IMK-14 – Istraživanje i razvoj u teškoj mašinogradnji 26(2020)1, SR 13-18
UDC 621 ISSN 0354-6829
doi: 10.5937/IMK2001013R

Optimizacija problema raspoređivanja poslova u Industriji 4.0

Olga Ristić[1*], Marjan Milošević[1], Sandra Milunović Koprivica[1], Milan Vesković[1], Veljko Aleksić[1]
[1]Fakultet tehničkih nauka, Univerzitet u Kragujevcu, Čačak (Srbija)

Rezime: Razvoj četvrte industrijske revolucije (Industry 4.0) primenom digitalizacije ima značajan uticaj na promene koje nastaju u proizvodnim sistemima. U industrijskim proizvodnim sistemima se povećava složenost raspoređivanja poslova na mašinama. Osnovni cilj raspoređivanja je prikupljanje podataka pomoću savremenih tehnologija kao što su: primena senzora na mašinama, klaud računarstvo, veštačka inteligencija, analiza velike količine podataka, softveri za raspoređivanje itd. U radu je prikazan primer primene softvera otvorenog koda za raspoređivanje poslova (LEKIN) koji se koristi za rešavanje inženjerskih problema u proizvodnim firmama. Primena algoritama za optimizaciju i softvera za raspoređivanje poslova uticaće na smanjenje proizvodnih troškova i minimizaciju ukupnog vremena završetka proizvodnje. U radu je dat primer testiranja problema i upoređivanje razlika između primene heuristike pomeranja uskih grla (Shifting Bottleneck Heuristic - SBH) i nekoliko pravila izvršavanja kao što su: najranije vreme početka (Earliest Due Date - EDD), pravilo prvi posao koji je na redu, prvi se izvrši (First Come First Served - FCFS) i najkraće vreme izvršenja (Shortest Processing Time - SPT). Primenjen je kriterijum evaluacije ukupnog vremena izvršavanja i kašnjenja operacija proizvodnje. Rezultati pokazuju da je SBH efikasnija u odnosu na primenjena pravila izvršavanja.

Ključne reči: Algoritmi raspoređivanja, Optimizacija, Job shop, industrija 4.0, Lekin

1. UVOD

Poslednjih godina sa razvojem industrije 4.0 složenost proizvodnih sistema je sve veća. Da bi se skratilo vreme proizvodnje i smanjili troškovi neophodno je izvršti optimizaciju poslova koji se obavljaju na mašinama. Poslednjih par decenija razvijeni su algoritmi raspoređivanja koji efikasno rešavaju različite probleme optimizacije kako u proizvodnim sistemima, tako i u uslužnim sistemima. U ovim sistemima nastaju problemi kao što su: određivanje rasporeda i redosleda proizvodnje proizvoda na mašinama, utvrđivanje redosleda plana aktivnosti na projektu, raspored izvršilaca u procesu pružanja usluga, ogranizovanje rasporeda transporta itd. Ovi problemi se svrstavaju u NP-teške probleme (nedeterminističke u polinomijalnom vremenu teško rešive probleme) zbog ograničenja u pogledu kapaciteta ljudi i opreme na kojoj se obavlja proces proizvodnje. Za rešavanje ovih problema se koriste jednostavne heuristike i to najčešće u vidu pravila raspoređivanja ili složene heuristike kao što su npr. metode veštačke inteligencije.

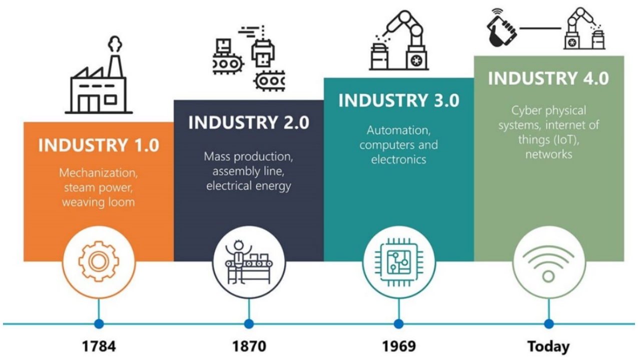
U radu je prvo prikazan razvoj industrijske revolucije, od nastanka prve industrijske revolucije, pa do četvrte. Poseban osvrt je dat na karakteristike industrijske revolucije četvrte generacije gde se primenjuju savremeni inteligentni sistemi za upravljanje procesom proizvodnje.

U narednom poglavlju je dat opis algoritama koji se koriste u optimizaciji procesa raspoređivanja. Izvršena je osnovna podela ovih algoritama na aproksimativne metode i optimizacione metode.

U četvrtom poglavlju su prikazani neki od najčešće korišćenih softvera za raspoređivanje. Dat je detaljan opis LEKIN sistema raspoređivanja, kao jednostavnog softvera koji se može koristiti kako u edukativne svrhe, tako i za raspoređivanje složenih poslova u proizvodnim sistemima. Dat je primer kojim se opisuje jednostavnost primene ovog softvera.

2. RAZVOJ INDUSTRIJE 4.0

Krajem XVIII veka nastala je prva industrijska revolucija nazvana generacija 1.0 koja se zasnivala na upotrebi parnih mašina, mehanizaciji, kao i primeni razboja u tekstilnoj industriji (slika 1) [1]. Već krajem XIX veka dolazi do masovne proizvodnje, pojave električne energije i proizvodnih linija, što je uticalo na pojavu druge industrijske revolucije. Sa pojavom automatizacije, računara i razvojem informacionih tehnologija, krajem 1960-tih godina nastaje treća industrijska revolucija. Poslednju deceniju karakteriše razvoj interneta, mreža, Internet stvari (Internet of Things - IoT), manipulacija sa velikom količinom podataka, primena senzora itd.



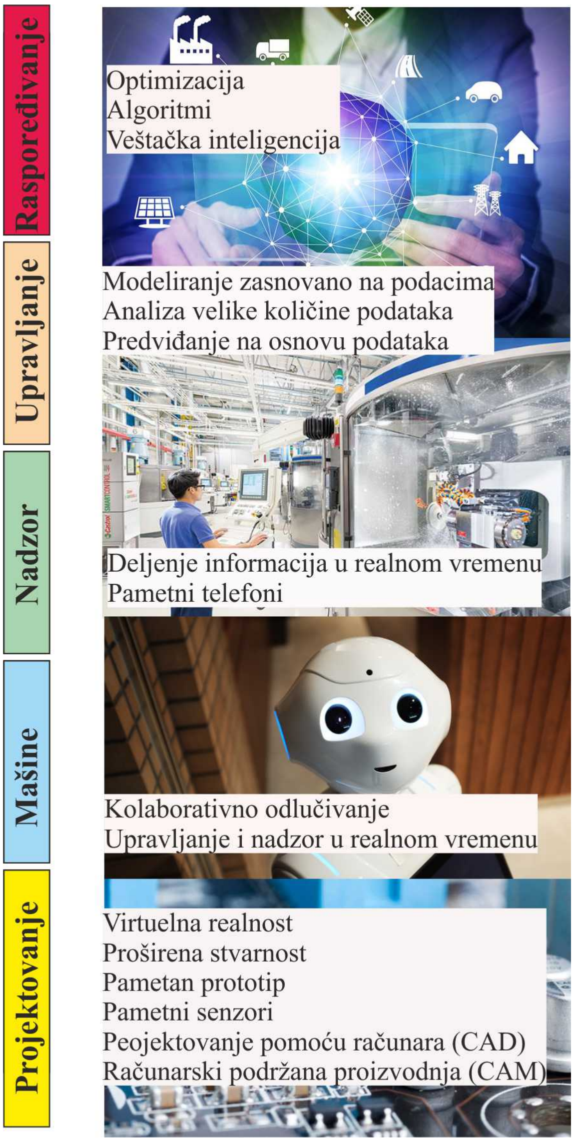
Slika 1: Razvoj industrijske revolucije

Sve je to uticalo da već 2011. godine započne nova era četvrte industrijske revolucije (Industry 4.0). Osnovni cilj je kreiranje inteligentnih fabrika sa proizvodnim tehnologijama koje se transformišu pod uticajem sajber sistema, klaud računarstva, mašinskog učenja i IoT [2,3]. Na ovaj način je omogućeno donošenje pametnih odluka komunikacijom između ljudi, mašina, senzora itd.

