign the value of zero to the tail dependence, while others represent positive lower-tail dependence and/or positive upper-tail dependence. Next, we present a copula-based definition of the tail dependence for a bivariate input model with copula C_2 [26]:

Definition 2. If a two-dimensional copula C_2 is such that $\lim_{u \downarrow 0} C_2(u, u)/u = \nu_L$ exists, then C_2 has lower-tail dependence if $\nu_L \in (0, 1]$ and no lower-tail dependence if $\nu_L = 0$. Similarly, if $\lim_{u \uparrow 1}(1-2u+C_2(u, u))/(1-u) = \nu_U$ exists, then C_2 has upper-tail dependence if $\nu_U \in (0, 1]$ and no upper-tail dependence if $\nu_U = 0$.

A close look at the existing literature reveals that one of the widely used copulas for bivariate input modeling is the two-dimensional normal distribution. The application of Definition 2 with the two-dimensional copula C_2 replaced by the two-dimensional standard normal cdf having correlation $\rho(1, 2) \in (-1, 1)$ between random variables Z_1 ($\equiv \Phi^{-1}(F_1(X_1))$) and Z_2 ($\equiv \Phi^{-1}(F_2(X_2))$) results in the computation of zero for both

Table 1
Bivariate copulas and their tail-dependence functions.

Copula	Parameter	ν_L	ν_U
Elliptically symmetric			
Normal	$-1 < \rho < 1$	0	0
t	$-1 < \rho < 1, 0 < d < \infty$	$2t_{d+1}\left(\sqrt{\frac{(d+1)(1-\rho)}{(1+\rho)}}\right)$	$2t_{d+1}\left(\sqrt{\frac{(d+1)(1-\rho)}{(1+\rho)}}\right)$
Archimedean			
Clayton	$\theta \geq 0$	$2^{-1/\theta}$	0
Gumbel	$\theta \geq 1$	0	$2-2^{1/\theta}$
	$\theta \geq 1$	$2^{-1/\theta}$	$2-2^{1/\theta}$
	$\theta_1 > 0, \theta_2 \geq 1$	$2^{-1/(\theta_1\theta_2)}$	$2-2^{1/\theta_2}$
Max-infinitely divisible	$\theta_1 \geq 0, \theta_2 \geq 1$	$2^{-1/\theta_1}$	$2-2^{1/\theta_2}$

the lower-tail dependence ($\nu_L(1,2) = 0$) and the upper-tail dependence ($\nu_U(1,2) = 0$) between random variables X_1 and X_2. Thus, regardless of the correlation $\rho(1,2) \in (-1,1)$ we choose, extreme events appear to occur independently in X_1 and X_2 when we go far enough into the tails. This explains why a bivariate input model building on the normal distribution fails to represent positive tail dependencies.

This discussion readily extends to the multi-dimensional setting with $k \geq 2$. A well known multivariate input model building also on the normal distribution is the NORTA random vector introduced by Cario and Nelson [32]. The goal of this input model is to match the pre-specified properties of the random vector $\mathbf{X} = (X_1, X_2, \ldots, X_k)'$, i.e., the marginal distributions $F_i,\ i = 1, 2, \ldots, k$ and the input correlations $\rho_{\mathbf{X}}(i,j),\ i,j = 1, 2, \ldots, k$, so as to drive the simulation with random vectors that have these properties. Therefore, the construction of the NORTA random vector builds on (1) applying the probability integral transform F_i to the input random variable X_i, which results in the uniform random variable U_i (i.e., $F_i(X_i) = U_i$); and (2) applying the inverse cdf Φ^{-1} to U_i from which the standard normal random variable Z_i is obtained (i.e., $Z_i = \Phi^{-1}(F_i(X_i))$). Consequently, we obtain the joint cdf F of the NORTA random vector $\mathbf{X}$ as follows:

$$
\begin{aligned}
&F(x_1, x_2, \ldots, x_k)\\
&\quad = \Pr(X_i \leq x_i;\ i = 1, 2, \ldots, k)\\
&\quad = \Pr(F_i(X_i) \leq F_i(x_i);\ i = 1, 2, \ldots, k)\\
&\quad = \Pr(U_i \leq u_i;\ i = 1, 2, \ldots, k)\\
&\quad = \Pr\left(\Phi^{-1}(U_i) \leq \Phi^{-1}(u_i);\ i = 1, 2, \ldots, k\right)\\
&\quad = \Pr\left(Z_i \leq \Phi^{-1}(u_i);\ i = 1, 2, \ldots, k\right)\\
&\quad = \Phi_k\left(\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_k); \boldsymbol{\Sigma}_k\right)\\
&\quad = \Phi_k\left(\Phi^{-1}(F_1(x_1)), \Phi^{-1}(F_2(x_2)), \ldots, \Phi^{-1}(F_k(x_k)); \boldsymbol{\Sigma}_k\right).
\end{aligned}
$$

In this representation, $\Phi_k(\cdot; \boldsymbol{\Sigma}_k)$ is the joint cdf of the standard normal random vector $\mathbf{Z} = (Z_1, Z_2, \ldots, Z_k)'$ with the correlation matrix $\boldsymbol{\Sigma}_k \equiv [\rho(i,j);\ i,j = 1, 2, \ldots, k]$, where $\rho(i,j)$ is the correlation between the random variables Z_i and Z_j. The function $\Phi_k(\cdot; \boldsymbol{\Sigma}_k)$, which is simply the k-dimensional normal copula, couples the arbitrary marginal cdfs $F_i,\ i = 1, 2, \ldots, k$ with the correlation matrix $\boldsymbol{\Sigma}_k$ to obtain the joint distribution function F. Thus, the dependence structure of the k-dimensional NORTA distribution is represented by the k-dimensional normal copula, explaining the failure of the NORTA distribution to represent non-zero tail dependencies. This result further extends to the multivariate normal distribution and the multivariate Johnson translation system, which are the special cases of the NORTA distribution. Specifically, we obtain the multivariate normal distribution by allowing each of the NORTA components to have a univariate normal distribution; we obtain the multivariate Johnson translation system by letting each component of the NORTA random vector have a univariate Johnson distribution [69].

A solution to the problem of capturing asymmetric dependence structures with positive tail dependencies is to replace the k-dimensional normal copula of the NORTA random vector $\mathbf{X}$ with a k-dimensional copula having the ability to match the pre-specified values of lower-tail dependencies $\nu_L(i,j),\ i,j = 1, 2, \ldots, k$ and upper-tail dependencies $\nu_U(i,j),\ i,j = 1, 2, \ldots, k$. Section 3.2 reviews the bivariate copula models that can be used for this purpose along with their tail dependence properties (i.e., $k = 2$). Focusing on random vectors with three or more components (i.e., $k \geq 3$), Section 3.3 considers the representation of tail dependencies by multivariate input models.

3.2. Bivariate copula models

The existing literature contains numerous parametric families of bivariate copulas, emphasizing different distributional properties [26,68,18]. In this survey, we consider the property of tail dependence. Table 1 provides the bivariate copulas that can be used for capturing this measure of dependence between any pair of random variables.

3.2.1. Elliptically symmetric copulas

Both the normal copula and the t copula fall into the class of elliptically symmetric copulas, which are introduced in [70] and discussed comprehensively in [71] as the generalizations of the normal copula to those with elliptically symmetric contours. Thus, the elliptically symmetric copulas inherit many of the tractable properties of the normal copula and maintain the advantage of being easy to sample. Specifically, the bivariate normal copula with the dependence parameter $\rho \in (-1,1)$ is given by

$$
C_2(u_1, u_2; \rho) = \int_{-\infty}^{\Phi^{-1}(u_1)} \int_{-\infty}^{\Phi^{-1}(u_2)} \frac{1}{2\pi\sqrt{1-\rho^2}} \times \exp\left\{-\frac{z_1^2 - 2\rho z_1 z_2 + z_2^2}{2(1-\rho^2)}\right\} dz_1 dz_2,
$$

while the bivariate t copula with the degrees of freedom $d \in (0, \infty)$ is given by

$$
C_2(u_1, u_2; \rho, d) = \int_{-\infty}^{t_d^{-1}(u_1)} \int_{-\infty}^{t_d^{-1}(u_2)} \frac{1}{2\pi\sqrt{1-\rho^2}} \times \left\{1 + \frac{z_1^2 - 2\rho z_1 z_2 + z_2^2}{d(1-\rho^2)}\right\}^{-(d+2)/2} dz_1 dz_2,
$$

where the parameter ρ corresponds to a dependence parameter when $d > 2$. An important distinction between these two copulas is that the normal copula assigns the value of zero to the tail dependencies, while both the lower-tail dependence and the upper-tail dependence captured by the t copula are given by $2t_{d+1}(\sqrt{d+1}\sqrt{1-\rho}/\sqrt{1+\rho})$, where t_d denotes the univariate t distribution function with d degrees of freedom [23]. Thus, the t copula assumes positive tail dependence even for $\rho = 0$, but the symmetry in the dependence structure (i.e., the equivalence between the lower-tail dependence and the upper-tail dependence) restricts its use for bivariate input modeling; i.e., a property that is shared by all elliptically symmetric copulas.

3.2.2. Archimedean copulas

A family of non-elliptical copulas with the ability to capture asymmetric dependence structures (i.e., different values for lower-tail and upper-tail dependencies) is the class of Archimedean copulas. These copulas are analytically tractable in the sense that many of their properties can be derived using elementary calculus [66]. Specifically, an Archimedean copula with parameter θ is of the form $C_2(u_1, u_2; \theta) = \phi^{-1}(\phi(u_1; \theta) + \phi(u_2; \theta); \theta)$, where $\phi^{-1}(\cdot; \theta)$ is the pseudo-inverse of $\phi(\cdot; \theta) : [0, 1] \to [0, \infty]$, which is a continuous, strictly decreasing, and convex generator function satisfying $\phi(1; \theta) = 0$. Different generator functions lead to different types of Archimedean copulas. For instance, the generator $\phi(t; \theta) = (t^{-\theta} - 1)/\theta$ produces the Clayton copula $C_2(u_1, u_2; \theta) = (u_1^{-\theta} + u_2^{-\theta} - 1)^{-1/\theta}$ with $0 \le \theta < \infty$, leading to a lower-tail dependence of $\nu_L(1, 2) = 2^{-1/\theta}$ between random variables X_1 and X_2 [72]. This is the copula function used for obtaining the first-row plots of Fig. 2, while the second-row plots are obtained from the generator function $\phi(t; \theta) = (-\log t)^\theta$, which leads to the Gumbel copula $C_2(u_1, u_2; \theta) = \exp(-((-\log u_1)^\theta + (-\log u_2)^\theta)^{1/\theta})$, $1 \le \theta < \infty$ with an upper-tail dependence of $\nu_U(1, 2) = 2 - 2^{1/\theta}$ [73]. Thus, neither the Clayton copula nor the Gumbel copula can simultaneously represent positive lower-tail and upper-tail dependencies. The generator function defined as $\phi(t; \theta) = (t^{-1} - 1)^\theta$, however, captures both the lower-tail dependence and the upper-tail dependence. More specifically, it produces the copula $C_2(u_1, u_2; \theta) = (1 + ((u_1^{-1} - 1)^\theta + (u_2^{-1} - 1)^\theta)^{1/\theta})^{-1}$ with $\theta \ge 1$, leading to a lower-tail dependence of $\nu_L(1, 2) = 2^{-1/\theta}$ and an upper-tail dependence of $\nu_U(1, 2) = 2 - 2^{1/\theta}$ [18]. However, the values $\nu_L(1, 2)$ and $\nu_U(1, 2)$ can take are limited by the copula parameter θ.

A way to overcome this limitation of a single-parameter copula is to construct a two-parameter Archimedean copula. This can be done by using a composite generator function of the form $\phi(t; \theta_1, \theta_2) = (\phi(t^{\theta_1}))^{\theta_2}$. For example, defining the composite generator function $\phi(t; \theta_1, \theta_2)$ as $(t^{-\theta_1} - 1)^{\theta_2}$ with $\theta_1 > 0$ and $\theta_2 \ge 1$ leads to the two-parameter Archimedean copula $C_2(u_1, u_2; \theta) = (((u_1^{\theta_1} - 1)^{\theta_2} + (u_2^{\theta_1} - 1)^{\theta_2})^{1/\theta_1} + 1)^{-1/\theta_1}$ with a lower-tail dependence of $\nu_L(1, 2) = 2^{-1/(\theta_1\theta_2)}$ and an upper-tail dependence of $\nu_U(1, 2) = 2 - 2^{1/\theta_2}$. The upper-tail dependence $\nu_U(1, 2)$ can take any value between 0 and 1, providing more flexibility than the single-parameter Archimedean copula in modeling the amount of dependence in the upper-quadrant tail of the underlying joint distribution function. However, the value the lower-tail dependence $\nu_L(1, 2)$ can assume is restricted by the value of the upper-tail dependence $\nu_U(1, 2)$. This modeling challenge is overcome by the max-infinitely divisible copulas introduced in the next section.

3.2.3. Max-infinitely divisible copulas

The lower-tail dependence $\nu_L(1, 2) \in (0, 1]$ and the upper-tail dependence $\nu_U(1, 2) \in (0, 1]$ between random variables X_1 and X_2 can be jointly represented by using a family of copulas of the form $C_2(u_1, u_2) = \Theta(-\log K(e^{-\Theta^{-1}(u_1)}, e^{-\Theta^{-1}(u_2)}))$, where Θ is a Laplace transformation and K is a max-infinitely divisible bivariate copula between random variables $U_1 = F_1(X_1)$ and $U_2 = F_2(X_2)$. Specifically, the distribution function K is max-infinitely divisible if K^α is a distribution function for every $\alpha > 0$ [74]. Furthermore, if K is an Archimedean copula, then C_2 is an Archimedean copula. For example, letting K be the one-parameter Clayton copula with parameter $\theta_1 \ge 0$ and allowing Θ to be the Laplace transformation satisfying $\Theta(t) = 1 - (1 - e^{-t})^{1/\theta_2}$ with parameter $\theta_2 \ge 1$ results in the two-parameter copula of the form $C_2(u_1, u_2; \theta_1, \theta_2) = \phi^{-1}(\phi(u_1; \theta_1, \theta_2) + \phi(u_2; \theta_1, \theta_2); \theta_1, \theta_2)$, where $\phi(t; \theta_1, \theta_2) = (1 - (1 - t)^{\theta_2})^{-\theta_1} - 1$ [26]. The application of Definition 2 to this copula function results in the lower-tail dependence $\nu_L(1, 2) = 2^{-1/\theta_1}$ and the upper-tail dependence $\nu_U(1, 2) = 2 - 2^{1/\theta_2}$. Thus, parameter θ_1 (θ_2) is used only for adjusting the lower-tail (upper-tail) dependence $\nu_L(1, 2)$ ($\nu_U(1, 2)$); i.e., the use of this two-parameter Archimedean copula for bivariate input modeling allows both the lower-tail dependence and the upper-tail dependence to assume arbitrary values in [0, 1]. Further discussion on two-parameter bivariate copulas can be found in [26].

3.3. Multivariate copula models

In this section, we discuss the extension of the bivariate copula models presented in Section 3.2 to three or more random variables. Specifically, Section 3.3.1 reviews the multivariate elliptical copulas; Section 3.3.2 presents the exchangeable multivariate Archimedean copulas; Section 3.3.3 provides the mixtures of max-infinitely divisible copulas; and finally Section 3.3.4 reviews the vine specifications that are known to be the most flexible multivariate copula models with the ability to represent asymmetric dependence structures with positive tail dependencies.

3.3.1. Multivariate elliptical copulas

Two widely used copulas for representing the joint distribution of three or more input random variables are the normal copula associated with the multivariate normal distribution and the t copula associated with the multivariate t distribution. Each of these copulas is a member of the elliptical copula family that is particularly easy to use for driving stochastic simulations with multiple inputs. Specifically, the multivariate normal copula with the positive definite correlation matrix $\mathbf{\Sigma}_k \equiv [\rho(i, j);\ i, j = 1, 2, \ldots, k]$ is given by

$$
\begin{aligned}
&C_k(u_1, u_2, \ldots, u_k; \mathbf{\Sigma}_k) \\
&\quad = \Phi_k\left(\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_k); \mathbf{\Sigma}_k\right), \\
&\quad = \int_{-\infty}^{\Phi^{-1}(u_1)} \cdots \int_{-\infty}^{\Phi^{-1}(u_k)} \frac{1}{(2\pi)^{k/2}|\mathbf{\Sigma}_k|^{1/2}} \\
&\qquad \times \exp\left(-\frac{1}{2}\mathbf{z}'\mathbf{\Sigma}_k^{-1}\mathbf{z}\right) dz_1 \ldots dz_k,
\end{aligned}
$$

where $\mathbf{z}$ denotes the vector $(z_1, z_2, \ldots, z_k)'$. The limitation of this multivariate copula in representing positive tail dependence becomes apparent in modeling investor's default risk [75,76]. For further discussion on the limitations of the multivariate normal copula, we refer the reader to Lipton and Rennie [77], Donnelly and Embrechts [78] and Brigo et al. [79].

The multivariate t copula with the parameters $\mathbf{\Sigma}_k \equiv [\rho(i, j); i, j = 1, 2, \ldots, k]$ and $d \in (0, \infty)$ is given by

$$
\begin{aligned}
&C_k(u_1, u_2, \ldots, u_k; \mathbf{\Sigma}_k, d) \\
&\quad = t\left(t_d^{-1}(u_1), t_d^{-1}(u_2), \ldots, t_d^{-1}(u_k); \mathbf{\Sigma}_k, d\right), \\
&\quad = \int_{-\infty}^{t_d^{-1}(u_1)} \cdots \int_{-\infty}^{t_d^{-1}(u_k)} \frac{\Gamma\left(\frac{d+k}{2}\right)|\mathbf{\Sigma}_k|^{-1/2}}{\Gamma\left(\frac{d}{2}\right)(d\pi)^{k/2}} \\
&\qquad \times \left(1 + \frac{1}{d}\mathbf{z}'\mathbf{\Sigma}_k^{-1}\mathbf{z}\right)^{-\frac{d+k}{2}} dz_1 \ldots dz_k.
\end{aligned}
$$

Even for $\rho(i, j) = 0$, the multivariate t copula represents symmetric tail dependencies $\nu_L(i, j) = \nu_U(i, j) = 2t_{d+1}(-\sqrt{d+1}\sqrt{1 - \rho(i, j)}/\sqrt{1 + \rho(i, j)})$, as can be deduced from Table 1 by focusing on the interaction between random variables X_i and X_j. Thus, the multivariate t copula fails to capture asymmetric dependence structures with different values for lower-tail and upper-tail dependencies. In the following section, this limitation of the multivariate t copula is overcome by using the exchangeable Archimedean copula for multivariate input modeling.

We conclude this section by noting that the class of the elliptically symmetric copulas includes the logistic copula [71], the exponential power copula [80], and the generalized t copula [81,82]. However, not all generalized t copulas are elliptically symmetric; they allow for different degrees of freedom and different types of dependencies among the components of the random vector. We refer the reader to Mendes and Arslan [82] for the characterization of the tail dependencies that can be captured by the generalized t copulas.

3.3.2. Exchangeable multivariate Archimedean copulas

The extension of the Archimedean copula introduced in Section 3.2 for bivariate input modeling is the k-dimensional Archimedean copula of the following form:

$$C_k(u_1, u_2, \ldots, u_k; \theta) = \phi^{-1}\left(\phi(u_1; \theta) + \phi(u_2; \theta) + \cdots + \phi(u_k; \theta); \theta\right).$$

$\phi(\cdot; \theta) : [0, 1] \to [0, \infty)$ is a continuous, strictly decreasing function that satisfies $\phi(0; \theta) = \infty$ and $\phi(1; \theta) = 0$. Additionally, $\phi^{-1}(\cdot; \theta)$ is a completely monotonic function on $[0, \infty)$ (i.e., $(-1)^\ell \partial^\ell \phi^{-1}(t; \theta)/\partial t^\ell \geq 0$ for all $t \in (0, \infty)$ and $\ell = 0, 1, \ldots, \infty$) [83]. This is a necessary and sufficient condition for the function $C_k(\cdot; \theta)$ to define a copula, and this condition is satisfied by both the k-dimensional Clayton copula and the k-dimensional Gumbel copula.

Specifically, the pseudo-inverse of the generator function of the Clayton copula; i.e., $\phi^{-1}(t; \theta) = (1 + t)^{-1/\theta}$ is completely monotonic on $[0, \infty)$. Therefore, the k-dimensional Clayton copula with $\theta > 0$ is given by

$$C_k(u_1, u_2, \ldots, u_k; \theta) = \left(u_1^{-\theta} + u_2^{-\theta} + \cdots + u_k^{-\theta} - k + 1\right)^{-1/\theta}.$$

Due to its ability to represent the joint probability of component random variables taking very small values together, Tehrani et al. [16] use this copula function to model dependent disruptions in supply chains caused by catastrophic events. Wagner et al. [15], on the other hand, use the multivariate Clayton copula for investigating the impact of positive default dependencies on the design of supplier portfolios. However, the k-dimensional Clayton copula assumes a lower-tail dependence of $2^{-1/\theta}$ and an upper-tail dependence of zero between all pairs of its components (i.e., $\nu_L(i, j) = 2^{-1/\theta}$ and $\nu_U(i, j) = 0$ for $i = 1, 2, \ldots, k$ and $j = i, i + 1, \ldots, k$), limiting its ability to perform flexible dependence modeling.

Similarly, the pseudo-inverse of the generator function of the Gumbel copula; i.e., $\phi^{-1}(t; \theta) = \exp(-t^{1/\theta})$ is completely monotonic on $[0, \infty)$. Thus, the two-dimensional Gumbel copula of Section 3.2 is generalized to the following k-dimensional copula with $\theta \geq 1$:

$$C_k(u_1, u_2, \ldots, u_k; \theta) = \exp\left(-\left((-\log u_1)^\theta + (-\log u_2)^\theta + \cdots + (-\log u_k)^\theta\right)^{1/\theta}\right).$$

The functional form of this copula leads to the tail dependencies of $\nu_L(i, j) = 0$ and $\nu_U(i, j) = 2 - 2^{1/\theta}$ for $i = 1, 2, \ldots, k$ and $j = i, i + 1, \ldots, k$.

The bivariate two-parameter Archimedean copulas can also be generalized to be the k-dimensional Archimedean copulas. As an example, we consider the composite generator function $\phi(t; \theta_1, \theta_2) = (t^{-\theta_1} - 1)^{\theta_2}$ with $\theta_1 > 0$ and $\theta_2 \geq 1$. Since the pseudo-inverse of this generator function is completely monotonic on $[0, \infty)$, we obtain the following k-dimensional copula:

$$C_k(u_1, u_2, \ldots, u_k; \theta_1, \theta_2) = \left(\left((u_1^{-\theta_1} - 1)^{\theta_2} + (u_2^{-\theta_1} - 1)^{\theta_2} + \cdots + (u_k^{-\theta_1} - 1)^{\theta_2}\right)^{1/\theta_2} + 1\right)^{-1/\theta_1}.$$

Therefore, the tail dependencies $\nu_L(i, j)$ and $\nu_U(i, j)$ are, respectively, identified as $2^{-1/(\theta_1\theta_2)}$ and $2 - 2^{1/\theta_2}$ for $i = 1, 2, \ldots, k$ and $j = i, i + 1, \ldots, k$.

3.3.3. Mixtures of max-infinitely divisible copulas

A flexible bivariate copula with the ability to represent any pair of lower-tail and upper-tail dependencies is the max-infinitely divisible bivariate copula (Section 3.2). Thus, a natural question to ask is whether it is possible to build on the mixtures of max-infinitely divisible bivariate copulas so as to construct a k-dimensional copula that would match arbitrary values of $\nu_L(i, j) > 0$ and $\nu_U(i, j) > 0$ for $i = 1, 2, \ldots, k$ and $j = i, i + 1, \ldots, k$. While the answer to this question is no, there exists a multivariate copula function, $C_k(F_1(x_1), F_2(x_2), \ldots, F_k(x_k))$ from the family of extreme value copulas, which captures arbitrary values for the positive upper-tail dependencies among the random variables $X_1, X_2, \ldots, X_k$, while modeling the positive lower-tail dependencies with limited flexibility:

$$\Theta\left(-\sum_{i=1}^{k}\sum_{j>i}^{k} \log K_{i,j}(e^{-\Theta^{-1}(F_i(x_i))/(\vartheta_i+k-1)}, e^{-\Theta^{-1}(F_j(x_j))/(\vartheta_j+k-1)}) + \sum_{i=1}^{k} \frac{\vartheta_i}{\vartheta_i + k - 1}\Theta^{-1}(F_i(x_i))\right).$$

For this representation, Joe [26] provides the interpretation that Laplace transformation Θ represents a minimal level of pairwise global dependence, bivariate copula $K_{i,j}$ adds individual pairwise dependence beyond the global dependence, and parameters ϑ_i, $i = 1, 2, \ldots, k$ lead to bivariate and multivariate asymmetry. Usually, the ϑ_i, $i = 1, 2, \ldots, k$ are nonnegative, although they can be negative if some of the copulas $K_{i,j}$ correspond to independence. What is important to recognize here is that the Laplace transformation Θ limits our ability to represent arbitrary values for $\nu_L(i, j) > 0$ and $\nu_U(i, j) > 0$. This can be shown by selecting the Galambos copula for $K_{i,j}$; i.e.,

$$K_{i,j}(u_i, u_j; \theta_{i,j}) = u_i u_j \exp\left(\left((-\log u_i)^{-\theta_{i,j}} + (-\log u_j)^{-\theta_{i,j}}\right)^{-1/\theta_{i,j}}\right)$$

with $0 \leq \theta_{i,j} < \infty$ [84], and by choosing the Laplace transformation Θ as gamma type; i.e., $\Theta(s) = (1 + s)^{-1/\delta}$ with parameter $\delta \geq 0$. In this case, we obtain an upper-tail dependence of

$$\nu_U(i, j) = \left((\vartheta_i + k - 1)^{\theta_{i,j}} + \left(\vartheta_j + k - 1\right)^{\theta_{i,j}}\right)^{-1/\theta_{i,j}}$$

and a lower-tail dependence of

$$\nu_L(i, j) = \left(2 - \left((\vartheta_i + k - 1)^{\theta_{i,j}} + \left(\vartheta_j + k - 1\right)^{\theta_{i,j}}\right)^{-1/\theta_{i,j}}\right)^{-1/\delta}$$

for $i = 1, 2, \ldots, k$ and $j = i, i + 1, \ldots, k$. The parameter $\theta_{i,j}$ is specific to the joint distribution of random variables X_i and X_j, while global parameter δ is shared among all k components X_i, $i = 1, 2, \ldots, k$. Therefore, we can represent any arbitrary value of upper-tail dependence between any pair of component random variables, but the global parameter δ limits our ability to have the same level of flexibility in representing lower-tail dependence.

To conclude, the use of mixtures of max-infinitely divisible copulas as well as multivariate Archimedean copulas for input modeling allows the representation of both the lower-tail dependence and the upper-tail dependence among the components of the random vector. However, the dependence structures captured by these copula families are restricted by the use of insufficient number of copula parameters. Although this particular limitation is overcome by the vine copula we present in the following section, the copula-vine method characterizes the dependence structure of the random vector using a mix of bivariate tail dependencies and bivariate conditional tail dependencies. Thus, in low-dimensional settings the simulation practitioner, who is interested in generating random vectors with pre-specified (unconditional) pair-wise

lower-tail and upper-tail dependencies, might find the use of the mixtures of max-infinitely divisible copulas of this section more convenient than the vines of the following section.

3.3.4. Vines

A vine is a graphical model introduced in [85], studied extensively in [86–89], and described comprehensively in [90] for constructing multivariate distributions using a mix of bivariate and conditional bivariate distributions of uniform random variables. Specifically, a k-dimensional vine is a nested set of $k-1$ spanning trees where the edges of tree j are the nodes of tree $j+1$, starting with a tree on a graph whose nodes are the k component random variables $X_i,\ i=1,2,\ldots,k$. A regular vine is, on the other hand, a vine in which two edges in tree j are joined by an edge in tree $j+1$ only if they share a common node [90, Section 4.4].

Fig. 3 presents a four-dimensional vine, which is a nested set of three spanning trees. Specifically, the first tree of this regular vine is the collection of three bivariate distributions; i.e., the joint distribution of the component random variables X_1 and X_2, the joint distribution of X_2 and X_3, and the joint distribution of X_3 and X_4. Since the solid line between X_1 and X_2 links these random variables, we associate this solid line with the joint distribution of X_1 and X_2, and label it as "1, 2", implying the copula density function $c_2(F_1(x_1), F_2(x_2))$. Similarly, we denote the solid line between X_2 and X_3 by "2, 3" associated with the copula density function $c_2(F_2(x_2), F_3(x_3))$, and use "3, 4" for the line between X_3 and X_4 to correspond to the copula density function $c_2(F_3(x_3), F_4(x_4))$. The second tree, on the other hand, contains two bivariate distributions, which are the joint distribution of $X_1|X_2$ and $X_3|X_2$ and the joint distribution of $X_2|X_3$ and $X_4|X_3$ with the respective copula density functions $c_2(F_{1|2}(x_1|x_2), F_{3|2}(x_3|x_2))$ and $c_2(F_{2|3}(x_2|x_3), F_{4|3}(x_4|x_3))$. Therefore, in Fig. 3 we mark the lines "…" of the second tree by "1, 3|2" and "2, 4|3". Finally, the third tree corresponds to the joint distribution of $X_1|X_2, X_3$ and $X_4|X_2, X_3$ represented by the line "–..–", which is further labeled by "1, 4|2, 3" implying the copula density function $c_2(F_{1|2,3}(x_1|x_2, x_3), F_{4|2,3}(x_4|x_2, x_3))$.

