



## Research paper

# A software tool for both the maximum tolerated dose and the optimal biological dose finding trials in early phase designs

Chen Li <sup>a,1</sup>, Hongying Sun <sup>b,1</sup>, Cheng Cheng <sup>c</sup>, Li Tang <sup>c,\*</sup>, Haitao Pan <sup>c,\*</sup>

<sup>a</sup> Department of Health Statistics, School of Preventive Medicine, Fourth Military Medical University, China

<sup>b</sup> RNA Computational Scientist, Takeda Pharmaceuticals, United States of America

<sup>c</sup> Department of Biostatistics, St. Jude Children's Research Hospital, United States of America

## ARTICLE INFO

## Keywords:

Maximum tolerated dose  
Optimal biological dose  
Dose-finding  
Phase I/II trials  
Bayesian adaptive design

## ABSTRACT

**Background:** Phase I and/or I/II oncology trials are conducted to find the maximum tolerated dose (MTD) and/or optimal biological dose (OBD) of a new drug or treatment. In these trials, for cytotoxic agents, the primary aim of the single-agent or drug-combination is to find the MTD with a certain target toxicity rate, while for the cytostatic agents, a more appropriate target is the OBD, which is often defined by considering of toxicity and efficacy simultaneously. Accessible software packages to achieve both these aims are needed.

**Results:** The objective of this study is to develop a software package that can provide tools for both MTD- and OBD-finding trials, which implements the Keyboard design for single-agent MTD-finding trials as reported by Yan et al. (2017), the Keyboard design for drug-combination MTD-finding trials by Pan et al. (2020), and a phase I/II OBD-finding method by Li et al. (2017) in a single R package, called Keyboard. For each of the designs, the Keyboard package provides corresponding functions such as `get.boundary(...)` for deriving the optimal dose escalation and de-escalation boundaries, `select.mtd(...)` for selecting the MTD when the trial is completed, `select.obd(...)` for selecting the OBD at the end of a trial, and `get.oc(...)` for generating the operating characteristics.

**Conclusion:** The Keyboard R package developed herein provides convenient tools for designing, conducting and analyzing single-agent, drug-combination, phase I/II OBD-finding trials.

## 1. Introduction

Phase I dose-finding clinical trials are critical in new drug/treatment development because they determine the dose that will be further investigated in the subsequent phase II or III trials. For the *cytotoxic* agent, one of the primary objectives is to find the maximum tolerated dose (MTD), which is defined as the highest dose that has a dose-limiting toxicity (DLT) rate less than or close to a prespecified target rate. The identified MTD will then be examined in later phases, e.g., phase II clinical trials. Statistical methods for the single-agent MTD-finding designs include algorithm-based designs, such as the 3 + 3 design [1], and the biased-coin design [2–4], the model-based designs, such as the continual reassessment method (CRM) [5,6], and model-assisted designs, such as mTPI [7], BOIN [8], and Keyboard [9] designs.

It should be noted that all the above methods were developed for the *cytotoxic* agent to identify the MTD. Cytotoxic drug development, however, rests on the premise that agents must be cytotoxic to be effective. By equating efficacy with toxicity, the traditional drug development, and progressing from phase I through phase III clinical studies,

has allowed us to find the highest effective dose that does not induce intolerable levels of toxicity. This essentially justifies the goal of finding the MTD for phase I dose-finding trials. This assumption, while reasonable for cytotoxic agents, however, may not hold true for *cytostatic* or *molecularly targeted* agents. For example, the chimeric antigen receptor (CAR) T therapies require a balance of a boosting of the immune system to combat cancer while avoiding over-stimulation. In this case, preliminary dose exploration should aim to capture effective biological activity rather than DLT alone. Therefore, the optimal biological dose (OBD), which usually refers to a safe dose with acceptable efficacy is more appropriate than the MTD since it considers both the toxicity and efficacy. In the literature, the associated methods for finding OBD are termed as phase I/II dose-finding designs. These include many designs, for example, the model-based methods [10–14] and model-assisted methods [15–20]. In addition to these designs, Li et al. [15] proposed a toxicity and efficacy probability interval (TEPI) design, which was shown to have desirable operating characteristics and was simple and

\* Corresponding author.

E-mail addresses: [li.tang@stjude.org](mailto:li.tang@stjude.org) (L. Tang), [haitao.pan@stjude.org](mailto:haitao.pan@stjude.org) (H. Pan).

<sup>1</sup> Chen Li and Hongying Sun contribute equally.

transparent to practitioners with a physician-elicited decision table that maps the two-dimensional probability intervals to a set of dosing decisions. The Keyboard package develops multiple R functions to implement this design since the method behind this design is also interval based which is the core of the Keyboard design. Noted that recently another method uTPI has been proposed [20], which uses approaches similar to that for the TEPI design [15]. Since the uTPI has provided the R code in their paper, we are not incorporating this method into the Keyboard package.

In modern clinical trials, it is common to treat patients with a combination of agents. This is especially true in case of cancer patients, as they are more effective and less susceptible to drug resistance than those undergoing single-agent trials. Trial designs for drug-combination studies involve several distinct features that are beyond the scope of the methods for single-agent studies. A major challenge in designing combination trials is that dose combinations are only partially ordered in terms of toxicity probability. For instance, consider a trial combining  $m$  doses of agent A and  $n$  doses of agent B, that is, we now have an  $n \times m$  dose matrix. A major challenge in designing combination trials is that dose combinations are only partially ordered in terms of toxicity probability. That is, *a priori*, we cannot fully rank the  $n \times m$  dose matrix from low to high based on their toxicity probabilities. Various designs have been proposed for the drug-combination MTD-finding trials: a design based on the order of the restricted inference [4], a copula-type regression model [19], latent contingency tables [20], the partial order CRM method (POCRM) [21], the sequential dose-finding strategy [22,23], a Bayesian optimal interval design [24], the Keyboard combination design [25], and Bayesian data augmentation for late-onset toxicity [26], among others. The Keyboard package implements the Keyboard combination design proposed by Pan et al. [25].

In summary, we have developed an R package, Keyboard, that includes three methods [9,15,25] for practitioners and these methods can be easily accessed. All these methods use the Bayesian interval-based method, which is the underlying theoretical basis for the Keyboard design. Therefore, we will incorporate these three methods in one package.

The paper is organized as follows. We will first concisely introduce the three designs, Keyboard single-agent method [9], Keyboard combination design [25], and Keyboard phase I/II design [15] in Section 2. Section 3 gives three exemplary trials and demonstrates how to use the Keyboard package to design trials for single-agent, drug combinations of MTD- or OBD-dose finding. Section 4 gives the conclusion. The appendix provides detailed step-by-step implementations of these methods using the Keyboard R package.

## 2. Material and methods

### 2.1. Design for the MTD dose-finding single-agent trials

The design of this package for the single-agent MTD-finding was proposed by Yan et al. [9], and its theoretical properties were further explored by Pan et al. [25]. We have termed this as the single-agent Keyboard design, or simply, the Keyboard design. This design uses the toxicity probability's posterior distribution to guide dose transitions. To decide whether to escalate or de-escalate the dose, by using the Keyboard design, we first need to specify a *target key*, which is an *equivalence interval* also used in the mTPI design, and then identify the *strongest key*, which is defined as the interval with the maximum posterior probability in terms of the toxicity rate, based on the current information. If the strongest key is to the left of the target key, we escalate the dose because data suggest that current dose is likely to be low. If the strongest key is to the right of the target key, then we de-escalate the dose because the observed data suggest that the current dose is likely to be over-toxic, and if the strongest key is the target key, then we stay at the current dose because the observed data support the

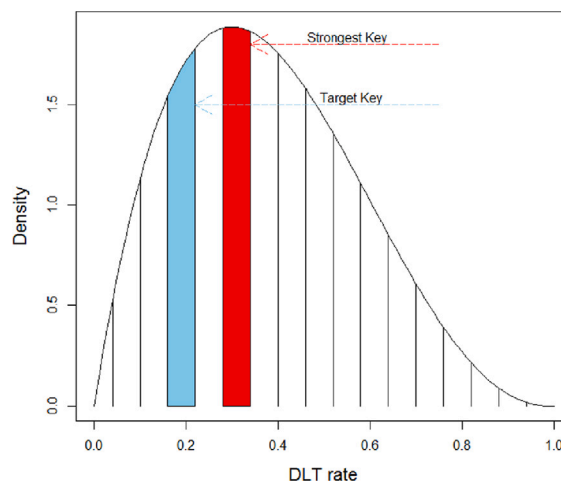


Fig. 1. Dose transition schema of the Keyboard design.

notion that the current dose is likely to be in the right dose interval (See Fig. 1).

Let  $\phi$  be the target toxicity rate specified by the investigator and  $p_i \in (0, 1)$  denote the toxicity probability of dose level  $i \in \{1, \dots, D\}$ .  $n_i$  refers to the number of patients treated at dose  $i$ , and  $y_i$  is the number of patients with DLT among the  $n_d$  treated patients. In this case, data can be denoted as  $D_i = (n_i, y_i)$ . The beta-binomial model is used in the Keyboard design:  $y_i | p_i \sim \text{Binomial}(p_i, n_i)$ ,  $p_i \sim \text{Uniform}(0, 1)$ . Given  $D_i$ , posterior distribution of the toxicity rate at dose  $i$  is  $p_i | D_i \sim \text{Beta}(y_i + 1, n_i - y_i + 1)$ . Here we denote the target key as  $I_{\text{target}} = (\phi - \epsilon_1, \phi + \epsilon_2)$ , with  $\epsilon_1$  and  $\epsilon_2$  being small positive values, for instance, both are 0.05. Then, a unit probability interval of  $[0, 1]$  can be partitioned into a series of equally-wide keys/intervals, denoted by  $I_k$ . To decide whether to escalate or de-escalate the dose, the Keyboard design calculates the strongest key  $I_{\text{max}}$ :

$$I_{\text{max}} = \underset{k \in \{1, \dots, K\}}{\text{argmax}} \{ \Pr(p_i \in I_k | D_i) \}.$$

Dose-transition rules for the Keyboard design are as follows:

- If  $I_{\text{max}} > I_{\text{target}}$ ,  $p_i$  is most likely to be overdosing, de-escalation to a lower dose will be applied.
- If  $I_{\text{max}} < I_{\text{target}}$ ,  $p_i$  is most likely to be underdosing, escalation to a higher dose will be applied.
- If  $I_{\text{max}} \equiv I_{\text{target}}$ ,  $p_i$  is most likely to be the proper dose, stay at the current dose will be applied.

A desirable feature of the Keyboard design is that the dose-transition rules can be provided before the start of the trial and no real-time computation is required during the trial. Practitioners only need the total number of patients and number of patients experiencing DLT at various dose levels to make the dose-transition rules. Table 1 below presents an example.