Industrijski proizvodni sistemi imaju niz prednosti u odnosu na ranije proizvodne sisteme, jer se na ovaj način primenjuju različite vrste softvera kojima se vrši optimizacija i donošenje odluka na svim nivoima upravljanja procesom proizvodnje. Neke od ovih tehnologija koriste veštačku inteligenciju koja omogućava proizvodnim sistemima da uče iz prethodnih iskustava industrijske prakse [4].

Kod industrijske revolucije 4.0 osnovni cilj je skraćenje vremena proizvodnje i donošenje pravih odluka u proizvodnim sistemima. Inteligentni sistemi upravljanja u proizvodnji se mogu podeliti na nekoliko nivoa (slika 2) [5]:

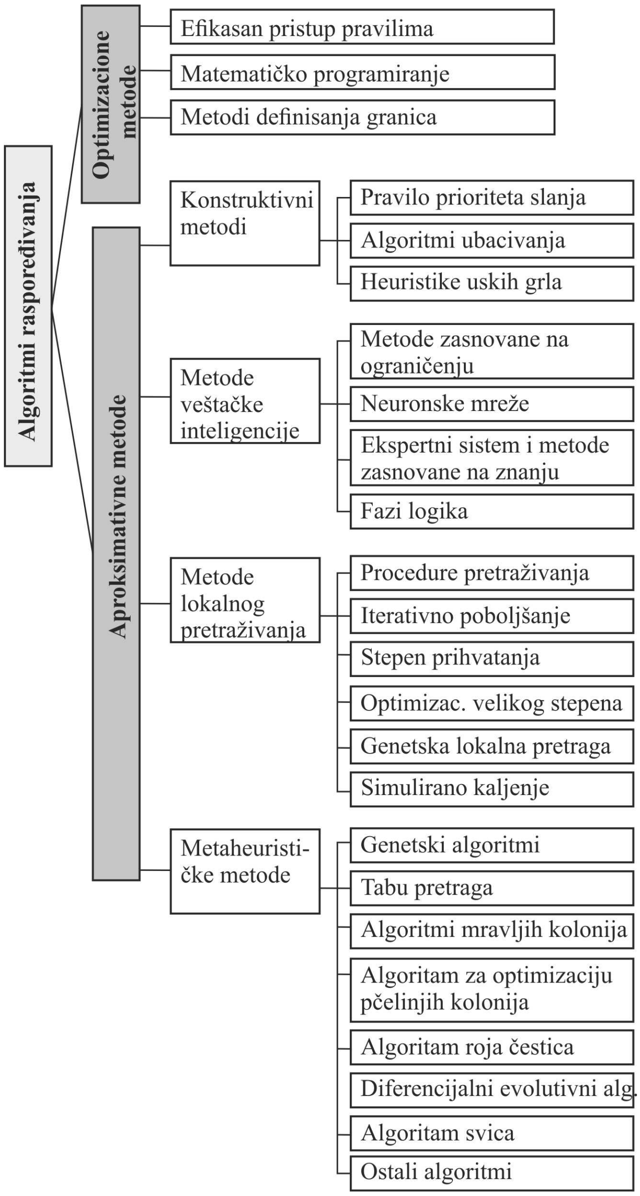
1. pametno projektovanje,
2. pametne mašine,
3. pametan nadzor
4. pametno upravljanje,
5. pametno raspoređivanje.



Slika 2: Prikaz sistema inteligentnog upravljanja u industriji 4.0 generacije

3. ALGORITMI OPTIMIZACIJE ZA PROCES RASPOREĐIVANJA

Sa razvojem industrije nove generacije 4.0 osnovni cilj je skraćenje vremena proizvodnje i smanjenje svih troškova (proizvodnje, održavanja opreme, radne snage, ...) [6]. U prvom koraku je neophodno izvršiti optimizaciju izvođenja poslova koji se obavljaju na određenim mašinama. Koji algoritam će se koristiti za optimizaciju zavisi od brojnih faktora koji se analiziraju. Sve složeniji proizvodni sistemi sa velikim brojem mašina i mašinskih centara u kojima se izvode razne operacije, uticali su da se poslednjih par decenija intenzivno razvijaju i primenjuju algoritmi raspoređivanja. Osnovna podela ovih algoritama je na aproksimativne metode i optimizacione metode (slika 3) [7].



Slika 3: Pregled algoritama optimizacije koji se koriste u procesu raspoređivanja poslova

Primena određenog algoritma zavisi prvenstveno od složenosti problema koji se rešava. Autori radova [6,7,8] su primenili različite algoritme kako bi došli do optimalnog rešenja u pogledu planiranja i raspoređivanja poslova koji se izvode u proizvodnim sistemima.

4. SOFTVERI ZA RASPOREĐIVANJE POSLOVA

4.1. Opis

Poslednjih par decenija razvijen je veliki broj softvera koji se koriste u procesu planiranja i raspoređivanja složenih poslova. Neki od njih su prvenstveno imali edukativnu ulogu, dok su drugi komercijalnog karaktera. Najčešće se primenjuju sledeći [9]:

- SAP,
- IBM sistem raspoređivanja,
- AMD sistem raspoređivanja,
- ORTEMS' – alat za agilnu proizvodnju,
- Asprova,
- Preactor,
- MS Project,
- LEKIN.

U radu ćemo se zadržati na prikazu LEKIN sistema raspoređivanja, jer pored komercijalne verzije ima i besplatnu verziju.

4.2. LEKIN – softver za raspoređivanje poslova

LEKIN softver se koristi za raspoređivanje poslova u proizvodnim i uslužnim sistemima. Prvobitno je razvijen kao softver za demonstriranje teorije i primene algoritama raspoređivanja. Profesor Michael L. Pinedo i Xiuli Chao vodili su projekat razvoja ovog softvera krajem 20-tog veka [9]. Besplatna verzija ovog softvera se može pronaći na sajtu [10].

LEKIN softver može podržavati različita radna okruženja u koje spadaju [11]:

1. *Single Machine Scheduling (SMS)* je proces kojim se dodeljuje grupa poslova na jednoj mašini. Poslovi se raspoređuju tako da se može optimizovati vremena izvršenja.

2. *Parallel Machines Scheduling (PMS)* je proces kod kojeg je veći broj mašina raspoloživ za realizaciju poslova (operacija). Nije bitno kojoj mašini je dodeljen koji posao, ali posao se ne može izvoditi na više od jedne mašine istovremeno.

3. *Flow Shop Scheduling (FSS)* je proces raspoređivanje tačno definisanog niza od *m* broja operacija na *n* mašina. Minimalni zastoj i minimalno vreme čekanja su ograničenja u neprekidnom toku procesa. Obično se pronalazi da proizvodni pogoni koriste FSS probleme. FSS problem je opšti oblik job shop problema fleksibilnih proizvodnih sistema. Ovde svaka mašina ima mogućnost obavljanja više operacija nekog posla.

