

Łukasz WITANOWSKI, Piotr LAMPART

Polska Akademia Nauk

Instytut Maszyn Przepływowych im. Roberta Szewalskiego

Fiszera 14, 80-231 Gdańsk

E-mail: lwitanowski@imp.gda.pl, lampart@imp.gda.pl

Przestrzenne kształtowanie łopatki turbiny parowej z wykorzystaniem algorytmu hybrydowego i sieci neuronowej

1 Wstęp

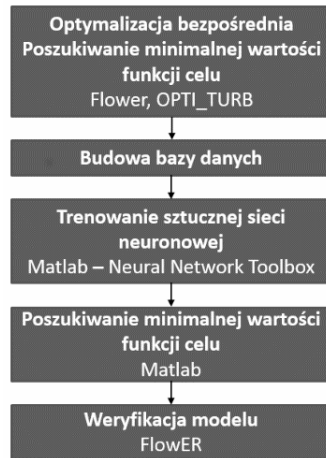
Głównym celem projektanta turbin parowych jest osiągnięcie produktu, trwałego, wysoko sprawnego przy zapewnieniu jak najniższej ceny i spełnieniu wymagań rynku zmieniających się w niezwykłym tempie. Jedną z technologii pozwalających na osiągnięcie powyższego celu w zakresie kształtowania układu przepływowego turbin jest CFD. Tradycyjnie CFD służyło jako narzędzie do analizy zmian zachodzących przy modyfikacji geometrii układu przepływowego. Rozwój wydajności i dokładności solverów pozwolił na sprzężenie ich z pakietami optymalizacyjnymi. Celem poniższej pracy jest podwyższenie sprawności przepływowej turbiny. Kolejnym zadaniem jest opracowanie metody poszukiwania najmniejszej wartości funkcji celu poprzez sieć neuronową, która bazuje na wynikach optymalizacji układu przepływowego. W artykule przedstawiono wyniki optymalizacji 2 ostatnich stopni układu przepływowego turbiny ORC9000 100kW (rys. 1) przy użyciu metody hybrydowej [1, 2] i sztucznej sieci neuronowej [3-7]. Optymalizowana turbina znajduje się w laboratorium Instytutu Maszyn Przepływowych PAN w Gdańsku jako element układu ORC.

2 Opis metody optymalizacji

W pierwszym etapie utworzono 100 zestawów parametrów geometrycznych za pomocą optymalizacji bezpośredniej na potrzeby trenowania sieci neuronowej oraz 100 zestawów geometrycznych w celu zweryfikowania sieci neuronowej. Każdy z zestawów zawierał 14 parametrów modyfikacji geometrycznych oraz wartość funkcji celu. Tworzenie pojedynczego zestawu polegało na przeprowadzeniu obliczeń CFD kanału przepływowego, który został utworzony w wyniku kolejnej iteracji algorytmu optymalizacyjnego. W celu porównania metod zbudowano bazę danych zawierającą 300 zestawów geometrycznych. W kolejnym etapie za pomocą pakietu programu Matlab Neural Network wytrenowano oraz przetestowano nowo powstałą sieć neuronową.

W procesie trenowania wykorzystano funkcję 'trainlm', która opiera się na algorytmie Levenberga-Marquardta. Użyta metoda jest powszechnie stosowana ze względu na szybki czas uzyskania zbieżności sieci oraz jej stabilność [8,9]. Kolejno wygenerowano 300 000 zestawów parametrów geometrycznych, poszukując

minimalnej wartości funkcji celu. Otrzymany zestaw zweryfikowano w solverze FlowER. Metodę optymalizacji przedstawiono na rysunku 1.



Rys. 1. Schemat procesu optymalizacji nową metodą hybrydową

Fig. 1. Scheme of new hybrid optimization process

3 Optymalizacja bezpośrednia

W metodzie hybrydowej na potrzeby optymalizacji bezpośredniej wykorzystano zmodyfikowany algorytm Hooke'a-Jeevesa połączony z metodą symulowanego wyżarzania. W algorytmie można wyróżnić dwie fazy obliczeń w jednej iteracji: etap próbny i etap roboczy.

