

Validacija kao bitan element provere kvaliteta softvera

Studija slučaja

Tanja Krunić

Visoka tehnička škola strukovnih studija u Novom Sadu

Novi Sad, Srbija

krunic@vtsns.edu.rs

Sažetak—U okviru ovog rada je prikazana validacija softvera za proračun zapremine tečnosti u položenim cilindričnim rezervoarima u proizvoljnoj tački po y-osi. Za potrebe validacije softvera urađena su dodatna merenja i primenom metode kubnih splajnova je sastavljena interpolacija funkcije zapremine rezervoara u zavisnosti od tačke visine tečnosti. Razlika dobijenih rezultata i rezultata originalnog softvera je u unapred zadatim granicama, što potvrđuje da ispitivani softver daje dobre rezultate.

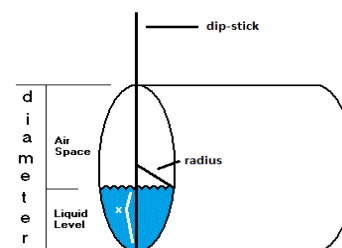
Ključne reči - validacija softvera, black box testing model, numerička analiza, kubni splajnovi, wolfram mathematica

I. UVOD

Validacija softvera je jedan od glavnih aspekata provere kvaliteta softvera. U suštini, to je proces kojim se proverava da li je softver isprogramiran tako da daje očekivane rezultate, [1]. Potpuno je jasno da bi šteta od softvera koji ne daje tačne rezultate bila ogromna. Prema [2], ekonomija Sjedinjenih Američkih država godišnje trpi štetu od 59.5 miliona dolara zbog neispravnih softvera. Iz ovog razloga je od izuzetne važnosti da se pre puštanja u rad softvera potvrdi njegova ispravnost.

U ovom radu je prikazan postupak validacije softvera koji je kreiran da bi proračunavao zapreminu tečnosti u horizontalnim cilindričnim rezervoarima proizvoljnih dimenzija u zavisnosti od visine tečnosti. Navedeni softver ispisuje tablice zapremine u litrama u zavisnosti od visine u centimetrima. Metoda proračuna zapremine koja se koristi je tzv. volumetrijska metoda koja podrazumeva pražnjenje rezervoara koji se potom puni određenom količinom vode, nakon čega se meri visina tečnosti. Na osnovu određenog broja dopunjavanja vode i merenja se numeričkom metodom proračunava zapremina tečnosti u zavisnosti od visine. Merenje visine tečnosti u rezervoarima se vrši mernim letvama (Sl. 1) čije merne jedinice su u milimetrima. Potom se iz tablice koja je programski generisana može pročitati odgovarajuća zapremina. Zapremina kao funkcija visine tečnosti se može računati geometrijskim i numeričkim putem. U ovom radu će biti reči o numeričkim metodama, dok se o geometrijskoj metodi više detalja može naći u [3], [4].

Originalan softver čiji je postupak validacije prikazan u ovom radu je kreiran u Microsoft Exel-u, i u njegovoj osnovi je numerička interpolacija funkcije zapremine u zavisnosti od nivoa tečnosti korišćenjem Lagranžovog interpolacionog polinoma. Za potrebe validacije ovog softvera izvršena je interpolacija funkcije zapremine rezervoara u zavisnosti od visine tečnosti na osnovu dodatno izvršenih testnih merenja. Pri tome je korišćena metoda interpolacije kubnim splajnovima. Ova metoda je opisana u poglavlju 2. U poglavlju 3 je opisan postupak validacije softvera. Pri validaciji softvera korišćen je programski paket *Wolfram Mathematica* [5], koji sadrži brojne ugrađene funkcije - između ostalog i metodu interpolacije kubnim splajnovima. Na kraju, poređenjem rezultata dobijamo da se izlazni podaci dva različita programa u datim tačkama razlikuju za manje nego 0.5%, što je savim zadovoljavajuće.



Slika 1. Merenje visine tečnosti u rezervoaru mernom letvom

II. INTERPOLACIJA FUNKCIJE METODOM KUBNIH SPLAJNOVA

Jedan od efektivnih metoda interpolacije funkcija je metoda splajnova. U nastavku je dat kratak opis metode splajnova, [6].

Neka je dat skup tačaka mreže $\{x_0, x_1, \dots, x_n\}$, $x_0 < x_1 < \dots < x_n$.

Neka su date redom vrednosti funkcije y_0, y_1, \dots, y_n u tačkama mreže.

