FUNCTIONAL STRUCTURE AND SPECIFIC CALCULATION ALGORITHMS FOR THE COMPUTER APPLICATION FOR SCHEDULING PRODUCTION OPERATIONS FOR MULTIFUNCTIONAL MACHINES REGARDING HYDRAULIC JACK CH.01.00

SZABO Loredana-Florentina
Facultatea de Inginerie Industrială și Robotică, Specializarea: Informatică Aplicată în Inginerie Industrială, Anul de studii: IV, e-mail: loredanaszabo16@yahoo.com

Conducător științific: Conf. dr. ing. **Gabriel-Mădălin CATANĂ**

ABSTRACT: This paper presents the functional structure of the computer application for scheduling production operations on multifunctional machines, which contains specific calculation methods. Starting from the functional structure, the specific programming algorithms for the presented calculation methods were presented in pseudocode format. This study will be performed on the unique parts of the Hydraulic Jack product.

CUVINTE CHEIE: programare, mașină multifuncțională, aplicație informatică, algoritmi de calcul, pseudocod.

1. Introducere

Programarea operațiilor de producție este o etapă de management pe termen scurt al producției, în care sunt stabilite datele de început și de sfârșit ale operațiilor de producție ale unor loturi (cantități) de articole și mașinile alocate producției în compartimentele de fabricație, astfel încât producția articolelor să se efectueze în anumite condiții de performanță dorite, în limitele capacității de producție existente în compartimentele de fabricație și în concordanță cu parametri de planificare a producției în planul necesarului de materiale (resurse) derivat din programul de producție director [3,4].

Programarea operațiilor de producție la mașini multifuncționale se aplică în cazul în care mai multe procese de producție uni-operaționale trebuie să fie realizate, de-a lungul unui orizont de timp, la o singură mașină disponibilă sau la un grup de mașini identice disponibile în acest orizont [3,4]. Universalitatea și flexibilitatea unei mașini multifuncționale permit ca procesul de producție al unui articol la mașină să poată fi concentrat într-o singură operație.

Metodele de programare a operațiilor la mașini multifuncționale se deosebesc în funcție de condițiile specifice luate în considerare la programarea operațiilor, definite prin:
- categoriile de date inițiale considerate privind operațiile de programat și mașinile disponibile pentru realizarea operațiilor;
- criteriul de performanță al programului operațiilor care va fi optimizat în cadrul metodei.

2. Structura aplicației

2.1. Prezentare generală a aplicației și a reperului Cric hidraulic CH.01.00

În cadrul proiectului de diplomă se are ca scop crearea unei aplicații informatice pentru programarea operațiilor de producție la o singură mașină multifuncțională sau la un grup de mai multe mașini identice multifuncționale.

Aplicația va consta în algoritmii de calcul specifici pentru fiecare metodă de optimizare a operațiilor de producție. Această aplicație va fi utilizată pentru programa operațiile de producție a pieselor unice/nestandardizare din mai multe tipuri de produse inclusiv produsul cric hidraulic CH.01.00.

În cadrul acestei lucrări se prezintă algoritmii de calcul ai metodele de programare a operațiilor de producție la o singură mașină multifuncțională, sub formă de pseudo-cod, privind piesele nestandardizate care fac parte din produsul cric hidraulic.

Cricul hidraulic este un dispozitiv utilizat cel mai frecvent în domeniul auto. Acest dispozitiv se utilizează pentru diversele tipuri de activități care necesită ridicarea obiectelor de mare greutate, având o baza stabilă și o capacitate de ridicare de aproximativ 12 tone, după cum se observă în fig.1.[6].

