

# Autoreferat

---

## 1 Imię i nazwisko

---

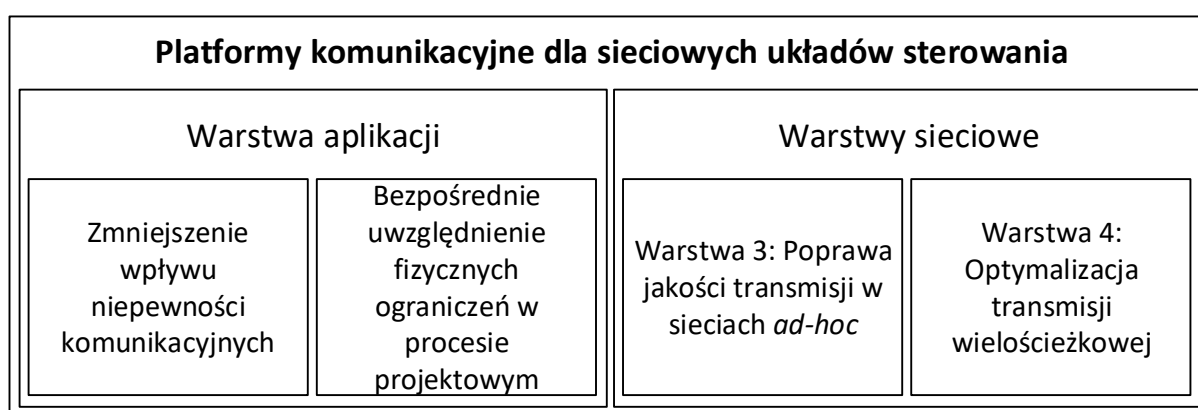
**dr inż. Michał Morawski**

## 2 Tytuł osiągnięcia naukowego

---

### **Odporne platformy komunikacyjne dla współczesnych sieciowych układów sterowania**

Dorobek przedstawiony w ramach osiągnięcia habilitacyjnego koncentruje się na metodach dostarczania wydajnych, a zarazem odpornych platform wymiany informacji dla nowoczesnych, sieciowych układów sterowania (ang. *Network Control Systems* – NCS). Efekty przedstawionych prac nie są powiązane z konkretną klasą układów regulacji ani z wybranym rodzajem sieci. Przeciwnie, mogą być zastosowane w wielu typach NCS już wprowadzonych w przemyśle, jak i tych rozważanych w niedalekiej przyszłości, związanych na przykład z Internetem Rzeczy. Poprawa właściwości omawianej klasy układów upatrywana jest w kilku obszarach zilustrowanych na Rysunku 1.



Rysunek 1. Główne obszary dorobku naukowego.

Omawiane osiągnięcie zrealizowane jest w ściśle ze sobą powiązanych obszarach badań zlokalizowanych zarówno w warstwie aplikacji jak i w warstwach sieciowych.

W warstwie aplikacji:

- Zaproponowano szereg metod ograniczających wpływ niepewności związanych z transmisją danych na tworzenie sieciowych układów sterowania, pozwalając przy tym utrzymać klasyczne zasady projektowania układów regulacji.  
Analiza odporności uzyskanych układów sterowania względem immanentnych w rozważanej klasie zaburzeń komunikacyjnych poparta została formalnymi dowodami matematycznymi.
- W przypadku sieciowych układów sterowania, z uwagi na opóźnienie w pętli sprzężenia zwrotnego, dopuszczenie do nasycenia układu wykonawczego jest szczególnie niebezpieczne i może prowadzić do utraty stabilności.  
W ramach przeprowadzonych badań zaproponowano metodę bezpośredniego włączenia analizy tych ograniczeń w proces projektowania układu zamkniętego, gwarantującą zachowanie stabilności.

W warstwach sieciowych:

- Wskazano metody uzyskania szybszej niż w rozwiązaniach standardowych reakcji sieci na zaburzenia kanałów transmisji danych. Zastosowanie tych rozwiązań prowadzi do zmniejszenia opóźnień w dostarczaniu danych i informacji zwrotnej oraz ograniczenia ich zmienności. Opracowane metody poprawiają zatem jakość działania platformy transportowej dla sieciowych układów sterowania oraz usług multimedialnych. Efekt ten osiągnięto poprzez zastosowanie rozmytych metryk sieci z jednej strony i teorii gier – z drugiej.
- Zbliżony efekt uzyskano również w wyniku zaproponowanego przez habilitanta sposobu optymalizacji wielościeżkowej transmisji danych. Optymalizacji poddana została warstwa transportowa sieci, co umożliwiło równoważenie obciążenia interfejsów terminali sieciowych.

Mój dodatkowy dorobek habilitacyjny związany jest z:

- Równoległym przetwarzaniem informacji, w szczególności z przetwarzaniem systolicznym polegającym na takiej dekompozycji problemu rachunkowego, by dane

opracowane przez jeden komponent układu był źródłem dla innych jego elementów. Celem tych działań jest zwiększenie szybkości obliczeń.

- Analizą czasowo-częstotliwościową wykorzystywaną do diagnostyki wskazującej niesprawne elementy pomiarowe we wdrożonych w praktyce przemysłowych układach sterowania.
- Implementacją układów sterowania, systemów diagnostyki, optymalizacji i wspomagania produkcji w przemyśle, a także uczestnictwem w gronach eksperckich oceniających nowoczesne rozwiązania informatyczne.

### **3 Posiadane dyplomy i stopnie naukowe**

---

1. Magister inżynier. Kierunek: *Elektronika*. Specjalność: *Elektroniczna aparatura medyczna*. Wydział Elektryczny (obecna nazwa: Wydział Elektrotechniki, Elektroniki, Informatyki i Automatyki), Politechnika Łódzka, 1991.
2. Doktor nauk technicznych w dyscyplinie Elektrotechnika. Instytut Elektrotechniki, Warszawa, ul. Pożaryskiego 28, 4 grudnia 1997. Tytuł rozprawy: *Serwomechanizm samonastrajający z zastosowaniem sieci neuronowej*. Praca wyróżniona.

### **4 Informacje o dotychczasowym zatrudnieniu w jednostkach naukowych**

---

1. 1990-1997: asystent w Pracowni Sterowania Optymalnego w Zakładzie Badań Podstawowych PAN w Instytucie Elektrotechniki w Warszawie, ul. Pożaryskiego 28;
2. 1998-2003: adiunkt w Pracowni Sterowania Optymalnego w Zakładzie Badań Podstawowych PAN w Instytucie Elektrotechniki w Warszawie, ul. Pożaryskiego 28;
3. 1998-dzisiaj: adiunkt w Samodzielnym Zakładzie Sieci Komputerowych, po zmianach organizacyjnych – w Instytucie Informatyki, Politechnika Łódzka, Łódź, Wólczańska 215.

## 5 Wskazanie osiągnięcia wynikającego z art. 16 ust. 2 ustawy z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym oraz o stopniach i tytule w zakresie sztuki (Dz. U. 2016 r. poz. 882 ze zm. w Dz. U. z 2016 r. poz. 1311.)

---

Punkty MNiSW podano według wykazu za lata 2013–2016. Wartości wskaźnika *Impact Factor* (IF) podano względem roku opublikowania pracy.

### 5.1 Główne publikacje (czasopisma z listy JCR, z dziedziny informatyka)

---

[MM-A] Michał Morawski, Przemysław Ignaciuk, 2018, *Energy-efficient scheduler for MPTCP data transfer with independent and coupled channels*, *Computer Communications*, 132, 56-64;

**IF = 2.613; MNiSW = 30.**

Mój wkład w powstanie tej publikacji polegał na opracowaniu podstaw teoretycznych i implementacji metody optymalnego, względem zużycia energii, podziału strumienia generowanego przez aplikację użytkownika. Udział procentowy szacuję na 70%.

[MM-B] Przemysław Ignaciuk, Michał Morawski, 2018, *Quasi-soft variable structure control of discrete-time systems with input saturation*, *IEEE Transactions on Control Systems Technology*, PP, 99(2), 1-6;

**IF = 4.883; MNiSW = 40.**

Mój wkład w powstanie tej pracy polegał na implementacji algorytmu w fizycznym układzie, przeprowadzeniu badań eksperymentalnych i opracowaniu wyników. Udział procentowy szacuję na 30%.

[MM-C] Michał Morawski, Przemysław Ignaciuk, 2017, *Network nodes play a game – a routing alternative in multihop ad-hoc networks*, *Computer Networks*, 122, 96-104;

**IF = 2.522; MNiSW = 35.**

Mój wkład w powstanie tej publikacji polegał na opracowaniu, opartej na teorii gier, platformy komunikacyjnej przeznaczonej do sterowania obiektami fizycznymi w sieci typu *ad-hoc*. Udział procentowy szacuję na 60%.

[MM-D] Michał Morawski, Przemysław Ignaciuk, 2016, *Reducing impact of network induced perturbations in remote control systems*, *Control Engineering Practice*, 55(10), 127–138;

**IF = 2.602; MNiSW = 30;**

Mój wkład w powstanie tej publikacji polegał na opracowaniu jednej z przedstawionych w niej metod kompensacji opóźnień oraz na implementacji sieciowych układów sterowania wykorzystujących proponowane algorytmy w rzeczywistym urządzeniu. Udział procentowy szacuję na 50%.

[MM-E] Michał Morawski, Antoni Zajązkowski, 2010, *Approach to the design of robust networked control systems*, International Journal of Applied Mathematics and Computer Science (AMCS), 20(4), 2010, 689-698;

**IF = 0.794; MNiSW = 25.**

Mój wkład w powstanie tej publikacji polegał na opracowaniu metody kompensacji opóźnień i jej implementacji w rzeczywistym urządzeniu oraz metody zarządzania ruchem w sieci wykorzystującej rozmyte metryki. Udział procentowy szacuję na 75%.

## 5.2 Pozostałe publikacje składające się na osiągnięcie habilitacyjne

---

[MM-F] Przemysław Ignaciuk, Michał Morawski, 2018, *Discrete sliding mode control of multipath TCP networks under input and output uncertainty*, 23<sup>rd</sup> International Conference on Methods and Model in Automation and Robotics (MMAR), Międzyzdroje, 363-368;

**MNiSW = 15 (WoS).**

Mój wkład w powstanie tej publikacji polegał na przeprowadzeniu dowodów twierdzeń 1-3, opisujących właściwości zaprojektowanego sieciowego układu regulacji. Udział procentowy szacuję na 50%.

[MM-G] Michał Morawski, Przemysław Ignaciuk, 2016, *Adjustable sampling rate – an efficient way to reduce the impact of network-induced uncertainty in Networked Control Systems?*, 23<sup>rd</sup> Conference Computer Networks (CN), 329-343. Best Paper Award;

**MNiSW = 15 (WoS).**

Mój wkład w powstanie tej publikacji polegał na opracowaniu metody projektowania przełączanego układu regulacji w środowisku sieciowym. Udział procentowy szacuję na 50%.

[MM-H] Michał Morawski, 2006, *Analysis of short latencies in industrial network Environments*, Journal of Applied Computer Sciences (JACS), 2/2006, 65-78;  
**MNiSW = 5.**

[MM-I] Michał Morawski, Andrzej Bartoszewicz, 2005, *Applying routing information to the TCP sender logic*, 2<sup>nd</sup> International Conference on Telecommunication and Computer Networks IADAT-tcn2005, Portsmouth, UK, 115-119;  
**MNiSW = 5.**

Mój wkład w powstanie tej publikacji polegał na opracowaniu metody płynnego uwzględniania w logice TCP informacji pochodzących z protokołów rutowania w celu ograniczenia stopy strat. Udział procentowy szacuję na 50%.

[MM-J] Michał Morawski, 2005, *Uncertain metrics applied to QoS multipath routing*, 5<sup>th</sup> IEEE International Workshop on Design of Reliable Communication Networks, DRCN'05, Island of Ischia, Naples – Italy, 353–360.  
**MNiSW = 15 (WoS).**

### 5.3 Omówienie celu naukowego ww. prac i osiągniętych wyników przedstawionych w postaci powiązanego tematycznie cyklu publikacji wraz z omówieniem ich wykorzystania

---

#### 5.3.1 Wstęp

---

Jeszcze przed ukończeniem studiów zostałem zatrudniony w Pracowni Sterowania Optymalnego w Zakładzie Badań Podstawowych PAN. Rozpoczęciu pracy zawodowej towarzyszył charakterystyczny dla tego okresu gwałtowny rozwój sieci komputerowych. Fascynacja systemami komunikacyjnymi i ich zastosowaniem w sterowaniu obiektami przemysłowymi zaważyła na mojej karierze zawodowej. Dlatego też jednym z podstawowych zadań podejmowanych w tym czasie było projektowanie i implementacja rozproszonych układów sterowania. Brałem wówczas udział w szeregu wdrożeń realizowanych we współpracy z przemysłem. Najważniejsze z nich to:

- Systemy wspomagania pracy dyspozytorów w kopalni węgla brunatnego w Bełchatowie. Ich celem było zbieranie danych z licznych czujników zainstalowanych w urządzeniach wydobywczych (koparkach, zwałowarkach, taśmociągach, podstacjach energetycznych itp.), przetwarzanie pozyskanych informacji dla celów sterowania i wspomaganie pracy dyspozytorów.

Komunikacja pomiędzy elementami składowymi odbywała się za pośrednictwem protokołów MAP (*Manufacturing Automation Protocols*).