W etapie próbnym przeszukiwane jest otoczenie punktu bazowego poprzez wykonanie kroków próbnych o zadanej długości wzdłuż wszystkich kierunków ortogonalnego układu współrzędnych. Dla n -wymiarowej funkcji celu nowy algorytm hybrydowy zawsze wykonuje tę samą liczbę kroków próbnych – równą $2n$ w kierunkach równoległych do ortogonalnego układu współrzędnych. W jednym kroku próbnym sprawdzane są wartości funkcji celu w wierzchołkach n -wymiarowego hiperprostokąta, którego długość boku w i -tym kierunku ortogonalnego układu współrzędnych jest równa sumie długości kroku $2i-1$ i długości kroku $2i$. Długość kroku próbnego nie jest stała, tak jak w wersji klasycznej, lecz zmienia się: jest zwiększana, jeśli w poprzedniej iteracji na danej osi ortogonalnego układu współrzędnych nastąpiło zmniejszenie wartości funkcji celu, i zmniejszana w przeciwnym wypadku.

W etapie roboczym następuje przejście od bieżącego punktu bazowego do nowego punktu bazowego zgodnie ze zdefiniowaną metodą przejścia wykorzystującą informacje na temat wartości funkcji celu z etapu próbnego. Krok roboczy jest wykonywany, jeśli co najmniej jeden z kroków próbnych był pomyślny, tzn. że w jego wyniku nastąpiło zmniejszenie wartości funkcji celu. W przeciwnym wypadku o wykonaniu kroku próbnego decyduje test Metropolis'a. W przypadku przejścia tego testu wykonywany jest krok roboczy, nawet jeśli żaden z kroków próbnych nie był pomyślny.

Niepomyślny wynik testu oznacza powrót do poprzedniego punktu bazowego, dla którego otrzymaliśmy co najmniej jeden krok próbny, zmniejszenie długości kroku i rozpoczęcie cyklu od początku.

4 Algorytm uczenia sieci neuronowej

Jedną z najskuteczniejszych metod uczenia sieci neuronowych jest algorytm Levenberga-Marquardta ze wsteczną propagacją. Jest to tak zwana metoda z nauczycielem. Nazwa algorytmu wywodzi się ze sposobu interakcji błędu z wagami połączeń między neuronami sieci. Proces propagacji błędu i zmiany wag połączeń odbywa się w kierunku odwrotnym niż kierunek przetwarzania wektora wejściowego.

Kluczową czynnością w procesie uczenia według algorytmu wstecznej propagacji jest utworzenia ciągu skojarzonych ze sobą danych: wejściowych i docelowych. Ważne, by dane wejściowe i odpowiadające im wartości docelowe pokrywały możliwy maksymalny zakres późniejszego działania sieci.

Po utworzeniu zbiorów danych uczących w ramach algorytmu uczącego wykonywane są kolejno czynności:

1. Sieć przetwarza sygnał wejściowy, generując aktualne wartości wyjściowe. W większości nie pokrywają się one z wartościami docelowymi.
2. Na podstawie porównania danych wyjściowych z danymi docelowymi sieć wylicza błąd sieci.
3. Błąd propaguje wstecz, w kierunku przeciwnym do przepływu danych, modyfikując wagi połączeń.
4. Po zmodyfikowaniu wszystkich wag sieci sygnał ponownie jest podawany na wejście i jeśli błąd po porównaniu nie przekracza zadanej przez użytkownika wartości, podawany jest następny sygnał.

Po przetworzeniu całego zbioru sieć uważa się za nauczoną. W kolejnym etapie następuje proces weryfikacji działania sieci na podstawie utworzonej do tego celu bazy danych.