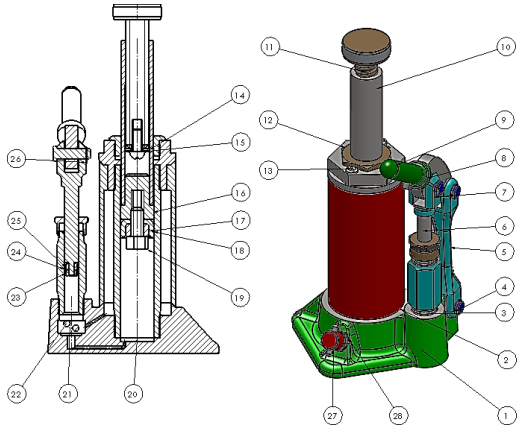

Fig.1. Ansamblu 3D Cric hidraulic

Tabel 1 Descrierea componentelor ansamblului

Nr. crt.	Cod	Denumire	Tip piesă	Nr. crt.	Cod	Denumire	Tip piesă
1	GH 2	Bază	Nestandardizat	15	GH 7	Șurub 4	Standardizat
2	GH 1	Cilindru	Nestandardizat	16	GH 8	Șurub 5	Standardizat
3	GH 14	Arbore	Nestandardizat	17	GH 9	Distanțier1	Standardizat
4	GH 21	Șurub 1	Standardizat	18	GH 11	Distanțier2	Standardizat
5	GH 20	Braț	Nestandardizat	19	GH 10	Șurub	Standardizat
6	GH 15	Bucșă 1	Nestandardizat	20	GH 3	Cilindru filetat exterior	Nestandardizat
7	GH 16	Braț 2	Nestandardizat	21	GH 25	Cilindru complex	Nestandardizat
8	GH 22	Mâner	Nestandardizat	22	GH 25 ESF	Bile	Standardizat
9	GH 13	Bucșă 3	Nestandardizat	23	GH 19	Piuliță	Standardizat
10	GH 3.2	Cilindru filetat interior	Nestandardizat	24	GH 18	Distanțier3	Standardizat
11	GH 4.5	Șurub 2	Standardizat	25	GH 17	Distanțier4	Standardizat
12	GH 12	Bucșă 2	Nestandardizat	26	GH 27	Șaibă 2	Standardizat
13	GH 26	Șurub 3	Standardizat	27	GH 24	Capac	Nestandardizat
14	GH 6	Șaibă 1	Standardizat	28	GH 23	Bucșă 4	Nestandardizat

*Standardizată – piesă ce poate fi achiziționată;

*Nestandardizată – piesă care se prelucrează.

2.2. Modul de funcționare al aplicației informatice

Utilizarea aplicației informatice pentru programarea operațiilor de producție la mașini multifuncționale pentru o gama largă de produse se va realiza în următoarele etape:

- Accesarea pagini web;
- Alegerea tipului de mașină: o singură mașină multifuncțională sau un grup de mașini identice multifuncționale;
- Alegerea metodei din listă. Fiecare metodă va avea restricții bine definite care să nu permită utilizatorului să introducă date greșite;
- Rularea aplicației, datele rezultate vor fi afișate și salvate într-o baza de date de unde vor putea fi accesate din nou.

Interfața Aplicației informatică va fi creată să fie cât mai intuitivă și ușor de utilizat astfel utilizatorii să nu întâmpine probleme la accesare acesteia. De asemenea scopul principal este să furnizeze informațiile aferente fiecărei cereri cât mai corect și mai rapid. Modul de funcționare al aplicației se observă în figura următoare (fig.2).

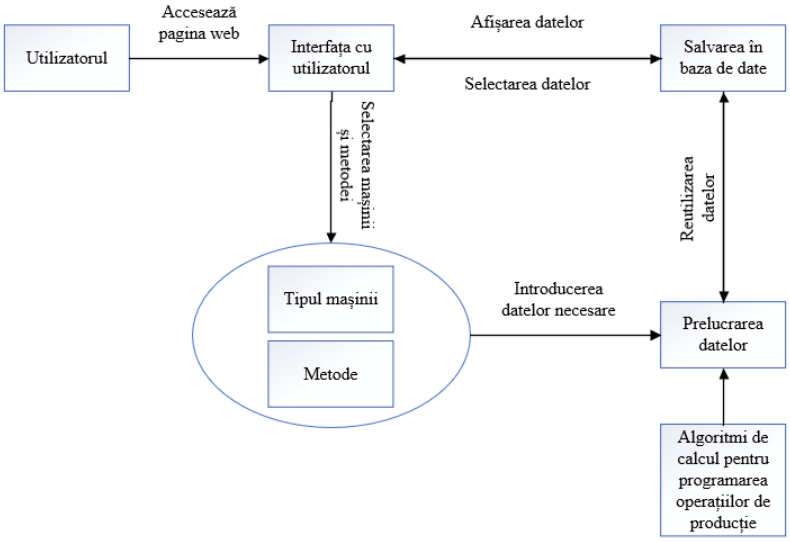

Fig.2. Schema bloc a aplicației informatice

3. Metode și algoritmi de calcul pentru programarea operațiilor la o singură mașină

3.1. Condițiile pentru aplicarea metodelor

Condițiile specifice de programare a operațiilor la una sau mai multe mașini multifuncționale pot fi sintetizate astfel [3,4]:

• Date privind operațiile de programat pentru articolele de fabricat (cu numere de ordine **g**):

- Termenul de lansare în producție a operației articolului ($\mathbf{T_{LPg}}$);
- Termenul de recepționare din producție a articolului după finalizarea operației ($\mathbf{T_{RPg}}$);
- Durata operației de producție a articolului la mașină ($\mathbf{TT_g}$);

• Date privind mașinile disponibile pentru operațiile de producție (cu numere de ordine **i**, astfel încât i=1 dacă este disponibilă o singură mașină și i>1 dacă sunt disponibile mai multe mașini pentru efectuarea operațiilor):

- Termenul de disponibilizare pentru producție a fiecărei mașini ($\mathbf{T_{Di}}$);

• Date privind programul operațiilor la mașinile alocate (rezultă în urma programării):

- Intervalul de timp programat pentru efectuarea operației de producție a articolului **g** la mașina **i**, [$\mathbf{DS_{g\,i}}$, $\mathbf{DF_{g\,i}}$], astfel încât $\mathbf{DF_{g\,i} = DS_{g\,i} + TT_g}$;
- Durata programată a ciclului de producție al articolului ($\mathbf{Tc_g}$), astfel încât $\mathbf{Tc_g = DF_{g\,i} - T_{LPg}}$;
- Întârzierea programată finalizării producției articolului ($\mathbf{\hat{I}_g}$), astfel încât $\mathbf{\hat{I}_g = max(DF_{g\,i} - T_{RPg}, 0)}$;

• Criteriul de performanță al programului operațiilor care se dorește a fi optimizat (minimizat):

- Data maximă programată pentru finalizarea operațiilor de producție a articolelor ($\mathbf{DF_{max}}$), astfel încât $\mathbf{DF_{max} = max(DF_{g\,i})}$;
- Durata totală programată pentru ciclurile de producție ale articolelor ($\mathbf{Tc_T}$), astfel încât $\mathbf{Tc_T = \Sigma\, Tc_g}$ sau, respectiv, durata totală ponderată ($\mathbf{Tc_{Tw}}$), definită prin $\mathbf{Tc_{Tw} = \Sigma\, w_g \cdot Tc_g}$;
- Întârzierea maximă programată finalizării producției articolelor ($\mathbf{\hat{I}_{max}}$), astfel încât $\mathbf{\hat{I}_{max} = max(\hat{I}_g)}$;
- Întârzierea totală programată pentru finalizarea producției articolelor ($\mathbf{\hat{I}_T}$), astfel încât $\mathbf{\hat{I}_T = \Sigma\, \hat{I}_g}$ sau, respectiv, întârzierea totală ponderată ($\mathbf{\hat{I}_{Tw}}$), definită prin $\mathbf{\hat{I}_{Tw} = \Sigma\, w_g \cdot \hat{I}_g}$;
- Numărul total programat de articole cu finalizare întârziată a producției ($\mathbf{N\hat{i}_T}$), astfel încât $N\hat{i}_T = \sum_{\hat{i}_g>0} 1$;

3.2. Metode de programare și pseudocoduri

• Procedura de programare pe baza timpului de realizare minim (SPT)

Algoritmul de programare descris în pseudocod pentru metoda SPT:

1. Introducerea de la tastatură a datelor inițiale:

- Articol g, g = 1,n (n – numărul total de articole diferite de programat pentru producție), T_{LPg}=0,, TT_g,;
- Mașina disponibilă i, i=1 deci T_{D1}=0;
2. Ordonarea crescătoare cu ajutorul unui algoritm de sortare prin interschimbare a parametrului TT_g și obținerea numărului de ordine la programare a articolului g, denumit P_g, P_g = 1,n.
3. Calcul iterativ (de n ori), în ordinea crescătoare a numerelor de ordine P_g:
- Stabilire intervale de timp în care are au loc operațiile articolelor g: [DS_{g1} , DF_{g1}];
    Data de început a operației articolului:
    DS_{g1} = 0, dacă P_g = 1
    DS_{g1} = $DF_{g'1}$, dacă P_g = $P_{g'}$ +1 (articolul g urmează articolului g' în ordinea de programare);
    Data de sfârșit a operației articolului: DF_{g1} = DS_{g1} + TT_g
4. Calcul parametru de calitate optimizat la programul de producție privind cele n articole:
    Durata totală programată pentru ciclurile de producție: $Tc_T = \Sigma_g (DF_{g1} - T_{LPg})$;
5. Afișare și salvare a datelor rezultate în urma programării operațiilor de producție:
- Date de început și sfârșit ale fiecărui articol g, g = 1,n: DS_{g1} , DF_{g1};
- Valoare parametru de calitate optimizat la programul de producție: $T_{CT.}$

