ISSN 0002-306X. Изв. НАН РА и ГИУА. Сер. ТН. 2014. Т. LXVII, № 4.

ՀՏԴ 621.382.13

ԱՎՏՈՄԱՏԱՑՈՒՄ ԵՎ ԿԱՌԱՎԱՐՄԱՆ ՀԱՄԱԿԱՐԳԵՐ

Լ.Գ. ԿԱՐԱՊԵՏՅԱՆ

ՊԱՏԱՀԱԿԱՆ ԹԵՍՏԱՎՈՐՄԱՆ ՀԱՐՄԱՐՎՈՂԱԿԱՆ ԱԼԳՈՐԻԹՄ

Առաջարկվում է պատահական թեստավորման հարմարվողական մեթոդ, որը նախատեսված է ոչ կետային տեսակի ձախողման պատկերներ ունեցող մուտքային տիրույթներով ծրագրային միջոցների համար: Փորձարկման ժամանակ մուտքային տարածության մեջ թեսթ-դեպքերի համաչափ բաշխման շնորհիվ առաջին սխալի բացահայտման հավանականությունը մեծանում է ավելի քիչ թեստ-դեպքերի օգտագործումով, քանի որ սովորական պատահական փորձարկման համար օգտագործվում են ավելի շատ թեստ-դեպքեր:

Առանցքային բառեր. պատահական թեստավորում, թեստ-դեպքերի գեներացում, հավասարաչափ բաշխում, մուտքային տվյալների ձախողումների պատկեր:

Ներածություն: Պատահական թեստավորման (ՊԹ) ավտոմատացումը հիմնվում է մուտքային տիրույթում պատահականության որևէ սկզբունքով թեստ-դեպքերի հաջորդական ընտրության՝ գեներացման վրա [1]: Հավասարաչափ կամ բաշխման այլ օրենքի կիրառմամբ թեստ-դեպքերի գեներացումը սովորաբար համարվում է ոչ միայն պարզ, այլ նաև դիմելու ինտուիտիվ ձև [1]: Մուտքագրման տիրույթի տարրերը հայտնի են որպես ձախողում պատճառող տվյալներ, եթե դրանք ստեղծում են սխալ արդյունքներ:

ՊԹ-ի արդյունավետությունը [2], վիճակագրական գնահատականների արտածման ունակությունն ու հուսալիությունը կարող են զգալիորեն բարելավվել՝ թեստային դրվագների արդյունավետ բաշխում կիրառելու և քիչ փորձարկմամբ առաջին ձախողումը բացահայտելու պարագայում: Թեստային դրվագների բաշխման տարբեր մոտեցումներ են առաջադրում հարմարվողական պատահական թեստավորման (ՀՊԹ) մեթոդները: Առկա են հարմարվողական պատահական թեստավորման շատ խնդիրներ [3], որոնք վերաբերում են, օրինակ, տարբեր չափանիշներով հավասարաչափ տարածմանը, թեկնածու թեստ-դեպքերի հավաքածուներ կազմելու ուղիներին: [3]-ում ապացուցվում է, որ թեստավորման ռազմավարության արդյունավետությունը կախված է ոչ միայն ձախողման չափից, այլև դա առաջացնող տվյալների երկրաչափական պատկերից: Սա նշանակում է, որ պատահական թեստավորման իրականացումը կարող է բարելավվել, եթե հաշվի առնվի ձախողում պատճառող մուտքային տվյալների երկրաչափական պատկերը:

Ըստ [4]-ի պատահական փորձարկումների՝ երկու ամենատարածված արդյունավետ չափումներն են առնվազն մեկ ձախողում հայտնաբերելու հավանականությունը (P-չափման միջոց) և հայտնաբերված ձախողումների ակնկալվող շարքը (E-չափ-