The two edges of a tree in Fig. 3 are joined only if they share a common component random variable to obtain an edge of the following tree. For example, the edges "1, 2" and "2, 3" of the first tree share the node associated with the random variable X_2 and they are combined for the edge "1, 3|2" of the second tree, while the edge "2, 4|3" of the second tree is obtained from the edges "2, 3" and "3, 4" sharing the node of the random variable X_3. Similarly, the edges "1, 3|2" and "2, 4|3", which share the nodes associated with the random variables X_2 and X_3 in the second tree, are joined by the edge "1, 4|2, 3" of the third tree in a manner that is consistent with the definition of a regular vine.

More specifically, the regular vine in Fig. 3 is known as the drawable vine ($\mathcal{D}$-vine). Its use for multivariate input modeling allows the four-dimensional copula density function to be represented by the product of the six bivariate linking copulas illustrated in Fig. 3; i.e.,

$$\begin{aligned} &c_4\left(F_1(x_1), F_2(x_2), F_3(x_3), F_4(x_4)\right) \\ &\quad = c_2\left(F_1(x_1), F_2(x_2)\right) \times c_2\left(F_2(x_2), F_3(x_3)\right) \\ &\qquad \times c_2\left(F_3(x_3), F_4(x_4)\right) \times c_2\left(F_{1|2}(x_1|x_2), F_{3|2}(x_3|x_2)\right) \\ &\qquad \times c_2\left(F_{2|3}(x_2|x_3), F_{4|3}(x_4|x_3)\right) \\ &\qquad \times c_2\left(F_{1|2,3}(x_1|x_2, x_3), F_{4|2,3}(x_4|x_2, x_3)\right). \end{aligned}$$

The bivariate linking copulas of the first row appear in the first tree of Fig. 3, while the bivariate linking copulas of the next two rows come from the second and third trees. Furthermore, this characterization of the four-dimensional random vector is easily generalized to a k-dimensional random vector; i.e., the joint

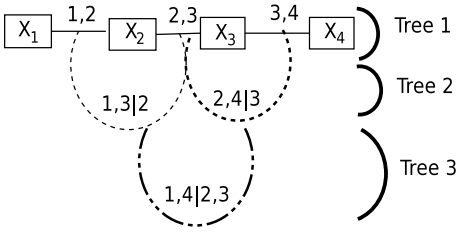


Fig. 3. A $\mathcal{D}$-vine specification on four dependent random variables.

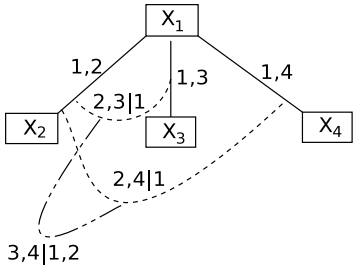


Fig. 4. A $\mathcal{C}$-vine specification on four dependent random variables.

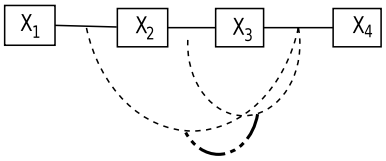


Fig. 5. An example of a non-regular vine on four dependent random variables.

density function of the k-dimensional $\mathcal{D}$-vine copula is obtained as a factorization of the univariate marginal density functions $\prod_{i=1}^{k} f_i(x_i)$ and the product of the bivariate (unconditional) linking copulas

$$\prod_{i=1}^{k-1} c_2\left(F_i(x_i), F_{i+1}(x_{i+1})\right)$$

of the first tree and the bivariate (conditional) linking copulas

$$\prod_{j=2}^{k-1}\prod_{i=1}^{k-j} c_2(F_{i|i+1,\ldots,i+j-1}(x_i|x_{i+1}, \ldots, x_{i+j-1}),$$
$$F_{i+j|i+1,\ldots,i+j-1}(x_{i+j}|x_{i+1}, \ldots, x_{i+j-1}))$$

of the remaining $k-2$ trees. Representing the dependence structure of the k-dimensional random vector via these $(k-1)(k-2)$ bivariate copulas instead of a single k-dimensional copula leads to computational tractability in the development of data-fitting algorithms in Section 4, goodness-of-fit tests in Section 5, and sampling procedures in Section 6 for copula-based multivariate input modeling.

Nevertheless, no unique regular vine exists for representing the dependence structure of a random vector. Fig. 4 presents another type of four-dimensional regular vine, the canonical vine ($\mathcal{C}$-vine), which is often used for multivariate input modeling. Fig. 5, on the other hand, provides an example of a four-dimensional non-regular vine. The comparison of this vine to the regular vines in Figs. 3 and 4 shows that the dependence structure of the regular vine is represented in terms of unconditional and conditional dependence measures that are algebraically independent of each other; i.e., they do not need to satisfy any algebraic constraints for positive definiteness. Therefore, all assignments of the numbers between -1 and 1 to the edges of the regular vine are consistent

in the sense that there is a joint distribution realizing these dependence measures.

A close look at the existing literature on vines reveals $\mathcal{C}$-vine and $\mathcal{D}$-vine to be the widely used regular vines for multi-dimensional dependence modeling. In recent years, the theory on vines has advanced considerably, accompanied by the development of several packages released by R software. In particular, we refer the reader to the recently released software packages CDVine, Vines, and Copula for statistical inference, distribution function evaluation, and simulation of $\mathcal{C}$-vine and $\mathcal{D}$-vine copulas. However, even if we narrow our choice of copulas to a $\mathcal{C}$-vine and a $\mathcal{D}$-vine, it is not clear which one of these vine structures would best capture the underlying dependence structure. Fortunately, this modeling challenge can be overcome by using the lattice-based algorithm of Maugis and Guegan [91] to find the vine copula representation that best models the dependence structure of the random vector of interest.

Joe et al. [92] show that vine copulas can cover a wide range of tail dependencies by choosing the bivariate linking copulas appropriately. However, the information set used to construct a vine copula is composed of a mix of unconditional bivariate tail dependencies and conditional bivariate tail dependencies. This raises the question of what the implied unconditional bivariate tail dependencies are for the given conditional tail dependencies. To explain this issue in detail, we consider the use of vines for constructing a three-dimensional random vector with components X_1, X_2, and X_3. If we are given bivariate joint cdfs $F(X_1, X_2)$ and $F(X_2, X_3)$ and the two-dimensional copula C_2 measuring the amount of conditional dependence between $X_1|X_2$ and $X_3|X_2$, then we first obtain the conditional marginal distributions $F_{1|2}$ and $F_{3|2}$ of the random variables $X_1|X_2$ and $X_3|X_2$, respectively, from $\partial F(x_1, x_2)/\partial x_2$ and $\partial F(x_2, x_3)/\partial x_2$. Then, we use these conditional marginal distributions and the copula C_2 together with the marginal distribution F_2 of the random variable X_2 to build the three-dimensional distribution

$$F(x_1, x_2, x_3) = \int_{-\infty}^{x_2} C_2(F_{1|2}(x_1|x), F_{3|2}(x_3|x))dF_2(x).$$

While the tail dependence parameters are explicitly chosen for the bivariate copulas, which link the random variables X_1 and X_2, the random variables X_2 and X_3, and the conditional random variables $X_1|X_2$ and $X_3|X_2$, the tail dependence between random variables X_1 and X_3 are not explicitly defined, even though it can be obtained from the joint cdf F. Under certain regularity conditions, Joe [26] proves the existence of an upper-tail dependence between X_1 and X_3 when both the bivariate copula linking X_1 and X_2 and the bivariate copula linking X_2 and X_3 have upper-tail dependencies. This result is further shown to hold for a k-dimensional random vector [92].

An alternative to the representation of the dependence structure of a multivariate random vector in terms of bivariate tail dependencies is to measure the multivariate tail dependence. Joe et al. [92] define the lower-tail dependence of the k-dimensional copula C_k as $\lim_{u\downarrow 0} C_k(u, u, \ldots, u)/u$ and the upper-tail dependence as $\lim_{u\downarrow 0} \bar{C}_k(1-u, 1-u, \ldots, 1-u)/u$ with $\bar{C}_k$ the survival function of C_k. It is shown that the multivariate tail dependence function of a vine copula can be expressed recursively by the tail dependence and conditional tail dependence functions of lower-dimensional margins. Also, a vine copula is identified as tail dependent if all the bivariate linking copulas associated with the first tree of the vine (i.e., baseline copulas) are tail dependent. More specifically, if the baseline copulas are all lower-tail (upper-tail) dependent, then the vine copula is lower-tail (upper-tail) dependent provided that $[0, 1]^2$ is the support of the bivariate linking copulas. However, if some of the baseline copulas are tail independent, then the vine copula is tail independent.

4. Fitting methods

In this section, we assume the availability of multivariate input data and review the methods of parameter estimation for the copulas of Section 3. Specifically, Section 4.1 discusses the use of frequentist methods (i.e., the methods that only rely on the available input data), while Section 4.2 reviews the limited literature on Bayesian methods (i.e., the methods that combine the input data with the prior information available about the joint distribution parameters) for copula-parameter estimation.

4.1. Frequentist method

This section reviews the methods of Maximum Likelihood Estimation (MLE) and the Inference For Margins (IFM) to estimate the parameters of the copula-based input models.

The application of the MLE method for parameter estimation requires the maximization of the joint likelihood function of the available input data $x_{i,t},\ i = 1, 2, \ldots, k,\ t = 1, 2, \ldots, n$ of length n with respect to the parameters of the joint density function. The utilization of Sklar's marginal-copula representation for the joint density function of the k-dimensional random vector $\mathbf{X} = (X_1, X_2, \ldots, X_k)'$ with marginal-distribution parameter vectors $\boldsymbol{\Psi}_i,\ i = 1, 2, \ldots, k$ and the copula parameter vector $\boldsymbol{\Upsilon}$ leads to the following log-likelihood function, where $\mathbf{x}$ denotes the vector of the available input data:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon}|\mathbf{x}) = &\sum_{i=1}^{k}\sum_{t=1}^{n} \log f_i(x_{i,t}; \boldsymbol{\Psi}_i) \\ &+ \sum_{t=1}^{n} \log\ c_k(F_1(x_{1,t}; \boldsymbol{\Psi}_1), \\ &F_2(x_{2,t}; \boldsymbol{\Psi}_2), \ldots, F_k(x_{k,t}; \boldsymbol{\Psi}_k); \boldsymbol{\Upsilon}).\end{aligned}$$

The maximum likelihood estimators $\tilde{\boldsymbol{\Psi}}_i^{\mathrm{MLE}},\ i = 1, 2, \ldots, k$, and $\tilde{\boldsymbol{\Upsilon}}^{\mathrm{MLE}}$ are obtained from the maximization of this log-likelihood function; i.e.,

$$\begin{aligned}&\left(\tilde{\boldsymbol{\Psi}}_1^{\mathrm{MLE}}, \tilde{\boldsymbol{\Psi}}_2^{\mathrm{MLE}}, \ldots, \tilde{\boldsymbol{\Psi}}_k^{\mathrm{MLE}}, \tilde{\boldsymbol{\Upsilon}}^{\mathrm{MLE}}\right)' \\ &\quad = \underset{\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon}}{\arg\max}\ \mathcal{L}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon}|\mathbf{x}).\end{aligned}$$

Under the regularity conditions of the asymptotic likelihood theory [93], $\tilde{\boldsymbol{\Psi}}_i^{\mathrm{MLE}},\ i = 1, 2, \ldots, k$, and $\tilde{\boldsymbol{\Upsilon}}^{\mathrm{MLE}}$ are strongly consistent and asymptotically normal; i.e., as $n \to \infty$, the random vector $\sqrt{n}((\tilde{\boldsymbol{\Psi}}_1^{\mathrm{MLE}}, \tilde{\boldsymbol{\Psi}}_2^{\mathrm{MLE}}, \ldots, \tilde{\boldsymbol{\Psi}}_k^{\mathrm{MLE}}, \tilde{\boldsymbol{\Upsilon}}^{\mathrm{MLE}})' - (\boldsymbol{\Psi}_1^*, \boldsymbol{\Psi}_2^*, \ldots, \boldsymbol{\Psi}_k^*, \boldsymbol{\Upsilon}^*)')$ converges to be normally distributed with the zero mean vector and the variance–covariance matrix $\tau^{-1}(\boldsymbol{\Psi}_1^*, \boldsymbol{\Psi}_2^*, \ldots, \boldsymbol{\Psi}_k^*, \boldsymbol{\Upsilon}^*)$, where $\boldsymbol{\Psi}_i^*,\ i = 1, 2, \ldots, k$, and $\boldsymbol{\Upsilon}^*$ are the true marginal-distribution and copula parameter vectors and $\tau(\boldsymbol{\Psi}_1^*, \boldsymbol{\Psi}_2^*, \ldots, \boldsymbol{\Psi}_k^*, \boldsymbol{\Upsilon}^*)$ is the Fisher's information matrix [94].

Despite the wide use of the MLE method for parameter estimation, the simultaneous estimation of the marginal-distribution parameters and the copula parameters gets computationally demanding with k, the number of component random variables. Fortunately, the log-likelihood function $\mathcal{L}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon}|\mathbf{x})$ can be partitioned into terms that separately relate to the component marginal density functions and the copula density function. This makes it possible to estimate the marginal distribution parameters and the copula parameters in two stages. The resulting method of parameter estimation is known as the IFM method [26, Chapter 10]. First, we independently estimate the marginal distribution parameters $\tilde{\boldsymbol{\Psi}}_i^{\mathrm{IFM}},\ i = 1, 2, \ldots, k$ from the log-likelihood functions associated with the marginal distributions

$\mathcal{L}_i(\boldsymbol{\Psi}_i|\mathbf{x}_i) = \sum_{t=1}^{n} \log f_i(x_{i,t};\boldsymbol{\Psi}_i),\ i = 1, 2, \ldots, k$; i.e., $\tilde{\boldsymbol{\Psi}}_i^{\text{IFM}} = \arg\max_{\boldsymbol{\Psi}_i} \mathcal{L}_i(\boldsymbol{\Psi}_i|\mathbf{x}_i)$ for $i = 1, 2, \ldots, k$. Then, we estimate the copula parameter vector $\tilde{\boldsymbol{\Upsilon}}^{\text{IFM}}$ from the part of the log-likelihood function that is associated with the copula density function; i.e.,

$$\mathcal{L}_c(\boldsymbol{\Upsilon}|\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \mathbf{x}) = \sum_{t=1}^{n} \log\ c_k\left(F_1(x_{1,t};\boldsymbol{\Psi}_1), F_2(x_{2,t};\boldsymbol{\Psi}_2), \ldots, F_k(x_{k,t};\boldsymbol{\Psi}_k);\boldsymbol{\Upsilon}\right),$$

by using the marginal distribution parameter estimates $\tilde{\boldsymbol{\Psi}}_i^{\text{IFM}}$, $i = 1, 2, \ldots, k$ obtained in the first stage; i.e., $\tilde{\boldsymbol{\Upsilon}}^{\text{IFM}} = \arg\max_{\boldsymbol{\Upsilon}} \mathcal{L}_c(\boldsymbol{\Upsilon}|\tilde{\boldsymbol{\Psi}}_1^{\text{IFM}}, \tilde{\boldsymbol{\Psi}}_2^{\text{IFM}}, \ldots, \tilde{\boldsymbol{\Psi}}_k^{\text{IFM}}, \mathbf{x})$. However, the use of a copula-vine specification to construct a k-dimensional input model as a nested set of $k-1$ spanning trees further decomposes the estimation of the copula parameter vector into $k-1$ stages.

The IFM estimators $\tilde{\boldsymbol{\Psi}}_i^{\text{IFM}}$, $i = 1, 2, \ldots, k$, and $\tilde{\boldsymbol{\Upsilon}}^{\text{IFM}}$ are generally not equivalent to the MLE estimators $\tilde{\boldsymbol{\Psi}}_i^{\text{MLE}}$, $i = 1, 2, \ldots, k$, and $\tilde{\boldsymbol{\Upsilon}}^{\text{MLE}}$ except for the normal copula with the normally distributed marginal distributions. Nevertheless, Joe [26] proves that the IFM estimators are also strongly consistent and asymptotically normal under certain regularity conditions; i.e., $\sqrt{n}((\tilde{\boldsymbol{\Psi}}_1^{\text{IFM}}, \tilde{\boldsymbol{\Psi}}_2^{\text{IFM}}, \ldots, \tilde{\boldsymbol{\Psi}}_k^{\text{IFM}}, \tilde{\boldsymbol{\Upsilon}}^{\text{IFM}})' - (\boldsymbol{\Psi}_1^*, \boldsymbol{\Psi}_2^*, \ldots, \boldsymbol{\Psi}_k^*, \boldsymbol{\Upsilon}^*)') \rightarrow \mathbf{N}(\mathbf{0}, \mathbf{G}^{-1}(\boldsymbol{\Psi}_1^*, \boldsymbol{\Psi}_2^*, \ldots, \boldsymbol{\Psi}_k^*, \boldsymbol{\Upsilon}^*))$ with $\mathbf{G}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon})$ the Godambe information matrix. If we define $\mathbf{S}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon})$ as a vector whose transpose is the collection of $\nabla_{\boldsymbol{\Psi}_i} \mathcal{L}_i(\boldsymbol{\Psi}_i|\mathbf{x}_i)$, $i = 1, 2, \ldots, k$, and $\nabla_{\boldsymbol{\Upsilon}} \mathcal{L}_c(\boldsymbol{\Upsilon}|\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \mathbf{x})$, then the Godambe information matrix $\mathbf{G}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon})$ is given by $\mathbf{D}^{-1}\mathbf{V}(\mathbf{D}^{-1})'$, where $\mathbf{D} = \mathrm{E}(\nabla\mathbf{S}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon})/\ \nabla\ (\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon}))$ and $\mathbf{V} = \mathrm{E}(\mathbf{S}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon})\mathbf{S}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon})')$ [44]. However, it is well known that the estimators obtained from the entire log-likelihood function are the most efficient estimators, in that they attain the minimum asymptotic variance bound, while the estimators obtained from the multi-stage maximum-likelihood estimation do not attain this bound. Furthermore, an inappropriate choice of models for the marginal distributions may have detrimental effects on the estimation of the copula parameter vector [95]. Therefore, the IFM estimators $\tilde{\boldsymbol{\Psi}}_i^{\text{IFM}}$, $i = 1, 2, \ldots, k$, and $\tilde{\boldsymbol{\Upsilon}}^{\text{IFM}}$ provide good starting points for obtaining more efficient estimators by solving $\mathcal{L}(\boldsymbol{\Psi}_1, \boldsymbol{\Psi}_2, \ldots, \boldsymbol{\Psi}_k, \boldsymbol{\Upsilon}|\mathbf{x})$ for both the marginal-distribution parameter vectors $\boldsymbol{\Psi}_i$, $i = 1, 2, \ldots, k$ and the copula parameter vector $\boldsymbol{\Upsilon}$ in a single stage.

4.2. Bayesian method

An alternative method to the use of the frequentist MLE and IFM methods for parameter estimation is the Bayesian method. The Bayesian method starts with the determination of a joint prior density function that quantifies the initial uncertainty about the multivariate distribution parameters. Then, the joint prior distribution is updated with the joint likelihood function of the available historical data, and we obtain the joint posterior distribution that captures the uncertainty associated with the multivariate distribution parameters. The derivation of the posterior density function generally requires the computation of high-dimensional integrals, limiting the use of the Bayesian method for parameter estimation in multi-dimensional settings. However, the advancement of the Markov Chain Monte Carlo (MCMC) method in recent years has made it possible to estimate distribution parameters with any posterior density function. In this section, we review Bayesian estimation using the MCMC method to obtain parameter estimates for three copula models: the multivariate normal copula, the multivariate t copula, and the $\mathcal{D}$-vine built on bivariate t copulas. Although the multivariate normal copula does not capture any tail dependence, we describe a Bayesian estimation for this copula due to its relation to the multivariate t copula.

Dalla Valle [96] builds a Bayesian model to estimate the correlation matrix $\boldsymbol{\Sigma}_k$ of the k-dimensional normal copula density function with a likelihood function of the following form, where $\mathbf{x}_t$ is the vector of input data $x_{i,t}$, $i = 1, 2, \ldots, k$ and $\mathbf{I}_k$ is the k-dimensional identity matrix:

$$|\boldsymbol{\Sigma}_k|^{-n/2} \exp\left(-\frac{1}{2}\sum_{t=1}^{n} \mathbf{x}_t'\left(\boldsymbol{\Sigma}_k^{-1} - \mathbf{I}_k\right)\mathbf{x}_t\right).$$

Dalla Valle [96] uses the Inverse Wishart density function as the conjugate prior for the correlation matrix $\boldsymbol{\Sigma}_k$; i.e., $\boldsymbol{\Sigma}_k \sim$ Inverse Wishart$(\alpha, \mathbf{B})$ [97]. The parameters of this conjugate prior are chosen as $\alpha \equiv k+1$ and $\mathbf{B} \equiv [\mathrm{diag}(\gamma_i);\ i = 1, 2, \ldots, k]$, where γ_i is the gamma random variable whose scale and shape parameters are 0.001 to minimize the impact of the prior density function on the posterior density function. Therefore, the posterior density function is identified as the Inverse Wishart with parameters $n/2 + \alpha$ and $\mathbf{B} + \sum_{t=1}^{n} \mathbf{x}_t\mathbf{x}_t'/2$. Dalla Valle [96] obtains an estimate of $\boldsymbol{\Sigma}_k$ from this posterior density function via the use of a Gibbs sampler algorithm, which is a widely used MCMC method that requires the sampling of $\boldsymbol{\Sigma}_k$ from its conditional posterior density function. The key idea behind any MCMC method is to simulate a random walk on the entire parameter space that converges to the joint posterior density function of the parameters [98]. Then, the parameters sampled in each replication of the Gibbs sampler algorithm are averaged to estimate the copula parameters.

Another distribution Dalla Valle [96] considers is the k-dimensional t copula with a likelihood function given by

$$\left[\Gamma\left(\frac{d+k}{2}\right)\right]^n \left[\Gamma\left(\frac{d+1}{2}\right)\right]^{-kn} \left[\Gamma\left(\frac{d}{2}\right)\right]^{n(k-1)} \times |\boldsymbol{\Sigma}_k|^{-n/2} \prod_{t=1}^{n}\left[1 + \frac{\mathbf{x}_t'\boldsymbol{\Sigma}_k^{-1}\mathbf{x}_t}{d}\right]^{-\frac{d+k}{2}} \prod_{t=1}^{n}\prod_{i=1}^{k}\left[1 + \frac{x_{it}^2}{d}\right]^{\frac{d+1}{2}}.$$

Similar to the selection of a joint prior density function for the normal copula parameters, Dalla Valle chooses the Inverse Wishart prior with parameters α and $\mathbf{B}$ for $\boldsymbol{\Sigma}_k$ of the k-dimensional t copula. Additionally, the truncated Poisson distribution with parameter h is selected as a prior for d. Consequently, the posterior density function of $\boldsymbol{\Sigma}_k$ is obtained as

$$|\boldsymbol{\Sigma}_k|^{-\frac{n}{2}-\alpha+\frac{k+1}{2}} \prod_{t=1}^{n}\left[1 + \frac{\mathbf{x}_t'\boldsymbol{\Sigma}_k^{-1}\mathbf{x}_t}{d}\right]^{-\frac{d+k}{2}} \exp\left[-\mathrm{tr}(\mathbf{B}\boldsymbol{\Sigma}_k^{-1})\right],$$

while the posterior density function of d is given by

$$\left[\Gamma\left(\frac{d+k}{2}\right)\right]^n \left[\Gamma\left(\frac{d+1}{2}\right)\right]^{-kn} \left[\Gamma\left(\frac{d}{2}\right)\right]^{n(k-1)} \times \prod_{t=1}^{n}\left[1 + \frac{\mathbf{x}_t'\boldsymbol{\Sigma}_k^{-1}\mathbf{x}_t}{d}\right]^{\frac{d+k}{2}} \prod_{t=1}^{n}\prod_{i=1}^{k}\left[1 + \frac{x_{it}^2}{d}\right]^{\frac{d+1}{2}} \frac{h^d}{d!}.$$

Both of these posterior density functions are in non-standard forms; therefore, Dalla Valle [96] resorts to the use of an MCMC method known as the Metropolis Hastings algorithm for estimating $\boldsymbol{\Sigma}_k$ and d. The implementation details of this algorithm can be found in [98].

Czado and Min [99], on the other hand, consider the Bayesian estimation of the parameters of a $\mathcal{D}$-vine built on bivariate t copulas with the following key assumption: The degrees of freedom parameter, the dependence parameters of the baseline

copulas, and the conditional dependence parameters of the conditional bivariate linking copulas are independent. Czado and Min use a uniform prior in the interval (1, 100) for the degrees of freedom parameter and a uniform prior in the interval $(-1, 1)$ for each of the dependence parameters. However, the resulting joint posterior density function of the copula parameters does not have a well-defined form; therefore, the authors also use the Metropolis Hastings algorithm for estimating the vine copula parameters. Details of implementing this algorithm for the $\mathcal{D}$-vine are available in [99].

5. Goodness-of-fit tests

The next step in multivariate input modeling is to assess the goodness of the estimated copula parameters in capturing the joint distributional characteristics of the available input data. A close look at the existing literature reveals several goodness-of-fit tests that have been built specifically for copulas; we refer the reader to Berg [100] and Genest et al. [101] for a comprehensive review as well as a comparison of these tests. In this section, we review only two of these tests; i.e., a test that is based on the empirical copula and a test proposed by Genest et al. [101] building on a variation of Rosenblatt's transformation. The reason for considering only these tests in our survey is that they work for any copula function and they are not sensitive to the grouping of the data. Genest et al. [101] call the tests with such characteristics the "blanket" tests. Among all blanket tests considered in [101], the ones that are reviewed here standout as the best blanket tests that have been developed for copula models.

To focus our presentation on the fit of the copula itself and avoid any distributional assumptions about the component marginal distributions, we consider the ranks of the historical data in this section and denote them by $b_{i,t},\ i = 1, 2, \ldots, k,\ t = 1, 2, \ldots, n$. Therefore, we define a pseudo-observation, $u_{i,t}$ by $b_{i,t}/(n+1)$. The pseudo-observations $u_{i,t},\ i = 1, 2, \ldots, k,\ t = 1, 2, \ldots, n$ can be interpreted as a sample from the underlying k-dimensional copula C_k, whose empirical counterpart $C_{k,n}(\mathbf{u})$ with $\mathbf{u} = (u_1, u_2, \ldots, u_k)' \in [0, 1]^k$ is given by

$$C_{k,n}(\mathbf{u}) = \frac{1}{n}\sum_{t=1}^{n} \mathcal{I}(u_{1,t} \le u_1, u_{2,t} \le u_2, \ldots, u_{k,t} \le u_k),$$

where $\mathcal{I}$ represents the indicator function that equals 1 if its argument is true, and 0 otherwise [102]. It is important to note that the pseudo-observations are not mutually independent and they are only approximately uniform on [0, 1]. Any inference procedure based on the pseudo-observations should take these features into account. A detailed discussion about the significance of these properties of the pseudo-observations on the performance of the goodness-of-fit tests is available in [101].

Building on the empirical copula $C_{k,n}$, Genest and Rémillard [103] propose the use of the rank-based version of Cramér von Mises test statistic,

$$S_n = \int_{[0,1]^k} n\left(C_{k,n}(\mathbf{u}) - C_k(\mathbf{u}; \hat{\boldsymbol{\Upsilon}}_n)\right)^2 dC_{k,n}(\mathbf{u})$$

and the rank-based version of the Kolmogorov–Smirnov test statistic,

$$T_n = \sup_{\mathbf{u}\in[0,1]^k} \left|\sqrt{n}\left(C_{k,n}(\mathbf{u}) - C_k(\mathbf{u}; \hat{\boldsymbol{\Upsilon}}_n)\right)\right|,$$

where $C_k(\cdot; \hat{\boldsymbol{\Upsilon}}_n)$ denotes the k-dimensional copula with the dependence parameter vector $\hat{\boldsymbol{\Upsilon}}_n$ estimated from multivariate data of length n. Small values of S_n and T_n indicate the goodness of the copula fit. The approximate p values of these tests can be obtained from their limiting distributions. However, in practice the limiting distributions of S_n and T_n depend on the copula family of interest and the unknown dependence parameter vector $\boldsymbol{\Upsilon}$. Therefore, the common practice is to obtain the asymptotic distribution of these tests and the approximate p values via Monte Carlo methods [101]. A parametric bootstrap procedure, which can be used for this purpose, is available in Appendix A of Genest et al. [101], while the consistency and the asymptotic convergence of these tests are proven in [103].

