

Arkadiusz GARDECKI

DOŚWIADCZENIA Z EKSPLOATACJI HUMANOIDALNYCH ROBOTÓW NAO

STRESZCZENIE *W artykule przedstawiono kilkuletnie doświadczenia z eksploatacji dwóch generacji autonomicznych humanoidalnych robotów NAO. Omówiono ważniejsze czujniki zastosowane w tej konstrukcji, które odpowiedzialne są za interakcje człowieka z robotem. Omówiono także możliwości i ograniczenia tego robota oraz spostrzeżenia wynikające z doświadczenia z pracy z dedykowanym oprogramowaniem: Choregraphe, Monitor oraz Webots for NAO. Zostały również omówione ważniejsze zmiany w konstrukcji robota oraz przykładowe zastosowania tej platformy robotycznej.*

Słowa kluczowe: robotyka, robot humanoidalny NAO

1. WSTĘP

Blisko czteroletnie doświadczenia z eksploatacji robotów humanoidalnych NAO skłoniły autora do próby podsumowania tego okresu.

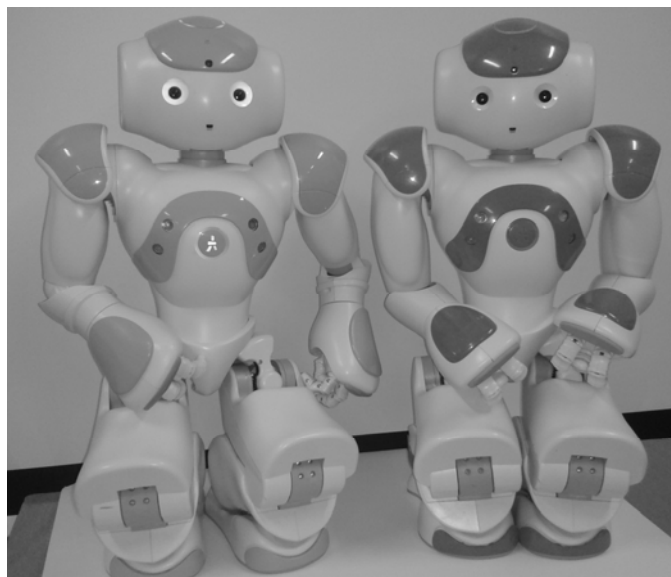
Robot NAO jest autonomicznym, mobilnym, humanoidalnym robotem zaprojektowanym przez francuską firmę Aldebaran Robotics [1]. Robot ten, opracowywany od 2004 roku, został zaferowany uczelniom oraz instytucjom edukacyjnym w 2008 roku. Główne obszary zastosowań tej konstrukcji to szeroko rozumiana edukacja oraz wspomagane przez producenta zastosowanie robota w terapii dzieci autystycznych [2, 3]. Istotną zaletą tej konstrukcji jest jej pełna programowalność. Producent dostarcza często aktualizowaną dokumentację oraz dedykowane środowisko programistyczne Choregraphe [4] i Webots for NAO [5].

W 2011 roku Instytut Układów Elektromechanicznych i Elektroniki Przemysłowej Politechniki Opolskiej zakupił pierwszego robota NAO w wersji V3.3, a w 2013 roku kolejnego robota w wersji V4. W wyniku mechanicznego uszkodzenia

dr inż. Arkadiusz GARDECKI
e-mail: a.gardecki@po.opole.pl

Politechnika Opolska, Instytut Układów Elektromechanicznych i Elektroniki Przemysłowej,
ul. Prószkowska 76 bud.1, 45-758 Opole