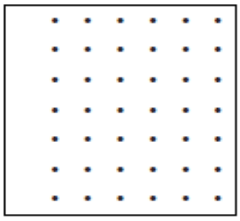
ման միջոց): Սակայն այդ միջոցները կատարյալ չեն, կան որոշ քննադատություններ դրանց վերաբերյալ՝ չնայած իրենց հանրաճանաչությանը: Այսինքն՝ երբեմն չկա E-չափման միջոցի կիրառության անհրաժեշտություն՝ ենթադրելով որոշ սխալներ, իսկ P-չափման միջոցը չի տարբերակում դեպքերը՝ ըստ խափանումների բացահայտման քանակի:

Ստորև P-չափման միջոցի ու E-չափման միջոցի փոխարեն առաջարկվում է արդյունավետության չափման մեկ այլ մեխանիզմ՝ առաջին ձախողումը բացահայտող դեպքերի անհրաժեշտ (ակնկալվող) քանակը՝ F-չափման միջոցը:

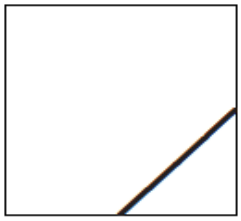
Եթե մուտքային D դոմենի չափը նշանակենք d-ով, իսկ m-ով և n-ով՝ համապատասխանաբար ձախողում պատճառող տվյալները և թեստերի ընդհանուր քանակը, ապա փորձարկման և խափանումների հաճախությունները կչափվեն համապատասխանաբար σ= n/d և θ=m/d բանաձևերով:

Դեպքերի փոխարինմամբ պատահական փորձարկման դեպքում F-չափման միջոցը կորոշվի որպես F=1/θ, կամ, որ համարժեք է, F=d/m: Այսինքն՝ F-չափման միջոցն արտացոլում է թեստավորման ռազմավարության արդյունավետությունը ավելի ուղղակիորեն և բնականորեն, քանի որ ստորին F-չափումը նշանակում է, որ ավելի քիչ թեսթ-դեպքեր են պահանջվում առաջին ձախողումը բացահայտելու համար: Գործնականում, երբ ձախողում է հայտնաբերվում, փորձարկումը սովորաբար կանգ է առնում, և սկսվում է կարգաբերումը: Թեստավորման փուլը սովորաբար վերսկսվում է միայն այն բանից հետո, երբ ամրագրվում է ձախողում: Հետևաբար, F-չափման միջոցը ոչ միայն ավելի ինտուիտիվ ձև է, այլև ավելի իրատեսական է գործնական տեսանկյունից:

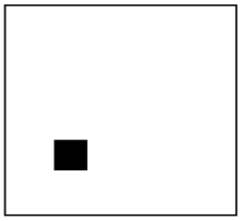
[3]-ում նշվում է, որ փորձարկման որոշ ռազմավարություններ բխում են ձախողում պատճառող մուտքային տվյալների երկրաչափական պատկերներից: Վերջինները դասակարգվում են ըստ երեք կատեգորիաների՝ կետային, շերտային և տեղամասային: Պարզության համար դիտարկվում են երկչափ մուտքային տիրույթների պատկերներ: Նկ.1-3-ում ցուցադրված են այդ պատկերները համապատասխանաբար կետային, շերտային և տեղամասային մոդելներով: Արտաքին սահմանները ներկայացնում են մուտքագրման տիրույթը, իսկ լրացված շրջանները մատնանշում են ձախողում պատճառող տվյալները՝ պատկերների կամ նախշերի տեսքով:



Նկ. 1. Կետային մոդել



Նկ. 2. Շերտային մոդել



Նկ. 3. Տեղամասային մոդել

Կետային օրինակում ձախողում պատճառող տվյալները միայնակ կետեր են, որոնք ձևավորում են շատ փոքր չափի շրջաններ՝ սփռված ամբողջ տիրույթով: Շերտի օրինակում ձախողում պատճառող տվյալները ձևավորում են նեղ շերտ [5], իսկ տեղամասային օրինակի հիմնական հատկանիշը ձախողում պատճառող տվյալների կենտրոնացումն է մեկ կամ մի քանի տարածաշրջաններում:

Դիտարկենք «կանոնավոր» երկրաչափություն ունեցող մուտքագրման մի D տիրույթ, որում հեշտ է գեներացնել ձախողում պատճառող տվյալներ՝ պատահական ձևով: Մասնավորապես, ենթադրենք D տիրույթը քառակուսի է, որի չափերն են՝ x-ը և y-ը, ընդ որում՝ $\{(x, y) \mid 0 \le x, y \le 10\}$: Ենթադրենք՝ սխալը ծրագրի $x+y>3$ պայմանական արտահայտության մեջ է, իսկ ճիշտ արտահայտությունը պետք է լինի $x+y >4$:

Այդ դեպքում ցանկացած ձախողում պատկանում է $\{(x, y) \mid 3 < x + y \le 4\}$ տեղամասին, որը D քառակուսային տիրույթում մեկ միավոր լայնությամբ շերտ է:

Պատահական փորձարկման կիրառումն այդ ծրագրում ենթադրում է այնպիսի կետերի գեներացում, որ թեստերի հաջորդականությունը լինի հնարավորինս ձգված, այսինքն՝ ընտրված հաջորդ թեստը գտնվի նախորդից առնվազն մեկ միավոր հեռավորության վրա:

Այսպիսով, առաջարկվում է պատահական փորձարկման ժամանակ նոր թեստի գեներացման դեպքում համոզվել, որ այն շատ մոտ չէ որևէ նախորդ թեստի: Նվազագույն F-ի հասնելու ճանապարհին է նախ՝ մի շարք պատահական թեստ-դեպքերի գեներացումը, ապա՝ դրանցից "լավագույն" մեկի ընտրությունը, որն ապահովում է դրանց հնարավորինս համաչափ տարածումը (սփռումը) ամբողջ մուտքային դաշտում:

Խնդրի դրվածքը և ալգորիթմի հիմնավորումը: Տրված են երկու՝ *գործարկված* $T = \{T_1, T_2, \cdot, T_N\}$ և *թեկնածու* $C = \{C_1, C_2, \cdot, C_K\}$ թեստ-դեպքերի հավաքածուններ, որոնք չեն հատվում $C \cap T = \varnothing$:

Գործարկված թեստ-դեպքը սահմանված է որպես ոչ մի ձախողում չբացահայտող արդեն կատարված թեստ, իսկ թեկնածուն սահմանված է որպես մուտքային տիրույթից պատահականորեն ընտրված որևէ թեստ:

Պահանջվում է գործարկված թեստ-դեպքերի հավաքածուն աստիճանաբար համալրել թեկնածու հավաքածուից ընտրված տարրով՝ մինչև ձախողման ի հայտ գալը: Թեկնածու հավաքածուի տարրը, որն ամենահեռուն է բոլոր կատարված թեստ-դեպքերից, ընտրվում է որպես հաջորդ թեստ-դեպք:

Հայտնի են հարմարվողական պատահական թեստավորման համար թեկնածու թեստ-դեպքերի հավաքածու գեներացնող և «ամենահեռուի» գաղափարն իրականացնող տարբեր ալգորիթմներ:

Առաջարկվող մեթոդը (թեկնածու թեստերի ընտրություն Հիլբերտի կորի կառուցմամբ՝ HT) հիմնավորելու համար այն համեմատության մեջ է դրված հարմարվողական պատահական թեստավորման (թեկնածու թեստերի ֆիքսված չափով՝

FSCST) ալգորիթմի [4] և սովորական պատահական թեստավորման ալգորիթմի (RT) հետ:

FSCST մեթոդի էությունը թեստ-դեպքերի հավասարաչափ տարածումն է գործարկված և մնացած թեստերի միջև հեռավորությունների մինիմումների մաքսիմումի չափանիշով:

Թեկնածու թեստերի բազմությունն ունի հաստատուն չափ (k), և յուրաքանչյուր անգամ, երբ ընտրվում է որևէ թեստ, ստեղծվում է թեկնածուների նոր հավաքածու: FSCST ալգորիթմը նկարագրում է, թե ինչպես է գեներացվում հարմարվողական պատահական փորձարկման այս տարբերակում թեկնածու թեստերի շարքը և ընտրվում փորձարկման թեստ-դեպքը:

Թեկնածու թեստերի C հավաքածուից ընտրվում է c_h տարրն այնպես, որ բոլոր j ∈ {1, 2, ·, k}-ի համար տեղի ունենա հետևյալ պայմանը՝

$$\min_{i=1}^{n} dist(c_h, t_i) \geq \min_{i=1}^{n} dist(c_j, t_i),$$

որտեղ *dist*-ը սահմանվում է որպես էվկլիդյան հեռավորություն, այսինքն՝ m-չափանի մուտքային տիրույթում երկու՝ $a=(a_1, a_2, \cdots, a_m)$ և $b=(b_1, b_2, \cdots, b_m)$ կետերի միջև հեռավորությունը որոշվում է հետևյալ բանաձևով՝

$$dist(a,b) = \sqrt{\sum_{i=1}^{m} (a_i - b_i)^2} :$$

Կատարված է նախնական ուսումնասիրություն՝ տեսնելու համար, թե ինչպես է թեկնածու հավաքածուի չափն ազդում հարմարվողական պատահական փորձարկման արդյունավետության վրա: Ընդհանուր առմամբ, որքան մեծ է k-ն՝ թեկնածու հավաքածուի չափը, այնքան փոքր թվով թեստ-դեպքեր են անհրաժեշտ առաջին ձախողումը բացահայտելու համար: Բացի այդ, k ≥ 10-ի դեպքում չկա առաջին ձախողումը բացահայտելու համար անհրաժեշտ թեստ-դեպքերի քանակի մեծ տարբերություն: Հետևաբար, հաստատված է k = 10 թիվը:

FSCS-ն ունի զգալիորեն ավելի փոքր F-չափ (ձախողում հայտնաբերելու համար կատարված թեստերի քանակ), քան RT-ն: Փորձնական արդյունքները ցույց են տալիս, որ FSCS-ն ունի RT-ին գերազանցելու լավ հնարավորություն (շատ զգալի չափով՝ ավելի քան 50%-ով): Մյուս կողմից, հաջորդ թեստ-դեպքն ընտրելիս FSCS-ին անհրաժեշտ են ավելի շատ ռեսուրսներ, քան պահանջում է RT-ն: Թող N-ը լինի առաջին ձախողում հայտնաբերելու համար կատարված թեստերի քանակը, իսկ K –ն՝ FSCS թեկնածու հավաքածուի չափը: Ընտրության բարդությունը KN^2 կարգի է:

Ալգորիթմ FSCST

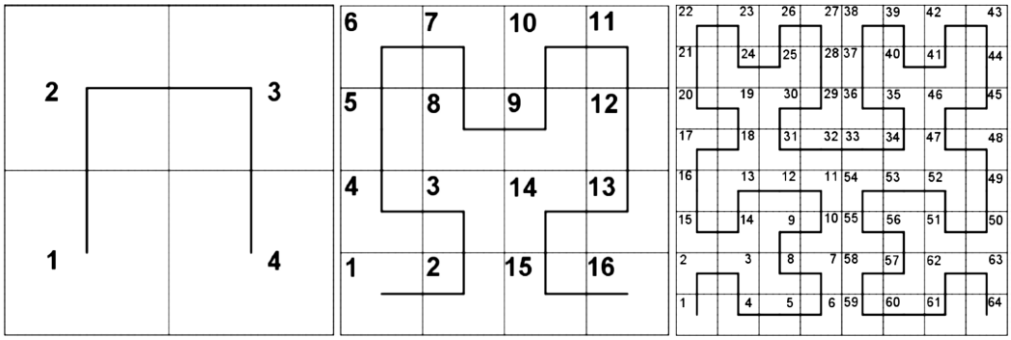
/*selected set := *{ test data already selected }*;
candidate set := *{}*;

```
total number of candidates := M*M; */
function Select The Best Test Data(selected set, candidate set,
                total number of candidates);
        best distance := –1.0;
        for i := 1 to total number of candidates do
                candidate := randomly generate one test data from the program
                        input domain, the test data cannot be in
                        candidate set nor in selected set;
                candidate set := candidate set + { candidate };
                min candidate distance := Max Integer;
                foreach j in selected set do
                        min candidate distance := Minimum(min candidate distance,
                                Euclidean Distance(j, candidate));
                end foreach
                if (best distance < min candidate distance) then
                        best data := candidate;
                        best distance := min candidate distance;
                end if
        end for
        return best data;
end function
```

