

# Nieliniowe sterowanie predykcyjne ramion manipulatorów

Piotr Tatjewski

Politechnika Warszawska, Nowowiejska 15/19, 00-665 Warszawa, Polska

**Streszczenie:** Przedmiotem artykułu są algorytmy sterowania predykcyjnego (typu MPC – Model Predictive Control) ramion manipulatorów sztywnych. Zastosowano MPC z modelem w przestrzeni stanów i wykorzystano najnowszą technikę tłumienia zakłóceń i błędów modelowania, pozwalającą uniknąć dynamicznego modelowania zakłóceń lub uciekania się do dodatkowych technik ich eliminowania, takich jak SMC. Rozważane są przede wszystkim najbardziej efektywne obliczeniowo algorytmy MPC-NPL (NPL – Nonlinear Prediction and Linearization), w dwóch wersjach: z optymalizacją QP (Quadratic Programming) z ograniczeniami i z jawną optymalizacją bez ograniczeń i spełnieniem ograniczeń nierównościowych a posteriori. Dla wszystkich rozważanych algorytmów przeprowadzono kompleksową analizę symulacyjną sterowania manipulatorem z napędem bezpośrednim, przy dwóch rodzajach zakłócenia: zewnętrznym i parametrycznym. Wyniki porównano z uzyskanymi dla znanego algorytmu CTC-PID (CTC – Computer Torque Control), uzyskując lepszą jakość regulacji algorytmami MPC. Zbadano wpływ długości okresu próbkowania i obliczeniowego opóźnienia sterowania na jakość regulacji, co jest istotne dla algorytmów z szybkim próbkowaniem opartych na modelach.

**Słowa kluczowe:** sterowanie manipulatorów, sterowanie nieliniowe, sterowanie predykcyjne, opóźnienie sterowania, szybkie próbkowanie

## 1. Wprowadzenie

Manipulatory i roboty są szeroko stosowane w coraz bardziej zautomatyzowanych systemach produkcji i montażu. Aby sprostać postawionym zadaniom, sterowniki ramion manipulatorów powinny zapewniać odpowiednią jakość śledzenia zadanych trajektorii przy jednoczesnym generowaniu szybkich, płynnych i bezpiecznych ruchów. Manipulatory to złożone, nieliniowe systemy mechatroniczne, w szczególności manipulatory z napędem bezpośrednim, z silnymi współzależnościami między ramionami. Stąd algorytmy sterowania muszą być projektowane dla obiektów nieliniowych i wielowymiarowych. W literaturze przedmiotu znaleźć można wiele rozwiązań tego problemu, poczynając od prostych klasycznych regulatorów PID, popularnych w praktyce, zwłaszcza dla sterowania silnikami z przekładniami, które skutecznie tłumią efekty sprzężeń dynamicznych między ramionami. Celem niniejszego artykułu nie jest przegląd struktur i algorytmów sterowania ramionami

manipulatorów. Skorzystamy z faktu, że ukazała się ostatnio praca przeglądowa na ten temat [23], z wyczerpującą listą piśmiennictwa, do której odsyłamy zainteresowanego czytelnika. Polecamy również znacznie wcześniejszą, obszerną pracę przeglądową [13]. Warto również wspomnieć o podręcznikach dotyczących sterowania ramionami manipulatorów, w tym standardowego sterowania PD/PID i bardziej rozbudowanych struktur: PD/PID z kompensacją momentów grawitacyjnych oraz PD/PID z wykorzystaniem linearyzacji modelu manipulatora przez nieliniowe sprzężenie zwrotne, zwanego sterowaniem CTC (ang. *Computed Torque Control*) lub sterowaniem z odwrotnością dynamiki (ang. *inverse dynamics control*) [10, 12].

Sterowanie predykcyjne MPC (ang. *Model Predictive Control*) jest obecnie uznaną zaawansowaną techniką sterowania, reprezentowaną przez liczne algorytmy i pakiety oprogramowania stosowane z powodzeniem w praktyce przemysłowej, zwłaszcza w sterowaniu procesami technologicznymi [7, 17, 2, 19, 24, 3, 27]. Algorytmy MPC znalazły zastosowanie w przemysłowych układach sterowania przede wszystkim dla obiektów wielowymiarowych, z silniejszymi oddziaływaniami skrośnymi, z aktywnymi ograniczeniami, o trudnej dynamice, gdzie klasyczne układy regulacji PID nie zapewniały zadowalającej jakości regulacji. MPC jest techniką opartą na modelach, wykorzystującą optymalizację on-line jako podstawowy mechanizm obliczania wartości sygnałów sterujących, stąd wymagającą większej mocy obliczeniowej. Dlatego początkowo stosowano ją do procesów o wolniejszej dynamice i z wykorzystaniem modeli liniowych. Aby zastosować algorytmy MPC do

### Autor korespondujący:

Piotr Tatjewski, piotr.tatjewski@pw.edu.pl

### Artykuł recenzowany

nadesłany 19.06.2022 r., przyjęty do druku 09.03.2023 r.



Zezwala się na korzystanie z artykułu na warunkach licencji Creative Commons Uznanie autorstwa 3.0

szybszych procesów, z jednej strony wykorzystano mocniejsze, szybsze mikroprocesory, a z drugiej opracowywano wersje algorytmów wymagające mniejszej liczby obliczeń, z reguły suboptymalne. Ta ostatnia opcja ma miejsce zwłaszcza przy konstruowaniu algorytmów MPC z nieliniowymi modelami procesów, przez zastosowanie uproszczeń modeli lub ich aproksymacji, takich jak modele liniowe o zmiennych parametrach, modele rozmyte czy sieci neuronowe [15, 27, 16].

Manipulatory to obiekty wykonujące z reguły szybkie ruchy, stąd sterowane z krótkimi okresami próbkowania – a więc z ograniczoną liczbą obliczeń w jednym okresie. Dlatego też bezpośrednie zastosowanie techniki MPC z modelem nieliniowym i nieliniową optymalizacją on-line było w tym przypadku bardzo trudne do zastosowania i spotkać je można w literaturze stosunkowo niedawno, i to z pewnymi specyficznymi modyfikacjami, aby optymalizacja była bardziej wydajna [21, 14]. Z drugiej strony, zadanie sterowania manipulatorem, wielowymiarowe z możliwymi silnymi interakcjami i aktywnymi ograniczeniami, wydaje się doskonale pasować do zastosowania MPC. Dlatego w literaturze opisano w ostatnich latach szereg prób opracowania prostszych algorytmów typu MPC do sterowania manipulatorem. Główną trudność, optymalizację nieliniową on-line, zastępowano zwykle optymalizacją kwadratową dzięki uproszczeniom prowadzącym do przybliżeń liniowych lub problemów optymalizacji typu LMI (ang. *Linear Matrix Inequality*). Jednym z podejść było tu zastosowanie modelu liniowego zmiennego w czasie LPV (ang. *Linear Parameter-Varying*), stanowiącego zbiór modeli liniowych [6]. Drugim, częściej spotykanym podejściem było wykorzystanie linearyzacji modelu nieliniowego. Wilson i inni [30] zastosowali algorytm DMC z modelem odpowiedzi skokowej przeliczanym w każdym okresie próbkowania przez symulację modelu nieliniowego manipulatora. W [1, 8] zastosowano linearyzację modelu w bieżącym punkcie, model liniowy wykorzystano następnie zarówno do predykcji wyjść, jak i optymalizacji kwadratowej (algorytm MPC-NSL [24]). Aby przezwyciężyć trudności wynikające z nieliniowości modelu, zastosowano algorytm MPC do uproszczonego modelu liniowego wynikającego z linearyzacji z nieliniowym sprzężeniem zwrotnym przez CTC [9, 11, 4, 18]. Wydaje się jednak, że zastosowanie algorytmu MPC do niezwykle prostego modelu liniowego uzyskanego po takiej linearyzacji nie jest dobrym rozwiązaniem. MPC jest bowiem algorytmem zdolnym do radzenia sobie z nieliniowościami, wyposażonym w silne mechanizmy tłumienia zakłóceń i niepewności. Przy niedokładnym modelowaniu, w obecności zakłóceń, jego potencjał nie może być dobrze wykorzystany w przypadku zastosowania do skrajnie prostego modelu uzyskanego po linearyzacji sprzężeniem zwrotnym.

Problem tłumienia zakłóceń, zarówno wynikających z niedokładności modelu (niepewność parametrów, niemodelowana dynamika), jak i zewnętrznych, jest istotny w projektowaniu układów sterowania, a w szczególności w projektowaniu MPC z modelami w przestrzeni stanów, jakimi są modele manipulatorów. Aby rozwiązać ten problem, proponowano kombinacje MPC z PID [30] lub z SMC (ang. *Sliding Mode Control*) [11, 18], co prowadziło do nadmiernej komplikacji systemu sterowania. Powodem tego był fakt, że dopiero stosunkowo niedawno zaproponowano wydajną technikę modelowania i tłumienia zakłóceń dla MPC z modelami w przestrzeni stanów, z pierwszymi zastosowaniami w systemach sterowania procesami przemysłowymi. Technika ta jest szczególnie efektywna w przypadku stanu mierzzonego, ponieważ eliminuje potrzebę stosowania obserwatorów czy filtrów zakłóceń dynamicznych [26–29]. W ostatnich z podanych prac rozważone są oba przypadki, ze stanem mierzonym i niemierzonym (obserwowanym, filtrowanym). Przypadek ze stanem mierzonym często spotykany jest w sterowaniu manipulatorami, gdzie na stan składają się zarówno położenia, jak i prędkości. Można tu też zaliczyć

przypadek, kiedy mierzone są jedynie położenia ramion manipulatora, a prędkości są odpowiednio wyliczane przez numeryczne różniczkowanie funkcji położenia.

Celem artykułu jest przedstawienie zastosowania najnowszych realizacji nieliniowych algorytmów MPC do sterowania manipulatorami robotów, bez uciekania się do dodatkowych technik tłumienia zakłóceń jak zastosowanie PID, SMC czy linearyzacji przez sprzężenie zwrotne typu CTC. Prezentację rozpoczyna zwięzłe przedstawienie ogólnego sformułowania algorytmu MPC z optymalizacją nieliniową on-line, wykorzystującą bezpośrednio nieliniowy model manipulatora (algorytm MPC-NO). Algorytm ten jest traktowany dalej głównie jako referencyjny. W kolejnym rozdziale opisany jest najbardziej efektywny, zdaniem autora, algorytm MPC z nieliniową predykcją i linearyzacją (algorytm MPC-NPL), w którym optymalizację nieliniową zastąpiono programowaniem kwadratowym (QP) z ograniczeniami. W Rozdziale 4 zaprezentowana zostanie najbardziej oszczędna obliczeniowo wersja analityczna (jawna) algorytmu MPC-NPL, z analitycznym (jawnym) wzorem na sygnał sterujący i przycinaniem tego sygnału do ograniczeń nierównościowych. Wszystkie wymienione algorytmy będą wykorzystywać technikę tłumienia zakłóceń [26, 28]. W Rozdziale 5 przedstawiona zostanie kompleksowa symulacyjna analiza porównawcza zastosowania rozważonych algorytmów MPC do sterowania manipulatorem z napędem bezpośrednim. Zostaną również podane i porównane wyniki dla algorytmu CTC-PID. Ponadto zbadany zostanie wpływ długości okresu próbkowania i opóźnienia wprowadzonego przez czas obliczeń na jakość sterowania. Artykuł kończy odpowiednie podsumowanie. Zgodnie z wiedzą autora, zastosowanie algorytmu MPC-NPL z efektywnym tłumieniem zakłóceń do sterowania manipulatorem nie zostało dotychczas zaproponowane i przeanalizowane w literaturze.