PRACE INSTYTUTU ELEKTROTECHNIKI, zeszyt 270, 2015



Rys. 1. Roboty humanoidalne NAO V3.3 (z głową w wersji V4) i V4

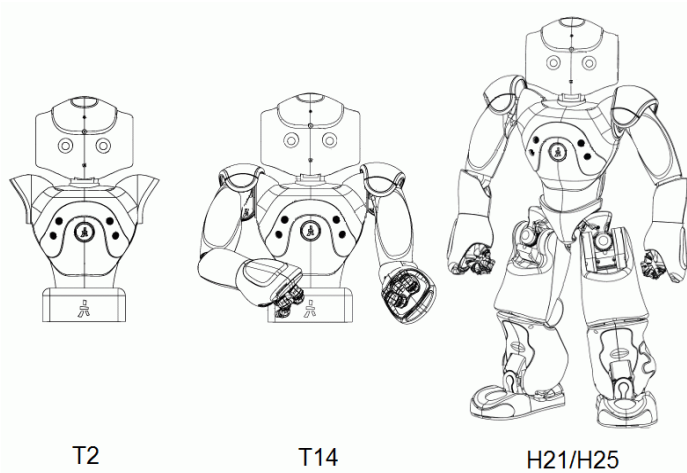
zębátky w przekładni łączy głowy zdecydowano się na wymianę całej głowy robota w wersji V3.3, na głowę robota w wersji V4. Działanie to było także podyktowane potrzebą zachowania kompatybilności oprogramowania wewnętrznego robotów (w głowie robota znajduje się główna jednostka obliczeniowa, pamięci, kamery, moduł WiFi itp).

2. SPECYFIKACJA I BUDOWA ROBOTA NAO

Roboty NAO oferowane były w kilku wersjach: torsu z głową T2, torsu z głową i kończynami górnymi T14, kompletnego robota z ograniczoną funkcjonalnością ręki H21, oraz pełna wersja H25. Liczby przy oznaczeniu typu oznaczają liczbę stopni swobody konstrukcji.

Robot kontrolowany jest przez własny system operacyjny NAOqi. Zawiera on szereg modułów, tworzących bibliotekę [7], umożliwiających kontrolę zasobów robota.

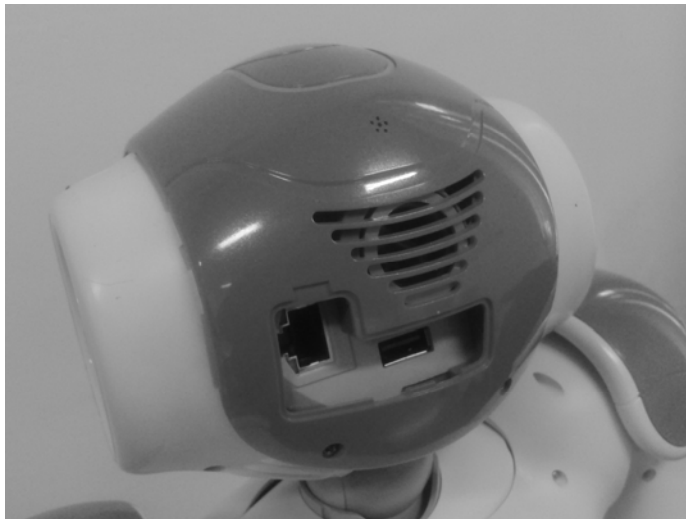
Ruchome łączy zasilane są za pomocą elektrycznych serwonapędów z wysokoobrotowymi silnikami DC, których moc i momenty dopasowane są do charakterystyki łączy. łączy są wyposażone w układ 12 bitowych aktuatorów, wykorzystujących efekt Halla, dzięki którym mamy zawsze informację o aktualnej pozycji robota. Ograniczeniem tej konstrukcji są zużywające się koła zębate przekładni wykonane z materiałów nieżelaznych. Szczególnie wrażliwe na uszkodzenia jest łączy szyi robota ze względu na przenoszone duże obciążania, np. podczas upadku robota. Wymiana takiej części jest kosztowna i wiąże się z koniecznością oddania robota do serwisu. Wraz ze wzrostem temperatury łączy, co następuje już po około godzinie



Rys. 2. Oferowane wersje robotów NAO [6]

pracy, odczuwalny jest spadek sztywności konstrukcji. Wiąże się to z większym prawdopodobieństwem upadku lub nieprawidłowym wykonaniu sekwencji ruchu. Rozwiązaniem tego problemu może być dodatkowe wymuszone chłodzenie (np. przenośny wentylator) lub przerwa w pracy.