- Elektroniczne systemy sterowania krosnami żakardowymi zastępujące wcześniej stosowane programatory mechaniczne. Systemy te uruchomiono w kilku fabrykach w Łodzi, a także w Prudniku. Prócz implementacji samego układu sterowania, rozwiązania te zawierały oprogramowanie wspomagające proces projektowania produkowanych tkanin, a także podsystemy komunikacji z maszynami.
- System kontroli naziemnych urządzeń nawigacyjnych *Control Flight Inspection System* dla przedsiębiorstwa Polskie Porty Lotnicze (we współpracy z firmą CAS). W skład tego rozwiązania wchodziło oprogramowanie wraz z zestawem specjalizowanych urządzeń nawigacyjnych. Komunikację zapewniały typowe tylko dla układów aeronautyki sieci komputerowe, na przykład ARINC 629. Omawiany system został zainstalowany w samolocie weryfikującym poprawność działania wszystkich urządzeń nawigacji lotniczej w Polsce.

Oprócz układów sterowania opartych na klasycznych metodach pomiarowych powstawały także rozwiązania wykorzystujące kamery wizyjne do uzyskania informacji zwrotnej o stanie obiektu.

- Wyniki prac teoretycznych i praktycznych związanych z tą aktywnością ujęte zostały w mojej pracy doktorskiej, zatytułowanej *Serwomechanizm samonastrajający się z zastosowaniem sieci neuronowej*, obronionej w Instytucie Elektrotechniki w Warszawie. Praca została wyróżniona.

W przedstawionym rozwiązaniu zbudowano analizujący obraz system komputerowy, który określał we współrzędnych związanych z pojedynczą kamerą położenie trójwymiarowych obiektów na scenie w 6 stopniach swobody. Zaproponowane rozwiązanie w sposób bezpośredni uwzględniało geometryczne i inne zaburzenia obrazu, na przykład przesłonięcia fragmentów sceny, cienie itp. Rozważane były też obrazy cyfrowe pozyskane z układu dwóch kamer.

Moja dalsza aktywność badawczo-naukowa związana była z projektowaniem rozproszonych systemów przetwarzania danych. W szczególności dotyczyło to:

- Przetwarzania systolicznego polegającego na takiej dekompozycji problemu obliczeniowego, by był on realizowalny równoległe przez układ nieskomplikowanych elementów połączonych ze sobą w specjalną strukturę. Struktura ta jest bezpośrednio związana z projektowanym algorytmem. W zaproponowanym podejściu elementy przetwarzające (ale także system dostarczania danych i odbierania wyników obliczeń) zorganizowane są tak, by wynik przetwarzania jednych z nich był źródłem danych dla kolejnych. W efekcie uzyskuje się przyspieszenie prowadzenia obliczeń. Ze względu na sposób działania algorytmy systoliczne nazywane są też algorytmami czasowo-przestrzennymi, albo potokowo-równoległymi. Omawiane prace realizowano ze wsparciem Komitetu Badań Naukowych w postaci grantu nr 8T10A 046 12, pt. *Numeryczno-analityczne metody przetwarzania równoległego w problemach sterowania układami elektromechanicznymi*, w którym byłem głównym wykonawcą.

Ta gałąź aktywności zaowocowała cyklem 7 artykułów i monografią: Ewa Lipowska-Nadolska, Michał Morawski, 1999, *Tablice systoliczne. Problemy wybrane*, PLJ.

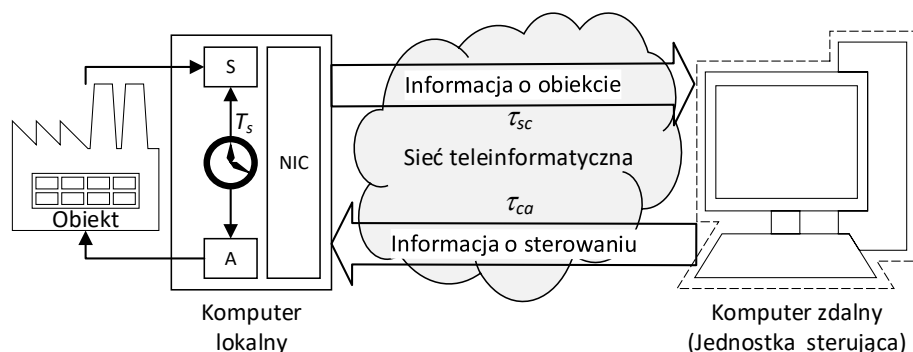
- Oprogramowania dla systemów diagnostycznych, których zadaniem było wykrywanie anomalii w pracy podsystemów składowych. Funkcjonowanie stworzonych programów opierało się na analizie zmienności transformat czasowo-częstotliwościowych sygnałów pomiarowych, w obecności zakłóceń i opóźnień, nieuniknionych w rozważanej klasie aplikacji. Ta część aktywności zaowocowała szeregiem publikacji i wdrożeniem przykładowego systemu w Elektrowni Bełchatów. O jakości, użyteczności i niezawodności wykonanych prac świadczy m. in. fakt, że systemy te, wdrażane w latach 2001-2003, działają do dziś. Zarówno przedmiotem i sposobem działania systemy te wpisują się we współczesną tematykę określaną paradygmatem *Industry 4.0*.

### 5.3.2 Główne osiągnięcie

---

Przez kolejne lata działalności zawodowej i akademickiej zajmowałem się projektowaniem i wdrażaniem innowacyjnych rozwiązań informatycznych w *sieciowych* układach sterowania. Specyfiką tej klasy układów regulacji automatycznej jest występowanie sieci komputerowej w pętli sprzężenia zwrotnego, jak pokazano na Rysunku 2. Sieć zajmuje się zarówno transportem danych pomiarowych wielkości





Rysunek 2. Schemat typowego sieciowego układu sterowania. A – podsystem wykonawczy, S – podsystem pomiarowy, NIC – interfejs komunikacyjny,  $T_s$  – okres impulsowania,  $\tau_{ca}$  – opóźnienie występujące pomiędzy jednostką sterującą a podsystemem wykonawczym,  $\tau_{sc}$  – opóźnienie występujące pomiędzy podsystemem pomiarowym a jednostką sterującą.

wyjściowych obiektu, jak i wartości sterujących dla celów przeprowadzenia procesu regulacji. Takie rozwiązanie pozwala przenieść logikę algorytmu sterowania do wysokowydajnego komputera, a w konsekwencji wykonalne jest zastosowanie wyrafinowanych metod teorii regulacji. Pojawia się także możliwość jednoczesnego nadzoru pracy wielu układów oraz synchronizacji ich działań [1–4]. Kolejną zaletą, wynikającą na przykład z zastosowania zapasowych kanałów i regulatorów, jest usprawnienie reakcji na ewentualną awarię. Ponadto, przy odpowiedniej konstrukcji sieciowych układów sterowania można zrezygnować z kosztownych systemów czasu rzeczywistego na rzecz znacznie tańszych rozwiązań ogólnodostępnych, niespełniających restrykcyjnych wymagań czasowych.

Niestety, zastosowanie sieci komputerowej w pętli sprzężenia zwrotnego ma też i swoje wady. Największą z nich jest występowanie opóźnień (Rysunek 2) między elementem pomiarowym i sterownikiem ( $\tau_{sc}$ ) oraz między sterownikiem a elementem wykonawczym ( $\tau_{ca}$ ), których nie można uniknąć w strukturze rozproszonej. Opóźnienia te powodują przesunięcie w czasie momentu pomiaru i poprzedzoną stosownymi obliczeniami zmianę wartości sterującej przyłożonej do (zdalnego) obiektu. Omawiane opóźnienie jest równe co najmniej jednemu okresowi impulsowania, jednak zwykle jest dużo większe, a co gorsza – zmieniające się w trakcie realizacji procesu regulacji w losowy lub trudny do przewidzenia sposób.

Zastosowanie znanych mechanizmów kompensacji opóźnień (na przykład predyktora Smitha) [5, 6] stanowi istotne wyzwanie w przypadku sieciowych układów sterowania. Powodem jest zarówno wspomniana zmienność opóźnień (pomiaru opisane w [MM-H] i

[MM-E] wskazują na wahania stanowiące nawet trzydziestokrotność wielkości nominalnej), jak i możliwość zagubienia niektórych danych pomiarowych lub poleceń skierowanych do układu wykonawczego, na przykład wskutek zakłócenia lub przeciążenia sieci. Tego rodzaju straty pojawiają się w przypadkowych chwilach i wynikają z działania sieci jako niepewnego medium transmisji danych. Zdarzają się także serie strat. W rezultacie może dojść do utraty stabilności układu, który pracowałby poprawnie w implementowanej lokalnie pętli sprzężenia zwrotnego.

Opisaną niedogodność można pominąć wyłącznie w przypadku obiektów o niewielkiej dynamice. Można też łagodzić jej wpływ stosując kosztowe rozwiązania techniczne, tj. specjalizowane sieci komputerowe, systemy operacyjne i aplikacje dostosowane do systemów czasu rzeczywistego. Ze względu na to, że przepustowość sieci jest zawsze ograniczona, utrudnione jest także zmniejszanie okresu impulsowania układu. Próby polepszania dynamiki układu zamkniętego tą metodą są często nieskuteczne, a nawet mogą obniżać jakość procesu regulacji [7]. Dzieje się tak dlatego, że w przypadku nadmiernego obciążenia sieci zwiększają się nierównomierności opóźnień, a nawet możliwe jest powstawanie strat występujących po sobie w seriach [8, 9] (zjawisko załamania sieci). Wówczas zamknięta pętla sterowania zostaje przerwana, a odzyskanie stabilności systemu po przywróceniu łączności może być już niemożliwe.

Szereg problemów związanych z komunikacją w sieciowych układach sterowania można wyeliminować wykorzystując konstrukcję układu bazującego na zarządzaniu zdarzeniami (ang. *Event Driven* – ED), zamiast klasycznego opartego o okresy impulsowania (ang. *Time Driven* – TD) [4]. Podejście takie wydaje się atrakcyjne, gdyż coraz szybsze i tańsze komputery powodują, że zwiększenie kosztów realizacji algorytmu sterowania [10] staje się coraz mniej istotne z praktycznego punktu widzenia. Niestety, wspomniane zyski są realne tylko przy założeniu, że informacje przesyłane przez sieci komputerowe nie giną, a opóźnienie transmisji jest znane. W praktycznych implementacjach sieciowych układów sterowania przedstawione założenia trudno spełnić. W jednym z projektów komercyjnych realizowanych przy moim współudziale, podczas rozruchu silników pomp dużej mocy, stopa strat występująca w sieci komunikacyjnej (niewłaściwe sumy kontrolne pakietów otrzymane wskutek zakłóceń elektromagnetycznych) sięgała aż 70%.

Ponieważ bardzo często warunkiem przeprowadzenia projektów jest niska cena ich wdrożenia, koniecznością staje się używanie sieci standardowych, dla których nawet w warunkach biurowych, zarówno odsetek utraconych pakietów, jak i zmienność opóźnień jest niepomijalna.

W sieciowych układach sterowania opartych o czas (TD) do łagodzenia wpływu opisanych powyżej zjawisk można podejść od strony:

**Aplikacyjnej**, polegającej na wprowadzeniu do wynikowego prawa sterowania mechanizmów zmniejszających wpływ opóźnień lub strat bez istotnego zwiększania nakładu obliczeniowego i wolumenu przepływu danych;

**Sieciowej**, polegającej na zmniejszeniu niepewności związanej z działaniem systemów łączności.

W rzeczywistości każde z tych podejść ma ograniczony zakres stosowalności. Nie sposób skompensować w warstwie aplikacji zbyt dużych wahań opóźnień, ponieważ z każdym fizycznym elementem wykonawczym związane są ograniczenia (na przykład maksymalny moment silnika). Także nadmierne zmniejszanie opóźnień i ich zmienności w sieciach teleinformatycznych jest kosztowne. Natomiast równoczesne stosowanie aplikacyjnych i komunikacyjnych metod ograniczania wpływu opisanych zjawisk pozwala na osiągnięcie wymiernej poprawy jakości działania układu regulacji przy stosunkowo niewielkich nakładach. Jako dodatkową korzyść należy wskazać poprawę odporności układu regulacji na występowanie awarii (na przykład implementację systemów zapasowych). Znaczenie tego ostatniego aspektu jest nie do przecenienia w przypadku wdrożenia.

Opisane poniżej prace związane z poprawą jakości działania platform komunikacyjnych w sieciowych układach sterowania wspierane były grantami, w których byłem głównym wykonawcą lub wykonawcą pomocniczym:

- 3T11C 043 26, (2004-2006), zatytułowany *Modele i algorytmy wielościeżkowego rutowania w sieciach komputerowych z gwarancją jakości usług*. Grant Komitetu Badań Naukowych realizowany w Zakładzie Sieci Komputerowych Politechniki Łódzkiej.
- 3T10A 037 28, (2005-2007) zatytułowany *Sterowanie wybranymi układami elektromechanicznymi przy wykorzystaniu sieci komputerowych*.

Grant Komitetu Badań Naukowych realizowany w Instytucie Automatyki Politechniki Łódzkiej.

- DEC-2012/05/D/ST6/03030, (2013-2016), zatytułowany *Projektowanie i weryfikacja algorytmów sterowania w sieciowych układach regulacji*.

Grant Narodowego Centrum Nauki realizowany w Instytucie Informatyki Politechniki Łódzkiej.

- 0156/IP2/2015/73, (2015–2017), zatytułowany *Optymalne a odporne algorytmy aktywnego zarządzania zasobami w układach sieciowych – projektowanie i analiza właściwości*.

Grant Ministerstwa Nauki i Szkolnictwa Wyższego realizowany w Instytucie Informatyki Politechniki Łódzkiej.

### **5.3.2.1 Aplikacyjne podejście do poprawy jakości procesu regulacji w sieciowych układach sterowania – ograniczanie wpływu niepewności**

---

Z natury rzeczy jakość działania systemu jest taka, jaka jest jakość działania najsłabszego elementu składowego. W przypadku analizowanej klasy układów takim elementem jest zwykle sieć teleinformatyczna wykorzystywana do komunikacji między podsystemami. W cyklu prac stanowiącym omawiane osiągnięcie naukowe, pojęcie sieć komputerowa dotyczy z założenia takich sieci, w których zapewnienie stałego i niewielkiego opóźnienia jest nierealizowalne w dłuższym okresie (niededykowana sieć ogólnego przeznaczenia). W związku z tym, koniecznością staje się rozważanie działania aplikacji w warunkach nieuniknionych niepewności towarzyszących przekazywaniu danych na odległość, tak jak to zasygnalizowano na Rysunku 2.