4. *Flexible Flow Shop Scheduling (FFSS)* je problem gde je moguće dupliranje mašina i mašine su grupisane u radne stanice na kojima se proizvode isti tipovi proizvoda.

5. *Job Shop Scheduling (JSS)* predstavlja problem raspoređivanja n proizvoda na m mašina u kojima svaki proizvod ima unapred određenu putanju koju treba slediti, a sastoji se od niza operacija na mašinama.

6. *Flexible Job Shop Scheduling (JSS)* je vrsta rasporeda radnih mesta u kojima se za jednu operaciju posla može koristiti više različitih mašina.

LEKIN sistem se sastoji od četiri interfejsa za prikaz podataka:

1. Job Pool Window
2. Sequence Window
3. Gantt Chart (Schedule) Window
4. Graphical User Interface

LEKIN softver u besplatnoj verziji može upravljati maksimalno sa 50 poslova, 20 radnih stanica i 100 mašina. Istalacija sistema je jednostavna. Kada se softver pokrene na glavnom meniju se može pristupiti klikom na opciju "File", a potom na "Start over". Nakon toga korisnik ručno unosi podatke o mašinama i operacijama koje se izvode na mašinama. Fajl sa unetim podacima može se izmeniti i sačuvati kao izmenjeni fajl pod novim imenom. Nove podatke korisnik unosi izborom sa padajućeg menija opcije Workspace. U dijalog-boksu korisnik unosi osnovne informacije o dostupnim radnim stanicama i broju poslova koje treba rasporediti. Tada se pojavljuje još jedan dijalog-boks u koji se unose detaljne informacije o radnoj stanici o dostupnosti, broju poslova, vremenima podešavanja itd.

U trećem dijalog-boksu korisnik mora uneti podatke o operacijama koje se izvode na mašinama (prioriteti ili težine, datumi rokova, datumi završetka, vreme obrade različitih operacija).

4.3. Pravila raspoređivanja poslova

Kada se skup poslova obradi različitim operacijama u LEKIN softveru, tada će se prema njihovim rokovima rasporediti poslovi kako bi se minimiziralo vreme završetka operacija. Postoje mnoga pravila za optimizaciju raspoređivanja poslova. Ovde ćemo objasniti neka od njih [9]:

1. First come – First Served (FCFS)

Kod ovog pravila, posao koji stigne prvi, prvo se izvršava. To je ujedno i najjednostavniji način pridržavanja pravila u procesu proizvodnje, jer je fer prema kupcu. Glavni nedostatak ovog pravila je što duži poslovi iziskuju od ostalih da čekaju što utiče na kašnjenje u roku isporuke i ostali poslovi čekaju duže.

2. Shortest Processing Time (SPT)

Kod ovog pravila se prvo izvršavaju poslovi koji traju kraće. Prednost se ogleda u minimiziranju vremena završetka svih operacija, kao i minimiziranje prosečnog broja poslova u sistemu. Nedostaci su u nedostatku informacije o vremenu početka obrade, a duži poslovi čekaju duže na izvršenje.

3. Earliest Due date (EDD)

Posao koji ima najbliži rok, prvo se izvršava, ali ako postoji više poslova sa istim rokom onda se prvo uzimaju poslovi koji traju kraće. Ukoliko je visok prioritet dospelih poslova, ignorišu se preostali poslovi i prvo izvode prioritetniji.

4. Critical Ratio (CR) Rule

Poslovi koji se mogu podeliti na vremenske intervale, uklapaju se u vreme obrade. Prvo se obavljaju poslovi sa najmanjim rokom dospeća i vremenom obrade. Redosled poslova zavisi od odnosa (Datum početka obrade)/(Preostalo vreme obrade), gde se preostalo vreme obrade odnosi na redosled, podešavanje, pokretanje, vreme čekanja i premještanja iz jednog mašinskog centra u drugi.

Informacije o svakom proizvodu prikazane su kroz prozor *Job Pool* (Slika 3). Poslovi koji se izvode na mašinama imaju sledeće parametre:

Rls *(Release date - r_j)* - vreme kada operacija za izradu proizvoda postaje raspoloživa.

Due *(Due date - d_j)* - vreme završetka proizvoda. Ukoliko se obrada proizvoda ne završi do tog vremena, dolazi do kašnjenja. Osnovni cilj raspoređivanje je smanjenje kašnjenja na najmanju moguću meru.

Wght *(Weight - w_j)* - težinski koeficijent koji određuje značajnost proizvoda. Ako je proizvod značajniji, onda je ovaj koeficijent veći, pa se tako smanjuje kašnjenje izrade tog proizvoda jer ima veći prioritet.

Pr.tm*. (Processing Time- p_j)* – vreme obrade proizvoda koje se dobija sabiranjem vremena obrade svih operacija obrade.

Stat *(Status)* - predstavlja status operacije u kom se definiše pripremno vreme operacije i obeležava se slovima od A-Z. Korisnik zadaje programu vodeći računa da operacije sa istim statusom imaju ista pripremna vremena.

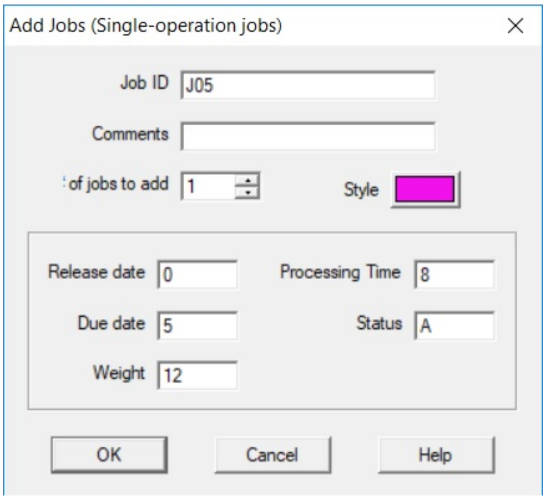
Bgn – period početka obrade proizvoda.

End – period završetka obrade proizvoda.

T – ukupno kašnjenje proizvoda.

wT – ukupno kašnjenje proizvoda pomnoženo težinskim koeficijentom.

(+)(Route) - putanja j-tog proizvoda kojom se određuje redosled operacija izrade na mašinama. Da bi se proverio redosled izrade proizvoda na mašinama, klikne se na + u prozoru Job Pool.



Slika 4: Dijalog prozor za dodavanje poslova

Nakon izbora odgovarajućih parametara korisnik bira komande iz menija Schedule - pravila raspoređivanja koja mogu biti:

- SPT, EDD, ATCS (Apparent Tardiness Cost with Setups), LPT (Longest Processing Time) - jednostavna pravila raspoređivanja,
- Shifting Bottleneck, Local Search - složene heuristike.
- Algoritme koje je kreirao korisnik i koji su dostupni u .exe verziji.
- Ručni unos redosleda poslova po mašinama.

