

Contents lists available at [ScienceDirect](https://www.sciencedirect.com)

Discrete Optimization

www.elsevier.com/locate/disopt

On the complexity of solving a decision problem with flow-depending costs: The case of the IJsselmeer dikes

A. Abiad^{a,b}, S. Gribling^c, D. Lahaye^d, M. Mnich^e, G. Regts^f, L. Vena^{f,g,*},
G. Verweij^h, P. Zwaneveld^h

^a Eindhoven University of Technology, Department of Mathematics and Computer Science, The Netherlands

^b Ghent University, Department of Mathematics: Analysis, Logic and Discrete Mathematics, Belgium

^c CWI, The Netherlands

^d Delft Institute for Applied Mathematics, The Netherlands

^e TU Hamburg, Institute for Algorithms and Complexity, 21073 Hamburg, Germany

^f University of Amsterdam, Korteweg de Vries Institute for Mathematics, The Netherlands

^g Charles University, Czech Republic

^h CPB Netherlands Bureau for Economic Policy Analysis, The Netherlands

ARTICLE INFO

Article history:

Received 16 July 2018

Received in revised form 24 October 2019

Accepted 20 November 2019

Available online xxxx

Keywords:

Cost–benefit analysis

Integer programming

Dynamic programming

ABSTRACT

We consider a fundamental integer programming (IP) model for cost–benefit analysis and flood protection through dike building in the Netherlands, due to Zwaneveld and Verweij (2017). Experimental analysis with data for the IJsselmeer shows that the solution of the linear programming relaxation of the IP model is integral. This naturally leads to question whether the polytope associated to the IP is always integral.

In this paper we first give a negative answer to this question by proving the non-integrality of the polytope. Secondly, we establish natural conditions that guarantee the linear programming relaxation of the IP model is integral. We show that these conditions are indeed satisfied by the recent data on flood probabilities, damage and investment costs of IJsselmeer. Finally, we show that the IP model can be solved in polynomial time when the number of dike segments, or the number of feasible barrier heights, are bounded.

© 2019 Elsevier B.V. All rights reserved.

1. Introduction

Protection against increasing sea levels is an important issue around the world, including the Netherlands. Optimal dike heights are of crucial importance to the Netherlands as almost 60% of its surface is under threat of flooding from sea, lakes, or rivers. This area is protected by more than 3500 kilometers of dunes and dikes, which require substantial yearly investments of more than one billion Euro [1].

* Corresponding author.

E-mail addresses: aida.abiad@ugent.be (A. Abiad), s.j.gribling@cwil.nl (S. Gribling), d.j.p.lahaye@tudelft.nl (D. Lahaye), mmnich@uni-bonn.de (M. Mnich), g.regts@uva.nl (G. Regts), lluis.vena@gmail.com (L. Vena), G.Verweij@cpb.nl (G. Verweij), P.J.Zwaneveld@cpb.nl (P. Zwaneveld).

<https://doi.org/10.1016/j.disopt.2019.100565>

1572-5286/© 2019 Elsevier B.V. All rights reserved.

Recently, Zwaneveld et al. [1,2] presented an integer programming (IP) model for a cost–benefit analysis to determine optimal dike heights that allows highly flexible input parameters for flood probabilities, damage costs and investment costs for dike heightening. Their model improves upon an earlier model by Brekelmans et al. [3], who presented a dedicated approach without guaranteed optimality, and which was, in turn, an improvement of the original model by van Dantzig [4] from 1956. The latter was introduced after a devastating flood in the Netherlands in 1953, with the goal of designing a long-lasting cost-efficient layout for a dike ring.

This paper considers an IP model for cost–benefit analysis and flood protection through dike building in the Netherlands. This model is based on the work of Bos and Zwaneveld [5], Zwaneveld and Verweij [6] and recent work by Zwaneveld and Verweij [2]. In the latter, the authors study the problem of economically optimal flood prevention in a situation in which multiple barrier dams and dikes protect the hinterland to both sea level rise as well as peak river discharges.

Current optimal flood prevention methods, e.g., by Kind [7], Brekelmans et al. [3], Zwaneveld, Verweij, and van Hoesel [1], only consider single dike ring areas with no interdependency between dikes. However, the assumption that the failure probability of a dike ring may not depend on the strength of other dikes or barrier dams “is clearly not valid for many [...] dike ring areas in The Netherlands” [2]. This fact motivated the Dutch ministry of Infrastructure and the Environment to ask for a model that is capable of assessing these interdependencies. Only such interdependencies are powerful enough to model the complex interactions that can be observed on real-life instances; for instance, Eijgenraam et al. [8] state that “if one is interested in an optimal investment policy (i.e., when, how much, and where to heighten dikes), then the model [...] for nonhomogeneous dike rings [should be used]”. The first such model being capable of meeting the ministry’s demands was presented by Zwaneveld and Verweij [2,6]. They present a graph-based IP model for a cost–benefit analysis to determine optimal dike heights with multiple interdependencies between dikes and barrier dams. Zwaneveld and Verweij [6] identify several solution approaches (e.g. a dynamic programming heuristic and branch-and-cut), and they also show that it can be solved quickly using a branch-and-cut approach for real world problem instances.

Zwaneveld and Verweij [2] made the interesting observation that their branch-and-cut approach yields, on all instances they tested, an integer optimal solution without any branching. Thus, on those instances, solving the linear programming (LP) relaxation of their IP model gives the optimal solution for the IP model. A natural question arising from this observation is:

Question 1. *Does the LP relaxation of the Zwaneveld–Verweij IP model always admit an integral optimal solution?*

The question was posed explicitly by the *Netherlands Bureau for Economic Policy Analysis* (CPB) at the 2017 Study Group Mathematics with Industry in Amsterdam, The Netherlands. A positive answer would have practical relevance, as it would imply that the computational cost of solving the IP is always reasonable regardless of the objective function (by using the LP relaxation of the problem). A negative answer to the question would imply that, for certain objective functions, the computational cost of solving the IP may increase significantly.

1.1. Our contribution

The contributions of the paper are as follows.

Our first contribution is a negative answer to [Question 1](#). This implies that despite what the experiments might suggested, the polytope related to the LP relaxation of the integer problem of model of Zwaneveld and Verweij [2,6] is not integral. We provide details in [Section 3](#).

Second, we derive sufficient conditions on costs that ensure the LP relaxation to be integral (see Proposition 2 and Corollary 1). Thus, the IP model can be solved using the linear relaxation when the conditions in Proposition 2 and Corollary 1 are met. Moreover, we experimentally verify that such conditions hold on numerous data related to the case we are considering. For that, we use recent data on flood probabilities, damage and investment costs that are currently used by the Dutch government. This shows that the conditions derived in Proposition 2 and Corollary 1 are of practical importance. Details can be found in Section 3.

Third, in Section 4 we propose two alternative approaches to solve the problem, by means of dynamic programming and by shortest-path computations. Those approaches lead to polynomial-time algorithms of the related IP model when some natural problem parameters – such as the number of dike segments, or the number of feasible barrier dam heights – are bounded.

Finally, in Section 5 we present a natural abstract version of the dike height problem, which allows for several variations and open problems.

In Section 6 we present the conclusions of this work.

2. The Zwaneveld–Verweij integer programming model

In this section we present the IP model formulated by Zwaneveld and Verweij [2]. Before going into the details of the IP model, we introduce some important terminology. A *dike* is a natural or artificial slope or wall that regulates water levels to protect a region from flooding, and a *dike segment* is a part of a dike. Several dike segments may jointly protect the same area, in which case they form a *dike ring*. In the Netherlands, around the IJsselmeer, dike ring areas and smaller dikes lie beneath the *Afsluitdijk* which is the outermost dike located in the north. The Afsluitdijk separates the North Sea and the IJsselmeer, an artificial lake. The Afsluitdijk is thus a *barrier dam* (or simply a *barrier*) that stops or restricts the flow of water.

The IP model uses the following data:

- T is the set of time periods at which changes to a dike segment can be made. Without loss of generality, we assume that $T = \{0, 1, \dots, T_{\max}\}$. The time period between time t and time $t + 1$ can be either one year or any other number of years.
- D is the set of dike segments. We identify D with the first $|D|$ positive integers $D = \{1, \dots, |D|\}$.
- H_D is the set of possible heights for a dike segment. For ease of notation, we do not let H_D depend on the dike segment, i.e., all dike segments have the same set of possible heights. The heights might receive an upgrade, and go from height h_1 at time $t - 1$, to a height h_2 at time t . We denote the height of the previous year (parameter in time T being equal to $t - 1$) by h_1 , and that of the current year by h_2 (parameter in time T being equal to t).
- Likewise, H_B is the set of possible heights for the barrier and we denote the height of the barrier in the previous year by h_1^B , and that of the current year by h_2^B .

The decision variables are:

- $B(t, h_1^B, h_2^B)$ is a binary variable taking value 1 if the barrier is updated from height h_1^B at time $t - 1$ to height h_2^B at time t . If $h_1^B = h_2^B$ then the barrier is not strengthened in period from $t - 1$ to t and remains at its previous height. This decision variable is used for bookkeeping investment (and maintenance) costs, flood probabilities, and related expected damage costs of the barrier.
- $CY(t, d, h_1, h_2)$ is a binary variable taking value 1 if dike ring segment d is updated from height h_1 at time $t - 1$ to height h_2 at time t . If $h_1 = h_2$ then the dike ring segment d is not strengthened in the time period from $t - 1$ to t and remains at its previous height. This decision variable is used for tracking investment (and maintenance) ‘costs’, hence the chosen name.

The following are auxiliary variables:

- $DY(t, d, h_2, h_2^B)$ is a binary variable taking value 1 if at time t the barrier has height h_2^B , and dike segment d is of height h_2 . Alternatively, this variable can be seen as a linearization of the 0-1 variable $\left(\sum_{h_1} CY(t, d, h_1, h_2)\right) \left(\sum_{h_1^B} B(t, h_1^B, h_2^B)\right)$: $DY(t, d, h_2, h_2^B)$ takes value 1 if both $\left(\sum_{h_1} CY(t, d, h_1, h_2)\right)$ and $\left(\sum_{h_1^B} B(t, h_1^B, h_2^B)\right)$ are 1. This variable is used to connect investments in dike segments (and the barrier) to expected damages. Indeed, the total cost depends not only on the individual heights of the barrier and the dike, but also on the combined pairs of heights of both (barrier height, dike d heights); the values of DY are forced by the constraints once B and CY values are chosen. The name DY reflects the ‘dummy’ nature of the variable.

The input parameters are:

- $D_{\text{cost}}(t, d, h_1, h_2)$ is the cost for investment and maintenance if the dike segment d is strengthened from height h_1 at time $t - 1$ to height h_2 at time t . If $h_1 = h_2$, the dike segment is not strengthened and these costs only represent maintenance costs.
- $D_{\text{expdam}}(t, d, h_2, h_2^B)$, is the expected damage, i.e.,

$$D_{\text{expdam}}(t, d, h_2, h_2^B) = \text{prob}(t, d, h_2, h_2^B) \times \text{damage}(t, d, h_2, h_2^B),$$

where $\text{prob}(t, d, h_2, h_2^B)$ and $\text{damage}(t, d, h_2, h_2^B)$ are the probability of failure and the expected damage cost (the latter given that there is a flooding) at time t given the height of segment h_2 and the height of the barrier h_2^B . Note that it is assumed that both the probability of failure and the expected damage upon failure of dike segment d only depend on the height of the dike segment d , on the height of the barrier and on the time t .

- $B_{\text{cost}}(t, d, h_1^B, h_2^B)$ is the cost for investment and maintenance if the barrier is strengthened from height h_1^B at time $t - 1$ to height h_2^B at time t . If $h_1^B = h_2^B$, the barrier is not strengthened and these costs only represent maintenance costs.
- $B_{\text{expdam}}(t, h_2^B)$ is the expected damage of a flooding of the barrier, i.e.,

$$B_{\text{expdam}}(t, h_2^B) = \text{prob}(t, h_2^B) \times \text{damage}(t, h_2^B).$$

Here, $\text{prob}(t, h_2^B)$ and $\text{damage}(t, h_2^B)$ are the probability of failure and the expected damage cost (the latter given that there is a flooding), in period t given the height of the barrier h_2^B . Both $\text{prob}(t, h_2^B)$ and $\text{damage}(t, h_2^B)$ depend only on the height of the barrier and on the time t .

The values of the input parameters (costs) are estimated for every year and then discounted to a present value, which is actually set at the year 2020. Thus, the cost estimate is a present net value calculation.

