

Model određivanja konstrukcijskih parametara u timskom radu pomoću Petrijevih mreža

Vlah, Lorena

Master's thesis / Diplomski rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:235:773575>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-03-07**

Repository / Repozitorij:

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Lorena Vlah

Zagreb, srpanj 2017. godina.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

**MODEL ODREĐIVANJA
KONSTRUKCIJSKIH
PARAMETARA U TIMSKOM
RADU POMOĆU PETRIJEVIH
MREŽA**

Mentor:

Prof. dr. sc. Neven Pavković, dipl. ing.

Student:

Lorena Vlah

Zagreb, srpanj 2017.

Izjavljujem da sam ovaj rad izradila samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem se obitelji na podršci tijekom studiranja i kolegama s fakulteta na nezaboravnim trenucima, timskom radu i grupnom učenju te mentoru prof. Pavkoviću na korisnim savjetima i usmjeravanju u pisanju diplomskog rada.

Lorena Vlah



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE



Središnje povjerenstvo za završne i diplomske ispite
 Povjerenstvo za diplomske ispite studija strojarstva za smjerove:
 procesno-energetski, konstrukcijski, brodstrojarski i inženjersko modeliranje i računalne simulacije

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa:	
Ur. broj:	

DIPLOMSKI ZADATAK

Student: **Lorena Vlah** Mat. br.: 0036464647

Naslov rada na hrvatskom jeziku: **Model određivanja konstrukcijskih parametara u timskom radu pomoću Petrijevih mreža**

Naslov rada na engleskom jeziku: **Model of design parameters determination in team work using Petri nets**

Opis zadatka:

U multidisciplinarnom timskom radu na razvoju proizvoda, koordinacija i komunikacija među članovima tima predstavlja problematiku od primarnog interesa. Istraživanja potvrđuju da današnji računalni sustavi i metodologije upravljanja razvojnim projektima još uvijek ne osiguravaju odgovarajuću potporu ažuriranju i propagaciji informacija u procesu upravljanja konstrukcijskim parametrima.

U ovom radu potrebno je napraviti analizu stanja razvoja metodologija i alata za podršku upravljanju konstrukcijskim parametrima u timskom radu (s naglaskom na problematiku spregnutih parametara). Temeljem analize u daljnjem tijeku rada treba se fokusirati na primjenu Petrijevih mreža, posebno na tzv. obojane Petrijeve mreže za modeliranje dinamike procesa određivanja konstrukcijskih parametara.

Pri tome je potrebno:

1. Analizirati konkretne primjere složenih proizvoda u kojima se vrijednosti skupova spregnutih parametara određuju komunikacijom u timskom radu;
2. Za analizirane proizvode prikazati zavisnosti konstrukcijskih parametara u obliku matrice strukturnih zavisnosti (Design Structure Matrix) i pri tome izdvojiti spregnute parametre;
3. Za skupove spregnutih parametara modelirati simulaciju procesa određivanja njihovih vrijednosti metodom obojanih Petrijevih mreža (Coloured Petri nets). Simulacija procesa određivanja vrijednosti parametara treba uključiti prikaz njihovih zavisnosti, iteracije, pregovaranja i dogovaranja članova tima, te vizualizaciju trenutnog stanja.

U radu je potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan:

11. svibnja 2017.

Rok predaje rada:

13. srpnja 2017.

Predviđeni datum obrane:

19., 20. i 21. srpnja 2017.

Zadatak zadao:

Prof. dr. sc. Neven Pavković

Predsjednica Povjerenstva:

Prof. dr. sc. Tanja Jurčević Lulić

SADRŽAJ

SADRŽAJ	II
POPIS SLIKA	IV
POPIS TABLICA.....	VI
POPIS KRATICA	VII
SAŽETAK.....	VIII
SUMMARY	IX
1. UVOD.....	1
2. Analiza stanja razvoja metodologija i alata za podršku upravljanja konstrukcijskim parametrima u timskom radu	2
2.1. Pregled i analiza literature.....	2
2.2. Petri mreže kao najnaprednija mrežna tehnika za modeliranje <i>Discrete-Event</i> dinamičkih procesa	4
2.3. Modeliranje i simulacija procesa razvoja novog proizvoda na više razina (<i>Multi-level modelling and simulation</i>)	7
2.4. Mrežni pristup povezivanju konstrukcijskih parametara	12
2.4.1. Konstruiranje kao niz transformacija između stanja modela	12
2.4.2. Modeliranje razvoja i procesa za analizu razvoja pomoću Petri mreža	14
2.4.3. Primjena upravljača procesa	15
2.5. Primjena Petri mreža u upravljanju tijekom rada (<i>Workflow Management</i>).....	17
2.5.1. Upravljanje tijekom rada.....	17
2.5.2. Obojene Petri mreže za modeliranje i analiziranje procesa tijekom rada	18
2.6. Tokovi parametara	21
2.6.1. Problemi u komunikaciji.....	21
2.6.2. Modeli procesa razvoja	23
2.7. Upravljanje parametrima, novi pristup u inženjerstvu sustava (Daimler AG).....	25
2.7.1. Trenutni pristupi i novi pristup upravljanja parametrima u inženjerstvu sustava.....	25
2.8. Upravljanje parametrima arhitekture vozila (Daimler AG).....	28
2.8.1. Aktivni lanci u pristupu upravljanja parametrima	28
2.9. Modeliranje dinamike konstrukcijskih parametara pomoću Petri mreža.....	30
2.10. Dimenzije komunikacije u razvoju proizvoda	32
2.11. Modeliranje komunikacijskih aktivnosti više agenata pomoću Petri mreža.....	36
2.11.1. Petri net okvir kao formalna metoda za modeliranje različitih oblika komunikacije u sustavima s više agenata	37
2.12. Istraživanje korelacija između faktora koji utječu na komunikaciju u kompleksnom razvoju proizvoda.....	38

2.13. Modeliranje i simulacija protokola višestranog pregovaranja pomoću obojenih Petri mreža.....	40
3. Obojene Petri mreže i alat CPN tools kao alat za modeliranje i simulaciju dinamičkih procesa pomoću Petri mreža.....	42
3.1. Mogućnosti CPN jezika za modeliranje prikazane na modelu komunikacijskog protokola	43
3.1.1. Korisničko sučelje.....	43
3.1.2. Model komunikacijskog protokola	44
3.1.3. Dodjeljivanje vremena i kreiranje hijerarhijski povezanih modula	46
3.2. Strukture podataka i naprednije mogućnosti korištene u modeliranju mreže za upravljanje konstrukcijskim parametrima u timskom radu	47
4. Složene konstrukcije u kojima su razmatrani tokovi konstrukcijskih parametara	49
4.1. Samohodni uređaj za čišćenje prirubnice spremnika	49
4.2. Robotski manipulator tip SCARA	52
5. Zavisnosti konstrukcijskih parametara prikazani u DSM matrici	55
5.1. DSM (Design structure matrix).....	55
5.2. DSM matrica veza između konstrukcijskih parametara za razmatrane podsklopove proizvoda Samohodni uređaj za čišćenje prirubnice spremnika.....	57
5.3. DSM matrica veza između konstrukcijskih parametara za robotski manipulator tip Scara.....	59
6. Model procesa određivanja vrijednosti konstrukcijskih parametara u timskom radu metodom obojenih Petrijevih mreža.....	62
6.1. Simulacija procesa određivanja vrijednosti konstrukcijskih parametara u timskom radu za razmatrane podsklopove Samohodnog uređaja za čišćenje prirubnice spremnika.....	62
6.2. Simulacija procesa određivanja vrijednosti konstrukcijskih parametara u timskom radu za Scara manipulator.....	70
7. ZAKLJUČAK.....	82
LITERATURA.....	84
PRILOZI.....	86

POPIS SLIKA

Slika 1: Bipartitni usmjereni graf [4]	5
Slika 2: Uvjeti za izvršenje tranzicije su ispunjeni (lijevo), nisu ispunjeni (desno) [24].....	5
Slika 3: Tijek izvršavanja tranzicija i vektorski prikaz stanja [24]	6
Slika 4: Model procesa na visokoj razini [2].....	8
Slika 5: Pretvorba DSM prikaza procesa u DSM mrežu [2]	9
Slika 6: Modeliranje procesa na više razina [2]	10
Slika 7: Primjer primjene pristupa modeliranja procesa s više razina [2].....	11
Slika 8: Razvoj proizvoda kao model transformacija [5].....	13
Slika 9: Hijerarhijsko detaljiranje tranzicije Petri mreže [5].....	14
Slika 10: Mrežni model procesa optimizacije uklještene grede IPE profila opterećene na savijanje koncentriranom silom [5].....	15
Slika 11: Raspoređivanje računalnih procesora i aktivnosti [5].....	16
Slika 12: Trodimenzijski prikaz tijeka rada	18
Slika 13: 4 tipa usmjeravanja procesa [7]	18
Slika 14: Sekvencijalno usmjeravanje [7].....	20
Slika 15: Paralelno usmjeravanje [7]	20
Slika 16: Uvjetno usmjeravanje [7].....	20
Slika 17: Iterativno usmjeravanje [7]	20
Slika 18: Interakcija u hijerarhijskom uređenju formalne komunikacije [3].....	21
Slika 19: Paralelni zadaci koji su koristili parametar i koji ga trenutno koriste [3].....	24
Slika 20: Tok više parametara kojim je prikazana interakcija između različitih parametara koji povezuju isti zadatak [3].....	24
Slika 21: Usporedba <i>Top-Down</i> SE pristupa temeljenih na modelu i pristupa Upravljanje parametrima kao <i>Bottom-Up</i> pristupom [11]	27
Slika 22: Integracija baze podataka parametara u CAD-PDM okruženje [12]	28
Slika 23: Inicijalni CPN model procesa razvoja [6].....	31
Slika 24: Primjer klasične Petri mreže za proces komunikacije (pregovaranja) [8], [9]	37
Slika 25: Mreža korelacijskih koeficijenata [13]	38
Slika 26: Obojena Petri mreža procesa pregovaranja (<i>Negotiation Property Page</i>) [16].....	41
Slika 27: Hijerarhijski prikaz elemenata modela procesa pregovaranja (glavna obojena Petri mreža i pripadne podmreže) [16]	41
Slika 28: Radni prostor aplikacije CPN Tools [15].....	43
Slika 29: Osnovni CPN model komunikacijskog protokola s inicijalnim markiranjem M_0 [15]	45
Slika 30: Index u kojem su deklarirani tipovi podataka, konstante i varijable [15].....	45
Slika 31: Vremenski CPN model protokola s inicijalnim markiranjem M_0 [15].....	46
Slika 32: Uređaj za čišćenje prirubnice sa silama koje na njega djeluju [21].....	49
Slika 33: Klizač mehanizma za poliranje prirubnice [21].....	50

Slika 34: Mehanizam za pogon četki [21].....	50
Slika 35: Scara manipulator [23].....	53
Slika 36: Servo aktuator Harmonic Drive [23]	53
Slika 37: Osnovne veze među elementima s obzirom na redoslijed izvođenja zadataka/dobivanja parametara [19].....	56
Slika 38: Particionirana DSM matrica [19].....	56
Slika 39: Tipovi podataka obuhvaćeni DSM-om [22]	56
Slika 40: DSM matrica bazirana na parametrima podsklopa za poliranje i podsklopa za pogon četki uređaja za čišćenje prirubnice spremnika	57
Slika 41: Particioniranje parametara mehanizma za poliranje prirubnice u alatu PSM ₃₂ (lijevo prije particioniranja, desno nakon particioniranja).....	58
Slika 42: DSM matrica bazirana na parametrima Scara manipulatora podijeljenog na dvije grupe parametara	59
Slika 43: DSM matrica grupe parametara konstrukcije i odabira pogona segmenta 2 nakon particioniranja.....	61
Slika 44: DSM matrica grupe parametara konstrukcije segmenta 1 i proračuna manipulatora nakon particioniranja	61
Slika 45: Hijerarhijski prikaz mreže uređaja za čišćenje prirubnice spremnika	62
Slika 46: Glavna obojena Petrijeva mreža procesa određivanja konstrukcijskih parametara u timu od dva člana za podsklopove samohodnog uređaja za čišćenje prirubnice ..	64
Slika 47: Podmreža određivanja parametara mehanizma za poliranje prirubnice – Dizajner 1	65
Slika 48: Podmreža određivanja parametara mehanizma za pogon četki – Dizajner 2	66
Slika 49: Podmreža procesa pregovaranja – uređaj za čišćenje prirubnice spremnika.....	67
Slika 50: Podmreža usklađivanja parametara – uređaj za čišćenje prirubnice spremnika.....	68
Slika 51: Podmreža procesa argumentiranja – uređaj za čišćenje prirubnice spremnika.....	69
Slika 52: Hijerarhijski prikaz mreže Scara manipulatora.....	70
Slika 53: Obojena Petrijeva mreža procesa određivanja konstrukcijskih parametara u timu od dva člana za Scara manipulator	72
Slika 54: Podmreža određivanja parametara za konstruiranje segmenta 2 i odabir motora 2 – Dizajner 1	73
Slika 55: Podmreža procesa modeliranja segmenta 1	74
Slika 56: Podmreža određivanja parametara za konstruiranje segmenta 1 i proračun manipulatora – Dizajner 2	75
Slika 57: Podmreža procesa modeliranja segmenta 1	76
Slika 58: Podmreža procesa pregovaranja – Scara manipulator	77
Slika 59: Podmreža procesa usklađivanja parametara – Scara manipulator	78
Slika 60: Podmreža procesa argumentiranja – Scara manipulator	79
Slika 61: Pozicija lista prošlih i trenutne dužine S2.....	80
Slika 62: Tranzicija donošenja odluke s mogućnošću odabira	81

POPIS TABLICA

Tablica 1: Pregled i analiza literature.....	2
Tablica 2: Elementi Petri mreže	4
Tablica 3: Problemi s kojima se dizajneri susreću u komunikaciji	22
Tablica 4: Dimenzije komunikacijskih situacija [14]	32
Tablica 5: Scenariji interakcije [14]	34
Tablica 6: Strukture podataka, varijable, funkcije i logički izrazi u modelu za upravljanje konstrukcijskim parametrima u timskom radu	47
Tablica 7: Razmatrani zahtjevi i parametri koji se javljaju u promatranim podsklopovima uređaja za čišćenje prirubnice spremnika.....	51
Tablica 8: Razmatrani zahtjevi i parametri koji se javljaju u promatranim podsklopovima SCARA manipulatora.....	54

POPIS KRATICA

Kratika	Puni naziv
PM	Petrijeva mreža
NPD	New Product Development
PLM	Product Lifecycle Management
DSM	Design Structure Matrix
WF	Workflow
BR	Business Rules
WRI-WF	Well-handled with Regular Iterations Workflow nets
DnPDP	Dynamic new-Product Design Process
WFMS	Workflow management systems
SE	System engineering
EO	Engineering Objects
SysML	Systems Modeling Language
SysMT	System Model and Management
CAD	Computer Aided Design
XML	Extensible Markup Language
CPN	Coloured Petri net

SAŽETAK

Razvoj kompleksnih proizvoda potrebno je softverski podržati kako bi informacije o proizvodu bile strukturirane, pohranjene i dostupne dizajnerima koji s njima rade. Te funkcije ostvarene su PLM sustavima, no pokazalo se da PLM sustavi ne podržavaju dinamičko praćenje tokova parametara i komunikacijske procese u razvoju proizvoda. U ovom radu potrebno je kreirati model upravljanja konstrukcijskim parametrima u timskom radu pomoću Petrijevih mreža. Prije analize parametara razmatranih konstrukcija, proučena je literatura te su sažete najbitnije činjenice trenutnog stanja istraživanja. Procesi koji su modelirani su proces konstruiranja samohodnog uređaja za čišćenje prirubnice i proces konstruiranja Scara manipulatora. Razmatrani složeni proizvodi konstruirani su u timskom radu od dva člana. Relacije između parametara prikazane su u DSM matrici te je proveden algoritam particioniranja kako bi se vidjelo koji se parametri određuju sekvencijalno, paralelno i iterativno. Kao alat za modeliranje procesa pomoću obojenih Petrijevih mreža odabran je *CPN Tools*. Navedene su osnovne značajke alata. Kreirani su modeli za upravljanje parametrima procesa konstruiranja razmatranih konstrukcija uz model procesa komunikacije. Simulacije procesa moguće je pratiti uz korisnikov odabir izvršenja tranzicija ili automatski.

Ključne riječi: tokovi parametara, dinamika procesa konstruiranja, timski rad, podrška komunikaciji, DSM, spregnuti parametri, obojene Petrijeve mreže, *CPN Tools*

SUMMARY

The development of complex products needs to be supported by software so that product information is structured, stored and accessible to the designers who work with those information. These functions were realized with PLM systems, but PLM systems did not support dynamic parameter trails and communication processes in product development. In this paper, it is necessary to create a model of design parameters determination in team work using Petri nets. Prior to analyzing the parameters of the considered constructions, the literature was studied and the most important facts of the current research state were summarized. Processes that have been modeled are the process of design a self-propelled flange cleaner and the process of constructing Scara manipulator. Considered complex products are constructed in team work of two members. The relationships between the parameters are shown in the DSM matrix and a partitioning algorithm has been implemented to see which parameters are sequentially, parallelly and iteratively determined. CPN Tools is selected as a modeling tool for the Petri nets. The basic features of the tool are listed. Models for managing the parameters of the design process of the considered structures with the modeling of the communication process were created. Process simulations can be tracked with the user's choice of transition execution or automatically.

Key words: parameter trails, design process dynamics, team work, communication support, DSM, coupled parameters, coloured Petri nets, CPN Tools

1. UVOD

Proces razvoja novog proizvoda je najčešće timski rad koji uključuje inženjere iz različitih područja. Osim upravljanja multidisciplinarnim timovima, izazov predstavlja i upravljanje kompleksnošću proizvoda. Timovi su često geografski distribuirani, te nije jednostavno ostvariti uspješnu suradnju i usklađivanje različitih modula konstrukcije proizvoda. Alat koji danas koriste poduzeća diljem svijeta, a koji je razvijen kako bi podržao upravljanje procesom konstruiranja i ostalim fazama životnog vijeka proizvoda, omogućio pohranu znanja o proizvodu te pružio uvid u sljedivost određenih podataka, jest PLM (*Product Lifecycle Management*). Uočeno je da PLM nema razvijenu podršku komunikaciji i koordinaciji među članovima različitih timova kojima su informacije o stanju proizvoda potrebne trenutno, odnosno da su trenutno ažurne. Provedena su istraživanja u kompanijama koje razvijaju nove kompleksne proizvode u kojima je multidisciplinarni razvoj nužan, ne samo zbog veće kvalitete i bolje ostvarene funkcionalnosti proizvoda, već i zbog bržeg lansiranja proizvoda na tržište, ostvarenog podjelom aktivnosti u razvoju. Proces razvoja proizvoda sklon je čestim promjenama zbog toga što dio znanja o proizvodu nastaje u procesu razvoja, a mogući problemi vode do zastoja u procesu konstruiranja. Stavka koja nije podržana PLM sustavima je upravljanje spregnutim parametrima. Spregnutim parametrima nazivaju se oni čije vrijednosti su međuovisne, tj. promjena vrijednosti jednog parametra zbog konstrukcijskih zahtjeva i ograničenja utječe na vrijednost drugog parametra i obratno. Ukoliko se radi o modularnoj konstrukciji, u kojoj je za razvoj svakog modula zadužen određeni konstruktor, oni moraju pravovremeno biti informirani o spregnutim parametrima te međusobno surađivati i zajedno odrediti vrijednosti tih parametara. Tokovi konstrukcijskih parametara u kompleksnim konstrukcijama podijeljenim u module danas su predmet istraživanja. Česte promjene u procesu konstruiranja novog proizvoda čine dinamiku procesa, a za modeliranje dinamičkih sustava i proučavanje ponašanja sustava razvijene su tzv. Petrijeve mreže. U ovom radu primijenjene su obojene Petrijeve mreže za modeliranje dinamike procesa određivanja konstrukcijskih parametara, od kojih se neki određuju iterativno, a neki su i spregnuti u odnosu na dizajnere koji ih određuju. Potrebno je razraditi uobičajene situacije suradnje dizajnera kako bi donijeli odluku o konačnim vrijednostima spregnutih parametara te ih prikazati u modeliranoj mreži.

2. Analiza stanja razvoja metodologija i alata za podršku upravljanja konstrukcijskim parametrima u timskom radu

U području razvoja proizvoda upravljanje konstrukcijskim parametrima izazovan je predmet istraživanja te potreba za razvojem metodologija i alata proizlazi iz prakse gdje je problem vizualizacije, praćenja vrijednosti parametara i njihovih promjena stalno prisutan. PLM sustavi podržavaju proces razvoja proizvoda, no ne pružaju podršku tipičnim komunikacijskim procesima u dizajnu.

2.1. Pregled i analiza literature

Tablica 1: Pregled i analiza literature

Autor(i)	Naziv znanstvenog članka	Kratak opis
Upravljanje konstrukcijskim parametrima		
Radošević	<i>Petri mreže i njihov odnos prema drugim poznatim mrežnim tehnikama</i>	Petri mreže – definicija, matematički model, primjena i prednosti
Taylor, Karniel, Reich	<i>Multi-level modelling and simulation of new product development processes</i>	Prikazana je metoda koja podržava modeliranje dinamike procesa razvoja novog proizvoda
Christoper, McMahon, Meng Xianyi	<i>A Network Approach to Parametric Design Integration</i>	Petri mreže kao metoda za modeliranje tokova parametara pri razvoju proizvoda, primjer FEM analiza grede IPE profila i konstrukcija koljenastog vratila
Van der Aalst	<i>The application of Petri Nets to Workflow Management</i>	Prikazana je namjena alata za podršku tijekom rada, te su istaknute prednosti obojenih Petrijevih mreža u modeliranju procesa
Flanagan, Eckert, Clarkson	<i>Parameter trails</i>	Istaknuti su problemi u komunikaciji pri razvoju proizvoda te su prikazani mogući modeli tokova parametara
Toepfer, Naumann	<i>Parameter Management, a Novel Approach in Systems Engineering</i>	Predstavljanje pristupa Upravljanja parametrima

Toepfer, Naumann	<i>Management of vehicle architecture parameters</i>	Pristup Upravljanja parametrima i prikaz aktivnih lanaca
Pavković, Juranić	<i>Modeling the design parameters dynamics with Petri nets</i>	Prikazan je pristup modeliranja dinamike procesa konstruiranja uz prethodnu DSM analizu (relacije između parametara, spregnuti parametri) te je istaknut problem komunikacije u dizajnu
Podrška komunikaciji u procesu konstruiranja		
Maier, Kreimeyer, Hepperle, Eckert, Lindemann, Clarkson	<i>Exploration of Correlations between Factors Influencing Communication in Complex Product Development</i>	Analizirani su faktori koji utječu na komunikaciju u razvoju proizvoda
Khosravifar	<i>Modeling Multi Agent Communication Activities with Petri Nets</i>	Prikazana je primjena Petrijevih mreža za modeliranje i praćenje tipičnih komunikacijskih procesa (pregovaranje, uvjeravanje, odlučivanje)
Eckert	<i>Dimensions of communication in design</i>	Provedeni su intervjui u velikim poduzećima te je napravljena analiza dimenzija komunikacije i scenarija interakcije
Obojene Petrijeve mreže i CPN-Tools		
Jensen, Kristensen, Wells	<i>Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems</i>	Prikazane su karakteristike i mogućnosti alata kroz primjer komunikacijskog protokola
Zaitsev, Shmeleva	<i>Simulating of Telecommunication Systems with CPN Tools</i>	Prezentacija <i>CPN Tools</i> značajki na jednostavnijim primjerima te uvod u naprednije <i>CPN Tools</i> mogućnosti
Bacarin, van der Aalst, Madeira, C. B. Medeiros	<i>Towards Modeling and Simulating a Multi-party Negotiation Protocol with Coloured Petri nets</i>	Složen CPN model procesa e-ugovaranja s 20 podmreža

2.2. Petri mreže kao najnaprednija mrežna tehnika za modeliranje *Discrete-Event* dinamičkih procesa

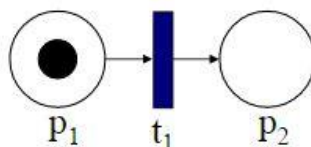
Petri mreže postavio je i prikazao u svojoj doktorskoj disertaciji Carl Adam Petri (1926.-2010.), njemački matematičar koji se bavio računalnim znanostima. Osnovna primjena Petri mreža je modeliranje dinamičkih sustava te proučavanje i analiziranje ponašanja sustava. Petri mreže već imaju široku primjenu u modeliranju informacijskih, upravljačkih i automatskih sustava. U posljednjih dvadesetak godina ova metoda i tehnika počinje se primjenjivati i u računarstvu, elektrotehnici, strojarstvu te se na područjima tih i drugih znanosti vrše daljnja istraživanja, razvijaju teorije i pronalaze primjene. Velika prednosti Petri mreža je što se pomoću njih i elementi sustava i njihova svojstva mogu matematički izraziti i istraživati pomoću računala. Petri mreže omogućuju definiranje raznih stanja i događaja u sustavu bolje od drugih poznatih mrežnih tehnika. Pomoću njih se mogu istraživati paralelni procesi, istovremenost događaja, ciklički procesi, postojanje uvjeta za događaje, smjer mogućeg razvoja procesa, eventualna konfliktnost događaja, vremenski odnosi u sustavu i dr. [4]

Tablica 2: Elementi Petri mreže

	Simbol i oznaka	Engleski naziv	Hrvatski naziv
1.		Place Condition	Pozicija Mjesto
2.		Transition	Tranzicija Prijelaz
3.		Arc	Strelica Luk
4.		Token	Marka

Petri mreže su najnaprednija mrežna tehnika za modeliranje *Discrete-Event* sustava. Petri mreža je bipartitni usmjereni multigraf (bipartitni jer ga čine dva osnovna elementa, multigraf zbog postojanja težina na strelicama koje predstavljaju uvjete za izvršenje tranzicije). Pozicije, tranzicije i strelice čine nepomični dio dijagrama. Pozicija može predstavljati stanja, objekte ili uvjete. Tranzicija može predstavljati događaj, aktivnost ili promjenu. Strelice povezuju pozicije i tranzicije. U četvrtom stupcu Tablice 2 nalazi se marka koja se može pomicati iz pozicije u poziciju prema unaprijed definiranim pravilima. Pokretljivost marki omogućuje da se procesi u mreži mogu simulirati te se njihovi ishodi mogu vizualno pratiti u

mreži. Marka (token) je pomičan simbol koji označava postoje li uvjeti za odvijanje procesa u određenom smjeru. [4]



Slika 1: Bipartitni usmjereni graf [4]

Za razvoj procesa i događaja u nekom smjeru potrebni su uvjeti (materijalni, informacijski, novčani i drugi), ovisno o prirodi procesa i sustava koji se modelira. Ti uvjeti nazivaju se resursi. Da bi se iz jedne pozicije moglo prijeći u drugu (iz jednog u drugo stanje) polazna pozicija mora raspolagati s dovoljno resursa. Postojanje odgovarajućih resursa i njihova količina izražava se brojem marki koje se stavljaju u tu poziciju. Pozicije međusobno uvijek odvajaju tranzicije. Njihova je uloga da dopuštaju prijelaz iz jedne pozicije u drugu samo u slučaju postojanja dovoljnih resursa, tj. tokena u početnoj poziciji. Postojanje dovoljnih resursa za prijelaz preko tranzicije naziva se postojanjem uvjeta, tj. postojanjem mogućnosti za izvršenje tranzicije (*execution*). Tranzicija se u modelu realizira na način da se tokeni iz početnih pozicija premjeste u slijedeće pozicije. Napuštanjem tranzicija dobije se novi raspored tokena po pozicijama te se time stvaraju novi uvjeti za izvršenje narednih tranzicija. U Petri mrežama početno stanje se zadaje početnim razmještajem tokena po pozicijama. Početni razmještaj marki naziva se inicijalno markiranje. Ono predstavlja ishodište analize, daljnjeg razmatranja, odnosno simulacije. Iz početnog markiranja u ovisnosti o uvjetima izvršenja tranzicije može se dobiti jedno ili više novih markiranja. Broj markiranja odnosno broj mogućih novih razmještaja tokena označava broj mogućih smjerova razvoja procesa u sustavu. [4]



Slika 2: Uvjeti za izvršenje tranzicije su ispunjeni (lijevo), nisu ispunjeni (desno) [24]

Mreža se kao matematički model prikazuje kao skup skupova matrica pozicija, tranzicija, ulaznih i izlaznih matrica, težinske matrice te vektora stanja.

$$PM = \{P, T, I, O, \Phi, m_0\}$$

$P = \{p_1, p_2, \dots, p_n\}$ skup pozicija u PM

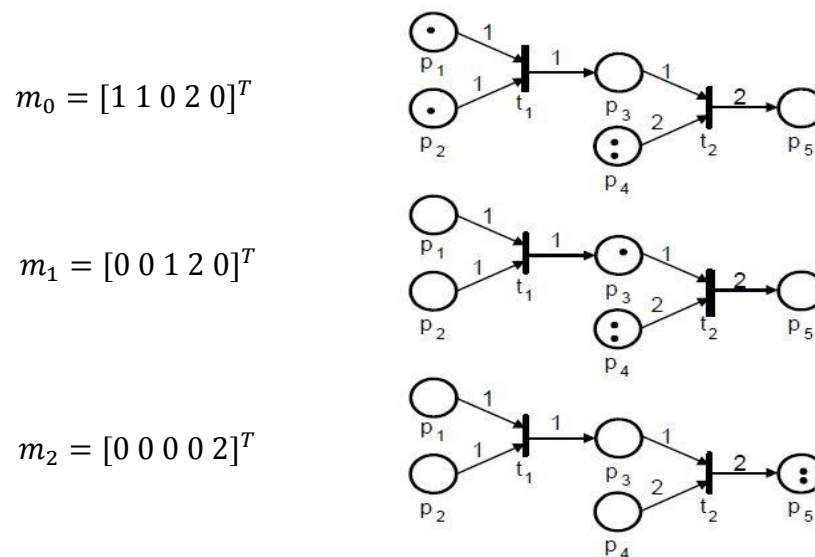
$T = \{t_1, t_2, \dots, t_n\}$ skup tranzicija u PM