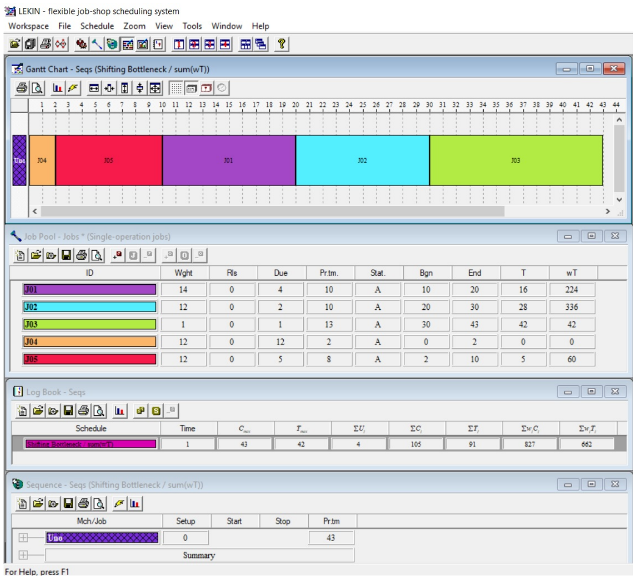
Da bi se na osnovu zadatih podataka za svaki posao i mašinu generisao gantogram, koriste se kriterijumske funkcije za merenje uspešnosti [12]. Funkcije koje su definisane u programu LEKIN su sledeće:

- Makespan - C_{max} (vreme obrade poslednjeg proizvoda)
- The maximum tardiness - T_{max} (maksimalno zakašnjenje, odnosno razlika između stvarnog završetka i očekivanog završetka obrade)
- The total number of late jobs - $\Sigma\ U_j$ (ukupan broj proizvoda koji kasne sa obradom)
- The total flow time - $\Sigma\ C_j$ (ukupno vreme završetka obrade svih proizvoda)
- The total tardiness - $\Sigma\ T_j$ (ukupno vreme kašnjenja obrade svih proizvoda)
- The total weighted flow time - $\Sigma\ w_jC_j$ (ukupno vreme završetka obrade svih proizvoda pomnoženo težinskom funkcijom)
- The total weighted tardiness - $\Sigma\ w_jT_j$ (ukupno vreme kašnjenja obrade svih proizvoda pomnoženo težinskom funkcijom)

4.4. Primer raspoređivanje poslova

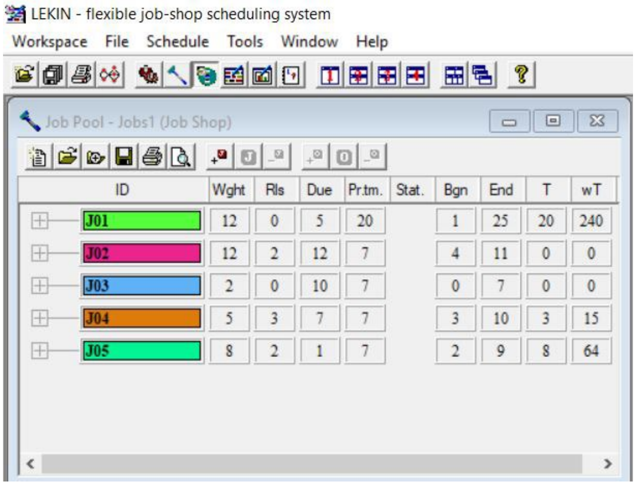
Primenom softvera LEKIN za raspoređivanje poslova na jednoj mašini potrebno je uneti za svaki posao ID, datum početka posla, vreme završetka (due date), težinski faktor, vreme obrade, status i stil (izbor boje koja se predstavljati dati posao u gantogramu). Na slici 4 je dati prikaz unosa podataka za jednu operaciju.

Za trivijalan primer raspoređivanja poslova na jednoj mašini uneti su podaci za pet poslova koji se mogu obaviti na mašini i na slici 5 je prikazan raspored svih poslova. Odabran je algoritam izbora poslova sa kritičnim vremenom završetka.

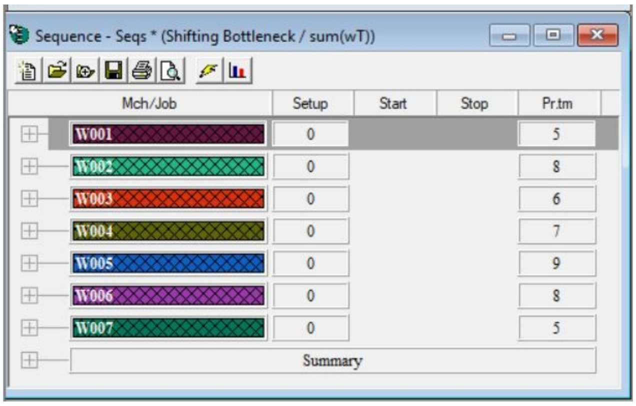


Slika 5: Primer raspoređivanja poslova na jednoj mašini

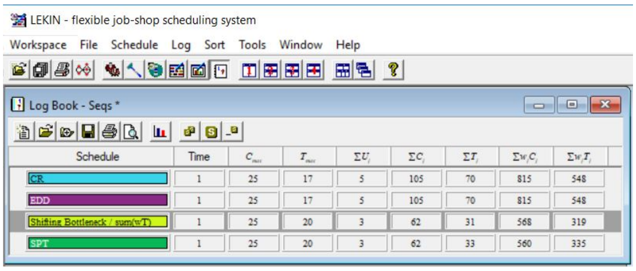
Jedan od složenijih problem raspoređivanja je raspoređivanjem operacija na mašinama. Ovde ćemo prikazati problem sa raspoređivanjem sedam radnih stanica i pet operacija koje se izvode na mašinama. Na slici 6 je prikazan Job Pool prozor za dati primer. Radne stanice i primena SBH za dati primer su prikazane na slici 7. Primenom različitih pravila raspoređivanja (CR, EDD i SPT) i heuristike SBH, prikazani su podaci o rasporedu operacija datih na slici 8. U ovom slučaju, minimalno kašnjenje poslova se dobija primenom heuristike SBH. Na slici 9 prikazana je Gant karta SBH sa rasporedom izvođenja operacija na mašinama.



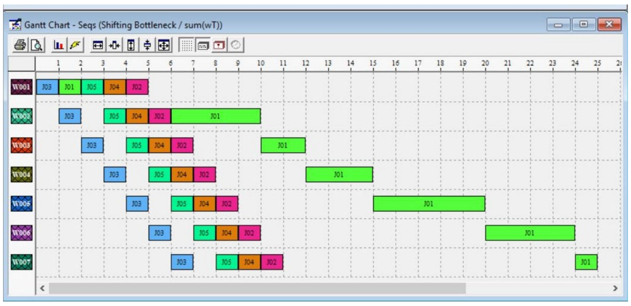
Slika 6: Problem raspoređivanja poslova prikazan u Job Pool prozoru



Slika 7: SBH za problem raspoređivanja poslova



Slika 8: Analiza podataka u Log Book za problem raspoređivanja poslova



Slika 9: Gantt karta za problem raspoređivanja poslova

5. ZAKLJUČAK

Primena algoritama raspoređivanja u proizvodnim sistemima je značajna u optimizaciji procesa proizvodnje. U ovom radu je prikazana primena LEKIN sofverskog sistema koji se koristi za optimizaciju planiranja i raspoređivanja u proizvodnim i uslužnim sistemima. LEKIN ima niz prednosti:

- jednostavnost upotrebe u obrazovanju, kao i u procesu istraživanja optimizacionih problema,
- mogućnost kreiranja optimalnog rasporeda rada mašina,
- u radu se mogu koristiti nove heurističke tehnike.

Primenom pravila raspoređivanja i heurističke tehnike pomeranja uskih grla (Shifting Bottleneck) dobijeni su rezultati o vremenima kašnjenja na primeru sedam radnih stanica i pet vrsta operacija koje se mogu izvoditi na mašinama. Može se iz datog primera zaključiti da je primena heurističke tehnike efikasnija u rešavanju složenih problema raspoređivanja operacija na mašinama u odnosu na jednostavna pravila raspoređivanja.