Problem chłodzenia pojawia się także w przypadku dłuższej pracy robota z wykorzystaniem komunikacji przez moduł WiFi (szczególnie w gorące dni). Wynika to z umiejscowienia procesora i tego modułu w niewielkiej przestrzeni wewnątrz głowy robota (rys. 3). Przy dłuższej pracy producent zaleca połączenie kablowe Ethernet zamiast komunikacji WiFi. Temperatura wewnętrzna robota jest stale monitorowana.



Rys. 3. Lokalizacja portu Ethernet i USB oraz wylot chłodzenia w tylnej części głowy (ver4) [6]

Wzrost temperatury grożący przegrzaniem jest zgłaszany poprzez komunikaty informujące o aktualnej podwyższonej temperaturze, aż do wyłączenia się awaryjnego włączanie.

W tabeli poniżej zestawiono ważniejsze parametry wersji V3.x, V4 i V5 robotów NAO.

TABELA 1

Specyfikacja wybranych modeli robotów NAO [6]

V3.x	V4	V5
Procesor X86 AMD Geode 500 MHz CPU	Procesor ATOM Z530 1.6 GHz CPU	Procesor ATOM Z530 1.6 GHz CPU
256 MB SDRAM	1 GB RAM	1 GB RAM
WiFi 802.11 b/g/n	WiFi 802.11 b/g/n	WiFi 802.11 a/b/g/n
2 kamery VGA@30fps (YUV422)	2 kamery 960p@30fps (YUV422 color space)	
Czujniki ultradźwiękowe (2 nadajniki, 2 odbiorniki)		
Czujnik naciski FSR, 8 sztuk – zakres 0-25 N		
2 głośniki i 4 mikrofony		
Akcelerometr 1 osiowy Żyroskop 2 osiowy	Akcelerometr 3 os. Żyroskop 3 osiowy	
Baterie litowo-jonowe (Li-Ion) 2.25 Ah		
Waga: 4,8 kg, wysokość: 58 cm		

Roboty różnych wersji, choć z pozoru z zewnątrz bardzo podobne, mogą różnić się wymiarami na przykład długością nóg i/lub ramion. Może to spowodować, że program działający poprawnie na jednej wersji robota może spowodować nawet upadek robota w innej wersji. Poza tą przyczyną upadki robota zdarzają się dosyć często, mimo bardzo ostrożnego obchodzenia się z nim. Przyczyn może być wiele np.: chwilowe nieprawidłowe działanie akcelerometru lub żyroskopu – powoduje nieuzasadnione wykrycie upadku, zatrzymanie lub zawieszenie programu, zerwanie komunikacji z robotem itp. Mimo to konstrukcja robota jest dosyć odporna i zaprojektowana z uwzględnieniem możliwości takich zdarzeń. Należy tylko szczególnie dbać aby robot nie spadł z wysokości np. stolika, podestu. Wykonana z odpornego tworzywa (ABS-PC) obudowa jest odporna na zarysowania i pęknięcia. Przy testach i badaniach gdzie robot może upaść (np. wchodzenie na schody itp.) warto zadbać o odpowiednie amortyzujące podłoże lub podwieszać robota na wyciągu. Podłoże po którym porusza się robot powinno być możliwie gładkie ale nie śliskie ani nie stawiające zbyt dużego oporu. Z doświadczeń poruszania się po różnych nawierzchniach dobrym wyborem wydaje się gładka wykładzina przemysłowa.

Czas ładowania baterii wynosi około 1,5 h a w pełni naładowana bateria wystarcza na około 40÷60 minut pracy w zależności od intensywności zadań. Robot także może być ładowany w trakcie pracy co znacząco wydłużaj czas jego aktywności.