$I : P \Rightarrow T$ ulazna matrica događaja

$O : T \Rightarrow P$ izlazna matrica događaja

$\Phi : (I, O) \Rightarrow \{1, 2, \dots\}$ matrica svih težinskih funkcija u PM

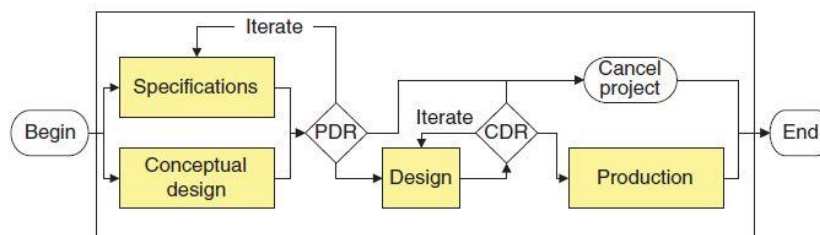
$m_0 : P \Rightarrow \{1, 2, 3, \dots\}$ vektor početnog stanja (*initial marking*)



Slika 3: Tijek izvršavanja tranzicija i vektorski prikaz stanja [24]

2.3. Modeliranje i simulacija procesa razvoja novog proizvoda na više razina (*Multi-level modelling and simulation*)

Na suvremenom globalnom natjecateljskom tržištu, razvoj novih proizvoda je razmatran kao kritični faktor uspješnosti za organizacije. Značajni naponi su uloženi u razvoju metoda i alata za poboljšavanje upravljanja procesom konstruiranja, osobito onih koji su povezani s razvojem novih proizvoda (NPD - *New Product Development*). Upravljanje životnim ciklusom proizvoda (PLM) je skup metoda i alata koji pomažu organizacijama da se nose s povećanom kompleksnošću današnjih proizvoda. One imaju temeljnu pretpostavku da je znanje o proizvodu poznato pa se planiranje procesa razvoja i njegovog procesa konstruiranja može napraviti unaprijed te zatim biti popraćeno kroz pažljivo upravljanje. Veliki broj pogrešaka kod uvođenja novog proizvoda na tržište sugerira suprotno. Pitanja upravljanja uključuju planiranje neočekivane promjene i iterativnog procesa te njihovog modeliranja za implementaciju, izvršenje, prikazivanje i simulaciju. Trenutno dostupni komercijalni alati (*workflow tools* unutar PLM-a) tipično podržavaju predefinirane procese, čija struktura je nepromjenjiva. Postojeće metode planiranja obično podržavaju unaprijed definirane procese znanja. Mnoge metode za modeliranje ne rješavaju iterativne procese. Plan razvoja procesa trebao bi odražavati znanje o proizvodu. Kontekst znanja o proizvodu dinamički se razvija tijekom cijelog procesa, uključujući znanje o proizvodu, zahtjeve, tehnologiju i druge čimbenike. Sve više i više znanja o proizvodu očituje se u oblikovanju promjena prethodno obavljenih aktivnosti. Slijedom toga, znanje o proizvodu trebalo bi uključivati utjecaj promjene izvedbe komponente proizvoda na potrebu redefiniranja dizajna drugih komponenti. Takve iteracije smatraju se glavnim izvorom povećanog razvoja proizvoda, vremena i troškova te je stoga potreban proces planiranja i simulacije. Razultat u praksi je da se planiranje i prikazivanje NPD procesa tipično radi ručno. Na taj način teško je upravljati između uloga konstruktora i projektnih menadžera. Integracija ažuriranih podataka o proizvodu u upravljanju procesom nije moguća te voditelji projekata i konstruktori temelje svoj odluke na prošlim, djelomično zastarjelim podacima o proizvodu, umjesto na nedavno generiranim podacima o razvijenom proizvodu. Zbog tog nedostatka PLM-a, troše se resursi, što potencijalno može prouzročiti neuspjeh projekata. [1]



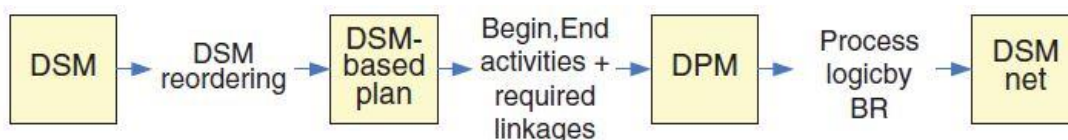
Slika 4: Model procesa na visokoj razini [2]

Kako bi se uklonili ti nedostaci, razvio se pristup modeliranja na više razina za podršku upravljanju razvojem novog proizvoda. Pristup pripada području dinamike procesa razvoja novog proizvoda. Pristup modeliranja procesa na više razina spaja nekoliko metoda za modeliranje procesa, a krajnji rezultat je dobivanje DSM mreže (*DSM net*) spremne za izvršenje i simulaciju procesa. Sustav koji implementira takav pristup sposoban je za modeliranje procesa i izvršavanje procesa prema planovima koji se mijenjaju na temelju znanja o proizvodu. Stoga možemo upravljati iterativnim procesima s dinamičkim promjenama strukture procesa. Simulacije potencijalnih promjena procesa su analizirane koristeći statističku analizu, a rezultati analize koriste se za donošenje odluka tijekom procesa. Kao takav, sustav se može koristiti kao pomoć pri donošenju odluka. Za simulaciju projektnih procesa, naročito NPD procesa, simulacija zahtijeva formalne metode za modeliranje dinamičkih procesnih struktura, tj. procesnih shema. Potpuno rješenje za upravljanje NPD procesima i simulacije zahtijeva planiranje procesa (i ponovno planiranje zbog ažuriranja znanja), dinamičko modeliranje procesa (prema ažuriranim planovima) i implementaciju dinamički promjenjivih procesnih shema, gdje se opseg posla koji je potreban za novi proizvod ne može definirati niti se može unaprijed planirati. Pristup uključuje dvije osnovne metode: *Design Structure Matrix* (DSM) i Petri mreže.

DSM metoda se smatra popularnim istraživačkim alatom za analizu znanja o proizvodu i upotrebljava se za postavljanje projektnih aktivnosti u odgovarajućem slijedu, kako bi informacije o osnovnim projektima bile dostupne na vrijeme. DSM se zatim treba pretvoriti u shemu procesa. Metoda DSM pretpostavlja potpuno znanje o proizvodu, tj. pretpostavlja se da se i planiranje i konverzija u shemu procesa obavljaju jednom. Međutim, u NPD procesima, redoslijed DSM-a (postavljanje modela procesnog plana) i njezino preoblikovanje u shemu procesa treba se ponavljati svaki put kada se znanje o proizvodu promijeni. Tumačenje plana

baziranog na DSM procesnoj shemi nije jedinstveno i njegovo prevođenje u procesni tijek rada (*process workflow*) može dovesti do problema implementacije.

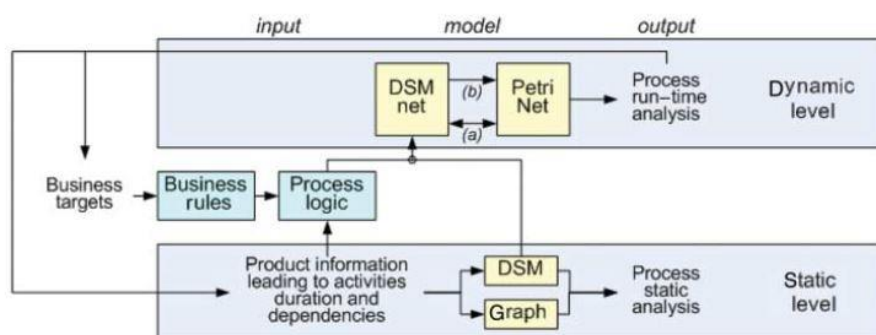
Nedostatak dosljedne metode transformacije plana temeljenog na DSM-u logički ispravnom istovremenom procesnom modelu u slučaju iterativnih aktivnosti, zahtijevao je prilagodbu koncepata verifikacije procesa (temeljenih na WF-mrežama) i razvoj poslovnih pravila. WF mreže, koje su posebne Petri mreže za proces tijekom rada, koriste se u uspostavljenom formalnom modelu za metode određivanja procesa i simulacije. Koriste se kao jamstvo potrebnih svojstava tijekom rada pri implementaciji koja su definirana kriterijima ispravnosti. Mreže radnog toka (*Workflow nets*) kao podklasa Petrijevih mreža, pružaju formalne alate za provjeru svojstava procesa i utvrđivanje kriterija ispravnosti. WRI-WF mreže su podklasa WF mreža koje mogu biti hijerarhijski izgrađene i održavati kriterij ispravnosti. Stoga omogućuju automatizirani pristup građenju procesa temeljen na promjenjivom znanju o proizvodu koji je potreban za simulacije NPD-a. Međutim, WRI-WF mreže koje su sposobne za modeliranje tipičnih procesa temeljenih na DSM-u ne podržavaju modeliranje posebnih slučajeva logike. Poslovna pravila koja su definirana u ovom radu koriste se za rješavanje općenitih logičkih slučajeva i definiranje specifične logike implementacije. Logika poslovnih pravila (BR) definira tumačenje DSM-a prema različitim poslovnim slučajevima i vodi automatsko prevođenje DSM-a na logički iterativni procesni plan. Poslovna pravila rješavaju logički sloj pretvorbe procesa u DSM mreže.



Slika 5: Pretvorba DSM prikaza procesa u DSM mrežu [2]

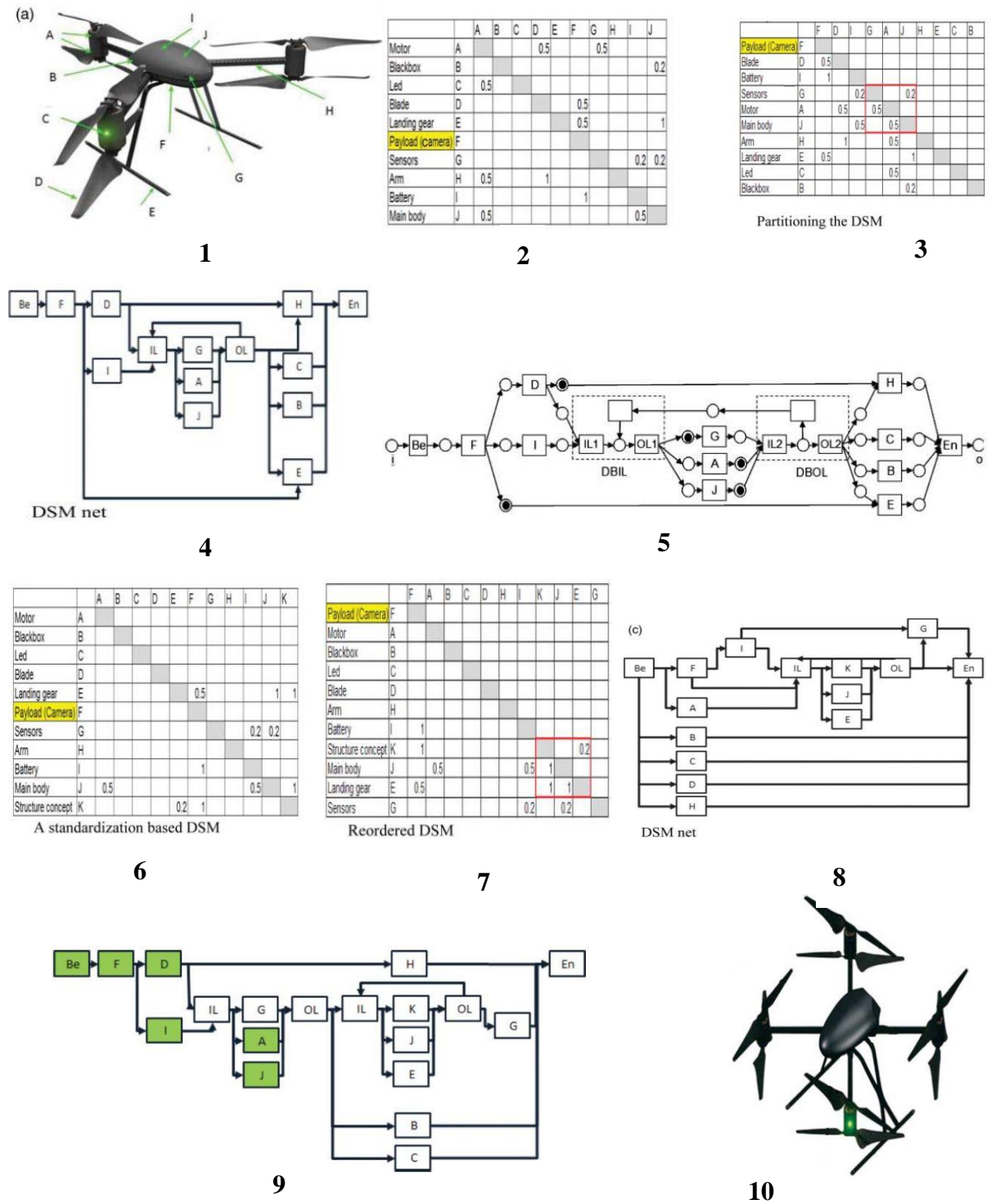
DSM mreže koriste se za modeliranje procesa, koristeći (dinamički modificirane) logičke aktivnosti. DSM mreže se generiraju prema DSM redosljedu u nekoliko faza prevođenja. Dobivena shema procesa plana može se simulirati, izvršiti, i formalno provjeriti. Različite postavke znanja o proizvodu se mijenjaju, što uzrokuje da su različite promjene strukture procesa konstruiranja provjerene, dok su analizirani različiti simulacijski parametri. Glavni nalazi (otkrića) analize studije su da simulacije mogu pomoći u donošenju odluka. Ipak, rezultati simulacije trebaju statističku analizu kako bi se razlikovali značajni rezultati od

beznačajnih. Proširenje i spajanje prezentiranih DSM koncepata i koncepta Petri mreže, omogućilo je integraciju planiranja, modeliranja i implementacije koja može biti automatizirana i može podržavati dinamičke modifikacije procesne sheme. Prikazani integracijski mostovi između DSM i Petri mreža, kao reorganizirani i prošireni DSM mogu generirati WRI-WF mreže.



Slika 6: Modeliranje procesa na više razina [2]

Temeljni zaključak je da se procesi NPD-a mogu upravljati alatima za dinamički tijekom rada (*dynamic workflow tools*), iako se dodatno znanje treba uzeti kao dio tekućeg procesa. Dinamički okvir razvoja novog proizvoda (DnPDP) je integrirani pristup koji uključuje proces planiranja i ponovnog planiranja u vremenu rada uslijed razvoja znanja o proizvodu, dinamičku implementaciju procesa, izvršenje procesa i simulaciju. Poboljšanje PLM sposobnosti da bi se prilagodilo kompleksnosti procesa korištenjem integriranog pristupa (planiranje bazirano na proizvodu, modeliranje procesa, izvršenje i simulacija) moglo bi pomoći u boljem upravljanju NPD procesima. Tvrdi se da bi integracija dinamičkih sposobnosti upravljanja shemama procesa na postojeće alate učinila korištenje računalno podržanih alata za upravljanje ili PLM alata učinkovitijima, a time i prevladavajućima. [2]



1 - inicijalni koncept	5 - implementacija Petri mreže	
2 - DSM matrica ovisnosti parametara	6 - DSM baziran na standardiziranim komponentama proizvoda	
3 - particioniranje DSM-a	7 - particioniranje novog DSM	9 - promijenjena DSM mreža
4 - kreiranje DSM mreže	8 - nova DSM mreža	10 - novi koncept proizvoda

Slika 7: Primjer primjene pristupa modeliranja procesa s više razina [2]

2.4. Mrežni pristup povezivanju konstrukcijskih parametara

Razvoj proizvoda je aktivnost suradnje više inženjera specijaliziranih za određena područja koji manipuliraju svojim vlastitim modelima razvoja kroz niz stanja modela te primjenjuju svoje znanje o konstrukciji i manipulaciji tim modelima. Proces razvoja je karakteriziran preko eksplicitnih i implicitnih atributa (zajedno s ostalim faktorima kao što su su vanjsko opterećenje i ograničenja). Eksplicitni atributi su jasno definirani, opisuju tijek razvoja, analize i testiranja te ih definira konstruktor. Oni utječu na proizvodnju objekta. Početna eksplicitna svojstva modela proizlaze iz zahtjeva specifikacija. Implicitni atributi su procjene ili pojavni atributi, ne trebaju biti poznati za proizvodnju objekta, ali se mogu pojaviti ili biti izvedeni iz eksplicitnih atributa te se koriste za procjenu prikladnosti objekta. Vrijednosti implicitnih atributa su procijenjene tijekom procesa razvoja. Tijekom razvoja konstrukcije razvojni tim radi s brojnim modelima za razne attribute – npr. geometrijski modeli za eksplicitne attribute oblika, modeli očekivanog vijeka trajanja za implicitni atribut dinamičke izdržljivosti itd. Pomoćni modeli mogu biti korišteni kao pomoć u razvoju procjena o implicitnim atributima. Transformacije između modela atributa formiraju mrežu interaktivnih aktivnosti koje mogu biti serijski (slijedno) ovisne, istovremene ili iterativne. Te aktivnosti se modeliraju kao tranzicije između stanja podataka (*data states*), a kao alat za modeliranje mogu se koristiti Petri mreže. One mogu jednostavno biti korištene za prikazivanje tokova razvojnih informacija i manipulacije tim informacijama. Njihov je značaj u automatizaciji i integraciji razvoja proizvoda u tome da mogu prikazivati dinamičke procese. [5]

2.4.1. *Konstruiranje kao niz transformacija između stanja modela*

Razvoj (*design*) “D” nekog proizvoda može biti prezentiran skupovima:

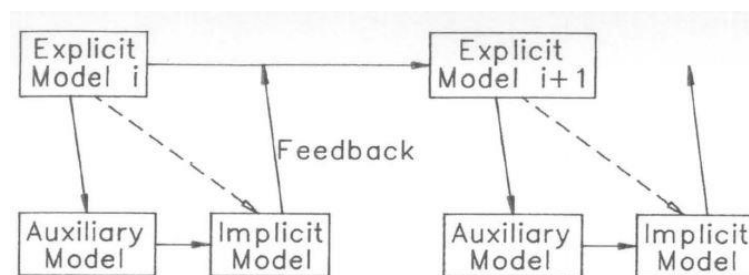
$$D = \{E, L, I\}$$

E – skup eksplicitnih atributa koji opisuju konstrukciju, kao što su dimenzijski parametri, vrijednosti atributa materijala koji čine proizvod i dr.

L – skup vrijednosti vanjskih utjecaja na proizvod kao što je opterećenje

I – skup implicitnih atributa koji opisuju karakteristike i ponašanje proizvoda, a proizlaze iz vanjskih utjecaja L (uključuju parametre kao što su čvrstoća, izdržljivost i dr.)

Zadatak konstruiranja je razviti eksplicitne attribute kao prikaz razvoja proizvoda (kroz dokumentaciju, dijagrame i računalne modele) i pritom razumjeti implicitna svojstva koja nastaju. U prilagodljivom razvoju ili razvoju više varijanti, početni eksplicitni model svojstava je predložen od zahtjeva specifikacija ili odabirom prototipa konstrukcije za daljni razvoj. Razvoj se tada promatra kao slijed transformacija modela eksplicitnih svojstava kako se konstrukcija razvija od inicijalnog stanja. Inicijalni model može imati samo osnovne parametre, dok se ostali parametri progresivno instanciraju i definiraju korak po korak kroz iterativni niz transformacija procesa. Dizajner može odabrati specifične transformacije kod razmatranja ili ispitivanja nastajućih implicitnih atributa, ili dobiti te attribute direktno iz eksplicitnog modela, ili stvarajući pomoćni među-model koji omogućuje konstrukciju modela implicitnih atributa (npr. FEM je pomoćni model korišten za određivanje raspodjele napreznja).

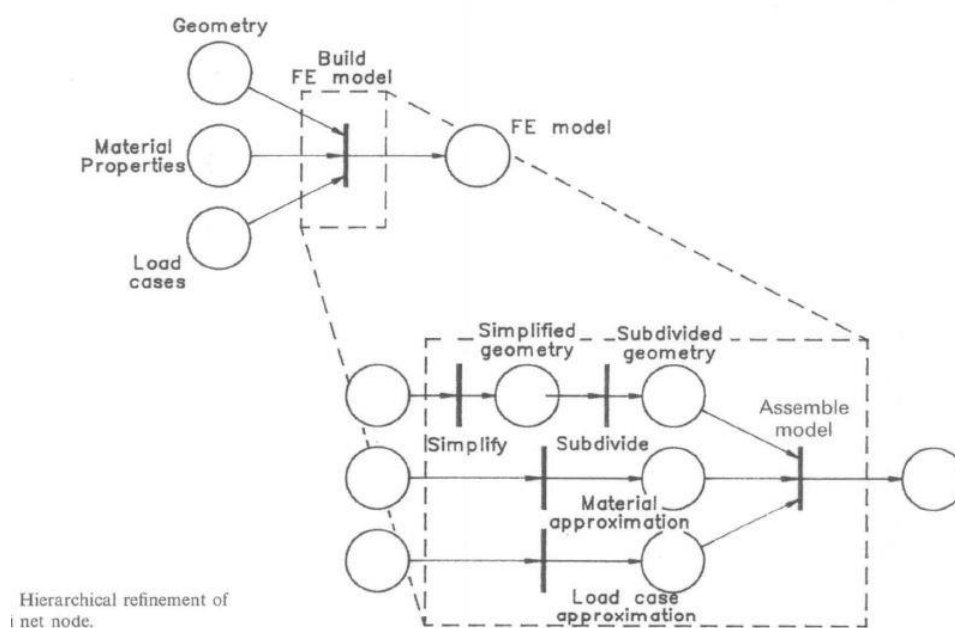


Slika 8: Razvoj proizvoda kao model transformacija [5]

Međutim, često je potrebno primijeniti višestruke procese nad eksplicitnim atributima kako bi se procijenile vrijednosti različitih implicitnih atributa. Npr. kako bi se procijenila otpornost komponente na zamor, detalji geometrije, materijala i nametnutog opterećenja će biti kombinirani s tehnikama za kinematsku i dinamičku analizu, te analizu napreznja, čiji će rezultati biti pohranjeni u algoritme za procjenu životnog vijeka komponente u odnosu na zamor. Zbog toga je prikladnije razmotriti mreže tranformacija između modeliranih atributa, koje se sastoje od dva glavna elementa, onih koji označavaju modelirane attribute i onih koji predstavljaju transformacije. Strelice koje se nalaze između elemenata predstavljaju tok informacija. Pokazalo se da su Petri mreže najprikladnije za modeliranje procesa razvoja proizvoda, jer se njima jednostavno prikazuju podaci i procesi, serijski i paralelni elementi, te se kroz simulaciju može pratiti dinamika procesa u vremenu.

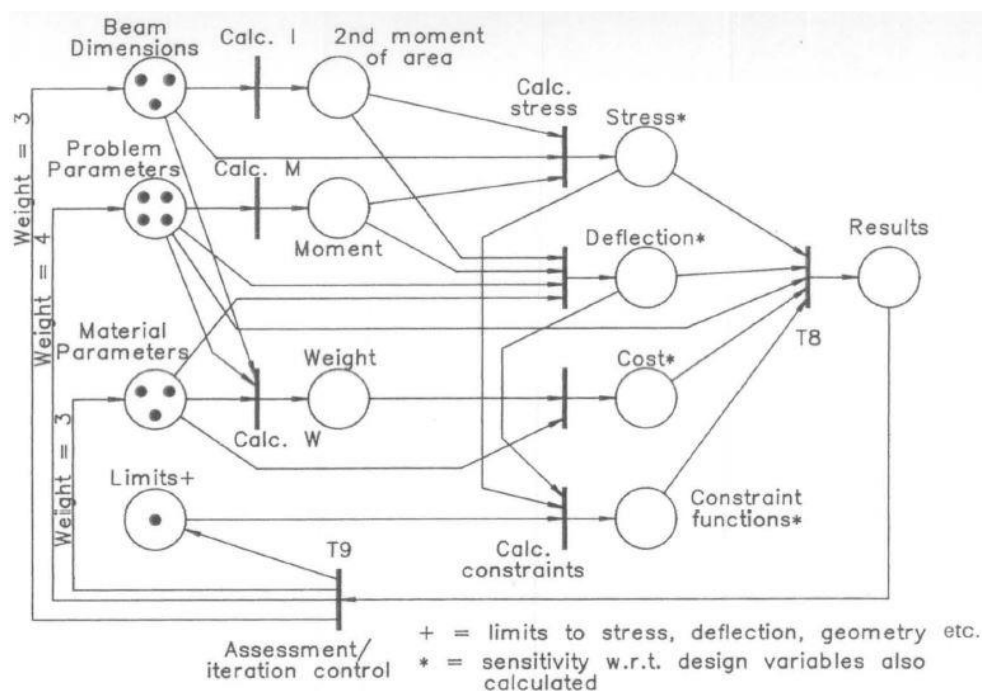
2.4.2. Modeliranje razvoja i procesa za analizu razvoja pomoću Petri mreža

U tehničkoj primjeni, pozicije predstavljaju stanja podataka u razvojnom procesu. Pozicija može predstavljati geometrijski model neke komponente, kolekciju svojstava materijala, matematički model i drugo, dok tokeni označavaju da je određena informacija prisutna u poziciji. Tranzicije predstavljaju operacije koje mogu biti učinjene nad podatkom kada su tokeni postavljeni na ulaznu poziciju.



Slika 9: Hijerarhijsko detaljiranje tranzicije Petri mreže [5]

Na slici 9. ulazne pozicije uključuju geometrijski model, podatke o svojstvima materijala i podatke o slučaju opterećenja koje se odnosi na tu komponentu. Kada su svi zahtijevani podaci prisutni, može se izvršiti tranzicija “*Build FE model*“, a prisutnost izlaznog (*output*) tokena će pokazati da se ta aktivnost uspješno izvršila i predstavlja novo stanje u mreži. Tranzicija “*Build FE model*“ zapravo uključuje više transformacija modela. Vjerojatno je da će geometrijski model korišten u razvoju biti pojednostavljen za analizu i da će se izvršiti aproksimacije kao što su simetrija oko neke osi ili razmatranje 2D modela, umjesto analiziranja kompliciranog 3D slučaja. Na sličan način, slučaj opterećenja i podaci o materijalu će se vjerojatno idealizirati u odnosu na realan slučaj. To se može modelirati koristeći Petri mreže i primjenjujući tranzicije više razine (*high-level*) kako bi se predstavila mreža na nižoj razini.



Slika 10: Mrežni model procesa optimizacije uključene grede IPE profila opterećene na savijanje koncentriranom silom [5]

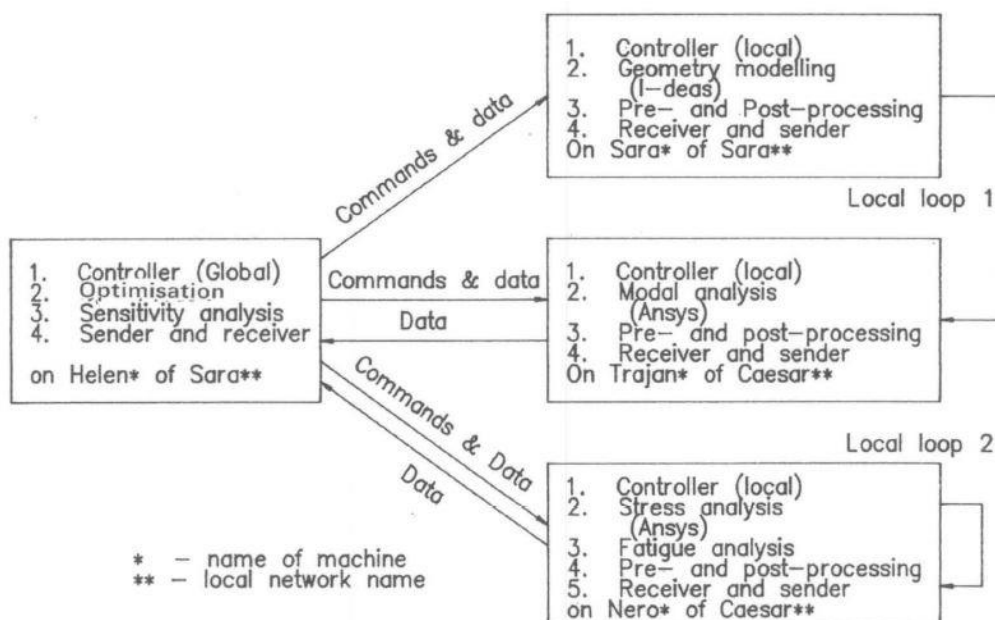
Na slici 10. je dijagram tipa Petri mreže koji prikazuje stanja podataka i transformacije korištene u optimizaciji grede. Prikazano je više tokena u pozicijama koje imaju višestruke veze prema tranzicijama, te su prikazane težine na vezama koje omogućuju nastajanje višestrukih tokena. Strelice iz tranzicije T9 prema pozicijama: *beam dimensions*, *problem parameters*, *material parameters* i *limits* imaju težine kako bi se osiguralo da je dovoljno tokena pozicionirano da se izvrše sve zahtijevane tranzicije. Iteracija je realizirana višekratnim okidanjem tranzicije T9 za svaki korak optimizacije.

2.4.3. Primjena upravljača procesa

U članku je prikazan razvoj koljenastog vratila automobila te je nastanak vratila opisan kroz tri faze. Prva faza definira dimenzije ograničenja prostora, druga dobivanje okvirnih proporcija, a zadnja konstrukciju dorađenu do detalja. Zatim je modelirana pojednostavljena Petri mreža procesa razvoja koljenastog vratila kako bi se mogla implementirati u upravljačku jedinicu (*controller*).

U razvoju koljenastog vratila najčešće sudjeluje najmanje dva dizajnera specijalizirana za određena područja. Svaki od njih treba izdvojiti podatke modela zahtijevane od drugih, te je primijećeno da se mnogo vremena troši na izdvajanje tih podataka. Uočeno je da se taj proces

može automatizirati jer su komponente parametrizirane. Pomoću Petri mreža razvijen je program koji simulira paralelne aktivnosti kojima koordinira interakciju između proračunskih procesa korištenih u analizi procesa. Upravljač automatski upravlja slijednim i paralelnim izvršenjima različitih procesa. Za pripajanje dodatnog procesa, kako bi se ispitali rezultati analize i modificirali ulazni parametri, uspostavljena je automatska optimizacija multiprocesora. Inicijalizacija mreže je ostvarena tako da upravljač učitava ulazni dokument koji je korisnik pripremio. Multiprocesorske operacije su postignute postavljanjem skupa kontrolera, po jedan na svakoj radnoj stanici, što doprinosi bržem izvršenju mreže. Glavna upravljačka jedinica upravlja cjelokupnim nizom izvršenja tranzicija procesa i komunicira sa upravljačima na svakoj radnoj stanici.