• Procedura de programare pe baza timpului ponderat de realizare minim (WSPT)

Algoritmul de programare descris în pseudocod pentru metoda WSPT:

1. Introducerea de la tastatură a datelor inițiale:
- Articol g, g = 1,n (n – numărul total de articole diferite de programat pentru producție), T_{LPg}=0,, TT_g, w_g,;
- Mașina disponibilă i, i=1 deci T_{D1}=0;
2. Ordonarea crescătoare cu ajutorul unui algoritm de sortare prin interschimbare a parametrului P_g, unde P_g = TT_g / w_g:
3. Calcul iterativ (de n ori), în ordinea crescătoare a numerelor de ordine P_g:
- Stabilire intervale de timp în care are au loc operațiile articolelor g: [DS_{g1} , DF_{g1}];
    Data de început a operației articolului:
    DS_{g1} = 0, dacă P_g = 1
    DS_{g1} = $DF_{g'1}$, dacă P_g = $P_{g'}$ +1 (articolul g urmează articolului g' în ordinea de programare);
    Data de sfârșit a operației articolului: DF_{g1} = DS_{g1} + TT_g;
4. Calcul parametru de calitate optimizat la programul de producție privind cele n articole:
- Durata totală programată pentru ciclurile de producție: $Tc_T = \Sigma_g (DF_{g1} - T_{LPg})$;
- Durata totală ponderată programată pentru ciclurile de producție: $Tc_{Tw} = \Sigma_g w_g * T_{CT}$;
5. Afișare și salvare a datelor rezultate în urma programării operațiilor de producție:
- Date de început și sfârșit ale fiecărui articol g, g = 1,n: DS_{g1} , DF_{g1};
- Valoare parametru de calitate optimizat la programul de producție: T_{CT}, $Tc_{Tw.}$

• Procedura de programare pe baza termenului de recepționare minim (EDD*)*

Algoritmul de programare descris în pseudocod pentru metoda EDD:

1. Introducerea de la tastatură a datelor inițiale:
- Articol g, g = 1,n (n – numărul total de articole diferite de programat pentru producție), T_{LPg}=0,, TT_g,, T_{RPg};
- Mașina disponibilă i, i=1 deci T_{D1}=0;
2. Ordonarea crescătoare cu ajutorul unui algoritm de sortare prin interschimbare a parametrului T_{RPg} și obținerea numărului de ordine la programare a articolului g, denumit P_g, P_g = 1,n.
3. Calcul iterativ (de n ori), în ordinea crescătoare a numerelor de ordine P_g:
- Stabilire intervale de timp în care are au loc operațiile articolelor g: [DS_{g1} , DF_{g1}];
    Data de început a operației articolului:
    DS_{g1} = 0, dacă P_g = 1
    DS_{g1} = $DF_{g'1}$, dacă P_g = $P_{g'}$ +1 (articolul g urmează articolului g' în ordinea de programare);
    Data de sfârșit a operației articolului: DF_{g1} = DS_{g1} + TT_g
4. Calcul parametru de calitate optimizat la programul de producție privind cele n articole:
    Întârzierea totală programată pentru finalizarea producției articolelor: $\hat{I}_g$= max($DF_{g\,i}$ –T_{RPg},0);
    Întârzierea maximă programată finalizării producției articolelor: $\hat{I}_{max}$ = max (max(DF_{g1} –T_{RPg},0));

5. Afișare și salvare a datelor rezultate în urma programării operațiilor de producție:
- Date de început și sfârșit ale fiecărui articol g, g = 1,n: DS_{g1} , DF_{g1};
- Valoare parametru de calitate optimizat la programul de producție: $\hat{I}_g$, $\hat{I}_{max.}$

• Procedura de programare Hodgson (Moore)

Algoritmul de programare descris în pseudocod pentru metoda Moore:

*Această metodă utilizează principiul de ordonare al metodei EDD, apoi se continuă cu ordonarea parametrilor neprogramați.

