



Politechnika Śląska

Wydział Mechaniczny Technologiczny

Katedra Wytrzymałości Materiałów i Metod
Komputerowych Mechaniki

Witold Grela

Optymalizacja ewolucyjna kształtu łopatek turbiny

Rozprawa doktorska

Promotor:

prof. dr hab. inż. Tadeusz Burczyński

Gliwice 2006

Praca została wykonana przy wsparciu Avio Polska Sp. z .o.o. (ul. Grażyńskiego 141, 43-300 Bielsko-Biała), która dostarczyła modele, dane materiałowe, warunki brzegowe oraz udostępniła stanowisko obliczeniowe i oprogramowanie Ansys.

Spis ilustracji	6
Spis tabel	9
Spis oznaczeń	10
1. Wprowadzenie	16
1.1. <i>Cel i zakres pracy.....</i>	<i>16</i>
1.2. <i>Organizacja pracy.....</i>	<i>19</i>
2. Teza pracy	21
3. Algorytmy ewolucyjne w optymalizacji konstrukcji.....	22
3.1. <i>Sformułowanie zadania optymalizacji ewolucyjnej</i>	<i>22</i>
3.2. <i>Omówienie dostępnej literatury</i>	<i>29</i>
3.3. <i>Zastosowanie optymalizacji ewolucyjnej w mechanice konstrukcji.....</i>	<i>30</i>
4. Środowisko i narzędzia optymalizacji łopatki turbiny przemysłowej.....	33
5. Parametryzacja modelu geometrycznego łopatki	39
5.1. <i>Model geometryczny łopatki turbiny przemysłowej</i>	<i>39</i>
5.2. <i>Modyfikacja chromosomu do postaci wykonalnej</i>	<i>43</i>
5.3. <i>Opis generacji modelu strefy optymalizowanej</i>	<i>47</i>
6. Model termomechaniczny łopatki turbiny przemysłowej	52
6.1. <i>Założenia upraszczające</i>	<i>52</i>
6.2. <i>Model matematyczny</i>	<i>56</i>
6.2.1. <i>Sformułowanie zadania brzegowego teorii termosprężystości</i>	<i>56</i>
6.2.2. <i>Dyskretyzacja modelu</i>	<i>62</i>
6.2.3. <i>Podstawowe równania MES</i>	<i>65</i>
6.2.4. <i>Globalizacja równań MES</i>	<i>68</i>
6.3. <i>Warunki brzegowe.....</i>	<i>69</i>
6.4. <i>Obciążenie cieplne. Zarys analizy przewodzenia ciepła w ujęciu MES.....</i>	<i>72</i>
6.5. <i>Własności materiałów</i>	<i>76</i>

6.6.	<i>Model dyskretny MES</i>	77
6.6.1.	<i>Model szczegółowy</i>	78
6.6.2.	<i>Model uproszczony</i>	78
6.7.	<i>Wyniki analizy MES segmentu rotora turbiny w warunkach roboczych</i>	80
7.	Ewolucyjna optymalizacja jednokryterialna kształtu łopatki ze względu na minimum maksymalnego naprężenia głównego	84
7.1.	<i>Sformułowanie zadania optymalizacji ewolucyjnej</i>	84
7.2.	<i>Etapy optymalizacji ewolucyjnej</i>	86
7.2.1.	<i>Reprodukcja</i>	87
7.2.2.	<i>Operacje genetyczne - krzyżowanie</i>	88
7.2.3.	<i>Operacje genetyczne - mutacja</i>	89
7.2.4.	<i>Sukcesja</i>	89
7.3.	<i>Opis kryterium zatrzymania obliczeń</i>	90
7.4.	<i>Omówienie wyników i wnioski</i>	91
8.	Ewolucyjna optymalizacja jednokryterialna kształtu łopatki ze względu na minimum masy	99
8.1.	<i>Sformułowanie zadania optymalizacji ewolucyjnej</i>	99
8.2.	<i>Etapy optymalizacji ewolucyjnej</i>	101
8.3.	<i>Opis kryterium zatrzymania obliczeń</i>	101
8.4.	<i>Omówienie wyników i wnioski</i>	101
9.	Ewolucyjna optymalizacja dwukryterialna kształtu łopatki ze względu na minimum masy i maksymalnego naprężenia głównego	109
9.1.	<i>Sformułowanie zadania optymalizacji ewolucyjnej</i>	109
9.2.	<i>Etapy optymalizacji ewolucyjnej</i>	111
9.3.	<i>Opis kryterium zatrzymania obliczeń</i>	111
9.4.	<i>Omówienie wyników i wnioski</i>	112
10.	Wpływ zmiany kształtu łopatki w strefie optymalizowanej na częstość drgań własnych łopatki	120
10.1.	<i>Cel analizy częstości drgań własnych łopatki</i>	120

10.2.	<i>Model matematyczny do analizy częstości drgań własnych łopatk</i>	120
10.3.	<i>Model symulacyjny MES do analizy częstości drgań własnych łopatk</i>	124
10.4.	<i>Wyniki analizy częstości drgań własnych łopatk</i>	125
11.	Podsumowanie i wnioski końcowe	126
12.	Streszczenie (Summary)	128
	Literatura	131
	Dodatek 1 - Kod źródłowy programu optymalizacyjnego (APDL)	134
	Dodatek 2 - Struktura plików wejścia-wyjścia programu optymalizacyjnego	159

SPIS ILUSTRACJI

Rys. 3.1 Ogólny schemat blokowy procesu optymalizacji ewolucyjnej.....	27
Rys. 4.2 Schemat blokowy procesu optymalizacji ewolucyjnej kształtu łopatki turbiny przemysłowej.	35
Rys.4.3 Budowa łopatki turbiny.....	37
Rys. 4.4 Łopaska turbiny - strefa optymalizowana.	38
Rys. 5.1 Turbina przemysłowa i model geometryczny rotora.	40
Rys. 5.2 Model geometryczny rotora turbiny przemysłowej – elementy i podział z wyodrębnieniem części optymalizowanej.....	40
Rys. 5.3 Szczegóły parametryzacji strefy optymalizowanej.....	42
Rys. 5.4 Schemat wystąpienia błędu geometrycznego nr 1.	45
Rys. 5.5 Schemat wystąpienia błędu geometrycznego nr 2.	46
Rys. 5.6 Schemat wystąpienia błędu geometrycznego nr 3.	47
Rys. 5.7 Model początkowy strefy optymalizowanej łopatki.	48
Rys. 5.8 Siatka krzywych opisująca powierzchnię strefy optymalizowanej.....	49
Rys. 5.9 Powierzchnie opisujące kształt strefy optymalizowanej.....	49
Rys. 5.10 Utworzenie modelu geometrycznego strefy optymalizowanej.....	50
Rys. 5.11 Schemat procedury optymalizacyjnej kształtu łopatki turbiny przemysłowej.....	51
Rys. 6.1 Sformułowanie zadania brzegowego teorii termosprężystości.....	56
Rys. 6.2 Element czworościenny czterowęzłowy.	62
Rys. 6.3 Element czworościenny dziesięciowęzłowy.....	64
Rys. 6.4 Model geometryczny rotora turbiny przemysłowej – przemieszczeniowe warunki brzegowe.	71
Rys. 6.5 Model geometryczny rotora turbiny przemysłowej – rozkład ciśnienia i temperatury.	71
Rys. 6.6 Sformułowanie dwuwymiarowego zadania przewodzenia ciepła w stanie ustalonym.	73
Rys. 6.7 Element trójkątny trójwęzłowy do obliczeń termicznych.....	74
Rys. 6.8 Własności materiałów wykorzystanych w rotorze turbiny przemysłowej.....	77
Rys. 6.9 Siatka elementów skończonych na modelu szczegółowym segmentu rotora turbiny.	79
Rys. 6.10 Siatka elementów skończonych na modelu uproszczonym segmentu rotora turbiny. Elementy o zredukowanej liczbie węzłów zaznaczone kolorem niebieskim.....	80

Rys. 6.11 Wyniki analizy MES na segmencie rotora turbiny w warunkach roboczych.	81
Rys. 6.12 Wyniki analizy MES na optymalizowanej części łopatki turbiny w warunkach roboczych - naprężenia.....	82
Rys. 6.13 Wyniki analizy MES na optymalizowanej części łopatki turbiny w warunkach roboczych - przemieszczenia.	82
Rys. 7.1 Strefa odczytu maksymalnego naprężenia głównego z optymalizowanej części łopatki (powierzchnie ciemnoszare).....	85
Rys. 7.2 Schemat wystąpienia warunku zatrzymania obliczeń.....	91
Rys. 7.3 Zmiana wartości funkcji przystosowania F (znormalizowanego naprężenia - $\bar{\sigma}$) – minimalizacja maksymalnego naprężenia głównego.....	93
Rys. 7.4 Zmiana wartości znormalizowanej masy (\bar{M}) i ilość ukaranych osobników – minimalizacja maksymalnego naprężenia głównego.....	93
Rys. 7.5 Zmiana odchylenia standardowego (h_k) funkcji przystosowania – minimalizacja naprężenia	94
Rys. 7.6 Zmiana średniej szybkości poprawy rozwiązania (v_k) – minimalizacja naprężenia .	95
Rys. 7.7 Zmiana średniego czasu oceny wartości funkcji przystosowania (w_k) – minimalizacja naprężenia	96
Rys. 7.8 Siatka elementów skończonych na modelu optymalnym – minimalizacja maksymalnego naprężenia głównego.....	96
Rys. 7.9 Porównanie kształtu modelu w konfiguracji „0” i modelu zoptymalizowanego – minimalizacja maksymalnego naprężenia głównego.....	97
Rys. 7.10 Porównanie przemieszczenia wypadkowego na modelu w konfiguracji „0” i modelu zoptymalizowanym – minimalizacja maksymalnego naprężenia (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).	97
Rys. 7.11 Porównanie rozkładu maksymalnego naprężenia głównego na modelu w konfiguracji „0” i modelu zoptymalizowanym – minimalizacja maksymalnego naprężenia głównego (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).	98
Rys. 8.1 Strefa obliczania masy optymalizowanej części łopatki (część ciemnoszara).....	100
Rys. 8.2 Zmiana wartości funkcji przystosowania (znormalizowanej masy \bar{M}) – minimalizacja masy.....	103
Rys. 8.3 Zmiana znormalizowanego naprężenia ($\bar{\sigma}$) i ilości ukaranych osobników – minimalizacja masy.....	103
Rys. 8.4 Zmiana odchylenia standardowego funkcji przystosowania (h_k) – minimalizacja masy.	104
Rys. 8.5 Zmiana średniej szybkości poprawy rozwiązania (v_k) – minimalizacja masy.....	105

Rys. 8.6 Zmiana średniego czasu oceny wartości funkcji przystosowania (w_k) – minimalizacja naprężeń.	106
Rys. 8.7 Siatka elementów skończonych na modelu optymalnym – minimalizacja masy. ...	106
Rys. 8.8 Porównanie kształtu modelu w konfiguracji „0” i modelu zoptymalizowanego – minimalizacja masy.	107
Rys. 8.9 Porównanie przemieszczenia wypadkowego na modelu w konfiguracji „0” i modelu zoptymalizowanym – minimalizacja masy (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).	107
Rys. 8.10 Porównanie rozkładu maksymalnego naprężenia głównego na modelu w konfiguracji „0” i modelu zoptymalizowanym – minimalizacja masy (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).	108
Rys. 9.1 Funkcja przystosowania – zmiana w trakcie optymalizacji dwukryterialnej z uwzględnieniem wyników minimalizacji masy i minimalizacji naprężenia.	114
Rys. 9.2 Funkcja przystosowania – zmiana w trakcie optymalizacji dwukryterialnej.	114
Rys. 9.3 Znormalizowana masa- zmiana w trakcie optymalizacji dwukryterialnej.	115
Rys. 9.4 Maksymalne znormalizowane naprężenie główne – zmiana w trakcie optymalizacji dwukryterialnej.	115
Rys. 9.5 Rozkład Pareto dla optymalizacji dwukryterialnej z uwzględnieniem punktów otrzymanych w trakcie minimalizacji masy i naprężenia oraz z zaznaczeniem punktu idealnego.	116
Rys. 9.6 Rozkład Pareto dla optymalizacji dwukryterialnej z uwzględnieniem wyłącznie punktów dla niezerowych współczynników a,b.	116
Rys. 9.7 Rozkład Pareto dla optymalizacji dwukryterialnej z wyodrębnieniem punktów niezdominowanych – front Pareto. Osobnik wybrany do szczegółowego omówienia wyróżniony na zielono.	117
Rys. 9.8 Siatka elementów skończonych na modelu optymalnym – optymalizacja dwukryterialna.	118
Rys. 9.9 Porównanie kształtu modelu w konfiguracji „0” i modelu zoptymalizowanego – optymalizacja dwukryterialna.	118
Rys. 9.10 Porównanie przemieszczenia wypadkowego na modelu w konfiguracji „0” i modelu zoptymalizowanym – optymalizacja dwukryterialna (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).	119
Rys. 9.11 Porównanie rozkładu maksymalnego naprężenia głównego na modelu w konfiguracji „0” i modelu zoptymalizowanym – optymalizacja dwukryterialna (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).	119
Rys. 10.1 Schemat modelu MES łopatki turbiny do analizy modalnej.	124

SPIS TABEL

Tabela 5.1. Wartości dopuszczalne parametrów modelu optymalizowanego (znormalizowane w stosunku do konfiguracji „0”).	44
Tabela 6.1. Porównanie wyników analizy MES modelu segmentu rotora turbiny w warunkach roboczych - konfiguracja uproszczona i szczegółowa.	78
Tabela 7.1. Wyniki minimalizacji maksymalnego naprężenia głównego.....	92
Tabela 8.1. Wyniki minimalizacji masy.....	101
Tabela 9.1. Wyniki optymalizacji dwukryterialnej z uwzględnieniem wyników minimalizacji masy i naprężenia.	113
Tabela 9.2. Wyniki optymalizacji dwukryterialnej dla wybranego osobnika ($a=0.75$, $b=0.25$, pokolenie 173).....	113
Tabela 10.1. Porównanie wyników analizy modalnej dla modeli łopatek w różnych konfiguracjach geometrycznych.....	125

SPIS OZNACZEŃ

Wielkości wektorowe i macierzowe oznaczone są drukiem **pogrubionym**.

$[\]$	składowe macierzy (tensora)
$\{ \}$	składowe wektora
a	współczynnik wagi dla masy
a_i, b_i, c_i, d_i, e_i	współczynniki funkcji kształtu ($i=1, 2, \dots, 10$)
b	współczynnik wagi dla naprężeń
B	macierz pochodnych funkcji kształtu
C	globalna macierz tłumienia modelu
$c1, c2$	punkty przecięcia chromosomu w trakcie krzyżowania
d	przedział uśredniania szybkości poprawy
D	macierz sztywności
E	moduł sprężystości materiału (moduł <i>Younga</i>)
e_{ikm}	trójwskaźnikowy symbol permutacyjny <i>Levi-Civity'ego</i>
f^e	elementowy wektor uogólnionych sił węzłowych (składowe f_{xn}, f_{yn}, f_{zn})
f_w	funkcja wagowa
F^i_{\max}	maksymalna wartość funkcji przystosowania dla pokolenia i
F^i_{\min}	minimalna wartość funkcji przystosowania dla pokolenia i
$F(T)$	wartość funkcji przystosowania dla chromosomu T
f_b	wektorowe pole sił bezwładności (składowe f_b^x, f_b^y, f_b^z)
f_b^e	elementowy wektor bezwładnościowych sił węzłowych
F	globalny wektor uogólnionych sił węzłowych
G	moduł odkształcenia postaciowego (moduł <i>Kirchhoffa</i>)
H	wysokość strefy optymalizowanej
h_i	odchylenie standardowe funkcji przystosowania w pokoleniu i

h	wysokość podziału powierzchni optymalizowanej
H_{max}	maksymalna dopuszczalna wartość wysokości H
H_{min}	minimalna dopuszczalna wartość wysokości H
I	macierz jednostkowa
J	wektor własny
K	globalna macierz sztywności modelu
K^e	macierz sztywności elementu skończonego
k_{max}	ilość pokoleń wygenerowanych w trakcie ewolucji do momentu zatrzymania obliczeń
k_{min}	minimalna ilość poprawionych rozwiązań
k_p	ilość rozwiązań poprawionych w procesie optymalizacji
L	macierz dolna trójkątna otrzymywana w wyniku triangulacji B-Ch
$L_F(c)$	zbiór poziomicy funkcji przystosowania określony na poziomie c
l_{min}	długość krótszej z dwóch krzywych dla których tworzony jest łuk RI
M	globalna macierz bezwładności modelu
M_0	masa optymalizowanej części pióra łopatki modelu w konfiguracji „0”
$M(T)$	masa części optymalizowanej obliczona dla osobnika o chromosomie T
M^e	macierz bezwładności elementu skończonego
N	macierz funkcji kształtu
N_i	podmacierz funkcji kształtu
N_i	funkcja interpolacyjna (funkcja kształtu)
p	pole sił powierzchniowych (składowe p_x, p_y, p_z)
$P()$	dystrybuanta funkcji
$p(T)$	prawdopodobieństwo selekcji osobnika o chromosomie T
p^e	elementowy wektor zastępczych sił węzłowych reprezentujących pole ciśnień (składowe p_{xm}, p_{ym}, p_{zm})

P_m	masowa funkcja kary
$P_{m\sigma}$	naprężeniowo-masowa funkcja kary
P_σ	naprężeniowa funkcja kary
Q	prędkość wytwarzania ciepła
\mathbf{q}	pole sił objętościowych (składowe q_x, q_y, q_z)
q_c	wydatek przepływu ciepła przez jednostkę powierzchni
\mathbf{q}^e	elementowy wektor zastępczych sił węzłowych reprezentujących siły objętościowe (składowe q_{xn}, q_{yn}, q_{zn})
q_{ni}	składowe obciążenia powierzchniowego na kierunek i
R	funkcja reszty
$r(T)$	ranga osobnika o chromosomie T
R	funkcja reszty
$R1$	większy (górny) z promieni zaokrąglenia krzywej przekroju
$R1_i$	promień górnego łuku w i -tym przekroju strefy sparametryzowanej
$R1_M$	wartość promienia $R1$ po modyfikacji
$R1_{\max}$	maksymalna dopuszczalna wartość promienia $R1$
$R1_{\min}$	minimalna dopuszczalna wartość promienia $R1$
$R2$	mniejszy (dolny) z promieni zaokrąglenia krzywej przekroju
$R2_i$	promień dolnego łuku w i -tym przekroju strefy sparametryzowanej
$R2_M$	wartość promienia $R2$ po modyfikacji.
$R2_{\max}$	maksymalna dopuszczalna wartość promienia $R2$
$R2_{\min}$	minimalna dopuszczalna wartość promienia $R2$
S	powierzchnia ciała
S^e	powierzchnia elementu skończonego
s	brzeg ciała

s_i	część i brzegu ciała
s^e	brzeg elementu skończonego
s_i^e	część i brzegu elementu skończonego
T	chromosom osobnika populacji bazowej (wektor parametrów geometrycznych)
T^i	chromosom dla którego funkcja $F(T)$ osiągnie optimum w pokoleniu i
T^*	chromosom dla którego funkcja $F(T)$ osiągnie optimum globalne
t'	wartość genu osobnika populacji potomnej
T'_i	i -ty osobnik (chromosom) populacji potomnej
$T_{0-\beta}$	różnica temperatury na brzegu (T_β) i temperatury otaczającego ciała ośrodka (T_0)
T_C	temperatura ciała
T_C^e	elementowy wektor temperatur węzłowych (składowe T_{Cn})
T_{C-REF}	różnica pomiędzy temperaturą ciała (T_C) i temperaturą odniesienia
t_i	i -ty gen chromosomu (zmienna projektowa)
T_i	i -ty chromosom (osobnik) populacji bazowej
T_{REF}	temperatura odniesienia
u	pole przemieszczeń (składowe u, v, w)
\dot{u}	pole prędkości (składowe $\dot{u}, \dot{v}, \dot{w}$)
\ddot{u}	pole przyspieszeń
U_A	globalny wektor amplitud (składowe u_A, v_A, w_A)
u^e	elementowy wektor przemieszczeń węzłowych (składowe u_n, v_n, w_n)
\dot{u}^e	elementowy wektor prędkości węzłowych
\ddot{u}^e	elementowy wektor przyspieszeń węzłowych
U	globalny wektor przemieszczeń węzłowych
\dot{U}	globalny wektor prędkości węzłowych
\ddot{U}	globalny wektor przyspieszeń węzłowych
V	objętość ciała

V_0	objętość optymalizowanej części pióra łopatki modelu w konfiguracji „0”
$V(T)$	objętość części optymalizowanej obliczona dla osobnika o chromosomie T
V^e	objętość elementu skończonego
v_i	średnia szybkość poprawy dla i -tego pokolenia obliczana na przedziale d
v_{min}	minimalna dopuszczalna szybkość poprawy rozwiązania
W	łączny czas obliczeń
w_i	średni czas oceny wartości funkcji przystosowania dla i -tego pokolenia
X	tensor sił masowych (składowe X_i)
Y_i	i -ty osobnik populacji tymczasowej v^i
y_i^j	i -ty gen osobnika populacji tymczasowej o chromosomie Y_j
z_i	i -ty gen osobnika z populacji tymczasowej ζ^i
Z_i	i -ty osobnik populacji tymczasowej po krzyżowaniu
α	kąt rozwarcia łuku o promieniu RI
β	kąt pomiędzy dwoma krzywymi łączonymi łukiem RI
$\sigma_1(T)$	maksymalne naprężenie główne na optymalizowanej części pióra łopatki dla osobnika o chromosomie T
σ_{10}	maksymalne naprężenie główne na optymalizowanej części pióra łopatki modelu w konfiguracji „0”
λ	stała <i>Lamego</i>
γ	współczynnik przejmowania ciepła
ω	częstość w radianach
Δ	operator różniczkowy
σ	wektorowa postać tensora naprężeń (składowe σ_{ij})
ε	wektorowa postać tensora odkształceń całkowitych (składowe ε_{ij})
α	wektor współczynników rozszerzalności cieplnej (składowe α_{ij})
Ω	przestrzeń (wektorowa) rozwiązań dopuszczalnych
τ^i	populacja bazowa pokolenia i

v^i	populacja tymczasowa pokolenia i po reprodukcji
ζ^i	populacja tymczasowa pokolenia i po krzyżowaniu
$\delta\epsilon$	wektor odkształceń przygotowanych (składowe $\delta\epsilon_{ij}$)
τ^i	populacja potomna pokolenia i
ϵ^C	wektorowa postać tensora odkształceń od obciążeń cieplnych (składowe ϵ^T_{ij})
$\xi_G(g_1, g_2)$	liczba losowana z rozkładem Gaussa, gdzie g_1 jest wartością oczekiwaną a g_2 odchyleniem standardowym
κ_i	przewodność cieplna w kierunku i
δ_{ij}	delta <i>Kroneckera</i> ($\delta_{ij}=1$ dla $i=j$, $\delta_{ij}=0$ dla $i\neq j$)
v_{jn}	cosinus kierunkowy między normalną zewnętrzną n do powierzchni obciążonej i półosią x_j układu współrzędnych.
ϵ^M	wektorowa postać tensora odkształceń od obciążeń mechanicznych (składowe ϵ^M_{ij})
$\bar{M}(T)$	znormalizowana masa modelu o chromosomie T
δU	praca przygotowana wykonana przez siły zewnętrzne
δu	pole przemieszczeń przygotowanych (składowe δu , δv , δw)
$\xi_U(r_1, r_2)$	liczba losowana z rozkładem normalnym przy górnej i dolnej granicy przedziału dopuszczalnego odpowiednio r_1, r_2
δu^e	elementowy wektor węzłowych przemieszczeń przygotowanych (składowe δu_n , δv_n , δw_n)
$\bar{\sigma}(T)$	znormalizowane maksymalne naprężenie główne dla osobnika o chromosomie T

1. WPROWADZENIE

1.1. Cel i zakres pracy

Sytuacja na rynku przewozów lotniczych powoduje ostrzejszą niż dotychczas konkurencję wśród producentów silników turbowentylatorowych. Wymusza to na nich nieustanne doskonalenie metod projektowania i narzędzi inżynierskich jakimi posługują się biura konstrukcyjne. Jednym ze stosunkowo mało zbadanych i wykorzystanych zagadnień pozostaje optymalizacja wytrzymałościowa kształtu łopatek turbin. Na szerszą skalę prowadzone są tylko prace dotyczące optymalizacji kształtu kanałów przepływowych i kształtu łopatek pod kątem poprawy ich parametrów aerodynamicznych, co wynika z analitycznego lub numeryczno-analitycznego charakteru rozwiązania zadania mechaniki płynów oraz względnej łatwości sformułowania funkcji celu w oparciu o bilans energetyczny [9]. Brak jest natomiast opracowań dotyczących problemu poprawy wytrzymałości łopatek i innych elementów turbin poprzez optymalizację ich kształtu.

Zwykle w trakcie projektowania łopatki wykonuje się jedynie optymalizację jej położenia osiowego względem gniazda na dysku, w celu minimalizacji sił na zamku jodełkowym oraz przebiegu kąta pochylenia osiowego (ang. *tilt*) i obwodowego (ang. *leaning*) łopatki w funkcji promienia, w celu kompensacji momentów gnących pochodzących od sił bezwładności i sił gazu. Oba te zadania optymalizacji przeprowadzane są jednak przy założeniu niezmienności przekroju łopatki w płaszczyźnie normalnej do promienia, tj. w przypadku minimalizacji sił na zamku przesuwa się łopatkę w całości względem półki dolnej, natomiast w przypadku optymalizacji kątów operuje się na kilku-kilkunastu przekrojach łopatki przesuwanym jako niezmiennie w płaszczyźnie obwodowo-osiowej. Nie dopuszcza się tym samym rzeczywistej zmiany kształtu przekroju łopatki, który w strefie przejścia pióro-półka dolna/górna mógłby zostać zmodyfikowany bez szkody dla parametrów aerodynamicznych łopatki, a ze znaczącym obniżeniem masy i poziomu naprężeń, które decydują o wytrzymałości całej łopatki.

Obecnie żaden z liczących się producentów silników lotniczych nie prowadzi takiej optymalizacji na skalę przemysłową, mimo że jej efekty znalazłyby natychmiastowe odbicie w parametrach decydujących o konkurencyjności silnika na rynku (masa, moment bezwładności rotora turbiny, obciążenie dysku, wymagania wobec instalacji rozruchowej, itp.). Efekt redukcji masy łopatki byłby tym bardziej istotny, że na jeden rotor przypada ich kilkadziesiąt (najczęściej ponad sto).

Powodem tego stanu rzeczy jest skomplikowany i długotrwały proces projektowania łopatk. Zarówno parametryzacja modelu geometrycznego jak i optymalizacja z wykorzystaniem tradycyjnych metod jest trudna i czasochłonna w stopniu wykluczającym wykorzystanie przemysłowe. Typowa łopatka turbiny jest bowiem elementem wymagającym bardzo złożonej symulacji inżynierskiej. Jej kształt jest zdeterminowany przez funkcje aerodynamiczne, ma ona bezpośredni kontakt z gorącym strumieniem gazów, działa w warunkach wysokich temperatur i obciążeń siłą odśrodkową. Ostateczna jej forma stanowi kompromis pomiędzy wymaganiami podyktowanymi sprawnością aerodynamiczną, dotyczącymi charakterystyki dynamicznej i wytrzymałości doraźnej, wytrzymałości na zmęczenie nisko- i wysokocyklowe, odporności na pękanie, wymaganiami technologicznymi, itp. Wymusza to zastosowanie najbardziej zaawansowanych narzędzi projektowania. W procesie projektowania łopatek wykorzystuje się powszechnie narzędzia wspomagane komputerowo, zarówno w trakcie tworzenia modeli geometrycznych (narzędzia *CAD* – *Computer Aided Design*), obliczeń i analiz inżynierskich (narzędzia *CAE* – *Computer Aided Engineering*) jak i przygotowania procesu technologicznego oraz samej produkcji (narzędzia *CAM* – *Computer Aided Manufacturing*). Najważniejszym narzędziem obliczeniowym jest *Metoda Elementów Skończonych (MES)*, stanowiąca od momentu upowszechnienia w zastosowaniach inżynierskich uznany w środowisku producentów turbin lotniczych standard, akceptowany również przez instytucje decydujące o dopuszczeniu silników do eksploatacji. Obecnie coraz większa część procesu certyfikacji silnika lotniczego i poszczególnych jego elementów oparta jest wyłącznie o wyniki obliczeń *MES* i stopniowo ogranicza się zakres wymaganych doświadczalnych badań stanowiskowych. **LIT?**

Problemy techniczne utrudniające zastosowanie na skalę przemysłową optymalizacji kształtu łopatek można podzielić na kilka kategorii:

1. Problemy związane z parametryzacją modeli geometrycznych łopatek (wykorzystujących opis powierzchniami typu *B-surface* i krzywymi typu *spline* [33]).
2. Problemy związane ze sformułowaniem funkcji celu i zapewnieniem warunków koniecznych do wykorzystania tradycyjnych algorytmów optymalizacyjnych (ciągła funkcja celu, wąski zakres zmienności parametrów, itp.).
3. Konieczność funkcjonowania procedury optymalizacyjnej w środowisku komercyjnego oprogramowania do obliczeń inżynierskich (ocena wartości funkcji celu może się odbywać tylko z wykorzystaniem *MES* ze względu na złożoność zadania i konieczność integracji z całym procesem projektowania łopatk).
4. Czas trwania obliczeń oraz zbieżność algorytmu optymalizacyjnego.

Zaproponowana metoda musi gwarantować uzyskanie pożądaných wyników w z góry określonym, jak najkrótszym czasie. Dodatkowo pożądané byłoby wykorzystanie oprogramowania inżynierskiego już powszechne stosowanego w przemyśle silników lotniczych, korzystającego ze standardowych danych wejściowych (modele geometryczne, dane materiałowe, opis warunków brzegowych, itp.). Spowoduje to znaczące ograniczenie czasu przygotowania danych dla procedury optymalizacyjnej, co w warunkach tak złożonej analizy ma duże znaczenie, pozwoli również na ograniczenie jej kosztów (koszt oprogramowania i ew. szkolenia pracowników), minimalizację czasu analizy wyników, tym samym umożliwiając natychmiastowe włącznie optymalizacji w stosowany dotychczas proces projektowania.

Obok klasycznych metod optymalizacji opartych na znajomości gradientu funkcji celu (metoda najszybszego spadku, metoda gradientów sprzężonych, metody zmiennej metryki LIT?) pojawiły się w ostatnich latach metody inspirowane zjawiskami i procesami biologicznymi do których należą algorytmy ewolucyjne i immunologiczne [7], [15], [29]. Metody te zaliczane są do metod optymalizacji globalnej i nie wymagają znajomości gradientu funkcji celu. Niniejsza rozprawa doktorska jest propozycją metody optymalizacji kształtu łopatek opartą na algorytmach ewolucyjnych sprzężonych z metodą elementów skończonych, która zapewnia możliwość uwzględnienia złożonego stanu obciążeń turbiny, także w zakresie nieliniowym, pozostawiając jednocześnie pełną swobodę wyboru funkcji celu (w tym przypadku funkcji przystosowania) i narzucenia dowolnych ograniczeń na parametry geometryczne. Metoda ta spełnia wszystkie wymienione wcześniej wymagania, stanowiąc dzięki temu narzędzie bardzo atrakcyjne dla zastosowań inżynierskich.

Pierwszym etapem optymalizacji jest parametryzacja części modelu geometrycznego łopatki turbiny i jego zapis w postaci wektora parametrów geometrycznych. Po losowym wygenerowaniu rodziny wektorów reprezentujących różne konfiguracje geometryczne łopatki określane mianem "osobników", są one realizowane jako modele geometryczne, a następnie modele dyskretnie metody elementów skończonych. Dla każdej konfiguracji rozwiązywane jest zadanie *MES* z warunkami brzegowymi odpowiadającymi warunkom roboczym turbiny, dostarczając informacji na temat wartości funkcji przystosowania danego osobnika, czyli jego jakości z punktu widzenia celu optymalizacji. Parametry geometryczne najlepszych osobników są potem przetwarzane zgodnie z algorytmem ewolucyjnym w celu uzyskania kolejnej rodziny konfiguracji, przeprowadzana jest znowu generacja modeli, analiza *MES* i ocena jakości rozwiązań. Proces ten jest powtarzany aż do spełnienia kryterium zakończenia obliczeń. Całość procedury optymalizacyjnej zrealizowana została jako program w języku *APDL* (*Ansys Parameter Design Language*) i działa w środowisku *Ansys* [2] stanowiącym powszechny standard w przemyśle napędów lotniczych.

Praca doktorska obejmie przeprowadzenie obliczeń dla trzech różnych postaci funkcji przystosowania, mających na celu kolejno minimalizację masy, maksymalnego naprężenia głównego oraz równoczesną minimalizację masy i naprężenia na części łopatki turbiny. Dostarczą one informacji na temat skuteczności i zbieżności algorytmu ewolucyjnego oraz kosztu numerycznego optymalizacji, które są najważniejszymi z punktu widzenia zastosowań przemysłowych.

Zrealizowanie wszystkich założeń pracy pozwoli na opracowanie metody optymalizacji kształtu łopatek, która może być bardzo atrakcyjna dla przemysłu i będzie stanowić istotny krok w kierunku udoskonalenia tradycyjnego procesu projektowania elementów silników lotniczych.

W trakcie pracy wykorzystano modele geometryczne, warunki brzegowe, dane materiałowe i sprzęt oraz oprogramowanie dostarczone przez firmę *Avio Polska*. Ponieważ zagadnienie dotyczy przemysłowego projektu łopatki rotora turbiny będącego obecnie w fazie wdrożenia, a przedstawione tu informacje mają charakter poufny, wszystkie dane i wyniki zostały znormalizowane względem wartości otrzymanych dla modelu w pierwotnej, wyjściowej konfiguracji geometrycznej (oznaczonej jako konfiguracja „0”).

1.2. Organizacja pracy

Pracę podzielono na 12 rozdziałów poświęconych algorytmom ewolucyjnym, metodzie elementów skończonych, budowie modelu symulacyjnego łopatki turbiny przemysłowej oraz opisowi procesu i wyników optymalizacji.

Na początku pracy znajduje się spis treści, spis ilustracji, spis tabel oraz spis oznaczeń.

Rozdział 1 „Wprowadzenie” zawiera ogólny opis problematyki pracy i poruszanych w niej zagadnień, ich genezę, a także ogólne omówienie zawartości.

Rozdział 2 formułuje tezę pracy.

Rozdział 3 „Algorytmy ewolucyjne w optymalizacji konstrukcji” poświęcony jest omówieniu genezy i zastosowania algorytmów ewolucyjnych w optymalizacji konstrukcji, a także zawiera wyjaśnienie podstawowych pojęć i omówienie najpopularniejszych metod wykorzystywanych w algorytmach ewolucyjnych.

Rozdział 4 „Środowisko i narzędzia optymalizacji łopatki turbiny przemysłowej” zawiera podstawowe informacje na temat wykorzystanego w pracy oprogramowania do analizy wytrzymałościowej za pomocą *MES*, oprogramowania stworzonego na potrzeby obsługi procesu optymalizacji, sposobu wymiany danych i platformy sprzętowej.

Rozdział 5 „Parametryzacja modelu geometrycznego łopatki” omawia sposób parametryzacji modelu geometrycznego łopatki i związane z nim ograniczenia tworzące

przestrzeń rozwiązań dopuszczalnych oraz zapis wektora parametrów geometrycznych jako genotypu.

Rozdział 6 „Model termomechaniczny łopatki turbiny przemysłowej” zawiera opis modelu symulacyjnego *MES* rotora turbiny przemysłowej a w szczególności łopatki z optymalizowaną strefą. Zostały w nim omówione: sformułowanie zadania brzegowego teorii termosprężystości i oparty na nim model matematyczny, dyskretyzacja modelu, podstawowe równania *MES*, opis wykorzystanych elementów skończonych, założenia upraszczające, warunki brzegowe i własności materiałów.

Rozdział 7 „Ewolucyjna optymalizacja jednokryterialna kształtu łopatki ze względu na minimum maksymalnego naprężenia głównego” prezentuje sformułowanie zadania optymalizacji jednokryterialnej ze względu na minimum naprężenia głównego, szczegółowy opis algorytmu ewolucyjnego, otrzymane wyniki i ich omówienie oraz porównanie modelu zoptymalizowanego z modelem wyjściowym.

Rozdział 8 „Ewolucyjna optymalizacja jednokryterialna kształtu łopatki ze względu na minimum masy” prezentuje sformułowanie zadania optymalizacji jednokryterialnej ze względu na minimum masy, opis algorytmu ewolucyjnego, otrzymane wyniki i ich omówienie oraz porównanie modelu zoptymalizowanego z modelem wyjściowym.

Rozdział 9 „Ewolucyjna optymalizacja dwukryterialna kształtu łopatki ze względu na minimum masy i maksymalnego naprężenia głównego” prezentuje sformułowanie zadania optymalizacji dwukryterialnej, opis algorytmu ewolucyjnego, otrzymane wyniki i ich omówienie oraz porównanie modelu zoptymalizowanego z modelem wyjściowym a także z modelami otrzymanymi w wyniku optymalizacji jednokryterialnych.

Rozdział 10 „Wpływ zmiany kształtu łopatki w strefie optymalizowanej na częstość drgań własnych łopatki” omawia analizę wpływu zmiany kształtu łopatki na częstość jej drgań własnych dla pierwszych 10 form modalnych, a także skrótowy opis modelu matematycznego i modelu *MES* wykorzystanego obliczeń.

Rozdział 11 „Podsumowanie i wnioski końcowe” zawiera podsumowanie otrzymanych wyników, wnioski oraz omówienie możliwości i celowości kontynuowania prac związanych z ewolucyjną optymalizacją kształtu łopatek turbin silników przemysłowych i lotniczych.

Rozdział 12 „Streszczenie (Summary)” zawiera streszczenia pracy w języku polskim i angielskim.

Dodatek 1 - Kod źródłowy programu optymalizacyjnego (APDL) - zawiera kod programu optymalizacyjnego w języku *APDL* z komentarzami.

Dodatek 2 - Struktura plików wejścia-wyjścia programu optymalizacyjnego – zawiera opis struktury plików wykorzystywanych przez program optymalizacyjny.

2. TEZA PRACY

Sformułowana w oparciu o spostrzeżenia zawarte w rozdziale pierwszym, teza pracy brzmi następująco:

„Zastosowanie optymalizacji kształtu opartej na algorytmie ewolucyjnym sprzężonym z metodą elementów skończonych pozwala na znaczne obniżenie masy i maksymalnego naprężenia głównego w strefie przejścia platforma – pióro łopatki turbiny przemysłowej w stosunku do rozwiązania znalezionego metodami tradycyjnymi.”

3. ALGORYTMY EWOLUCYJNE W OPTYMALIZACJI KONSTRUKCJI

3.1. Sformułowanie zadania optymalizacji ewolucyjnej

Zadania optymalizacji (minimalizacji) polegają na przeszukaniu pewnej przestrzeni rozwiązań dopuszczalnych Ω , w której została zdefiniowana funkcja $F(T)$ zwana funkcją celu. Szukane jest takie $T^* \in \Omega$ aby zachodził związek:

$$\forall T \in \Omega \quad F(T^*) \leq F(T) \quad (3.1)$$

W ogólnym przypadku zadanie to jest nierozwiązywalne, ponieważ wymaga sprawdzenia wartości $F(T)$ w każdym punkcie przestrzeni Ω , co dla przestrzeni ciągłych jest niemożliwe, a dla dyskretnych najczęściej niewykonalne ze względu na zbyt dużą ilość punktów. Dodatkowo obowiązuje tzw. „przekleństwo wymiarowości”, w przestrzeni kartezjańskiej liczba punktów w sąsiedztwie o ustalonym promieniu wokół dowolnego punktu rośnie wykładniczo. Aby rozwiązać problem, trzeba poczynić szereg założeń upraszczających które umożliwiają osłabienie warunku 3.1. Narzucają one na przykład określoną klasę funkcji $F(T)$ lub określają maksymalną ilość operacji jakie mają zapewnić znalezienie rozwiązania. Można też posłużyć się algorytmem opartym na doświadczeniu i „inteligentnej” regule postępowania, które określa się też mianem algorytmów „niepełnowartościowych”, umożliwiających znalezienie w akceptowalnym czasie rozwiązania problemu dostatecznie bliskiego rozwiązaniu rzeczywistemu.

Algorytmy ewolucyjne należą do tej ostatniej kategorii, są rodzajem tzw. „metaheurystyk” (z greckiego „heuriskein” – znaleźć, odkryć). Jako pierwszy zaproponował algorytm ewolucyjny John Holland w 1975r., był to tzw. „prosty algorytm genetyczny” służący do modelowania procesu ewolucji. Rozwiązanie poszukiwane jest poprzez modyfikację i przetwarzanie rozwiązań już istniejących, a jego jakość sprawdzana jest „empirycznie” („rozpoznanie walką”), w praktyce jest to metoda prób i błędów w której błędy są eliminowane a sukcesy kumulowane [3], [18].

Wykorzystanie mechanizmu zjawiska biologicznego spowodowało przejście z niego również terminologii stosowanej do opisu procesu. Elementy algorytmu optymalizacyjnego mają swoje odpowiedniki w elementach występujących w środowisku naturalnym.

Środowisko w którym zachodzą procesy ewolucji stanowi funkcja celu (w tym przypadku nazywana „*funkcją przystosowania*”) i przestrzeń rozwiązań dopuszczalnych. Algorytm ewolucyjny przetwarza iteracyjnie zbiór (lub kilka zbiorów) proponowanych rozwiązań problemu stanowiący „*populację*” złożoną z „*osobników*”. Każdy osobnik reprezentowany jest przez zbiór cech stanowiących „*genotyp*” złożony z „*chromosomów*” (najczęściej jeden genotyp odpowiada jednemu chromosomowi). Genotyp złożony jest z elementarnych jednostek, tzw. „*genów*”. Jednorazowo w środowisku ewolucji funkcjonuje jedna grupa osobników nazywana „*pokoleniem*”, zawierająca populację na bieżącym poziomie ewolucji. Każdy osobnik reprezentowany jest przez obliczoną dla niego wartość funkcji przystosowania, która pozwala stwierdzić jak „dobry” jest on z punktu widzenia poszukiwanego rozwiązania problemu.

Odpowiedniki tych elementów w opisywanym tutaj procesie ewolucyjnej optymalizacji kształtu łopatki turbiny są następujące:

1. Gen – parametr geometryczny modelu optymalizowanej części łopatki.
2. Chromosom (genotyp) – wektor parametrów geometrycznych wystarczający do jednoznacznego opisu modelu (konfiguracja geometryczna modelu).
3. Osobnik – model geometryczny optymalizowanej części łopatki będący realizacją chromosomu.
4. Populacja bazowa – zbiór osobników (modeli geometrycznych) dla których określone są wartości funkcji przystosowania.
5. Populacja tymczasowa – zbiór osobników na których przeprowadzane są operacje genetyczne.
6. Populacja potomna – zbiór osobników, które stają się podstawą populacji bazowej kolejnego pokolenia.
7. Pokolenie – populacja na określonym etapie ewolucji.
8. Funkcja przystosowania – odpowiednik funkcji celu, funkcja obliczana dla każdego osobnika, której wartość określa jak „dobry” jest on z punktu widzenia poszukiwanego rozwiązania problemu.

Dla potrzeb optymalizacji kształtu łopatki turbiny wybrano dla chromosomu reprezentację rzeczywistoliczbową, gdzie każdy gen jest wprost interpretowany jako wartość parametru geometrycznego modelu łopatki. Chromosom reprezentujący model geometryczny można więc przedstawić jako wektor złożony ze zmiennych projektowych (genów):

$$\mathbf{T}_j = \{ t_1, t_2, \dots, t_n \} \quad (3.2)$$

gdzie:

T_j – j -ty osobnik (chromosom) populacji bazowej
 t_i – i -ty gen (parametr geometryczny, zmienna projektowa)

Każda populacja bazowa danego pokolenia składa się z K osobników i może być zapisana jako:

$$\tau^k = [T_1, T_2, \dots, T_K] \quad (3.3)$$

gdzie:

τ^k – populacja bazowa k -tego pokolenia,

Populacja bazowa pierwszego pokolenia generowana jest losowo. Losowanie wartości genów przeprowadzane jest do momentu utworzenia K osobników. Każdy chromosom jest następnie przekazywany jako dane wejściowe do procedury optymalizacyjnej, która po zbudowaniu i modelu obliczeniowego strefy optymalizowanej i analizie *MES* podaje wartość jego funkcji przystosowania. Po zbudowaniu wektora wartości funkcji przystosowania dla całej populacji bazowej bieżącego pokolenia następuje generacja populacji potomnej. Składa się ona z trzech etapów: reprodukcji, operacji genetycznych i sukcesji. Operacje reprodukcji i sukcesji określa się czasem wspólną nazwą selekcji [6]. Schemat ten naśladuje naturalny proces rozmnażania, przeprowadzając operacje na chromosomach najlepszych osobników populacji bazowej i krzyżując je na podobieństwo łańcuchów *DNA* w naturze.

Wyróżnia się dwie odmiany selekcji. W pierwszej z nich populacja bazowa kolejnego pokolenia powstaje wyłącznie z osobników populacji potomnej. W drugiej, wykorzystanej w niniejszej pracy tzw. selekcji elitarniej, populacja bazowa powstaje zarówno z osobników populacji potomnej, jak i z najlepszych osobników poprzedniej populacji bazowej. W ten sposób zagwarantowane zostaje przeżycie najlepszych osobników poprzedniego pokolenia, dzięki temu funkcja przystosowania przybiera wartości niemalejące (przy omawianych tutaj zagadnieniach minimalizacji). Wykorzystany schemat selekcji elitarniej gwarantuje przetrwanie jednego, najlepszego osobnika populacji bazowej.

Pierwszym etapem selekcji jest reprodukcja. Wyróżnia się wiele typów reprodukcji (proporcjonalna, progowa, rangowa, turniejowa) różniących się przede wszystkim sposobem określenia prawdopodobieństwa wyboru osobnika do populacji tymczasowej. Prawdopodobieństwo to jest jednak z reguły uzależnione od wartości funkcji przystosowania danego osobnika. W niniejszej pracy wykorzystano schemat reprodukcji progowej. Prawdopodobieństwo wyboru osobnika do populacji tymczasowej przybiera wartość 1 lub 0 zależnie od tego czy wartość funkcji przystosowania osobnika jest mniejsza czy większa od średniej dla populacji bazowej. W wyniku reprodukcji do dalszych etapów przetwarzania

chromosomu wybranych zostaje $K/2$ osobników o najniższych wartościach funkcji przystosowania, stają się one populacją tymczasową ν w danym pokoleniu.

Populacja tymczasowa poddawana jest następnie działaniu operatorów genetycznych. Za ich pomocą przeprowadzane jest krzyżowanie i mutacja.

Krzyżowanie jest operacją w wyniku której na podstawie populacji tymczasowej $K/2$ osobników po reprodukcji zostaje utworzona nowa populacja tymczasowa licząca $K-1$ osobników. W tym celu przetwarzane są chromosomy osobników po reprodukcji. Wykorzystuje się wiele typów krzyżowania, dzieląc je zasadniczo na wymieniające (jednopunktowe, wielopunktowe, równomierne) w którym chromosomy osobników rodzicielskich są rozcinane i składane na nowo w chromosomy osobników potomnych zachowując oryginalne wartości genów, lub uśredniające gdzie wartości genów osobników potomnych są funkcjami wartości genów osobników rodzicielskich. W obu metodach wyboru osobników rodzicielskich dokonuje się w sposób losowy, losowe mogą być także punkty rozcięcia chromosomów rodzicielskich lub funkcje określające wartości genów uśrednionych.

W niniejszej pracy wykorzystano schemat krzyżowania wymieniającego z dwoma losowo wybranymi i jednym stałym punktem rozcięcia oraz z trzema osobnikami rodzicielskimi. Wynika to w sposób naturalny z budowy chromosomu (szczegółowe omówienie w rozdziale 5 i 7.2). Osobniki rodzicielskie losowane są z zestawu $K/2$ osobników populacji tymczasowej, zagwarantowany został brak możliwości powtórzenia się ich kombinacji w obrębie jednego pokolenia. Po losowym wyborze punktów rozcięcia chromosomów rodzicielskich, jest z nich składany jeden chromosom nowej populacji tymczasowej. Operacja ta powtarzana $K-1$ razy.

Kolejną operacją procesu generacji populacji potomnej jest mutacja, która polega na losowej zmianie wartości dwóch losowo wybranych genów każdego osobnika populacji tymczasowej. Zakres zmiany wartości określany jest losowo i wynosi $\pm 10\%$. W ten sposób tworzona jest populacja potomna.