Slika 11: Raspoređivanje računalskih procesora i aktivnosti [5]

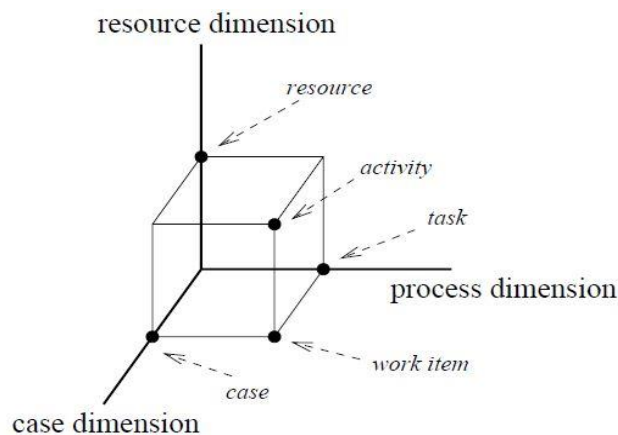
2.5. Primjena Petri mreža u upravljanju tijekom rada (*Workflow Management*)

Informacijski sustavi u današnje vrijeme kontroliraju, prikazuju i podržavaju logističke aspekte poslovnih procesa. Današnji poslovni sustavi isporučuju veliku količinu proizvoda i usluga, što rezultira time da se broj radnih procesa unutar tvrtki značajno povećao. Životni vijek proizvoda se u zadnjih nekoliko desetljeća smanjio, poslovni procesi su skloni čestim promjenama, a složenost procesa se znatno povećala. Navedene promjene okruženja u prosječnim poslovnim organizacijama ukazuju na potrebu za konceptima, tehnikama i alatima za podršku upravljanju tijekom rada što predstavlja bitnu značajku u razvoju današnjih informacijskih sustava. Sustav upravljanja tijekom rada (*WFMS - Workflow management system*) je softverski alat koji omogućuje definiranje, izvršenje, registraciju i kontrolu procesa.

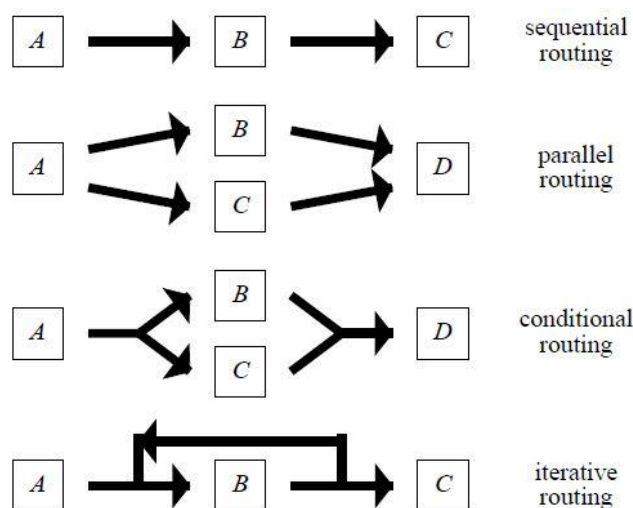
2.5.1. Upravljanje tijekom rada

Tijek rada temeljen je na slučaju (*case-based*), svaki dio rada izvršen je za specifičan slučaj. Slučajevi se obrađuju izvršavanjem zadataka određenim redoslijedom. Da bi se zadaci izvršavali određenim redom, nužno je definirati uvjete koji odgovaraju uzročnim ovisnostima između zadataka. Uvjet može biti ispunjen ili neispunjen (*true or false*). Svaki zadatak ima preduvjete za izvršenje i postuvjete koje određuje to izvršenje.

Tijek rada ima tri dimenzije: (1) dimenzija slučaja, (2) dimenzija procesa i (3) dimenzija resursa. Dimenzija slučaja označava činjenicu da su svi slučajevi obrađeni individualno. Sa stajališta tijeka rada, slučajevi ne utječu direktno jedni na druge već utječu jedni na druge indirektno putem dijeljenja resursa i podataka. U dimenziji procesa specificirani su zadaci i usmjeravanje zadataka. Identificirana su 4 tipa usmjeravanja: sekvencijalno (izvršenjem jednog zadatka slijedi drugi zadatak), paralelno (dva ili više zadataka izvršava se u isto vrijeme), uvjetno (izvršenje ili prvog ili nekog drugog zadatka), iteracijsko (izvršavanje nekog zadatka više puta).



Slika 12: Trodimenzijski prikaz tijeka rada



Slika 13: 4 tipa usmjeravanja procesa [7]

2.5.2. Obojene Petri mreže za modeliranje i analiziranje procesa tijekom rada

Petri mreže su dobro utemeljena tehnika modeliranja procesa, koje je šezdesetih godina osmislio Carl Adam Petri. Od tada su korištene za modeliranje i analiziranje raznih procesa za primjenu od protokola, proizvodnih sustava, interakcije među korisnicima do poslovnih procesa. U zadnja dva desetljeća klasične Petri mreže proširene su s bojom, vremenom i hijerarhijom. Ta proširenja olakšavaju modeliranje kompleksnih procesa gdje su podaci i vrijeme važni faktori. Procesu tijekom rada modelirani pomoću Petri mreža imaju jasnu i preciznu definiciju jer je semantika klasičnih Petri mreža i nekoliko poboljšanja (boja, vrijeme, semantika) definirana formalno. Petri mreže omogućavaju primjenu raznih tehnika analize. Te

tehnike mogu se koristiti da dokažu svojstva (nepromjenjivost svojstava, zastoji i dr.) te da se izračunaju procjene performansi (vrijeme odgovora, vrijeme čekanja i sl.). Moguće je otkriti alternativne tijekove rada koristeći standardne alate za analizu.

Klasične Petri mreže omogućuju modeliranje stanja, događaja, uvjeta, sinkronizacije, paralelizma, izbora i iteracije. No, realni procesi koje Petri mreže opisuju skloni su biti kompleksni i ekstremno veliki. Osim toga, klasične Petri mreže ne omogućuju modeliranje strukture podataka i vremena. Tri dobro poznata proširenja osnovnih Petri modela su: proširenje s bojom za modeliranje podataka, proširenje s vremenom i proširenje s hijerarhijom kako bi se strukturirali veliki modeli.

- **Proširenje s bojom**

Tokeni najčešće predstavljaju objekte (resurse, dobra, ljude) u modeliranom sustavu. Ponekad je potrebno postaviti attribute na te objekte (npr. ime, broj, datum, iznos). Zbog toga se Petri mreže proširuju s bojom ili tipom podatka, odnosno tokena. U obojenim Petri mrežama svaki token ima vrijednost koja se naziva "bojom". Tranzicije određuju vrijednosti proizvedenih tokena na osnovi vrijednosti potrošenih tokena, odnosno tranzicije opisuju relaciju između vrijednosti ulaznih tokena i vrijednosti izlaznih tokena. Moguće je specificirati preduvjete koji zahtjevaju ispunjenje boje tokena kako bi ga uzeli u obzir.

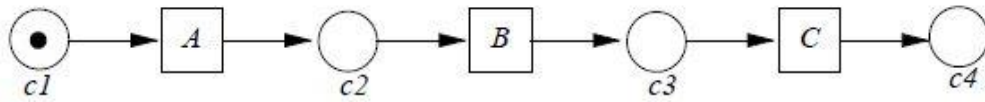
- **Proširenje s vremenom**

Za realne sustave često je važno opisati vremensko ponašanje sustava, odnosno potrebno je modelirati trajanje i kašnjenja. Vrijeme može biti povezano s tokenima, pozicijama i tranzicijama.

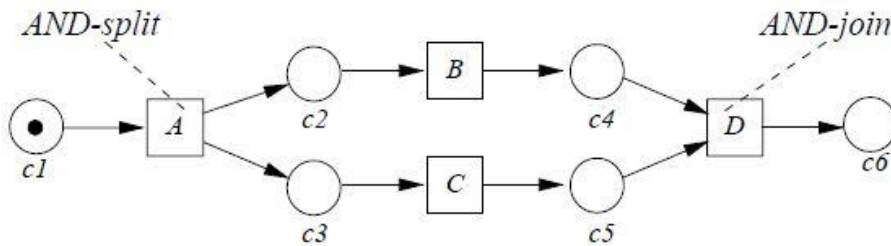
- **Proširenje s hijerarhijom**

Iako vremenske obojene Petri mreže omogućuju sažeti opis mnogih poslovnih procesa, precizne specifikacije realnih sustava imaju tendenciju da mreže postanu velike i kompleksne. Iz tog razloga, omogućena je konstrukcija hijerarhije, koja se naziva podmreža (*subnet*). Podmreža može biti skupina određenog broja pozicija, tranzicija i podsustava. Takva konstrukcija koristi se kako bi se strukturirali složeni procesi. Na jednoj razini želi se dati jednostavan opis procesa. Proširenje s hijerarhijom omogućuje takav pristup.

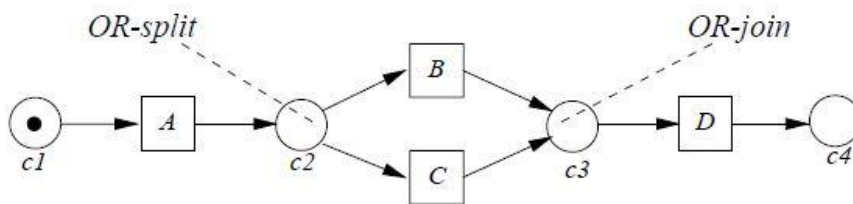
U modeliranju procesa pomoću Petri mreža, koriste se tzv. građevni blokovi (*building blocks*) kao što su *AND-split*, *AND-join*, *OR-split* i *OR-join* kako bi se dobila sekvencijalna, uvjetna, paralelna i iterativna usmjeravanja.



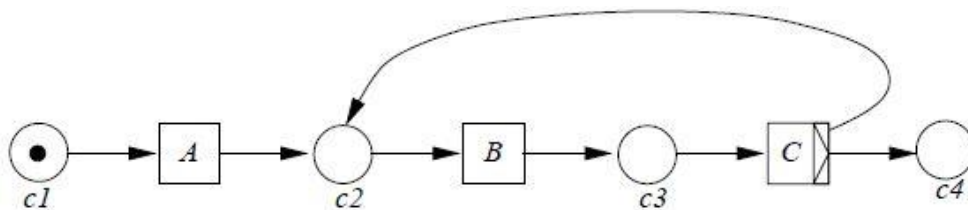
Slika 14: Sekvencijalno usmjeravanje [7]



Slika 15: Paralelno usmjeravanje [7]



Slika 16: Uvjetno usmjeravanje [7]



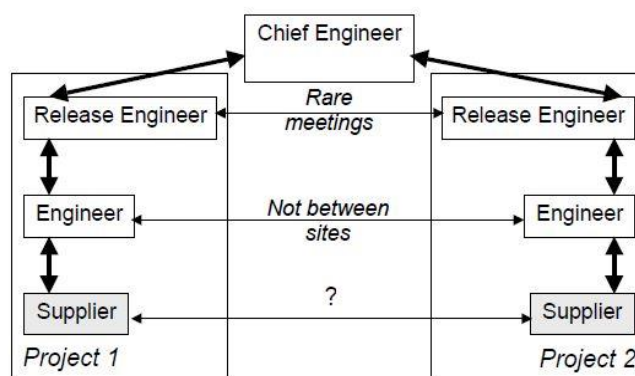
Slika 17: Iterativno usmjeravanje [7]

2.6. Tokovi parametara

Komunikacija je ključna za uspješnost bilo kojeg razvojnog procesa. No, u razvoju složenih proizvoda, ona često ne uspijeva i utječe na smanjenje efikasnosti procesa i kvalitetu proizvoda. U razvoju kompleksnih proizvoda sudjeluje veliki broj ljudi, iz različitih specijaliziranih područja koji rade na različitim zadacima. Pojedinci često imaju loš uvid u to kako se njihovi zadaci uklapaju u kontekst šireg pogleda na proizvod unatoč tome što su specijalizirani za svoja područja i razumiju zadatke osoba s kojima često rade. Konstruktori ponekad ne znaju odakle je došla informacija, odnosno parametar, te tko će njegovu vrijednost dalje koristiti. To otežava pregovaranje između konstruktora i vodi prema gubitku resursa. Pogreška u razmjeni znanja je često posljedica problema u komunikaciji. [3]

2.6.1. Problemi u komunikaciji

Konstruktori stvaraju i daju specifikacije svojih zadataka, što je izraženo u obliku parametara, uglavnom u formi skica ili numeričkih vrijednosti. Tokovi parametara odnose se na put koji oni prođu u procesu razvoja proizvoda i podatke koji utječu na vrijednosti parametara, kao što su zahtjevi i ograničenja. Više konstruktora često koristi vrijednosti istih parametara, a da pritom nisu svjesni jedni drugih. U tom slučaju, ako jedan od njih treba promijeniti vrijednost, drugi nisu o tome obaviješteni. Kod modeliranja tokova parametara tijekom procesa razvoja, dizajneri mogu pratiti informacije generirane tijekom procesa i zatražiti obrazloženja za određene vrijednosti. Razvojni timovi mogu pregovarati o zadacima i raspravljati o vrijednostima parametara te biti informirani o promjenama. Razumijevanje kompleksnosti i podrška toku informacija tijekom projekta je ključna za uspješnost razvoja proizvoda. [3]



Slika 18: Interakcija u hijerarhijskom uređenju formalne komunikacije [3]

Kompleksni proizvodi su podijeljeni u module s relativno jednostavnom interakcijom kako bi se smanjila zahtjevnost upravljanja razvojim procesom i ubrzao proces razvoja. No, nedostatak svijesti o interakcijama komponenata i međuovisnosti razvojih procesa rezultira nizom problema. Dizajneri ne znaju koje informacije moraju dati u koje vrijeme, a koje informacije trebaju prihvatiti. Posebno se ističu sljedeća pitanja:

Tablica 3: Problemi s kojima se dizajneri susreću u komunikaciji

Nepotpuna povijest informacije
Članovi tima ne mogu ući u trag informacijama, kao što su specifikacije i vrijednosti parametara, odnosno saznati koji su dizajneri odgovorni za njih. Stoga ne mogu dovesti u pitanje te vrijednosti ili mijenjati prijašnje odluke, iako bi te promjene poboljšale cjelokupni proizvod.
Paralelna (istovremena) primjena informacija je nepoznata
Dizajneri ne znaju tko sve istovremeno koristi istu informaciju ili koje posljedice promjena određenog parametra ima na druge aspekte razvoja. Zbog organizacijske strukture poduzeća teško je dobiti informacije kako bi se razmotrili paralelni parametri.
Nepoznata odredišta za informacije
U mnogim slučajevima dizajneri ne znaju na koga će utjecati njihove odluke. Nepoznati su odredište i primjena kreiranih informacija te njihov utjecaj. Dizajneri ne pružaju svojim kolegama odgovarajuće informacije, posebice podatke o odlukama koje su privremene. Ne pružaju se informacije o parametrima čija ograničenja se mogu mijenjati. To rezultira proizvoljnim odlukama koje vode nepotrebnim ograničenjima kasnije u projektu.

Posljedice nerazumijevanja tokova parametara u razvoju su velike. Bitnim zadacima se ne dodjeljuje prioritet, što rezultira nepotrebnim kašnjenjima za druge. To je posebice tako s malim zadacima, kao što je naručivanje komponenata, koje može imati veliki utjecaj na projekt ako nije učinjeno na vrijeme. Dizajneri stoga ne uspijevaju uvidjeti mogućnosti za poboljšanje vlastite izvedbe. [3]

2.6.2. *Modeli procesa razvoja*

Design Structure Matrices (DSM) su kvadratne matrice s identičnim retcima i stupcima s naznačenom dijagonalom i koriste se za označavanje zavisnosti jednog elementa prema drugom. Te matrice prikazuju svojstva procesa. Može se promijeniti redoslijed redaka (zajedno sa stupcima) kako bi se postigao minimalan broj iteracija. No, DSM ne sadrži informaciju kako su zadaci povezani i zbog toga se ne može koristiti za stvaranje tokova parametara.

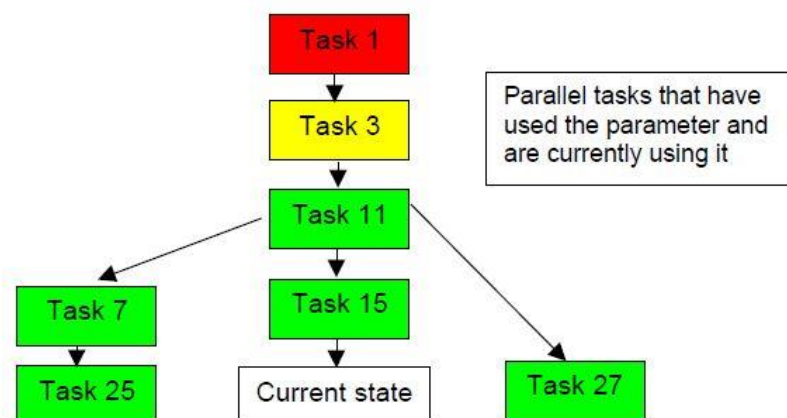
Signposting je model dinamičkog razvojnog procesa temeljen na vezama unutar zadatka kroz parametre. Izlazni parametri jednog zadatka su korišteni kao ulazni za drugi zadatak. Inicijalno je *Signposting* model korišten da uputi dizajnere na sljedeći zadatak, pokazujući im one zadatke za koje imaju dovoljno ulaznih podataka. Tim modelom su generirani mogući putevi kroz razvojni proces. Za razliku od drugih modela, *Signposting* sadrži dovoljno informacija da prikazuje tokove parametara te na taj način dizajneri mogu saznati trenutni status parametra.

U ovom članku razina pouzdanosti nekog zadatka prikazana je pomoću boja. Crvena boja znači da je vrlo vjerojatno će se taj zadatak izvršiti opet, što znači da vrijednosti parametara koje su proizašle iz te radnje nisu pouzdane. To su uglavnom početne vrijednosti s kojima se ulazi u iteracijski proces. Žuta boja predstavlja zadatke koji imaju veću razinu pouzdanosti. To upućuje da su vrijednosti parametara generirane tim zadatkom već sigurnije, no još uvijek postoji mogućnost njihove promjene i ponovnog izvođenja zadatka. Zelena boja predstavlja zadatke koji su rezultirali konačnim vrijednostima parametara. U dijagramu je moguće vidjeti i koji se zadatak trenutno izvodi. Ukoliko se vrijednost parametra treba promijeniti, dizajneru je potrebna informacija koji sve zadaci koriste tu vrijednost. Osim paralelnih zadataka, tokovi parametara pokazuju i zadatke koji su već izvršeni, a utjecali su na zadatak koji se trenutno izvodi.

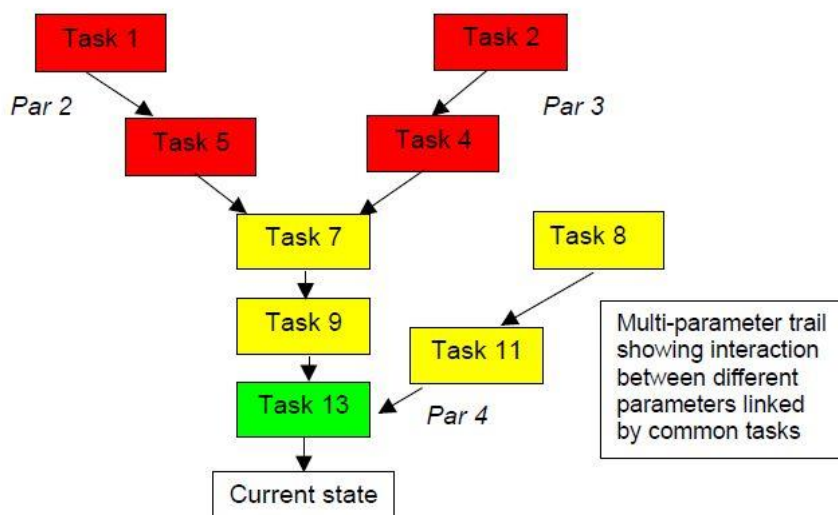
U stablu koje prikazuje tokove više parametara, dizajner može vidjeti kako drugi parametri utječu na razmatranu vrijednost (npr. ako konstruktor želi promijeniti vijak, htio bi znati o povezanim naprezanjima i dimenziju promjera kroz koji vijak mora proći). Takvo parametarsko stablo utvrđuje veze između različitih parametara zahtijevanih da bi se obavio određeni zadatak. To omogućuje dizajneru uvid u druge parametre zahtijevane kao ulazne za zadatak gdje je trenutno razmatrani parametar potreban. Osim toga, moguće je vidjeti slično

stablo za bilo koji parametar, a ne samo onaj koji je trenutno razmatran te je pružen pogled na bilo koji parametar u kontekstu cjelokupnog razvoja. [3]

U drugom dijelu ovog diplomskog rada prikazati će se tokovi parametara, utvrditi će se koji parametri su nužni za obavljanje određenih razvojnih zadataka. Nakon obavljanja određenih zadataka, generirati će se pripadni parametri te će biti moguće pratiti paralelne zadatke i trenutno stanje, što se pokazuje kao nužno za uspješan proces razvoja proizvoda.



Slika 19: Paralelni zadaci koji su koristili parametar i koji ga trenutno koriste [3]



Slika 20: Tok više parametara kojim je prikazana interakcija između različitih parametara koji povezuju isti zadatak [3]

2.7. Upravljanje parametrima, novi pristup u inženjerstvu sustava (Daimler AG)

Inženjerstvo sustava je interdisciplinarno polje inženjerstva i inženjerskog upravljanja koje se fokusira na to kako razvijati i upravljati kompleksnim sustavima kroz životni vijek proizvoda. *SE (Systems Engineering)* koristi alate koji uključuju modeliranje, simulaciju i analizu zahtjeva kako bi upravljali kompleksnošću [10]. Trenutni pristupi u inženjerstvu sustava temeljenog na modelu nemaju široku primjenu u razvoju automobilskih proizvoda zbog napora modeliranja i visoke kompleksnosti modela sustava. Članak opisuje novi pristup inženjerstvu sustava temeljen na sustavnom upravljanju inženjerskim parametrima kao kvantificiranim, povezanim i sljedivim karakteristikama inženjerskih objekata u razvoju proizvoda. Pristup ima za cilj podržati razvoj kompleksnih proizvoda tako da daje inženjerskim procesima pravu informaciju u pravo vrijeme omogućavajući sljedivost u implicitno generiranim modelima sustava na razini inženjerskih parametara. [11]

Strogo progonjeni CO₂ strategijama, kako bi se držali legalnih zahtjeva, te razvili održive i inovativne proizvode, automobilska industrija je prisiljena integrirati nove koncepte i pristupe po pitanju snage vozila, materijala i funkcionalnosti. Kao glavni izvor inovacija spominje se primjena multidisciplinarnih funkcija, realiziranih kroz mehatroničke sustave, uključujući i spregnute podsustave, koji kreiraju visoko kompleksan i umreženi proizvod. Razvoj takvih sustava zahtijeva kooperaciju više disciplina (mehanika, mehatronika i informatika) i tako pojačava organizacijsku i proceduralnu kompleksnost, vodeći prema značajnom povećanju interakcije i komunikacije među dizajnerima. [11]

SE pristupi se razvijaju kako bi pomogli uspješnom provođenju interdisciplinarnog razvojnog procesa. Znatan aspekt SE pristupa je eksplicitno modeliranje tehničkih sustava kako bi opisali svojstva i ponašanje sustava, specificirali i provjerili disciplinu koja obuhvaća koncept rješenja te omogućili sljedivost između mnoštva inženjerskih objekata (*Engineering Objects EO's*). Ovaj članak opisuje novi pristup u SE i njegovu implementaciju prototipom koja pruža podršku procesu razvoja kompleksnih proizvoda na razini inženjerskih parametara. [11]

2.7.1. Trenutni pristupi i novi pristup upravljanja parametrima u inženjerstvu sustava

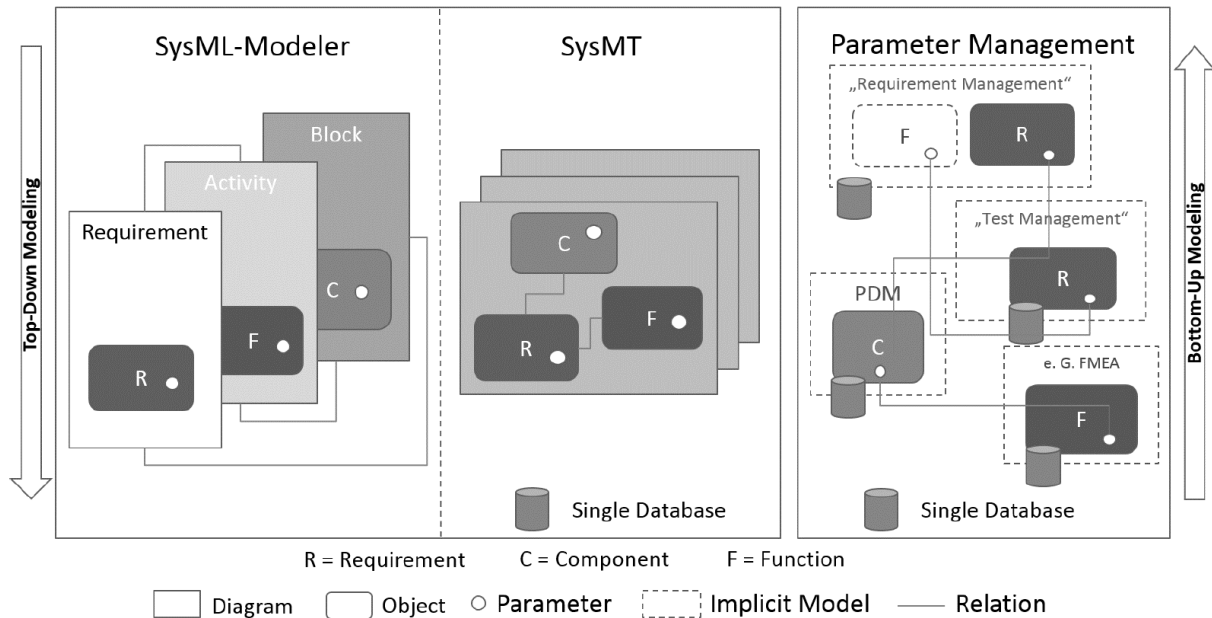
Pristup inženjerstva sustava temeljen na modelu (*Model-Based Systems Engineering*) produžuje SE pristup generirajući i koristeći centralno dostupne digitalne modele sustava kako bi olakšali interakciju između različitih softverskih alata, omogućili sljedivost između EO od

kojih se sustav sastoji. Razvoj u automobilske industriji zahtijeva suradnju dizajnera koji moraju pružiti detaljne informacije o komponentama ili modulima, iako ne profitiraju od tih informacija, dok ih drugi koriste i od njih profitiraju. Postavlja se pitanje kako SE pristup treba biti konceptualiziran da bi savladao izazove kompleksnosti sustava ujedinjujući inženjerske objekte i njihove odnose u jedan stalno ažuran prikaz sustava bez nekompatibilne grupe alata i pretjeranog napora modeliranja. [11]

Provedeni su intervjui u različitim odjelima te su analizirani tokovi rada, tokovi informacija te obrađivani tipovi podataka u automobilske industriji. Sve što je stvar percepcije, materijalno ili nematerijalno, smatra se objektom, karakteriziranim svojstvima. Kako bi se objekti međusobno razlikovali, potrebno je identificirati i definirati svojstva specifična za određeni objekt – karakteristike objekta. Inženjerski parametar predstavlja karakteristiku kvalitete i relaciju koja može specifično biti opisana kvantitetom. Parametar često nosi ime koje opisuje karakteristiku i kvantitetu (numeričke vrijednosti). [11]

Pristupi temeljeni na modelu u SE koriste stroju čitljive grafičke jezike za modeliranje koji omogućuju konstrukciju sustava i analizu. *SysML (Systems Modeling Language)* pruža predefiniranu grupu dijagrama koji omogućuju modeliranje ponašanja sustava, zahtjeva, strukture i parametara u holističkom modelu sustava. Po kreiranju modela sustava, inženjerski objekti mogu biti praćeni te informacije koje se odnose na njihove parametre mogu biti preuzete i razmijenjene. To se naziva *top-down* pristupom jer slijedi put od kreiranja holističkog modela sustava, pripajanja inženjerskih objekata, do stjecanja parametara iz EO. Taj pristup je ograničen sa složenosti modela zbog nehomogenosti inženjerskih alata koji drugačije predstavljaju parcijalne modele. Na temelju pristupa napravljena je implementacija prototipa *SysMT (System Model and Management Tool)*. Pristup baziran na predlošku omogućava dobivanje podataka iz inženjerskih objekata i sljedivost kroz relacije EO. Upravljanje parametrima je *bottom-up* pristup u inženjerstvu sustava. Za razliku od pristupa temeljenih na modelu, upravljanje parametrima ne zahtijeva izradu modela cjelovitog sustava. Umjesto toga, parametri su definirani i upravljani u jedinstvenoj (*single source*) aplikaciji. Parametri posjeduju samo reference prikaza inženjerskih objekata od kojih su nastali. Već postojeći EO i njihove implicitne veze jednog prema drugom su prikazani preko parametara, umjesto da su eksplicitno prikazani u modelu. Razvijen je prototip (aplikacija) implementacije pristupa upravljanja parametrima. Tri glavna aspekta implementirana u prototipu su: model podataka za

organizaciju parametara, funkcije unutar aplikacije koja upravlja bazom podataka kako bi kreirala radnu okolinu i funkcionalna poboljšanja na osnovu središnjeg modela podataka.



