

# Optimizacija periodične razvrstitve procesov pri robotski manipulaciji

Tomaž KORITNIK, Darko KORITNIK

**Izveček:** Optimizacija urnika, v mednarodni literaturi opisana kot problem Job-Shop, je aritmetično izjemno zahteven problem, ki spada v skupino NP-kompleksnih problemov, kjer časovna zahtevnost glede na število izdelkov in strojev v procesu narašča eksponencialno. Že pri manj obsežnih procesih, npr. 6 izdelkov na 6 strojih, eksaktni postopki zaradi časovne potratnosti odpovedo. Za razvrščanje opravil v obsežnejših procesih so uporabni le aproksimativni postopki, ki pa ne zagotavljajo optimalnosti. Prispevek prikazuje optimizacijo konkretne robotske celice, kjer sta zaradi nenehnega ponavljanja istega procesa za uspešno optimizacijo ključni prekrivanje in periodičnost. Izvedljivost urnika in veljavnost predvidenih taktičnih časov smo pred izvedbo robotske celice preverili tudi s simulacijo, ki je pokazala, da je glede na konfiguracijo procesa urnik optimalen.

**Ključne besede:** urnik, optimizacija, industrijski procesi, robotska manipulacija

## ■ 1 Razvrščanje operacij

V vsakdanjem življenju se s časovno optimizacijo procesov spopadamo praktično pri vsakem delu, v veliki meri tudi nezavedno. Pomagajo nam človeška prilagodljivost, iznajdljivost, izdatnost senzornih informacij, subjektivna presoja pomembnosti posameznih opravil, elastična večopravnost ter takojšnja abstrakcija nepričakovanih zapletov (ali z eno besedo: improvizacija) [1]. Opravila hkrati razčlenjujemo in integriramo, nenazadnje znamo dinamično prilagajati tudi zadane cilje. Še pri dobro načrtovanih opravilih vse naštetu poteka tudi ad-hoc. Inteligentno sprejemanje odločitev v realnem času bolj ali manj dobro prikrije matematično in logično zahtevnost razvrščanja [1]. Ta se pokaže pri kompleksnih procesih, ki morajo biti strogo načrtovani, definirani in deterministični. Takšni so industrijski procesi, ki tam, kjer potekajo, običajno predstavljajo jedro ekonomske dejavnosti.

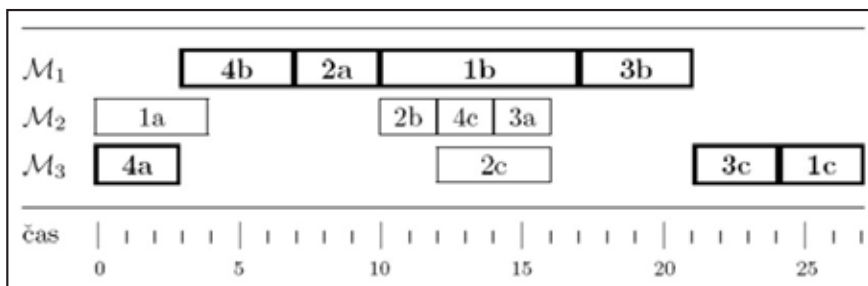
Dr. Tomaž Koritnik, univ. dipl. inž., Darko Koritnik, univ. dipl. inž., oba DAX, d. o. o., Trbovlje

Raziskave na področju optimizacije so tu pomembne zato, ker se ekonomska raba sredstev neposredno in takoj odraža v produktivnosti in posledično na konkurenčnosti [2]. Povedano v kontekstu: če ima nek stroj maksimalno kapaciteto npr. 60 izdelkov na uro, želimo imeti izdelek vsako minuto, tako da se fiksni stroški v času obratovanja porazdelijo na čim več izdelanih enot, ki bodo na trgu zato bolj konkurenčne. Če izdelava poteka na več strojih v več fazah, se zastajanje v eni fazi odraža kot mrtvi čas na vseh ostalih. Cilj je torej s smotrno razvrstitvijo v vsakem trenutku obratovanja dosegati čim večjo zasedenost vseh proizvodnih kapacitet. Glavni problem pri tem je neobvladljivo hitro naraščanje števila možnih razvrstitev glede na število faz in mest njihovega izvajanja.