Ostatnia operacja prowadząca do powstania nowej populacji bazowej, to sukcesja. Polega ona na uzupełnieniu populacji potomnej złożonej z $K-1$ osobników o chromosom najlepszego osobnika poprzedniej populacji bazowej. W ten sposób powstaje pełna populacja bazowa kolejnego pokolenia złożona z K osobników. Po utworzeniu nowej populacji bazowej przeprowadzana jest kontrola położenia genów w przedziałach dopuszczalnych i ewentualna modyfikacja ich wartości do ich granic.

Operacje na chromosomach nie kończą się jednak w momencie wygenerowania populacji bazowej. Ze względu na charakter kodowania chromosomu (zapis rzeczywistoliczbowy odpowiadający bezpośrednio wektorowi zmiennych projektowych) oraz kształt strefy optymalizowanej, geny są ze sobą powiązane – także w sposób niejawni. Należy sprawdzić wykonalność chromosomów jako modeli geometrycznych i w razie

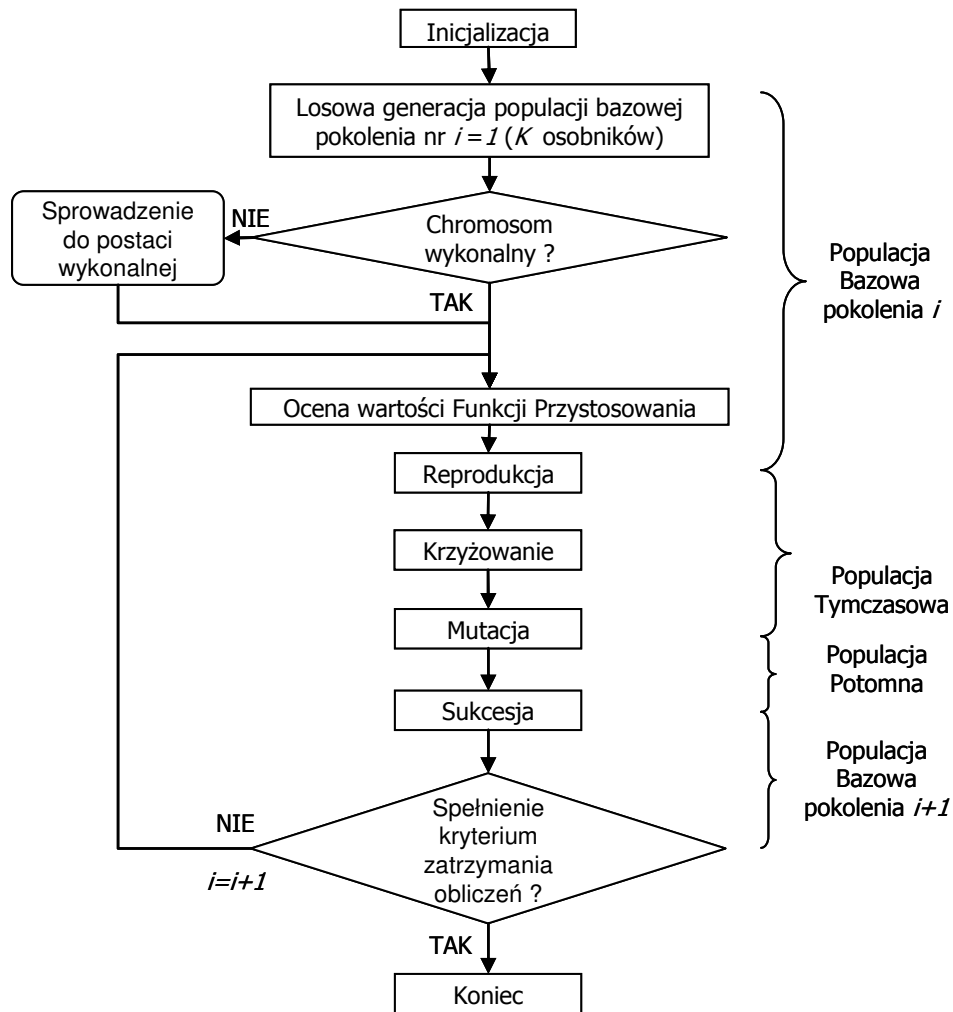
potrzeby zmodyfikować wartości genów które uniemożliwiają realizację chromosomu. W trakcie generacji modelu sprawdzany jest szereg warunków (położenie punktów krzywych i krawędzi powierzchni opisujących optymalizowaną bryłę, itp.). W razie potrzeby uruchamiany jest mechanizm modyfikacji chromosomu (opisany szczegółowo w rozdziale 5.2.), którego zadaniem jest zapewnienie sprowadzenie chromosomu do postaci wykonalnej przy jak najmniejszej zmianie oryginalnych wartości jego genów. Proces ten przeprowadzany jest iteracyjnie.

Jednym z elementów algorytmu ewolucyjnego, oprócz schematu selekcji i operatorów genetycznych oraz sposobu sprowadzania chromosomu do postaci wykonalnej, jest także kryterium zatrzymania obliczeń. Stosuje się różne kryteria (minimalnej szybkości poprawy rozwiązania, minimalnej ilości poprawionych rozwiązań, różnorodności populacji bazowej, czasu obliczeń, itp.) a ich wybór podyktowany jest zawsze specyfiką zagadnienia i charakterem przebiegu funkcji przystosowania. W niniejszej pracy zastosowano kryteria oparte na szybkości poprawy funkcji kształtu dla optymalizacji jednokryterialnych, uzupełnione o kryterium minimalnej ilości poprawionych rozwiązań dla optymalizacji dwukryterialnej.

Ogólny schemat działania algorytmu ewolucyjnego przedstawia rys. 3.1. Wszystkie elementy algorytmu ewolucyjnego opisane są szczegółowo w rozdziałach poświęconych przedstawieniu wyników optymalizacji ewolucyjnej (najszerzej w rozdziale 7). Parametryzacja modelu i budowa chromosomu jako wektora zmiennych projektowych omówiona jest w rozdziale 5.

W naturze proces ten jest o wiele bardziej złożony, środowisko ewolucji ulega ciągłej zmianie (funkcja przystosowania nie jest stała, lecz ustalona bieżącymi warunkami panującymi w środowisku), oprócz mutacji genowych zachodzą także mutacje chromosomowe (translokacja, duplikacja, inwersja), dochodzi do interakcji między populacjami egzystującymi w różnych warunkach, działają czynniki losowe prowadzące na przykład do eliminacji osobników dobrze przystosowanych, itp. Informacja przekazywana w procesie rozmnażania nie ogranicza się do informacji „genotypowej” zawartej w chromosomie, lecz jest wzbogacona o informację „makroskopową”, będącą wiedzą nabytą osobnika przejawiającą się w adaptacji, walce o byt, itp., która ulega ciągłemu wzbogaceniu. „Wiedza ewolucji jest empiryczna i doraźna, a swą pozorną doskonałość zawdzięcza olbrzymim otchłaniom przestrzeni i czasu które przemierzyła, w których więcej było jednak, jeśli próbować bilansowania, klęsk niż sukcesów” (S. Lem, *Summa Technologiae*).

Algorytm ewolucyjny naśladuje w sposób maksymalnie uproszczony proces przetwarzania informacji jaki zachodzi w naturze, jest obciążony „naturalnymi” błędami”, a mimo to okazuje się narzędziem bardzo efektywnym. Z punktu widzenia zastosowania związanego z optymalizacją, zasadniczą kwestią jest jego zbieżność.



Rys. 3.1 Ogólny schemat blokowy procesu optymalizacji ewolucyjnej.

Zbieżność algorytmu ewolucyjnego może być rozważana jedynie w kategoriach probabilistycznych, ponieważ jest on techniką niedeterministyczną. Zostało sformułowane twierdzenie o zbieżności, które mówi że po odpowiednio długim czasie populacja bazowa zawierać będzie chromosomy o przystosowaniu zbliżonym do poszukiwanego maksimum globalnego [6]. Jeśli przez T^* oznaczmy poszukiwany chromosom, dla którego otrzymamy maksimum globalne funkcji przystosowania, a zbiór poziomowy funkcji przystosowania przez:

$$L_F(c) = \{T \in \Omega \mid F(T) > c\} \quad (3.4)$$

gdzie:

$L_F(c)$ – zbiór poziomowy funkcji przystosowania określony na poziomie c ,

T – chromosom populacji bazowej,
 Ω – przestrzeń rozwiązań dopuszczalnych,
 $F(T)$ – wartość funkcji przystosowania dla chromosomu T .

oraz wprowadzimy:

$$T^i = \arg \max_{T \in \tau^i} F(T) \quad (3.5)$$

gdzie:

T^i – chromosom dla którego funkcja $F(T)$ osiągnie maksimum w pokoleniu i ,
 τ^i – populacja bazowa pokolenia i .

to jeśli spełnione będą następujące założenia:

1. Istnieje takie $c > 0$, że $\forall d < c$ $L_F(d)$ jest niezerowej miary.
2. Jeśli Ω jest zbiorem skończonym, to T^* (maksimum globalne) nie należy do jego brzegu.
3. Zachodzi $|T^*| < \infty$ i $F(T^*) < \infty$ (gdzie $|T^*|$ oznacza metrykę chromosomu T^*).
4. Stosuje się algorytm ewolucyjny z selekcją elitarną (zapewniającą przeżycie co najmniej najlepszemu osobnikowi).
5. Stosuje się mutację zapewniającą osiągnięcie każdego genotypu z prawdopodobieństwem większym od zera.

dla dowolnej populacji początkowej τ^1 , takiej że $\forall T \in \tau^1$ $|T| < \infty$

i dla każdego $\varepsilon > 0$ zachodzi:

$$\lim_{i \rightarrow \infty} P[T^i \in L_F(F(T^*) - \varepsilon)] = 1 \quad (3.6)$$

gdzie:

$P()$ – dystrybuanta funkcji

Czyli przy odpowiednio dużej liczbie pokoleń dąży do jedności prawdopodobieństwo odwiedzenia zbioru poziomicowego dowolnie bliskiego maksimum globalnemu.

Twierdzenie to najczęściej udowadnia się korzystając z twierdzenia *Borela – Cantellego* o zbieżności ciągu zdarzeń losowych. Należy zauważyć że zbieżność algorytmu ewolucyjnego ma charakter asymptotyczny i probabilistyczny, nie może więc służyć na

przykład do sformułowania kryterium zatrzymania algorytmu, ale założenia jakie należy spełnić są łatwe do zrealizowania i realistyczne [6].

Mocną stroną algorytmu ewolucyjnego jest jego uniwersalność i stosunkowo niewielkie wymagania wobec naszej wiedzy o problemie. W praktyce musimy określić tylko sposób kodowania genotypu (sparametryzować model geometryczny w przypadku optymalizacji kształtu i określić sposób zapisu wektorowego), określić metodę reprodukcji i sukcesji, zdefiniować operatory genetyczne (krzyżowanie, mutacja) zapewnić możliwość oceny funkcji przystosowania i zdefiniować przestrzeń rozwiązań dopuszczalnych.

W przypadku rozwiązań specyficznych problemów, takich jak na przykład optymalizacja kształtu, warunki te są stosunkowo łatwe do zrealizowania. Największym problemem może się okazać parametryzacja modelu i zapewnienie realizacji chromosomu w postaci modelu geometrycznego, ponieważ dowolny zestaw parametrów z przestrzeni dopuszczalnej niekoniecznie musi być wykonalny, a parametry (które mogą być ze sobą niejawnie powiązane) mogą się wykluczać. Jest to etap kluczowy dla całego procesu optymalizacji, ponieważ sposób parametryzacji modelu decyduje o kształcie strefy optymalizowanej, przebiegu późniejszej ewolucji i postaci ostatecznego rozwiązania. Należy zapewnić swobodę kształtowania strefy optymalizowanej przy jak najniższej ilości parametrów, ponieważ długość chromosomu ma wpływ na koszt obliczeń („przekleństwo wymiarowości”) i pośrednio szybkość zbieżności funkcji przystosowania.

3.2. Omówienie dostępnej literatury

Pierwsze prace w których pojawiły się wzmianki o numerycznym symulowaniu procesów ewolucyjnych pojawiły się w latach 50-tych XX wieku. Początkowo próbowano naśladować naturalne procesy ewolucyjne, a dopiero później uogólniono i zaadaptowano wykształcone w ten sposób metody do innych zagadnień. Pionierami tych prac byli m.in. G.E.P. Box, W. Bledsoe, A.S. Fraser i inni.

W ciągu następnych 25 lat wykształciły się główne kierunki rozwoju symulacji procesów ewolucyjnych. Ideę programowania ewolucyjnego opracowano w połowie lat 60-tych XX wieku w Stanach Zjednoczonych (L. Fogel). Teorię schematów dla algorytmów genetycznych opracował J.H. Holland (Stany Zjednoczone, połowa lat 70-tych XX wieku) i spopularyzował D. Goldberg. Strategie ewolucyjne zostały opracowane głównie w Niemczech przez I. Rechenberga i H.P. Schwefela (lata 1973-1981).

Już pierwsi badacze dostrzegali możliwości tkwiące w algorytmach ewolucyjnych i ich wszechstronność, prognozując dużą popularność tych metod (J. Holland). W trakcie ich rozwoju dochodziło do ścierania się różnych poglądów dotyczących doboru i znaczenia

operatorów genetycznych, ostatnie 20 lat to rozwój algorytmów ewolucyjnych w kierunku zwiększenia ich wydajności, stabilności, skrócenia czasu obliczeń, ułatwienia implementacji w różnych dziedzinach. Jest to faza charakterystyczna dla metody we wczesnym stadium zastosowań inżynierskich.

Można powiedzieć, że zagadnienia optymalizacji jednokryterialnej nie stanowią już większego problemu w zastosowaniach praktycznych. Szczególnie intensywny rozwój przeżywają obecnie metody pozwalające na optymalizację wielokryterialną. W ciągu ostatnich 10 lat ukazało się w periodykach naukowych około 500 publikacji na ten temat (za C. A. Cello¹).

W Polsce algorytmy ewolucyjne stają się również coraz popularniejsze. Podstawową książką omawiającą systematycznie tematykę algorytmów ewolucyjnych pozostaje podręcznik J. Arabasa [3]. Zastosowania praktyczne i bieżące kierunki ich rozwoju omawiają materiały z konferencji naukowych, także organizowanych w Polsce (*AI-METH*, *KAeIOG*). Ukazują się czasopisma poświęcone tym zagadnieniom („*Evolutionary Computation Journal*”, „*IEEE Transactions on Evolutionary Computation*”), również w Polsce. **LIT?**

3.3. Zastosowanie optymalizacji ewolucyjnej w mechanice konstrukcji

Można zaryzykować stwierdzenie, że metody optymalizacji w zagadnieniach inżynierskich mechaniki konstrukcji są obecnie na etapie porównywalnym z *MES* sprzed około 20 lat. Zagadnienie to jest dobrze opracowane teoretycznie, integracja z inżynierskimi narzędziami obliczeniowymi stanowi coraz mniejszy problem, pojawia się komercyjne oprogramowanie inżynierskie które ma umożliwić wzbogacenie tradycyjnego procesu projektowania z wykorzystaniem *MES* o fazę optymalizacji ewolucyjnej, także wielokryterialnej (programy *modeFRONTIER*, *TOSCA*). W zastosowaniach związanych z *MES* stosuje się optymalizację na poziomie siatki elementów skończonych, gdzie genotyp o liczności dziesiątek tysięcy stanowią współrzędne węzłów siatki modelu. Każdy liczący się na rynku inżynierski pakiet do obliczeń *MES* posiada moduł przeznaczony do optymalizacji metodami klasycznymi, a wprowadzenie metod ewolucyjnych jest tylko kwestią czasu.

Istnieją dwie zasadnicze przeszkody w upowszechnieniu optymalizacji (także ewolucyjnej) w zastosowaniach inżynierskich związanych z wykorzystaniem *MES*, są to:

¹ źródło <http://www.lania.mx/~ccoello/EMOO/EMOOjournals.html>

1. Koszt numeryczny optymalizacji, który ciągle uniemożliwia jej wykorzystanie w dużych inżynierskich zadaniach *MES*.
2. Brak uniwersalnej metody doboru parametrów metody, która uniemożliwia przeprowadzenie obliczeń w rozsądnym czasie bez specjalistycznej wiedzy i doświadczenia.

Trzecią, mniej oczywistą przyczyną jest stosunkowo słaba znajomość zagadnień optymalizacji przez inżynierów i niechęć do metod stosunkowo słabo sprawdzonych praktycznie. Brak także powszechnie dostępnych informacji o potencjalnych korzyściach płynących z zastosowania optymalizacji.

Podobne problemy stały w początkach lat 80-tych ubiegłego stulecia przez *MES*, ich przezwyciężenie trwało ok. 10 lat – można się spodziewać że za kilka lat nastąpi wymuszone przez rynkowe wymogi konkurencyjności upowszechnienie metod optymalizacji w zastosowaniach inżynierskich i sprowadzenie optymalizacji do roli kolejnego niezbędnego narzędzia w procesie projektowania.

W tej chwili publikacje na temat zastosowań przemysłowych optymalizacji ewolucyjnej stanowią znikomy ułamek wśród wszystkich publikacji dotyczących algorytmów ewolucyjnych, które mają ciągle charakter głównie naukowy. Główne kierunki widoczne w zastosowaniach związanych z mechaniką konstrukcji obejmują następujące dziedziny:

1. Optymalizacja wymiarów przekrojów poprzecznych konstrukcji (*size optimization*).
2. Optymalizacja własności materiałowych.
3. Optymalizacja kształtu.
4. Optymalizacja topologiczna.

Większość z nich ma charakter prac badawczo-rozwojowych, wiele wykorzystywanych jest jako przykłady mające upowszechnić metodę i narzędzia w zastosowaniach przemysłowych, są one rozpowszechniane przez producentów oprogramowania przeznaczonego do optymalizacji w oparciu o sprawdzone metody (*MES*). Przekrój tych zastosowań jest bardzo szeroki. Optymalizowane są konstrukcje budowlane (np. pod kątem wytrzymałości na obciążenia sejsmiczne), rdzenie reaktorów jądrowych [1], elementy instalacji przemysłowych – zarówno pod kątem wytrzymałości doraźnej jak i zmęczeniowej oraz charakterystyki dynamicznej [15], [35]. Typowe zastosowania w przemyśle motoryzacyjnym to elementy układów hamulcowych [25], [30], węzłów łożyskowych, elementów zawieszenia, tłoczonych elementów nadwozia, silników [1], itp.

Z zastosowań związanych z przemysłem lotniczym znane są prace dotyczące optymalizacji kształtu skrzydeł w różnym zakresie prędkości [40], [10], rozmieszczenia napędów elementów mechanizacji płata [38] w celu zapewnienia odpowiednich charakterystyk sterowania i inne.

Z zastosowań związanych z napędami lotniczymi najbardziej znane są przykłady optymalizacji kanałów przepływowych turbin i ołopatkowania rotorów oraz statorów [39] wykorzystujące algorytmy ewolucyjne jako jedną z metod poszukiwania optymalnego zestawu parametrów geometrycznych w celu otrzymania na przykład pożądanego rozkładu ciśnień na piórze łopatyki lub też szukania ekstremum funkcji celu (funkcji przystosowania) związanej ze sprawnością, stratami przepływowymi, mocą turbiny, itp. Znane są także zastosowania algorytmów genetycznych w automatyce sterowania silników turbowentylatorowych [13].

Najbliższe tematycznie prezentowanym w niniejszej rozprawie badaniom są prace dotyczące optymalizacji wielokryterialnej kształtu łopatek turbin (optymalizacja kształtu zamka jodełkowego łopatyki, minimalizacja masy rotora i ilości łopatek wieńca, maksymalizacja grubości profilu, maksymalizacja sprawności aerodynamicznej) prowadzone między innymi na zlecenie firmy *Avio Group* – producenta elementów turbin dla przemysłu lotniczego [14], [34] a także innych producentów w branży napędów lotniczych (np. *MTU Aero Engines* [20]).

4. ŚRODOWISKO I NARZĘDZIA OPTIMALIZACJI ŁOPATKI TURBINY PRZEMYSŁOWEJ

Optimalizacja kształtu łopatki przeprowadzana jest w całości w środowisku programu do modelowania i obliczeń metodą elementów skończonych *Ansys*. Środowisko to wybrane zostało z kilku względów, najważniejsze z nich to:

1. Powszechność wykorzystania w przemyśle napędów lotniczych, gdzie *Ansys* jest (obok *MSC.Nastran*) obowiązującym światowym standardem – wykorzystywany jest także w *Avio Polska Sp. z o.o.*, która dostarczyła modele geometryczne rotora turbiny i dane do analizy *MES*.
2. Zaawansowane możliwości języka programowania *APDL* (*Ansys Parameter Design Language* [2]) stanowiącego podstawę interfejsu użytkownika programu *Ansys*, które umożliwiły realizację algorytmu ewolucyjnego w postaci kodu *APDL* i jego pełną integrację z oprogramowaniem komercyjnym.
3. Szerokie możliwości oferowane przez moduł *pre-processor* (przygotowanie modelu obliczeniowego) programu *Ansys*, w tym możliwość eksportu i importu siatki *MES* oraz modelu geometrycznego w kilku formatach.

Algorytm ewolucyjny zrealizowany został jako program komputerowy w postaci kodu języka *APDL*. Można w nim wyróżnić trzy zasadnicze części:

1. *Generator Modeli Parametrycznych* – jest to procedura przeprowadzająca generację modelu geometrycznego optymalizowanej strefy łopatki, dane wejściowe stanowi chromosom (wektor parametrów geometrycznych) w postaci pliku tekstowego w formacie *ASCII* oraz wstępny model geometryczny półki dolnej łopatki w formacie *IGES*. Jeśli nie można zrealizować chromosomu jako modelu geometrycznego w jego oryginalnej formie, dodatkowo uruchamiany jest *Moduł Modyfikacji Chromosomu*, przeprowadza on modyfikację chromosomu do postaci wykonalnej poprzez iteracyjną zmianę wartości wybranych genów. Wynikiem działania tej części programu jest model geometryczny optymalizowanej części łopatki w postaci bryły gotowej do dyskretyzacji siatką *MES*. Wszystkie operacje przeprowadzane są w środowisku *pre-processor* programu *Ansys* za pomocą kodu *APDL*. Szczegółowy opis działania tej części kodu przedstawia rozdział 5.3.

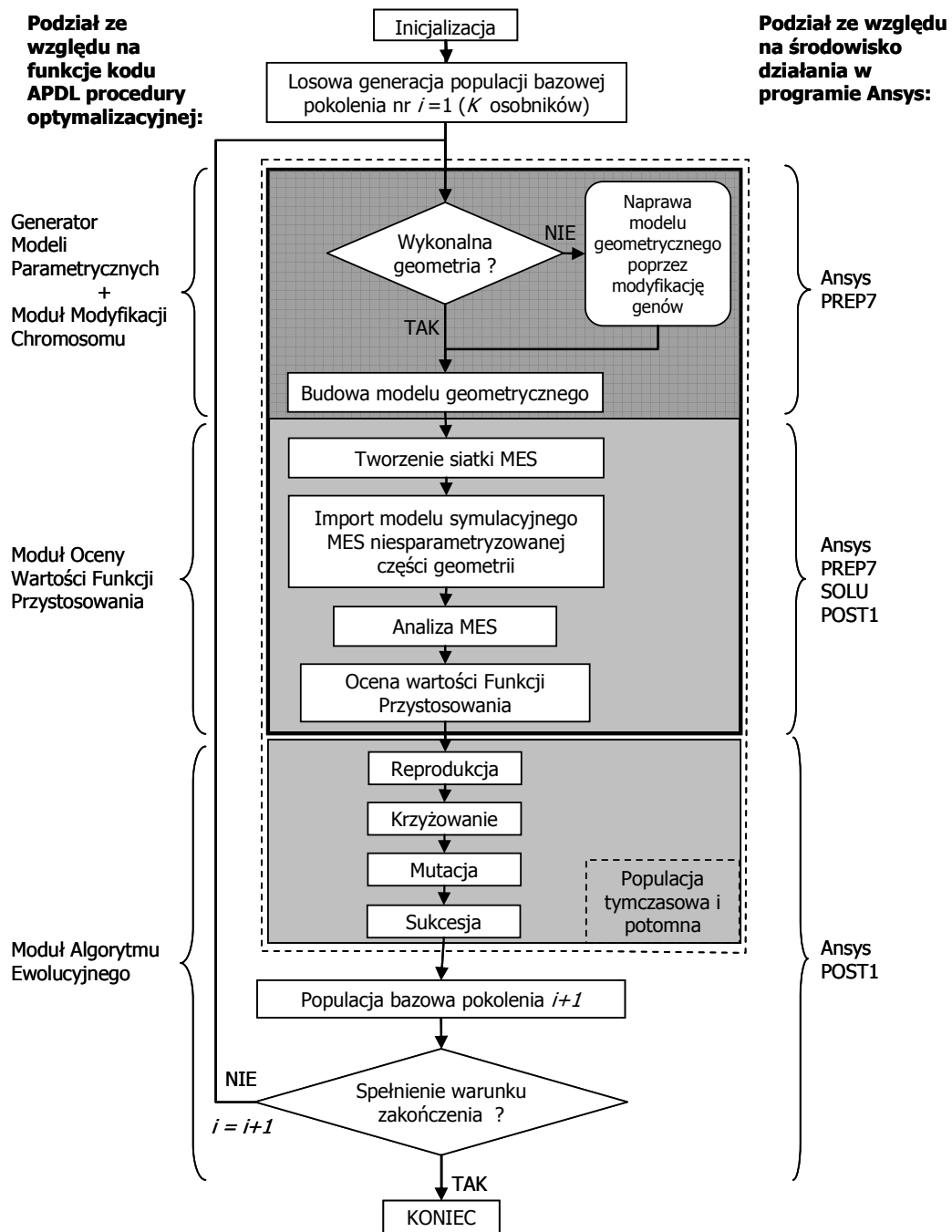
2. *Moduł Oceny Wartości Funkcji Przystosowania* – jest to część programu przeprowadzająca budowę i analizę modelu dyskretnego *MES* segmentu rotora turbiny. Wszystkie wymagane operacje, w postaci utworzenia siatki elementów skończonych w oparciu o model geometryczny, przyłożenia warunków brzegowych i zdefiniowania własności materiałowych przeprowadzane są tylko na optymalizowanej (sparametryzowanej) części modelu, część niesparametryzowana importowana jest jako gotowy do analizy model *MES* z odpowiednio przygotowanego pliku tekstowego *ASCII* programu *Ansys*. Dane wejściowe to model geometryczny będący wynikiem działania modułu nr 1 oraz plik z opisem modelu w standardzie *Ansys* zawierający niesparametryzowaną część modelu *MES* rotora turbiny, który zawiera także informacje konieczne do przygotowania modelu części sparametryzowanej (dane materiałowe, warunki brzegowe). Efektem działania tej części programu jest rozwiązanie zadania *MES* w postaci masy i pola naprężeń na optymalizowanej części łopatki. Wszystkie operacje przeprowadzane są w środowisku *pre-processor*, *solver* i *post-processor* programu *Ansys* za pomocą kodu *APDL*.

3. *Moduł Algorytmu Ewolucyjnego* – jest to część programu przeprowadzająca proces generacji kolejnego pokolenia na podstawie wartości funkcji przystosowania i chromosomu pokolenia bieżącego. Dane wejściowe to wartości maksymalnych naprężeń głównych na optymalizowanej części łopatki oraz masy dla wszystkich osobników pokolenia bieżącego, uzyskane jako wynik działania modułu nr 2 oraz chromosomy pokolenia bieżącego zapisane jako plik tekstowy w formacie *ASCII*. Na podstawie wartości funkcji przystosowania następuje reprodukcja osobników z bieżącego pokolenia, ich krzyżowanie, mutacja i selekcja prowadząca do utworzenia kolejnego pokolenia. Wszystkie operacje przeprowadzane są w środowisku *post-processor* programu *Ansys* za pomocą kodu *APDL*. Algorytm ewolucyjny zmienia się zależnie od celu optymalizacji.

Rys. 4.2 przedstawia schemat działania procedury optymalizacyjnej z uwzględnieniem podziału ze względu na środowisko *Ansys* oraz moduły programu optymalizacyjnego.

Budowa typowej łopatki turbiny oraz sposób jej zamontowania na dysku turbiny pokazuje rys.4.3. Funkcjonalnie, profil łopatki podzielony jest na cztery części, tj. stronę natarcia (ang. *leading edge*), stronę ciśnącą (ang. *pressure side*), stronę spływu (ang. *trailing edge*), stronę ssącą (ang. *suction side*). W kierunku promieniowym można podzielić łopatkę na następujące strefy: zamek jodełkowy osadzony w gnieździe dysku turbiny (ang. *firtree*), stopę (ang. *shank*), półkę dolną (ang. *bottom platform*), dolną strefę zmiany przekroju półka-

pióro (ang. *hub fillet*), pióro (ang. *airfoil*), górną strefę zmiany przekroju pióro-półka (ang. *tip fillet*) oraz półkę górną (ang. *shroud*).



Rys. 4.2 Schemat blokowy procesu optymalizacji ewolucyjnej kształtu łopatki turbiny przemysłowej.

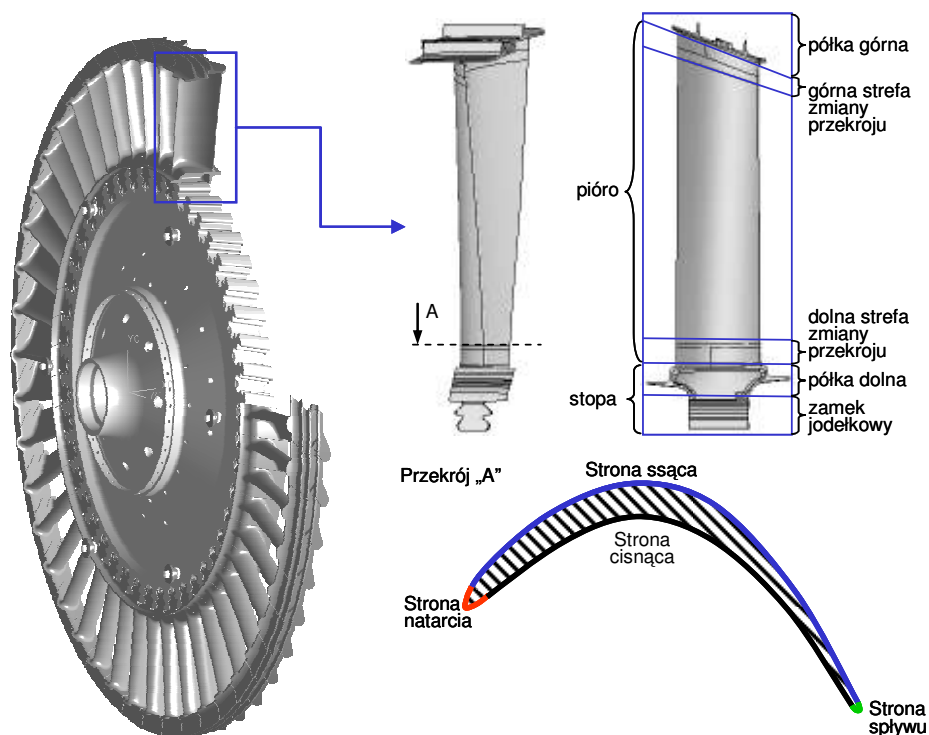
W warunkach roboczych łopatka znajduje się w złożonym termo-mechanicznym stanie obciążenia. Obciążenia mechaniczne powodowane są oddziaływaniem sił gazu oraz

siłami bezwładności pochodzącymi od prędkości kątowej rotora turbiny. Napływający na łopatkę od strony sprężarki strumień gazu zostaje rozdzielony na powierzchni natarcia, a kształt profilu powoduje powstanie siły nośnej na łopatkę wywołanej różnicą ciśnień pomiędzy stroną cisnącą i stroną ssącą. Wypadkowe siły pól ciśnień na piórze powodują powstanie momentu skręcającego łopatkę w płaszczyźnie osiowo-stycznej oraz momentu gnącego w płaszczyźnie promieniowo-osiowej. Jednak główną przyczyną obciążeń mechanicznych łopatki jest siła odśrodkowa wywołana obrotami rotora z prędkością kilku tysięcy obrotów na minutę, powodująca powstanie głównie momentu gnącego w płaszczyźnie promieniowo-stycznej, o znaku przeciwnym do momentu wynikającego z obciążeń strumieniem gazu [11], [41], [26]. Te dwa główne momenty gnące są kompensowane dla warunków roboczych turbiny w procesie projektowania łopatki podczas wyznaczania optymalnego kąta pochylenia osiowego (ang. *tilt*) i obwodowego (ang. *leaning*), proces ten jest przeprowadzany na podstawie analizy statystycznej wyników dla kilku konfiguracji geometrycznych po wstępnym wyznaczeniu geometrii optymalizowanej w niniejszej pracy strefy przejścia półka dolna – pióro łopatki. W jego trakcie nie dopuszcza się jednak do zmiany kształtu przekroju łopatki w płaszczyźnie normalnej do promienia, operując na kilku – kilkunastu przekrojach przesuwanych jako niezmiennie w płaszczyznach o stałych promieniach.

Obciążenia termiczne wywołane są kontaktem łopatki ze strumieniem gorącego gazu oraz wymianą ciepła z innymi elementami rotora (dysk). Przestrzenne pole temperatury można przybliżyć w celu uproszczenia zagadnienia jednowymiarowym pola zmiennym tylko w funkcji promienia, z wystarczającą dla obliczeń wytrzymałościowych dokładnością. W trakcie analizy naprężeń na samej łopatkę, pomija się rozszerzalność cieplną i analizuje się model od razu w konfiguracji „cieplej”, tj. przeskalowanej zgodnie z uproszczonym rozkładem temperatur i średnimi wartościami współczynników ekspansji termicznej, pozwala to na uwzględnienie zmiany średnicy rotora prowadzącego do przyrostu siły odśrodkowej na łopatkę. W przypadku analizy innych elementów rotora (dyski, uszczelnienia) uwzględnia się rozszerzalność cieplną, która w przypadku połączeń śrubowych i wcisków montażowych (dysk/uszczelnienia międzystopniowe) ma zasadnicze znaczenie dla rozkładu naprężeń. Upraszczając, można powiedzieć że zasadniczym problemem w przypadku projektowania dysków i uszczelnień międzystopniowych jest wytrzymałość na zmęczenie niskocyklowe związane z rozszerzalnością cieplną i siłami odśrodkowymi, natomiast w przypadku łopatek jest to wytrzymałość na zmęczenie wysokocyklowe związana z drganiami własnymi łopatki w strumieniu gazu. W rzeczywistości naprężenia wywoływane ekspansją termiczną łopatki kompensowane są poprzez jej przemieszczenia w strefie zamka. W zasadzie jedynym znaczącym z wytrzymałościowego punktu widzenia efektem zmiany temperatury łopatki jest zmiana jej własności materiałowych, uwzględniona w obliczeniach *MES*.

Wyniki analizy wytrzymałościowej *MES* rotora turbiny w warunkach roboczych, wraz z omówieniem stanu przemieszczeń i naprężeń przedstawia rozdział 6.7. Szczegółowy opis warunków roboczych turbiny przedstawia rozdział 6.3.

Opisane zjawiska powodują, że jest strefa przejścia półka dolna – pióro łopatki jest newralgiczna zarówno ze względu na wytrzymałość doraźną jak i zmęczeniową, z uwagi na złożony stan obciążeń mechaniczno-termicznych oraz gwałtowną zmianę przekroju. Z tego względu została ona wybrana do optymalizacji kształtu. Niewielkie zmiany kształtu w tym rejonie powodują bardzo duże zmiany w rozkładzie naprężeń i przemieszczeń, znacznie wpływając również na odkształcenia całej łopatki.

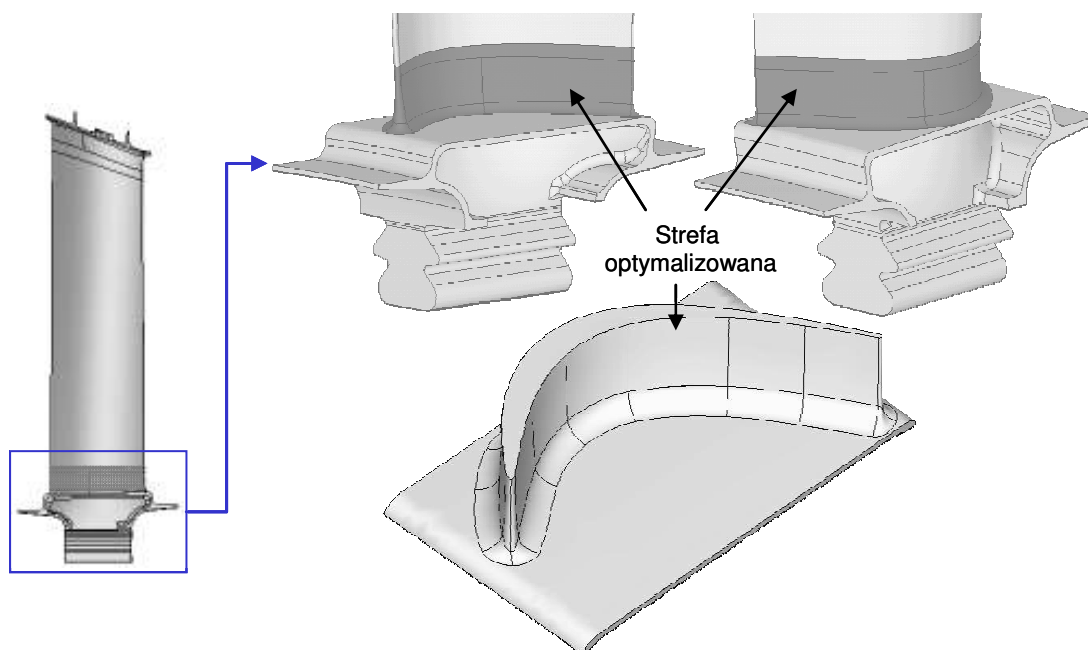


Rys.4.3 Budowa łopatki turbiny.

Strefa optymalizowana łopatki rozciąga się od półki dolnej do 10% wysokości pióra (rys. 4.4). W tej części kształt pióra może być zmieniany w bardzo szerokim zakresie bez znaczącego wpływu na sprawność aerodynamiczną łopatki i parametry kanału przepływowego (rozkład ciśnień na piórze łopatki – rozdział 6.3). Zachowane muszą być jednak ograniczenia wynikające z funkcji aerodynamicznych łopatki, czyli:

1. Styczność powierzchni strefy zmiany przekroju do powierzchni pióra łopatki na wysokości 10% pióra.

2. Styczność powierzchni strefy przejścia do powierzchni półki dolnej osiągana poprzez odpowiednie ukształtowanie dolnej części pióra.
3. Styczność powierzchni wokół obwodu pióra łopatk.



Rys. 4.4 Łopaska turbiny - strefa optymalizowana.

Strefa optymalizowana jest szczególnie narażona na koncentracje naprężeń prowadzące do zniszczenia. Wynika to z gwałtownej zmiany jej przekroju i związanej z nią zmiany sztywności. Momenty zginający i skręcający wynikające z działania sił bezwładności oraz pól ciśnień powodują powstanie koncentracji naprężeń w okolicy krawędzi spływu po stronie cisałej i krawędzi natarcia po stronie ssącej łopaski. Również nieuwzględnione w tej analizie naprężenia dynamiczne wywołane drganiem własnym łopaski dla niższych częstotliwości (postaci własne giętne i skrętne) koncentrują się w tym rejonie [26]. Dodatkowo występuje tam wysoki gradient temperatury spowodowany tym, że półka dolna opływana jest od strony pióra głównym gorącym strumieniem gazu z kanału przepływowego turbiny, natomiast od dołu chłodzona jest strumieniem wewnętrznym, co powoduje gwałtowną zmianę własności materiałowych.

5. PARAMETRYZACJA MODELU GEOMETRYCZNEGO ŁOPATKI

5.1. Model geometryczny łopatki turbiny przemysłowej

Model geometryczny rotora turbiny przemysłowej reprezentuje jeden segment kątowy dysku o rozwartości $360^\circ/N$ gdzie N odpowiada ilości łopatek w wieńcu rotora. Na powierzchniach lateralnych modelu narzucone zostały warunki symetrii przemieszczeń. Model składa się z następujących elementów:

1. Łopaska.
2. Dysk.
3. Uszczelnienie międzystopniowe przednie (fragment).
4. Uszczelnienie międzystopniowe tylne (fragment).

Orientacyjną pozycję modelu na schemacie całej turbiny przedstawia rys. 5.1, poszczególne elementy modelu przedstawia rys. 5.2. Do opisu modelu wykorzystane zostały układy współrzędnych pokazane na rysunku, kierunki obrotu rotora i napływu gazu został zaznaczony strzałkami. Model geometryczny przygotowany został z wykorzystaniem programu *Unigraphics* przez *Avio Polska Sp. z o.o.* i udostępniony do wykorzystania w tej pracy doktorskiej.

Kształt strefy optymalizowanej opisany jest za pomocą powierzchni typu *B-surface*. Ten typ opisu jest matematycznie najbardziej elastyczną metodą przedstawienia powierzchni w modelu geometrycznym, ich kształt tworzony jest przy pomocy punktów kontrolnych [33]. Powierzchnie te zbudowane są w oparciu o wybrane przekroje na łączeniach wydzielonych części obwodu pióra łopatki. W konstrukcji łopatek turbin niskiego ciśnienia używa się do ich opisu zwykle par łuków o zadanych promieniach (ang. *compound fillet*) lub krzywej elipsoidalnej.

W oparciu o założenia sformułowane w rozdziale 4, model geometryczny strefy optymalizowanej został sparametryzowany po wybraniu przekrojów umożliwiających jej jednoznaczny i dokładny opis. Cały proces parametryzacji przeprowadzony został w oparciu o niesparametryzowany model wyjściowy stanowiący oryginalną konfigurację geometryczną



modelu łopatki będącą jednym z etapów przejściowych projektu inżynierskiego. Model ten oznaczony zostanie jako konfiguracja „0”. Zostało sprawdzone, że sparametryzowany model geometryczny zbudowany na podstawie wektora zmiennych projektowych jest w granicach tolerancji numerycznych identyczny z wyjściowym modelem niesparametryzowanym w konfiguracji „0” jeśli chodzi o kształt opisujących go powierzchni.

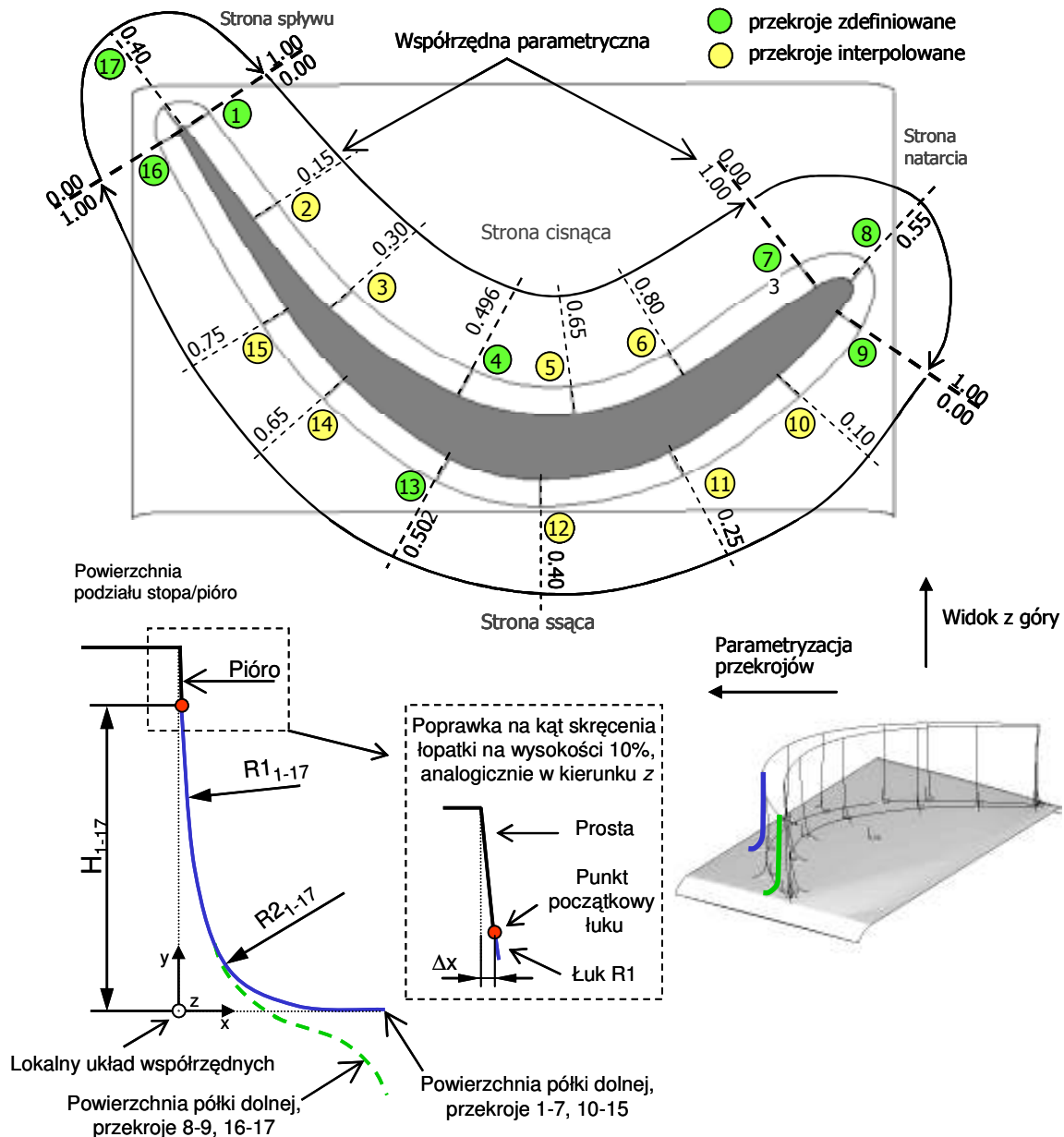
Każda z czterech części obwodu pióra podzielona została w wybranych miejscach, oznaczonych z wykorzystaniem współrzędnej parametrycznej zdefiniowanej wzdłuż krawędzi pióra. Każdy z przekrojów zdefiniowany został w lokalnym kartezyjskim układzie współrzędnych za pomocą trzech parametrów.

- R1* – większy (górny) z promieni zaokrąglenia krzywej przekroju,
- R2* – mniejszy (dolny) z promieni zaokrąglenia krzywej przekroju,
- H* – wysokość strefy optymalizowanej w danym przekroju.

Lokalne układy współrzędnych w których opisane są przekroje tworzone są na powierzchni półki dolnej łopatki, wzdłuż krzywej obrysu profilu powstałej poprzez projekcję przekroju łopatki z wysokości 10% na półkę dolną. Projekcja została przeprowadzona w sposób uwzględniający zmienny z wysokością łopatki kąt natarcia profilu (skręcenie łopatki), a do pozycji punktów początkowych krzywych definiujących przekroje parametryczne są doliczane współczynniki (Δx , Δz) uwzględniające różnicę położenia odpowiadających sobie punktów profilu na półce dolnej i na wysokości 10% pióra. Szczegóły parametryzacji profilu pokazuje rys. 2.1.

W trakcie przygotowania modelu okazało się że opis w 17 przekrojach wystarcza na dokładne odwzorowanie kształtu modelu w konfiguracji „0”. Taki opis wprowadza $17 \times 3 = 51$ parametrów definiujących strefę optymalizowaną. Jednocześnie można zauważyć, że wiele sąsiadujących przekrojów można wyznaczyć poprzez interpolację, definiując tylko niektóre, decydujące o obrysie profilu. Pozwala to na uproszczenie opisu do 8 przekrojów, pomiędzy którymi wartości promieni i wysokości wyznaczone są poprzez liniową interpolację pomiędzy najbliższymi zdefiniowanymi przekrojami. Jeśli dodatkowo przyjmiemy, że wysokość strefy optymalizowanej jest stała wokół obwodu pióra łopatki, to otrzymamy zestaw $8 \times 2 + 1 = 17$ parametrów opisujących model. Ten zestaw jest wystarczający aby na podstawie opisu parametrycznego zbudować model identyczny geometrycznie z konfiguracją „0”.

Rozkład definiowanych przekrojów jak również ich ilość może zostać dowolnie zmieniona poprzez wprowadzenie niewielkich zmian w kodzie *APDL* programu optymalizacyjnego, również wysokość strefy optymalizowanej może być definiowana indywidualnie w każdym przekroju, założenia dotyczące budowy wektora parametrów opisujących model są parametrami metody.



Rys. 5.3 Szczegóły parametryzacji strefy optymalizowanej.