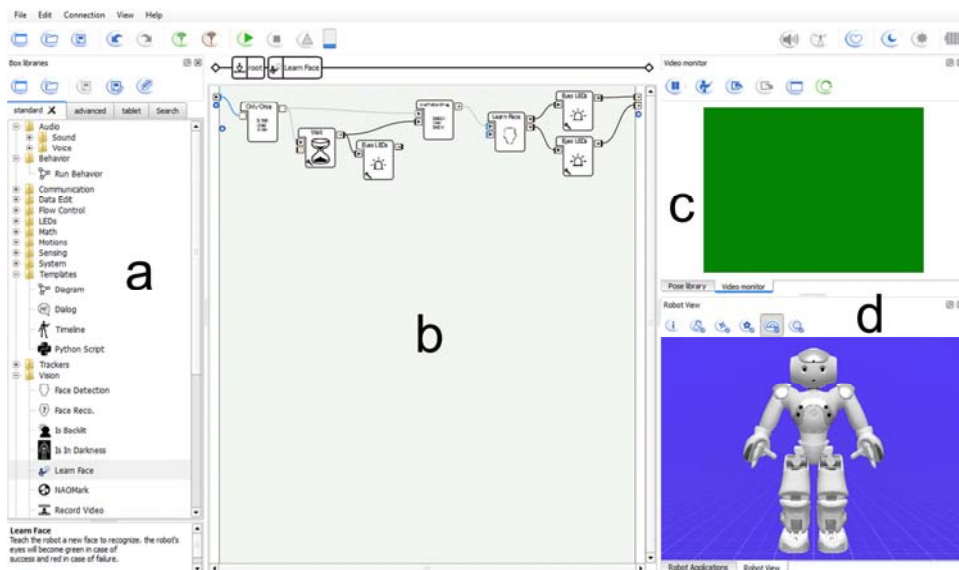
3. PROGRAMOWANIE I STEROWANIE ROBOTA NAO

Dedykowane oprogramowanie składa się z programu Choregraphe (rys. 4), Monitor oraz dodatku Webots for NAO (rys. 6).

Środowisko Choregraphe umożliwia budowę oraz dowolne modyfikacje już istniejących obiektów pogrupowanych w tematycznych bibliotekach. Dzięki nim łatwo i intuicyjnie można budować aplikację o różnym stopniu skomplikowania. Posługując się obiektami graficznymi nawet dzieci są w stanie zbudować prosty program. Obiekty te mają ukrytą warstwę reprezentowaną przez kod w języku Python (rys. 5). Na tym poziomie mamy dostęp do modułów i zmiennych reprezentujących stany i sygnały z czujników oraz do wbudowanych algorytmów sterujących robotem.

Program Choregraphe ma wiele możliwości dostosowania go do aktualnego zadania – kontrola wykonywania kodu, debugging, uczenie robota, projektowanie sekwencji ruchowej (bloki motion) i wiele innych. Obserwując rozwój tego oprogramowania można dostrzec wyraźny postęp w poprawie jego niezawodności i funkcjonalności. Zdarzają się sporadycznie problemy z utratą komunikacji z NAOqi OS, problem z komunikacją z rzeczywistymi robotami oraz niestabilności przy współpracy z Webots for NAO ale można uznać to za poziom akceptowalny.

Program Monitor umożliwia podgląd i rejestrację najistotniejszych parametrów robota. Dzięki temu możemy obserwować np. zmiany obciążenia poszczególnych złączy w trakcie ruchu. Działa on tylko podczas współpracy z rzeczywistym robotem i ustanowionej z nim komunikacji.