For the safety of patients, in practice, the following *safety rules* are applied to the Keyboard design:

- (i) If  $n_i \geq 3$  and  $\Pr(p_i > \phi | D_i) > 95\%$ , we eliminate the current dose  $i$  and any higher doses from the trial to avoid exposing future patients to unacceptably toxic doses.
- (ii) If current dose  $i$  is the lowest dose in (i), the trial is terminated and no dose is selected as the MTD.

### 2.2. Keyboard design for drug-combination phase I dose-finding trials

Recent advances in drug discovery have intensified the interest in using dual agents in phase I clinical trials. The design and conduct of

**Table 1**  
Pre-tabulated decision rules of the Keyboard design for single-agent.

	Number of patients treated at the current dose															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\phi = 0.2$ with the target key = (0.17, 0.23)																
Escalate if number of DLTs $\leq$	0	0	0	0	0	1	1	1	1	1	1	2	2	2	2	2
De-escalate if number of DLTs $\geq$	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4	4
$\phi = 0.3$ with the target key=(0.25, 0.35)																
Escalate if number of DLTs $\leq$	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3
De-escalate if number of DLTs $\geq$	1	1	2	2	2	3	3	3	4	4	4	5	5	5	6	6

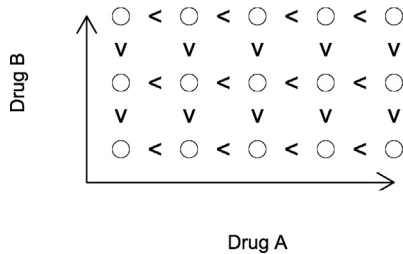


Fig. 2. Two dimensional drug-combination trial.

the phase 1 combination trial present specific challenges. A fundamental assumption for cytotoxic/biologic agents is monotonicity between toxicity and doses, For single agents, this assumption induces a complete ordering of the doses. However, in the case of drug combination treatment where the two agents are allowed to vary, it induces a partial ordering constraint on the probabilities of toxicities (see Fig. 2). The monotonicity assumption coupled with the small sample size in phase I clinical trials and higher dimension of the dose space, makes the design of combination trials challenging [25,27].

Herein, we list some of the proposed combination designs in literature [12,20,25,27]. Among these methods, the Keyboard combination design proposed by Pan et al. [25] is one method that has empirically and theoretically demonstrated to have desirable properties.

Let  $p_{jk}$  be the toxicity probability of the dual agents agent A at level  $j$  and agent B at level  $k$ ,  $n_{jk}$  be the number of patients treated at the dose combination  $(j, k)$  and  $y_{jk}$  be the number of patients who experienced DLT at the dose combination  $(j, k)$ .

The algorithm of the Keyboard combination design is given as below:

- Step 1. Treat the first cohort of patients at the dose combination (1, 1) or a specified dose combination.
- Step 2. At the dose combination  $(j, k)$ , given the observed data  $D_{jk} = (n_{jk}, y_{jk})$ , find the strongest key  $I_{\max}$  based on the updated posterior distribution of  $p_{jk}$ :
  - If  $I_{\max} < I_{\text{target}}$ , then we escalate the dose.
  - If  $I_{\max} > I_{\text{target}}$ , then we deescalate the dose.
  - Otherwise, if  $I_{\max} \equiv I_{\text{target}}$ , then we stay at the current dose.

Step 3. Continue this process until the pre-specified maximum sample size  $N$  is achieved.

Note: Here,  $I_1 < (>)I_2$  means that  $I_1$  is at the left (right) of the  $I_2$ , and  $\equiv$  means that the two intervals overlap.

However, in the above algorithm, it is not clear how to decide to escalate/de-escalate, since for combination trials with dual-agents, if the decision is, for example, to escalate, there exist more than one directions. For example, we can escalate only agent A, only agent B, or both simultaneously.

The Keyboard design uses the admissible dose escalation/de-escalation sets to derive the dose-transition path explicitly. The

admissible sets are defined as:

$$\mathcal{A}_E = \{(j + 1, k), (j, k + 1)\}$$

and

$$\mathcal{A}_D = \{(j - 1, k), (j, k - 1)\}$$

For instance,  $\mathcal{A}_E$  means that escalation can be only allowed to move agent A from  $j$  to  $j+1$ , or move agent B from  $k$  to  $k+1$ . That is, diagonal movement is prohibited in escalation/de-escalation in  $\mathcal{A}_{E_1}$  if there are safety concerns. The admissible set  $\mathcal{A}_D$  can be interpreted similarly.

The dose-transition algorithm is then given as follows:

- Escalation: Escalate to the dose combination that belongs to  $\mathcal{A}_E$  and has the highest value of  $\Pr(p_{j'k'} \in I_{\text{target}} | D_{jk})$  where  $(j', k') \in \mathcal{A}_E$ .
- De-escalation: De-escalate to the dose combination that belongs to  $\mathcal{A}_D$  and has the highest value of  $\Pr(p_{j'k'} \in I_{\text{target}} | D_{jk})$  where  $(j', k') \in \mathcal{A}_D$ .

If there are multiple optimal dose combinations with the same value of  $\Pr(p_{j'k'} \in I_{\text{target}} | D_{jk})$ , we will randomly choose one. The trial is completed when the maximum sample size is reached. After seeing all observed data at the completion of the trial, the matrix isotonic regression will be used to obtain the estimates of  $p_{jk}$ 's value and select the MTD as the combination with a estimated toxicity probability that is closest to the target. The same set of safety rules for the single-agent Keyboard design are also used here.

### 2.3. Design for the single-agent OBD-finding trials

Traditionally, the purpose of having a phase I dose-finding design in cancer has been to find the MTD based solely on the toxicity and traditional designs considering the DLT data implicitly assumes a monotonically increasing relationship between dose and response efficacy. Monotone efficacy may be a reasonable assumption for cytotoxic agents. However, for molecular targeted agents, little toxicity may arise within the therapeutic dose range and the dose-response curves may not be strictly monotone. This challenges the conventional principle of *more is better*. Instead, the OBD, which is defined as the lowest dose with the highest rate of efficacy while safe, is a more appropriate endpoint. A study by Corbaux et al. [28] showed that the dose approved by the Food and Drug Administration (FDA) is consistent with an OBD of 83% of drugs in a total of 87 completed trials for evaluating molecular targeted agents.

Li et al. [15] proposed a TEPI phase I design for finding the OBD. This approach incorporates efficacy outcomes to inform dosing decisions to optimize efficacy and safety simultaneously, with the purpose of finding the dose with the most desirable outcome for safety and efficacy. The key features of TEPI are its simplicity, flexibility, and transparency, because all decision rules can be prespecified prior to trial initiation.

Assume that there are  $D$  doses in a trial. We still assume that the toxicity probability  $p_i$  increases with dose level  $i$ . However, the efficacy probability  $q_i$  may increase initially and then reach a plateau from which minimal improvement or even decreasing efficacy may be seen with increasing dose. For this reason,  $q_i$  is assumed to be not monotone

**Table 2**  
An example of a pre-specified decision table for the OBD-finding design.

		Efficacy.low (0,0.25)	Efficacy.moderate (0.25,0.45)	Efficacy.high (0.45,0.65)	Efficacy.superb (0.65,1)
Toxicity.low	(0,0.15)	E	E	E	E
Toxicity.moderate	(0.15,0.25)	E	E	E	S
Toxicity.high	(0.25,0.35)	D	S	S	S
Toxicity.unacceptable	(0.35,1.0)	D	D	D	D

with  $i$ . In the TEPI design,  $p_i$  and  $q_i$  are assumed to be independent for simplicity. Suppose that the current dose is  $i$ , number of patients treated at this dose level is  $n_i$ , number of patients who experienced toxicity is  $x_i$ , and the number of responses is  $y_i$ . The trial data can be represented as  $\text{Data} = (n_i, x_i, y_i), i = 1, \dots, D$ .

Similar to Keyboard design in Section 2.1, in TEPI design, we partition the unit intervals (0, 1) for  $p_i$  and  $q_i$  into subintervals by using  $(a, b)$  and  $(c, d)$  as generic subintervals in the partition for  $p_i$  and  $q_i$  respectively. The interval combination  $(a, b) \times (c, d)$  forms the basis for dose-finding decisions. A dose-finding decision table should firstly be elicited by investigators for all interval combinations. Li et al. call this a “preset table” for the TEPI design, which is fixed and elicited prior to the trial. An example of such a two-dimensional table is given in Table 2. We can see from this table there are four subintervals for toxicity (rows) and efficacy (columns), the intersection of which forms sixteen interval combinations. Each of the 16 combinations corresponds to a specific decision. Decision “E” denotes escalation (i.e., treating the next cohort of patients at the next higher dose). Decision “S” denotes staying at the current dose for treating the next cohort of patients. Decision “D” denotes de-escalation (i.e., treating the next cohort of patients at the next lower dose). These reflect the practical clinical actions needed when a particular combination of toxicity and efficacy data are observed at a certain dose level. For example, Table 2 shows that the interval combination  $(0.15, 0.25) \times (0, 0.25)$  for  $p_i$  and  $q_i$  corresponds to an action of “E”-escalation. This means that if the observed toxicity rate for a dose falls in  $(0.15, 0.25)$  and the observed efficacy rate is in  $(0, 0.25)$  with a high probability, the next patient cohort would be recommended to be treated at the next higher dose level. In order to formulate this table, we should have determined the following: (i) the maximum tolerated toxicity rate,  $p_T$ , and (ii) the minimum acceptable efficacy rate,  $q_E$ , at which the clinician is willing to treat future patients at the current dose level but should not be lower. For example, in Table 2, the values elicited by the clinical team for this trial are based on  $p_T = 0.35$  and  $q_E = 0.25$ .

Based on the preset table, Li et al. proposed a “local” decision-theoretic framework and derived a Bayes rule, which is equivalent to computing the joint unit probability mass (JUPM) for the TEPI. For a given region A, the JUPM is defined as the ratio of probability of the region versus the size of the region. Considering the two-dimensional unit square  $(0, 1) \times (0, 1)$ , the JUPM for each interval combination  $(a, b) \times (c, d)$  is

$$\text{JUPM}_{(a,b)}^{(c,d)} = \frac{\Pr(p_i \in (a, b), q_i \in (c, d) | \text{Data})}{(b - a) \times (d - c)}, 0 < a < b < 1, 0 < c < d < 1$$

Here,  $\Pr(p_i \in (a, b), q_i \in (c, d) | D)$  is the posterior probability of  $p_i$  and  $q_i$  falling in the subinterval  $(a, b)$  and  $(c, d)$ . Assume the priors for both  $p_i$  and  $q_i$  follow independent beta distributions  $\text{Beta}(\alpha_p, \beta_p)$  and  $\text{Beta}(\alpha_q, \beta_q)$  independently. The posterior distributions for  $p_i$  and  $q_i$  are  $\text{Beta}(\alpha_p + x_i, \beta_p + n_i - x_i)$  and  $\text{Beta}(\alpha_q + y_i, \beta_q + n_i - y_i)$ . Using these posterior distributions to update the JUPMs for all sixteen combination rectangular areas, we can find the highest JUPM value and its corresponding combination rectangular area  $(a^*, b^*)$  and  $(c^*, d^*)$ , which can be used to guide the dose-transition to treat the next cohort of patients.