Ten sposób parametryzacji umożliwia wygenerowanie modelu o wymaganej konfiguracji na podstawie istniejącego wcześniej modelu bazowego, opisującego niesparametryzowaną część łopatki w postaci pióra od wysokości 10% oraz półki dolnej z obrysem profilu, stopą i zamkiem a także zestawu 17 parametrów przekazywanych do programu w formie pliku tekstowego. Cała generacja modelu odbywa się za pomocą kodu APDL w środowisku *pre-processor* programu Ansys.

5.2. Modyfikacja chromosomu do postaci wykonalnej

Ponieważ w trakcie optymalizacji konieczne będzie wygenerowanie w sposób całkowicie automatyczny kilku tysięcy modeli, niezbędne jest zapewnienie stabilności procesu. Proces uznajemy za stabilny jeśli za pomocą algorytmu modyfikacji parametrów wejściowych (zrealizowanego jako *Moduł Modyfikacji Chromosomu*), każdy zestaw parametrów może zostać sprowadzony do postaci wykonalnej, tj. zapewniającej utworzenie na jego podstawie zestawu zamkniętych, wzajemnie stycznych powierzchni które jednoznacznie definiują bryłę będącą modelem geometrycznym optymalizowanej części łopaty. W początkowym okresie generacji modeli bardzo rzadko zdarza się że wygenerowany zestaw parametrów nadaje się od razu do budowy modelu, w miarę postępowania procesu optymalizacji zakresy zmienności parametrów ulegają znacznemu zawężeniu i proces modyfikacji danych wejściowych wykorzystywany jest coraz rzadziej.

Sprowadzenie zestawu parametrów do postaci wykonalnej odbywa się w dwóch etapach:

1. Modyfikacja parametrów do postaci wykonalnej przed rozpoczęciem generacji modelu.

Dopuszczalne wartości parametrów otrzymywane w drodze losowania (dla pierwszego pokolenia) oraz przetwarzania parametrów wejściowych (dla kolejnych pokoleń) muszą leżeć w obszarze zapewniającym wykonalność modelu ze względu na poziom tolerancji modelowania oraz założenia o kształcie definiowanego przekroju. Wartości dopuszczalne zdefiniowane zostały poprzez normalizację w stosunku do wartości parametrów modelu w konfiguracji „0” i przyjęcie odpowiednich mnożników, tabela 5.1 przedstawia ich zakres.

Założenie o kształcie przekroju narzuca wymagania $R2 \leq R1$ i $R2 \leq H$. Dla $R2=H$ przekrój sprowadza się do opisanego jednym promieniem i wysokością, ponieważ łuk o promieniu $R1$ jest realizowany na kącie o rozwartości 0° (patrz rys. 5.3). Podobnie ze względu na maksymalną wartość promienia $R1$ modelu w konfiguracji „0”, nie ma sensu dopuszczanie wartości $R1$ większych niż $6.75R10$ (gdzie $R10$ to średnia wartość promienia $R1$ modelu w konfiguracji „0”), ponieważ dalsze powiększanie $R1$ nie powoduje już żadnej zmiany rozwiązania a łuk można przybliżyć z wystarczającą dokładnością odcinkiem prostej. Podobnie wartości promienia $R2 < 0.05R20$ (gdzie $R20$ to średnia wartość promienia $R2$ modelu w konfiguracji „0”) są na poniżej tolerancji wykonawczych łopaty, również więc nie ma sensu ich dopuszczanie.

Tabela 5.1. Wartości dopuszczalne parametrów modelu optymalizowanego (znormalizowane w stosunku do konfiguracji „0”).

Parametr	Minimum, -	Maksimum, -
$R1$	0.02	6.75
$R2$	0.05	H dla danego osobnika*
H	0.4	1.20

* w przypadku $R2 = H$ przekrój redukowany jest do kształtu z opisanego jednym promieniem

Oznaczenia:

$R1$ – promień znormalizowany względem średniej wartości promienia odpowiadającego $R1$ modelu w konfiguracji „0”,

$R2$ – promień znormalizowany względem średniej wartości promienia odpowiadającego $R2$ modelu w konfiguracji „0”,

H – wysokość znormalizowana względem wysokości H modelu w konfiguracji „0”.

2. Modyfikacja parametrów do postaci wykonalnej w trakcie generacji modelu.

Ograniczenia narzucone na wartości skrajne parametrów nie zapewniają wykonalności modelu, ponieważ każdy z przekrojów jest funkcją trzech definiowanych niezależnie parametrów, a dodatkowo generowane powierzchnie muszą zachować styczność do walcowej powierzchni półki dolnej. Przed rozpoczęciem generacji modelu nie da się określić czy dany zestaw parametrów będzie możliwy do zrealizowania, więc konieczne jest modyfikowanie wartości parametrów w trakcie budowy modelu, w sposób iteracyjny. Odbywa się to w trzech etapach, na podstawie sprawdzenia trzech warunków geometrycznych:

2.1. Sprawdzenie wartości promienia $R1$.

Pierwszym warunkiem jaki jest sprawdzany po zapewnieniu, że wartości parametrów leżą w zakresach dopuszczalnych, jest warunek istnienia jednego punktu przebicia powierzchni półki dolnej krzywą o promieniu $R1$ i kącie α zakreślaną z wysokości H każdego przekroju. Schemat wystąpienia błędu wywołanego niespełnieniem tego warunku pokazany został na rys. 5.4. Kąt α obliczany jest na podstawie wzoru:

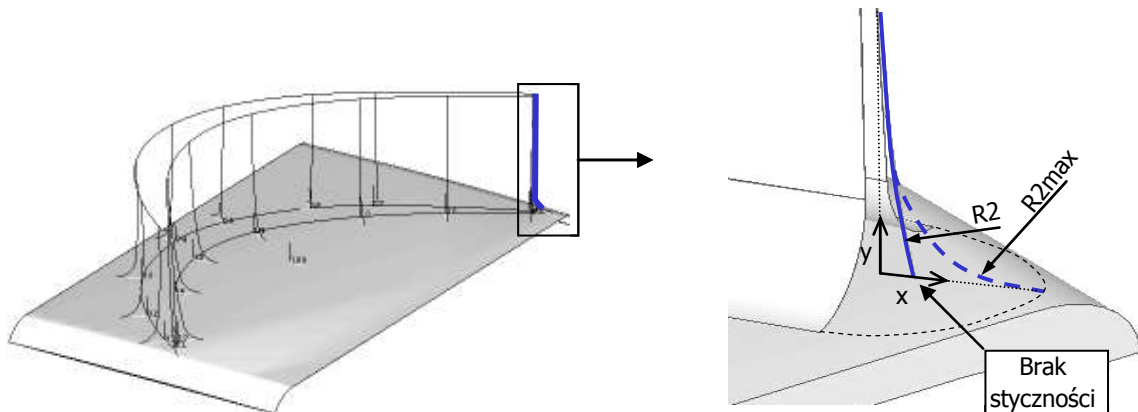
$$\alpha = 3 \arctan\left(\frac{H}{R1}\right) \quad (5.1)$$

gdzie:

α – kąt rozwarcia łuku o promieniu $R1$,

H – wysokość przekroju strefy optymalizowanej,

l_{min} – długość krótszej z dwóch krzywych dla których tworzony jest łuk $R1$,
 β – kąt pomiędzy dwoma krzywymi łączonymi łukiem $R1$.



Rys. 5.5 Schemat wystąpienia błędu geometrycznego nr 2.

2.3. Sprawdzenie warunku rozdzielności dolnego obrys profilu z krawędzią półki dolnej.

Ostatnim ze sprawdzanych warunków jest warunek rozdzielności dolnego obrysu profilu z krawędzią półki dolnej. Dla kilku fragmentów obrysu profilu może dojść do przecięcia obrysu profilu z krawędzią półki dolnej, co powoduje utworzenie niedomkniętego zestawu powierzchni. Schemat wystąpienia błędu spowodowanego niespełnieniem tego warunku pokazuje rys. 5.6. W przypadku wystąpienia tego błędu, w przekroju najbliższym punktowi przecięcia obrysu profilu z krawędzią półki dolnej wartości obu promieni zaokrągłeń zmieniają się iteracyjnie aż do spełnienia warunku 2.3 według równania (5.4):

$$\begin{aligned} R1_M &= 1.05R1 \\ R2_M &= 0.95R2 \end{aligned} \quad (5.4)$$

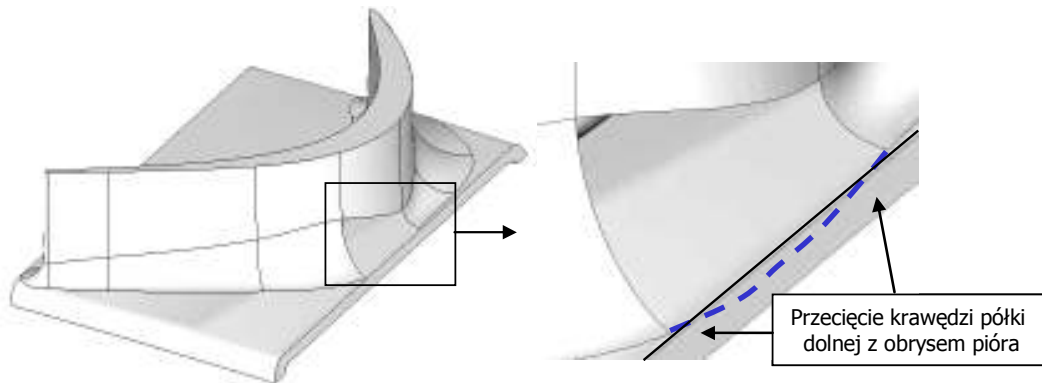
gdzie:

$R2_M$ – wartość promienia $R2$ po modyfikacji,

$R1_M$ – wartość promienia $R1$ po modyfikacji.

Po spełnieniu warunków 2.1 – 2.3, tworzona jest siatka krzywych definiujących przekroje strefy optymalizowanej. Dolny łańcuch krzywych łączących punkty na wysokości 0% pióra łopatki jest następnie projektowany w kierunku normalnym na powierzchnię półki dolnej w celu zapewnienia przylegania powierzchni strefy parametryzowanej do powierzchni półki. Proces ten jest przeprowadzany w dwóch etapach. Najpierw odbywa się projekcja

poprzez wyciągnięcie powierzchni z każdej projektowanej krzywej i znalezienie nowej krzywej powstałej poprzez przecięcie powierzchni półki dolnej oraz powierzchni utworzonej przez wyciągnięcie krzywej. Ze względu na niedoskonałości opisu powierzchni w programie *Ansys*, zdarza się że krzywa na przecięciu dwóch powierzchni nie zostaje utworzona. W takim wypadku uruchamiany jest alternatywny proces tworzenia krzywej polegający na projektowaniu na powierzchni półki dolnej zestawu punktów, które służą następnie do rekonstrukcji oryginalnej krzywej.



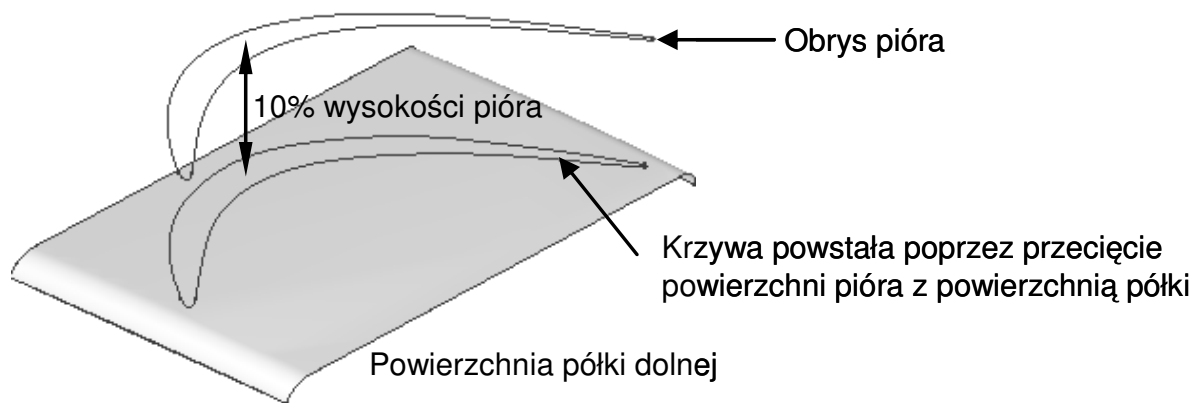
Rys. 5.6 Schemat wystąpienia błędu geometrycznego nr 3.

5.3. Opis generacji modelu strefy optymalizowanej

Model *MES* optymalizowanej strefy łopatki budowany jest na podstawie utworzonego wcześniej modelu geometrycznego. Proces ten przeprowadzany jest w środowisku *pre-processor* programu *Ansys* za pomocą procedury optymalizacyjnej w języku *APDL*. Można go przedstawić w następujących etapach:

1. Import modelu bazowego strefy optymalizowanej w formacie *IGES*.

Model początkowy składa się z modelu powierzchni półki dolnej łopatki oraz dwóch zestawów krzywych: obrysu pióra łopatki na wysokości 10% oraz obrysu pióra łopatki na wysokości 0% (leżącego na powierzchni półki dolnej). Krzywe utworzone są zgodnie z opisanym w rozdziale 5.1 podziałem obwodu pióra łopatki, na część reprezentującą powierzchnię cisnącą, ssącą, natarcia i spływu. Rys. 5.7 przedstawia model początkowy strefy optymalizowanej.



Rys. 5.7 Model początkowy strefy optymalizowanej łopatki.

2. Budowa siatki krzywych opisujących powierzchnię modelu geometrycznego.

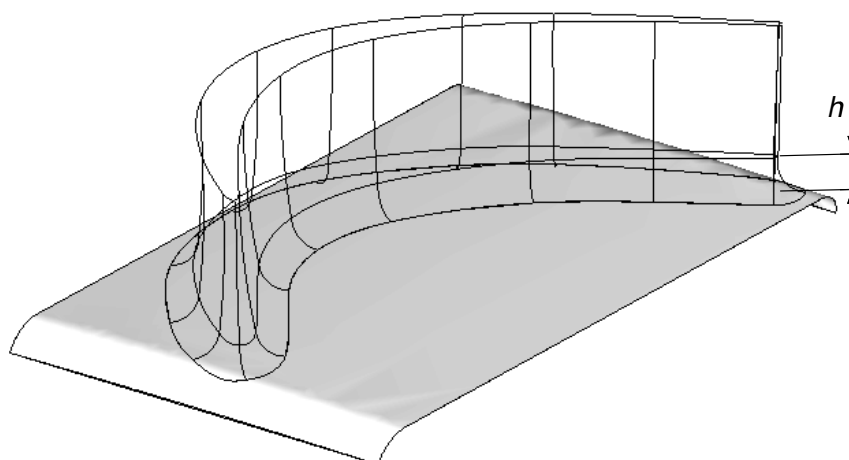
Po zaimportowaniu modelu początkowego rozpoczyna się budowa siatki krzywych opisujących kształt strefy optymalizowanej zgodnie z opisem w rozdziale 5.2. Geny realizowane są jako parametry geometryczne modelu, chromosom zostaje zaimportowany z pliku tekstowego. Powstaje 17 przekrojów, z których każdy złożony jest z trzech krzywych reprezentujących łuki o promieniach $R1$ i $R2$ oraz fragment pióra znajdujący się powyżej strefy przejścia pióro-półka i opisany w lokalnym kartezjańskim układzie współrzędnych. W trakcie tworzenia krzywych przeprowadzane jest sprawdzanie błędów geometrycznych opisane w rozdziale 5.2. Następnie krzywe te są w każdym przekroju łączone i dzielone powtórnie na dwie części, w celu ułatwienia opisu powierzchni. Wysokość podziału h jest stała dla całego obwodu i odpowiada podziałowi na modelu w konfiguracji „0”. Kolejnym krokiem jest utworzenie dwóch łańcuchów krzywych typu *spline* wokół obrysu profilu na poziomie półki dolnej oraz na wysokości h . Następnie krzywe z poziomu półki dolnej są projektowane w kierunku normalnym na powierzchnię półki.

Rys. 5.8 przedstawia siatkę krzywych opisującą powierzchnię strefy optymalizowanej.

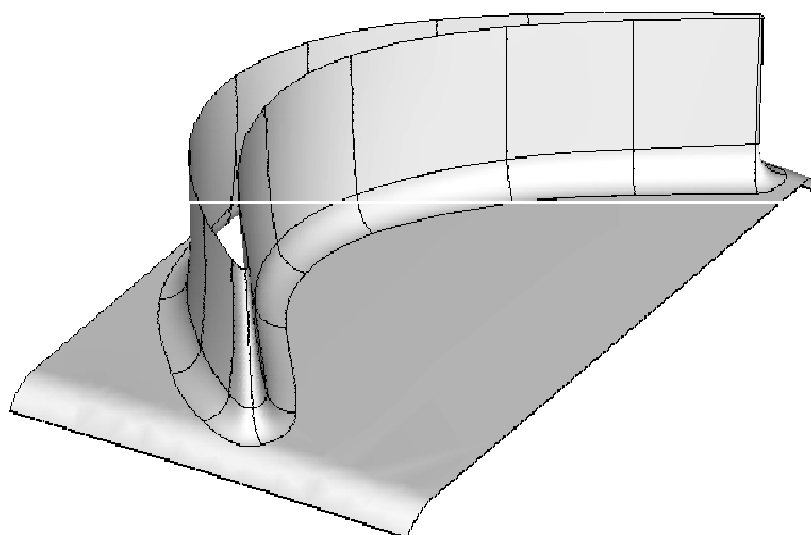
3. Budowa powierzchni opisujących bryłę modelu geometrycznego.

Po utworzeniu krzywych budowane są powierzchnie opisujące bryłę modelu geometrycznego strefy optymalizowanej. Są to powierzchnie typu *B-spline* opisane za pomocą z 4 krzywych stanowiących krawędzie. Dodatkowo następuje wycięcie

obrysu pióra z powierzchni łopatki . Kompletny zestaw sparametryzowanych powierzchni pokazuje rys. 5.9.



Rys. 5.8 Siatka krzywych opisująca powierzchnię strefy optymalizowanej.

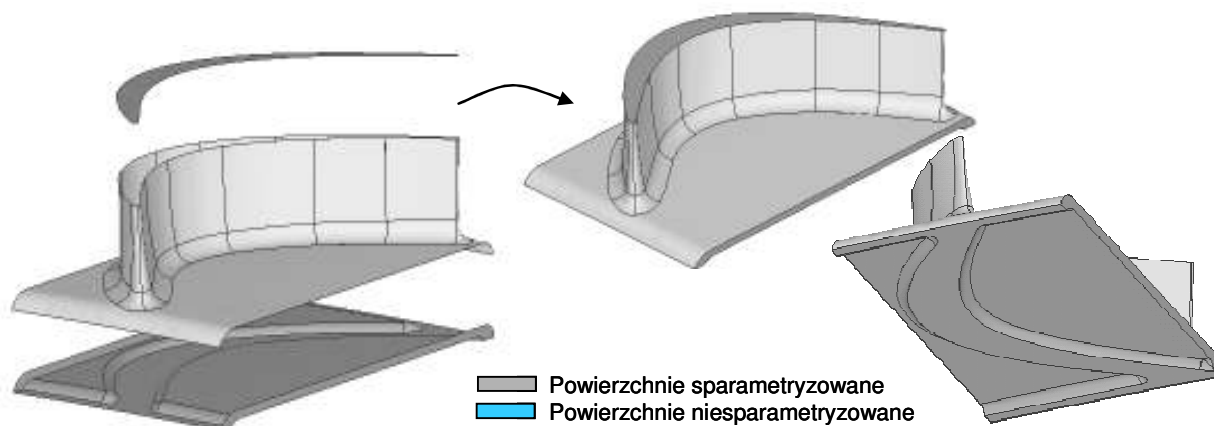


Rys. 5.9 Powierzchnie opisujące kształt strefy optymalizowanej.

4. Utworzenie bryły.

Po utworzeniu sparametryzowanych powierzchni następuje import powierzchni koniecznych do zamknięcia bryły stanowiącej model geometryczny i jej utworzenie. Operację tę pokazuje rys. 5.10.

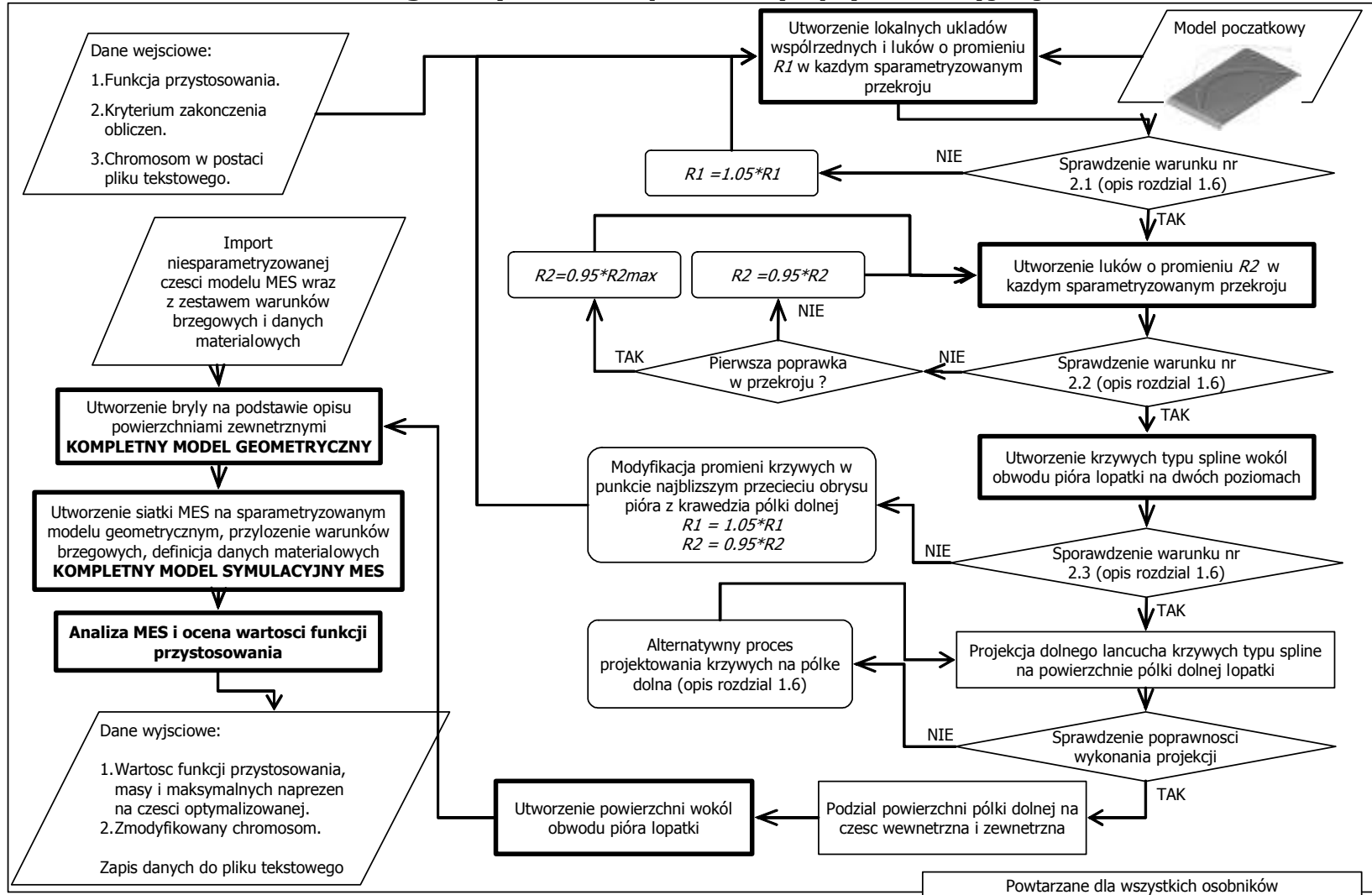
Kolejnym krokiem jest budowa siatki *MES* na modelu geometrycznym części optymalizowanej. Gęstości siatek powierzchniowych są zróżnicowane w zależności wymaganej dokładności odwzorowania danej strefy. Wykorzystywany jest automatyczny generator siatek. Po utworzeniu siatki następuje przypisanie własności materiałowych i warunków brzegowych na nowo utworzonej siatce.



Rys. 5.10 Utworzenie modelu geometrycznego strefy optymalizowanej.

Rys. 5.11 pokazuje szczegółowo proces generacji modelu obliczeniowego rotora turbiny przemysłowej za pomocą schematu blokowego, z uwzględnieniem wszystkich operacji prowadzących do utworzenia modelu geometrycznego, modelu *MES* i sprawdzenia wartości funkcji przystosowania.

Szczegółowy schemat procedury optymalizacyjnej



Rys. 5.11 Schemat procedury optymalizacyjnej kształtu łopatki turbiny przemysłowej

6. MODEL TERMOMECHANICZNY ŁOPATKI TURBINY PRZEMYSŁOWEJ

6.1. Założenia upraszczające

W celu zbudowania modelu obliczeniowego rotora turbiny przemysłowej zawierającego optymalizowaną łopatkę przyjęto szereg założeń upraszczających. Ich uzasadnienie opiera się na porównaniach wyników obliczeń z wynikami prób stanowiskowych silników, gdzie monitoruje się tensometrycznie stan naprężeń na łopatkach i innych elementach turbin (które najczęściej objęte są tajemnicą przemysłową i w zasadzie się ich nie publikuje). Potwierdzeniem słuszności przyjętych uproszczeń są też wyniki analiz nieliniowych *MES*, uwzględniające cały szereg zjawisk pominiętych w analizie liniowej (kontakt z tarciami, nieliniowość geometryczna, uplastycznienie, itp.). Publikacje naukowe najczęściej poruszają te problemy w sposób bardzo ogólny, lub nie poświęcają zbyt wiele miejsca porównaniu wyników obliczeń z badaniami doświadczalnymi. Powszechnie przyjmuje się współczynniki bezpieczeństwa na poziomie zapewniającym uwzględnienie błędu w wykorzystywanym modelu fizycznym. Autorowi znana jest praca omawiająca wpływ linearyzacji modelu *MES* na wyniki obliczeń, która w pełni potwierdza słuszność wymienionych poniżej założeń [26], inne dostępne źródła omawiają te problemy dość ogólnie ale również potwierdzają te założenia [11], [14], [21], [41].

Podsumowując, należy podkreślić że większość przyjętych założeń upraszczających jest dopuszczalna tylko pod warunkiem, że interesują nas wyniki analizy wytrzymałościowej wyłącznie dla łopatki – wyniki obliczeń dla dysku i uszczelnień międzystopniowych mogą być traktowane tylko jako szacunkowe, dla potrzeb inżynierskich uzyskuje się je w analizach z wykorzystaniem bardziej szczegółowych modeli, w których z kolei upraszcza się samą łopatkę (szczegółowe zamodelowanie połączeń śrubowych na dysku, uwzględnienie kontaktu z tarciami na wszystkich powierzchniach kontaktu, przy jednoczesnym zastąpieniu większej części łopatki pojedynczym elementem masowym z zestawem sił skupionych odpowiadających obciążeniom aerodynamicznym).

Założenia upraszczające można podzielić na:

1. Założenia dotyczące ujęcia zadania w ramach mechaniki ciała stałego:

- 1.1. Założenie o małych odkształceniach elementów rotora pozwalające potraktować zadanie jako liniowo sprężyste. Założenie to pozwala na znaczne skrócenie czasu analizy *MES*, a jego słuszność została potwierdzona w trakcie analiz porównawczych w pełnym zakresie nieliniowym. Wprowadzony w ten sposób błąd w uzyskanych wartościach naprężeń na łopatkach turbiny jest na poziomie 1%.
- 1.2. Założenie o liniowości zjawiska kontaktu pomiędzy elementami rotora (kontakt zamknięty w trakcie całej analizy, bez możliwości separacji i penetracji, bez uwzględnienia tarcia, dopuszczający tylko względne przemieszczenie styczne powierzchni), zamodelowany jako warunek wzajemnie równych przemieszczeń normalnych na powierzchniach będących w kontakcie. Pominięcie nieliniowości kontaktu na zamku jodełkowym łopatk/dysk turbiny na wyniki naprężeń na stopie łopatki nie przekracza 5% na korzyść rozwiązania liniowego, a pominięte w analizie liniowej otwarcie kontaktu zachodzi tylko na ok. 2% powierzchni styku na zamku jodełkowym (wartości uzyskane w trakcie analiz porównawczych, por. [26]).

2. Założenia dotyczące przemieszczeniowych warunków brzegowych:

- 2.1. Założenie o symetrii cyklicznej pola przemieszczeń rotora turbiny, które pozwala na modelowanie tylko jednego segmentu reprezentatywnego dla całego rotora, o rozpiętości kątowej $360^\circ/N$, gdzie N odpowiada liczbie łopatek w wieńcu dysku. Na powierzchniach lateralnych dysku nałożone są warunki symetrii przemieszczeń. Założenie to wynika bezpośrednio z konfiguracji geometrycznej rotora i cyklicznie symetrycznego charakteru obciążeń, nie wprowadza żadnego zauważalnego błędu do wyniku obliczeń.
- 2.2. Założenie o pomijalnym wpływie na rozwiązanie zastąpienia połączenia śrubowego dysku z uszczelnieniami międzystopniowymi warunkami liniowego kontaktu (dopuszczalne są jedynie przemieszczenia styczne do powierzchni kontaktu, bez możliwości separacji i penetracji i bez tarcia). Założenie dopuszczalne wyłącznie przy analizie stanu naprężeń na łopatkach, w rzeczywistości połączenie śrubowe na dysku stanowi jeden z najważniejszych węzłów konstrukcyjnych rotora i jest szczegółowo analizowane za pomocą wyspecjalizowanych modeli *MES*, zawierających dokładny model śruby z wprowadzonym napięciem wstępnym. Złożone zjawiska zachodzące na powierzchniach współpracujących elementów spowodowane m.in. ekspansją termiczną, siłami odśrodkowymi i tarciami

powodują że rzeczywista ocena stanu naprężeń w tym rejonie jest bardzo trudna, zwykle strefa ta stanowi lokację krytyczną rotora ze względu na zmęczenie niskocyklowe.

- 2.3. Założenie o pomijalnym wpływie na rozwiązanie oddziaływania sąsiadujących stopni turbiny i wału oraz wynikające z niego pominięcie rzeczywistych przemieszczeń i sił na powierzchniach uszczelnień międzystopniowych, które pozwala na utwierdzenie osiowe powierzchni czołowej dysku. Założenie to jest dopuszczalne pod warunkiem że chcemy uzyskać wystarczająco dokładne rozwiązanie tylko na łopacie rotora, pozycja osiowa zamka jodełkowego przy odpowiednim zadaniu warunków brzegowych na łopacie nie ma znaczenia dla wyników w interesującej nas strefie.
- 2.4. Założenie o przyjęciu maksymalnego kąta skrzywienia montażowego daszka łopatki, które pozwala na zasymulowanie nominalnych, montażowych przemieszczeń normalnych na powierzchniach kontaktu półki górnej z innymi łopatkami w wieńcu. Założenie to wynika z nieznajomości rzeczywistego kąta skrzywienia łopatki, który ulega zmianie wskutek zużycia powierzchni roboczych daszka łopatki oraz zmienia się zależnie od warunków roboczych. Próby stanowiskowe rotora wykazują że nie przekracza on wartości nominalnej, więc przyjęta wartość jest najbardziej krytyczna z punktu widzenia wytrzymałości łopatki.
- 2.5. Założenie o możliwości zastąpienia rzeczywistego utwierdzenia osiowego łopatki w postaci zamka mechanicznego narzuceniem warunku wzajemnie równych przemieszczeń osiowych na parę węzłów na łopacie i dysku turbiny. Założenie to wynika ze znikomego wpływu sposobu osiowego utwierdzenia łopatki na stan naprężeń w rejonie pióra i ostateczną konfigurację roboczą łopatki, rzeczywisty sposób montażu łopatki pozwala na jej nieznaczne przemieszczenia osiowe w trakcie pracy turbiny, co znajduje odbicie w stanie naprężeń w rejonie zamka jodełkowego.

3. Założenia dotyczące obciążenia ciśnieniem i temperaturą:

- 3.1. Założenie o pomijalnym wpływie na rozwiązanie ciśnień w kanale głównym i kanale wtórnym turbiny na obciążenie wszystkich powierzchni rotora z wyjątkiem pióra łopatki. Założenie to pozwala na obciążenie polem ciśnień tylko powierzchni pióra, na pozostałych powierzchniach łopatki ciśnienie nie jest

przykładane. Jest ono dopuszczalne w analizach łopatek turbin niskiego ciśnienia i zostało potwierdzone w trakcie prób stanowiskowych.

- 3.2. Założenie o pomijalnym wpływie rozszerzalności cieplnej na pole naprężeń na łopacie turbiny (w modelu *MES* uwzględniona została tylko zmiana własności materiałowych z temperaturą, ekspansja termiczna jest pominięta poprzez przyjęcie zerowych współczynników rozszerzalności cieplnej). To założenie jest dopuszczalne tylko w przypadku obliczeń dotyczących samej łopatki (czyli w przypadku analizy będącej przedmiotem niniejszej pracy) – dyski i uszczelnienia międzystopniowe analizowane są z uwzględnieniem rozszerzalności cieplnej, według innej metodologii, inne są też kryteria ich dopuszczalności do eksploatacji (patrz rozdział 4). Podstawą przyjęcia tego uproszczenia jest fakt, że rzeczywista rozszerzalność cieplna materiału łopatki jest niewielka (ok. 10 razy mniejsza niż rozszerzalność cieplna materiału dysku), a sposób zamocowania łopatki w dysku (zamek jodełkowy) powoduje, że ewentualne odkształcenia wywoływane ekspansją termiczną łopatki kompensowane są poprzez jej przemieszczenia w strefie zamka. Z tego samego powodu nie ma też naprężeń powodowanych różnicami rozszerzalności cieplnej obu materiałów. Zamodelowanie rzeczywistego zamka jodełkowego w analizie liniowej jest niemożliwe, a uproszczenie polegające na pominięciu ekspansji termicznej nie wprowadza żadnego błędu w obliczenia rozkładu naprężeń na łopacie – potwierdzają to wyniki prób stanowiskowych silników w których monitoruje się tensometrycznie stan naprężeń na łopatkach.

Efekty rozszerzalności cieplnej w postaci zmiany średnicy dysku oraz wydłużenia łopatki, które mogłyby wpłynąć na wzrost sił bezwładności pochodzących od prędkości kątowej rotora są w modelu uwzględnione poprzez jego przeskalowanie konfiguracji „cieplej”, tj. przygotowanie modelu geometrycznego już w konfiguracji odkształconej wskutek wzrostu temperatury – kształt ten określa się w sposób szacunkowy z wykorzystaniem obliczeń analitycznych.

- 3.3. Założenie upraszczające rzeczywisty rozkład temperatury do pola, które jest funkcją tylko jednej zmiennej przestrzennej (promienia). Założenie to nie wprowadza znaczącego błędu i jest dopuszczalne ze względu na dużą zgodność rozkładu uproszczonego z rzeczywistym.

Wszystkie te założenia są przyjmowane w trakcie inżynierskich analiz rotorów dopuszczających do certyfikacji zarówno łopatek turbiny jak też silników. **LIT?**

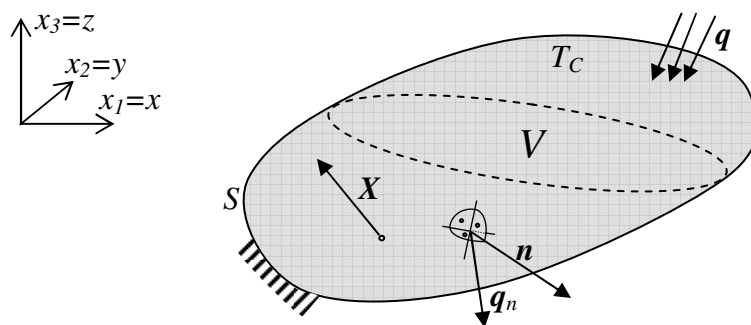
6.2. Model matematyczny

Opis modelu matematycznego segmentu rotora sprowadza się do opisu sformułowania zadania brzegowego teorii termosprężystości i jego rozwiązania za pomocą metody elementów skończonych, która nie będzie tutaj szczegółowo omawiana (*MES* jest przedmiotem szeregu publikacji począwszy od lat 60-tych XX wieku – autor korzystał z [37], [36], [8], [22], [2], [4], [12], [32]). Omówione zostaną jedynie główne zależności prowadzące do sformułowania układu równań algebraicznych stanowiącego faktyczny model matematyczny rotora zbudowany w oparciu o *MES*, na bazie przemieszczeniowego sformułowania wariacyjnego. Takie ujęcie wykorzystywane jest w programie obliczeniowym *Ansys*.

6.2.1. Sformułowanie zadania brzegowego teorii termosprężystości

W celu oceny wartości funkcji przystosowania dla każdego osobnika konieczne jest wyznaczenie pola przemieszczeń i naprężeń dla modelu segmentu rotora, a w szczególności dla interesującej nas strefy łopatki.

W celu sformułowania zadania teorii termosprężystości, zakładamy że mamy do czynienia z kontinuum o objętości V , opisanym w układzie kartezjańskim o osiach x_1, x_2, x_3 (równoważnym układowi x, y, z), o powierzchni zewnętrznej S , którego materiał charakteryzuje się znanymi własnościami izotropowymi liniowo sprężystymi i gęstością ρ . Ciało znajduje się w znanej temperaturze T_C , działają na nie znane siły powierzchniowe q i znane wewnętrzne siły masowe X , na części powierzchni jest podparte, pozostaje w równowadze.



Rys. 6.1 Sformułowanie zadania brzegowego teorii termosprężystości.

Poszukujemy wektorowego pola przemieszczeń u oraz tensorowych pól odkształceń ϵ i przemieszczeń σ , wszystkie pola są funkcją trzech współrzędnych przestrzennych. Zagadnienie brzegowe teorii termosprężystości opisane jest równaniem różniczkowym (pole przemieszczeń w funkcji sił zewnętrznych) z zestawem warunków brzegowych, do rozwiązania którego dysponujemy [24], [4], [31]:

1. Równaniami równowagi wewnętrznej w postaci:

$$\sigma_{ij,j} + X_i \rho = 0 \quad (6.1)$$

$$\sigma_{ij} = \sigma_{ji} \quad (6.2)$$

gdzie:

σ_{ij} – składowe tensora naprężeń,

X_i – składowe sił masowych,

ρ – gęstość materiału,

$i = 1, 2, 3$

$j = 1, 2, 3$

$$\frac{\partial}{\partial x_j} () = ()_{,j}$$

Inna postać tego równania z której będziemy korzystać wyprowadzając równania elementów skończonych to:

$$\Delta^T \sigma + X = 0 \quad (6.3)$$

gdzie:

$\sigma = \{ \sigma_{11} \ \sigma_{22} \ \sigma_{33} \ \sigma_{12} \ \sigma_{23} \ \sigma_{31} \}^T$ – wektorowa (skrótowa) postać tensora naprężeń,

$$\Delta^T = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 & 0 & \frac{\partial}{\partial x_2} & 0 & \frac{\partial}{\partial x_3} \\ 0 & \frac{\partial}{\partial x_2} & 0 & \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_3} & 0 \\ 0 & 0 & \frac{\partial}{\partial x_3} & 0 & \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} \end{bmatrix} \text{ – operator różniczkowy,}$$

$X = \{ X_1 \ X_2 \ X_3 \}^T$ – wektor sił masowych.

2. Równaniami geometrycznymi:

2.1. *Cauchy'ego* (związki pomiędzy składowymi stanu odkształcenia i przemieszczenia) z uwzględnieniem odkształceń wstępnych wywołanych rozszerzalnością cieplną w postaci:

$$\varepsilon_{ij} = \varepsilon^M_{ij} + \varepsilon^C_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}) + \alpha_{ij}(T_C - T_{REF}) \quad (6.4)$$

$$\varepsilon_{ij} = \varepsilon_{ji} \quad (6.5)$$

gdzie:

ε_{ij} – składowe tensora odkształceń całkowitych,

ε^M_{ij} – składowe tensora odkształceń od obciążeń mechanicznych,

ε^C_{ij} – składowe tensora odkształceń od obciążeń cieplnych,

u_i – składowe wektora przemieszczeń,

T_C – temperatura ciała,

T_{REF} – temperatura odniesienia dla której $\varepsilon^C_{ij} = 0$,

α_{ij} – składowe 6-cio elementowego wektora współczynników rozszerzalności cieplnej (zgodnie z przyjętymi założeniami $\alpha_{ij} \neq 0$ dla $i=j$, $\alpha_{ij} = 0$ dla $i \neq j$)

Inna postać tego równania z której będziemy korzystać wyprowadzając równania elementów skończonych to:

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}^M + \boldsymbol{\varepsilon}^C = \Delta \mathbf{u} + T_{C-REF} \boldsymbol{\alpha} \quad (6.6)$$

gdzie:

$\boldsymbol{\varepsilon} = \{\varepsilon_{11} \ \varepsilon_{22} \ \varepsilon_{33} \ \varepsilon_{12} \ \varepsilon_{23} \ \varepsilon_{31}\}^T$ – wektorowa (skrótowa) postać tensora odkształceń całkowitych,

$\boldsymbol{\varepsilon}^M = \{\varepsilon^M_{11} \ \varepsilon^M_{22} \ \varepsilon^M_{33} \ \varepsilon^M_{12} \ \varepsilon^M_{23} \ \varepsilon^M_{31}\}^T$ – wektorowa (skrótowa) postać tensora odkształceń od obciążeń mechanicznych,

$\boldsymbol{\varepsilon}^C = \{\varepsilon^C_{11} \ \varepsilon^C_{22} \ \varepsilon^C_{33} \ \varepsilon^C_{12} \ \varepsilon^C_{23} \ \varepsilon^C_{31}\}^T$ – wektorowa (skrótowa) postać tensora odkształceń od obciążeń cieplnych,

$\mathbf{u} = \{u \ v \ w\}^T$ – pole przemieszczeń,

T_{C-REF} – różnica pomiędzy temperaturą ciała i temperaturą odniesienia,

$\boldsymbol{\alpha} = \{\alpha_{11} \ \alpha_{22} \ \alpha_{33} \ 0 \ 0 \ 0\}^T$ – wektor współczynników rozszerzalności cieplnej.

2.2. *De Saint-Venanta* (warunki ciągłości - nierozdzielności odkształceń) w postaci:

$$e_{ikm} e_{jln} \varepsilon_{kl, mn} = 0 \quad (6.7)$$

gdzie:

e_{ikm} – trójwskaznikowy symbol permutacyjny *Levi-Civita'ego* zdefiniowany jako:

$$\begin{aligned}
& 0 \text{ jeśli } i=k \text{ lub } i=m \\
e_{ikm} &= 1 \text{ jeśli } i,k,m \text{ tworzą permutację cykliczną } 1,2,3 \\
& -1 \text{ jeśli } i,k,m \text{ tworzą permutację cykliczną } 1,3,2
\end{aligned}$$

3. Równaniami sprężystości (uogólnione prawo *Hooke'a*):

$$\varepsilon_{ji} = \frac{1+\nu}{E} \sigma_{ij} - \frac{\nu}{E} \sigma_{kk} \delta_{ij} + \alpha_{ij} \Delta T_{C-REF} \quad (6.8)$$

gdzie:

δ_{ij} – delta *Kroneckera* ($\delta_{ij}=1$ dla $i=j$, $\delta_{ij}=0$ dla $i \neq j$),

E – moduł sprężystości materiału (moduł *Younga*),

ν – współczynnik *Poissona*,

$k = 1,2,3$.

Inna postać tego równania z której będziemy korzystać wyprowadzając równania elementów skończonych to:

$$\sigma = D(\varepsilon - \varepsilon^c) \quad (6.9)$$

gdzie:

$$D^{-1} = \begin{bmatrix} \frac{1}{E_1} & -\frac{\nu_{21}}{E_2} & -\frac{\nu_{31}}{E_3} & 0 & 0 & 0 \\ -\frac{\nu_{12}}{E_1} & \frac{1}{E_2} & -\frac{\nu_{32}}{E_3} & 0 & 0 & 0 \\ -\frac{\nu_{13}}{E_1} & -\frac{\nu_{23}}{E_2} & \frac{1}{E_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{12}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{23}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{31}} \end{bmatrix} \text{ – macierz sztywności}$$

Zgodnie z opisem w rozdziałach 4 i 6.1, w naszym zadaniu rozszerzalność cieplna została pominięta (poprzez przyjęcie zerowych wartości składowych wektora rozszerzalności cieplnej). Prezentowane powyżej równania mają charakter ogólny, zgodny z rzeczywistym modelem matematycznym będącym podstawą sformułowania wykorzystanego w programie *Ansys*. Dla potrzeb naszego zadania możemy uprościć zagadnienie wprowadzając od razu zależność:

$$\alpha_{ij} \equiv 0 \Rightarrow \varepsilon^C_{ij} \equiv 0 \quad (6.10)$$

Co prowadzi do równań:

$$\varepsilon_{ji} = \varepsilon^M_{ij} = \frac{1+\nu}{E} \sigma_{ij} - \frac{\nu}{E} \sigma_{kk} \delta_{ij} \quad (6.11)$$

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon} \quad (6.12)$$

W trzech grupach mamy do dyspozycji 15 równań i szukamy 15 niewiadomych (po 6 składowych tensora naprężenia i odkształcenia, 3 składowe wektora przemieszczenia). W przypadku wyznaczania w pierwszej kolejności składowych tensora naprężenia (rozwiązanie w naprężeniach) możemy dojść do równań *Beltramiiego-Michella*. W przypadku wyznaczania w pierwszej kolejności składowych tensora odkształcenia (rozwiązanie w przemieszczeniach) możemy dojść do równań *Naviera-Lamego* w postaci:

$$G u_{i,jj} + (\lambda + G) u_{i,jj} + \rho X_i = 0 \quad (6.13)$$

gdzie:

G – moduł odkształcenia postaciowego (moduł *Kirchhoffa*), $G = \frac{E}{2(1+\nu)}$,

λ – stała *Lamego*, $\lambda = \frac{2G\nu}{1-2\nu}$.

Do rozwiązania tego równania konieczna jest znajomość warunków brzegowych, które można podzielić na:

1. Warunki brzegowe przemieszczeniowe (*Dirichleta*) w postaci:

$$u_i|_S = u' \quad (6.14)$$

gdzie:

$u_i|_S$ – składowa i przemieszczenia na powierzchni S ,

u' – zadana funkcja (w szczególności stała).

2. Warunki brzegowe dla sił powierzchniowych (*Neumanna*) w postaci:

$$q_{ni} = \sigma_{ji} \nu_{jn} \quad (6.15)$$

gdzie:

q_{ni} – składowe obciążenia powierzchniowego q_n na kierunek i ,

u_{jn} – cosinus kierunkowy między normalną zewnętrzną n do powierzchni obciążonej i półosią x_j układu współrzędnych.

Można jeszcze wyróżnić mieszane warunki brzegowe reprezentujące podłoże sprężyste, opisane równaniem zawierającym sumę iloczynu przemieszczeń na powierzchni i odpowiednich składników macierzy sztywności podłoża oraz naprężenia, ale nie są one wykorzystane w interesującym nas zadaniu.

Równania *Naviera-Lamego* są syntezą statycznej strony zadania teorii sprężystości gdyż wyrażają one warunki równowagi każdego elementu ośrodka i stanowią bezpośredni związek pomiędzy polem przemieszczeń i siłami oddziałującymi na ciało.

W przypadku kiedy jest zdefiniowana postać ciała (model geometryczny), jego własności sprężyste, siły masowe i warunki brzegowe na całej jego powierzchni, równania *Naviera-Lamego* pozwalają na sformułowanie modelu matematycznego i teoretycznie pozwalają znaleźć pole przemieszczeń oraz jego pochodne (odkształcenia) i naprężenia w każdym punkcie ciała. Matematycznie jednak rozwiązanie tego układu równań jest możliwe tylko dla szczególnych przypadków geometrii. Zasadniczo metody rozwiązania zadania brzegowego dzielą się na dwie grupy, jedna z nich zawiera metody w których pomija się niektóre składowe naprężenia jako małe na podstawie fizycznej strony zagadnienia, druga polega na wprowadzeniu hipotez dotyczących rozkładu naprężeń lub stanu odkształcenia (na przykład poprzez przybliżenie pola przemieszczeń funkcją danego typu). W ogólnym przypadku pole przemieszczeń poszukiwane jest w postaci:

$$u = c_0 N_0 + c_1 N_1 + c_2 N_2 + \dots + c_n N_n \quad (6.16)$$

gdzie:

N_i – funkcje interpolacyjne (funkcje kształtu),

c_i – stałe współczynniki.

Metody te prowadzą do przybliżonego rozwiązania układu równań różniczkowych. Funkcje N_i dobierane są tak, aby spełnić warunki brzegowe, ale ich wybór jest w znacznej mierze dowolny. Sposoby wyznaczania wartości stałych współczynników podali m.in. Ritz (1908 r.) i Galerkin (1915 r.). W przypadku skomplikowanej geometrii, problem doboru funkcji N_i staje się bardzo trudny.