The second goodness-of-fit test we present for copula-based input modeling is based on Rosenblatt's transformation [104], which is a general procedure used in simulation for decomposing a random vector with a given distribution into mutually independent components that are uniformly distributed on the unit interval. Specifically, Rosenblatt [104] transforms the random vector $\mathbf{U} = (U_1, U_2, \ldots, U_k)' \in [0, 1]^k$ to the random vector $\mathcal{R}(\mathbf{U}) = (Y_1, Y_2, \ldots, Y_k)' \in [0, 1]^k$ with $Y_1 = U_1$ and

$$Y_i = \frac{\left(\partial^{i-1} C_k\left(U_1, U_2, \ldots, U_{i-1}, U_i, 1, \ldots, 1; \boldsymbol{\Upsilon}\right)\right) / \left(\partial U_1 \partial U_2 \cdots \partial U_{i-1}\right)}{\left(\partial^{i-1} C_k\left(U_1, U_2, \ldots, U_{i-1}, 1, 1, \ldots, 1; \boldsymbol{\Upsilon}\right)\right) / \left(\partial U_1 \partial U_2 \cdots \partial U_{i-1}\right)}$$

for $i = 2, 3, \ldots, k$. The use of this probability integral transform of the k-dimensional copula $C_k(\cdot; \boldsymbol{\Upsilon})$ for measuring the goodness of a copula fit leads to the interpretation that the pseudo-observations $\mathbf{y}_t = (y_{1,t}, y_{2,t}, \ldots, y_{k,t})',\ t = 1, 2, \ldots, n$, which are obtained from $\mathcal{R}(\mathbf{u}_t),\ t = 1, 2, \ldots, n$ with $\mathbf{u}_t = (u_{1,t}, u_{2,t}, \ldots, u_{k,t})'$, correspond to a sample from the independence copula $C_\perp(\mathbf{u})$ (i.e., $C_\perp(u_1, u_2, \ldots, u_k) = u_1 u_2 \cdots u_k$) [101].

Building on this result, Genest et al. [101] devise two tests that are based on the empirical distribution function

$$D_{k,n}(\mathbf{u}) = \frac{1}{n}\sum_{t=1}^{n} \mathcal{I}(\mathbf{y}_t \le \mathbf{u})$$

associated with the pseudo-observations $\mathbf{y}_t,\ t = 1, 2, \ldots, n$. Any discrepancy between this empirical copula $(D_{k,n}(\mathbf{u}))$ and the independence copula $(C_\perp(\mathbf{u}))$ is interpreted as an indication of a lack of fit. The statistics of the resulting Cramér von Mises tests are as follows:

$$\begin{aligned} S_n^{(B)} &= n\int_{[0,1]^k} \left(D_{k,n}(\mathbf{u}) - C_\perp(\mathbf{u})\right)^2 d\mathbf{u}, \\ &= \frac{n}{3^k} - \frac{1}{2^{k-1}}\sum_{t=1}^{n}\prod_{i=1}^{k}(1 - y_{i,t}^2) \\ &\quad + \frac{1}{n}\sum_{s=1}^{n}\sum_{t=1}^{n}\prod_{i=1}^{k}(1 - \max(y_{i,s}, y_{i,t})). \\ S_n^{(C)} &= n\int_{[0,1]^k} \left(D_{k,n}(\mathbf{u}) - C_\perp(\mathbf{u})\right)^2 dD_{k,n}(\mathbf{u}), \\ &= \sum_{t=1}^{n} \left(D_{k,n}(\mathbf{y}_t) - C_\perp(\mathbf{y}_t)\right)^2. \end{aligned}$$

We refer the reader to Appendix D of Genest et al. [101] for a parametric bootstrap procedure using the test statistics $S_n^{(B)}$ and $S_n^{(C)}$ to measure the goodness of a copula fit.

A comprehensive experimental study conducted by Genest et al. [101] shows that the test statistics based on the Cramér–von Mises distance perform better than the test statistics based on the Kolmogorov–Smirnov distance; i.e., the test statistics S_n, $S_n^{(B)}$, and $S_n^{(C)}$ outperform the test statistic T_n. Furthermore, among the test statistics based on Cramér–von Mises distance, S_n and $S_n^{(B)}$ are the most promising tests for various copula models including the normal copula, the t copula, the Clayton copula, and the Gumbel copula. Recent applications of these test statistics to financial data can be found in [105].

6. Sampling

We discuss the well known methods of generating random vectors from multivariate normal and t copulas in Section 6.1, from Archimedean copulas via conditional sampling in Section 6.2, and from $\mathcal{C}$-vine and $\mathcal{D}$-vine specifications in Section 6.3.

6.1. Sampling from the normal copula and the t copula

The well known approach to the sampling of a uniform random vector $(U_1, U_2, \ldots, U_k)'$ from the k-dimensional normal copula with correlation matrix $\mathbf{\Sigma}_k$ is based on using the Cholesky decomposition for variate generation [106]. Specifically, we first find the Cholesky decomposition of the correlation matrix $\mathbf{\Sigma}_k$; i.e., $\mathbf{\Sigma}_k = \mathbf{A}\mathbf{A}'$. Then, we generate k independent standard normal random variates $z_i,\ i = 1, 2, \ldots, k$; construct vector $\mathbf{z} = (z_1, z_2, \ldots, z_k)'$; and set $\mathbf{x} = (x_1, x_2, \ldots, x_k)'$ to $\mathbf{Az}$. Finally, we apply the transformation $u_i = \Phi(x_i)$ for $i = 1, 2, \ldots, k$.

We can similarly sample a uniform random vector from the k-dimensional t copula with parameters $\mathbf{\Sigma}_k$ and d. First, we obtain the vector $\mathbf{x}$ as described above for the k-dimensional normal copula. Then, we generate a random variate s from a chi-square distribution with d degrees of freedom and set $\mathbf{m} = \sqrt{(d/s)}\mathbf{x}$. Finally, we obtain $u_i,\ i = 1, 2, \ldots, k$ from the transformation $t_d(m_i),\ i = 1, 2, \ldots, k$.

6.2. Sampling from the Archimedean copulas

The generation of a two-dimensional random vector of the form $(u_1, u_2)'$ from the copula C_2 starts with the sampling of two independent uniform random variates u_1 and w. Then, we obtain the uniform random variate u_2 from the quasi-inverse function of the conditional distribution $\Pr(U_2 \le u_2|U_1 = u_1)$; i.e., $u_2 = C_{2|1}^{-1}(w|u_1)$. Specifically, we obtain the conditional copula $C_{2|1}(u_2|u_1)$ from $\partial C_2(u_1, u_2)/\partial u_1$. In the case of a k-dimensional setting with $k \ge 3$, we similarly generate k independent uniform random variates, $u_1, w_1, w_2, \ldots, w_{k-1}$, and obtain the random variates $u_2, u_3, \ldots, u_k$ by

$$u_2 = C_{2|1}^{-1}(w_1|u_1),$$

$$u_3 = C_{3|1,2}^{-1}(w_2|u_1, u_2),$$

$$\vdots$$

$$u_k = C_{k|1,2,\ldots,k-1}^{-1}(w_{k-1}|u_1, u_2, \ldots, u_{k-1}),$$

where $C_{k|1,2,\ldots,k-1}(u_k|u_1, u_2, \ldots, u_{k-1}) = \Pr(U_k \le u_k|U_1 = u_1, U_2 = u_2, \ldots, U_{k-1} = u_{k-1})$ is given by

$$\frac{\left(\partial^{k-1}C_k\left(u_1, u_2, \ldots, u_k\right)\right) / \left(\partial u_1 \ldots \partial u_{k-1}\right)}{\left(\partial^{k-1}C_{k-1}\left(u_1, u_2, \ldots, u_{k-1}\right)\right) / \left(\partial u_1 \ldots \partial u_{k-1}\right)}.$$

The functional form of the conditional copula for the k-dimensional Archimedean copula reduces to the following representation, where $\phi^{-1(i)}(\cdot; \theta)$ stands for the ith-order derivative of the pseudo-inverse of the generator function $\phi(\cdot; \theta)$ [44]:

$$C_{i|1,2,\ldots,i-1}(u_i|u_1, u_2, \ldots, u_{i-1}; \theta) = \frac{\phi^{-1(i-1)}(\phi(u_1; \theta) + \phi(u_2; \theta) + \cdots + \phi(u_i; \theta); \theta)}{\phi^{-1(i-1)}(\phi(u_1; \theta) + \phi(u_2; \theta) + \cdots + \phi(u_{i-1}; \theta); \theta)},$$
$$i = 2, 3, \ldots, k.$$

Therefore, we can sample from the k-dimensional Clayton copula with

$$\phi^{-1(i)}(t; \theta) = (-1)^i \frac{(\theta+1)(\theta+2)\ldots(\theta+i-1)}{\theta^i}(t+1)^{-1/\theta-i}$$

as follows: First, we generate k independent, uniformly distributed random variates $w_i,\ i = 1, 2, \ldots, k$. Then, we set $u_1 = w_1$ and $w_2 = C_{2|1}(u_2|w_1)$ with $c_1 = \phi(u_1; \theta) = u_1^{-\theta} - 1$, $c_2 = \phi(u_1; \theta) + \phi(u_2; \theta) = u_1^{-\theta} + u_2^{-\theta} - 2$, and $C_{2|1}(u_2|w_1) = \phi^{-1(1)}(c_2)/\phi^{-1(1)}(c_1)$. Therefore, we obtain $w_2 = ((u_1^{-\theta} + u_2^{-\theta} - 1)/u_1^{-\theta})^{-1/\theta-1}$, which leads to $u_2 = ((w_1^{-\theta}(w_2^{-\theta/(\theta+1)} - 1)) + 1)^{-1/\theta}$. Similarly, we set $w_3 = C_{3|1,2}(u_3|u_1, u_2) = \phi^{-1(2)}(c_3)/\phi^{-1(2)}(c_2) = ((u_1^{-\theta} + u_2^{-\theta} + u_3^{-\theta} - 2)/(u_1^{-\theta} + u_2^{-\theta} - 1))^{-1/\theta-2}$, which results in

$$u_3 = \left(w_3^{-\theta/(1+2\theta)}\left(w_1^{-\theta} + w_2^{-\theta} - 1\right) - w_1^{-\theta} - w_2^{-\theta} + 2\right)^{-1/\theta}.$$

As a result of using this recursion, we obtain

$$w_k = C_{k|1,2,\ldots,k-1}(u_k|u_1, u_2, \ldots, u_{k-1}) = \left(\frac{u_1^{-\theta} + u_2^{-\theta} + \cdots + u_k^{-\theta} - k + 1}{u_1^{-\theta} + u_2^{-\theta} + \cdots + u_{k-1}^{-\theta} - k + 2}\right)^{-1/\theta-k+1}$$

and thus,

$$u_k = ((u_1^{-\theta} + u_2^{-\theta} + \cdots + u_{k-1}^{-\theta} - k + 2) \times (w_k^{\theta/(\theta(1-k)-1)} - 1) + 1)^{-1/\theta}.$$

The generation of k random variates from the k-dimensional Gumbel copula also starts with the sampling of k independent, uniformly distributed random variates $w_i,\ i = 1, 2, \ldots, k$. Then, we set $u_1 = w_1$ and $w_2 = C_{2|1}(u_2|w_1) = \phi^{-1(1)}(c_2)/\phi^{-1(1)}(c_1)$ with $c_1 = \phi(u_1; \theta) = (-\log(u_1))^\theta$ and $c_2 = \phi(u_1; \theta) + \phi(u_2; \theta) = (-\log(u_1))^\theta + (-\log(u_2))^\theta$, and solve the resulting equation for u_2. We obtain $u_3, u_4, \ldots, u_k$ in a similar manner by setting $w_i = C_{i|1,2,\ldots,i-1}(u_i|u_1, u_2, \ldots, u_{i-1}) = \phi^{-1(i-1)}(c_i; \theta)/\phi^{-1(i-1)}(c_{i-1})$ and solving this equation for $u_i,\ i = 3, 4, \ldots, k$. Thus, unlike the Clayton copula, there is no recursive formula for the inverse of the generator function of the k-dimensional Gumbel copula. Nevertheless, we can easily obtain $u_1, u_2, \ldots, u_k$ via the use of numerical search procedures.

Another way of sampling a k-dimensional Gumbel copula is to use the Marshall and Olkin generation method [107] with the $(1/\theta)$-stable random variable γ and the Laplace transform $\Theta(\gamma) = \exp(-\gamma^{1/\theta})$[4]. Specifically, we first generate a random variable γ from the Stable (1, 0, 0) distribution with parameter $1/\theta$. We do this by sampling a uniform random variable r on the interval $(-\pi/2, \pi/2)$ and an exponential random variable ξ with mean 1, and setting

$$\gamma = \frac{\sin(r/\theta)}{(\cos(r))^\theta}\left[\frac{\cos((1-1/\theta)r)}{\xi}\right]^{(1-1/\theta)/(1/\theta)}.$$

Then, independent of the previous step, we generate k independent uniform random variates $w_i,\ i = 1, 2, \ldots, k$ and apply the transformation $u_i = \Theta(-\log(w_i)/\gamma)$ for $i = 1, 2, \ldots, k$. Similarly, the Marshall and Olkin generation method can be used for sampling from a k-dimensional Clayton copula with γ defined as a gamma random variable having shape parameter 1 and scale parameter $1/\theta$ and the Laplace transform $\Theta(\gamma)$ defined by $(1+\gamma)^{-1/\theta}$ [108].

6.3. Sampling from the $\mathcal{C}$-vine and $\mathcal{D}$-vine specifications

The sampling of a random vector with a copula-vine specification starts with the generation of k independent uniform random variates $w_i,\ i = 1, 2, \ldots, k$; proceeds by traversing the regular vine of interest in a specific order; and applying successive inversions of the conditional distributions derived from the two-dimensional copulas of each edge. Specifically, the sampling algorithm associated with a $\mathcal{C}$-vine specification is based on the graphical representation of the k-dimensional distribution with

(conditional) tail dependencies assigned to the following edges of the vine:

$1, 2$	$1, 3$	$1, 4$	$\cdots$	$1, k$
	$2, 3\|1$	$2, 4\|1$	$\cdots$	$2, k\|1$
		$3, 4\|1, 2$	$\cdots$	$3, k\|1, 2$
			$\ddots$	$\vdots$
				$k-1, k\|1, 2, \ldots, k-2$

In particular, "$1, k$" refers to the edge between the (unconditional) random variables X_1 and X_k, while "$k-1, k|1, 2, \ldots, k-2$" represents the edge associated with the conditional random variables $X_{k-1}|X_1, X_2, \ldots, X_{k-2}$ and $X_k|X_1, X_2, \ldots, X_{k-2}$. Thus, we sample the uniform random vector $(U_1, U_2, \ldots, U_k)'$ with a k-dimensional distribution function represented by a $\mathcal{C}$-vine as follows:

$$
\begin{aligned}
u_1 &\leftarrow w_1.\\
u_2 &\leftarrow C_{2|1}^{-1}\left(w_2|u_1\right).\\
u_3 &\leftarrow C_{3|1}^{-1}\left(C_{3|2:1}^{-1}\left(w_3|u_2\right)|u_1\right).\\
u_4 &\leftarrow C_{4|1}^{-1}\left(C_{4|2:1}^{-1}\left(C_{4|3:1,2}^{-1}\left(w_4|u_3\right)|u_2\right)|u_1\right).\\
&\vdots\\
u_k &\leftarrow C_{k|1}^{-1}(C_{k|2:1}^{-1}(C_{k|3:1,2}^{-1}(\cdots(C_{k|k-1:1,2,\ldots,k-2}^{-1}\\
&\quad (w_k|u_{k-1})|u_{k-2})\cdots)|u_3)|u_2)|u_1).
\end{aligned}
$$

In this sampling algorithm, $C_{i|j}$ is the conditional distribution associated with the bivariate copula between random variables U_i and U_j (i.e., $C_{i|j}(u_i|u_j) = \partial C_2(u_i, u_j)/\partial u_j$), while $C_{i|j:1,2,\ldots,j-1}$ is the conditional distribution associated with the conditional copula $C_{i,j|1,2,\ldots,j-1}$ between the random variables $U_i|U_1, U_2, \ldots, U_{j-1}$ and $U_j|U_1, U_2, \ldots, U_{j-1}$; i.e.,

$$C_{i|j:1,2,\ldots,j-1}(u_i|u_j) = \frac{\partial C_2(u_i, u_j|u_1, u_2, \ldots, u_{j-1})}{\partial u_j}.$$

The uniform random variates $w_1, w_2, \ldots, w_k$ are generated independently and $u_1, u_2, \ldots, u_k$ are obtained by applying successive inverse cdfs. Furthermore, the first four rows of this sampling algorithm describe the generation of random variates u_1, u_2, u_3, and u_4 from the four-dimensional $\mathcal{C}$-vine illustrated in Fig. 4.

When the joint distribution function of the random variables $U_1, U_2, \ldots, U_k$ is specified by a $\mathcal{D}$-vine, which is illustrated in Fig. 3 for a four-dimensional random vector, the graphical representation is obtained in terms of the dependence measures associated with the following edges of the vine:

$1, 2$	$1, 3\|2$	$1, 4\|2, 3$	$\cdots$	$1, k\|2, \ldots, k-1$
	$2, 3$	$2, 4\|3$	$\cdots$	$2, k\|3, \ldots, k-1$
		$3, 4$	$\cdots$	$3, k\|4, \ldots, k-1$
			$\ddots$	$\vdots$
				$k-2, k\|k-1$
				$k-1, k$

Therefore, we first generate k independent uniform random variates $w_1, w_2, \ldots, w_k$ and then sample the uniform random vector $(U_1, U_2, \ldots, U_k)'$ represented with the $\mathcal{D}$-vine by calculating

$$
\begin{aligned}
u_1 &\leftarrow w_1,\\
u_2 &\leftarrow C_{2|1}^{-1}\left(w_2|u_1\right),\\
u_3 &\leftarrow C_{3|2}^{-1}\left(C_{3|1,2}^{-1}\left(w_3|C_{1|2}(u_1|u_2)\right)|u_2\right),\\
u_4 &\leftarrow C_{4|3}^{-1}\left(C_{4|2,3}^{-1}\left(C_{4|1,2,3}^{-1}\left(w_4|C_{1|2,3}(u_1|u_2, u_3)\right)|C_{2|3}(u_2|u_3)\right)|u_3\right),\\
&\vdots\\
u_k &\leftarrow C_{k|k-1}^{-1}(C_{k|k-2,k-1}^{-1}(\cdots C_{k|1,2,\ldots,k-1}^{-1}(w_k|C_{1|2,3,\ldots,k-1}\\
&\quad (u_1|u_2, u_3, \ldots, u_{k-1}))\cdots|C_{k-2|k-1}(u_{k-2}|u_{k-1}))|u_{k-1}).
\end{aligned}
$$

Thus, the sampling procedure for $\mathcal{D}$-vine uses both conditional distributions and inverse conditional distributions. The first four rows of this sampling procedure further describe the sampling of a uniform random vector from the four-dimensional $\mathcal{D}$-vine in Fig. 3.

7. Conclusion

As large-scale stochastic simulation becomes a tool that is used routinely for the design and analysis of complex systems, it is important to develop multivariate input models that are flexible enough to capture the joint distributional properties of the system inputs. A close look at the existing literature shows that the development of a large number of multivariate input models builds on the use of the normal copula for dependence modeling. However, the normal copula fails to represent the dependence structures with tail dependencies that arise in the context of extreme events.

In this survey, we reviewed the copula-based input models for stochastic simulations with dependent inputs by focusing on the tail dependence. First, we reviewed the bivariate copula models along with their tail dependence properties and then extended our discussion to the multivariate copula models with three or more component random variables. Finally, we discussed how to fit these copula models to the available historical data sets, and how to generate random vectors from the resulting joint distributions.

The case of bivariate copulas with tail-dependence power has been well studied, but these do not readily extend to multiple dimensions. Recently, several multivariate parametric copulas have been introduced, but the study of their tail-dependence properties is still in its infancy. We believe that the study of the tail-dependence characteristics of the existing multivariate copulas as well as the development of new multivariate copulas that have the ability to capture a wide variety of asymmetric dependence structures are the promising research areas to enhance the field of multivariate input modeling for stochastic simulations.

References

[1] S. Vincent, Input data analysis, in: J. Banks (Ed.), Handbook of Simulation, John Wiley & Sons, New York, 1998, pp. 55–91.
[2] A.M. Law, Simulation Modeling and Analysis, fourth ed., McGraw-Hill, New York, 2007.
[3] M.E. Kuhl, J.S. Ivy, E.K. Lada, N.M. Steiger, M.A. Wagner, J.R. Wilson, Univariate input models for stochastic simulation, Journal of Simulation 4 (2010) 81–97.
[4] S.H. Xu, Structural analysis of a queuing system with multi-classes of correlated arrivals and blocking, Operations Research 47 (1999) 264–276.
[5] M. Livny, B. Melamed, A.K. Tsiolis, The impact of autocorrelation on queuing systems, Management Science 39 (1993) 322–339.
[6] P.P. Ware, T.W. Page, B.L. Nelson, Automatic modeling of file system workloads using two-level arrival processes, ACM Transactions on Modeling and Computer Simulation 8 (1998) 305–330.
[7] B. Biller, I. Civelek, A. Scheller-Wolf, Impact of dependence on single-server queueing systems, Tepper Working Paper 2010-E18, Carnegie Mellon University, Pittsburgh, PA, 2010.
[8] A.A.B. Pritsker, D.L. Martin, J.S. Reust, M.A.F. Wagner, J.R. Wilson, M.E. Kuhl, J.P. Roberts, O.P. Daily, A.M. Harper, E.B. Edwards, L. Bennett, J.F. Burdick, M.D. Allen, Organ transplantation policy evaluation, in: C. Alexopoulos, K. Kang, W. R. Lilegdon, D. Goldsman (Eds.), Proceedings of the 1995 Winter Simulation Conference, 1995, pp. 1314–1323.
[9] S. Ghosh, M.S. Squillante, Analysis and control of web server queues, Computer Communications 27 (2004) 1771–1785.
[10] A.N. Avramidis, A. Deslauriers, P. L'Ecuyer, Modeling daily arrivals to a telephone call center, Management Science 50 (2004) 896–908.

[11] P. Kouvelis, P. Su, The Structure of Global Supply Chains, in: Research Monograph for Book Series on Foundations and Trends® in Technology, Information, and Operations Management, NOW, Boston, Delft, 2007.
[12] B. Biller, S. Ghosh, Multivariate input processes, in: B.L. Nelson, S.G. Henderson (Eds.), Handbooks in Operations Research and Management Science: Simulation, Elsevier Science, Amsterdam, 2006, pp. 123–153.
[13] A.J. Patton, Modeling asymmetric exchange rate dependence, International Economic Review 47 (2005) 527–556.
[14] C.J. Corbett, K. Rajaram, A generalization of the inventory pooling effect to nonnormal dependent demand, Manufacturing and Service Operations Management 8 (2006) 351–358.
[15] S.M. Wagner, C. Bode, P. Koziol, Supplier default dependencies: empirical evidence from the automotive industry, European Journal of Operational Research 199 (2009) 150–161.
[16] B.M. Tehrani, S.H. Xu, S. Kumara, H. Li, A single-period analysis of a two-echelon inventory system with dependent supply uncertainty, Transportation Research Part B: Methodological 45 (2011) 1128–1151.
[17] T.A. Craney, N. White, Distribution selection with no data using VBA and excel, Quality Engineering 16 (2004) 643–656.
[18] R.B. Nelsen, An Introduction to Copulas, second ed., Springer-Verlag, New York, 2006.
[19] K. Jong-Dev, Measures of dependence, in: P.R. Krishnaiah, P.K. Sen (Eds.), Handbook of Statistics, Elsevier Science, North-Holland, Amsterdam, 1984, pp. 79–88.
[20] G. Yule, M.G. Kendall, An Introduction to the Theory of Statistics, Griffin, London, 1965.
[21] W. Whitt, Bivariate distributions with given marginals, The Annals of Statistics 4 (1976) 1280–1289.
[22] H. Hotelling, M.R. Pabst, Rank correlation and tests of significance involving no assumption of normality, The Annals of Mathematical Statistics 7 (1936) 29–43.
[23] P. Embrechts, A. McNeil, D. Straumann, Correlation and dependence in risk management: properties and pitfalls, in: M.A.H. Dempster (Ed.), Risk Management: Value at Risk and Beyond, Cambridge University Press, Cambridge, UK, 2002, pp. 176–223.
[24] W. Hoeffding, Masstabinvariante korrelations theorie, Schrijtfen des Mathematischen Instituts und des Instituts fr Angewandte Mathematik det Universitt Berlin 5 (1940) 179–233.
[25] P.J. Danaher, M.S. Smith, Modeling multivariate distributions using copulas: applications in marketing, Marketing Science 30 (2011) 4–21.
[26] H. Joe, Multivariate Models and Dependence Concepts, Chapman and Hall, London, 1997.
[27] B. Biller, Copula-based multivariate input models for stochastic simulation, Operations Research 57 (2009) 878–892.
[28] R.D. Cook, M.E. Johnson, A family of distributions for modeling non-elliptically symmetric multivariate data, Journal of the Royal Statistical Society. Series B 43 (1981) 210–218.
[29] R.D. Cook, M.E. Johnson, Generalized Burr–Pareto-logistic distributions with applications to an uranium exploration data set, Technometrics 28 (1986) 123–131.
[30] Y.L. Tong, The Multivariate Normal Distribution, Springer-Verlag, New Jersey, 1990.
[31] N.L. Johnson, Bivariate distributions based on simple translation systems, Biometrika 36 (1949) 297–304.
[32] M.C. Cario, B.L. Nelson, Modeling and generating random vectors with arbitrary marginal distributions and correlation matrix, Working Paper, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, 1997.
[33] I. Fortin, C. Kuzmics, Tail-dependence in stock-return pairs, International Journal of Intelligent Systems in Accounting, Finance and Management 11 (2002) 89–107.
[34] A. Ang, J. Chen, Asymmetric correlations of equity portfolios, Journal of Financial Economics 63 (2002) 443–494.
[35] A. Ang, G. Bekaert, International asset allocation with regime shifts, Review of Financial Studies 15 (2002) 1137–1187.
[36] K.-H. Bae, G.A. Karolyi, R.M. Stulz, A new approach to measuring financial contagion, Review of Financial Studies 16 (2003) 717–764.
[37] M.-H. Zhang, Modeling total tail dependence along diagonals, Insurance: Mathematics and Economics 42 (2008) 73–80.
[38] Y. Sun, X. Wu, A nonparametric study of dependence between S & P 500 index and market volatility index (VIX), Working Paper, 2009.
[39] U. Cherubini, E. Luciano, Value-at-risk trade-off and capital allocation with copulas, Economic Notes 30 (2001) 235–256.
[40] P. Embrechts, F. Lindskog, A. McNeil, Modeling dependence with copulas and applications to risk management, in: S.T. Rachev (Ed.), Handbook of Heavy Tailed Distributions in Finance, Elsevier, North-Holland, 2003.
[41] P. Embrechts, A. Höing, Extreme VaR scenarios in higher dimensions, Extremes 9 (2006) 177–192.
[42] J.V. Rosenberg, Nonparametric pricing of multivariate contingent claims, Journal of Derivatives 10 (2003) 9–26.
[43] M.N. Bennett, J.E. Kennedy, Quanto pricing with copulas, Journal of Derivatives 12 (2004) 26–45.
[44] U. Cherubini, E. Luciano, W. Vecchiato, Copula Methods in Finance, John Wiley & Sons, Chichester, UK, 2004.
[45] R.W.J. van den Goorbergh, C. Genest, B.J.M. Werker, Multivariate option pricing using dynamic copula models, Insurance: Mathematics and Economics 37 (2005) 101–114.
[46] M. Salmon, C. Schleicher, Pricing multivariate currency options with copulas, in: J. Rank (Ed.), Copulas: From Theory to Application in Finance, Risk Books, London, 2006.
[47] D.X. Li, On default correlation: a copula function approach, Journal of Fixed Income 9 (2000) 43–54.
[48] K. Giesecke, Correlated default with incomplete information, Journal of Banking and Finance 28 (2004) 1521–1545.
[49] A.J. Patton, On the out-of-sample importance of skewness and asymmetric dependence for asset allocation, Journal of Financial Econometrics 2 (2004) 130–168.
[50] R. Garcia, G. Tsafack, Dependence structure and extreme co-movements in international equity and bond markets, Journal of Banking and Finance 35 (2011) 1954–1970.
[51] A.J. Patton, Copula-based models for financial time series, in: Handbook of Financial Time Series, Springer-Verlag, 2009, pp. 767–781 (Chapter V-3).
[52] R.T. Clemen, T. Reilly, Correlations and copulas for decision and risk analysis, Management Science 45 (1999) 208–224.
[53] M.N. Jouini, R.T. Clemen, Copula models for aggregating expert opinions, Operations Research 44 (1996) 444–457.
[54] D. Vedenov, Application of copulas to estimation of joint crop yield distributions, in: Proceedings of the American Agricultural Economics Association Annual Meeting, Orlando, FL, 2008.
[55] J. Palomo, D. Rios Insua, F. Ruggeri, Modeling external risks in project management, Risk Analysis 27 (2007) 961–978.
[56] W. Yi, V.W. Bier, An application of copulas to accident precursor analysis, Management Science 44 (1998) 257–270.
[57] A. Meel, W.D. Seider, Real-time risk analysis of safety systems, Computers and Chemical Engineering 32 (2008) 827–840.
[58] C.R. Bhat, N. Eluru, A copula-based approach to accommodate residential self-selection effects in travel behavior modeling, Transportation Research Part B: Methodological 43 (2009) 749–765.
[59] K. Takahasi, Note on the multivariate Burr's distribution, Annals of the Institute of Statistical Mathematics 17 (1965) 257–260.
[60] N.L. Johnson, S. Kotz, Distributions in Statistics: Continuous Multivariate Distributions, John Wiley & Sons, New York, 1972.
[61] S.P. Satterthwaite, T.P. Hutchinson, A generalization of Gumbel's bivariate logistic distribution, Metrika 25 (1978) 163–170.
[62] R.L. Plackett, A class of bivariate distributions, Journal of American Statistical Association 60 (1965) 516–522.
[63] D. Morgenstern, Einfache beispiele zweidimensionaler verteilungen, Mitteilingsblatt fr Mathematishe Statistik 8 (1956) 234–235.
[64] N.L. Johnson, S. Kotz, On some generalized Farlie–Gumbel–Morgenstern distributions, Communications in Statistics 4 (1975) 415–427.
[65] M.E. Johnson, Multivariate Statistical Simulation, Wiley, New York, 1987.
[66] C. Genest, R.J. MacKay, The joy of copulas: bivariate distributions with uniform marginals, The American Statistician (American Statistical Association) 40 (1986) 280–283.
[67] B. Schweizer, Thirty years of copulas, in: G. Dall'Aglio, S. Kotz, G. Salinetti (Eds.), Advances in Probability Distributions with Given Marginals, Kluwer, Dordrecht, Netherlands, 1991, pp. 13–50.
[68] P. Trivedi, D. Zimmer, Copula modeling: an introduction for practitioners, Foundations and Trends in Econometrics 1 (2005) 1–111.
[69] N.L. Johnson, Systems of frequency curves generated by methods of translation, Biometrika 36 (1949) 149–176.
[70] D. Kelker, Distribution theory of spherical distributions and location-scale parameter generalization, Sankhyā 32 (1970) 419–430.
[71] K. Fang, S. Kotz, K.W. Ng, Symmetric Multivariate and Related Distributions, Chapman & Hall, London, 1987.
[72] G. Kimeldorf, A.R. Sampson, Uniform representations of bivariate distributions, Communications in Statistics 4 (1975) 617–627.
[73] E.J. Gumbel, Distributions des valeurs extrêmes en plusiers dimensions, Publications de LiInstitute de Statistique 9 (1960) 171–173.
[74] A.A. Balkema, S.I. Resnick, Max-infinite divisibility, Journal of Applied Probability 14 (1977) 309–319.
[75] D. MacKenzie, End-of-the-world trade, in: London Review of Books, Vol. 30, 2008, pp. 24–26.
[76] F. Salmon, Recipe for disaster: the formula that killed Wall Street, 2009. Wired 02/23/2009.
[77] A. Lipton, A. Rennie, Credit Correlation: Life After Copulas, World Scientific Publishing Co., Hackensack, NJ, 2007.
[78] C. Donnelly, P. Embrechts, The devil in the tails: actuarial mathematics and the subprime mortgage crisis, ASTIN Bulletin 40 (2010) 1–33.
[79] D. Brigo, A. Pallavicini, R. Torresetti, Credit Models and the Crisis: A Journey into CDOs, Copulas, Correlations, and Dynamic Models, in: The Wiley Finance Series, 2010.
[80] E. Gomeza, M.A. Gomez-Viilegasa, J.M. Marina, A multivariate generalization of the power exponential family of distributions, Communications in Statistics: Theory and Methods 27 (1998) 589–600.
[81] Z.M. Landsman, E.A. Valdez, Tail conditional expectations for elliptical distributions, Technical Report N 02-04, 2002.
[82] B.V.M. Mendes, O. Arslan, Multivariate skew distributions based on the GT copula, Brazilian Review of Econometrics 26 (2006) 235–255.
[83] C.H. Kimberling, A probabilistic interpretation of complete monotonicity, Aequationes Mathematicae 10 (1974) 152–164.
[84] J. Galambos, Order statistics of samples from multivariate distributions, Journal of American Statistical Association 70 (1975) 674–680.