Analizując metody obliczania wartości sterujących w sieciowych układach regulacji w literaturze przedmiotu często rozważa się występowanie niepewności wyłącznie w kanale SC, rzadziej w kanale CA (Rysunek 2). W rzeczywistych rozwiązaniach takie założenia są nierealne. W cyklu prac [MM-D], [MM-E], [MM-G] założono, że niedostatki jakości kanałów komunikacyjnych (losowe opóźnienia, straty) mogą występować w dowolnym miejscu toru układu regulacji ze sprzężeniem zwrotnym. Dzięki temu możliwa jest implementacja opracowanych metod w układach fizycznych, a zatem i praktyczna (eksperymentalna) weryfikacja poprawności ich działania.

Choć procesy fizyczne modeluje się zwykle jako układy z czasem ciągłym, to postać ta jest niemożliwa do zastosowania w sieciowych układach sterowania. Założono tutaj, że model z czasem ciągłym poddany jest linearyzacji w punkcie pracy oraz, że jest to układ stacjonarny. Zakładając ponadto tryb pracy TD (tj. oparty o stałe okresy impulsowania) w wyniku odpowiednich przekształceń [11–13] sprowadza się analizowany model do postaci dyskretnej

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k) \end{aligned}, \quad (1)$$

gdzie  $\mathbf{x}(k)$  jest wektorem stanu układu,  $\mathbf{u}(k)$  – wektorem wejścia,  $\mathbf{y}(k)$  – wektorem wyjścia, zaś macierze  $\mathbf{A}$  i  $\mathbf{B}$  odpowiednich wymiarów opisują dynamikę układu i wpływ wartości wejściowych.  $k = 0, 1, 2, \dots$  określa ciąg dyskretnych chwil i jest skrótowym zapisem wyrażenia  $kT_s$ , w którym  $T_s$  jest okresem impulsowania.

W proponowanym podejściu zakłada się, że dla równania (1) wektor wejściowy  $\mathbf{u}(k)$  uzyskuje się na podstawie stanu układu jako

$$\mathbf{u}(k) = -\mathbf{K}\mathbf{x}(k), \quad (2)$$

gdzie macierz  $\mathbf{K}$  jest określana jedną z metod dostępnych w literaturze [11–13]. Niestety warunkiem zastosowania tego nieskomplikowanego schematu jest możliwość pomiaru wektora stanu  $\mathbf{x}(k)$  i dostarczenie wcześniej obliczonej wartości sterującej  $\mathbf{u}(k)$  w tej samej chwili czasu. W sieciowym układzie sterowania jest to jednak nierealizowalne. Zatem równanie (1) należy rozważyć w postaci

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k-h(k)) + \mathbf{d}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k) \end{aligned}, \quad (3)$$

gdzie  $\mathbf{d}(k)$  odzwierciedla nieznaną wektor zakłóceń, także błędów pomiarowych, zaś  $h(k) \geq 1$  – funkcję opóźnienia dostarczania informacji.  $h(k)$  jest dyskretnym odpowiednikiem sumy czasów  $\tau_{ca}$  i  $\tau_{sc}$ , tj.

$$h(k) = \lceil (\tau_{sc}(k) + \tau_{ca}(k)) / T_s \rceil. \quad (4)$$

Często wartość  $h(k)$  modeluje się statystycznie na przykład rozkładami Poissona, wykładniczym lub Pareto [9, 14, 15]. Badania przedstawione w [MM-E] i [MM-H] wskazują jednak wątpliwą skuteczność takiego podejścia dla krótkich odcinków czasu, rzędu co najwyżej kilkudziesięciu milisekund. We współczesnych sieciach

komputerowych minimalne możliwe do praktycznego, trwałego (tj. powtarzającego się co najmniej w kilkudziesięciu kolejnych cyklach) osiągnięcia opóźnienie to około

- 1 ms – dla sieci Ethernet,
- 2-4 ms – dla sieci bezprzewodowych zbudowanych w oparciu o protokoły standardu 802.11,
- kilkanaście ms – w sieciach opartych o rozwiązania klasy 802.15. Sieci tego typu desygnowane są właśnie do celów gromadzenia pomiarów i sterowania,
- co najmniej kilkadziesiąt ms – w sieciach komórkowych.

Maksymalne dopuszczalne czasy opóźnień zależą od własności obiektu regulacji. Opisywane algorytmy badane były w zastosowaniu między innymi do obiektów strukturalnie niestabilnych (odwrócone wahadło i układ lewitacji magnetycznej), gdzie brak możliwości dostarczenia wartości sterującej przez czas od kilku do kilkudziesięciu milisekund (w zależności od konkretnej realizacji), powodował nieodwracalną utratę stabilności. Nierzadko sieci komputerowe nie są w stanie podołać takim wymaganiom w sposób trwały. Powodem wydłużenia czasu dostarczenia danych w sieci komputerowej może być:

- występowanie zatorów, nawet krótkotrwałych i niezauważalnych dla typowych aplikacji biurowych,
- działanie procedur dostępu do medium,
- zakłócenia,
- inne transmisje we współdzielonych kanałach komunikacyjnych,
- procedury związane z bezpieczeństwem komunikacji.

W przypadku układu opisanego równaniem (3) można projektować dedykowany układ regulacji. Jednak metody bezpośredniego uwzględniania opóźnień w konstrukcji prawa sterowania są uciążliwe w implementacji [16]. Dlatego też często stosowaną metodą [17] realizacji tego zadania jest zamiana opóźnień na zmienne stanu w taki sposób, że (3) przekształca się w

$$\begin{aligned} \mathbf{x}_e(k+1) &= \mathbf{A}_e \mathbf{x}_e(k) + \mathbf{B}_e \mathbf{u}(k) + \mathbf{d}_e(k) \\ \mathbf{y}(k) &= \mathbf{C}_e \mathbf{x}_e(k) \end{aligned} \quad (5)$$

gdzie macierze powiązane są ze sobą w następujący sposób:

$$\mathbf{x}_e(k) = \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{u}(k-h) \\ \vdots \\ \mathbf{u}(k-2) \\ \mathbf{u}(k-1) \end{bmatrix}, \mathbf{A}_e = \begin{bmatrix} \mathbf{A} & \mathbf{B} & 0 & \dots & 0 \\ 0 & 0 & \mathbf{I} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \mathbf{I} \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}, \mathbf{B}_e = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{I} \end{bmatrix}, \mathbf{d}_e = \begin{bmatrix} \mathbf{d} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \mathbf{C}_e = \begin{bmatrix} \mathbf{C}^T \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \quad (6)$$

przy czym  $\mathbf{I}$  jest macierzą jednostkową wymiaru takiego, jaka jest liczba wejść układu, zaś  $h$  opóźnieniem, które należy uwzględnić przy projektowaniu regulatora oraz analizy właściwości układu zamkniętego. Ponieważ opóźnienie w sieciowym układzie sterowania zwykle nie jest z góry znane, odpowiednie parametry regulatorów należy dobrać oddzielnie dla opóźnień w całym zakresie pracy układu.

W przypadku równań wysokiego rzędu powstających w wyniku przekształcenia (5) przeprowadzenie formalnej procedury projektowej nasyca trudności analitycznych [MM-D]. Z tego względu poszukuje się innych metod korygowania problemu nieznanymi *a priori* opóźnień. Można na przykład zastosować kompensator opóźnień. Wówczas prawo sterowania jest takie samo, jak w przypadku układu bez opóźnienia (2). Zatem do jego syntezy można wykorzystać metody klasyczne [11–13].

W pracach [MM-D], [MM-E], [MM-G] zaproponowano kilka podejść do projektowania takich kompensatorów:

1. Podstawienie (2) do (1) dla kolejnych chwil  $k$ , ponownie wykorzystanie (2), znów podstawienie do (1), aż do osiągnięcia maksymalnej wartości opóźnienia, czyli

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k), \\ \mathbf{u}(k+1) &= -\mathbf{K}\mathbf{x}(k+1), \\ \mathbf{x}(k+2) &= \mathbf{A}\mathbf{x}(k+1) + \mathbf{B}\mathbf{u}(k+1), \\ \mathbf{u}(k+2) &= -\mathbf{K}\mathbf{x}(k+2), \\ &\dots \end{aligned} \quad (7)$$

2. Wykorzystując wielkość pomocniczą zbudowaną na bazie wcześniejszych wartości sterujących

$$\mathbf{v}(k) = \mathbf{A}^h \mathbf{x}(k) + \sum_{i=1}^h \mathbf{A}^{i-1} \mathbf{B} \mathbf{u}(k-i), \quad (8)$$

gdzie  $h$  określa długość opóźnienia, dla którego projektowany jest regulator. Wówczas prawo sterowania będzie miało postać

$$\mathbf{u}(k) = -\mathbf{K}\mathbf{v}(k), \quad (9)$$

przy czym macierz  $\mathbf{K}$  jest identyczna jak w przypadku układu bez opóźnień, zaś (9) bezpośrednio zastępuje (2).

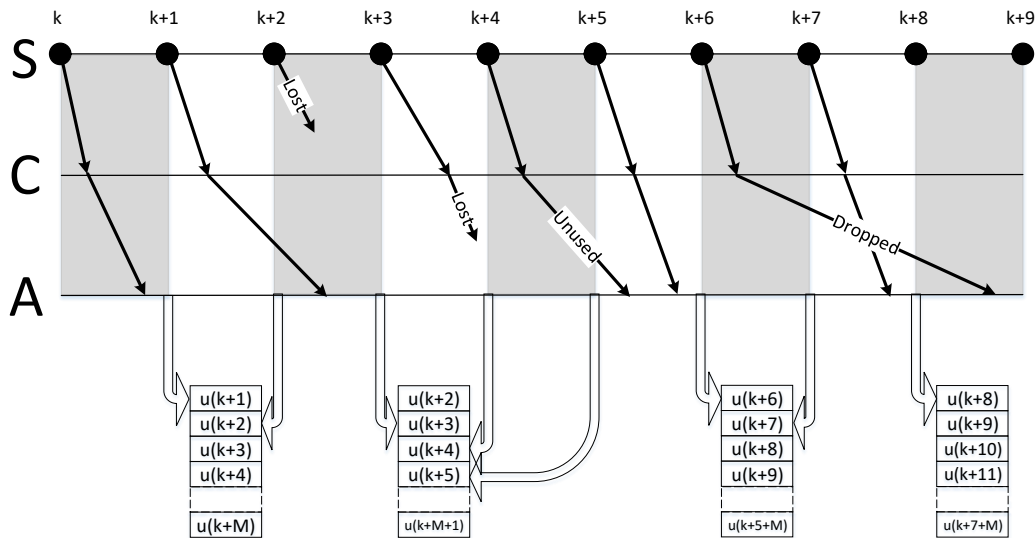
W obu przypadkach kolejne elementy wektora  $\mathbf{u}(k)[i]$  mogą być wyznaczone rekurencyjnie, na podstawie wcześniej obliczonej wartości  $\mathbf{u}(k)[i - 1]$ , co jest istotne ze względu na efektywność obliczeniową.

Mierząc średnie opóźnienie w układzie –  $h(k) = h = \lceil \text{av}(\tau_{sc} + \tau_{ca}) / T_s \rceil$  – na podstawie znaczników czasu – tzw. stempli (ang. *time stamps*) generowanych przez podsystem pomiarowy i przekazywanych w pakietach z danymi, komputer sterujący może wyznaczyć wartości wykorzystywane następnie przez elementy wykonawcze. Tak zbudowany układ będzie jednak działał poprawnie, jeśli rzeczywista wariancja  $h(k)$  będzie bliska zeru. W przeciwnym przypadku (a także, gdy filtr obliczający średnie opóźnienie  $\text{av}(\cdot)$  trudno jest odpowiednio dobrać) należy stosować inne podejście.

Zamiast określać wartość średniego opóźnienia –  $\text{av}(\cdot)$  – można jednocześnie (w jednym pakiecie) przysyłać sekwencję wartości pomiarowych oraz innych danych potrzebnych do obliczeń. Komputer sterujący może wówczas obliczyć kilka alternatywnych wielkości wejściowych układu i przesłać je w tym samym pakiecie do układu wykonawczego. Podsystem wykonawczy, znając aktualną wielkość opóźnienia w pętli sprzężenia zwrotnego, używa odpowiedniej wartości do indeksowania tablicy propozycji (Rysunek 3).

W porównaniu z poprzednią metodą, zaletą omawianego podejścia jest uodpornienie układu na szybkie fluktuacje opóźnień i straty ([MM-H]), zaś wadą zwiększenie obciążenia sieci. Analiza typowych protokołów sieciowych, a w szczególności wielkości różnego rodzaju nagłóweków i procedur dostępu do medium pokazuje jednak, iż w przypadku wielu rzeczywistych technologii sieciowych zwiększenie wielkości komunikatów nie powoduje istotnych zmian obciążenia sieci. Trzeba jednak pamiętać, że nie jest to bezwzględna reguła i nie zawsze koszt przekazania dodatkowych informacji jest akceptowalny. W pracy [MM-D] dokonano oszacowania wymaganej wydajności transmisji w obu rozpatrywanych przypadkach.