## 2. MPC z nieliniową optymalizacją (MPC-NO)

Przyjmijmy następujący, znany nieliniowy model manipulatora [13, 10, 12, 23], z czasem ciągłym,

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + F(\dot{q}) + g(q) = u, \quad (1)$$

gdzie  $q, \dot{q}, \ddot{q} \in R^n$  to wektory położenia, prędkości i przyspieszenia ramion manipulatora,  $M(q)$  jest macierzą inercji,  $C(q, \dot{q})$  macierzą sił dośrodkowych i Coriolisa,  $F(\dot{q})$  jest funkcją opisującą tarcie (lepkie i suche),  $g(q)$  jest wektorem momentów grawitacyjnych a  $u$  jest wektorem momentów sterujących.

Zdefiniujemy wektor stanu w postaci  $x^T = [q^T \ \dot{q}^T] \in R^{2n}$ , wówczas model (1) może być zapisany w standardowej postaci

$$\dot{x} = \begin{bmatrix} \dot{q} \\ -M(q)^{-1} [C(q, \dot{q})\dot{q} + F(\dot{q}) + g(q)] + M(q)^{-1} u \end{bmatrix}. \quad (2)$$

Dyskretyzując układ równań (2) schematem Eulera z okresem  $T_c$  i przyjmując wektor wyjść postaci  $y(t) = q(t)$ , uzyskujemy model z czasem dyskretnym:

$$x(k+1) = x(k) + \begin{bmatrix} T_c \dot{q}(k) \\ -T_c M(q(k))^{-1} [C(x(k))\dot{q}(k) + F(\dot{q}(k)) + g(q(k)) - u(k)] \end{bmatrix}, \quad (3)$$

$$y(k) = q(k). \quad (4)$$

Przy projektowaniu algorytmów MPC wykorzystamy uproszczony w stosunku do (3), za to ciągły i różniczkowalny model uwzględniający jedynie tarcie lepkie, tzn. z  $F(\dot{q}) = F_v \dot{q}$ , z macierzą diagonalną  $F_v$ . W opisie algorytmów model ten będziemy zapisywać w standardowej, ogólnej i zwężonej postaci:

$$x(k+1) = f(x(k), u(k)), \quad (5)$$

$$y(k) = \mathbf{C}x(k). \quad (6)$$

Oznaczmy przez  $N$  i  $N_u$  liczby okresów próbkowania  $T_c$  definiujące, odpowiednio, horyzont predykcji i horyzont sterowania. Zadanie optymalizacji dynamicznej algorytmu MPC-NO przyjmujemy w postaci

$$\begin{aligned} x(k+1|k) &= f(x(k), u(k|k)) + v(k), & y(k+1|k) &= \mathbf{C}x(k+1|k), \\ x(k+2|k) &= f(x(k+1|k), u(k+1|k)) + v(k), & y(k+2|k) &= \mathbf{C}x(k+2|k), \\ &\vdots & & \\ x(k+N|k) &= f(x(k+N-1|k), u(k+N-1|k)) + v(k), & y(k+N|k) &= \mathbf{C}x(k+N|k). \end{aligned} \quad (11)$$

$$\begin{aligned} \min_{u(k|k), \dots, u(k+N_u-1|k)} &= \left\{ J(k) = \sum_{p=1}^N \|y^{zad}(k+p|k) - y(k+p|k)\|_{\Psi}^2 + \right. \\ &\left. + \sum_{p=0}^{N_u-1} \|u(k+p|k) - u(k+p-1|k)\|_{\Lambda}^2 \right\} \end{aligned} \quad (7)$$

z ograniczeniami:

- równania predykcji wyliczające  $y(k+p|k)$  z wykorzystaniem modelu, dla danych wartości  $u(k|k), \dots, u(k+N-1|k)$ ,  $p = 1, \dots, N$ ,
- $u_{\min} \leq u(k+p|k) \leq u_{\max}$ ,  $p = 0, \dots, N_u-1$ ,
- $-\Delta u_{\max} \leq u(k+p|k) - u(k+p-1|k) \leq \Delta u_{\max}$ ,  $p = 0, \dots, N_u-1$ ,
- $y_{\min} \leq y(k+p|k) \leq y_{\max}$ ,  $p = 1, \dots, N$ ,

gdzie  $y(k+p|k)$ ,  $u(k+p|k)$  itd., oznaczają wartości wyjść i sterowań predykowanych w bieżącej chwili próbkowania  $k$ , ale na chwilę  $k+p$  [27, 28]. Dla  $p=0$  mamy  $u(k+0-1|k) = u(k-1)$ , czyli sterowania manipulatora wyliczone w poprzedniej chwili próbkowania. Dla  $p \geq N_u$  przyjmujemy  $u(k+p|k) = u(k+N_u-1|k)$ , zgodnie z definicją horyzontu sterowania.  $\Psi$ ,  $\Lambda$  to diagonalne macierze dodatnich współczynników wagowych (spełniających też rolę skalujących)

cych), a z definicji  $\|x\|_{\Psi} = \sqrt{x^T \Psi x}$ . Wektorem zmiennych decyzyjnych  $U(k)$  zadania (7) jest

$$U(k)^T = [u(k|k)^T \ u(k+1|k)^T \ \dots \ u(k+N_u-1|k)^T]. \quad (8)$$

Dla sformułowania równań predykcji wyjść na całym horyzoncie predykcji  $N$ , kluczowe jest zastosowanie odpowiedniego modelu niepewności prowadzącego do eliminacji wpływu zakłóceń i błędów modelowania. W [25] zaprezentowano tzw. „model stałych zakłóceń stanu” (ang. *constant state disturbance model*) zapewniający regulację z zerowymi błędami w stanach ustalonych, dla MPC z liniowymi modelami w przestrzeni stanów. W [26, 28] model ten został uogólniony dla nieliniowych równań stanu. Jest on postaci

$$\begin{aligned} v(k) &= x(k) - x(k|k-1) \\ &= x(k) - f(x(k-1), u(k-1)), \end{aligned} \quad (9)$$

tzn. jest to różnica między stanem  $x(k)$  zmierzonym w chwili  $k$  i stanem predykowanym na tę chwilę na podstawie modelu w chwili poprzedniej  $k-1$ . Różnica ta jest niezerowa w sytuacji błędów modelowania i nieuwzględnianych w modelu (niemierzonych) zakłóceń zewnętrznych. Rozsądnie jest założyć na horyzoncie predykcji te same błędy modelu i z reguły nie

mamy w chwili  $k$  informacji o zmianach niemierzonych zakłóceń w najbliższych  $N$  chwilach – stąd przyjmujemy, że różnica między stanem rzeczywistym a predykowanym będzie taka sama w każdej z chwil  $k+p$ ,  $p = 1, \dots, N$  horyzontu predykcji, tzn.

$$v(k+1|k) = v(k+2|k) = \dots = v(k+N|k) = v(k), \quad (10)$$

stąd w określeniu model „stałych” zakłóceń.

Po zdefiniowaniu modelu zakłóceń, można na horyzoncie predykcji sformułować równania predykcji stanu, a stąd i wyjść, dla zadanej trajektorii sterowań (8):

$$\begin{aligned} x(k+1|k) &= f(x(k), u(k|k)) + v(k), & y(k+1|k) &= \mathbf{C}x(k+1|k), \\ x(k+2|k) &= f(x(k+1|k), u(k+1|k)) + v(k), & y(k+2|k) &= \mathbf{C}x(k+2|k), \\ &\vdots & & \\ x(k+N|k) &= f(x(k+N-1|k), u(k+N-1|k)) + v(k), & y(k+N|k) &= \mathbf{C}x(k+N|k). \end{aligned} \quad (11)$$

Zadanie optymalizacji MPC (7) jest teraz w pełni zdefiniowane i może być rozwiązane. Pierwszy element  $\hat{u}(k|k)$  optymalnej trajektorii sterowania  $\hat{U}(k)$  jest wysyłany do elementów wykonawczych ramion manipulatora jako bieżący sygnał sterujący  $u(k)$ . W następnej chwili próbkowania mamy nowe pomiary i algorytm MPC jest powtarzany.

Zadania optymalizacji (7) różnią się w kolejnych krokach algorytmu zwykle niewiele, wartościami kilku parametrów: nowym pomiarem stanu  $x(k)$  i ostatnią wartością sterowania  $u(k-1)$ . W takich przypadkach przyspieszyć obliczenia może znana w technice optymalizacji tzw. technika „ciepłego startu”, tj. wykorzystanie ostatniej optymalnej trajektorii sterowania  $\hat{U}(k-1)$  jako podstawy następnej początkowej trajektorii sterowania  $\hat{U}(k)$ . Dla przesuwanego horyzontu, polega to na zmodyfikowaniu  $\hat{U}(k-1)$  przez pominięcie pierwszego elementu  $\hat{u}(k-1|k-1)$  i powtórzenie ostatniego. Ciepły start jest możliwy dla horyzontu sterowania  $N_u \geq 2$ , bywa efektywny głównie przy większych wartościach  $N_u$ . Gdy wymagane jest szybkie próbkowanie i mały nakład obliczeń, jak przy sterowaniu manipulatorami, stosowane są zwykle bardzo krótkie horyzonty sterowania i „ciepły start” nie jest zalecany.

Algorytm MPC-NO będzie dalej traktowany jako wzorzec optymalny do porównania z algorytmami szybszymi (z mniejszym nakładem obliczeń), ale suboptymalnymi. Jednak dzięki rozwojowi coraz wydajniejszych mikroprocesorów oraz udoskonalaniu procedur optymalizacji nieliniowej poszerza się obszar możliwych zastosowań algorytmów z optymalizacją nieliniową [21].

### 3. Algorytmy z nieliniową predykcją i linearyzacją (MPC-NPL)

#### 3.1. Algorytm NPL numeryczny

Jeśli w każdej chwili próbkowania zlinearyzujemy model nieliniowy, a następnie stosujemy standardowy algorytm MPC z modelem liniowym do predykcji wyjść i optymalizacji, to uzyskamy tzw. algorytm MPC-NSL (ang. *Nonlinear with Successive Linearizations*), zastosowany do sterowania manipulatorem [8, 27]. Nie jest to jednak podejście zalecane. Obliczeniowo porównywalnym i bardziej skutecznym jest algorytm wykonujący w każdej chwili próbkowania predykcję początkowej trajektorii wyjść przy użyciu pełnego modelu nieliniowego, a następnie linearyzację i zastosowanie optymalizacji z przyrostowym modelem zlinearyzowanym, w celu poprawy trajektorii, np. [27].