All in all, the IP model, which we refer to as the Zwaneveld–Verweij IP model, then reads as follows:

$$\min \sum_{t \in T} \sum_{d \in D} \sum_{h_1 \in H_D} \sum_{h_2 \geq h_1} D_{\text{cost}}(t, d, h_1, h_2) \cdot CY(t, d, h_1, h_2) \tag{1}$$

$$+ \sum_{t \in T} \sum_{d \in D} \sum_{h_2 \in H_D} \sum_{h_2^B} D_{\text{expdam}}(t, d, h_2, h_2^B) \cdot DY(t, d, h_2, h_2^B) \tag{2}$$

$$+ \sum_{t \in T} \sum_{h_1^B \in H^B} \sum_{h_2^B \geq h_1^B} (B_{\text{cost}}(t, h_1^B, h_2^B) + B_{\text{expdam}}(t, h_2^B)) \cdot B(t, h_1^B, h_2^B) \tag{3}$$

subject to

$$CY(0, d, 0, 0) = 1, CY(0, d, h_1, h_2) = 0 \quad \forall d \in D, h_1, h_2 \in H_D \subset \mathbb{N}, h_2 \geq h_1 \text{ and } h_2 > 0 \tag{4}$$

$$\sum_{h_1 \leq h_2} CY(t - 1, d, h_1, h_2) = \sum_{h_3 \geq h_2} CY(t, d, h_2, h_3) \quad \forall t \in T \setminus \{0\}, d \in D, h_2 \in H_D \tag{5}$$

$$\sum_{h_1 \leq h_2} \text{CY}(t, d, h_1, h_2) = \sum_{h_2^B} \text{DY}(t, d, h_2, h_2^B) \quad \forall t \in T, d \in D, h_2 \in H_D \quad (6)$$

$$\text{B}(0, 0, 0) = 1, \text{B}(0, h_1^B, h_2^B) = 0 \quad \forall h_1^B, h_2^B \in H_B \subset \mathbb{N}, h_2^B \geq h_1^B \text{ and } h_2^B > 0 \quad (7)$$

$$\sum_{h_1^B \leq h_2^B} \text{B}(t-1, h_1^B, h_2^B) = \sum_{h_3^B \geq h_2^B} \text{B}(t, h_2^B, h_3^B) \quad \forall t \in T \setminus \{0\}, d \in D, h_2^B \in H_B \quad (8)$$

$$\sum_{h_1^B \leq h_2^B} \text{B}(t, h_1^B, h_2^B) = \sum_{h_2} \text{DY}(t, d, h_2, h_2^B) \quad \forall t \in T, d \in D, h_2^B \in H_B \quad (9)$$

$$\text{CY}(t, d, h_1, h_2) \in \{0, 1\} \quad \forall t \in T, d \in D, h_1, h_2 \in H_D, h_2 \geq h_1 \quad (10)$$

$$\text{DY}(t, d, h_2, h_2^B) \in \{0, 1\} \quad \forall t \in T, d \in D, h_2 \in H_D, h_2^B \in H_B \quad (11)$$

$$\text{B}(t, h_1^B, h_2^B) \in \{0, 1\} \quad \forall t \in T, d \in D, h_1^B, h_2^B \in H_B, h_2^B \geq h_1^B \quad (12)$$

The objective function (1) + (2) + (3) gives the total expected cost for a given sequence of height upgrades, both for the dam barrier and dike segments, considering the costs of upgrading the dikes as well as the expected damages upon dike failure. Eqs. (6) and (9) are the linking constraints between the barrier and the dike segments using the variables DY. We shall observe that the heights of the dikes, or the barrier, are not allowed to decrease, so the variables $\text{CY}(t, d, h_1, h_2)$ and $\text{B}(t, h_1, h_2)$ exist only when $h_1 \leq h_2$. The remaining constraints are explained in the following section.

2.1. Graph representation of the Zwaneveld–Verweij IP model

The IP problem can be modeled as a directed graph (see Fig. 1 for how to represent Example 1 detailed in Section 3.1.)

Such graph can be built as a union of bipartite graphs, one for each dike segment $G_1, \dots, G_{|D|}$ and one for the barrier G_B . Each of these bipartite graphs has $|T|$ stable vertex sets, and each of these stable sets contains either $|H_D|$ (for $G_1, \dots, G_{|D|}$) or $|H_B|$ (for G_B) vertices. Hence the vertices are the triplets (t, d, h) with $t \in T, h \in H_D$ and $d \in D$, while the vertex set of G_B contains all the pairs (t, h^B) with $t \in T$ and $h^B \in H_B$. As a consequence, the vertex set for the graph of the IP problem is the union of the vertex sets for $G_i, i \in D$, and the vertex set for G_B .

The edges in the graph correspond, one-to-one, to variables in the IP problem. Each graph G_d for $d \in D$, contains an edge corresponding to the variable $\text{CY}(t, d, h_1, h_2)$, from the vertex $(t-1, d, h_1)$ (in the stable set corresponding to the time $t-1$) to the vertex (t, d, h_2) (in the stable set corresponding to the time t). The graph G_B contains an edge from the vertex $(t-1, h_1^B)$ (in the stable set $t-1$) to the vertex (t, h_2^B) (in the stable set t) whenever there is a corresponding variable $\text{B}(t, h_1^B, h_2^B)$. Additionally, for $d \in D$ there is an undirected edge between the vertex (t, d, h_2) in G_d and the vertex (t, h_2^B) in G_B for each connecting variable $\text{DY}(t, d, h_2, h_2^B)$.

2.2. Explanation of the constraints and of the Zwaneveld–Verweij IP model

With the above representation, Eqs. (5) and (8) can be seen as flow constraints on the graphs $G_1, \dots, G_{|D|}$ and G_B , respectively. However, the presence of the linking constraints (6) and (9) implies that the whole problem cannot be seen as a minimum-cost flow problem on directed graphs. Eqs. (4) and (7) are the initial and boundary conditions. Eqs. (10)–(12) are integrality constraints. The heights of the dike segments and the barrier heights never decrease.

Observe that, by the initializations (4), (7) and the constraints (5), (8) the following equalities also hold:

$$\sum_{h_1^B \leq h_2^B} \text{B}(t, h_1^B, h_2^B) = 1, \text{ and } \sum_{h_1 \leq h_2} \text{CY}(t, d, h_1, h_2) = 1, \text{ for all } t \in T, d \in D.$$

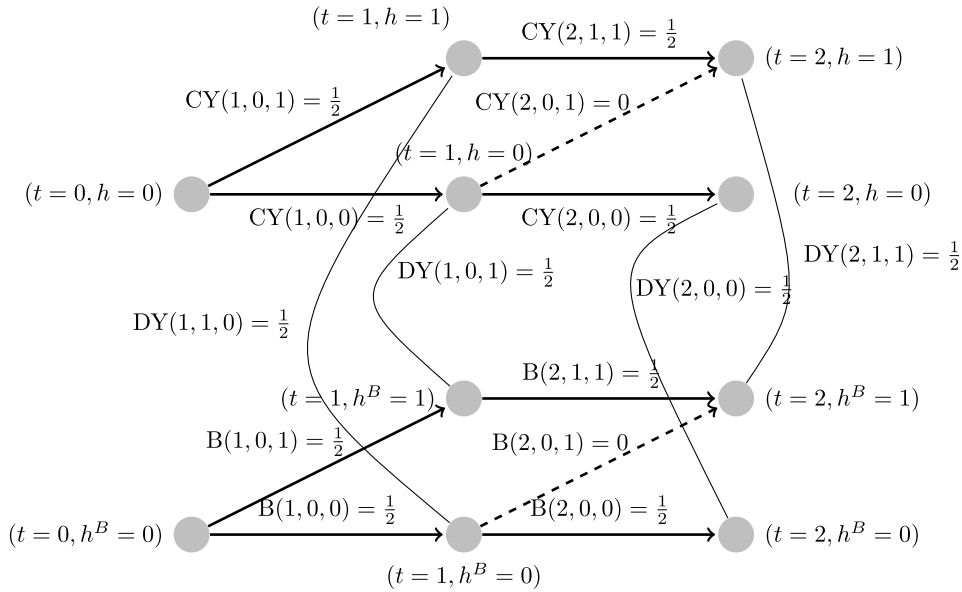


Fig. 1. Example of a non-integral point of the polytope associated with the Zwaneveld–Verweij IP model. The 0-valued variables are either omitted or dashed. The edges between the vertices for $t = 2$ are crossing. The initialization and contour variables from (4) and (7) are omitted.

Intuitively speaking, the above model consists of many smaller models corresponding to the heightening of each dike segment, which are joined together into a big model by the barrier (which can be also heightened and which interacts with the dike segments). This joining together is the source of the model’s difficulty and is characteristic for flood protection problems which often involve ‘small measures’ affecting a single area, and ‘big measures’ affecting multiple areas. The complexity results that we derive in the following sections are logical since they state that, essentially, the per-dike problem is simple but it is the joining of all dikes that makes it difficult.

2.3. Comparison with other models

Other models to solve the problem of determining the heights of the dikes have been considered, e.g., by Brekelmans et al. [3], Eijgenraam et al. [8]. Those models consider the dike heights as continuous decision variables. In general, a continuous optimization model may not be convex, and therefore the solution found may be a local minimum. In contrast, the Zwaneveld–Verweij IP model presented in the previous section considers a discretization of these decisions (when to update the heights and by how much). As a benefit, the problem is now linear and the optimal solution is a global optimum. As a drawback, when the problem is discretized the number of decision variables increases, the problem must be solved over the integers, and some information might be lost due to the used discretization. Indeed, the discretization may not capture the global optimum of the original problem (for instance if the coordinates of the optimal solution are irrational numbers). The latter can be addressed by considering a refined discretization.

In Postek et al. [9], the authors use an integer programming model as the Zwaneveld–Verweij IP model from Section 2 to determine the minimal cost for determining the height of the dikes in a dike system. They introduce the possibility of uncertainty in the calculations for some of the costs, probabilities, and safety parameters (for instance, due to the uncertainty in the sea level rise in the next 30 years). Their model allows to consider interactions between the dikes and a barrier (through what in [9] is referred to as *large-scale measures*). The model from [9] assumes a certain required safety level, and aims to minimize the

Table 1

Values of the variables for the vertex p of P_1 from Example 1 taking non-zero values; all omitted variables take value zero.

(t, h_1, h_2) or (t, h_2, h_2^B)	(0, 0, 0)	(1, 0, 1)	(1, 0, 0)	(2, 1, 1)	(2, 0, 0)	(1, 1, 0)	(1, 0, 1)
CY(t, h_1, h_2)	1	1/2	1/2	1/2	1/2	0	0
B(t, h_1, h_2)	1	1/2	1/2	1/2	1/2	0	0
DY(t, h_2, h_2^B)	1	0	0	1/2	1/2	1/2	1/2

total cost of updating the dikes subject to such conditions. In contrast, the Zwaneveld–Verweij IP model focuses on making a cost–benefit analysis, and as such, the required safety levels in [9] exist through the expected costs involved if a failure occurs; the uncertainty allowed in the computations by [9] ought to be linked to the probability of failure in the Zwaneveld–Verweij IP model.

3. On the integrality of the polytope

3.1. The polytope is non-integral

In this section, we follow the definitions and conventions of Schrijver [10]. All polytopes considered are convex polytopes. Recall that a polytope in \mathbb{R}^k is said to be *non-integral* if it has a vertex (facet of dimension 0) with at least one coordinate that is not an integer. Additionally, every vertex in the polytope defined by the constraints can be obtained as the optimal solution of a linear program with an adequate objective function.

The Zwaneveld–Verweij IP model from Section 2 defines a family of integer linear programs, one for each choice of sets D, H_B, H_D, T and objective function (1)+(2)+(3). In its linear relaxation, the integer variables (10)–(12) taking values in $\{0, 1\}$ are allowed to vary in the interval $[0, 1]$. Therefore, the constraints of the linear relaxation of the integer linear program define a polytope in $\mathbb{R}^{\#\{\text{variables}\}} = \mathbb{R}^{|T| \left(|D| \binom{|H_D|+1}{1} + \binom{|H_B|+1}{1} \right)}$, by identifying the coordinates of each element of the polytope with the variables in the linear program. We call the IP model *non-integral* if there exist sets D, H_B, H_D, T for which the polytope induced by the constraints of the linear relaxation is non-integral.

Our main result in this section is as follows:

Proposition 1. *The Zwaneveld–Verweij IP model from Section 2 is non-integral.*

We will provide two different proofs of Proposition 1, one in Section 3.1.1 and one in Section 3.1.2.

The sets from the following example induce a polytope that has a non-integral vertex.

Example 1.

- $T = \{0, 1, 2\}$, that is, three time periods.
- $D = \{1\}$, that is, one dike segment. Since there is only one dike segment, we omit the corresponding indices (for instance, $CY(t, d, h_1, h_2)$ simplifies to $CY(t, h_1, h_2)$).
- $H_D = H = \{0, 1\}$, $H_B = \{0, 1\}$. That is, two different heights for the barrier and for the dikes.

The number of variables in the example is 21. We denote by P_1 the polytope induced by the linear relaxation of Example 1. The values of the variables for some non-integral vertex p of P_1 are indicated in Table 1. In Fig. 1 we consider a visual representation of the vertex p as a solution of a flow problem on graphs; the variables and their values are indicated near the edge that represents them.

One can check that p is a feasible solution of the LP relaxation (a point in the polytope). Indeed, the flow conditions are satisfied, as are the equations linking the dummy variables DY and the CY’s and B’s (Eqs. (6) and (9)). Next, we will prove that p is indeed a *vertex* of the polytope P_1 .

3.1.1. p is a vertex of P_1 . First proof of Proposition 1

Our goal is to show that p is a vertex of P_1 . Each convex polytope P is defined by a finite set of linear inequalities $Ax \leq 0$. Each vertex v of P is defined as the unique solution of the relations from $Ax \leq 0$ that are satisfied with equality in $Av = 0$. In particular, the row vectors of the relations satisfied with equality in $Av \leq 0$ span the space of solutions $\mathbb{R}^{\dim(x)} = \mathbb{R}^{\#\{\text{variables}\}}$.

Thus, for Example 1, if we want to show that p is a vertex of the polytope P_1 , we shall show that the span of the row vectors of the matrix A coming from Eqs. (5), (6), (8), (9), (4), (7), and the linear relaxation of (10), (11), (12), that are satisfied with equality in $Ap \leq 0$ span \mathbb{R}^{21} .