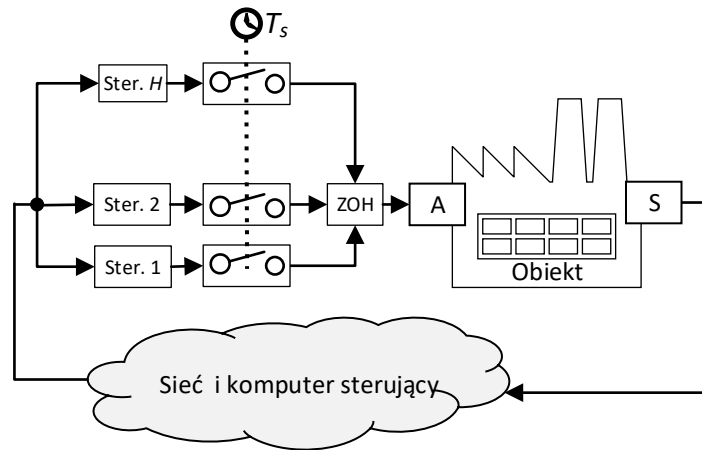
Rysunek 3. Schemat systemu wyboru odpowiedniej wartości sygnału sterującego ze zbioru propozycji. S – podsystem pomiarowy, C – podsystem sterujący, A – układ wykonawczy,  $k, k+1, k+2, \dots$ , – kolejne chwile czasu w układzie ze stałym okresem impulsowania  $T_s$ .

Zaproponowałem też inną, nowatorską metodę wyznaczania wielkości sterujących. Może być ona zastosowana zarówno dla kompensatorów jednowartościowych (4) jak i wielowartościowych (Rysunek 3). Chociaż jest ona bardzo podobna w implementacji układu wykonawczego do metod opisanych wcześniej, to z matematycznego punktu widzenia jest to zupełnie inne rozwiązanie. Zastosowano w nim układ z przełączaną strukturą [18] przedstawiony na Rysunku 4. Zamiast kompensatora opóźnień zaproponowano użycie kilku schematów, w których układ opisany tym samym równaniem stanu dla czasu ciągłego jest dyskretyzowany dla różnych opóźnień. Innymi słowy dla każdego opóźnienia  $i \in [1, H]$ , gdzie  $H$  jest maksymalnym opóźnieniem w danym układzie regulacji, wyznacza się macierze

$$\begin{aligned} \mathbf{A}_i &= e^{\mathbf{A}_c i T_s}, \\ \mathbf{B}_i &= \mathbf{B}_c \int_0^{i T_s} e^{\mathbf{B}_c (i T_s - \lambda)} d\lambda, \end{aligned} \quad (10)$$

gdzie indeks dolny „c” oznacza wartości określone dla czasu ciągłego wynikające wprost z modelu obiektu fizycznego. Macierze  $\mathbf{K}_i$  (2) zostały wyznaczone w taki sam sposób [11–13] dla każdego podsystemu  $i$ .

W badaniach zaprezentowanych w cyklu prac [MM-D], [MM-E], [MM-G] zbadano odpowiedzi rzeczywistych obiektów sterowanych za pomocą metod opisanych



Rysunek 4. Schemat systemu z przełączaną strukturą (ang. *switched system*). W zależności od wielkości opóźnienia układ wykonawczy wybiera odpowiednie prawo sterowania. A – podsystem wykonawczy, S – podsystem pomiarowy, ZOH (ang. *Zero-Order Hold*) – układ podtrzymujący sygnał wejściowy.

równaniami (7), (9), (2) + (10) – Rysunek 4 – przy użyciu kompensatorów jednowartościowych (4), jak i wielowartościowych (Rysunek 3) na różne scenariusze problemów indukowanych przez sieć (zmiennosc opóźnień, serie strat itp.). W pracach tych wykazano, że układ przełączany opisany równaniami (2) i (10) – Rysunek 4 – jest mniej wrażliwy od układów z kompensatorami na wielkość zakłócenia (także błędy pomiarowe), za to bardziej na błędy w identyfikacji parametrów obiektu (dokładność wyznaczenia macierzy  $A_c$  i  $B_c$ , obecność zjawisk nieuwzględnionych przy tworzeniu modelu, na przykład luzów, sprężystości cięgien itp.). Natomiast nakład obliczeniowy konieczny do ich zastosowania jest w obu przypadkach zbliżony. Warto też zauważyć, że metoda (7) nie ogranicza się do układów liniowych.

Ponadto w pracy [MM-E] zaprezentowałem działanie systemu redundancji typu *active-active* demonstrującego zalety wprowadzenia sieciowego układu sterowania. W rozwiązaniu tym główny układ sterujący został nagle wyłączony, a jego zadanie przejął w czasie rzeczywistym układ zapasowy. Dzięki odpowiedniej konstrukcji sieci i regulatorów [MM-E], nadzorowany obiekt (układ lewitacji magnetycznej) nie utracił stabilności, a czas reakcji (zastąpienia) wyniósł zaledwie kilka milisekund. Aby była możliwa tak sprawna reakcja na zakłócenia transmisji danych, konieczne jest stosowanie regulatorów bez pamięci lub z pamięcią bardzo krótką. Jest tak dlatego, że informacja w głównej i zapasowej jednostce sterującej może się różnić, potęgując w ten sposób niestabilność układu sterowania. Wszystkie rozwiązania zaproponowane powyżej spełniają takie wymagania.

### 5.3.2.2 *Aplikacyjne podejście do poprawiania własności sieciowych układów sterowania – ograniczanie wielkości sterujących*

---

Opracowując prawa sterowania dąży się zwykle do szybkiego sprowadzenia błędu regulacji do założonego minimum. Wiąże się to jednak z potrzebą zastosowania znacznych wielkości sterujących – często przekraczających fizyczne możliwości układu wykonawczego. Nierzadko zdarza się tak, że twórcy układu regulacji ignorują ograniczenie na wejściu. Niestety, układ z nasyceniem (pojawiającym się wskutek przekroczenia parametrów technicznych układu wykonawczego) nie jest już układem, dla którego zaprojektowane zostało prawo sterowania. Może to prowadzić do utraty stabilności [4]. Szczególnie narażone na ten efekt są układy z opóźnieniami, zatem także sieciowe układy sterowania.

Z drugiej strony, wskutek zaprojektowania takiego prawa sterowania, które gwarantuje pozostawanie w liniowym zakresie pracy układu wykonawczego, uzyskuje się układy o niskiej dynamice, wolno reagujące na zmiany warunków pracy.

Istnieją też praktyczne argumenty wskazujące na potrzebę tworzenia układów z łagodnymi przejściami wielkości sterujących. W przypadku gwałtownych zmian tych wartości urządzenia elektromechaniczne, w wyniku większych obciążeń, zużywają się szybciej i potrzebują do realizacji żądanych działań więcej energii. Ponadto modele matematyczne obiektów fizycznych zwykle pomijają szereg zjawisk, w tym przede wszystkim nieliniowych, takich jak sprężystości, drgania, poślizgi, czy luzy mechaniczne, ze względu na ich niewielki wpływ na zachowanie układu jako całości. Gwałtowne zmiany wartości sterujących powodują, że wpływ ten staje się znaczący.

Analizując problem dokładniej, spostrzec można, że w sytuacji niewielkich odchyłeń układu od stanu równowagi można stosować stosunkowo duże wartości sterujące. Jest tak dlatego, że w wyniku pomnożenia macierzy sprzężenia zwrotnego  $\mathbf{K}$  przez wektor stanu  $\mathbf{x}(k)$  wartości maksymalne (co do modułu) wektora sterującego  $\mathbf{u}(k)$  nie przekroczą fizycznych możliwości układu (2). W miarę zwiększania się błędu sterowania, wielkości znajdujące się w macierzy  $\mathbf{K}$  powinny się zmniejszać. Powstaje jednak pytanie, jak powinny się zmieniać wartości  $\mathbf{K}(k)$  w takim nieliniowym układzie sterowania, aby zagwarantować stabilność układu zamkniętego.

W pracy [MM-B] zaproponowałem podejście polegające na zastosowaniu układu o zmiennej strukturze (ang. *Variable Structure Control – VSC*) podobnego w swojej idei do pokazanego na Rysunku 4, jednak przełączanego **nie** w sposób skokowy na podstawie zmiennej związanej z czasem, tak jak to było zaproponowane w pracy [MM-D], ale płynnie, na podstawie odchylenia wektora stanu od punktu równowagi. W [MM-B] zastosowano prawo sterowania

$$\mathbf{u}(k) = -\mathbf{K}(k)\mathbf{x}(k) = -[\mathbf{K}_1 + g(k)\mathbf{K}_2]\mathbf{x}(k), \quad (11)$$

w którym macierze sprzężeń zwrotnych  $\mathbf{K}_1$  i  $\mathbf{K}_2$  wyznaczone są przy różnych założeniach w stosunku do dynamiki zamkniętego układu liniowego. W pracy [MM-B] wyznaczono i formalnie udowodniono w jaki sposób powinna być modyfikowana wartość zmiennej  $g(k)$ , aby zagwarantować stabilność układu w różnych warunkach pracy i jednocześnie nie przekraczać dopuszczalnych wielkości sterujących.

Własności tego rozwiązania zweryfikowano nie tylko symulacjami, ale także poprzez ich implementację w układzie odwróconego wahadła. Otrzymany w ten sposób układ regulacji porównano z rozwiązaniami klasycznymi – liniowym, liniowym z nasyceniem i z regulatorem ślizgowym opartym o wzorcowe prawo osiągnięcia (ang. *reaching law*) [19]. W zaprojektowanym regulatorze typu soft VSC uzyskano najlepsze wyniki, tj. najbardziej łagodne przebiegi wielkości sterujących przy zachowaniu wysokiej dynamiki układu zamkniętego.

Należy pamiętać, że prawo sterowania zrealizowane w oparciu o metodologię VSC wymaga większych nakładów obliczeniowych niż stosowanie rozwiązań klasycznych, na przykład regulatorów liniowych. Jednak wobec obserwowanego w ostatnich latach obniżenia ceny mocy obliczeniowej, koszt ten należy ocenić jako uzasadniony, biorąc pod uwagę korzyści jakościowe w układzie regulacji.

### 5.3.2.3 Sieciowe podejście do poprawiania własności sieciowych układów sterowania – sieci ad-hoc

---

Sieciowe układy sterowania mają sens tylko wówczas, gdy jakość działania sieci i jej niezawodność jest wystarczająca do realizacji nadrzędnego celu regulacji. W przypadku stosowania typowych sieci przemysłowych takich jak CAN, Profibus, RTEthernet, ARINC i innych podobnych [20, 21], sama konstrukcja opartych o nie rozwiązań gwarantuje

odpowiednią jakość komunikacji rozumianą jako gwarancję niewielkich i mało zmiennych opóźnień, czy rzadkich strat. Jest to jednak okupione dużym kosztem instalacji i utrzymania takich systemów. Ponadto, wszystkie te rozwiązania wymagają użycia dedykowanego okablowania i płaskiej struktury sieci. Co jednak zrobić w przypadku, gdy instalacja jest trudna lub niemożliwa w realizacji, a realna jest wyłącznie komunikacja bezprzewodowa – zwykle wykorzystująca fale elektromagnetyczne o częstotliwościach radiowych (współcześnie 400 MHz – 6 GHz), nazywana dalej komunikacją radiową, ale też akustyczna czy optyczna (ang. *Free Space Optics*)? Czy w takich przypadkach stosowanie sieciowych układów sterowania jest z góry skazane na niepowodzenie? Pytanie to staje się szczególnie istotne współcześnie, w erze dążenia do tworzenia systemów określanych paradygmatem *Industry 4.0* [22], a w szczególności – Internetu Rzeczy (ang. *Internet of Things* – IoT) oraz Przemysłowego Internetu Rzeczy (ang. *Industrial IoT* – IIoT) [23], stanowiącego fragment rynku o znaczącym potencjale ekonomicznym.

Innym problemem związanym z paradygmatem *Industry 4.0* [22] jest trudność realizacji sieciowych *rozległych* układów sterowania. Przeniesienie jednostki sterującej do wysokowydajnego komputera często pociąga za sobą konieczność umieszczenia go poza siecią lokalną, w centrum dystrybucji danych (w tzw. „chmurze”). Elementy pośredniczące w transmisji (rutery, bramy, przełączniki, systemy bezpieczeństwa) nie są projektowane i produkowane dla specjalizowanych sieci przemysłowych. Czy zastosowanie komunikacji rozległej również wyklucza wprowadzenie wydajnych sieciowych układów sterowania? Warto zauważyć, że pojęcie *sieci rozległych* jest tu użyte w znaczeniu zbioru sieci lokalnych połączonych ruterami, niekoniecznie rozległych geograficznie.

W pracach [MM-E], [MM-J] zaproponowałem wykorzystanie arytmetyki rozmytej (ang. *fuzzy arithmetics*) do zarządzania ruchem w sieciach komputerowych. Sieć komputerowa zbudowana na zaproponowanych zasadach pozwala na to, by dane użytkownika były przesyłane jednocześnie wieloma ścieżkami. Zwykle uważa się [25], że metryki opisujące połączenia sieci nie powinny mieć związku z chwilowym obciążeniem poszczególnych łączy, ponieważ powodowałoby to powstawanie zjawiska tzw. łopotania tras (ang. *flapping/fluttering*), niekorzystnie wpływającego na jakość transmisji. Zaproponowane rozwiązanie nie jest jednak podatne na ten efekt, ponieważ skutek zmian obciążeń łączy,

jedynie część transmitowanych danych przenoszona jest na kolejne ścieżki. Dzieje się tak ze względu na zastosowanie arytmetyki rozmytej, w której tradycyjne, „ostre” relacje pomiędzy wartościami („mniejszy”, „większy”, „równy”) zastąpione są „miękkimi” („dużo mniejszy”, „trochę mniejszy”, „prawie równy” itp.).