Jedynym sposobem rozwiązania zadania brzegowego z wystarczającą dokładnością dla przypadków złożonych modeli geometrycznych są metody numeryczne, polegające na fizycznej (*Metoda Elementów Skończonych*, *Metoda Elementów Brzegowych*) lub matematycznej (*Metoda Różnic Skończonych*, *Metody Bezsiatkowe*) dyskretyzacji obszaru rozwiązania. Wśród wymienionych metod najbardziej rozpowszechniony w zastosowaniach

inżynierskich jest *MES*, którego istotą jest podział rozpatrywanego modelu na podobszary (tzw. elementy skończone) o określonych kształtach w obrębie których obowiązują proste funkcje interpolacyjne. Metoda ta została wykorzystana w tej pracy do oceny wartości funkcji przystosowania generowanych osobników, w oparciu o nią działa program obliczeniowy *Ansys*. Opis wykorzystanych w modelu dyskretnym segmentu rotora elementów skończonych podaje następny rozdział.

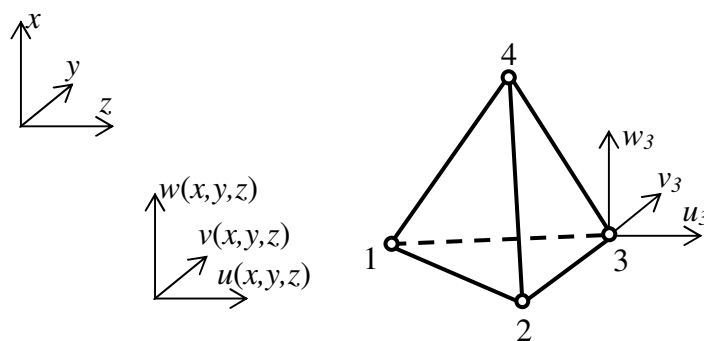
6.2.2. Dyskretyzacja modelu

Metoda Elementów Skończonych polega na dyskretyzacji rozpatrywanego obszaru, czyli jego podziale na podobszary w których pole przemieszczeń przybliżane jest prostymi funkcjami interpolacyjnymi. Każdy z takich obszarów ma określony kształt i wyodrębnione punkty (węzły) w których łączy się z sąsiednimi podobszarami i w których obliczane są wartości wektorowego pola przemieszczeń, będące funkcjami współrzędnych węzłowych. W przypadku wykorzystanych w budowie modelu segmentu rotora elementów te same funkcje wykorzystywane są do opisu geometrii elementu oraz pól sił zewnętrznych (są to tzw. elementy izoparametryczne).

W części dotyczącej *MES* dotychczasowy układ indeksów 1, 2, 3 zostanie zastąpiony indeksami x, y, z , a układ osi x_1, x_2, x_3 równoważnym układem x, y, z . Wynika to z wygody zapisu – stosowany wcześniej zapis wskaźnikowy był wygodniejszy przy użyciu oznaczeń liczbowych, w tym rozdziale indeksy liczbowe będą miały inne znaczenie.

W modelu *MES* segmentu rotora turbiny wykorzystano dwa typy elementów:

1. Element czworosieczny czterowęzłowy typu *SOLID45*.



Rys. 6.2 Element czworosieczny czterowęzłowy.

Element ma w każdym z węzłów (1, 2, 3, 4) trzy przemieszczeniowe stopnie swobody. Pole przemieszczeń ma postać identyczną z wykorzystaną w związku (6.6):

$$\mathbf{u} = \{u(x,y,z) \ v(x,y,z) \ w(x,y,z)\}^T \quad (6.17)$$

Wektor elementowych przemieszczeń węzłowych ma postać:

$$\mathbf{u}^e = \{u_1 \ v_1 \ w_1 \ u_2 \ v_2 \ w_2 \ u_3 \ v_3 \ w_3 \ u_4 \ v_4 \ w_4\}^T \quad (6.18)$$

gdzie:

u_i, v_i, w_i – komponenty przemieszczeń węzłowych dla węzła i ,

Jako funkcje opisujące komponenty pola przemieszczeń wybrane zostały wielomiany trzech zmiennych z czterema składnikami (por. 6.16):

$$u(x,y,z) = c_1 + c_2x + c_3y + c_4z \quad (6.19)$$

$$v(x,y,z) = d_1 + d_2x + d_3y + d_4z \quad (6.20)$$

$$w(x,y,z) = e_1 + e_2x + e_3y + e_4z \quad (6.21)$$

gdzie:

u, v, w – składowe pola przemieszczeń,

$i = 1, 2, 3, 4$ - numer węzła elementu skończonego,

c_i, d_i, e_i – stałe.

Taka funkcja zapewnia liniową zmienność przemieszczeń oraz stałe odkształcenia i naprężenia wzdłuż krawędzi elementu.

Stałe c_i, d_i, e_i można wyznaczyć w funkcji przemieszczeń węzłowych u_i, v_i, w_i wypisując powyższe równania dla każdego z węzłów w postaci:

$$u_1 = c_1 + c_2x_1 + c_3y_1 + c_4z_1 \quad (6.22)$$

i podobnie dla pozostałych komponentów i węzłów. Otrzymujemy układ równań, z którego można wyznaczyć szukane stałe i po wstawieniu ich do równań (6.19) – (6.21) - dostajemy związki w postaci

$$\mathbf{u} = \mathbf{N} \mathbf{u}^e \quad (6.23)$$

gdzie:

$\mathbf{N} = [N_1 \ N_2 \ N_3 \ N_4]$ – macierz funkcji kształtu

Każda z podmacierzy N_i ma następującą budowę:

$$N_i = \begin{bmatrix} N_i & 0 & 0 \\ 0 & N_i & 0 \\ 0 & 0 & N_i \end{bmatrix} \quad (6.24)$$

gdzie:

$$N_i = \frac{1}{6V^e} (a_i + b_i x + c_i y + d_i z) \quad (6.25)$$

a V^e jest objętością elementu czworobocznego obliczaną jako:

$$V^e = \frac{1}{6} \begin{vmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{vmatrix} \quad (6.26)$$

Współczynniki $a_i - d_i$ obliczane są według wzorów:

$$a_i = \begin{vmatrix} x_j & y_j & z_j \\ x_m & y_m & z_m \\ x_p & y_p & z_p \end{vmatrix} \quad b_i = - \begin{vmatrix} 1 & y_j & z_j \\ 1 & y_m & z_m \\ 1 & y_p & z_p \end{vmatrix} \quad c_i = - \begin{vmatrix} x_j & 1 & z_j \\ x_m & 1 & z_m \\ x_p & 1 & z_p \end{vmatrix} \quad d_i = - \begin{vmatrix} x_j & y_j & 1 \\ x_m & y_m & 1 \\ x_p & y_p & 1 \end{vmatrix} \quad (6.27)$$

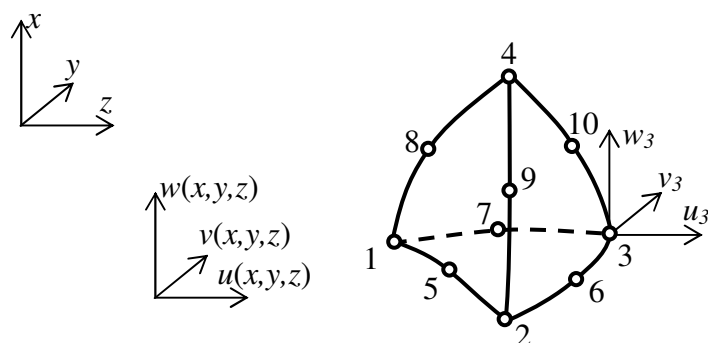
przy czym wartości indeksów i, j, m, p podlegają cyklicznej zmianie.

Można teraz na podstawie związków (6.19 – 21) zapisać składowe wektora przemieszczeń w postaci:

$$u(x, y, z) = \sum_{i=1}^4 N_i u_i = \frac{1}{6V^e} \sum_{i=1}^4 (a_i + b_i x + c_i y + d_i z) u_i \quad (6.28)$$

i podobnie dla $v(x, y, z), w(x, y, z)$.

2. Element czworoboczny dziesięciowęzłowy typu SOLID92.



Rys. 6.3 Element czworoboczny dziesięciowęzłowy.

Element ma w każdym z węzłów (1 - 10) trzy przemieszczeniowe stopnie swobody.

Jako funkcje opisujące komponenty pola przemieszczeń wybrane zostały pełne wielomiany kwadratowe trzech zmiennych z dziesięcioma składnikami:

$$u(x, y, z) = c_1 + c_2 x + c_3 y + c_4 z + c_5 x^2 + c_6 y^2 + c_7 z^2 + c_8 xy + c_9 yz + c_{10} xz \quad (6.29)$$

Taka funkcja zapewnia liniową zmienność odkształceń i naprężeń oraz zmienność przemieszczeń zgodnie z wielomianem drugiego stopnia wzdłuż krawędzi elementu.

Procedura wyznaczania funkcji kształtu jest taka sama jak w przypadku elementu czterowęzłowego, ulegają jedynie odpowiedniemu zwiększeniu rozmiary macierzy, wektorów i wskaźniki sumowania (z 4 do 10).

W praktyce w opisie elementów skończonych wykorzystuje się tzw. współrzędne barycentryczne, które transformuje się na współrzędne globalne za pomocą specjalnych macierzy (macierze *Jakobiego*).

Wstawiając równanie pola przemieszczeń w postaci (6.28) do równania *Cauchy'ego* w postaci (6.4) otrzymujemy związek wyrażający odkształcenia elementowe w funkcji przemieszczeń węzłowych:

$$\boldsymbol{\varepsilon} = \frac{1}{2} \boldsymbol{\Delta} \mathbf{N} \mathbf{u}^e = \mathbf{B} \mathbf{u}^e \quad (6.30)$$

gdzie:

$\mathbf{B} = \boldsymbol{\Delta} \mathbf{N}$ – macierz pochodnych funkcji kształtu.

Wstawiając równanie (6.30) do związku (6.12), otrzymujemy związek wyrażający naprężenia elementowe w funkcji przemieszczeń węzłowych:

$$\boldsymbol{\sigma} = \mathbf{D} \mathbf{B} \mathbf{u}^e \quad (6.31)$$

W przypadku elementu 4-węzłowego, odkształcenia i naprężenia są stałe w obrębie elementu, co wynika z postaci funkcji kształtu. Najczęściej interpretuje się je jako dokładne w środku ciężkości elementu. W przypadku elementu 10-węzłowego, odkształcenia i naprężenia są liniową funkcją współrzędnych geometrycznych.

6.2.3. Podstawowe równania *MES*

Podstawowe równania *MES* wyprowadzone zostaną w oparciu o *Zasadę Prac Przygotowanych (ZPP)* [22], [12], [19], ze względu na syntetyczność opisu. Można je wyprowadzić również w oparciu o równania *Naviera-Lamego* [4] poprzez przekształcenie do sformułowania słabego.

Poszukujemy zależności pomiędzy przemieszczeniami węzłowymi i siłami w obrębie elementu skończonego. *Zasada Prac Przygotowanych* mówi, że suma prac wykonanych na przemieszczeniach przygotowanych przez siły wewnętrzne i zewnętrzne dla ciała pozostającego w równowadze musi być równa zero. Przemieszczeniem przygotowanym $\boldsymbol{\delta} \mathbf{u}$ nazywamy każde dopuszczalne, to znaczy zgodne z więzami i nienaruszające ciągłości ciała przemieszczenie.

Pracę przygotowaną wykonaną przez siły wewnętrzne można zapisać jako:

$$\delta U = \int_V \boldsymbol{\sigma}^T \delta \boldsymbol{\varepsilon} dV \quad (6.32)$$

gdzie:

δU – praca przygotowana wykonana przez siły zewnętrzne,

$\delta \boldsymbol{\varepsilon} = \{ \delta \varepsilon_{11} \delta \varepsilon_{22} \delta \varepsilon_{33} \delta \varepsilon_{12} \delta \varepsilon_{23} \delta \varepsilon_{31} \}^T$ – wektor odkształceń wywołanych przemieszczeniami przygotowanymi,

V – objętość ciała.

Korzystając z równań wyprowadzonych na zakończenie poprzedniego rozdziału możemy łatwo przekształcić ten związek do postaci zdyskretyzowanej, wyrażającej pracę w elemencie skończonym w funkcji przemieszczeń węzłowych. Wstawiając równania (6.30) i (6.31) do równania (6.32) otrzymujemy związek:

$$\delta U^e = \delta \mathbf{u}^{eT} \int_{V^e} \mathbf{B}^T \mathbf{D} \mathbf{B} dV^e \quad (6.33)$$

gdzie:

V^e – objętość elementu skończonego,

$\delta \mathbf{u}^e = \{ \delta u_1 \delta v_1 \delta w_1 \dots \delta u_n \delta v_n \delta w_n \}^T$ – elementowy wektor węzłowych przemieszczeń przygotowanych (dla elementu n -węzłowego)

Pracę przygotowaną wykonaną przez siły zewnętrzne można zapisać jako:

$$\delta L = \int_S \mathbf{p}^T \delta \mathbf{u} dS + \int_V \mathbf{q}^T \delta \mathbf{u} dV \quad (6.34)$$

gdzie:

$\mathbf{p} = \{ p_x p_y p_z \}^T$ – dane pole sił powierzchniowych,

$\mathbf{q} = \{ q_x q_y q_z \}^T$ – dane pole sił objętościowych,

S – powierzchnia ciała,

$\delta \mathbf{u} = \{ \delta u \delta v \delta w \}^T$ – pole przemieszczeń przygotowanych.

W przypadku działania sił skupionych, można je przedstawić jako obciążenie ciągłe korzystając z delty *Diraca* [4]. Aby wyrazić pracę sił zewnętrznych w formie zdyskretyzowanej dla pojedynczego elementu, należy zastąpić ciągłe pola sił powierzchniowych i objętościowych wektorami sił węzłowych a pole przemieszczeń

wektorem przemieszczeń węzłowych. W tym celu należy skorzystać ze związku (6.23) obowiązującego w obrębie elementu skończonego i wstawić go do równania (6.34) w formie zapisanej dla przemieszczeń przygotowanych:

$$\delta \mathcal{L}^e = \delta \mathbf{u}^e \left(\int_{S^e} \mathbf{p}^T N dS + \int_{V^e} \mathbf{q}^T N dV \right) \quad (6.35)$$

gdzie:

S^e – powierzchnia elementu skończonego.

Związki w nawiasie reprezentują odpowiednio węzłowe siły powierzchniowe i węzłowe siły objętościowe w elemencie skończonym. Możemy napisać:

$$\int_{S^e} \mathbf{p}^T N dS + \int_{V^e} \mathbf{q}^T N dV = \mathbf{p}^e + \mathbf{q}^e = \mathbf{f}^e \quad (6.36)$$

gdzie:

$\mathbf{p}^e = \{p_{x1} \ p_{y1} \ p_{z1} \dots \ p_{xn} \ p_{yn} \ p_{zn}\}^T$ – elementowy wektor zastępczych sił węzłowych reprezentujących pole ciśnień (dla elementu n – węzłowego),

$\mathbf{q}^e = \{q_{x1} \ q_{y1} \ q_{z1} \dots \ q_{xn} \ q_{yn} \ q_{zn}\}^T$ – elementowy wektor zastępczych sił węzłowych reprezentujących siły objętościowe (dla elementu n – węzłowego),

$\mathbf{f}^e = \{f_{x1} \ f_{y1} \ f_{z1} \dots \ f_{xn} \ f_{yn} \ f_{zn}\}^T$ – elementowy wektor uogólnionych sił węzłowych (wektor obciążeń dla elementu n – węzłowego).

Można teraz sformułować ZPP dla elementu skończonego wykorzystując związki (6.33), (6.35), (6.36):

$$\begin{aligned} \delta U^e + \delta \mathcal{L}^e &= \delta \mathbf{u}^e{}^T \int_{V^e} \mathbf{B}^T \mathbf{D} \mathbf{B} dV \mathbf{u}^e + \delta \mathbf{u}^e{}^T \mathbf{f}^e = \\ &= \delta \mathbf{u}^e{}^T \left(\int_{V^e} \mathbf{B}^T \mathbf{D} \mathbf{B} dV \mathbf{u}^e + \mathbf{f}^e \right) = 0 \end{aligned} \quad (6.37)$$

Ponieważ wyrażenie to musi być prawdziwe dla dowolnych przemieszczeń przygotowanych, część w nawiasie musi być równa 0, więc:

$$\int_{V^e} \mathbf{B}^T \mathbf{D} \mathbf{B} dV \mathbf{u}^e + \mathbf{f}^e = 0 \quad (6.38)$$

Otrzymaliśmy równanie w postaci:

$$\mathbf{K}^e \mathbf{u}^e + \mathbf{f}^e = 0 \quad (6.39)$$

gdzie:

$$\mathbf{K}^e = \int_{V^e} \mathbf{B}^T \mathbf{D} \mathbf{B} dV - \text{macierz sztywności elementu skończonego.}$$

Przy danych polach sił powierzchniowych i objętościowych równanie to jest funkcjonałem N ponieważ zgodnie z równaniem (6.30) $\mathbf{B}=\Delta\mathbf{N}$. Szukaną niewiadomą pozostaje wektor przemieszczeń \mathbf{u}^e . Niektóre jego elementy można wyznaczyć na podstawie warunków brzegowych (utwierdzeń). Dodatkowo w przypadku ogólnym nie wszystkie elementy wektora \mathbf{f}^e są znane, ponieważ zawierają one nieznanne siły wewnętrzne. Równanie to nie może być wtedy rozwiązane na poziomie elementu.

W ujęciach algorytmicznych macierz sztywności elementu wyznacza się poprzez całkowanie numeryczne, polegające na obliczeniu sumy ważonej wyrażenia podcałkowego w tzw. punktach *Gaussa* [43], [36], [12], których ilość, położenie oraz wagi określone są z góry dla każdego typu elementu.

6.2.4. Globalizacja równań *MES*

Przedstawione równanie obowiązuje w obrębie jednego elementu skończonego. Cała objętość modelu geometrycznego jest przybliżona wieloma takimi elementami. Formuluje się dla nich globalną postać równania (6.39), która zawiera pełne wektory przemieszczeń i obciążeń dla wszystkich węzłów modelu, wykorzystując przy tym addytywność pracy jako wielkości skalarnej (suma prac przygotowanych w poszczególnych elementach równa jest sumie pracy w całym ich zestawie tworzącym model dyskretny). Obowiązują przy tym warunki ciągłości przemieszczeń i warunki równowagi. Współczynniki macierzy sztywności elementów we wspólnych węzłach wyrażane są zależnościami powstałymi w wyniku sumowania odpowiednich wyrazów macierzy elementowych.

Etap tworzenia globalnej macierzy sztywności nosi nazwę *agregacji* elementów skończonych. W programach komputerowych wykorzystuje się do tego tzw. *macierze alokacji*, które dla każdego z elementów zawierają numerację jego węzłów w macierzach globalnych, w celu identyfikacji stopni swobody elementu ze stopniami swobody całego modelu *MES*. Budowa globalnej macierzy sztywności na podstawie macierzy alokacji polega na sumowaniu wyrazów elementowych macierzy sztywności przy użyciu indeksów pochodzących z macierzy alokacji. Podobnie przekształca się elementowe wektory obciążeń węzłowych i przemieszczeń. W ujęciu przeznaczonym do zastosowań w oprogramowaniu obliczeniowym przeprowadza się szereg dodatkowych operacji mających na celu zmniejszenie rozmiaru zadania (np. kondensacja macierzy sztywności, przekształcenie macierzy z wykorzystaniem jej symetrii i pasmowości). Otrzymujemy równanie (6.39) w postaci globalnej:

$$KU + F = 0 \quad (6.40)$$

gdzie:

K – globalna macierz sztywności modelu,

F – globalny wektor uogólnionych sił węzłowych,

U – globalny wektor przemieszczeń węzłowych.

Bezpośrednim rozwiązaniem układu równań jest wektor przemieszczeń węzłowych. Na jego podstawie, dysponując elementowymi wektorami przemieszczeń węzłowych i korzystając z równań (6.30 – 31) można wyliczyć elementowe pola przemieszczeń i naprężeń. W praktyce nie interesują nas pola elementowe, lecz wartości średnie dla każdego elementu. Obliczane są one w punktach *Gaussa*, tych samych które służą wcześniej do całkowania numerycznego w celu wyznaczenia elementowych macierzy sztywności, otrzymane w ten sposób wartości zapewniają największą dokładność rozwiązania (w przeciwieństwie do np. wartości węzłowych lub liczonych w środku ciężkości elementu) [43], [36], [12].

W najbardziej ogólnym przypadku etapy rozwiązania zadania *MES* są następujące:

1. Budowa elementowych macierzy sztywności, obciążeń i przemieszczeń.
2. Budowa wektorów alokacji.
3. Budowa globalnych macierzy sztywności, obciążeń i przemieszczeń na podstawie macierzy lokalnych i wektorów alokacji (agregacja).
4. Rozwiązanie układu równań algebraicznych i obliczenie przemieszczeń oraz sił węzłowych.
5. Obliczenie odkształceń i naprężeń elementowych w punktach *Gaussa*.

W trakcie optymalizacji części łopatki turbiny do oceny wartości funkcji przystosowania wykorzystane zostały elementowe naprężenia główne, wyznaczone na podstawie elementowych komponentów naprężeń w punktach *Gaussa*, oraz masa modelu *MES*, wyznaczana na podstawie zgranulowanej macierzy bezwładności o budowie opisanej w rozdziale 10.3.

6.3. Warunki brzegowe

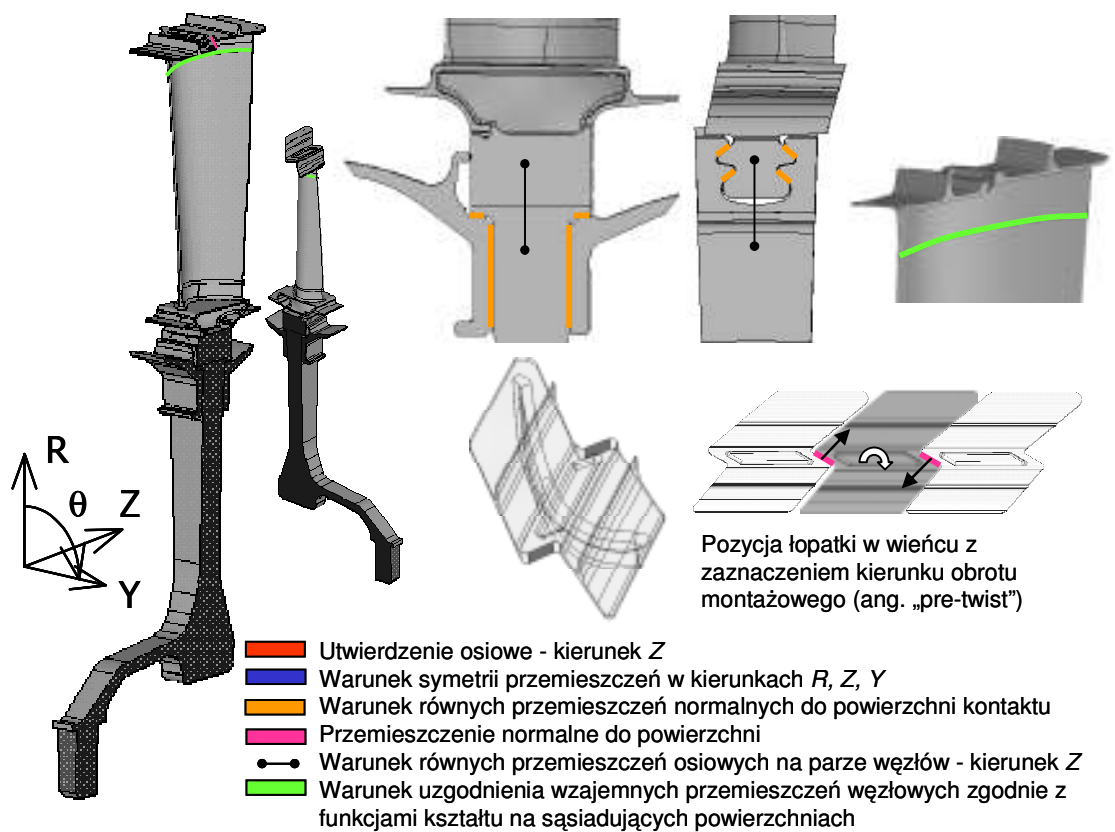
Warunki brzegowe nałożone na model *MES* rotora turbiny przemysłowej dzielą się na trzy typy:

1. Przemieszczeniowe warunki brzegowe (rys. 6.4)

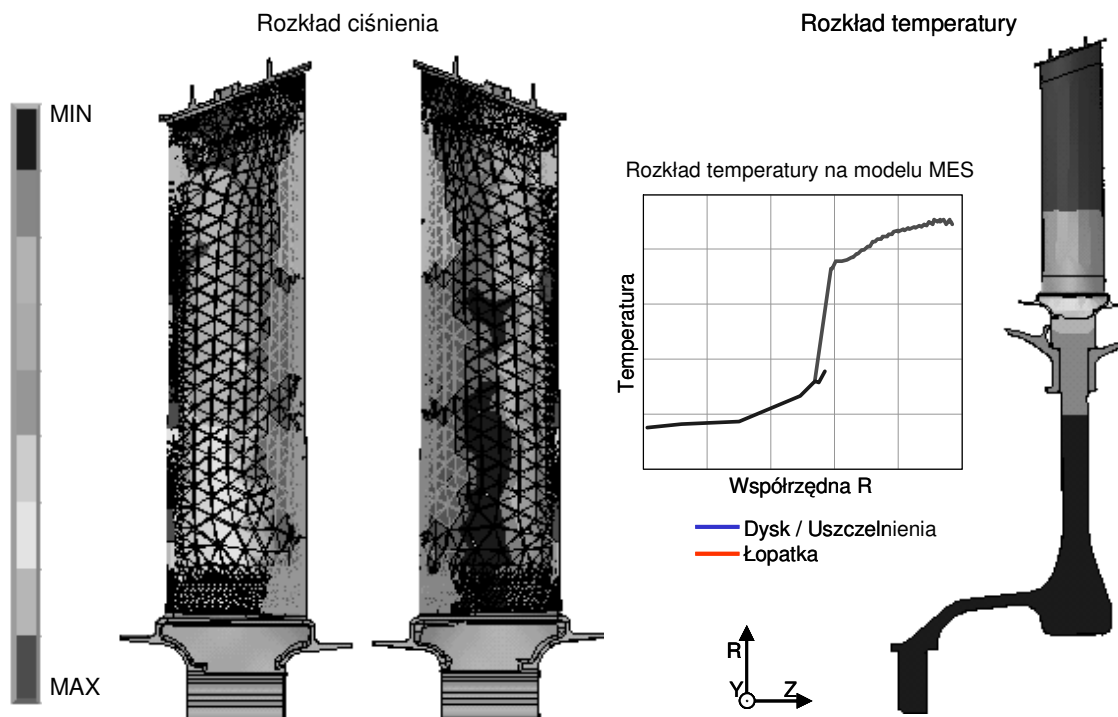
- 1.1. Symetria przemieszczeń na powierzchniach lateralnych dysku i fragmentów uszczelnień międzystopniowych, polegająca na narzuceniu przemieszczeniowych równań równowagi na wszystkie translacyjne stopnie swobody punktów węzłowych znajdujących się na tych powierzchniach.
 - 1.2. Utwierdzenie osiowe na czołowej powierzchni dysku współpracującej z wałem.
 - 1.3. Przemieszczenie normalne na powierzchniach roboczych daszka łopatki odpowiadające skręceniu montażowemu daszka.
 - 1.4. Warunek wzajemnie równych przemieszczeń osiowych na parze węzłów dysk – łopatka symulujące działanie zamka mechanicznego ustalającego osiowe położenie łopatki w dysku.
 - 1.5. Warunek wzajemnie równych przemieszczeń normalnych wszystkich powierzchniach elementów rotora znajdujących się w kontakcie (uszczelnienia międzystopniowe – dysk, łopatka – dysk).
 - 1.6. Warunek uzgodnienia wzajemnych przemieszczeń węzłowych na powierzchni podziału siatki *MES* na wysokości 90% pióra zgodnie z funkcjami kształtu elementów. Warunek ten został wprowadzony ze względu na konieczność opisu obwodu pióra różną ilością krzywych na powierzchniach styku środkowej i górnej części pióra, co uniemożliwiło wprowadzenie tam pokrywających się siatek powierzchniowych elementów skończonych.
2. Warunki brzegowe od sił powierzchniowych (ciśnienia - rys. 6.5).

Ciśnienie na powierzchni pióra łopatki pochodzące od strumienia gazu w kanale głównym turbiny, w postaci dwuwymiarowego pola w płaszczyźnie promieniowo – osiowej przyłożonego na wszystkie powierzchnie pióra łopatki. Rozkład ciśnień otrzymany został jako wynik numerycznej analizy przepływu w kanale głównym turbiny.

3. Prędkość kątowna odpowiadająca prędkości obrotowej rotora w warunkach roboczych.



Rys. 6.4 Model geometryczny rotora turbiny przemysłowej – przemieszczeniowe warunki brzegowe.



Rys. 6.5 Model geometryczny rotora turbiny przemysłowej – rozkład ciśnienia i temperatury.

6.4. Obciążenie cieplne. Zarys analizy przewodzenia ciepła w ujęciu MES.

Do całego modelu rotora przyłożone jest obciążenie cieplne w postaci pola temperatury, uproszczone dla potrzeb analizy strukturalnej do pola jednowymiarowego w funkcji współrzędnej promieniowej. Uproszczenie to jest dopuszczalne ze względu na bardzo dużą zgodność z rzeczywistym rozkładem. Sposób uwzględnienia obciążenia cieplnego w obliczeniach omawiają rozdziały 4 i 6.1 – pod uwagę brana jest tylko zmiana własności materiałowych z temperaturą, zaniedbywana jest natomiast rozszerzalność cieplna materiału. Zmiana średnicy dysku i wydłużenie łopatki na skutek obciążeń cieplnych powodujące wzrost obciążeń bezwładnościowych uwzględniane są poprzez budowę modelu geometrycznego już w konfiguracji przeskalowanej do warunków roboczych (konfiguracja „ciepła”).

Rozkład temperatury wyznaczany jest jako wynik analizy przepływu ciepła z wykorzystaniem MES dla całej turbiny. Obliczenia wykonuje się z wykorzystaniem modelu 2D (przekrój turbiny w płaszczyźnie promieniowo-osiowej), zawierającego całą sieć przepływową kanału turbiny, w stanie ustalonym. Szczegółowy opis tego modelu wykracza poza ramy niniejszej pracy, stanowiłoby to też naruszenie poufności danych przekazanych przez właściciela (*Avio Polska Sp. z o.o.*). Z tego względu problem przewodzenia ciepła w ujęciu MES omówiony zostanie ogólnie. Szczegółowe informacje zawierają publikacje źródłowe [43], [36], [19].

Ponieważ model wykorzystany w analizie przepływu ciepła w turbinie jest dwuwymiarowy, ograniczymy się do przedstawienia podstawowych równań dla funkcji dwóch zmiennych. Obszar A ograniczony brzegiem s wewnątrz którego panuje rozkład temperatury $T_C = T_C(x, y)$. Funkcja opisująca pole temperatury musi spełniać równanie quasi-harmoniczne, będące odpowiednikiem równania równowagi w zagadnieniu równowagi statycznej, w postaci:

$$\frac{\partial}{\partial x} \left(\kappa_x \frac{\partial T_C}{\partial x} \right) + \frac{\partial}{\partial y} \left(\kappa_y \frac{\partial T_C}{\partial y} \right) + Q = 0 \quad (6.41)$$

gdzie:

κ_i – przewodność cieplna w kierunku i ,

Q – prędkość wytwarzania ciepła w obszarze A .

Wyróżnia się dwa rodzaje warunków brzegowych dla rozpatrywanego obszaru, są to warunki w postaci zadanej temperatury T_β na części s_1 brzegu s (warunek *Dirichleta*):

$$T_C|_{s_1} = T_\beta \quad (6.42)$$

i warunki w postaci zadanego przepływu ciepła na części s_2 brzegu s (warunek *Neumanna*).

Przy założeniu $s_1 \cup s_2 = s$ można je sformułować następująco:

$$\left(\kappa_x \frac{\partial T_c}{\partial x} \right) v_{xn} + \left(\kappa_y \frac{\partial T_c}{\partial y} \right) v_{yn} + q_c + \gamma T_{\beta-0} = 0 \quad (6.43)$$

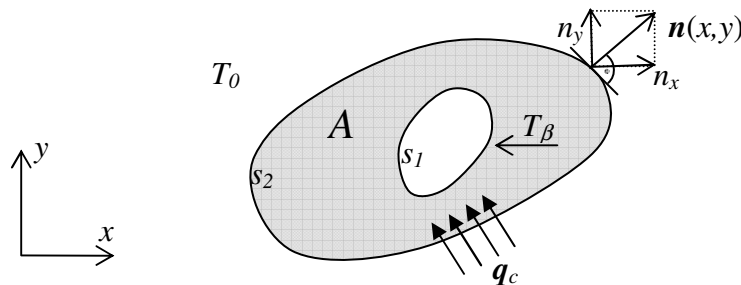
gdzie:

v_{jn} – cosinus kierunkowy między normalną zewnętrzną n do powierzchni obciążonej i półosią j układu współrzędnych,

q_c – wydatek przepływu ciepła przez jednostkę powierzchni,

γ – współczynnik przejmowania ciepła,

$T_{\beta-0} = (T_\beta - T_0)$ – różnica temperatury na brzegu (T_β) i temperatury otaczającego obszar A ośrodka (T_0).



Rys. 6.6 Sformułowanie dwuwymiarowego zadania przewodzenia ciepła w stanie ustalonym.

Zadanie opisane równaniem (6.24) można rozwiązać analitycznie w niewielu przypadkach. Przybliżone rozwiązanie można otrzymać metodą elementów skończonych, dyskretyzując obszar A poprzez podzielenie go na elementy, zbudowane analogicznie jak w przypadku zadania teorii termosprężystości. Rolę przemieszczeń węzłowych przejmują w tym przypadku temperatury węzłowe, do opisu pola temperatury wykorzystuje się te same funkcje kształtu pozwalające zarówno na opis geometrii elementu jak i pól elementowych.

Wykorzystany w modelu termicznym *MES* element ma postać trójwęzłowego trójkąta (analogicznego do elementu *CST* w analizie termomechanicznej) o liniowych funkcjach kształtu i liniowo zmiennej temperaturze wzdłuż krawędzi.

Pole temperatury w elemencie można przedstawić następująco (por. rozdział 6.2.2):

$$T_{C\phi}(x,y) = NT_C^e \quad (6.44)$$

gdzie elementowy wektor temperatur węzłowych ma postać

$$T_C^e = \{T_{C1} \ T_{C2} \ T_{C3}\}^T \quad (6.45)$$

macierz (wektor) funkcji kształtu ma postać

$$N = \{N_1 \ N_2 \ N_3\} \quad (6.46)$$

poszczególne funkcje kształtu opisane są wzorami

$$N_i = \frac{1}{2S^e} \cdot (a_i + b_i x + c_i y) \quad (6.47)$$

S^e jest powierzchnią elementu skończonego obliczaną jako:

$$S^e = \frac{1}{2} \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} \quad (6.48)$$

natomiast współczynniki $a_i - c_i$ obliczane są według wzorów:

$$a_i = \begin{vmatrix} x_j & y_j \\ x_m & y_m \end{vmatrix} \quad b_i = - \begin{vmatrix} 1 & y_j \\ 1 & y_m \end{vmatrix} \quad c_i = - \begin{vmatrix} 1 & x_j \\ 1 & x_m \end{vmatrix} \quad (6.49)$$

przy czym wartości indeksów i, j, m podlegają cyklicznej zmianie.

Zakładając, że funkcja $T_{C\varphi}(x,y)$ spełnia warunki brzegowe, możemy ją podstawić do wzoru (6.43) otrzymując spełnione na obszarze elementu skończonego równanie:

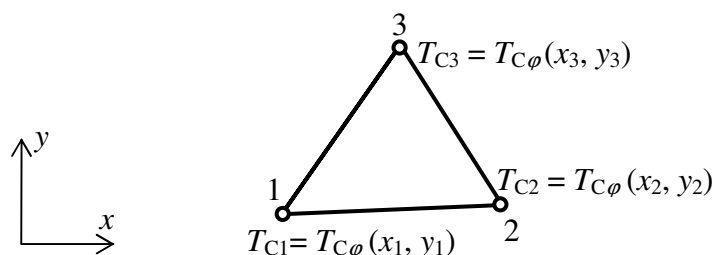
$$\frac{\partial}{\partial x} \left(\kappa_x \frac{\partial T_{C\varphi}}{\partial x} \right) + \frac{\partial}{\partial y} \left(\kappa_y \frac{\partial T_{C\varphi}}{\partial y} \right) + Q = R \neq 0 \quad (6.50)$$

gdzie:

R – funkcja reszty.

W ogólnym przypadku równanie (6.50) nie może być spełnione dokładnie przez funkcję $T_{C\varphi}(x,y)$. Szukamy rozwiązania, które minimalizuje błąd w całym obszarze elementu S^e . Rozwiązanie Jeśli przyrównamy R do zera tożsamościowo, to spełnione będzie równanie:

$$\int_{S^e} f_w R dS^e = 0 \quad (6.51)$$



Rys. 6.7 Element trójkątny trójwęzłowy do obliczeń termicznych.

gdzie f_w jest dowolną funkcją ciągłą zwaną funkcją wagową. Można podstawić równanie (6.50) do równania (6.51) i po uwzględnieniu związku (6.44) otrzymać równanie z trzema niewiadomymi temperaturami węzłowymi. Aby uzyskać układ trzech równań, zamiast funkcji f_w możemy postawić kolejno trzy funkcje wagowe pochodzące z wektora N . Otrzymamy w ten sposób trzy równania postaci:

$$\int_{S^e} N_i \left(\frac{\partial}{\partial x} \left(\kappa_x \frac{\partial T_{C\varphi}}{\partial x} \right) + \frac{\partial}{\partial y} \left(\kappa_y \frac{\partial T_{C\varphi}}{\partial y} \right) \right) dS^e + \int_{S^e} N_i Q dS^e = 0 \quad (6.52)$$

Poszczególne składniki pierwszego członu równania (6.51) można rozpisać następująco:

$$\int_{S^e} N_i \frac{\partial}{\partial x} \left(\kappa_x \frac{\partial T_{C\varphi}}{\partial x} \right) dS^e = \int_{S^e} \frac{\partial}{\partial x} \left(N_i \kappa_x \frac{\partial T_{C\varphi}}{\partial x} \right) dS^e - \int_{S^e} \frac{\partial N_i}{\partial x} \kappa_x \frac{\partial T_{C\varphi}}{\partial x} dS^e \quad (6.53)$$

Podobnie dla kierunku y . Kompletne równanie (6.52) po rozpisaniu wg schematu (6.53) i pogrupowaniu składników będzie wyglądać następująco:

$$\begin{aligned} & \int_{S^e} \left[\frac{\partial}{\partial x} \left(N_i \kappa_x \frac{\partial T_{C\varphi}}{\partial x} \right) + \frac{\partial}{\partial y} \left(N_i \kappa_y \frac{\partial T_{C\varphi}}{\partial y} \right) \right] dS^e + \int_{S^e} N_i Q dS^e = \\ & = \int_{S^e} \left(\frac{\partial N_i}{\partial x} \kappa_x \frac{\partial T_{C\varphi}}{\partial x} + \frac{\partial N_i}{\partial y} \kappa_y \frac{\partial T_{C\varphi}}{\partial y} \right) dS^e \end{aligned} \quad (6.54)$$

Do pierwszego członu lewej strony równania można zastosować twierdzenie *Grena-Gaussa-Ostrogradskiego* i zamienić całkę powierzchniową po elemencie S^e na całkę po brzegu elementu s^e . Pamiętając o podziale brzegu na dwie części ze względu na zadane warunki brzegowe, możemy zamiast całki powierzchniowej zapisać całkę po brzegu i otrzymać równanie w postaci:

$$\begin{aligned} & \int_{s_1^e} N_i \left(k_x \frac{\partial T_{C\varphi}}{\partial x} v_{xn} + k_y \frac{\partial T_{C\varphi}}{\partial y} v_{yn} \right) ds + \int_{s_2^e} N_i \left(k_x \frac{\partial T_{C\varphi}}{\partial x} v_{xn} + k_y \frac{\partial T_{C\varphi}}{\partial y} v_{yn} \right) ds + \int_{S^e} N_i Q dS^e = \\ & = \int_{S^e} \left(\frac{\partial N_i}{\partial x} k_x \frac{\partial T_{C\varphi}}{\partial x} + \frac{\partial N_i}{\partial y} k_y \frac{\partial T_{C\varphi}}{\partial y} \right) dS^e \end{aligned} \quad (6.55)$$

Zgodnie z warunkiem brzegowym *Dirichleta* temperatura na brzegu s_1^e jest znana i całka po s_1^e może zostać zamieniona w stałą, która do układu równań nic nie wnosi. Można ją pominąć dla uproszczenia dalszego rozumowania. Zgodnie z warunkiem brzegowym *Neumanna* całkę po brzegu s_2^e możemy zapisać następująco:

$$\int_{s_2^e} N_i \left(k_x \frac{\partial T_{C\varphi}}{\partial x} v_{xn} + k_y \frac{\partial T_{C\varphi}}{\partial y} v_{yn} \right) ds = - \int_{s_2^e} N_i q_C ds - \int_{s_2^e} N_i \gamma (T_\beta - T_0) ds \quad (6.56)$$

Po podstawieniu równania (6.55) do równania (6.54) i uporządkowaniu otrzymujemy:

$$\int_{S^e} \left(\frac{\partial N_i}{\partial x} k_x \frac{\partial T_{C\varphi}}{\partial x} + \frac{\partial N_i}{\partial y} k_y \frac{\partial T_{C\varphi}}{\partial y} \right) dS^e = \int_{s_2^e} N_i q_C d\psi + \int_{s_2^e} N_i \gamma (T_\beta - T_0) ds - \int_{S^e} N_i Q dS^e \quad (6.57)$$

Jeśli teraz skorzystamy z równania (6.44) i zastąpimy funkcję $T_{C\phi}(x,y)$ iloczynem wektorów temperatur węzłowych i funkcji kształtu oraz wyrazimy temperaturę T_β jako całkę po brzegu ψ_2 , otrzymamy układ równań:

$$\begin{aligned} & \left(\int_{s^e} \left(\frac{\partial N_i}{\partial x} k_x \frac{\partial N}{\partial x} + \frac{\partial N_i}{\partial y} k_y \frac{\partial N}{\partial y} \right) dS^e + \int_{s^e_2} N_i \gamma \mathcal{N} ds \right) \mathbf{T}_C^e = \\ & = \int_{s^e_2} N_i q_C d\psi - \int_{s^e} N_i Q dS^e + \int_{s^e_2} N_i \gamma \mathcal{T}_0 ds \end{aligned} \quad (6.58)$$

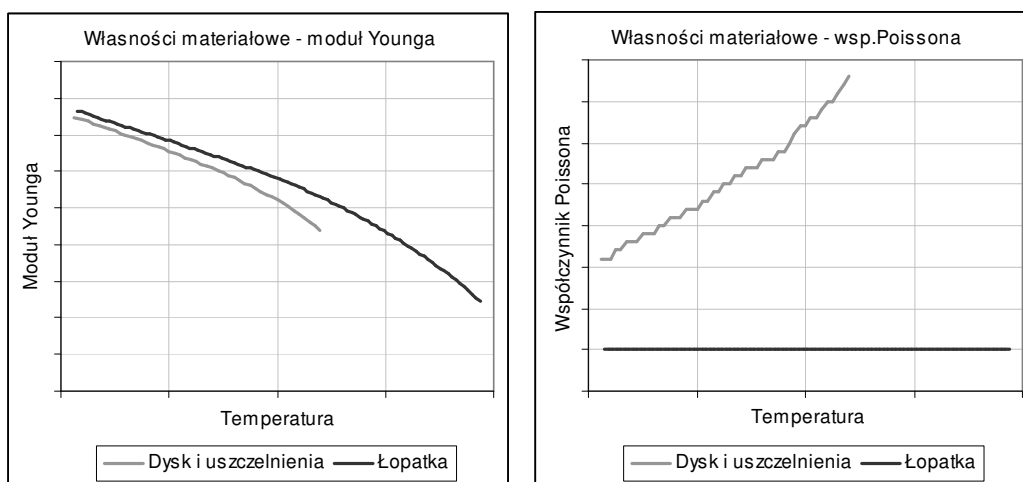
Ten układ równań jest odpowiednikiem układu (6.38) w zadaniu termosprężystości – jedyną niewiadomą jest wektor temperatur węzłowych. Szukając analogii można powiedzieć, że macierzy sztywności odpowiada suma całek po lewej stronie równania (z uwzględnieniem warunku sprężystego podłoża w postaci konwekcji na brzegu), natomiast prawa strona wyraża warunki brzegowe odpowiadające kolejno warunkom sił powierzchniowych i warunkom przemieszczeniowym.

Przedstawione wyżej równania dotyczą pojedynczego elementu skończonego. Podobnie jak w przypadku analizy wytrzymałościowej, cały obszar modelu podzielony jest na wiele elementów, których macierze podlegają agregacji i przekształceniu do postaci globalnej z wykorzystaniem macierzy alokacji (w sposób analogiczny do opisanego w rozdziale 6.2.4), wektorem niewiadomych staje się globalny wektor temperatur węzłowych – istnieje więc tylko jedna, niezależna od układu współrzędnych niewiadoma w każdym węźle. Rozwiązanie układu n równań algebraicznych z n niewiadomymi nie stanowi problemu technicznego.

6.5. Własności materiałów

Zarówno dysk z uszczelnieniami międzystopniowymi jak i łopátka turbiny wykonane są ze stopów żarowytrzymałych na osnowie niklu o własnościach izotropowych, silnie uzależnionych od temperatury. Jak zostało zaznaczone wcześniej, własności materiałowe nie uwzględniają rozszerzalności cieplnej, która jest przyjmowana jako zerowa. Wynika to z metodologii obliczeń wytrzymałościowych łopatek turbin opisaną w rozdziale 4 i 6.1.

Łopátka wykonana jest w procesie odlewania, natomiast dysk i uszczelnienia to elementy kute. Odlew łopátki poddawany jest obróbce skrawaniem w bardzo ograniczonym zakresie, tylko na powierzchniach kontaktu z dyskiem i sąsiednimi łopatkami, natomiast całość powierzchni elementu jest śrutowana. Dysk i uszczelnienia poddawane są obróbce skrawaniem i śrutowaniu na wszystkich powierzchniach.



Rys. 6.8 Własności materiałów wykorzystanych w rotorze turbiny przemysłowej.

6.6. Model dyskretny MES

W celu minimalizacji kosztu numerycznego obliczeń, zostały opracowane dwa modele *MES* różniące się dokładnością siatki. Model dokładniejszy („szczegółowy”) służył w celu sprawdzenia poprawności parametryzacji modelu, działania procedury i dostarczenia wyników porównawczych dla konfiguracji „0”, natomiast model mniej dokładny („uproszczony”) wykorzystany został bezpośrednio w procesie optymalizacji.

Oba modele zbudowane zostały z wykorzystaniem elementów skończonych typu czworościennego, dziesięciowęzłowych i czterowęzłowych, o funkcjach kształtu w postaci wielomianu drugiego i pierwszego stopnia. Każdy węzeł ma trzy przemieszczeniowe stopnie swobody. W strefie optymalizowanej siatka elementów skończonych jest identyczna dla obu modeli. W modelu uproszczonym elementy na dysku i uszczelnieniach międzystopniowych (nieoptymalizowana część modelu), gdzie istotne było jedynie poprawne zamodelowanie sztywności, zostały zredukowane do elementów czterowęzłowych o liniowej funkcji kształtu. Pozwoliło to na znaczne obniżenie ilości stopni swobody i kosztu numerycznego obliczeń.

Tabela 6.1 pokazuje porównanie parametrów obu modeli. Wyniki naprężenia, czas analizy, ilość stopni swobody i masa zostały znormalizowane względem modelu szczegółowego. Można stwierdzić, że model uproszczony stanowi wystarczające przybliżenie modelu szczegółowego dla rozpatrywanych warunków brzegowych, ponieważ wprowadzane na skutek uproszczenia błędy masy całego modelu wynoszą ok. 3%, przy identycznej masie części optymalizowanej i różnicy naprężeń w interesującym nas rejonie na poziomie 0.4%, pozwalając przy tym na skrócenie czasu analizy ponad 80%, co stanowi kluczowe zagadnienie ze względu na konieczność minimalizacji czasu oceny funkcji przystosowania w

trakcie optymalizacji. Należy też zwrócić uwagę, że na skutek uproszczenia siatki elementów skończonych model prostszy zwraca wyniki nieco zawyżone, a więc można się spodziewać że wyniki otrzymane w trakcie optymalizacji będą po „bezpiecznej stronie”. Znormalizowane wartości naprężenia i masy w trakcie optymalizacji obliczane były względem modelu w konfiguracji uproszczonej.