Notice that some rows of A are vectors $e_i = (0, \dots, 0, 1, 0, \dots, 0)$, (that is, vectors of the canonical base of \mathbb{R}^{21}). Since we want to show that some row vectors span \mathbb{R}^{21} , both the rows containing these canonical vectors e_i , as well as the columns indexed by the corresponding unique non-zero coordinate i , may be omitted from our considerations. This submatrix of the rows A that are satisfied with equality at p is indicated in Fig. A.7.

Now, using Gaussian elimination, the matrix from Fig. A.7 is row-equivalent to that in Fig. A.8. Since

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & -1 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 0 & -1 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & -1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

we show that the vectors induced by the equations that are satisfied with equality for the evaluation given by Fig. 1 span $\mathbb{R}^{\#\{\text{variables}\}} = \mathbb{R}^{21}$, and hence the point p is a vertex of the polytope.

3.1.2. p is a vertex of P_1 . Second (alternative) proof of Proposition 1

Again, our goal is to show that p is a vertex of P_1 . To argue that the point p , given by the values from Table 1, is indeed a vertex of the polytope P_1 , we shall show that for every line with non-zero direction vector $\vec{x} = (\vec{x}_1, \dots, \vec{x}_{21})$ and for every $\varepsilon > 0$, either $p + \varepsilon\vec{x}$ or $p - \varepsilon\vec{x}$ is outside the polytope. Every coordinate \vec{x}_i of \vec{x} corresponds, uniquely, to a variable B, CY, or DY.

First observe that if \vec{x}_i is the coordinate related to a variable that is either 0 or 1 in p , then $\vec{x}_i = 0$, as otherwise, for any $\varepsilon > 0$, either $p + \varepsilon\vec{x}$ or $p - \varepsilon\vec{x}$ would be outside of the polytope P_1 . Hence, the only coordinates \vec{x}_i that may be non-zero are those for which the coordinate i in p is in the open interval $(0, 1)$.

The argument proceeds by contradiction. It assumes a sign on a non-zero coordinate of \vec{x} , say \vec{x}_i . By a chain of implications we conclude that another coordinate \vec{x}_j is non-zero and has the same sign as \vec{x}_i . By a different chain of implications we also conclude that \vec{x}_j has a different sign than \vec{x}_i , reaching a contradiction. Thus, initially the coordinate \vec{x}_i is zero. Furthermore, the fact that initially the coordinate \vec{x}_i is zero will imply that the coefficients used in the chains of implications are also zero. The details of the concrete argument are given below.

Remark 1. In Example 1, every equation from (5), (6), (8), (9) involves at most 2 variables on each side of the equality, one of them being either 0 or 1.

This observation forces the implications written below.

Assume, for instance, that the coefficient \vec{x}_i corresponding to B(2, 1, 1) in \vec{x} is negative. The following chain of implications can be drawn:

- by the flow constraints (8), the coefficient of B(1, 0, 1) is negative;
- then, by the flow constraints (8), the coefficient of B(1, 0, 0) is positive;
- then, by the flow constraints (8), the coefficient of B(2, 0, 0) is positive.

Now, using Eqs. (9) that link the variables B and DY, $B(2, 0, 1) + B(2, 1, 1) = DY(2, 1, 1) + DY(2, 1, 0)$, we obtain that the coefficient of $DY(2, 1, 1)$ is negative, which implies that

- the coefficient of $CY(2, 1, 1)$ in \vec{x} is negative;
- then, by flow constraints (5), the coefficient of $CY(1, 0, 1)$ is negative;
- then, by flow constraints (5), the coefficient of $CY(1, 0, 0)$ is positive;
- then, by flow constraints (5), the coefficient of $CY(2, 0, 0)$ is positive.

This implies that the coefficient of $DY(2, 0, 0)$ has to be positive. However, let us now look at the coefficients of $DY(1, 0, 1)$ and the one corresponding to $DY(1, 1, 0)$.

If we use Eqs. (9), the linking constraints between the variables DY and B, the coefficients \vec{x}_i' and \vec{x}_i'' corresponding to the variables $DY(1, 0, 1)$ and $DY(1, 1, 0)$ in \vec{x} have to be negative and positive, respectively. However, if we look at the constraints (6) linking the variables DY and CY, the coefficients should have the opposite sign (positive and negative, respectively). Thus we reach a contradiction, which shows that the coefficient \vec{x}_i corresponding to $B(2, 1, 1)$ is 0.

The fact that the coefficient \vec{x}_i corresponding to $B(2, 1, 1)$ is 0 implies, by again applying the same arguments as above using Remark 1, that all the other coefficients involved in the chains of implications have to be zero. This shows that no such non-zero vector \vec{x} exists.

3.2. Sufficient conditions for integrality of the optimal solution of the linear relaxation

Our findings in Section 3.1 show that the polytope is, in general, non-integral. Thus, a natural next step is to find some conditions on the objective function (1)+(2)+(3) that guarantee that either the linear relaxation of the integer program finds an integral point as a solution, or that there is an integral point in the optimal face of the convex polytope and a procedure to find it. Such conditions are presented in Proposition 2 and Corollary 1.

Proposition 2. Consider the Zwaneveld–Verweij IP model from Section 2, and assume that the coefficients of the objective function (1)+(2)+(3) satisfy:

- (i) for every $t, d, h_2, h_2', h_2^B, h_2'^B$ such that $h_2 \leq h_2'$ and $h_2^B \leq h_2'^B$,

$$\begin{aligned} & D_{\text{expdam}}(t, d, h_2', h_2^B) + D_{\text{expdam}}(t, d, h_2, h_2'^B) \\ & \geq D_{\text{expdam}}(t, d, h_2, h_2^B) + D_{\text{expdam}}(t, d, h_2', h_2'^B) \end{aligned} \tag{13}$$

- (ii) for every $t, h_1^B, h_1'^B, h_2^B, h_2'^B$ such that $h_1^B \leq h_1'^B$ and $h_2^B \leq h_2'^B$,

$$\begin{aligned} & B_{\text{cost}}(t, h_1^B, h_2'^B) + B_{\text{cost}}(t, h_1'^B, h_2^B) \\ & \geq B_{\text{cost}}(t, h_1^B, h_2^B) + B_{\text{cost}}(t, h_1'^B, h_2'^B) \end{aligned} \tag{14}$$

- (iii) for every $t, d, h_1, h_1', h_2, h_2'$ such that $h_1 \leq h_1'$ and $h_2 \leq h_2'$,

$$\begin{aligned} & D_{\text{cost}}(t, d, h_1, h_2') + D_{\text{cost}}(t, d, h_1', h_2) \\ & \geq D_{\text{cost}}(t, d, h_1, h_2) + D_{\text{cost}}(t, d, h_1', h_2'). \end{aligned} \tag{15}$$

Then, there is an optimal solution of the LP relaxation of the Zwaneveld–Verweij IP model with integer coordinates.

Note that the term involving $B_{\text{expdam}}(t, h_2^B)$ from (3) does not appear in condition (ii) of Proposition 2 as it appears on both sides of the inequality.

The conditions (i)–(iii) are satisfied by a wide range of reasonable cost functions. Observe that all inequalities that appear in conditions (i)–(iii) from Proposition 2 are of a similar form: for $a \leq a'$ and $b \leq b'$ we have that some function $c(\cdot, \cdot)$ satisfies $c(a, b) + c(a', b') \leq c(a, b') + c(a', b)$. Such an inequality for c is naturally satisfied (in fact with equality) if c is of the form $c(x, y) = f(y) - f(x) + c_0$ for some function f and constant c_0 . In the context of conditions (ii) and (iii) from Proposition 2, such a form is somewhat reasonable to expect: the cost of rising a dike from level x to y compares to the cost of rising the dike from level 0 to y , minus the effort already made to rise it from 0 to x , plus perhaps some extra cost c_0 .

Moreover, assuming an exponential economical growth, it seems reasonable to assume that if we want to protect an area by increasing the height of a dike, we will try to protect the same area in the future even more. Thus, the increments in the dike heights are absolutely ordered up to a point where the costs are higher than the benefits, and hence there is a unique optimum path.

From a technical perspective, each of the condition sets (i), (ii) and (iii) from Proposition 2 independently provides structural information on the feasible set which help towards finding the integer solution of the IP problem (see Steps 1, 2 and 3 in the proof of Proposition 2). Indeed, conditions (i)–(iii) guarantee the optimal solution of the LP relaxation to be integral via the following argument: the conditions (i)–(iii) induce an ordering on the possibly fractional solutions of the barrier and the dikes.

The partial solutions of the barrier can be paired with the partial solutions of the dikes in a way compatible with the ordering. This compatibility permits us to view the problem as a min-cost flow problem (and it is known that the solution to the LP relaxation of the min-cost flow problem is integral.) Both the ordering of the partial solutions and the mentioned pairing can be achieved without altering the optimality of the solution thanks to the conditions (i)–(iii).

Analogous arguments apply to conditions (ii), (iii) and (iv) of Corollary 1.

Proof of Proposition 2. The Zwaneveld–Verweij IP model can be thought of as several min-cost flow problems, one for each dike and one for the barrier. However, these problems are not independent; they are linked to each other using the variables DY. An example of this interpretation can be seen in Fig. 1. The graph G that codifies the IP problem is the one constructed in Section 2.1. We use the same notation in this proof. Observe that, for each $d \in D$ and $t \in T$, the vertices $\{(t, d, h)\}_{h \in H_D}$ can be ordered using the height h :

$$v = (t, d, h) <_v u = (t, d, h') \text{ if and only if } h < h'.$$

This induces a partial order on the vertices of G_d for $d \in D$. The vertices in G_B can be similarly partially ordered.

The problem is translated as follows. Variables $B(0, 0, 0)$ and $D(0, d, 0, 0)$ at $t = 0$ represent a flow input of 1. Eqs. (5), together with initializations (4) get translated into the property that, for each $d \in D$, there is a flow of one unit through G_d (the graph induced by each of the dikes). Eqs. (8), together with the initializations (7), get translated into the following property: there is a flow of 1 unit through G_B (the graph induced by the barrier). Recall that the flow through a vertex v is the sum of the inflows of the edges attached to the vertex v . Eqs. (6) and (9) make the variables DY behave as pairing variables: if $\text{DY}(t, d, h, h^B) = f$ then a flow of f through the vertex (t, d, h) is paired with a flow of f through the vertex (t, h^B) . Finally, ((1)+(2)+(3)) is the linear objective function for this intertwined min-cost flow problem. Now, any solution of the IP gives a flow, and vice-versa: each of the variables B and CY indicates the amount of flow through the corresponding edge, and the quantity DY indicates the amount paired between the two ends of the edge. See Section 5 for further details.

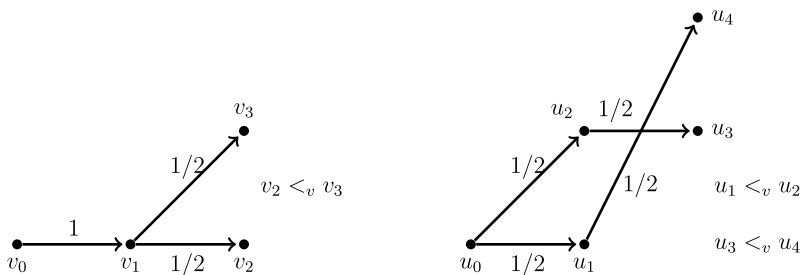


Fig. 2. Flow path $v_0v_1v_3$ is above flow path $v_0v_1v_2$ and they form a layered flow. Flow paths $u_0u_1u_4$ and $u_0u_2u_3$ do not form a layered flow. Note that edges u_1u_4 and u_2u_3 form a crossing.

Let \vec{x} be a solution satisfying Eqs. (5), (6), (8) and (9) (satisfying the equations of an intertwined flow problem). Then edges $e_1 = u_1v_1$ and $e_2 = u_2v_2$ are said to form a *crossing* in the solution \vec{x} if the pairs of vertices (u_1, u_2) and (v_1, v_2) are comparable, and $u_1 <_v u_2$ and $v_1 >_v v_2$, and the value of both e_1 and e_2 in \vec{x} is non-zero. If both edges e_1 and e_2 belong to the same group of variables, say B, CY, or DY, then we say that e_1 and e_2 form a B-crossing, a CY-crossing or a DY-crossing, respectively. For an illustration, in Fig. 1, edges $DY(1, 1, 0)$ and $DY(1, 0, 1)$ form a DY-crossing if we order the vertices in each layer (t, d, \cdot) by its height h or h_B .

The proof of Proposition 2 consists of three steps:

Step 1 First, we reroute the flow in the graph (modifying the variables B and CY) to obtain a “layered” flow (see the definition below). The first step will be completed without increasing the total cost due to conditions (ii) and (iii).

Step 2 In the second step we modify the variables DY while keeping B and CY constant. For the second step we utilize condition (i) to see that the solution can be modified while not increasing the total cost.

Step 3 In the third step we use standard flow-rerouting arguments in graphs to find the desired integral solution (while not increasing the total cost) using a Ford–Fulkerson-algorithm-type argument.

Step 1 and Step 2 are the preparation for Step 3, which uses the properties of the solution in Step 1 and Step 2 to finish the argument. Namely, we first create a layered flow (Step 1), then do the uncrossings of the DY in conjunction with the layered flow (Step 2). Afterwards we pair up the flow paths: top path (of the barrier) with top path (from the dikes) and bottom path with bottom path; We use the fact that the DY variables are not crossing to iteratively redirect the flow into a single path (Step 3).