Funkcija

$$S_p(x) = \sum_{i=0}^p s_{i,k}(x - x_k)^i, \quad x \in [x_k, x_{k+1}],$$

gde su koeficijenti

$$s_{i,k} \in R, i = 0, 1, \dots, p, k = 0, 1, \dots, n,$$

se naziva splajn stepena p sa čvorovima iz mreže $\{x_0, x_1, \dots, x_n\}$.

Splajn $S_p(x)$ se naziva interpolacioni splajn funkcije $f(x)$ na mreži x_0, x_1, \dots, x_n , $x_0 < x_1 < \dots < x_n$ u koliko važi

$$S_p(x_k) = f(x_k), \quad k = 0, 1, \dots, n.$$

Specijalno, za $p=3$ imamo kubni splajn koji se veoma često koristi u interpolaciji funkcija. Tada kažemo da je funkcija $S(x)$ kubni splajn ako postoji polinom trećeg stepena

$$S_k(x) = s_{k,0} + s_{k,1}(x - x_k) + s_{k,2}(x - x_k)^2 + s_{k,3}(x - x_k)^3,$$

$k=0, 1, \dots, n$ takvih da važi

(i) za svako $k=0, 1, \dots, n-1$,

$$S(x) = S_k(x), \quad x \in [x_k, x_{k+1}],$$

(ii) splajn prolazi kroz zadate čvorove, tj.

$$S(x_k) = y_k, \quad k = 0, 1, \dots, n,$$

(iii) splajn je neprekidna funkcija, tj.

$$S_k(x_{k+1}) = S_{k+1}(x_{k+1}), \quad k = 0, 1, \dots, n-1,$$

(iv) splajn je glatka funkcija, tj.

$$S'_k(x_{k+1}) = S'_{k+1}(x_{k+1}), \quad k = 0, 1, \dots, n-1,$$

(v) drugi izvod splajna je neprekidna funkcija, tj.

$$S''_k(x_{k+1}) = S''_{k+1}(x_{k+1}), \quad k = 0, 1, \dots, n-1.$$

Pri tome, polinom trećeg stepena $S_k(x)$ ima četiri nepoznata koeficijenta, $s_{k,0}, s_{k,1}, s_{k,2}, s_{k,3}$, što znači da je za određivanje splajna potrebno odrediti $4n$ koeficijenata. Iz uslova (ii) dobija se $n+1$ jednačina, a svaki od uslova (iii)-(v) određuje još $n-1$ jednačinu, što čini ukupno $4n-2$ jednačina. Drugim rečima, neophodno je rešiti sistem od $4n-2$ jednačine i $4n$ nepoznate. Da bi sistem bio određen dodaju se konturni uslovi $S''(x_0) = S''(x_n) = 0$.

Originalan softver čija validacija je vršena je napisan u Exel-u, i bazira se takođe na interpolaciji funkcije zapremine, ali u ovom slučaju primenom Lagranžovog interpolacionog polinoma, koji je oblika $L_n(x) = \sum_{i=0}^n y_i \prod_{j=0, j \neq i}^n \frac{x-x_j}{x_i-x_j}$, [7]. Validacija je izvršena u programskom paketu *Wolfram Mathematica*. Ovaj program je izabran zbog velikog broja ugrađenih matematičkih funkcija. U njemu je dostupna već gotova funkcija interpolacije kubnim splajnovima. S obzirom na složenost gore opisanog problema, jasno je da ovakav pristup znatno skraćuje vreme i trošak postupka validacije. Drugim

rečima, za potrebe validacije nije neophodno napisati komplikovan programski kod. Još jedna od prednosti primene navedenog programskog paketa je postojanje obimne dokumentacije, [8].

III. POSTUPAK VALIDACIJE SOFTVERA ZA PRORAČUN ZAPREMINE TEČNOSTI U ZAVISNOSTI OD NIVOA TEČNOSTI

Kao što je već napomenuto, softver za ispisivanje tablice zapremine tečnosti u zavisnosti od nivoa tečnosti je napisan u *Microsoft Exel*-u. Na Sl. 2 je prikazana tablica zapremine originalnog softvera.