1. Introducerea de la tastatură a datelor inițiale:
- Articol g, g = 1,n (n – numărul total de articole diferite de programat pentru producție), T_{LPg}=0,, $TT_{g,}$, T_{RPg};
- Mașina disponibilă i, i=1 deci T_{D1}=0;
2. Ordonarea crescătoare cu ajutorul unui algoritm de sortare prin interschimbare a parametrului T_{RPg} și obținerea numărului de ordine la programare a articolului g, denumit P_g, P_g = 1,n.
3. Calcul iterativ (de n ori), în ordinea crescătoare a numerelor de ordine P_g:
- Stabilire intervale de timp în care are au loc operațiile programate ale articolelor g: [DS_{g1} , DF_{g1}];

  Data de început a operației articolului:

  DS_{g1} = 0, dacă P_g = 1

  DS_{g1} = $DF_{g'1}$, dacă P_g = $P_{g'}$ +1 (articolul g urmează articolului g' în ordinea de programare);

  Data de sfârșit a operației articolului: DF_{g1} = DS_{g1} + TT_g
- Stabilirea operației care determină întârzierea operației întârziate și care nu a fost realocată într-o etapă anterioară trebuie să respecte două condiții: $DF_{g1} > T_{RPg}$; unde g = 1,n, ;

  $\hat{I}_g$= max($DF_{g\,i}$ –T_{RPg},0), unde $\hat{I}_g > 0$;
- Termenul P_g care respectă aceste condiții este alocat la sfârșitul liste ordonate;
4. Calcul parametru de calitate optimizat la programul de producție privind cele n articole:
- Numărul total programat de articole cu finalizare întârziată a producției: $N\hat{i}_T = \sum_{\hat{i}_g>0} 1$;
5. Afișare și salvare a datelor rezultate în urma programării operațiilor de producție:
- Date de început și sfârșit ale fiecărui articol g, g = 1,n: DS_{g1} , DF_{g1};
- Valoare parametru de calitate optimizat la programul de producție: $\hat{I}_{g.}$, $N\hat{I}_T$

• Procedura de programare pe baza termenului de lansare minim (ERD)

Algoritmul de programare descris în pseudocod pentru metoda ERD:

1. Introducerea de la tastatură a datelor inițiale:
- Articol g, g = 1,n (n – numărul total de articole diferite de programat pentru producție), T_{LPg} >=0,, $TT_{g,,}$ T_{RPg};
- Mașina disponibilă i, i=1 deci T_{D1}=0;
2. Calcul iterativ (de n ori), în ordinea crescătoare a numerele de ordine P_g:
- Stabilire intervale de timp în care are au loc operațiile programate ale articolelor g: [DS_{g1} , DF_{g1}];

  Data de început a operației articolului:

  DS_{g1} = 0, dacă P_g = 1

  DS_{g1} = $DF_{g'1}$, dacă P_g = $P_{g'}$ +1 (articolul g urmează articolului g' în ordinea de programare);

  Data de sfârșit a operației articolului: DF_{g1} = DS_{g1} + TT_g
- Stabilirea operațiilor neprogramate:

  DF_{gn} = DS_{gn} + TT_g, unde DF_{gn} este sfârșitul ultimei operații programate;

  T_{LPg} >= DF_{gn}, unde T_{LPg} este data de început a primei operații neprogramate;
3. Ordonarea crescătoare cu ajutorul unui algoritm de sortare prin interschimbare a parametrului T_{LPg} și obținerea numărului de ordine la programare a articolului g, denumit P_g, P_g = 1,n.
4. Alocarea operațiilor neprogramate ordonate crescător (P_g) la sfârșitul listei de operații programate: T_{LPg} >= 0, unde g = 1,n și T_{LPg} = DF_{gn};
5. Calcul parametru de calitate optimizat la programul de producție privind cele n articole:

   Data maximă programată pentru finalizarea operațiilor de producție a articolelor:

   DF_{max}= max($DF_{g\,1}$), unde g= 1,n;
6. Afișare și salvare a datelor rezultate în urma programării operațiilor de producție:

   Date de început și sfârșit ale fiecărui articol g, g = 1,n: DS_{g1} , DF_{g1}, T_{LPg};

Valoare parametru de calitate optimizat la programul de producție: $DF_{max.}$.