[85] R. Cooke, Markov and entropy properties of tree and vines-dependent variables, in: Proceedings of the ASA Section on Bayesian Statistical Science, American Statistical Association, Alexanria, VA, 1997, pp. 166–175.
[86] T. Bedford, R.M. Cooke, Probability density decomposition for conditionally dependent random variables modeled by vines, Annals of Mathematics and Artificial Intelligence 32 (2001) 245–268.
[87] T. Bedford, R.M. Cooke, Vines—a new graphical model for dependent random variables, The Annals of Statistics 30 (2002) 1031–1068.
[88] D. Kurowicka, R. Cooke, A parameterization of positive definite matrices in terms of partial correlation vines, Linear Algebra and its Applications 372 (2003) 225–251.
[89] K. Aas, C. Czado, A. Frigessi, H. Bakken, Pair-copula constructions of multiple dependence, Insurance: Mathematics and Economics 44 (2009) 182–198.
[90] D. Kurowicka, R. Cooke, Uncertainty Analysis with High Dimensional Dependence Modeling, in: Wiley Series in Probability and Statistics, 2006.
[91] P.A. Maugis, D. Guegan, New prospects on vines, Insurance Markets and Companies: Analysis and Actuarial Computations 1 (2010) 15–22.
[92] H. Joe, L. Haijun, A.K. Nikoloulopoulos, Tail dependence functions and vine copulas, Journal of Multivariate Analysis 101 (2010) 252–270.
[93] R.J. Serfling, Approximation Theorems of Mathematical Statistics, Wiley, New York, 1980.
[94] P. Billingsley, Statistical Inference for Markov Processes, University of Chicago Press, Chicago, 1961.
[95] G. Kim, M.J. Silvapulle, P. Silvapulle, Comparison of semi-parametric and parametric methods for estimating copulas, Communications in Statistics: Simulation and Computation 51 (2007) 2836–2850.
[96] L. Dalla Valle, Bayesian copulae distributions, with application to operational risk management, Methodology and Computing in Applied Probability 11 (2009) 95–115.
[97] P.E. Rossi, G.M. Allenby, R. McCulloch, Bayesian Statistics and Marketing, in: Wiley Series in Probability and Statistics, 2006.
[98] W.R. Gilks, S. Richardson, D.J. Spiegelhalter, Markov Chain Monte Carlo in Practice, Chapman and Hall, London, UK, 1996.
[99] C. Czado, A. Min, Bayesian inference for $\mathcal{D}$-vines: estimation and model selection, in: D. Kurowicka, H. Joe (Eds.), Dependence Modeling: Vine Copula Handbook, World Scientific Publishing Co., Singapore, 2011, pp. 249–264.
[100] D. Berg, Copula goodness-of-fit testing: an overview and power comparison, The European Journal of Finance 15 (2009) 675–701.
[101] C. Genest, B. Rémillard, D. Beaudoin, Goodness-of-fit tests for copulas: a review and a power study, Insurance: Mathematics and Economics 44 (2009) 199–213.
[102] P. Deheuvels, La fonction de dépendance empirique et ses propriétés: Un test non paramétrique d'imdépendance, Académie Royale de Belgique: Bulletin de la Classe des Sciences, 5e Série 65 (1979) 274–292.
[103] C. Genest, B. Rémillard, Validity of parametric bootstrap for goodness-of-fit testing in semiparametric models, Annals of Henri Poincaré 44 (2008) 1096–1127.
[104] M. Rosenblatt, Remarks on a multivariate transformation, The Annals of Mathematical Statistics 23 (1952) 470–472.
[105] G.N.F. Weiss, Are Copula-GoF tests of any practical use? empirical evidence for stocks, commodities, and FX futures, The Quarterly Review of Economics and Finance 51 (2011) 173–188.
[106] E.M. Scheuer, D.S. Stoller, On the generation of normal random vectors, Technometrics 4 (1962) 278–281.
[107] A.W. Marshall, I. Olkin, Families of multivariate distributions, Journal of American Statistical Association 83 (1988) 834–841.
[108] E.W. Frees, E. Valdez, Understanding relationships using copulas, North American Actuarial Journal 2 (1998) 1–25.

Surveys in Operations Research and Management Science 17 (2012) 85–96

Contents lists available at SciVerse ScienceDirect

Surveys in Operations Research and Management Science

journal homepage: www.elsevier.com/locate/sorms

Review

A tutorial on fundamental model structures for railway timetable optimization

Steven S. Harrod *

University of Dayton, United States

ARTICLE INFO

Article history:
Received 22 June 2012
Accepted 8 August 2012

ABSTRACT

This guide explains the role of railway timetables relative to all other railway scheduling activities, and then presents four fundamental timetable formulations suitable for optimization. Timetabling models may be classified according to whether they explicitly model the track structure, and whether the timetable is intended to be periodic or not (aperiodic). The presentation of models is organized to facilitate the selection of a model by planning objective and available data, regardless of the specific traffic carried or network size.

Contents

1. Introduction

Railway operations involve large sums of money both in infrastructure and direct operating expenses, and their services are valued by both the traveling public and primary industries. A fully defined timetable specifies the paths that trains follow on a railway network including track lines used, junctions or stations traveled, connections between trains for passengers or freight, and various interactions between trains necessary for safe operation, with planned timings for all events. The quality of a timetable determines the utilization of the railway network, the sustainable flow, and the robustness of the service commitments to passengers and freight recipients. For example, Netherlands Railways was the subject of the 2008 Franz Edelman Award (INFORMS), for which they documented a profit increase of € 40 million annually due to improvements to a timetable of 5500 trains [1].

Operation by timetable is frequently confused with the North American term "scheduled railroading", but North American railroads have not compiled and followed timetables for over fifty years. Instead, they attempt to maintain a homogeneous flow along their networks and rely on experienced human dispatchers to issue orders granting authority for moves directly to train crews. Freight train timekeeping records are measured in hours. This practice, "timetable free" operation, is the result of a legacy of "tonnage" dispatching, where trains only departed when they reached full length or tonnage. This in turn was the result of the elimination of most North American passenger services by 1960 and the simultaneous loss of priority or perishable freight traffic to road carriage. Although increasing network congestion, introduction of intermodal services, and re-introduction of passenger services are apparent throughout North America, for the moment timetable free operation remains standard practice.

Timetable optimization formulations are commonly labeled according to their application: passenger or freight, single or double track, and main lines or junctions. However, frequently the same mathematical structures and capabilities can be found spread amongst these different applications. This leads to a lack of continuity between these conceptual developments and sometimes a lack of recognition as well. From the user's point

* Tel.: +1 937 312 9252.
E-mail address: steven.harrod@udayton.edu.

1876-7354/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.sorms.2012.08.002

of view, the originating application is irrelevant so long as the formulation supports the objectives at hand, the available data, and the available computing resources.

This paper addresses timetabling models by their structure and capabilities rather than their prior application. The primary options to consider in selecting a timetabling formulation are whether it explicitly models the track structure, and whether the timetable is intended to be periodic or aperiodic (without regularly sequenced repeating train paths). The four models discussed in this paper are aligned by these characteristics in Fig. 1. Timetables in Europe are frequently periodic. For example, trains leave Manchester for Birmingham, England at seven and twenty-seven minutes past the hour daily, and an ICE train leaves Munich for Nurnberg at sixteen minutes past the hour daily. This means that care is taken to structure the timetable so that passengers may expect a specific service to depart at the same time each hour or half hour. In the United States, intercity or suburban rail passenger services are not offered with periodic timetables, with few exceptions.

All models, regardless of their application, can be segregated according to whether or not they explicitly represent the limitations of the track network. Many managerial questions concern the economic value of a segment of track, or seek to prioritize a limited budget for track investment, and these questions are more easily answered when the track is explicitly considered in the model. The lack of explicit track representation also limits the ability of the model to estimate or forecast line capacity, where capacity is the volume of train paths supported. This is not to be confused with the alternate measure of capacity used in some circles, where capacity is measured as a function of experienced delay or the makespan of trip times. Examples of this measure can be found in the United States, where network performance is frequently measured by the sum of hours of train waiting time [2]. Models that do not explicitly represent the track structure typically require that the initial problem data set supports at least one feasible timetable containing all trains in the data set, whereas models that consider the underlying track resource may return solutions of some subset of the initial train requests.

Note the lack of a entry in Fig. 1 at the intersection of periodic and explicit track features. To date there are no efficient methods of providing both of these features in the same model. It is possible to dimension various aperiodic mixed-integer program models with time scales that "wrap around" to form a periodic decision space [3,4]. Harrod finds that similar problems are more difficult to optimize in their periodic form than in their aperiodic form.

Many of these models may be extended to consider timetable robustness (stability under stochastic delays or incidents), for example using stochastic programming. This tutorial is limited to deterministic timetable optimization, but some brief references are offered here. Kroon et al. [5] determines an optimal periodic timetable from a large sample of stochastic train operation realizations. The objective minimizes a reduced set of "primary disturbances". Liebchen et al. [6] constrains the feasible range of decision variables to attain a more robust timetable solution in a single optimization step. A significant body of literature describes post analysis of timetable stability, and max-plus system theory is a good starting point [7]. Goerigk and Schöbel [8] consider what timetable robustness expectations are reasonable under two distributions of network delay and four network delay response policies.

1.1. Taxonomy of railway scheduling

Scheduling activities occur at all levels of railway management. At the strategic level, scheduling may determine the frequency of train operations or the origins and destinations served.

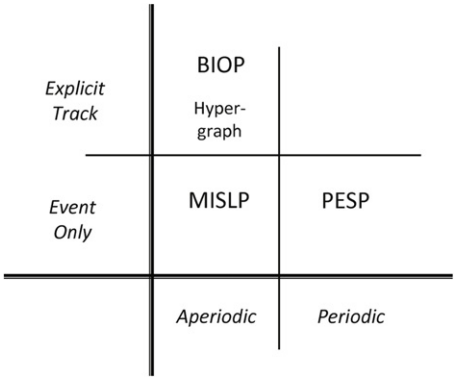


Fig. 1. Model feature distribution.

Interconnections are also a strategic scheduling task. In the case of carload freight, "blocking" is the strategy of grouping cars together to minimize the coupling and uncoupling of individual cars at yards (which may then interact with the choice of route and accumulated distance). For passengers, strategic scheduling determines connections between trains at stations for both variety in destinations served and passenger convenience. All of these activities are frequently collected under the terms "service design", "network design", or "strategic operating plan" [9]. The planning of locomotives, rolling stock, and crews also frequently appears under the heading "scheduling" [10].

Timetables are a tactical scheduling activity. As previously described, timetables determine the timings of trains at stations or signal control points. They ensure that a train which departs as scheduled will have a contiguous, conflict-free path to its destination. Conflicts may include trains moving in the opposing direction, slower trains in the same direction blocking the path, or tracks out of service due to maintenance. An iterative cycle may occur between the tactical and strategic scheduling activities. For example, a desired service design may be infeasible at the tactical level, and require either compromises in the service design or changes to the infrastructure. Andersen [11] describes how the Swiss Bahn 2000 service design required changes to the track network in order to attain desired periodic timetables.

Operational scheduling activities include live dispatching of established timetables, and network recovery from delays or incidents. Recovery can take the form of attempting to return trains to their original timetables, or generating new temporary timetables for the remainder of their journeys. There is no conceptual difference between operational and tactical scheduling of trains. Only the solution quality expectations and available processing time differ. Tactical timetable planning seeks a high quality solution and hours or days of processing time are acceptable. On the other hand, operational schedules must be determined in minutes, sometimes seconds, and the first feasible schedule returned may be acceptable.

1.2. Prior references on railway scheduling

A number of prior surveys can be recommended for further reading in railway scheduling research. Lusby et al. [12] presents a comprehensive technical reference to timetabling models organized by track structure (single track, double track, or station tracks). Caprara et al. [13] surveys passenger railway topics in Europe, but specifically excludes freight service topics. Kroon et al. [14] presents a detailed survey and exposition of periodic timetabling, dominated by European literature, with particular emphasis on the periodic event scheduling problem (PESP). Törnquist [15] reviews a sample of 48 timetabling and dispatching papers from both North American and European theaters over

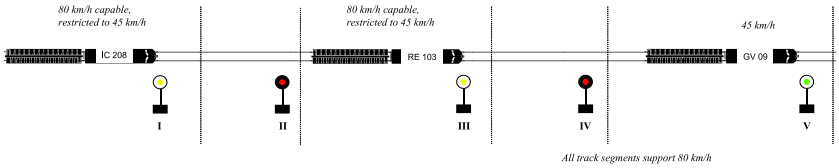


Fig. 2. Train headway times defined by leading train.

the period 1973 to 2005, and includes detailed comparisons by network constraints, objective, solution method, and problem size.

The most recent comprehensive survey of the North American operating plan literature, [16], encompasses many applications of optimization including blocking, scheduling, and locomotive assignment. The research cited primarily represents the advancement and application of mathematical programming. In contrast, Leilich [17] focuses solely on simulation applications in North America, with many line capacity examples. An additional survey of commercial train performance and dispatch simulators may be found in [18]. Bussieck et al. [19] surveys primarily European passenger railway scheduling problems, with an emphasis on periodic scheduling and applications of the then very new formulation of PESP. The introduction of Jovanović and Harker [20] includes a lengthy review of computer applications in North American dispatching, notes the objective functions devised in each experience, and discusses the components that should be considered in formulating an objective. A well known paper, Assad [21] presents a comprehensive review of North American research prior to 1979, with especially rich coverage of the 1960s, including the topics of yard (marshaling) operations, blocking, timetables, and dispatching. Finally, Yabe [22] presents an introduction to the relatively unknown application of operations research to railway management in Japan from the 1940s on.

1.3. Primary model classifications and their origins

This paper groups timetable generation models into four fundamental structures. Only formulations that offer the possibility of attaining an optimal objective value as a function of the solved timetable are considered. Published models which do not formulate a mathematical program and relevant objective, or which cannot be solved to optimality on at least small problems, are not discussed.

The oldest model of interest is the *Mixed Integer Sequencing Linear Program* (MISLP), discussed in Section 2. This formulation configures a decision space composed of binary variables that establish the sequencing of pairs of trains, and real valued variables that specify train timing at control points. It originates from sponsored research at the Wharton School, University of Pennsylvania in the late 1980s [23]. The sponsor was Burlington Northern Railroad, at the time a major U.S. railroad operating in the region bounded by Chicago, Dallas, and Seattle. The scheduling research was intended to support a very early form of positive train control, the Advanced Railroad Electronics System (ARES). However, Burlington Northern's interest in advanced scheduling waned, partly because the model's solution time of one hour on a 17.9 MFlop computer was impractical for field applications, and the project was canceled.

The next class of model is the *Binary Integer Occupancy Program* (BIOP), discussed in Section 3. The focus of this formulation is the explicit modeling of the track structure, through binary variables which represent the assigned occupation of a controlled track segment during a discrete time interval. This model originates from [24], where it addresses the timetabling of heavy haul ore trains on single track lines of the Pilbara region of Australia. Mees was unable to resolve commercially relevant problems under its specific formulation and available computing power, but later derivations from other authors were capable of solving larger problems.

The *Hypergraph Formulation* of Section 4 addresses a critical omission in the BIOP structure by mapping binary decision variables to contiguous pairs of track occupations. First appearing in [25], the model is applied to the calculation of the absolute capacity limit of a single track line traveled by trains of different speeds. This study derives a complete timetable for a route without any presumption of specific dispatch technique or artifact of an initial feasible solution. The timetable is optimal with respect to the defined objective.

The last model in this classification paradigm, the *Periodic Event Scheduling Program* of Section 5 supports the specialized needs of railway networks that operate a repeating schedule on a fixed periodic time cycle. The application of PESP to timetable generation originates with [26] (see also [1]). That problem, and all subsequent implementations of PESP, concerns the scheduling of passenger trains on "clockface" schedules, typical of European services. To better market services to passengers, European railways frequently advertise cyclical departures, such as an InterRegio train departing Zurich for Luzern twice an hour, every hour, at four and thirty-five minutes past the hour. Of the four model classifications presented here, PESP enjoys the status of being the only timetable generation model that is actually in regular commercial service (see for example [27]).

1.4. Modeling significance of explicit track representation

The value of explicitly modeling track infrastructure in a timetabling or dispatching formulation is best understood by examining typical train path interactions. It is frequently believed that an operating constraint based upon a fixed headway time between trains is sufficient and equivalent to explicit modeling of track constraints, but it will be demonstrated with these examples that fixed headway time is a relaxation of the true constraint. In both cases, the difficulty is that the true timed headway between two trains is frequently a function of the dispatch pattern of additional proximate train paths and the routing of the train path, both of which are frequently decision variables. A MISLP model addresses this dependency through its sequencing variables. Alternatively, these constraints would have to be nonlinear to accurately represent these conditions without explicit track occupancy modeling.

Fig. 2 displays an example where three trains are traveling in the same direction along a signaled line with a typical red/yellow two block separation between any two trains. A fixed headway time constraint would establish minimum timed separations between any pair of consecutive trains, but this timing would be based upon the independent performance of each train, without interference. In the figure, two passenger trains, RE 103 and IC 208, are caught behind a slower freight train, GV 09. The timed headway between trains RE 103 and IC 208 would normally be a function of their track speed, but in this case their headway must be lengthened because train RE 103 is forced to travel at the slower speed of train GV 09. Thus, the headway between any two trains is not only a function of their joint performance, but the performance of any preceding train. This limiting function in fleets of trains is called "signal wake", and is documented in [28].

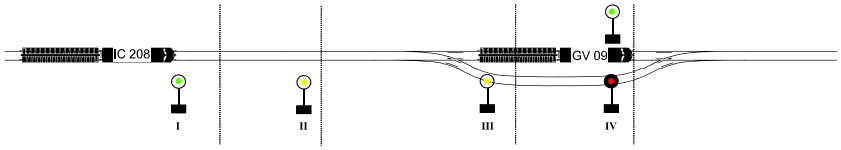


Fig. 3. Train headway times defined by siding selection.

Table 1
Variables, sets, and parameters of problem (MISLP).

Component	Type	Description
$d_{r,i}$	Real variable	The time that train r departs from point i
$a_{r,i}$	Real variable	The time that train r arrives at point i
$I_{r,r',i,j}$	Binary variable	An indicator which is true if trains r and r' interact, meet or pass, between points i and j
$F_{r,r',i,j}$	Binary variable	An indicator which is true if train r departs point i after train r' arrives at point i
$G_{r,r',i,j}$	Binary variable	An indicator which is true if train r' departs point j after train r arrives at point j
R	Set	The set of trains considered in the planning problem
B	Set	The set of control points or track segments, where B_r is a subset of control points reachable by train r; $B_r \subseteq B$
$d^*_{r,i}$	Parameter	The preferred departure time for train r from point i
$a^*_{r,i}$	Parameter	The preferred arrival time for train r at point i
$s_{r,i,j}$	Parameter	The unimpeded travel time from point i to point j
$f_{r,r',i,j}$	Parameter	The estimated delay that train r will incur if meeting or passing train r' between points i and j
$\beta_{r,i}$	Parameter	The penalty coefficient for late departures by train r from point i
$\alpha_{r,i}$	Parameter	The penalty coefficient for late arrivals by train r at point i
ϕ	Parameter	A scale factor controlling the shape of the exponential penalty cost function
M	Parameter	A dominant large positive number

A second example of the limitations of fixed headway timing is displayed in Fig. 3. In this example, train IC 208 is following GV 09. As well as train IC 208 just continuing to follow GV 09, there is the option of train GV 09 taking the siding and allowing train IC 208 to pass. However, if GV 09 takes the siding, it must slow down even further to negotiate the siding curvature and brake to a stop. This will require an even longer headway time separation between GV 09 and the following IC 208. In actual operation, this would be enforced by yellow signal indications on the intervening track blocks. For the timetabling model, this means that the headway time is not fixed, but is instead a function of the ultimate train path solution.

Substituting timed headways for physical headways may be acceptable if the physical track network is sufficiently dense such that it is possible to post process the solution and correct any train path conflicts. Most of the timetabling implementations cited here expect some review and adjustment of the timetable solutions, usually by skilled railway dispatchers. As the physical track network becomes less dense, and the siding and signaling resources become sparse, it becomes less likely that post processing of a timetabling solution may find a feasible operating plan. In operating scenarios such as Figs. 2 and 3, explicit modeling of the track structure is necessary to more accurately represent the necessary operating headways on lines with limited signaling and switching resources, especially those typical of North America.

2. Mixed integer sequencing linear programs

The mixed integer sequencing linear program (MISLP) decomposes the scheduling decision into binary variables that decide the *sequencing* of pairs of conflicting trains at control points (stations, sidings, etc.) and real variables that determine the event times at control points. An example of a basic mixed integer sequencing formulation is presented below, abbreviated from [29]. The variables, parameters, and sets of the formulation are provided in Table 1.

(MISLP)

$$\min \sum_{\substack{r\in R\\ i\in B_r}} [\beta_{r,i}|d_{r,i} - d^*_{r,i}|^{\phi} + \alpha_{r,i}|a_{r,i} - a^*_{r,i}|^{\phi}] \tag{1}$$

s.t.

$$d_{r,i} \geq a_{r,i} \quad \forall r \in R; i \in B_r \tag{2}$$

$$I_{r,r',i,j} + F_{r,r',i,j} + G_{r,r',i,j} = 1 \quad \forall r, r' \in R; i, j \in B_r \cap B_{r'} \tag{3}$$

$$a_{r,i} - d_{r,j} \geq s_{r,i,j} + \sum_{\substack{r'\in R\\ i,j\in B_r\cap B_{r'}}} f_{r,r',i,j} I_{r,r',i,j} \quad \forall r \in R; i, j \in B_r \tag{4}$$

$$\begin{aligned}
&d_{r,i} - a_{r',i} \geq M(F_{r,r',i,j} - 1) \quad \forall r, r' \in R; i, j \in B_r \cap B_{r'}\\
&d_{r',j} - a_{r,j} \geq M(G_{r,r',i,j} - 1) \quad \forall r, r' \in R; i, j \in B_r \cap B_{r'}\\
&a_{r',i} - d_{r,i} \geq M(I_{r,r',i,j} - 1) \quad \forall r, r' \in R; i, j \in B_r \cap B_{r'}\\
&a_{r,j} - d_{r',j} \geq M(I_{r,r',i,j} - 1) \quad \forall r, r' \in R; i, j \in B_r \cap B_{r'}
\end{aligned} \tag{5}$$

$$a, d \geq 0 \quad I, F, G \in \{0, 1\}.$$

This particular example displays a nonlinear objective, but variations with linear objectives also appear in the literature. Eq. (1) is a nonlinear cost function of delays at arrivals and departures, each individually priced. The language implies a flow from arrival to departure, so that at each control point a train is visualized as first arriving and then departing. Constraint (2) enforces chronological continuity in the decision variables, that is, a train cannot physically depart at a time earlier than its arrival at each control point. Constraint (3) enforces a logical set of alternatives between a pair of trains and a pair of control points; either the trains meet/pass, or the trains pass control points in either direction such that they do not interact between the stated pair of control points. Constraint (4) enforces the necessary travel time between control points, including estimated delays due to meets/passes. Finally, constraints (5) enforce the necessary arrivals and departures at control points according to the paired interaction selected by the indicator variables I, F, and G.

MISLP models are suitable for applications where the primary managerial concern is adjustments to the timing of trains, within a known feasible schedule. The input data set of trains must itself have at least one feasible schedule solution. There is no provision for the selection of a preferred subset of trains from the input data set. The MISLP model is less suitable for network capacity analysis, because it does not explicitly represent the track network. Note also that the MISLP model perceives all train interactions as

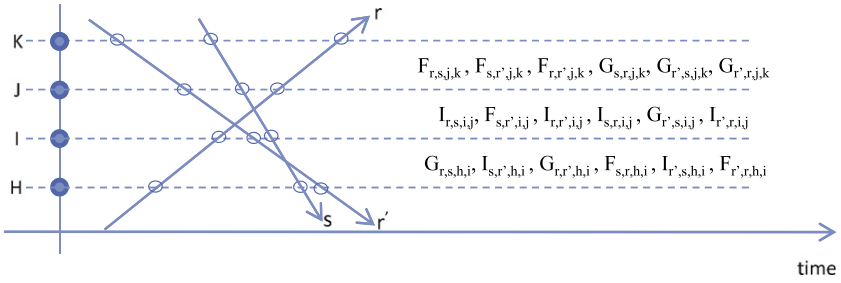


Fig. 4. Example of the meaning and encoding of indicator variables in the MISLP model.

pairwise events. Thus the MISLP model becomes significantly more complex when modeling complex junctions where more than two trains may interact simultaneously.

For example, Fig. 4 demonstrates the meaning and values of the indicator variables I, F, and G for a set of three train paths with both meets and passes. The variables listed in the figure are all true or $= 1$. Note that three indicator variables are defined for every permutation of pairs of trains over each track segment, not merely combinations of pairs of trains.

The invention of the MISLP model can be traced back to [30]. This was the first model to explicitly formulate the train meet/pass problem as a linear program. The formulation and solution algorithm was adopted from the job-shop problem of [31]. In this formulation, instead of branching on key decision variables (binary flags representing the sequencing of pairs of jobs or trains), branching is accomplished by dynamically adding constraints implied by the value of the branching decision variable, which dramatically speeds the solution time of the problem at each node of the solution tree. The decision variables remaining in the formulation represent the real valued times at which trains pass control points. This formulation is unique because it eliminates the direct manipulation of integer variables, but at the cost of extreme assumptions such as unlimited siding capacity.