The basic dose-finding concept of TEPI is as follows. We assume that the current patient cohort is treated at dose  $i$ . After the current cohort completes DLT and response evaluation, we compute the JUPMs for all the interval combinations in Table 2. The TEPI design recommends “E”, “S”, or “D” corresponding to the combination with the highest

JUPM value. Because for a given trial there are a finite number of possible toxicity and efficacy outcomes as binomial counts, for any toxicity and efficacy counts that can be observed in the trial, the TEPI dose-finding decisions can be precalculated. For example, if given the current computation, the combination rectangular area  $(0.15, 0.25) \times (0.25, 0.45)$  has the largest JUPM, in Table 2, the corresponding decision is “E”, that is, then we should escalate the dose.

To address ethical constraints, the *Safety rule* and *futility rule* are used as shown below.

- Safety rule: If  $\Pr(p_i > p_T | \text{Data}) > \eta$  for a  $\eta$  close to 1 (say, 0.95), exclude dose  $i, i + 1, \dots, D$  from future use for this trial (i.e., these doses will never be tested again in the trial) and the next cohort of patients will be treated at dose  $i - 1$ . This corresponds to a dosing action of “DU<sub>T</sub>” to de-escalate due to unacceptably high toxicity.
- Futility rules: If  $\Pr(q_i > q_E | \text{Data}) < \epsilon$  for a small  $\epsilon$  (say 0.3), then exclude dose  $i$  from future use in the trial. This corresponds to a dosing action of “EU” - escalate and never return due to unacceptable low efficacy - or “DU<sub>E</sub>” - de-escalate and never return this dose due to unacceptable low efficacy.

A dose level is considered “available” if it satisfies both the safety and futility rules. Only these doses can be used to treat subsequent patients.

At the end of the trial, we select the most desirable dose based on a utility score to balance between the toxicity and efficacy tradeoff. Utility-based decision criteria have been adopted widely in recent dose-finding trials. An elicited utility function for safety and efficacy should be constructed based on  $p_T$  and  $q_E$  by discussing with clinicians. In the developed Keyboard package, we provide the following three utility function options.

The first utility function is a function of the toxicity  $f_1(p)$  and the efficacy  $f_2(q)$ , where  $p$  and  $q$  denote the toxicity and efficacy rate. Both  $f_1(\cdot)$  and  $f_2(\cdot)$  are truncated linear functions, given by

$$f_1(p) = \begin{cases} 1, & p \in [0, p_1^*) \\ 1 - \frac{p - p_1^*}{p_2^* - p_1^*}, & p \in [p_1^*, p_2^*) \\ 0, & p \in [p_2^*, 1] \end{cases} \quad (1)$$

$$f_2(q) = \begin{cases} 0, & q \in [0, q_1^*) \\ 1 - \frac{q - q_1^*}{q_2^* - q_1^*}, & q \in [q_1^*, q_2^*) \\ 1, & q \in [q_2^*, 1] \end{cases} \quad (2)$$

where the  $p^*$ 's and  $q^*$ 's are prespecified cutoff values. From the above, we can see that utility  $f_1(p)$  is 1 if  $p \leq p_1^*$ , and 0 if  $p > p_2^*$ , and linearly decreasing with  $p$  if  $p \in [p_1^*, p_2^*)$ . Utility  $f_2(q)$  is set in a similar manner.

By assuming that the toxicity and efficacy are independent, the utility function that quantifies the benefit-risk trade-off at the current dose  $j$  that is defined as follows:

$$U(p_i, q_i) = f_1(p_i) \times f_2(q_i) \quad (3)$$

For each dose  $i$ , we can use a numerical approximation approach to compute the posterior expected utility,  $E[U(p_i, q_i) | \text{Data}]$ . For example, we can generate a total of  $T$  random samples from the posterior distributions. For each sample  $t$ , we generate  $p^t = (p_1^t, \dots, p_D^t)$  and  $q^t = (q_1^t, \dots, q_D^t)$  as a random sample of  $D$  probabilities of toxicity



**Table 3**  
Dose escalation and de-escalation rules for the Keyboard design.

	Number of patients treated									
	3	6	9	12	15	18	21	24	27	30
Escalate if number of DLT $\leq$	0	1	2	2	3	4	5	5	6	7
De-escalate if number of DLT $\geq$	2	3	4	5	6	7	8	9	10	11
Eliminate if number of DLT $\geq$	3	4	5	7	8	9	10	11	12	14

and efficacy, respectively. We perform the isotonic transformation on  $p^i$  to obtain  $\hat{p}^i = (\hat{p}_1^i, \dots, \hat{p}_D^i)$ , where  $\hat{p}_i^i \leq \hat{p}_j^i$  if  $i < j$ , which ensures the non-decreasing toxicity with the increased dose. For each dose  $i$ , based on the samples  $\hat{p}_i^i$  and  $\hat{q}_i^i$ , the corresponding utility score is  $U^i(\hat{p}_i^i, \hat{q}_i^i) = f_1(\hat{p}_i^i) \times f_2(\hat{q}_i^i)$ . Thus, the estimated posterior expected utility is given by  $\hat{E}[U(p_i, q_i)|Data] = \frac{1}{T} \sum_{t=1}^T U^i(\hat{p}_i^i, \hat{q}_i^i)$ .

Finally, the selected the optimal dose  $\hat{d}$  is given by

$$\hat{d} = \{i : \operatorname{argmax}_{i \in \{1, \dots, D\}} \hat{E}[U(p_i, q_i)|Data]\}.$$

Another utility function depends on a marginal toxicity probability  $\pi_{T,i} = \Pr(Y_T = 1|i = d)$  and a marginal efficacy probability  $\pi_{E,i} = \Pr(Y_E = 1|i = d)$ , which is defined as follows:

$$U(p_i, q_i) = \pi_{E,i} - w_1 \times \pi_{T,i} \tag{4}$$

where,  $w_1$  is a pre-specified weight. This trade-off function describes how many patients are willing to trade an increase of  $w_1$  in the DLT rate for a unit increase in the efficacy rate. If  $w_1 = 0$ , we have a special case in which the dose with the highest efficacy is the most desirable.

The third utility function also depends on the marginal toxicity probability  $\pi_{T,i} = \Pr(Y_T = 1|i = d)$  and the efficacy probability  $\pi_{E,i} = \Pr(Y_E = 1|i = d)$ , but it puts an additional penalty for over-toxicity and is defined as follows:

$$U(p_i, q_i) = \pi_{E,i} - w_1 \times \pi_{T,i} - w_2 * \pi_{T,i} * I(\pi_{T,i} > \rho) \tag{5}$$

where  $w_1$  and  $w_2$  are pre-specified weights,  $I(\cdot)$  is an indicator function, and  $\rho$  is a pre-specified toxicity threshold deemed of substantial concern and can be chosen as the target toxicity rate. Compared to the above second utility function of (4), this trade-off function is more flexible and allows for imposing a higher penalty (i.e.,  $w_1 + w_2$ ) when the true DLT rate  $\pi_{T,i}$  exceeds the threshold  $\rho$ .

Once the utility score is computed for all the doses, then the OBD can be identified by:

$$d_{OBD} = \{i : \operatorname{argmax}_{i \in D} [U(p_i, q_i)|Data]\}$$

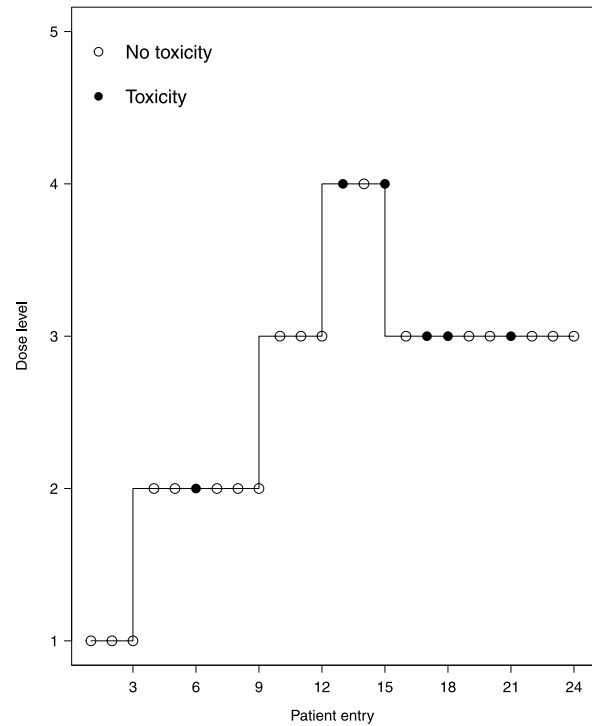
### 3. Trial examples

In this section, we demonstrate how to design various phase I dose-finding trials with examples. Detailed steps of how to implement the Keyboard R package are included in [Appendix](#).

#### 3.1. Single-agent phase I trial

Consider a single-agent phase I trial with five dose levels, in which the objective is to find the MTD with a target DLT rate of 0.3. The maximum sample size is 30 patients with a cohort size of 3. To design and conduct this trial, we first use the function `get.boundary.kb(target = 0.3, ncohort = 10, cohortsize = 3)` to yield the dose escalation and de-escalation decision rules shown in [Table 3](#).

If we assume that the trial starts by treating the first cohort of three patients at dose 1, based on [Table 3](#), none of the patients have the DLT (0/3). Therefore, we should escalate to dose 2 to treat the next cohort of patients. If at dose level 2, none of the patients experience the DLT (0/3), we should escalate to dose 3. If at dose level 3, two of three patients experience the DLT (2/3), based on [Table 3](#), we should de-escalate from dose 3 to dose 2. Note that as of now, 6 patients will



**Fig. 3.** Dose-transition path of a single-agent phase I dose-finding trial using the Keyboard design.

be treated at dose 2. If one of the patients among these six has DLT (1/6), we will escalate to dose 3 again. Currently, we have 6 patients at dose 3. If among these 3 newly enrolled patients, none has DLT, that is, at dose 3, there are a total of 2 DLTs among 6 (2/6), based on [Table 3](#), we should stay. That is, we continue to treat the next cohort of patients at dose 3. Note that the number of patients at dose 3 will finally be 9 in total. This process will continue to make dose transitions until exhausting the maximum sample size or meeting early stopping criteria. In [Fig. 3](#), we show the complete path of dose-transition from this trial.