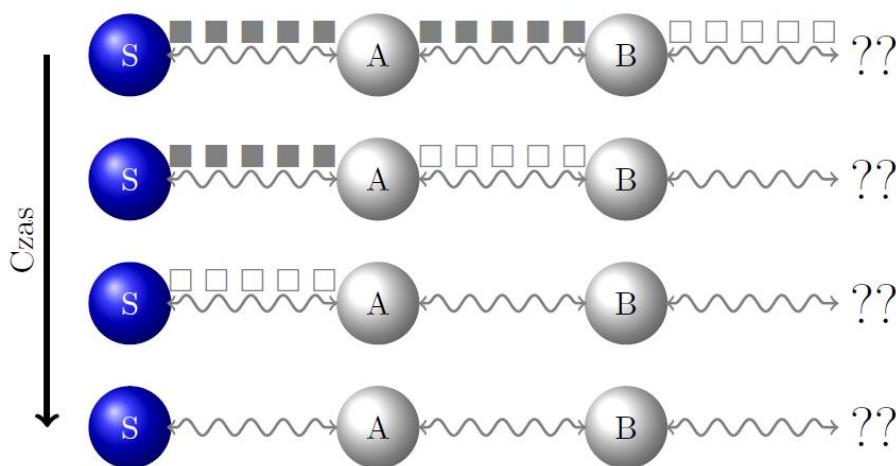
Stosując rozmyte metryki szczególnie korzystne efekty uzyskuje się, gdy koszty łącz odzwierciedlają długości kolejek wyjściowych w ruterach, a zatem pośrednio – czas transmisji. Wówczas nieliczne pakiety docierają do celu w innej kolejności niż ta, w której były wysłane, co poprawia funkcjonowanie protokołów wyższych warstw.

W pracy [MM-C] rozwinięto tę koncepcję. Zaproponowałem w niej nowatorskie podejście do konstrukcji bezprzewodowych sieci przeznaczonych do transmisji danych w czasie rzeczywistym skonstruowane na bazie teorii gier. Teoria ta została zastosowana do znalezienia optymalnych ścieżek transmisji za pomocą szacowania reputacji węzłów pośredniczących. W przedstawionym w [MM-C] rozwiązaniu węzeł ma tym większą reputację, im większy odsetek komunikatów jest prawidłowo retransmitowany. Warto zwrócić uwagę na to, że nie ma tu znaczenia powód niedostarczenia danych, na przykład zakłócenia, wadliwa ścieżka transmisji, czy wyczerpanie baterii.

Opracowane rozwiązanie charakteryzuje się bardzo szybką reakcją na zmiany zachodzące w sieci (mobilność węzłów, zmiana ich obciążenia, poziom zakłóceń). Jednocześnie nie zakłada ono żadnej specyficznej topologii. Należy podkreślić, że zastosowanie teorii gier w systemach komunikacji jest podejściem stosunkowo nowym [24], chociaż jej osiągnięcia są szeroko dyskutowane w innych dziedzinach wiedzy, w szczególności w ekonomii.

Zaproponowane rozwiązanie nie jest odmianą rutingu (czyli klasycznej metody wyznaczania najlepszej ścieżki transmisji), ponieważ powstająca trasa tworzy się jako wynik współzawodnictwa pakietów na drodze do celu bez udziału zewnętrznego protokołu. W ten sposób uniknięto także niekorzystnego zjawiska łopotania tras będącego częstym efektem działania protokołów rutowania wrażliwych na obciążenie [25].

W wielu pracach dotyczących sieci bezprzewodowych zakłada się istnienie fizycznych połączeń pomiędzy poszczególnymi węzłami. Takie założenie jest realne w sieciach optycznych, akustycznych, ale bardzo rzadko w sieciach radiowych, w których węzły wyposażone są w antenę dookólną. Ze względu na sposób transmisji wszystkie węzły



Rysunek 5. Schemat działania systemu analizy reputacji opartej na teorii gier. Pełne kwadraty obrazują komunikaty, które zostają uznane za potwierdzone, puste – komunikaty uznane za niepotwierdzone.

znajdujące się w odpowiedniej odległości odbiorą komunikat jednocześnie. Niestety możliwe jest także, że komunikat ten zakłóci inną transmisję niepowiązaną z analizowanym przepływem danych. Takie zachowanie sieci postrzegane jest zwykle jako wada. W zaproponowanym rozwiązaniu wada ta została obrócona w zaletę i użyta do znalezienia najlepszej ścieżki transmisji danych uwzględniającej aktywność pozostałych węzłów sieci.

Schemat działania zaproponowanego rozwiązania zilustrowany jest na Rysunku 5. Pokazany jest na nim proces decyzyjny prowadzący do zaprzestania komunikacji przez węzeł „S” ze względu na to, że odbiorca komunikatów nie znajduje się na ścieżce „SAB”. Kluczowym spostrzeżeniem w analizowanym przypadku jest fakt, że jeżeli komunikat wysłany przez węzeł „S” jest przekazywany na swojej potencjalnej trasie do celu, na przykład przez węzeł „A”, to węzeł „S” również odbierze tę wiadomość i uzna ją za potwierdzenie. W żadnym momencie węzły nie wybierają przekaźnika komunikatów. Transmisje mają charakter rozgłoszeniowy, co wynika z fizycznej zasady działania systemu transmisji danych wykorzystujących fale elektromagnetyczne o częstotliwościach radiowych. W przykładzie z Rysunku 5, węzeł „S” początkowo zakłada, że dane przez niego generowane przekazywane są do skutecznemu celu, ponieważ docierają do niego potwierdzenia pochodzące z węzła „A”. Podobnie zachowują się węzły „A” i „B”. Po pewnym czasie węzeł „B” podejmuje decyzję o nieprzekazywaniu dalej danych pochodzących z węzła „S”, ponieważ nikt ich nie odbiera. Później taką samą decyzję podejmuje węzeł „A” (ponieważ przesyłane przez niego wiadomości przestały być potwierdzane) i za chwilę węzeł „S” podejmie podobną decyzję. Jeżeli jednak jakkolwiek

inny węzeł będzie przekazywał dalej wiadomości, to dany węzeł będzie kontynuował ich wysyłanie. Jeżeli ta sama wiadomość przekazywana jest przez kilka węzłów, a odbiorca potwierdza tylko pierwszą z nich, ignorując pozostałe, to z punktu widzenia węzłów znajdujących się na „dłuższej” trasie ścieżka się kończy.

Poszczególne węzły są autonomiczne i dopóki nie podejmują prób transmisji danych, nie znają położenia innych węzłów w swoim sąsiedztwie. Jednak w przypadku odebrania wiadomości, podejmując decyzję czy przekazać ją dalej, czy też nie, węzeł zachowuje się samolubnie (ang. *selfish*). Oznacza to, że podejmuje próbę przekazania informacji, o ile jego zysk z tego tytułu przekroczy koszty (nakłady) konieczne do przeprowadzenia transmisji. Wspomniany zysk zdefiniowany jest następującą macierzą korzyści zależną od prawdopodobieństwa decyzji podejmowanych przez węzły „X” i „Y” – odpowiednio  $q_x$  i  $q_y$ :

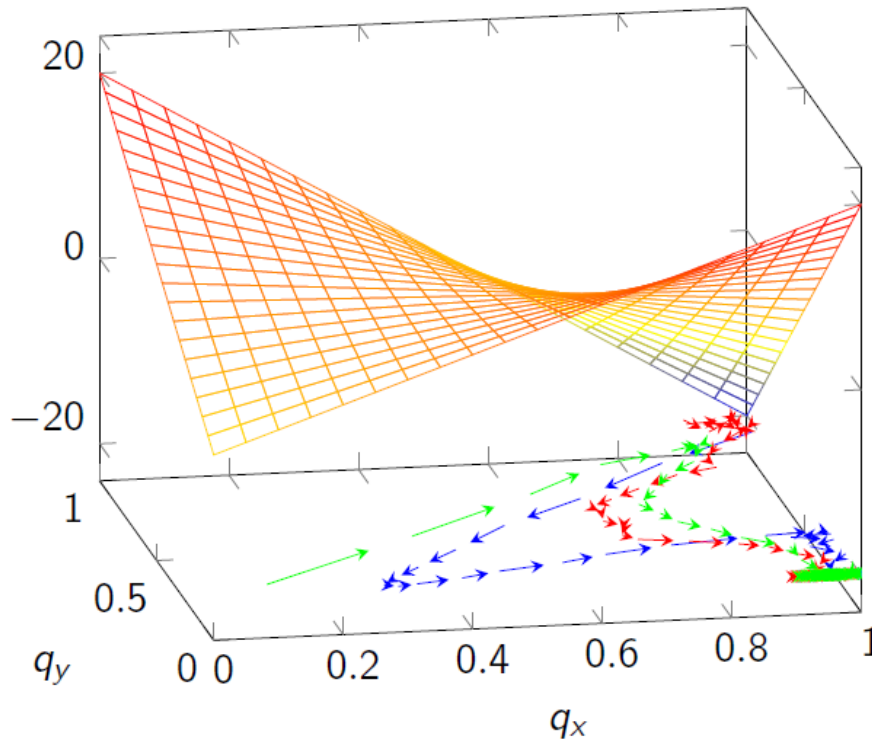
|           |                        |                     |                        |      |
|-----------|------------------------|---------------------|------------------------|------|
|           |                        | Węzeł „Y”           |                        | (12) |
|           |                        | Przekaż – ( $q_y$ ) | Odrzuć – ( $1 - q_y$ ) |      |
| Węzeł „X” | Przekaż – ( $q_x$ )    | $(x_{ff}, y_{ff})$  | $(x_{fd}, y_{df})$     |      |
|           | Odrzuć – ( $1 - q_x$ ) | $(x_{df}, y_{fd})$  | $(x_{dd}, y_{dd})$     |      |

W (12) zapis  $(x_{fd}, y_{df})$  należy rozumieć w ten sposób, że jeżeli węzeł „X” podejmie decyzję „Przekaż”, a węzeł „Y” – decyzję „Odrzuć”, to zysk węzła „X” będzie określony liczbą  $x_{fd}$ , zaś węzła „Y” –  $y_{df}$ . Pozostałe parametry należy rozpatrywać w ten sam sposób. Wyzwanie stanowi fakt, że węzeł „X” nie zna ani parametrów zysków węzła „Y”, ani – co ważniejsze – prawdopodobieństwa  $q_y$ , z jakim ów węzeł będzie próbował przekazać komunikat. Wartości te są szacowane na podstawie skutków podejmowanych decyzji.

W pracy [MM-C] zaproponowano, zrealizowano i przebadano eksperymentalnie ewolucyjną strategię [26] korygowania prawdopodobieństw transmisji oraz doboru zysku węzłów (współczynników  $x_{ff}, x_{fd}, x_{df}, x_{dd}, y_{ff}, y_{fd}, y_{df}, y_{dd}$ ), tak by sieć zachowywała się w sposób bliski optymalnemu w sensie Pareto.

W oparciu o macierz zysków (12) każdy z węzłów tworzy funkcję użyteczności zależną zarówno od znanych jemu parametrów, jak i od tych szacowanych na podstawie analizy zjawisk zachodzących w sieci, czyli (przy założeniu, że  $x_{df} = y_{df} = 0$ ):





Rysunek 6. Przykład adaptacji prawdopodobieństw wysyłania komunikatów (poniżej) wraz z początkową powierzchnią zysków (powyżej).

|                     |                |                 |                              |                                   |  |                                     |
|---------------------|----------------|-----------------|------------------------------|-----------------------------------|--|-------------------------------------|
| Adres przeznaczenia | Adres źródłowy | Adres nadajnika | Adres poprzedniego nadajnika | Unikalny identyfikator komunikatu | Inne pola niezwiązane z projektowanym protokołem | Zawartość określona przez aplikację |
|---------------------|----------------|-----------------|------------------------------|-----------------------------------|--|-------------------------------------|

Rysunek 7. Ramka zaproponowanego protokołu wymiany danych. Adres źródłowy, przeznaczenia, identyfikator komunikatu i zawartość wiadomości nie są zmieniane podczas przekazywania wiadomości. Adres nadajnika i adres poprzedniego nadajnika zmieniają się przy każdym przeskoku.

$$\begin{aligned}
 V_x &= q_x(x_{ff}q_y + (x_{fd} - x_{dd})(1 - q_y)) + x_{dd}(1 - q_y), \\
 V_y &= q_y(y_{ff}q_x + (y_{fd} - y_{dd})(1 - q_x)) + y_{dd}(1 - q_x).
 \end{aligned}
 \tag{13}$$

Suma tych funkcji tworzy powierzchnię hiperboloidy parabolicznej (Rysunek 6), przy czym parametry  $x_*$  i  $y_*$  są korygowane poprzez nagradzanie i karanie węzłów. Efektem tych działań jest zmiana kształtu powierzchni (w szczególności przesunięcie punktu równowagi). Natomiast zmiana zachowania węzłów realizowana jest poprzez zmianę prawdopodobieństwa przekazania informacji ( $q_x$  i  $q_y$ ) według zależności:

$$\begin{aligned}\frac{\partial V_x}{\partial q_x} &= (x_{ff} - x_{fd} + x_{dd})q_y + (x_{fd} - x_{dd}), \\ \frac{\partial V_y}{\partial q_y} &= (y_{ff} - y_{fd} + y_{dd})q_x + (y_{fd} - y_{dd}).\end{aligned}\tag{14}$$

Nieznane prawdopodobieństwa wysyłania komunikatów – odpowiednio  $q_y$  i  $q_x$  – szacowane są na podstawie opracowanego protokołu wymiany danych. Warto podkreślić jest to, że opisywany protokół zakłada interpretację nagłówków każdego komunikatu i nie jest protokołem kontrolnym stosowanym w typowych rozwiązaniach sieciowych. Rysunek 6 przedstawia przykładową adaptację systemu korygowania parametrów współzawodniczących węzłów w obecności zakłóceń, natomiast proponowana postać ramki komunikatu zilustrowana jest Rysunkiem 7.

