

A. Jančařík; Tomáš Kepka  
Skupinové testování – oddělující systémy

*Pokroky matematiky, fyziky a astronomie*, Vol. 66 (2021), No. 4, 230–237

Persistent URL: <http://dml.cz/dmlcz/149294>

## Terms of use:

© Jednota českých matematiků a fyziků, 2021

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library*  
<http://dml.cz>

# Skupinové testování – oddělující systémy

*Antonín Jančařík, Tomáš Kepka*

*Abstrakt.* Otázkami spojenými s testováním vzorků se v souvislosti s pandemií covid-19 začala zabývat i širší veřejnost. Jednou z otázek, která byla v souvislosti s testováním diskutována, byla i otázka tzv. poolování. Cílem předkládaného článku je představit jeden z matematických nástrojů – oddělující systémy, který lze při spojování vzorků a jejich následném testování efektivně využít. Všechna odvození jsou realizována jen s využitím elementární matematiky tak, aby bylo možné dosažené výsledky nejen použít při vlastním testování, ale také jako příklad praktické aplikace ve výuce matematiky či informatiky.

## Úvod

V souvislosti s pandemií covid-19 se široká veřejnost začala zabývat otázkami spojenými s testováním vzorků. Součástí každodenního zpravodajství byly informace o počtu realizovaných testů, počtu pozitivních výskytů, počtu nemocných. Mnoho odborníků se zabývalo možnostmi, jak zvýšit kapacitu testování, zvýšit jeho spolehlivost i snížit cenu. Cílem tohoto článku je představit základy metody skupinového testování (combinatorial group testing), které stojí na pomezí mezi matematikou a informatikou a mohou být za jistých podmínek využity i pro zefektivnění testování v průběhu pandemie. Představení metod skupinového testování je jedním ze způsobů, jak propojit výuku informatiky a matematiky s reálným světem a ukázat praktickou uplatnitelnost některých „teoretických“ témat matematiky, jako je například zápis čísla v soustavách o jiném základu, při řešení situací z běžného života.

Obvykle využívaným nástrojem pro popis algoritmů pro skupinové testování je maticový zápis. Ten však může být někdy na překážku vysvětlení principů testování žákům nižších ročníků. Právě s ohledem na použití pokud možno jen nástrojů elementární matematiky byl zvolen jiný přístup, který využívá pouze množiny a s nimi spojenou terminologii. Oba přístupy jsou samozřejmě navzájem převoditelné. Autorům článku však není známo, že by problém byl již podobným způsobem přeformulován, nebo pojem oddělujícího systému ve stejném smyslu byl definován již jinde.

## 1. Skupinové testování

Skupinové testování (group testing) bylo poprvé představeno v roce 1943 jako nástroj pro ekonomicky výhodnější testování vzorků krve [3]. Cílem skupinového testování je pomocí co nejmenšího počtu testů odhalit vzorky, které se liší (jsou infikovány) od normálu. Při testování se předpokládá, že lze současně testovat velké množství vzorků tak, že přítomnost jednoho vadného (infikovaného) vzorku kontaminuje celou skupinu a ta

---

Doc. RNDr. ANTONÍN JANČAŘÍK, Ph.D., Katedra matematiky a didaktiky matematiky PedF UK, Magdalény Rettigové 4, 116 39 Praha 1, e-mail: [antonin.jancarik@pedf.cuni.cz](mailto:antonin.jancarik@pedf.cuni.cz),  
prof. RNDr. TOMÁŠ KEPKA, DrSc., Katedra algebry MFF UK, Sokolovská 83, 186 75 Praha 8, e-mail: [kepka@karlin.mff.cuni.cz](mailto:kepka@karlin.mff.cuni.cz)

je testem rozeznána. Cílem skupinového testování je výrazným způsobem snížit počet potřebných testů. V praxi jsou využívány jak metody, které používají testování všech vzorků ve stejném okamžiku, kterým se budeme dále věnovat, tak i metody, které pracují s více kroky [4]. Ukazuje se, že praktické nasazení matematických metod by mohlo velmi efektivně přispět ke zvýšení dostupnosti PCR testování a snížení jeho ceny [1]. Myšlenka skupinového testování, vycházející původně z potřeb zdravotnictví, se ukázala jako uplatnitelná v mnoha dalších oblastech, jako je genetika [5], informatika [2], či technika [7]. Diskuze o možném využití adaptivních a neadaptivních vícekrokových algoritmů byla v českém prostředí otevřena i v rámci tohoto časopisu [6]. V tomto textu si dovoluujeme navázat a představit základní matematický aparát využitelný pro testování, při kterém je možné všechny potřebné testy realizovat současně, bez nutnosti dalších návazných kroků.