In what follows, we give the details of the above steps.

Step 1 Consider a graph G' . A path $p = (v_1, e_1, v_2, e_2, \dots, v_n)$ in G' , with vertices v_1, v_2, \dots, v_n and edges e_1, e_2, \dots, e_{n-1} , is a *flow path* when the amount of flow through each edge of the path is the same (that is, there is a valid flow through p from v_1 to v_n , even though the edges in G' of the path may carry a different flow). If the vertices of the graph have a partial order $<_v$, then a flow path $p_2 = v_1, e_1, v_2, e_2, \dots, v_n$ through the vertices v_1, v_2, \dots, v_n is said to be *above* a flow path $p_1 = w_1, e'_1, w_2, e'_2, \dots, w_n$ on the vertices w_1, w_2, \dots, w_n when $v_i \geq_w w_i$ for all i , and we denote it by $p_1 <_a p_2$. The *above* relation, $<_a$, induces a partial order on the set of paths. A flow in a graph G' is said to be *layered* if there exist a positive integer ℓ and flow paths p_1, \dots, p_ℓ that are totally ordered ($p_i <_a p_j$ for $i < j$) and that decompose the flow: the addition of the flows through the flow paths p_1, \dots, p_ℓ is the original flow. See Fig. 2 for an example.

Claim 1. For each feasible solution \vec{y}_0 of the IP, and under the assumptions of Proposition 2, there exists a feasible solution \vec{y}_1 for which

- for each $i \in D \cup \{B\}$, the flow induced in G_i by \vec{y}_1 is layered,
- the value of the objective at \vec{y}_1 is not larger than at \vec{y}_0 ,
- restricted to the variables DY , the values of \vec{y}_0 and \vec{y}_1 are equal.

In particular, some optimal solution induces a layered flow on each induced graph G_i , $i \in D \cup \{B\}$.

Proof of Claim 1. Let us begin by showing the statement for the graphs induced by the dikes G_d . The modifications to the argument for the case of the barrier G_B are discussed towards the end of the proof.

For each dike d , and each time t , let $h_{t,d}$ denote the maximum height in H_D for which the vertex $(t, d, h_{t,d})$ in G_d has a non-zero flow through it:

$$h_{t,d} = \max \left\{ h, \text{ subject to: } h \in H_D \text{ and } \sum_{h_1 \leq h} \text{CY}(t, d, h_1, h) > 0 \right\}.$$

Note that $\sum_{h_1 \leq h} \text{CY}(t, d, h_1, h) > 0$ indicates that there is a non-zero flow through the vertex (t, d, h) . Let $p_{\ell,d}$ be a flow path with vertices $\{(t, d, h_{t,d})\}_{t \in T}$ and with flow

$$\text{fl}(p_{\ell,d}) = \min_{t \in T} \left\{ \sum_{h_1 \leq h_{t,d}} \text{CY}(t, d, h_1, h_{t,d}) > 0 \right\}.$$

That is, the value of the flow in $p_{\ell,d}$ is restricted by the flow of the vertex in $p_{\ell,d}$ having the minimum amount of flow among the vertices of $p_{\ell,d}$. The flow path $p_{\ell,d}$ contains the edges $\text{CY}(t, d, h_{t-1,d}, h_{t,d})$ between vertices $(t-1, d, h_{t-1,d})$ and $(t, d, h_{t,d})$; thus, a flow of $\text{fl}(p_{\ell,d})$ through $p_{\ell,d}$ is allowed by the feasible solution \vec{y}_0 if and only if, for each $t \in T$,

$$\text{CY}(t, d, h_{t-1,d}, h_{t,d}) \geq \text{fl}(p_{\ell,d}). \tag{16}$$

If a flow $\text{fl}(p_{\ell,d})$ through $p_{\ell,d}$ is allowed by \vec{y}_0 (conditions (16) are satisfied), then we subtract a flow of $\text{fl}(p_{\ell,d})$ from the edges of $p_{\ell,d}$, inducing a valid $[1 - \text{fl}(p_{\ell,d})]$ -valued flow through the graph G_d . The same procedure is then applied to this new flow in G_d . If conditions (16) are satisfied for the new $[1 - \text{fl}(p_{\ell,d})]$ -flow through G_d , then the procedure can be repeated. Since the number of vertices with flow through it strictly decreases from one step to the next, the procedure is guaranteed to eventually finish.

Repeating this procedure $i \geq 0$ times – so (16) is satisfied at each of the i steps – we obtain flow i paths $p_{\ell,d}, \dots, p_{\ell-i+1,d}$ such that $p_{\ell-i+1,d} <_a p_{\ell-i+2,d} <_a \dots <_a p_{\ell,d}$. If after finding $i \geq 0$ flow paths, conditions (16) are not satisfied, then by a pigeonhole argument we can find a CY-crossing between the variables $\text{CY}(t, d, h_1, h_{t,d})$ and $\text{CY}(t, d, h_{t-1,d}, h_2)$. That is, there exist t, h_1, h_2 with $h_1 < h_{t-1,d}$ and $h_2 < h_{t-1,d}$, for which either

$$0 < \text{CY}(t, d, h_1, h_{t,d}) \leq \text{CY}(t, d, h_{t-1,d}, h_2) < 1 \tag{17}$$

or

$$1 > \text{CY}(t, d, h_1, h_{t,d}) \geq \text{CY}(t, d, h_{t-1,d}, h_2) > 0. \tag{18}$$

Then we modify the feasible solution \vec{y}_0 via one of the following transformations (see illustration in Fig. 3), if (17) occurs, then (\rightarrow indicates “turns into”)

$$\begin{cases} \text{CY}(t, d, h_1, h_2) & \rightarrow \text{CY}(t, d, h_1, h_2) + \text{CY}(t, d, h_1, h_{t,d}) \\ \text{CY}(t, d, h_1, h_{t,d}) & \rightarrow \text{CY}(t, d, h_1, h_{t,d}) - \text{CY}(t, d, h_1, h_{t,d}) = 0 \\ \text{CY}(t, d, h_{t-1,d}, h_2) & \rightarrow \text{CY}(t, d, h_{t,d}, h_2) - \text{CY}(t, d, h_1, h_{t,d}) \\ \text{CY}(t, d, h_{t-1,d}, h_{t,d}) & \rightarrow \text{CY}(t, d, h_{t-1,d}, h_{t,d}) + \text{CY}(t, d, h_1, h_{t,d}); \end{cases} \tag{19}$$

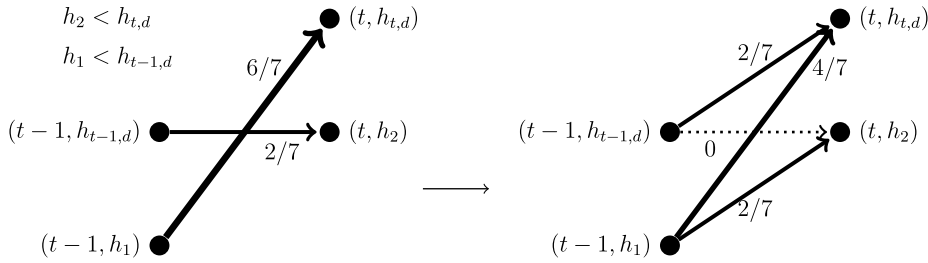


Fig. 3. Assuming (18), thus using (20) to undo a CY (or B) crossing.

if (18) occurs, then

$$\begin{cases}
 \text{CY}(t, d, h_1, h_2) & \rightarrow \text{CY}(t, d, h_1, h_2) + \text{CY}(t, d, h_{t-1,d}, h_2) \\
 \text{CY}(t, d, h_1, h_{t,d}) & \rightarrow \text{CY}(t, d, h_1, h_{t,d}) - \text{CY}(t, d, h_{t-1,d}, h_2) \\
 \text{CY}(t, d, h_{t-1,d}, h_2) & \rightarrow \text{CY}(t, d, h_{t-1,d}, h_2) - \text{CY}(t, d, h_{t-1,d}, h_2) = 0 \\
 \text{CY}(t, d, h_{t-1,d}, h_{t,d}) & \rightarrow \text{CY}(t, d, h_{t-1,d}, h_{t,d}) + \text{CY}(t, d, h_{t-1,d}, h_2).
 \end{cases} \tag{20}$$

This transformation provides a new feasible solution \vec{y}_1 not worse than \vec{y}_0 . Indeed, the value of the objective function at \vec{y}_1 has not increased with respect to the value of the objective function at \vec{y}_0 by condition (15). Feasibility of \vec{y}_1 follows from three facts.

The first fact is that, if there is a crossing between $\text{CY}(t, d, h_1, h_{t,d})$ and $\text{CY}(t, d, h_{t-1,d}, h_2)$, then $h_{t-1,d} \leq h_2$. By definition of $h_{t-1,d}$ we have $h_1 \leq h_{t-1,d}$; hence $h_1 \leq h_2$ and $\text{CY}(t, d, h_1, h_2)$ is a valid variable and hence an edge of the graph. Since $\text{CY}(t, d, h_1, h_{t,d})$ and $\text{CY}(t, d, h_{t-1,d}, h_2)$ are assumed to be crossing, and by definition of $h_{t,d}$ we have $h_2 \leq h_{t,d}$, then $h_2 < h_{t,d}$ and also $h_{t-1,d} < h_{t,d}$; thus $\text{CY}(t, d, h_{t-1,d}, h_{t,d})$ is also a valid variable and hence an edge of the graph. This fact guarantees that the operations in (19) or (20) are allowed (the variables are defined).

The second fact is that the sums $\sum_{h_1 \leq h} \text{CY}(t-1, d, h_1, h)$ and $\sum_{h_1 \geq h} \text{CY}(t, d, h, h_1)$ are equal for both \vec{y}_0 and \vec{y}_1 (the same value is added and subtracted to these sums); then, since the variables DY and the B's remain constant, all Eqs. (4), (5), (6), (7), (8), (9) are satisfied. Moreover, the linear relaxations of (11) and of (12), are also satisfied (the variables DY and B have not been modified).

The third and final fact states that the linear relaxation of (10) is also satisfied. Indeed, if (17) holds, then the variables are modified by transformation (19), and all variables CY remain non-negative (some are not modified, and those from which something is subtracted, their non-negativity is preserved). In case of (18) being satisfied, the conclusion on the non-negativity of CY holds as well. Moreover, values of the variables CY remain below 1 as the total flow through G_d is 1, and we have already shown that the flow through each edge is positive (non-negative), while the flow equations for G_d are satisfied by \vec{y}_1 , hence the flow through each edge is between 0 (non-negative) and 1 (the maximum total flow). This shows that (10) is also satisfied for \vec{y}_1 .

After feasibility and the fact that the value of the objective function at \vec{y}_1 is as good as the value of the objective function at \vec{y}_0 has been established, we continue with the main argument. The flow induced by \vec{y}_1 in G_d contains one less CY-crossing. After all CY-crossings have been eliminated using the transformations (19) and (20), condition (16) is satisfied and a new flow path $p_{\ell-i,d}$ can be found. Thus, for $G_1, \dots, G_{|D|}$ we obtain layered flows.

This procedure can be applied analogously to the barrier; in this case we shall use condition (14) instead of (15) to undo the B-crossings without increasing the cost (while remaining a feasible solution), and hence finding a layered flow for the graph G_B of the barrier. This completes the proof of Claim 1, as the 'in particular' part of the statement is an immediate consequence of the main part of the claim. \square

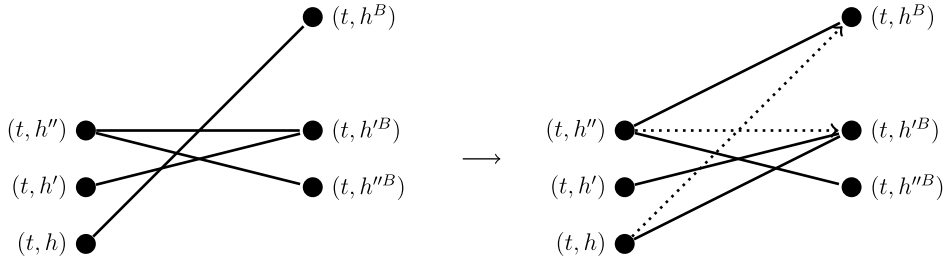


Fig. 4. On the left: crossing of (h, h^B) , (h', h'^B) is on top of crossing (h, h^B) , (h'', h''^B) which is on top of the crossing (h', h'^B) , (h'', h''^B) . Crossing of (h, h^B) , (h, h'^B) is the topmost one. On the right, we undo the topmost crossing using (21) or (22); other crossings are created but they are below the original one.

Step 2 Let us now move to the second step.

Claim 2. If \bar{y}_0 is a feasible solution of the linear relaxation, and under the assumptions of Proposition 2, then either \bar{y}_0 contains no DY-crossing, or there exists another feasible solution \bar{y}_1 such that

- \bar{y}_1 has no DY-crossings,
- the vector \bar{y}_1 , restricted to the variables B and CY , equals \bar{y}_0 ,
- the value of the objective function for \bar{y}_1 is not larger than the value of the objective function for \bar{y}_0 .

In particular, an optimal solution can be assumed to have no DY-crossings by modifying only the values of the variables DY .