Zarówno sposób wykorzystania metryk rozmytych, jak i mechanizmu gry (w sensie stosowanym w teorii gier) do znajdowania optymalnego rozłożenia ruchu pomiędzy węzłami sieci jest oryginalnym dorobkiem habilitanta. Porównanie działania zaproponowanego mechanizmu [MM-C] z rozwiązaniem typowym polegającym na implementacji protokołów rutowania [27] wskazuje na podobne własności transmisyjne w przypadku danych strumieniowych (transmisji ogólnego przeznaczenia) i znacznie lepsze w przypadku komunikatów służących do realizacji funkcji układu sterowania [28].

#### **5.3.2.4 Sieciowe podejście do poprawy własności sieciowych układów sterowania – transmisja wielościeżkowa**

W pracach [MM-A] i [MM-F] przedstawiono metodę optymalizacji prowadzącą do poprawy jakości transmisji w sieci opartej o wielościeżkową wersję protokołu TCP. Zastosowanie protokołu TCP w układach sterowania wymaga jednak komentarza. Protokół ten był projektowany do przekazywania danych strumieniowych. W przypadku przesyłania tego rodzaju informacji stracone części (segmenty) są przez protokół odzyskiwane i przesyłane ponownie. W przypadku informacji wykorzystywanych do sterowania w systemach typu TD jest to nie tylko zbędne (odzyskane segmenty, gdy już dotrą do celu są bezużyteczne – Rysunek 3), ale wręcz szkodliwe, ponieważ retransmisje powodują wydłużenie opóźnienia w dostarczaniu aktualnych informacji. Dlaczego zatem celowe jest rozważanie protokołu TCP w układach regulacji automatycznej? Powody są dwa:

- transport danych w układach sterowania typu ED, który wymaga potwierdzenia dostarczonych danych;
- konstrukcja zabezpieczeń w sieciach komputerowych przeciwdziałających szkodliwym aktywnościom użytkowników i aplikacji.

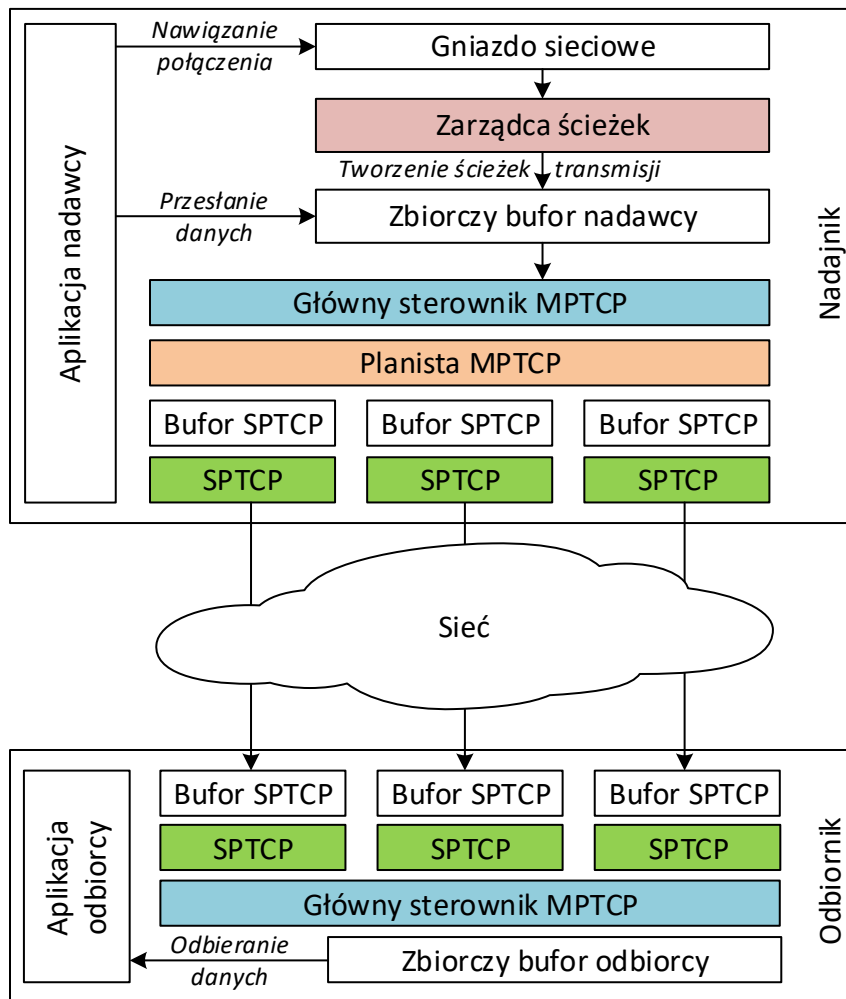
Z tych względów ruch sieciowy oparty o protokoły bezpołączeniowe (UDP, RTP) jest często ograniczany do niezbędnego minimum. W praktyce tylko dane wykorzystujące protokół TCP mogą przenikać przez zabezpieczenia administracyjne, zatem tam, gdzie stosowane jest sterowanie hierarchiczne, lub w sieciach rozległych, projektanci są niejako „skazani” na jego użycie.

W swojej tradycyjnej wersji protokół TCP nie sprawdza się jednak zupełnie w zadaniach związanych ze sterowaniem [4, 29, 30]. W pracy [MM-I] do poprawienia jakości jego działania wykorzystano informacje pochodzące od protokołów rutowania. Uzyskano w ten sposób regulację prędkości transmisji przeciwdziałającą powstawaniu zatorów i przez to zmniejszenie liczby odrzucanych segmentów. Niestety, praktyczne zastosowanie tego rozwiązania jest trudne ze względu na ograniczenia administracyjne związane z bezpieczeństwem sieci.

Gdyby jednak można było zastąpić typową, pojedynczą ścieżkę komunikacji ich zbiorem, to wady zastosowania protokołu TCP w sieciowych układach sterowania zostałyby w znacznym stopniu ograniczone [30, 31]. Ponadto przy takim podejściu wzrasta odporność układu na awarie i przeciążenia sieci, gdyż prawdopodobieństwo wystąpienia tego rodzaju zjawisk na wielu ścieżkach jednocześnie jest niewielkie. W pracach [MM-E], [MM-J] rozważano metody rutowania równoważące obciążenie, ale niepodatnego na efekt łopotania tras (ang. *flapping/fluttering*) [25]. Metoda równoważenia obciążenia łącz zaproponowana w pracach [MM-E] i [MM-J] nie pozwalała jednak na równoważenie ruchu w punktach końcowych (komunikujących się komputerach).

Ostatnio [32] wzrosła popularność rozwiązań polegających na stosowaniu wielu ścieżek transmisji równocześnie, właśnie z punktu widzenia terminali końcowych (ang. *peer/endpoint*). Jednym z najczęściej analizowanych protokołów tej klasy jest protokół MPTCP (ang. *MultiPath TCP*) [33, 34], rzadziej protokoły bezpołączeniowe takie jak MP RTP (ang. *MultiPath RTP*) [35], czy iPRP (ang. *Parallel Redundancy Protocol*) [29].

Ze względów praktycznych i jego zastosowanie w szerokim zakresie usług sieciowych [36, 37], dalszej analizie został poddany protokół MPTCP. Protokół ten zbudowany jest w



Rysunek 8. Architektura protokołu MPTCP według implementacji referencyjnej [42]. Kolorami wyróżniono podstawowe moduły. Pozostałe bloki oznaczają części systemu wpływające pośrednio na przebieg transmisji.

oparciu o architekturę pokazaną na Rysunku 8. Po nawiązaniu tradycyjnego połączenia TCP na wybranej arbitralnie ścieżce podstawowej, jeżeli to możliwe, rozpoczyna swoje działania moduł zarządcy ścieżek (ang. *path manager*), który zajmuje się tworzeniem i usuwaniem kanałów transmisji. Moduł ten działa przez cały czas istnienia połączenia niezależnie od aktualnego stanu transmisji danych. Dane przesyłane przez aplikację umieszczane są w zbiorczym buforze protokołu, skąd pobiera je główny sterownik MPTCP. Tempo pobierania danych z tego bufora zależy zarówno od aktywności sieci, aplikacji, jak i celu realizowanego przez omawiany komponent protokołu. Takimi celami może być zwiększenie prędkości transmisji, ale może też być zapewnienie jej sprawiedliwości (ang. *fairness*) lub inne. Choć w typowych implementacjach MPTCP, do nadzorowania przepływu strumienia danych aplikacji wykorzystywane są uproszczone wersje tradycyjnych implementacji sterowników TCP [MM-F], to tak być nie musi. W tej

warstwie bowiem nie ma strat, które koryguje się w innych modułach obsługujących proces komunikacji (SPTCP).

W pracy [MM-F] zaproponowałem podejście do projektowania modułu głównego regulatora oparte o aparat matematyczny dyskretnego sterowania ślizgowego (ang. *Discrete Sliding Mode Control* – DSMC) [38, 39]. W pracy [MM-F] udowodniłem w sposób formalny własności układu sterowania zbudowanego w oparciu o takie podejście.

Główny sterownik przesyła dane do modułów SPTCP (ang. *SinglePath TCP*) za pośrednictwem planisty (ang. *scheduler*), którego zadaniem jest podział strumienia danych generowanego przez aplikację na poszczególne kanały wymiany danych. Segmenty kierowane do każdego z tych kanałów (ścieżek transmisji) umieszczane są w odpowiadającym mu buforze SPTCP. Sama transmisja nadzorowana jest klasycznymi, jednościeżkowymi wersjami protokołu TCP – SPTCP [40]. W literaturze poddano analizie kilkadziesiąt odmian tego protokołu [41], a kilkanaście zaimplementowano we współczesnych dystrybucjach systemu operacyjnego Linux.

Każdy z modułów wchodzących w skład protokołu MPTCP projektowany jest niezależnie od pozostałych. Pomiedzy nimi, aplikacją i urządzeniami sieciowymi pośredniczącymi w transmisji można zaobserwować jednak wiele powiązań i nieoczywistych sprzężeń. Ponieważ rozwój referencyjnej implementacji protokołu MPTCP [42] skupia się przede wszystkim na usuwaniu niekonsekwencji projektowych i niepożądanych zależności, to wymienione komponenty mogą być udoskonalane w sposób autonomiczny korzystając z zalet architektury modularnej.

W pracy [MM-A] opisano dwa podejścia do tworzenia modułu planisty. W obu przypadkach celem jest minimalizacja energii zużywanej przez urządzenie końcowe. Chociaż to ważne zagadnienie poruszane było w wielu innych pracach [43-48], rozwiązania tam proponowane były ograniczone rodzajem przesyłanych danych lub też wymagały modyfikacji stosu sieciowego.

Jeżeli celem optymalizacji miałyby być zmniejszenie zużycia energii przez komunikującą się stację, to wydawać by się mogło, że jeżeli stacja dysponuje  $N$  interfejsami logicznymi o różnej wydajności energetycznej, to należy używać tylko jednego z nich – tego, który ma najmniejsze wymagania względem zasilania. Tak jednak nie jest, ponieważ nie tylko interfejsy komunikacyjne wymagają zasilania, ale znacząca ilość energii wydatkowana jest na aktywność aplikacji, w tym uruchomienie dodatkowych rdzeni procesora,

podświetlenie ekranu itp. Zakłada się tu, że jeżeli aplikacja zakończy transmisję, to zakończy również aktywność obliczeniową, tak jak to ma miejsce w przypadku typowego wzorca działania użytkownika przeglądarki internetowej. Ilość energii rozpraszanej przez interfejsy i część obliczeniową systemu w ujęciu średnim jest znacząca, a jej chwilowe zużycie jest silnie zmienne [MM-A]. Wydatek energetyczny interfejsów przewodowych jest co najmniej o trzy rzędy wielkości mniejszy niż interfejsów sieci komórkowych. Dla tego samego interfejsu sieci bezprzewodowej może się w czasie jednej sekundy zmieniać o rząd wielkości [MM-A]. Różnice wolumenu energii zużytej przez elementy systemu, inne niż te związane z siecią, są jeszcze większe.

Rozwiązania modułu planisty zaproponowane przez autorów protokołu [42] sprawdzają się doskonale w przypadku, gdy ścieżki są w przybliżeniu symetryczne (mają podobne przepustowości i opóźnienia), a także, gdy czas dostarczania danych nie ma zasadniczego znaczenia. W przeciwnym razie działają nieoptymalnie (zwiększając opóźnienia w sieci, zużycie energii, a czasem zmniejszenie prędkości transmisji) [MM-A]. W przypadku NCS wady te są szczególnie dotkliwe [31].

Dlatego też w [MM-A] zaproponowałem rozwiązanie optymalizujące zużycie energii ( $\mathcal{G}$ ) przez stację i jej interfejsy sieciowe równocześnie, w postaci kryterium

$$\mathcal{G} = \sum_{i=1}^N \mathcal{G}_i + D \max_i T_i, \quad (15)$$

gdzie  $N$  jest liczbą ścieżek,  $D$  – mocą pobieraną przez podsystemy niezwiązane z transmisją sieciową,  $T_i$  – czasem transmisji poprzez ścieżkę  $i$ .

W (15) energia rozproszona przez interfejs logiczny  $i$  –  $\mathcal{G}_i$  określona jest jako

$$\mathcal{G}_i = \int_0^{T_i} p_i(t) dt, \quad (16)$$

gdzie  $p_i(t)$  jest funkcją opisującą zmienność w czasie mocy pobieranej przez ten interfejs.