Slika 21: Usporedba *Top-Down* SE pristupa temeljenih na modelu i pristupa Upravljanje parametrima kao *Bottom-Up* pristupom [11]

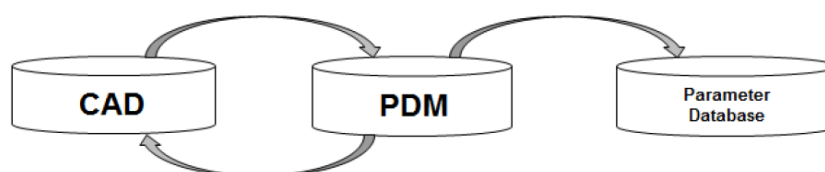
No, nedostatak pristupa je da nije podržana odgovarajuća komunikacija vezano na promjene u specifikacijama arhitekture proizvoda. Prednosti su da se promjene parametara dohvaćaju izravno iz PDM i CAD sustava, a omogućena je i sljedivost parametara. Pristup izvještava o promjeni i omogućuje trenutnu akciju. Proces dostave informacija iz parcijalnih modela u holistički model je tako sveden na minimum.

2.8. Upravljanje parametrima arhitekture vozila (Daimler AG)

Izložena studija nastavak je na prethodno obrađeni članak u kojem je predstavljen pristup upravljanja parametrima. Taj pristup olakšava upravljanje procesom dobivanja i praćenja funkcionalnih i geometrijskih parametara u aplikaciji za bazu podataka. Pristup baze podataka omogućava kvalitativno upravljanje međuovisnostima parametara definirajući tzv. aktivne lance. Aplikacija baze podataka može biti povezana s CAD modelima te omogućava inicijalnu integraciju CAD parametara i njihovo konstantno ažuriranje u bazi podataka. [12]

2.8.1. Aktivni lanci u pristupu upravljanja parametrima

Modeliranje sustava omogućava prikaz međuovisnosti unutar modela sustava. U kontekstu inženjerstva sustava, sljedivost predstavlja središnji aspekt. Sljedivost je sposobnost opisivanja, dokumentiranja i praćenja ovisnosti među artefaktima kroz cijeli razvojni proces. Danas postoji mnoštvo industrijskih alata koji podržavaju modeliranje sustava i sljedivost. No, u praksi implementacija modela nije česta zbog zahtjevnosti procesa modeliranja. Pristup upravljanja parametrima temelji se na rješenju s bazom podataka. Ovisnosti između parametara mogu se jednostavno modelirati u obliku aktivnih lanaca. Parametri mogu biti funkcionalnog ili geometrijskog karaktera. Pristup omogućava definiciju parametara u bilo koje vrijeme, unutar bilo kojeg alata. Geometrijski parametri su već implementirani u CAD modele. Ti modeli su zatim upravljani u specifičnim konfiguracijama proizvoda u PDM sustavima. CAD i PLM su sastavni dijelovi postojećeg IT okruženja. Povezivanjem CAD i PLM sustava, parametri nisu samo upravljani bez redundancije, već i ažurirani. Kako bi se parametri vratili i interpretirali u CAD podatke, informacija o parametru i njegovoj vrijednosti mora biti ugrađena tako da je čitljiva stroju. Ta funkcionalnost ostvarena je korištenjem XML tehnologije. [12]



Slika 22: Integracija baze podataka parametara u CAD-PDM okruženje [12]

Funkcionalne i geometrijske međuovisnosti unutar i između komponenata mogu biti dokumentirane spajanjem parametara specifičnih za komponentu u aktivni lanac. Aktivni lanac je spremište u kojem su parametri zajedno okupljeni. Isti parametar može biti dio nekoliko aktivnih lanaca. Kada dođe do promjene parametra, to može imati utjecaj na bilo koji drugi parametar koji je dio aktivnog lanca ili na parametre drugih lanaca ako je izmijenjeni parametar povezan s nekoliko aktivnih lanaca. Predstavljeni pristup nije ograničen niti na specifične procese niti sa razinom kompleksnosti sustava. Parametri mogu biti definirani u bilo koje vrijeme procesa razvoja i povezani s bilo kojim drugim parametrima. [12]

Korištenje baze podataka za upravljanje parametrima predstavlja pristup za kreiranje transparentnosti i dosljednosti u visoko umreženom razvojnom procesu. Povezivanje parametara u aktivne lance predstavlja pristup modeliranja s malom zahtjevnošću koji bi dizajneri mogli lako prihvatiti. [12]

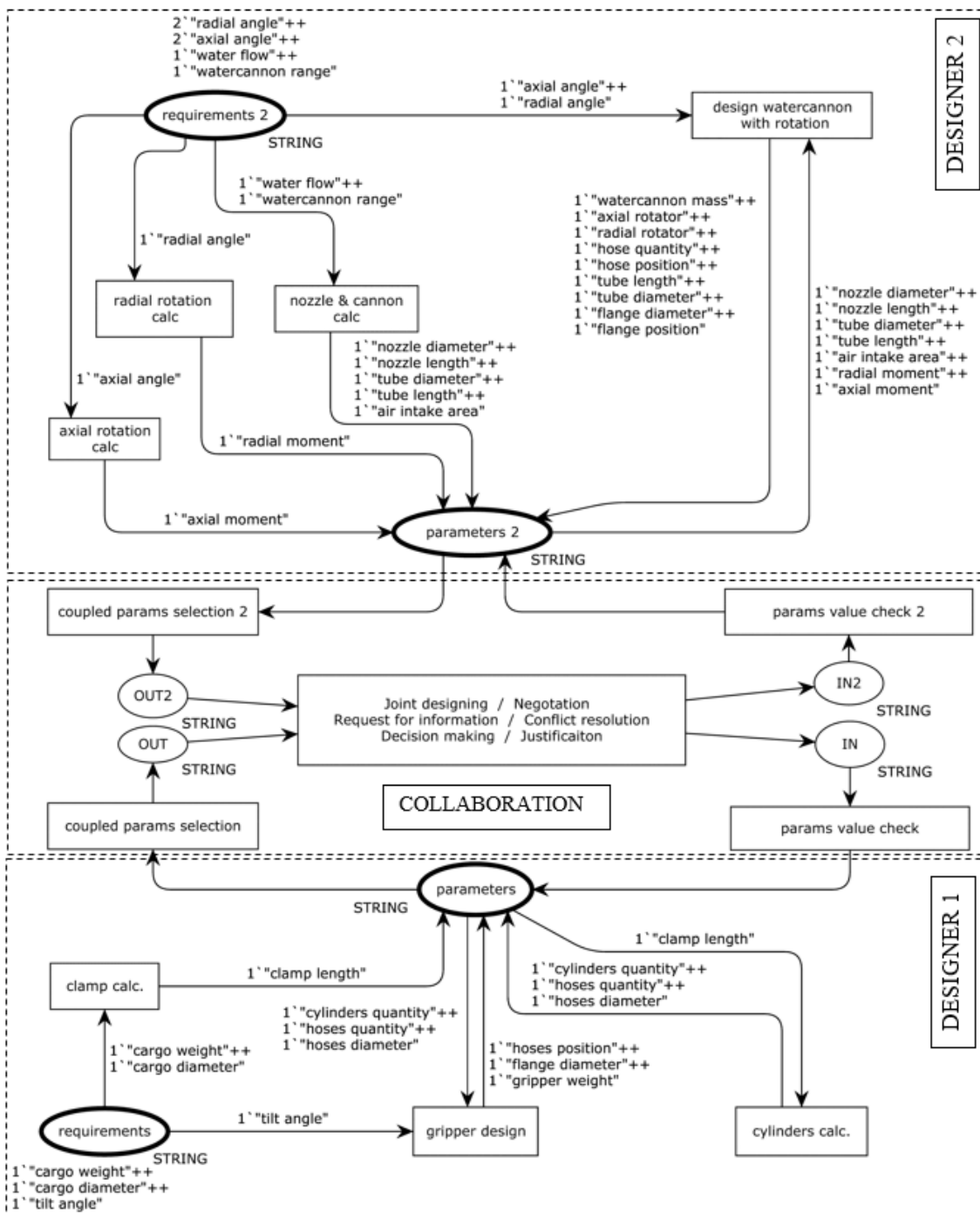
2.9. Modeliranje dinamike konstrukcijskih parametara pomoću Petri mreža

Koordinacija i komunikacija među članovima tima koji zajedno rade na kompleksnim proizvodima postale su problem od primarnog interesa. Složenost proizvoda se prikazuje kroz funkciju, tehnologiju i spajanje svih sustava na nižoj ili višoj razini kompleksnosti. Promatraju se veliki broj aktivnosti, karakteristika sudionika, karakteristike timova i organizacija te veze među njima. PLM (*Product lifecycle management*) podržava upravljanje podacima i procesima vezanim uz razvoj proizvoda, kao što su vizualizacija, veza s CAD alatima, inženjerstvo sustava, suradnja i veze s drugim poslovnim informacijskim sustavima. No, primjećeno je kako PLM ne pruža trenutno ažuriranje informacija, što dovodi do problema u razmjeni znanja između članova tima, osobito u razmjeni vrijednosti parametara ili usklađivanju njihovih vrijednosti. Niti vizualizacija toka parametara nije podržana. Potrebno je istražiti metode i alate kojima bi se uspješno upravljalo dinamikom konstrukcijskih parametara u okolnostima razvoja kompleksnih proizvoda u timskom radu koji uključuje veliki broj spregnutih parametara. Spregnute parametre moguće je prepoznati u DSM matrici, ali samo na statičkoj razini. Spregnutim parametrima nazivaju se oni čije vrijednosti utječu na konstrukciju više modula nekog proizvoda, a za svaki modul zadužen je jedan dizajner. Dizajneri moraju suradnjom utvrditi vrijednosti takvih parametara. [6]

U primjeni klasičnih Petri mreža uočeno je da modeli postaju preveliki zbog toga što manipulacija podacima mora biti prikazana direktno u strukturi mreže, a i nije razvijena podrška hijerarhijskom modeliranju (izrada podmodela). Kako bi se ti problemi uklonili, razvijene su Petri mreže više razine. *High-level Petri nets* prepoznate su kao najprikladnija mrežna tehnika za opis i analizu sinkronizacije, komunikacije i razmjenu resursa između paralelnih procesa. Obojene Petri mreže (*Coloured Petri Nets*) predstavljaju jezik za modeliranje *Discrete-Event* sustava, kombinirajući sposobnosti Petri mreža sa sposobnostima programskog jezika. CPN's pružaju i kreiranje strukture podataka i hijerarhijsku dekompoziciju. Modeliranje kompleksnog procesa CP mrežom nije trivijalno, ali ukoliko postoje ponavljajući modelirani problemi, moguće je ubrzati modeliranje koristeći tzv. uzorke (*design patterns*). [6]

U članku su analizirani parametri multifunkcionalne robotske ruke koju je razvijao tim od dva dizajnera. Svaki dizajner bio je zadužen za konstrukciju jednog modula. Odnosi između parametara analizirani su u DSM matrici, gdje je prepoznato nekoliko spregnutih parametara

čije konačne vrijednosti moraju biti donesene u procesu suradnje. U alatu CPN-Tools prikazan je inicijalni model toka parametara.



Slika 23: Inicijalni CPN model procesa razvoja [6]

2.10. Dimenzije komunikacije u razvoju proizvoda

Studija istražuje različite interakcije u dizajnu i razvija skup dimenzija za klasificiranje scenarija komunikacije u razvoju proizvoda. Sustavi podrške kolaborativnom dizajnu su razvijeni za specifične scenarije, dok razmatranje šireg raspona korištenja može pokazati više zahtjeva i potencijalnih zamki. Različite računalne tehnologije omogućuju dizajnerima da učinkovito surađuju na velikim projektima, unatoč geografskim udaljenostima. Sustavi za upravljanje informacijama koji podržavaju komunikaciju kroz vođenje evidencije su bitna stavka razvoja, kao i sustavi za sastanke licem u lice koji omogućuju dijeljenje računalnih prikaza informacija. Postoje sustavi koji podržavaju video konferencije te kolaborativno konstruiranje na dijeljenim CAD modelima. Nakon provedenog istraživanja potvrđeno je kako razgovori o dizajnu gotovo uvijek uključuju skice, crteže, prototipove i druge vizualne prikaze. Komunikacija u skupnom konstruiranju je realizirana kroz kombinaciju govora, crteža i gesta. Primijećeno je da su sporna pitanja kompleksnih proizvoda najčešće rješavana socijalnim procesima argumentiranja i pregovaranja. [14]

Tablica 4: Dimenzije komunikacijskih situacija [14]

Oblik komunikacije	Mjesto	Licem u lice ili geografska udaljenost
	Vrijeme	Komunikacija je interaktivna u realnom vremenu ili asinkrona
	Veličina	Interakcija između dvoje ili mnogo sudionika
	Identitet	Primatelji su poznati (razgovor, privatne poruke) ili nepoznati (evidentiranje, otvorena publika)
Oblik zadatka	Cilj zadatka	Stvaranje ideja ili alternativnih rješenja / konvergentno rješavanje problema / donošenje odluka na osnovu alternativa / stjecanje ili prenošenje već postojećih informacija
	Podjela odlučivanja	Zajedničko rješavanje problema / predaja nakon pregovaranja / sekvencijalno rješavanje problema
	Hijerarhija odluka	Zadaci različitih sudionika su od jednake važnosti / neki zadaci su podređeni drugima
	Trajanje	Komunikacijska aktivnost je kratka / aktivnost je produžena

	Tip informacije	Činjenice, prijedlozi, specifikacije / mišljenja, prosudbe ili prognoze
	Pritisak vremena	Zadatak je vremenski kritičan / zadatak nije hitan
Stručnost	Jednakost stručnosti	Sudionici imaju jednake razine stručnosti / neki sudionici imaju više znanja od drugih
	Stanje stručnosti	Sudionici su podijelili znanje (koriste iste kocepte i mogu interpretirati jedni drugima pojmove i prikaze) / sudionici imaju komplementarne stručnosti
	Mentalni prikazi	Sudionici konceptualiziraju temu na sličan način / na potpuno drugačiji način
	Poznanstva	Sudionici se poznaju / sudionici ne mogu imati pretpostavke o tuđem znanju
	Kontekst	Sudionici dijele kontekstualne informacije / sudionici imaju različito znanje o kontekstu
Stručnost korištenja alata	Kompetentnost pri korištenju alata	Iskusan čest korisnik / početnik
Organizacija	Hijerarhija	Sudionici na istoj razini hijerarhije / sudionici imaju različiti status
	Interes	Sudionici unutar iste tvrtke / sudionici rade za različite tvrtke
	Sigurnost	Sve informacije se mogu dijeliti / neke informacije se ne smiju dijeliti
Prikaz informacija	Medij	Govor, geste, ručno crtane skice, printani ispisi tekstualnih datoteka ili CAD modela, web stranice, dijeljene datoteke, fizički objekti kao prototipovi
	Oblik informacije	Tekst, tablice, dijagrami, kod, fotografije, tehnička dokumentacija
	Notacija	Neka područja imaju alternativne notacije za istu informaciju

Neke od tih dimenzija određuju većinu karakteristika situacija interakcije te dovode do različitih scenarija. Ti scenariji predstavljaju tipične radne situacije koje zahtjevaju računalnu podršku.

Tablica 5: Scenariji interakcije [14]

Predaja zadatka
Jedna osoba obavlja zadatak razvoja, a zatim zadatak prelazi na daljnji razvoj od strane drugog specijalista s drugačijim aspektima dizajna, uz pismenu ili usmenu specifikaciju. Pretpostavlja se da će drugi dizajner nastaviti razvoj prema zahtijevanim specifikacijama radije nego da promijeni specifikacije. Sudionici su često dislocirani i komunikacija je asinkrona. Zadaci koji se kasnije odvijaju se često promatraju kao podređeni, te dolazi do pregovaranja kada nastane problem.
Zajedničko konstruiranje
Grupa ljudi radi zajedno na problemu, najčešće u isto vrijeme na istom prostoru. Pojedinci bi mogli raditi na dijelovima problema, ali imaju jednostavan pristup jedan prema drugome te rasprave probleme odmah kada nastanu. To je čest slučaj kada se radi o grupi ljudi sličnih stručnosti, dobrih poznanstava i kada rješavanje problema utječe na sve.
Sučelje pregovaranja
U paralelnom razvoju različiti ljudi iz različitih područja rade na dizajnu u isto vrijeme. Njihovi zadaci imaju međusobno zavisne ulaze. Da bi se postigla potpuna istovremenost, oni trebaju raditi s procijenjenim vrijednostima parametara i pregovarati kako bi se postigla međusobno dosljedna rješenja za njihove individualne probleme. Paralelni procesi razvoja napreduju najbolje u projektnim timovima jer se komunikacija odvija neformalno razgovorima jedan na jedan i sastancima.
Zahtjev za informacijom
Dizajneri obično shvate da trebaju više informacija, a glavni izvor tih informacija su njihovi kolege. Zahtjev za informacijom će se češće pojaviti u predaji zadataka ili istovremenim situacijama nego na zajedničkim razvojnim sastancima.
Pregovaranje radi jasnoće i ograničenja pregovaranja
Sudionici u raspravi moraju osigurati da razumiju pozicije jedni drugih, tj. moraju postići kompatibilne interpretacije situacije. To često zahtjeva razumijevanje ograničenja koja drugi

moraju ispuniti te razumijevanje ograničenja vlastitih aktivnosti. Pregovaranje radi jasnoće često dovodi do pregovaranja preko ograničenja. To je posebno važno kada se zadaci trebaju predati od jednog specijalista drugom.

Generiranje ideja

U mnogim procesima dizajna, koji su u osnovi sekvencijalni, generiranje ideja provodi se kao zajednička aktivnost na sastancima. Potrebno je sagledati sva moguća rješenja uz obvezujuće vrijeme i raspodjelu resursa. Dizajneri često koriste ideje iz prošlih projekata ili drugih izvora.

Rješavanje sukoba

Sastanci su često dogovoreni kako bi se riješili sukobi između elemenata dizajna. Rješenja konfliktnih situacija variraju ovisno o tome da li je prisutna osoba koja ima autoritet i sposobnost da presudi ili nametne svoju odluku stranama koje su u konfliktu.

Donošenje odluke

Dizajn obuhvaća istraživanje mogućnosti nakon čega slijedi konačna odluka. Odluke moraju dati ustupke koje je potrebno napraviti, u rješavanju konflikata, vrednovanju koncepata i dr. Ako pojedinci donose vlastite odluke, moraju ih opravdati. Odluke se najčešće donose zajednički na sastancima ili od pojedinca na višoj poziciji poduzeća.

Opravdanje

Dizajneri moraju često opravdati svoja rješenja ili odluke, ili usmeno na sastancima ili u izvješćima. Primatelj ne može pretpostaviti da ima isto znanje kao i osoba koja mora opravdati rješenje. Opravdanje mora biti jednako, kako nadređenoj osobi tako i svima drugima te predstavljeno tako da ga primatelj razumije. Specifična opravdanja su često potrebna kod aktivnosti predaja zadataka.

2.11. Modeliranje komunikacijskih aktivnosti više agenata pomoću Petri mreža

Sustavi više agenata tradicionalno koriste komunikacijske protokole, koji su niz aktivnosti, pri čemu neki od tih protokola ne vode brigu o autonomiji agenata u komunikaciji. Prikazati će se tok informacija između različitih entiteta komunikacije agenata pomoću Petri mreža. Dok komuniciraju, svaki agent može neovisno okinuti tranziciju, što utječe na stanje dijaloga.

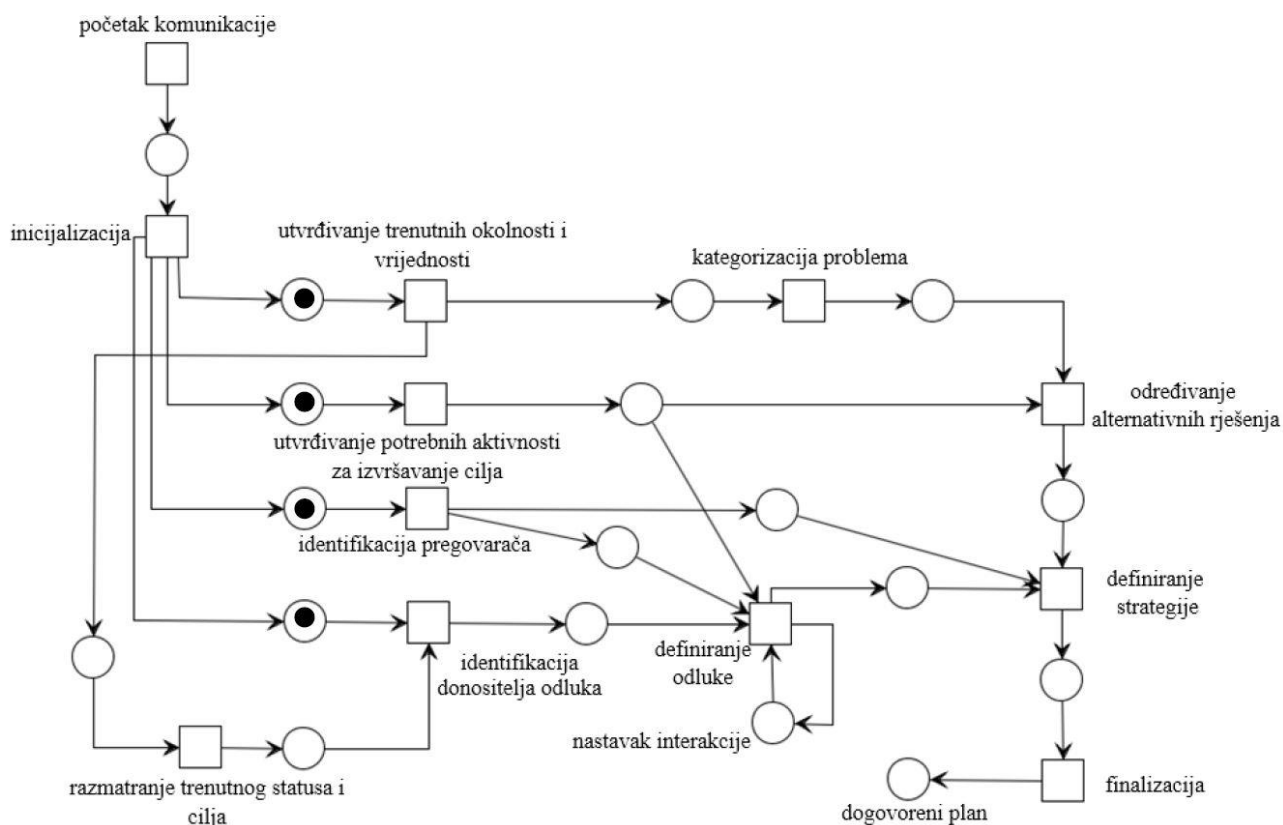
Multi-agent sustav je sastavljen od nekoliko agenata koji imaju za cilj dosegnuti ostvarenja koja su neostvariva bez doprinosa pojedinačnih agenata. Svojstva agenata su njihova autonomija i međudjelovanje koje izvode s okruženjem. Oni djeluju neovisno na temelju svog mentaliteta (načina razmišljanja) i onoga što nauče od drugih. Izrazi izmijenjeni između agenata reflektiraju njihov stil govorenja i pripadaju različitim kategorijama kao što su: (1) pregovaranje, u kojemu se agenti nastoje složiti oko podjele nekih rijetkih resursa; (2) uvjeravanje, u kojemu svaki agent nastoji uvjeriti druge da podrže njegove tvrdnje; (3) traženje informacija, u kojima agenti nastoje odgovoriti na neka pitanja drugih; (4) upit, u kojem nekoliko agenata povezano nastoji odgovoriti na neka pitanja drugih; (5) ograničavanje, u kojem se agenti nastoje povezano složiti oko nekih specifičnih situacija. U osnovi protokoli eksplicitno izražavaju komunikacijske izraze koje tipični agent izgovori tijekom interakcije s aktivnom okolinom. Ti protokoli su često specificirani uz preduvjete (ili naknadne uvjete) i pokretani pomoću mehanizma za donošenje odluka koji omogućuje agentima da razumno prihvate ili odbiju izraz izgovoren od drugog autonomnog agenta. Ponekad tipični agent treba objašnjenja ako trenutni argumenti nisu dovoljni da donese racionalnu odluku. Uobičajeno agent treba obraniti svoju prijašnju izjavu. Obrana spomenutog agenta može imati dva razloga. Prvi razlog reflektira znanje agenta, koje određuje da li je njegova izjava točna. Na taj način, on se uvijek slaže sa određenom tvrdnjom sve dok nije kontradiktorna sa njegovom već dobro poznatom informacijom. Drugi razlog je temeljen na prethodnim stajalištima koja je zastupao. Pri svakom stajalištu, agent sprema ili ažurira sve informacije razmijenjene između sebe samog i drugih agenata. Stoga, očito je da što se više agenata složi oko spomenute izjave, dobivanje tvrdnje s visokom sigurnošću postaje više povjerljivo. Štoviše, u toj funkciji više faktora, pouzdanost agenata doprinosi mogućnosti da daju težine različitim agentima, i tako dobivaju mudriju odluku. [8]

2.11.1. Petri net okvir kao formalna metoda za modeliranje različitih oblika komunikacije u sustavima s više agenata

Petri mreže se najčešće koriste za modeliranje sustava za procesuiranje (obradu i manipulaciju) podataka s istovremenim događajima. Uz prikaz i analizu, moguće je implementirati u sustav istovremenost, sinkronizaciju, konflikte i različite izbore. Petri mreže su razmatrane kao matematički model za sustave s reguliranim tokom informacija i s obzirom na to da je struktura bipartitni usmjereni graf, različiti rezultati interakcije agenata kroz različite puteve izvršenja, mogu biti praćeni i analizirani. [8]

- **Okolina pregovaranja**

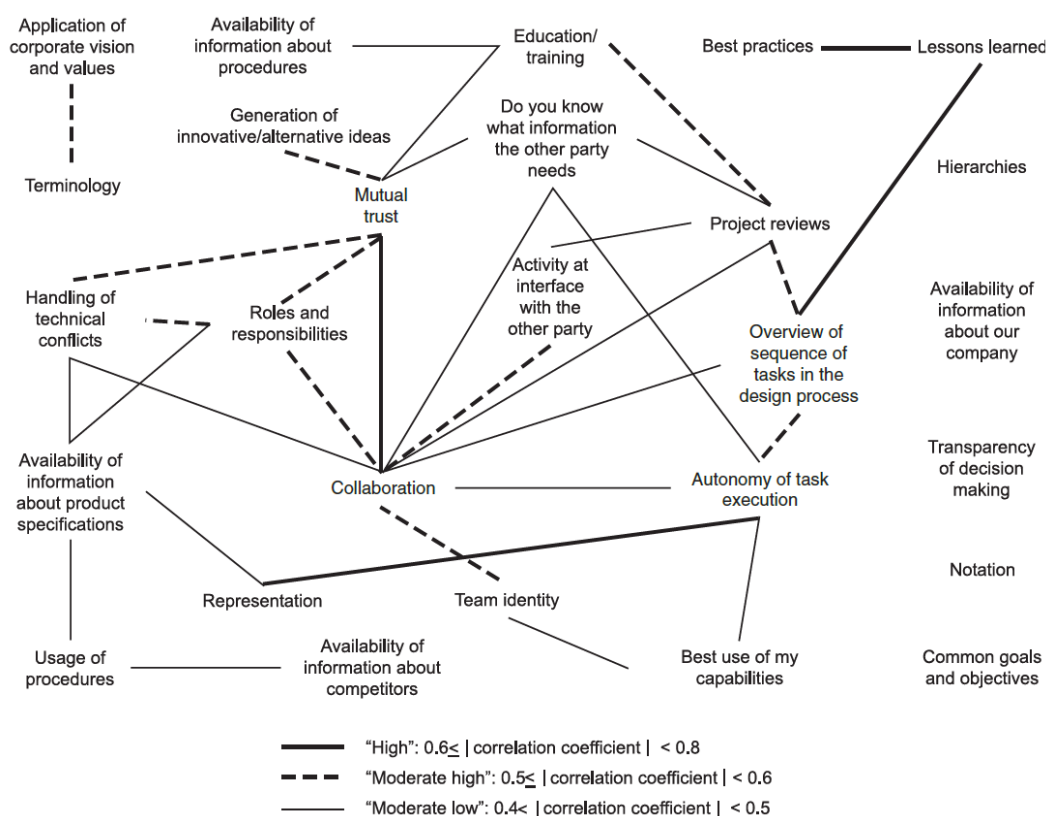
Pregovaranje između agenata je koordinirano u predefiniranoj okolini pregovaranja. Pregovaranje omogućuje više od jedne sesije (aktivnosti) u vremenu.