Proof of Claim 2. Let $<_v$ be a partial order on the vertices. The crossing formed by the edges $e_1 = u_1v_1$ and $e_2 = u_2v_2$ is said to be *on top* of the crossing formed by $e_3 = u_3v_3$, $e_4 = u_4v_4$ if (v_1, v_2, u_2, u_1) (with $v_1 >_v v_2$) is larger than (v_3, v_4, u_4, u_3) (with $v_3 >_v v_4$) in the lexicographical order: each coordinate of (v_1, v_2, u_2, u_1) can be compared with the corresponding coordinate in (v_3, v_4, u_4, u_3) and the value of the first coordinate in which they differ, is strictly larger in (v_1, v_2, u_2, u_1) (either $v_1 >_v v_3$, or if $v_1 = v_3$ then $v_4 >_v v_2$, and so on). See Fig. 4 for an illustration.

To make the argument clearer, we shall assume that the vertices are ordered by non-decreasing height.

For each $t \in T$ and $d \in D$, we undo all crossings between the edges $DY(t, d, \cdot, \cdot)$. Let the topmost DY-crossing be between the edges $DY(t, d, h', h^B)$ and $DY(t, d, h, h'^B)$, with $h < h'$, $h^B < h'^B$. Then we can modify the solution of the linear relaxation \bar{y}_0 to obtain another feasible solution \bar{y}_1 as follows. See Fig. 4 for an illustration of the procedure (from the left-hand figure, to the right-hand figure).

If $DY(t, d, h', h^B) \geq DY(t, d, h, h'^B)$ then (\rightarrow indicates “turns into”)

$$\begin{cases} DY(t, d, h', h^B) & \rightarrow DY(t, d, h', h^B) - DY(t, d, h, h'^B) \geq 0 \\ DY(t, d, h, h'^B) & \rightarrow DY(t, d, h, h'^B) - DY(t, d, h, h'^B) = 0 \\ DY(t, d, h, h^B) & \rightarrow DY(t, d, h, h^B) + DY(t, d, h, h'^B) \\ DY(t, d, h', h'^B) & \rightarrow DY(t, d, h', h'^B) + DY(t, d, h, h'^B). \end{cases} \tag{21}$$

If $DY(t, d, h', h^B) < DY(t, d, h, h'^B)$ then

$$\begin{cases} DY(t, d, h', h^B) & \rightarrow DY(t, d, h', h^B) - DY(t, d, h', h^B) = 0 \\ DY(t, d, h, h'^B) & \rightarrow DY(t, d, h, h'^B) - DY(t, d, h', h^B) > 0 \\ DY(t, d, h, h^B) & \rightarrow DY(t, d, h, h^B) + DY(t, d, h', h^B) \\ DY(t, d, h', h'^B) & \rightarrow DY(t, d, h', h'^B) + DY(t, d, h', h^B). \end{cases} \tag{22}$$

The cost of \bar{y}_1 is at most \bar{y}_0 due to (13). Furthermore, $DY(t, d, h, h'^B)DY(t, d, h', h^B) = 0$, so $DY(t, d, h', h^B)$ and $DY(t, d, h, h'^B)$ do not cross in \bar{y}_1 .

Even though some DY-crossing between edges $DY(t, d, \cdot, \cdot)$ might have been created in \vec{y}_1 with respect to \vec{y}_0 , all are smaller in the lexicographical order. Indeed, all the crossings involving one edge from the set $\{DY(t, d, h', h^B), DY(t, d, h, h'^B)\}$, either do not exist as the variable has value 0 or the crossing is smaller in the lexicographical order. The latter is shown by observing that the modified edges $DY(t, d, h', h'^B)$ and $DY(t, d, h, h^B)$ do not cross any edge from the set $\{DY(t, d, h', h^B), DY(t, d, h, h'^B)\}$, and the crossings involving one edge from the set $\{DY(t, d, h', h^B), DY(t, d, h, h'^B)\}$ already occur for \vec{y}_0 , but by assumption we picked the topmost DY-crossing in \vec{y}_0 . This fact is argued in the following two paragraphs.

Each crossing involving the edge $DY(t, d, h, h^B)$ in \vec{y}_1 is below the $\{DY(t, d, h', h^B), DY(t, d, h, h'^B)\}$. Indeed, if $\{DY(t, d, h'', h''^B), DY(t, d, h, h^B)\}$ cross in \vec{y}_1 , and $h''^B < h^B$, then the crossing (h^B, h''^B, h'', h) is below (h^B, h^B, h', h) as, recall, $h'^B > h^B$. In the case that $h^B < h''^B$, then $h'' < h$ and there is a crossing $\{DY(t, d, h', h^B), DY(t, d, h'', h''^B)\}$ in \vec{y}_0 (as $h < h'$ and $h'^B > h^B$) with (h''^B, h^B, h', h'') being larger than (h''^B, h^B, h, h'') . So the $\{DY(t, d, h'', h''^B), DY(t, d, h, h^B)\}$ crossing in \vec{y}_1 is below the maximal one in $\{DY(t, d, h', h^B), DY(t, d, h, h'^B)\}$ for \vec{y}_0 .

Now, assume that the edge $DY(t, d, h', h'^B)$ creates a crossing with $DY(t, d, h'', h''^B)$ in \vec{y}_1 , and that is larger in lexicographical order than the one between $DY(t, d, h', h^B)$ and $DY(t, d, h, h'^B)$. If $h'^B < h''^B$, then $h'' < h'$, and the edges $DY(t, d, h'', h''^B)$ and $DY(t, d, h', h^B)$ also cross in \vec{y}_0 while (h''^B, h'^B, h, h'') is larger in the lexicographical order than (h'^B, h^B, h', h) , contradicting the maximality of the (h'^B, h^B, h', h) cross in \vec{y}_0 . Therefore, we should have $h''^B < h'^B$ and thus $h' < h''$ (as they are crossing). By the assumption that (h'^B, h''^B, h'', h') is larger in the lexicographical order than (h'^B, h^B, h', h) we have $h''^B \geq h^B$. Hence, the edges $DY(t, d, h'', h''^B)$ and $DY(t, d, h, h'^B)$ cross for \vec{y}_0 (as $h''^B < h'^B$ and $h < h' < h''$) but then (h'^B, h''^B, h'', h) is larger than (h'^B, h^B, h', h) in the lexicographical order, contradicting the maximality of the crossing between $DY(t, d, h, h'^B)$ and $DY(t, d, h', h^B)$ in \vec{y}_0 . As this procedure that removes crossings can be conducted in such a way that the topmost crossing in the lexicographical order strictly decreases from one step to the next, the process finishes in a finite number of steps.

Additionally, the values on the variables B and CY have not been modified by the transformations (21) or (22) in \vec{y}_1 with respect to \vec{y}_0 . Furthermore, since the sums

$$\sum_{h \in H_D} DY(t, d, h, h^B) \text{ and } \sum_{h^B \in H_B} DY(t, d, h, h^B)$$

maintain its value in \vec{y}_1 with respect to \vec{y}_0 , then \vec{y}_1 is also a feasible solution of the linear relaxation. Therefore, this process finishes and a feasible solution \vec{y}_1 is found which is at least as good as \vec{y}_0 , with no DY-crossings and with the same values as \vec{y}_1 for the variables B and CY as claimed. This completes the proof of Claim 2. \square

Step 3 We can finally move to the third step.

Claim 3. *If \vec{y}_0 is a feasible solution of the linear relaxation, and under the assumptions of Proposition 2, then there exists a feasible solution \vec{y}_1 such that*

- \vec{y}_1 is integral
- the objective function at \vec{y}_1 is not larger than the one for \vec{y}_0 .

Proof. By Claim 1 the flows induced by \vec{y}_0 on each $G_i, i \in D \cup \{B\}$ can be assumed to be layered. Since Claim 2 does not modify the edges CY, the edges DY can be assumed to be pairwise non-crossing. Hence we have a layered flow with non-crossing edges DY in \vec{y}_0 .

Let $p_{1,i}, \dots, p_{\ell_i,i}$ denote the flow paths of the layered flow in the graph G_i induced by \vec{y}_0 , with $i \in D \cup \{B\}$. Assume moreover that, for each $i \in D \cup \{B\}$, $p_{j_1,i}$ is below $p_{j_2,i}$ whenever $j_1 < j_2$ (recall that this is denoted

by $p_{j_1,i} <_a p_{j_2,i}$). Let $L_1, \dots, L_{|D|}$ denote the (lower) flow paths $p_{1,1}, \dots, p_{1,|D|}$; that is, L_d is the minimal non-zero flow path in G_d with respect to the $<_a$ ordering on the flow paths. Likewise, L_B is the minimal flow path for the graph G_B , so $L_B = p_{1,B}$. Similarly, $U_1, \dots, U_{|D|}$ denote the topmost flow paths $p_{\ell_1,1}, \dots, p_{\ell_{|D|},|D|}$ of the layered flow; that is, $U_i = p_{\ell_i,i}$. Likewise, $U_B = p_{\ell_B,B}$, is the topmost flow path induced by \vec{y}_0 in the graph G_B .

Assume \vec{y}_0 is non-integral, whence at least a flow variable CY or B is non-integral, and there exists an $i \in D \cup \{B\}$ for which $U_i \neq L_i$. The plan is to reroute as much flow as possible, either from $[U_1, \dots, U_{|D|}, U_B]$ to $[L_1, \dots, L_{|D|}, L_B]$, or vice-versa, depending on the cost function; we say that a flow of value f is rerouted from $[U_1, \dots, U_{|D|}, U_B]$ to $[L_1, \dots, L_{|D|}, L_B]$ if for each $i \in D \cup \{B\}$, the flow through U_i is lowered by f , and the flow through L_i is increased by f .

Consider

$$\begin{cases} l_{\text{in}} := \min_{i \in D \cup \{B\}} \{1 - \text{fl}(L_i)\} & \text{max. flow the lower paths can take} \\ l_{\text{out}} := \min_{i \in D \cup \{B\}} \{\text{fl}(L_i)\} & \text{max. flow the lower paths can give} \\ u_{\text{in}} := \min_{i \in D \cup \{B\}} \{1 - \text{fl}(U_i)\} & \text{max. flow the upper paths can take} \\ u_{\text{out}} := \min_{i \in D \cup \{B\}} \{\text{fl}(U_i)\} & \text{max. flow the upper paths can give.} \end{cases}$$

Since (1)+(2)+(3) is linear, we can determine whether rerouting from $[U_1, \dots, U_{|D|}, U_B]$ to $[L_1, \dots, L_{|D|}, L_B]$ does not increase the value of the objective function, or vice-versa. Assume the first case. Then we reroute a flow of through-flow $\text{trfl} = \min\{u_{\text{out}}, l_{\text{in}}\}$ from $[U_1, \dots, U_{|D|}, U_B]$ to $[L_1, \dots, L_{|D|}, L_B]$. That is, consider the flow paths with edges

$$\begin{aligned} U_i &= (1, i, h_{0,u_i}, h_{1,u_i}), \dots, (|T|, i, h_{|T|-1,u_i}, h_{|T|,u_i}), \\ L_i &= (1, i, h_{0,u_i}, h_{1,l_i}), \dots, (|T|, i, h_{|T|-1,l_i}, h_{|T|,l_i}), \quad i \in D \cup \{B\}; \end{aligned}$$

then a feasible solution \vec{y}_1 with cost value at most that of \vec{y}_0 , is obtained from \vec{y}_0 by updating the variables as follows (\rightarrow indicates “turns into”):

$$\begin{cases} \text{CY}(t, d, h_{t-1,u_d}, h_{t,u_d}) \rightarrow \text{CY}(t, d, h_{t-1,u_d}, h_{t,u_d}) - \text{trfl} & \text{for } d \in D, t \in T \\ \text{B}(t, h_{t-1,u_B}, h_{t,u_B}) \rightarrow \text{B}(t, h_{t-1,u_B}, h_{t,u_B}) - \text{trfl} & \text{for } t \in T \end{cases} \quad (23)$$

$$\begin{cases} \text{CY}(t, d, h_{t-1,l_d}, h_{t,l_d}) \rightarrow \text{CY}(t, d, h_{t-1,l_d}, h_{t,l_d}) + \text{trfl} & \text{for } d \in D, t \in T \\ \text{B}(t, h_{t-1,l_B}, h_{t,l_B}) \rightarrow \text{B}(t, h_{t-1,l_B}, h_{t,l_B}) + \text{trfl} & \text{for } t \in T \end{cases} \quad (24)$$

$$\text{DY}(t, d, h_{t,u_d}, h_{t,u_B}) \rightarrow \text{DY}(t, d, h_{t,u_d}, h_{t,u_B}) - \text{trfl} \quad \text{for } d \in D, t \in T \quad (25)$$

$$\text{DY}(t, d, h_{t,l_d}, h_{t,l_B}) \rightarrow \text{DY}(t, d, h_{t,l_d}, h_{t,l_B}) + \text{trfl} \quad \text{for } d \in D, t \in T \quad (26)$$

The variables that do not appear in (23), (24), (25), (26) are not modified. Some variables that appear may be untouched when $L_i = U_i$ at a particular edge.

Let us show that the updated vector \vec{y}_1 is a feasible solution. It is clear that \vec{y}_1 satisfies (4), (5), (6), (7), (8), (9). It remains to show that it also satisfies the linear relaxations of (10), (11) and (12).

Since the flow is layered, and by the definition of trfl , each variable represented in (23) is non-negative, and each variable represented in (24) is at most 1.