Przy założeniach, że

- transmisja jest długotrwała (czas potrzebny na jej zainicjowanie jest pomijalnie mały),
- kanały są od siebie niezależne,

udowodniłem formalnie, że minimalne zużycie energii względem kryterium (15) ma miejsce wtedy, gdy stosunek ilości danych wysłanych kanałem  $i$  i  $j$  określony jest zależnością

$$\forall_{i,j \in [1,N]} \frac{\beta_i}{\beta_j} = \frac{P_j \text{av}(c_j)}{P_i \text{av}(c_i)}, \quad (17)$$

gdzie  $\beta_i, \beta_j$  są wolumenami wysyłanych danych,  $\text{av}(c_i), \text{av}(c_j)$  – średnią przepustowością rozważanych kanałów, zaś współczynniki  $P_i, P_j$  – odpowiednio skonstruowanymi współczynnikami mocy możliwymi do określenia w każdej chwili na podstawie prowadzonych pomiarów [MM-A].

Równanie (17) jest uogólnieniem optimum Kleinrocka [49], którego znalezienie, ze względu na wahania przepustowości i konieczność rozpraszania obliczeń, w zdecentralizowanym systemie transmisyjnym, zostało uznane za niemożliwe [50]. Nie znaczy to jednak, że nie można go z wystarczającą dokładnością oszacować korzystając z faktu, że w analizowanym przypadku obliczenia przepustowości dla każdego kanału są realizowane praktycznie w tej samej chwili. Wskazałem, że dobrym przybliżeniem jest

$$\text{av}(c_i) \approx \frac{f_i}{\tau_i}, \quad (18)$$

gdzie  $f_i$  jest liczbą segmentów wysłanych, lecz niepotwierdzonych (ang. *in-flight data*), zaś  $\tau_i$  jest uśrednionym czasem dostarczenia danych (ang. *Smoothed Round Trip Time* – SRTT). Obie te wartości są znane w każdej współczesnej implementacji protokołu TCP.

Jako dodatkową korzyść przeprowadzonej analizy ustaliłem też próg wolumenu danych, poniżej którego nie warto zestawiać dodatkowych połączeń. Okazuje się, że w przypadku krótkotrwałej transmisji stosowanie wielu ścieżek prowadzi do zmniejszenia szybkości przesyłania danych i zwiększenia zużycia energii, jest zatem nieopłacalne. Wcześniej ów próg opłacalności określany był empirycznie.

Warunkiem stosowalności podziału (17) jest m. in. niezależność kanałów. Nietrudno się jednak przekonać, że spełnienie tego warunku bywa problematyczne, ponieważ poszczególne strumienie mogą na siebie oddziaływać w dowolnym miejscu ścieżki. Jeśli tak jest, to optymalizacja wyrażenia (15) powinna ten fakt uwzględniać. Zwykle metody optymalizacyjne zawiodą przy rozwiązaniu postawionego w ten sposób zadania ze względu na to, że wielkość wzajemnego wpływu podstrumieni na siebie nie jest znana z wyprzedzeniem i może się zmieniać w trakcie przebiegu transmisji. Narzucająca się w

tym przypadku metoda polegająca na wyznaczeniu macierzy korelacji nie prowadzi do pożądaných rezultatów, gdyż wymaga dużej liczby danych wejściowych dotyczących wydajności kanałów, co nie jest realne przy typowym użyciu protokołu MPTCP [36, 37]. Dlatego też do poszukiwania optymalnego zadania zastosowano, zmodyfikowane pod kątem implementacji w środowisku sieciowym, podejście nazywane *Extremum Seeking Control* – ESC [51].

Stosując ESC, by znaleźć minimum funkcji (15), należy tak dobrać parametry podziału ruchu, by spełnić warunek

$$\dot{g} = \frac{\partial g}{\partial c_1} \dot{c}_1 + \frac{\partial g}{\partial c_2} \dot{c}_2 + \dots + \frac{\partial g}{\partial c_N} \dot{c}_N \leq 0, \quad (19)$$

co jest równoważne

$$\dot{g} = -\sum_{i=1}^N \left( \rho_i \frac{\beta_i}{av^2(c_i)} \dot{c}_i \right) - D_w \frac{\sum_{i=1}^N \beta_i}{\left[ \sum_{i=1}^N av(c_i) \right]^2} \sum_{i=1}^N \dot{c}_i \leq 0, \quad (20)$$

przy czym współczynniki  $\rho_i$ ,  $D_w$  są współczynnikami zapotrzebowania na energię uzyskanymi na podstawie pomiarów.

Mimo, że uzyskane zależności wydają się stosunkowo złożone, to przy odpowiedniej implementacji programowej możliwe jest ich obliczanie w czasie rzeczywistym, nieznacznie tylko pogarszając efektywność uzyskaną w przypadku rozwiązania przybliżonego (17). Poruszony aspekt wydajności obliczeń jest o tyle istotny, że powiązane operacje muszą być przeprowadzone dla każdego pakietu (segmentu) i dla każdego interfejsu oddzielnie.

Oba rozwiązania zaimplementowałem w systemie Linux. W czasie ich weryfikacji praktycznej uzyskano poprawę efektywności energetycznej transmisji MPTCP w stosunku do rozwiązania referencyjnego [42] rzędu kilkunastu procent, a w warunkach dużego obciążenia sieci – nawet ponad 30%. Uzyskano też dodatkowe korzyści:

- W wielu przypadkach doszło do równoczesnego przyspieszenia szybkości transmisji w stosunku do rozwiązania zaproponowanego przez autorów protokołu [42], co jest związane z ograniczeniem wpływu zjawiska pęcznienia buforów (ang. *Buffer Bloat*) [52] na podział strumienia danych.
- Ze względu na bardziej równomierne obciążenie ścieżek transmisji wyraźnie zmniejszono opóźnienie w dostarczaniu danych i jego wahania, co w przypadku



zastosowania opisywanego rozwiązania w torze głównym układu regulacji jest bardzo istotne.

- Ponadto zastosowanie proponowanych mechanizmów minimalizuje prawdopodobieństwo wystąpienia efektu blokowania kolejek (ang. *Head-of-Line blocking*).

Aby możliwie dokładnie zbadać efekt zaproponowanych modyfikacji, odpowiednie badania przeprowadzone zostały nie tylko w laboratorium, czy za pomocą symulacji, ale korzystając z rzeczywistych urządzeń i z sieci publicznej, w różnych godzinach i dniach (przy różnym natężeniu ruchu), tak aby odzwierciedlić typowy sposób korzystania z badanej klasy terminali. Badania te wykazały, że zalety opracowanych rozwiązań szczególnie mocno uwydatniają się w przypadku mocno obciążonych ścieżek komunikacji, gdy przepustowość sieci podlega znaczącym wahaniom.

#### 5.4 Podsumowanie

---

Obszar badań będący głównym osiągnięciem zgłoszonym w postępowaniu habilitacyjnym dotyczy zagadnień teleinformatyki. Koncentruje się on na poprawieniu działania szczególnego, istotnego zarówno w sferze użytkowej i przemysłowej, rodzaju aplikacji, jakimi są sieciowe układy sterowania. Cel ten został zrealizowany poprzez redukcję wpływu niepożądanych efektów towarzyszących transmisji danych poprzez niededykowane sieci teleinformatyczne. W pracach [MM-A]–[MM-J] opisałem i zaimplementowałem w rzeczywistych obiektach, szereg metod usprawniających działanie sieciowych układów sterowania zarówno od strony aplikacyjnej jak i sieciowej.

Wśród metod leżących po stronie aplikacyjnej należy wyróżnić:

- ograniczenie wrażliwości układu regulacji na wahania opóźnień w transmisji danych poprzez sieci komputerowe,
- obniżenie wymagania technicznych stawianych wobec kanałów komunikacyjnych,
- wygładzenie (ograniczenie zmienności) przebiegów sygnałów sterujących.

Wśród metod leżących po stronie działania sieci teleinformatycznych, których celem jest ograniczenie amplitudy fluktuacji opóźnień oraz zwiększenie niezawodności transmisji warto wyeksponować:

- w warstwie sieciowej – zastosowanie arytmetyki rozmytej do rozpraszania ruchu w sieciach na wiele łączy równocześnie,

- w warstwie sieciowej – zastosowanie teorii gier do wyboru optymalnej trasy przesyłania danych,
- w warstwie transportowej – optymalizację sposobu rozpraszania przepływów danych pomiędzy wiele możliwych ścieżek.

W przedstawionym do oceny cyklu prac

- zastosowałem formalne metody matematyczne do udowodnienia własności sieciowych układów sterowania zmodyfikowanych w sposób wskazany w poprzednich punktach;
- zweryfikowałem empirycznie skuteczność ich działania;
- wskazałem jakie zalety niesie rezygnacja z heurystycznych metod zarządzania ruchem w sieciach na rzecz metod formalnych, podawanych dla klasycznych układów sterowania;
- wskazałem na możliwość zastosowania, w procesie projektowania sieciowych układów sterowania i w zarządzaniu ruchem w sieciach komputerowych, metod typowo proponowanych w ramach innych dziedzin wiedzy (na przykład arytmetyka rozmyta, czy teoria gier). Tego rodzaju podejście jest szczególnie użyteczne, gdy w systemie pojawiają się niepewności, których opis formalny jest złożony. Opisane techniki mogą być wartościowym uzupełnieniem wyników osiągniętych metodami klasycznymi.

Rezultaty przeprowadzonych badań, z uwagi na ich praktyczny charakter, mogą mieć realne przełożenie na gospodarkę, w szczególności w aspekcie współcześnie rozpatrywanego paradygmatu *Industry 4.0*.

## 5.5 Omówienie pozostałych osiągnięć naukowo-badawczych

---

Prócz zainteresowań zogniskowanych na sieciowych układach sterowania, zajmowałem się także innymi aspektami działania systemów rozproszonych. Należy tu wyróżnić trzy główne kierunki badań.

### 5.5.1 Tablice systoliczne

---

Klasyczne podejście do tworzenia systemów równoległych określane jest często akronimami SIMD (ang. *Single Instruction, Multiple Data*) i MIMD (ang. *Multiple Instructions, Multiple Data*). W tego typu rozwiązaniach kolejne procesory wykonują te

same (SIMD) rozkazy przetwarzając inne dane lub (MIMD) każdy procesor przetwarza dane według innego algorytmu. Istnieje też jednak inna metoda przeprowadzania równoległych obliczeń – przetwarzanie potokowe (ang. *pipeline processing*). Stosując tego rodzaju podejście, dane wejściowe umieszcza się w kolejce, z której pierwszy z procesorów je pobiera, przetwarza częściowo, przekazuje kolejnym, a te kontynuują obliczenia, lecz już w innej chwili i w oparciu o wyniki pracy swojego poprzednika. Operacje wykonywane przez procesory są zwykle bardzo proste i zajmują tyle samo taktów zegara w każdym z nich (w szczególności może to być jeden takt). Jeżeli taki – przypominający swoją konstrukcją taśmę produkcyjną – system zrealizuje się w co najmniej w dwóch wymiarach, to mamy do czynienia z tablicą systoliczną, przetwarzającą dane potokowo-równolegle (stosowana jest też nazwa – przetwarzanie czasowo-przestrzenne). Klasyczne przetwarzanie potokowe jest szczególnym przypadkiem tablicy systolicznej.

W pracach [SUP-A], [SUP-E]–[SUP-G] przedstawiono szereg systolicznych rozwiązań problemów obliczeniowych. Rozwiązania te zostały zaimplementowane i zweryfikowane na rzeczywistym komputerze o odpowiedniej architekturze. Komputer ten składał się z 1024 procesorów umieszczonych w tablicy 32×32. Cykl prac związanych z omawianym sposobem przetwarzania informacji został podsumowany monografią [SUP-A].

### 5.5.2 Diagnostyka Sygnałów Wejściowych

---

Współcześnie proces wytwarzania dóbr w przedsiębiorstwach jest zwykle nadzorowany przez systemy typu SCADA (ang. *Supervisory Control And Data Acquisition*). Ich rola polega na dokonywaniu odczytu wartości z setek lub nawet tysięcy czujników i na ich podstawie – korygowanie nastaw regulatorów. Taki system funkcjonuje zgodnie z założeniami projektowymi, gdy wyniki pomiarów są poprawne. Wskutek występowania zanieczyszczeń, zakłóceń, niewłaściwej temperatury, uszkodzeń i innych podobnych powodów, zmierzone wartości nie zawsze odzwierciedlają rzeczywiste wielkości fizyczne z oczekiwaną dokładnością.

Dlatego każda pomierzona wartość zawsze powinna być poddawana weryfikacji. Niestety w większości przypadków weryfikacja ta jest zbyt pobieżna. W szczególności zwykle nie są brane pod uwagę własności dynamiczne śledzonego obiektu, tj. powiązanie między wartościami (także ich zmiennością) pochodzącymi z różnych jego części. Dzieje się tak dlatego, że model dynamiczny obiektu (na przykład kotła w elektrowni) nie jest znany z

wystarczającą dokładnością. Powstała zatem potrzeba, by analizować grupy mierzonych wielkości i weryfikować ich spójność.

Skoro jednak model obiektu nie jest znany i nie jest możliwe stosowanie metod klasycznych (na przykład obserwatorów stanu), to trzeba zadanie zrealizować inaczej. W pracach [SUP-B]–[SUP-D] zaproponowano system, który uczy się na podstawie danych wskazanych przez eksperta, jak wygląda poprawna dynamika zmian w analizowanej grupie mierzonych wielkości. Na podstawie różnic pomiędzy sygnałem bieżącym i wcześniej nauczoną, oceniana jest wiarygodność szeregu czasowego mierzonych wartości.

Do realizacji tego celu zastosowano wyniki analizy czasowo-częstotliwościowej falkowej (ang. *wavelet*) sygnałów pobieranych w czasie rzeczywistym z systemu SCADA za pośrednictwem interfejsu OPC (ang. *OLE for Process Control*). System ten został wdrożony i działa do chwili obecnej w Elektrowni Bełchatów.