## ■ 2 Eksponentna rast kompleksnosti

Proizvodni proces opišemo z naborem izdelkov ( $J_1, \dots, J_N$ ) in naborom postaj oz. strojev ( $S_1, \dots, S_M$ ), ki so potrebni v fazah izdelave. Za vsak stroj obstaja  $N!$  možnih razvrstitev. Ker je strojev  $M$ , obstaja v splošnem

$(N!)^M$  možnih rešitev razvrščanja oz. urnikov [3]. Iskanje optimalne rešitve v tej množici spada med NP-zahtevne probleme. To pomeni, da čas, potreben za optimalno rešitev, narašča eksponencialno z dimenzijo problema. Konkretno: problem  $2 \times 2$  ima 4 rešitve,  $3 \times 3$  jih ima 216,  $4 \times 4$  že 331776,  $5 \times 5$  pa že skoraj 25 milijard. Računalnik, ki bi vsako nanosekundo obdelal eno možno rešitev, bi za problem  $6 \times 6$  potreboval 4,4 leta, za problem  $7 \times 7$  pa že 2,6 milijard let. Vseh atomov v znanem vesolju naj bi bilo okrog  $10^{80}$  [4]; če bi vsakemu naredili unikatno ime, jih (psevdo)naključno pomešali, nato pa nekemu naročili, naj med njimi poišče točno določenega, bi to opravil hitreje, kot bi našel optimalno rešitev problema  $11 \times 11$ . Problem  $20 \times 20$  ima že  $5,28 \cdot 10^{367}$  rešitev itd. Ob tem postane jasno, da se moramo odpovedati eksaktnim postopkom razvrščanja in s tem zagotovljeni optimalni rešitvi ter se zaneesti na aproksimativne postopke, ki so toliko manj kompleksni, da so praktično uporabni. Glede na način tvorjenja urnika ločimo konstruktivne in iterativne postopke. Prvi tvorijo urnik od začetka proti koncu, drugi iterativno izboljšujejo obstoječi urnik. Najpogostejši pristopi so



Slika 1. Primer Ganttovega diagrama

omejevanje preiskovalnega drevesa, prioriteta pravila, lokalno iskanje, metahevistika, postopno ohlajanje, iskanje s tabu seznamom in genetski algoritmi. Kreirani urnik običajno nazorno ponazorimo z Ganttovim diagramom, kjer vodoravna os predstavlja časovne enote, navpična os posamezne stroje, okviri pa trajanje operacije na nekem izdelku v določeni fazi (slika 1).

### ■ 3 Optimizacija robotske manipulacije

Prikažimo urnik za konkretno aplikacijo, kjer mora merjenec opraviti več tehnoloških faz z različnimi časi izvajanja na različnih strojih, manipulacijo med stroji pa izvaja robot:

1. odjem merjenca z dovajalnega traku,
2. lasersko označevanje A (5 s),
3. utekanje (100 s),
4. VN-preizkus (12 s),

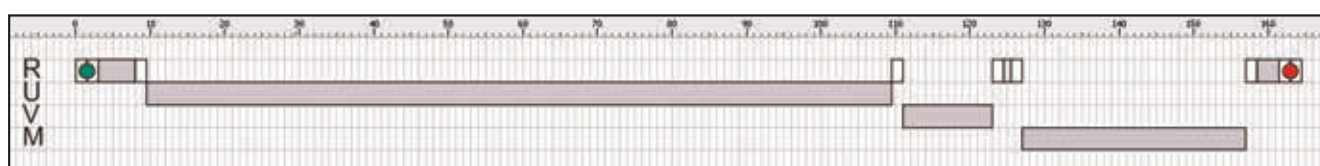
5. magnetni preizkus (< 1 s),
6. magnetenje v magnetilni tuljavi (30 s),
7. lasersko označevanje B (3 s),
8. odlaganje merjenca na odlagalni trak

Taktni čas mora biti krajši od 30 s. Ganttov diagram procesa je prikazan na sliki 2. Robot mora med laserskim označevanjem in magnetnim preizkusom držati merjenec v prijemalu. Ker je R(obot) med fazama na strojih P(reizkus) in L(aser) zaseden, ju združimo na stroj R.