ZAHVALNICA

Ovaj rad je podržan od strane Ministarstva prosvete, nauke i tehnološkog razvoja Republike Srbije kroz projekte III46001 i III47003 za period 2011-2019.

LITERATURA

[1] http://www.isitan.com/en/technology-industry-40-ready.html

[2] A. Nicolae, A. Korodi And I. Silea, “An Overview of Industry 4.0 Development Directions in the Industrial Internet of Things Context”, Romanian Journal Of Information Science And Technology, Vol 22(3–4), pp. 183–201, (2019)

[3] J. Zhang, G. Ding, Y. Zou, S. Qin and J. Fu, “Review of job shop scheduling research and its new perspectives under Industry 4.0”, Journal of Intelligent Manufacturing, Vol. 30, issue 4, No 19, pp. 1809-1830, (2019)

[4] D. Li, Å. Fast-Berglund, & D. Paulin, “Current and future Industry 4.0 capabilities for information and knowledge sharing”, International Journal of Advanced Manufacturing Technology, Vol. 105, pp. 3951–3963, (2019)

[5] Alexandre Dolgui, Dmitry Ivanov, Suresh P. Sethi & Boris Sokolov, “Scheduling in production, supply chain and Industry 4.0 systems by optimal control: fundamentals, state-of-the-art and applications”, International Journal of Production Research, Vol. 57(2), pp. 411-332, (2019)

[6] G. Miodragović, R. Bulatović, S. Ivanović, M. Bošković, “The Use of Biologically-inspired Algorithms for the Optimization of Machining Parameters”, IMK-14 – Research & Development in Heavy Machinery, Vol. 21(3), pp. 93-97, (2015)

[7] M. A. Cruz-Chávez *et al.*, "Hybrid Micro Genetic Multi-Population Algorithm With Collective Communication for the Job Shop Scheduling Problem", *IEEE Access*, Vol. 7, pp. 82358-82376, (2019)

[8] H. Chang, Y. Chen, T. Liu and J. Chou, "Solving the Flexible Job Shop Scheduling Problem With Makespan Optimization by Using a Hybrid Taguchi-Genetic Algorithm", in *IEEE Access*, Vol. 3, pp. 1740-1754, (2015)

[9] M. L. Pinedo, “Scheduling-Theory, Algorithms, and Systems”, Springer, p. 670, (2016)

[10] http://web-static.stern.nyu.edu/om/software/lekin/ Last time access 12.12.2019.

[11] M. Dave and K. Choudhary, "Job shop scheduling algorithms — A shift from traditional techniques to non-traditional techniques," 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, pp. 169-173,(2016)

[12] Z. Rakićević and J. Kojić, “Software “Lekin” for solving scheduling problems in planning of production and servicing”, INFO M, Vol. 46, pp. 27-33, (2013)

IMK-14 – Research & Development in Heavy Machinery 26(2020)1, EN 13-18
UDC 621 ISSN 0354-6829
doi: 10.5937/IMK2001013R

Optimization of the Job Shop Scheduling Problem in Industry 4.0

Olga Ristić[1*], Marjan Milošević[1], Sandra Milunović Koprivica[1], Milan Vesković[1], Veljko Aleksić[1]
[1]Faculty of Technical Sciences, University of Kragujevac, Čačak (Srbija)

Abstract: The Industry 4.0 trend has brought significant transformation in the manufacturing process through digitalization. In Intelligent Manufacturing Systems (IMS) there is an increase in the complexity of scheduling jobs on machines. The scheduling aims to collect data through the support of novel and emerging technologies such as: inclusion of machine sensors, cloud computing, artificial intelligence, big data analytics, scheduling software, etc. In this paper, we present an example of open-source job shop scheduling software (LEKIN) to solve real-time engineering problems in a manufacturing company. With using optimization algorithms and scheduling software there is likely to be a reduction of production costs and minimization of the total order completion time (make span). A test problem is running to evaluate the difference between the implementation of Shifting Bottleneck Heuristic (SBH) and some dispatching rules, such as Earliest Due Date (EDD), First Come First Served (FCFS), and Shortest Processing Time (SPT). The evaluation criteria used were the make span and the total weighted tardiness. The results have shown that the SBH outstripped the dispatching rules.

Keywords: Scheduling algorithm, Optimization, Job shop, Industry 4.0, Lekin

1. INTRODUCTION

In recent years, with the development of Industry 4.0, the complexity of production systems has been increasing. In order to shorten production time and reduce costs, it is necessary to optimize the jobs performed on machines. In the last couple of decades, scheduling algorithms have been developed that effectively solve various optimization problems in production and service systems. In these systems, problems arise such as: determining the schedule and order of job shops on machines, determining the order of the project activity plan, scheduling in the service delivery process, limiting transportation schedules, scheduling maintenance activities, etc. These problems are categorized as NP-hardness problems because of the limitations on the capacity of the people and the equipment on which the production process takes place. Simple heuristics are being used to solve these problems, most often in the form of scheduling rules or complex heuristics such as artificial intelligence methods.

In the section 2, paper presents the development of the industrial revolution, from Industry 1.0 to Industry 4.0. Particular attention is given to the characteristics of the fourth-generation industrial revolution where modern intelligent systems for managing the production process are applied.

Section 3 describes the algorithms used in optimizing the scheduling process. Here, the basic division of these algorithms into approximate methods and optimization methods is performed.

Section 4 presents some of the most commonly used scheduling software. A detailed description of the LEKIN scheduling system is given, as simple software that can be used for educational purposes and for scheduling complex jobs in production systems. An example is given to describe the ease of deployment of this software.

2. DEVELOPMENT OF INDUSTRY 4.0

At the end of the 18th century, the first industrial revolution evolved and it was called Industry 1.0. It was based on the use of steam engines, mechanization, and the application of weaving loom in the textile industry (Figure 1) [1]. Already at the end of the 19th century, there was a mass production, the emergence of electricity and production lines, which influenced the development of the second industrial revolution. With the advent of automation, computers and the information technology, the third industrial revolution began in the late 1960s. The last decade has been characterized by the development of the Internet, networks, the big data, the deployment of sensors, etc.

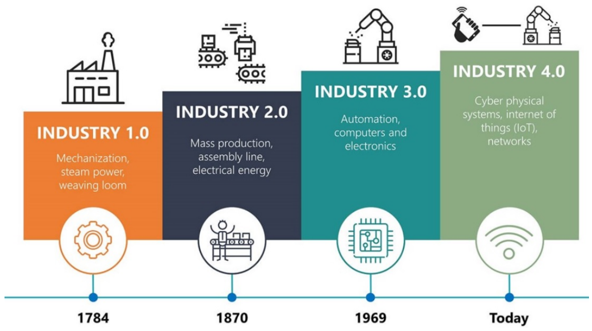


Figure 1: The development of industrial revolution

All of this influenced the launching of a new era of the fourth industrial revolution (Industry 4.0) in 2011th. The main goal is to create intelligent factories with productive technologies that are transformed by cyber-physical systems, cloud computing, machine learning and Internet of Thing (IoT) [2,3]. This enables smart decisions to be made by communication between humans, machines, sensors, etc.

*Corresponding author: Faculty of Technical Sciences, Svetog Save 65, 32000 Čačak, olga.ristic@ftn.kg.ac.rs

IMS has a number of advantages over earlier manufacturing systems. The different types of software are implemented that optimises and make decisions at all levels of managing the production process. Some of these technologies use artificial intelligence that enables manufacturing systems to learn from previous industrial practice experiences [4].