Upon finishing the trial, we would have selected the MTD. We assume that the observed number of patients and DLTs across five doses are  $n = c(3, 6, 12, 3, 0)$  and  $y = c(0, 1, 3, 2, 0)$ , respectively, that is, this study finally stopped with a total of sample size 24 instead of 30 pre-planned in the protocol. We can use the function `select.mtd.kb(target = 0.3, ntox = y, npts = n)`, which recommends dose 3 as the MTD, with an estimated DLT rate of 25.0% with 95% CI of (6%, 52%).

#### 3.2. Drug-combination trial

Consider a drug-combination trial, there are 3 doses for agent A and 5 doses for agent B. The objective is to find the MTD with a target DLT rate of 0.3. The maximum sample size is 30 patients, treated in cohorts with the size of 3. We assume that the trial starts by treating the first

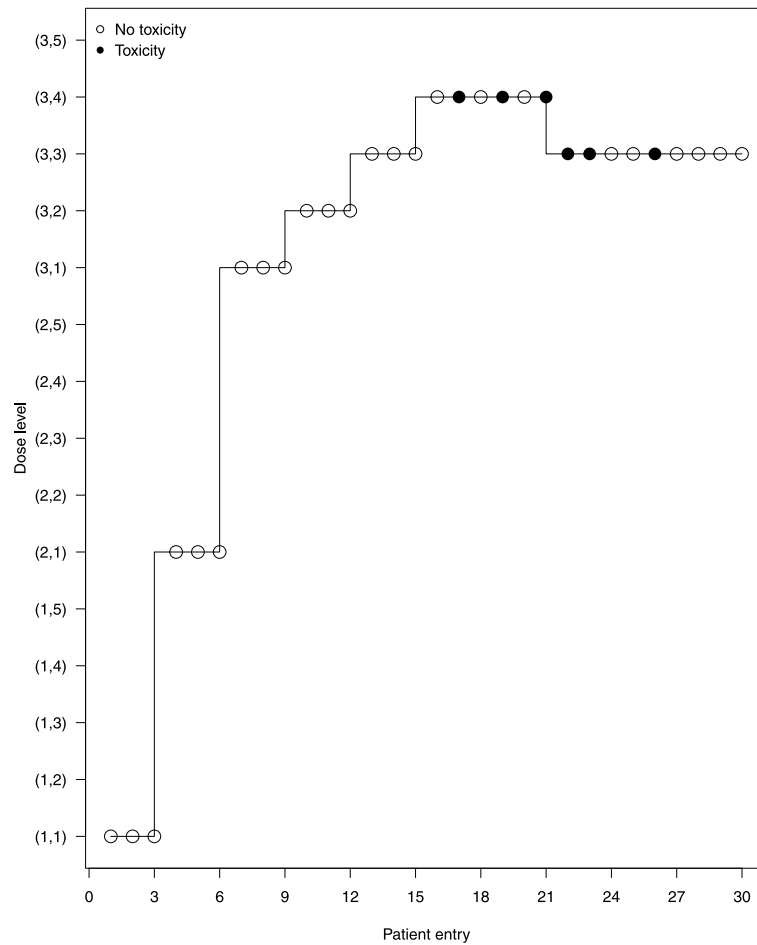


Fig. 4. Illustration of a 3 × 5 combination trial with the cohort size of three to find the MTD. Open circle indicates patients without the DLT, and solid circle denotes patients with the DLT.

cohort of 3 patients at the lowest dose combination (1,1), which means dose level 1 is combination of agent A and B. If no DLT was observed, the observed data can be described in the following matrices.

$$n = \begin{pmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

where  $n$  records the number of patients treated at each dose combination, and  $y$  records the number of patients who experience DLT at each dose combination. In matrices  $n$  and  $y$ , each entry  $(j, k)$  records the observed data associated with combination  $(j, k)$ .

To determine the dose for the second cohort of patients, we use the function `next.comb.kb(target = 0.3, npts = n, ntox = y, dose.curr = c(1, 1))`, which recommends escalating the dose to combination (2,1). If after treating the second cohort of patients, none experiences the DLT, the updated data matrices are

$$n = \begin{pmatrix} 3 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

To determine the dose for the third cohort of patients, we use `next.comb.kb(target = 0.3, npts = n, ntox = y, dose.curr = c(2, 1))` with updated  $y$ ,  $n$  and `dose.curr` to treat the third cohort of patients. We repeat this procedure until the maximum sample size was reached. Fig. 4 shows the dose assignments path for all 30 patients. For example, at dose combination (3,2), there were 0

DLTs; the function recommends escalating the dose to combination (3,3). When the trial is completed, the number of patients treated at each dose combination and the corresponding number of patients who experienced toxicity at each dose combination are

$$n = \begin{pmatrix} 3 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 \\ 3 & 3 & 12 & 6 & 0 \end{pmatrix}, \quad y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 3 & 0 \end{pmatrix}.$$

We use the function `select.mtd.comb.kb(target = 0.3, npts = n, ntox = y)`, which recommends the dose combination (3, 3) as the MTD.

### 3.3. Phase I/II trial to find the optimal biological dose (OBD)

Consider a single-agent phase I trial with five dose levels, the primary objective is to find the OBD. The maximum sample size is 30 patients, treated in cohorts with the size of 3. Clinical team determines that the maximum tolerated toxicity rate  $p_T$  is 0.20 and the minimum acceptable efficacy rate  $q_E$  is 0.40.

To design and show how to conduct this trial, we use the following function in the first place to set up the “preset table” shown in Table 2. Partial outputs of decision rules are presented in Table 4, which can be generated by executing the following code.

```
R> oc.obd2.kb <- get.decision.obd.kb
      (toxicity.low = 0.15, toxicity.moderate = 0.25,
       +           toxicity.high = 0.35,
```

**Table 4**  
Dose escalation and de-escalation table for the phase I/II OBD-finding design (partial outputs).

	N	T	R	Decision
1	3	0	0	EUE
2	3	0	1	E
3	3	0	2	E
4	3	0	3	E
5	3	1	0	DUE
6	3	1	1	S
7	3	1	2	S
8	3	1	3	S
9	3	2	0	DUE
10	3	2	1	D
11	3	2	2	D
12	3	2	3	D
13	3	3	0	DUT
14	3	3	1	DUT
15	3	3	2	DUT
16	3	3	3	DUT
17	6	0	0	EUE
18	6	0	1	EUE
19	6	0	2	E
20	6	0	3	E
21	6	0	4	E
22	6	0	5	E
23	6	0	6	E
24	6	1	0	EUE
25	6	1	1	EUE
26	6	1	2	E
27	6	1	3	E
28	6	1	4	E
29	6	1	5	S
30	6	1	6	S
31	6	2	0	DUE
32	6	2	1	DUE
33	6	2	2	S
⋮	⋮	⋮	⋮	⋮
99	9	3	3	S
⋮	⋮	⋮	⋮	⋮
210	12	3	5	S
⋮	⋮	⋮	⋮	⋮

```
+ efficacy.low = 0.25,
+ efficacy.moderate = 0.45,
+ efficacy.high = 0.65,
+ target.toxicity = 0.20,
+ target.efficacy = 0.40,
+ ncohort = 10, cohortsize = 3)
```

To show how to use the above decision table, for example, if a trial starts by treating the first cohort of 3 patients at dose level 1 with none of the patients having the DLT and one patient showing response, according to the second row - (N = 3, T = 0, R = 1), the 'Decision' is 'E', that is, we should escalate the dose to level 2 to treat the second cohort of 3 patients. We also can run the function `decision.finding(out.matrix = output.matrix, n = 3, t = 0, r = 1)` to guide the next recommend dose level.

If currently at dose 2, one of three patients experienced DLT and two have responses, by using the decision table of looking at the sixth row - (N = 3, T = 1, R = 1) or by using the command `decision.finding(out.matrix = output.matrix, n = 3, t = 1, r = 1)`, the 'Decision' is 'S', that is, to continue to treat the next cohort of patients at dose 2. If at dose 2, among the newly enrolled three patients, none has the DLT and one has the responses, that is, currently, (N = 6, T = 1, R = 2), the 'Decision' is 'E', that is, we should escalate to treat next cohort of patients at dose 3. We repeat this process until maximum sample size reached or stopping rules are satisfied. The complete dose-transition path is shown in Fig. 5.

```
> target.toxicity <- 0.2
> target.efficacy <- 0.4
```

```
> npts <- c(3,6,12,3,0)
> ntox <- c(0,1,3,2,0)
> neff <- c(1,2,5,2,0)
> obd <- select.obd.kb
(target.toxicity=target.toxicity,
+ target.efficacy=target.efficacy,
+ npts = npts,
+ ntox = ntox, neff = neff)
> print(obd)
$name
[1] "select.obd.kb"

$obd1
[1] 3

$obd2
[1] 3

$obd3
[1] 3
```

For the above trial path plot, we can see that, at the end of the trial, at dose 1, there were 3 patients with 0 DLT and 1 responses, at dose 2, there were 6 patients with 1 DLT and 2 responses, at dose 3, there were 12 patients with 3 DLTs and 5 responses, and at dose 4, there were 3 patients with 2 DLTs and 2 responses. These numbers can be in the format of  $n = c(3,6,12,3,0)$ , and  $t = c(0,1,3,2,0)$ , and  $r = c(1,2,5,2,0)$  and we use function `select.obd.kb(target.toxicity = 0.2, target.efficacy = 0.4, npts = n, ntox = t, neff = r)` to select the OBD. The recommended dose level using utility functions of (3) as an example is dose 3.

#### 4. Conclusion

In this paper, we review three interval-based Bayesian design methods, the keyboard single- and dual-agent phase I dose-finding design for the MTD and the TEPI design for identifying the OBD. These designs have been demonstrated to have desirable performances and their usages are simple and transparent due to pre-tabulated decision tables. We developed an R package Keyboard, which can assist practitioners to design, conduct, and analyze trials by using these methods. We also use examples to demonstrate how to use this package and step-by-step codes are included in Appendix.

#### 5. Availability and requirements

Project name: Keyboard

Project home page:

<https://cran.r-project.org/web/packages/Keyboard/index.html>

Operating system(s): Platform independent

Programming language: R

Other requirements: R 3.4.0 or above

License: GPL-2

Any restrictions to use by non-academics: none

#### 6. Abbreviations

MTD: maximum tolerated dose; OBD: optimal biological dose; DLT: dose-limiting toxicity; CRM: continual reassessment method; TEPI: toxicity and efficacy probability interval; JUPM: joint unit probability mass; BOIN: Bayesian optimal interval design.