The core of the MISLP formulation presented here is developed in [23]. In its initial application, patented under the name "SCAN I", no objective function was defined. The solution algorithm sought feasibility only, and a second stage algorithm computed the reliability statistic of the returned schedule. The reliability statistic was defined as the proportion of schedule samples that remained feasible after the free running times of the trains were perturbed over a sample distribution. The feasible solution search algorithm was derived from the branching algorithm of [32].

This model has been extended with a variety of objective functions. Kraay et al. [33] formulates a nonlinear, but convex, objective function for the minimization of fuel consumption. The authors find that significant fuel savings can be achieved by pacing trains at just the required speeds to conform to the master schedule, while retaining an improved meet/pass plan. The MISLP formulation presented here then appears in [29] as part of the earlier described research at Burlington Northern. Based on the SCAN engine, the objective considered train times, car connections, and a very limited range of crew change decisions.

Mills et al. [34] formulate a MISLP problem where the meet/pass plan (variables I, F, and G) is supplied as input data, and is not part of the decision. The intended application is dynamic rescheduling, or live dispatching. The nonlinear multi-objective of minimizing fuel consumption and delay at the destination is manipulated by managing the intermediate travel times. Mills et al. also present an alternative discrete time formulation and a sequential scheduling algorithm, which solves much faster while achieving a solution value only a half of a percent less than the nonlinear formulation. Cai and Goh [35] offer a greedy heuristic for this problem.

Carey and Lockwood [36] apply the MISLP model to a typical British single direction, high speed passenger network. The solution method is sequentially solved for the meet/pass variables one train at a time. Carey [37] extend this application to complex networks of junctions, multiple station platforms, and alternate routes. Carey [38] extend the model again to two-way traffic with the surprising conclusion that the two-way model had a smaller solution space and faster solution than the one-way model. Even so, the last model produced a very large MIP formulation, and a problem containing only 4 trains and 6 track links required days of computing time on a Sun 4 computer.

Higgins et al. [39,40] (original work in [41]) make a unique application of MISLP to the location of sidings along a typical Australian single track line. Using a method analogous to Bender's decomposition, the problem is split into sub-problems which are: first, solve for optimal siding locations and train times while holding the meet/pass plan fixed, and then second, solve for an optimal meet/pass plan and schedule while holding track design fixed. The examples presented found a siding track reduction of 45% from standard allocations for comparable performance on a single track line. Higgins et al. [42] investigate the application of the traditional Bender's decomposition to this model with a nonlinear objective and fixed track structure, and derives a lower bound for the meet/pass plan train delay which assists in the branch and bound node pruning.

More recently, Zhou and Zhong [43] apply MISLP to a double track high-speed railway in China and develop a Pareto optimal frontier for two distinct objectives: minimization of the variance of interdeparture times and minimization of the total travel time. Later, Zhou and Zhong [44] extend the same model to a single track line and proposes a simple and computationally efficient lower bound rule to estimate the total train delay due to resolving existing crossing conflicts in a partial schedule. Wong et al. [45] apply a MISLP formulation to an urban transit network with the objective of minimizing the sum of passenger transfer waiting times. Törnquist and Persson [46] address the live rescheduling of trains already dispatched, and in [47] report that local optimization over a rolling time horizon achieves desirable computational results relative to optimization over the complete time horizon. Aronsson et al. [48] reprise the MISLP model to consider complex resources such as parallel tracks and compare three variations of resource constraint formulation. Dollevoet et al. [49] combine the sequencing elements of MISLP with assignment variables (see Section 3) to enforce capacity limits in stations, and find that formulation variations with explicit assignment variables are easier to solve than those with aggregate capacity limits. Corman et al. [50] formulate the sequencing variables as *alternative arcs* in an alternative graph (see Section 3.2).

3. Binary integer occupancy programs

Many train pathing problems are less concerned with the exact timing of trains and more focused on the combinatorial solution of many trains competing for a limited track network. Example problems include the allocation of tracks in complex

Table 2
Variables, sets, and parameters of problem (BIOP).

Component	Type	Description
$x^r_{i,t}$	Binary variable	An indicator that is true if train r occupies block i during discrete time interval t
R	Set	The set of trains considered in the planning problem
B^r	Set of pairs	The set of pairs of blocks (control points or track segments) and initial times that compose one or more feasible paths for train r, determined before consideration of conflicting trains
$\Phi^r_{i,t}$	Set of pairs	The set of all time indexed blocks on which train r may continue its journey from block i at time t
$\Omega_{i,t}$	Set of pairs	The set of all time indexed blocks which will conflict with an occupancy of block i at time t
$c^r_{i,t}$	Parameter	The cost coefficient associated with variable $x^r_{i,t}$
o^r	Parameter	The block at which a train enters the line
d^r	Parameter	The block at which a train leaves the line

stations, the mixing of disparate or contractually separate trains on a shared network, and the valuation of infrastructure. These problems are better served by a model that explicitly considers the track network. The binary integer occupancy program (BIOP) accomplishes this by expressing the finite *occupancy* of a segment of track by a single train for a discrete time duration as a *binary* variable. The feasible decision space is constructed from the fundamental operating rule that only one train may occupy a controlled track segment at any time. In some models the decision space is further condensed by fully enumerating each of the feasible paths for a train as binary decision variables and cross referencing these variables to occupation constraints indexed by controlled track segment and discrete time.

A generic example of a basic binary integer occupancy formulation of the author is presented below. This example presumes a simple single route line. The variables, parameters, and sets of the formulation are provided in Table 2.
(BIOP)

$$\min \sum_{\substack{r\in R\\(i,t)\in B^r}} c^r_{i,t}x^r_{i,t} \tag{6}$$

s.t.

$$\sum_{(i,t)\in B^r|i=o^r} x^r_{i,t} = 1 \quad \forall r \in R \tag{7}$$

$$x^r_{i,t} = \sum_{(k,l)\in\Phi^r_{i,t}} x^r_{k,l} \quad \forall r \in R;\ (i,t) \in B^r | i \neq d^r \tag{8}$$

$$\sum_{(i,t)\in B^r|i=d^r} x^r_{i,t} = 1 \quad \forall r \in R \tag{9}$$

$$\sum_{\substack{m\in R\\(k,l)\in B^m\cap(\Omega_{i,t}\cup\{(i,t)\})}} x^m_{k,l} \leq 1 \quad \forall (i,t) \in \bigcup_{r\in R} B^r \tag{10}$$

$x \in \{0, 1\}$.

The objective is a linear cost function of each individual track occupation, defined as is appropriate for individual circumstances (cost of track, arrival or departure emphasis, etc.). It should be apparent that penalties for delays can simply be assigned to each variable that represents a delayed destination block and time. A single commodity, unidirection flow, is defined by constraints (7)–(9). Side constraints (10) define a multicommodity flow.

This decision variable structure originates with Mees [24], which also is the first example of a time expanded graph representation of a railway track structure. The application addresses heavy tonnage trains on long single track railways. A solution algorithm is presented that loops through an ordered list of trains, rescheduling one train at a time while holding the other trains fixed. Train paths are valued according to Lagrangian multipliers assigned to each arc (block indexed by time), which are updated at the beginning of each pass through the list of trains. The algorithm obtains locally optimal solutions, and continues until a full pass through the list of trains is made without any objective improvement. Goh and Mees [51] extend this graph visualization to a nonlinear model of the train acceleration dynamics, but find the model intractable for practical problem sizes.

Brännlund et al. [52] enumerate complete paths for each train and represent each path alternative as a binary decision variable. Occupancy constraints are enumerated for each controlled track segment in a time expanded graph, and decision variables are grouped in cliques such that only one path variable may occupy a track segment at a discrete time. Lagrangian relaxation is then applied to the block occupancy constraints, and the market value of track time is measured as the value of the assigned Lagrangian multipliers. Individual train paths are enumerated sequentially in pre-processing and the order of processing is discussed in detail in [53]. The pathing approach is applied to live dispatching by [54], where the paths represent simulated feasible options for recovery from a network disruption.

Caprara et al. [55] model a single direction of dense traffic on a main line, where there are no capacity limits at intermediate stations. The track segments between stations are controlled, and the occupancy constraints are indexed by pairs of incompatible arcs (trains indexed by discrete time). The solution method again is a sequential scheduling of trains, ranked by their profit as determined by Lagrangian multipliers. Extensions to limit capacity at stations and other operating constraints are provided in [56]. Ahuja et al. [57] explicitly model occupancy constraints on both the main line and within the stations or sidings by distinguishing between occupancies while in motion and occupancies while standing.

Borndörfer et al. [58] generalize the concept of clique sets of incompatible occupancies further to generate timetables and train paths for the Hannover–Kassel–Fulda network of 31 stations, 45 line segments, and 946 candidate trains, over a 6 h time horizon. Timetables are generated repeatedly as part of a Parkes "i-bundle" auction of train paths. In this auction, the rail authority collects bids from commercial operators for train paths, and then generates a timetable selectively from the bids that maximizes bid revenue. The selected bids are announced, and the commercial operators have the option to make additional bids. The timetable generation and bidding cycle is repeated until the revenue objective ceases to improve.

Recent solution advances seek to simplify the large constraint matrix of this problem. Fischer et al. [59] examine the Lagrangian dual of the problem, relaxing the capacity and clique constraints. Groups of trains in the relaxed solution are fixed to integer path assignments by a rounding heuristic. The solution is repeated with subsequent additional groups of trains fixed until a complete timetable is rendered. Fischer and Helmburg [60] minimize the memory required to solve the problem by only generating the path time-space graph as alternative solutions require it. The initial graph is limited to the earliest available departure path for each train. Borndörfer et al. [61] describe a depth-first rapid branching technique that rapidly fixes variables by progressively modifying the coefficients of variables so that they converge to integer values in successive solutions.

3.1. Path enumeration models

A small group of models dispense with representing individual track segments in the formulation altogether. Some argument could be made for segregating these models into a fifth form (perhaps "PATH"), but they can be visualized as a simplification of the BIOP model. These models enumerate each train path alternative as a binary decision variable, and then every potentially conflicting path alternative is enumerated. These conflicts are then collected as clique sets which allow only one of the path alternatives to be selected. The resulting model is a node packing problem and is effectively a BIOP problem with the flow conservation constraints simplified or removed. To date this model has only been applied to railway junctions or stations, where the distances and times are very short, but the paths are very complex.

Zwaneveld et al. [4] model a dense Dutch railway junction in this manner and the objective seeks to maximize the quantity of feasible timetable paths. Occupancy constraints are indexed by cliques of incompatible paths. The problem is cast as a node packing problem and solved with a branch and cut method. A similar application is made to a French railway junction by [62,63] and solved with constraint programming. Caimi et al. [64] present an interesting reduction technique for a similar model where the feasible path set is reduced by deleting paths that are dominated by other alternatives. For example, within a set of paths with equivalent endpoint timings, paths with a higher conflict count are deleted.

Limitations of this model are that the decisions made within the optimization itself are very limited and the data preparation or pre-processing task is very demanding. This model neither decides timings nor constructs routes within its optimization process. Both of these decisions must be determined prior in the input data. On the other hand, there is no limit to the scenarios of conflict between paths which may be modeled. More recently, Caimi et al. [65] start from the pathing model perspective and then unwind the paths so that once again individual track resources are explicitly considered. Arranging track resources in "Resource Trees", Caimi et al. find the formulation to be more compact and efficient at obtaining solutions.

3.2. Alternative graph models

A large number of BIOP models appear in the literature that are formulated as *alternative graphs*. An argument can be made that they are conceptually identical, that only the preferred solution method differs. However, there is a subtle and powerful difference: the arc and node variables are no longer tied to a rigid discrete time index. Train path alternatives can be freely defined in real time, and the conflicts between them fully represented in the graph. Of course, the graphs may yet be very large.

In alternative graph theory, the larger problem graph contains subgraphs, each of which represents the feasible moves for an individual train in time and space. The nodes or vertices of these subgraphs are connected to any other nodes or vertices that represent conflicting train paths. For example, nodes that represent conflicting trains on a track segment are connected together, and form a clique where only one node may be occupied. The arcs connecting these clique members are "alternative" arcs, and any combinatorial solution to the flow on the graph requires that one each from these cliques of alternative arcs be selected. This is really not different than the resolution of path or resource constraints in the binary integer occupancy models already presented, only these authors might identify themselves as graph theorists, and utilize flow determination algorithms instead of linear programming.

Mascis and Pacciarelli [66] would appear to be the earliest application of alternative graph theory to timetabling, or any sort of railway scheduling. The authors approach the train timetabling problem with a strong foundation in job-shop scheduling, where the railway scenario is analogous to a shop with blocking and no-waiting behavior. The authors demonstrate that traditional job-shop algorithms are not appropriate for these conditions, and present a better algorithm called Avoid Maximum Current Cmax (AMCC). In short, this algorithm compares two alternative arcs, determines which selection would result in the worst solution, and avoids selecting that arc. Thus, it reduces the feasible solution set by iteratively deleting undesirable slices of the feasible set, until only one solution remains.

Mazzarello and Ottaviani [67] directly apply Mascis and Pacciarelli's formulation to dynamic rescheduling after delays on two Netherlands Railways networks, Nieuw Vennep–Amsterdam Lelylaan–Amsterdam Zuid WTC and Roosendaal–Willemsdorp–Breda. A multi-objective of minimizing delays and minimizing fuel consumption was formulated. The authors solve problems using a rolling time horizon which they refer to as "schedule windows", similar to Törnquist. D'Ariano et al. [68] also apply Mascis and Pacciarelli's model to a rescheduling problem on the Netherlands Railways Schiphol rail network (see also [69]). The authors set a benchmark calculation time of 120 s to reschedule an hourly cyclic service of 54 trains. In order to meet this objective, the authors enhanced the AMCC algorithm by enhancing the selection of alternative arcs. The authors compiled sets of "static implications", train path choices that are also implied by specific alternative arc choices. Thus at each alternative arc selection, many other related arcs are fixed. D'Ariano et al. [70] test the same model for dynamic timetabling for dispatching support (see [71] for a thorough exposition of this dispatching problem).

Corman et al. [50] demonstrate an alternative graph formulation of the MISLP model (see Section 2). The alternative arcs replace the sequencing variables of MISLP. The authors seek to minimize delay propagation in a dynamic dispatch response to primary delays. The authors decompose the global scheduling problem into local scheduling problems centered on dispatch territories and then compose a coordinating optimization problem for the territory boundary.

4. Hypergraph formulation

The hypergraph model of train movements revises the decision variables of the BIOP model so that they indicate the sequential occupancy, or the *transition*, between two controlled track segments, over an interval of one *or more* discrete time units. The controlled track segments are individually indexed by discrete time units over the planning horizon, and form nodes within a time expanded graph. Additional nodes define zones of transition conflict between adjacent track segments. The binary decision variables are directed arcs on this graph that potentially enclose multiple nodes. The removal of the identity that restricts an arc to two graph nodes classifies this model as a hypergraph.

A hypergraph is a graph in which the definition of an edge is expanded to include any non-empty subset of nodes (see [72] for a formal definition). In the case of a railway line, a track segment has frequently been modeled as an arc with nodes at the endpoints of the track segment. However, occupation of a track segment in practice may conflict with other adjacent track segments. A discrete time graph model of the network likely will also require a train movement decision variable to register the occupation of multiple graph nodes each representing a labeled network resource for a discrete time unit. Thus a hypergraph provides the flexibility to encapsulate all of the variety of implied conflicts of a train movement variable.

The hypergraph model originated to address a fatal flaw in the BIOP model when applied to single track lines. In the BIOP model,

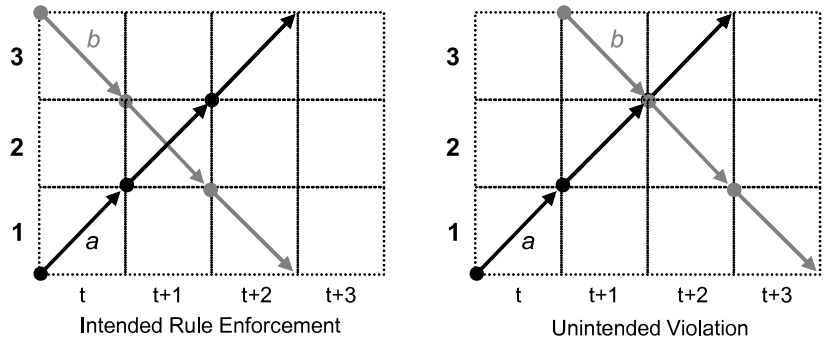


Fig. 5. Example of operational infeasibility under simple occupational constraint.

because of the discrete time clock, train paths may pass each other without triggering the occupancy constraint. An example of this operationally infeasible move is depicted in Fig. 5. In this figure, each square is an available track segment ($\{1, 2, 3\}$) over a discrete time interval ($\{t, t+1, t+2, t+3\}$), and also in this case the track is single track without any holding capacity or station at the track endpoints. A standard BIOP constraint intends that only one train path may occupy each track segment, and frequently also implies that trains may not pass each other on this diagram. However, as can be seen on the "Unintended Violation" portion of the diagram, two trains will be scheduled to pass each (illegally) on this single track because if phased selectively they will not be included in the track occupancy constraint.

A hypergraph with its representative track network is shown in Fig. 6. Hollow nodes are track segment occupancies, while solid nodes represent potential transition conflicts between track segments, and are only defined where paths represent separate trains. Note the conflict between the two labeled arcs would not register in a BIOP model, unless the paths were fully enumerated and clique conflict sets determined in pre-processing.

An example of a hypergraph formulation is presented below, abbreviated from [73]. The variables, parameters, and sets of the formulation are provided in Tables 3 and 4.

(Hypergraph)

$$\max \sum_{\substack{r\in R \\ (p^r_o,j,u,v)\in\Psi^r}} (c^r_p + c^r_e(u - p^r_e))x^r_{p^r_o,j,u,v} + \sum_{\substack{r\in R \\ (p^r_d,\mathbf{e}^r,u,v)\in\Psi^r}} c^r_l(p^r_l - v)x^r_{p^r_d,\mathbf{e}^r,u,v} + \sum_{\substack{r\in R \\ (i,j,u,v)\in\Psi^r|i=j}} c^r_s x^r_{i,j,u,v}. \quad (11a)$$

Linear Network Constraints

$$\sum_{(p^r_o,j,u,v)\in\Psi^r} x^r_{p^r_o,j,u,v} \le 1 \quad \forall r \in R \quad (11b)$$

$$\sum_{(a,i,u,t)\in\Psi^r} x^r_{a,i,u,t} = \sum_{(i,j,t,v)\in\Psi^r} x^r_{i,j,t,v}$$

$$\forall r \in R, \{i \in B | i \ne p^r_o\}, t \in T \quad (11c)$$

$$\sum_{(p^r_d,\mathbf{e}^r,u,v)\in\Psi^r} x^r_{p^r_d,\mathbf{e}^r,u,v} \le 1 \quad \forall r \in R \quad (11d)$$

$$x \in \{0, 1\}. \quad (11e)$$

Side Constraints

$$\sum_{\substack{r\in R \\ (i,j,u,v)\in\Psi^r|u\le t<v}} x^r_{i,j,u,v} \le b^i_t \quad \forall i \in B, t \in T \quad (12a)$$

$$\sum_{\substack{r\in R^N \\ v\in\{t+1-\epsilon,\dots,t+1+\delta\} \\ (i,j,u,v)\in\Psi^r|j=a+1,j\ne i}} x^r_{i,j,u,v} + \sum_{\substack{r\in R^S \\ v\in\{t+1-\epsilon,\dots,t+1+\delta\} \\ (i,j,u,v)\in\Psi^r|j=a,j\ne i}} x^r_{i,j,u,v} \le \upsilon^a_t$$

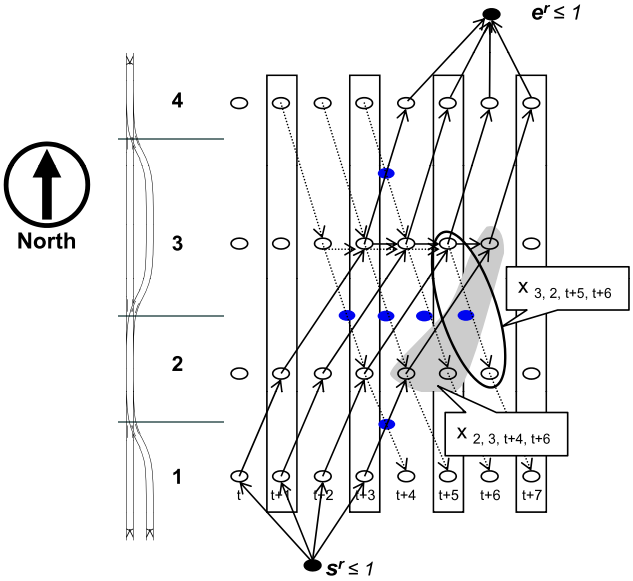


Fig. 6. Hypergraph of train paths, arcs $x_{2,3,t+4,t+6}$ and $x_{3,2,t+5,t+6}$ do not conflict in block occupancy but do conflict in transition at cell $(2, t+5)$.

$$\forall (a,t) \in \Upsilon \quad (12b)$$

$$\sum_{\substack{r\in R^N|h^r\ge 1 \\ a\in\{i-h,\dots,i-1\} \\ (a,j,u,v)\in\Psi^r|u\le t<v,a\ne j}} x^r_{a,j,u,v} + \sum_{\substack{r\in R^N \\ (i,j,u,v)\in\Psi^r|u\le t<v}} x^r_{i,j,u,v} \le b^i_t$$

$$\forall i \in B, t \in T \quad (12c)$$

$$\sum_{\substack{r\in R^S|h^r\ge 1 \\ a\in\{i+1,\dots,i+h\} \\ (a,j,u,v)\in\Psi^r|u\le t<v,a\ne j}} x^r_{a,j,u,v} + \sum_{\substack{r\in R^S \\ (i,j,u,v)\in\Psi^r|u\le t<v}} x^r_{a,j,u,v} \le b^i_t$$

$$\forall i \in B, t \in T. \quad (12d)$$

Objective and constraints (11) define a set of independent single commodity flows. Side constraints (12) bind these flows together and regulate them as a multicommodity flow. Constraint (11b) enforces a single departure for each train, and because it is an inequality also offers the option of removing the train from the schedule if it is not productive, while constraint (11c) enforces conservation of mass at the nodes (enforcing a single train path) and sinking constraint (11d) ensures a single terminal arrival.

Side constraint (12a) enforces a common block occupancy limit, but instead of enumerating every track segment with a capacity of 1, parallel segments such as sidings or multiple through track are defined as a single block with capacity $b^i_t \ge 0$. Other variations of this constraint, such as enumerating every track segment, are possible. Constraint (12b) enforces the limits on transitions between track segments. The sensitivity of these limits are adjustable

Table 3
Variables, parameters, and nodes of problem (P).

Component	Type	Description
$x^r_{i,j,u,v}$	Binary variable	Occupancy arc representing the possession of node i at time u and the exit into node j at time v of train r
p^r_o	Parameter	Origin of train r
p^r_d	Parameter	Destination of train r
p^r_e	Parameter	Earliest allowed time of origination of train r
p^r_l	Parameter	Latest allowed time of termination of train r
$\mathbf{e}^r$	Artificial node	Artificial sink node designating train r is off the network
c^r_p	Parameter	Fixed value (or "profit") of train r completing its journey
c^r_e	Parameter	Incentive per unit time for later origination of train r
c^r_l	Parameter	Incentive per unit time for earlier termination of train r
c^r_s	Parameter	Cost per unit time of enroute waiting (stopped) of train r
b^i_t	Parameter	Capacity (count of trains) of block i at time t
υ^i_t	Parameter	Capacity (count of trains) of cell i at time t
ϵ	Parameter	Dimension of leading transition window
δ	Parameter	Dimension of lagging transition window

Table 4
Sets of problem (P).

Set	Description
T	The discrete time horizon, ordered with starting value $t = 1$
R	The set of all trains
B	The set of all track blocks, ordered by a common reference direction of travel, such as "north" or "south"
R^N	The set of trains, $R^N \subset R$, traveling in the direction defined by increasing track block index
R^S	The set of trains, $R^S \subset R$, traveling in the opposite direction of trains in set R^N, $R^N \cup R^S \equiv R$
Ψ^r	The set of feasible path arcs (i, j, u, v) for train r supplied from pre-processing
Υ	The set of transition nodes between track segments

through the "window" parameters. Finally, constraints (12c) and (12d) limit the headways or follow on spacing of trains. All of these features are presented in detail in [73].

Harrod [25] applies this model to theoretical studies of the capacity of single track lines under various siding configurations and mixes of train speeds, and demonstrates that all traffic on a single track line benefits when priority trains operate at higher speeds.

5. Periodic event scheduling problems

Cyclical timetables are typical of passenger services in Europe. These services require that most, if not all, train paths repeat in time with period T. To accomplish this, the periodic event scheduling problem labels train positions at control points as events and defines span constraints between all potentially conflicting events. Like MISLP, PESP consists of timing variables, but unlike MISLP, there are no sequencing variables. This difference reflects the contrast in the motivating application of MISLP and PESP. The European cyclical timetables exist primarily on networks of multiple track main lines which are typically dispatched with a single direction to each track, whereas the North American MISLP examples are primarily single track with sidings. The timed headways between trains in the European scenario are reliably predictable, whereas in the single track meet/pass problem the headways are a function of the dispatch solution (encoded in the sequencing variables I, F, and G).

The span constraints are formulated as in (13), where the time of periodically repeating event j, τ_j, follows the time of event i, τ_i, by duration no less than l_{ij} and no greater than u_{ij}. This is graphically presented in Fig. 7 where events i and j repeat uniformly with period T. It is feasible to shift event j anywhere within the interval $[l_{ij}, u_{ij}]$, so long as the span between events i and j remains identical over all cycles of period T. The span constraints may also be compactly represented by the clockface chart in Fig. 8:

$$(\tau_j - \tau_i - l_{ij}) \bmod T \leq u_{ij} - l_{ij}. \tag{13}$$

Previous aperiodic network models perform poorly when modified for periodic timetable modeling. For example, Harrod [3]

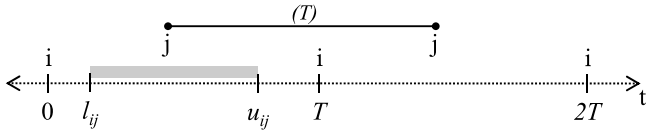


Fig. 7. Periodic span relationship between events j and i.

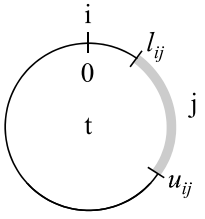


Fig. 8. Periodic span relationship between events j and i viewed as a clockface.

demonstrates that the hypergraph model is terribly difficult to solve to optimality when modified to represent a cyclical timetable. PESP, first proposed by [74], isolates the characteristic cyclical pattern as a smaller, more manageable problem. However, this functionality sacrifices the routing capability. The physical assignment of trains to track routes must be included in the input data set for PESP. Schrijver and Steenbeck [26] is widely cited as the central component of CADANS, the central element of Netherlands Railways' timetable planning system; however this reference is not available in English. CADANS is reported to apply constraint generation to obtain feasible solutions, with some local optimization capabilities.

Odijk [75] models the train paths through Arnhem CS station, The Netherlands, as a PESP problem. The problem is formulated as a graph of event time nodes connected by arcs representing the span constraints. The algorithm presented either returns a feasible timetable or proves infeasibility of the input data set. Different timetables are obtained by modifying the algorithm starting vector. The sample problem consisted of 12 trains and 54 spanning constraints over a period of $T = 30$ min. Timetable generation time ranged from 82 to 225 ms.

Table 5
Variables, sets, and parameters of problem (IP-PESP).

Component	Type	Description
x_a	Integer variable (arc)	The periodic tension between two events, $a = (i, j) \in A$, as defined in Eq. (14)
z	Integer variable (vector)	A vector of the period bases for the cycles of the graph D. For each sequence of related train path events or *cycle*, this determines how many periods, T, best encompass the cycle.
D	Graph	The graph composed of train events as nodes V and spans between events as arcs A
A	Set	The set of controlled pairs of events $\{(i, j)\}$
Γ	Matrix	The arc-cycle incidence matrix. Each element of the matrix asserts that a tension arc $a \in A$ either has a forward/backward direction in a cycle $(1/-1)$ or is not contained in a cycle (0)
$\mathbb{Z}$	Set of integers	
w_a	Parameter	The cost coefficient associated with increasing periodic tension x_a (increasing the time between events).
l_a	Parameter	The lower bound of the periodic tension x_a
u_a	Parameter	The upper bound of the periodic tension x_a
T	Parameter	The duration of the periodic cycle
ν	Parameter	The cyclomatic number, $\nu = \lvert A\rvert - \lvert V\rvert + 1$

Nachtigall and Voget [76] formulate and demonstrate a very complex bi-criteria application of genetic algorithms and linear programming in order to generate a Pareto-optimal frontier of potential timetables with reduced passenger waiting times at various levels of capital investment in infrastructure. The linear program contains binary decision variables which represent an available combination of infrastructure investments. A key constraint limits the weighted sum of passenger waiting times by a ceiling parameter. The program is too complex to solve as required, but a lower bound is derived for the infrastructure cost as a function of waiting time ceiling. Actual timetables for a given level of investment are determined as genetic algorithm solutions of a PESP formulation. Kroon and Peeters [77] devise a PESP extension that provides for variable train movement times, allowing the solution algorithm to adjust the train paths within defined limits to resolve an initially infeasible input data set.