Slika 24: Primjer klasične Petri mreže za proces komunikacije (pregovaranja) [8], [9]

2.12. Istraživanje korelacija između faktora koji utječu na komunikaciju u kompleksnom razvoju proizvoda

U članku je prezentirana mreža faktora koji utječu na komunikaciju. Mreža je dobivena na temelju intervjuiranja inženjera, menadžera i timova. Prikazani faktori mogu poslužiti boljem shvaćanju cjelokupnog kompleksnog tehničko-socijalnog sustava te pomoći pri modeliranju alata za podršku komunikaciji. [9]



Slika 25: Mreža korelacijskih koeficijenata [13]

Istaknute su četiri varijable za koje se pokazalo da imaju najviše veza prema ostalim varijablama te su zbog toga nazvane jezgrene ili temeljne varijable u komunikaciji. To su:

- Suradnja
- Uzajamno povjerenje
- Pregled slijeda izvođenja zadataka u razvojnom procesu
- Autonomija izvođenja zadataka

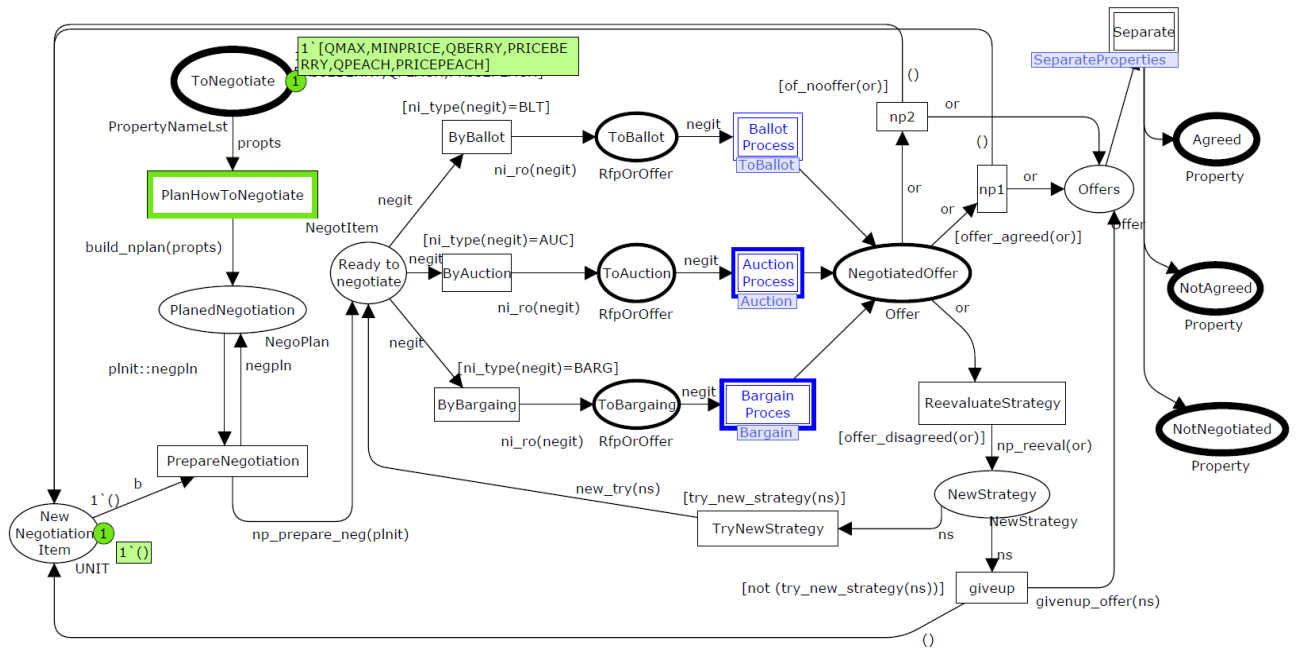
Mijenjanje statusa pojedinog faktora može imati značajan utjecaj na velik broj ostalih faktora ukoliko je razina njihove povezanosti visoka. Također, mijenjanje jednog faktora koji se javlja u više ciklusa procesa može imati neplanirane efekte na faktore koji nisu direktno povezani s istim. Mreža faktora koji utječu na komunikaciju postoji kao rezultat istovremenih lokaliziranih radnji. Takve se mreže uobičajeno modeliraju kao višestruki čvorovi u kojima svaki čvor predstavlja varijablu stanja s određenom vrijednošću. Potrebno je utvrditi jesu li utjecaji na komunikaciju linearni ili nelinearni te simetrični ili asimetrični. Odmakom od linearnosti i simetričnosti komunikacija se komplicira te je za donošenje zaključaka o takvim procesa potrebno provesti daljnja istraživanja. Također je konstatirano da nije moguće pratiti sve uzročne obrasce povezane s faktorima koji utječu na komunikaciju zbog njihove kompleksnosti i različitosti. [9]

2.13. Modeliranje i simulacija protokola višestranog pregovaranja pomoću obojenih Petri mreža

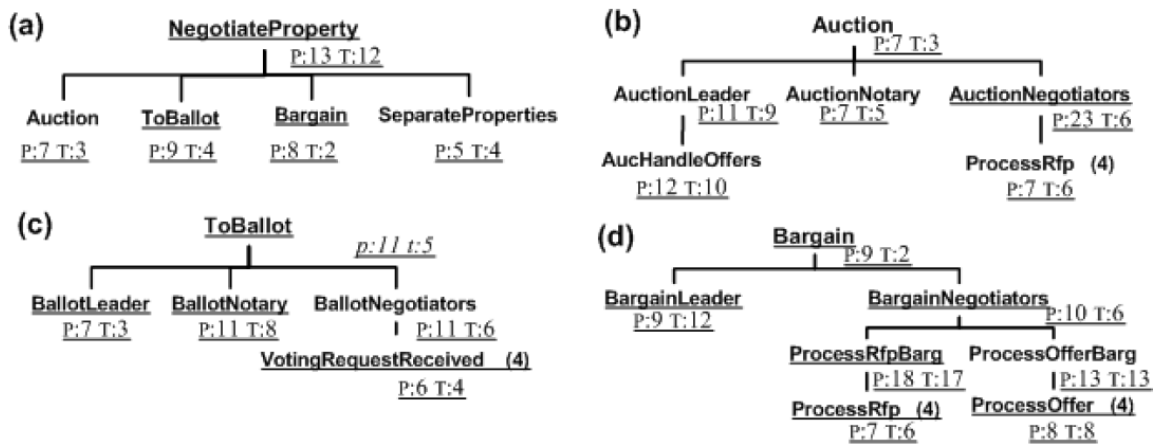
Članak obrađuje pristup modeliranju protokola višestranog pregovaranja, te je prikazano kako različiti načini pregovaranja (npr. aukcije, pogađanje) mogu biti modelirani na jedinstven i dosljedan način. U članku je opisan protokol pregovaranja koji kombinira tri različita načina: pogodba, aukcija i glasanje. Rezultat pregovaranja je e-ugovor koji će biti potpisan od nekoliko partnera. Model je realiziran pomoću obojenih Petri mreža. Za kreiranje i analizu obojenih Petri mreža korišten je CPN Tools. Simulacije su korištene za validaciju i analizu izvođenja, dok se analiza stanja može koristiti za utvrđivanje mogućih grešaka u protokolima. Obrađuje se problem e-ugovaranja, za reguliranje odnosa između virtualnog poduzeća i partnera koji dobivaju razne proizvode. E-ugovaranje se može realizirati putem web servisa te je važno utvrditi pravila za uspješno ugovaranje poslova koji zadovoljavaju unaprijed utvrđene uvjete (parametre), npr. količina, cijena proizvoda i rok isporuke. [16]

Automatsko pregovaranje nije trivijalan zadatak. Većina sustava za pregovaranje ima ograničen doseg: podržavaju samo jednu vrstu pregovaranja, najčešće pogodbe ili aukcije na ograničenom tržištu. Rezultat su ugovori između dva pregovarača. Ako u dogovoru sudjeluje više od dva partnera, mora se napraviti više dvostranih ugovora. U sporazumu o suradnji između više partnera, to može dovesti do gubitka značenja pojedinih stavaka. Jedan ugovor između nekoliko partnera može izraziti njihov zajednički cilj pomoću skupa složenih odnosa. Dijeljenje tog ugovora u nekoliko ugovora između dva partnera može dovesti do gubitka nekih odnosa bitnih za druge partnere. Ovaj članak obrađuje i automatsko pregovaranje i višestrano ugovaranje. [16]

Fokus članka je na SPICA (*SuPply chain Integration, Coordination, conctracting and Auditing framework*) protokolu koji predstavlja model procesa pregovaranja kojim upravlja voditelj pregovora. Nakon uspješnog pregovaranja kreira se novi ugovor na osnovu ugovorenih vrijednosti. Ugovor je predefiniрани predložak koji se popunjava vrijednostima koje su dogovorene između pregovarača. [16]



Slika 26: Obojena Petri mreža procesa pregovaranja (*Negotiation Property Page*) [16]



Slika 27: Hijerarhijski prikaz elemenata modela procesa pregovaranja (glavna obojena Petri mreža i pripadne podmreže) [16]

3. Obojene Petri mreže i alat CPN tools kao alat za modeliranje i simulaciju dinamičkih procesa pomoću Petri mreža

Kako bi se inženjeri lakše nosili s kompleksnošću modernih paralelnih sustava, potrebno je pružiti metode koje omogućuju pronalaženje i ispravljanje grešaka (*debugging*) te testiranje dijelova procesa razvoja prije njegove implementacije. Konstruiranje modela i simuliranje njegova ponašanja daje novi uvid u razvoj i razmatranje operacija sustava te pruža sistematsko istraživanje mogućih scenarija što može smanjiti broj pogrešaka u razvoju. [15]

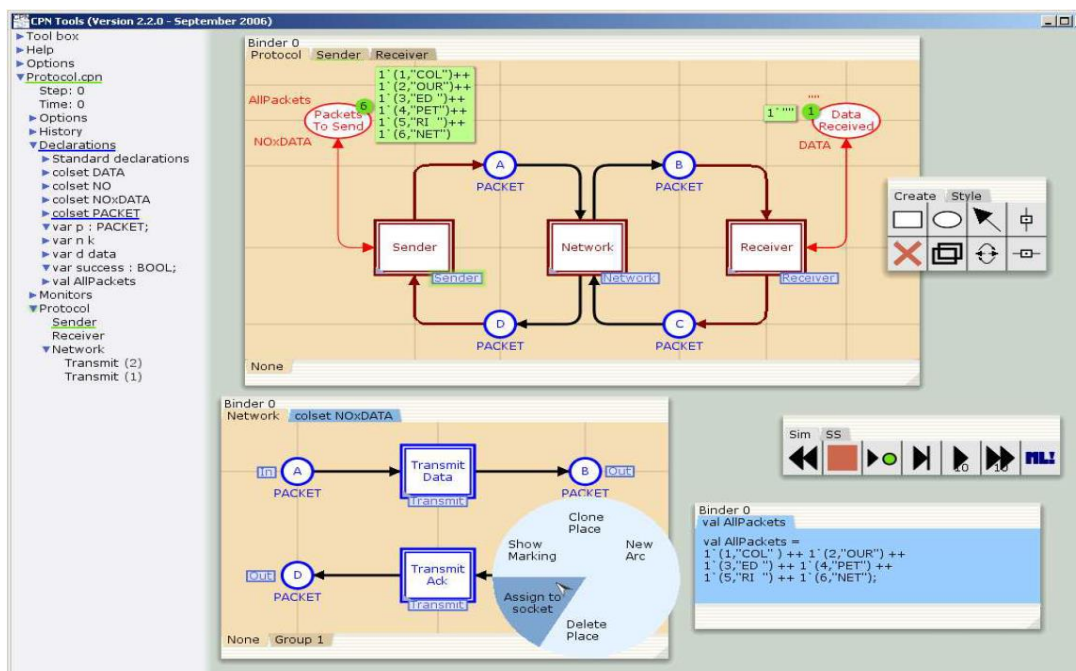
CPN Tools je grafički realizirani jezik za izradu modela paralelnih sustava i analizu njihovih svojstava. Alat je razvijen na Sveučilištu u Aarhus-u (Danska). *CPN Tools* je alat za uređivanje, simulaciju i analizu obojenih Petri mreža. Značajka alata je da inkrementalno provjerava sintaksu i generira kod koji nastaje kada se mreža modelira. Simulator je brz, učinkovito prikazuje i vremenski ovisne mreže procesa. Funkcijski programski jezik *Standard ML* pruža platformu za definiranje tipova podataka, za opisivanje manipulacije podacima te za kreiranje parametriziranih modela. *CP-nets* je dodatak na postojeće jezike i metode za modeliranje (kao što je *UML*) i može biti korišten zajedno s njima ili biti integriran u njih. Licenca za *CPN-Tools* može biti dobivena besplatno i za komercijalno korištenje. [15]

Tipična područja primjene su komunikacijski protokoli, mreže podataka, poslovni procesi, proizvodni sustavi i dr. CPN modeli se koriste za modeliranje i određivanje ponašanja paralelnih sustava. Interaktivna simulacija omogućuje pogled u različite ishode i provjerava da li model radi kao što se očekuje. Automatska simulacija izvršava CPN model što brže i učinkovitije može, bez korisnikova utjecaja. Pomoću *CPN-Tools* alata moguće je provjeriti svojstva modela uz *state-space* analizu i kontrolu modela te provesti *simulation-based performance* analizu. CP mreže uključuju koncept vremena što omogućuje definiranje kašnjenja događaja u sustavu te je moguće organizirati model kao grupu modula pri čemu moduli mogu pripadati različitim razinama hijerarhije. CPN Tools omogućuje kreiranje složenih tipova podataka, polja, liste, te sadrži nekolicinu predefiniраниh funkcija. U zadnjih nekoliko godina razvijeni su uzorci tzv. *patterns*, koji ubrzavaju modeliranje time što je moguće module mreže koji se ponavljaju više puta “zalijepiti“ u već postojeću mrežu. [15]

3.1. Mogućnosti CPN jezika za modeliranje prikazane na modelu komunikacijskog protokola

3.1.1. Korisničko sučelje

Korisničko sučelje sastoji se od *Indexa* i radnog prostora za modeliranje sustava. U *Index* području nalaze se alati za kreiranje deklaracija, strukture mreže, natpisa i hijerarhijskih modela te *Tool box* (omogućuju kreiranje i kloniranje CPN elemenata). Postoje i dodatni alati kojima se može poboljšati vizualizacija i čitljivost modela, uključujući alate za promjenu boje linija, promjenu debljine crta te primjenu elemenata za poravnanje. *Index* omogućuje pregled svih modela koji su trenutno otvoreni. Iz *Indexa* se željene kartice s alatima mogu odvući u radni prostor (npr. za kreiranje elemenata mreže i simulaciju, slika 28.). Tipove podataka (*color sets*), varijable i funkcije koje su koriste u modeliranju potrebno je deklarirati u *Declarations* sekciji *Indexa*. Nakon kreiranja nove pozicije, tranzicije ili strelice, potrebno je upisati imena elemenata, *TAB* tipkom pozicijama dodati tip podatka koji će se u njima nalaziti te dodati inicijalne podatke ukoliko je potrebno. Svaka pozicija može biti markirana s jednim ili više tokena, a svaki token ima vrijednost podatka koji mu je pridružen. Klikom na strelice dodaje se izraz (*expression*), kojim se definira uvjet koji mora biti ispunjen za izvršenje tranzicije (izraz na ulaznoj strelici) te manipulacija tokenima nakon izvršenja tranzicije (izraz na izlaznoj strelici). [15]

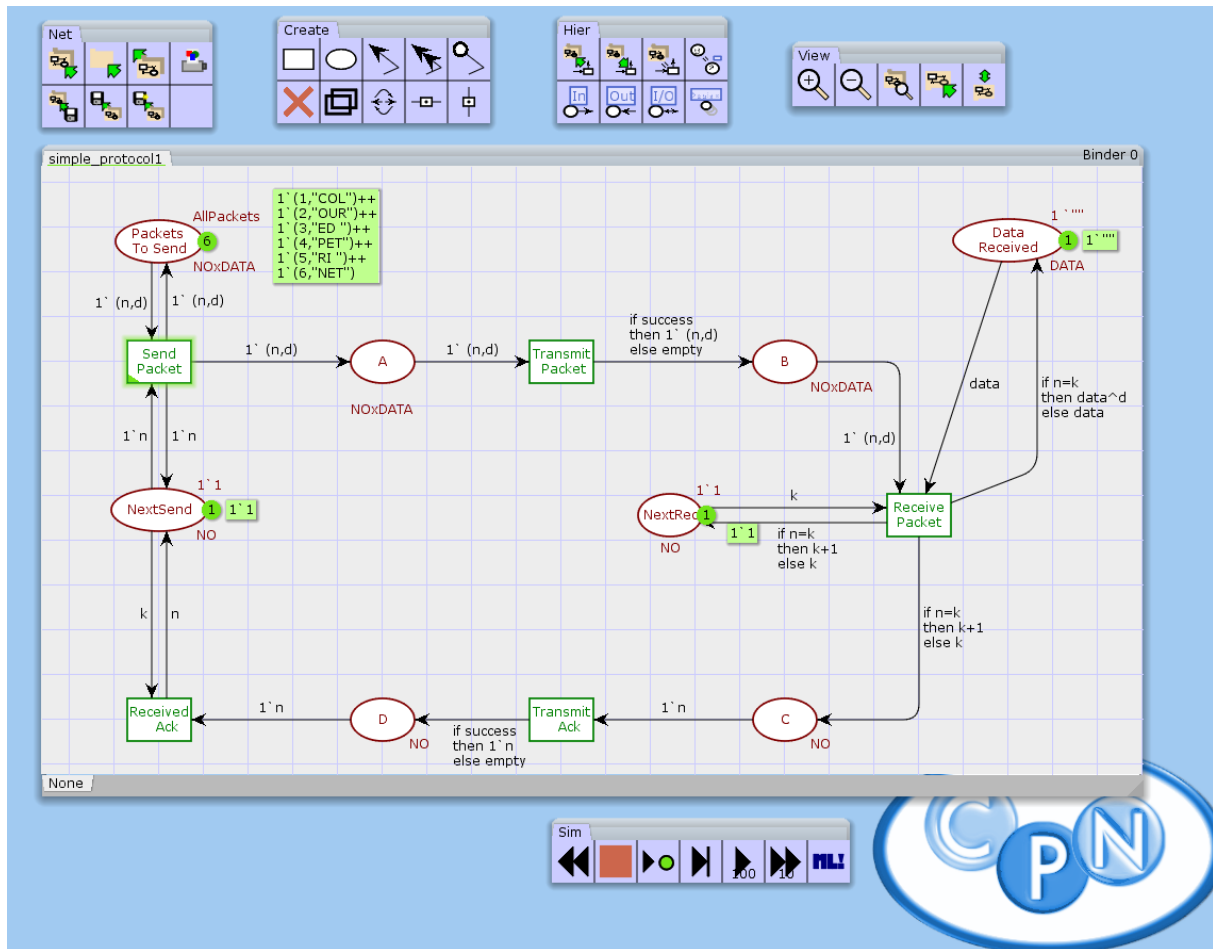


Slika 28: Radni prostor aplikacije CPN Tools [15]

3.1.2. Model komunikacijskog protokola

Protokolom se želi poslati šest paketa, određenim redom, a svaki od paketa “nosi” neku boju. U ovom primjeru boja je definirana kao *NOxDATA* što znači da se tokenom prenosi neki cijeli broj i string. Tokeni se pohranjuju u poziciji *Data Received* te se pomoću određenog operatora stringovi spajaju redom kako stižu u poziciju. Kada je evidentirano da je paket stigao, potrebno je poslati potvrdu te zahtjev za slanjem idućeg paketa. To je ostvareno povećanjem brojača (varijabla *k*) te povratnom vezom.

Komunikacijski protokol sastoji se od osam pozicija i pet tranzicija (slika 29.). Nakon što su kreirani elementi i veze među njima, potrebno je deklarirati tipove podataka i varijable koje će se koristiti u kreiranju uvjeta (slika 29.). Ukoliko se podaci sastoje od nekoliko različitih tipova, koristi se operator *Product*. Poziciji *Packets To Send* inicijalno je dodijeljeno šest tokena, tipa *DATAxNO*, a pozicijama *NextSend* i *NextRec*, tipa cijeli broj, inicijalno su dodijeljeni tokeni koji nose vrijednost 1. Na strelicama se nalaze varijable u koje će se pohranjivati tokeni pri kolanju kroz mrežu. Varijable će omogućiti samo pohranu onih podataka koji odgovaraju tipu deklaracije varijabli. S obzirom na to da se paketi mogu izgubiti pri prijenosu na određene strelice potrebno je postaviti logički uvjet. Ukoliko varijabla *success* tipa *boolean* slučajnim odabirom poprimi vrijednost *true*, podaci će se proslijediti, a ako dođe do greške u prijenosu, proces se zaustavlja (konstanta *empty*). Idući paket može se poslati tek kada je string podatak uspješno pohranjen u poziciji *Data Received*, a to se kontrolira uvjetom na ulaznoj strelici. Ako redni broj podatka koji je stigao odgovara pomoćnoj varijabli *k*, ona se poveća za 1 i poprima redni broj idućeg paketa, te šalje dalje u povratnu granu koja omogućuje generiranje potvrde i šalje zahtjev za slanjem idućeg paketa. Izrazom *data^d* svaki dolazni string “lijepi” se na prethodni kako bi se na kraju dobio izraz *COLOURED PETRI NET*. [15]



Slika 29: Osnovni CPN model komunikacijskog protokola s inicijalnim markiranjem M_0 [15]

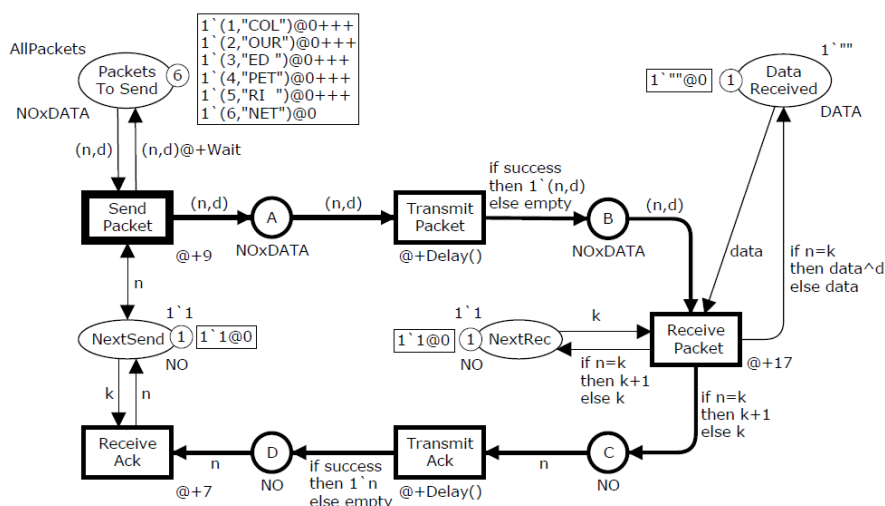
```

▼ Declarations
▶ Standard priorities
▼ Standard declarations
  ▼ val AllPackets = 1` (1,"COL")+1` (2,"OUR")+1` (3,"ED ") +1` (4,"PET")+1` (5,"RI ") +1` (6,"NET");
  ▼ colset DATA = string;
  ▼ colset NO = int;
  ▼ colset NOxDATA = product NO*DATA;
  ▼ var n : NO;
  ▼ var d : DATA;
  ▼ var data : DATA;
  ▼ var k : NO;
  ▼ colset UNIT = unit;
  ▼ colset BOOL = bool;
  ▼ var success : BOOL;
  ▼ colset INT = int;
  ▼ colset INTINF = intinf;
  ▼ colset TIME = time;
  ▶ colset REAL
  ▶ colset STRING
▶ Monitors
  simple_protocol1
    
```

Slika 30: Index u kojem su deklarirani tipovi podataka, konstante i varijable [15]

3.1.3. Dodjeljivanje vremena i kreiranje hijerarhijski povezanih modula

U alatu *CPN Tools* moguće je modelima dodati vremensku informaciju. Vremensko kašnjenje, odnosno realno trajanje može se dodijeliti tokenima u pozicijama, događajima predstavljenim tranzicijama te uvjetima na strelicama. Vremenski CPN modeli uz boju tokena nose još jednu vrijednost tzv. *time stamp*. Kod deklaracije tipova podataka potrebno je napisati riječ *timed*. Na slici 31. prikazana je sintaksa za dodjeljivanje vremenskih konstanti. Izvršenje vremenskog CPN modela kontrolirano je preko globalnog sata i radi na sličan način kao niz događaja u *Discrete Event* simulacijama. Model ostaje u određenom trenutku tako dugo dok se ne izvrši neka od aktivnih (*colour enabled*) tranzicija. Također je moguće pomoću predefiniране funkcije *Delay()* tranzicijama dodati vremensko kašnjenje koje odgovara nekom broju iz proizvoljno definiranog intervala. [15]



Slika 31: Vremenski CPN model protokola s inicijalnim markiranjem M_0 [15]

Kompleksnu mrežu moguće je podijeliti na podmreže (module), a uvjet je da svaki modul mora biti ograničen s tranzicijama. Alati za kreiranje hijerarhijske mreže nalaze se u *Hierarchy* paleti, a potrebno je u radnom postoru imati i *View* paletu. Prvi korak je kreiranje nove grupe klikom na opciju *Creates a new group* u izborniku *View*. Zatim se odaberu elementi koji pripadaju određenoj grupi klikom na opciju *Toggles whether an element belongs to a group*. U paleti *Hierarchy* nakon odabira elemenata selektira se opcija koja miče elemente i rubne tranzicije u podkarticu s imenom grupe. Na Slici 28. modelirana su tri modula komunikacijskog protokola. Moguće je grupirati i tranzicije koje imaju različite tipove ulaznih podataka, koristeći tip podatka *union* (slika 28. *PACKET*). [15]

3.2. Strukture podataka i naprednije mogućnosti korištene u modeliranju mreže za upravljanje konstrukcijskim parametrima u timskom radu

Prije kreiranja obojene Petri mreže u alatu *CPN Tools* potrebno je deklarirati strukture podataka i varijable te inicijalizirati varijable. Osim uvjeta za izvršenje tranzicije koji se nalaze na ulaznim strelicama (*arc expressions*), moguće je postaviti i dodatne uvjete na same tranzicije (tzv. *transition guards*). Alat nudi i nekoliko predefiniраниh funkcija koje najčešće obavljaju određene matematičke operacije. [18]

Tablica 6: Strukture podataka, varijable, funkcije i logički izrazi u modelu za upravljanje konstrukcijskim parametrima u timskom radu

COLOR SET	SINTAKSA DEKLARACIJE
colset B	colset B = bool with (no,yes);
colset NO	colset NO = int;
colset DATA	colset DATA = string;
colset NOxDATA	colset NOxDATA = product NO*DATA;
colset DATAxNO	colset DATAxNO = product DATA*NO;
colset Range	colset Range = int with 500..700;
colset Lista	colset Lista = list NO;
colset BOOL1	colset BOOL1 = bool;
colset BOOL2	colset BOOL2 = bool;
colset REAL	colset REAL = real;
colset DATA1	colset DATA1 = string;
colset DATA1xREAL	colset DATA1xREAL = product STRING*REAL;
colset ZAHTJEVI_1	colset ZAHTJEVI_1 = DATA1*REAL;
colset ZAHTJEVI_2	colset ZAHTJEVI_2 = DATA1*REAL;
colset RJESENJE	colset RJESENJE = with D1 D2 NOVE_DIMENZIJE;
colset DONOS_ODL	colset DONOS_ODL = Designer1 Designer2 Koordinator;
VARIJABLE	INICIJALIZACIJA VARIJABLI
d	var d : DATA;

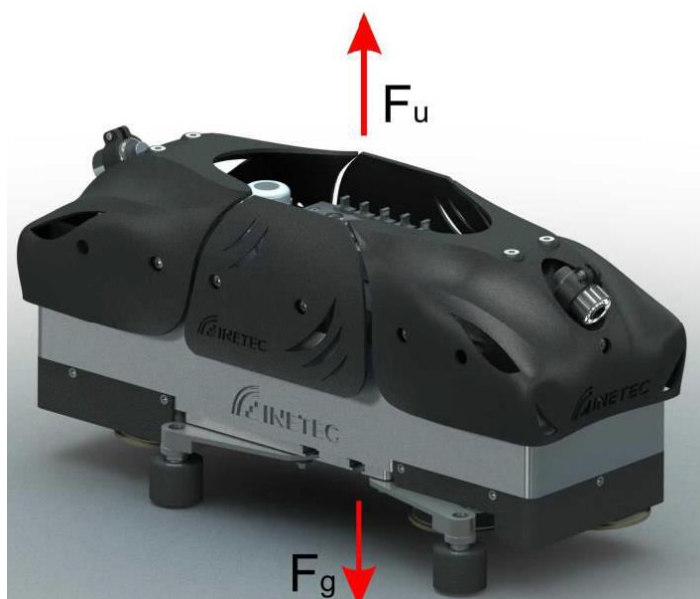
f	var f : DATAxNO;
l2	var l2 : NO;
ListaDuzina	var ListaDuzina : Lista;
NE_ODGOVARA	var NE_ODGOVARA : BOOL1;
ODLUKA	var ODLUKA : RJESENJE
DONOSITELJ	var DONOSITELJ : DONOS_ODL;
ODGOVARA	var ODGOVARA : BOOL;
GOTOVO	var GOTOVO : BOOL;
NE_ZADOVOLJAVA	var NE_ZADOVOLJAVA : BOOL2;
NE_ODGOVARA1	var NE_ODGOVARA1 : BOOL;
FUNKCIJE	
fun ranRange() = Range.ran() ; <i>ran</i> - predefinirana funkcija koja vraća slučajno odabrani cjelobrojni broj iz raspona tipa <i>Range</i> (od 500mm do 700mm)	
LOGIČKI IZRAZI	
if ODLUKA = D1 then 1` “Nove dimenzije“ else if ODLUKA = NOVE_DIMENZIJE then 1`“Nove dimenzije“ else 1`“Promjena nije potrebna“	
if ODGOVARA = false then 2` false else empty	
DODATNI UVJET NA TRANZICIJI (GUARD)	
[#2f = l2] – uzima drugi tip podatka, broj, iz varijable l2 tipa DATAxNO	

4. Složene konstrukcije u kojima su razmatrani tokovi konstrukcijskih parametara

Za analizu tokova parametara odabrala sam dvije složene konstrukcije koje sam konstruirala u timu od dva člana u dosadašnjem tijeku studija. Razmatrano je generiranje parametara, analiza tokova parametara, njihove međuovisnosti te određivanje spregnutih parametara. U ovom poglavlju ukratko su objašnjeni načini rada i prikazane konstrukcije.