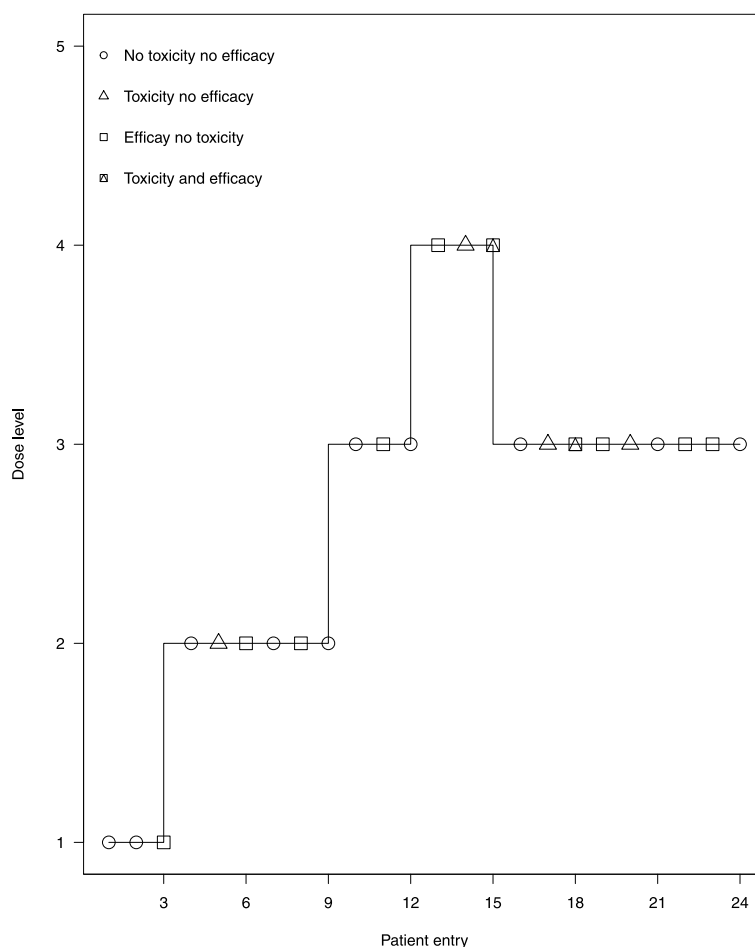


Fig. 5. Illustration of a single-agent phase I/II trial for finding the optimal biology dose (OBD).

## Funding

Cheng's work was partially supported by NIH/NCI cancer Center Support Grant, United States of America CA21765. Li's work was partially supported by National Natural Science Foundation of China Grant 82273728. Pan, Tang, Sun, and Cheng's work were partially supported by American Lebanese Syrian Associated Charities (ALSAC), United States of America.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

We thank the editor and two anonymous reviewers for their constructive comments, which helped us to improve the manuscript. We also thank Dr. Vani Shanker for scientific editing of this manuscript.

## Appendix

R package Keyboard contains functions that implement the designs introduced in the main paper. In this Appendix, we show how to implement these methods step-by-step by using the Keyboard R package. We firstly list the corresponding functions for the three design methods introduced in the main paper in the following.

### Single-agent MTD-finding design

- `get.boundary.kb(...)`: This function is used to generate the optimal dose-escalation and de-escalation boundaries for conducting a single-agent trial with the Keyboard design.
- `select.mtd.kb(...)`: This function is used to select the MTD after the single-agent trial is completed.
- `get.oc.kb(...)`: This function is used to generate the operating characteristics of the Keyboard design for single-agent trials.

### Drug-combination MTD-finding design

- `get.boundary.comb.kb(...)`: This function is used to generate the optimal dose escalation and de-escalation boundaries for conducting a drug-combination trial with the Keyboard design.
- `next.comb.kb(...)`: This function is used to determine the dose combination for the next cohort of patients in drug-combination trials that aim to find one MTD.
- `select.mtd.comb.kb(...)`: This function is used to select the MTD after the drug-combination trial is completed.
- `get.oc.comb.kb(...)`: This function is used to generate the operating characteristics of the Keyboard design for drug-combination trials.

### Phase I/II trial OBD-finding design

- `get.decision.obd.kb(...)`: This function is used to generate the boundary table and decision matrix for single-agent phase I/II trials designed to find the OBD.
- `select.obd.kb(...)`: This function is used to select the OBD at the end of a trial.
- `get.oc.obd.kb(...)`: This function is used to generate operating characteristics.



```

> get.boundary.kb(target=0.3, ncohort=10, cohortsizes=3)
$boundary_tab

Number of patients treated 3 6 9 12 15 18 21 24 27 30
Escalate if # of DLT <= 0 1 2 2 3 4 5 5 6 7
de-escalate if # of DLT >= 2 3 4 5 6 7 8 9 10 11
Eliminate if # of DLT >= 3 4 5 7 8 9 10 11 12 14

$full_boundary_tab

Number of patients treated 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
Escalate if # of DLT <= 0 0 0 0 1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4 5 5 5 5 6 6 6 6 7 7
de-escalate if # of DLT >= 1 1 2 2 2 3 3 3 4 4 4 5 5 5 6 6 6 7 7 7 7 8 8 9 9 9 10 10 10 11 11
Eliminate if # of DLT >= NA 2 3 3 4 4 5 5 6 6 7 7 8 8 8 9 9 9 10 10 11 11 11 12 12 12 13 13 14

```

Fig. 6. Boundary table of a phase I dose-finding example for the single-agent.

### Phase I/II trial-automatically generating decision tables

Different from the above, the following functions do not rely on pre-specified decision tables (Details and example are introduced shortly).

- `get.decision.obd2.kb(...)`: This function is used to provide the boundary table and decision matrix for the phase I/II trials automatically.
- `select.obd.kb(...)`: This function is used to select the optimal biological dose (OBD) at the end of a single-agent phase I/II trial.
- `get.oc.obd2.kb(...)`: This function is used to generate the operating characteristics, and the decision tables is generated automatically.

#### A.1. Phase I dose-finding trial with a single agent

To design a phase I dose-finding trial with a single agent, we can use the function `get.boundary.kb(.)` to obtain the dose escalation and de-escalation boundaries. This function has the following arguments:

- `target` The target dose-limiting toxicity (DLT) rate.
- `ncohort` The total number of cohorts.
- `cohortsizes` The number of patients in the cohort.
- `marginL` The difference between the target and the lower bound of the “target key” (proper dosing interval) to be defined — the default value is 0.05.
- `marginR` The difference between the target and the upper bound of the “target key” (proper dosing interval) to be defined — the default value is 0.05.
- `n.earlystop` The early stopping parameter. If the number of patients treated at the current dose reaches `n.earlystop`, then stop the trial and select the MTD based on the observed data.
- `cutoff.eli` The cutoff value to eliminate the over-toxic dose and all higher doses for safety — the default value is 0.95.
- `extrasafe` Set `extrasafe = TRUE` to impose a stricter stopping rule for extra safety, expressed as the stopping boundary value in the result.

For the `extrasafe`, it means that a dose  $d$  will be eliminated due to toxicity if  $\Pr(p_d > \phi | D_d = (n_d, y_d)) > \text{cutoff.eli} - \text{offset}$ . For instance, the default value `offset = 0.05`.

Here we show an example. Assuming we want to conduct a phase I trial with  $D = 5$  dose levels and the target toxicity rate of  $\phi = 0.3$ . The maximum sample size is 30 patients, and patients are treated in cohorts of size 3. Using the default values of `marginL`, `marginR`, and `cutoff.eli`, we can obtain the decision boundary by applying `get.boundary.kb(.)` with the outputs shown as in Fig. 6

For developing the trial protocol, we need to provide operating characteristics (OCs) of scenarios interested by the clinical team. There is a `get.oc.kb(.)` can be used by simulations to provide the empirical OCs. For example, if we want to explore the OC of a scenario with assumed true toxicity rates of (0.05, 0.15, 0.3, 0.45, 0.6) for five doses. For this scenario, we assume that dose 3 is the true MTD. We can use the Fig. 7 code to get the empirical OCs for this scenario.

From the above outputs, many results are reported. For instance, there is 54.3% chance to select dose 3 to be the MTD if using the keyboard design, among 30 patients, on average, there will have 10.995 patients being treated at dose 3. All the other outputs can be found detailed explanations in the R package manual.

When a trial completes, we should select the MTD based on the data. For example, assuming that observed number of patients and DLTs across five doses are  $n = c(3, 6, 12, 3, 0)$  and  $y = c(0, 1, 3, 2, 0)$ , respectively, that is, this study finally stopped with a total of sample size 24 instead of 30 pre-planned in the protocol. We can use the function `select.mtd.kb(target = 0.3, ntox = y, npts = n)`, which recommends dose 3 as the MTD, with the estimated DLT rate of 25.0% with 95% CI of (6%, 52%) (see Fig. 8).

#### A.2. Phase I dose-finding drug combination trial

We are now showing how to use the Keyboard R package to design a drug combination phase I dose-finding trial (see Fig. 9).

#### Conduct the trial

The function `next.comb.kb(.)` can be used to find which dose should be assigned to new cohort of patients based on the updated data. This function shares a similar set of arguments with the function `get.boundary.kb(.)` described previously:

- `npts` A matrix recording the number of patients treated at each dose combination.
- `ntox` A matrix recording the number of patients who experienced toxicity at each dose combination.
- `dose.curr` The current dose combination (the dose combination that was used to treat the most recently enrolled cohort of patients).