Kolejne kroki algorytmu MPC-NPL, w  $k$ -tym okresie próbkowania:

1. Mierzmy stan manipulatora  $x(k)$ , wyliczamy estymatę zakłóceń stanu  $v(k)$  (9).
2. Wyznaczamy początkową trajektorię sterowań  $U^0(k)$  na horyzoncie predykcji,

$$U^0(k)^T = [u^0(k|k)^T \ u^0(k+1|k)^T \ \dots \ u^0(k+N-1|k)^T]. \quad (12)$$

Używając modelu nieliniowego obliczamy dla sterowań  $U^0(k)$  początkową trajektorię stanu  $X^0(k)$  (składającą się z  $x^0(k+p|k)$ ,  $p = 1, \dots, N$ ), a następnie początkową trajektorię wyjść  $Y^0(k)$  (składającą się z  $y^0(k+p|k)$ ,  $p = 1, \dots, N$ ):

$$x^0(k+p|k) = f(x^0(k+p-1|k), u^0(k+p-1|k)) + v(k), \quad (13)$$

$$y^0(k+p|k) = \mathbf{C}x^0(k+p|k), \quad p = 1, \dots, N, \quad (14)$$

gdzie  $x^0(k|k) = x(k)$  jest stanem zmierzonym.

3. Linearyzujemy równania stanu w punkcie  $(x(k), u^0(k|k))$ . Obliczamy macierz  $\mathbf{M}(k)$  określającą liniową zależność między przyrostami sterowań (wokół trajektorii początkowej sterowań) i przyrostami wyjść (wokół trajektorii początkowej wyjść),

$$\Delta Y(k) = \mathbf{M}(k) \cdot \Delta U(k). \quad (15)$$

4. Rozwiązujemy zadanie optymalizacji kwadratowej (20) z równaniami predykcji stanowiącymi sumę trajektorii początkowej  $Y^0(k)$  i przyrostów (15), z rozwiązaniem  $\widehat{\Delta u}(k+p|k)$ ,  $p = 0, \dots, N_u - 1$ .

5. Obliczamy suboptymalną trajektorię sterowań:

$$\begin{aligned} \hat{u}(k+p|k) &= u^0(k+p|k) + \sum_{j=0}^p \widehat{\Delta u}(k+j|k), \quad p = 0, \dots, N_u - 1, \\ &= u^0(k+N_u-1|k) + \sum_{j=0}^{N_u-1} \widehat{\Delta u}(k+j|k), \quad p = N_u, \dots, N-1, \end{aligned} \quad (16)$$

6. Pierwszy podwektor optymalnej trajektorii sterowań  $\hat{u}(k|k)$  przesyłany jest do efektorów manipulatora jako wektor aktualnych wartości sterowań  $u(k)$ .

Początkową trajektorię sterowań  $U^0(k)$  (12) można przyjąć jako „stałą”, tj. wektor  $U^0(k)$  jest złożony z  $N$  podwektorów  $u(k-1)$ ,  $u^0(k+p|k) = u(k-1)$ ,  $p = 0, \dots, N-1$ , albo skonstruować wg zasady „ciepłego startu”, tzn.

$$u^0(k+p|k) = \hat{u}(k+p|k-1), \quad p = 0, \dots, N-2,$$

$$u^0(k+N-1|k) = \hat{u}(k+N-2|k-1).$$

Trajektoria początkowa stała jest efektywna dla krótkich horyzontów sterowania  $N_u$ , co jest typowe dla obiektów o szybkiej dynamice, jak przy rozważanym w niniejszej pracy sterowaniu manipulatorami. Stąd dalsza prezentacja będzie dotyczyła tego wariantu. Zastosowanie „ciepłego startu” jest możliwe dla dłuższych  $N_u$ , ale wymaga rozszerzenia funkcji  $J(k)$  w (20).

W wyniku linearyzacji uzyskujemy model liniowy w przestrzeni stanu:

$$x(k+1) = \mathbf{A}(k)x(k) + \mathbf{B}(k)u(k). \quad (17)$$

Model (3) jest liniowy względem sterowania  $u(k)$ , stąd linearyzacja względem  $u$  jest niepotrzebna do obliczenia  $\mathbf{B}(k)$ ,  $\mathbf{B}(k) = T_c M(q(k))^{-1}$ . Aby uzyskać  $\mathbf{A}(k)$ , należy wyznaczyć pochodne cząstkowe względem  $x$  jedynie ostatnich  $n$  równań stanu w (3), gdyż

$$\mathbf{A}(k) =$$

$$= \begin{bmatrix} \mathbf{0}_n & \mathbf{I}_n \\ \frac{\partial}{\partial x(k)} \left[ -T_c M(q(k))^{-1} \left( (C(x(k)) + F_v) \dot{q}(k) + g(q(k)) \right) \right]_{n \times 2n} \end{bmatrix} \quad (18)$$

gdzie  $\frac{\partial}{\partial x(k)}[\cdot]$  oznacza macierz pochodnych cząstkowych funkcji w nawiasach kwadratowych.

Macierze  $\mathbf{A}(k)$ ,  $\mathbf{B}(k)$  i  $\mathbf{C}$  służą do obliczenia macierzy  $\mathbf{M}(k)$ , w sposób standardowy jak dla algorytmu MPC z liniowym modelem w przestrzeni stanu, np. [27],

$$\mathbf{M}(k) = \begin{bmatrix} \mathbf{CB} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{C(I+A)B} & \mathbf{CB} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C(I+\dots+A^{N_u-1})B} & \mathbf{C(I+\dots+A^{N_u-2})B} & \dots & \mathbf{CB} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{C(I+\dots+A^{N-1})B} & \mathbf{C(I+\dots+A^{N-2})B} & \dots & \mathbf{C(I+\dots+A^{N-N_u})B} \end{bmatrix} \quad (19)$$

gdzie  $\mathbf{A} = \mathbf{A}(k)$ ,  $\mathbf{B} = \mathbf{B}(k)$ , aby wzór (19) zmieścił się w szerokości strony.

Zadanie programowania kwadratowego (QP), liczone w punkcie 4., ma postać:

$$\min_{\Delta U(k)} \left\{ J(k) = \sum_{p=1}^N \|y^{zad}(k+p|k) - y^0(k+p|k) - \mathbf{M}(k)\Delta U(k)\|_{\Psi}^2 + \sum_{p=0}^{N_u-1} \|\Delta u(k+p|k)\|_{\Lambda}^2 \right\} \quad (20)$$

z ograniczeniami:

$$-u_{\min} \leq u(k-1) + \sum_{j=0}^p \Delta u(k+j|k) \leq u_{\max}, \quad p = 0, \dots, N_u-1,$$

$$-\Delta u_{\max} \leq \Delta u(k+p|k) \leq \Delta u_{\max}, \quad p = 0, \dots, N_u-1,$$

$$-y_{\min} \leq y^0(k+p|k) + \mathbf{M}(k)\Delta U(k) \leq y_{\max}, \quad p = 1, \dots, N,$$

Jest to standardowa postać zadania optymalizacji MPC z liniowym modelem obiektu [27], z wektorem zmiennych decyzyjnych  $\Delta U(k)$ ,

$$\Delta U(k)^T = [\Delta u(k|k)^T \ \Delta u(k+1|k)^T \ \dots \ \Delta u(k+N_u-1|k)^T], \quad (21)$$

gdzie  $\Delta u(k|k) = u(k|k) - u(k-1)$ ,  
 $\Delta u(k+1|k) = u(k+1|k) - u(k|k)$ , itd.

Podstawowa różnica między (20) a problemem QP dla standardowego algorytmu MPC z modelem liniowym polega na tym, że początkowa trajektoria wyjść,  $y^0(k+p|k)$ ,  $p = 1, \dots, N$ , jest teraz obliczana na modelu nieliniowym, a nie liniowym, a macierz dynamiczna  $\mathbf{M}(k)$  podlega adaptacji w każdej chwili próbkowania. Ponadto górne i dolne granice ograniczeń nierównościowych w (20) powinny być tak przeformułowane, aby określały ograniczenia dla zmiennych przyrostowych modelu zlinearyzowanego.



### 3.2. Algorytm MPC-NPL analityczny

Rozwiązywanie on-line problemu programowania kwadratowego z ograniczeniami może być zbyt czasochłonne lub kosztowne dla sterowania szybkimi manipulatorami. Aby jeszcze bardziej zredukować nakład obliczeń on-line, opracowano analityczną (jawną) wersję algorytmu MPC-NPL. W algorytmie tym najpierw znajdujemy punkt minimalizujący samą kwadratową funkcję celu  $J(k)$  zadania (20), pomijając chwilowo ograniczenia nierównościowe. Obliczone w ten sposób wartości wektora sterowań są następnie przycinane do ograniczeń nierównościowych. Podejście to wprowadza dodatkową suboptymalność przy aktywności ograniczeń, ale znacznie zmniejsza obciążenie obliczeniowe. Co więcej, przykłady zastosowań pokazały, że taki algorytm może być równie lub podobnie efektywny jak algorytm z optymalizacją QP z ograniczeniami, przede wszystkim dla przypadków bez lub z ograniczoną aktywnością ograniczeń nierównościowych.

Funkcja  $J(k)$  w (20) jest ściśle wypukła, stąd jej minimum jest jednoznaczne i wynika z rozwiązania liniowych równań warunku koniecznego optymalności. Z postaci funkcji  $J(k)$  wynika, że warunek ten można sformułować w postaci [27]:

$$\left[ \mathbf{M}(k)^T \underline{\Psi} \mathbf{M}(k) + \underline{\Lambda} \right] \Delta U(k) = \mathbf{M}(k)^T \underline{\Psi} \left[ Y^{zad}(k) - Y^0(k) \right], \quad (22)$$

gdzie  $\underline{\Psi} = \text{diag}\{\Psi, \dots, \Psi\}$ ,  $\underline{\Lambda} = \text{diag}\{\Lambda, \dots, \Lambda\}$  są macierzami blokowo-diagonalnymi, składającymi się odpowiednio, z  $N$  bloków  $\Psi$  i z  $N_u$  bloków  $\Lambda$ , oraz

$$Y^{zad}(k) = \begin{bmatrix} y^{zad}(k+1|k) \\ y^{zad}(k+2|k) \\ \vdots \\ y^{zad}(k+N|k) \end{bmatrix}_{n_y N \times 1}, \quad Y^0(k) = \begin{bmatrix} y^0(k+1|k) \\ y^0(k+2|k) \\ \vdots \\ y^0(k+N|k) \end{bmatrix}_{n_y N \times 1} \quad (23)$$

Rozwiązanie  $\widehat{\Delta U}(k)$  równania (22) jest suboptymalną trajektorią przyrostów sterowania (wokół trajektorii początkowej) na horyzoncie sterowania,

$$\widehat{\Delta U}(k)^T = \left[ \widehat{\Delta u}(k|k)^T \quad \widehat{\Delta u}(k+1|k)^T \quad \dots \quad \widehat{\Delta u}(k+N_u-1|k)^T \right], \quad (24)$$

Elementy podwektora  $\widehat{\Delta u}(k) = \widehat{\Delta u}(k|k)$  przycinamy do ograniczeń szybkości zmian:

$$\begin{aligned} & \text{if } \widehat{\Delta u}(k)_j > \Delta u_{j,\max}, \quad \overline{\Delta u}(k)_j = \Delta u_{j,\max}; \\ & \text{elseif } \widehat{\Delta u}(k)_j < -\Delta u_{j,\max}, \quad \overline{\Delta u}(k)_j = -\Delta u_{j,\max}; \\ & \text{else } \overline{\Delta u}(k)_j = \widehat{\Delta u}(k)_j; \quad \text{end, } \quad j = 1, \dots, n_u. \end{aligned} \quad (25)$$