Tabela 6.1. Porównanie wyników analizy MES modelu segmentu rotora turbiny w warunkach roboczych - konfiguracja uproszczona i szczegółowa.

Model MES części nieoptymalizowanej	Model MES części optymalizowanej	Ilość stopni swobody całego modelu*	Masa całego modelu*	Czas analizy MES*	Maks. naprężenie główne na części optymalizowanej*
Szczegółowy	Identyczny	1.000	1.000	1.000	1.000
Uproszczony		0.359	1.033	0.187	1.004

*znormalizowane względem modelu szczegółowego

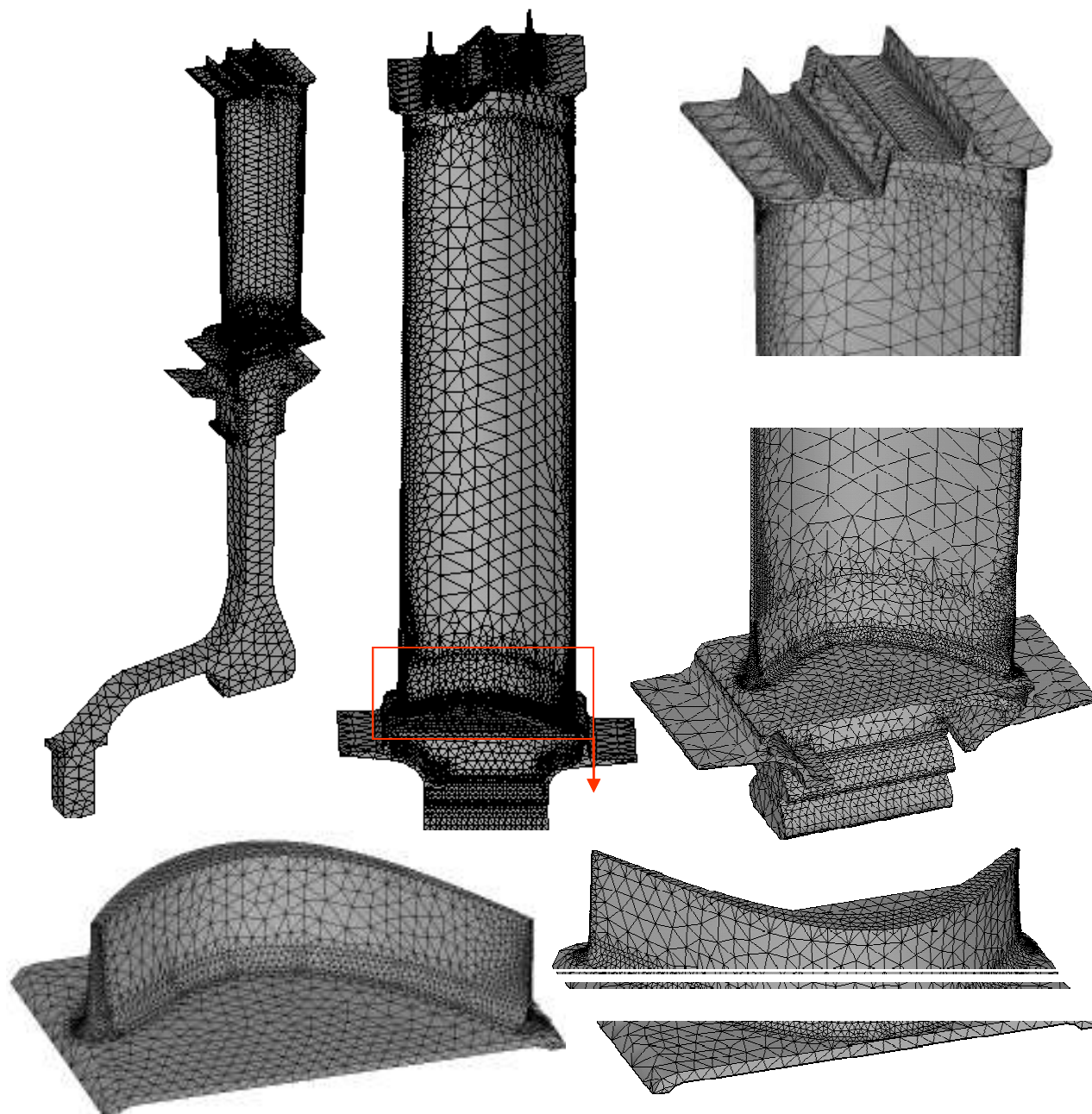
6.6.1. Model szczegółowy

Model szczegółowy opracowany został w celu sprawdzenia poprawności sposobu parametryzacji strefy optymalizowanej, dostarczenia dokładnych wyników porównawczych dla konfiguracji „0” i sprawdzenia działania procedury optymalizacyjnej.

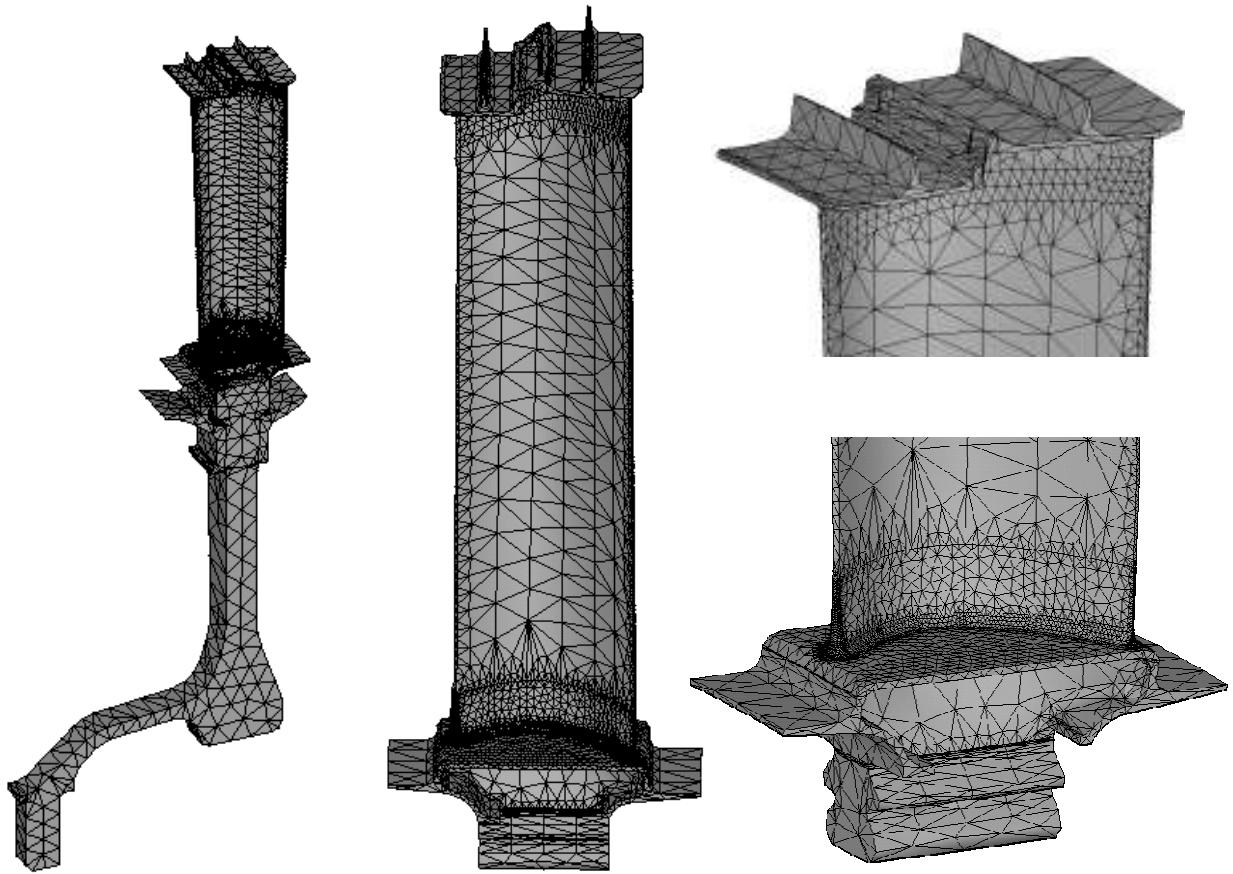
Siatka elementów skończonych (rys. 6.9) zbudowana została przy zachowaniu wysokiej zgodności z modelem geometrycznym, przy różnicy mas między modelem geometrycznym i MES poniżej 1%. Cały model wykonany został z wykorzystaniem elementów czworościennych dziesięciowęzłowych o kwadratowej funkcji kształtu.

6.6.2. Model uproszczony

Model uproszczony opracowany został w celu bezpośredniego wykorzystania w procedurze optymalizacyjnej. Model łopatki wykonany został z wykorzystaniem elementów czworościennych dziesięciowęzłowych o kwadratowej funkcji kształtu. Model dysku oraz uszczelnień międzystopniowych zbudowany został z wykorzystaniem elementów czworościennych czterowęzłowych o liniowej funkcji kształtu w celu zmniejszenia ilości stopni swobody. Rys. 6.10 pokazuje siatkę elementów skończonych na modelu uproszczonym, elementy o liniowej funkcji kształtu zaznaczono kolorem niebieskim.



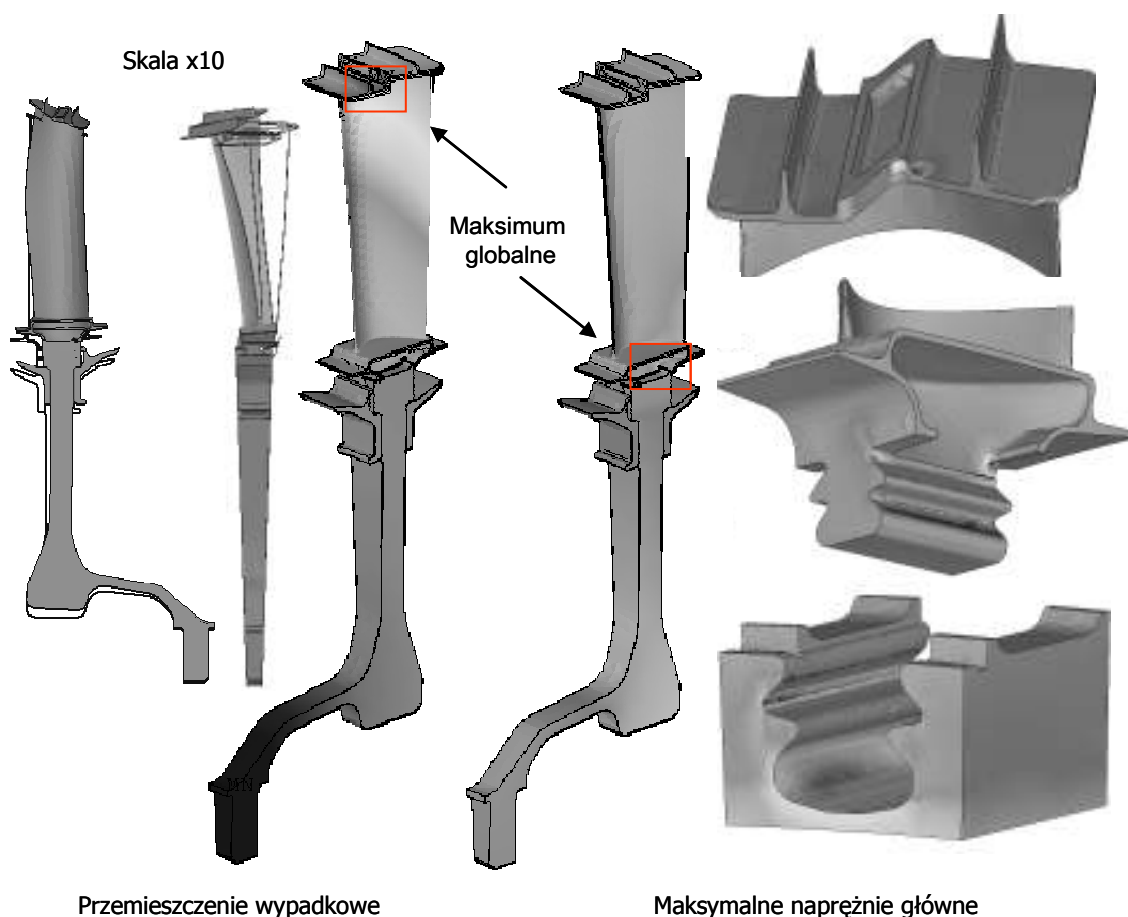
Rys. 6.9 Siatka elementów skończonych na modelu szczegółowym segmentu rotora turbiny.



Rys. 6.10 Siatka elementów skończonych na modelu uproszczonym segmentu rotora turbiny. Elementy o zredukowanej liczbie węzłów zaznaczone kolorem niebieskim.

6.7. Wyniki analizy MES segmentu rotora turbiny w warunkach roboczych

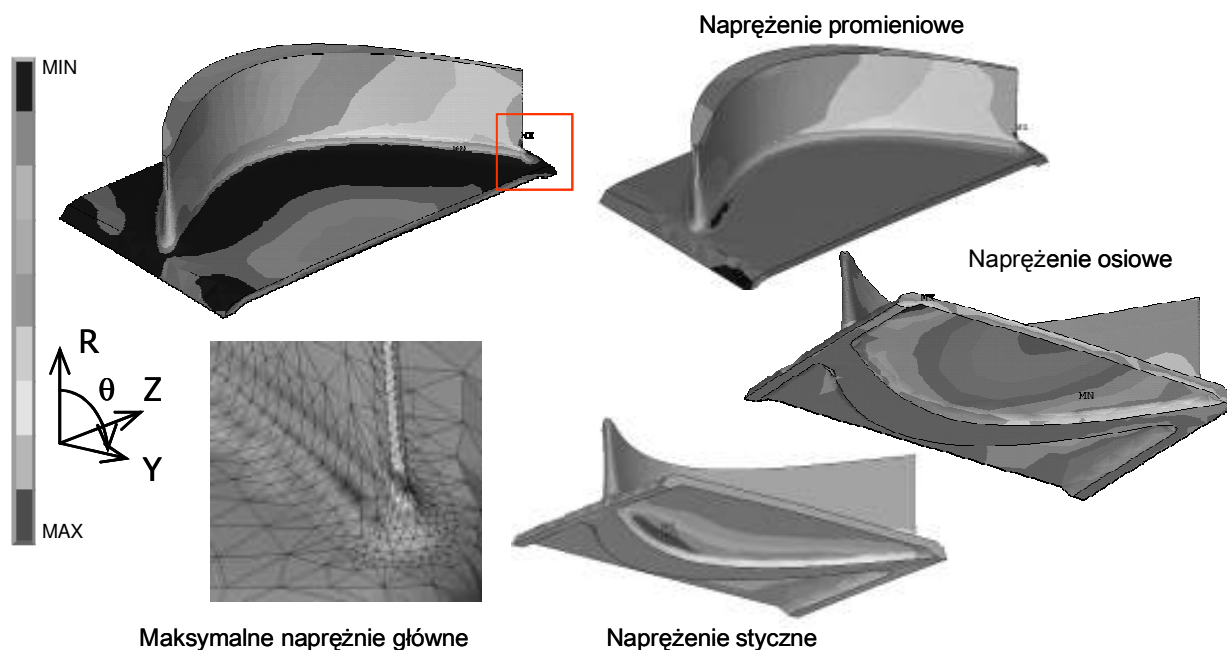
Przedstawione wyniki pochodzą z modelu szczegółowego. Warunki brzegowe analizy opisane zostały w rozdziale 6.3. Ponieważ przedmiotem pracy nie jest analiza stanu przemieszczeń lub naprężeń na elementach rotora, wyniki zaprezentowane zostały skrótowo. Rys. 6.11 przedstawia kształt segmentu po odkształceniu i rozkład maksymalnego naprężenia głównego na całym modelu. Rys. 6.12 przedstawia szczegółowe wyniki analizy *MES* na optymalizowanej części łopaty w postaci wszystkich komponentów naprężenia w układzie analizy oraz maksymalnego naprężenia głównego. Rys. 6.13 przedstawia szczegółowe wyniki analizy *MES* na optymalizowanej części łopaty w postaci wszystkich komponentów przemieszczenia w układzie analizy oraz przemieszczenia wypadkowego.



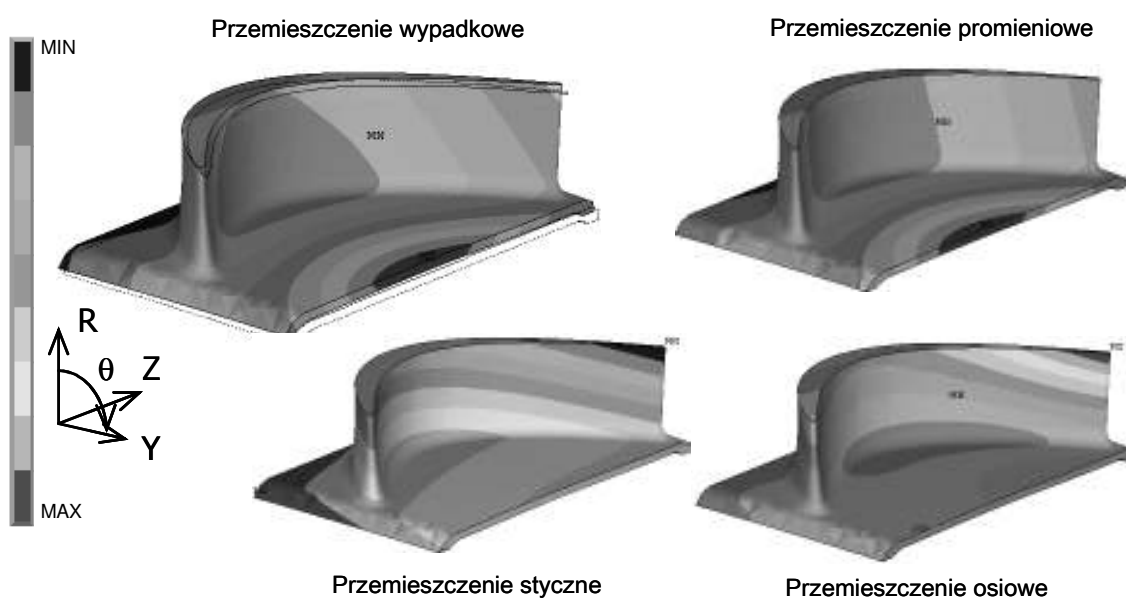
Rys. 6.11 Wyniki analizy MES na segmencie rotora turbiny w warunkach roboczych.

Łopatką poddawana jest głównie zginaniu (na skutek obciążeń aerodynamicznych) i rozciąganiu (na skutek sił odśrodkowych spowodowanych prędkością kątową dysku). Efekt sił odśrodkowych widać też w odkształceniach dysku.

Globalne maksimum naprężeń głównych w modelu występuje na łopatkce, po stronie cisnącej, na połączeniu zamka jodełkowego ze stopą. Spowodowane jest ono zginaniem pochodzącym od sił gazu, łopatką jest osadzona w zamku powyżej którego następuje skokowa zmiana przekroju i występuje karb będący miejscem spiętrzenia naprężeń. Kolejne wyteżone miejsca łopatkki to strefa zaokrąglenia w pobliżu powierzchni kontaktu z sąsiednią łopatką w wieńcu. Oba te miejsca można jednak stosunkowo łatwo poddać modyfikacji kształtu (np. zwiększając promienie zaokrąglenia lub podnosząc krawędź półki dolnej), nie pełnią one żadnej istotnej funkcji aerodynamicznej, ograniczenia narzucają jedynie parametry procesu technologicznego i konieczność utrzymania jak najniższej masy łopatkki. Zmiany te wprowadza się do projektu stosunkowo późno, po wyważeniu łopatkki i ustaleniu kształtu pióra, a także strefy przejścia pomiędzy piórem i półką dolną.



Rys. 6.12 Wyniki analizy MES na optymalizowanej części łopatki turbiny w warunkach roboczych - naprężenia.



Rys. 6.13 Wyniki analizy MES na optymalizowanej części łopatki turbiny w warunkach roboczych - przemieszczenia.

Na dysku największe naprężenia główne występują w zamku jodełkowym i spowodowane są rozciąganiem na skutek oddziaływania łopatki. Wyniki otrzymane dla dysku należy jednak traktować jako szacunkowe, w rzeczywistej analizie inżynierskiej elementów rotora należy uwzględnić cały szereg zjawisk pominiętych tutaj dla uproszczenia analizy

(patrz rozdział 6.1). W rzeczywistości maksimum naprężeń na dysku występuje najczęściej w rejonie połączenia śrubowego z uszczelnieniami międzystopniowymi, które w naszym modelu zostało uproszczone do warunków liniowego kontaktu – nie wprowadza to zauważalnego błędu w wyniki otrzymane dla samej łopatki.

W strefie optymalizowanej, zgodnie z oczekiwaniami maksimum naprężeń głównych występuje w interesującym nas rejonie, nieco powyżej półki dolnej, na krawędzi zaokrąglenia pomiędzy piórem i półką dolną, na krawędzi spływu. Spowodowane jest ono głównie rozciąganiem, skręceniem i zginaniem łopatki pochodzącymi od sił odśrodkowych i sił aerodynamicznych. Skokowa zmiana przekroju jaką stanowi strefa przejścia pióro – półka dolna stanowi naturalną strefę koncentracji naprężeń wywołanych odkształceniami pióra łopatki.

7. EWOLUCYJNA OPTIMALIZACJA JEDNOKRYTERIALNA KSZTAŁTU ŁOPATKI ZE WZGLĘDU NA MINIMUM MAKSYMALNEGO NAPRĘŻENIA GŁÓWNEGO

7.1. Sformułowanie zadania optymalizacji ewolucyjnej

Celem obliczeń była minimalizacja maksymalnych naprężeń głównych na optymalizowanej części łopatkki przy zachowaniu masy poniżej wartości dopuszczalnej, czyli znalezienie chromosomu T^* spełniającego warunki:

$$\forall T \in \Omega \quad F(T^*) \leq F(T) \quad (7.1)$$

przy ograniczeniu

$$\bar{M}(T^*) \leq 1 \quad (7.2)$$

gdzie:

Ω – przestrzeń rozwiązań dopuszczalnych,

T – chromosom (wektor parametrów geometrycznych) osobnika,

T^* – chromosom dla którego funkcja $F(T)$ osiągnie optimum globalne,

$F(T)$ – wartość funkcji przystosowania dla osobnika o chromosomie T ,

$\bar{M}(T)$ – znormalizowana masa dla osobnika o chromosomie T obliczana jako stosunek masy modelu o chromosomie T do masy modelu w konfiguracji „0”:

$$\bar{M}(T) = \frac{M(T)}{M_0} \equiv \frac{\int \rho dV}{\int_{V_0} \rho dV} \quad (7.3)$$

$M(T)$ – masa obliczona dla osobnika o chromosomie T ,

$V(T)$ – objętość obliczona dla osobnika o chromosomie T ,

ρ – gęstość,

M_0 – masa optymalizowanej części pióra łopatkki modelu w konfiguracji „0”,

V_0 – objętość optymalizowanej części pióra łopatkki modelu w konfiguracji „0”.

Sformułowane wyżej zadanie optymalizacji warunkowej sprowadzone zostało do zagadnienia optymalizacji bezwarunkowej poprzez zastosowanie metody funkcji kary.

Funkcja przystosowania może zostać przedstawioną za pomocą następującego wzoru:

$$F(\mathbf{T}) = \bar{\sigma}(\mathbf{T}) \cdot P_m(\mathbf{T}) \quad (7.4)$$

gdzie:

$\bar{\sigma}(\mathbf{T})$ – znormalizowane maksymalne naprężenie główne dla osobnika o chromosomie \mathbf{T} obliczane wg wzoru:

$$\bar{\sigma}(\mathbf{T}) = \frac{\sigma_1(\mathbf{T})}{\sigma_{10}} \quad (7.5)$$

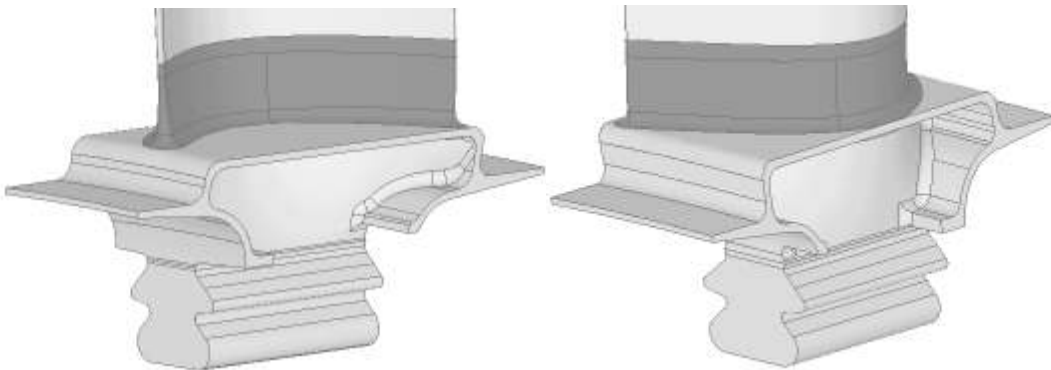
$\sigma_1(\mathbf{T})$ – maksymalne naprężenie główne na optymalizowanej części pióra łopatki dla osobnika o chromosomie \mathbf{T} ,

σ_{10} – maksymalne naprężenie główne na optymalizowanej części pióra łopatki modelu w konfiguracji „0”,

$P_m(\mathbf{T})$ – masowa funkcja kary dla osobnika o chromosomie \mathbf{T} obliczana wg wzoru:

$$P_m(\mathbf{T}) = \begin{cases} 1 & \text{gdy } \bar{M}(\mathbf{T}) \leq 1 \\ 2 & \text{gdy } \bar{M}(\mathbf{T}) > 1 \end{cases} \quad (7.6)$$

Naprężenie główne odczytywane jest z węzłów powierzchniowych pióra łopatki do wysokości strefy optymalizowanej (rys. 7.1). Funkcja kary została wprowadzona w celu zapobiegania przyrostowi masy części optymalizowanej powyżej masy części w konfiguracji geometrycznej „0”.



Rys. 7.1 Strefa odczytu maksymalnego naprężenia głównego z optymalizowanej części łopatki (powierzchnie ciemnoszare).

W celu zapewnienia czytelności przedstawionych danych, osobniki o funkcji kary $P_m(\mathbf{T}) = 2$ zostały wykluczone z obliczeń odchylenia standardowego funkcji przystosowania i wartości średnich funkcji przystosowania, masy i naprężenia. Obliczenia te są przeprowadzane tylko dla osobników o dopuszczalnej masie. Na wykresie pokazującym zmianę masy umieszczona została krzywa obrazująca ilość ukaranych osobników.

7.2. Etapy optymalizacji ewolucyjnej.

Ogólny opis działania algorytmu ewolucyjnego podaje rozdział 3. Poniżej opis ten zostanie sprecyzowany i rozszerzony.

Każdy chromosom identyfikowany jest przez dwie zmienne – numer pokolenia oznaczany poniżej jako k i pozycję w pokoleniu oznaczaną poniżej jako j . Do identyfikacji genu konieczne jest jeszcze podanie jego pozycji w chromosomie, oznaczanej poniżej jako i . Ze względu na przejrzystość tekstu, notacja ta nie jest zachowywana konsekwentnie dla każdego równania, tam gdzie rozważania mają charakter ogólny i obowiązują dla tak samo wszystkich pokoleń, chromosomów lub genów, pomijane są wartości wskaźników i, j, k .

Chromosom osobnika jest zapisu rzeczywistoliczbowego i zgodnie z opisem w rozdziale 5, można go przedstawić jako:

$$\mathbf{T}_j = \{ t_1, \dots, t_8, t_9, \dots, t_{16}, t_{17} \} = \{ R1_1, \dots, R1_8, R2_1, \dots, R2_8, H_j \} \quad (7.7)$$

gdzie:

\mathbf{T}_j – j -ty osobnik (chromosom) populacji bazowej,

t_i – i -ty gen chromosomu j ,

$R1_k$ – promień górnego łuku w k -tym przekroju strefy sparametryzowanej,

$R2_k$ – promień dolnego łuku w k -tym przekroju strefy sparametryzowanej,

H_j – wysokość strefy optymalizowanej danego osobnika.

Każda populacja bazowa danego pokolenia składa się z K osobników i może być zapisana jako:

$$\boldsymbol{\tau}^k = [\mathbf{T}_1, \mathbf{T}_2, \dots, \mathbf{T}_K] \quad (7.8)$$

gdzie:

$\boldsymbol{\tau}^k$ – populacja bazowa k -tego pokolenia,

Przyjęto, że rozmiar populacji osobników jest stały w całym procesie ewolucji i wynosi $K = 30$.

Populacja bazowa w pierwszym pokoleniu ($\boldsymbol{\tau}^1$) składa się z losowo wygenerowanych chromosomów, dopuszczalny zakres każdego z parametrów podaje tabela 5.1. Losowy proces generacji genów dla osobnika o chromosomie k może być przedstawiony jako:

$$t_i = \begin{cases} \xi_U(R1_{\min}, R1_{\max}) & \text{gdy } i \in \langle 1, 8 \rangle \\ \xi_U(H_{\min}, H_{\max}) & \text{gdy } i = 17 \\ \xi_U(R2_{\min}, t_{17}) & \text{gdy } i \in \langle 9, 16 \rangle \end{cases} \quad (7.9)$$

gdzie:

$\xi_U(r_1, r_2)$ – zmienna uzyskana w procesie losowej generacji z rozkładem normalnym przy górnej i dolnej granicy przedziału dopuszczalnego odpowiednio r_1, r_2 ,

$R1_{\min}, R1_{\max}$ – minimalna i maksymalna dopuszczalna wartość promienia $R1$ (tabela 5.1),

$R2_{\min}$ – minimalna dopuszczalna wartość promienia $R2$ (tabela 5.1),

H_{\min}, H_{\max} – minimalna i maksymalna dopuszczalna wartość wysokości H (tabela 5.1).

Losowanie wartości genów przeprowadzane jest do momentu utworzenia K osobników. Każdy chromosom jest następnie przekazywany jako dane wejściowe do procedury optymalizacyjnej, która po zbudowaniu i modelu MES strefy optymalizowanej, analizie MES i ocenie wartości funkcji przystosowania przeprowadza budowę nowego pokolenia. Składa się ona z etapów przedstawionych poniżej.

7.2.1. Reprodukacja.

Schemat reprodukcji opiera się na reprodukcji progowej (szczególnym przypadku reprodukcji rangowej) ze stałą w obrębie pokolenia wartością prawdopodobieństwa wyboru osobnika [6], [3]. Osobniki sortowane są według rosnącej wartości funkcji przystosowania i każdemu z nich nadawana jest ranga odpowiadająca numerowi osobnika w tym uszeregowaniu. Na jej podstawie obliczane jest prawdopodobieństwo reprodukcji osobnika. Proces ten może zostać przedstawiony jako:

$$p(\mathbf{T}) = \begin{cases} 0 & \text{gdy } F(\mathbf{T}) \geq (F^k_{\max} - F^k_{\min})/2 \\ 1 & \text{gdy } F(\mathbf{T}) < (F^k_{\max} - F^k_{\min})/2 \end{cases} \quad (7.10)$$

lub:

$$p(\mathbf{T}) = \begin{cases} 0 & \text{gdy } r(\mathbf{T}) > K/2 \\ 1 & \text{gdy } r(\mathbf{T}) \leq K/2 \end{cases} \quad (7.11)$$

gdzie:

$p(\mathbf{T})$ – prawdopodobieństwo reprodukcji osobnika o chromosomie \mathbf{T} ,

F^k_{\min} – minimalna wartość funkcji przystosowania dla pokolenia k ,

F^k_{\max} – maksymalna wartość funkcji przystosowania dla pokolenia k ,

$r(\mathbf{T})$ – ranga osobnika o chromosomie \mathbf{T} obliczana jako pozycja w wektorze posortowanym wg rosnącej wartości funkcji przystosowania.

W wyniku reprodukcji do dalszych etapów przetwarzania chromosomu wybranych zostaje $K/2 = 15$ osobników o najniższych wartościach funkcji przystosowania, stanowią one populację tymczasową \mathbf{v} w danym pokoleniu. Proces reprodukcji można przedstawić jako:

$$\boldsymbol{\tau}^k = [T_1, T_2, \dots, T_{30}] \xrightarrow{\text{Reprodukcja}} \mathbf{v}^k = [Y_1, Y_2, \dots, Y_{15}] \quad (7.12)$$

gdzie:

\mathbf{v}^k – populacja tymczasowa k -tego pokolenia po reprodukcji,

Y_j – j -ty osobnik populacji tymczasowej k -tego pokolenia po reprodukcji.

7.2.2. Operacje genetyczne - krzyżowanie

Proces krzyżowania oparty jest na schemacie wymiany z dwoma losowo wybranymi i jednym stałym punktem przecięcia chromosomu, z zagwarantowaniem przetrwania najlepszego osobnika. Do utworzenia jednego osobnika potomnego wymagane są trzy osobniki rodzicielskie. W wyniku krzyżowania powstaje nowa populacja tymczasowa $\boldsymbol{\zeta}$ w danym pokoleniu. Proces ten może zostać przedstawiony w sposób następujący:

$$\left. \begin{array}{l} Y_I \\ Y_{II} \\ Y_{III} \end{array} \right\} \rightarrow \mathbf{Z}_m = [y_1^I, \dots, y_{c1}^I, y_{c1+1}^{II}, \dots, y_{c2}^{II}, y_{c2+1}^{III}, \dots, y_{17}^{III}] \quad (7.13)$$

gdzie:

Y_I, Y_{II}, Y_{III} – losowo wybrane osobniki populacji tymczasowej \mathbf{v}^k ,

\mathbf{Z}_m – m -ty osobnik nowej populacji tymczasowej $\boldsymbol{\zeta}^k$, $m=2,3\dots K$,

$c1, c2$ – punkty rozcięcia chromosomu osobnika \mathbf{Z}_m losowane z rozkładem normalnym: $c1 = \xi_U(1,8)$, $c2 = \xi_U(9,16)$,

y_i^n – i -ty gen osobnika o chromosomie Y_n ,

lub:

$$\mathbf{v}^k = [Y_1, Y_2, \dots, Y_{15}] \xrightarrow{\text{Krzyżowanie}} \boldsymbol{\zeta}^k = [Z_1, Z_2, \dots, Z_{29}] \quad (7.14)$$

gdzie :

$\boldsymbol{\zeta}^k$ – nowa populacja tymczasowa pokolenia k po krzyżowaniu.

Krzyżowanie przeprowadzane jest $K-1$ razy, aż do zapełnienia populacji tymczasowej $\boldsymbol{\zeta}^k$. W trakcie krzyżowania zagwarantowane zostało, że chromosomy Y_I, Y_{II} nie mogą się

powtórzyć dla tego samego osobnika oraz że kombinacja Y_I, Y_{II}, Y_{III} nie może zostać powtórzona w tym samym pokoleniu.

7.2.3. Operacje genetyczne - mutacja

Mutacja polega na losowej zmianie wartości dwóch losowo wybranych genów każdego osobnika populacji tymczasowej ζ^k i może zostać opisana w następujący sposób:

$$t_i' = z + \xi_G(0,0.1) \cdot z_i \quad (7.15)$$

gdzie:

t_i' – wartość i -tego genu osobnika populacji potomnej,

z_i – wartość i -tego genu osobnika z populacji tymczasowej ζ^k ,

$\xi_G(g_1, g_2)$ – liczba losowana z rozkładem Gaussa, gdzie g_1 jest wartością oczekiwaną a g_2 odchyleniem standardowym.

lub:

$$\zeta^k = [Z_1, Z_2, \dots, Z_{29}] \xrightarrow{\text{Mutacja}} \tau^{k'} = [T'_1, T'_2, \dots, T'_{29}] \quad (7.16)$$

gdzie:

τ^k – populacja potomna pokolenia k ,

T'_j – j -ty osobnik populacji potomnej.

7.2.4. Sukcesja

Po przeprowadzeniu operacji genetycznych następuje sukcesja elitarna, polegająca na utworzeniu populacji bazowej kolejnego pokolenia na podstawie bieżącej populacji potomnej i poprzedniej populacji bazowej. Może ona zostać przedstawiona jako:

$$\tau^{k'} = [T'_1, T'_2, \dots, T'_{29}] \xrightarrow{\text{Sukcesja}} \tau^{k+1} = [T'_1, T'_2, \dots, T'_{29}, T^k] \quad (7.17)$$

gdzie:

τ^{k+1} – populacja bazowa pokolenia $k+1$,

T^k – osobnik dla którego funkcja przystosowania osiągnęła minimum w pokoleniu k .

Po utworzeniu populacji bazowej kolejnego pokolenia przeprowadzana jest kontrola położenia genów w przedziałach dopuszczalnych (które podaje tabela 5.1). Wartości genów,

które leżą poza obszarem dopuszczalnym, są zamieniane na odpowiednio dolną lub górną granicę przedziału dopuszczalnego.

7.3. Opis kryterium zatrzymania obliczeń

Kryterium zatrzymania obliczeń zostało sformułowane na podstawie szybkości poprawy rozwiązania, czyli szybkości zmiany wartości funkcji przystosowania. Dla każdego pokolenia z zadanego przedziału:

$$k \in \langle d + 1, d + 2, \dots, k_{\max} \rangle \quad (7.18)$$

gdzie:

k – numer pokolenia,

d – przedział uśredniania szybkości poprawy,

k_{\max} – ilość pokoleń wygenerowanych w trakcie ewolucji do momentu zatrzymania obliczeń.

obliczana jest średnia wartość szybkości poprawy wg wzoru:

$$v_k = \frac{F_{\min}^k - F_{\min}^{k-d}}{d} \quad (7.19)$$

gdzie:

F_{\min}^k – minimalna funkcji przystosowania w pokoleniu k ,

v_k – średnia szybkość poprawy dla pokolenia k obliczana na przedziale d .

W trakcie obliczeń na podstawie obserwacji przebiegu funkcji przystosowania, zostały określone następujące parametry kryterium zatrzymania obliczeń:

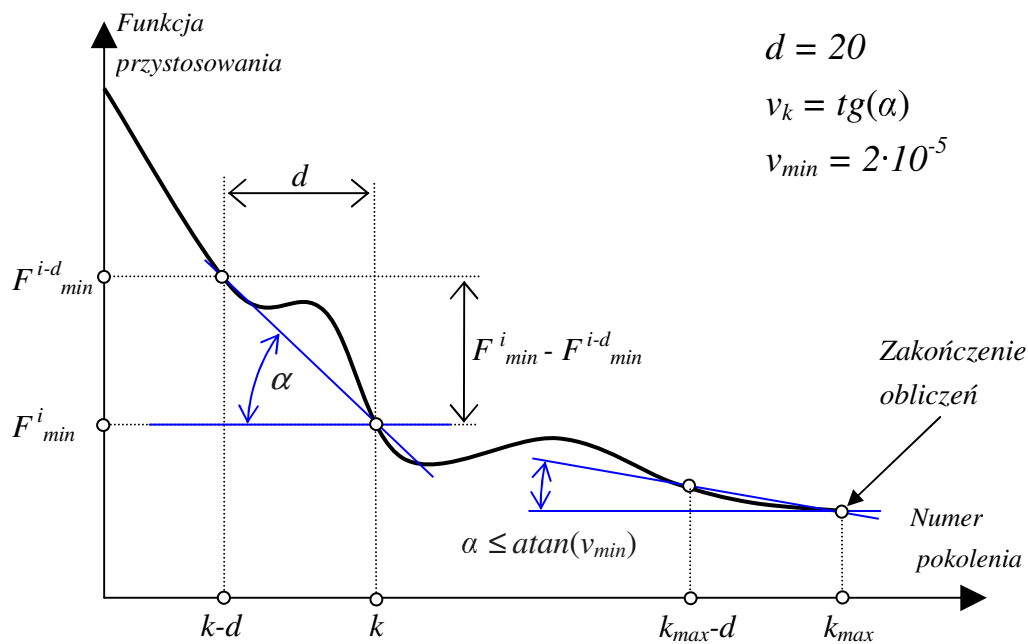
$$d = 20$$

$$v_{\min} = 2 \cdot 10^{-5} \quad (7.20)$$

gdzie:

v_{\min} – minimalna dopuszczalna szybkość poprawy rozwiązania,

Schemat sprawdzania warunku zatrzymania obliczeń przedstawia rys. 7.2.



Rys. 7.2 Schemat wystąpienia warunku zatrzymania obliczeń.

7.4. Omówienie wyników i wnioski

Wyniki obliczeń przedstawia tabela 7.1. Obliczenia zakończone zostały po 72 pokoleniach przy redukcji masy optymalizowanej części łopatki o 15.93% i redukcji maksymalnego naprężenia głównego o 0.02%. Średni czas oceny funkcji przystosowania dla jednego osobnika wyniósł 5 min 33 s na stacji roboczej wyposażonej w procesor Pentium IV o częstotliwości 3.2GHz oraz pamięć RAM 1GB.

Wszystkie wartości średnie i odchylenie standardowe pokazane na wykresach obliczone zostały tylko dla osobników o dopuszczalnej masie, czyli nieobjętych działaniem funkcji kary. Na wykresie pokazującym zmianę masy w trakcie optymalizacji została również pokazana ilość osobników które przekroczyły masę dopuszczalną, zostały wykluczone z obliczeń średniej i ukarane funkcją kary zgodnie z równaniem (7.6).

Rys. 7.3 przedstawia zmianę minimalnej i średniej wartości funkcji przystosowania dla kolejnych pokoleń. Dla losowo wygenerowanego pokolenia nr 1 minimalna i średnia wartość funkcji przystosowania wyniosła odpowiednio 0.93 i 1.06. Najbardziej gwałtowny spadek wartości funkcji przystosowania nastąpił między 2 i 4 pokoleniem (redukcja wartości minimalnej i średniej odpowiednio o 0.06 i 0.10). Dla kolejnych pokoleń przebieg funkcji przystosowania ulegał stopniowemu wypłaszczeniu, malała zarówno częstość zmian

najlepszego rozwiązania jak i ich zakres. Do dwóch najbardziej znaczących spadków doszło między pokoleniami 16-17 (spadek wartości minimalnej 0.0057) i 24-25 (spadek wartości minimalnej 0.0064). Pierwszy dłuższy okres stagnacji nastąpił między pokoleniami 34-46, kolejny między pokoleniami 51-72. Obliczenia zostały zatrzymane po 72 pokoleniach. Ostateczny wynik optymalizacji to redukcja funkcji przystosowania (maksymalnego naprężenia głównego) części optymalizowanej łopatki o 15.93%. Średnia wartość masy w pokoleniu została zmniejszona o 14.98%, co świadczy o otrzymaniu całej rodziny rozwiązań znacznie lepszych od modelu w konfiguracji „0”.

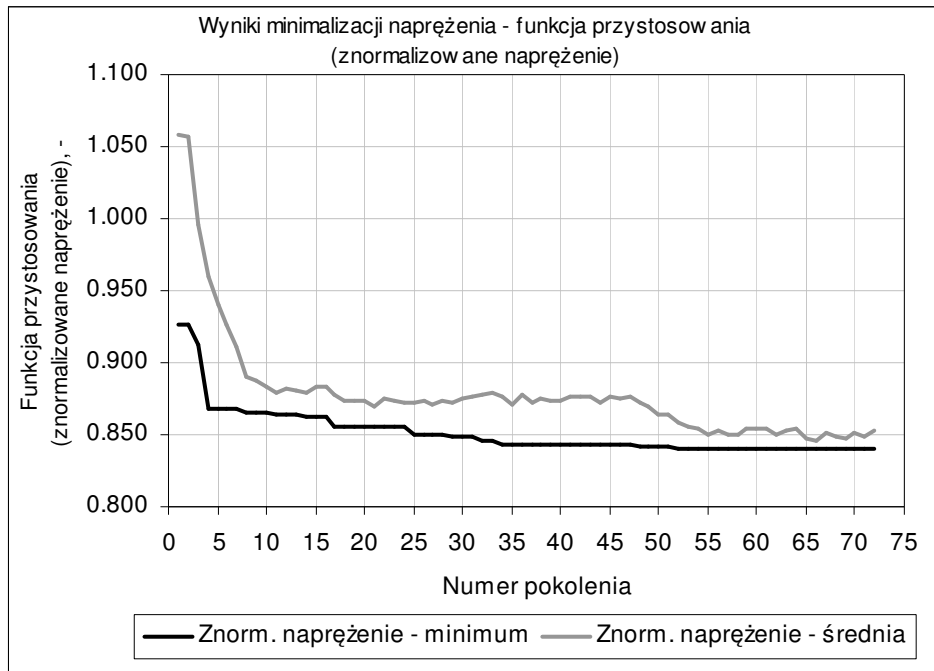
Tabela 7.1. Wyniki minimalizacji maksymalnego naprężenia głównego.

Konfiguracja	Pokolenie	Znorm. masa	Funkcja Przystosowania (Znormalizowane naprężenie)	Odch. stand. w pokoleniu k_{max}	Średnia szybkość poprawy w pokoleniu k_{max}	Śr. czas oceny wartości f.p.*	Łączny czas obliczeń*
	k_{max}	\bar{M} , kg/kg	F_1 - ($\bar{\sigma}$, MPa/MPa)	h_k	v_k -	w_k , mm:ss	W , ggg:mm
Konfiguracja „0”	-	1.000	1.000	-	-	-	-
Model optymalny	72	1.000	0.841	1.49E-01	0.00E+00	05:32	199:12

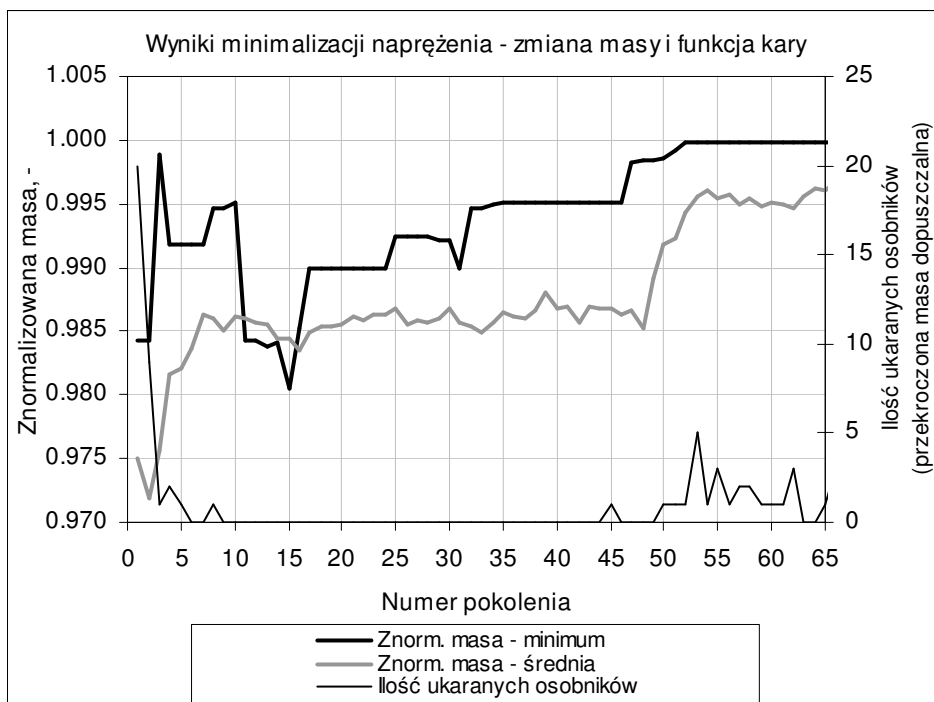
Efekt optymalizacji	- 0.021%	- 15.930%
---------------------	----------	-----------

* komputer PC PentiumIV 3.2GHz,1GB RAM, wynik łączny dla k_{max} pokoleń

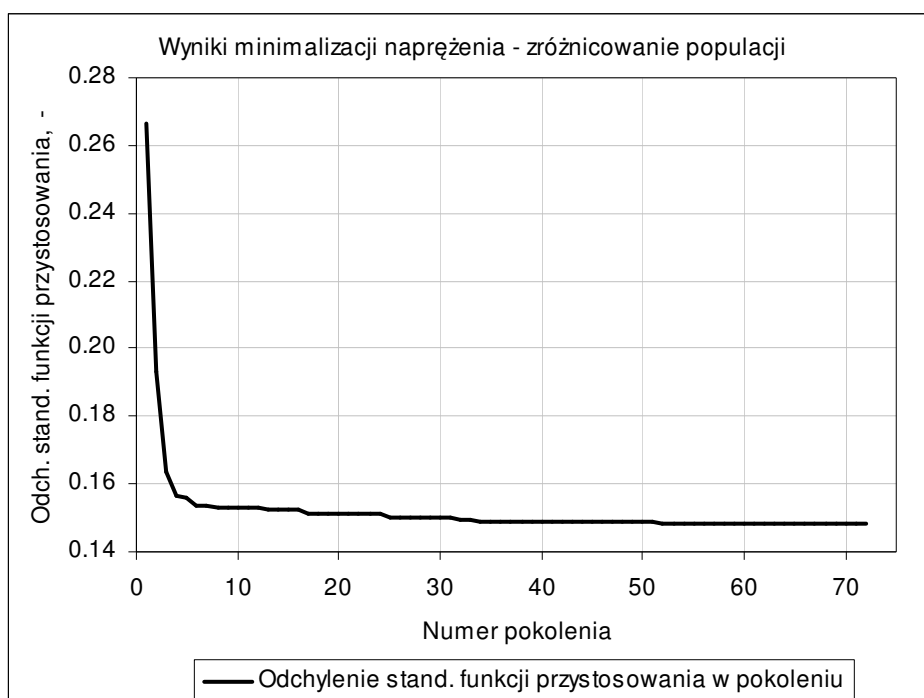
Uzyskany poziom redukcji maksymalnego naprężenia głównego przy zachowanej masie strefy optymalizowanej można uznać za wysoki z punktu widzenia kryteriów wytrzymałościowych stawianych łopacie (wytrzymałość doraźna, wytrzymałość na zmęczenie nisko/wysokocyklowe, odporność na pełzanie w podwyższonej temperaturze).