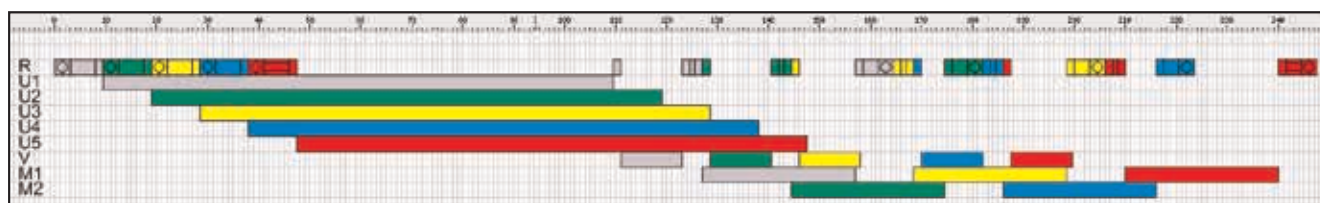
Vsota trajanja tehnoloških faz za posamezen merjenec traja 165 s. Če hočemo zadostiti zahtevi po taktnem času, je jasno, da bo treba procese izvajati vzporedno, s podvajanjem kapacitet. Na sliki 3 vidimo, da stroj R večino časa čaka, v glavnem zaradi strojev U(tekanje) in M(magnetenje). Če kapaciteto U

popeterimo in M podvojimo, brez nadaljnje optimizacije razvrstitve dosežemo takt 5 merjencev v 248 s (cca 50 s na merjenec, slika 3). Še z enim dodatnim strojem U dosežemo takt 6 merjencev v 265 s (cca 45 s na merjenec, slika 4).

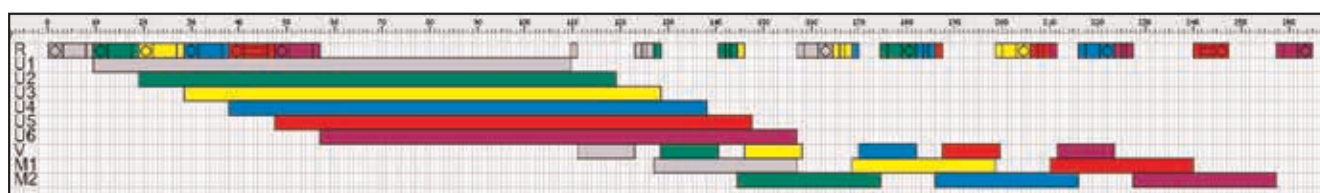
Enostavno podvajanje očitno ni prava pot, saj še vedno opazimo dolge čase nezasedenosti strojev. Pristop k optimizaciji urnika bomo nekoliko spremenili v tem, da na vertikalni osi ne bomo izhajali iz posameznega stroja, temveč iz posameznega merjenca. Pri tem je potrebna večja pozornost nedovoljenemu prekrivanju zasedenosti postaj in predpisanemu vrstnemu redu, saj iz modificiranega diagrama ni več na prvi pogled jasno, kje so kršitve tehnoloških zahtev. Prostor za optimizacijo je v dejstvu, da gre pri proizvodnji za ciklično ponavljanje urnika, kjer pozkušamo z razmikanjem faz po časovni osi doseči začetek naslednjega cikla, še preden je prejšnji dokončan. Omejimo se na periodične rešitve, saj le tako zagotovimo ponovljivost taktnege časa za vsak cikel. Bolj kot trda znanost nam je pri tem lahko v praktično pomoč umetnost. Vzporednice našemu problemu lahko najdemo v ustvarjanju skladatelja J. S. Bacha in slikarja M. C. Escherja. Prvi je znan