Suppose that we have a  $3 \times 4$  drug-combination trial with 3 dose levels of agent A and 4 dose levels of agent B, aiming to find a MTD that has a target toxicity rate of 0.3. The maximum sample size is assumed to be 48 patients, and patients are treated in cohort sizes of 3. Let  $(j, k)$  denote the combination of the  $j$ th dose level of agent A and the  $k$ th dose level of agent B. The trial can be conducted as follows. We start the trial by treating the first cohort of 3 patients at the lowest dose (2, 2). Assuming that none of the patients experienced DLT, the data from the first cohort of patients are given by

$$n = \begin{pmatrix} 3 & 0 & 0 & 0 & 0 \\ 7 & 6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

where  $n$  records the number of patients treated at each dose combination, and  $y$  records the number of patients who experienced toxicity at each dose combination. In matrixes  $y$  and  $n$ , entry  $(j, k)$  records the data associated with combination  $(j, k)$ . To determine the dose for the second cohort of patients, we call function `next.comb.kb(.)`:

From the above output, the recommended dose combination for next cohort of patients is (3, 2). Therefore, we should escalate to dose

```

> oc <- get.oc.kb(target=0.3, p.true=c(0.05, 0.15, 0.3, 0.45, 0.6),
+               ncohort=10, cohortsize=3, ntrial=1000)
>
> oc
$selpercent
[1] 1.1 22.9 54.3 20.6 1.1

$npatients
[1] 4.161 9.228 10.995 4.881 0.735

$ntox
[1] 0.195 1.345 3.283 2.192 0.424

$totaltox
[1] 7.439

$totaln
[1] 30

$percentstop
[1] 0

$overdose60
[1] 2.9

$overdose80
[1] 0

$simu.setup
  target p.true ncohort cohortsize startdose marginL marginR cutoff.eli extrasafe offset ntrial dose
1    0.3  0.05     10           3         1    0.05  0.05    0.95   FALSE  0.05  1000   1
2    0.3  0.15     10           3         1    0.05  0.05    0.95   FALSE  0.05  1000   2
3    0.3  0.30     10           3         1    0.05  0.05    0.95   FALSE  0.05  1000   3
4    0.3  0.45     10           3         1    0.05  0.05    0.95   FALSE  0.05  1000   4
5    0.3  0.60     10           3         1    0.05  0.05    0.95   FALSE  0.05  1000   5

```

Fig. 7. Empirical operating characteristics (OCs) of a phase I dose-finding example for the single-agent.

```

> n <- c(3, 6, 12, 3, 0)
> y <- c(0, 1, 3, 3, 0)
>
> selmtd <- select.mtd.kb(target=0.3, npts=n, ntox=y)
>
> selmtd
$target
[1] 0.3

$MTD
[1] 3

$sp_est
  dose phat      CI
1    1 0.02 (0.00,0.20)
2    2 0.17 (0.01,0.53)
3    3 0.25 (0.06,0.52)
4    4 0.98 (0.80,1.00)
5    5 ---- (----,----)

$sp_overdose
[1] "0.01" "0.18" "0.32" "1.00" "----"

```

Fig. 8. Selecting the MTD of a phase I dose-finding example for the single-agent.

3 for agent A and still at dose 2 for agent B. We repeat this procedure until the maximum sample size is reached.

### Obtain the operating characteristics

The function `get.oc.comb.kb()` can be used to obtain the operating characteristics of the Keyboard drug-combination design. This function also shares the same set of arguments as the function `get.oc.kb()`, except that `p.true` is now a matrix (rather than a vector) and `startdose` is a vector of length 2 (rather than a scalar) (see Fig. 11)

Consider a  $3 \times 5$  combination trial scenario with the true toxicity probabilities

$$p.true = \begin{pmatrix} 0.01 & 0.03 & 0.10 & 0.20 & 0.30 \\ 0.03 & 0.05 & 0.15 & 0.30 & 0.60 \\ 0.08 & 0.10 & 0.30 & 0.60 & 0.75 \end{pmatrix},$$

and the target toxicity rate of 0.30. Here, we show how to obtain the operating characteristics (OCs) based on 1000 simulated trials with a maximum sample size of 45 by a cohort size of 3. Partial outputs are shown as in Fig. 10.

### Select a MTD when the trial is completed

When the trial is completed, based on the observed data, we can select a MTD using the function `select.mtd.comb.kb()`. This function has seven arguments: `target`, `npts`, `ntox`, `cutoff.eli`, `extrasafe`, and `offset`, where the descriptions of `cutoff.eli`, `extrasafe`, and `offset` are the same as those in `get.boundary.kb()`. Assume that the number of patients treated at each dose combination and the corresponding number of patients who experienced toxicity at each dose combination are

$$n = \begin{pmatrix} 6 & 3 & 0 & 0 & 0 \\ 6 & 24 & 9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 5 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

From the above output, we can see that dose combination (2, 2), that is, combination of agent A of dose 2 and agent B of dose 2 is selected as the MTD.

### A.3. Phase I trial for the optimal biological dose (OBD)

**Design and conduct the trial** To design a phase I/II trial for finding the OBD, we need to run the function `get.decision.obd.kb()` to obtain the boundary table and decision table. This function needs the following arguments to input:

```

> n<-matrix(c(3, 0, 0, 0, 0,7, 6, 3, 0, 0, 0, 0, 0, 0),
+           ncol=5, byrow=TRUE)
> y<-matrix(c(0, 0, 0, 0, 0, 1, 1,3, 0, 0, 0, 0, 0, 0),
+           ncol=5, byrow=TRUE)
> next.comb.kb(target=0.3, npts=n, ntox=y, dose.curr=c(2, 2))
$next_dc
[1] 3 2

```

Fig. 9. Recommend a next dose for a phase I drug combination trial.

```

> p.true<-matrix(c(0.01, 0.03, 0.10, 0.20, 0.30, 0.03, 0.05, 0.15,
+                 0.30, 0.60, 0.08, 0.10, 0.30, 0.60, 0.75),
+               byrow=TRUE, ncol=5)
> get.oc.comb.kb(target=0.3, p.true, ncohort=15, cohortsize=3,
+               n.earlystop=12, startdose=c(1, 1), ntrial=100)
$name
[1] "get.oc.comb.kb"

$target
[1] 0.3

$sp.true
  [,1] [,2] [,3] [,4] [,5]
[1,] 0.01 0.03 0.10 0.2 0.30
[2,] 0.03 0.05 0.15 0.3 0.60
[3,] 0.08 0.10 0.30 0.6 0.75

$ncohort
[1] 15

$cohortsize
[1] 3

$total
[1] 45

$ntrial
[1] 100

$selpercent
  [,1] [,2] [,3] [,4] [,5]
[1,] 0 0 2 3 5
[2,] 0 0 12 20 0
[3,] 0 10 44 4 0

$pcs
[1] 69

$nmtd
[1] 32.32227

$nmtd.over
[1] 12.70142

$nmtd.over.sd
[1] 0.04181732

$nptsdose
  [,1] [,2] [,3] [,4] [,5]
[1,] 3.03 1.41 1.08 1.05 0.72
[2,] 1.77 1.98 3.30 2.79 0.93
[3,] 0.90 2.88 6.72 2.82 0.27

$ntoxdose
  [,1] [,2] [,3] [,4] [,5]
[1,] 0.01 0.03 0.09 0.22 0.17
[2,] 0.03 0.07 0.50 0.75 0.58
[3,] 0.08 0.29 2.09 1.75 0.21

$totaltox
[1] 6.9

```

Fig. 10. Empirical operating characteristics (OCs) of a phase I drug combination trial.

- `toxicity.low` The upper boundary for the low toxicity interval.
- `toxicity.moderate` The upper boundary for the moderate toxicity interval.
- `toxicity.high` The upper boundary for the high toxicity interval.
- `efficacy.low` The upper boundary for the low efficacy interval.
- `efficacy.moderate` The upper boundary for the moderate efficacy interval.
- `efficacy.high` The upper boundary for the high efficacy interval.

```

> n<-matrix(c(6, 3, 0, 0, 0, 6, 24, 9, 0, 0, 0, 0, 0, 0),
+           ncol=5, byrow=TRUE)
> y<-matrix(c(0, 0, 0, 0, 0, 1, 5, 4, 0, 0, 0, 0, 0, 0),
+           ncol=5, byrow=TRUE)
> select.mtd.comb.kb(target=0.30,npts=n, ntox=y)
$target
[1] 0.3

$MTD
      DoseA DoseB
[1,]     2     2

$p_est
      [,1] [,2] [,3] [,4] [,5]
[1,] 0.01 0.02  NA  NA  NA
[2,] 0.17 0.21 0.45  NA  NA
[3,]  NA  NA  NA  NA  NA

```

Fig. 11. Selecting the MTD for a phase I drug combination trial.

```

> toxicity.low <- 0.15
> toxicity.moderate <- 0.25
> toxicity.high <- 0.35
> efficacy.low <- 0.25
> efficacy.moderate <- 0.45
> efficacy.high <- 0.65
> target.toxicity=0.20
> target.efficacy=0.40
> cohortsize=3
> ncohort=10
> decision.obd <- get.decision.obd.kb(toxicity.low = toxicity.low,
+                                     toxicity.moderate= toxicity.moderate,
+                                     toxicity.high = toxicity.high,
+                                     efficacy.low = efficacy.low,
+                                     efficacy.moderate = efficacy.moderate,
+                                     efficacy.high = efficacy.high,
+                                     target.toxicity=target.toxicity,
+                                     target.efficacy=target.efficacy,
+                                     cohortsize=cohortsize, ncohort=ncohort)

```

Fig. 12. Code to drive the decision table of a phase I trial to find the optimal biological dose (OBD).

- `target.toxicity` The target DLT rate.
- `target.efficacy` The target efficacy rate.
- `ncohort` The total number of cohorts.
- `cohortsize` The number of patients in the cohort.
- `cutoff.eli.toxicity` The cutoff value to eliminate a dose with an unacceptably high toxicity for safety — the default value is 0.95.
- `cutoff.eli.efficacy` The cutoff value to eliminate a dose with unacceptably low efficacy — the default value is 0.3.

As an example, suppose we want to conduct a phase trial for finding the OBD with  $D = 5$  dose levels with a target toxicity rate of 0.20 and a target efficacy rate of 0.40. Maximum number of patients is 30 and patients are treated by the cohort size of 3. We can use the function `get.decision.obd.kb(.)` to derive the decision table (see Fig. 12).

The “preset” table of 4-by-4 can be obtained by applying `decision.obd$boundary.table` (see Fig. 13).

Detailed decision lists can be obtained by applying `decision.obd$decision.matrix` and part of outputs are shown in Fig. 14.

Here are explanations for the above decision actions. Assuming the current does is  $j$ ,

- D: deescalate to a previous dose  $j - 1$ .
- E: escalate to a next dose  $j + 1$ .
- S: stay at the current dose  $j$ .
- EUE: escalate to a higher dose  $j + 1$  due to the unacceptable low efficacy.

- DUE: deescalate to a lower dose  $j - 1$  due to the unacceptable low efficacy.
- DUT: deescalate to a lower dose  $j - 1$  due to the unacceptable high toxicity.
- Safety stopping rule: if  $\Pr(p_j > p_T | \text{Data}) > \eta$ , the default value of  $\eta$  is 0.95, then any dose  $\geq j$  will be excluded from the trial.
- Default futility rule: if  $\Pr(q_j > q_E | \text{Data}) < \epsilon$ , the default value of  $\epsilon$  is 0.3, then dose  $j$  will be excluded from the trial.

From the above output, for example, we can see that if there are 3 patients on a specific dose level with 1 toxicity and 1 response, the decision would be “S” (the sixth row in Fig. 14), that is, we will continue to administer the current dose to the next cohort of patients.