Napomenimo najpre da je pri testiranju softvera korišćen tzv. model crne kutije (engl. *Black box testing model*) koji podrazumeva da tester nije upućen u sadržaj i strukturu koda programa, već testira njegovu funkcionalnost i upoređuje dobijene rezultate sa očekivanim rezultatima, [9]. U ovom

ЦЕНТИ-МЕТРИ	0	1	2	3	4	5	6	7	8	9
0										
10	290	334	380	428	478	528	580	634	689	746
20	804	864	925	987	1051	1116	1182	1249	1317	1386
30	1456	1527	1599	1672	1746	1820	1895	1971	2048	2125
40	2203	2282	2361	2441	2522	2603	2685	2767	2850	2933
50	3017	3101	3186	3271	3342	3442	3528	3615	3702	3789
60	3877	3965	4053	4141	4230	4319	4408	4497	4586	4675
70	4765	4855	4944	5034	5124	5214	5304	5394	5484	5574
80	5664	5753	5843	5932	6021	6110	6198	6287	6374	6462
90	6549	6635	6722	6808	6893	6978	7063	7147	7230	7314
100	7396	7478	7559	7639	7719	7799	7878	7956	8034	8112
110	8189	8267	8344	8420	8495	8570	8644	8718	8790	8861
120	8932	9001	9069	9136	9201	9265	9326	9386	9445	9501
130	9556	9610	9662	9712	9760	9807				

Slika 2. Tablica originalnog softvera - funkcija zapremine u zavisnosti od nivoa tečnosti

slučaju kod originalnog softvera nije bio poznat. Jedino što je bilo poznato jeste, kao što je već rečeno, da je program napisan na osnovu određenog broja merenja visine tečnosti i zapremine, te da je potom izvršena numerička interpolacija pomoću Lagranžovog polinoma. Proces izvođenja testa je primenjen na osnovu principa poznate strategije testa, [10]: Na osnovu ulaznih podataka programa, vrši se posmatranje izlaznih podataka prema osmišljenoj testnoj strategiji. Izlazni podaci se potom porede sa očekivanim rezultatima. Razlika između posmatranih i očekivanih rezultata mora zadovoljiti unapred zadati kriterijum. U našem slučaju, ulazni podaci koji se posmatraju predstavljaju izlazne vrednosti testiranog programa, tj. podatke iz izlazne tablice. Strategija testa je podrazumevala da se na osnovu izvršenih testnih merenja sastavi splajn interpolacija zavisnosti zapremine tečnosti u funkciji visine. Iz ove interpolacije izdvojeni su parovi tipa {visina(cm), zapremina (l)}. Ovi rezultati su potom upoređeni sa podacima iz tablice originalnog softvera. Prema strategiji testiranja, program je uspešno prošao test ukoliko njegovi rezultati (izlazni podaci) zadovolje neki unapred zadati kriterijum. U našem slučaju, kriterijum je podrazumevao da

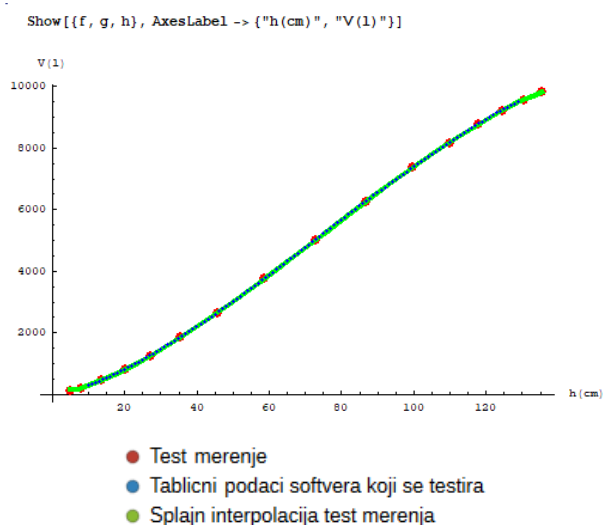
izmerenih nivoa tečnosti u *mm* koje su potom automatski preko petlje pretvorene u *cm*. Ispod vidimo parove tačaka testnih merenja, petlju koja vrednosti izmerenih visina pretvara u centimetre, kao i primenu ugrađene funkcije kubne splajn interpolacije.

```
pts={{50,126.59},{80,201.46},{135,452.79},{200,804.15},{271,125
5.64},{355,1857.56},{457,2659.73},{587,3762.34},{728,5015.35},{
868,6267.88},{997,7370.39},{1098,8172.68},{1178,8774.58},{1244
,9226.03},{1304,9577.35},{1355,9828.88}}
```

```
Do[pts[[i,1]]=pts[[i,1]]/10/N,{i,Length[pts]}
```

```
<<NumericalMath`SplineFit`
sp = SplineFit[pts, Cubic]
```

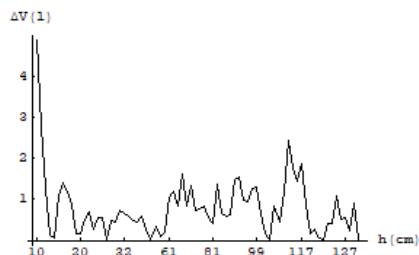
Na Sl. 7 vidimo grafike testnog merenja, njegove interpolacije kubnim splajnom, kao i izlazne podatke iz softvera koji se testira. Naizgled, imamo jako dobro poklapanje, što je ipak trebalo proveriti numerički računanjem razlike podataka iz tablice testiranog softvera i funkcije splajn interpolacije testnog merenja u istim tačkama.