Slika 2. Ganttov diagram procesa



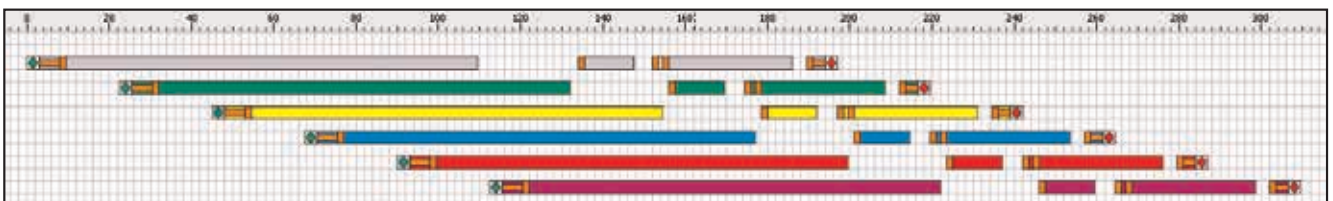
Slika 3. 5 x stroj U, 2 x stroj M



Slika 4. 6 x stroj U, 2 x stroj M



Slika 5. Časovni diagram kanonične glasbene forme – fuge



Slika 6. Časovni diagram glede na izdelek

po svojih številnih fugah in kanonih, ki jih lahko za naše potrebe opišemo kot časovno zamaknjeno prepletanje glasbene teme same s sabo v več vzporednih izvajanjih. Glasbena tema je analogija urniku, zahteva po ustreznem prekrivanju zasedenosti strojev pa je analogna zahtevi po harmonični ustreznosti. Izziv tu ni čim hitrejša izvajanje, temveč napisati temo, ki bo omogočila čim več vzporednih izvajanj. Na *sliki 5* je del ene izmed tripartitnih Bachovih fug, *slika 6* pa prikazuje izsek periodičnega urnika, kjer je vsak izdelek prikazan s svojo barvo. Medsebojna grafična analogija obeh slik izhaja iz kanoničnosti postopka v obeh formah. Na *sliki 7* je ena od Escherjevih ilustracij: če je bil cilj pokazati, kako na list papirja spraviti čim-



Slika 7. Periodičnost in prekrivanje v dveh dimenzijah

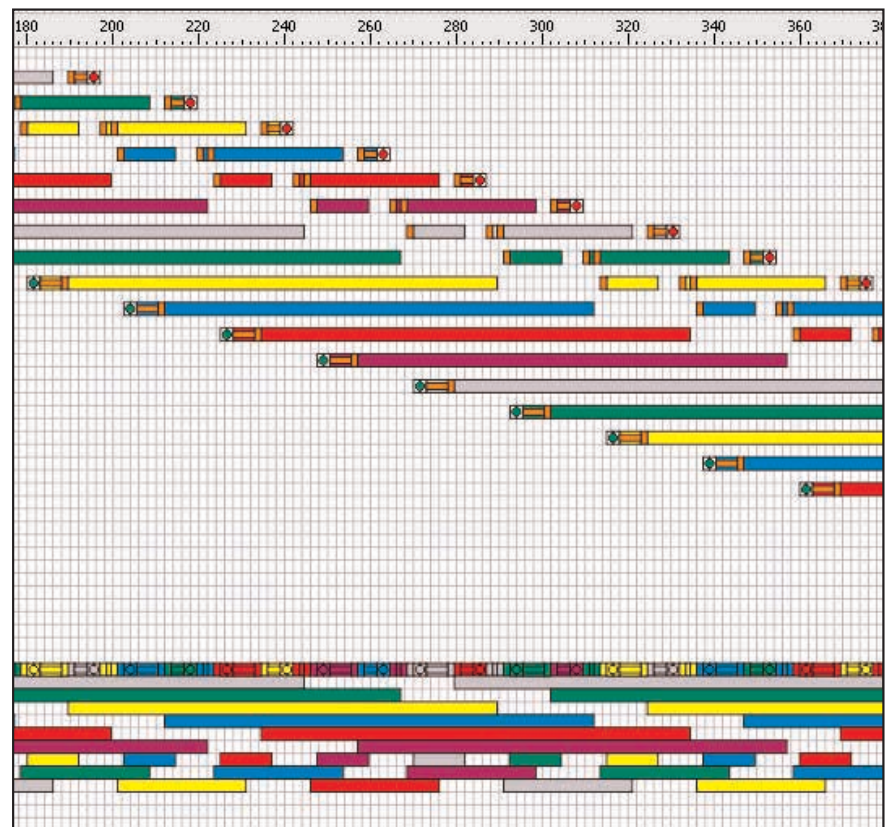
več ptic (analogija s čim več ponovitvami takta v časovni enoti), avtor ni šel v smeri čim bolj natlačenih figur, marveč je z razmikanjem po ravnini (analogija z razmikanjem po časovni osi) pustil med njimi ravno toliko pro-