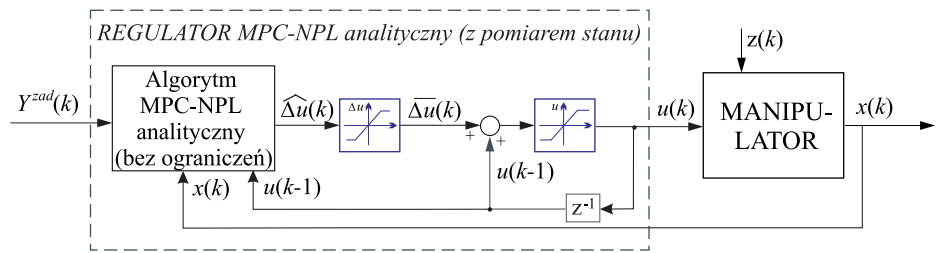
Następnie składowe wektora  $\bar{u}(k) = u(k-1) + \overline{\Delta u}(k)$  przycinamy do dolnej i górnej granicy ograniczeń amplitudy,  $u_{\min}$  i  $u_{\max}$ , w analogiczny sposób. Uzyskany w ten sposób wektor sterujący  $u(k)$  spełnia wszystkie ograniczenia. Struktura przedstawionego algorytmu, nazwanego „analitycznym”

(MPC-NPLa), pokazana jest schematycznie na Rys. 1. Jest on „analityczny”, gdyż  $\widehat{\Delta U}(k)$  można wyrazić wzorem analitycznym

$$\widehat{\Delta U}(k) = \left[ \mathbf{M}(k)^T \underline{\Psi} \mathbf{M}(k) + \underline{\Lambda} \right]^{-1} \mathbf{M}(k)^T \underline{\Psi} \left[ Y^{ref}(k) - Y^0(k) \right]. \quad (26)$$

Jednak rozwiązanie układu równań liniowych (22) wymaga znacznie mniej obliczeń niż wyznaczenie  $\widehat{\Delta U}(k)$  ze wzoru (26). Może to być ważne przy obliczeniach on-line, ponieważ macierz  $\mathbf{M}(k)$  jest przeliczana, ogólnie w każdym okresie próbkowania.

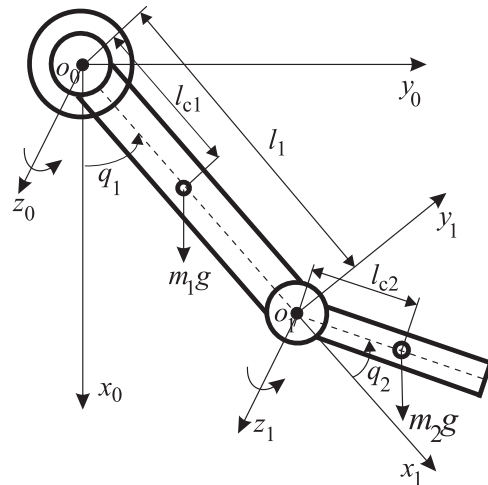
Dla odróżnienia od MPC-NPLa, algorytm MPC-NPL omówiony w poprzednim rozdziale określamy będziemy akronimem MPC-NPLn (numeryczny).



Rys. 1. Struktura algorytmu MPC-NPLa  
Fig. 1. Structure of the MPC-NPLa algorithm

### 3.3. Modele do symulacji i do sterowania manipulatorem

Zastosujemy rozważane algorytmy do sterowania eksperymentalnym dwuwymiarowym planarnym manipulatorem z napędem bezpośrednim (ang. *direct drive*), opisanym szczegółowo w [20]. Schemat manipulatora przedstawiony jest na Rys. 2.



Rys. 2. Schemat eksperymentalnego manipulatora z napędem bezpośrednim

Fig. 2. Diagram of the experimental direct drive manipulator)

Manipulator operuje w płaszczyźnie pionowej  $0_0x_0y_0$ , wektor położenia  $q(t) = [q_1(t) \quad q_2(t)]^T$  składa się z kątów poszczególnych ramion (Rys. 2), zaś wektor sterowań  $u(t) = [u_1(t) \quad u_2(t)]^T$  składa się z momentów obrotowych generowanych przez silniki. Dynamikę manipulatora opisuje model (1), z następującymi postaciami elementów tego modelu:

$$M(q) = \begin{bmatrix} p_1 + 2p_3 \cos(q_2) & p_2 + p_3 \cos(q_2) \\ p_2 + p_3 \cos(q_2) & p_2 \end{bmatrix}, \quad (27)$$

$$C(q, \dot{q}) = \begin{bmatrix} -p_3 \sin(q_2) \dot{q}_2 & -p_3 \sin(q_2) (\dot{q}_1 + \dot{q}_2) \\ -p_3 \sin(q_2) \dot{q}_1 & 0 \end{bmatrix}, \quad (28)$$

$$F(\dot{q}) = \begin{bmatrix} f_{v1} & 0 \\ 0 & f_{v2} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} f_{c1} & 0 \\ 0 & f_{c2} \end{bmatrix} \begin{bmatrix} \text{sgn}(\dot{q}_1) \\ \text{sgn}(\dot{q}_2) \end{bmatrix} = \\ = F_v \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + F_C \begin{bmatrix} \text{sgn}(\dot{q}_1) \\ \text{sgn}(\dot{q}_2) \end{bmatrix}, \quad (29)$$

$$g(q) = \begin{bmatrix} p_4 \sin(q_1) + p_5 \sin(q_1 + q_2) \\ p_5 \sin(q_1 + q_2) \end{bmatrix}. \quad (30)$$

Z danych fizycznych [20] wynikają następujące wartości parametrów:

$$p_1 = 2,352 \text{ Nms}^2, p_2 = 0,102 \text{ Nms}^2, p_3 = 0,084 \text{ Nms}^2, \\ p_4 = 38,466 \text{ Nm}, p_5 = 1,825 \text{ Nm} \\ f_{v1} = 2,288 \text{ Nms}, f_{v2} = 0,175 \text{ Nms}, \\ f_{c1} = 7,17 \text{ dla } \dot{q}_1 > 0 \text{ Nm}, \\ 8,05 \text{ dla } \dot{q}_1 < 0 \text{ Nm}, f_{c2} = 1,734 \text{ Nm}, \\ \text{wraz z ograniczeniami momentów sterujących:} \\ u_{1\text{max}} = 200 \text{ Nm}, u_{2\text{max}} = 15 \text{ Nm}.$$

Do symulacji dynamiki manipulatora w pętli sterowania ze sprzężeniem zwrotnym użyto modelu dyskretnego (3)–(4) z bardzo krótkim krokiem całkowania  $T_c = T_p = 0,0001 \text{ s}$ . Wówczas model ten z bardzo dużą dokładnością odzwierciedla dynamikę czasu ciągłego. Aby uzyskać do projektowania regulatorów MPC model ciągły i różniczkowalny, wykorzystano model (3)–(4) z pominięciem tarcia suchego, tj. z  $F(\dot{q}) = F_v \dot{q}$ . Stąd tarcie suche (Coulomba) traktowane będzie w układzie regulacji z MPC jako dynamika niemodelowana. Jako podstawowy okres próbkowania regulatorów przyjęto  $T_c = 0,005 \text{ s}$ , zbadano również algorytmy z innymi wartościami  $T_c$ .

Zastosowano następujący scenariusz symulacji:

- Długość horyzontu symulacji 2,5 s, trajektorie zadane z dwoma zmianami końcowych położenia kątowych ramion manipulatora (w radianach), realizowane jako trajektorie sześciennicze trwające 0,5 s każda, pierwsza rozpoczynająca się w chwili 0,15 s, druga w chwili 1,5 s.
- Zakłócenie:
  - zewnętrzne (Zu): moment obrotowy 4 Nm (przy maksymalnym momencie obrotowym drugiego siłownika 15 Nm) dodany do wejścia sterującego drugiego ramienia od chwili 1,25 s (połowa horyzontu symulacji); zakłócenie typowe dla testowania jakości regulacji, w tym manipulatorów [10]; lub
  - parametryczne (P1): zmiana dynamiki przez dodanie na końcu drugiego ramienia obciążenia 1 kg, co skutkuje zmianami parametrów obiektu do wartości:  $p_1 = 2,596 \text{ Nms}^2, p_2 = 0,144 \text{ Nms}^2, p_3 = 0,154 \text{ Nms}^2, p_4 = 42,88 \text{ Nm}, p_5 = 3,351 \text{ Nm}$ . Parametry te są jedynie w modelu manipulatora jako obiektu w pętli regulacji, regulatory projektowane są dla modelu nominalnego (bez zakłóceń). Zauważmy, że zewnętrzne zakłócenia wymagają tłumienia, ale nie zmieniają dynamiki pętli sprzężenia zwrotnego. Zmiany parametrów obiektu zmieniają tę dynamikę.

- Jakość sterowania zdefiniowana przez kryterium ISE (ang. *Integrated Squared Error*):

$$\text{ISE} = \sum_{k=1}^{N_{\text{sym}}} (e_1(k)^2 + e_2(k)^2) * T_c, \quad (31)$$

gdzie  $N_{\text{sym}}$  to liczba okresów próbkowania  $T_c$  określająca horyzont symulacji.

Przyjęto następujące wartości parametrów w funkcji celu zadań optymalizacji MPC:  $[\psi_1 \ \psi_2] = [10 \ 10], [\lambda_1 \ \lambda_2] = [0,0001 \ 0,001]$ . Przyjęto okres próbkowania  $T_c = 0,005 \text{ s}$ , ale zbadano również sterowanie z innymi wartościami. Przetestowano różne horyzonty predykcji, dla  $T_c = 0,005 \text{ s}$  wybrano  $N = 18$ . Przyjęte wartości zapewniają, dla krótkich horyzontów sterowania  $N_u$ , kompromis między jakością regulacji, nakładem obliczeniowym i odpornością. Różnice w wartościach wag wynikają, w znacznym stopniu, z różnic w zakresach wartości zmiennych, co odpowiada skalowaniu zadania optymalizacji.

### 3.4. Wyniki sterowania z regulatorami MPC

Porównanie rezultatów sterowania przedstawionymi w pracy algorytmami MPC przedstawiono zbiorczo w Tabeli 1, w sposób zagregowany przez podanie wartości indeksu ISE (31), dla różnych horyzontów sterowania  $N_u$  oraz różnych zakłóceń. Symulacje przeprowadzono w środowisku MATLAB, wykorzystując do optymalizacji nieliniowej procedurę „fmincon”, a do QP procedurę „quadprog”. Do linearyzacji modelu zastosowano przybliżenie numeryczne pochodnych cząstkowych, ze względu na bardzo złożone wzory analityczne (toolbox „Symbolic Math”) – uzyskując wyniki praktycznie równoważne, niemające wpływu na symulację układu sterowania. Rozważono trzy przypadki: tylko z dynamiką niemodelowaną (tarcie suche, Coulomba), z dynamiką niemodelowaną i dodanym zakłóceniem zewnętrznym (Zu) lub parametrycznym (P1).