Slika 7. Grafik testnog merenja, njegove interpolacije i tabličnih podataka izvornog softvera

Na Sl. 8 vidimo grafik razlike izlaznih vrednosti testnog i testiranog programa u svim tačkama posmatranog intervala. Kao što vidimo, razlika u litrama ne prelazi 5, čime smo dokazali da je unapred postavljeni kriterijum za validnost testiranog softvera zadovoljen. Razlika u zapremini u raznim tačkama visine se kretala od 0.013 do 4.863, dok je prosečna razlika oko 0.8. Najveća razlika u zapremini, tek nešto malo ispod granice dozvoljenog je izmerena na početku intervala, tj. pri veoma maloj vrednosti izmerene visine tečnosti od 5 cm. Ovo je sasvim logično zbog toga što postoji mogućnost deformisanja dna praznog rezervoara.

```
ListPlot[razlika, AxesLabel -> {"h (cm)", "ΔV (l)"},
PlotJoined -> True, PlotRange -> All, Ticks -> {tacke, Automatic}]
```



Slika 8. Grafik razlike izlaznih vrednosti testiranog i testnog programa

IV. POREĐENJE SA DRUGIM METODAMA VALIDACIJE

U [10] je predstavljen softver koji koristi geometrijsku metodu koja se bazira na proračunu zapremine rezervoara na osnovu spoljnih dimenzija uz primenu traka za merenje obima i ultrazvučnog uređaja za merenje debljine zidova rezervoara. U ovom obliku, geometrijsku metodu je moguće primeniti isključivo kod nadzemnih rezervoara, dok je kod podzemnih rezervoara situacija znatno komplikovanija. Rezervoar mora da se isprazni, degazira i očisti od mulja i taloga, kako bi se omogućio silazak radnika u unutrašnjost rezervoara da bi izmerio njegove dimenzije po sve tri koordinatne ose. Drugim rečima, ova metoda kod podzemnih rezervoara postaje skupa i potencijalno opasna po zdravlje radnika. U [12] i [13] je prikazan softver koji je zasnovan na laserskoj metodi pri čemu se parametri rezervoara mere korišćenjem laserskog uređaja u velikom broju tačaka. U [13] se ističe da ova metoda za razliku od volumetrijske metode nema troškova koji se odnose na količinu vode koja se sipa u rezervoar, organizaciju cisterne za prevoz, kao i odlaganje zaprljane vode. S druge strane, neophodna je nabavka skupog laserskog uređaja. U [11] se za izradu odgovarajućeg softvera koristi *Microsoft Excel*, u [12] se ne spominje softverski alat koji je korišćen, dok je u [13] odgovarajući softver kreiran u *MATLAB*-u.

Prema [10], validacija softvera za proračun zapremine tečnosti u horizontalnim cilindričnim rezervoarima u zavisnosti od visine tečnosti može da se izvrši poređenjem rezultata dobijenih drugim poznatim metodama. Generalno, u praksi se kod validacije ove vrste softvera javlja problem dostupnosti softvera koji je kreiran primenom druge metode, jer softveri ove vrste nisu javno dostupni. Stoga je poželjno kreirati sopstveni alternativni program čiji se rezultati mogu porediti sa rezultatima softvera koji testiramo. U ovom radu smo prikazali validaciju softvera koji je kreiran u *Microsoft Excelu* volumetrijskom metodom uz primenu Lagražovog interpolacionog polinoma. Validacija je izvršena poređenjem rezultata koji se dobijaju takođe volumetrijskom metodom, ali u ovom slučaju primenom metodom splajnova u programu *Wolfram Mathematica*. Jedna od prednosti primene ove numeričke metode u validaciji softvera je i činjenica da je metoda splajnova podesnija za numeričku aproksimaciju funkcije na osnovu podataka dobijenih eksperimentalnim putem, jer se kod Lagranžovih polinoma reda većeg od deset pojavljuju oscilacije, dok metoda splajnova daje glatke funkcije, [14]. Pored toga, prednost je i u izboru softverskog alata, tj. primeni programa *Wolfram Mathematica* koji

poseduje gotove procedure za primenu interpolacije metodom splajnova, pa se relativno brzo može napisati alternativni program za testiranje. Testiranje je vršeno po principu crne kutije, tj. autor test aplikacije nije imao uvid u algoritam i kod autora softvera čija validacija se tražila, čime je obezbeđena objektivnost validacije. Još jedna od prednosti ove metode je što nije neophodno posedovati laserske uređaje čija je cena relativno visoka. S druge strane, postoje troškovi odlaganja zaprljane vode, što svakako predstavlja nedostatak primene volumetrijske metode.