Rys. 4. Widok głównego okna programu Choregraphe 2.1.2.17

a – biblioteki obiektów, b – obszar budowy aplikacji,

c – widok z kamery robota, d – widok aktualnej pozycji robota

```

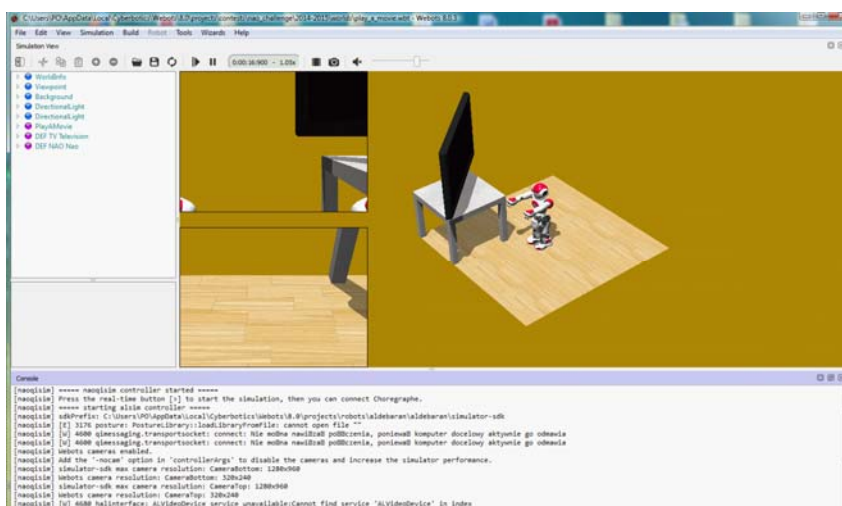
1 class MyClass(GeneratedClass):
2     def __init__(self):
3
4
5     def onLoad(self):
6         self.ok = [False]*2
7
8     def onUnload(self):
9
10
11
12     def onStart(self, nInput):
13         self.ok[nInput-1] = True
14         bOutput = True
15         for bOk in self.ok:
16             bOutput = bOutput and bOk
17         if( bOutput ):
18             self.ok = [False]*2
19             self.signalsReceived()
20
21     def onInput_signal1(self):
22         self.onStart(1)
23
24     def onInput_signal2(self):
25         self.onStart(2)

```



Rys. 5. Obiekt *Wait for Signals* i kod w języku Python opisujący działanie tego obiektu.

Program Webots for NAO (rys. 6) jest dedykowanym i dostarczanym przez Aldebaran wersji ograniczonym jedynie do symulacji robotów NAO środowiskiem 3D. Możemy w nim testować jego zachowanie w odwzorowanym wirtualnie środowisku. Dzięki dołączonym Bibliotekom mamy możliwość dodawania i edytowania różnych obiektów, przykładowych gotowych środowisk (pokój, boisko dla NAO) a także dodatkowych robotów NAO. Wszystkie te obiekty są bogato sparametryzowane dzięki temu mamy szerokie możliwości dopasowania ich do wymagań symulacji. Sterowanie symulacji odbywa się z poziomu NAOqi za pośrednictwem programu Choregraphe. Ustalamy połączenie z symulowanym robotem a następnie możemy przesłać do niego program podobnie jakbyśmy wysyłali program do rzeczywistego robota. Obsługa symulacji jest stosunkowo łatwa i intuicyjna. Najistotniejszym ograniczeniem tego programu są jego wymagania dotyczące mocy obliczeniowej komputera i karty graficznej. Jest to jednak naturalne w tego typu programach.



Rys. 6. Widok głównego okna programu Webots for NAO 8

Oprócz dedykowanego oprogramowania są podejmowane próby stworzenia lub wykorzystania istniejących środowisk do sterowania robotem NAO. Ułatwia to fakt, iż system zarządzający robotem NAOqi bazuje na systemie Linux (Embedded Linux 32bit x86 ELF) i także jest udostępniany na zasadach *open-source licence*. Do ciekawych propozycji należy Robot Operating System (ROS) i budowane w jego ramach aplikacje sterujące wybranymi funkcjonalnościami robota NAO [8]. Można także pisać aplikacje bezpośrednio w Pythonie z wykorzystaniem bibliotek NAOqi. Istnieje także możliwość zaimplementowania oprogramowania pisanego w innych językach np.: C++, Java, URBI czy MATLAB.