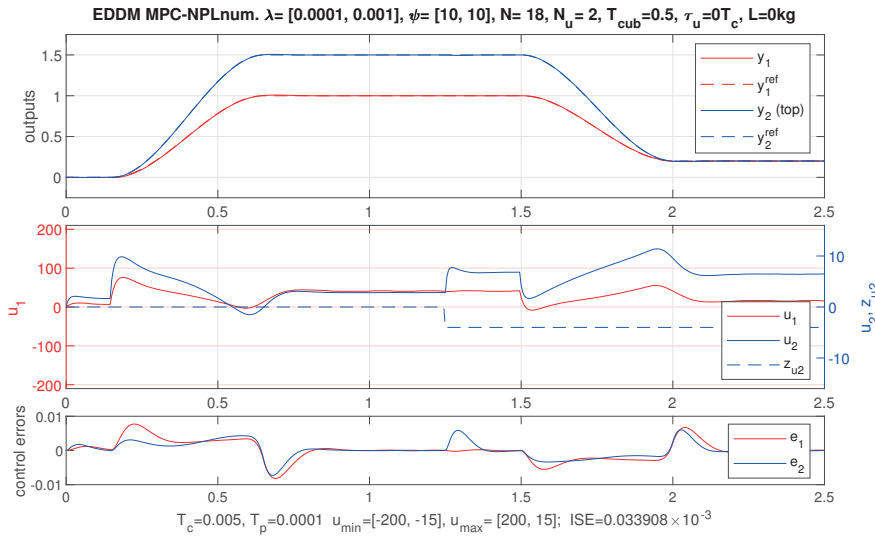
Horyzont sterowania w największym stopniu wpływa na nakład obliczeń, bo definiuje wymiar  $N_u \times n_u$  zadania optymalizacji, dla  $N_u = 1$  i  $N_u = 2$  odpowiednio 2 i 4. Wybór  $N_u = 2$  skutkuje wyraźnie lepszymi wartościami ISE, dla  $N_u = 3$  i  $N_u = 4$  poprawa jest już niewiele większa, przy zwiększonym obciążeniu obliczeniowym. Dla każdego z przypadków zakłócenia, wyniki dla wszystkich trzech algorytmów, NO, NPLn i NPLa, są praktycznie identyczne. Porównując MPC-NO i MPC-NPL, różnice występują na dalszych miejscach po przecinku, z reguły niewidocznych w Tabeli 1, trajektorie są wizualnie nie do odróżnienia. Wyniki dla NPLn i MPLa są dokładnie identyczne, ze względu na nieaktywne ograniczenia sterowania. Przykładowe trajektorie przedstawiono dla algorytmu MPC-NPLn z  $N_u = 2$ , na Rys. 3 i Rys. 4, dla zakłó-

Tab. 1. Wskaźnik ISE dla algorytmów MPC, dla różnych wartości  $N_u$  i zakłóceń  
Tab. 1. Values of ISE for MPC algorithms, for different values of  $N_u$  and disturbances

Zakłócenie	$N_u$	MPC-NO	MPC-NPLn	NPL-NPLa
Dynamika niemodelowana (tarcie suche)	1	$0,1120 \cdot 10^{-3}$	$0,1122 \cdot 10^{-3}$	$0,1122 \cdot 10^{-3}$
	2	$0,0321 \cdot 10^{-3}$	$0,0321 \cdot 10^{-3}$	$0,0321 \cdot 10^{-3}$
	3	$0,0234 \cdot 10^{-3}$	$0,0234 \cdot 10^{-3}$	$0,0234 \cdot 10^{-3}$
	4	$0,0230 \cdot 10^{-3}$	$0,0230 \cdot 10^{-3}$	$0,0230 \cdot 10^{-3}$
Dynamika niemodelowana + zewnętrzne Zu	1	$0,1120 \cdot 10^{-3}$	$0,1122 \cdot 10^{-3}$	$0,1122 \cdot 10^{-3}$
	2	$0,0339 \cdot 10^{-3}$	$0,0339 \cdot 10^{-3}$	$0,0339 \cdot 10^{-3}$
	3	$0,0250 \cdot 10^{-3}$	$0,0250 \cdot 10^{-3}$	$0,0250 \cdot 10^{-3}$
Dynamika niemodelowana + parametryczne P1	1	$0,0983 \cdot 10^{-3}$	$0,0983 \cdot 10^{-3}$	$0,0983 \cdot 10^{-3}$
	2	$0,0310 \cdot 10^{-3}$	$0,0310 \cdot 10^{-3}$	$0,0310 \cdot 10^{-3}$
	3	$0,0252 \cdot 10^{-3}$	$0,0252 \cdot 10^{-3}$	$0,0252 \cdot 10^{-3}$

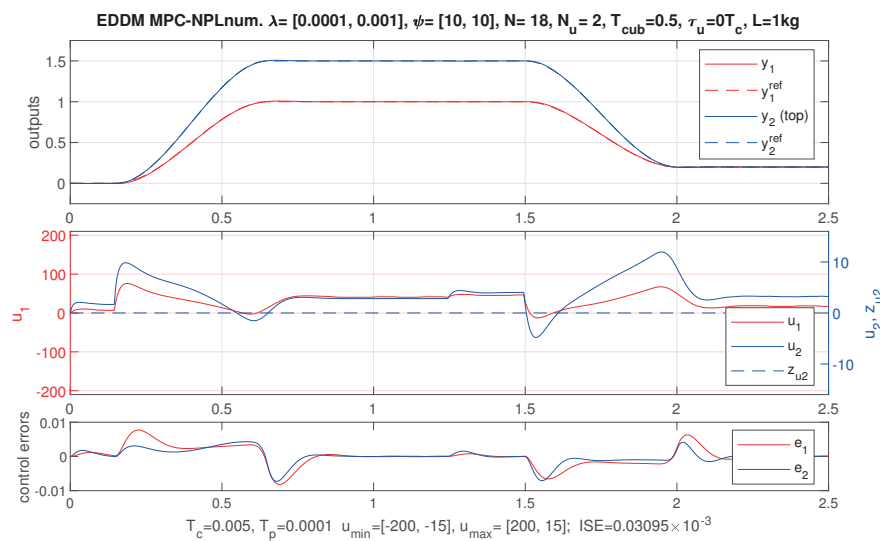
Rys. 3. Trajektorie dla MPC-NPLn z  $N_u = 2$  i zakłóceniem zewnętrznym

Fig. 3. Trajectories for MPC-NPLn with  $N_u = 2$  and external disturbance



Rys. 4. Trajektorie dla MPC-NPLn z  $N_u = 2$  i zakłóceniem parametrycznym

Fig. 4. Trajectories for MPC-NPLn with  $N_u = 2$  and parametric disturbance



czenia zewnętrznego  $Z_u$  i parametrycznego P1. Porównując dane w Tabeli 1 widać, że wyniki bez zakłóceń i z zakłóceniami bardzo niewiele się różnią, zastosowany w równaniach stanu algorytmów MPC model uwzględniania zakłóceń (9)–(10) jest bardzo efektywny.

Przedstawione wyniki pokazują, że dla rozważanego manipulatora prostsze algorytmy MPC-NPL mogą być stosowane bez utraty jakości sterowania, zamiast bardziej złożonego i mniej pewnego obliczeniowo algorytmu MPC-NO (optymalizacja nieliniowa jest bardziej zawodna niż optymalizacja kwadratowa).

### 3.5. Porównania ze sterowaniem algorytmem CTC-PID

Dla rzetelnej oceny wydajności, przydatność algorytmów MPC do sterowania ramionami manipulatorów, należy przeprowadzić porównanie ze stosowanymi algorytmami. Najczęściej spotykanym jest klasyczny algorytm PID [12, 22, 23]. Jest on szczególnie skuteczny, gdy silniki są wyposażone w przekładnie redukujące wzajemne interakcje między ramionami. W przypadku manipulatorów z napędem bezpośrednim zalecane są algorytmy oparte na modelach. Znany rozwiązaniem jest tu algorytm CTC-PID, linearyzujący dynamikę manipulatora nieliniowym sprzężeniem zwrotnym, z regulatorami PD/PID ze sprzężeniem wyprzedzającym (ang. *feedforward*) od przyspieszenia [12, 22, 23]. Porównanie MPC z tym algorytmem wydaje się trafne, ponieważ oba są nieliniowe i oparte na modelach. Przedstawimy teraz wyniki takiego porównania.

Przypomnijmy odpowiadający modelowi (1) algorytm CTC-PID [12, 23]:

$$u = M(q)(\ddot{q}^{zad} + u_{PID}) + C(q, \dot{q})\dot{q} + F(\dot{q}) + g(q), \quad (32)$$

gdzie  $u_{PID}$  jest wyjściem wielopętlowego regulatora PID. Wstawienie (32) wraz z równaniami regulatora PID do (1) skutkuje dynamiką pętli regulacji składającą się z  $n$  układów dynamicznych SISO trzeciego rzędu. Popularnym podejściem do doboru nastaw regulatora PID jest lokowanie biegunów (ang. *pole placement*) [12, 5]. Zakładając wartości biegunów równań charakterystycznych otrzymujemy wzory na wzmocnienia regulatorów PID. Popularnym wyborem jest przyjęcie potrójnego bieguna,  $-\alpha_j$  ( $\alpha_j > 0$ ), co umożliwi wygodne strojenie wszystkich wzmocnień jednym parametrem,  $k_{p_j} = 3\alpha_j^2$ ,  $k_{d_j} = 3\alpha_j$ ,  $k_{i_j} = \alpha_j^3$ ,  $j = 1, 2$ . Regulator PID implementowano jako dyskretny,

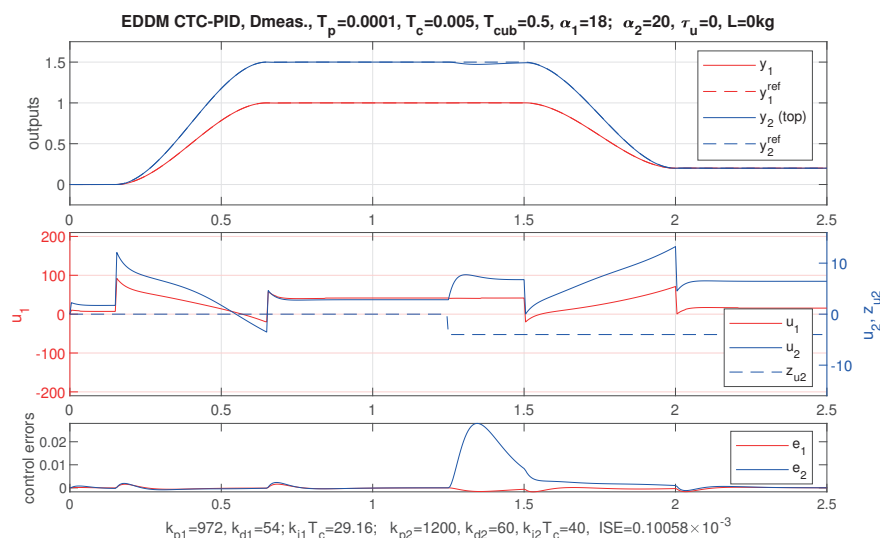
$$u_{i_j}(k) = u_{i_j}(k-1) + k_{i_j} T_c e_j(k), \\ u_{PID_j}(k) = k_{p_j} e_j(k) + k_{d_j} \dot{e}_j(k) + u_{i_j}(k), \quad j = 1, 2. \quad (33)$$

gdzie  $u_{i_j}(k)$  jest stanem  $j$ -tego integratora, a  $e_j(k)$  jest  $j$ -tym błędem regulacji,  $j = 1, 2$ .