Rys. 7.3 Zmiana wartości funkcji przystosowania F (znormalizowanego naprężenia - $\bar{\sigma}$) – minimalizacja maksymalnego naprężenia głównego.



Rys. 7.4 Zmiana wartości znormalizowanej masy (\bar{M}) i ilość ukaranych osobników – minimalizacja maksymalnego naprężenia głównego.

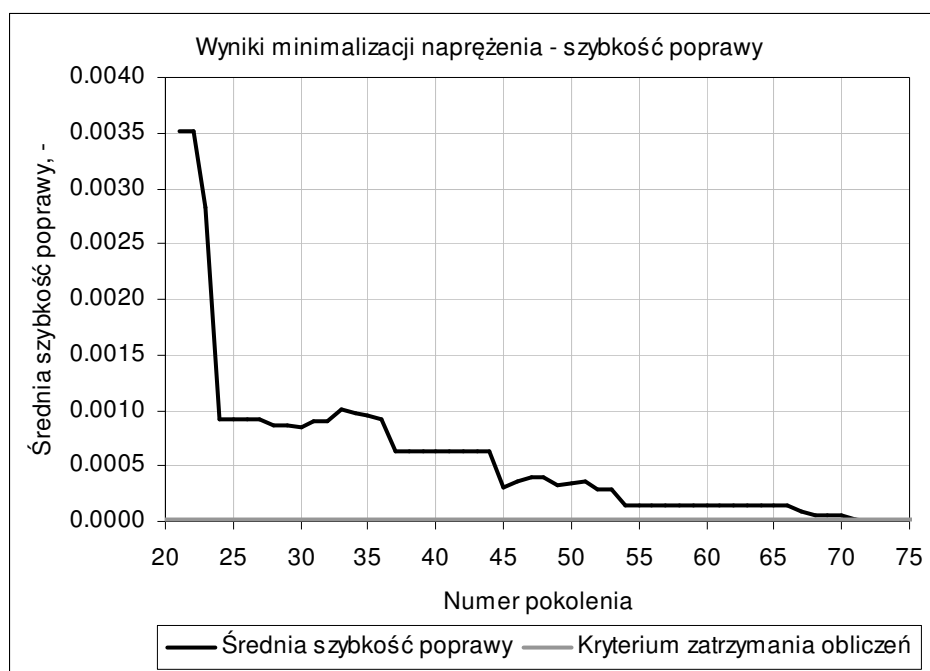


Rys. 7.5 Zmiana odchylenia standardowego (h_k) funkcji przystosowania – minimalizacja naprężenia .

Zmianę znormalizowanej masy (\bar{M}) w funkcji numeru pokolenia przedstawia rys. 7.4, pokazana została wartość odpowiadająca rozwiązaniu o najniższej funkcji przystosowania (linia czarna) i wartość średnia w pokoleniu (linia szara). Dodatkowo pokazana została ilość ukaranych osobników które przekroczyły masę dopuszczalną. Osobniki te zostały wykluczone z obliczeń wartości średnich dla każdego pokolenia. Zgodnie z oczekiwaniami wartość naprężenia zbliża się stopniowo do wartości dopuszczalnej. Ilość osobników po okresie przystosowania do warunków narzuconych przez funkcję kary maleje od wartości 20 w pierwszym pokoleniu do wartości 0 w pokoleniu szóstym. Późniejsze jej wahania przybierają zdecydowany charakter dopiero w pokoleniu 53 (wartość 5) i odpowiadają pierwszemu pokoleniu w którym najlepszy osobnik osiągnął masę dopuszczalną.

Zmianę zróżnicowania pokolenia w trakcie ewolucji pokazuje rys. 7.5. Zgodnie z oczekiwaniami odchylenie standardowe (h_k) funkcji przystosowania w pokoleniu ma stałą tendencję malejącą.

Zmianę średniej szybkości poprawy najlepszego rozwiązania (v_k) pokazuje rys. 7.6. Utrzymuje się stała tendencja malejąca z okresami stagnacji (pokolenia 27-34, 45-56).



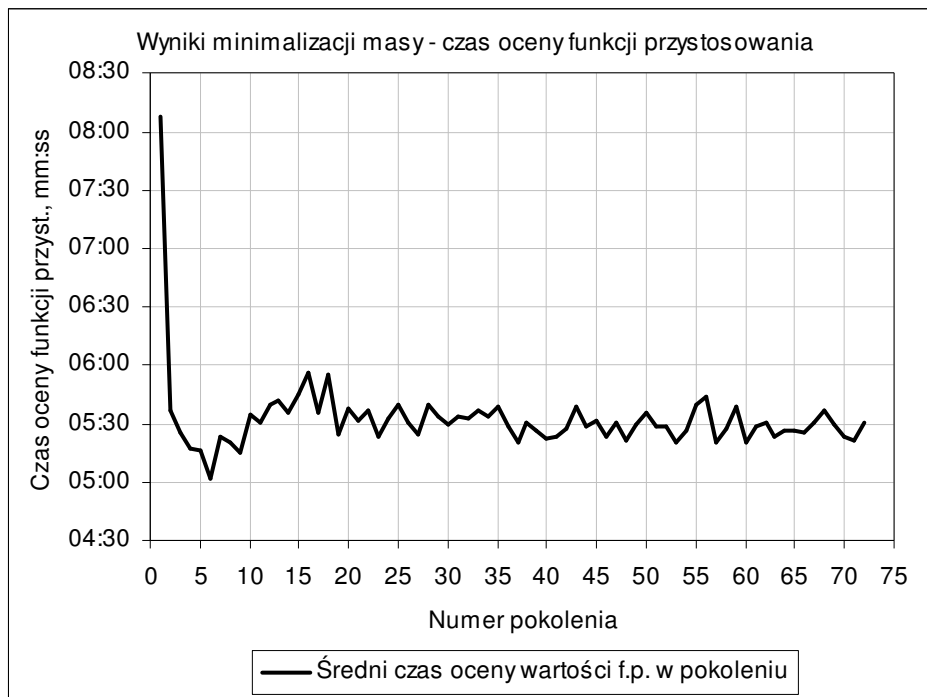
Rys. 7.6 Zmiana średniej szybkości poprawy rozwiązania (v_k) – minimalizacja naprężenia .

Rys. 7.7 pokazuje zmianę średniego czasu oceny funkcji przystosowania (w_k) w trakcie optymalizacji. Początkowa wartość powyżej 8 minut w ciągu jednego pokolenia obniża się do ok. 5 minut i z nieznacznymi wahaniami w granicach 40 sekund pozostaje stała do końca obliczeń. Początkowa wysoka wartość oraz późniejsze wahanie wywołane są procesem modyfikacji niedopuszczalnych chromosomów, pierwsze generowane losowo pokolenie modyfikowane jest w całości, kolejne tylko w części w miarę jak osobniki adaptują się do narzuconych ograniczeń.

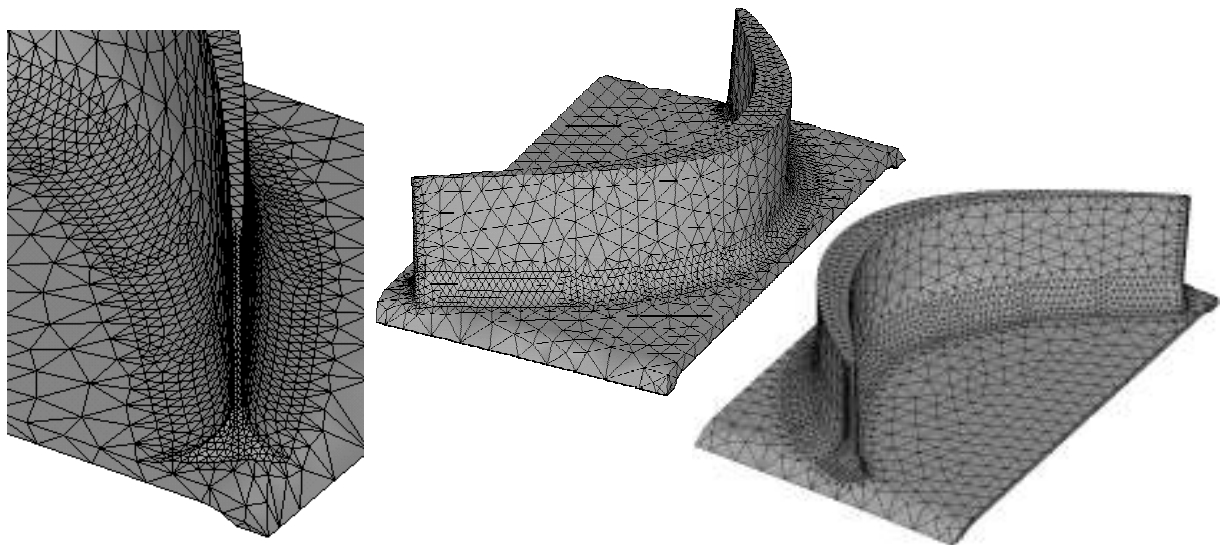
Siatka elementów skończonych na modelu zoptymalizowanym jest zagęszczona w rejonie spodziewanej koncentracji naprężeń (rys. 7.8). Porównanie kształtu modelu w konfiguracji „0” i zoptymalizowanego (rys. 7.9) pokazuje, że uległa dużej redukcji objętość strefy zmiany przekroju łopatki w pobliżu powierzchni natarcia i stopniowo powiększa się ona w kierunku krawędzi sływu. Sama powierzchnia sływu ukształtowana jest w sposób zapewniający dużą sztywność giętą pióra w płaszczyźnie stycznej do szkieletowej profilu.

Porównanie rozkładu maksymalnych naprężeń głównych na modelu zoptymalizowanym (rys. 7.10) pokazuje, że lokalne maksimum naprężeń uległo przesunięciu ku górze pióra, pojawiła się też strefa podwyższonych naprężeń w pobliżu krawędzi natarcia po stronie ssącej. W pobliżu krawędzi sływu naprężenia rozłożone są w sposób bardziej równomierny. Łopatka pracuje w większym stopniu na skręcanie a w mniejszym na zginanie.

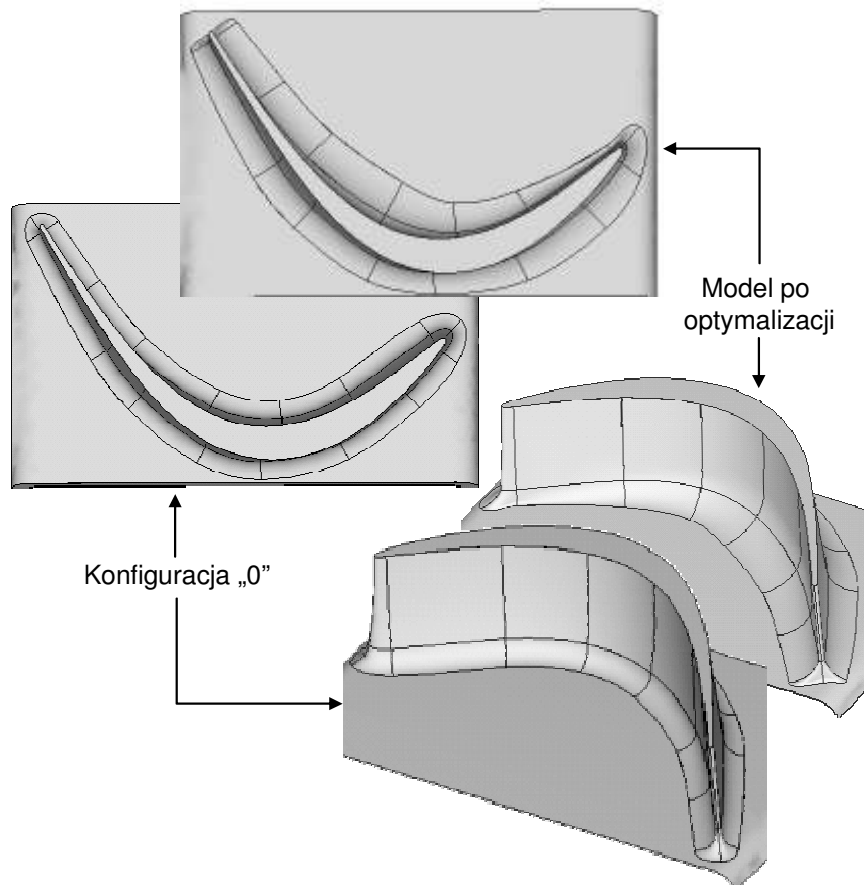
Porównanie przemieszczenia wypadkowego na modelu w konfiguracji „0” i modelu optymalnym pokazuje rys. 7.11. Poza nieznacznym zwiększeniem przemieszczeń na modelu optymalnym nie doszło do zmiany ich rozkładu.



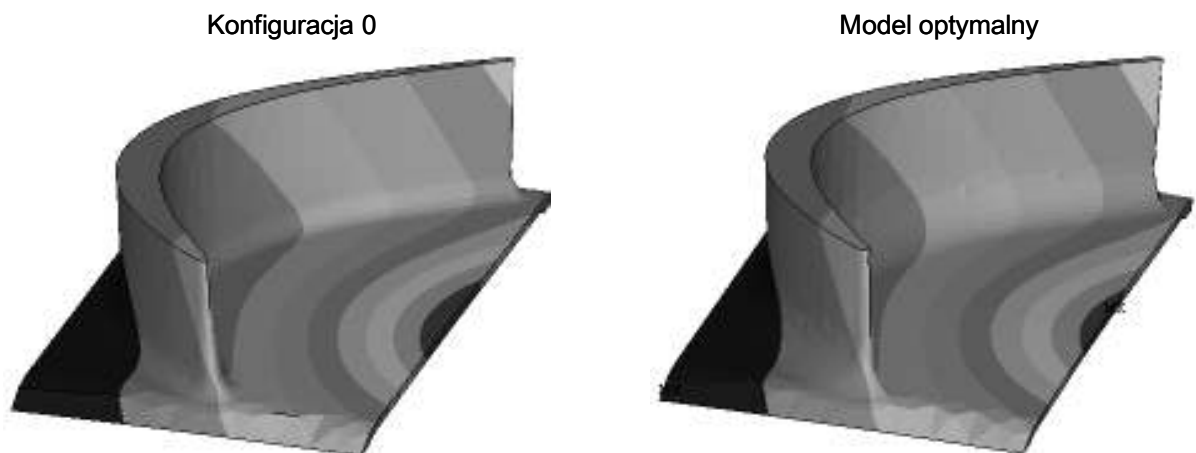
Rys. 7.7 Zmiana średniego czasu oceny wartości funkcji przystosowania (w_k) – minimalizacja naprężenia .



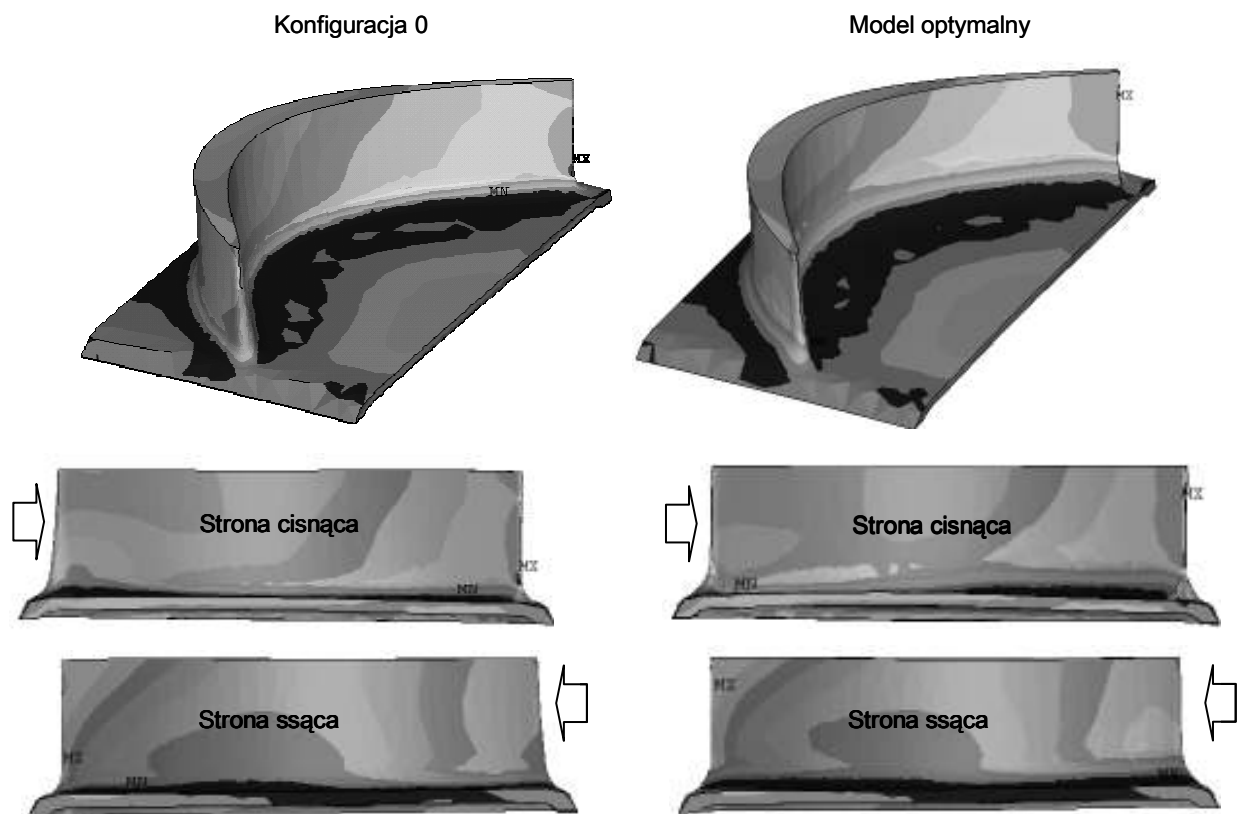
Rys. 7.8 Siatka elementów skończonych na modelu optymalnym – minimalizacja maksymalnego naprężenia głównego.



Rys. 7.9 Porównanie kształtu modelu w konfiguracji „0” i modelu zoptymalizowanego – minimalizacja maksymalnego naprężenia głównego.



Rys. 7.10 Porównanie przemieszczenia wypadkowego na modelu w konfiguracji „0” i modelu zoptymalizowanym – minimalizacja maksymalnego naprężenia (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).



Rys. 7.11 Porównanie rozkładu maksymalnego naprężenia głównego na modelu w konfiguracji „0” i modelu zoptymalizowanym – minimalizacja maksymalnego naprężenia głównego (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).

8. EWOLUCYJNA OPTIMALIZACJA JEDNOKRYTERIALNA KSZTAŁTU ŁOPATKI ZE WZGLĘDU NA MINIMUM MASY

8.1. Sformułowanie zadania optymalizacji ewolucyjnej

Celem obliczeń była minimalizacja masy optymalizowanej części łopatki przy zachowaniu maksymalnego naprężenia głównego poniżej wartości dopuszczalnej, czyli znalezienie chromosomu T^* spełniającego warunki:

$$\forall T \in \Omega \quad F(T^*) \leq F(T) \quad (8.1)$$

przy ograniczeniu

$$\bar{\sigma}(T^*) \leq 1 \quad (8.2)$$

gdzie:

Ω – przestrzeń rozwiązań dopuszczalnych,

T – chromosom (wektor parametrów geometrycznych) osobnika,

T^* – chromosom optymalny,

$F(T)$ – wartość funkcji przystosowania dla osobnika o chromosomie T ,

$\bar{\sigma}(T)$ – znormalizowane maksymalne naprężenie główne dla osobnika o chromosomie T :

$$\bar{\sigma}(T) = \frac{\sigma_1(T)}{\sigma_{10}} \quad (8.3)$$

$\sigma_1(T)$ – maksymalne naprężenie główne na optymalizowanej części pióra łopatki dla osobnika o chromosomie T ,

σ_{10} – maksymalne naprężenie główne na optymalizowanej części pióra łopatki modelu w konfiguracji „0”.

Sformułowane wyżej zadanie optymalizacji warunkowej sprowadzone zostało do zagadnienia optymalizacji bezwarunkowej poprzez zastosowanie metody funkcji kary.

Funkcja przystosowania może zostać przedstawiona za pomocą następującego wzoru:

$$F(\mathbf{T}) = \overline{M}(\mathbf{T}) \cdot P_{\sigma}(\mathbf{T}) \quad (8.4)$$

gdzie:

$\overline{M}(\mathbf{T})$ – znormalizowana masa dla osobnika o chromosomie \mathbf{T} obliczana jako stosunek masy modelu o chromosomie \mathbf{T} do masy modelu w konfiguracji „0”:

$$\overline{M}(\mathbf{T}) = \frac{M(\mathbf{T})}{M_0} \equiv \frac{\int_{V(\mathbf{T})} \rho dV}{\int_{V_0} \rho dV} \quad (8.5)$$

$M(\mathbf{T})$ – masa obliczona dla osobnika o chromosomie \mathbf{T} ,

$V(\mathbf{T})$ – objętość obliczona dla osobnika o chromosomie \mathbf{T} ,

ρ – gęstość,

M_0 – masa optymalizowanej części pióra łopatki modelu w konfiguracji „0”,

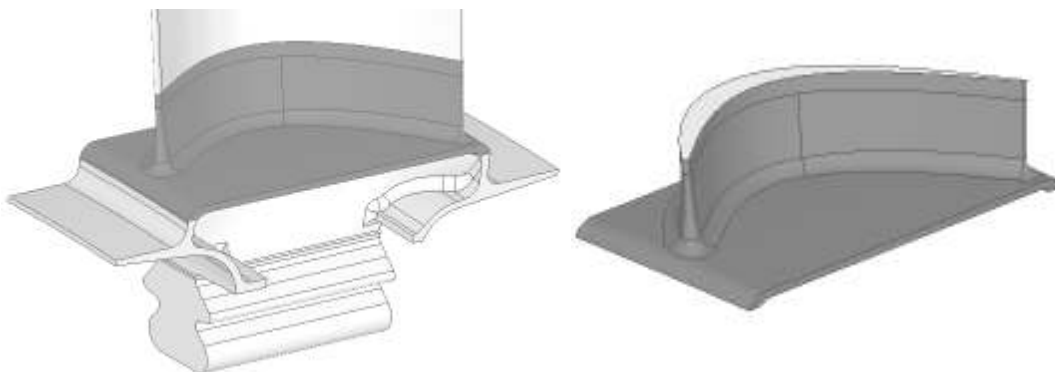
V_0 – objętość optymalizowanej części pióra łopatki modelu w konfiguracji „0”.

P_{σ} – naprężeniowa funkcja kary obliczana wg wzoru:

$$P_{\sigma}(\mathbf{T}) = \begin{cases} 1 & \text{gdy } \overline{\sigma}_1(\mathbf{T}) \leq 1 \\ 2 & \text{gdy } \overline{\sigma}_1(\mathbf{T}) > 1 \end{cases} \quad (8.6)$$

Masa obliczana jest dla modelu *MES* optymalizowanego fragmentu łopatki (rys. 8.1). Funkcja kary została wprowadzona w celu zapobiegania przyrostowi masy części optymalizowanej powyżej masy części w konfiguracji geometrycznej „0”.

W celu zapewnienia czytelności przedstawionych danych, osobniki o funkcji kary $P_{\sigma}(\mathbf{T}) = 2$ zostały wykluczone z obliczeń odchylenia standardowego i wartości średnich funkcji przystosowania. Obliczenia te są przeprowadzane tylko dla osobników o dopuszczalnym naprężeniu. Na wykresie pokazującym zmianę naprężenia umieszczona została krzywa pokazująca ilość ukaranych osobników.



Rys. 8.1 Strefa obliczania masy optymalizowanej części łopatki (część ciemnoszara).

8.2. Etapy optymalizacji ewolucyjnej.

Schemat procesu optymalizacji ewolucyjnej jest identyczny z przedstawionym w poprzednim rozdziale (rozdział 7.2).

8.3. Opis kryterium zatrzymania obliczeń

Kryterium zatrzymania obliczeń jest identyczne z przedstawionym w poprzednim rozdziale (rozdział 7.3).

8.4. Omówienie wyników i wnioski

Wyniki obliczeń przedstawia tabela 8.1. Obliczenia zakończone zostały po 106 pokoleniach przy redukcji masy optymalizowanej części łopatki o 6.59% i redukcji maksymalnego naprężenia głównego o 3.20%. Średni czas oceny funkcji przystosowania dla jednego osobnika wyniósł 4 min na stacji roboczej wyposażonej w procesor PentiumIV o częstotliwości 3.2GHz oraz pamięć RAM 1GB.

Tabela 8.1. Wyniki minimalizacji masy.

Konfiguracja	Pokolenie	Funkcja Przystosowania (Znormalizowana masa)	Znorm. naprężenie	Odch. stand. w pokoleniu k_{max}	Średnia szybkość poprawy w pokoleniu k_{max}	Śr. czas oceny wartości f.p.*	Łączny czas obliczeń*
	k_{max}	$F, - (\bar{M}, \text{kg/kg})$	$\bar{\sigma}, \text{MPa/MPa}$	h_k	$v_k, -$	$w_k, \text{mm:ss}$	$W, \text{ggg:mm}$
Konfiguracja „0”	-	1.000	1.000	-	-	-	
Model optymalny	116	0.934	0.995	1.65E-01	1.86E-05	04:00	232:00

Efekt optymalizacji	- 6.616%	- 0.505%
---------------------	----------	----------

* komputer PC PentiumIV 3.2GHz,1GB RAM, wynik łączny dla k_{max} pokoleń

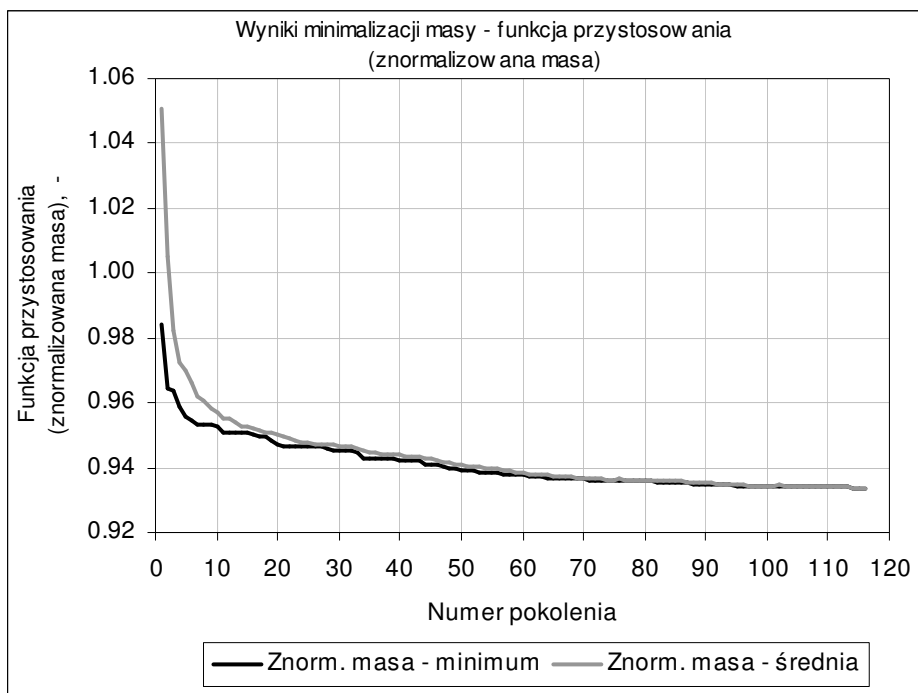
Wszystkie wartości średnie i odchylenie standardowe pokazane na wykresach obliczone zostały tylko dla osobników o dopuszczalnej masie, czyli nieobjętych funkcją kary.

Na wykresie pokazującym zmianę masy w trakcie optymalizacji została również pokazana ilość osobników które przekroczyły masę dopuszczalną, zostały wykluczone z obliczeń średniej i ukarane.

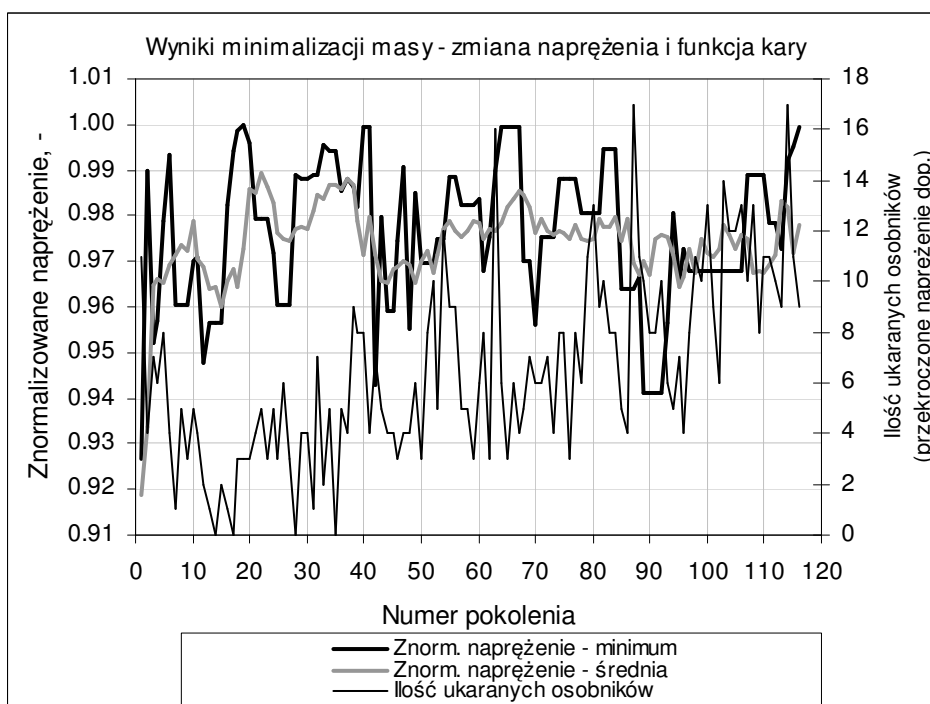
Zmianę minimalnej i średniej wartości funkcji przystosowania dla kolejnych pokoleń przedstawia rys. 8.2. Dla losowo wygenerowanego pokolenia nr 1 minimalna i średnia wartość funkcji przystosowania wyniosła odpowiednio 0.98 i 1.05. Najbardziej gwałtowny spadek wartości funkcji przystosowania nastąpił między 1 i 6 pokoleniem (redukcja wartości minimalnej i średniej odpowiednio o 0.03 i 0.08). Dla kolejnych pokoleń przebieg funkcji przystosowania ulegał stopniowemu wypłaszczeniu, malała zarówno częstość zmian najlepszego rozwiązania jak i ich zakres. Spadek wartości funkcji przystosowania miał charakter zbliżony do krzywej logarytmicznej. Obliczenia zostały zatrzymane po 116 pokoleniach. Ostateczny wynik optymalizacji to redukcja funkcji przystosowania (masy) części optymalizowanej łopatki o 6.6% i spadek maksymalnego naprężenia głównego o 3.2%. Średnia wartość funkcji przystosowania w pokoleniu wynosi 0.9342, natomiast dla najlepszego osobnika 0.9341, co świadczy o otrzymaniu całej rodziny rozwiązań znacznie lepszych od modelu w konfiguracji „0” i bardzo dobrej eksploracji dopuszczalnego zakresu parametrów. Można z dużym prawdopodobieństwem stwierdzić, że otrzymane rozwiązanie znajduje się blisko minimum globalnego.

Zmianę znormalizowanego naprężenia w funkcji numeru pokolenia przedstawia rys. 8.3, pokazana została wartość odpowiadająca rozwiązaniu o najniższej funkcji przystosowania (linia czarna) i wartość średnia w pokoleniu (linia szara). Dodatkowo pokazana została ilość ukaranych osobników które przekroczyły naprężenie dopuszczalne. Osobniki te zostały wykluczone z obliczeń wartości średnich dla każdego pokolenia. Krzywe mają charakter nieustalony, wykazują jedynie globalną tendencję wzrostową dla pokoleń 1-20 oraz globalną tendencję spadkową dla pokoleń 85-116. Generalnie zakres wahań naprężenia nie przekracza 5%.

Te wahania wartości można wyjaśnić bardzo dużą wrażliwością rozkładu naprężeń w analizowanej strefie na niewielkie zmiany geometrii, także te mieszczące się w granicach mutacji genów. Można natomiast wykluczyć wpływ zmiany siatki elementów skończonych na otrzymywane wartości naprężeń dla podobnych konfiguracji geometrycznych, ponieważ strefa odczytu naprężeń dyskretyzowana jest bardzo dokładnie. Porównanie wykresów zmiany masy (funkcji przystosowania) i naprężeń pokazuje że występują duże wahania naprężeń wywołane lokalną zmianą wartości promieni $R1$ i $R2$, które nie mają wpływu na masę z powodu zbyt małego przyrostu lub obniżenia objętości optymalizowanej strefy. Na przykładzie otrzymanych wyników widać celowość optymalizacji kształtu łopatki w tym obszarze, a zwłaszcza minimalizacji naprężeń które można obniżyć zmieniając konfigurację geometryczną modelu „0” w bardzo niewielkim zakresie. Charakterystyczny jest także



Rys. 8.2 Zmiana wartości funkcji przystosowania (znormalizowanej masy \bar{M}) – minimalizacja masy.

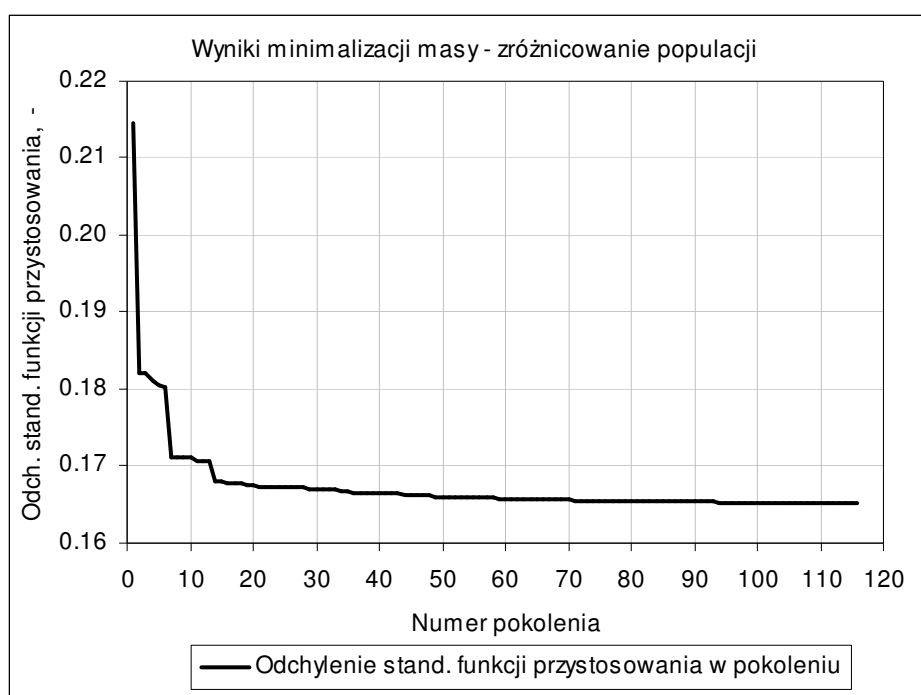


Rys. 8.3 Zmiana znormalizowanego naprężenia ($\bar{\sigma}$) i ilości ukaranych osobników – minimalizacja masy.

jednoczesny, kilkuprocentowy spadek masy i naprężenia, który pokazuje możliwości zdecydowanego poprawienia konstrukcji zarówno pod względem wytrzymałości jak i masy, które są czynnikami decydującymi o parametrach eksploatacyjnych całego rotora a w związku z tym także silnika.

Zmianę zróżnicowania pokolenia w trakcie ewolucji pokazuje rys. 8.4. Zgodnie z oczekiwaniami odchylenie standardowe (h_k) funkcji przystosowania w pokoleniu ma stałą tendencję malejącą.

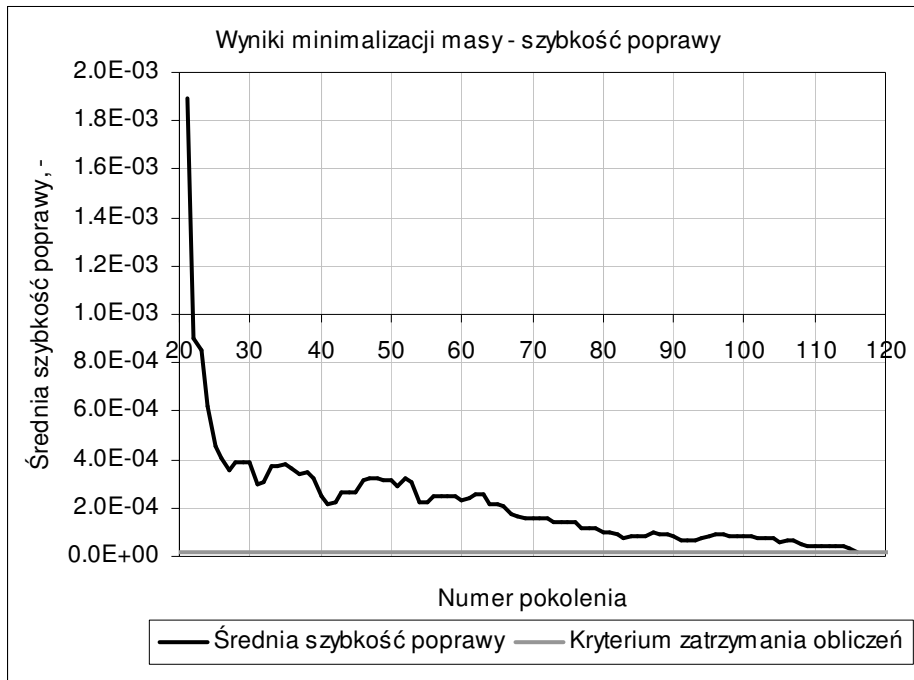
Zmianę średniej szybkości poprawy najlepszego rozwiązania (v_k) pokazuje rys. 8.5. Utrzymuje się stała tendencja malejąca z niewielkimi wahaniami.



Rys. 8.4 Zmiana odchylenia standardowego funkcji przystosowania (h_k) – minimalizacja masy.

Rys. 8.6 pokazuje zmianę średniego czasu oceny funkcji przystosowania (w_k) w trakcie optymalizacji. Początkowa wartość powyżej 8 minut w ciągu jednego pokolenia obniża się do ok. 4 minut i z nieznacznymi wahaniami w granicach 10 sekund pozostaje stała do końca obliczeń. Początkowa wysoka wartość oraz późniejsze wahania wywołane są procesem modyfikacji niedopuszczalnych chromosomów, pierwsze generowane losowo pokolenie modyfikowane jest w całości, kolejne tylko w części w miarę jak osobniki adaptują się do narzuconych ograniczeń. Różnicę w czasie oceny funkcji przystosowania w stosunku do obliczeń zw. z minimalizacją naprężeń należy wytłumaczyć przede wszystkim mniejszymi masami i co za tym idzie mniejszymi rozmiarami modeli i mniejszą ilością stopni swobody

co redukuje rozmiar zadania *MES* jakie należy rozwiązać w celu oceny wartości funkcji przystosowania.

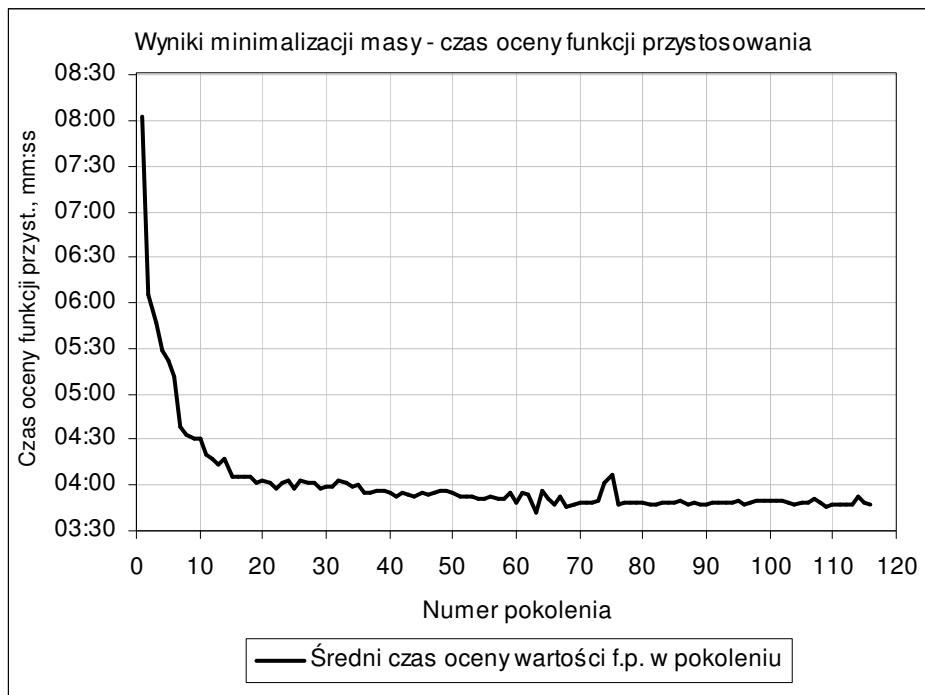


Rys. 8.5 Zmiana średniej szybkości poprawy rozwiązania (v_k) – minimalizacja masy.

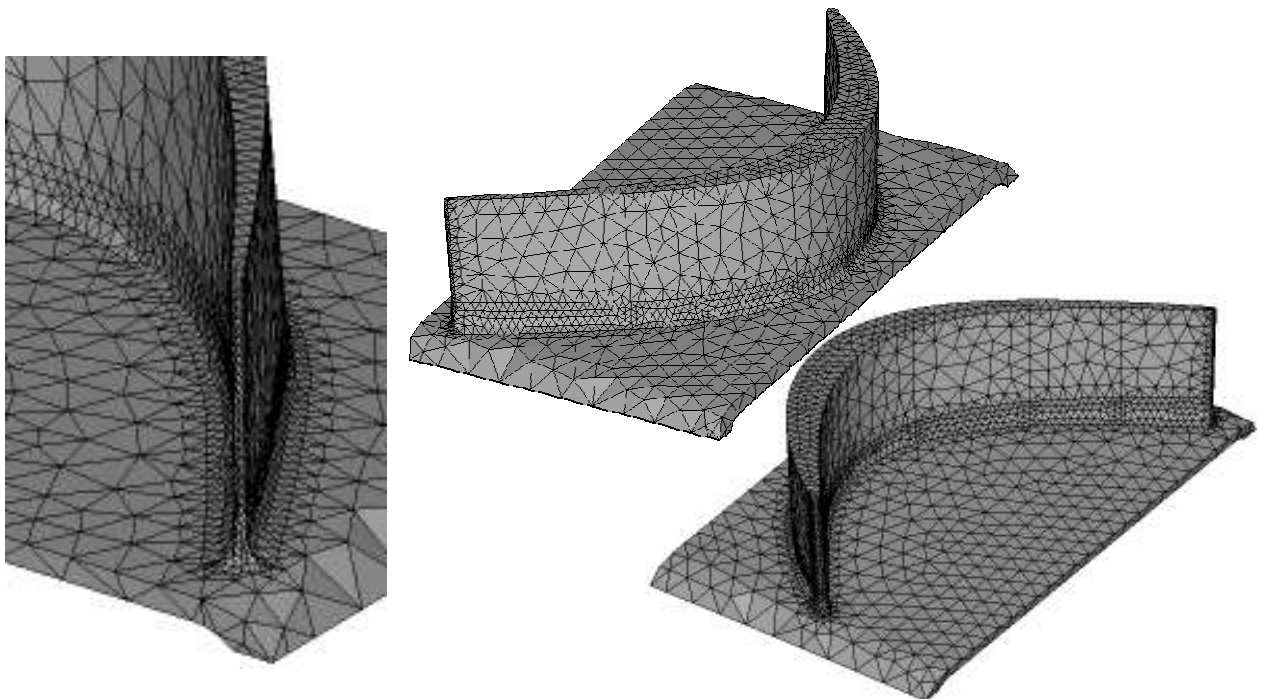
Szczegóły siatki elementów skończonych na modelu zoptymalizowanym (zagęszczenie w rejonie spodziewanej koncentracji naprężeń) pokazuje rys. 8.7. Porównanie kształtu modelu w konfiguracji „0” i zoptymalizowanego (rys. 8.8) pokazuje, że uległa znacznej redukcji objętość strefy zmiany przekroju łopatki w pobliżu powierzchni natarcia i nieco mniejszej redukcji strefa w pobliżu powierzchni spływu.

Porównanie rozkładu maksymalnych naprężeń głównych na modelu zoptymalizowanym (rys. 8.10) pokazuje, że strefa maksimum lokalnego uległa rozciągnięciu na znacznej długości obwodu, naprężenia rozłożone są w sposób bardziej równomierny niż w modelu wyjściowym, zaczęła pracować strefa na powierzchni cisnącej w pobliżu krawędzi spływu oraz na powierzchni ssącej w pobliżu krawędzi natarcia.

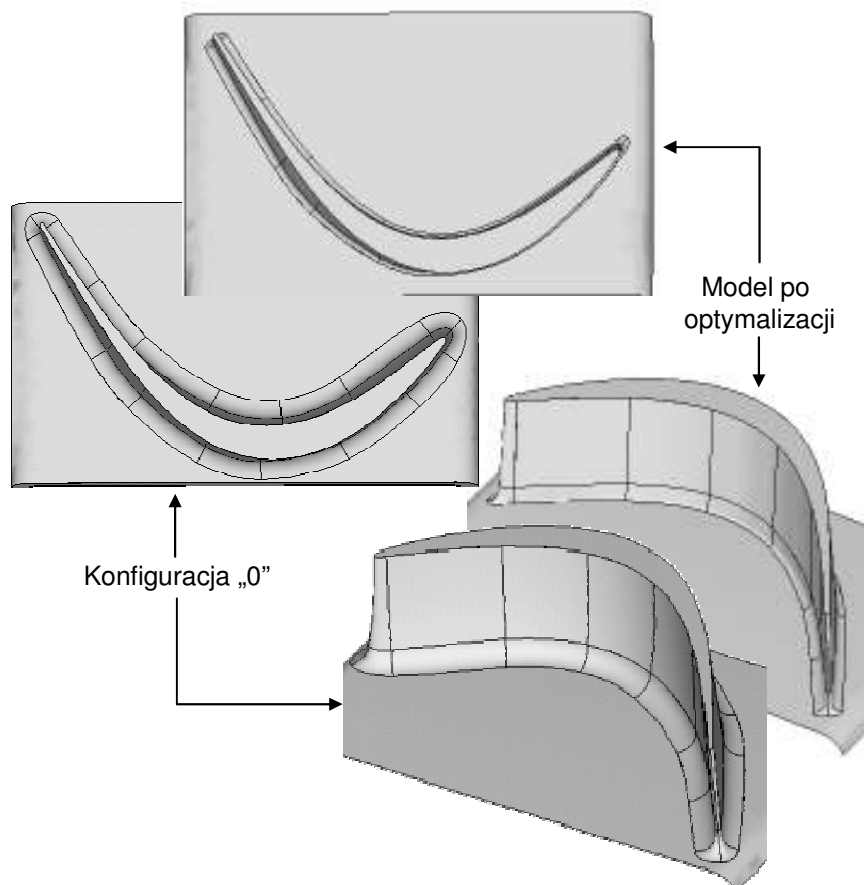
Rys. 8.9 pokazuje porównanie przemieszczenia wypadkowego na modelu w konfiguracji „0” i modelu optymalnym. Nie doszło do znaczącej zmiany ich rozkładu.