stora pravih oblik (analogija z omejitvami glede zasedenosti strojev), da ta prostor definira še enkrat toliko ptic.

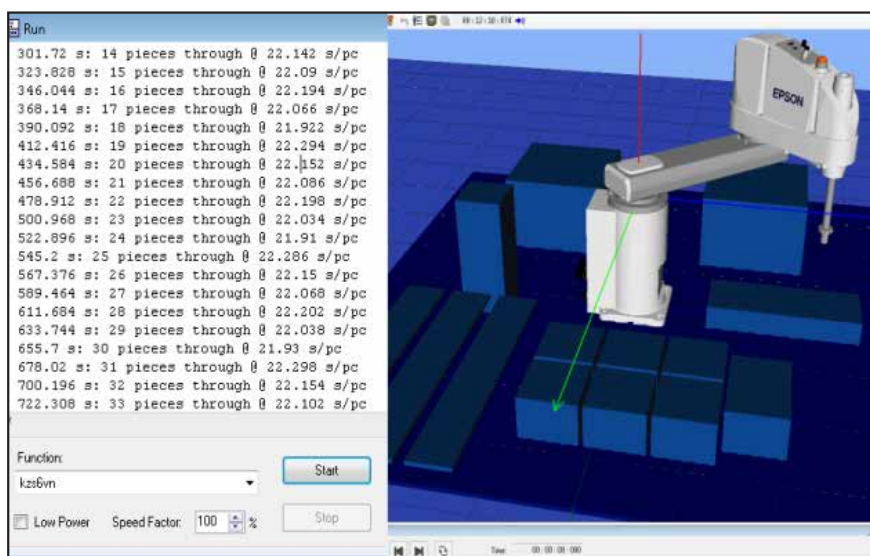
Urnik na *sliki 6* zagotavlja taktni čas 310 s za 6 izdelkov, kar je sicer slabše od izhodišča, vendar njegovo smiselnost pokaže *slika 8*: cikel naslednjih 6 izdelkov se ne začne po 310 s, temveč periodično vsakih 135 s, s čimer dosežemo taktni čas 22,5 s, s čimer dosežemo vsakih 8–9 merjencev, klasični Ganttov diagram (*slika 8* spodaj) pa pokaže 100-odstotno zasedenost stroja R, kar pomeni, da smo glede na dano konfiguracijo dosegli optimalen urnik.

#### 4 Simulacija izvajanja urnika

Izvedljivost končnega urnika smo preverili tudi s simulacijo v robotskem programskem okolju. Dina-



Slika 8. Izsek optimalnega urnika, spodaj Ganttov diagram



Slika 9. Simulacija procesa in izvedbe urnika

mična parametrizacija hitrosti gibanja robota je pokazala še to, da je optimalen taktni čas dosežen že pri 43 % maksimalne dinamike robota; zmanjšanje hitrosti in pospeška pomeni tudi povečanje taktnege časa, povečanje teh vrednosti pa časa ne skrajša, kar pomeni, da robot ni ozko grlo procesa, čeprav je od strojev najbolj zaseden.