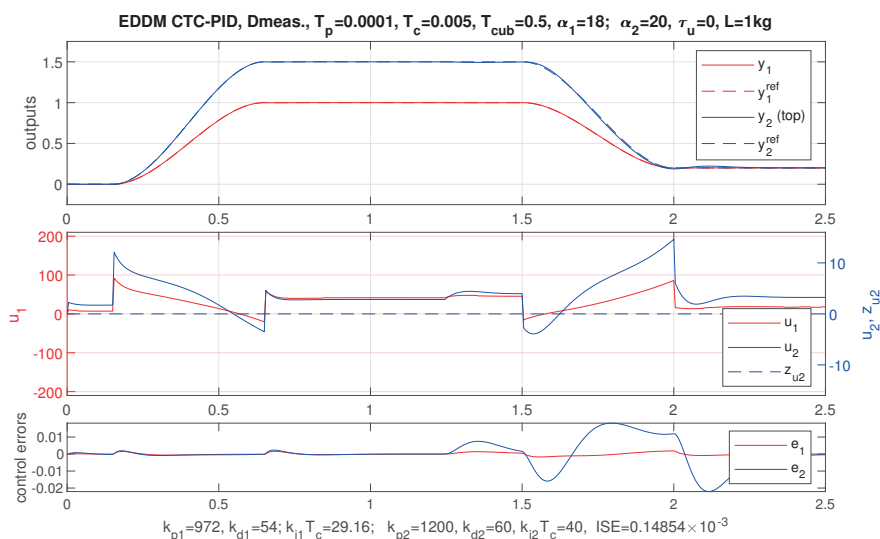
W Tabeli 2 przedstawiono wartości ISE uzyskane dla sterowania CTC-PID przy  $\alpha_1 = 18$ ,  $\alpha_2 = 20$ , co skutkuje szybką regulacją (duże wzmocnienia  $k_{p1} = 972$ ,  $k_{p2} = 1200$ ). Nato-

miast na Rys. 5 i Rys. 6 pokazano odpowiednie trajektorie, dla zakłócenia zewnętrznego i parametrycznego. Dla łatwości porównania, w Tabeli 2 umieszczono również wyniki uzyskane w analogicznych warunkach dla algorytmu MPC-NPL. Przy braku jakichkolwiek zakłóceń i błędów modelowania algorytm CTC-PID jest najlepszy (w sprzężeniu zwrotnym używa ideal-

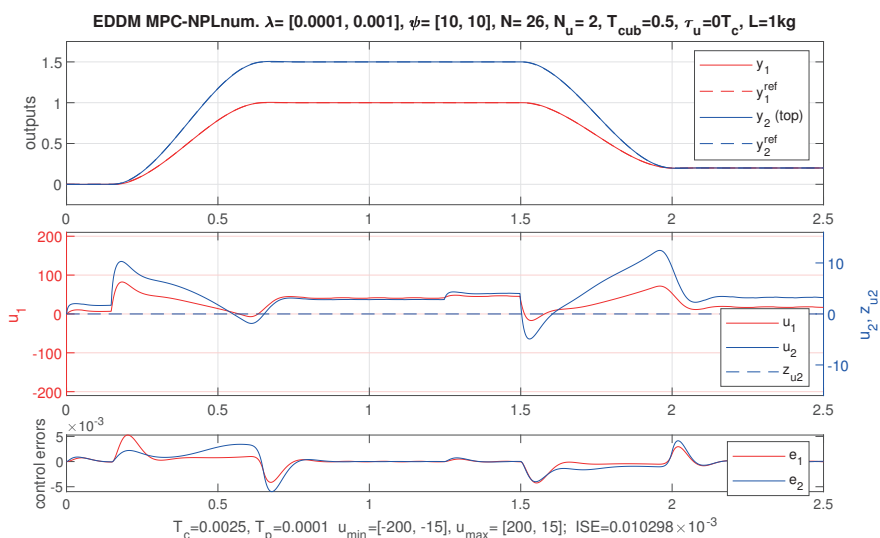
nego modelu, również z tarcie suchym, w MPC tarcie suche jest traktowane jako dynamika niemodelowana), ale jakość regulacji znacznie spada przy zakłóceniach. Natomiast algorytm MPC jest znacznie odporniejszy, znakomicie wytlumia zakłócenie zewnętrzne i błędy modelowania, bez utraty jakości sterowania, zapewniając w tych warunkach lepsze wartości



Rys. 5. Trajektorie dla CTC-PID z  $\alpha_1 = 18$ ,  $\alpha_2 = 20$  i zakłóceniem zewnętrznym  
Fig. 5. Trajectories for CTC-PID with  $\alpha_1 = 18$ ,  $\alpha_2 = 20$  and external disturbance



Rys. 6. Trajektorie dla CTC-PID z  $\alpha_1 = 18$ ,  $\alpha_2 = 20$  i zakłóceniem parametrycznym  
Fig. 6. Trajectories for CTC-PID with  $\alpha_1 = 18$ ,  $\alpha_2 = 20$  and parametric disturbance



Rys. 7. Trajektorie dla MPC-NPLn z  $T_c = 0,0025$  s i zakłóceniem parametrycznym  
Fig. 7. Trajectories for MPC-NPLn with  $T_c = 0.0025$  s and parametric disturbance



Tab. 2. Wartości ISE dla algorytmów CTC-PID i MPC-NPL, dla różnych zakłóceń  
 Tab. 2. Values of ISE for CTC-PID and MPC-NPL algorithms, for different disturbances

CTC-PID ( $\alpha_1 = 18, \alpha_2 = 20$ )		MPC-NPL ( $N = 18, N_u = 2$ )	
zakłócenie	ISE	zakłócenie	ISE
brak	$0,0015 \cdot 10^{-3}$	niemodelowana dynamika	$0,0321 \cdot 10^{-3}$
zewnętrzne (Zu)	$0,1006 \cdot 10^{-3}$	niemodelowana dynamika +Zu	$0,0339 \cdot 10^{-3}$
parametryczne (P1)	$0,1485 \cdot 10^{-3}$	niemodelowana dynamika +P1	$0,0310 \cdot 10^{-3}$

wskaźnika ISE. Porównując trajektorie na Rys. 5 i 6 z tymi na Rys. 3 i 4 dla regulatora MPC-NPL widzimy, że ten ostatni zapewnia również bardziej płynne trajektorie sterowania.

### 3.6. Wpływ okresu próbkowania i opóźnienia sterowania

Algorytmy oparte na modelach wykonują więcej obliczeń niż proste algorytmy PID, co może skutkować opóźnieniem w implementacji sygnałów sterujących. W związku z tym zbadano wpływ długości okresu próbkowania i opóźnienia sterowania na działanie rozpatrywanych algorytmów, w szczególności MPC-NPL i CTC-PID. Najistotniejsze praktycznie jest opóźnienie równe jednemu okresowi próbkowania, tzn. kiedy sterowanie obliczone na podstawie pomiarów uzyskanych na początku  $k$ -tego okresu jest wysyłane do elementów wykonawczych pod koniec tego okresu (lub na początku następnego).

Zaletą algorytmów MPC jest możliwość uwzględnienia opóźnienia sterowania w modelu manipulatora. Dla opóźnienia  $\tau_u = T_c$  powoduje to rozszerzenie stanu manipulatora do postaci  $x^T = [q^T \dot{q}^T z^T] \in R^6$  i równania stanu (3) do postaci

$$x(k+1) = \begin{bmatrix} q(k) + T_c \dot{q}(k) \\ \dot{q}(k) - T_c M(q(k))^{-1} [C(x(k))\dot{q}(k) + F(\dot{q}(k)) + g(q(k)) - z(k)] \\ u(k) \end{bmatrix}. \quad (34)$$

W Tabeli 3 zestawiono wybrane wyniki porównań, dla sytuacji z zakłóceniami zewnętrznym (Zu) i parametrycznym (P1), przez „nd” oznaczono niemodelowaną dynamikę, tzn. tarcie Coulomba traktowane jako zakłócenie w algorytmach MPC. Dla ułatwienia porównania podano również wartości ISE dla sytuacji bez opóźnień sterowania. Badano sterowanie z trzema okresami próbkowania:  $T_c = 0,01$  s,  $0,005$  s i  $0,002$  s. Wpływ opóźnienia sterowania  $\tau_u$  na działanie algorytmu MPC badano w dwóch wersjach: z opóźnieniem jako dodatkowym zakłóceniem oraz z opóźnieniem uwzględnionym w modelu użytym do projektowania regulatora. Dla algorytmu CTC-PID opóźnienie sterowania jest dodatkowym zakłóceniem. W algorytmach MPC musi być zachowana odpowiednia relacja (fizycznie) długości horyzontu predykcji  $N$  do szybkości dynamiki procesu. Dlatego dłuższe horyzonty predykcji  $N$ , podane w liczbach okresów próbkowania, odpowiadają krótszym wartościom  $T_c$ . Przyjęto wartości  $N$  równe 12, 18 i 26, kolejno dla malejących wartości  $T_c$ . Natomiast zastosowano  $N_u = 2$  we wszystkich przypadkach, jako efektywny obliczeniowo i regulacyjny horyzont sterowania.

Dane zawarte w Tabeli 3 pokazują, że dla algorytmu MPC zmniejszanie  $T_c$  poprawia wartości ISE. Zmniejszenie dwukrotne, z  $0,01$  s do  $0,005$  s, zmniejsza ISE nawet bardziej, to samo ma miejsce dla zmniejszenia z  $0,005$  s do  $0,0025$  s. Jeśli chodzi o wpływ opóźnienia sterowania, to dla mniejszych wartości  $T_c$ ,  $0,005$  s i  $0,0025$  s, wpływu tego praktycznie nie widać.

Natomiast dla  $T_c = 0,01$  wpływ negatywny jest wyraźnie widoczny. Opóźnienie traktowane jako zakłócenie spowodowało oscylacje sygnału sterującego  $u_2$ . Zwiększenie współczynnika wagowego  $\lambda_2$  do wartości  $0,01$  stłumiło te oscylacje, ale kosztem wzrostu ISE, i to do wartości większej niż w wariancie z opóźnieniem uwzględnionym w modelu (gdzie nie powoduje oscylacji). Stąd wniosek, że szczególnie przy dłuższych okresach próbkowania należy uwzględniać opóźnienie w modelu. Wraz ze spadkiem wartości ISE błędy regulacji maleją podobnie, przykładowym potwierdzeniem jest porównanie Rys. 7, pokazującego trajektorie dla  $T_c = 0,0025$  s, z Rys. 4.