HT-մեթոդի էությունը թեկնածու թեստերի հավաքածուի ձևավորումն է Հիլբերտի կորի դիսկրետ մոտարկման կառուցմամբ և այդ հավաքածուից հերթական «հեռու» թեստի ընտրությունը:

Մաթանալիզի տեսության համաձայն՝ երկչափ միավոր քառակուսու (կամ ավելի ընդհանրական՝ N-չափանի հիպերխորանարդի) մեջ սահմանափակված տարածությունը կարելի է համաչափ լցնել շարունակական կորով: Երկչափ կամ եռաչափ (կամ ավելի բարձրչափանի) շարունակական կորը ինտուիտիվ կարող է դիտվել որպես անընդհատ շարժվող կետ: Սակայն այն սահմանվում է նաև որպես անսահմանափակ իրական առանցքի անընդհատ կոր կամ միավոր (0, 1) բաց միջակայքի կոր: Առավել լավ ուսումնասիրված են երկչափ կամ եռաչափ էվկլիդյան տարածությունը շարունակական կորով լցնելու դեպքերը (ալգորիթմները): Տարածությունը համաչափ լցնող անընդհատ ֆրակտալը՝ Հիլբերտի կորը, հայտնաբերել է գերմանացի մաթեմատիկոս Դավիթ Հիլբերտը 1891թ., որպես Գաուս-Պիանոյի հայտնաբերած տիեզերական տարածությունները լցնող կորերի մի տարբերակ:

Հիլբերտի կորի դիսկրետ մոտարկումները շատ օգտակար են երկչափ և եռաչափ տարածությունները «հսկելու» համար (նկ. 4) և կարող են կիրառվել նաև պատահական թեստավորման դեպքում:



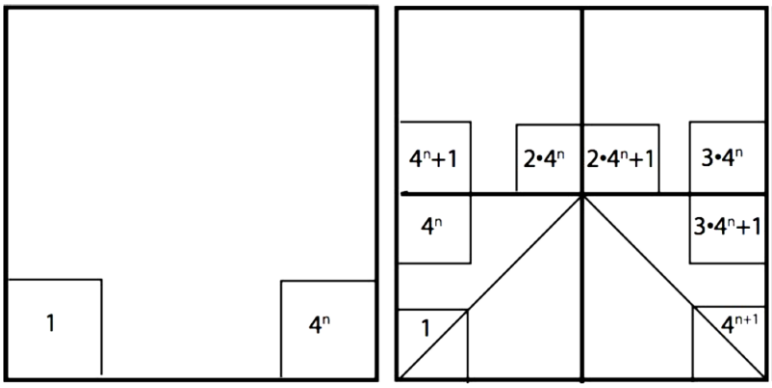
Նկ. 4. Հիլբերտի կորի երեք մակարդակները

Այս մեթոդով տարածությունը դիսկրետացվում է՝ յուրաքանչյուր չափը բաժանելով 2-ի աստիճան հանդիսացող m թվով բջիջների (բջջի չափը կախված է ձախողման տիրույթի ենթադրյալ չափերից), և սկսվում է կորի կառուցումը [m x m] չափերով տիրույթի P(0,0) բջջից մինչև P(m,m) բջիջը:

Կառուցելով մուտքային տիրույթում n-րդ աստիճանի կոր ($n = \log_2 m$), կարելի է այն դիտարկել որպես թեկնածու թեստերի վրայով անցնող ճանապարհ, որի կետերի քանակը 4^n է: Այսինքն՝ H_n կորի n-րդ դիսկրետ մոտարկման էվկլիդյան երկարությունը, կորի մակարդակից կախված, աճում է էքսպոնենցիալ օրենքով, միաժամանակ մնալով սահմանափակված վերջավոր տիրույթում:

Եթե P(x, y)-ը որոշում է որևէ թեստ, իսկ այդ թեստի դիրքը կորի վրա d-ն է, ապա d-ին մոտ հեռավորության վրա կգտնվի P(x, y)-ին մոտիկ որևէ այլ թեստ: Հակառակ պնդումը, սակայն, միշտ չէ ճշմարիտ: Նկ. 4-ում պարզ երևում է, որ կետերի բազմաթիվ զույգեր գտնվում են մեկ միավոր էվկլիդյան հեռավորության վրա, բայց կորի վրա գտնվում են միմյանցից շատ ավելի հեռու:

Պատահական հարմարվողական թեստավորման այս մեթոդում թեստի ընտրությունը (մինչև առաջին ձախողումը) կատարվում է Հիլբերտի կորի երկարությամբ (նկ. 5)՝ պահպանելով տեղաշարժի որոշակի հեռավորություն: n աստիճանի կորի դեպքում n իտերացիաներից հերթական i-րդը տեղի է ունենում i-րդ կետից $d=4^{n-i}$ քայլով, որպեսզի բացառվի «մոտակա-հեռու» թեստի փորձարկումը:



Նկ. 5. Հիլբերտի կորով փորձարկումների իտերացիաների քայլերը n և n+1 մակարդակներում

Այս ալգորիթմով փորձարկումներն իրականացնելիս վատագույն դեպքում կունենանք F- գնահատականի 4^n առավելագույն քանակ:

Ըստ այս մեթոդի՝ կիրառված են Հիլբերտի կորի կառուցման ռեկուրսիվ և կորի երկարությամբ իտերացիոն որոնումներ իրականացնող ալգորիթմները:

Հիլբերտի կորի կառուցման ընթացակարգը

```
/*Algorithm Hilbert-curve */
        procedure hilbert(x, y, xi, xj, yi, yj, n) /*n-current  level */
                if (n <= 0) then    LineTo(x  + (xi + yi)/2, y  + (xj + yj)/2");
                else
                  {
                  hilbert(x,          y,          yi/2, yj/2,  xi/2,  xj/2, n-1);
                  hilbert(x+xi/2,     y+xj/2 ,    xi/2, xj/2,  yi/2,  yj/2, n-1);
                  hilbert(x+xi/2+yi/2, y+xj/2+yj/2, xi/2, xj/2,  yi/2,  yj/2, n-1);
                  hilbert(x+xi/2+yi,  y+xj/2+yj,  -yi/2,-yj/2, -xi/2, -xj/2, n-1);   }
                end procedure
```