Simulacija pokaže, da je urnik koherenten (taktni cikli s časom ne divergirajo) in da so variance taktnih časov zaradi različno dolgih robotskih gibov za posluževanje posameznega stroja v povprečju zanemarljive.

## ■ 5 Zaključek

Pokazali smo, da problematika optimizacije razvrščanja z dimenzijo pro-

blema hitro postane tako zahtevna, da za najsplošnejše probleme, večje od 6 izdelkov na 6 strojih, enostavno ni uporabnih postopkov razvrščanja, ki bi zagotovili optimalne rešitve, na voljo imamo le eksakten opis problema. Nedoseganje zahtevanih taktnih časov se večkrat rešuje s podvajanjem kapacitet, in ne z ustrezno optimizacijo urnika. V procesih, v katerih je podvajanje proizvodnih kapacitet zaradi ekonomike zelo kritično, imamo na voljo omejen nabor metod optimizacije, pa še izmed teh jih velik del postane neuporaben z naraščajočo kompleksnostjo in specifičnostjo tehnoloških zahtev. Vedno pa ostaja motiv, da ne vemo, ali je neka rešitev optimalna, in da jo morda lahko izboljšamo. Ker so časovni prihranki tu kumulativni, je lahko ekonomski učinek precejšen. Niso pa industrijski

procesi edino področje optimizacije. Prihranki v vseh pojavnih oblikah so možni na zelo različnih področjih, pomislimo tu na urnike v šolah, obilo mrtvega časa v logistiki, špediciji in storitvah, nenazadnje na čakalne dobe v zdravstvu. Skupno vsem področjem je to, da se običajno ne zavedamo zahtevnosti razvrščanja opravil, dokler ne naletimo na jasno definiran problem. Tudi na področju razvrščanja so simulacijska orodja zelo uporabna, saj spodbujajo tudi lastno iznajdljivost in kar je še človeških lastnosti, ki smo jih naštevali v uvodu.

## Literatura

- [1] Stein E. W.: Improvisation as Model for Real-Time Decision Making, v: Supporting Real Time Decision-Making, Annals of Information Systems 13; pog. 2, Springer Science and Business Media, 2011. DOI 10.1007/978-1-4419-7406-8\_2.
- [2] Seidel, R. H. A., Arndt, G.: Productivity Improvement in Job Shop Production, CIRP Annals – Manufacturing Technology, Vol. 37, No. 01, 1988, 421–424.
- [3] Ploydabi, K., Mungwattana, A.: Algorithm for Solving Job Shop Scheduling Problem Based on machine availability constraint, International Journal on Computer Science and Engineering, Vol. 02, No. 05, 2010, 1919–1925.
- [4] <http://www.wolframalpha.com/input/?i=how+many+atoms+in+the+universe&lk=4>.

### Optimization of periodic job-shop scheduling in robot manipulation

**Abstract:** Schedule optimization of industrial processing, generally referred to as Job-Shop problem, involves exceptionally demanding arithmetic which puts it among NP-hard problems, where time consumption grows exponentially against problem size. Exact solving procedures fail to be useful for all but the smallest job-shop problems. Even modest processes defined by 6 machines and 6 jobs or larger need to be addressed by using approximative scheduling procedures which perform faster but do not guarantee the optimal solution. The paper presents time-optimization of a given robot cell where overlapping and periodicity are crucial to achieve shorter cycle times of a continually repeating process. Before commissioning the actual cell, feasibility of the resulting schedule and validity of predicted cycle times were tested in a simulation environment where optimality of the process solution was confirmed.

**Keywords:** job-shop, schedule, optimization, industrial process, robot manipulation