Dla algorytmu CTC-PID, zmniejszenie  $T_c$  z  $0,01$  s do  $0,005$  s poprawiło wartości ISE, przy dalszym zmniejszeniu do  $T_c = 0,0025$  s zmiany są już niewielkie. Wynika to z faktu, że przy szybkim próbkowaniu strojenie dyskretnego regulatora PID zbliża się do strojenia regulatora ciągłego. Im szybsze próbkowanie, tym lepiej regulator dyskretny PID naśladuje regulator z czasem ciągłym i stąd mniejsze różnice przy zmniejszaniu okresu próbkowania. Natomiast wprowadzenie opóźnienia sterowania przy algorytmie CTC-PID zwiększa istotnie wartości ISE dla  $T_c = 0,01$  s, dla  $T_c = 0,005$  s i  $0,0025$  s jest to już wzrost mniejszy. Porównując wyniki dla obu algorytmów, przy  $T_c = 0,005$  s i  $0,0025$  s jakość regulacji algorytmem MPC jest znacznie lepsza.

### 3.7. Sterowanie przy skokowych zmianach wartości zadanych

Nawet w zadaniach swobodnego, szybkiego przechodzenia między ustalonymi położeniami ramion manipulatorów, z reguły nie jest zalecane zadawanie skokowych zmian wartości zadanych położenia, a więc kształtowanie trajektorii przejścia przez dynamikę pętli regulacji, zależną od nastaw regulatora. Albowiem bezpośrednio zastosowanie wielu algorytmów, w tym algorytmów wielopętlowych PID czy CTC-PID, prowadzi z reguły do przebiegów z silnymi przeregulowaniami i do wydłużenia czasów regulacji, z wchodzeniem w nasycenie sygnałów sterujących.

Interesującą cechą algorytmów MPC jest możliwość zastosowania wewnętrznego mechanizmu tzw. trajektorii referencyjnych, przekształcających zmiany skokowe wartości zadanych w odpowiednie łagodniejsze zmiany typu wykładniczego [17, 27]. Polega to na zastąpieniu pierwszej sumy w funkcji celu  $J(k)$  zadania (7) sumą

$$J_1(k) = \sum_{p=1}^N \|y^{ref}(k+p|k) - y(k+p|k)\|_{\Psi}^2 \quad (35)$$

gdzie

$$y^{ref}(k+p|k) = \left[1 - (T_{ref})^p\right] y^{zad}(k) + (T_{ref})^p y(k), \quad p = 1, \dots, N \quad (36)$$

zaś  $0 \leq T_{ref} < 1$  jest parametrem użytkownika. Dla  $T_{ref} = 0$ ,  $y^{ref}(k+p|k) = y^{zad}(k)$ , natomiast im większa jest wartość  $T_{ref} > 0$ , tym bardziej spowolniona jest zmiana  $y^{ref}(k+p|k)$

Tab. 3. Wartości ISE dla algorytmów MPC-NPL i CTC-PID, przy różnych  $T_c$  i  $\tau_u$   
 Tab. 3. Values of ISE for MPC-NPL and CTC-PID algorithms, with various  $T_c$  and  $\tau_u$

$T_c$	$\tau_u$	MPC-NPL		CTC-PID	
		zakłócenia	ISE	zakłócenia	ISE
0,01	0	nd+Zu	$0,1010 \cdot 10^{-3}$	Zu	$0,1742 \cdot 10^{-3}$
	$T_c$	nd+Zu+ $\tau_u$	$0,2700 \cdot 10^{-3(a)}$	Zu+ $\tau_u$	$0,2278 \cdot 10^{-3}$
	$T_c$	nd+Zu <sup>(b)</sup>	$0,1525 \cdot 10^{-3}$		
	0	nd+P1	$0,1033 \cdot 10^{-3}$	P1	$0,1561 \cdot 10^{-3}$
	$T_c$	nd+P1+ $\tau_u$	$0,2290 \cdot 10^{-3(a)}$	P1+ $\tau_u$	$0,2374 \cdot 10^{-3}$
	$T_c$	nd+P1 <sup>(b)</sup>	$0,1676 \cdot 10^{-3}$		
0,005	0	nd+Zu	$0,0339 \cdot 10^{-3}$	Zu	$0,1006 \cdot 10^{-3}$
	$T_c$	nd+Zu+ $\tau_u$	$0,0326 \cdot 10^{-3}$	Zu+ $\tau_u$	$0,1074 \cdot 10^{-3}$
	$T_c$	nd+Zu <sup>(b)</sup>	$0,0424 \cdot 10^{-3}$		
	0	nd+P1	$0,0310 \cdot 10^{-3}$	P1	$0,1485 \cdot 10^{-3}$
	$T_c$	nd+P1+ $\tau_u$	$0,0294 \cdot 10^{-3}$	P1+ $\tau_u$	$0,1651 \cdot 10^{-3}$
	$T_c$	nd+P1 <sup>(b)</sup>	$0,0426 \cdot 10^{-3}$		
0,0025	0	nd+Zu	$0,0113 \cdot 10^{-3}$	Zu	$0,1004 \cdot 10^{-3}$
	$T_c$	nd+Zu+ $\tau_u$	$0,0111 \cdot 10^{-3}$	Zu+ $\tau_u$	$0,1021 \cdot 10^{-3}$
	$T_c$	nd+Zu <sup>(b)</sup>	$0,0127 \cdot 10^{-3}$		
	0	nd+P1	$0,0103 \cdot 10^{-3}$	P1	$0,1467 \cdot 10^{-3}$
	$T_c$	nd+P1+ $\tau_u$	$0,0100 \cdot 10^{-3}$	P1+ $\tau_u$	$0,1521 \cdot 10^{-3}$
	$T_c$	nd+P1 <sup>(b)</sup>	$0,0121 \cdot 10^{-3}$		

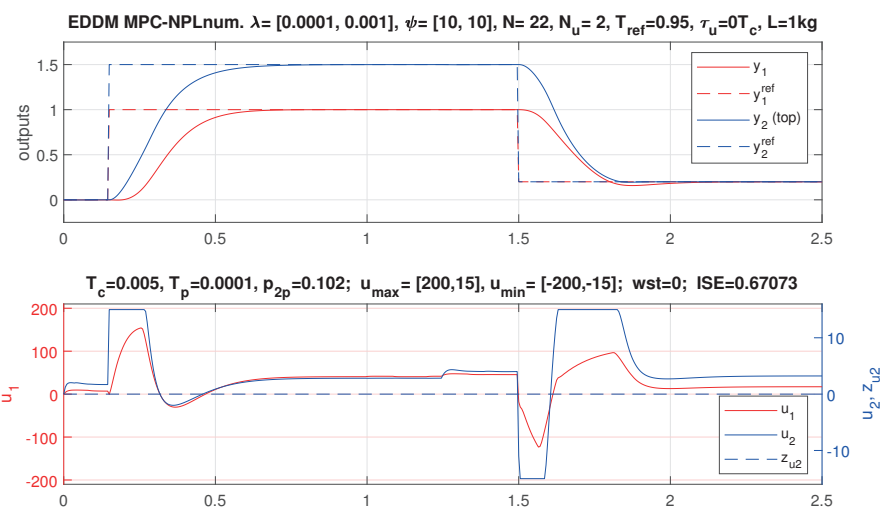
(a) wartość ISE dla wartości  $\lambda_2$  zwiększonej do 0,01 dla stłumienia oscylacji sygnału sterującego  
 (b) opóźnienie  $\tau_u$  uwzględnione w modelu manipulatora użytym do projektowania regulatora

w kierunku  $y^{ad}(k)$  na horyzoncie predykcji. Na Rys. 8 przedstawiono przykładowe przebiegi trajektorii uzyskane przy sterowaniu manipulatora algorytmem MPC-NPLn z  $T_{ref} = 0,95$ , przy odcinkami stałych, skokowo zmiennych wartościach zadanych, z zakłóceniem parametrycznym P1 (od chwili 1,25 s). Jak widać z porównania z przedstawionymi na Rys. 4 przebiegami uzyskanymi dla trajektorii zadanych kubicznych, zmiany położenia ramion następują też w czasie ok. 0,5 s i są równie płynne, podobnej jakości. Natomiast przebiegi momentów sterujących są zupełnie inne, dla drugiego silnika z okresami nasycenia. Projektowanie i analiza sterowania manipulatorami dla okresami stałych, skokowo zmiennych wartości zadanych położenia wykracza poza zakres niniejszej pracy, zamiarem było tu jedynie zasygnalizowanie potencjału algorytmów MPC w tym zakresie.

### 4. Podsumowanie

W pracy przedstawiono zastosowanie algorytmów MPC z nieliniowymi modelami przestrzeni stanu do sterowania manipulatorami, z wykorzystaniem najbardziej aktualnej techniki tłumienia zakłóceń [26, 28]. Pozwala ona uniknąć dynamicznego modelowania zakłóceń, zarówno zewnętrznych, jak i wewnętrznych, czy też stosowania innych, dodatkowych technik tłumienia zakłóceń, takich jak SMC (ang. *Sliding Mode Control*), spotykanych w artykułach dotyczących sterowania MPC ramion manipulatorów.

Przedstawiono zwięźle algorytm MPC z nieliniową optymalizacją (algorytm MPC-NO), którego bezpośrednie zastosowanie do sterowania manipulatorami może być trudne, ze względu na nakład obliczeń on-line w każdym okresie próbkowania. Algorytm ten jest traktowany jako punkt odniesienia dla bardziej efektywnych obliczeniowo, ale generalnie suboptymalnych algorytmów MPC-NPL, które stanowią podstawową treść artykułu. Technika NPL, z nieliniową predykcją i linearyzacją, dzieli proces obliczeniowy on-line na dwie części. W pierwszej przeprowadza się predykcję trajektorii początkowych z wykorzystaniem pełnego modelu nieliniowego, natomiast w drugiej trajektorie są poprawiane przez optymalizację z przyrostowym modelem zlinearyzowanym w aktualnym stanie manipulatora, co prowa-



Rys. 8. Trajektorie dla MPC-NPLn z trajektorią referencyjną  $T_{ref} = 0,95$ , przy skokowych zmianach stałych wartości zadanych, z zakłóceniem parametrycznym od 1,25 s  
 Fig. 8. Trajectories for MPC-NPLn with reference trajectory  $T_{ref} = 0.95$ , for step changes of constant set-points, with parametric disturbance from 1.25 s

dzi do szybkiej optymalizacji liniowo-kwadratowej. Linearyzację można spotkać w poprzednich próbach stosowania algorytmów MPC, co zostało pokrótce omówione we wstępie. Model liniowy był wówczas używany zarówno do wszystkich predykcji, jak i do optymalizacji. Jest to różnica istotna, albowiem z doświadczenia wynika, że predykcja nieliniowa odgrywa kluczową rolę. Przy tym nie zwiększa znacząco obciążenia obliczeniowego, ponieważ jest wykonywana tylko raz w okresie próbkowania.