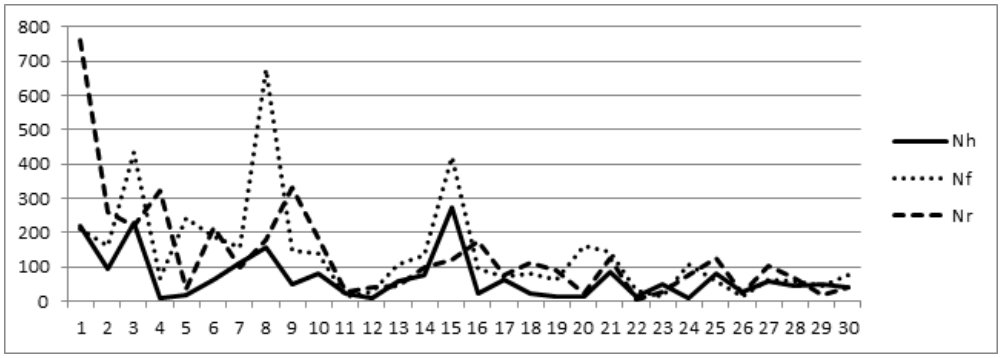
Իտերացիոն որոնումների ֆունկցիան

```
/*Algorithm recursiv search */

candidate set := {Hilbert curve};
total number of candidates := M*M;
Nf=0;
function Select The Best Test Data(candidate set, total number of candidates);
        level := n;

        for i := 1 to n do
                for j := i to total number of candidates step 4^(n-i)
                 do
```

```
            Nf++;
            candidate := candidate set[j];
            if (candidate is best test) then
            break; break;
            end if
        end for
        return Nf;
end function
```

Վերլուծական ուսումնասիրության (մեթոդների համեմատության) իրականացման համար պատահականորեն ընտրված են մի քանի մուտքային դոմեններ՝ ձախողման տիրույթների տարբեր դիրքերով և չափերով: Յուրաքանչյուր մուտքային դոմենի համար ստացվել են (N_h, N_f, N_r) թվերը, որոնք համապատասխանում են HT, FSCST և RT ալգորիթմներով փորձարկման դեպքերում առաջին ձախողումը բացահայտելու համար անհրաժեշտ փորձարկումների քանակներին: Արդյունքները, որոնք բերված են աղյուսակում և նկ.6-ում, ցույց են տալիս, որ միջին հաշվով $N_h<N_f<N_r$, այսինքն՝ առաջին ձախողումը բացահայտելու համար HT-ի համար պահանջվում են միջին հաշվով ավելի քիչ թեսթի-դեպքեր, քան FSCST-ի, առավել ևս՝ RT-ի դեպքում: Արտադրողականության բարելավման հաշվարկը կատարված է հետևյալ բանաձևերով՝

$$\frac{N_r - N_h}{N_r} \times 100\,,\ \frac{N_f - N_h}{N_f} \times 100:$$



Նկ. 6. Փորձնական արդյունքների համեմատության դիագրամը

Աղյուսակ

N_f, N_r, H_h արժեքների համեմատական ամփոփում

N	Ձախողման տիրույթի սկզբնակետի կոորդինատները		Ձախողման տիրույթի չափերը		N_h	N_f	N_r	N_h-ի բարելավման %-ը N_f-ի համեմատ	N_h-ի բարելավման %-ը N_r-ի համեմատ
	X	Y	W	H					
1	655	427	91	85	18	243	34	93%	47%
2	149	774	125	175	11	29	3	62%	-267%
3	748	14	118	153	20	79	112	75%	82%
4	634	282	80	174	76	134	100	43%	24%
5	72	141	63	51	92	163	259	44%	64%
6	709	24	122	103	20	12	27	-67%	26%
7	391	866	156	167	47	45	17	-4%	-176%
8	327	174	142	169	27	13	20	-108%	-35%
9	326	817	155	142	47	13	25	-262%	-88%
10	518	676	138	148	14	163	23	91%	39%
11	615	2	166	141	80	57	124	-40%	35%
12	271	796	125	148	11	63	87	83%	87%
13	666	874	176	78	59	107	44	45%	-34%
14	665	610	61	51	219	208	759	-5%	71%
15	436	230	57	171	111	157	97	29%	-14%
16	65	742	163	156	44	60	67	27%	34%
17	856	104	126	171	83	144	123	42%	33%
18	588	629	69	68	226	433	218	48%	-4%
19	127	546	167	163	38	75	39	49%	3%
20	293	244	87	85	7	67	323	90%	98%
21	101	132	174	92	23	95	175	76%	87%
22	802	444	130	114	271	419	119	35%	-128%
23	165	419	130	102	8	26	40	69%	80%
24	522	118	103	119	78	137	181	43%	57%
25	202	687	92	117	158	674	179	77%	12%
26	815	784	62	137	61	187	214	67%	71%
27	674	882	160	158	59	63	102	6%	42%
28	266	219	176	127	7	106	77	93%	91%
29	878	848	171	95	64	69	74	7%	14%
30	316	565	83	139	51	147	331	65%	85%