Since the total flow through G_d , $d \in D$ and G_B is 1, it suffices to show that the new variables in (25) are non-negative in order to prove that the new variables in (26) are at most 1 as well. Let us

proceed by contradiction and assume that there is a negative variable due to the operation (25); denote by $(t, d, h_{t,u_d}, h_{t,u_B})$ the variable for which $DY(t, d, h_{t,u_d}, h_{t,u_B}) - \text{trfl} < 0$. By the relations (6) and (9), and since there is a flow of at least trfl through the vertices (t, d, h_{t,u_d}) and (t, h_{t,u_B}) in the original solution \vec{y}_0 , we have

$$\sum_{h \in H_D} DY(t, d, h, h_{t,u_B}) \geq \text{trfl} \quad \text{and} \quad \sum_{h^B \in H_B} DY(t, d, h_{t,u_d}, h^B) \geq \text{trfl}.$$

By a pigeonhole argument, and since we are considering the uppermost flow paths U_B and U_d , if $DY(t, d, h_{t,u_d}, h_{t,u_B}) - \text{trfl} < 0$ then there exist $h_0^B < h_{t,u_B}$ and $h_0 < h_{t,u_d}$ with $DY(t, d, h_{t,u_d}, h_0^B) > 0$ and $DY(t, d, h_0, h_{t,u_B}) > 0$ for the solution \vec{y}_0 . Thus there is a DY-crossing between $DY(t, d, h_{t,u_d}, h_0^B)$ and $DY(t, d, h_0, h_{t,u_B})$, which is a contradiction as \vec{y}_0 was assumed to have no DY-crossings.

An analogous argument shows that rerouting an amount of $\min\{l_{\text{out}}, u_{\text{in}}\}$ from the L 's to the U 's also creates a feasible solution.

After applying this argument once, either the total number of flow paths is reduced by at least 1, either because the flow through a path has been reduced to 0, or the flow through another flow path has been increased to 1. Since the total number of flow paths is finite, this procedure finishes and all the flow is rerouted towards a single flow path for each G_d and G_B . Due to the linearity of the objective function, this flow rerouting can be achieved without increasing the value of the objective function. This finishes the proof of Claim 3. \square

The proof of Proposition 2 is finished after observing that Claim 3 directly shows its statement. \square

Next, we show that condition (i) in Proposition 2 can be replaced by condition (iv) in Corollary 1 to obtain the same conclusion.

Corollary 1. *Assume that conditions (ii), (iii) from Proposition 2 hold. Assume further that the following conditions on the coefficients of the objective function (1)+(2)+(3) hold:*

(iv) *for each dike $d \in D$, either*

$$\begin{aligned} & D_{\text{expdam}}(t, d, h'_2, h_2^B) + D_{\text{expdam}}(t, d, h_2, h_2'^B) \\ & \leq D_{\text{expdam}}(t, d, h_2, h_2^B) + D_{\text{expdam}}(t, d, h'_2, h_2'^B) \end{aligned} \tag{27}$$

for every $t, h_2, h'_2, h_2^B, h_2'^B$ such that $h_2 \leq h'_2$ and $h_2^B \leq h_2'^B$, or

$$\begin{aligned} & D_{\text{expdam}}(t, d, h'_2, h_2^B) + D_{\text{expdam}}(t, d, h_2, h_2'^B) \\ & \geq D_{\text{expdam}}(t, d, h_2, h_2^B) + D_{\text{expdam}}(t, d, h'_2, h_2'^B) \end{aligned} \tag{28}$$

for every $t, h_2, h'_2, h_2^B, h_2'^B$ such that $h_2 \leq h'_2$ and $h_2^B \leq h_2'^B$.

Then the linear relaxation of the Zwaneveld–Verweij IP model from Section 2 admits an integral optimal solution.

An important observation is that (iv) is more general than condition (i): if (i) are satisfied, then (iv) are also satisfied. The arguments of the proof of Claim 1, however, are easily reduced to those used in the proof of Proposition 2.

Proof. We modify the argument of the proof of Proposition 2, as follows. For $t \in T$ and $d \in D$, we modify the order on the vertices of G_d as follows:

$$(t, d, h) <_v (t, d, h') \iff \begin{cases} h < h' & \text{if (28) is satisfied for } d, \\ h > h' & \text{if (27) is satisfied for } d. \end{cases} \quad (29)$$

That is, we consider the usual ordering on heights of G_d when (28) is satisfied ((i) in Proposition 2), and the reversed order (so the largest height is considered as the minimum element in the new order) when (27) is satisfied. The order of the vertices on the barrier is the same as in Proposition 2.

Mimicking Step 1, we obtain a layered path decomposition as well, as the conditions (ii) and (iii) from Proposition 2 are maintained. Even though the ordering on the vertices might have changed, the flow path decomposition is the same but the order might be reversed: if for dike d , (27) is satisfied instead of condition (i), then the lowermost path in the flow path decomposition with the new ordering of the vertices is the topmost one from Proposition 2.

Mimicking Step 2, we want to avoid the DY-crossings and the same argument goes through, but, importantly, with the updated version of the ordering on the vertices. This updated ordering (29) allows us to claim that undo the crossings does not increase the value of the objective function.

Then mimicking Step 3, we reroute flow from $[U'_1, \dots, U'_{|D|}, U_B]$ to $[L'_1, \dots, L'_{|D|}, L_B]$ or vice-versa, with

$$\begin{cases} L'_d = L_d \text{ and } U'_d = U_d & \text{if (28) is satisfied for the dike } d, \\ L'_d = U_d \text{ and } U'_d = L_d & \text{if (27) is satisfied for the dike } d, \end{cases}$$

where L_d and U_d denote, respectively, the lowermost and the uppermost paths in the argument of Proposition 2 (where the ordinary ordering by height on all the vertices (t, d, \cdot) is considered), and where L'_d and U'_d denote the lowermost and the uppermost flow paths in the new ordering. \square

3.3. Computational results

Recall that the original problem that we wish to solve is the integer program from Section 2. In order to solve it, we first implement and solve the LP relaxation, and within a few minutes we obtain a solution that turns out to be integer. Therefore, with the set of parameter data that we have (D_{cost} , D_{expdam} , etc.), all the variables can be continuous.

In this section we test whether conditions of Corollary 1, which include conditions (iv) found in Corollary 1 and conditions (ii) and (iii) from Proposition 2 are satisfied by the most recent data on flood probabilities, damage and investment costs. Our findings show that most of the tested instances indeed satisfy these conditions.

It is worth noting that we check conditions (iv) from Corollary 1 and conditions (ii) and (iii) from Proposition 2 separately, since, in light of the proofs of Proposition 2 and Corollary 1, each of them provides structural information in the solution set.

The details of our findings are given in Figs. A.9 and A.10. In each table, the first column specifies the years that we used in our study: 5-year periods until the year 2100, and 10-year periods after the year 2100. In the first row we specify each of the dike rings. The description of the dike rings around the IJsselmeer and the IJsseldelta is as follows (the numbers are also used in Fig. 5):

zfw = Zuid-West Friesland = 6–4	nop = Noord-Oost Polder = 7–1
nfl = Noord-Oost Flevoland = 8–1	wfn = West-Friesland Noord = 13–2
wie = Wieringen = 12–1	ijd = IJsseldelta = 11–1
mas = Mastenbroek = 10–1	vol = Vollenhove = 9–1
sal = Salland = 53–1	ovl = Oost-Veluwe = 52–1



Fig. 5. Dike rings around the IJsselmeer and the IJsseldelta in the Netherlands.

3.3.1. Remarks on Fig. A.9

In this table, we summarize how many instances in our real data do satisfy condition (iv) in Corollary 1.

As height levels we included 14 levels (that is $|H_D| = 14$) for the dike rings and also 14 levels for the Afsluitdijk (that is, $|H_B| = 14$). As a result, for each dike ring $14 + 13 + \dots + 1 = \frac{1}{2} \cdot 14 \cdot 15 = 105$ combinations of height, with $h \leq h'$, levels for both the dike rings and the Afsluitdijk could be evaluated. Hence, in total we tested $105 \cdot 105 = 11025$ instances (numbers in row immediately below the names). The rest of numbers in the tables correspond to the number of instances that fulfill the corresponding condition

and are calculated for all dike rings (listed in the first row by its short name) and year (listed in the first column) combinations.

Note that the year 2020 is the initial year considered, thus the conditions of [Corollary 1](#) and [Proposition 2](#) are trivially satisfied for each dike ring (zwf, nop, nfl, wfn, wie, ijd, mas, vol, sal, ovl) and barrier (Afsluitdijk); the “2020” row is hence added for comparison purposes.

3.3.2. Remarks on [Fig. A.10](#)

[Fig. A.10](#) contains the summary of the number of constraints that satisfy conditions (ii) and (iii) in [Proposition 2](#); (ii) for the barrier and (iii) for the dikes. The conditions are analogous for the dikes and the barrier.

There are $\binom{15}{2}$ pairs $h \leq h'$ where each height h has 14 options. However, from all the possible $\binom{15}{2}$ pairs of pairs $((h_1, h_2), (h_3, h_4))$ with $h_1 \leq h_2, h_3 \leq h_4, h_2 \leq h_4$, we shall only consider those where $h_1 \leq h_2 \leq h_3 \leq h_4$; there are $\binom{14+3}{4} = 2380$ of those.

Indeed, if $h_3 < h_2$, then there is no edge $CY(t, d, h_3, h_2)$. Alternatively, the cost of decreasing the dike height d can be thought of being unreasonably high, making (iii) in [Proposition 2](#) obviously true.

We shall observe that in both cases, the number of instances being satisfied does not depend on the year. Indeed, by a usual assumption in water safety research, the cost of updating and maintaining the dike (or the barrier) is independent of the year. That is to say, the only effect on the cost is the rates used to bring the cost to a present day value calculation, which for a fix year, is a constant multiplicative factor on the costs for a given year; this makes the satisfaction of (ii) and (iii) independent of the year.

3.3.3. Implementation

Regarding the practical implementation of the Zwaneveld–Verweij IPmodel, there was the option to install pumps on the Afsluitdijk as an additional strengthening measure or not installing them.¹ As the installation of pumps affects the cost, we ran the simulation twice: once considering the installation of pumps on the Afsluitdijk and once considering it without. It turns out that the minimum cost was achieved by installing the pumps.

We implemented the IP model in GAMS and solved the model with CPLEX. No preprocessing techniques and default GAMS and CPLEX settings were used. The tested problem instances could be solved to optimality within a few minutes.

4. Alternative approaches

A feasible solution to the IP presented in [Section 2](#) can be interpreted as a choice of height $h^d(t)$ for each dike segment at each time period t , and a height $h^B(t)$ of the barrier. Abstractly, the cost of these height series can be written as a sum of cost terms which depend only on the ‘upgrade’ done in period t to segment d (i.e., a heightening of the dike, or merely the maintenance cost); we denote this by $\text{cost}^d(h^d(t-1), h^d(t), t)$ for segment d , and by $\text{cost}^B(h^B(t-1), h^B(t), t)$ for the barrier. Finally, there is also an expected damage cost for upgrading the dike and barrier to heights $h^d(t)$ and $h^B(t)$ in period t , denoted by $\text{dam}^{d,B}(h^B(t), h^d(t), t)$.

¹ Water is dislodged from the IJsselmeer by gravity; when the sea water level is below the IJsselmeer water level, sluices allow the water to flow from the IJsselmeer to the sea. These sluices prevent the water from the sea to enter into the mere when the sea water level is high, such as in high tide. The installation of pumps allows to increase the water flow capacity and to dislodge water from the IJsselmeer even during high tide; these pumps are additional strengthening measures meant to prevent, for instance, flooding in some areas around the IJsselmeer in events of exceptionally high tide. Therefore the installation of pumps affects the costs of the problem.

The Zwaneveld–Verweij IP model can thus be written as follows:

$$\begin{aligned}
 & \min \sum_{t \in T} \left[\text{cost}^B(h^B(t-1), h^B(t), t) \right. \\
 & \quad + \sum_{d \in D} \left[\text{cost}^d(h^d(t-1), h^d(t), t) \right. \\
 & \quad \left. \left. + \text{dam}^{d,B}(h^B(t), h^d(t), t) \right] \right] \tag{30} \\
 & \text{s.t. } h^d(t) \in H_D, h^B(t) \in H_B \text{ for } d \in D, t \in T \\
 & \quad h^d(t) \geq h^d(t-1) \text{ for } d \in D, t \in T \\
 & \quad h^B(t) \geq h^B(t-1) \text{ for } t \in T
 \end{aligned}$$

Next, we provide two algorithms which solve problem (30) to optimality. The first algorithm is a dynamic program, which runs in polynomial time for bounded $|D|$. The second algorithm enumerates all possible height profiles of the barrier, and for each profile solves shortest path problems on small graphs.