#### Obtain the operating characteristics

The function `get.oc.obd.kb(.)` can be used to obtain the operating characteristics (OCs) for the phase I trial to find the OBD. This function needs the following arguments to input:

- `toxicity.low` The upper boundary for the low toxicity interval.
- `toxicity.moderate` The upper boundary for the moderate toxicity interval.
- `toxicity.high` The upper boundary for the high toxicity interval.
- `efficacy.low` The upper boundary for the low efficacy interval.
- `efficacy.moderate` The upper boundary for the moderate efficacy interval.

```
> decision.obd$boundary.table
      T1      T2      EF1      EF2      DECISION
[1,] "0"      "0.15" "0"      "0.25" "E"
[2,] "0"      "0.15" "0.25" "0.45" "E"
[3,] "0"      "0.15" "0.45" "0.65" "E"
[4,] "0"      "0.15" "0.65" "1"      "E"
[5,] "0.15"   "0.25" "0"      "0.25" "E"
[6,] "0.15"   "0.25" "0.25" "0.45" "E"
[7,] "0.15"   "0.25" "0.45" "0.65" "E"
[8,] "0.15"   "0.25" "0.65" "1"      "S"
[9,] "0.25"   "0.35" "0"      "0.25" "D"
[10,] "0.25"  "0.35" "0.25" "0.45" "S"
[11,] "0.25"  "0.35" "0.45" "0.65" "S"
[12,] "0.25"  "0.35" "0.65" "1"      "S"
[13,] "0.35"  "1"      "0"      "0.25" "D"
[14,] "0.35"  "1"      "0.25" "0.45" "D"
[15,] "0.35"  "1"      "0.45" "0.65" "D"
[16,] "0.35"  "1"      "0.65" "1"      "D"
```

Fig. 13. Code to generate the “preset” table of a phase I trial to find the optimal biological dose (OBD).

```
> decision.obd$decision.matrix
  N T R Decision
1  3 0 0     EUE
2  3 0 1       E
3  3 0 2       E
4  3 0 3       E
5  3 1 0     DUE
6  3 1 1       S
7  3 1 2       S
8  3 1 3       S
9  3 2 0     DUT
10 3 2 1     DUT
11 3 2 2     DUT
12 3 2 3     DUT
13 3 3 0     DUT
14 3 3 1     DUT
15 3 3 2     DUT
16 3 3 3     DUT
17 6 0 0     EUE
18 6 0 1     EUE
19 6 0 2       E
20 6 0 3       E
21 6 0 4       E
22 6 0 5       E
23 6 0 6       E
24 6 1 0     EUE
25 6 1 1     EUE
26 6 1 2       E
27 6 1 3       E
28 6 1 4       E
29 6 1 5       S
30 6 1 6       S
31 6 2 0     DUE
32 6 2 1     DUE
33 6 2 2       S
34 6 2 3       S
35 6 2 4       S
36 6 2 5       S
37 6 2 6       S
38 6 3 0     DUT
39 6 3 1     DUT
40 6 3 2       S
41 6 3 3       S
42 6 3 4       S
43 6 3 5       S
44 6 3 6       S
45 6 4 0     DUT
46 6 4 1     DUT
47 6 4 2     DUT
48 6 4 3     DUT
```

Fig. 14. Decision lists (part) of a phase I trial to find the optimal biological dose (OBD).

- `efficacy.high` The upper boundary for the high efficacy interval.
- `target.toxicity` The target DLT rate.
- `target.efficacy` The target efficacy rate.
- `ncohort` The total number of cohorts.
- `cohortsize` The number of patients in a cohort.
- `n.early` The early stopping parameter. If the number of patients treated at the current dose reaches `n.early`, then we stop the trial and select the MTD based on the observed data.
- `startdose` The starting dose level.
- `p.true` A vector containing the true toxicity probabilities of the investigational dose levels.
- `q.true` A vector containing the true efficacy probabilities of the investigational dose levels.
- `ntrial` The total number of trials to be simulated.
- `seed` The random seed for simulation.
- `p1` The cutoff lower limit for safety utility function (3):  $U = f(p) \times f(q)$ .  $f(p)$  is the function for toxicity probability, and  $f(q)$  is the function for efficacy probability.
- `p2` The cutoff upper limit for safety utility function (3).
- `q1` The cutoff lower limit for efficacy utility function (3).
- `q2` The cutoff upper limit for efficacy utility function (3).
- `cutoff.eli.toxicity` The cutoff value to eliminate a dose with unacceptably high toxicity for safety. The default value is 0.95.
- `cutoff.eli.efficacy` The cutoff value for the futility rule, the acceptably lower efficacy. The default value is 0.30.
- `w1.toxicity` The weight for toxicity utility function (4) and (5). The recommended  $\rho$  is the target toxicity rate.
- `w2.toxicity` The weight for toxicity utility function (5).
- `indicator` The indicator cutoff value for utility function (5).

Using the same setting as above and by assuming a scenario in which the true toxicity rates are `p.true = (0.08, 0.20, 0.60, 0.7, 0.80)` and the true responses are `q.true = (0.25, 0.40, 0.45, 0.50, 0.52)`; we can use the function `get.oc.obd.kb()` to obtain the operating characteristics. Fig. 15 are outputs based on 1000 simulated trials.

The outputs show that dose 3 is recommended as the OBD by simulations for this scenario irrespective of which utility function is employed.

#### Select the OBD when the trial is completed

When a trial is completed, we can select the OBD based on the observed data using the function `select.obd.kb(...)`. This function needs the following arguments to input:

- `target.toxicity` The target DLT rate.
- `target.efficacy` The target efficacy rate.



```

> p.true <-c(0.08,0.20,0.60,0.7,0.80)
> q.true <- c(0.25,0.40,0.45,0.50,0.52)
> oc.obd.kb <- get.oc.obd.kb(toxicity.low = toxicity.low,
+                             toxicity.moderate= toxicity.moderate,
+                             toxicity.high = toxicity.high,
+                             efficacy.low = efficacy.low,
+                             efficacy.moderate = efficacy.moderate,
+                             efficacy.high = efficacy.high,
+                             target.toxicity=target.toxicity,
+                             target.efficacy= target.efficacy,
+                             p.true= p.true, q.true= q.true)
> summary.kb(oc.obd.kb)
selection percentage at each dose level (%) using utility function 1:
24.1 50.7 1.1 0.0 0.0
selection percentage at each dose level (%) using utility function 2:
26.2 48.3 1.3 0.1 0.0
selection percentage at each dose level (%) using utility function 3:
32.2 42.7 1.0 0.0 0.0
average number of patients treated at each dose level:
6.4 12.1 10.2 0.5 0.0
average number of toxicity observed at each dose level:
0.5 2.5 6.2 0.4 0.0
average number of efficacies observed at each dose level:
1.5 4.8 4.6 0.3 0.0
average number of toxicities: 9.5
average number of efficacies: 11.3
average number of patients: 29.2
percentage of early stopping due to toxicity: 2.9%
percentage of trial termination using utility function 1: 24.1%
percentage of trial termination using utility function 2: 24.1%
percentage of trial termination using utility function 3: 24.1%

```

Fig. 15. Empirical operating characteristics (OCs) of a phase I trial to find the optimal biological dose (OBD).

```

> target.toxicity=0.2
> target.efficacy=0.4
> npts <- c(3,6,12,3,3)
> ntox <- c(1,2,4,2,3)
> neff <- c(0,0,5,1,1)
> obd <- select.obd.kb (target.toxicity=target.toxicity,
+                       target.efficacy= target.efficacy, npts = npts,
+                       ntox = ntox, neff = neff)
> print(obd)
$name
[1] "select.obd.kb"

$obd1
[1] 3

$obd2
[1] 3

$obd3
[1] 3

```

Fig. 16. Selecting an OBD of a phase I trial to find the optimal biological dose (OBD).

```

> decision <- get.decision.obd2.kb(target.toxicity=0.2,
+                                 target.efficacy=0.4, cohortsize=3, ncohort=10,
+                                 decision=c("E", "E", "S", "S", "S", "S", "D", "D", "D"))
> decision$boundary.table

```

	T1	T2	EF1	EF2	Decision
[1,]	"0"	"0.16"	"0"	"0.27"	"E"
[2,]	"0"	"0.16"	"0.27"	"0.52"	"E"
[3,]	"0"	"0.16"	"0.52"	"1"	"S"
[4,]	"0.16"	"0.24"	"0"	"0.27"	"S"
[5,]	"0.16"	"0.24"	"0.27"	"0.52"	"S"
[6,]	"0.16"	"0.24"	"0.52"	"1"	"S"
[7,]	"0.24"	"1"	"0"	"0.27"	"D"
[8,]	"0.24"	"1"	"0.27"	"0.52"	"D"
[9,]	"0.24"	"1"	"0.52"	"1"	"D"

Fig. 17. A 3-by-3 "preset" table.

```

> decision$decision.matrix
      N T R Decision
1     3 0 0      EUE
2     3 0 1       E
3     3 0 2       S
4     3 0 3       S
5     3 1 0       S
6     3 1 1       S
7     3 1 2       S
8     3 1 3       S
9     3 2 0      DUT
10    3 2 1      DUT
11    3 2 2      DUT
12    3 2 3      DUT
13    3 3 0      DUT
14    3 3 1      DUT
15    3 3 2      DUT
16    3 3 3      DUT
17    6 0 0      EUE
18    6 0 1      EUE
19    6 0 2       E
20    6 0 3       E
21    6 0 4       S
22    6 0 5       S
23    6 0 6       S
24    6 1 0       S
25    6 1 1       S
26    6 1 2       S
27    6 1 3       S
28    6 1 4       S
29    6 1 5       S
30    6 1 6       S
31    6 2 0       S
32    6 2 1       S
33    6 2 2       S
34    6 2 3       S
35    6 2 4       S
36    6 2 5       S
37    6 2 6       S
38    6 3 0      DUT
39    6 3 1      DUT
40    6 3 2      DUT

```

Fig. 18. A part of decision lists.

- `npts` The vector containing the total number of patients treated at each dose level.
- `ntox` The vector containing the number of subjects at each dose level who have the toxicities.
- `neff` The vector containing the number of subjects at each dose level who have the responses.
- `p1` The cutoff lower limit for safety utility function (3):  $U = f(p) \times f(q)$ .  $f(p)$  is the function for toxicity probability, and  $f(q)$  is the function for efficacy probability.
- `p2` The cutoff upper limit for safety utility function (3).
- `q1` The cutoff lower limit for efficacy utility function (3).
- `q2` The cutoff upper limit for efficacy utility function (3).
- `cutoff.eli.toxicity` The cutoff value to eliminate a dose with unacceptable high toxicity for safety — the default value is 0.95.
- `cutoff.eli.efficacy` The cutoff value for the utility rule, the acceptable lowest efficacy — the default value is 0.30.

- `w1.toxicity` The weight for toxicity utility function (4) and (5) — the recommended  $p$  is the target toxicity rate as the default value.
- `w2.toxicity` The weight for toxicity utility function (5).
- `indicator` The indicator cutoff for utility function (5).