Եզրակացություն. Պատահական փորձարկումը հայեցակարգով պարզ է և հեշտ է իրականացնել: Բացի այդ, կարելի է անել հուսալիության և վիճակագրական հաշվարկներ: Հետևաբար, այն օգտագործվում է բազմաթիվ թեստավորողների կողմից: Քանի որ պատահական փորձարկման դեպքում չի օգտագործվում որևէ տեղեկություն գեներացված թեստ-դեպքերի մասին, այն չի կարող լինել թեստավորման հզոր մեթոդ, և դրա կատարումը կախված է միայն ձախողման ինտենսիվության չափերից:

Ոչ կետային ձախողման մոդելների համար պատահական հարմարվողական փորձարկումները՝ Հիլբերտի կորի երկարությամբ թեստ-դեպքերի համաչափ տա-

րածմամբ, խափանումների արագ հայտնաբերման և դրանց օրինաչափությունը գտնելու լավ հնարավորություններ են տալիս:

Այս գաղափարի օգտակարությունը հիմնավորվել է բազմաթիվ փորձարկումների միջոցով:

ԳՐԱԿԱՆՈՒԹՅԱՆ ՑԱՆԿ

1. **White, L.J.** Software testing and verification // Advances in Computers. - 1987. - 26. - P. 335–391.
2. **Hamlet, R.** Random testing. Encyclopedia of Software Engineering / Edited by Marciniak.- J. Wiley, 1994.
3. **Chan, F.T., Chen, T.Y., Mak, I.K., Yu, Y.T.** Proportional sampling strategy: guidelines for software testing practitioners // Information and Software Technology. - 1996. - 38. - P. 775–782.
4. **Chen, T.Y., Yu, Y.T.** On the relationship between partition and random testing // IEEE Transactions on Software Engineering. - 1994. - 20. – P. 977–980.
5. **Chen, T.Y., Eddy, G., Merkel, R., Wong, P.K.** Adaptive random testing through dynamic partitioning // Proceedings of the 4th International Conference on Quality Software (QSIC 04), IEEE Computer Society Press. - 2004.

Եվրոպական տարածաշրջանային ակադեմիա: Նյութը ներկայացվել է խմբագրություն 09.04.2014:

Л.Г. КАРАПЕТЯН

АЛГОРИТМ АДАПТИВНОГО СЛУЧАЙНОГО ТЕСТИРОВАНИЯ

Предлагается метод адаптивного случайного тестирования, который предназначен для тестирования программных средств с неточечным изображением отказов входной области. За счет равномерного распределения тестовых кейсов при тестировании во входном пространстве вероятность обнаружения первой ошибки увеличивается с меньшим использованием тестовых кейсов, так как для обычного случайного тестирования используется намного больше тестовых кейсов.

Ключевые слова: случайное тестирование, генерация тестовых кейсов, равномерное распределение, изображение неудачных входных данных.

L.G. KARAPETYAN

AN ALGORITHM FOR ADAPTIVE RANDOM TESTING

A method for adaptive random testing is proposed for testing softwares with non-point types of patterns of failure-causing inputs. At the expense of an even spread of test cases in the input space, the probability of finding the first error increases with little application of test cases, as for ordinary random testing, many more test cases are used.

Keywords: random testing, developing test cases, even distribution, failure input patterns.