## 2. Trochu matematiky

Účelně zavedeme vhodné matematické prostředky a značení.

Nechť  $M$  je konečná množina obsahující alespoň dva prvky. Počet prvků této množiny označíme  $m$  (tj.  $m = |M| \geq 2$ ). Jak je zvykem, množinu všech podmnožin množiny  $M$  označíme  $P(M)$  (tzv. potence množiny  $M$ ,  $|P(M)| = 2^m$ ). Označme ještě  $P^*(M)$  množinu všech vlastních neprázdných podmnožin množiny  $M$  (pak  $|P^*(M)| = 2^m - 2$ ).

Pro naše účely bude množina  $M$  představovat množinu vzorků, které chceme otestovat. Pokud testujeme tak, že několik vzorků spojíme (seskupíme, slijeme atd.) a vyhodnocujeme současně, nevyhodnocujeme testy jednotlivých vzorků – prvků množiny  $M$ , nýbrž podmnožin množiny  $M$ , tedy vyhodnocujeme jednotlivé prvky potence množiny  $M$ . Při konstrukci testu tak budeme vybírat, které prvky potence množiny  $M$  chceme testovat a které nikoli. Testem je tedy následně podmnožina  $P(M)$ , neboli prvek množiny  $P(P(M))$ , který budeme nadále nazývat systémem podmnožin množiny  $M$ .

Pro danou množinu  $M$  existuje  $2^{2^m}$  systémů podmnožin. Tento počet zcela vylučuje, aby i pro relativně malé množiny  $M$  bylo reálně možné procházet všechny systémy podmnožin množiny  $M$  a hledat takové, které jsou pro testování vhodné. V informatice hrají úlohy, jejichž řešení vyžaduje projití všech podmnožin (např. problém batohu – knapsack problem, viz [8]) významnou roli. Neexistence známého algoritmu, který by úlohu byl schopen řešit v polynomiálním čase, je velkou překážkou pro praktické řešení i pro relativně malé množiny. Procházení všech podmnožin množiny všech podmnožin pak představuje z hlediska praktického nasazení programátorskou noční můru. Pokud má množina  $M$  pouhých sto prvků, existuje 1 267 650 600 228 229 401 496 703 205 376 podmnožin této množiny. Počet systémů podmnožin množiny  $M$  pak představuje astronomické číslo, které má 381 600 857 690 147 056 244 358 827 361 míst.

Ve skutečnosti nás pro praktické řešení zajímají jen takové systémy podmnožin, které mají méně než  $m$  prvků (jsou lepší, než testování každého vzorku samostatně). I tak se může jednat o neúměrně velký počet. V dalším textu si ukážeme, jaké vlastnosti musí mít systém podmnožin, aby byl prakticky využitelný jako test, který detekuje určitý, předem daný, počet infikovaných vzorků. Proto zavedeme následující užitečné značení. Nechť  $\mathcal{A}$  je systém podmnožin množiny  $M$  (tj.  $\mathcal{A} \subset P(M)$ ). Je-li  $\mathcal{A} \neq \emptyset$ , pak

definujeme

$$\begin{aligned}\tilde{\mathcal{A}} &= \bigcap_{A \in \mathcal{A}} A, \\ \check{\mathcal{A}} &= \bigcup_{A \in \mathcal{A}} A, \\ \bar{\mathcal{A}} &= \{A', A \in \mathcal{A}\}.\end{aligned}$$

Všimněte si, že  $\bar{\bar{\mathcal{A}}} = \mathcal{A}$ ,  $\overline{P(M)} = P(M)$  a  $\overline{P^*(M)} = P^*(M)$ . Nelze zaměňovat systém doplňků  $\bar{\mathcal{A}}$  s doplňkovým systémem  $\mathcal{A}' = P(M) \setminus \mathcal{A}$ .