With the Industry 4.0, the primary goal is to reduce production time and make right decisions in manufacturing systems. IMS can be classified into several levels (Figure 2) [5]:

1. smart design,
2. smart machines,
3. smart monitoring,
4. smart management,
5. smart scheduling.

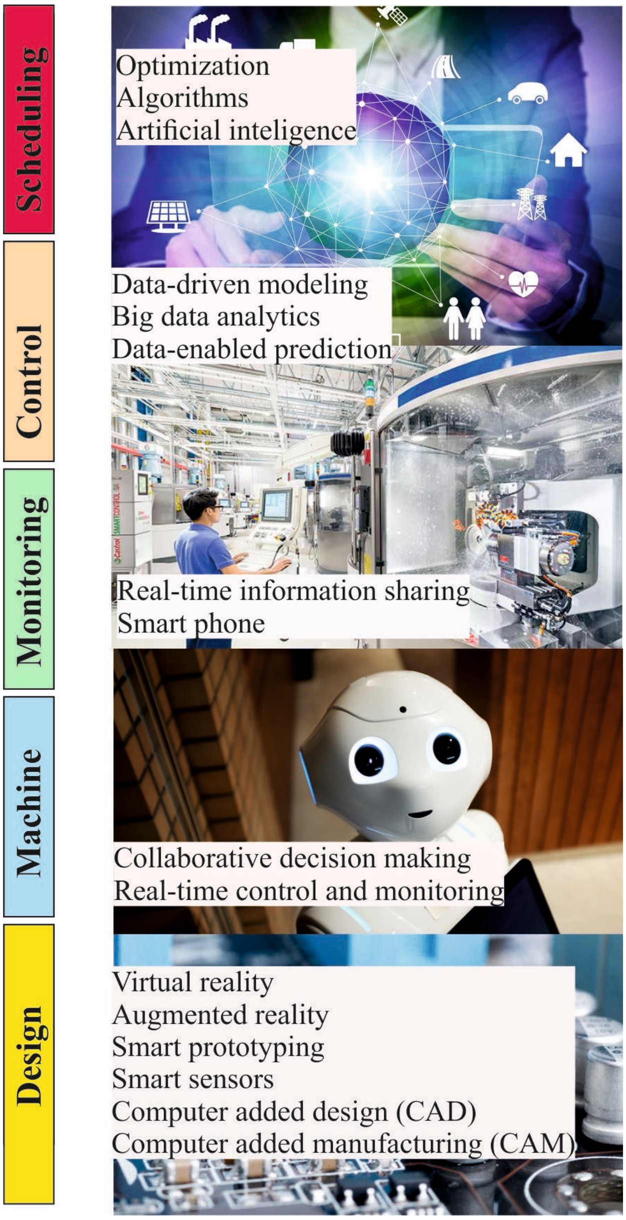


Figure 2: The infrastructure of the industry 4.0 IMS

3. OPTIMIZATION ALGORITHMS FOR SCHEDULING

With the development of the new generation 4.0 industry, the main goal is to optimize makespan time and reduce all costs (manufacturing, equipment maintenance, cost of labour work, etc) [6]. In the first step, it is necessary to optimize the execution of jobs performed on specific machines. Which algorithm will be used for optimization depends on a number of factors which could be analysed. Increased complexity of manufacturing systems with a large number of machines and workstations where various operations are performed have influenced the scheduling algorithms to be intensively developed and implemented over the last few decades. The basic classification of these algorithms is into approximate methods and optimization methods (Figure 3) [7].

The application of a particular algorithm depends primarily on the complexity of the problem which is being solved. The authors of [6,7,8] applied different algorithms to optimize solution in terms of planning and scheduling jobs that are performed in manufacturing systems.

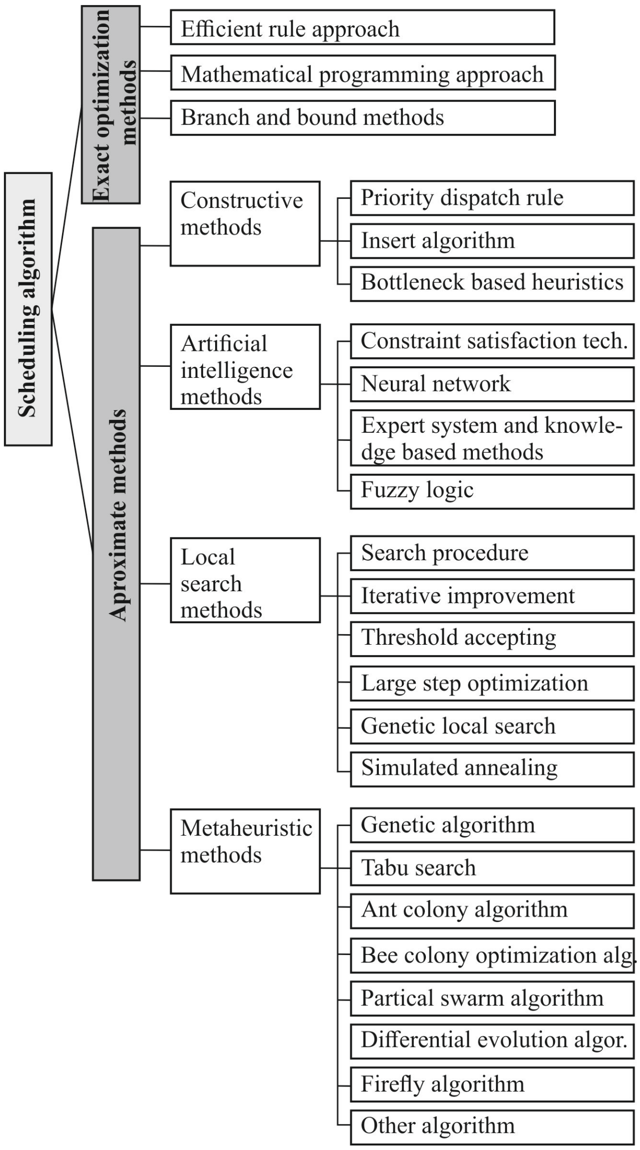


Figure 3: Optimization algorithms for scheduling

4. REVIEW OF SCHEDULING SOFTWARE

4.1. Description

Over the past few decades, a large number of software products have been developed that is used in the process of planning and scheduling complex jobs. Some of them were primarily for educational purposes, while others were commercial. The following software products are most commonly applied [9]:

- SAP,
- IBM scheduling system,
- AMD scheduling system,
- ORTEMS' Agile Manufacturing Suite,
- Asprova,
- Preactor,
- MS Project,
- LEKIN.

We will present the presentation of the LEKIN scheduling system since it has a free version in addition to the commercial version.

4.2. LEKIN system for scheduling

The LEKIN system is used for scheduling in production or service industry. It was developed as an educational tool for demonstration scheduling theory and applications. Professor Michael L. Pinedo and Xiuli Chao are leading in developing this tool at the end of the 20th century [9]. The free version of this system can be found on web-site [10].

The LEKIN system can handle a number of different scheduling machines environments involves [11]:

1. *Single Machine Scheduling (SMS)* is the process of assigning a group of jobs to a single machine. The jobs are arranged so that one or many performance measures may be optimized.

2. *Parallel Machines Scheduling (PMS)* is a process where more than one machine is available for processing the jobs. It does not matter which machine a job is assigned to, but it cannot be processed on more than one machine at the same time.