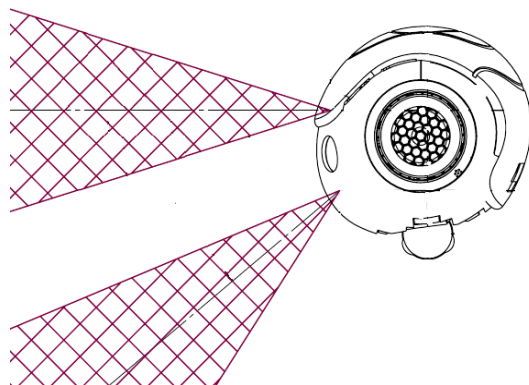
Podsumowując ten rozdział można powiedzieć, że środowisko programistyczne robota NAO jest spójną i przemyślaną koncepcją firmy Aldebaran (NAOqi jest także używane w innych robotach tej firmy np. Romeo, Pepper). Mamy możliwość kontroli sygnałów z czujników, przetwarzania tych informacji i sterowania serwonapędami. Taki pełny programowalny dostęp do konstrukcji wiąże się także z odpowiedzialnością. Poprzez nieprawidłowe sterowanie możemy uszkodzić robota. Producenci zadbali i o to, wiele krytycznych operacji jest dodatkowo zabezpieczonych programowo, jednak zabezpieczenia te można usunąć (np. Fall detection – moduł zabezpieczający konstrukcję przed konsekwencjami upadku – ustawia robota w pozycji potencjalnie korzystnej z punktu widzenia konsekwencji upadku i wyłącza zasilanie złączy). Aby ją skutecznie wyłączyć, na przykład gdy chcemy aby robot usiadł na krzeselku, trzeba zmienić ustawienia na kilku poziomach dostępu.

4. INTERAKCJA Z CZŁOWIEKIEM

Komunikacja z robotem może odbywać się na wiele sposobów. Możemy wykorzystać czujniki dotykowe, analizę obrazu z kamer lub rozpoznawanie mowy.

Robot posiada wiele czujników które reagują na dotyk. Zaliczamy do nich włącznik główny na środku korpusu, czujniki pojemnościowe na głowie, wyłączniki krańcowe z przodu stopy, czujniki na przedramieniu (są opcją w zależności od wersji robota) oraz czujniki nacisku na spodzie stóp. Najczęściej wykorzystywanymi czujnikami dotykowymi ze względu na łatwość dostępu do nich są trzy podświetlane, pojemnościowe czujniki dotykowe na górze głowy. Mogą one uzupełniać lub nawet zastępować komunikację werbalną z robotem. Ma to szczególnie znaczenie w środowisku w którym hałas lub odgłosy mogą zakłócać działanie modułu identyfikacji mowy. Działają one niezawodnie, ale ze względu na swoje położenie należy je programowo zabezpieczyć przed przypadkowym wyzwoleniem (np. dzieci intuicyjnie dotykają głowy robota).

Ciekawym mechanizmem, który możemy także interpretować jako czujnik dotyku jest wykorzystanie sygnałów z aktuatorów w złączach. Rejestrowana jest każda zmiana położenia złącza, a tym samym możemy użyć jej do wykrycia, np. zmiany położenia złącza ręki, spowodowanej ruchem podczas uścisk ręki robota. Takie wykorzystanie tych czujników umożliwia realizację ciekawych algorytmów interakcji człowieka i robota. Można także regulować stopień sztywności złączy i w połączeniu z informacjami z aktuatorów możemy przy pomocy takiego złącza sterować robotem



Rys. 7. Zasięg i umiejscowienie kamer robota NAO

(np. algorytm prowadzenia za rękę robota). Podobnie możemy wykorzystać informację o zmianie obciążenia złącza poprzez analizę zmian poboru prądu przez złącze [9].

Robot wyposażony jest we dwie przełączalne kamery (rys. 7), których obszar roboczy częściowo się pokrywa. Dzięki temu możemy zwiększyć obszar obserwacji robota, który w połączeniu z ruchomą głową jest całkiem akceptowalny. Obraz z kamery (analizowana jest zawsze tylko z jednej wybranej kamery) możemy wykorzystać jako punkt wyjścia dla autorskich algorytmów przetwarzania obrazu lub posłużyć się wbudowanym modulem. Gotowe moduły umożliwiają np. rozpoznawanie twarzy lub wybranych obiektów (mała piłka). Możemy także w trybie nauki przesłać do robota obrazy własnych obiektów z przyporządkowanymi im nazwami. Moduł rozpoznawania zwróci zmienną typu string z nazwą rozpoznanego obiektu. Robot NAO rozpoznaje także specjalne znaczniki NAOMark, dzięki którym możemy oznaczać przedmioty lub kodować pod nimi operacje, które ma wykonać robot. Wbudowane oprogramowanie działa z reguły poprawnie, ważne jest bardzo dobre oświetlenie obszaru pracy robota (równomierne, najlepiej bez oświetlenia punktowego, które oślepia kamerę) oraz w przypadku śledzenia niezbyt szybkie ruchy obiektu śledzonego.