Assume that the number of patients treated at four doses is `npts = (3, 6, 12, 3, 3)`, the corresponding number of patients who experienced toxicity is `ntox = (1, 2, 4, 2, 3)`, and the corresponding number of patients who experienced efficacy is `neff = (0, 0, 5, 1, 1)`. From the Fig. 16 outputs, we can see the dose 3 is selected as the OBD based on these data irrespective of which utility functions being used.

#### A.4. Option 2: Alternative approach to partition the toxicity and efficacy intervals

We provide another function which can partition the toxicity and efficacy intervals without pre-specifying the parameters of `toxicity.low`, `toxicity.moderate`, ..., `efficacy.high` in the last section. We use the Bayesian optimal interval design (BOIN) as a tool to automatically partition the toxicity and efficacy intervals but with “clinical” rationales. To be specific, given the target toxicity rate, the BOIN algorithm will produce two cutoffs, which can be used to divide the toxicity interval into `toxicity.low`, `toxicity-acceptable`, `toxicity-high` sub-intervals. For example, if the target toxicity rate is 0.2, by executing the following code of using the function `get.boundary(.)` from the BOIN R package, the toxicity unit probability interval can be divided into three sub-intervals, (0, 0.16), (0.16, 0.24), and (0.24, 1), which correspond to the `toxicity.low`, `toxicity-acceptable`, `toxicity-high` sub-intervals.

```

R> > library(BOIN)
> bound <- get.boundary(target=0.2,
  ncohort=10, cohortsize=3)
> bound$lambda_e
[1] 0.1572423
> bound$lambda_d
[1] 0.2384624

```

In a similar vein, to generate the sub-intervals for the efficacy, we would firstly input the *target failure rate* to the above `get.boundary(.)` function from BOIN R package to get the two cutoffs. The *target failure rate* is defined as  $1 - \text{target efficacy rate}$ , that is, if the target efficacy rate is 0.4, then, the target failure rate is 0.6. Then, we can use find two cutoffs by using the target failure rate as the value for the target parameter in the `get.boundary(.)` function. Since the BOIN algorithm is for finding the MTD with assumption of the monotone increasing in terms of toxicity with the dose, we firstly reverse the two cutoffs and then subtract the two cutoffs by 1 to generate the sub-intervals in terms of the efficacy outcome. In literature, a similar strategy was adopted by Zohar et al. [29] in which the continual reassessment method (CRM) was adapted to find the minimal effective dose (MinED) in a phase II trial.

Here is an example. If the target efficacy rate is 0.4, by executing the following code, we get two original cutoffs of 0.48 and 0.73.

```

R> bound <- get.boundary(target=0.6,
  ncohort=10, cohortsize=3)
> bound$lambda_e
[1] 0.4791901
> bound$lambda_d
[1] 0.7314159

```

Then, the efficacy interval can be obtained as: (0, 1-0.73), (1-0.73, 1-0.48), (1-0.48, 1), that is, (0, 0.27), (0.27, 0.52), and (0.52, 1) correspond to the `efficacy.low`, `efficacy-acceptable`, `efficacy-high` sub-intervals.

From the above, we can generate the following sub-intervals for toxicity:

- The low interval for toxicity is (0, toxicity.lower.boundary).
- The acceptable interval for toxicity is (toxicity.lower.boundary, toxicity.upper.boundary).
- The high interval for toxicity is (toxicity.upper.boundary, 1).

and the sub-intervals for efficacy:

- The low interval for efficacy is (0, efficacy.lower.boundary).
- The acceptable interval for efficacy is (efficacy.lower.boundary, efficacy.upper.boundary).
- The high interval for efficacy is (efficacy.upper.boundary, 1).

If a “preset” table is given (an example shown below), we can then use the corresponding functions to design a trial.

		Efficacy.low (0,0.27)	Efficacy.moderate (0.27,0.52)	Efficacy.high (0.52,1)
Toxicity.low	(0,0.16)	E	E	S
Toxicity.moderate	(0.16,0.24)	S	S	S
Toxicity. high	(0.24,1)	D	D	D

For example, in the above “preset” table, the interval combination (0, 0.16) × (0.27,0.52) corresponds to a decision “S”.

We develop a function `get.decision.obd2.kb(.)`, which can automatically generate the sub-intervals based on the procedures we described above and provide the decision table.

For example, for a trial, there are 5 doses with the target toxicity rate of 0.2 and target efficacy rate of 0.4. Patients are assumed to be treated in cohorts size of 3 and the number of cohorts is 10. We can see from the following code that we are only needed to input `target.toxicity` and `target.efficacy` without requiring to input a set of parameters `toxicity.low`, `toxicity.moderate`, ..., `efficacy.high` if using the function `get.decision.obd.kb(.)` in the last section.

The 3-by-3 “preset” table can be obtained (see Fig. 17).

A part of the decision lists are shown as Fig. 18.

We also develop the function `get.oc.obd2.kb(.)` to conduct simulations to get the operating characteristics and the function `select.obd.kb(...)` to select the OBD in this context. Detailed examples can be found in Keyboard R package manual.

### References

[1] B.E. Storer, Design and analysis of phase I clinical trials, *Biometrics* 45 (3) (1989) 925–937.  
 [2] S.D. Durham, F.W.F. Rosenberger, A random walk rule for phase I clinical trials, *Biometrics* 53 (2) (1997) 745–760.  
 [3] A. Ivanova, A. Montazer-Haghighi, S.G. Mohanty, S.D. Durham, Improved up-and-down designs for phase I trials, *Stat. Med.* 22 (1) (2003) 69–82.  
 [4] M. Stylianou, D.A. Follmann, The accelerated biased coin up-and-down design in phase I trials, *J. Biopharm. Statist.* 14 (1) (2004) 249–260.  
 [5] J. O’Quigley, M. Pepe, L. Fisher, Continual reassessment method. a practical design for phase I clinical trials in cancer, *Biometrics* 46 (1) (1990) 33–48.

[6] Cheung Ying Kuen, Dose Finding By the Continual Reassessment Method, CRC Press, 2011.  
 [7] Y. Ji, P. Liu, Y. Li, B. Nebiyou Bekele, A modified toxicity probability interval method for dose-finding trials, *Clin. Trials* 7 (6) (2010) 653.  
 [8] S. Liu, Y. Yuan, Bayesian optimal interval designs for phase I clinical trials, *J. R. Stat. Soc. Series C (Appl. Stat.)* 64 (3) (2015) 507–523.  
 [9] F. Yan, S.J. Mandrekar, Yuan Y. Keyboard, A novel Bayesian toxicity probability interval design for phase I clinical trials, *Clin. Cancer Res.* 23 (15) (2017) 3994–4003.  
 [10] P.F. Thall, J.D. Cook, Dose-finding based on efficacy–toxicity trade-offs, *Biometrics* 60 (3) (2004).  
 [11] G. Yin, L.Y. Ji, Bayesian dose-finding in phase I/II clinical trials using toxicity and efficacy odds ratios, *Biometrics* 62 (3) (2006) 777–784.  
 [12] Zang Lee, J.J. Yuan, Adaptive designs for identifying optimal biological dose for molecularly targeted agents, *Clin. Trials J. Soc. Clin. Trials* 11 (3) (2014) 319–327.  
 [13] S. Liu, V.E. Johnson, A robust Bayesian dose-finding design for phase I/II clinical trials, *Biostatistics* 17 (2) (2016) 249–263.  
 [14] H. Sato, A. Hirakawa, C. Hamada, An adaptive dose-finding method using a change-point model for molecularly targeted agents in phase I trials, *Stat. Med.* 35 (23) (2016) 4093–4109.  
 [15] D.H. Li, J.B. Whitmore, W. Guo, Y. Ji, Toxicity and efficacy probability interval design for phase I adoptive cell therapy dose-finding clinical trials. *Clinical cancer research* ., *Off. J. Am. Assoc. Cancer Res.* 23 (1) (2017) 13–20.  
 [16] R. Lin, G. Yin, STEIN: A simple toxicity and efficacy interval design for seamless phase I/II clinical trials, *Stat. Med.* 36 (26) (2017) 4106–4120.  
 [17] R. Lin, Y. Zhou, F. Yan, D. Li, Y. Yuan, BOIN12. Bayesian optimal interval phase I/II trial design for utility-based dose finding in immunotherapy and targeted therapies, *JCO Prec. Oncol.* 4 (2020) 1393–1402.  
 [18] D.H. Li, J. Whitmore, Y. Ji, Abstract 423. Toxicity and efficacy probability intervals design for phase I dose-finding in oncology trials, *Cancer Res.* 76 (14 Supplement) (2016) 423.  
 [19] Y. Zhou, J.J. Lee, Y. Yuan, A utility-based Bayesian optimal interval (U-BOIN) phase I/II design to identify the optimal biological dose for targeted and immune therapies, *Stat. Med.* 38 (28) (2019) S5299–316.  
 [20] H. Shi, J. Cao, Y. Yuan, R. Lin, uTPI: a utility-based toxicity probability interval design for phase I/II dose-finding trials, *Stat. Med.* 40 (11) (2021) 2626–2649.  
 [21] G. Yin, Y. Yuan, Bayesian dose finding in oncology for drug combinations by copula regression, *J. R. Stat. Soc. Series C (Appl. Stat.)* 58 (2) (2009) 211–224.  
 [22] G. Yin, Y. Yuan, A latent contingency table approach to dose finding for combinations of two agents, *Biometrics* 65 (3) (2009) 866–875.  
 [23] N.A. Wages, M.R. Conaway, J. O’Quigley, Continual reassessment method for partial ordering, *Biometrics* 67 (4) (2011) 1555–1563.  
 [24] Y. Yuan, G. Yin, Sequential continual reassessment method for two-dimensional dose finding, *Stat. Med.* 27 (27) (2010) 5664–5678.  
 [25] H. Pan, R. Lin, Y. Zhou, Y. Yuan, Keyboard design for phase I drug-combination trials, *Contemp. Clin. Trials* 92 (2020) 105972.  
 [26] S. Liu, J. Ning, A Bayesian dose-finding design for drug combination trials with delayed toxicities, *Bayesian Anal.* 8 (3) (2013) 703–722.  
 [27] S. Razaee Zahra, Galen Wien-Cook, Mourad Tighiouart, A nonparametric Bayesian design for drug combination cancer trials, 2019, arXiv preprint arXiv: 1910.09163.  
 [28] P. Corbaux, M.E. Madani, M. Tod, J. Péron, B. You, Clinical efficacy of the optimal biological dose in early-phase trials of anti-cancer targeted therapies, *Eur. J. Cancer* 120 (2019) 40–46.  
 [29] S. Zohar, M. Resche-Rigon, S. Chevret, Using the continual reassessment method to estimate the minimum effective dose in phase II dose-finding studies. a case study, *Clin. Trials* 10 (3) (2013) 414–421.