3. *Flow Shop Scheduling (FSS)* is a process where a strict set sequence of m number of jobs operations and n number of machines. Minimal downtime and minimal waiting time are the constraints in the continuous flow of processes. Production facilities are generally found to be using flow shop scheduling problems. A scheduling problem for flow shop is a generalized version of the problem for job shop scheduling of flexible manufacturing systems. Here each machine has the ability to perform more than one operation for a particular job.

4. *Flexible Flow Shop Scheduling (FFSS*) is a problem where is possible duplication of the machine and the machines are grouped into the workstations where the product of the same types are processed.

5. *Job Shop Scheduling (JSS)* is a problem of scheduling n products on m machines in which each product has a predetermined different route to follow, consisting of a series of operations on the machines.

6. *Flexible Job Shop Scheduling (JSS)* is a type of job shop scheduling in which more than one machine of the same type can be used for one job operation.

The LEKIN scheduling system has four interfaces:

1. Job Pool Window
2. Sequence Window
3. Gantt Chart (Schedule) Window
4. Graphical User Interface

The LEKIN system can manage a maximum of 50 jobs, 20 workstations and 100 machines in the free version. The installation of the system is reasonably easy. When LEKIN is run the main menu can be accessed during a scheduling session clicking on "File" and then "Start over". The user then manually enters all machine data and job data. The file with entering data can be modified and save the modified file under a new name. New data set user must enter by selected a machine environment. In the dialog box user will enter basic information about available workstations and the number of jobs to be scheduled. Then another dialog box will appear and must enter detailed workstation information about availability, the number of jobs, set up times etc.

In the third dialog box user has to enter information about the job (priorities or weights, due dates, release dates, processing times of the various operations).

4.3. Job shop scheduling with Lekin

When the set of jobs are processed with different operations in LEKIN software, then according to their due dates processing time jobs will be scheduled to minimize makespan. There are many priority scheduling rules to obtain an optimum sequence. Here we will explain and use some of them [9]:

1. First come – First Serve Rule (FCFS)

With this rule, the job that arrives first is first executed. It is also the simplest way to comply with the rules in the production process because it is fair to the customer. The main disadvantage of this rule is that longer jobs require others to wait which affects the tardiness in delivery and other jobs wait longer.

2. Shortest Processing Time (SPT)

With this rule, jobs that take shorter periods are first executed. The advantage is minimizing the completion time of all operations, as well as minimizing the average number of jobs in the system. The disadvantages are the lack of information about the due time of processing, and longer jobs wait longer for execution.

3. Earliest Due date (EDD)

The job that has the closest deadline is first executed, but if there are more jobs with a deadline then earliest due date jobs are first taken. If the priority of due date jobs is high, the remaining jobs are ignored and the first priority is performed.

4. Critical Ratio (CR) Rule

Jobs that can be divided into time intervals fit into the processing time. First, the jobs with the least maturity and processing time are performed. The job scheduling depends on the relationship (Due Date)/ (Remaining Shop Time), where the remaining processing time refers to the order, setting, starting, waiting time, and moving from one machine centre to another.

The information about each job is added using the Job Pool Window (Figure 4). Jobs performed on machines have the following parameters:

Rls *(Release date - r_j)* - the time when the product operation becomes available.

Due *(Due date - d_j)* - present time to finish job. If the processing of the job is not completed by this time, there will be tardiness. The main objective of scheduling is to minimize tardinesss.

Wght *(Weight - w_j)* - weight coefficient that determines the significance of the job. If the job is more significant, then this coefficient is higher, thus reducing the tardiness in the production because it has a higher priority.

Pr.tm. *(Processing Time- p_j)* – the processing time of the jobs obtained by summing the processing time of all processing operations.

Stat *(Status)* - presents the status of the jobs, which defines the processing time of the operation and is marked with letters A-Z. The user assigns to the program taking into account that operations with the same status have similar processing times.

Bgn – the period of the beginning of job processing.

End – the period of job finishing.

T – total product tardiness.

wT – total product tardiness multiplied by a weight coefficient.

(+)(Route) - the path of the j-th product that determines the order of the machine-building operations. To check the order in which products are created on machines, click + in the Job Pool window.

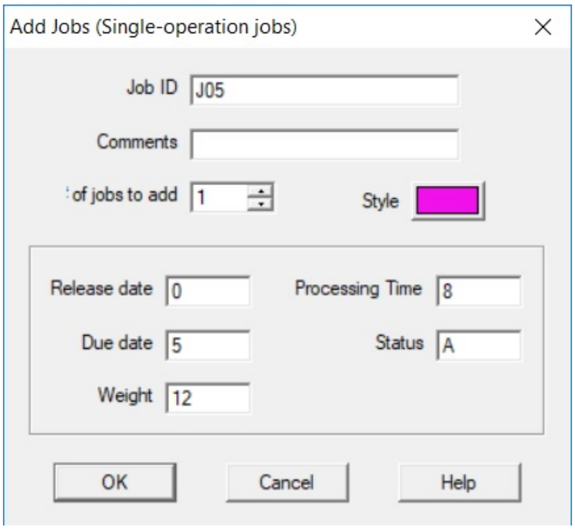


Figure 4: Dialog box for jobs adding

After selecting the appropriate parameters, the user selects commands from the Schedule menu of the scheduling rules, which can be:

- SPT, EDD, ATCS (Apparent Tardiness Cost with Setups), LPT (Longest Processing Time) – easy rules of scheduling,
- Shifting Bottleneck, Local Search - complex heuristics.
- Algorithms which was user-created and available in .exe version.
- User manually enters jobs order by machine.

The criteria functions are used to measure the performance of job shop scheduling problem and create Gantt chart based on the given data for each job and machine [12]. The functions defined in LEKIN are as follows:

- Makespan - Cmax (the processing time of the last job)
- The maximum tardiness - T_{max} (the difference between the actual completion and the expected completion of job)
- The total number of late jobs - $\Sigma\ U_j$
- The total flow time - $\Sigma\ C_j$
- The total tardiness - $\Sigma\ T_j$
- The total weighted flow time - $\Sigma\ w_jC_j$ (total flow time of all jobs multiplied by weight function)
- The total weighted tardiness - $\Sigma\ w_jT_j$ (total tardiness time of all jobs multiplied by weight function)

4.4. The example of job shop scheduling

Using the LEKIN software for scheduling jobs on single machine, it is necessary to enter for each job the ID, date of start of the job, due date, weight factor, processing time, status and style (colour selection to represent the given job in the Gantt chart). In the figure 5 is presented an example of single machine scheduling problem. Five different operations are performed on one machine. Data on each operation were entered and a heuristic method for the calculation of bottlenecks was applied. The appearance of the Gant chart and the shortest tardiness time of all operations were obtained.

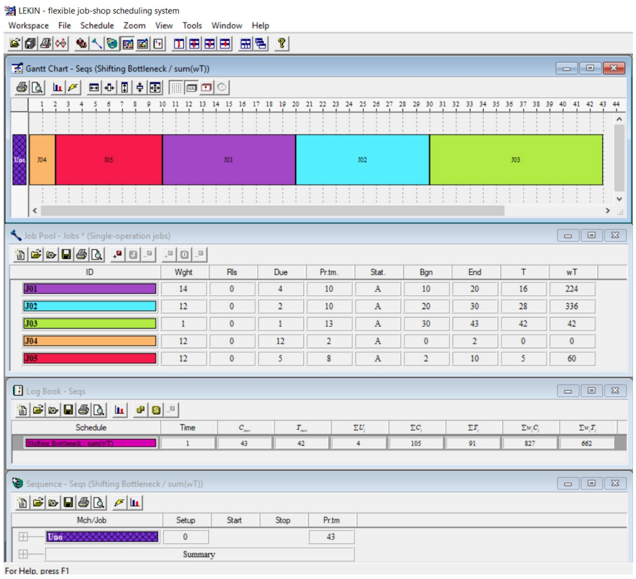


Figure 5: Example of a single machine scheduling problem

The more complex scheduling problem is a job shop scheduling problem. We have presented the scheduling problem with seven work centers and five jobs. In figure 6 is presented Job Pool window for given example. Working centre for given example and SBH have been presented on figure 7. Appling different scheduling rules (CR, EDD and SPT) and SBH of job tardiness data have presented in Log Book (figure 8). In this case, the minimum tardiness of job shop scheduling is obtained with SBH. The Gantt Chart of SBH is present in figure 9.