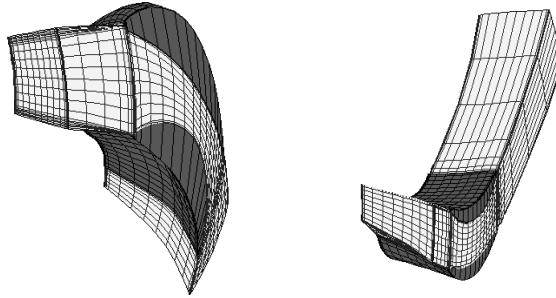
5 Parametry wejściowe

W obliczeniach przepływowych wykorzystano program FlowER. Jest to solver modelu RANS, tj. rozwiązuje układy równań Naviera-Stokesa w uśrednieniu Reynoldsa zapisany w krzywoliniowym układzie współrzędnych. Jako domknięcie układu RANS stosuje dwa modele turbulencji – algebraiczny dwuwarstwowy model Baldwina-Lomaxa lub dwurównaniowy model Mentera $k-\omega$ SST, który stanowi kombinację modeli $k-\epsilon$ i $k-\omega$, a ponadto zakłada liniową zależność pomiędzy energią kinetyczną turbulencji a głównym naprężeniem turbulentnym w warstwie przyściennej. W pre-processorze programu wprowadzono geometrię układu przepływowego i zbudowano siatkę strukturalnej typu H ze zgęszczeniem w pobliżu ścianek profilu i ograniczeń zewnętrznych oraz w pobliżu krawędzi wlotowej i wylotowej (rys.2). Obliczenia prowadzono w dwóch ostatnich stopniach mikroturbiny ORC9000 100kW. Warunki brzegowe przedstawiono w tabeli 1.

Tab. 1. Zestawienie ustawień wejściowych

Tab. 1. Inlet parameters

Solwer przepływowy	FlowER
Temperatura na wejściu do pierwszego wieńca	519.8 K
Ciśnienie na wejściu do pierwszego wieńca	84.77 kPa
Ciśnienie na wyjściu z turbiny	17 kPa
Przepływ	stacjonarny
Model przepływu	k- ω SST
Płyn	Model gazu doskonałego
Funkcja celu	Suma strat energii kinetycznej i energii wylotowej
Ilość modyfikowanych parametrów	14



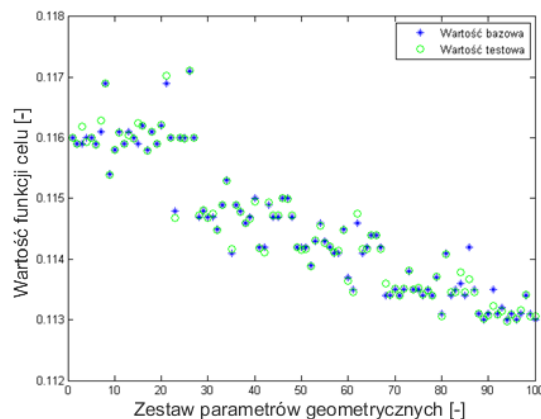
Rys. 2. Przestrzenna siatka obliczeniowa (z lewej kierownica pierwszego stopnia, z prawej wirnik pierwszego stopnia)

Fig. 2. Computational grid of stator (left) and rotor (right) of the original first stage

6 Wyniki obliczeń, wnioski

Rysunek 3 przedstawia porównanie bazowych wartości funkcji celu otrzymanych z wytrenowanej sieci neuronowej. Największy błąd, a więc różnicę pomiędzy bazową wartością funkcji celu a wartością otrzymaną z wytrenowanej sieci neuronowej, uzyskano na poziomie 0,00032.

*Przestrzenne kształtowanie łopatki turbiny parowej
z wykorzystaniem algorytmu hybrydowego i sieci neuronowej*



Rys. 3. Porównanie sztucznej sieci neuronowej z wartościami bazowymi

Fig. 3. Results of training a neural network

W wyniku weryfikacji zestawu danych parametrów geometrycznych uzyskano redukcję strat energii kinetycznej i energii wylotowej z dwóch ostatnich stopni turbiny ORC9000 o wartość 1,4%, z 13,44% do 12,04%. W tabeli 2 przedstawiono zmodyfikowane parametry.

Tab. 2. Zestawienie zmodyfikowanych parametrów geometrycznych

Tab. 2. Change of the optimized parameters in two stages

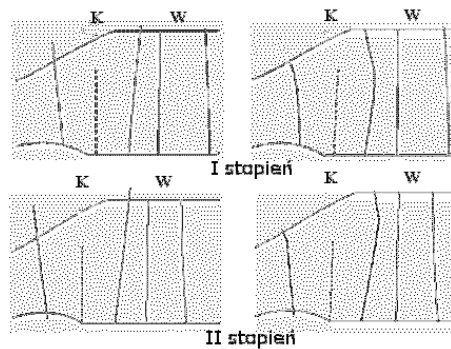
ID	Parametr	Wartość
1	Kąt ustawienia profili łopatki – wieniec 1	-0,308°
2	Kąt ustawienia profili łopatki – wieniec 2	-0,177°
3	Kąt ustawienia profili łopatki – wieniec 3	0,146°
4	Kąt ustawienia profili łopatki – wieniec 4	0,7°
5	Proste skręcenie łopatki – wieniec 2 (wirnik 1)	-0,576°
6	Proste skręcenie łopatki – wieniec 4 (wirnik 2)	-0,523°
7	Kąt liniowego pochylenia obwodowego – wieniec 1	-7,53°
8	Kąt liniowego pochylenia obwodowego – wieniec 3	-0,744°
9	Kąt liniowego pochylenia osiowego – wieniec 1	-2,96°
10	Kąt liniowego pochylenia osiowego – wieniec 3	-2,30°
11	Pochylenie szablowe przy stopie i wierzchołku – wieniec 1	0,228 ¹
12	Pochylenie szablowe przy stopie i wierzchołku – wieniec 3	0,228 ¹
13	Złożone pochylenie osiowe – wieniec 1	0,209 ¹
14	Złożone pochylenie osiowe – wieniec 3	0,128 ¹