4.1. Samohodni uređaj za čišćenje prirubnice spremnika

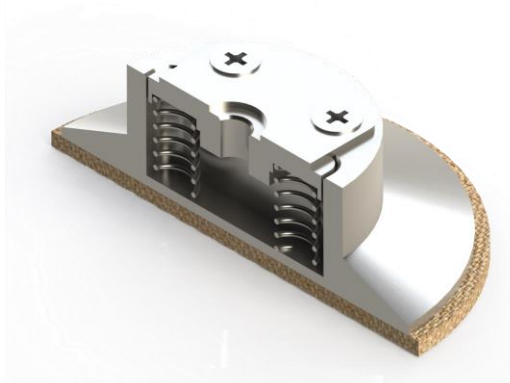
Bilo je potrebno konstruirati samohodni uređaj za čišćenje žlijebova na prirubnici spremnika pod vodom. Uređaj se morao moći samostalno gibati po površini prirubnice vertikalno postavljenog spremnika koja se nalazi 3 m pod vodom, te biti sposoban vizualno pregledavati i čistiti žlijebove brtvi. Čestice koje nastanu tijekom čišćenja bilo je potrebno usisavati i odvoditi s površine prirubnice. Bilo je potrebno razmotriti mogućnost čišćenja prirubnica unutrašnjeg promjera od 3180mm i vanjskog promjera 3492,5mm. Očekivana kontinuirana brzina gibanja uređaja bila je 700mm/min. [21]



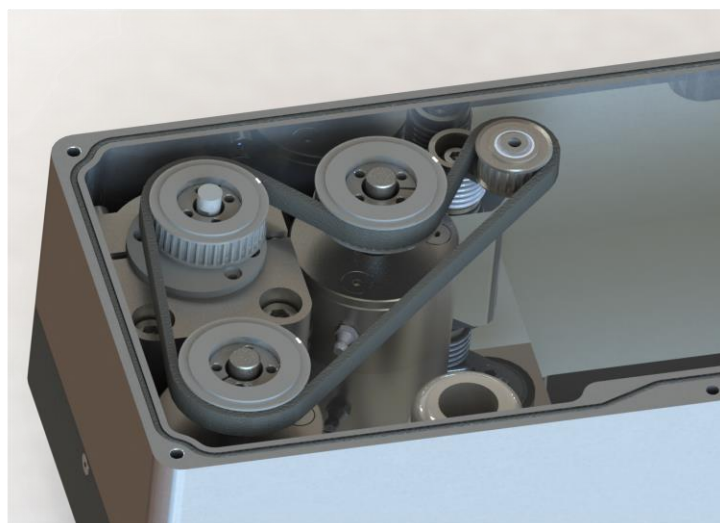
Slika 32: Uređaj za čišćenje prirubnice sa silama koje na njega djeluju [21]

Radi jednostavnosti analize tokova parametara, razmotrena su dva modula: mehanizam za pogon četki i mehanizam za poliranje prirubnice. Proces konstruiranja podijeljen je na dva dizajnera. Prvi dizajner konstruira mehanizam za poliranje prirubnice, a drugi mehanizam za pogon četki. Pri konstruiranju mehanizma za poliranje prirubnice potrebno je odabrati

dimenzije opruga te na temelju njih modelirati rotirajuće diskove (inicijalne dimenzije), na temelju zahtjeva i dimenzija klizača proračunati potreban broj i dimenzije opruga te se iterativno vratiti u proces odabira novih opruga. Drugi dizajner odabire zupčasti remenski prijenos, elektromotor potrebne snage i modelira čahure i stege za prihvat elemenata za kućište. Dizajneri se moraju dogovoriti oko vrijednosti parametara potrebnih za prihvat klizača (komponente mehanizma za poliranje) na gonjena vratila remenica (mehanizam za pogon četki). U tablici 7. nalazi se popis zahtjeva i parametara razmatranih podsklopova. [21]



Slika 33: Klizač mehanizma za poliranje prirubnice [21]



Slika 34: Mehanizam za pogon četki [21]

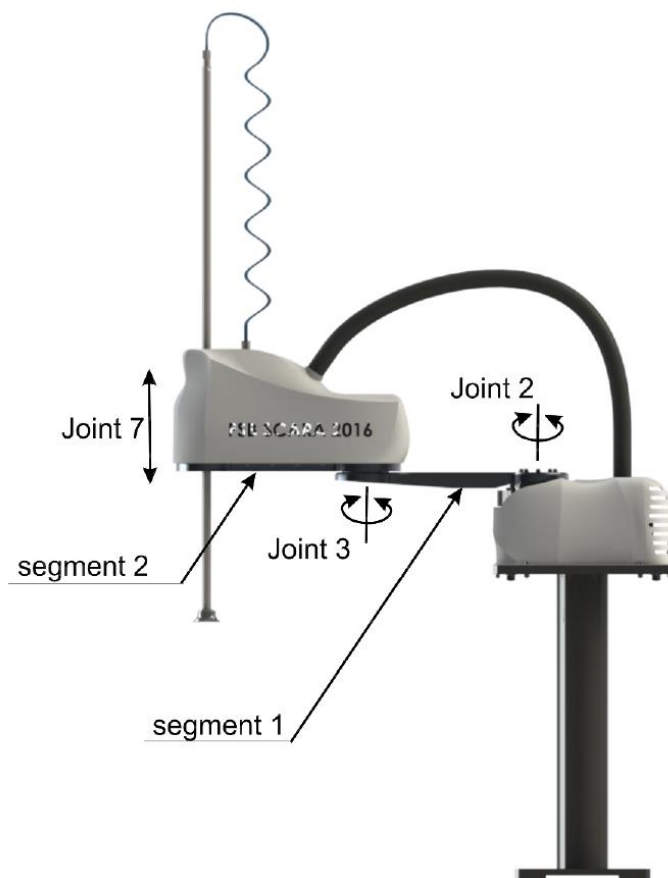
Tablica 7: Razmatrani zahtjevi i parametri koji se javljaju u promatranim podsklopovima uređaja za čišćenje prirubnice spremnika

DIZAJNER2 - MEHANIZAM ZA POGON ČETKI	ZAHTJEVI	Širina kućišta	DIZAJNER1 - MEHANIZAM ZA POLIRANJE PRIRUBNICE	ZAHTJEVI	Normalna sila u opruzi F_o
		Moment torzije diskova			Modul krutosti mat. opruge G
		Sila zatezne remenice			Dopušteno smično naprezanje materijala
		Brzina vrtnje diska			
	PARAMETRI	Promjer pogonske remenice		Dužina navojnog provrta	
		Korak ozubljenja p		Navojni provrt na klizaču	
		Profil remena		Ugradbena duljina opruga	
		Broj gonjenih remenica		Maksimalni hod klizača	
		Promjer gonjene remenice		Promjer žice opruga	
		Duljina remena		Broj opruga	
		Položaj osi gonjenih vratila		Pomak opruge	
		Promjer provrta gonjene remenice		Broj diskova	
		Dužina donjeg rukavca gonjenog vratila		Položaj diskova	
		Navoj donjeg rukavca		Promjer diska četke	
		Unutarnji promjer uležištenja		Širina utora klizača	
		Širina uležištenja			
		Promjer kućišta EM			
		Promjer stege			
		Broj vijaka			
		Promjer čahure			
Položaj zatezne remenice					
Promjer EM					
Pogonski moment torzije					
Brzina vrtnje n_{EM}					

4.2. Robotski manipulator tip SCARA

Bilo je potrebno konstruktivno razraditi mehanizam manipulatora, tip SCARA. Masa predmeta bila je 100 g, tip zahvatnog mehanizma sredstva vakum sisaljka, a medij komprimirani zrak. Projicirana površina radnog prostora bila je $1,14\text{m}^2$, visina radnog prostora $l=0,82$ m, kut rotacije glavnog zgloba 270° , vrijeme ubrzanja glavnog zgloba $0,95$ s, a ostala ubrzanja $0,5$ s. Krakove je bilo potrebno odrediti prema projiciranoj površini. Za dani model bilo je potrebno odrediti dinamičke karakteristike sustava primjenom programskog paketa. [22]

SCARA je manipulator koji ima dva rotacijska i jedan translacijski zglob. Zadatak smo počeli tako da smo u *Solidworks-u* mijenjali parametre dužina segmenata 1 i 2 dok nismo dobili dužine segmenata takve da je bilo moguće dohvatiti svaku točku u zadanom radnom prostoru. Iz poznatih upravljanih koordinata (maksimalnih kuteva zakreta zglobova), vrijednosti masa i momenata inercije segmenata, te zadanih vremena ubrzanja proveli smo direktnu kinematiku i dinamiku robota, kako bi dobili vrijednosti potrebnih momenata motora, maksimalnu brzinu i akceleraciju koje se javljaju u gibanju manipulatora. Ukupni potreban moment u prvoj upravljanoj koordinati jednak je zbroju momenta potrebnog za pokretanje prvog segmenta T11 i momenta potrebnog za pokretanje drugog segmenta T12. Potreban moment motora u drugoj upravljanoj koordinati jednak je momentu za pokretanje drugog zgloba T22. Motori su servo aktuatori *Harmonic Drive* (slika 36.) koji imaju dvije prirubnice, svaka za prihvat na jedan segment. Proces konstruiranja podijeljen je na dva dizajnera. Prvi dizajner vrši modeliranje segmenta 2, dinamičku analizu za pokretanje segmenta 2, odabire motor 2, linearni motor i vakum sisaljku. Nakon toga iterativno se vraća na modeliranje segmenta 2 i usklađuje dimenzije segmenta s dimenzijama odabranog motora. Određene parametre prosljeđuje drugom dizajneru te on kreće s modeliranjem segmenta 1, dinamičkom analizom cijelog manipulatora i vrši FEM analizu. Oko dimenzija prihvata segmenata na motor 2 dizajneri se moraju uskladiti, odnosno moraju surađivati i donesti zajedničku odluku te prilagoditi svaki svoj podsuklop dimenzijama koje su dogovorene pregovaranjem. Ukoliko glavni stup konstrukcije ne zadovolji dopuštena savojna naprezanja, potrebno je promijeniti dimenzije segmenata, odnosno ponoviti proces konstruiranja manipulatora. [35]



Slika 35: Scara manipulator [23]



Slika 36: Servo aktuator Harmonic Drive [23]

Radi preglednosti uvedene su sljedeće oznake: S1 – dužina segmenta 1

S2 – dužina segmenta 2

M2 – *Harmonic Drive* aktuator

Tablica 8: Razmatrani zahtjevi i parametri koji se javljaju u promatranim podsklopovima SCARA manipulatora

DIZAJNER1 – KONSTRUKCIJA I POGON DRUGOG SEGMENTA	ZAHTJEVI	Vrijeme ubravanja drugog rotacijskog zgloba	DIZAJNER2 – KONSTRUKCIJA PRVOG SEGMENTA I PRORAČUN SCARE	ZAHTJEVI	Vrijeme ubravanja glavnog rotacijskog zgloba
		Površina radnog prostora			Kut rotacije glavnog zgloba
		Masa predmeta koji se prenosi			Dopušteno savojno naprezanje za aluminij
		Visina radnog prostora			
		Vrijeme ubravanja translacijskog zgloba			
	PARAMETRI	Masa S2		Masa S1	
		Moment inercije S2		Moment inercije S1	
		Dužina S2		Max. moment u prvom zglobu potreban za pokretanje S2	
		Max. moment u drugom zglobu potreban za pokretanje S2		Max. moment u prvom zglobu potreban za pokretanje S1	
		Promjer S2 za pričvršćivanje vijcima za prirubnicu M2		Dužina S1	
		Broj provrta za vijke na S2 za pričvršćivanje vijcima za M2		Promjer S1 na pričvršćivanje vijcima za M2	
		Moment inercije M2		Broj provrta za vijke na S1 za pričvršćivanje na M2	
		Masa M2		Najveće naprezanje u postolju manipulatora	
		Promjer prirubnice M2 za pričvršćivanje vijcima za S1			
Broj provrta za vijke na M2 za pričvršćivanje vijcima za S1					
Masa linearnog motora					
Potreban podtlak					

5. Zavisnosti konstrukcijskih parametara prikazani u DSM matrici

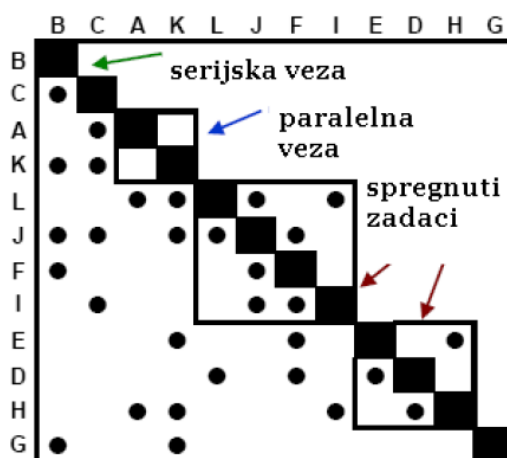
U ovom poglavlju promatrani parametri konstrukcija uređaja za čišćenje prirubnice i Scara manipulatora, prikazani su u obliku DSM matrica, te su prikazane relacije između parametara. Na matrični prikaz relacija između parametara primijenjen je algoritam particioniranja kako bi se uočili i grupirali parametri koji se određuju iteracijom. Particioniranje je provedeno pomoću alata PSM₃₂.

5.1. DSM (Design structure matrix)

DSM je matrični prikaz relacija između komponenti nekog sustava. Matrica sadrži informacije o tome koji su parametri potrebni da bi se pokrenula određena aktivnost i koji zadaci ili parametri koriste izlazne informacije. Matrice su podatne za uporabu na računalu, a pomoću njih se lako mogu opisivati i modelirati sustavi jer mogu ukazati na postojanje ili nedostatak veza između elemenata sustava. Prvi stupac i prvi redak čine nazivi elemenata matrice. Ukoliko između dva elementa postoji veza, uobičajeno je staviti oznaku "X". Kada su unesene sve relacije između promatranih elemenata, potrebno je provesti analizu kako bi se utvrdilo koji elementi imaju paralelnu ili serijsku vezu, a posebno je važno uočiti koji elementi su spregnuti. Takvi elementi određuju se povratnim petljama, za njihovo dobivanje potrebne su ulazne informacije koje nisu dostupne u trenutku izvođenja. Dobivanje zavisnog parametra biti će temeljeno na početnoj pretpostavci. Moguće je manipulirati matricom kako bi uočile povratne petlje te eventualno smanjile. Ta manipulacija naziva se algoritam particioniranja ili preslagivanja (slika 38.). Jednom kada je DSM particioniran, identificiraju se zadaci u serijama i izvršavaju se sekvencijalno. Također se uočavaju paralelni zadaci koji se mogu izvršavati paralelno. Za spregnute zadatke ili parametre teško je odrediti redosljed izvođenja i često se sastoje od nekoliko iterativnih ciklusa. Oznake u pojedinačnim redovima prikazuju sve zadatke čiji izlaz je neophodan za izvršenje zadatka koji odgovara tom redu. Isto tako, čitanje pojedinačnog stupca pokazuje koji zadatak prima informacije od zadatka koji odgovara tom stupcu. [19]

OSNOVNE VEZE MEĐU ELEMENTIMA			
VEZA	PARALELNI (nezavisni zadaci)	SEKVENCIJALNI (zavisni zadaci, serijski)	MEĐUOVISNI (zavisni zadaci, spregnuti)
BINARNA MATRICA			

Slika 37: Osnovne veze među elementima s obzirom na redoslijed izvođenja zadataka/dobivanja parametara [19]



Slika 38: Particionirana DSM matrica [19]

TIPOVI DSM-a	INTERPRETACIJA	PRIMJENA	METODE
(«task-based») baziran na zadacima	Zadatak/ ulazno stanje/izlazne veze	Raspored projektnih zadataka, redoslijed aktivnosti	Preslagivanje ili particioniranje , cijepanje (cijepanje), vrednovanje
(«parametar-based») baziran na parametrima	Parametri, bodovanja i neophodni slučajevi	Apstraktni nivo procesa	Particioniranje , cijepanje (cijepanje), vrednovanje
(«team-based») baziran na timovima	Karakteristike veza između više timova	Integracija timova, organizacijski menadžment	Klaster analiza
«component-based» baziran na komponentama	Mnogostruke veze između komponenti	definicija modula sustava – integracija u podsustave	Klaster analiza

Slika 39: Tipovi podataka obuhvaćeni DSM-om [22]

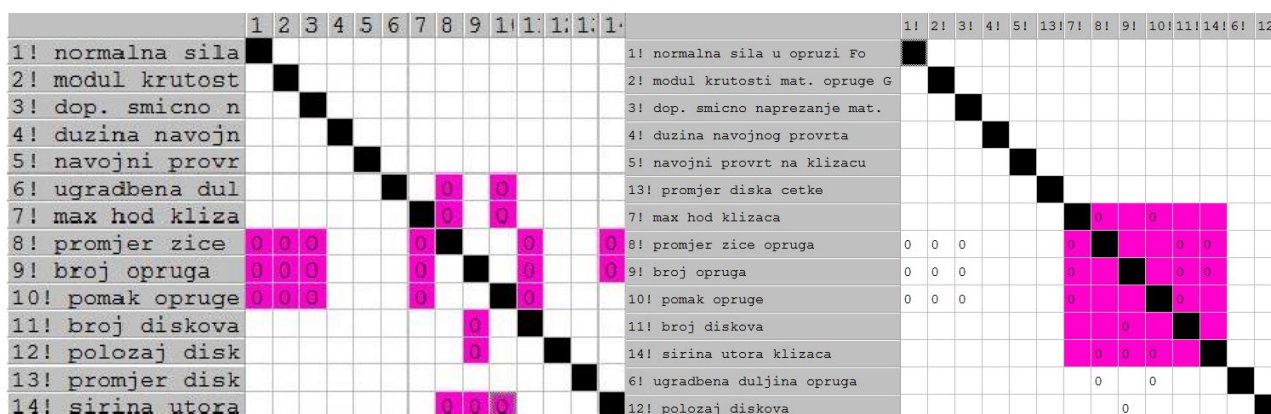
5.2. DSM matrica veza između konstrukcijskih parametara za razmatrane podsklopove proizvoda Samohodni uređaj za čišćenje prirubnice spremnika

MEHANIZAM ZA POLIRANJE PRIRUBNICE	MEHANIZAM ZA POGON ČETKI																					
	širina utora klizača																					
promjer diska četke																						
položaj diskova																						
broj diskova																						
pomak opruge																						
broj opruga																						
promjer žice opruga																						
max hod klizača																						
ugradbena dužina opruga																						
navojni provrt na klizaču																						
dužina navojnog provrta																						
dop. smično naprezanje mat.																						
modul krutosti mat. opruge G																						
normalna sila u opruzi Fo																						
brzina vrtnje n																						
pogonski moment torzije																						
promjer EM																						
položaj zatezne remenice																						
promjer čahure																						
broj vijaka																						
promjer stege																						
promjer kućišta EM																						
širina uležištenja																						
navoj donjeg rukavca																						
dužina donjeg rukavca goni. vratila																						
promjer provrta goni. vratila																						
položaj osi goni. vratila																						
dužina remena																						
promjer goni. remenice																						
broj goni. remenica																						
profil remena																						
korak ozubljenja p																						
promjer pogonske remenice																						
brzina vrtnje diska																						
sila zatezne remenice																						
moment torzije diskova																						
širina kućišta																						
MEHANIZAM ZA POGON ČETKI																						
MEHANIZAM ZA POLIRANJE PRIRUBNICE																						

Slika 40: DSM matrica bazirana na parametrima podsklopa za poliranje i podsklopa za pogon četki uređaja za čišćenje prirubnice spremnika

Kreirana je matrica koja prikazuje relacije između grupe parametara iz oba promatrana podsklopa, tj dvije matrice relacija unutar istog podsklopa (slika 40. plava boja) i dvije matrice relacija između parametara različitih podsklopova (slika 40. roza boja). Veze između parametara su ili funkcijske zavisnosti ili veze nastale zbog fizikalnih ograničenja, tj. vrijednosti određenih parametara koje fizikalno ograničavaju druge parametre te na taj način utječu na njihove iznose. Crvenom bojom naznačeni su spregnuti parametri, odnosno parametri podsklopa za poliranje koji utječu na vrijednosti parametara pogona četki, i obratno parametri podsklopa za pogon četki koji utječu na vrijednosti parametara mehanizma za poliranje. Prvu grupu spregnutih parametara čine: broj diskova, položaj diskova, broj gonjenih remenica i položaj osi gonjenih vratila. Drugu grupu spregnutih parametara čine: promjer provrta gonjene remenice, dužina donjeg rukavca gonjenog vratila, dužina navojnog provrta, i vrsta navojnog provrta na klizaču. Nakon procesa pregovaranja i uvjeravanja, dizajneri donose odluku vezanu za vrijednosti spregnutih parametara, te je potrebno poslati poruku za promjenom dimenzija onome koji mora mijenjati parametre na podsklopu koji konstruira.

Particioniranje je provedeno u alatu PSM₃₂, koji se može besplatno preuzeti s *Interneta* [20]. Analizom parametara mehanizma za poliranje prirubnice uočena je grupa parametara čije se vrijednosti određuju iterativno, s time da se kao inicijalne vrijednosti uzmu pretpostavljene vrijednosti (slika 41). Particioniranjem mehanizma za pogon četki uočeno je da su sve veze između parametara sekvencijalne ili paralelne.



Slika 41: Partitioniranje parametara mehanizma za poliranje prirubnice u alatu PSM₃₂ (lijevo prije partitioniranja, desno nakon partitioniranja)

Za proizvod Scara manipulator također je kreirana matrica koja prikazuje relacije između dvije grupe parametara, tj dvije matrice relacija unutar iste grupe parametara (slika 42. plava boja) i dvije matrice relacija između parametara različitih grupa (slika 42. roza boja). Veze između parametara su ili funkcijske zavisnosti ili veze nastale zbog fizikalnih ograničenja, tj. vrijednosti određenih parametara koje fizikalno ograničavaju druge parametre te na taj način utječu na njihove iznose. Crvenom bojom naznačeni su spregnuti parametri, odnosno parametri konstrukcije i pogona drugog segmenta koji utječu na vrijednosti parametara konstrukcije prvog segmenta i odabir glavnog pogona, i obratno parametri podsklopa konstrukcije prvog segmenta i koji utječu na vrijednosti parametara konstrukcije i pogona drugog segmenta. Spregnuti parametri su oni čije vrijednosti je potrebno uskladiti kako bi prihvat segmenta 1 (dizajner 2) na prirubnicu motora 2 (dizajner 1) bio ostvariv. Spregnuti parametri su: promjer segmenta 1 za pričvršćivanje vijcima za motor 2, broj provrta za vijke na segmentu 1 za pričvršćivanje vijcima za motor 2, promjer prirubnice motora 2 za pričvršćivanje vijcima za segment 1 te broj provrta za vijke na prirubnici motora 2 za pričvršćivanje vijcima za segment 1. Nakon što se dizajneri u procesu pregovaranja odluče za dimenzije prirubnice motora, odgovarajuće dimenzije segmenta 1 ili za potpuno nove dimenzije, potrebno je poslati poruku za promjenom, ukoliko je to potrebno.

Algoritam particioniranja proveden je u alatu PSM₃₂. Analizom grupe parametara vezanih za konstrukciju i pogon segmenta 2, uočena je skupina parametara čije vrijednosti se određuju iterativnim postupkom (slika 43.). Algoritmom particioniranja i među parametrima s kojima radi drugi dizajner (konstrukcija prvog segmenta i proračun manipulatora), uočena je skupina parametara čije vrijednosti se određuju iteracijom (slika 44).

	1!	2!	3!	6!	4!	5!	7!	8!	9!	10!	11!	12!	13!
1! vrijeme ubrzavanja drugog rotacijskog zgloba	█												
2! površina radnog prostora		█											
3! masa predmeta koji se prenosi			█										
6! dužina S2				0									
4! masa S2				0		█	█	0	0				
5! moment inercije S2				0		█	█	0	0				
7! max moment u drugom zglobu za pokret	0	0	0	0	0	0	█	█					
8! promjer S2 za pricvrscivanje vijcima							█	█	0				
9! broj provrta za vijke na S2 za pricvrscivanje							█	█	0				
10! moment inercije M2										█			
11! masa M2										0			
12! promjer M2 za pricvrscivanje vijcima										0			
13! broj provrta za vijke na M2 za pricvrscivanje										0			

Slika 43: DSM matrica grupe parametara konstrukcije i odabira pogona segmenta 2 nakon particioniranja

	1!	2!	3!	4!	5!	9!	10!	6!	7!	8!	11!
1! vrijeme ubrzavanja glavnog rotacijskog zgloba	█										
2! kut rotacije glavnog zgloba		█									
3! dopusteno savojno naprezanje za aluminij			█								
4! masa S1				█							
5! moment inercije S1					█						
9! promjer S1 za pricvrscivanje vijcima na M2						█					
10! broj provrta za vijke na S1 za pricvrscivanje na M2							█				
6! max moment u prvom zglobu potreban za pokretanje S2	0	0		0	0			█	█	0	█
7! max moment u prvom zglobu potreban za pokretanje S1	0	0		0	0			█	█	0	█
8! dužina S1								█	█	█	0
11! najveće naprezanje u postolju manipulatora				0	0			0	0	0	█

Slika 44: DSM matrica grupe parametara konstrukcije segmenta 1 i proračuna manipulatora nakon particioniranja

6. Model procesa određivanja vrijednosti konstrukcijskih parametara u timskom radu metodom obojenih Petrijevih mreža

Provedenom DSM analizom procesa određivanja konstrukcijskih parametara, dobiven je prikaz procesa na statičkoj razini. Radi praćenja dinamike procesa, kreirani su modeli procesa određivanja vrijednosti konstrukcijskih parametara u timskom radu za oba razmatrana proizvoda. U modele su uključeni prikazi zavisnosti parametara, iteracija te procesi pregovaranja i dogovaranja oko vrijednosti spregnutih parametara. Modeli su kreirani pomoću obojenih Petrijevih mreža u alatu *CPN Tools* te su pokrenute simulacije procesa.

6.1. Simulacija procesa određivanja vrijednosti konstrukcijskih parametara u timskom radu za razmatrane podsklopove Samohodnog uređaja za čišćenje prirubnice spremnika

Dobivena je složena mreža procesa konstruiranja, te je zbog toga hijerarhijski podijeljena na nekoliko podmreža (slika 45.). Pri automatskom pokretanju simulacije, kartice s podmrežama koje sadrže tranzicije spremne za izvršavanje (*enabled*), automatski se prikazuju.

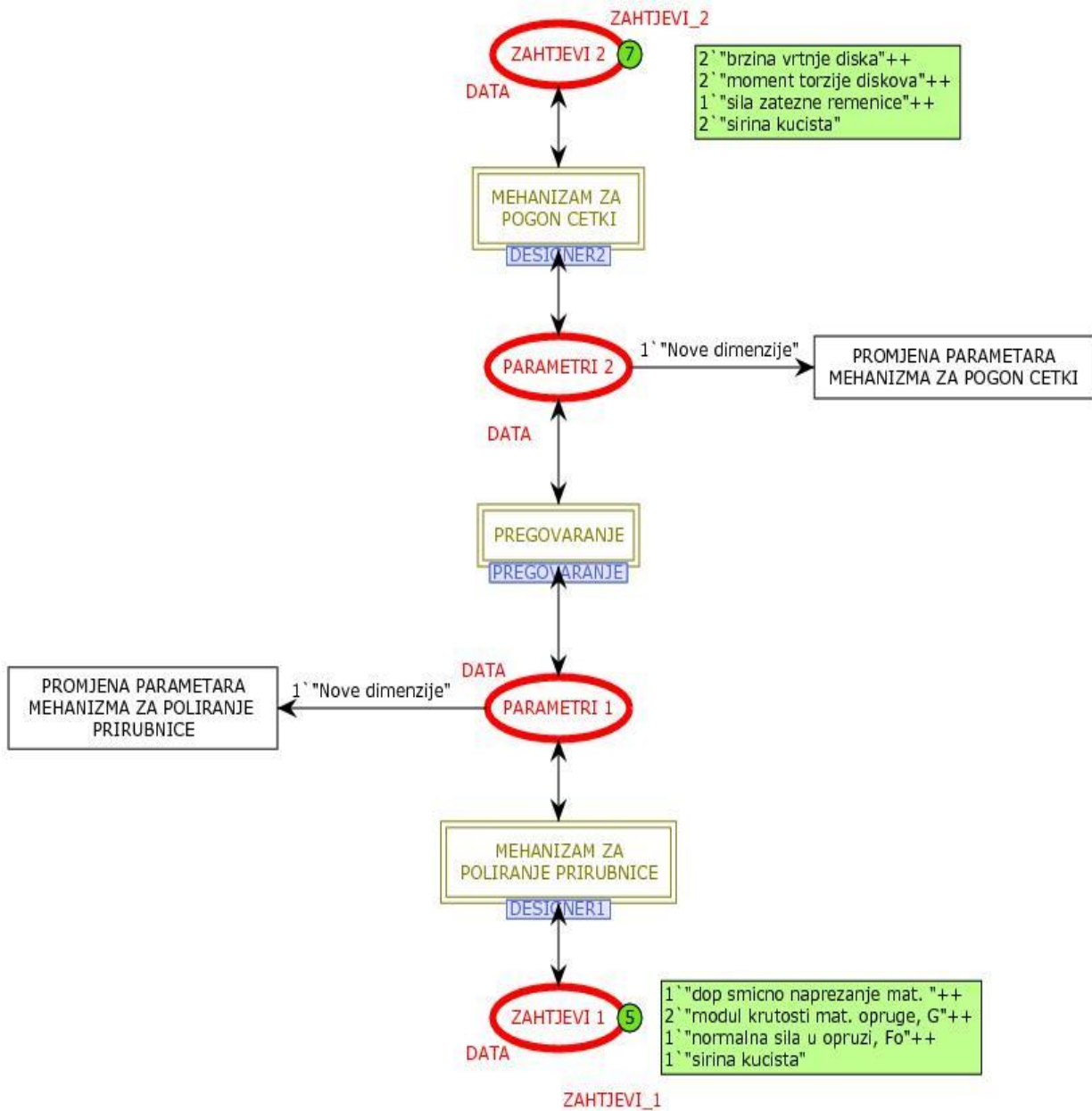


Slika 45: Hijerarhijski prikaz mreže uređaja za čišćenje prirubnice spremnika

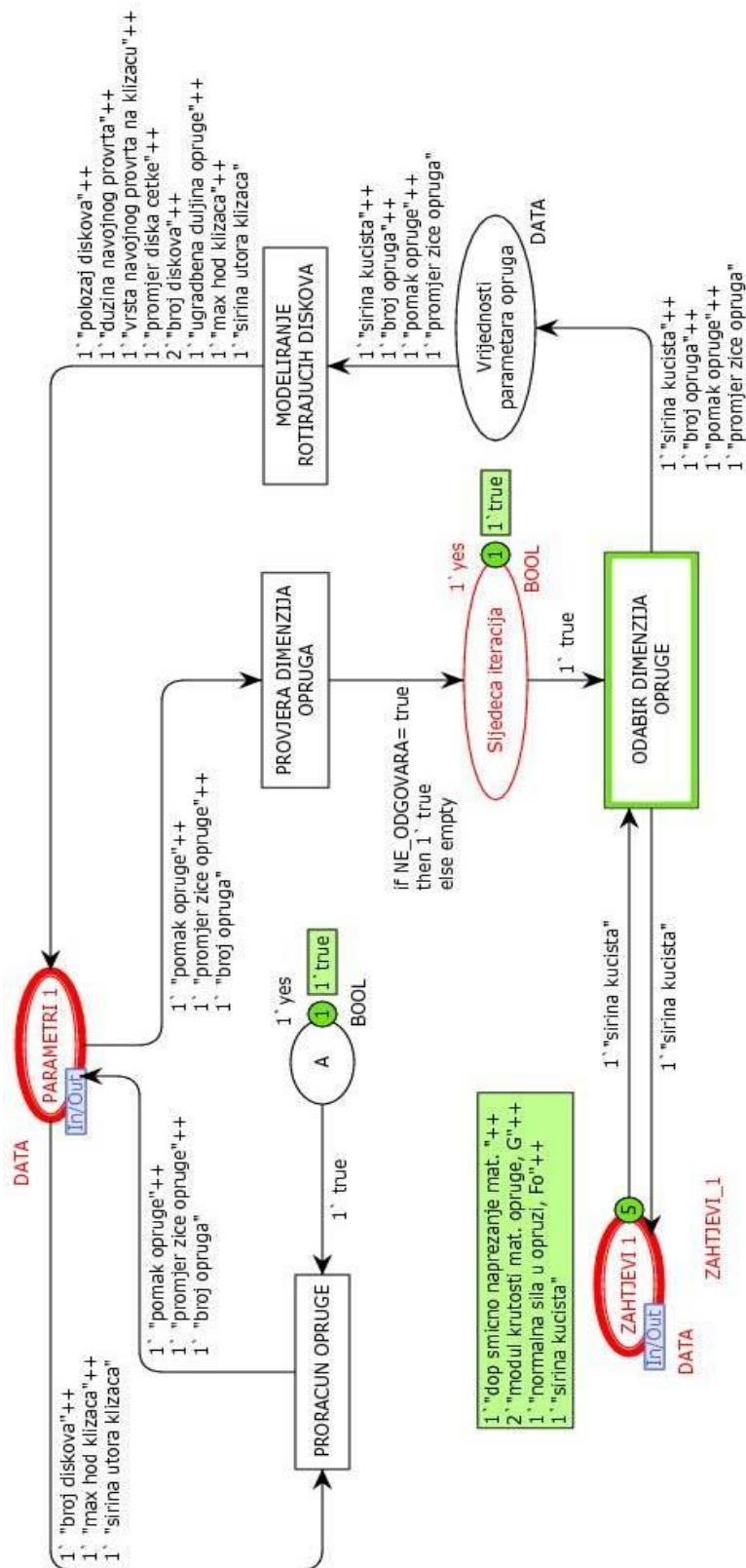
Proces konstruiranja počinje paralelnim konstruiranjem mehanizma za poliranje prirubnice (dizajner 1) i mehanizma za pogon četki (dizajner 2). Dizajner 1 odabire inicijalne dimenzije opruga, na temelju čega modelira rotirajuće diskove. Sada su poznati parametri broj diskova, maksimalan hod klizača i širina utora klizača te je moguće proračunom odrediti kakve su opruge potrebne za ostvarivanje pritisne sile za poliranje. Dizajner se iterativno vraća na