Wbudowany mechanizm rozpoznawania mowy działa akceptowalnie w warunkach laboratoryjnych. W warunkach zakłóceń np. hałasu, wielu źródeł dźwięku jego użycie do sterowania robotem jest raczej dyskusyjne. Robot ma także możliwość generowania mowy. Można zainstalować dwa języki w pamięci robota, w tym polski. Nasze roboty mają wybrane język angielski i język polski. O ile generacja mowy w języku angielskim jest bezproblemowa, o tyle generacja mowy w języku polskim nie działa idealnie. Dotyczy to wymowy niektórych polskich znaków diakrytycznych (np. ś, ć – są nierozpoznawalne). Uwaga ta dotyczy algorytmu z 2014 roku.

W oczach robota zlokalizowane są nadajniki emiterów podczerwieni. Potencjalnie mogą one być wykorzystane do komunikacji pomiędzy robotami, lub z innymi urządzeniami, ale producent sam nie rekomenduje ich używania. Do komunikacji robotów ze sobą bezpieczniej jest użyć modułów WiFi. Taka komunikacja nie wymaga ich bliskiej obecności, w trakcie niej mogą odwoływać się do współdzielonych zasobów.

Złącze USB, typowo używane do aktualizacji oprogramowania, znajduje się w tyle głowy. Umożliwia ono podłączenie zewnętrznych urządzeń takich jak Kinect, Asus 3D lub urządzeń Arduino [10], co znacząco może rozszerzyć możliwości robota.

Robot może także wchodzić w interakcję z człowiekiem poprzez zmianę oświetlenia diod LED, które mają możliwość zmiany kolorów. Najważniejsze obszary zawierające takie oświetlenie to oczy robota, okolice czujników dotykowych na głowie, okolice głośników, przycisk włączający robota oraz górna część stopy. Jest kilka wbudowanych mechanizmów zmiany tego oświetlenia (np. rozładowanie baterii powoduje, że oczy robota z koloru białego zmieniają się stopniowo na kolor czerwony oznaczający wyczerpanie baterii). Oświetleniem tym możemy dowolnie sterować wzmacniając działania robota o te efekty.

Reakcja na roboty NAO wśród odbiorców jest zdecydowanie pozytywna. Należy tu jednak wyróżnić dwie grupy odbiorców, wśród których mogą pojawić się początkowo problemy. W pierwszym przypadku dotyczy to małych dzieci które początkowo boją się robota, potrzebują czasu do oswojenia się z obecnością tej konstrukcji. Podobnie jest w grupie dzieci z zaburzeniami takimi jak autyzm. Przy czym wiek dzieci ma tu mniejsze znaczenie. Często autyzmowi towarzyszą inne zaburzenia (np. słuchu) i stosowanie w terapii robotów NAO wymaga wiele uwagi i powolnego wprowadzania w środowisko dziecka. Opisywane na forach doświadczenia terapeutów używających robotów wskazują na tego typu trudności. Czas akceptacji obecności robota może trwać nawet kilka tygodni. Jednak po okresie zaakceptowania i zrozumienia zasad interakcji z robotem doświadczenia terapeutów są pozytywne.

5. PODSUMOWANIE

Konstrukcja robota NAO jest dostatecznie odporna na typowe użytkowanie. Sporadyczne problemy w trakcie eksploatacji są do zaakceptowania w porównaniu z zaletami tej konstrukcji. Odzworowanie możliwości ruchowych człowieka jest wystarczające, a subiektywnie oceniając nawet lepsze niż można by się spodziewać po tak niewielkiej konstrukcji i jej uproszczeniach. Uproszczenie konstrukcji dłoni do trzech palców działających jako proste chwytaki także nie stanowi problemu. Udźwig robota jest niewielki ale umożliwia utrzymanie pisaka, czy podniesienie niedużego przedmiotu.