### 2.1. Oddělující systémy

Aby bylo možné nějaký systém podmnožin použít jako test pro vyhledávání infekčních, resp. vadných vzorků, musí tento systém mít jisté velmi specifické vlastnosti, které se pokusíme v dalším textu popsat. Nejprve zavedeme několik podmínek, které může neprázdný systém  $\mathcal{A}$  splňovat:

- (p1)  $\tilde{\mathcal{A}} = \emptyset$ ;
- (p2)  $|\tilde{\mathcal{A}}| \leq 1$ ;
- (p3)  $\check{\mathcal{A}} = M$ ;
- (p4)  $|M \setminus \check{\mathcal{A}}| \leq 1$ ;
- (p5)  $\mathcal{A}$  je podmnožinou  $P^*(M)$ ;
- (p6) je-li  $\mathcal{B}$  neprázdňá vlastní část systému  $\mathcal{A}$ , pak průnik  $\tilde{\mathcal{B}}$  a  $\tilde{\overline{(\mathcal{A} \setminus \mathcal{B})}}$  je nejvýše jednoprvkový (tj. je buď prázdný, nebo jednoprvkový);
- (p7)  $A \cap B = \emptyset$  pro všechna  $A, B \in \mathcal{A}$ ,  $A \neq B$ ;
- (p8)  $A \cup B = M$  pro všechna  $A, B \in \mathcal{A}$ ,  $A \neq B$ .

Je zjevné, že mezi jednotlivými podmínkami platí nejrůznější vzájemné vztahy, kterými se ale nyní nebudeme zabývat. Pouze na tomto místě konstatujeme, že podmínka (p1) je silnější než podmínka (p2) a podmínka (p3) je silnější než podmínka (p4). Pomocí uvedených podmínek zavedeme pojem, který bude pro další úvahy zásadním.

Nechť  $M$  je konečná množina,  $m = |M| \geq 2$ . Systém  $\mathcal{A}$  podmnožin množiny  $M$  nazveme oddělujícím, jestliže  $\mathcal{A}$  splňuje podmínky (p2), (p4) a (p6). Navíc o oddělujícím systému  $\mathcal{A}$  budeme říkat, že je:

- prvního druhu, jestliže jsou splněny podmínky (p1) a (p3);
- druhého druhu, jestliže je splněna podmínka (p3), nikoli však (p1);
- třetího druhu, jestliže je splněna podmínka (p1), nikoli však (p3);
- čtvrtého druhu, jestliže není splněna ani jedna z podmínek (p1) a (p3);
- řádný, jestliže splňuje podmínku (p5).

## 2.2. Použití oddělovacích systémů

Nechť  $\mathcal{A}$  je neprázdný systém neprázdných podmnožin množiny  $M$ ,  $m = |M| \geq 2$ . Pro každé  $a \in M$  definujeme  $R(a) = \{A \in \mathcal{A} : a \in A\}$  (tj. množinu všech podmnožin z  $\mathcal{A}$ , které obsahují prvek  $a$ ). Následují snadná pozorování:

- $R(a) \neq \emptyset$ , právě když  $a \in \bigcup \mathcal{A}$ .
- Je-li  $\bigcup \mathcal{A} = M$ , pak  $R(a) \neq \emptyset$  pro každé  $a \in M$ .
- $R(a) \neq \mathcal{A}$ , právě když  $a$  není prvkem  $\bigcap \mathcal{A}$ .
- Je-li  $\bigcap \mathcal{A} = \emptyset$ , pak  $R(a) \neq \mathcal{A}$  pro každé  $a \in M$ .
- $a \in \bigcap R(a)$  pro každé  $a \in M$ .
- Jestliže  $\mathcal{A}$  splňuje podmínku (p2) a jestliže  $a, b \in M$  jsou takové prvky, že  $R(a) = \mathcal{A} = R(b)$ , pak  $a = b$ .
- Jestliže  $\mathcal{A}$  splňuje (p4) a jestliže  $a, b \in M$  jsou takové prvky, že  $R(a) = \emptyset = R(b)$ , pak  $a = b$ .
- Jestliže  $\mathcal{A}$  splňuje (p6) a jestliže  $a, b \in M$  jsou takové prvky, že  $R(a) = R(b) \neq \emptyset$ , pak  $a = b$ .
- Jestliže  $\mathcal{A}$  je oddělovací systém, potom systémy  $R(a)$ ,  $a \in M$ , jsou po dvou různé.
- Jestliže  $\mathcal{A}$  je oddělovací systém prvního druhu, pak  $\emptyset, \mathcal{A} \neq R(a)$  a systémy  $R(a)$ ,  $a \in M$ , jsou po dvou různé.

Nyní již můžeme přistoupit k vlastnímu využití oddělovacích systémů pro testování. Předpokládejme, že máme množinu  $M$  vzorků, které jsou dvou druhů – vadné a bezvadné. Máme k použití test (či metodu), který odhalí, zdali v nějaké skupině (rozuměj podmnožině) vzorků je obsažen alespoň jeden vadný vzorek. Samozřejmě, pokud otestujeme každý vzorek samostatně, nalezneme snadno všechny vadné vzorky. Tento postup však vyžaduje  $m = |M|$  testů, což, zejména pro velké množství vzorků, může být nevýhodné či dokonce nepřijatelné. Při testování nám jde o to, abychom při co nejmenším množství testů zjistili o množině vadných vzorků co nejvíce, tedy pokud možno našli všechny vadné vzorky. Také nám z různých důvodů může jít o to, aby všechny testy proběhly (takřka) současně. Nelze tedy upravovat testované skupiny na základě částečných výsledků testování.

Ze vzorků lze utvářet skupiny, které pak vytvoří systém podmnožin množiny vzorků. Podívejme se nyní na to, co nám mohou přinést oddělovací systémy.

Nechť  $\mathcal{A}$  je řádný oddělovací systém prvního druhu (neprázdných, vlastních) podmnožin (podskupin) množiny vzorků  $M$ . Začneme tím, že otestujeme všechny skupiny systému  $\mathcal{A}$ . To činí  $|\mathcal{A}|$  testů. Co se může přihodit:

1. Žádná ze skupin neobsahuje vadné vzorky. Pak ale jsou všechny vzorky bezvadné.
2. Pokud alespoň jedna ze skupin vykazovala vadný vzorek, tak označíme  $G \neq \emptyset$  systém všech skupin, u kterých se při testování odhalila přítomnost vadného vzorku. Nyní rozlišíme dva případy, kdy buď  $G = R(z)$  pro nějaké  $z \in M$ , nebo  $G \neq R(z)$  pro všechna  $z \in M$ .

3. Nejprve se zabývejme případem, kdy pro všechna  $z \in M$  platí  $G \neq R(z)$ . V tomto případě jsou minimálně dva vzorky vadné a navíc musí platit, že pokud je  $z$  vadný vzorek, tak  $R(z)$  je podmnožinou  $G$ . Neplatí však obecně obrácená implikace. V současnosti jsou známy metody, jak konstruovat oddělující systémy, kde platí i opačná implikace, za předpokladu, že počet vadných vzorků nepřekročí předem dané číslo. Těmi se však nebudeme nyní zabývat.
4. Zbývá případ, kdy  $G = R(z)$  pro nějaké  $z \in M$ . Jestliže z jakéhokoli důvodu víme nebo předpokládáme, že pouze jeden vzorek je vadný, pak je to vzorek  $z$ . Nevíme-li to, pak by bylo možné se pomocí jediného dalšího testu přesvědčit, zda  $z$  je vadný vzorek či nikoli. Pokud  $z$  není vadný vzorek, existují alespoň dva vadné vzorky. Je-li  $z$  vadný vzorek, pak ovšem nevíme, zda existuje další vadný vzorek či nikoli. Zkusme tedy předpokládat, že existuje ještě jiný vadný vzorek  $x$ ; nutně platí, že  $R(x)$  je (vlastní) podmnožinou  $R(z)$ . Potom v systému  $\mathcal{A}$  existuje skupina, která obsahuje pouze prvek  $z$ , ale nikoli prvek  $x$ . Použijme tedy systém  $\bar{\mathcal{A}}$ . Je-li systém  $\mathcal{A}$  řádný oddělující systém prvního druhu, je i systém  $\bar{\mathcal{A}}$  řádný oddělující systém prvního druhu. Otestujme tedy systém  $\bar{\mathcal{A}}$  a označme  $\bar{G} \neq \emptyset$  systém všech skupin, u kterých se při testování systémem  $\bar{\mathcal{A}}$  odhalila přítomnost vadného vzorku. Mohou nastat dva případy:
- $\bar{G} = \bar{R}(x)$  a  $x$  je jediným vadným vzorkem,
  - $\bar{G} \neq \bar{R}(x)$  a v množině  $M$  existují minimálně dva vadné vzorky.