odabir opruga te prema tome mijenja dimenzije rotirajućih diskova. Dizajner 2 odabire komponente zupčastog remenskog prijenosa, elektromotor te modelira sklop zatezne remenice. Ukoliko je potrebno, radi ostvarivanja zadanog prijenosnog omjera, odabire druge komponente remenskog prijenosa. Na temelju zahtjeva i dobivenog parametra promjera provrta gonjene remenice vrši se dimenzioniranje gonjenog vratila i odabir uležištenja. Sada su poznati svi parametri za modeliranje čahura za uležištenje gonjenih vratila, prihvata za kućište te stege za prihvrat odabranog elektromotora. U pozicijama Parametri 1 i Parametri 2 sada se nalaze vrijednosti spregnutih parametara i započinje proces pregovaranja. Nakon dohvaćanja njihovih vrijednosti ulazi se u proces usklađivanja parametara. U tranziciji provjere kompatibilnosti provjerava se da li su parametri već usklađeni. Ukoliko nisu usklađeni, unutar pod mreže argumentiranje dizajneri iznose argumente zašto su odabrali određene vrijednosti spregnutih parametara, te obrazlažu zašto bi ili nebi bilo moguće te vrijednosti mijenjati. Proces je modeliran kao povratna petlja. Nakon argumentiranja slijedi uvjeravanje i odabir donositelja odluke (unaprijed su definirani mogući kandidati: dizajner1, dizajner2 ili koordinator). Donosi se odluka te se odabrana vrijednost pohranjuje u varijablu ODLUKA. Zahtjev za promjenom prosljeđuje se onom dizajneru koji mora mijenjati dimenzije u svom podsklopu (moguće je da oba dizajnera trebaju napraviti promjene ako su odabrane potpuno nove vrijednosti parametara). Dizajneru čije vrijednosti spregnutih parametara su jednake odlučnim šalje se poruka da promjena nije potrebna.

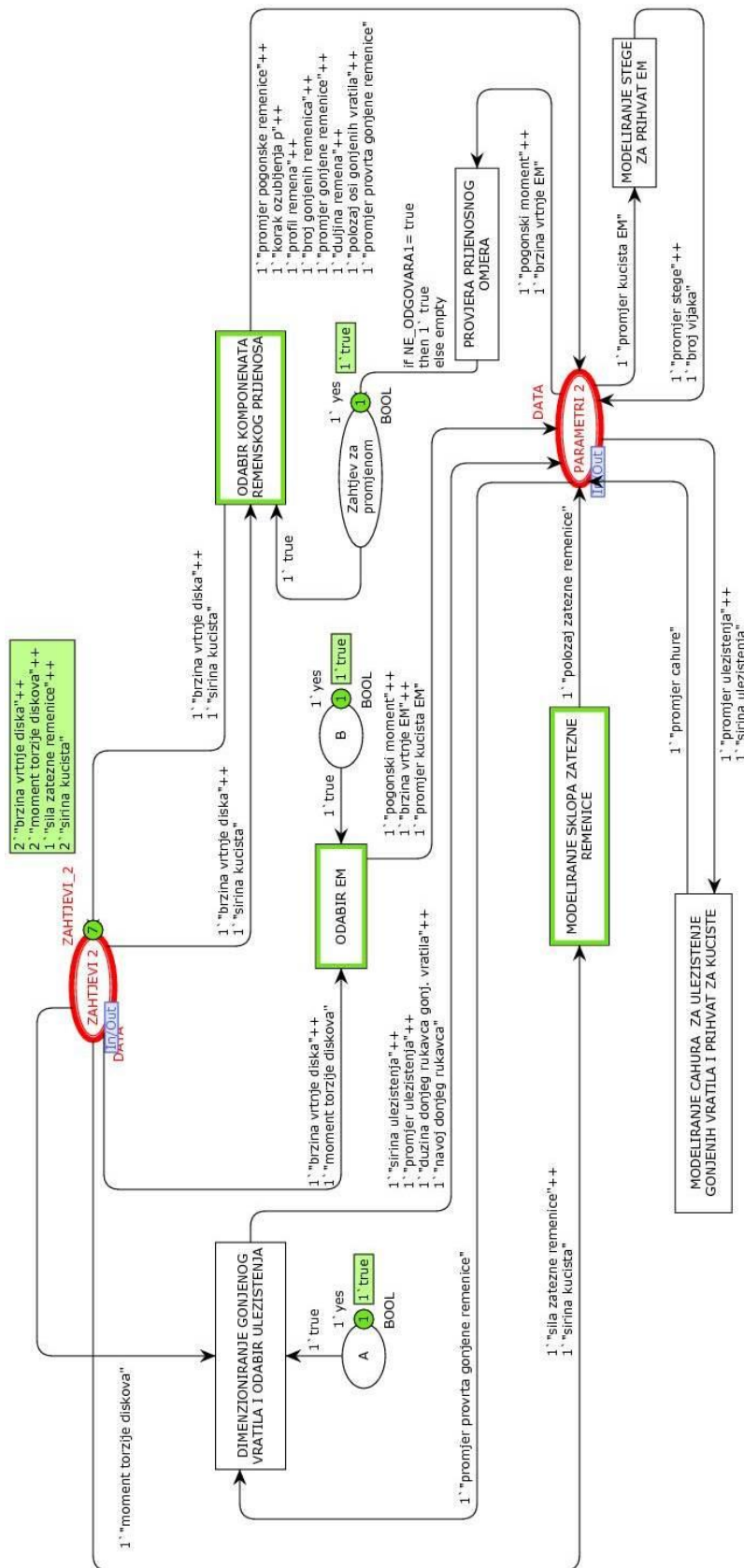
Pri ručnom odabiru izvršavanja omogućenih tranzicija potrebno je odabrati vrijednosti koje će varijable poprimiti. Taj postupak naziva se izvršavanje tranzicije s omogućenim ograničenjem. Ukoliko simulaciju procesa pokrećemo automatski i definiramo broj koraka, varijable će poprimiti slučajne vrijednosti iz zadanog raspona.



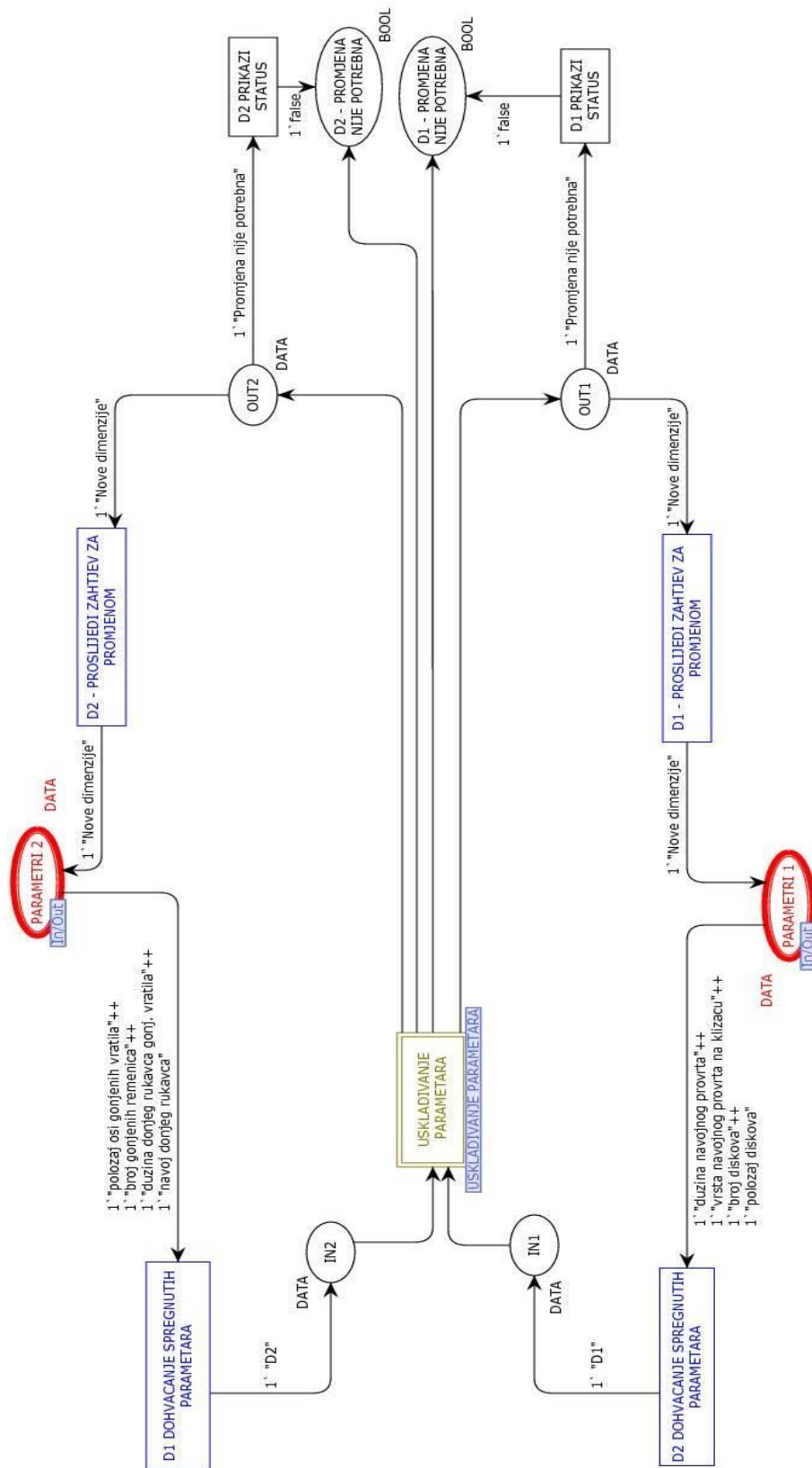
Slika 46: Glavna obojena Petrijeva mreža procesa određivanja konstrukcijskih parametara u timu od dva člana za podsklopove samohodnog uređaja za čišćenje prirubnice



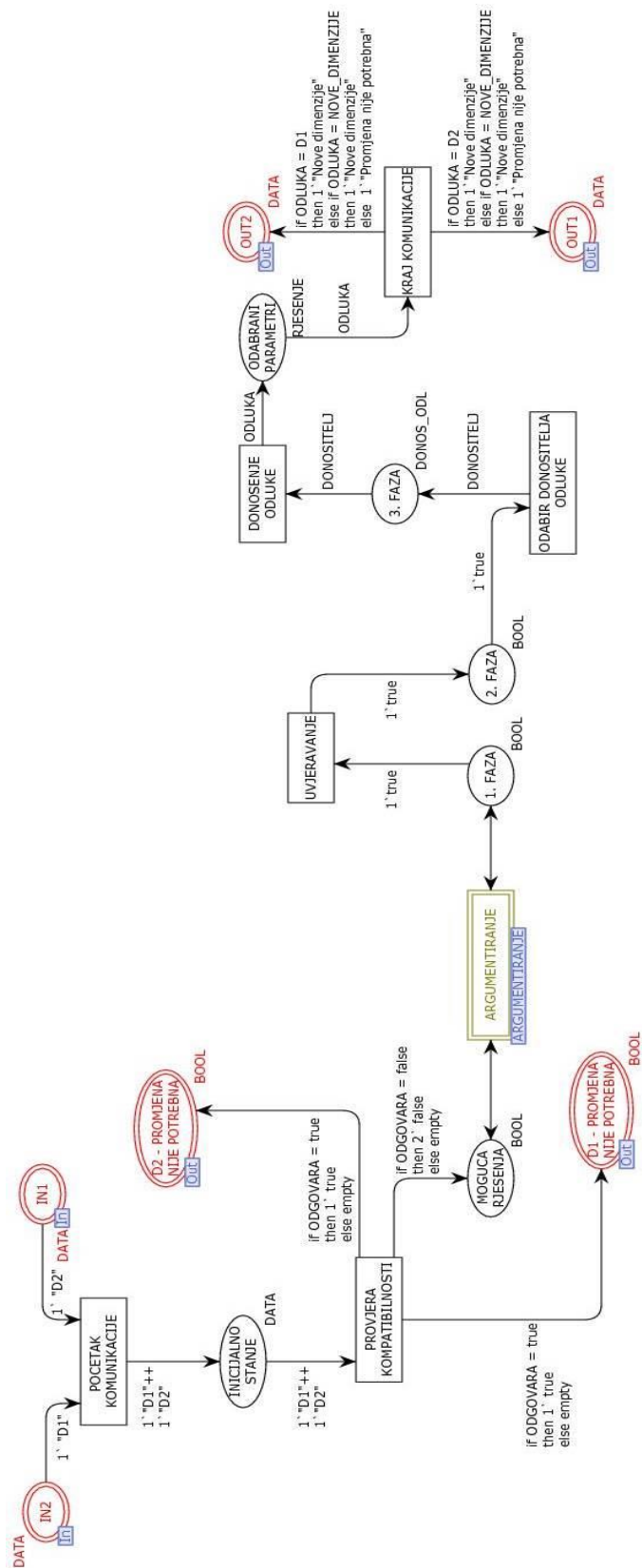
Slika 47: Podmreža određivanja parametara mehanizma za poliranje prirubnice – Dizajner 1



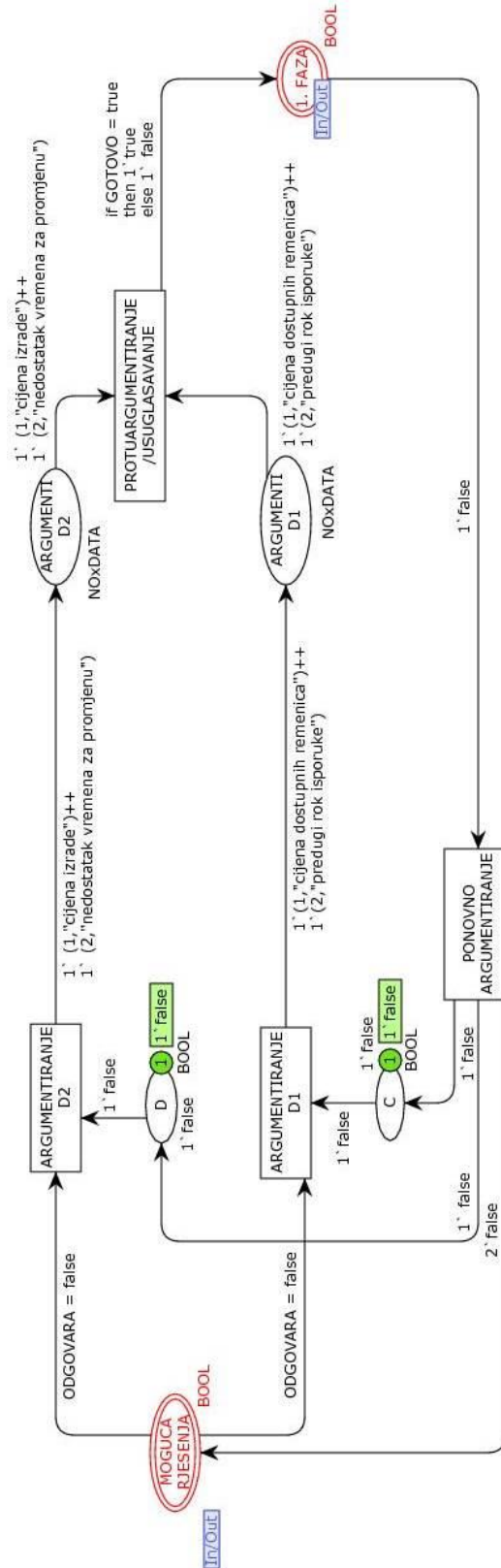
Slika 48: Podmreža određivanja parametara mehanizma za pogon četki – Dizajner 2



Slika 49: Podmreža procesa pregovaranja – uređaj za čišćenje prirubnice spremnika



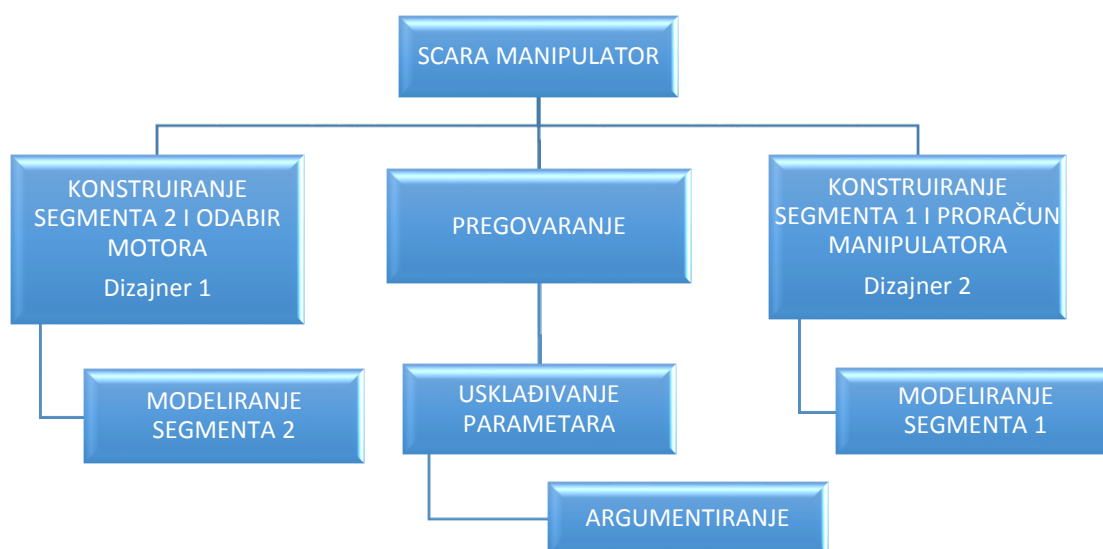
Slika 50: Podmreža usklađivanja parametara – uređaj za čišćenje prirubnice spremnika



Slika 51: Podmreža procesa argumentiranja – uređaj za čišćenje prirubnice spremnika

6.2. Simulacija procesa određivanja vrijednosti konstrukcijskih parametara u timskom radu za Scara manipulator

Dobivena je složena mreža procesa konstruiranja, te je zbog toga hijerarhijski podijeljena na nekoliko podmreža (slika 52.). Pri automatskom pokretanju simulacije, kartice s podmrežama koje sadrže tranzicije spremne za izvršavanje (*enabled*), automatski se prikazuju.

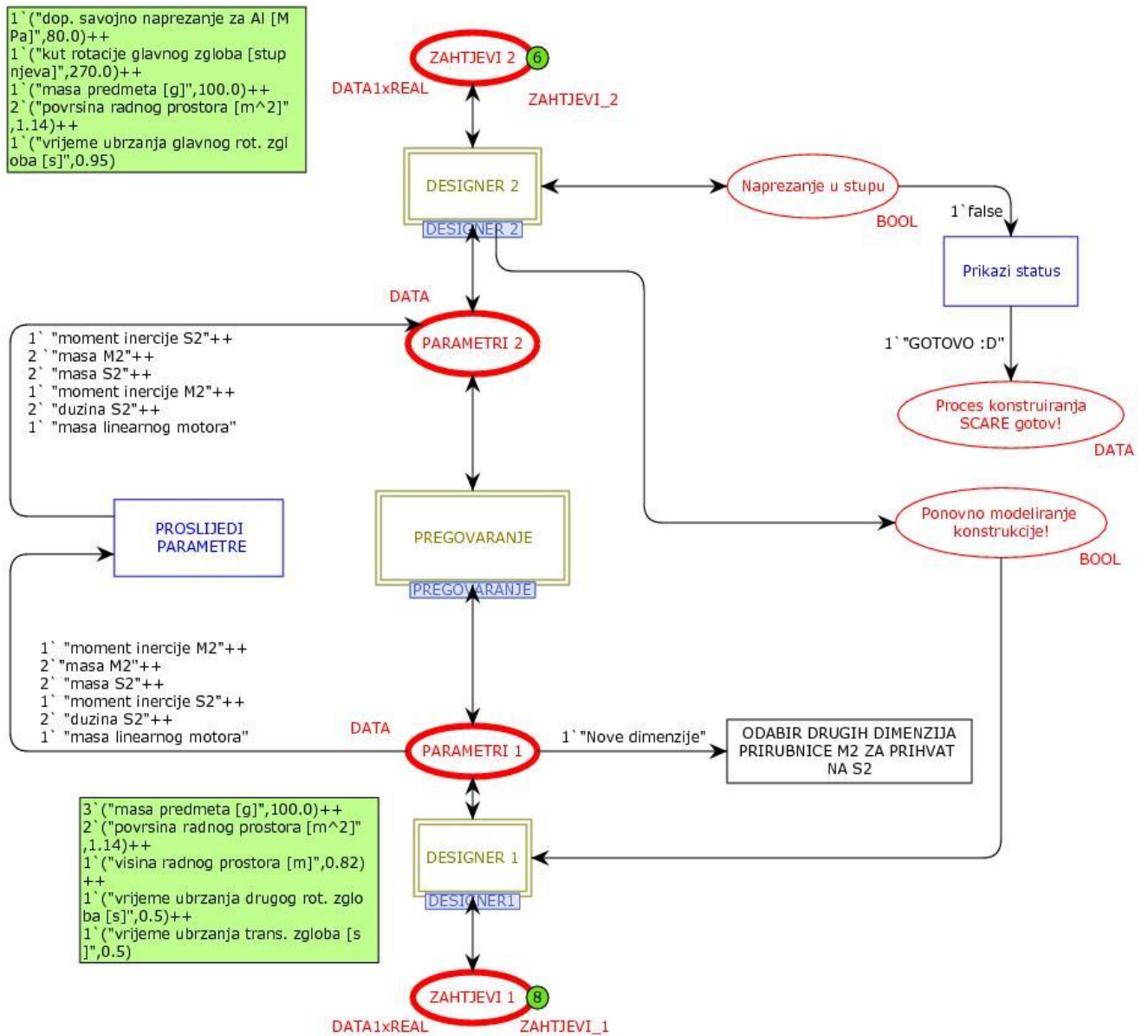


Slika 52: Hijerarhijski prikaz mreže Scara manipulatora

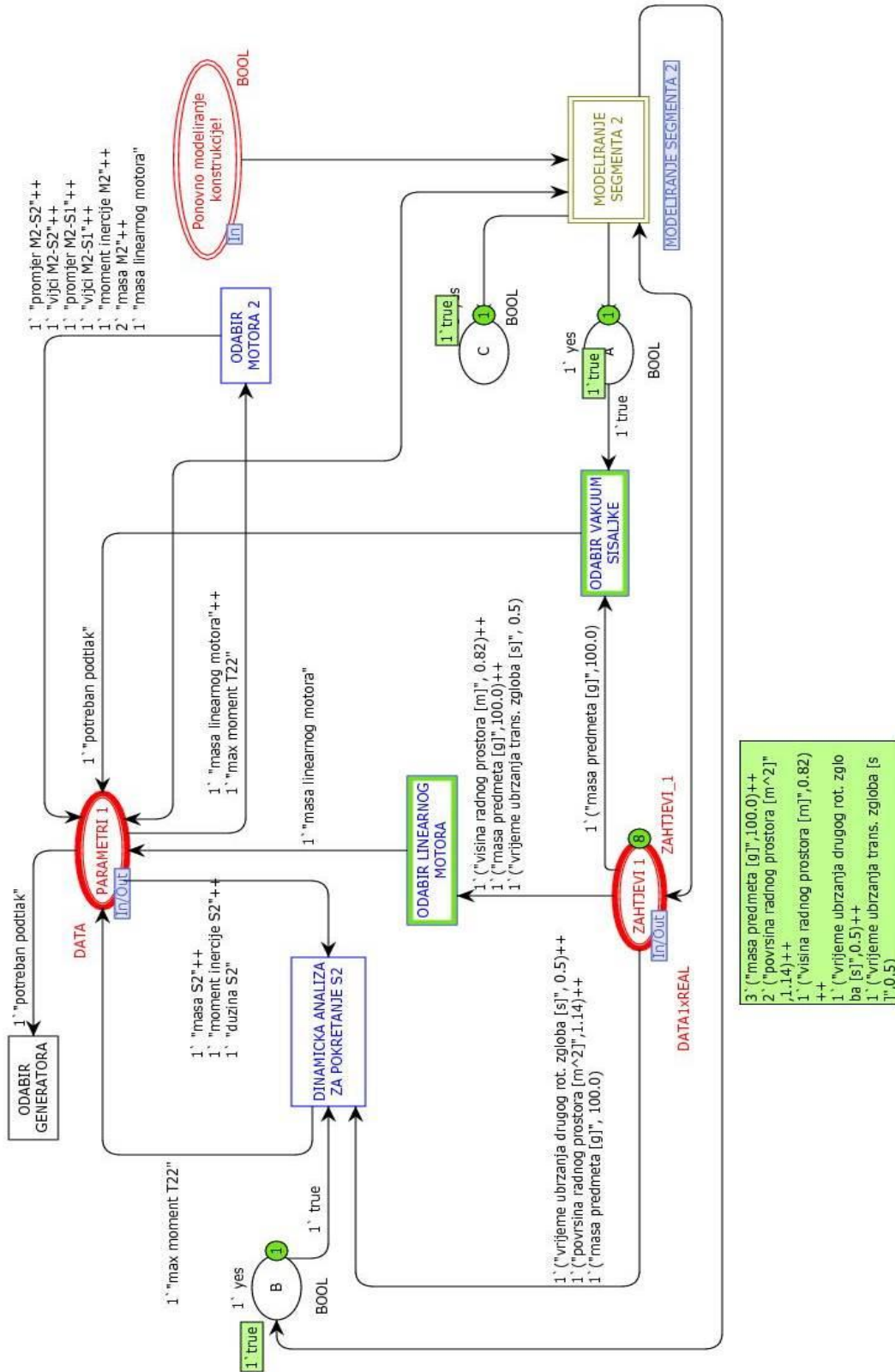
Proces konstruiranja počinje paralelnim modeliranjem segmenata, prvi dizajner modelira segment 2, dok drugi modelira segment 1. Pritom je moguće pratiti vrijednosti dužina segmenta 2, inicijalno odabranu pomoću predefinirane *random* funkcije iz zadanog raspona te svaku sljedeću vrijednosti, generiranu potrebnim iteracijama (slika 61.). Paralelno s modeliranjem segmenata provodi se odabir linearnog motora i vakuum sisaljke. Nakon što su dobiveni potrebni parametri vrši se dinamička analiza, dobivaju se potrebni momenti torzije i odabire se motor 2. Sada je dizajner 2 u mogućnosti da provjeri da li dimenzije prirubnice motora 2 odgovaraju dimenzijama prihvaća segmenta 2, te ukoliko je potrebno, vrši ponovno modeliranje tog segmenta. Uz omogućeno izvršenje tranzicije dinamičke analize za pokretanje segmenta 2, paralelno se vrši odabir generatora. U pozicijama Parametri 1 i Parametri 2 sada se nalaze sve vrijednosti koje dizajner 1 mora proslijediti dizajneru 2 kako bi proveo dinamičku analizu cijelog manipulatora. U parametrima se nalaze i spregnuti parametri, te je omogućeno izvršavanje procesa pregovaranja oko njihovih vrijednosti. Nakon dohvaćanja spregnutih parametara dolazi se u podmrežu usklađivanje parametara. U tranziciji provjera kompatibilnosti

provjerava se da li vrijednosti već odgovaraju te ukoliko promjena nije potrebna, šalje se odgovarajuća poruka. Ukoliko je potrebno pregovarati i odlučiti se za vrijednosti koje će biti konačne za ostvarivanje prihvata segmenta za motor 2, ulazi se u podmrežu argumentiranja. U argumentiranju svaki dizajner iznosi svoja obrazloženja zašto su se odlučili za trenutne vrijednosti spregnutih parametara te je moguće taj proces ponoviti više puta što je ostvareno povratnom petljom. Nakon toga oni se međusobno uvjeravaju, definira se donositelj odluke te se poruka o konačnoj odluci prosljeđuje u mreže svakog dizajnera. Nakon pregovaranja, dizajner 2 provodi *FEM* analizu naprezanja u glavnom stupu manipulatora, te ukoliko dopušteno naprezanje nije zadovoljeno, omogućeno je ponovno izvršavanje modeliranja segmenata 1 i 2 kako bi se promjenom dužina segmenata, promijenio položaj motora 2 koji najviše savojno opterećuje glavni stup manipulatora. Ukoliko je *FEM* analiza zadovoljena, proces konstruiranja manipulatora je gotov.