1) parametry przestrzennego kształtowania łopatek opisano w [1]

Na rysunku 4 przedstawiono rzut merydionalny 1 i 2 stopnia modyfikowanego układu przepływowego. Rzut merydionalny jest rzutem prostopadłym wieńców na płaszczyznę

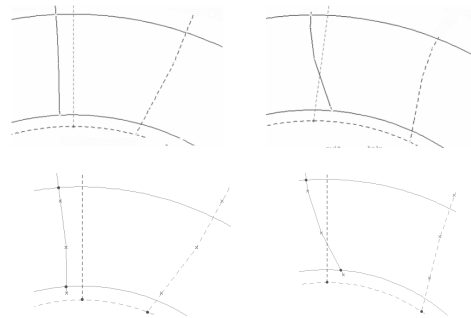
merydionalną, ilustrującym krawędzie wlotową i wylotową kierownicy i wirnika. W obu przypadkach obserwuje się zmiany dla liniowego pochylenia osiowego [1]. Wartość uległa zmianie kolejno o kąt $-2,96^\circ$ i $-2,30^\circ$.

W największym stopniu zmianie uległ kąt liniowego pochylenia obwodowego. Modyfikacje przedstawiono na rysunku 5 w rzucie promieniowym, który jest rzutem z przodu wieńca prostopadłym do osi turbiny, ilustrującym krawędź wlotową i wylotową wieńca.



Rys. 4. Rzut merydionalny kanału łopatkowego pierwszego (rys. górne) i drugiego (rys. dolne) stopnia turbiny (geometria zoptymalizowana – rys. prawy), K – kierownica, W – wirnik

Fig. 4. Original (left) and final (right) geometry of the stage group in meridional view, K – stator, W – rotor

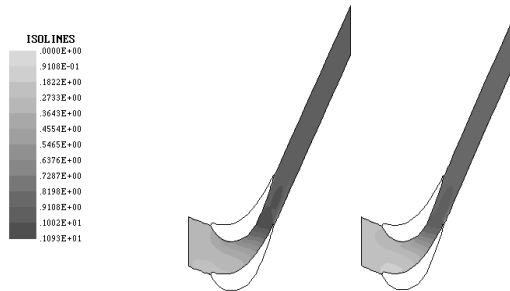


Rys. 5. Kanał łopatkowy w rzucie promieniowym ukazującym krawędź wlotową i wylotową łopatki kierowniczej w pierwszym (rys. górny) i drugim (rys. dolny) stopniu turbiny (geometria zoptymalizowana – rys. prawy)

Fig. 5. Original (left) and final (right) shape of stator blades first (top) and second (bottom) stage in circumferential view

Rysunek 6 przedstawia porównanie izolinii liczb Macha na średnicy podziałowej w wirniku w 2 stopniu badanej turbiny. W obszarze wylotowym zredukowano prędkości okołodźwiękowe, co w rezultacie przełożyło się na ograniczenie strat wynikających z prędkości wylotowej z turbiny. Ponadto w wyniku optymalizacji

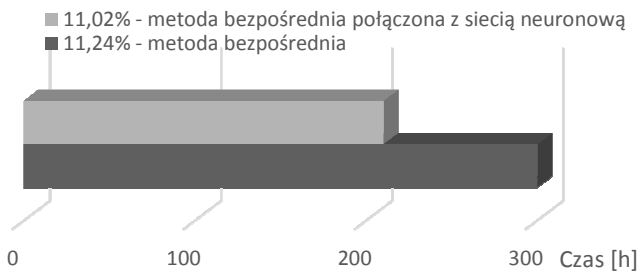
zredukowano straty przepływowe w warstwach przyściennych i straty przepływów wtórnych.