4.1. Dynamic programming

There are two key observations to be made. First, the second part of the objective function decomposes naturally into a sum of $|D|$ terms, each of which depends only on the barrier height and one segment. Secondly, for each time period the cost only depends on the dike/barrier heights at times $t-1$ and t . Together this allows us to solve the problem using a dynamic program. The recursion will be on the time period. We maintain a table which stores values $\text{opt}(h^B, \vec{h}^d, t)$ for all $t \in T, h^B \in H_B, \vec{h}^d \in (H_D)^D$. Their interpretation is as that $\text{opt}(h^B, \vec{h}^d, t)$ is equal to the minimum cost made, up to time t , when the barrier and segments are of height h^B and \vec{h}^d at time period t respectively. We can compute the entries of this table by means of the following recursion:

$$\begin{aligned}
 \text{opt}(h^B, \vec{h}^d, t) = \min \left\{ \text{opt}(h^B - i^B, \vec{h}^d - \vec{i}^d, t-1) + \text{cost}^B(h^B - i^B, h^B, t) + \right. \\
 \left. \text{cost}(\vec{h}^d - \vec{i}^d, \vec{h}^d, t) + \text{dam}(h^B, \vec{h}^d, t) : \right. \\
 \left. h^B - i^B \in H_B, \vec{h}^d - \vec{i}^d \in (H_D)^{|D|} \right\}
 \end{aligned}$$

It follows that each entry of the table can be computed in time $\mathcal{O}(|H_B||H_D|^{|D|})$. Hence, all entries of the table can be filled in time $\mathcal{O}((|H_B||H_D|^{|D|})^2 \cdot |T|)$. Using the interpretation of $\text{opt}(h^B, \vec{h}^d, t)$, it follows that the optimum of (30) is equal to

$$\min_{h^B \in H_B, \vec{h}^d \in (H_D)^{|D|}} \text{opt}(h^B, \vec{h}^d, t).$$

This shows the following result:

Theorem 1. *The optimal value of (30) can be found in time $\mathcal{O}((|H_B||H_D|^{|D|})^2 \cdot |T|)$.*

4.2. Shortest paths

We now present an algorithm that runs in polynomial time when the number of possible barrier heights is bounded. The algorithm computes the optimal value of (30) in time

$$\mathcal{O} \left(\overbrace{|D|}^{\# \text{ segments}} \cdot \underbrace{(|T| \cdot |H_D|)^2}_{\text{complexity shortest path}} \cdot \overbrace{|T|^{|H_B|}}^{\# \text{ barrier height profiles}} \right).$$

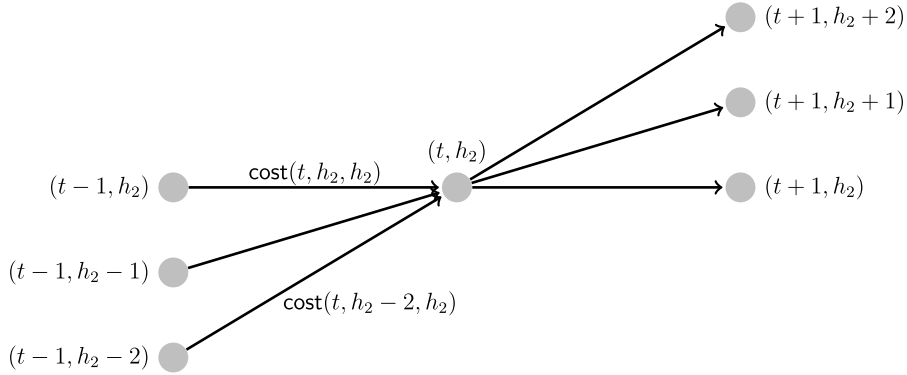


Fig. 6. Incoming and outgoing arcs of a node (t, h_2) are sketched for some $0 < t < T_{\max}$ and $h_2 \in H_D$.

The algorithm is based on shortest paths computations. To illustrate the basic idea we first discuss the algorithm for the setting of one dike segment and no barrier. Thereafter, a barrier is added, and from that the generalization to multiple dike segments and barriers easily follows.

4.2.1. One dike segment, no barrier

First consider the situation with only one dike segment and no barrier. In this case the problem of minimizing the cost at time period t becomes equivalent to finding a shortest p - q path in the following graph. The source $p = (0, 0)$ is the initial height of the dike at time 0. Then, for each time $t \in \{1, \dots, T_{\max}\}$ and each possible height of the dike h , we create a node (t, h) . Finally we define a sink node q . The edges are defined as follows. We first add an edge between $(0, 0)$ and $(1, h)$ for each $h \in H_D$, with weight $\text{cost}(0, h, 1)$, similarly for each $t \in \{1, \dots, T_{\max}\}$ and height pair $h_1 \leq h_2$ there is an edge from $(t-1, h_1)$ to (t, h_2) with weight $\text{cost}(h_1, h_2, t)$ equal to the financial cost associated to the decision of raising the dike segment from height h_1 to h_2 in time period t . Notice that since there is no barrier, we can assume that the expected damage cost $\text{dam}(t, h)$ is incorporated in $\text{cost}(h_1, h_2, t)$. Finally, the nodes (t, h) are all connected to the sink q . In Fig. 6, the incoming and outgoing arcs of a node (t, h_2) are sketched for some $0 < t < T_{\max}$ and $h_2 \in H_D$.

One observes that, indeed, any shortest p - q path corresponds to an optimal strategy of heightening this dike segment.

Recall that a shortest p - q path in a graph $G = (V, E)$ with non-negative edge weights can be found in time $\mathcal{O}(|V|^2)$ using Dijkstra’s algorithm.

4.2.2. One dike segment, a barrier

We now consider the case of a single dike segment and a barrier. The observation we need to make is that the total financial costs incurred by upgrading the dike segment from height h_1 to height h_2 in time period t no longer only depend on the dike segment, they also depend on the height of the barrier at time point t . This means that we cannot solve a shortest path problem for the barrier and dike segment separately: the costs on the dike segment graph depend on the path chosen in the barrier graph.

The key idea is that if we fix the height of the barrier at each time t , then we reduce to the previous setting where all the costs are known. Hence, the optimal value of (30) can be found by minimizing over the possible height profiles $h^B(t)$ of the barrier over time, the minimum cost of a p - q path in the network defined in the previous section (using the costs associated to $h^B(t)$) plus the cost of implementing height profile $h^B(t)$. The outer minimization over the possible height profiles $h^B(t)$ is performed by enumeration, which takes time $\mathcal{O}(|T|^{|H^B|})$. This means that the optimal investment strategy for both the dike segment

and barrier can be found in time

$$\mathcal{O} \left((|T| \cdot |H_D|)^2 \cdot \binom{|T|}{|H_B|} \right) = \mathcal{O} \left((|T| \cdot |H_D|)^2 \cdot |T|^{|H_B|} \right).$$

4.2.3. Multiple dike segments and a barrier

The approach of the previous section easily generalizes to the setting of multiple dike segments and a barrier. Once a height profile $h^B(t)$ of the barrier is fixed, the optimal height profiles of each of the different dike segments can be computed independently. Hence the problem of finding the optimal investment strategy for multiple dike segments and a barrier can be solved in time

$$\mathcal{O} \left(|D| \cdot (|T| \cdot |H_D|)^2 \cdot |T|^{|H_B|} \right).$$

This approach generalizes to the setting of multiple barriers and dike segments (where the costs of a dike segment at time t may depend on the height of several barriers). The complexity will be of the form

$$\mathcal{O} \left(|D| \cdot (|T| \cdot |H_D|)^2 \cdot |T|^{|H_B||B|} \right),$$

where $|B|$ is the number of barriers. One should note that the above approach assumes the same discretization in time of the barrier and dike segments. It seems reasonable to assume a coarser discretization for the barrier of say T_B steps, this would reduce the above-mentioned formula to

$$\mathcal{O} \left(|D| \cdot (|T| \cdot |H_D|)^2 \cdot |T_B|^{|H_B||B|} \right).$$

5. An abstraction of the dike heightening problem

In this section we present a natural abstract version of the dike heightening problem, which allows for several variations and questions, which we believe have not been considered in the literature before. We believe that studying these variations may shed additional light on the complexity of the dike height problem.

In the dike height problem we essentially have two directed graphs where each path in one of the two graphs (the one modeling the height of the barrier) influences the cost of arcs in the other graph. It is not difficult to show that if we were to allow any kind of influence of the path from one of the graphs on the cost of arcs in the other graph, the problem would become NP-hard. Indeed, one can easily show that in this case the problem contains the problem of finding two vertex disjoint paths in a directed graph, which is NP-complete [11].

For this reason, we consider the following restricted problem.

Definition 1. For $k \in \mathbb{N}$, a digraph D with vertex set V and arc set A is k -layered if its vertex set $V(D)$ can be partitioned into layers $V(D) = V_0 \cup V_1 \cup \dots \cup V_k \cup V_{k+1}$ such that each $a \in A$ is from V_i to V_{i+1} for some $i = 0, \dots, k$, where V_0 and V_{k+1} both consist of a single vertex and $|V_1| = |V_2| = \dots = |V_k|$. We denote the arcs between V_i and V_{i+1} by $A[V_i, V_{i+1}]$ and refer to $|V_1|$ as the partition size.

With this in mind, we define the MINIMUM INTERTWINED COST PATH problem as follows. The problem takes as input k -layered graphs $G^1 = (V^1, A^1), \dots, G^{d+1} = (V^{d+1}, A^{d+1})$ with partitions $V^j = V_0^{(j)} \cup V_1^{(j)} \cup \dots \cup V_{k+1}^{(j)}$, $|V_0^{(j)}| = |V_{k+1}^{(j)}| = 1$, and cost functions $c^j : A^j \rightarrow \mathbb{R}_{\geq 0}$ for $j = 1, \dots, d + 1$, and for each $i = 1, \dots, k$ and $t = 2, \dots, d + 1$ maps $m_i^t : V_i^{(t)} \times A^1[V_{i-1}^1, V_i^1] \rightarrow \mathbb{R}_{\geq 0}$.

For $j = 1, \dots, d + 1$ let $P^j = (v_0^j, a_1^j, v_1^j, a_2^j, v_2^j, \dots, a_k^j, v_k^j, a_{k+1}^j, v_{k+1}^j)$ be a path from $V_0^{(j)}$ to $V_{k+1}^{(j)}$ with $a_i^j = (v_{i-1}^j, v_i^j)$ for $j = 1, \dots, d + 1$. Define the cost of the $(d + 1)$ -tuple (P^1, \dots, P^{d+1}) as

$$\text{cost}(P^1, \dots, P^{d+1}) = \sum_{i=1}^{k+1} \sum_{t=1}^{d+1} c^t(a_i^t) + \sum_{i=1}^{k+1} \sum_{t=2}^{d+1} m_i^t(v_i^t, a_i^1).$$

The objective is to compute a $(d + 1)$ -tuple of paths $(P^{1*}, \dots, P^{d+1*})$ with minimum cost over all such $(d + 1)$ -tuples.

In the MINIMUM INTERTWINED COST PATH problem, the dependence of $\text{cost}(P^1, \dots, P^{d+1})$ on the paths P^2, \dots, P^{d+1} is linear in the edges of P^2, \dots, P^{d+1} . Note that the Zwaneveld–Verweij IP model is the specific case of the MINIMUM INTERTWINED COST PATH problem where the barrier acts as P^1 , each of the dikes is represented by one path P^j , $j = 2, \dots, d + 1$, and the cost functions m_i^t only depend on the vertices $m_i^t(v_i^t, a_i^1) = m_i^t(v_i^t, v_i^1)$, in addition to the edges between V_i and V_{i+1} being restricted (only connecting vertices of non-decreasing heights).

This particular fact allowed us in Section 4.2.2 to give an algorithm for the problem, which runs in polynomial time if we consider the size of the sets in the partition of the vertices of G^1 as a constant. If the bipartite graphs between $V_i^{(1)}$ and $V_{i+1}^{(1)}$ are complete, the cost factor $|T|^{|H_B|}$ turns into $|H_B|^{|T|}$ and then this dynamic programming approach will not work. It would be interesting to find out if some other approach may yield an efficient algorithm.

We end this section with some concrete questions.

Question 2. *Is the MINIMUM INTERTWINED COST PATH problem NP-hard if the number of possible heights is unbounded?*

We do not have an answer for Question 2, but we remark the following: with an appropriate cost function on the updating of the heights of one dike, instances of the KNAPSACK problem can be seen as optimizing the height of one dike. Indeed, the decision of updating the height of a dike at time $t \in \mathbb{N}$ corresponds to the decision of adding a certain number of copies of an item to the knapsack; the total height of the dike at time t corresponds to the accumulated weight of the chosen items (counting multiplicities) to be carried among the first t items. The cost function of upgrading the height at time t by kw_t units corresponds to the profit of adding k copies of the item t , whose weight is w_t . The cost function of upgrading the dikes is such that once the capacity of the knapsack is exceeded by a set of items, then the cost of keeping or upgrading the dike height is unreasonable high. With this correspondence, we observe that the optimal solution of the KNAPSACK problem corresponds to the optimal solution of the dike height. Computing an optimal solution to the KNAPSACK problem is well-known to be NP-hard. One of the inputs of the KNAPSACK problem is the logarithm of the total weight of the knapsack bag. Thus the dynamic program proposed in Section 4.1 is an exponential time algorithm. Note that Question 2 asks whether MINIMUM INTERTWINED COST PATH is NP-hard with respect to the input “the total weight of the knapsack bag”, instead of its logarithm.

If Question 2 has a positive answer, the problem becomes hard to solve in general, and hence it makes sense to consider the following more restricted configurations.