Shrňme uvedené úvahy o řádném oddělujícím systému prvního druhu  $\mathcal{A}$ . Pokud jsme přesvědčeni o existenci nejvýše jednoho vadného vzorku, pomocí současně provedených  $|\mathcal{A}|$  testů jsme schopni rozhodnout, zda některý ze vzorků je vadný a tento vzorek určit.

Pokud si nejsme jisti tím, že vadný vzorek je nejvýše jeden, otestujeme vzorky současně testy  $\mathcal{A}$  a  $\bar{\mathcal{A}}$ , tedy provedeme  $2|\mathcal{A}|$  měření. Na základě výsledků těchto testů jsme schopni nejen určit, zda systém obsahuje vadný vzorek, ale také zda je tento vadný vzorek právě jeden a tento vzorek identifikovat, anebo zda existuje vadných vzorků více.

Nyní si výše uvedené úvahy ukážeme na příkladu.

**Příklad 1.** Nechť  $M$  je množina obsahující  $2^n$  vzorků. Označme vzorky z této množiny celými čísly od 0 do  $2^n - 1$ . Systém  $\mathcal{A}$  sestavíme z  $n$  podmnožin množiny  $M$  tak, že podmnožina  $A_i$  obsahuje právě ty prvky množiny  $M$ , které mají v binárním zápise přiděleného čísla na pozici  $i$  číslici 1.

Nejprve ukážeme, že systém  $\mathcal{A}$  je řádný oddělující systém.

- $\bar{\cap} \mathcal{A}$  je jednoprvkový a je tvořen vzorkem, jehož číslo je v binární soustavě tvořeno samými jedničkami, tedy vzorkem, který je označen číslem  $2^n - 1$ . Systém  $\mathcal{A}$  tedy splňuje podmínku (p2).
- $\bar{\cup} \mathcal{A}$  obsahuje všechny vzorky, jejichž číselné označení v binárním zápise obsahuje alespoň jednu jedničku, tedy neobsahuje pouze jediný vzorek, a to ten, jehož číselné označení je 0. Systém  $\mathcal{A}$  tedy splňuje podmínku (p4).

- Je-li  $\mathcal{B}$  neprázdná vlastní část systému  $\mathcal{A}$ , pak průnik  $\tilde{\mathcal{B}} \cap \tilde{\mathcal{A}} \overline{(\mathcal{A} \setminus \mathcal{B})}$  obsahuje pouze jediný vzorek, a to ten, jehož číslo je v binární soustavě tvořeno samými jedničkami na pozicích odpovídajících prvkům  $\mathcal{A}$  a nulami na zbývajících pozicích. Systém  $\mathcal{A}$  tedy splňuje podmínku (p6).
- Je zjevné, že systém  $\mathcal{A}$  neobsahuje prázdnou ani celou množinu, proto splňuje podmínku (p5).

Systém  $\mathcal{A}$  je tedy řádný oddělující systém čtvrtého druhu.