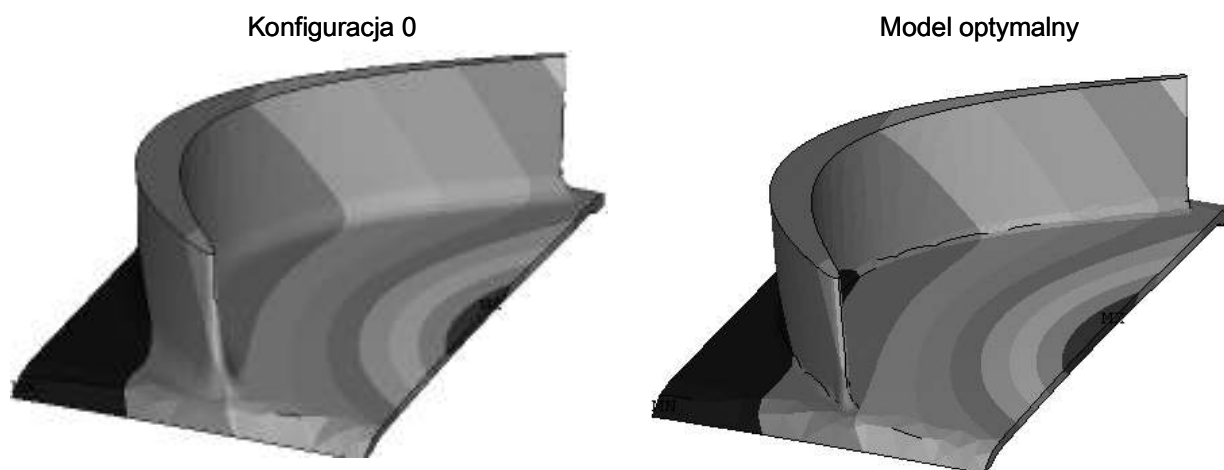
Rys. 8.6 Zmiana średniego czasu oceny wartości funkcji przystosowania (w_k) – minimalizacja naprężeń.



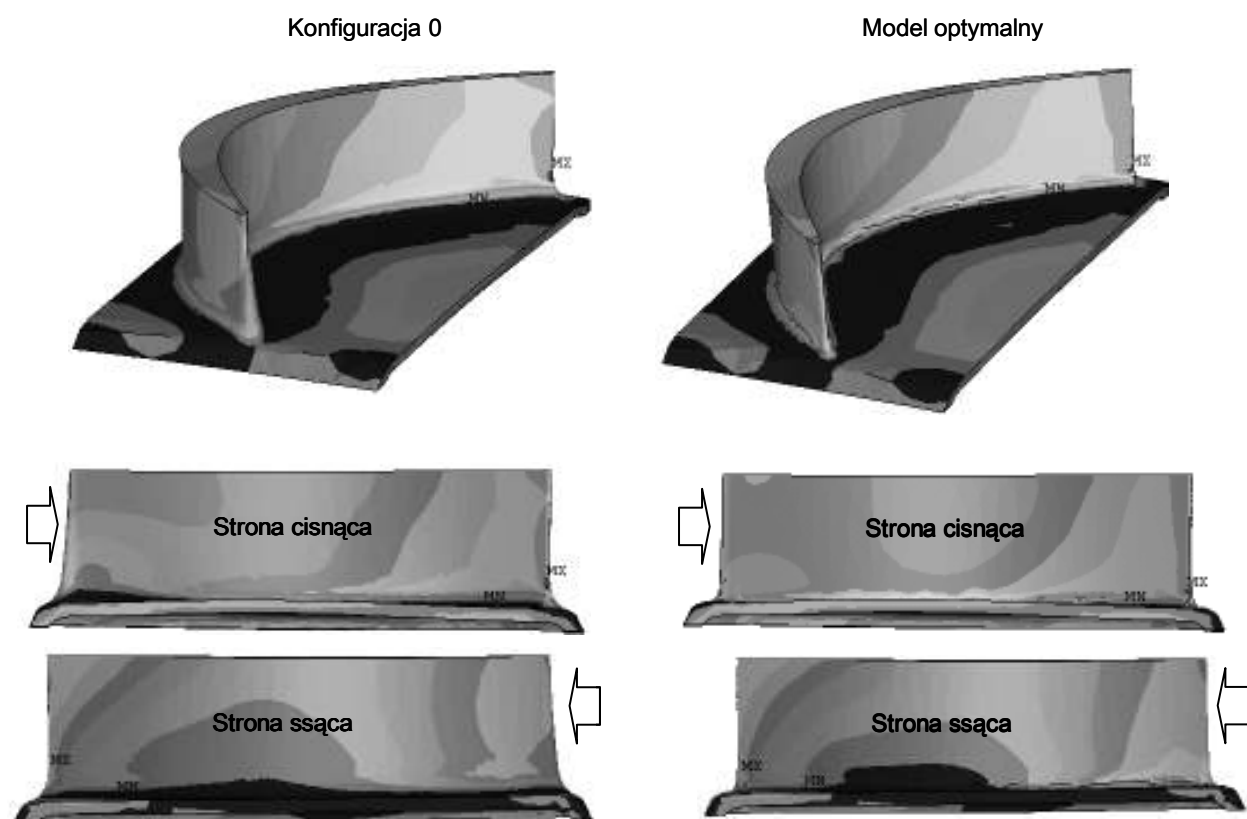
Rys. 8.7 Siatka elementów skończonych na modelu optymalnym – minimalizacja masy.



Rys. 8.8 Porównanie kształtu modelu w konfiguracji „0” i modelu zoptymalizowanego – minimalizacja masy.



Rys. 8.9 Porównanie przemieszczenia wypadkowego na modelu w konfiguracji „0” i modelu zoptymalizowanym – minimalizacja masy (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).



Rys. 8.10 Porównanie rozkładu maksymalnego naprężenia głównego na modelu w konfiguracji „0” i modelu zoptymalizowanym – minimalizacja masy (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).

9. EWOLUCYJNA OPTYMALIZACJA DWUKRYTERIALNA KSZTAŁTU ŁOPATKI ZE WZGLĘDU NA MINIMUM MASY I MAKSYMALNEGO NAPRĘŻENIA GŁÓWNEGO

9.1. Sformułowanie zadania optymalizacji ewolucyjnej

Rozważane w poprzednich rozdziałach funkcje przystosowania są antagonistyczne, tzn. minimalizacja masy prowadzi do maksymalizacji naprężeń a minimalizacja naprężeń do maksymalizacji masy. Celem optymalizacji dwukryterialnej jest równoczesna minimalizacja tych dwóch parametrów, czyli znalezienie chromosomu T^* spełniającego warunki:

1. Minimalizacja maksymalnego naprężenia głównego

$$\forall T \in \Omega \quad \bar{\sigma}(T^*) \leq \bar{\sigma}(T) \quad (9.1)$$

2. Minimalizacja masy

$$\forall T \in \Omega \quad \bar{M}(T^*) \leq \bar{M}(T) \quad (9.2)$$

przy jednoczesnych ograniczeniach:

$$\bar{\sigma}(T^*) \leq 1 \quad (9.3)$$

$$\bar{M}(T^*) \leq 1 \quad (9.4)$$

gdzie:

Ω – przestrzeń rozwiązań dopuszczalnych,

T – chromosom (wektor parametrów geometrycznych) osobnika,

T^* – chromosom optymalny,

$F(T)$ – wartość funkcji przystosowania dla osobnika o chromosomie T ,

$\bar{\sigma}(T)$ – znormalizowane maksymalne naprężenie główne dla osobnika o chromosomie T :

$$\bar{\sigma}(T) = \frac{\sigma_1(T)}{\sigma_{10}} \quad (9.5)$$

$\sigma_1(T)$ – maksymalne naprężenie główne na optymalizowanej części pióra łopatki dla osobnika o chromosomie T ,

σ_{10} – maksymalne naprężenie główne na optymalizowanej części pióra łopatki modelu w konfiguracji „0”.

$\bar{M}(T)$ – znormalizowana masa obliczona dla osobnika o chromosomie T obliczana jako stosunek masy modelu o chromosomie T do masy modelu w konfiguracji „0”:

$$\bar{M}(T) = \frac{M(T)}{M_0} \equiv \frac{\int_{V(T)} \rho dV}{\int_{V_0} \rho dV} \quad (9.6)$$

$M(T)$ – masa obliczona dla osobnika o chromosomie T ,

$V(T)$ – objętość obliczona dla osobnika o chromosomie T ,

ρ – gęstość,

M_0 – masa optymalizowanej części pióra łopatki modelu w konfiguracji „0”,

V_0 – objętość optymalizowanej części pióra łopatki modelu w konfiguracji „0”.

Zadanie sprowadzono do minimalizacji funkcji przystosowania w postaci:

$$F(T) = (a \cdot \bar{M}(T) + b \cdot \bar{\sigma}(T)) \cdot P_{m\sigma}(T) \quad (9.7)$$

gdzie:

a – współczynnik wagi dla masy,

b – współczynnik wagi dla naprężeń,

$P_{m\sigma}$ – naprężeniowo-masowa funkcja kary obliczana wg wzoru:

$$P_{m\sigma} = \begin{cases} 1 & \text{gdy } \bar{\sigma}_1 \leq 1 \text{ i } \bar{M} \leq 1 \\ 2 & \text{gdy } \bar{\sigma}_1 > 1 \text{ lub } \bar{M} > 1 \end{cases} \quad (9.8)$$

Masa obliczana jest dla modelu *MES* optymalizowanego fragmentu łopatki (rys. 8.1), naprężenie główne odczytywane jest z węzłów powierzchniowych pióra łopatki do wysokości strefy optymalizowanej, jak (rys. 7.1). Funkcja kary została wprowadzona w celu zapobieżenia przyrostowi masy lub naprężeń na części optymalizowanej powyżej masy lub naprężeń na modelu w konfiguracji geometrycznej „0”.

Optymalizacja dwukryterialna została przeprowadzona przy 6 zestawach współczynników a i b , obliczanych przy założeniu, że:

$$a + b = 1 \quad (9.9)$$

W szczególności, można przedstawić w formie pokazanego wyżej równania także funkcję przystosowania dla obliczeń zw. z minimalizacją masy (wtedy $a = 1.0$, $b = 0.0$) oraz minimalizacją naprężeń (wtedy $a = 0.0$, $b = 1.0$). Jeśli uznamy wyniki minimalizacji za minima globalne, uwzględnienie wyników optymalizacji jednokryterialnej pozwoli na szacunkową ocenę jak daleko od teoretycznego rozwiązania idealnego (minimalna masa przy minimalnym naprężeniu) znajduje się nasz wynik.

Wartość współczynnika a była stopniowo zwiększana od wartości 0.50 do wartości 0.80 w celu zwiększenia udziału znormalizowanej masy w funkcji przystosowania, ponieważ obliczenia przy wagach $a=0.50$ i $b=0.50$ wykazały że minimalizacji ulegają głównie naprężenia. Przeprowadzono także obliczenia przy współczynnikach $a=0.53$ i $b=0.47$, które odpowiadają uzyskanym w trakcie optymalizacji jednokryterialnym proporcjom masy i naprężenia teoretycznego modelu idealnego (minimum globalne masy przy minimum globalnym naprężenia).

Obliczenia z kilkoma zestawami współczynników a i b pozwoliły na wykreślenie frontu Pareto i uzyskanie zestawu osobników niezdominowanych. Wszystkie wykorzystane zestawy współczynników a i b przedstawia tabela 9.1.

9.2. Etapy optymalizacji ewolucyjnej.

Schemat procesu optymalizacji ewolucyjnej jest identyczny z przedstawionym w rozdziale 7.2.

9.3. Opis kryterium zatrzymania obliczeń

Kryterium zatrzymania obliczeń sformułowane zostało za pomocą dwóch warunków:

1. Warunek konieczny sformułowany w sposób zaprezentowany w rozdziale 7.3, z wyjątkiem wartości minimalnej szybkości poprawy rozwiązania, która dla optymalizacji dwukryterialnej ustalona została jako:

$$v_{min} = 1.2 \cdot 10^{-5} \quad (9.10)$$

gdzie:

v_{min} – minimalna dopuszczalna szybkość poprawy rozwiązania.

2. Warunek wystarczający - ponieważ w trakcie obliczeń okazało się, że niektóre zestawy współczynników wagowych a, b (na przykład $a=0.65, b=0.35$, lub $a=0.75, b=0.25$) prowadzą do przedwczesnej zbieżności objawiającej się małą ilością poprawionych rozwiązań k_p , wprowadzono dodatkowe kryterium. Dla zestawów współczynników a, b , które pozwalały na spełnienie warunku narzuconego na v_k przed osiągnięciem minimalnej ilości poprawionych rozwiązań k_{min} stosowano warunek w postaci:

$$k_{min}=22 \quad (9.11)$$

gdzie:

k_{min} – minimalna ilość poprawionych rozwiązań.

Zostało więc ustalone, że obliczenia mogą zostać zatrzymane po spełnieniu warunku 1, jeśli jednocześnie zachodzi warunek 2. Jeśli w momencie spełnienia warunku 1 nie był jednocześnie spełniony warunek 2, to warunek 2 stawał się jedynym koniecznym warunkiem zatrzymania obliczeń.

9.4. Omówienie wyników i wnioski

Wyniki obliczeń przedstawia tabela 9.1. Poza wynikami optymalizacji dwukryterialnej przedstawione w niej zostały także wyniki minimalizacji masy i minimalizacji naprężeń, będące przypadkiem szczególnym optymalizacji w którym jeden ze współczynników wagi przyjmuje wartość 0. W dwóch przypadkach brak jest danych na temat czasu oceny funkcji przystosowania, ponieważ nie zostały one zarejestrowane. Trzy ostatnie zestawy współczynników reprezentują obliczenia zatrzymane tylko na podstawie kryterium minimalnej ilości poprawionych rozwiązań.

Rys. 9.1 pokazuje zmianę funkcji przystosowania trakcie optymalizacji dwukryterialnej, wraz z wynikami minimalizacji masy i minimalizacji naprężenia. Rys. 9.2 pokazuje zmianę funkcji przystosowania trakcie optymalizacji dwukryterialnej tylko dla współczynników a i b o wartościach > 0 .

Rys. 9.3 pokazuje zmianę znormalizowanej masy w trakcie optymalizacji dwukryterialnej. Rys. 9.4 pokazuje zmianę znormalizowanego naprężenia w trakcie optymalizacji dwukryterialnej. Rys. 9.5 pokazuje rozkład Pareto dla optymalizacji dwukryterialnej z uwzględnieniem punktów otrzymanych w trakcie minimalizacji masy i naprężenia oraz z zaznaczeniem punktu idealnego.

Rys. 9.6 pokazuje rozkład Pareto dla optymalizacji dwukryterialnej z uwzględnieniem wyłącznie punktów otrzymanych dla niezerowych współczynników a , b .

Rys. 9.7 pokazuje front Pareto na tle wszystkich otrzymanych rozwiązań. Do dalszego szczegółowego omówienia wybrany został osobnik uzyskany w pokoleniu 173 dla zestawu współczynników $a=0.75$, $b=0.25$ wyróżniony na rysunku na zielono, szczegóły rozwiązania dla tego osobnika pokazuje tabela 9.2.

Tabela 9.1. Wyniki optymalizacji dwukryterialnej z uwzględnieniem wyników minimalizacji masy i naprężenia.

L.p.	Cel optymalizacji	Waga masy		Pokolenie k_{max}	Znorm. masa - najlepszy osobnik \bar{M} kg/kg	Znorm. naprężenie - najlepszy osobnik $\bar{\sigma}$ MPa/MPa	Wart. funkcji przystosowania w pokoleniu k_{max} $a * \bar{M} + b * \bar{\sigma}$	Odch. stand. w pokoleniu k_{max} h_k	Średnia szybkość poprawy w pokoleniu k_{max} v_{min}	Ilość poprawionych rozwiązań k_p	Śr. czas oceny wartości funkcji przystosowania* W_k mm:ss	Łączny czas obliczeń* W ggg:mm
		a	b									
1	Min. masy	1.00	0.00	116	0.934	0.995	0.934	0.165	1.86E-05	64	04:02	233:56
2	Min. naprężeń	0.00	1.00	72	1.000	0.841	0.841	0.149	0.00E+00	22	05:32	199:12
3	Min. sumy ważonej masy i naprężenia	0.50	0.50	73	0.992	0.850	0.921	0.163	9.83E-06	24	06:01	219:36
4		0.53	0.47	106	0.985	0.860	0.925	0.164	1.20E-05	26	b. d.	b. d.
5		0.55	0.45	100	0.982	0.846	0.920	0.163	1.13E-05	28	b. d.	b. d.
6		0.60	0.40	99	0.973	0.846	0.922	0.163	0.00E+00	22	04:13	208:43
7		0.65	0.35	121	0.973	0.855	0.932	0.165	1.58E-04	22**	04:32	274:16
8		0.75	0.25	182	0.959	0.886	0.940	0.166	5.20E-05	22**	04:51	441:21
9		0.80	0.20	89	0.952	0.920	0.946	0.167	9.32E-05	22**	03:55	174:17

* komputer PC PentiumIV 3.2GHz,1GB RAM, wynik łączny dla k_{max} pokoleń

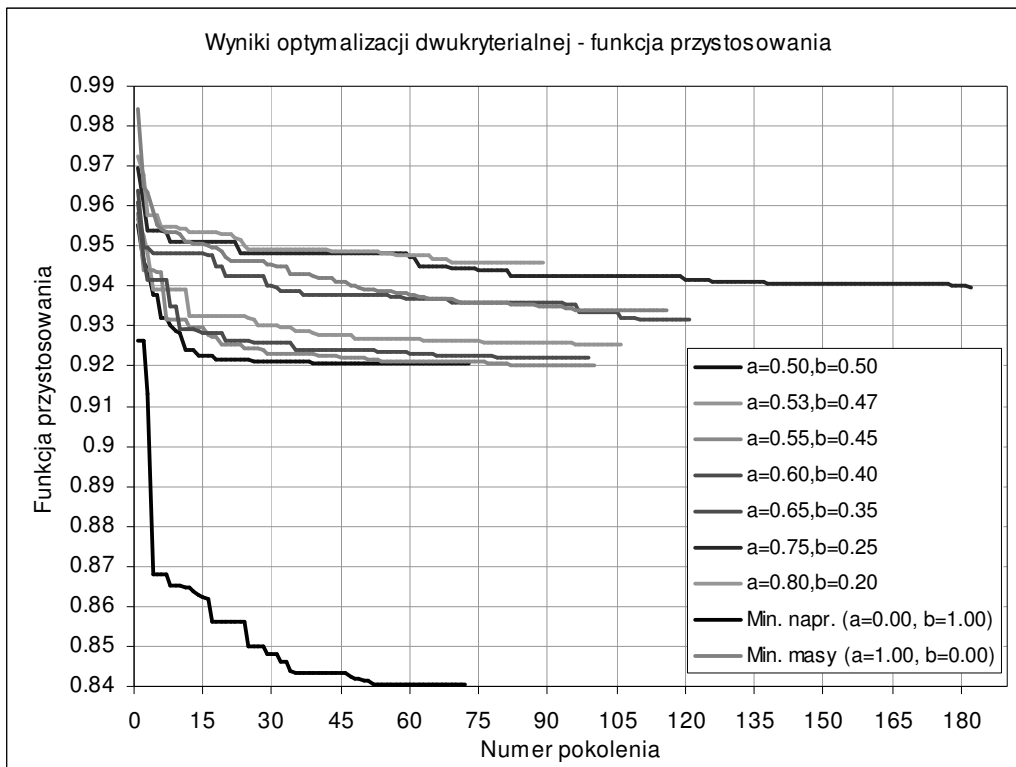
** kryterium minimalnej ilości poprawionych rozwiązań

Tabela 9.2. Wyniki optymalizacji dwukryterialnej dla wybranego osobnika ($a=0.75$, $b=0.25$, pokolenie 173).

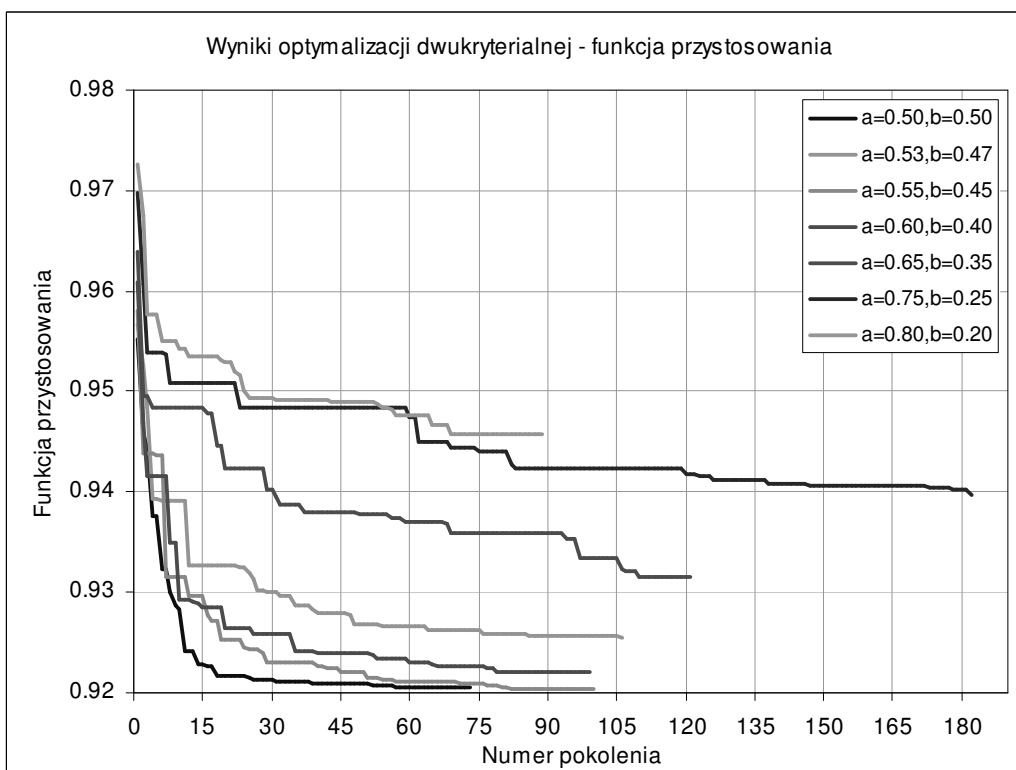
Konfiguracja	Pokolenie k_{max}	Znorm. masa \bar{M} , kg/kg	Znorm. naprężenie $\bar{\sigma}$, MPa/MPa	Wartość Funkcji Przystosowania F , -	Odch. stand. w pokoleniu k_{max} h_k , -	Średnia szybkość poprawy* w pokoleniu k_{max} v_k , -
Konfiguracja „0”	-	1.000	1.000	1.000	-	-
Model optymalny	173	0.958	0.890	0.940	0.166	1.17E-05

Efekt optymalizacji	- 4.247%	- 11.036%
---------------------	----------	-----------

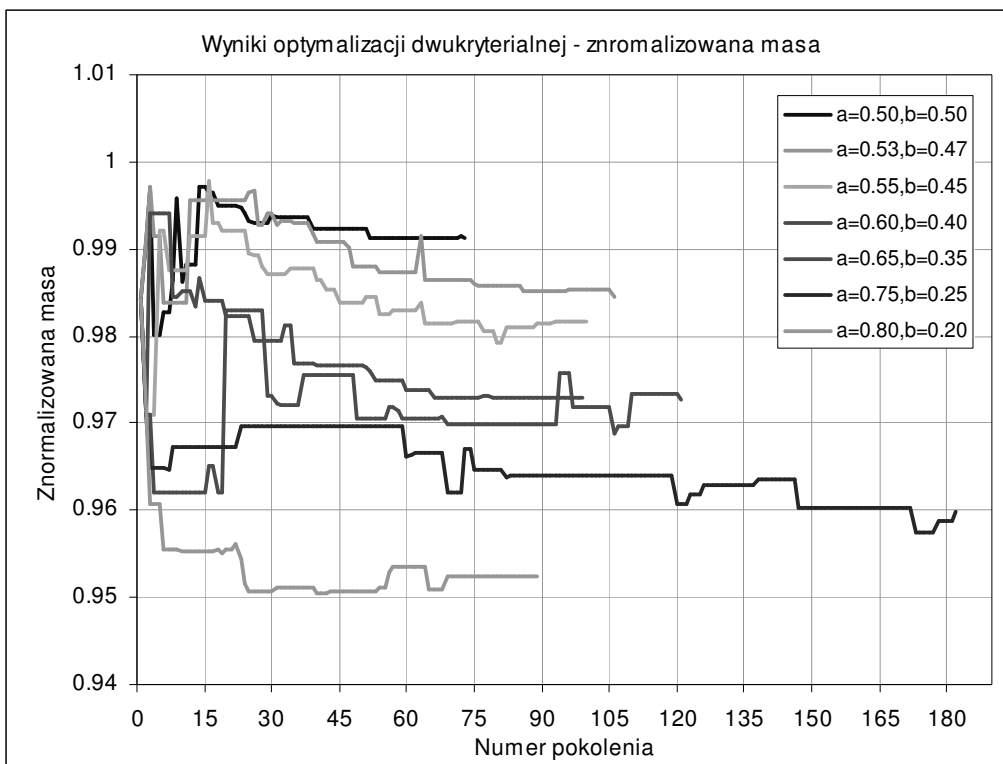
* komputer PC PentiumIV 3.2GHz,1GB RAM, wynik łączny dla k_{max} pokoleń



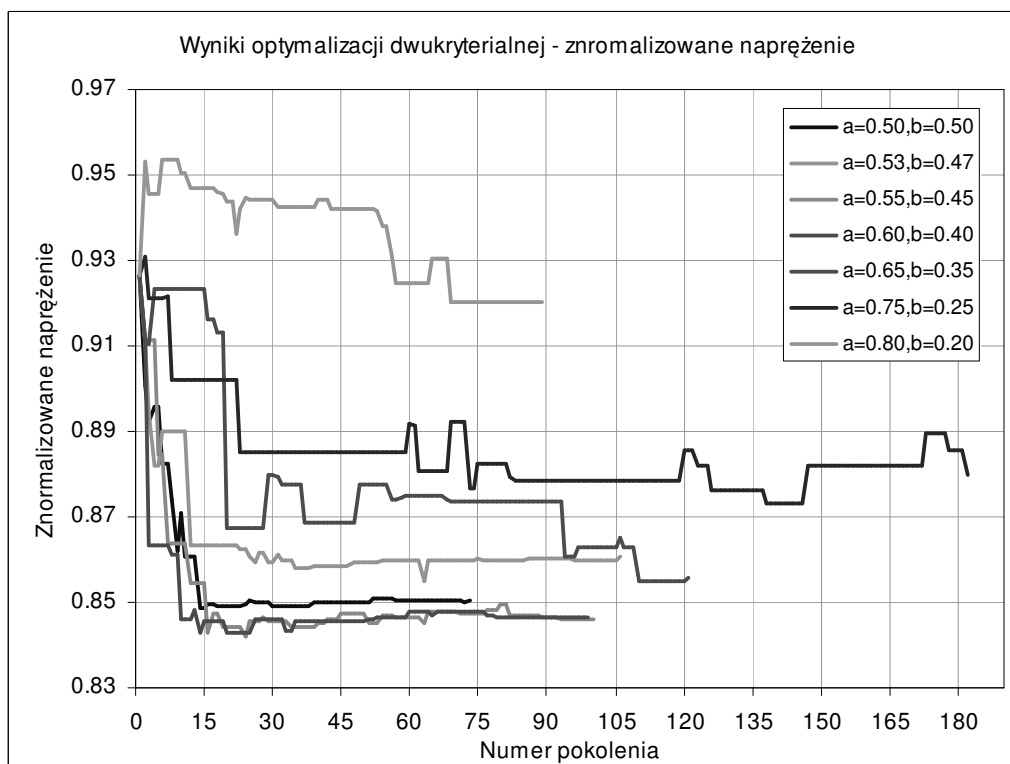
Rys. 9.1 Funkcja przystosowania – zmiana w trakcie optymalizacji dwukryterialnej z uwzględnieniem wyników minimalizacji masy i minimalizacji naprężenia.



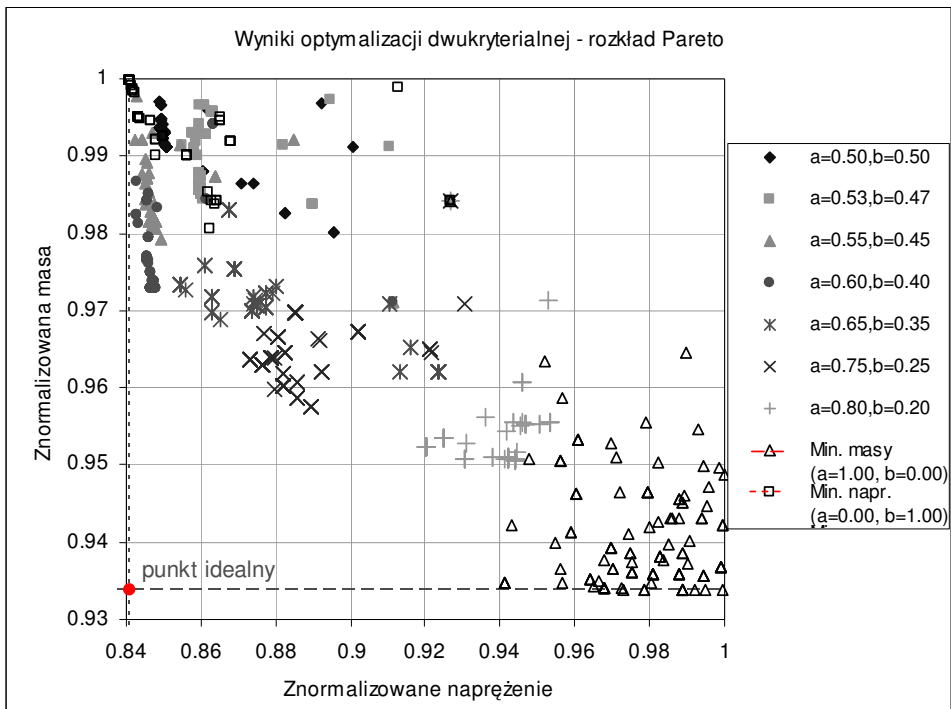
Rys. 9.2 Funkcja przystosowania – zmiana w trakcie optymalizacji dwukryterialnej.



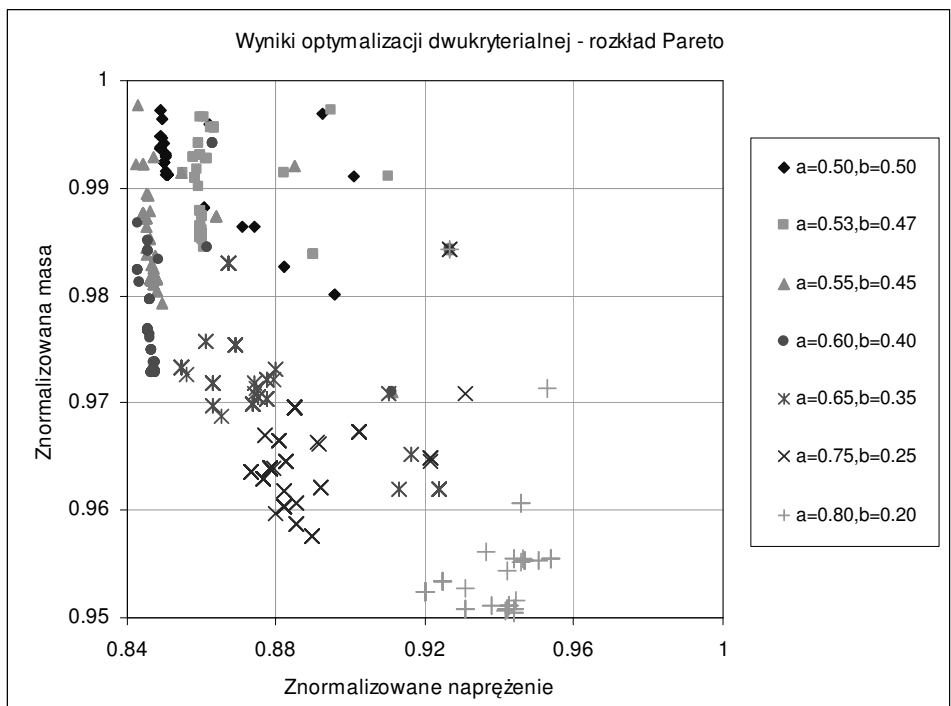
Rys. 9.3 Znnormalizowana masa- zmiana w trakcie optymalizacji dwukryterialnej.



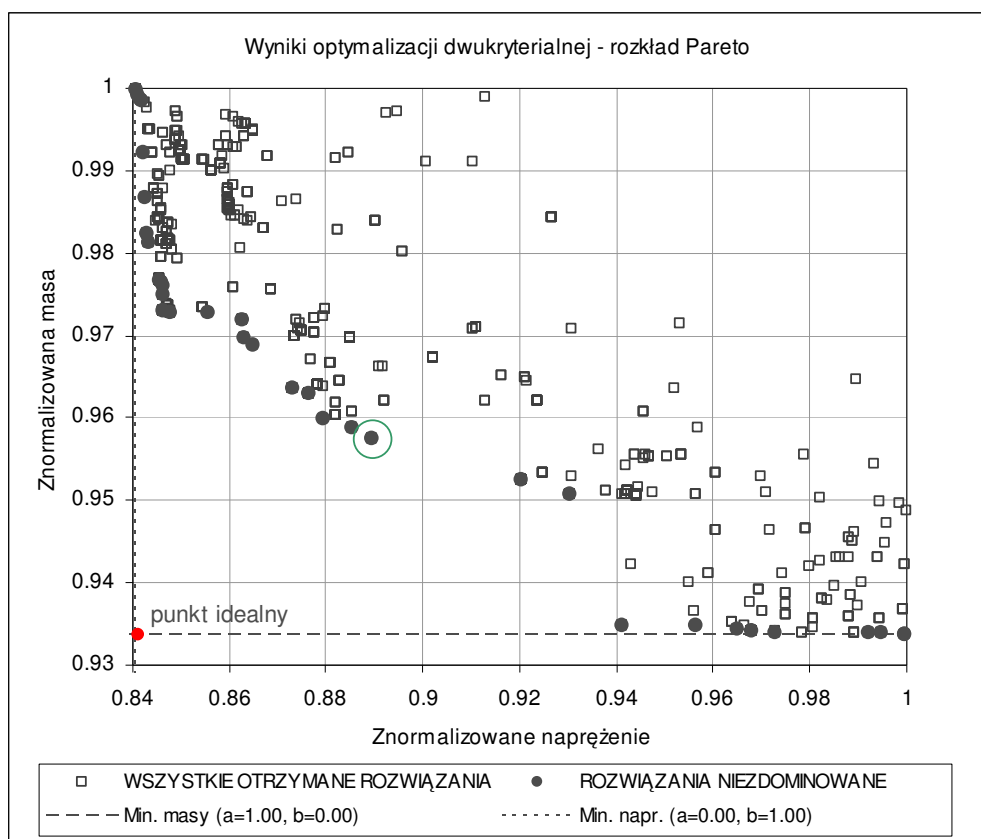
Rys. 9.4 Maksymalne znormalizowane naprężenie główne – zmiana w trakcie optymalizacji dwukryterialnej.



Rys. 9.5 Rozkład Pareto dla optymalizacji dwukryterialnej z uwzględnieniem punktów otrzymanych w trakcie minimalizacji masy i naprężenia oraz z zaznaczeniem punktu idealnego.



Rys. 9.6 Rozkład Pareto dla optymalizacji dwukryterialnej z uwzględnieniem wyłącznie punktów dla niezerowych współczynników a, b .

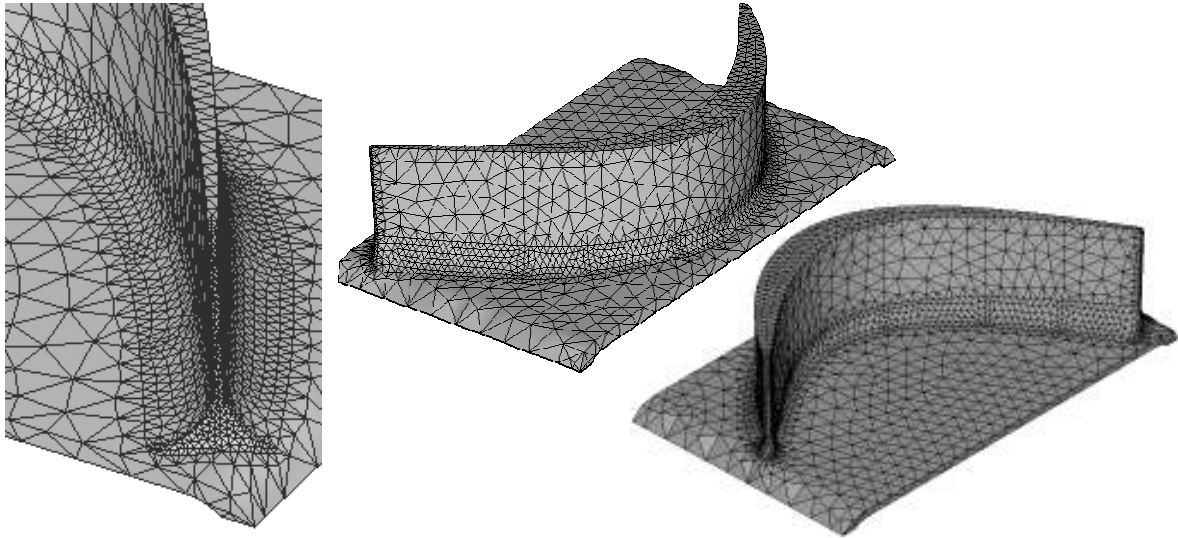


Rys. 9.7 Rozkład Pareto dla optymalizacji dwukryterialnej z wyodrębnieniem punktów niezdominowanych – front Pareto. Osobnik wybrany do szczegółowego omówienia wyróżniony na zielono.

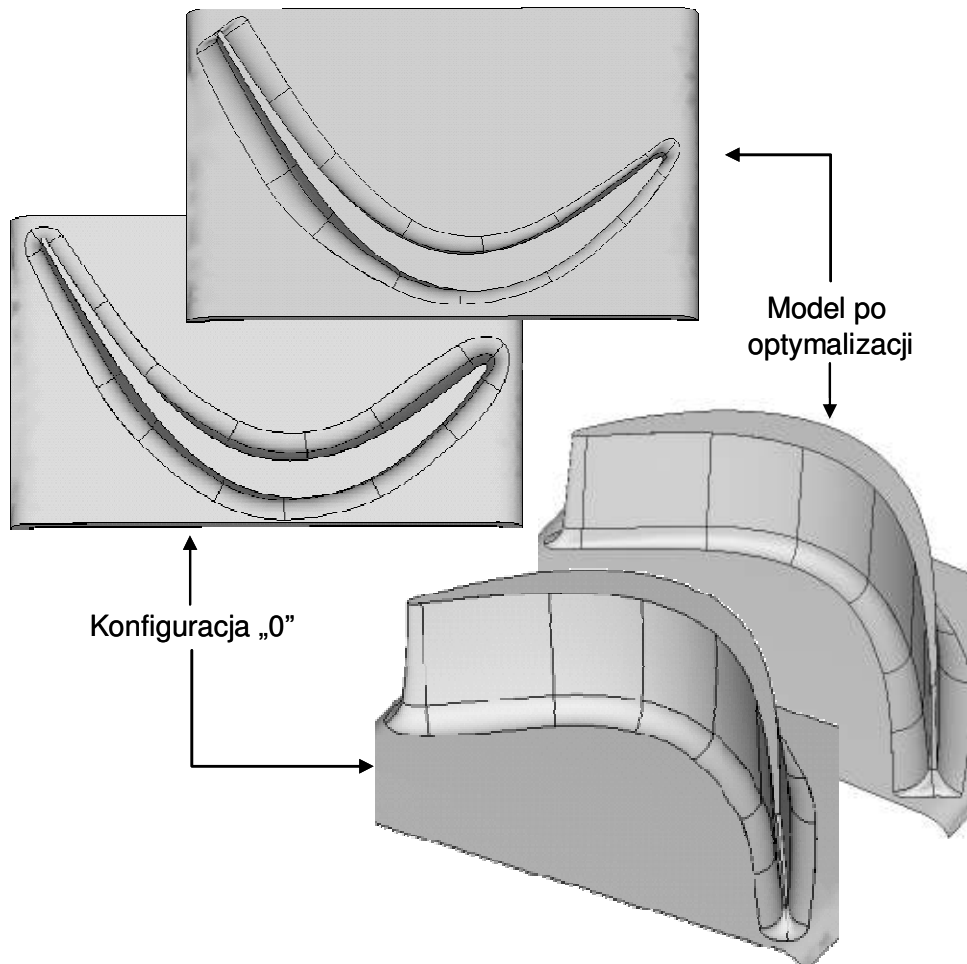
Szczegóły siatki elementów skończonych na modelu zoptymalizowanym (zagęszczenie w rejonie spodziewanej koncentracji naprężeń) pokazuje rys. 9.8. Porównanie kształtu modelu w konfiguracji „0” i zoptymalizowanego (rys. 9.9) pokazuje, że uległa znacznej redukcji objętość strefy zmiany przekroju łopatkę w pobliżu powierzchni natarcia i powiększeniu w kierunku powierzchni spływu.

Porównanie rozkładu maksymalnych naprężeń głównych na modelu zoptymalizowanym (rys. 8.10) pokazuje, że strefa maksimum lokalnego uległa rozciągnięciu na znacznej długości obwodu, naprężenia rozłożone są w sposób bardziej równomierny niż w modelu wyjściowym, zaczęła pracować strefa na powierzchni cisnącej w pobliżu krawędzi spływu oraz na powierzchni ssącej w pobliżu krawędzi natarcia. Model przypomina kształtem konfigurację uzyskaną w trakcie minimalizacji maksymalnego naprężenia głównego.

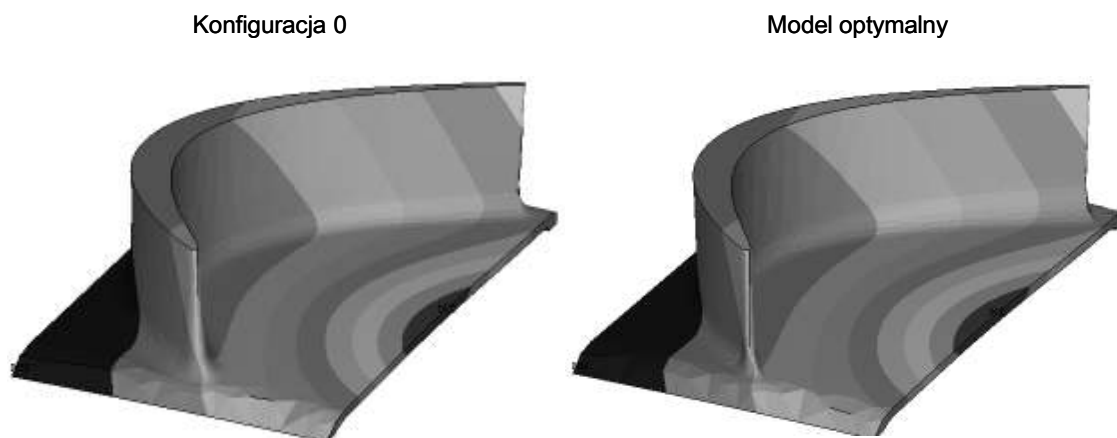
Rys. 8.9 pokazuje porównanie przemieszczenia wypadkowego na modelu w konfiguracji „0” i modelu optymalnym. Nie doszło do znaczącej zmiany.



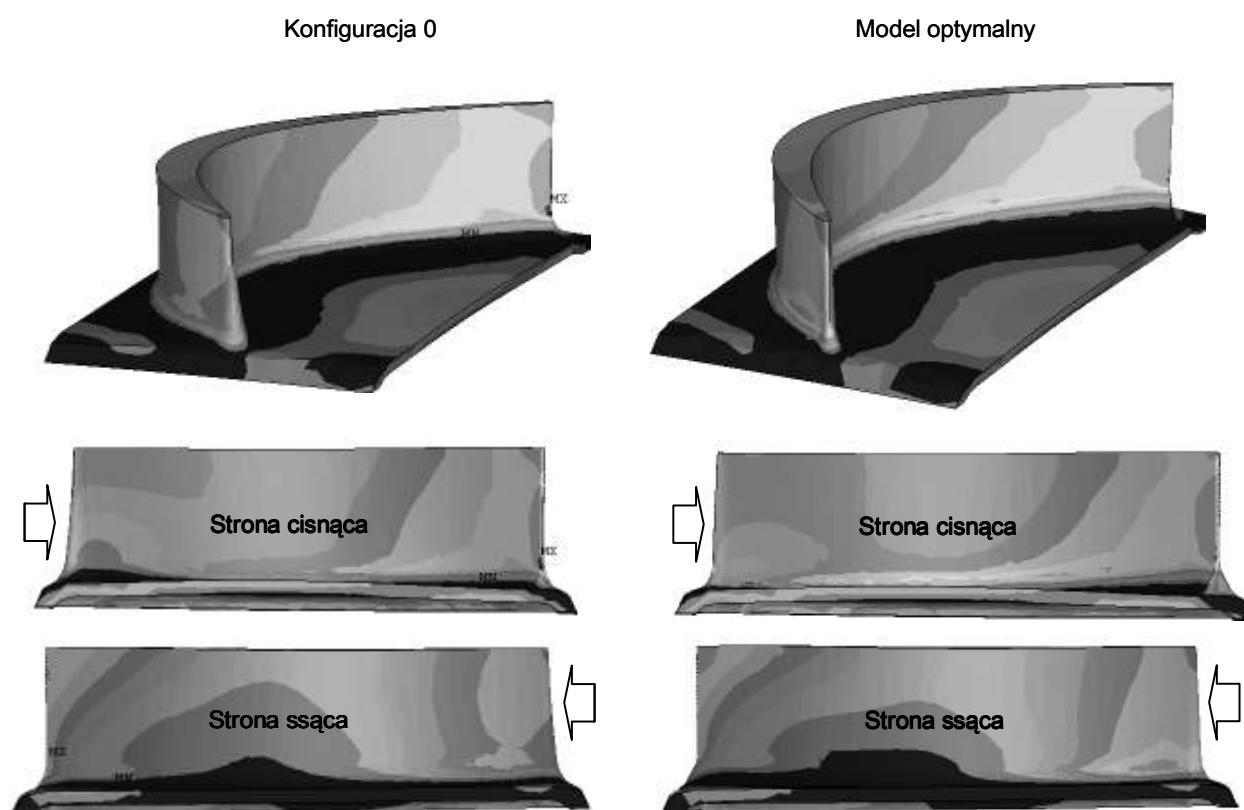
Rys. 9.8 Siatka elementów skończonych na modelu optymalnym – optymalizacja dwukryterialna.



Rys. 9.9 Porównanie kształtu modelu w konfiguracji „0” i modelu zoptymalizowanego – optymalizacja dwukryterialna.



Rys. 9.10 Porównanie przemieszczenia wypadkowego na modelu w konfiguracji „0” i modelu zoptymalizowanym – optymalizacja dwukryterialna (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).



Rys. 9.11 Porównanie rozkładu maksymalnego naprężenia głównego na modelu w konfiguracji „0” i modelu zoptymalizowanym – optymalizacja dwukryterialna (zachowane te same skale kolorystyczne, kierunek napływu gazu zaznaczony strzałkami).

10. WPŁYW ZMIANY KSZTAŁTU ŁOPATKI W STREFIE OPTYMALIZOWANEJ NA CZĘSTOŚĆ DRGAŃ WŁASNYCH ŁOPATKI

10.1. Cel analizy częstości drgań własnych łopatki

W celu sprawdzenia wpływu zmiany kształtu łopatki na jej sztywność oraz charakterystykę modalną, została przeprowadzona analiza częstości drgań własnych łopatki w konfiguracji „0” oraz w konfiguracjach zoptymalizowanych.