Rys. 6. Porównanie izolinii liczby Macha na średnicy podziałowej (geometria zoptymalizowana – rys. prawy)

Fig. 6. Mach number contours in the last stage rotor for original (left) and optimized (right) geometry

Na rysunku 7 zestawiono czasy optymalizacji układu przepływowego. W wyniku tradycyjnej metody bezpośredniej z wykorzystaniem algorytmu zmodyfikowanego Hooke'a-Jeevesa uzyskano w czasie 300 h pracy wartość funkcji celu na poziomie 11,24%. Korzystając z połączenia metody optymalizacji bezpośredniej i sieci neuronowej, w czasie 210 h pracy otrzymano wartość funkcji celu na poziomie 11,02%. Dalsze próby obniżenia poszukiwanej wartości nie przyniosły pożądanego efektu. Stwierdzono, iż dalsze obniżenie wartości funkcji celu może nastąpić jedynie w przypadku wytrenowania sztucznej sieci neuronowej na większej bazie danych. W przypadku walidacji sieci należy znaleźć minimalną liczbę zestawów geometrycznych, które zapewnią potwierdzenie prawidłowej pracy sieci.



Rys. 7. Porównanie procesów optymalizacji

Fig. 7. Comparison of optimization process

Literatura

1. Lampart P., Hirt Ł.: Complex multidisciplinary optimization of turbine blading systems, *Arch. Mech.*, vol. 64, pp. 153 -175, 2012
2. Ostanin A.: *Metody optymalizacji z Matlab*, Nakom, 2009

3. Pierret S., Van den Braembussche R.A.: Turbomachinery Blade Design Using a Navier-Stokes Solver and Artificial Neural Network, *Trans. ASME*, vol. 121, 1999
4. Tadeusiewicz R.: *Sieci neuronowe*, AOW, 1993
5. Ostanek J.K.: Improving Pin-Fin Heat Transfer Predictions Using Artificial Neural Networks, *J. Turbomach.*, vol. 136, pp. 051010-1:9, 2013
6. Mengistu T., Ghaly W., Mansour T.: *Aerodynamic Shape Optimization of Turbine Blades Using a Design-Parameter-Based Shape Representation*, vol. 6 Turbo Expo 2007, Parts A B, pp. 1395-1404, 2007
7. Kosowski K., Tucki K., Kosowski A.: Turbine stage design aided by artificial intelligence methods, *Expert Syst. Appl.*, vol. 36, pp. 11536-11542, 2009
8. Ghaffari A., Abdollahi H., Khoshayand M.R., Bozchalooi I.S., Dadgar A., Rafiee-Tehrani M.: Performance comparison of neural network training algorithms in modeling of bimodal drug delivery., *Int. J. Pharm.*, vol. 327, pp. 126-138, 2006
9. Sakamoto H., Matsumoto K., Kuwahara A., Hayami Y.: *Acceleration and Stabilization Techniques for the Levenberg-Marquardt Method* - Version details - Trove, Oxford Univ. Press, 2005

Streszczenie

W artykule przedstawiono metodę oraz wyniki optymalizacji sprawnościowej ołopatkowania przestrzennego dwóch ostatnich stopni mikroturbiny ORC9000 100kW z wykorzystaniem pakietu OPT_TURB oraz Matlab Toolbox - Neural Network. Wyznaczono funkcję celu z post-processingu obliczeń 3D części przepływowej przy użyciu programu FlowER – solvera modelu RANS. Funkcje celu zdefiniowano, jako sumę strat energii kinetycznej oraz energii wylotowej. Zaproponowano model sztucznej sieci neuronowej wytrenowanej przy użyciu algorytmu Lavenberga-Marquardta.

Słowa kluczowe: optymalizacja, sieć neuronowa, ORC, CFD, sprawność

3D shape optimization of steam turbine blade using a hybrid algorithm and artificial neural network

Summary

The paper presents the methods and results of optimization of two last stages of a microturbine ORC9000 100kW using a software package OPTI_TURB and Matlab Toolbox – Neural Network. Values of the minimised objective function, that is losses of kinetic energy and exit velocity are found in the first step from 3D RANS computation (from a FlowER code) of geometries changed during the direct process optimization. In the second step an indirect method is used. The applied neural network is trained over flow data with the help of Lavenberg – Marquardt backpropagation.

Keywords: optimization, neural network, ORC, CFD, efficiency