Liebchen [27] pursues direct optimization of the PESP instance by integer programming. In this model the decision variables are the *periodic tension* values of the event graph as in Eq. (14) (see also [78]). This PESP variant is sometimes referred to in the literature as the "Cyclic Periodicity Formulation":

$$x_{ij} := (\tau_j - \tau_i - l_{ij}) \bmod T + l_{ij}. \tag{14}$$

The implemented integer programming formulation of [27] is presented below (in matrix notation). In this model the objective is to make the timings between events (typically passenger connections between trains and idle train time) as short as possible. The variables, parameters, and sets of the formulation are provided in Table 5.

(IP-PESP)

$$\min \sum_{a \in A} w_a x_a \tag{15}$$

s.t.

$$\Gamma^T x = Tz \tag{16}$$

$$l \le x \le u \tag{17}$$

$$z \in \mathbb{Z}^{\nu} \tag{18}$$

$$x \in \mathbb{Z}^{|A|}. \tag{19}$$

Liebchen [27] produces an optimal operating timetable for the Berlin U-bahn network with a period of 10 min. The bare IP formulation alone, before the specification of ν and any other bounds on z, is not suitable for solving practical problems. In particular, the number of z variables can be exponential and as initially unrestricted integer variables the problem is formidable. Liebchen and Peeters [79] document that it is appropriate in this problem to limit the dimension of z by the cyclomatic number, ν. Next, additional box constraints and inequalities are derived which limit the search space (see also [80]). The problem is separated first into a search for a minimal cycle basis z and then a solution of Problem (IP-PESP). The determination of a minimal z is very important, for it reduces the problem search space, reduces the solution time, and improves the quality of the solution. Nachtigall and Opitz [81] document a solution method that addresses the z variables, the "modulo parameters", by minimizing periodic slack with a modulo simplex algorithm.

Caimi [82] demonstrates a variant of PESP for cases where the train paths are mixed periodic and aperiodic. Caimi et al. [83] replace the tension variables with pairs of tensions representing minimum and maximum time intervals, which better supports solution feasibility at station stops (see also [84]).

6. Summary and comment

Timetable optimization models to date may be classified according to whether they are periodic or not, and whether they explicitly represent the constraints of the track structure. The four model structures presented here, MISLP, BIOP, hypergraph, and PESP, support these options as shown earlier in Fig. 1, but note the absence of a model that explicitly represents the track structure of a periodic network. Only Nachtigall and Voget [76] investigate infrastructure investment within a periodic timetable, but the computation is complex and the potential infrastructure investments are suggested by the input data set, rather than identified by the analysis.

MISLP, the oldest timetable optimization formulation, is appropriate for real valued scheduling of single period schedules. However, it is limited to pairwise train dispatching decisions, the starting data set must be known feasible, and it does not provide any direct analysis of the track infrastructure. Limiting the solution to pairwise train interactions significantly reduces the feasible solution set, as proven in [85], especially on single track lines. On the other hand, BIOP specifically models the track structure at discrete time intervals, and can self-select an optimal group of trains from within a larger input set of trains. BIOP is especially suited for the combinatorial packing of train paths within a complex network or the economic valuation of a segment of track.

The hypergraph model addresses a significant omission in the constraint structure of BIOP. There are a number of likely train operation scenarios that are infeasible that cannot be controlled by the decision variable and constraint structure of BIOP. The hypergraph model is capable of explicitly representing the track structure and self-selecting the solution set of trains in the same manner as the BIOP model, and thus it may replace and supersede the BIOP model in many relevant applications.

PESP remains the only timetabling model in regular service. Even the well-publicized aperiodic dispatching tool of [32] in the

United States has been out of service for many years. Practical applications of PESP have had the most measurable effect on railway operations in Europe. It is ideally suited for the solution of periodic timetables with many interrelated passenger connections between trains. It does, however, share the characteristic of MISLP in that it cannot resolve data sets where a feasible timetable does not exist. Other algorithms or user intervention are necessary to remove trains from the data set or make other adjustments to allow PESP to resolve the timetable.

Timetabling and dynamic dispatching of trains remains a challenging problem, but there is no doubt that even an imperfect timetabling, rescheduling, or dispatching tool is preferable over no tool at all, or relying on manual dispatching skill. This has been demonstrated over and over again in the literature, including [86,32,68,87,27,88]. It should be apparent to all that mathematical programming software and computing power have advanced dramatically in twenty years. Many of the older implementations in the literature which were discarded as too computationally demanding may warrant a fresh evaluation.

Finally, one of the motivations here in presenting these models by structure rather than by application is to stimulate ideas for alternative applications of these models. Dramatic changes to the fundamental infrastructure and train control systems are underway in North America. Electro-pneumatic braking is currently in operation on selected freight trains [89], and is expected to dramatically reduce stopping distances and speed the handling of "drag" freight. The Railway Safety Improvement Act of 2008 requires implementation of positive train control by December 31, 2015 [90]. Why not apply PESP to the timetabling of unit coal trains in Wyoming?

References

[1] L. Kroon, D. Huisman, E. Abbink, P.-J. Fioole, M. Fischetti, G. Maróti, L. Schrijver, A. Steenbeck, R. Ybema, The new dutch timetable: the or revolution, Interfaces 39 (1) (2009) 1–12.

[2] T. White, Alternatives for railroad traffic simulation analysis, Transportation Research Record 1916 (2005) 34–41.

[3] S. Harrod, Railway capacity management and planning. Ph.D. Thesis, University of Cincinnati, 2007.

[4] P.J. Zwaneveld, L.G. Kroon, H.E. Romeijn, M. Salomon, S. Dauzère-Pérès, S.P.M. Van Hoesel, H.W. Ambergen, Routing trains through railway stations: model formulation and algorithms, Transportation Science 30 (3) (1996) 181–194.

[5] L. Kroon, G. Maróti, M.R. Helmrich, M. Vromans, R. Dekkar, Stochastic improvement of cyclic railway timetables, Transportation Research Part B 42 (2008) 553–570.

[6] C. Liebchen, M. Lübbecke, R. Möhring, S. Stiller, The concept of recoverable robustness, linear programming recovery, and railway applications, in: Ravindra K. Ahuja, Rolf H. Möhring, Christo D. Zaroliagis (Eds.), Robust and Online Large-Scale Optimization, in: Lecture Notes in Computer Science, vol. 5868, Berlin, 2009, pp. 1–27.

[7] R.M. Goverde, Railway timetable stability analysis using max-plus system theory, Transportation Research Part B 41 (2007) 179–201.

[8] M. Goerigk, A. Schöbel, An empirical analysis of robustness concepts for timetabling, in: Erlebach, T., L M. (Eds.), 10th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems, ATMOS'10. Liverpool, 2010, pp. 100–113.

[9] T.G. Crainic, Service network design in freight transportation, European Journal of Operational Research 122 (2000) 272–288.

[10] M.F. Gorman, S. Harrod, Operations research approaches to asset management in freight rail, in: Wiley Encyclopedia of Operations Research and Management Science, John Wiley & Sons, Inc., 2011.

[11] B. Andersen, A survey of the swiss public transport system and policy, Transport Reviews 13 (1) (1993) 61–81.

[12] R.M. Lusby, J. Larsen, M. Ehrgott, D. Ryan, Railway track allocation: Models and methods, OR Spectrum 33 (4) (2011) 843–883.

[13] A. Caprara, L. Kroon, M. Monaci, M. Peeters, P. Toth, Passenger railway optimization, in: C. Barnhart, G. Laporte (Eds.), Transportation; Handbooks in Operations Research and Management Science, Vol. 14, North-Holland, Amsterdam, 2007, pp. 129–187.

[14] L. Kroon, D. Huisman, G. Maróti, Optimisation models for railway timetabling. in: Hansen, I. A., Pachl, J. (Eds.), Railway Timetable and Traffic. Hamburg, 2008.

[15] J. Törnquist, Computer-based decision support for railway traffic scheduling and dispatching: A review of models and algorithms, in: 5th Workshop on Algorithmic Methods and Models for Optimization of Railways, Internationales Begegnungs- und Forschungszentrum f Dagstuhl, Germany, 2006.

[16] J.-F. Cordeau, P. Toth, D. Vigo, A survey of optimization models for train routing and scheduling, Transportation Science 32 (4) (1998) 380–404.

[17] R.H. Leilich, Application of simulation models in capacity constrained rail corridors, in: D.J. Medeiros, E.FR. Watson, J.S. Carson, M.S. Manivannan (Eds.), Proceedings of the 1998 Winter Simulation Conference, Winter Simulation Conference Foundation, 1998, pp. 1125–1133.

[18] J. Gibson, Train performance calculators and simulation models, in: TRB Workshop on Railroad Capacity and Corridor Planning. Transportation Research Board, Washington, D.C. January 2002. unpublished, obtained from participants/presenters.

[19] M. Bussieck, T. Winter, U. Zimmerman, Discrete optimization in public rail transport, Mathematical Programming 79 (1997) 415–444.

[20] D. Jovanović, P.T. Harker, A decision support system for train dispatching: an optimization-based methodology, Journal of the Transportation Research Forum 31 (1) (1990) 25–37.

[21] A.A. Assad, Models for rail transportation, Transportation Research Part A 14 (1980) 205–220.

[22] M. Yabe, A glimpse of the history and activities of operational research in japanese national railways and a case study by the author, Journal of the Operations Research Society of Japan 10 (1-2) (1967) 27–65.

[23] D. Jovanović, P.T. Harker, Tactical scheduling of rail operations: the scan I system, Transportation Science 25 (1) (1991) 46–64.

[24] A. Mees, Railway scheduling by network optimization, Mathematical and Computer Modelling 15 (1) (1991) 33–42.

[25] S. Harrod, Capacity factors of a mixed speed railway network, Transportation Research Part E 45 (2009) 830–841.

[26] A. Schrijver, A. Steenbeck, Dienstregelingontwikkeling voor railned (timetable construction for railned). Tech. rep., Center for Mathematics and Computer Science, Amsterdam, 1994 (in Dutch).

[27] C. Liebchen, The first optimized railway timetable in practice, Transportation Science 42 (4) (2008) 420–435.

[28] N. Welch, J. Gussow, Expansion of Canadian National Railway's line capacity, Interfaces 16 (1) (1986) 51–64.

[29] D.R. Kraay, P.T. Harker, Real-time scheduling of freight railroads, Transportation Research Part B 29B (3) (1995) 213–229.

[30] B. Szpigel, Optimal train scheduling on a single track railway, in: Operational Research '72, Proceedings of the Sixth IFORS International Conference on Operational Research. IFORS, North-Holland Publishing, London, 1973, pp. 343–352.

[31] H.H. Greenberg, A branch-bound solution to the general scheduling problem, Operations Research 16 (2) (1968) 353–361.

[32] R.L. Sauder, W.M. Westerman, Computer aided train dispatching: Decision support through optimization, Interfaces 13 (6) (1983) 24–37.

[33] D. Kraay, P.T. Harker, B. Chen, Optimal pacing of trains in freight railroads: model formulation and solution, Operations Research 39 (1) (1991) 82–99.

[34] R. Mills, S. Perkins, P. Pudney, Dynamic rescheduling of long-haul trains for improved timekeeping and energy conservation, Asia-Pacific Journal of Operational Research 8 (1991) 146–165.

[35] X. Cai, C. Goh, A fast heuristic for the train scheduling problem, Computers and Operations Research 21 (5) (1994) 499–510.

[36] M. Carey, D. Lockwood, A model, algorithm and strategy for train pathing, Journal of the Operational Research Society 46 (8) (1995) 988–1005.

[37] M. Carey, A model and strategy for train pathing with choice of lines, platforms, and routes, Transportation Research Part B 28B (5) (1994) 333–353.

[38] M. Carey, Extending a train pathing model from one-way to two-way track, Transportation Research Part B 28B (5) (1994) 395–400.

[39] A. Higgins, E. Kozan, L. Ferreira, Modelling the number and location of sidings on a single track railway, Computers and Operations Research 24 (3) (1997) 209–220.

[40] A. Higgins, E. Kozan, L. Ferreira, Heuristic techniques for single line train scheduling, Journal of Heuristics 3 (1997) 43–62.

[41] A. Higgins, E. Kozan, L. Ferreira, Optimising the position of crossing and overtaking loops on a single track railway, Tech. Rep. 94-16, Queensland University of Technology, Physical Infrastructure Centre, June 1994.

[42] A. Higgins, E. Kozan, L. Ferreira, Optimal scheduling of trains on a single line track, Transportation Research Part B 30 (2) (1996) 147–161.

[43] X. Zhou, M. Zhong, Bicriteria train scheduling for high-speed passenger railroad planning applications, European Journal of Operational Research 167 (2005) 752–771.

[44] X. Zhou, M. Zhong, Single-track train timetabling with guaranteed optimality: branch-and-bound algorithms with enhanced lower bounds, Transportation Research Part B 41 (2007) 320–341.

[45] R.C. Wong, T.W. Yuen, K.W. Fung, J.M. Leung, Optimizing timetable synchronization for rail mass transit, Transportation Science 42 (1) (2008) 57–69.

[46] J. Törnquist, J.A. Persson, N-tracked railway traffic re-scheduling during disturbances, Transportation Research Part B 41 (2007) 342–362.

[47] J. Törnquist, Railway traffic disturbance management — an experimental analysis of disturbance complexity, management objectives and limitations in planning horizon, Transportation Research Part A 41 (2007) 249–266.

[48] M. Aronsson, M. Bohlin, P. Kreuger, MILP formulations of cumulative constraints for railway scheduling — a comparative study, in: J. Clausen, G.D. Stefano (Eds.), 9th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems, ATMOS '09, Dagstuhl Publishing, Copenhagen, 2009.

[49] T. Dollevoet, M. Schmidt, A. Schöbel, Delay management including capacities of stations, in: Alberto Caprara, Spyros Kontogiannis (Eds.), 11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems, in: OpenAccess Series in Informatics (OASIcs), vol. 20, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2011, pp. 88–99.
[50] F. Corman, A. D'Ariano, D. Pacciarelli, M. Pranzo, Optimal inter-area coordination of train rescheduling decisions, Transportation Research Part E 48 (1) (2012) 71–88.
[51] C. Goh, A. Mees, Optimal control on a graph with application to train scheduling problems, Mathematical and Computer Modelling 15 (2) (1991) 49–58.
[52] U. Brännlund, P. Lindberg, A. Nôu, J. Nilsson, Railway timetabling using lagrangian relaxation, Transportation Science 32 (4) (1998) 358–369.
[53] A. Nõu, Railway timetabling — lagrangian heuristics, trita/mat-97-os12. Tech. rep., Department of Mathematics, Royal Institute of Technology, Stockholm, August 1997.
[54] J. Rodriguez, A constraint programming model for real-time train scheduling at junctions, Transportation Research Part B 41 (2007) 231–245.
[55] A. Caprara, M. Fischetti, P. Toth, Modeling and solving the train timetabling problem, Operations Research 50 (5) (2002) 851–861.
[56] A. Caprara, M. Monaci, P. Toth, P.L. Guida, A lagrangian heuristic algorithm for a real-world train timetabling problem, Discrete Applied Mathematics 154 (2006) 738–753.
[57] R.K. Ahuja, C.B. Cunha, G. Şahin, Network models in railroad planning and scheduling, in: Harvey J. Greenberg (Ed.), Tutorials in Operations Research, Emerging Theory, Methods, and Applications, Institute for Operations Research and the Management Sciences, Hanover, MD, 2005, pp. 54–101.
[58] R. Borndörfer, M. Grötschel, S. Lukac, K. Mitusch, T. Schlechte, S. Sören, A. Tanner, An auctioning approach to railway slot allocation, Competition and Regulation in Network Industries 1 (2) (2006) 163–196.
[59] F. Fischer, C. Helmburg, J. Janßen, B. Krostitz, Towards solving very large scale train timetabling problems by lagrangian relaxation, in: M. Fischetti, P. Widmayer (Eds.), 8th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems, ATMOS '08, Dagstuhl Publishing, Karlsruhe, 2008.
[60] F. Fischer, C. Helmburg, Dynamic graph generation and dynamic rolling horizon techniques in large scale train timetabling, in: Erlebach, T., L M. (Eds.), 10th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems, ATMOS '10, Liverpool, 2008.
[61] R. Borndörfer, T. Schlechte, S. Weider, Railway track allocation by rapid branching, in: Erlebach, T., L M. (Eds.), 10th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems, ATMOS '10, Liverpool. 2008.
[62] X. Delorme, J. Rodriguez, X. Gandibleux, Heuristics for railway infrastructure saturation, Electronic Notes in Theoretical Computer Science 50 (1) (2001) 1–15. http://www.sciencedirect.com/science/journal/15710661.
[63] J. Rodriguez, X. Delorme, X. Gandibleux, Railway infrastructure saturation using constraint programming approach, in: J. Allan, R.J. Hill, C.A. Brebbia, G. Sciutto, S. Sone (Eds.), Computers in Railways VIII, Wessex Institute of Technology, WIT Press, Southampton, 2002, pp. 807–816.
[64] G. Caimi, D. Burkolter, T. Herrmann, F. Chudak, M. Laumanns, Design of a railway scheduling model for dense services, Networks and Spatial Economics 9 (2009) 25–46.
[65] G. Caimi, F. Chudak, M. Fuchsberger, M. Laumanns, R. Zenklusen, A new resource-constrained multi-commodity flow model for conflict-free train routing and scheduling, Transportation Science 45 (2) (2011) 212–227.
[66] A. Mascis, D. Pacciarelli, Job-shop scheduling with blocking and no-wait constraints, European Journal of Operational Research 143 (2002) 498–517.
[67] M. Mazzarello, E. Ottaviani, A traffic management system for real-time traffic optimisation in railways, Transportation Research Part B 41 (2007) 246–274.
[68] A. D'Ariano, D. Pacciarelli, M. Pranzo, A branch and bound algorithm for scheduling trains in a railway network, European Journal of Operational Research 183 (2007) 643–657.
[69] A. D'Ariano, Real-time train dispatching: Models, algorithms, and applications. Ph.D. Thesis, Delft University of Technology, 2008.
[70] A. D'Ariano, D. Pacciarelli, M. Pranzo, Assessment of flexible timetables in real-time traffic management of a railway bottleneck, Transportation Research Part C 16 (2008) 232–245.
[71] F. Corman, Real-time railway traffic management: Dispatching in complex, large, and busy railway networks. Ph.D. Thesis, Delft University of Technology, 2008.
[72] G. Gallo, G. Longo, S. Nguyen, S. Pallottino, Directed hypergraphs and applications, Discrete Applied Mathematics 42 (1993) 177–201.
[73] S. Harrod, Modeling network transition constraints with hypergraphs, Transportation Science 45 (1) (2011) 81–97.
[74] P. Serafini, W. Ukovich, A mathematical model for periodic scheduling problems, SIAM Journal of Discrete Mathematics 2 (4) (1989) 550–581.
[75] M.A. Odijk, A constraint generation algorithm for the construction of periodic railway timetables, Transportation Research Part B 30 (6) (1996) 455–464.
[76] K. Nachtigall, S. Voget, Minimizing waiting times in integrated fixed interval timetables by upgrading railway tracks, European Journal of Operational Research 103 (1997) 610–627.
[77] L.G. Kroon, L.W. Peeters, A variable trip time model for cyclic railway timetabling, Transportation Science 37 (2) (2003) 198–212.
[78] C. Liebchen, Periodic timetable optimization. Ph.D. Thesis, Technical University Berlin, 2006.
[79] C. Liebchen, L. Peeters, On cyclic timetabling and cycles in graphs. Tech. Rep. No. 761/2002, Technische Universität Berlin, 2002.
[80] C. Liebchen, L. Peeters, Integral cycle bases for cyclic timetabling, Discrete Optimization 6 (2009) 98–109.
[81] K. Nachtigall, J. Opitz, Solving periodic timetable optimisation problems by modulo simplex calculations, in: M. Fischetti, P. Widmayer (Eds.), 8th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems, ATMOS '08, Dagstuhl Publishing, Karlsruhe, 2008.
[82] G. Caimi, The periodic service intention as a conceptual framework for generating timetables with partial periodicity, Transportation Planning and Technology 34 (4) (2011) 323–339.
[83] G. Caimi, M. Fuchsberger, M. Laumanns, K. Schüpbach, Periodic railway timetabling with event flexibility, Networks 57 (1) (2011) 3–18.
[84] G. Caimi, Algorithmic decision support for train scheduling in a large and highly utilised railway network. Ph.D. Thesis, ETH Zurich, 2009.
[85] O. Frank, Two-way traffic on a single line, Operations Research 14 (5) (1966) 801–811.
[86] A. Cherniavsky, A program for timetable compilation by a look-ahead method, Artificial Intelligence 3 (1972) 61–76.
[87] P. Pudney, A. Wardrop, Generating train plans with problem space search, in: M. Hickman, P. Mirchandani, S. Vo (Eds.), Computer-Aided Systems in Public Transport 123, in: Lecture Notes in Economics and Mathematical Systems, vol. 123, Springer-Verlag, Berlin, 2008, pp. 195–207.
[88] A. D'Ariano, F. Corman, D. Pacciarelli, M. Pranzo, Reordering and local rerouting strategies to manage train traffic in real time, Transportation Science 42 (4) (2008) 405–419.
[89] M. Luczak, ECP: How soon? Railway Age 202 (1) (2001) 56–59.
[90] P.A. Hansen, Railroads vow to make PTC work, Trains 69 (1) (2009) 10–11.

Surveys in Operations Research and Management Science 17 (2012) 97–106

Contents lists available at SciVerse ScienceDirect

Surveys in Operations Research and Management Science

journal homepage: www.elsevier.com/locate/sorms

Review

Non-convex mixed-integer nonlinear programming: A survey

Samuel Burer [a,*], Adam N. Letchford [b]

[a] Department of Management Sciences, Tippie College of Business, University of Iowa, United States

[b] Department of Management Science, Lancaster University, United Kingdom

ARTICLE INFO

Article history:
Received 9 July 2012
Accepted 8 August 2012

ABSTRACT

A wide range of problems arising in practical applications can be formulated as Mixed-Integer Nonlinear Programs (MINLPs). For the case in which the objective and constraint functions are convex, some quite effective exact and heuristic algorithms are available. When non-convexities are present, however, things become much more difficult, since then even the continuous relaxation is a global optimization problem. We survey the literature on non-convex MINLPs, discussing applications, algorithms, and software. Special attention is paid to the case in which the objective and constraint functions are quadratic.

Contents

1. Introduction

A *Mixed-Integer Nonlinear Program* (MINLP) is a problem of the following form:

$$\min \left\{ f^0(x, y) : f^j(x, y) \leq 0 \, (j = 1, \ldots, m), \; x \in \mathbb{Z}^{n_1}_+, \; y \in \mathbb{R}^{n_2}_+ \right\},$$

where n_1 is the number of integer-constrained variables, n_2 is the number of continuous variables, m is the number of constraints, and $f^j(x, y)$ for $j = 0, 1, \ldots, m$ are arbitrary functions mapping $\mathbb{Z}^{n_1}_+ \times \mathbb{R}^{n_2}_+$ to the reals.

MINLPs constitute a very general class of problems, containing as special cases both *Mixed-Integer Linear Programs* or MILPs (obtained when the functions $f^0, \ldots, f^m$ are all linear) and *Nonlinear Programs* or NLPs (obtained when $n_1 = 0$). This generality enables

* Corresponding author. Tel.: +1 3193350931.
E-mail addresses: samuel-burer@uiowa.edu (S. Burer), A.N.Letchford@lancaster.ac.uk (A.N. Letchford).

1876-7354/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.sorms.2012.08.001

one to model a very wide range of problems, but it comes at a price: even very special kinds of MINLP usually turn out to be $\mathcal{NP}$-hard.

It is useful to make a distinction between two kinds of MINLP. If the functions $f^0, \ldots, f^m$ are all *convex*, the MINLP is itself called *convex*; otherwise it is called *non-convex*. Although both kinds of MINLP are $\mathcal{NP}$-hard in general, convex MINLPs are much easier to solve than non-convex ones, in both theory and practice.

To see why, consider the *continuous relaxation* of an MINLP, which is obtained by relaxing the integrality condition from $x \in \mathbb{Z}^n_+$ to $x \in \mathbb{R}^n_+$. In the convex case, the continuous relaxation is itself convex, and therefore likely to be tractable, at least in theory. A variety of quite effective exact solution methods for convex MINLPs have been devised based on this fact. Examples include generalized Benders' decomposition [1], branch-and-bound [2], outer approximation [3], LP/NLP-based branch-and-bound [4], the extended cutting-plane method [5], branch-and-cut [6], and the hybrid methods described in [7,8]. These methods are capable of solving instances with hundreds or even thousands of variables.

By contrast, the continuous relaxation of a non-convex MINLP is itself a global optimization problem, and therefore likely to be $\mathcal{NP}$-hard (see, e.g., [9,10]). In fact, the situation is worse than this. Several simple cases of non-convex MINLPs, including the case in which all functions are quadratic, all variables are integer constrained, and the number of variables is fixed, are known to be not only $\mathcal{NP}$-hard, but even *undecidable* [11]. We refer the reader to the excellent surveys [12,13] for details.

As it happens, all of the proofs that non-convex MINLPs can be undecidable involve instances with an unbounded feasible region. Fortunately, in practice, the feasible region is usually bounded, either explicitly or implicitly. Nevertheless, the fact remains that some relatively small non-convex MINLPs, with just tens of variables, can cause existing methods to run into serious difficulties.

Several good surveys on MINLPs are available, e.g., [14–17, 12,18]. They all cover the convex case, and some cover the non-convex case. There is even research on the *pseudo-convex* case [19], involving non-convex functions that nevertheless have convex level sets. In this survey, on the other hand, we concentrate on the non-convex case. Moreover, we pay particular attention to a special case that has attracted a great deal of attention recently, and which is also of interest to ourselves: namely, the case in which all of the nonlinear functions involved are quadratic. We note that the quadratic case actually subsumes the case when all functions f^j are polynomials, although there may be substantial overhead when expressing a polynomial program as a quadratic one (see the beginning of Section 5 for details).

The paper is structured as follows. In Section 2, we review some applications of non-convex MINLPs. In Section 3, we review the key ingredients of most exact methods, including convex under-estimating functions, separable functions, factorization of non-separable functions, and standard branching versus spatial branching. In Section 4, we then show how these ingredients have been used in a variety of exact and heuristic methods for general non-convex MINLPs. Next, in Section 5, we cover the literature on the quadratic and polynomial cases. In Section 6, we list some of the available software packages, and, in Section 7, we end the survey with a few brief conclusions and topics of current and future research.

2. Applications

Many important practical problems are naturally modeled as non-convex MINLPs. We list a few examples here and recommend the references provided for further details and even more applications.

The field of chemical engineering gives rise to a plethora of non-convex MINLPs. Indeed, some of the first and most influential research in MINLPs has occurred in this field. For example, Grossmann and Sargent [20] discuss the design of chemical plants that use the same equipment "in different ways at different times". Misener and Floudas [21] survey the so-called pooling problem, which investigates how best to blend raw ingredients in pools to form the desired output. Luyben and Floudas [22] analyze the simultaneous design and control of a process, and Yee and Grossmann [23] examine heat exchanger networks in which heat from one process is used by another. See Floudas [24] and Misener and Floudas [25] for comprehensive lists of references of MINLPs arising in chemical engineering.

Another important source of non-convex MINLPs is network design. This includes, for example, water [26], gas [27], energy [28], and transportation [29] networks.

Non-convex MINLPs arise in other areas of engineering as well. These include avoiding trim-loss in the paper industry [30], airplane boarding [31], oil-spill response planning [32], ethanol supply chains [33], concrete structure design [34], and load-bearing thermal insulation systems [35]. There are also medical applications, such as seizure prediction [36].

Adams and Sherali [37] and Freire et al. [38] discuss applications of MINLPs with non-convex bilinear objective functions in production planning, facility location, distribution, and marketing.

Finally, many standard and well-studied optimization problems, each with its own selection of applications, can also be viewed quite naturally as non-convex MINLPs. These include, for example, maximum cut (or binary quadratic programming (QP)) and its variants [39–41], clustering [42], non-convex QP with binary variables [43], quadratically constrained QP [44], the quadratic traveling salesman problem (TSP) [45], TSP with neighborhoods [46], and polynomial optimization [47].

3. Key concepts

In this section, some key concepts are presented, which together form the main ingredients of all existing exact algorithms (and some heuristics) for non-convex MINLPs.

3.1. Under- and over-estimators

As mentioned above, even solving the continuous relaxation of a non-convex MINLP is unlikely to be easy. For this reason, a further relaxation step is usual. One way to do this is to replace each non-convex function $f^j(x, y)$ with a *convex under-estimating function*, i.e., a convex function $g^j(x, y)$ such that $g^j(x, y) \leq f^j(x, y)$ for all (x, y) in the domain of interest. Another way is to define a new variable, say z^j, which acts as a place holder for $f^j(x, y)$, and to add constraints which force z^j to be approximately equal to $f^j(x, y)$. In this latter approach, one adds constraints of the form $z^j \geq g^j(x, y)$, where $g^j(x, y)$ is again a convex under-estimator. One can also add constraints of the form $z^j \leq h^j(x, y)$, where $h^j(x, y)$ is a *concave over-estimating* function. If one wishes to solve the convex relaxation using an LP solver, rather than a general convex programming solver, one must use linear under- and over-estimators.