Algorytmy MPC-NPL zostały przedstawione w artykule w dwóch wersjach: pierwsza (zwana numeryczną) jest z optymalizacją kwadratową (QP) z ograniczeniami, druga (zwana analityczną), z najmniejszym nakładem obliczeń, jest z jawną (analityczną) optymalizacją bez ograniczeń i następnie z dopasowaniem (przycięciem) sterowań do ograniczeń nierównościowych. Zarówno dla algorytmu MPC-NPL, jak i algorytmu MPC-NO przedstawiono wyniki obszernych badań symulacyjnych z eksperymentalnym manipulatorem z napędami bezpośrednimi. Przedstawione wyniki pokazały, że prostsze obliczeniowo algorytmy NPC-NPL realizują nadążanie za gładkimi trajektoriami zadanymi z praktycznie taką samą dokładnością jak algorytm MPC-NO. Wyniki porównano z uzyskanymi dla znanego algorytmu CTC-PID, będącego również „model-based” z wykorzystaniem nieliniowego modelu manipulatora. Z porównań wynika, że przedstawiony w pracy algorytm MPC jest konkurencyjną alternatywą dla CTC-PID, wymagającą nieco więcej obliczeń, ale zapewniającą lepszą jakość sterowania i większą odporność, dla obu rodzajów rozpatrywanych zakłóceń, zewnętrznego i parametrycznego. Zasygnalizowano też potencjalne możliwości skutecznego zastosowania algorytmów MPC do sterowania manipulatorów przy odcinkami stałych, skokowo zmiennych trajektoriach wartości zadanych.

Zbadano również wpływ długości okresu próbkowania i opóźnienia obliczeniowego sterowania na działanie algorytmów MPC i CTC-PID. Jest to ważne przy projektowaniu algorytmów sterowania opartych na modelach i przy szybkim próbkowaniu, ponieważ większy nakład obliczeń może skutkować opóźnieniem implementacji sterowania.

Wyniki symulacji przedstawione w artykule są fragmentem badań autora prowadzonych nad sterowaniem manipulatorów algorytmami MPC. Ograniczona objętość pracy nie pozwoliła na prezentację szerszych wyników badań, ale konkluzje dla innych manipulatorów były analogiczne, uwiarygodniając wnioski przedstawione w pracy. Autor ma nadzieję, że przedstawione rezultaty będą interesujące zarówno dla badań stosowanych, jak i praktyki przemysłowej.

## Bibliografia

1. Bemmiran A., *Predictive optimizing reference governor for constrained 2 dof's robot with abrupt set-point trajectories*. „Journal of Applied Science”, Sabratha University, No. 1, 2018, 39–49, DOI: 10.47891/sabujas.v1i1.39-49.
2. Blevins T.L., McMillan G.K., Wojsznis W.K., Brown M.W., *Advanced Control Unleashed*. The ISA Society, Research Triangle Park, NC, 2003.
3. Blevins T.L., Wojsznis W.K., Nixon M., *Advanced Control Foundation: Tools, Techniques, and Applications*. The ISA Society, Research Triangle Park, NC, 2013.
4. Bouzoualegh S., Guechi E., Zennir Y., *Model predictive control of a three degrees of freedom manipulator robot*. [In:] Proceedings of the 3rd International Conference on Advanced Systems and Emergent Technologies, Hammamet, Tunisia, 2019, 84–89, DOI: 10.1109/ASET.2019.8870999.
5. Bożek A., Trybus L., *Tuning PID and PI-PI servo controllers by multiple pole placement*. „Bulletin of the Polish Academy of Sciences Technical Sciences”, Vol. 70, No. 1, 2022, 1–12, DOI: 10.24425/bpasts.2021.139957.
6. Bumroongsri P., Kheawhom S., *Interpolation-based off-line MPC for LPV systems*. [In:] Proceedings of the 10th IFAC International Symposium on Dynamics and Control of Process Systems, Mumbai, India, 2013.
7. Camacho E.F., Bordons C., *Model Predictive Control*. Springer Verlag, London, 1999.
8. Cisneros P.S.G., Sridharan A., Werner H., *Constrained predictive control of a robotic manipulator using quasi-LPV representations*. „IFAC-PapersOnLine”, Vol. 51, No. 26, 2018, 118–123, DOI: 10.1016/j.ifacol.2018.11.158.
9. Ferrara A., Incremona G.P., Magni L., *A robust MPC/ISM hierarchical multiloop control scheme for robot manipulators*. [In:] Proceedings of the 52nd Conference on Decision and Control, Florence, Italy, 2013, DOI: 10.1109/CDC.2013.6760430.
10. Lewis F.L., Dawson D.M., Abdallah C.T., *Robot Manipulator Control Theory and Practice*. CRC Press, Boca Raton, 2003.
11. Incremona G.P., Ferrara A., Magni L., *MPC for robot manipulators with integral sliding modes generation*. „IEEE/ASME Transactions on Mechatronics”, Vol. 22, No. 3, 2017, 1299–1307, DOI: 10.1109/TMECH.2017.2674701.
12. Kelly R., Santibáñez V., Loría A., *Control of Robot Manipulators in Joint Space*. Springer, London 2005, DOI: 10.1007/b135572.
13. Khan M.F., Islam R., Iqbal J., *Control strategies for robotic manipulators*. [In:] Proceedings of the 2012 International Conference of Robotics and Artificial Intelligence, Rawalpindi, Pakistan, DOI: 10.1109/ICRAI.2012.6413422.
14. Kleff S., Meduri A., Budhiraja R., Mansard N., Righetti L., *High-frequency nonlinear model predictive control of a manipulator*. [In:] Proceedings of the IEEE International Conference on Robotics and Automation, Xi'an, China, 2021, DOI: 10.1109/ICRA48506.2021.9560990.
15. Ławryńczuk M., *Computationally Efficient Model Predictive Control Algorithms: A Neural Network Approach*. „Studies in Systems, Decision and Control”, Vol. 3. Springer Verlag, Heidelberg 2014, DOI: 10.1007/978-3-319-04229-9.
16. Ławryńczuk M., Tatjewski P., *Offset-free state-space nonlinear predictive control for Wiener systems*. „Information Sciences”, Vol. 511, 2020, 127–151, DOI: 10.1016/j.ins.2019.09.042.
17. Maciejowski J.M., *Predictive Control with Constraints*. Prentice Hall, Harlow, England, 2002.
18. Nicolis D., Allevi F., Rocco P., *Operational space model predictive sliding mode control for redundant manipulators*. „IEEE Transactions on Robotics”, Vol. 36, No. 4, 2020, 1348–1355, DOI: 10.1109/TRO.2020.2974092.
19. Qin S.J., Badgwell T.A., *A survey of industrial model predictive control technology*. „Control Engineering Practice”, Vol. 11, No. 7, 2003, 733–764, DOI: 10.1016/S0967-0661(02)00186-7.
20. Reyes F., Kelly R., *Experimental evaluation of identification schemes on a direct drive robot*. „Robotica”, Vol. 15, No. 5, 1997, 563–571, DOI: 10.1017/S0263574797000659.
21. Rybus T., Seweryn K., Sasiadek J.Z., *Application of predictive control for manipulator mounted on a satellite*. „Archives of Control Sciences”, Vol. 28, No. 1, 2018, 105–118, DOI: 10.24425/119079.
22. Spong M., Hutchinson S., Vidyasagar M., *Robot Modeling and Control*. J. Wiley and Sons, 2005.
23. Spong M.W., *An historical perspective on the control of robotic manipulators*. „Annual Reviews of Control, Robotics, and Autonomous Systems”, Vol. 5, 2022, 1–31, DOI: 10.1146/annurev-control-042920-094829.
24. Tatjewski P., *Advanced Control of Industrial Processes. Structures and Algorithms*. Springer Verlag, London 2007, DOI: 10.1007/978-1-84628-635-3.

25. Tatjewski P., *Disturbance modeling and state estimation for offset-free predictive control with state-spaced process models*. „International Journal of Applied Mathematics and Computer Science”, Vol. 24, No. 2, 2014, 313–323.
26. Tatjewski P., *Offset-free nonlinear predictive control with measured state and unknown asymptotically constant disturbances*. [In:] Malinowski K., Józefczyk J., Świątek J. (eds), „Aktualne problemy automatyki i robotyki”, 288–299, Academic Publisher EXIT, Warszawa 2014.
27. Tatjewski P., *Sterowanie zaawansowane procesów przemysłowych. Struktury i algorytmy*. Wydanie drugie, zmienione (e-book). Akademicka Oficyna Wydawnicza EXIT, Warszawa 2016.
28. Tatjewski P., *Offset-free nonlinear Model Predictive Control with state-space process models*. „Archives of Control Sciences”, Vol. 27, No. 4, 2017, 595–615, DOI: 10.1515/acsc-2017-0035.
29. Tatjewski P., Ławryńczuk M., *Algorithms with state estimation in linear and nonlinear model predictive control*. „Computers and Chemical Engineering”, Vol. 143, 2020, DOI: 10.1016/j.compchemeng.2020.107065.
30. Wilson J., Charest M., Dubay R., *Non-linear model predictive control schemes with application on a 2 link vertical robot manipulator*. „Robotics and Computer-Integrated Manufacturing”, Vol. 41, 2016, 23–30, DOI: 10.1016/j.rcim.2016.02.003.

## Nonlinear Predictive Control of Manipulator Arms

**Abstract:** The subject of the article are predictive control algorithms (of MPC type – Model Predictive Control) for rigid manipulator arms. MPC with a state-space model and with the latest disturbance and modeling error suppression technique was applied, which avoids dynamic disturbance modeling or resorting to additional disturbance cancellation techniques, such as SMC. First of all, the most computationally efficient MPC-NPL (Nonlinear Prediction and Linearization) algorithms are considered, in two versions: the first with constrained QP (Quadratic Programming) optimization and the second with explicit (analytical) optimization without constraints and satisfying a posteriori inequality constraints. For all considered algorithms, a comprehensive simulation analysis was carried out for a direct drive manipulator, with two kinds of disturbances: external and parametric. The obtained results were compared with those for the well-known CTC-PID algorithm (CTC – Computer Torque Control), showing better control quality with MPC algorithms. In addition, the influence of the length of the sampling period and of the computational delay on control quality was investigated, which is important for model-based algorithms with fast sampling.

**Keywords:** manipulator control, nonlinear control, model predictive control, control delay, fast sampling

### Prof. dr hab. inż. Piotr Tatjewski

piotr.tatjewski@pw.edu.pl  
ORCID: 0000-0002-2596-8937



W 1972 r. ukończył studia magisterskie na kierunku elektronika na Politechnice Warszawskiej (PW). W latach 1976 i 1988 uzyskał stopnie doktora i doktora habilitowanego nauk technicznych w zakresie automatyki i robotyki. Tytuł profesora nauk technicznych uzyskał w 2003 r, od 2006 r. profesor zwyczajny na PW. Pracuje w Instytucie Automatyki i Informatyki Stosowanej PW od 1972 r., w latach 1996–2008 dyrektor tego instytutu. W latach 2012–2020 prodziekan ds. nauki Wydziału Elektroniki i Technik Informatycznych PW. Autor lub współautor ponad 80 artykułów w czasopismach i rozdziałów w monografiach, autor 8 książek, w tym 4 w języku angielskim (w tym "Advanced Control of Industrial Processes", Springer, 2007; "Iterative Algorithms of Multilayer Optimizing Control", Imperial College Press/World Scientific, 2005, współautor M.A. Brdys). Członek Komitetu Automatyki i Robotyki PAN od 2003 r., członek Centralnej Komisji ds. Stopni i Tytułów w kadencji 2017–2020. Zainteresowania naukowe obejmują sterowanie wielowarstwowe, sterowanie predykcyjne procesów liniowych i nieliniowych, „soft computing” w modelowaniu i sterowaniu, optymalizację punktów pracy, metody hierarchiczne w optymalizacji i sterowaniu.