Roboty NAO znakomicie spisują się jako roboty edukacyjne, są lubiane przez odbiorców niezależnie od wieku. Wciąż potrafią zadziwić odbiorców swoimi możliwościami ruchowymi i interakcji z otoczeniem. Jako stosunkowo niedrogi robot zrewolucjonizował ten obszar robotyki poprzez swoje upowszechnienie (ponad 5000 sprzedanych egzemplarzy) a dostarczane oprogramowanie umożliwia budowę zaawansowanych aplikacji. Firma zapewnia wystarczające wsparcie dotyczące nauki programowania, organizowane są tematyczne interaktywne webinaria, zorganizowane jest forum użytkowników na którym aktywni są także konstruktorzy firmu Aldebaran. Czasem na odpowiedź trzeba odczekać trochę czasu ale kontakt jest ostatecznie skuteczny. Dzięki popularności robota można korzystać z wielu gotowych aplikacji dostarczanych przez użytkowników z całego świata. Są one dostępne na stronach producenta lub bezpośrednio udostępniane przez użytkowników.

LITERATURA

1. Gouaillier D., Hugel V., Plazevic P., Kilner C., Monceaux J., Lafourcade P., Marnier B., Serre J. and Maisonnier B., The NAO humanoid: a combination of performance and affordability, CoRR abs/0807.3223, July 2008.
2. Strona projektu Ask NAO: <https://asknao.aldebaran.com/>
3. Shamsuddin S., Yussof H., Ismail L., Hanapiach F. A., Mohamed S., Piah H. A. and Zahiri N. I., Initial response of autistic children in human-robot interaction therapy with humanoid robot NAO, 2012 IEEE 8th International Colloquium on Signal Processing and its Applications (CSPA), 23-25 March 2012, IEEE, 2012, s. 188–193.
4. Dokumentacja: http://doc.aldebaran.com/2-1/news/2.0/choregraphe_rn2.0.html#
5. Dok. http://doc.aldebaran.com/2-1/software/webots/webots_index.html#webots
6. Dokumentacja NAO: http://doc.aldebaran.com/2-1/home_ nao.html
7. <http://doc.aldebaran.com/2-1/naoqi/core/index.html#naoqi-core>
8. Strona projektu ROS: www.ros.org
9. Calderon J. M., Elibol E., Moreno W., Weitzenfeld A., Current Usage Reduction Through Stiffness Control in Humanoid Robot, 13th IEEE-RAS International Conference of Humanoid Robots, Atlanta 2013.
10. www.humarobotics.com/en/robotics-lab/nao-and-arduino.html

Przyjęto do druku dnia 03.10.2015 r.

THE EXPERIENCE FROM THE OPERATION OF THE NAO HUMANOID ROBOTS

Arkadiusz GARDECKI

ABSTRACT *The paper presents several years of the operation experience with two generations of the autonomous humanoid robot NAO. Major sensors, responsible for this human robot interactions have been described. The possibilities and limitations of the robot and some observations that result from the experience with dedicated software (Choregraphe, Monitor and Webots for NAO) have been also discusses. Major changes in the design of the robot and examples of the use of this robotic platform (e.g., education, Autism therapy) have been discussed as well.*

Keywords: *robotics, humanoid robot NAO*



Dr inż. Arkadiusz GARDECKI urodził się w 1973 roku w Opolu. W 1997 r. uzyskał stopień magistra inżyniera na Politechnice Opolskiej, a w 2006 r. obronił doktorat na Wydziale Elektrotechniki, Automatyki i Informatyki tejże Politechniki. Obecnie jest tam zatrudniony na stanowisku adiunkta. Obszar jego zainteresowań obejmuje: roboty mobilne, techniki sterowania silników elektrycznych, metody numerycznego rozwiązywania układów równań różniczkowych oraz zagadnienia związane z pomiarami jakości energii elektrycznej.