For some specific functions, and some specific domains, one can characterize the so-called *convex* and *concave envelopes*, which are the tightest possible convex under-estimator and concave over-estimator. A classical example, due to McCormick [48], concerns the quadratic function y_1y_2, over the rectangular domain defined by $\ell_1 \leq y_1 \leq u_1$ and $\ell_2 \leq y_2 \leq u_2$. If z denotes the additional variable, the convex envelope is defined by the two linear inequalities $z \geq \ell_2 y_1 + \ell_1 y_2 - \ell_1\ell_2$ and $z \geq u_2 y_1 + u_1 y_2 - u_1 u_2$, and the concave envelope by $z \leq u_2 y_1 + \ell_1 y_2 - \ell_1 u_2$ and $z \leq \ell_2 y_1 + u_1 y_2 - u_1\ell_2$. In this case, both envelopes are defined using only linear constraints.

Many other examples of under- and over-estimating functions, and convex and concave envelopes, have appeared in the literature. See the books by Horst and Tuy [49] and Tawarmalani and Sahinidis [10] for details.

We mention another important paper, that of Androulakis et al. [50]. Their approach constructs convex under-estimators of general, twice-differentiable, non-convex functions whose domain is a box (also known as a hyper-rectangle). The basic idea is to add a convex quadratic term that takes the value zero on the corners of the box, and the choice of the quadratic term is governed by a vector $\alpha \geq 0$. For example, suppose the function $f(x)$ is defined on $B := \{x : 0 \leq x \leq e\}$, where e is the all ones vector, and let $\text{Diag}(\alpha)$ be the diagonal matrix having α as its diagonal. Then $f(x)$ is under-estimated by

$$f_\alpha(x) := f(x) + x^T \text{Diag}(\alpha) x - \alpha^T x,$$

since $x^T \text{Diag}(\alpha) x - \alpha^T x \leq 0$ for all $x \in B$ (and in fact equals zero on the corners of B). If α is chosen large, then $f_\alpha(x)$ will also be convex because its Hessian will be dominated by $\text{Diag}(\alpha)$. On the other hand, as α increases, the quality of the resulting under-estimation by $f_\alpha(x)$ worsens, so the choice of α is critical.

3.2. Separable functions

A function $f(x, y)$ is said to be *separable* if there exist functions $g(x_i)$ for $i = 1, \ldots, n_1$ and functions $h(y_i)$ for $i = 1, \ldots, n_2$ such that

$$f(x, y) = \sum_{i=1}^{n_1} g(x_i) + \sum_{i=1}^{n_2} h(y_i).$$

Separable functions are relatively easy to handle in two ways. First, if one has a useful convex under-estimator for each of the individual functions $g(x_i)$ and $h(y_i)$, the sum of those individual under-estimators is an under-estimator for $f(x, y)$. The same applies to concave over-estimators. Second, even if one does not have useful under- or over-estimators, one can use the following approach, due to Beale [51] and Tomlin [52].

1. Approximate each of the functions $g(x_i)$ and $h(y_i)$ with a piece-wise linear function.
2. Introduce new continuous variables, g_i and h_i, representing the values of these functions.
3. Add one binary variable for each 'piece' of each piece-wise linear function.
4. Add further binary variables, along with linear constraints, to ensure that the variables g_i and h_i take the correct values.

In this way, any non-convex MINLP with separable functions can be approximated by an MILP.

3.3. Factorization

If an MINLP is not separable, and it contains functions for which good under- or over-estimators are not available, one can often apply a process called *factorization*, also due to McCormick [48]. Factorization involves the introduction of additional variables and constraints, in such a way that the resulting MINLP involves functions of a simpler form.

Rather than presenting a formal definition, we give an example (see [53] for more details). Suppose an MINLP contains the (non-linear and non-convex) function $f(y_1, y_2, y_3) = \exp\left(\sqrt{y_1 y_2 + y_3}\right)$, where y_1, y_2, y_3 are continuous and non-negative variables. If one introduces new variables w_1, w_2, and w_3, along with the constraints $w_1 = \sqrt{w_2}$, $w_2 = w_3 + y_3$, and $w_3 = y_1 y_2$, one can rewrite the function f as $\exp(w_1)$. Then, one needs under- and over-estimators only for the relatively simple functions $\exp(w_1)$, $\sqrt{w_2}$, and $y_1 y_2$.

3.4. Branching: standard and spatial

The branch-and-bound method for MILPs, usually attributed to Land and Doig [54], is well known. The key operation, called *branching*, is based on the following idea. If an integer-constrained variable x_i takes a fractional value x_i^* in the optimal solution to the continuous relaxation of a problem, then one can replace the problem with two subproblems. In one of the subproblems, the constraint $x_i \leq \lfloor x_i^* \rfloor$ is added, and, in the other, the constraint $x_i \geq \lceil x_i^* \rceil$ is added. Clearly, the solution to the original relaxation is not feasible for either of the two subproblems.

In the global optimization literature, one branches by partitioning the domain of *continuous* variables. Typically, this is done by taking a continuous variable y_i, whose current domain is $[\ell_i, u_i]$, choosing some value β with $\ell_i < \beta < u_i$, and creating two subproblems, one with domain $[\ell_i, \beta]$ and the other with domain $[\beta, u_i]$. In addition, when solving either of the subproblems, one can replace the original under- and over-estimators with stronger ones, which take advantage of the reduced domain. This process, called 'spatial' branching, is necessary for two reasons: (i) the optimal solution to the relaxation may not be feasible for the original problem, and (ii) even if it is feasible, the approximation of the cost function in the relaxation may not be sufficiently accurate. Spatial branching is also due to McCormick [48].

We illustrate spatial branching with an example. Suppose that the continuous variable y_1 is known to satisfy $0 \leq y_1 \leq u_i$ and that, in the process of factorization, we have introduced a new variable z_i, representing the quadratic term y_i^2. If we intended to use a general convex programming solver, we could obtain a convex relaxation by appending the constraints $z_i \geq y_i^2$ and $z_i \leq u_i y_i$, as shown in Fig. 1(a). If, on the other hand, we preferred to use an LP solver, we could add instead the constraints $z_i \geq 0, z_i \geq u_i^2 - 2u_i y_i$, and $z_i \leq u_i y_i$, as shown in Fig. 1(b).

Now, suppose the solution of the relaxation is not feasible for the MINLP, and we decide to branch by splitting the domain of y_1 into the intervals $[0, \beta]$ and $[\beta, u_i]$. Also suppose for simplicity that we are using LP relaxations. Then, in the left branch we can tighten the relaxation by adding $\beta^2 - 2\beta y_i \leq z_i \leq \beta y_i$, while in the right branch we can add $\beta y_i \leq z_i \leq u_i y_i$ (see Fig. 2(a) and (b)).

Since MINLPs contain both integer-constrained and continuous variables, one is free to apply both standard branching or spatial branching where appropriate. Moreover, even if one applies standard branching, one may still be able to tighten the constraints in each of the two subproblems.

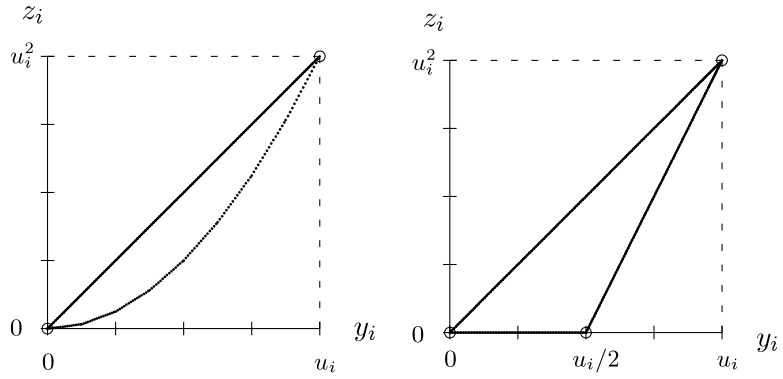
4. Algorithms for the general case

Now that we are armed with the concepts described in the previous section, we can go on to survey specific algorithms for general non-convex MINLPs.

4.1. Spatial branch-and-bound

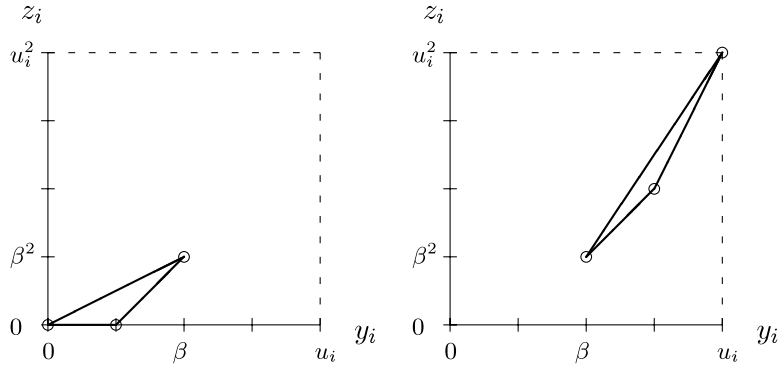
Branching, whether standard or spatial, usually has to be applied recursively, leading to a hierarchy of subproblems. As in the branch-and-bound method for MILPs [54], these subproblems can be viewed as being arranged in a tree structure, which can be searched in various ways. A subproblem can be removed from further consideration (also known as *fathomed* or *pruned*) under three conditions: (i) it is feasible for the original problem and its cost under the relaxed objective equals it true cost (to within some specified tolerance), (ii) the associated lower bound is no better than the best upper bound found so far, or (iii) it is infeasible.

This overall approach was first proposed by McCormick [48] in the context of global optimization problems. Later on, several authors (mostly from the chemical process engineering community) realized that the approach could be applied just as well to problems



(a) Convex. (b) Linear.

Fig. 1. Convex and linear approximations of the function $z_i = y_i^2$ over the domain $[0, u_i]$.



(a) Left. (b) Right.

Fig. 2. Improved linear approximations after spatial branching.

with integer variables. See, for example, Smith and Pantelides [55] or Lee and Grossmann [56].

4.2. Branch-and-reduce

A major step forward in the exact solution of non-convex MINLPs was the introduction of the *branch-and-reduce* technique by Ryoo and Sahinidis [57,58]. This is an improved version of spatial branch-and-bound in which one attempts to reduce the domains of the variables, beyond the reductions that occur simply as a result of branching. More specifically, one adds the following two operations: (i) before a subproblem is solved, its constraints are checked to see whether the domain of any variables can be reduced without losing any *feasible* solutions; (ii) after the subproblem is solved, sensitivity information is used to see whether the domain of any variables can be reduced without losing any *optimal* solutions.

After domain reduction has been performed, one can then generate better convex under-estimators. This in turn enables one to tighten the constraints, which can lead to improved lower bounds. The net effect is usually a drastic decrease in the size of the enumeration tree.

Branch-and-reduce is usually performed using LP relaxations, rather than more complex convex programming relaxations, due to two important facts. First and foremost, LPs can be solved more efficiently and with greater numerical stability. Second, sensitivity information is more readily available (and easier to interpret) in the case of LPs.

Tawarmalani and Sahinidis [59,60] added some further refinements to this scheme. In [59], a unified framework is given for domain reduction strategies, and, in [60], it is shown that, even when a constraint is convex, it may be helpful (in terms of tightness of the resulting relaxation) to introduce additional variables and split the constraint into two constraints. Some further enhanced rules for domain reduction, branching variable selection, and branching value have also been given by Belotti et al. [61].

4.3. α-branch-and-bound

Androulakis et al. [50] proposed an exact spatial branch-and-bound algorithm for global optimization of non-convex NLPs in which all functions involved are twice differentiable. This method, called α-BB, is based on their general technique for constructing under-estimators, which was mentioned in Section 3.1. In Adjiman et al. [62,63], the algorithm was improved by using tighter and more specialized under-estimators for constraints that have certain specific structures, and reserving the general technique only for constraints that do not have any of those structures. Later on, Adjiman et al. [64] extended the α-BB method to the mixed-integer case.

One advantage that α-BB has, with respect to the more traditional spatial branch-and-bound approach, or indeed branch-and-reduce, is that usually no additional variables are needed. That is to say, one can often work with the original objective and constraint functions, without needing to resort to factorization. This is because the under-estimators used do not rely on functions being factored. On the other hand, to solve the relaxations, one needs a general convex programming solver, rather than an LP solver.

4.4. Conversion to an MILP

Another approach that one can take is to factorize the problem (if necessary) as described in Section 3.3, approximate the resulting

separable MINLP by an MILP as described in Section 3.2, and then solve the resulting MILP using any available MILP solver. To our knowledge, this approach was first suggested by Beale and Tomlin [65]. The conversion into an MILP leads to sets of binary variables with a certain special structure. Beale and Tomlin call these sets *special ordered sets of type* 2 (*SOS2*), and propose a specialized branching rule. This branching rule is now standard in most commercial and academic MILP solvers. Beale and Forrest [66] discuss a method for updating the MILP approximations dynamically and an improved branching strategy for the SOS2 variables.

Keha et al. [67] compare several different ways of modeling piece-wise linear functions (PLFs) using binary variables. In their follow-up paper [68], the authors present a branch-and-cut algorithm that uses the SOS approach in conjunction with strong valid inequalities. Vielma and Nemhauser [69] also present an elegant way to reduce the number of auxiliary binary variables required for modeling PLFs.

A natural way to generalize this approach is to construct PLFs that approximate functions of more than one variable. (In fact, this was already suggested by Beale [70] and Tomlin [52] in the context of non-convex NLPs.) A recent exploration of this idea was conducted by Martin et al. [27]. As well as constructing such PLFs, they also propose adding cutting planes to tighten the relaxation. Geißler et al. [71] is another recent reference.

Leyffer et al. [72] show that the naive use of PLFs can lead to an infeasible MILP, even when the original MINLP is clearly feasible. They propose a modified approach, called 'branch-and-refine', in which piece-wise-linear under- and over-estimators are constructed. This ensures that all of the original feasible solutions for the MINLP remain feasible for the MILP. Also, instead of branching spatially or on special ordered sets, they branch in the classical way. Finally, they refine the PLFs each time a subproblem is constructed.

4.5. Some other exact approaches

For completeness, we mention a few other exact approaches.

- Benson and Erenguc [73] and Bretthauer et al. [74] present exact algorithms for MINLPs with linear constraints and a concave objective function. Their algorithms use LP relaxations, specialized penalty functions, and cutting planes that are similar to the well-known concavity cuts of Tuy [75].
- Kesavan et al. [76] present special techniques for MINLPs in which separability occurs at the level of the vectors x and y, i.e., the functions $f^j(x, y)$ can be expressed as $g^j(x) + h^j(y)$. In fact, the authors assume that the functions $h^j(\cdot)$ are linear and that y is binary.
- Karuppiah and Grossman [77] use Lagrangian decomposition to generate lower bounds and cutting planes for general non-convex MINLPs.
- D'Ambrosio et al. [78] present an exact algorithm for MINLPs in which the non-convexities are solely manifested as the sum of non-convex univariate functions. In this sense, while the whole problem is not necessarily separable, the non-convexities are. Their algorithm, called `SC-MINLP`, involves an alternating sequence of convex MINLPs and non-convex NLPs.

4.6. Heuristics

All of the methods mentioned so far in this section have been *exact* methods. To close this section, we mention some *heuristic* methods, i.e., methods designed to find good, but not provably optimal, solutions quickly.

It is sometimes possible to convert exact algorithms for convex MINLPs into heuristics for non-convex MINLPs. Leyffer [79] does this using an MINLP solver that combines branch-and-bound with sequential quadratic programming. Nowak and Vigerske [80] do so by using quadratic under- and over-estimators of all nonlinear functions, together with an exact solver for convex all-quadratic problems.

Other researchers have adapted classical heuristic (and meta-heuristic) approaches, normally applied to 0–1 LPs, to the more general case of non-convex MINLPs. For example, Exler et al. [81] present a heuristic, based on tabu search, for certain non-convex MINLP instances arising in integrated systems and process control design. A particle-swarm optimization for MINLPs is presented in [82], [83] studies an enhanced genetic algorithm, and [84] considers an ant-colony approach. Two recent examples are that of Liberti et al. [85], whose approach involves the integration of variable neighborhood search, local branching, sequential quadratic programming, and branch-and-bound, and that of Berthold [86], who conducts large neighborhood local search by rounding the fractional solution from a relaxation. Finally, D'Ambrosio et al. [87] and Nannicini and Belotti [88] have recently presented heuristics that involve the solution of an alternating sequence of NLPs and MILPs.

5. The quadratic case (and beyond)

In this section, we focus on the case in which all of the nonlinear objective and constraint functions are *quadratic*. This case has received much attention, not only because it is the most natural generalization of the linear case, but also because it has a very wide range of applicability. Indeed, all MINLPs involving *polynomials* can be reduced to quadratic MINLPs by using additional constraints and variables (e.g., the cubic constraint $y_2 = y_1^3$ can be reduced to the quadratic constraints $y_2 = y_1 w$ and $w = y_1^2$, where w is an additional variable). The papers [89,90] provide further discussion of such transformations. Moreover, even functions that are not polynomials can often be well approximated by quadratic functions in the domain of interest.

5.1. Quadratic optimization with binary variables

The simplest quadratic MINLPs are those in which all variables are binary. The literature on such problems is vast, and several different approaches have been suggested for tackling them. Among them, we mention the following.

- A seminal result due to Fortet [91] (see also [92,93]) is that a quadratic function of n binary variables can be *linearized* by adding $\mathcal{O}(n^2)$ additional variables and constraints. More precisely, any term of the form $x_i x_j$, with $i \neq j$, can be replaced with a new binary variable x_{ij}, along with constraints of the form $x_{ij} \leq x_i$, $x_{ij} \leq x_j$, and $x_{ij} \geq x_i + x_j - 1$. Note the match with McCormick's approximation of the function $y_i y_j$ in the continuous case, mentioned in Section 3.1.
- Glover [94] showed that, in fact, one can linearize such functions using only $\mathcal{O}(n)$ additional variables and constraints. See, e.g., [95,96] for related formulations. Chaovalitwongse et al. [97] and Sherali and Smith [98] provide recent, conceptually different $O(n)$ linearization approaches.
- Hammer and Rubin [99] showed that non-convex quadratic functions in binary variables can be *convexified* by adding or subtracting appropriate multiples of terms of the form $x_i^2 - x_i$ (which equal zero when x_i is binary). This approach was improved by Körner [100].
- Hammer et al. [101] present a bounding procedure, called the *roof dual*, which replaces each quadratic function with a tight linear under-estimator. Extensions of this are surveyed in Boros and Hammer [102].

- Pardalos and Rodgers [103] solve unconstrained 0–1 QPs within a branch-and-bound algorithm involving careful preprocessing and computational efficiencies.
- Poljak and Wolkowicz [104] examine several bounding techniques for unconstrained 0–1 QPs, and show that they all give the same bounds.
- Caprara [105] shows how to compute good bounds efficiently using Lagrangian relaxation, when the linearized version of the problem can be solved efficiently.

There are three other well-known approaches, that are not only highly effective, but can be adapted to quadratic problems that have a mixture of binary, integer-constrained, and/or continuous variables. These are discussed in the following three subsections.

5.2. The reformulation-linearization technique (RLT)

In their seminal 1986 paper, Adams and Sherali [106] proposed the following approach to 0–1 quadratic programs. First, the additional x_{ij} variables are introduced, along with the constraints of Fortet [91] mentioned in the previous subsection. Next, new valid constraints are derived as follows.

- Each linear inequality, say $a^T x \leq b$, is multiplied by each variable in turn, to obtain n valid quadratic inequalities of the form $(a^T x)x_k \leq bx_k$. Replacing each product of the form $x_i x_k$ with the single variable x_{ik}, and replacing x_k^2 with x_k, one obtains the following valid linear inequalities:

$$\sum_{i \neq k} a_i x_{ik} \leq (b - a_k)x_k \quad (k = 1, \ldots, n).$$

- Similarly, multiplying each linear inequality by terms of the form $1 - x_k$, one obtains n more valid quadratic inequalities of the form $(a^T x)(1 - x_k) \leq b(1 - x_k)$. This yields the linear inequalities:

$$\sum_{i \neq k} a_i (x_i - x_{ik}) \leq (b - a_k)(1 - x_k) \quad (k = 1, \ldots, n).$$

  The original linear inequalities can then be discarded, as they are implied by the new ones.
- Next, each linear equation, say $c^T x = d$, is multiplied by each variable in turn, to obtain n valid quadratic equations. Linearizing as usual, one obtains

$$\sum_{i \neq k} c_i x_{ik} = (d - c_k)x_k \quad (k = 1, \ldots, n).$$

  Unlike in the case of inequalities, there is no need to multiply equations by $1 - x_k$, since the resulting equations would be implied by the original equations and the new ones.

Later on, Sherali and Adams [107] realized that, if the above procedure is applied to a 0–1 *linear* program, the continuous relaxation of the transformed instance is stronger than that of the original instance. They also showed that one could obtain a hierarchy of increasingly stronger relaxations, by introducing variables representing products of three variables, products of four variables, and so on. They named the entire scheme the *Reformulation-Linearization Technique* (RLT).

Since then, the RLT has been extended to cover several other classes of convex and non-convex MINLPs, beyond pure 0–1 linear and quadratic problems. We will mention some of these extensions in Sections 5.5 and 5.6, but, for a full treatment, the reader is referred to the book [9].

5.3. Semidefinite relaxation

Another popular approach for generating strong relaxations of non-convex quadratic optimization (and other) problems is based on *semidefinite programming* (SDP). The starting point of this approach is as follows. Given an arbitrary vector $x \in \mathbb{R}^n$ of decision variables, define the matrix $X = xx^T$. Note that a quadratic function of x is a linear function of X. Therefore, any optimization problem involving quadratic functions can be reformulated as an optimization problem involving linear functions, together with the single non-convex constraint $X = xx^T$.

Now, note that X is real, symmetric and *positive semidefinite* (psd), and that, for $1 \leq i \leq j \leq n$, the entry X_{ij} represents the product $x_i x_j$ (and is thus analogous to the term x_{ij} in Sections 5.1 and 5.2). Moreover, as pointed out in [108,109], the augmented matrix

$$\hat{X} := \begin{pmatrix} 1 \\ x \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix}^T = \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix}$$

is also psd. This fact enables one to construct useful SDP relaxations of various quadratic optimization problems (e.g., [110–112,109,113,114,108]).

Note that, for a Mixed-Integer Quadratic Program (i.e., an MINLP with a quadratic objective but linear constraints), one can easily combine the RLT and SDP, to obtain a relaxation that dominates those obtained by using either technique alone. Anstreicher [115] shows that this can yield significant benefits in terms of bound strength, though running times can be high.

Buchheim and Wiegele [116] use SDP relaxations and a tailored branching scheme for a special kind of Mixed-Integer Quadratic Program, in which the only constraints present are ones that enforce each variable to belong to a specified subset of $\mathbb{R}$. Note that this includes unconstrained problems with any mixture of continuous, binary and general-integer variables.

A *completely positive* matrix is one that can be factored as NN^T, where N is a component-wise nonnegative matrix. Clearly, if $x \in \mathbb{R}^n_+$, then $\hat{X}$ is completely positive rather than merely psd. One can use this fact to derive even stronger SDP relaxations; see the survey [117]. Chen and Burer [118] use such an approach within branch-and-bound to solve non-convex QPs having continuous variables and linear constraints.

5.4. Polyhedral theory and convex analysis

We have seen, in the previous three subsections, that a popular way to tackle quadratic MINLPs is to introduce new variables representing products of pairs of original variables. Once this has been done, it is natural to study the convex hull of feasible solutions, in the hope of deriving strong linear (or at least convex) relaxations.

Padberg [40] tackled exactly this topic when he introduced a polytope associated with unconstrained 0–1 quadratic programs, which he called the *Boolean quadric polytope*. The Boolean quadric polytope of order n is defined as

$$\mathrm{BQP}_n = \mathrm{conv}\left\{x \in \{0,1\}^{n+\binom{n}{2}} : x_{ij} = x_i x_j \ (1 \leq i < j \leq n)\right\}.$$

Note that here, just as in the original version of the RLT, the variable x_{ij} is not defined when $i = j$. This is because squaring a binary variable has no effect.

Padberg [40] derived various valid and facet-defining inequalities for BQP_n, called *triangle*, *cut*, and *clique* inequalities. Since then, a wide variety of valid and facet-defining inequalities have been discovered. These are surveyed in the book by Deza and Laurent [119].

There are several other papers on polytopes related to quadratic versions of traditional combinatorial optimization problems. Among them, we mention [120] on the quadratic assignment polytope, [121] on the quadratic semi-assignment polytope, and [111] on the quadratic knapsack polytope. Padberg and Rijal [122] studied several quadratic 0–1 problems in a common framework.

There are also three papers on the following (non-polyhedral) convex set [123–125]:

$$\text{conv}\left\{x \in [0,1]^n,\ y \in \mathbb{R}^{\binom{n+1}{2}},\ x_{ij} = x_i x_j\ (1 \le i \le j \le n)\right\}.$$

This convex set is associated with non-convex quadratic programming with box constraints, a classical problem in global optimization. Burer and Letchford [124] use a combination of polyhedral theory and convex analysis to analyze this convex set. In a follow-up paper, Burer and Letchford [126] apply the same approach to the case in which there are unbounded continuous and integer variables.

Complementing the above approaches, several researchers have looked at the convex hull of sets of the form $\{(z, x) \in \mathbb{R}^{n+1} : z = q(x), x \in D\}$, where $q(x)$ is a given quadratic function and D is a bounded (most often simple) domain [127–129]. While slightly less general than convexifying in the space of all pairs x_{ij} as done above, this approach much more directly linearizes and convexifies the quadratics of interest in a given problem. It can also be effectively generalized to the non-quadratic case (see, for example, Section 2 of [53]).

5.5. Some additional techniques

Saxena et al. [130,131] have derived strong cutting planes for non-convex MIQCQPs (mixed-integer quadratically constrained quadratic programs). In [130], the cutting planes are derived in the extended quadratic space of the X_{ij} variables, using disjunctions of the form $(a^T x \le b) \vee (a^T x \ge b)$. In [131], the cutting planes are derived in the original space by projecting down certain relaxations from the quadratic space. See also the recent survey Burer and Saxena [132]. Separately, Galli et al. [133] have adapted the 'gap inequalities', originally defined in [134] for the max-cut problem, to non-convex MIQPs.

Berthold et al. [135] present an exact algorithm for MIQCQPs that is based on the integration of constraint programming and branch-and-cut. The key is to use quadratic constraints to reduce domains, wherever possible. Misener and Floudas [25] present an exact algorithm for non-convex mixed 0–1 QCQPs that is based on branch-and-reduce, together with cutting planes derived from the consideration of polyhedra involving small subsets of variables.

Billionnet et al. [136] revisit the approach for 0–1 quadratic programs, mentioned in Section 5.1, due to Hammer and Rubin [99] and Körner [100]. They show that an optimal reformulation can be derived from the dual of an SDP relaxation. Billionnet et al. [137] then show that the method can be extended to general MIQPs, provided that the integer-constrained variables are bounded and the part of the objective function associated with the continuous variables is convex.

Adams and Sherali [37] and Freire et al. [38] present algorithms for *bilinear* problems. A *bilinear* optimization problem is one in which all constraints are linear, and the objective function is the product of two linear functions (and therefore quadratic). The paper [37] is concerned with the case in which one of the linear functions involves binary variables and the other involves continuous variables. The paper [38], on the other hand, is concerned with the case in which all variables are integer-constrained.

Finally, we mention that Nowak [138] proposes using Lagrangian decomposition for non-convex MIQCQPs.

5.6. Extensions to polynomial optimization

Many researchers have extended ideas from quadratic programs to the much broader class of polynomial optimization problems. A simple way to linearize polynomials involving binary variables was given by Glover and Woolsey [92]. The RLT approach of Sherali and Adams [9] explained in Section 5.2 creates a hierarchy of ever-tighter LP relaxations of polynomial problems. Some successful applications of the RLT approach include the solution of 0–1 polynomial programs [107], mixed-integer polynomial programs [139], and mixed-discrete problems having non-convex polynomial constraints and general convex constraints [140].

Recently, some sophisticated approaches have been developed for mixed 0–1 polynomial programs that draw on concepts from real algebraic geometry, commutative algebra, and moment theory. Relevant works include Nesterov [141], Parrilo [142], Lasserre [47], Laurent [143], and De Loera et al. [144]. The method of Lasserre [145] works for integer polynomial programs when each variable has an explicit lower and upper bound.

Michaels and Weismantel [146] make an important observation for Integer Polynomial Programming. They note that, given a non-convex polynomial, say $f(x)$, there may exist a convex polynomial, say $f'(x)$, that achieves the same value as $f(x)$ at all integer points. In principle, this could allow such non-convex programs to be made convex.

6. Software

There are five software packages that can solve non-convex MINLPs to proven optimality:

`BARON`, α-`BB`, `LINDO-Global`, `Couenne`, and `GloMIQO`.