• Procedura de programare pe baza termenului de recepționare minim la data de lansare (EDD-R)

Algoritmul de programare descris în pseudocod pentru metoda EDD-R:

1. Introducerea de la tastatură a datelor inițiale:
- Articol g, g = 1,n (n – numărul total de articole diferite de programat pentru producție), $T_{LPg} >= 0$,, $TT_{g,}$, T_{RPg};
- Mașina disponibilă i, i=1 deci $T_{D1}=0$;
2. Calcul iterativ (de n ori), în ordinea crescătoare a numerele de ordine P_g:
- Stabilire intervale de timp în care are au loc operațiile programate ale articolelor g: [DS_{g1} , DF_{g1}];

  Data de început a operației articolului:

  $DS_{g1} = 0$, dacă $P_g = 1$

  $DS_{g1} = DF_{g'1}$, dacă $P_g = P_{g'} +1$ (articolul g urmează articolului g' în ordinea de programare);

  Data de sfârșit a operației articolului: $DF_{g1} = DS_{g1} + TT_g$
- Stabilirea operațiilor neprogramate:

  $DF_{gn} = DS_{gn} + TT_g$, unde DF_{gn} este sfârșitul ultimei operații programate;

  $T_{LPg} >= DF_{gn}$, unde T_{LPg} este data de început a primei operații neprogramate;
3. Ordonarea crescătoare cu ajutorul unui algoritm de sortare prin interschimbare a parametrului T_{RPg} și obținerea numărului de ordine la programare a articolului g, denumit P_g, P_g = 1,n.
4. Alocarea operațiilor neprogramate ordonate crescător (P_g) la sfârșitul listei de operații programate: $T_{LPg} >= 0$, unde g = 1,n și $T_{LPg} = DF_{gn}$;
5. Calcul parametru de calitate optimizat la programul de producție privind cele n articole:

   Întârzierea totală programată pentru finalizarea producției articolelor: $\hat{I}_g = \max(DF_{g\,i} - T_{RPg}, 0)$;

   Întârzierea maximă programată finalizării producției articolelor: $\hat{I}_{max} = \max(\max(DF_{g1} - T_{RPg}, 0))$;
6. Afișare și salvare a datelor rezultate în urma programării operațiilor de producție:

   Date de început și sfârșit ale fiecărui articol g, g = 1,n: DS_{g1} , DF_{g1};

   Valoare parametru de calitate optimizat la programul de producție: $\hat{I}_g$, $\hat{I}_{max}$.

4. Concluzii

În această lucrare a fost prezentată structura funcțională a aplicației informatice de programare a operațiilor de producție, care conține metode de programare specifice pentru o singură mașină multifuncțională, algoritmii de calcul specifici ai acestor metode fiind prezentați sub format pseudocod.

În cadrul proiectului de diplomă se va continua acest studiu prezentând și metodele de programare a operațiilor de producție la mai multe mașini identice multifuncționale, algoritmii de calcul specifici acestor metode și prezentarea lor sub format pseudocod.

Aceste metode se vor utiliza cel puțin pentru optimizarea proceselor de producție ale pieselor unice din cadrul produsului Cric Hidraulic.

5. Bibliografie

[1]. Anil Kumar, S., Suresh, N., *Operations Management*, New Age International Publishers, New Delhi, 2009

[2]. Baker, K.R., Trietsch, D., *Principles of Sequencing and Scheduling*, 2nd Edition, John Wiley & Sons, Inc., Hoboken, New Jersey, 2019.

[3]. Catană, M., *Managementul producției: Baze teoretice*, Editura Politehnica Press, Bucureşti, 2019

[4]. Catană, M., *Planificarea asistată a proceselor industriale - Partea I: Suport de curs*, https://fiir.curs.pub.ro/2019/

[5]. Pinedo, M.L., *Scheduling: Theory, Algorithms, and Systems*, 5th Edition, Springer Science + Business Media, LLC, New York, 2016.

[6]. Szabo L.F., *Proiect Proiectarea și Fabricarea Produselor,* Facultatea Inginerie Industrială și Robitică, 2020-2021.