K čemu lze takový systém použít? Pokud víme, že množina  $M$  obsahuje právě jeden vadný vzorek, můžeme jej pomocí systému  $\mathcal{A}$  nalézt. Testování pomocí systému  $\mathcal{A}$  díky testu  $A_i$  ověří, zda v číselném zápise vadného prvku je na pozici  $i$  číslice 1 (pokud je test  $A_i$  pozitivní), nebo číslice 0 (pokud je test  $A_i$  negativní).

**Příklad 2.** Testování pomocí systému  $\mathcal{A}$  z předchozího příkladu funguje díky tomu, že víme, že množina obsahuje právě jeden vadný vzorek. Co však v případě, kdy víme, že množina  $M$  obsahuje maximálně jeden vadný vzorek? V této situaci nelze systém  $\mathcal{A}$  použít, protože pokud všechny testy vyjdou negativní, nejsme schopni rozhodnout, zda množina  $M$  neobsahuje vadný vzorek, nebo je vadným vzorkem ten, co je označený číslem 0. Je to důsledek faktu, že  $\tilde{\mathcal{A}} \neq M$ .

Můžeme však velmi snadno vytvořit systém  $\dot{\mathcal{A}}$ , který se od systému  $\mathcal{A}$  liší pouze tím, že obsahuje ještě jeden test navíc. Test testující množinu obsahující jediný vzorek, a to vzorek označený právě číslem 0.

Je zjevné, že takto vytvořený systém  $\dot{\mathcal{A}}$  je řádný oddělující systém prvního druhu (splňuje podmínky (p1) a (p3)), a proto jej lze použít v situaci, kdy víme, že množina  $M$  obsahuje maximálně jeden vadný vzorek. Na základě aplikace systému  $\dot{\mathcal{A}}$  jsme schopni rozhodnout, zda množina  $M$  obsahuje vadný vzorek. A za použití výše uvedeného předpokladu, že množina  $M$  obsahuje nejvýše jeden vadný vzorek, můžeme také tento vadný vzorek jednoznačně určit.

**Příklad 3.** Pokud jsme v situaci, kdy očekáváme, že množina  $M$  bude obsahovat maximálně jeden vadný vzorek, ale může teoreticky obsahovat i více vadných vzorků, nejsme schopni pomocí systému  $\dot{\mathcal{A}}$  z předchozího příkladu ve většině případů (s výjimkou detekce právě jednoho vadného vzorku s číslem 0) určit, jestli se v případě zachycení jednoho vadného vzorku jedná o jediný vadný vzorek, nebo soubor obsahuje více vadných vzorků. Podle výše popsané teorie můžeme použít soubor  $\ddot{\mathcal{A}} = \dot{\mathcal{A}} \cup \bar{\dot{\mathcal{A}}}$ , pomocí kterého můžeme rozhodnout, zda soubor obsahuje žádný, jeden, nebo více vadných vzorků. Pokud soubor obsahuje právě jeden vadný vzorek, můžeme jej i jednoznačně určit.

Soubor  $\ddot{\mathcal{A}}$  pro každou pozici  $i$  obsahuje test  $A_i$ , kterým ověřujeme, zda vadný vzorek má na pozici  $i$  v binárním zápise svého číselného označení číslici 1, a test  $\bar{A}_i$  pomocí kterého ověřujeme, zda vadný vzorek má na pozici  $i$  v binárním zápise svého číselného označení číslici 0. Navíc soubor  $\ddot{\mathcal{A}}$  obsahuje test, kterým je testován jediný vzorek, a to ten, který je označen číslicí 0, a dále doplněk tohoto testu, kterým jsou současně testovány všechny vzorky s nenulovým označením. Je zjevné, že pro použití souboru  $\ddot{\mathcal{A}}$  nemají poslední dva testy žádný přínos, protože nám nepřinášejí nové informace a můžeme je tedy ze souboru vynechat. Pro otestování množiny  $M$  můžeme se stejným výsledkem použít soubor  $\ddot{\mathcal{A}} = \mathcal{A} \cup \bar{\mathcal{A}}$ , přičemž  $|\ddot{\mathcal{A}}| = 2n$ .