V. ZAKLJUČAK

Da bi se izbegla finansijska šteta zbog neispravnosti softvera, njegovo testiranje pre puštanja u rad je od velikog značaja. U ovom radu je prikazan slučaj testiranja softvera namenjenog proračunu zapremine cilindričnog rezervoara u zavisnosti od izmerenog nivoa visine tečnosti. U procesu validacije je primenjen metod *black box testing model* kada osoba koja vrši testiranje softvera nije upoznata sa kodom programa. U cilju testiranja ispravnosti rada softvera, izvršena su dodatna test merenja i na osnovu njih je napisan test program čiji rezultati su poređeni sa rezultatima originalnog programa. Razlika u izlaznim podacima je zadovoljila unapred zadati kriterijum, što je potvrdilo validnost programa. Testirani program je u upotrebi od kraja 2015. godine, i do danas nisu prijavljene primedbe na njegov rad.

LITERATURA

- [1] *Verification and validation in Software quality management*, <https://www.simplilearn.com/software-quality-management-verification-and-validation-rar100-article>, 2017
- [2] M., Newman, *Software errors cost US economy 59.5 billion annually. NIST assesses technical needs for industry to improve software testing*, 2002, http://www.abeacha.com/NIST_press_release_bugs_cost.htm
- [3] K. Mealy, *Cylindrical tank problem*, <https://www.pleacher.com/mp/probweek/p2016/a0516c.html>, 2016
- [4] R. Allain, *Cylindrical tank problem*, <https://www.wired.com/2010/11/car-talk-cylindrical-fuel-tank-problem/>, 2010

- [5] *Wolfram Mathematica*, <https://www.wolfram.com/mathematica/>
- [6] D. Herceg., N., Krejić, *Numerička analiza*, Prirodno-matematički fakultet Novi Sad, 1997
- [7] J. Lambers, *Lagrange interpolations, Lecture 5 Notes*, <http://www.math.usm.edu/lambers/mat772/fall10/lecture5.pdf>
- [8] *Wolfram Mathematica Documentation*, <https://reference.wolfram.com/language/?source=nav>
- [9] *Software testing fundamentals*, <http://softwaretestingfundamentals.com/black-box-testing/>
- [10] T. Galinac Grbac, *Testiranje softverskog proizvoda*, http://www.riteh.uniri.hr/zav_katd_sluz/zr/nastava/proginz/materijali/Testiranje%20programskog%20proizvoda.pdf
- [11] O.O. Agboola, P. P. Ikubanni, R. A. Ibikunle, A. A. Adediran, B. T. Ogunsemi, *Generation of Calibration Charts for Horizontal Petroleum Storage Tanks Using Microsoft Excel*, MAPAN Journal of Meterology Society of India, ISSN: 0970-3950, Volume 32, Issue 4, pp. 321–327, 2017
- [12] Ming-Sen Hu, Chia-Rei Ta, *Calculation of oil tank volume and report generation system with trim and list corrections*, Transactions of the Canadian Society for Mechanical Engineering, Vol. 40, No. 5, pp. 835-845, 2016
- [13] V. Knyva, M. Knyva, *New Method for Calibration of Horizontal Fuel Tanks*, ELEKTRONIKA IR ELEKTROTEHNIKA, ISSN 1392-1215, VOL. 18, NO. 9, pp. 91-94, 2012
- [14] J. Betten, *Comparison between LAGRANGE and Spline Interpolation*, Mathematical Models in Materials Science and Continuum Mechanics, file:///C:/Users/bs/AppData/Local/Temp/comparison2-1.pdf

ABSTRACT

Herein we show the validation of a software for calculation the liquid volume in a horizontal cylindrical tank in an arbitrary point of the y-axis. For the purpose of the software validation additional measures are performed. Then using the method of cubic splines, an appropriate interpolation function of the volume of the liquid depending on the height of the liquid is formed. The difference between the obtained results and the values obtained by the original software is within an predetermined interval. This approves that the software gives valid results.

VALIDATION AS AN IMPORTANT PART OF SOFTWARE QUALITY

A case study

Tanja Krunić