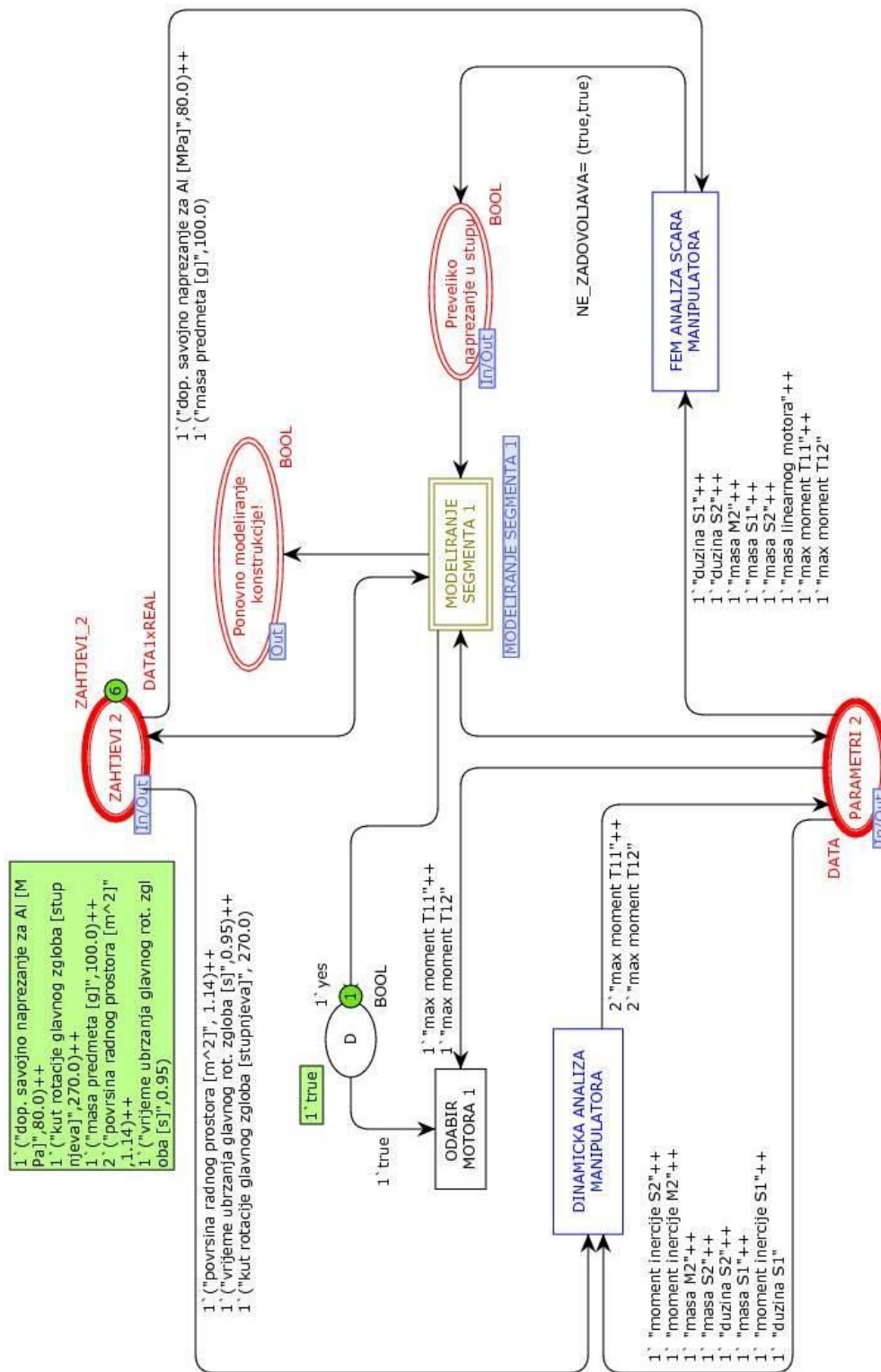
Pri ručnom pokretanju simulacije moguće je odabrati željene ishode izvršenja tranzicija ako se na izlaznim strelicama nalaze varijable. Pri automatskom pokretanju simulacije varijable poprimaju slučajne vrijednosti iz zadanih raspona. Prikazanim modelom moguće je pratiti tokove parametara u procesu konstruiranja cijelog proizvoda, a podržan je i proces pregovaranja oko vrijednosti spregnutih parametara.



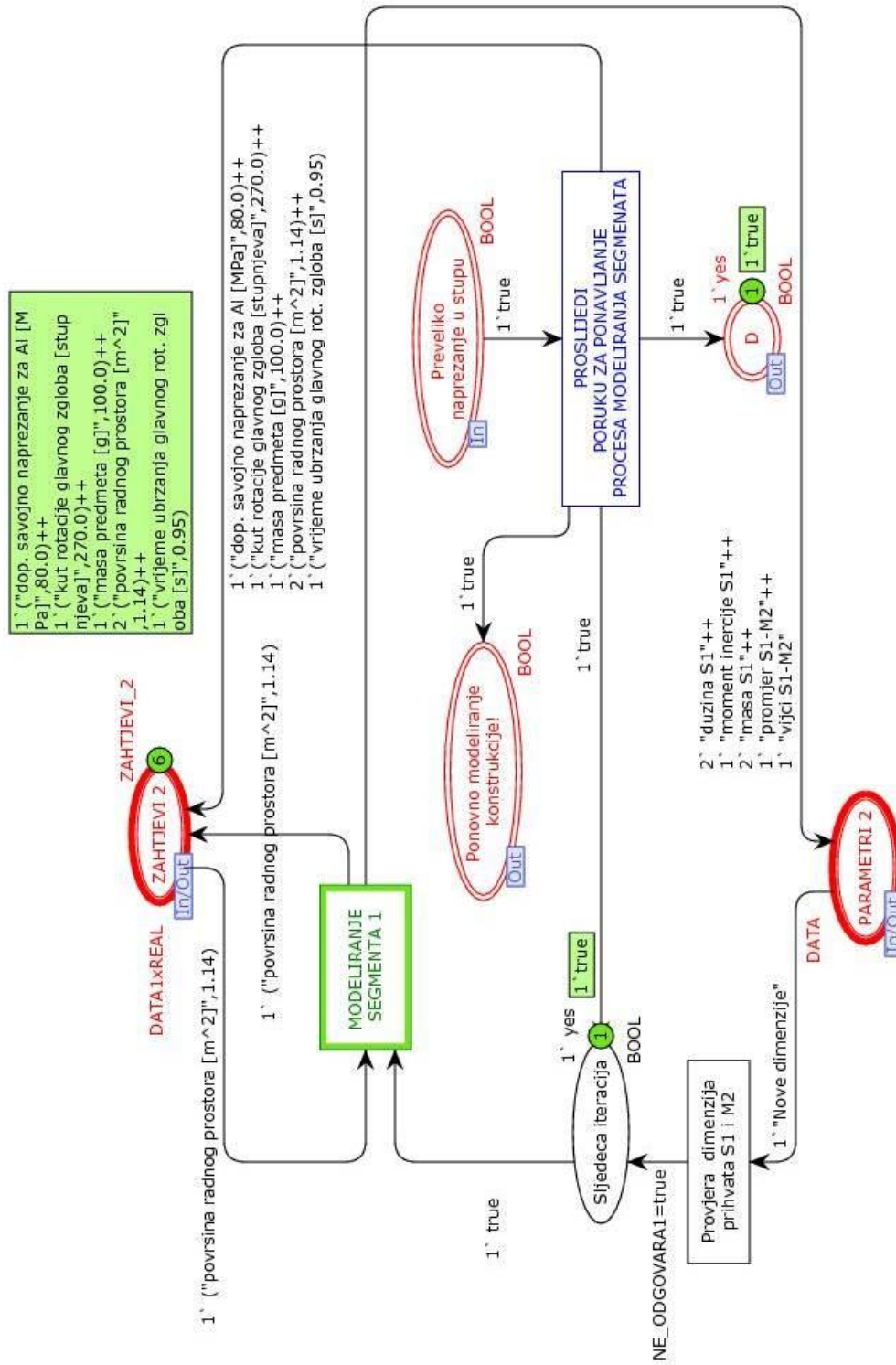
Slika 53: Obojena Petrijeva mreža procesa određivanja konstrukcijskih parametara u timu od dva člana za Scara manipulator



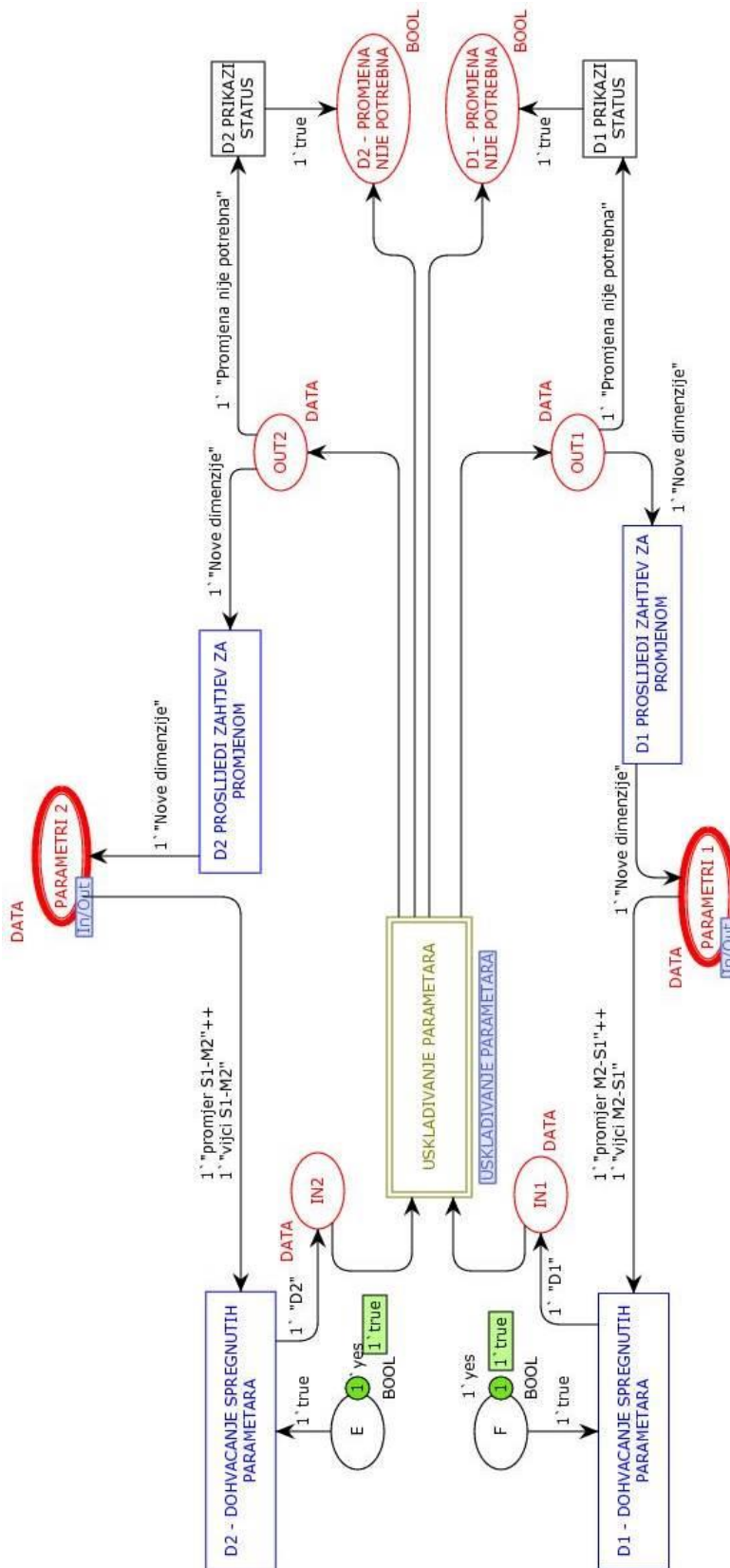
Slika 54: Podmreža odredivanja parametara za konstruiranje segmenta 2 i odabir motora 2 – Dizajner 1



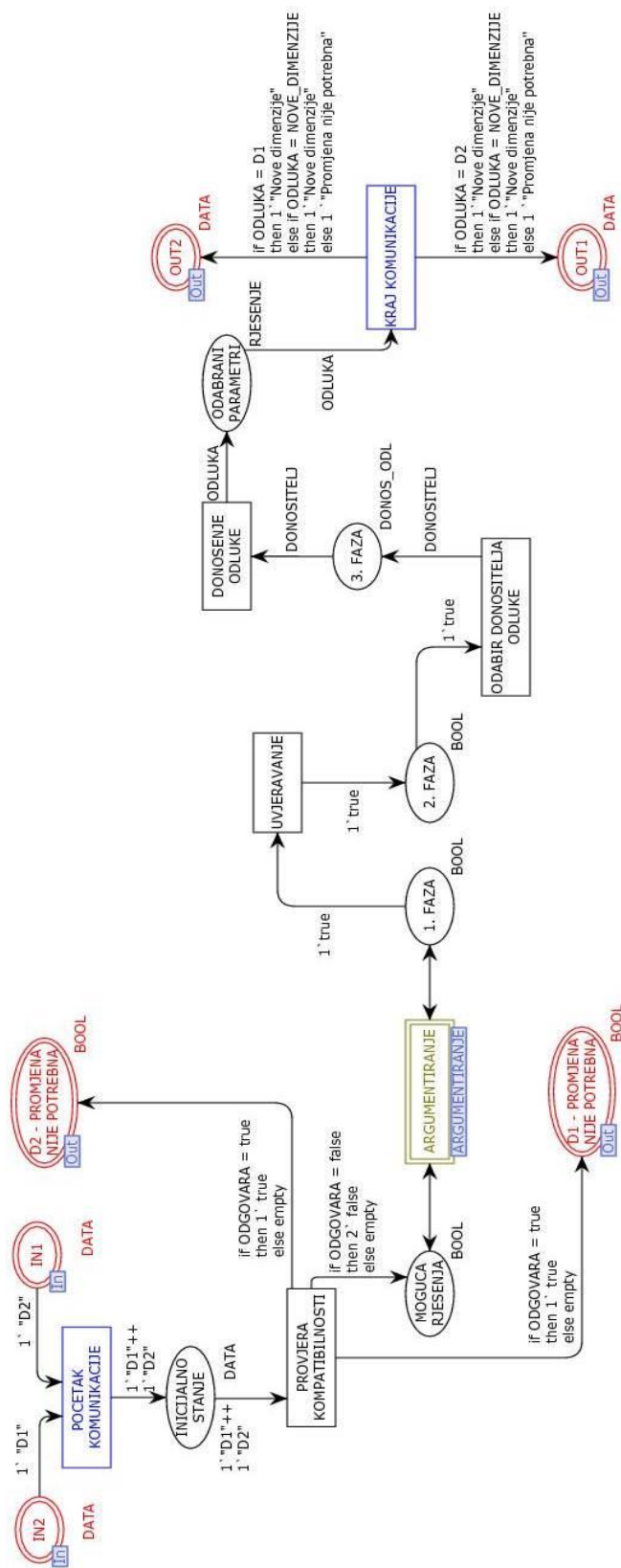
Slika 56: Podmreža određivanja parametara za konstruiranje segmenta 1 i proračun manipulatora – Dizajner 2



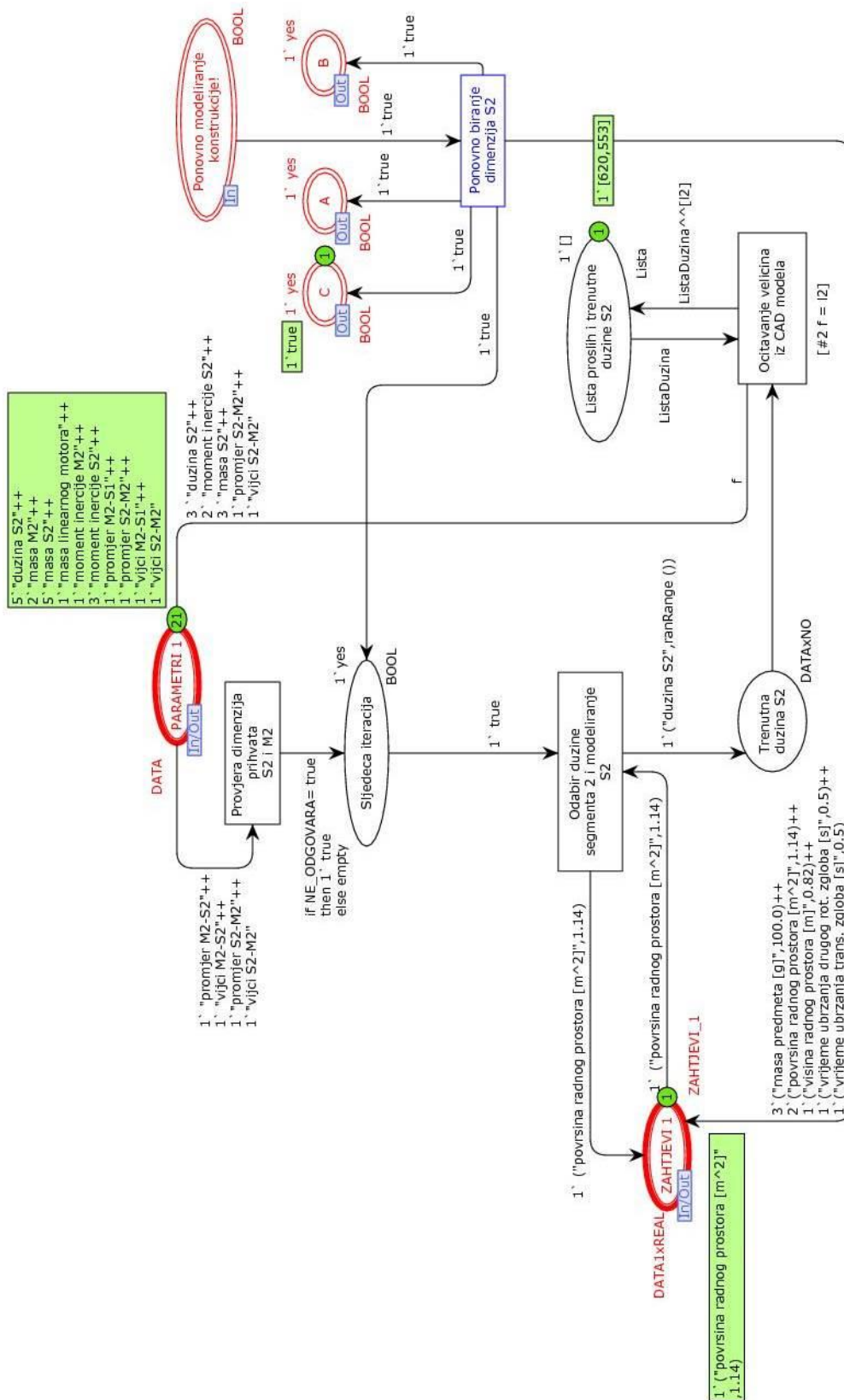
Slika 57: Podmreža procesa modeliranja segmenta 1



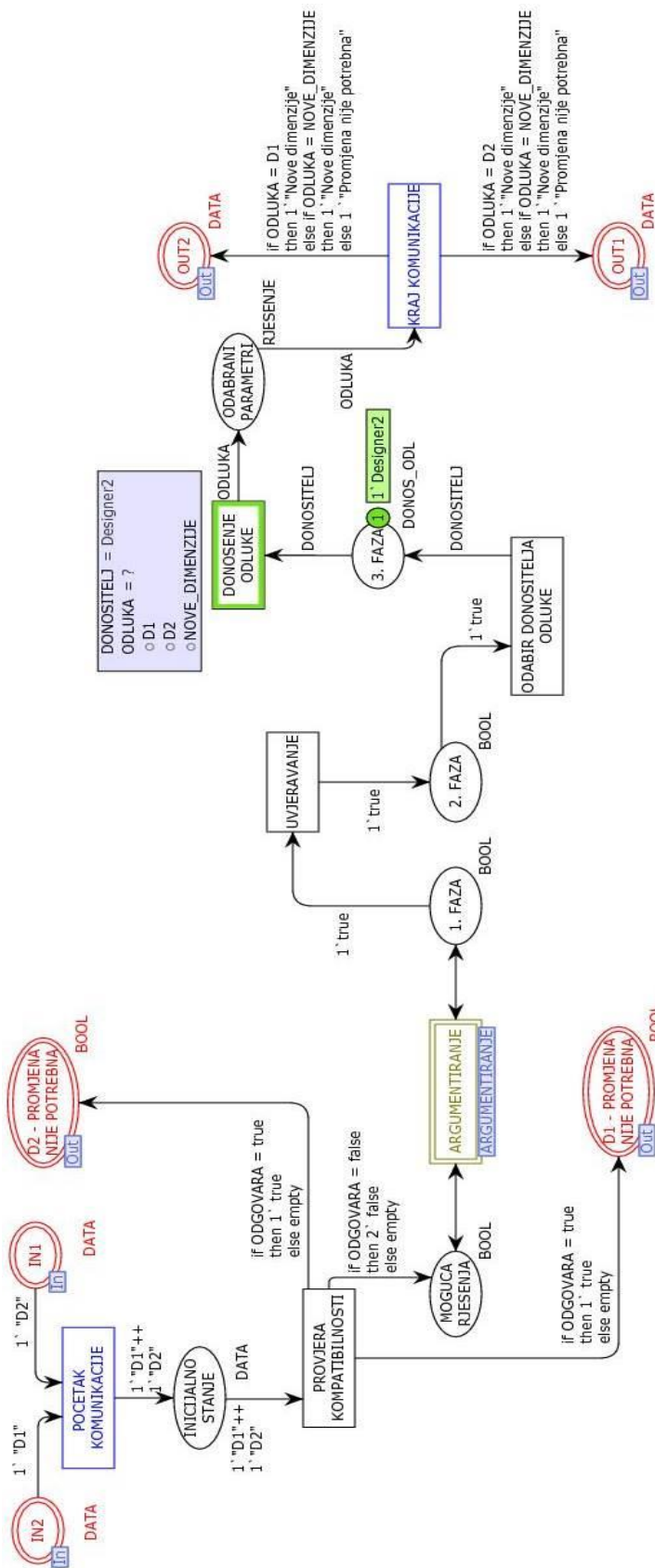
Slika 58: Podmreža procesa pregovaranja – Scara manipulator



Slika 59: Podmreža procesa usklađivanja parametara – Scara manipulator



Slika 61: Pozicija lista prošlih i trenutne dužine S2



Slika 62: Tranzicija donošenje odluke s mogućnošću odabira

7. ZAKLJUČAK

U procesu konstruiranja složenih proizvoda, timski rad je danas uvelike prisutan kako bi se ubrzao proces konstruiranja i poboljšala kvaliteta proizvoda objedinjujući znanje dizajnera specijaliziranih za različita područja. Kompleksnost proizvoda predmet je mnogih istraživanja te su kao podrška procesu razvijeni razni alati za vizualizaciju komponenata proizvoda, veza između komponenata ili drugih objekata u razvoju proizvoda te praćenje sljedivosti informacija o proizvodu. Mnoge studije ukazuju na potrebu praćenja generiranja parametara tijekom procesa konstruiranja, te vizualizaciju promjene njihovih vrijednosti. Cilj istraživanja je razviti alat koji bi podržao praćenje tokova parametara kako bi dizajneri imali uvid odakle dolaze parametri s kojima trenutno rade te tko će preuzeti njihove vrijednosti u konstruiranju drugih modula. Alat bi trebao trenutno ažurirati svaku promjenu vrijednosti parametara i omogućiti pregled utjecaja te promjene na ostale parametre. Osim toga, alat bi trebao pružiti podršku procesu komunikacije te dogovaranju oko vrijednosti spregnutih parametara. U literaturi se kao tipični procesi komunikacije u razvoju proizvoda navode proces pregovaranja, uvjeravanja, donošenja odluke i dr.

U ovom diplomskom radu realiziran je model za određivanje konstrukcijskih parametara u timskom radu pomoću obojenih Petrijevih mreža za dva složena proizvoda koja su konstruirana tijekom studija u timskom radu od dva člana. Pomoću DSM matrice temeljene na parametrima napravljena je analiza procesa, uočene su grupe parametara čije vrijednosti se određuju iterativno te su istaknuti spregnuti parametri. Obojene Petrijeve mreže omogućuju praćenje dinamike procesa konstruiranja, a za razliku od klasičnih Petrijevih mreža, tokeni imaju definirane tipove podataka. Model je kreiran u alatu *CPN Tools* koji omogućava podjelu kompleksne mreže na podmreže radi lakšeg praćenja procesa. *CPN Tools* nudi nekoliko opcija analize procesa kao što je *State Space* analiza koju bi u daljnjem istraživanju trebalo provesti nad modelima. Kao još jednu prednost alata *CPN Tools*, navela bih da je inicijalne vrijednosti parametara moguće učitati u model, npr. iz *XML* datoteke. U obrađenoj literaturi prikazano je da je model kreiran pomoću Petrijeve mreže moguće implementirati u upravljačke jedinice, tzv. kontrolere, koji mogu na temelju modela upravljati procesom. Sljedeći korak u istraživanju mogao bi biti razmatranje programskog rješenja, odnosno implementiranje modela u nekom višem programskom jeziku u obliku aplikacije. Također je potrebno aktivnosti u procesima konstruiranja i komunikacije sistematizirati na apstraktnoj razini te kreirati model koji bi

podržavao upravljanje parametrima u procesu konstruiranja različitih proizvoda te podržao proces komunikacije. Cilj istraživanja je razviti nove pristupe i naprednije alate kojima bi se lakše mogli kontrolirati i izvoditi razvojni procesi. To bi rezultiralo boljom organizacijom aktivnosti u poduzeću, optimiziranim i kvalitetnijim proizvodima.

LITERATURA

- [1] A. Karniel, Y. Reich: Managing the Dynamics of New Product Development Processes A New Product Lifecycle Management Paradigm, School of Mechanical Engineering, Tel Aviv University
- [2] A. Karniel, Y. Reich: Multi-level modelling and simulation of new product development processes – Journal of Engineering Design, 2013. School of Mechanical Engineering, Tel Aviv University, Israel
- [3] T. L. Flanagan, C. M. Eckert, P. J. Clarkson: Parameter trails International Conference on Engineering Design, ICED 03 Stockholm
- [4] Radošević: Petri mreže i njihov odnos prema ostalim mrežnim tehnikama Fakultet organizacije i informatike, Varaždin
- [5] C. A. McMahon, M. Xianyi: A Network Approach to Parametric Design Integration Department of Mechanical Engineering, University of Bristol, UK
- [6] N. Pavković, J. Juranić: Modelling the design parameters dynamics with Petri nets Katedra za razvoj proizvoda, Fakultet strojarstva i brodogradnje Zagreb International Conference on Engineering Design, ICED17
- [7] W. M. P. Van der Aalst: The Application of Petri Nets to Workflow Management Department of Mathematics and Computing Science, Eindhoven University of Technology
- [8] S. Khosravifar: Modeling Multi Agent Communication Activities with Petri Nets International Journal of Information and Education Technology
- [9] E. Kišmartin: Diplomski rad - Analiza podrške komunikaciji u procesu konstruiranja Fakultet strojarstva i brodogradnje, Zagreb
- [10] <https://ocw.mit.edu/courses/engineering-systems-division>
- [11] F. Toepfer, T. Naumann: Parameter Management, a Novel Approach in Systems Engineering – Daimler AG
- [12] F. Toepfer, T. Naumann: Management od Vehicle Architecture Parameters – Damiler AG International Design Conference – Design 2016., Dubrovnik-Croatia
- [13] A. M. Maier, M. Kreimeyer, C. Hepperle, C. M. Eckert, U. Lindemann, P. J. Clarkson: Exploration of Correlations between Factors Influencing Communication in Complex Product Development Engineering Design Centre, University of Cambridge, England Technical University of Munich, Institute of Product Development, Germany
- [14] C. Eckert, M. Stacey: Dimensions of Communication in Design International Conference on Engineering Design, ICED Glasgow
- [15] K. Jensen, L. M. Kristensen, L. Wells: Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems Department of Computer Science, University of Aarhus
- [16] E. Bacarin, W. M. P. van der Aalst, E. Madeira, C. B. Medeiros: Towards Modeling and Simulating a Multi-party Negotiation Protocol with Colored Petri Nets Department of Computer Science Londrina, PR Brazil, Department of Mathematics and Computer Science, Eindhoven University of Technology, Institute of Computing, Brazil
- [17] <http://cpntools.org/>
- [18] D. A. Zaitsev, T. R. Shmeleva: Simulating of Telecommunication Systems with CPN Tools

- Department of Communication Networks, Ministry of Transport and Communications of Ukraine
- [19] T. R. Browning: Process Integration using Design Structure Matrix, 2001.
- [20] D. V. Steward: An Introduction To The PSM32 Program, 1999.
- [21] N. Kuzmić, L. Vlah, D. Rohde, M. Štorga: Samohodni uređaj za čišćenje prirubnice spremnika, Programski zadatak iz kolegija TK-Praktikum
Fakultet strojarstva i brodogradnje, Zagreb
- [22] A. Čamber, D. Marjanović: Design Structure Matrix, Završni rad
Fakultet strojarstva i brodogradnje, Zagreb
- [23] N. Kuzmić, L. Vlah, Z. Domitran: SCARA manipulator, Programski zadatak iz kolegija
Konstrukcijski elementi robota
Fakultet strojarstva i brodogradnje, Zagreb
- [24] https://www.fer.unizg.hr/_download/repository/FPS_predavanja_2%5B1%5D.pdf

PRILOZI

I. CD-R disc