Navržený soubor  $\ddot{A}$  je schopen rozeznat, zda množina  $M$  obsahuje žádný, jeden, nebo více vadných vzorků. Pokud množina  $M$  obsahuje dva vadné vzorky, dokáže je systém  $\ddot{A}$  jednoznačně určit pouze v případě, že se jejich číselné označení v binární soustavě liší na jediné pozici. V ostatních případech není možné pomocí systému  $\ddot{A}$  určit, zda množina  $M$  obsahuje dva, nebo více vadných vzorků, ani je jednoznačně určit. Test však může přispět k částečné eliminaci, tedy určení podmnožiny množiny  $M$ , která nemůže obsahovat vadné vzorky. Tuto množinu tvoří takové vzorky, pro něž alespoň jeden test, do kterého byly zařazeny, vyšel negativní.

Je nutné ale mít na paměti, že i v situaci, kdy množina  $M$  obsahuje právě dva vadné vzorky, může nastat situace, kdy všechny testy vyjdou pozitivní. Tedy není možné uvedeným způsobem označit žádný vzorek jako negativní.

### 3. Závěr

Cílem tohoto článku bylo představit koncept skupinového vyhledávání a zavést základní pojem oddělovacího systému. Aplikace, která byla zmíněna v příkladech v závěru článku, vychází z teoretického předpokladu, že můžeme současně testovat libovolný počet vzorků a že daný soubor neobsahuje více než jeden vadný vzorek. Oba tyto předpoklady v praxi obvykle nejsou naplněny. Technické limity, především senzitivita testu v případě PCR testování, omezují počet současně testovaných vzorků na jednotky, maximálně desítky. Návrh testu pak musí vycházet z konkrétních technických omezení.

Druhé omezení, předpoklad nejvýše jednoho vadného vzorku, se při velmi nízké úrovni záchyťů (ve zlomcích promile) v plošném testování (např. ve školách), nemusí jevit jako nereálný. Přesto je samozřejmě lepší navrhnout testovací systém tak, aby byl schopen zachytit a přesně určit více než jeden vadný vzorek. Takové systémy jsou v současnosti známé. Je možné je vytvořit pro každý předem daný nejvyšší očekávaný počet vadných vzorků. Ne vždy je ale možné požadovaný systém vytvořit tak, aby jeho použitím došlo ke snížení počtu potřebných testů.

### L i t e r a t u r a

- [1] BARAK, N., et al.: *Lessons from applied large-scale pooling of 133,816 SARS-CoV-2 RT-PCR tests*. Science Translational Medicine 13 (2021), eabf2823, <https://doi.org/10.1126/scitranslmed.abf2823>.
- [2] DE BONIS, A., DI CRESCENZO, G.: *Combinatorial group testing for corruption localizing hashing*. In: B. Fu, D.-Z. Du: Computing and Combinatorics, 17th Annual International Conference COCOON 2011, Dallas, TX, USA, August 14–16, 2011, Proceedings, Springer, 2011, 579–591.
- [3] DORFMAN, R.: *The detection of defective members of large populations*. Ann. Math. Statist. 14 (1943), 436–440.
- [4] ELIAZ, Y., DANOVICH, M., GASIC, G. P.: *Poolkeh finds the optimal pooling strategy for a population-wide COVID-19 testing (Israel, UK, and US as test cases)*. MedRxiv, 2020, <https://doi.org/10.1101/2020.04.25.20079343>



- [5] FARACH, M., et al.: *Group testing problems with sequences in experimental molecular biology*. Proceedings, Compression and Complexity of SEQUENCES 1997, IEEE, 1997, 357–367.
- [6] GENEST, CH., ROUSSEAU, CH.: *Skupinový screening*. PMFA 66 (2021), 73–80.
- [7] GOODRICH, M. T., HIRSCHBERG, D. S.: *Improved adaptive group testing algorithms with applications to multiple access channels and dead sensor diagnosis*. J. Comb. Optim. 15 (2008), 95–121.
- [8] HABIBALLA, H.: *Základy teoretické informatiky*. Ostravská univerzita v Ostravě, 2017. Dostupné z: <http://hashim.sweb.cz/joomla/files/uvdoi.pdf>