### 5.5.3 Prace na rzecz przemysłu

---

Poza aktywnością ściśle badawczą, brałem udział w szeregu przedsięwzięć na rzecz przemysłu. Większość z nich polegała na opracowaniu i wdrożeniu rozwiązań z zakresu informatyki w ramach projektów komercyjnych realizowanych przez różnego rodzaju podmioty gospodarcze. W rozwiązaniach tych przeważały rozproszone systemy diagnostyki i sterowania. Portfolio uzupełniają programy do projektowania wzorów zakardowych, specjalizowane aplikacje zarządzania produkcją i składowaniem oraz algorytmy optymalizacji procesów logistycznych.

Ponadto od 2007 roku do chwili obecnej pełnię rolę eksperta w Programie Innowacyjna Gospodarka (POIG), Narodowego Centrum Nauki (NCN), Narodowego Centrum Badań i Rozwoju (NCBiR), Ministerstwa Inwestycji i Rozwoju (MIiR), Ministerstwa Nauki i Szkolnictwa Wyższego (MNiSW), Łódzkiej Agencji Rozwoju Regionalnego (ŁARR). W ramach prac związanych z tą działalnością oceniłem około 150 wniosków zgłaszanych przez przedsiębiorców i instytucji o dofinansowanie inicjatyw badawczo-rozwojowych i komercjalizacji ich wyników. Liczba ta obejmuje także ocenę protestów.

Ponadto jestem autorem i współautorem licznych ekspertyz dotyczących bezpieczeństwa sieci realizowanych na zlecenie przedsiębiorców, a także ekspertyz sądowych. Współpracuję również w tworzeniu standardu komunikacji przemysłowej OPC UA (*OPC*

Unified Architecture) oraz rozproszonych repozytoriów wykorzystujących technologię *blockchain* w organizacji produkcji.

#### 5.5.4 Publikacje

---

##### 5.5.4.1 Monografia

---

[SUP-A] Ewa Lipowska – Nadolska, Michał Morawski, 1999, *Tablice systoliczne. Problemy wybrane*, PLJ, Warszawa.

Mój wkład w powstanie tej publikacji polegał na opisie praktycznej realizacji wybranych algorytmów na komputerze systolicznym SYSTOLA 1024. Udział procentowy szacuję na 50%.

##### 5.5.4.2 Inne publikacje

---

[SUP-B] Michał Morawski, 2004, *Time-frequency analysis in supervising data acquisition on the large industry plant*, 2<sup>nd</sup> IEEE International Conference on Industrial Informatics INDIN'04. Collaborative Automation-One key for intelligent industrial environments, 131-136.

**MNiSW=15 (WoS).**

[SUP-C] Czesław Andryjowicz, Michał Morawski, 2004, *Ciągła ocena wiarygodności sygnałów pomiarowych i korzyści ekonomiczne z niej wynikające*, SYSTEMS, Journal of Transdisciplinary Systems Science, 9, special issue 2/1, Wrocław, 26-32.

**MNiSW=5.**

Mój wkład w powstanie tej pracy polegał opracowaniu metody oceny wiarygodności sygnału w czasie rzeczywistym. Udział procentowy szacuję na 75%.

[SUP-D] Michał Morawski, 2003, *Serwis WWW do oceny danych diagnostycznych procesy przemysłowego*, XI Konferencja Sieci i Systemy Informatyczne, 169-176.

[SUP-E] Michał Morawski, 1998, *Porównanie systolicznej i szeregowej realizacji algorytmów cyfrowego przetwarzania sygnałów*, XXI konferencja SPETO. Ustroń, 461-464.

[SUP-F] Michał Morawski, 1998, *Komputer systoliczny jako sieciowy serwer kryptograficzny*, VI Konferencja naukowo-technicznej „Sieci i Systemy Informatyczne – teoria, projekty, wdrożenia”, Łódź, 317-326.

[SUP-G] Michał Morawski, 1998, *Systolic realization of 1D and 2D digital filtering using the Isatec SYSTOLA 1024 parallel computer*, International Conference PARELEC'98, Białystok, 307-309.

## Bibliografia

---

- [1] P. Antsaklis, J. Baillieul (eds.), 2007, *Technology of networked control systems*, special issue: Proceedings of IEEE, 95(1), 1-320.

- [2] D. Hristu-Varsekelis, W.S. Levine, 2005, *Handbook of networked and embedded control systems*, Birkhäuser, Boston, MA.
- [3] J. P. Hespanha, P. Naghshtabrizi, Y. Xu, 2007, *A survey of recent results in networked control systems*, Proceedings of the IEEE, 95(1), 138-162.
- [4] Ch. Peng, D. Yue, Q.-L.Han, 2012, *Communication and control for networked complex systems*. Springer-Verlag.
- [5] J. E Normey-Rico, E.F. Camacho, 2008, *Dead-time compensators: A survey*. Control Engineering Practice, 16(4), 407–428.
- [6] K. Warwick, D. Rees (eds), 1988, *Industrial Digital Control Systems*, 2E, Institution of Electrical Engineers.
- [7] M. Heemels, A. Teel, N. van de Wouw, D. Nešić, 2010, *Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance*, IEEE Transactions on Automatic Control, 55(8), 1781–1796.
- [8] Ch. Peng, M. Fei, E. Tian, Y. Guan, 2014, *On hold or drop out-of-order packets in networked control systems*, Information Sciences, 268, 436–446.
- [9] L. Zhang, H. Gao, O. Kaynak, 2013, *Network-induced constraints in networked control systems – a survey*, IEEE Transactions on Industrial Informatics, 9(1), 403–416.
- [10] D. Truong, K. Ahn, 2015, *Robust variable sampling period control for networked control systems*, IEEE Transactions on Industrial Electronics, 62(9), 5630–5643.
- [11] K. Ogata, *Discrete-time control systems*, 2E, Prentice Hall, 1995.
- [12] K. Aström, B. Wittenmark, 1997, *Computer-controlled systems*, 3E., Prentice Hall.
- [13] F. L. Lewis, V. L. Syrmos, 1995, *Optimal control*, 2E, John Wiley & Sons.
- [14] D. Bertsekas, R. Gallager, 1987, *Data networks*, Prentice Hall, ch. 3.
- [15] Y. Yang, Y. Wang, S.-H. Yang, 2005, *A networked control system with stochastically varying transmission delay and uncertain process parameters*, Proceedings of the 16<sup>th</sup> IFAC World Congress, Prague, Czech Republic, (no. WeM15\_TO/3)
- [16] J.-P. Richard, 2003, *Time-delay systems: an overview of some recent advances and open problems*, Automatica, 39, 1667–1694.
- [17] T. Vyhldal, J.F. Lafay, R. Sipahi (eds.), 2014, *Delay systems. From theory to numerics and applications* Springer-Verlag.
- [18] M. Jungers, J. Daafouz, 2013, *Guaranteed cost certification for discrete-time linear switched systems with a dwell time*, IEEE Transactions on Automatic Control, 58(3), 768–772.
- [19] V.I. Utkin, 1992, *Sliding Modes in control and optimization*, Springer-Verlag.
- [20] A. Willig, K. Matheus, A. Wolisz, 2005, *Wireless technology in industrial networks*, Proceedings of IEEE, 93(6): 1130–1151.
- [21] A. Willig, 2008, *Recent and emerging topics in wireless industrial communications: a selection*, IEEE Transactions on Industrial Informatics, 4(2), 102–124.
- [22] R.-A. Koutsiamanis, G. Z. Papadopoulos, X. Fafoutis, J. M. Del Fiore, P. Thubert, N. Montavont, 2018, *From best effort to deterministic packet delivery for wireless Industrial IoT networks*, IEEE Transactions on Industrial Informatics, 14(10), 4468–4480.
- [23] L. Xu, W. He S. Li, 2014, *Internet of Things in industries: A survey*, IEEE Transactions on Industrial Informatics, 10(4), 2233-2243.
- [24] Z. Han, D. Niyato, W. Saad, T. Başar, A. Hjørungnes, 2012, *Game theory in wireless and communication networks: Theory, models, and applications*, Cambridge University Press.
- [25] A. Khanna, J. Zinky, 1989, *The revised ARPANET routing metric*, Proceedings of Symposium on Communications Architectures & Protocols (SIGCOMM'89), ACM, Austin, TX, USA, 45–56.
- [26] W.H. Sandholm, 2009, *Evolutionary game theory*, Encyclopedia of Complexity and Systems Science, Springer, 3176–3205.
- [27] C. Murthy, B. Manoj, 2004, *Ad-hoc wireless networks*, Prentice Hall Communication.
- [28] J. Broch, D.A. Maltz, D.B. Johnson, Y.-C. Hu, J. Jetcheva, 1998, *A performance comparison of multi-hop wireless ad hoc network routing protocols*, Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '98), ACM, New York, NY, USA, 85–97.

- [29] M. Popovic, M. Mohiuddin, D.-C. Tomozei, J.-Y. Le Boudec, 2016, *iPRP—The Parallel Redundancy Protocol for IP networks: Protocol design and operation*, IEEE Transactions on Industrial Informatics, 12(5), 1842-1854.
- [30] M. Polese, F. Chiariotti, E. Bonetto, F. Rigotto, A. Zanella, M. Zorzi, 2018, *A Survey on Recent Advances in Transport Layer Protocols*, IEEE Communications Surveys and Tutorials, preprint, arXiv:1810.03884.
- [31] S. Ferlin, S. Kuceray, H. Claussen, Ö. Alay, 2018, *MPTCP meets FEC: Supporting latency-sensitive applications over heterogeneous networks*, IEEE Transactions on Networking, preprint, arXiv:1807.11059v1.
- [32] C. Xu, J. Zhao, G. Muntean, 2016, *Congestion control design for multipath transport protocols: A survey*, IEEE Communications Surveys and Tutorials 18(4), 2948-2969.
- [33] S. Barré, C. Paasch, O. Bonaventure, 2011, *MultiPath TCP: From theory to practice*, Proceedings of IFIP Networking, Valencia, Spain, 444-457.
- [34] Q. Peng, A. Walid, J. Hwang, S.H. Low, 2016, *Multipath TCP: Analysis, design, and implementation*, IEEE/ACM Transactions on Networking, 24(1), 596-609.
- [35] V. Singh, S. Ahsan, J. Ott, 2013, *MP RTP: multipath considerations for real-time media*, Proceedings of 4<sup>th</sup> ACM Multimedia Systems Conference, Oslo, Norway, 190-201.
- [36] V. Tran, Q. De Coninck, B. Hesmans, R. Sadre, O. Bonaventure, 2016, *Observing real Multipath TCP traffic*, Computer Communications, 94, 114-122.
- [37] O. Bonaventure, C. Paasch, G. Detal, 2017, *Use cases and operational experience with Multipath TCP*, RFC 8041, IETF.
- [38] P. Ignaciuk, A. Bartoszewicz, 2011, *Discrete-time sliding-mode congestion control in multisource communication networks with time-varying delay*, IEEE Transactions on Control Systems Technology, 19(4), 852-867.
- [39] P. Ignaciuk, A. Bartoszewicz, 2013, *Congestion control in data transmission networks – sliding mode and other designs*, Springer-Verlag.
- [40] M. Welzl, 2005, *Network congestion control. Managing Internet Traffic*, Wiley & Sons.
- [41] A. Afanasyev, N. Tilley, P. Reiher, L. Kleinrock, 2010, *Host-to-host congestion control for TCP*, IEEE Communication Surveys and Tutorials. 12(3), 304-342.
- [42] S. Barré, C. Paasch, *MultiPath TCP – Linux kernel implementation*, on-line: <http://www.multipath-tcp.org>.
- [43] C. Paasch, S. Ferlin, O. Alay, O. Bonaventure, 2014, *Experimental evaluation of multipath TCP schedulers*, Proceedings of ACM SIGCOMM CSWS, Chicago, USA, 27-32.
- [44] Q. Peng, A. Walid, M. Chen, S. Low, *Energy efficient Multipath TCP for mobile devices*, 2014, Proceedings of 15<sup>th</sup> ACM International Symposium on Mobile Ad Hoc Network and Computing, Philadelphia, USA, 257-266.
- [45] B. Arzani, A. Gurney, S. Cheng, R. Guerin, B. T. Loo, 2014, *Impact of path characteristics and scheduling policies on MPTCP performance*, Proceedings of IEEE AINA Workshop, 743-748.
- [46] Y. S. Lim, Y. C. Chen, E. M. Nahum, D. Towsley, R. J. Gibbens, E. Cecchet, 2015, *Design, implementation, and evaluation of energy-aware multi-path TCP*, Proceedings of ACM CoNEXT, Heidelberg, Germany, 30:1-30:13.
- [47] B. Han, F. Qian, L. Ji, and V. Gopalakrishnan, 2016, *MP-DASH: Adaptive video streaming over preference-aware multipath*, Proceedings of ACM CoNEXT, Irvine, California, USA, 129-143.
- [48] D. H. Bui, K. Lee, S. Oh, I. Shin, H. Shin, H. Woo, D. Ban, 2013, *GreenBag: Energy-efficient bandwidth aggregation for real-time streaming in heterogeneous mobile wireless networks*, 2013 IEEE 34<sup>th</sup> Real-Time Systems Symposium, Vancouver, BC, Canada, 57-67.
- [49] L. Kleinrock, 1979, *Power and deterministic rules of thumb for probabilistic problems in computer communications*, Proceedings of International Conference on Communications, 43.1.1-43.1.10.
- [50] J. Jaffe, 1981, *Flow control power is nondecentralizable*, IEEE Transactions on Communications, 29(9), 1301-1306.
- [51] K. Ariyur, M. Krstić, 2003, *Real-time optimization by Extremum-Seeking Control*, Wiley & Sons.
- [52] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, V. Jacobson, 2017, *BBR: Congestion-based congestion control*, Communications of the ACM, 60(2) 58-66.