`BARON` is due to Sahinidis and colleagues [57,58,10], α-`BB` is due to Adjiman et al. [64], and `LINDO-Global` is described in Lin and Schrage [147]. `Couenne` is due to Belotti et al. [43], and GloMIQO [148] relates to the technique of Misener and Floudas [25] described in Section 5.5.

Some packages can be used to find *heuristic* solutions for non-convex MINLPs:

`BONMIN`, `DICOPT`, `LaGO`, and `MIDACO`.

The first three are actually packages for convex MINLPs, while the fourth is based on ant-colony optimization. The algorithmic approach behind `BONMIN` is described in [8], and `DICOPT` has been developed by Grossmann and co-authors (e.g., Kocis and Grossmann [149]). `LaGO` is described in Nowak and Vigerske [80], and `MIDACO` is presented in [84] and available at `midaco-solver.com`.

The package due to Liberti et al. [85], described in Section 4.6, is called *RECIPE*. The paper by Berthold et al. [135] presents an MIQCP solver for the software package `SCIP`. Finally, `GloptiPoly` [150] can solve general polynomial optimization problems.

7. Conclusions

Because non-convex MINLPs encompass a huge range of applications and problem types, the depth and breadth of techniques used to solve them should come as no surprise. In this survey, we have tried to give a fair and up-to-date introduction to these techniques.

Without a doubt, substantial successes in the fields of MILP and global optimization have played critical roles in the development of algorithms for non-convex MINLPs, and we suspect further successes will have continued benefits for MINLPs. We believe, also, that even more insights can be achieved by studying MINLPs specifically. For example, analyzing — and generating cutting planes for — the various convex hulls that arise in MINLPs (see Section 5.4) will require aspects of both polyhedral theory and convex analysis to achieve best results.

We also advocate the development of algorithms for various special cases of non-convex MINLPs. While general-purpose

algorithms for MINLPs are certainly needed, since MINLP are so broad, there will always be a need for handling important special cases. Special cases can also allow the development of newer techniques (e.g., semidefinite relaxations), which may then progress to more general techniques.

Finally, we believe there will be an increasing place for heuristics and approximation algorithms for non-convex MINLPs. Most techniques so far aim for globally optimal solutions, but in practice it would be valuable to have sophisticated approaches for finding near-optimal solutions.

Acknowledgments

The first author was supported by the National Science Foundation under grant CCF-0545514, and the second author was supported by the Engineering and Physical Sciences Research Council under grant EP/D072662/1.

The authors are in debt to two anonymous referees whose comments and suggestions greatly improved the paper.

References

[1] A.M. Geoffrion, Generalized benders decomposition, J. Optim. Theory Appl. 10 (1972) 237–260.
[2] O.K. Gupta, V. Ravindran, Branch and bound experiments in convex nonlinear integer programming, Management. Sci. 31 (1985) 1533–1546.
[3] M.A. Duran, I.E. Grossmann, An outer-approximation algorithm for a class of mixed-integer nonlinear programs, Math. Program. 36 (1986) 307–339.
[4] I. Quesada, I.E. Grossmann, An LP/NLP based branch and bound algorithm for convex MINLP optimization problems, Comput. Chem. Eng. 16 (1992) 937–947.
[5] T. Westerlund, F. Pettersson, A cutting plane method for solving convex MINLP problems, Comput. Chem. Eng. 19 (1995) S131–S136.
[6] R.A. Stubbs, S. Mehrotra, A branch-and-cut method for 0–1 mixed convex programming, Math. Program. 86 (1999) 515–532.
[7] K. Abhishek, S. Leyffer, J.T. Linderoth, FilMINT: an outer-approximation-based solver for nonlinear mixed integer programs, INFORMS J. Comput. 22 (2010) 555–567.
[8] P. Bonami, L.T. Biegler, A.R. Conn, G. Cornuéjols, I.E. Grossmann, C.D. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, An algorithmic framework for convex mixed integer nonlinear programs, Discrete Optim. 5 (2008) 186–204.
[9] H.D. Sherali, W.P. Adams, A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems, Kluwer, Dordrecht, 1999.
[10] M. Tawarmalani, N.V. Sahinidis, Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming, Springer, 2003.
[11] R. Jeroslow, There cannot be any algorithm for integer programming with quadratic constraints, Oper. Res. 21 (1) (1973) 221–224.
[12] R. Hemmecke, M. Köppe, J. Lee, R. Weismantel, Nonlinear integer programming, in: M. Jünger, et al. (Eds.), 50 Years of Integer Programming 1958–2008, Springer, Berlin, 2010, pp. 561–618.
[13] M. Köppe, On the complexity of nonlinear mixed-integer optimization, in: J. Lee, S. Leyffer (Eds.), Mixed-Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 533–558.
[14] P. Bonami, M. Kilinç, J. Linderoth, Algorithms and software for convex mixed integer nonlinear programs, in: J. Lee, S. Leyffer (Eds.), Mixed Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 1–40.
[15] C. D'Ambrosio, A. Lodi, Mixed integer nonlinear programming tools: a practical overview, 4OR 9 (2011) 329–349.
[16] C.A. Floudas, C.E. Gounaris, A review of recent advances in global optimization, J. Global Optim. 45 (2009) 3–38.
[17] I.E. Grossmann, Review of nonlinear mixed-integer and disjunctive techniques, Optim. Eng. 3 (2002) 227–252.
[18] S. Leyffer, J.T. Linderoth, J. Luedtke, A. Miller, T. Munson, Applications and algorithms for mixed integer nonlinear programming, J. Phys.: Conf. Ser. 180 (2009) 012–014.
[19] R. Pörn, T. Westerlund, A cutting plane method for minimizing pseudo-convex functions in the mixed integer case, Comput. Chem. Eng. 24 (2000) 2655–2665.
[20] I.E. Grossmann, R.W.H. Sargent, Optimum design of multipurpose chemical plants, Ind. Eng. Chem. Process Des. Dev. 18 (1979) 343–348.
[21] R. Misener, C.A. Floudas, Advances for the pooling problem: modeling, global optimization, and computational studies (survey), Appl. Comput. Math. 8 (2009) 3–22.
[22] M. Luyben, C.A. Floudas, A multiobjective optimization approach for analyzing the interaction of design and control, part 1: theoretical framework and binary distillation synthesis, Comput. Chem. Eng. 18 (1994) 933–969.
[23] T. Yee, I. Grossmann, Simultaneous optimization models for heat integration II: heat exchanger network synthesis, Comput. Chem. Eng. 14 (1990) 1165–1184.
[24] C.A. Floudas, Mixed integer nonlinear programming, in: C.A. Floudas, P.M. Pardalos (Eds.), Encyclopedia of Optimization, Springer, 2009, pp. 2234–2247.
[25] R. Misener, C.A. Floudas, Global optimization of mixed-integer quadratically-constrained quadratic programs (MIQCQP) through piecewise-linear and edge-concave relaxations, Math. Program. Series B (2012) http://dx.doi.org/10.1007/s10107-012-0555-6.
[26] C. Bragalli, C. D'Ambrosio, J. Lee, A. Lodi, P. Toth, An MINLP solution method for a water network problem, in: Proc. of ESA 2006, in: Lecture Notes in Computer Science, vol. 4168, Springer, Berlin, 2006, pp. 696–707.
[27] A. Martin, M. Möller, S. Moritz, Mixed integer models for the stationary case of gas network optimization, Math. Program. 105 (2006) 563–582.
[28] W. Murray, U. Shanbhag, A local relaxation approach for the siting of electrical substations, Comput. Optim. Appl. 33 (2006) 7–49.
[29] A. Fügenschuh, H. Homfeld, H. Schülldorf, S. Vigerske, Mixed-integer nonlinear problems in transportation applications, in: H. Rodrigues et al. (eds.) Proceedings of the 2nd International Conference on Engineering Optimization.
[30] I. Harjunkoski, T. Westerlund, R. Pörn, Numerical and environmental considerations on a complex industrial mixed integer non-linear programming (MINLP) problem, Comput. Chem. Eng. 23 (1999) 1545–1561.
[31] M. van den Briel, J. Villalobos, G. Hogg, T. Lindemann, A. Mule, America west airlines develops efficient boarding strategies, Interfaces 35 (2005) 191–201.
[32] F. You, S. Leyffer, Mixed-integer dynamic optimization for oil-spill response planning with integration of a dynamic oil weathering model, AIChE J. 57 (2011) 3555–3564.
[33] G. Corsano, A.R. Vecchietti, J.M. Montagna, Optimal design for sustainable bioethanol supply chain considering detailed plant performance model, Comput. Chem. Eng. 35 (8) (2011) 1384–1398.
[34] A. Guerra, A.M. Newman, S. Leyffer, Concrete structure design using mixed-integer nonlinear programming with complementarity constraints, SIAM J. Optim. 21 (2011) 833–863.
[35] K. Abhishek, S. Leyffer, J.T. Linderoth, Modeling without categorical variables: a mixed-integer nonlinear program for the optimization of thermal insulation systems, Optim. Eng. 11 (2010) 185–212.
[36] P.M. Pardalos, W. Chaovalitwongse, L.D. Iasemidis, J. Chris Sackellares, D.-S. Shiau, P.R. Carney, O.A. Prokopyev, V.A. Yatsenko, Seizure warning algorithm based on optimization and nonlinear dynamics, Math. Program. 101 (2004) 365–385.
[37] W.P. Adams, H.D. Sherali, Mixed-integer bilinear programming problems, Math. Program. 59 (1993) 279–305.
[38] A.S. Freire, E. Moreno, J.P. Vielma, An integer linear programming approach for bilinear integer programming, Oper. Res. Lett. 40 (2) (2012) 74–77.
[39] M.X. Goemans, D. Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, J. ACM 42 (1995) 1115–1145.
[40] M.W. Padberg, The boolean quadric polytope: some characteristics, facets and relatives, Math. Program. 45 (1989) 139–172.
[41] M. Anjos, A. Vannelli, Computing globally optimal solutions for single-row layout problems using semidefinite programming and cutting planes, INFORMS J. Comput. 20 (2008) 611–617.
[42] H. Sherali, J. Desai, A global optimization RLT-based approach for solving the hard clustering problem, J. Global Optim. 32 (2005) 281–306.
[43] S. Burer, On the copositive representation of binary and continuous nonconvex quadratic programs, Math. Program. 120 (2009) 479–495.
[44] J. Linderoth, A simplicial branch-and-bound algorithm for solving quadratically constrained quadratic programs, Math. Program. 103 (2005) 251–282.
[45] A. Fischer, C. Helmberg, The symmetric quadratic traveling salesman problem, Manuscript, Fakultät für Mathematik, Technishe Universität Chemnitz, Germany, 2011.
[46] I. Gentilini, F. Margot, K. Shimada, The traveling salesman problem with neighborhoods: MINLP solution. Manuscript, Tepper School of Business, Carnegie Mellon University. Optim. Methods Softw., 2011 (in press).
[47] J.B. Lasserre, Global optimization with polynomials and the problem of moments, SIAM J. Optim. 11 (2001) 796–817.
[48] G.P. McCormick, Computability of global solutions to factorable nonconvex programs. part I. convex underestimating problems, Math. Program. 10 (1976) 146–175.
[49] R. Horst, H. Tuy, Global Optimization: Deterministic Approaches, second ed., Springer-Verlag, Berlin, 1993.
[50] I.P. Androulakis, C.D. Maranas, C.A. Floudas, α-BB: a global optimization method for general constrained nonconvex problems, J. Global Optim. 7 (1995) 337–363.
[51] E.M.L. Beale, Mathematical Programming in Practice, Pitmans, London, 1968.
[52] J. Tomlin, A suggested extension of special ordered sets to non-separable non-convex programming problems, Ann. Discrete Math. 11 (1981) 359–370.
[53] P. Belotti, Disjunctive cuts for non-convex MINLP, in: Mixed Integer Nonlinear Programming, in: The IMA Volumes in Mathematics and its Applications, vol. 154, Springer, 2012, pp. 117–144.
[54] A.H. Land, A.G. Doig, An automatic method of solving discrete programming problems, Econometrica 28 (1960) 497–520.

[55] E.M.B. Smith, C.C. Pantelides, Global optimisation of non-convex MINLPs, Comput. Chem. Eng. 21 (1997) S791.
[56] S. Lee, I.E. Grossmann, A global optimization algorithm for non-convex generalized disjunctive programming and applications to process systems, Comput. Chem. Eng. 25 (2001) 1675–1697.
[57] H.S. Ryoo, N.V. Sahinidis, Global optimization of nonconvex NLPs and MINLPs with applications in process design, Comput. Chem. Eng. 19 (1995) 551–566.
[58] H.S. Ryoo, N.V. Sahinidis, A branch-and-reduce approach to global optimization, J. Global Optim. 8 (1996) 107–138.
[59] M. Tawarmalani, N.V. Sahinidis, Global optimization of mixed-integer nonlinear programs: a theoretical and computational study, Math. Program. 99 (2004) 563–591.
[60] M. Tawarmalani, N.V. Sahinidis, A polyhedral branch-and-cut approach to global optimization, Math. Program. 103 (2005) 225–249.
[61] P. Belotti, J. Lee, L. Liberti, F. Margot, A. Wächter, Branching and bounds tightening techniques for non-convex MINLP, Optim. Methods Softw. 24 (2009) 597–634.
[62] C.S. Adjiman, I.P. Androulakis, C.A. Floudas, A global optimization method, α-BB, for general twice-differentiable constrained NLPs: II. implementation and computational results, Comput. Chem. Eng. 22 (1998) 1159–1179.
[63] C.S. Adjiman, S. Dallwig, C.A. Floudas, A. Neumaier, A global optimization method, α-BB, for general twice-differentiable NLPs: I. theoretical advances, Comput. Chem. Eng. 22 (1998) 1137–1158.
[64] C.S. Adjiman, I.P. Androulakis, C.A. Floudas, Global optimization of mixed-integer nonlinear problems, AIChE J. 46 (2000) 1769–1797.
[65] E.M.L. Beale, J.A. Tomlin, Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables, in: J. Lawrence (Ed.), Proc. of the 5th Nat. Conf. in Oper. Res., Tavistock, London, 1970, pp. 447–454.
[66] E.M.L. Beale, J.J.H. Forrest, Global optimization using special ordered sets, Math. Program. 10 (1) (1976) 52–69.
[67] A.B. Keha, I.R. de Farias Jr., G.L. Nemhauser, Models for representing piecewise linear cost functions, Oper. Res. Lett. 32 (2004) 44–48.
[68] A.B. Keha, I.R. de Farias Jr., G.L. Nemhauser, A branch-and-cut algorithm without binary variables for nonconvex piecewise linear optimization, Oper. Res. 54 (2006) 847–858.
[69] J.P. Vielma, G.L. Nemhauser, Modeling disjunctive constraints with a logarithmic number of binary variables and constraints, Math. Program. 128 (2011) 49–72.
[70] E.M.L. Beale, Branch and bound methods for mathematical programming systems, Ann. Discrete Math. 5 (1979) 201–219. Discrete optimization (Proc. Adv. Res. Inst. Discrete Optimization and Systems Appl., Banff, Alta., 1977), II.
[71] B. Geißler, A. Martin, A. Morsi, L. Schewe, Using piecewise linear functions for solving MINLPs, in: J. Lee, S. Leyffer (Eds.), Mixed Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2012, pp. 287–314.
[72] S. Leyffer, A. Sartenaer, E. Wanufelle, Branch-and-refine for mixed-integer nonconvex global optimization. ANL/MCS-P1547-0908, Argonne National Laboratory, 2008, Preprint.
[73] H.P. Benson, S.S. Erenguc, An algorithm for concave integer minimization over a polyhedron, Naval Res. Logist. 37 (1990) 515–525.
[74] K.M. Bretthauer, A.V. Cabot, M.A. Venkataramanan, An algorithm and new penalties for concave integer minimization over a polyhedron, Naval Res. Logist. 41 (1994) 435–454.
[75] H. Tuy, Concave programming under linear constraints, Soviet Math. 5 (1964) 1437–1440.
[76] P. Kesavan, R.J. Allgor, E.P. Gatzke, P.I. Barton, Outer approximation algorithms for separable non-convex mixed-integer nonlinear programs, Math. Program. 100 (2004) 517–535.
[77] R. Karuppiah, I.E. Grossman, A Lagrangian based branch-and-cut algorithm for global optimization of nonconvex mixed-integer nonlinear programs with decomposable structures, J. Global Optim. 41 (2008) 163–186.
[78] C. D'Ambrosio, J. Lee, A. Wächter, An algorithmic framework for MINLP with separable non-convexity, in: J. Lee, S. Leyffer (Eds.), Mixed-Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 315–348.
[79] S. Leyffer, Integrating SQP and branch-and-bound for mixed integer nonlinear programming, Comput. Optim. Appl. 18 (2001) 259–309.
[80] I. Nowak, Vigerske, LaGO: a (heuristic) branch and cut algorithm for nonconvex MINLPs, Cent. Eur. J. Oper. Res. 16 (2008) 127–138.
[81] O. Exler, L.T. Antelo, J.A. Egea, A.A. Alonso, J.R. Banga, A tabu search-based algorithm for mixed-integer nonlinear problems and its application to integrated process and control system design, Comput. Chem. Eng. 32 (2008) 1877–1891.
[82] Y. Luo, X. Yuan, Y. Liu, An improved PSO algorithm for solving non-convex NLP/MINLP problems with equality constraints, Comput. Chem. Eng. 31 (2007) 153–162.
[83] C.-T. Young, Y. Zheng, C.-W. Yeh, S.-S. Jang, Information-guided genetic algorithm approach to the solution of MINLP problems, Ind. Eng. Chem. Res. 46 (2007) 1527–1537.
[84] M. Schlüter, J.A. Egea, J.R. Banga, Extended ant colony optimization for non-convex mixed integer nonlinear programming, Comput. Oper. Res. 36 (7) (2009) 2217–2229.
[85] L. Liberti, G. Nannicini, N. Mladenovic, A recipe for finding good solutions to MINLPs, Math. Program. Comput. 3 (2011) 349–390.
[86] T. Berthold, RENS: the optimal rounding. Zib-report 12-17, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Berlin, Germany, April 2012.
[87] C. D'Ambrosio, A. Frangioni, L. Liberti, A. Lodi, Experiments with a feasibility pump approach for nonconvex MINLPs, in: P. Festa (Ed.), Proceedings of the 9th Symposium on Experimental Algorithms, in: Lecture Notes in Computer Science, vol. 6049, Springer, Berlin, 2010.
[88] G. Nannicini, P. Belotti, Rounding-based heuristics for nonconvex MINLPs, Math. Program. Comput. 4 (1) (2012) 1–31.
[89] N.Z. Shor, Dual quadratic estimates in polynomial and Boolean programming, Ann. Oper. Res. 25 (1–4) (1990) 163–168. Computational methods in global optimization.
[90] H.D. Sherali, C.H. Tuncbilek, Comparison of two reformulation-linearization technique based linear programming relaxations for polynomial programming problems, J. Global Optim. 10 (4) (1997) 381–390.
[91] R. Fortet, L'Algèbre de Boole et ses applications en recherche opérationnelle, Cahiers Centre Etudes Rech. Oper. no. 4 (1959) 5–36.
[92] F. Glover, E. Woolsey, Converting the 0–1 polynomial programming problem to a 0–1 linear program, Oper. Res. 22 (1974) 180–182.
[93] P. Hansen, Methods of nonlinear 0–1 programming, Ann. Discr. Math. 5 (1979) 53–70.
[94] F. Glover, Improved linear integer programming formulations of nonlinear integer problems, Management. Sci. 22 (1975) 455–460.
[95] W.P. Adams, R.J. Forrester, A simple recipe for concise mixed 0–1 linearizations, Oper. Res. Lett. 33 (2005) 55–61.
[96] W.P. Adams, R.J. Forrester, F.W. Glover, Comparisons and enhancement strategies for linearizing mixed 0–1 quadratic programs, Discrete Optim. 1 (2004) 99–120.
[97] W. Chaovalitwongse, P.M. Pardalos, O.A. Prokopyev, A new linearization technique for multi-quadratic 0–1 programming problems, Oper. Res. Lett. 32 (6) (2004) 517–522.
[98] H.D. Sherali, J.C. Smith, An improved linearization strategy for zero–one quadratic programming problems, Optim. Lett. 1 (1) (2007) 35–47.
[99] P.L. Hammer, A.A. Rubin, Some remarks on quadratic programming with 0–1 variables, RAIRO 3 (1970) 67–79.
[100] F. Körner, A tight bound for the boolean quadratic optimization problem and its use in a branch and bound algorithm, Optimization 19 (1988) 711–721.
[101] P.L. Hammer, P. Hansen, B. Simeone, Roof duality, complementation and persistency in quadratic 0–1 optimization, Math. Program. 28 (1984) 121–155.
[102] E. Boros, P.L. Hammer, Pseudo-Boolean optimization, Discrete Appl. Math. 123 (2002) 155–225.
[103] P.M. Pardalos, G.P. Rodgers, Computational aspects of a branch and bound algorithm for quadratic zero–one programming, Computing 45 (2) (1990) 131–144.
[104] S. Poljak, H. Wolkowicz, Convex relaxations of (0, 1)-quadratic programming, Math. Oper. Res. 20 (1995) 550–561.
[105] A. Caprara, Constrained 0–1 quadratic programming: Basic approaches and extensions, European J. Oper. Res. 187 (2008) 1494–1503.
[106] W.P. Adams, H.D. Sherali, A tight linearization and an algorithm for zero–one quadratic programming problems, Management Sci. 32 (10) (1986) 1274–1290.
[107] H.D. Sherali, W.P. Adams, A hierarchy of relaxations between the continuous and convex hull representations for zero–one programming problems, SIAM J. Discrete Math. 3 (3) (1990) 411–430.
[108] N.Z. Shor, Quadratic optimization problems, Sov. J. Comput. Syst. Sci. 25 (1987) 1–11.
[109] L. Lovász, A.J. Schrijver, Cones of matrices and set-functions and 0–1 optimization, SIAM J. Optim. 1 (1991) 166–190.
[110] T. Fujie, M. Kojima, Semidefinite programming relaxation for nonconvex quadratic programs, J. Global Optim. 10 (1997) 367–380.
[111] C. Helmberg, F. Rendl, R. Weismantel, A semidefinite programming approach to the quadratic knapsack problem, J. Comb. Optim. 4 (2002) 197–215.
[112] C. Lémaréchal, F. Oustry, SDP relaxations in combinatorial optimization from a Lagrangian viewpoint, in: N. Hadjisawas, P.M. Pardalos (Eds.), Advances in Convex Analysis and Global Optimization, Kluwer, Dordrecht, 2001.
[113] S. Poljak, F. Rendl, H. Wolkowicz, A recipe for semidefinite relaxation for (0, 1)-quadratic programming, J. Global Optim. 7 (1995) 51–73.
[114] M. Ramana, An algorithmic analysis of multiquadratic and semidefinite programming problems, Ph.D. Thesis, Johns Hopkins University, Baltimore, MD, 1993.
[115] K.M. Anstreicher, Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming, J. Global Optim. 43 (2009) 471–484.
[116] C. Buchheim, A. Wiegele, Semidefinite relaxations for non-convex quadratic mixed-integer programming. Math. Program. 1–18. http://dx.doi.org/10.1007/s10107-012-0534-y.
[117] M. Dür, Copositive programming: a survey, in: M. Diehl, F. Glineur, E. Jarlebring, W. Michiels (Eds.), Recent Advances in Optimization and its Applications in Engineering, Springer, Berlin, 2010.
[118] J. Chen, S. Burer, Globally solving nonconvex quadratic programming problems via completely positive programming, Math. Program. Comput. 4 (1) (2012) 33–52.
[119] M.M. Deza, M. Laurent, Geometry of Cuts and Metrics, Springer, Berlin, 1997.
[120] M. Jünger, V. Kaibel, Box-inequalities for quadratic assignment polytopes, Math. Program. 91 (2001) 175–197.

[121] H. Saito, T. Fujie, T. Matsui, S. Matuura, A study of the quadratic semi-assignment polytope, Discrete Optim. 6 (2009) 37–50.
[122] M.W. Padberg, M.P. Rijal, Location, Scheduling, Design and Integer Programming, Springer, Berlin, 1996.
[123] K.M. Anstreicher, S. Burer, Computable representations for convex hulls of low-dimensional quadratic forms, Math. Program. 124 (2010) 33–43.
[124] S. Burer, A.N. Letchford, On non-convex quadratic programming with box constraints, SIAM J. Optim. 20 (2009) 1073–1089.
[125] Y. Yajima, T. Fujie, A polyhedral approach for nonconvex quadratic programming problems with box constraints, J. Global Optim. 13 (1998) 151–170.
[126] S. Burer, A.N. Letchford, Unbounded convex sets for non-convex mixed-integer quadratic programming. Working paper, Lancaster University, Lancaster, UK, 2011.
[127] C.A. Meyer, C.A. Floudas, Convex envelopes for edge-concave functions, Math. Program. 103 (2005) 207–224.
[128] O. Günlük, J. Lee, J. Leung, A polytope for a product of a real linear functions in 0/1 variables, in: J. Lee, S. Leyffer (Eds.), in: IMA Volumes in Mathematics and its Applications, Springer, 2011, pp. 513–532.
[129] J. Luedtke, M. Namazifar, J. Linderoth, Some results on the strength of relaxations of multilinear functions. Working paper, University of Wisconsin-Madison, 2010.
[130] A. Saxena, P. Bonami, J. Lee, Convex relaxations of non-convex mixed integer quadratically constrained programs: extended formulations, Math. Program. 124 (2010) 383–411.
[131] A. Saxena, P. Bonami, J. Lee, Convex relaxations of non-convex mixed integer quadratically constrained programs: projected formulations, Math. Program. 130 (2011) 359–413.
[132] S. Burer, A. Saxena, The MILP road to MIQCP, in: J. Lee, S. Leyffer (Eds.), Mixed-Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 373–406.
[133] L. Galli, K. Kaparis, A.N. Letchford, Gap inequalities for non-convex mixed-integer quadratic programs, Oper. Res. Lett. 39 (2011) 297–300.
[134] M. Laurent, S. Poljak, Gap inequalities for the cut polytope, SIAM J. Matrix Anal. 17 (1996) 530–547.
[135] T. Berthold, S. Heinz, S. Vigerske, Extending a CIP framework to solve MIQCPs, in: J. Lee, S. Leyffer (Eds.), Mixed-Integer Nonlinear Programming, in: IMA Volumes in Mathematics and its Applications, vol. 154, Springer, Berlin, 2011, pp. 427–445.
[136] A. Billionnet, S. Elloumi, M.-C. Plateau, Improving the performance of standard solvers for quadratic 0–1 programs by a tight convex reformulation: the QCR method, Discrete Appl. Math. 157 (2009) 1185–1197.
[137] A. Billionnet, S. Elloumi, A. Lambert, Extending the QCR method to general mixed-integer programs, Math. Program. 131 (2012) 381–401.
[138] I. Nowak, Lagrangian decomposition of block-separable mixed-integer all-quadratic programs, Math. Program. 102 (2005) 295–312.
[139] H.D. Sherali, W.P. Adams, A hierarchy of relaxations and convex hull characterizations for mixed-integer zero–one programming problems, Discrete Appl. Math. 52 (1) (1994) 83–106.
[140] H.D. Sherali, W.P. Adams, A reformulation-linearization technique (RLT) for semi-infinite and convex programs under mixed 0–1 and general discrete restrictions, Discrete Appl. Math. 157 (6) (2009) 1319–1333.
[141] Y. Nesterov, Global quadratic optimization via conic relaxation, in: R. Saigal, L. Vandenberghe, H. Wolkowicz (Eds.), Handbook of Semidefinite Programming, Kluwer Academic Publishers, 2000.
[142] P. Parrilo, Structured semidefinite programs and semi-algebraic geometry methods in robustness and optimization, Ph.D. Thesis, California Institute of Technology, 2000.
[143] M. Laurent, A comparison of the Sherali-Adams, Lovász-Schrijver, and Lasserre relaxations for 0–1 programming, Math. Oper. Res. 28 (2003) 470–496.
[144] J.A. De Loera, J. Lee, P.N. Malkin, S. Margulies, Hilbert's Nullstellensatz and an algorithm for proving combinatorial infeasibility, in: Proceedings of ISSAC 2008, ACM Press, New York, 2008, pp. 197–206.
[145] J.B. Lasserre, Polynomials nonnegative on a grid and discrete optimization, Trans. Amer. Math. Soc. 354 (2002) 631–649.
[146] D. Michaels, R. Weismantel, Polyhedra related to integer-convex polynomial systems, Math. Program. 105 (2006) 215–232.
[147] Y. Lin, L. Schrage, The global solver in the LINDO API, Optim. Methods Softw. 24 (2009) 657–668.
[148] R. Misener, C.A. Floudas, GloMIQO: global mixed-integer quadratic optimizer. Manuscript, Princeton University, Princeton, NJ, 2012. J. Global Optim. (in press). http://dx.doi.org/10.1007/s10898-012-9874-7.
[149] G.R. Kocis, I.E. Grossmann, Computational experience with DICOPT solving MINLP problems in process synthesis, Comput. Chem. Eng. 13 (1989) 307–315.
[150] D. Henrion, J.-B. Lasserre, J. Löfberg, GloptiPoly 3: moments, optimization and semidefinite programming, Optim. Methods Softw. 24 (2009) 761–779.