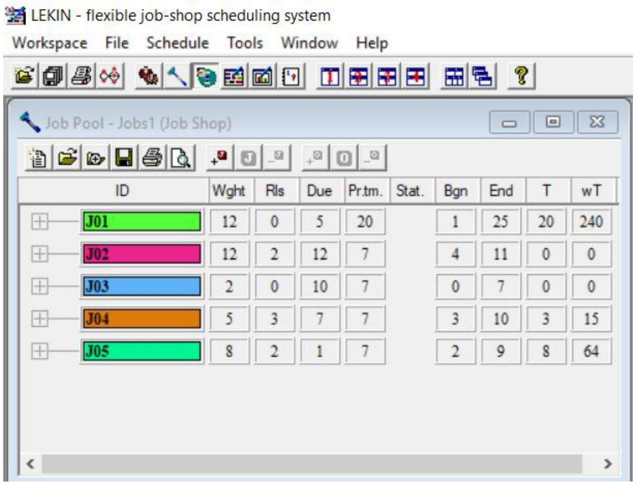


Figure 6: problem Job Pool window for job shop scheduling

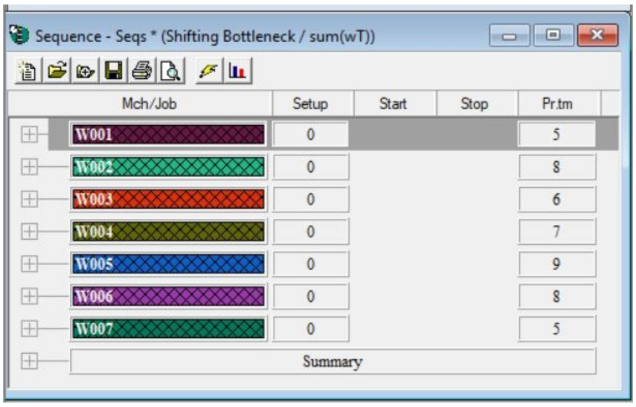


Figure 7: SBH for job shop scheduling problem

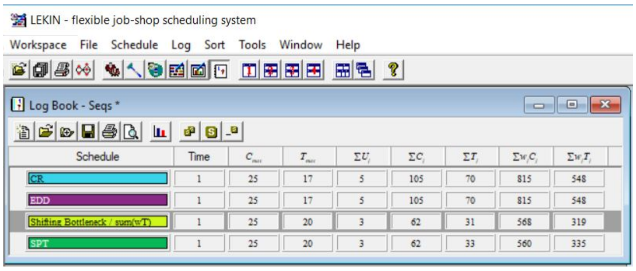


Figure 8: Data analysis in Log Book of job shop scheduling problem

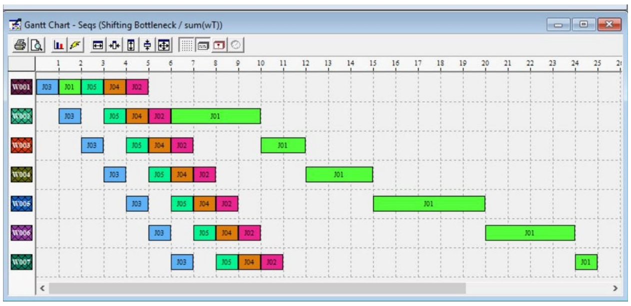


Figure 9: Gantt Chart of job shop scheduling problem

5. CONCLUSION

The application of scheduling algorithms in production systems is significant in optimising the production process. This paper presents the application of the LEKIN software system, which is a significant tool and uses to optimize planning and scheduling in production and service systems. LEKIN has a number of advantages:

- ease of use in education as well as in the process of exploring optimization problems,
- the ability to create an optimal schedule for machine operation,
- uses new heuristic techniques.

Applying scheduling rules and a SBH technique, we obtained results on tardiness times, for example, seven workstations and five types of operations that can be performed on machines. From this example, it can be concluded that the application of the heuristic technique is more efficient in solving the complex problems of scheduling operations on machines.

ACKNOWLEDGEMENTS

The work presented in this paper was funded by grant III46001 and III47003 for the period 2011-2019 by the Ministry of Education, Science, and Technological Development of the Republic of Serbia.

REFERENCES

[1] http://www.isitan.com/en/technology-industry-40-ready.html

[2] A. Nicolae, A. Korodi And I. Silea, “An Overview of Industry 4.0 Development Directions in the Industrial Internet of Things Context”, Romanian Journal Of Information Science And Technology, Vol 22(3–4), pp. 183–201, (2019)

[3] J. Zhang, G. Ding, Y. Zou, S. Qin and J. Fu, “Review of job shop scheduling research and its new perspectives under Industry 4.0”, Journal of Intelligent Manufacturing, Vol. 30, issue 4, No 19, pp. 1809-1830, (2019)

[4] D. Li, Å. Fast-Berglund, & D. Paulin, “Current and future Industry 4.0 capabilities for information and knowledge sharing”, International Journal of Advanced Manufacturing Technology, Vol. 105, pp. 3951–3963, (2019)

[5] Alexandre Dolgui, Dmitry Ivanov, Suresh P. Sethi & Boris Sokolov, “Scheduling in production, supply chain and Industry 4.0 systems by optimal control: fundamentals, state-of-the-art and applications”, International Journal of Production Research, Vol. 57(2), pp. 411-332, (2019)

[6] G. Miodragović, R. Bulatović, S. Ivanović, M. Bošković, “The Use of Biologically-inspired Algorithms for the Optimization of Machining Parameters”, IMK-14 – Research & Development in Heavy Machinery, Vol. 21(3), pp. 93-97, (2015)

[7] M. A. Cruz-Chávez *et al*., "Hybrid Micro Genetic Multi-Population Algorithm With Collective Communication for the Job Shop Scheduling Problem", *IEEE Access*, Vol. 7, pp. 82358-82376, (2019)

[8] H. Chang, Y. Chen, T. Liu and J. Chou, "Solving the Flexible Job Shop Scheduling Problem With Makespan Optimization by Using a Hybrid Taguchi-Genetic Algorithm", in *IEEE Access*, Vol. 3, pp. 1740-1754, (2015)

[9] M. L. Pinedo, “Scheduling-Theory, Algorithms, and Systems”, Springer, p. 670, (2016)

[10] http://web-static.stern.nyu.edu/om/software/lekin/ Last time access 12.12.2019.

[11] M. Dave and K. Choudhary, "Job shop scheduling algorithms — A shift from traditional techniques to non-traditional techniques," 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, pp. 169-173,(2016)

[12] Z. Rakićević and J. Kojić, “Software “Lekin” for solving scheduling problems in planning of production and servicing”, INFO M, Vol. 46, pp. 27-33, (2013)