Question 3. *Under which conditions on the bipartite graphs $G^j[V_i^{(j)}, V_{i+1}^{(j)}]$, ($j = 1, 2, i = 1, \dots, k$) can the MINIMUM INTERTWINED COST PATH problem be solved in polynomial time?*

Question 4. *Suppose the partition size of G^1 is constant. Under which conditions on the bipartite graphs $G^j[V_i^{(j)}, V_{i+1}^{(j)}]$ ($j = 1, 2, i = 0, \dots, k$) is the MINIMUM INTERTWINED COST PATH problem polynomial-time solvable?*

6. Conclusions

In this work we considered the Zwaneveld–Verweij IP model, detailed in Section 2, for cost–benefit analysis and flood protection through dike building in the Netherlands. The contributions of this paper are as follows:

$CY(t, h_1, h_2)$				$B(t, h_1^B, h_2^B)$				$DY(t, h_2, h_2^B)$				Eq.
1,0,1	1,0,0	2,1,1	2,0,0	1,0,1	1,0,1	2,1,1	2,0,0	1,1,0	1,0,1	2,1,1	2,0,0	
1	1	0	0									(5)
1	0	-1	0									(5)
0	1	0	-1									(5)
				1	1	0	0					(8)
				1	0	-1	0					(8)
				0	1	0	-1					(8)
1	0	0	0					-1	0	0	0	(6)
0	1	0	0					0	-1	0	0	(6)
0	0	1	0					0	0	-1	0	(6)
0	0	0	1					0	0	0	-1	(6)
				1	0	0	0	0	-1	0	0	(9)
				0	1	0	0	-1	0	0	0	(9)
				0	0	1	0	0	0	-1	0	(9)
				0	0	0	1	0	0	0	-1	(9)

Fig. A.7. Relevant submatrix of inequalities satisfied with equality at p .

				1	1	0	0
				1	0	-1	0
				0	1	0	-1
				1	1	0	0
				0	1	-1	0
				1	0	0	-1
1	0	0	0	-1	0	0	0
0	1	0	0	0	-1	0	0
0	0	1	0	0	0	-1	0
0	0	0	1	0	0	0	-1
				1	0	0	0
				0	1	0	0
				0	0	1	0
				0	0	0	1

Fig. A.8. Row reduced matrix from Fig. A.7.

- we show that despite what is suggested by experiments, the polytope related to the LP relaxation of the integer problem is not integral;
- we formulate conditions on costs in this integer program for the polytope to be integral and check that such conditions hold on numerous data related to the case the authors consider;
- the above condition satisfaction seems to explain why just with the LP relaxation we already obtain an integer solution to the original IP from Section 2;
- in the end, we provide a theoretical result on the polynomial-time solution to the related IP model when some parameters (number of dike segments, or the number of feasible barrier heights) are constant, and provide a complexity-theoretic abstraction of the considered model.

To sum up, we resolve issues related to solving the above mentioned model efficiently. Roughly speaking, the model consists of many “smaller models” corresponding to heightening of each dike segment, which are joined together into a big model by the barrier (which can be also heightened and which interacts with the dike segments). This joining together is the source of the model’s difficulty and is characteristic for flood protection problems which often involve “small measures” affecting a single area, and “big measures”

year	zwf	nop	nfl	wfn	wie	ijd	mas	vol	sal	ovl
2020	11025 (00)	11025 (00)	11025 (00)	11025 (00)	11025 (00)	11025 (00)	11025 (00)	11025 (00)	11025 (100)	11025 (00)
2021	10930 (99)	11025 (100)	9484 (86)	10935 (99)	11025 (100)	10604 (96)	10105 (91)	10484 (95)	10634 (96)	10869 (99)
2026	10815 (98)	11022 (100)	9494 (86)	10909 (99)	11025 (100)	10602 (96)	10117 (92)	10488 (95)	10639 (96)	10869 (99)
2031	10750 (98)	11023 (100)	9510 (86)	10874 (99)	11025 (100)	10600 (96)	10125 (92)	10491 (95)	10645 (97)	10856 (98)
2036	10936 (99)	11025 (100)	9530 (86)	10857 (98)	11025 (100)	10598 (96)	10130 (92)	10496 (95)	10656 (97)	11024 (100)
2041	10970 (100)	11025 (100)	9553 (87)	10832 (98)	11025 (100)	10596 (96)	10138 (92)	10499 (95)	10670 (97)	10996 (100)
2046	10992 (99)	11025 (100)	9588 (87)	10821 (98)	11025 (100)	10594 (96)	10148 (92)	10503 (95)	10683 (97)	10996 (100)
2051	11000 (100)	11025 (100)	9618 (87)	10811 (98)	11025 (100)	10594 (96)	10148 (92)	10506 (95)	10683 (97)	10963 (99)
2056	11010 (100)	11025 (100)	9666 (88)	10807 (98)	11025 (100)	10594 (96)	10148 (92)	10511 (95)	10683 (97)	10875 (99)
2061	11017 (100)	11025 (100)	9674 (88)	10807 (98)	11025 (100)	10594 (96)	10156 (92)	10515 (95)	10683 (97)	10809 (98)
2066	11021 (100)	11025 (100)	9677 (88)	10807 (98)	11025 (100)	10594 (96)	10170 (92)	10517 (95)	10683 (97)	10736 (97)
2071	11024 (100)	11025 (100)	9698 (88)	10803 (98)	11025 (100)	10594 (96)	10184 (92)	10520 (95)	10694 (97)	10676 (97)
2076	11025 (100)	11025 (100)	9715 (88)	10799 (98)	11025 (100)	10594 (96)	10194 (92)	10526 (95)	10708 (97)	10619 (96)
2081	11025 (100)	11025 (100)	9762 (89)	10803 (98)	11025 (100)	10594 (96)	10198 (92)	10530 (96)	10709 (97)	10566 (96)
2086	11025 (100)	11025 (100)	9777 (89)	10803 (98)	11025 (100)	10600 (96)	10205 (93)	10533 (96)	10713 (97)	10510 (95)
2091	11025 (100)	11025 (100)	9812 (89)	10803 (98)	11025 (100)	10600 (96)	10222 (93)	10537 (96)	10721 (97)	10487 (95)
2096	11025 (100)	11025 (100)	9847 (89)	10807 (98)	11025 (100)	10600 (96)	10239 (93)	10542 (96)	10736 (97)	10383 (94)
2101	11025 (100)	11025 (100)	9895 (90)	10807 (98)	11025 (100)	10600 (96)	10266 (93)	10542 (96)	10751 (98)	8185 (74)
2111	11025 (100)	11025 (100)	9927 (90)	10809 (98)	11025 (100)	10600 (96)	10297 (93)	10531 (96)	10763 (98)	8051 (73)
2121	11025 (100)	11025 (100)	9993 (91)	10894 (99)	11025 (100)	10601 (96)	10297 (93)	10506 (95)	10771 (98)	7986 (72)
2131	11025 (100)	11025 (100)	10020 (91)	10954 (99)	10934 (99)	10606 (96)	10297 (93)	10498 (95)	10771 (98)	7937 (72)
2141	10975 (100)	11025 (100)	10098 (92)	10988 (100)	10752 (98)	10612 (96)	10304 (93)	10472 (95)	10776 (98)	7920 (72)
2151	10854 (98)	11025 (100)	10197 (92)	10987 (100)	10492 (95)	10633 (96)	10311 (94)	10470 (95)	10799 (98)	7876 (71)
2161	10636 (96)	11025 (100)	10236 (93)	10953 (99)	10206 (93)	10633 (96)	10316 (94)	10439 (95)	10825 (98)	7858 (71)
2171	10207 (93)	11025 (100)	10289 (93)	10912 (99)	9751 (88)	10654 (97)	10331 (94)	10388 (94)	10831 (98)	7842 (71)
2181	9608 (87)	11025 (100)	10365 (94)	10783 (98)	9205 (83)	10696 (97)	10369 (94)	10273 (93)	10847 (98)	7796 (71)
2191	9257 (84)	11025 (100)	10447 (95)	10436 (95)	9114 (83)	10747 (97)	10423 (95)	10120 (92)	10857 (98)	7755 (70)
2201	9153 (83)	11025 (100)	10456 (95)	9881 (90)	9114 (83)	10779 (98)	10492 (95)	9919 (90)	10874 (99)	7727 (70)
2211	8533 (77)	11025 (100)	10473 (95)	9299 (84)	8477 (77)	10815 (98)	10603 (96)	9745 (88)	10876 (99)	7607 (69)
2221	8359 (76)	11025 (100)	10520 (95)	9050 (82)	8477 (77)	10853 (98)	10681 (97)	9519 (86)	10874 (99)	7419 (67)
2231	7936 (72)	11025 (100)	10607 (96)	8802 (80)	8477 (77)	10868 (99)	10730 (97)	9261 (84)	10823 (98)	7384 (67)
2241	7517 (68)	11025 (100)	10640 (97)	8459 (77)	7840 (71)	10901 (99)	10773 (98)	7748 (70)	6573 (60)	7352 (67)
2251	7067 (64)	11025 (100)	10708 (97)	8251 (75)	7749 (70)	10921 (99)	10815 (98)	4884 (44)	5581 (51)	7339 (67)
2261	6579 (60)	11025 (100)	10745 (97)	7849 (71)	7293 (66)	10933 (97)	10846 (98)	3719 (34)	5222 (47)	7270 (66)
2271	5994 (54)	11025 (100)	10864 (99)	7501 (68)	6930 (63)	10954 (99)	10881 (99)	3403 (31)	5052 (46)	7276 (66)
2281	5475 (50)	11025 (100)	10935 (99)	7096 (64)	6930 (63)	10954 (99)	10923 (99)	3245 (29)	5021 (46)	7276 (66)
2291	4786 (43)	11025 (100)	10975 (100)	6653 (60)	6111 (55)	10973 (100)	10925 (100)	3212 (29)	5026 (46)	7253 (66)

Fig. A.9. Output of the numerical experiment: checking whether conditions (iv) from Corollary 1 hold. In parenthesis, the percentage.

year	zwf	nop	nfl	wfn	wie	ijd	mas	vol	sal	ovl	afsluitdijk
2020	2380 (100)	2380 (100)	2380 (100)	2380 (100)	2380 (100)	2380 (100)	2380 (100)	2380 (100)	2380 (100)	2380 (100)	2380 (100)
2021	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2026	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2031	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2036	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2041	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2046	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2051	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2056	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2061	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2066	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2071	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2076	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2081	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2086	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2091	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2096	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2101	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2111	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2121	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2131	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2141	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2151	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2161	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2171	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2181	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2191	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2201	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2211	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2221	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2231	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2241	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2251	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2261	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2271	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2281	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)
2291	2162 (91)	2188 (92)	2016 (85)	2089 (88)	2104 (88)	2018 (85)	2217 (93)	2180 (92)	2210 (93)	2085 (88)	2016 (85)

Fig. A.10. Output of the numerical experiment: checking whether conditions (iii) from Proposition 2 hold for the dikes and conditions (ii) from Proposition 2 for the barrier (afsluitdijk).

affecting multiple areas. On that angle, our complexity results demonstrate that, essentially, the per-dike problem is simple, but linking the dikes makes the problem difficult.

Acknowledgments

We thank Kees Roos for helpful discussions in an early stage of this work, André Woning from Rijkswaterstaat for useful background information and Gary Greaves for reading the manuscript.

Parts of our results were obtained in the 2017 *Study Group Mathematics with Industry*, for which we thank the organizers. M. Mnich is supported by a DFG, Germany grant MN 59/4-1. L. Vena is supported by the European Research Council under the European Unions Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement 339109, and by the project 18-13685Y of the Czech Science Foundation (GAČR). We thank two anonymous referees whose comments and suggestions have enhanced the accessibility of this article and made the proofs easier to follow.

Appendix. Figures and tables

See Figs. A.7–A.10.

References

- [1] P.J. Zwaneveld, G. Verweij, S. van Hoesel, Safe dike heights at minimal costs: an integer programming approach, *European J. Oper. Res.* 270 (1) (2018) 294–301.
- [2] P.J. Zwaneveld, G. Verweij, *Economic Decision Problems in Multi-Level Flood Prevention: A New Graph-Based Approach Used for Real World Applications*, CPB Discussion Paper 380, 2018, The Hague, The Netherlands.
- [3] R. Brekelmans, D. den Hertog, K. Roos, C. Eijgenraam, Safe dike heights at minimal costs: the nonhomogeneous case, *Oper. Res.* 60 (6) (2012) 1342–1355.
- [4] D. van Dantzig, Economic decision problems for flood prevention, *Econometrica* 24 (3) (1956) 276–287.
- [5] F. Bos, P.J. Zwaneveld, Een snelle kosten-effectiviteitsanalyse voor het deltaprogramma ijsselmeergebied, in: CPB Communication (Including a CPB Background Document), 2012, The Hague, The Netherlands, <https://www.cpb.nl/persbericht/3212337/ondanks-zeespiegelstijging-hoeft-waterpeil-ijsselmeer-niet-omhoog-komende-eeuw>.
- [6] P.J. Zwaneveld, G. Verweij, Economisch optimale waterveiligheid in het ijsselmeergebied. mkba waterveiligheid: afsluitdijk, houtribdijk, ijsselmeer, ijssel- en vechtdelta en markermeer, in: CPB Communication, 2014, The Hague, The Netherlands, <https://www.cpb.nl/publicatie/economisch-optimale-waterveiligheid-het-ijsselmeergebied>.
- [7] J. Kind, *Maatschappelijke Kosten-Batenanalyse Waterveiligheid 21e Eeuw (MKBA WV21)*, Deltares Report, 2011, Delft, The Netherlands.
- [8] C. Eijgenraam, R. Brekelmans, D. den Hertog, K. Roos, Optimal strategies for flood prevention, *Manage. Sci.* 63 (5) (2015) 1644–1656.
- [9] K. Postek, D. den Hertog, J. Kind, C. Pustjens, Adjustable robust strategies for flood protection, *Omega* 82 (2019) 142–154.
- [10] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*, vol. 24, Springer Science & Business Media, 2003.
- [11] S. Fortune, J. Hopcroft, J. Wyllie, The directed subgraph homeomorphism problem, *Theoret. Comput. Sci.* 10 (1980) 111–121.