Charakterystyka modalna łopatki turbiny jest decydująca ze względu na konieczność spełnienia kryteriów wytrzymałości na zmęczenie wysokocyklowe, należy zagwarantować położenie wszystkich częstości drgań własnych łopatki poza częstotliwościami rezonansowymi turbiny w całym zakresie parametrów pracy silnika (wykorzystuje się w tym celu tzw. diagram *Campbella* [21] przedstawiający zależność pomiędzy częstotliwościami rezonansowymi turbiny i częstościami drgań własnych łopatki w funkcji prędkości obrotowej rotora). Dodatkowo sprawdza się wytrzymałość na zmęczenie wysokocyklowe łopatki dla podstawowych postaci drgań własnych z wykorzystaniem tzw. diagramu *Goodmana* przedstawiającego zależność między naprężeniami statycznymi i dynamicznymi w najbardziej wyteżonych punktach pióra łopatki na tle krzywej *Wöhlera*.

Ponieważ ani analiza częstości drgań własnych ani wytrzymałość zmęczeniowa łopatki nie są przedmiotami tej pracy i stanowią jedynie dodatek do przedstawionych wyników obliczeń, zagadnienie to zostanie omówione skrótowo, bardziej szczegółowe informacje dostępne są w wielu publikacjach, np. [21], [11], [36], [41], [43].

10.2. Model matematyczny do analizy częstości drgań własnych łopatki

W celu uproszczenia opisu zagadnienia zajmiemy się od razu przedstawieniem modelu *MES* układu drgającego wychodząc od razu od elementowego równania równowagi statycznej (6.39) wyprowadzonego w rozdziale 6.2, uzupełniając je o człon bezwładnościowy

i człon wynikający z sił tłumienia, oraz uzależniając elementowy wektor przemieszczeń węzłowych od czasu [36], [43]. W tym celu konieczne będzie przedstawienie sposobu wyznaczania macierzy bezwładności elementu. Można w tym celu wykorzystać elementowe funkcje kształtu (tzw. macierz konsekwentna) lub przyjąć masy skupione w węzłach elementu (tzw. macierz zgranulowana). Zgodnie z równaniem (6.23) pole przemieszczeń elementowych można przedstawić jako iloczyn macierzy funkcji kształtu i wektora przemieszczeń elementowych. Jeśli przedstawimy pole przemieszczeń jako funkcję czasu, to różniczkując równanie (6.23) po czasie otrzymamy:

$$\dot{\mathbf{u}} = \mathbf{N}\dot{\mathbf{u}}^e \quad (10.1)$$

$$\ddot{\mathbf{u}} = \mathbf{N}\ddot{\mathbf{u}}^e \quad (10.2)$$

gdzie:

$\dot{\mathbf{u}}$ – wektorowe pole prędkości jako funkcja czasu: (zgodnie z równaniem (6.17):

$$\dot{\mathbf{u}} = \{\dot{u}(x, y, z, t) \quad \dot{v}(x, y, z, t) \quad \dot{w}(x, y, z, t)\}^T,$$

$\ddot{\mathbf{u}}$ – wektorowe pole przyspieszeń (jak wyżej),

$\dot{\mathbf{u}}^e$ – elementowy wektor prędkości węzłowych,

$\ddot{\mathbf{u}}^e$ – elementowy wektor przyspieszeń węzłowych.

W celu zdefiniowania macierzy bezwładności elementu wykorzystujemy *zasadę d’Alamberta* – zachodzi związek:

$$\mathbf{f}_b = -\rho\ddot{\mathbf{u}} \quad (10.3)$$

gdzie:

$\rho = \rho(x, y, z)$ – gęstość,

\mathbf{f}_b – pole sił bezwładności ($\mathbf{f}_b = \{f_b^x(x, y, z) \quad f_b^y(x, y, z) \quad f_b^z(x, y, z)\}^T$).

Dla modelu dyskretnego bezwładnościowe siły węzłowe otrzymamy wykorzystując elementowe funkcje kształtu ze wzoru:

$$\mathbf{f}_b^e = \int_V \mathbf{N}^T \mathbf{f}_b dV = -\int_V \mathbf{N}^T \rho \mathbf{N} dV \ddot{\mathbf{u}}^e = -\mathbf{M}^e \ddot{\mathbf{u}}^e \quad (10.4)$$

gdzie:

\mathbf{f}_b^e – elementowy wektor bezwładnościowych sił węzłowych,

$\mathbf{M}^e = \int_V \mathbf{N}^T \rho \mathbf{N} dV$ - macierz bezwładności elementu.

Przedstawione równania dotyczą jednego elementu skończonego. Podlegają one tym samym etapom globalizacji (agregacja) jak równania równowagi statycznej z wykorzystaniem macierzy alokacji (opis w rozdziale 6.2.4), w wyniku czego otrzymujemy macierze i wektory globalne, opisujące pola w całej objętości modelu dyskretnego. Kolejne równania przedstawione będą dla całego modelu dyskretnego.

Dysponując macierzą bezwładności możemy wrócić do wyprowadzonego w rozdziale 6.2 równania sił statycznych i uzupełnić je dodatkowe człony związane z siłami bezwładności i siłami tłumienia. Pełna postać równania ruchu opisującego drgania swobodne układu jest następująca:

$$M\ddot{U} + C\dot{U} + KU + F = 0 \quad (10.5)$$

z warunkami początkowymi $\dot{U} = \dot{U}_0$ i $\ddot{U} = \ddot{U}_0$

gdzie:

M , C , K – odpowiednio globalne macierze sił bezwładności, tłumienia i sztywności,
 F – globalny wektor uogólnionych sił węzłowych.

Równanie ruchu może przyjąć kilka szczególnych postaci, dla naszego zagadnienia (drgania swobodne nietłumione), elementy macierzy C przyjmą wartości zerowe, podobnie przy braku wymuszenia składowe wektora F będą równe zero. Na układ działa tylko wstępne wzbudzenie. Ostatecznie równanie będzie postaci:

$$M\ddot{U} + KU = 0 \quad (10.6)$$

Jest to układ jednorodnych równań różniczkowych o stałych współczynnikach. Rozwiązanie układu ma postać:

$$U = U_A \sin(\omega t) \quad (10.7)$$

gdzie:

U_A – globalny wektor amplitud,
 ω - częstość w radianach.

Dwukrotnie różniczkując równanie 10.7 względem czasu otrzymujemy:

$$\ddot{U} = -\omega^2 U_A \sin(\omega t) \quad (10.8)$$

Po wstawieniu równań 10.7 oraz 10.8 do równania 10.6 otrzymamy:

$$(K - \eta M)U_A = 0 \quad (10.9)$$

gdzie:

$$\eta = \omega^2$$

W przypadku zadania w zakresie liniowo-sprężystym macierze \mathbf{K} i \mathbf{M} są niezależne od ω . Odrzucając trywialny przypadek zerowych przemieszczeń, rozwiązanie równania 10.9 znajdziemy w postaci:

$$\det(\mathbf{K} - \eta\mathbf{M}) = 0 \quad (10.10)$$

W praktyce rozwiązanie równania 10.10 przez wyznaczenie pierwiastków wielomianu uzyskanego na podstawie wyznacznika jest trudne i równanie to rozwiązuje się poprzez sprowadzenie do układu równań algebraicznych. Po obustronnym pomnożeniu równania 10.9 przez \mathbf{M}^{-1} otrzymujemy:

$$(\mathbf{M}^{-1}\mathbf{K} - \eta\mathbf{I})\mathbf{U}_A = 0 \quad (10.11)$$

gdzie:

\mathbf{I} – macierz jednostkowa.

Jeśli za pomocą triangulacji Banasiewicza-Choleskiego przedstawimy macierz bezwładności jako iloczyn macierzy trójkątnych, to otrzymamy:

$$\mathbf{M} = \mathbf{L}\mathbf{L}^T \quad (10.12)$$

oraz:

$$\mathbf{U} = (\mathbf{L}^T)^{-1}\mathbf{J} \quad (10.13)$$

gdzie:

\mathbf{L} – macierz dolna trójkątna otrzymywana w wyniku triangulacji Banasiewicza-Choleskiego [36], [21],

\mathbf{J} – wektor własny.

Podstawiając równania 10.12 i 10.13 do równania 10.11 i mnożąc obie strony równania przez \mathbf{L}^T otrzymamy zagadnienie własne w postaci:

$$(\mathbf{H} - \Lambda\mathbf{I})\mathbf{J} = 0 \quad (10.14)$$

gdzie:

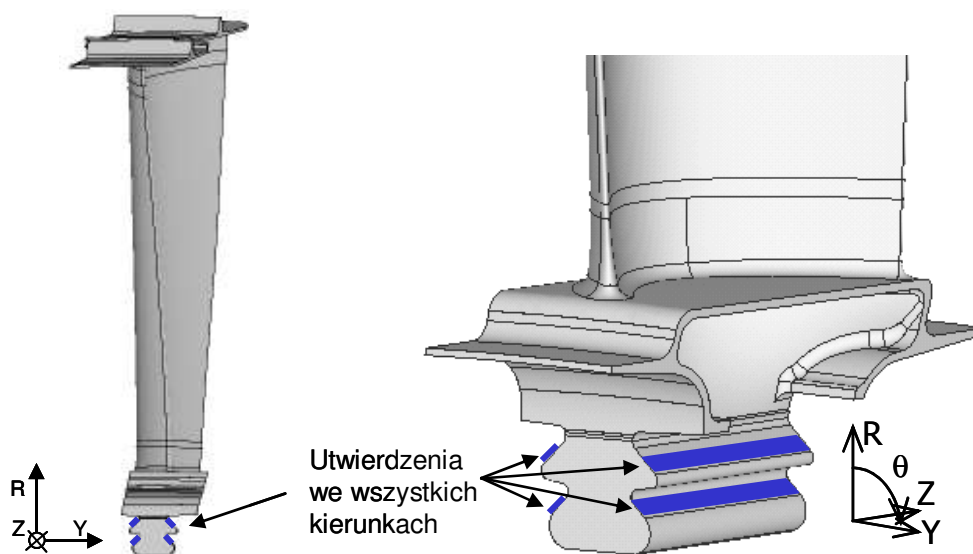
$$\mathbf{H} = \mathbf{L}^{-1}\mathbf{K}(\mathbf{L}^{-1})^T$$

$$\Lambda = \eta(\mathbf{L}^{-1})^T$$

Równanie 10.14 jest układem liniowych równań algebraicznych i można go rozwiązać jedną z wielu powszechnie znanych metod algebry liniowej (np. metodą bezpośrednią) [36], [43].

10.3. Model symulacyjny MES do analizy częstości drgań własnych łopatk

Do analizy wpływu zmiany kształtu łopatki na częstość jej drgań własnych został wykorzystany model łopatki utwierdzonej we wszystkich kierunkach na powierzchniach kontaktu z zamkiem jodełkowym dysku. Modele *MES* przygotowane były poprzez wyodrębnienie łopatki z pełnego modelu segmentu rotora w konfiguracji optymalnej dla każdego rozpatrywanego przypadku, siatka elementów skończonych na łopatkce była więc identyczna z wykorzystaną w analizie wytrzymałościowej. Obliczenia przeprowadzono w programie *Ansys*. Analiza modalna przeprowadzona została w temperaturze otoczenia i przy zatrzymanej turbinie (bez uwzględnienia usztywnienia wynikającego z prędkości obrotowej rotora). Schemat modelu *MES* łopatki turbiny do analizy modalnej przedstawia rys. 10.1.



Rys. 10.1 Schemat modelu *MES* łopatki turbiny do analizy modalnej.

10.4. Wyniki analizy częstości drgań własnych łopatk

Przeprowadzono obliczenia dla pierwszych pięciu postaci drgań własnych łopatki wszystkich modeli zoptymalizowanych i modelu w konfiguracji początkowej. Wyniki zostały znormalizowane względem częstości drgań własnych modelu w konfiguracji „0”, przedstawia je tabela 10.1.

Tabela 10.1. Porównanie wyników analizy modalnej dla modeli łopatek w różnych konfiguracjach geometrycznych.

Numer postaci własnej	Konfiguracja „0”	Model o zminimalizowanej masie	Model o zminimalizowanych naprężeniach głównych	Model po optymalizacji dwukryterialnej
1	1.00	0.99	1.00	1.02
2	1.00	0.99	1.00	1.00
3	1.00	0.99	1.00	0.99
4	1.00	0.96	1.00	1.00
5	1.00	0.99	1.00	0.99

Jak widać maksymalne obniżenie częstości drgań własnych zachodzi dla postaci nr 4 modelu o zminimalizowanej masie, gdzie różnica wynosi 4% na korzyść modelu w konfiguracji „0”. Jest to związane z oczekiwanym obniżeniem sztywności tego modelu przy redukcji masy, wszystkie częstości drgań własnych zostały obniżone. Nie ma znaczącej zmiany częstości dla modelu o zminimalizowanych naprężeniach. Dla modelu po optymalizacji dwukryterialnej maksymalna różnica wynosi 2% na korzyść modelu optymalnego, pozostałe różnice nie przekraczają 1%.

Wyniki tych obliczeń pozwalają stwierdzić, że optymalizacja kształtu łopatki w rozpatrywanej strefie nie będzie miała dużego wpływu na jej częstości drgań własnych a co za tym idzie na charakterystykę dynamiczną rotora turbiny. Można się jedynie spodziewać niewielkich różnic związanych ze zmianą masy łopatki.

11. PODSUMOWANIE I WNIOSKI KOŃCOWE

Podsumowując otrzymane wyniki obliczeń można stwierdzić, że zasadnicze założenia pracy zostały zrealizowane a jej teza udowodniona. Algorytm ewolucyjny połączony z *MES* okazał się metodą skuteczną i efektywną w zastosowaniu do optymalizacji kształtu łopatkę turbiny. Procedura optymalizacyjna została przygotowana w sposób prawidłowy, stworzone oprogramowanie funkcjonowało bezbłędnie. Wyniki mają bardzo dużą wartość poznawczą.

Zostały osiągnięte następujące rezultaty:

1. Obniżenie wartości maksymalnych naprężeń głównych o 15.9% przy zachowaniu dotychczasowej masy optymalizowanego fragmentu łopatkę.
2. Obniżenie masy optymalizowanego fragmentu łopatkę o 6.6% przy zachowaniu dotychczasowej wartości maksymalnych naprężeń głównych.
3. Równoczesne obniżenie masy o 4.2% i wartości maksymalnych naprężeń głównych o 11.0% oraz uzyskanie na froncie Pareto całej rodziny wyników lepszych od rozwiązania wyjściowego.

Na podstawie charakteru zmienności rozwiązań w trakcie optymalizacji, można stwierdzić z dużym prawdopodobieństwem, że są to wartości leżące w bliskim sąsiedztwie optimum globalnych. W dostępnej obecnie literaturze nie została zaprezentowana metoda pozwalająca na osiągnięcie zbliżonych wyników. Czas obliczeń (wynoszący poniżej 10 dni dla optymalizacji jednokryterialnej i ponad 70 dni dla optymalizacji dwukryterialnej) może zostać znacząco skrócony poprzez zastosowanie bardziej wydajnego stanowiska obliczeniowego, w trakcie pracy przy zaprezentowanych tutaj wynikach korzystano ze stanowiska o bardzo ograniczonych możliwościach. Przy wykorzystaniu jednej z wielu profesjonalnych inżynierskich stacji wieloprocesorowych średniej klasy dostępnych obecnie na rynku, można się spodziewać nawet 10-krotnego skrócenia czasu obliczeń. To umożliwiłoby uwzględnienie w obliczeniach kilku przypadków obciążenia, lub rozciągnięcie strefy optymalizowanej na górną część pióra łopatkę, gdzie geometria jest bardzo podobna. Możliwe jest również zastosowanie metody w optymalizacji innych części turbiny o znacznie mniej złożonym kształcie (uszczelnienia międzystopniowe, zamek jodełkowy), gdzie parametryzacja modelu i generacja siatki elementów skończonych nie przedstawiają tak dużych problemów praktycznych jak w wypadku części pióra łopatkę.

Ze względu na krótki okres czasu jaki upłynął od momentu uzyskania wyników nie ma dotychczas informacji o ich zastosowaniach praktycznych. Mogą zostać wykorzystane bezpośrednio wyniki obliczeń jak i sama metoda, lub jej elementy. Prace prowadzone były w

porozumieniu z inżynierami firmy *Avio Group* i ich wyniki spotkały się z bardzo dużym zainteresowaniem. W szczególności metoda parametryzacji modelu oraz sposób integracji środowiska *CAD* z programem obliczeniowym (bezpośrednia generacja modelu 3D na podstawie siatki sparametryzowanych krzywych w środowisku *Ansys*) są obiektem zainteresowania i istnieje duże prawdopodobieństwo ich wykorzystania w procesie projektowania podobnych elementów w przyszłości. Także wykorzystanie części programu optymalizacyjnego generującego modele *MES* na podstawie zestawu parametrów geometrycznych w innych zastosowaniach (*Robust Design, DOE*) jest bardzo prawdopodobne. Charakter otrzymanych wyników może wskazać możliwe kierunki modyfikacji istniejących rozwiązań i pokazać jakie możliwości niesie ze sobą wykorzystanie optymalizacji ewolucyjnej w tego typu zastosowaniach.

Jeśli planowane byłoby wykorzystanie przemysłowe, należałoby uogólnić metodę przygotowania modelu wyjściowego oraz doboru dopuszczalnego zakresu wartości genów. Można to zrealizować na przykład poprzez iteracyjną generację modeli ze stopniowym zwiększaniem wartości poszczególnych parametrów, co jest łatwe do zautomatyzowania i nie wymaga szczegółowej analizy kształtu modelu przez człowieka. Najbardziej wskazane kierunki dalszych badań zapoczątkowanych tą pracą mogłyby obejmować:

1. Jednoczesną optymalizację kształtu stref przejścia pióro – półka dolna/górna łopatki.
2. Optymalizację kształtu całego pióra łopatki w celu zbilansowania momentów gnących od sił gazu i sił odśrodkowych (np. poprzez zmianę przebiegu szkieletowej pióra).
3. Uwzględnienie w funkcji przystosowania kryterium wytrzymałości na zmęczenie wysokocyklowe i wymagań wobec charakterystyki dynamicznej łopatki.
4. Wykorzystanie stworzonego środowiska optymalizacji w celu analizy typu *Robust Design*.

12.STRESZCZENIE (SUMMARY)

Typowa łopatka turbiny jest elementem wymagającym bardzo złożonej symulacji inżynierskiej. Jej kształt jest zdeterminowany przez funkcje aerodynamiczne, ma ona bezpośredni kontakt z gorącym strumieniem gazów, działa w warunkach wysokich temperatur i obciążeń siłą odśrodkową. Ostateczna forma łopatki stanowi kompromis pomiędzy wymaganiami podyktowanymi sprawnością aerodynamiczną, dotyczącymi charakterystyki dynamicznej i wytrzymałości doraźnej, wytrzymałości na zmęczenie nisko- i wysokocyklowe, odporności na pękanie, wymaganiami technologicznymi, itp.

Optymalizacja kształtu łopatek w procesie projektowania przemysłowego jest zwykle przeprowadzana w bardzo ograniczonym zakresie, za pomocą serii analiz metodą prób i błędów, ponieważ złożoność problemu sprawia, że tradycyjne algorytmy optymalizacyjne są trudne do zastosowania.

Algorytmy ewolucyjne wydają się tutaj interesującą alternatywą, ponieważ szukają optymalnego rozwiązania tylko na podstawie wartości funkcji przystosowania i nie wymagają spełnienia dodatkowych warunków narzucanych przez tradycyjne algorytmy optymalizacyjne (ciągła funkcja celu, wąska przestrzeń rozwiązań dopuszczalnych).

Niniejsza rozprawa prezentuje optymalizację kształtu dolnego fragmentu pióra łopatki turbiny przemysłowej za pomocą algorytmu ewolucyjnego sprzężonego ze środowiskiem do analizy za pomocą *MES* (program *Ansys*) z wykorzystaniem wewnętrznego języka skryptowego (*APDL – Ansys Parameter Design Language*). Optymalizacja została przeprowadzona na trójwymiarowym modelu z pełnym zestawem warunków brzegowych reprezentujących warunki robocze turbiny.

Zostały przeprowadzone trzy oddzielne analizy:

1. Ewolucyjna optymalizacja jednokryterialna kształtu łopatki ze względu na minimum masy z narzuconym warunkiem maksymalnego naprężenia głównego – uzyskano redukcję masy o 6.6% w stosunku do modelu wyjściowego.
2. Ewolucyjna optymalizacja jednokryterialna kształtu łopatki ze względu na minimum naprężenia głównego z narzuconym warunkiem maksimum masy – uzyskano redukcję naprężenia o 15.9% w stosunku do modelu wyjściowego.
3. Ewolucyjna optymalizacja dwukryterialna kształtu łopatki ze względu na minimum masy i naprężenia głównego z narzuconymi warunkami maksymalnej masy i maksymalnego naprężenia – uzyskano równoczesne obniżenie masy o 4.2% i wartości maksymalnych naprężeń głównych o 11.0%.

Przeprowadzono szczegółową analizę otrzymanych wyników oraz zaproponowano dalsze prace badawcze oparte na opracowanej metodzie, z uwzględnieniem innych stref łopatk (pióro, strefa połączenia pióro-daszek) oraz innych postaci funkcji przystosowania (wyważenie łopatki w celu zbilansowania momentów gnących na piórze, optymalizacja charakterystyki modalnej).

A typical turbine blade is an element requiring a very complex design process. It operates very often in conditions of extremely high temperature, inertial and gas pressure loads. The blade shape results usually from a detailed engineering analysis and is a compromise between aerodynamic efficiency, dynamic characteristics, structural stress state, low- and high-cycle fatigue strength, creep resistance, technological requirements, etc.

Blade shape optimization in industrial design process is performed usually in a very limited range, with a series of trial-and-error analyses, as complexity of the problem causes the traditional optimizations algorithms to be difficult to implement or operate. The evolutionary algorithm seems an interesting alternative here, as it searches for the optimal solution basing only on fitness function value and does not require meeting the limitations of traditional optimizations algorithms (continuous objective function, narrow search domain, etc.).

This dissertation presents optimization of a turbine blade shank shape with the evolutionary algorithm coupled with *FEM* analysis environment (*Ansys*) by means of internal scripting language (*APDL – Ansys Parameter Design Language*). The optimization was performed on 3D-geometry with a full structural and thermal loads set representing turbine operating conditions.

Three separate analyses were performed:

1. Single-objective evolutionary optimization with a goal of 1st principal stress minimization and maximum allowable mass constraint imposed – obtained stress reduction with respect to original configuration is 15.9 %.
2. Single-objective evolutionary optimization with a goal of mass minimization and maximum allowable 1st principal stress constraint imposed - obtained mass reduction with respect to original configuration is 6.6 %.
3. Multi-objective evolutionary optimization with a goal of simultaneous 1st principal stress and mass minimization, with maximum allowable stress and mass constraint imposed – obtained stress and mass reduction with respect to original configuration is respectively 11.0% and 4.2%.

Detailed analysis of obtained results is presented together with proposal of further research based on the same approach, taking into account other zones of the blade (airfoil, airfoil – shroud transition zone) and other types of fitness functions (balancing of bending moments along blade airfoil, optimization of dynamic characteristics).

LITERATURA

- [1] Alander J. T. (Ed.), *Proceedings of the Second Nordic Workshop on Genetic Algorithms and their Applications (2NWGA)*, Vaasa, Finland, 1996
- [2] Ansys Inc. , *Ansys 6.1 Documentation*, SAS IP Inc., 2002
- [3] Arabas J., *Wykłady z algorytmów ewolucyjnych*, WNT, Warszawa, 2001
- [4] Bąk R., Burczyński T., *Wytrzymałość materiałów z elementami ujęcia komputerowego*, WNT, Warszawa, 2001
- [5] Burczyński T. (Ed.), *Computational Sensitivity Analysis and Evolutionary Optimization of Systems with Geometrical Singularities*, Scientific Publications of Department for Strength of Materials and Computational Mechanics No.1, Politechnika Śląska, Gliwice, 2002
- [6] Burczyński T., Cholewa W. (eds.), *Methods of Artificial Intelligence in Mechanics and Mechanical Engineering*, AI-MECH Series, Gliwice, 2000
- [7] Burczyński T., Osyczka A.(Eds.), *Evolutionary Methods in Mechanics*, Kluwer Academic Publishers , Dordrecht, 2004
- [8] Cichoń Cz., Cecot W., Krok J., Pluciński P., *Metody komputerowe w liniowej mechanice konstrukcji*, Wydawnictwo Politechniki Krakowskiej, Kraków, 2002
- [9] Cravero C. , Satta A., *A Hierarchical Optimisation Approach For Automatic Turbomachinery Blade Design*, AIAA-2001-3044, AIAA Fluid Dynamics Conference and Exhibit, 31st, Anaheim, CA, June 11-14, 2001
- [10] D'Angelo S., Fantetti M. Minisci. E., *Hang-Glider Wing Design by Genetic Optimization*, IUTAM Symposium on Evolutionary Methods in Mechanics, Kluwer Academic Publishers , Dordrecht/Boston/London, 2004
- [11] Dźygadło Z., Łyżwiński M., Otyś J., Szczeciński S., Wiatrek R., *Zespoły wirnikowe silników turbinowych*, WKiŁ, Warszawa, 1982
- [12] Felippa C. A., *Introduction to Finite Element Method - Course Material* - Department of Aerospace Engineering Science and Center for Aerospace Structures University of Colorado, USA, www.colorado.edu/engineering/Aerospace/CAS/courses.d/IFEM.d
- [13] Fonseca C. M. , Fleming P.J., *Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms II: Application Examples*, Research Report 565, Dept. Automatic Control and Systems Eng., University of Sheffield, Sheffield S1 4DU, U.K., 1995
- [14] Frola C., *Blade Fir Tree Design By Genetic Algorithm Method*, 2004 modeFRONTIER Users' Meeting, Trieste, 2004

- [15] Gätzi R., Uebersax M., König O., *Structural Optimization Tool using Genetic Algorithms and Ansys*, 18th CAD-FEM Users Meeting, Friedrichshafen, 2000
- [16] Grela W., Burczyński T., *Evolutionary Shape Optimization of a Turbine Blade Shank With APDL Language*, Recent Developments in Artificial Intelligence Methods, AI-METH Series, Gliwice, 2004
- [17] Grela W., Burczyński T., *Evolutionary Stress Minimization on a Turbine Blade Shank*, Computer Assisted Mechanics and Engineering Sciences, Vol. 12, No. 2-3, Warszawa, 2005, pp. 147-161
- [18] Grygiel K., *Wstęp do obliczeń ewolucyjnych i neuronowych – wykłady*, Instytut Informatyki UW, Warszawa, 2003
- [19] Hsu Tai-Ran, *The Finite Element Method in Thermomechanics*, Kluwer Academic Publishers, Boston, 1996
- [20] Hubner W., Fischer M., Himmler G., *Stress Reduction in an Interstage Air Seal using TOSCA (MSC.Construct) for increasing the LCF Fatigue Life*, 3rd FENet Annual Industry Meeting & AGM, Hamburg, 2003
- [21] Janecki S., Krawczuk M., *Dynamics of Steam Turbine Rotor Blading*, Ossolineum, Wrocław, 1998.
- [22] Juraszek J., Babiński L., Thorpe J., *Metoda elementów skończonych*, Bielsko – Biała, Wydawnictwo Politechniki Łódzkiej, 1998
- [23] Kleiber M (Ed.), *Handbook of Computational Solid Mechanics*, Berlin, Springer, 1998.
- [24] Kozłowski T., *Zarys teorii sprężystości*, Arkady, Warszawa, 1968
- [25] Krenich S., Osyczka A., *Optimal Design of Multiple Clutch Brakes Using a Multistage Evolutionary Method*, IUTAM Symposium on Evolutionary Methods in Mechanics, Kluwer Academic Publishers , Dordrecht/Boston/London, 2004, pp. 219—228
- [26] Lach G., J. Stadnicki (promotor) , *Analiza strukturalna oraz modalna wycinka wirnika turbiny gazowej* (Praca Dyplomowa Inżynierska nr 52), Akademia Techniczno-Humanistyczna, Bielsko-Biała, 2003
- [27] Michalewicz Z., *Genetic Algorithms + Data Structures = Evolutionary Programs*, Springer-Verlag, AI Series, New York, 1992
- [28] Miettinen K., Neittaanmaki P., Mäkelä M. M., Périaux J. (Eds.), *Evolutionary Algorithms in Engineering and Computer Science: Recent Advances in Genetic Algorithms*, Evolution Strategies, Evolutionary Programming, Genetic Programming and Industrial Applications, Wiley, 1999
- [29] Murawski K. , *Obliczenia ewolucyjne – geneza i zastosowanie*, Biuletyn Instytutu Automatyki i Robotyki WAT Nr 15, Warszawa, 2001
- [30] Noack J. , *Topology Optimization with Casting and Machining Restrictions by the Example of Brake Parts (TRW Automotive)*, CAD-FEM Users' Meeting, Poczdam, 2003.

- [31] Nowacki W. , *Teoria sprężystości*, PWN, Warszawa, 1970
- [32] Nowotarski I. , *Obliczenia statyczne i dynamiczne turbinowych silników lotniczych metodą elementów skończonych*, Biblioteka Naukowa Instytutu Lotnictwa, Warszawa, 2001
- [33] Piegl L. , Tiller W., *The NURBS Book*, Springer-Verlag, New York, 1997.
- [34] Poloni C., *Evolutionary Algorithms in modeFRONTIER, 2nd FENet Technology Workshop*, Zurich, 2002
- [35] Pühlhofer T., Gleichmar R., Langer H., Müller M., Müller U.C., Baier H., *Multidisciplinary Optimization and Statistical Evaluations of Structures and Electro-Mechanical Components with ANSYS on a PC-Cluster*, 20th CAD-FEM Users Meeting, Friedrichshafen, 2002
- [36] Rakowski G., Kacprzyk, Z., *Metoda elementów skończonych w mechanice konstrukcji*, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa, 1993
- [37] Rakowski G., *Sprężystość – problemy i rozwiązania. Metody analityczne i numeryczne*, Wydawnictwo Politechniki Świętokrzyskiej, Kielce, 2001
- [38] Rogers J.L., *A Parallel Approach To Optimum Actuator Selection With A Genetic Algorithm*, NASA Langley Research Center, Hampton, VA, USA, AIAA-2000-4484.
- [39] Shahpar S., *A Comparative Study Of Optimisation Methods For Aerodynamic Design Of Turbomachinery Blades*, ASME, 2000-GT-523, 2000
- [40] Stettner M., Haase W., *Multi-Objective Aeroelastic Optimisation, NATO meeting on Aerodynamic Design and Optimisation of Flight Vehicles in a Concurrent Multi-Disciplinary Environment*, Ottawa, 1999
- [41] Szczeciński S. , *Dwuwirnikowe i dwuprzepływowe lotnicze silniki turbinowe*, WKiŁ, Warszawa, 1971
- [42] Van Veldhuizen D.A., Lamont G. B. , *Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art*, Evolutionary Computation, 8(2), 2000, pp. 125-147
- [43] Zienkiewicz O.C., Taylor R.L., *The Finite Element Method*, Butterworth – Heinemann, Bristol, 2000

DODATEK 1 - KOD ŹRÓDŁOWY PROGRAMU OPTYMALIZACYJNEGO (APDL)

1. Procedura MASTER

```
FINISH
/CLEAR,START
/UIS,MSGPOP,3
/NERR,-1,1.0E+9,,1
/PREP7
/PAGE,1000,1000,1000,200 ! page width for printout
! setting environment
!/dev,font,2,Courier*New,400,0,-20,0,0,,
/COLOR,PBAK,OFF
!/TRIAD,LBOT
*AFUN,DEG
SHPP,OFF ! shape checking off
BOPT,NUMB,OFF

generat=150! number of generations to run
chromos=30 ! number of chromosomes in generation
gene=17 ! number of genes
max_div = 8 ! max no. of divisions for one profile part
part_no = 4 ! number of profile sections

ro = 1/(chromos/2) ! selection pressure ratio
mut_std_dev = 0.1 ! mutation std. deviation
stress_max = 65500! maximum allowable stress - if higher, mass is doubled
mass_max = 0.0004022654563
a1 = 0.5 ! coefficients for FF calculation
a2 = 0.5

G =1! generations counter setting - starting generation

! GEOMETRIC PARAMETERS
R1_min = 2
R1_max = 750
Hmax = 0.595 ! maximum compound fillet height
Hmin = 0.2
R2_min = 0.005 !minimum R2_radius
R2_max = Hmax! in this case, compound fillet is reduced to simple fillet
Hmax2 = 0.5978! radial distance between platform and cutting plane
DIST_MIN = 0.012 !min. distance between edge of spline and platform edge
DIST_PLTF = 0.1 ! minimum distance of rotated point line from the platform

! matrix to store actual radii if lower fillet which can be
! modified by program to fit fillet on the platform
*DIM,radius_check,array,max_div,part_no,2
! %max_div% rows - radii at divisions
! %part_no% columns - PS, SS, LE, TE
! plane 1 - R1
```

```

! plane 2 - R2
! dimensioned here because values have to be preserved in case
! of model parameters modification

*DIM,COM_NUM,array,part_no,1,3 ! lines at the airfoil edge and platform
COM_NUM(1,1,1)=5,6,7,8 ! lines of spline at the platform
COM_NUM(1,1,2)=1,3,4,2 ! lines at the shank/airfoil cutting plane
COM_NUM(1,1,3)=11,12,9,10 ! platform edges
! sequence PS_LINE,LE_LINE,SS_LINE,TE_LINE

! array of local coordinate systems rotation angles (rot about axis X)
*DIM,CROT,array,part_no, 1
! %part_no% rows - 4 THYZ angles (PS, SS, LE, TE)
! 1 column - 1 set of rotation angles

! array of divisions parametric positions along profile section
*DIM,DIV,array,1+max_div,part_no,1
! (1 + max_div) rows - divisions no. + parametric positions values
! %part_no% columns - PS, SS, LE, TE

! array of starting and ending kps of arcs and fillet lines
*DIM,ENT_NO,array,max_div+1,11,part_no
! max_div rows - max no. of divisions for one profile part
! 18 columns - starting (1), intermediate (2) and bottom (3),
! free column(4) and number of upper (5),lower fillet line (6)
! line connecting points on upper fillets points(7)
! line connecting points on intermediate fillets points(8)
! line connecting points on lower fillets points(9)
! col 10 - areas around upper fillet
! col 11 - areas around lower fillet
! %part_no% planes - PS, SS, LE, TE,

! matrix containing geometric parameters of generation
*DIM,param,array,chromos,gene+3,1,
! matrix containing geometric parameters of next generation
*DIM,param_new,array,chromos,gene,1,

! matrix containing geometric parameters of one model
*DIM,chromosome,table,3,3,part_no
! first row - R1, second row - R2, third row - H
! for now max. number of defined params for each section = 0.0, 1.0
! key defining section in which radii are defined
*DIM, key, array, max_div, part_no

*DIM, old_sequence,array,chromos,1 ! for sorting, dummy
*DIM, evolution, array, generat, gene+2 ! 1st column - generation#, 2nd col -FF value
*DIM, mutation1, array, chromos,gene+3,1,
*DIM, mutation2, array, chromos,gene+3,1, ! matrices for reproduction operation
*DIM, seq, array, chromos, 4!selection matrix
*DIM, mut_mask,,gene,1 ! mutation mask vector
*DIM, penalty, array, chromos, 1!selection matrix

! number of divisions along profile circumference
! watch for the first parameter, equal to divisions number
div(1,1) = 7, 0.0, 0.15, 0.3, 0.496,0.65, 0.8, 1.0 !PS
div(1,3) = 8, 0.0, 0.1, 0.25, 0.4, 0.5025, 0.65, 0.75, 1.0 !SS
div(1,2) = 1, 0.555 !LE

```

```

div(1,4) = 1, 0.6 !TE

! calculation of total number of divisions
tot_div=0
*DO, p, 1, part_no, 1
  tot_div=tot_div+div(1,p)
*ENDDO

! angles of rotation of local csys to align with profile edge
crot(1,1)=90, -90, 90, 90

! here I declare which sections are defined and which will be interp.
key(1,1) = 1,0,0,1,0,0,1
key(1,3) = 1,0,0,0,1,0,0,1
key(1,2) = 1
key(1,4) = 1
! definition of index values for table chromosome
*DO, p, 1, part_no, 1
  chromosome(0,0,p)=p ! indexes of planes for each prof. section
  chromosome(1,0,p)=1 ! indexes of values of R1
  chromosome(2,0,p)=2 ! indexes of values of R2
  chromosome(3,0,p)=3 ! indexes of values of H
  i_loop = 1 ! counter for row position in array chromosome
  *DO, i, 1, div(1,p), 1
    *IF,key(i,p),EQ,1,THEN
      chromosome(0,i_loop,p)=div(i+1,p)
      *IF,p,EQ,2,OR,p,EQ,4, THEN
        chromosome(0,2,p) = 1
        chromosome(0,2,p) = 1 ! to nie ma znaczenia bo zawsze interpoluje
        chromosome(0,3,p) = 1 ! w granicach dla 1 plane'a
        chromosome(0,3,p) = 1
      *ENDIF
      ! parametric positions of defined radii for each section
      ! value for 0 parametric positon
      i_loop = i_loop+1
    *ENDIF
  *ENDDO
*ENDDO

*IF, G, EQ, 1, THEN
  *VFILL, param(1, gene), RAND, Hmin, Hmax
  *DO, j, 1, (gene-1)/2, 1
    *VFILL, param(1,j), RAND, R1_min, R1_max
  *DO,i, 1, chromos, 1
    param(i,(gene-1)/2+j) = RAND(R2_min, param(i, gene))
    ! R2 cannot be larger than H, in case R2=H we have simple fillet
  *ENDDO
  *ENDDO
  *MWRITE, param, generation_%G%, 001, , JJK, gene, chromos, 1, 0
  (17F15.8)
*ENDIF

:GENERATE ! generation loop

! compound fillet parameters - file header
! File defining geometric parameters of models in generation 1
!R11 R12 R13 R14 R15 R16 R17 R18 R21 R22 R23 R24 R25 R26 R27 R28 H

```



```

*VREAD, param(1,1,1) , generation_%G%, 001, , JIK, gene, chromos, 1, 0
(17F15.8)

start_chrom = 1
*IF, G, GT, 1, THEN
  start_chrom = 2
  *VREAD, param(1,1,1) , generation_%G-1%, 003, , JIK, gene+3, 1, 1, 0
  (20F15.8)
*ENDIF
! calling subroutine to build the model
*DO, u, start_chrom , chromos,1
  ! defining model parameters for chromosome u
  j_loop=1 ! counter for column positions in row u of param array
  *DO, p, 1, part_no, 1
    i_loop = 1
    *DO, i, 1, div(1,p), 1
      *IF,key(i,p),EQ,1,THEN
        chromosome(1,i_loop,p)=param(u,j_loop) ! values of R1
        chromosome(2,i_loop,p)=param(u,j_loop+(gene-1)/2)! val.of R2
        chromosome(3,i_loop,p)=param(u,gene)! for now H is constant
      *IF, p, EQ, 2, OR, p, EQ, 4, THEN
        chromosome(1,2,p)=param(u,j_loop) ! values of R1
        chromosome(2,2,p)=param(u,j_loop+(gene-1)/2)! val.of R2
        chromosome(3,2,p)=param(u,gene)! for now H is constant
        chromosome(1,3,p)=param(u,j_loop) ! values of R1
        chromosome(2,3,p)=param(u,j_loop+(gene-1)/2)! val.of R2
        chromosome(3,3,p)=param(u,gene)! for now H is constant
      *ENDIF
      j_loop = j_loop+1
      i_loop = i_loop+1
    *ENDIF
  *ENDDO
*ENDDO

m_gen15 ! calling subroutine to build the model

! substitution of original parameters with modified ones
j_loop = 1 ! counter for column position in param array
*DO, p, 1, part_no, 1
  *DO, i, 1, div(1,p), 1
    *IF,key(i,p),EQ,1,THEN
      param(u,j_loop+(gene-1)/2)=chromosome(2,div(i+1,p),p)!val.of R2
      param(u,j_loop) = chromosome(1,div(i+1,p),p)!val.of R1
      j_loop = j_loop+1
    *ENDIF
  *ENDDO
*ENDDO
param(u,gene+3) = a1*(MAX_1st_PRINC/stress_max)+ a2*(MASS/mass_max) ! FF
param(u,gene+2) = MAX_1st_PRINC ! max 1st principal stress
param(u,gene+1) = MASS ! mass
*ENDDO

! saving modified parameters into file
!*MWRITE, param , generation_%G%, 002, , JIK, gene, chromos, 1, 0
!(17F15.8)

!!! REPRODUCTION !!!

```

```

! penalty for all chromosomes with stress GT stress_max

*VOPER, seq(1,1), param(1, gene+2), GT, stress_max
*VOPER, seq(1,2), param(1, gene+1), GT, mass_max
*VOPER, seq(1,1), seq(1,1), ADD, seq(1,2)

*VMASK, seq(1,1)
*VOPER, param(1, gene+3), param(1, gene+3), MULT, 2
! penalty function, previous best chromosome is not punished if exists
*MOPER, old_sequence, param, SORT, param(1, gene+3)! sorting acc. to FF value

*MWRITE, param, generation_%G%, 003, JIK, gene+3, chromos, 1, 0
(20F15.8)

! random choosing of chromosomes for crossover
*VFILL, seq(1,3), RAMP, 1, 0
:SEQUENCE2
*VMASK, seq(1,3)
*VFILL, seq(1,1), RAND, 1, chromos/2
*VMASK, seq(1,3)
*VFUN, seq(1,1), NINT, seq(1,1)
:SEQUENCE1
*VMASK, seq(1,3)
*VFILL, seq(1,2), RAND, 1, chromos/2
*VMASK, seq(1,3)
*VFUN, seq(1,2), NINT, seq(1,2)
*VMASK, seq(1,3)
*VOPER, seq(1,3), seq(1,1), EQ, seq(1,2)
*VSCFUN, seq_rep, MAX, seq(1,3)
*IF, seq_rep, EQ, 1, :SEQUENCE1
! how many the same numbers in the selected set - maximum chromos/2 ?
! first column
*VFILL, seq(1,3), RAMP, 0, 0
*VFILL, seq(1,4), RAMP, 0, 0
*DO, j, 1, chromos, 1
  c_seq = seq(j,1)
  *VLEN, j, 0
  *VOPER, seq(1,4), seq(1,1), EQ, c_seq
  *VLEN, j, 0
  *VSCFUN, seq(j,3), SUM, seq(1,4)
*ENDDO
*VOPER, seq(1,3), seq(1,3), GT, (chromos/2)-1
*VSCFUN, seq_rep, MAX, seq(1,3)
*IF, seq_rep, EQ, 1, :SEQUENCE2
! how many the same numbers in the selected set - maximum chromos/2 ?
! second column
*VFILL, seq(1,3), RAMP, 0, 0
*VFILL, seq(1,4), RAMP, 0, 0
*DO, j, 1, chromos, 1
  c_seq = seq(j,2)
  *VLEN, j, 0
  *VOPER, seq(1,4), seq(1,2), EQ, c_seq
  *VLEN, j, 0
  *VSCFUN, seq(j,3), SUM, seq(1,4)
*ENDDO
*VOPER, seq(1,3), seq(1,3), GT, (chromos/2)-1

```

```

*VSCFUN, seq_rep, MAX, seq(1,3)
*IF, seq_rep, EQ, 1, :SEQUENCE1
! how many the same pairs in all rows ?
*VFILL, seq(1,3),RAMP, 0,0
*DO, j, 1, chromos, 1
  *DO, k, j+1, chromos, 1
    *IF, seq(j,1),EQ,seq(k,1),AND,seq(j,2),EQ,seq(k,2),THEN
      seq(k,3)=1
    *ENDIF
  *ENDDO
*ENDDO
*VSCFUN, seq_rep, MAX, seq(1,3)
*IF, seq_rep, EQ, 1, :SEQUENCE2
*MOPER, old_sequence, seq, SORT, seq(1,1)
! third gene
*VFILL, seq(1,3), RAND,1,chromos/2
*VFUN, seq(1,3), NINT, seq(1,3)
! selection done

! CROSSOVER
*DO, j, 1, gene, 1
  *VFUN, param_new(1,j), COPY, param(1,j)! first chromosome passed with no change
*ENDDO

*DO, j, 2, chromos, 1
  cut_1 = NINT(RAND(1,8))
  cut_2 = NINT(RAND(9,16))
  *DO, k, 1, cut_1, 1
    param_new(j,k)= param(seq(j,1),k)! first part crossed , gene 1-8
  *ENDDO
  *DO, k, cut_1+1, cut_2, 1
    param_new(j,k)= param(seq(j,2),k)! first part crossed , gene 1-8
  *ENDDO
  *DO, k, cut_2+1, 17, 1
    param_new(j,k)= param(seq(j,3),k)! first part crossed , gene 1-8
  *ENDDO
*ENDDO
! crossover done

!MUTATION
*DO, j, 2, chromos, 1
  mut1 = NINT(RAND(1,8))
  mut2 = NINT(RAND(9,17))
  param_new(j,mu1)= param_new(j,mu1)+ param_new(j,mu1)*GDIS(0,mu_std_dev )
  param_new(j,mu2)= param_new(j,mu2)+ param_new(j,mu2)*GDIS(0,mu_std_dev )
*ENDDO
! mutation done, two genes in all chromosomes mutated with ksi=GDIS(0,0.1)

*DO, j, 1, gene, 1
  ! truncating of obtained gene values to min-max range
  *IF, j, LE, (gene-1)/2, THEN
    *VFILL, seq(1,2) , RAMP, R1_min, 0
    *VOPER, seq(1,1) , param_new(1,j), LT, R1_min
    *VMASK, seq(1,1) ! masking all genes below R1_min
    *VFUN, param_new(1,j), copy, seq(1,2)
    *VFILL, seq(1,2) , RAMP, R1_max, 0
    *VOPER, seq(1,1) , param_new(1,j), GT, R1_max
  *ENDIF

```

```

*VMASK, seq(1,1) ! masking all genes GT R1_max
*VFUN, param_new(1,j), copy, seq(1,2)
*ELSEIF, j, GT, (gene-1)/2, AND, j, LT, gene
*VFILL, seq(1,2) , RAMP, R2_min, 0
*VOPER, seq(1,1) , param_new(1,j), LT, R2_min
*VMASK, seq(1,1) ! masking all genes below R2_min
*VFUN, param_new(1,j), copy, seq(1,2)
*VFILL, seq(1,2) , RAMP, R2_max, 0
*VOPER, seq(1,1) , param_new(1,j), GT, R2_max
*VMASK, seq(1,1) ! masking all genes GT R2_max
*VFUN, param_new(1,j), copy, seq(1,2)
*ELSE
*VFILL, seq(1,2) , RAMP, Hmin, 0
*VOPER, seq(1,1) , param_new(1,j), LT, Hmin
*VMASK, seq(1,1) ! masking all genes below Hmin
*VFUN, param_new(1,j), copy, seq(1,2)
*VFILL, seq(1,2) , RAMP, Hmax, 0
*VOPER, seq(1,1) , param_new(1,j), GT, Hmax
*VMASK, seq(1,1) ! masking all genes over Hmax
*VFUN, param_new(1,j), copy, seq(1,2)
*ENDIF
! parameters truncated
*ENDDO
!checking if R2 are not greater than H
*DO, j, (gene-1)/2+1, gene-1 , 1
*VOPER, seq(1,1) , param_new(1,j), GT, param_new(1,gene)
*VMASK, seq(1,1)
*VFUN, param_new(1,j),copy, param_new(1,gene)
! substitution of values of R2 greater than H with H values
! it reduces the cross-sections to simple fillet
*ENDDO

G=G+1 ! next generation

*MWRITE, param_new , generation_%G%, 001, , JIK, gene, chromos, 1, 0
(17F15.8)

*IF, G, LE, generat, :GENERATE ! generate next generation if needed

/SYS, del *.err
/SYS, del *.log
/SYS, del *.emat
/SYS, del *.esav
/SYS, del *.mntr
/SYS, del *.osav
/SYS, del *.full
/SYS, del *.bak

```

2. Procedura MGEN

```

! reading the preliminary model into database
FINISH
/NERR,-1,1.0E+9,
/UIS, MSGPOP, 3
/SHOW, OFF
BOPT,NUMB,OFF
/FILNAME,model_G%G%_CH%u%,1
/SYS, del *.err

```

```

/SYS, del *.log
/prep7

REPEAT_R2=0 ! variable eq. to 0 if model is prepared for the 1st time
! repeat=1 if model is repeated with changed values of R2
:R2_BEGIN

! reading base model into database
CDREAD,SOLID,,,,shank_iges_62,iges,

! boolean options
BOPT,NUMB,OFF
! compressing, storing and setting entities numbers
numcmp,all
pl_no=1

! creation of csys100
LOCAL,R5.0,LOC, 100, 0, 25.1236305 , 0.292623478 , 316.488130
LOCAL,R5.0,ANG, 100, 0, 0.663914945 , 8.361585464E-04,-0.710375113
LOCAL,R5.0,PRM, 100, 0, 1.00000000 , 1.00000000
CSCIR, 100, 0, 0
*AFUN,DEG

*DO, p, 1,part_no,1

! profile section selection and number storing
cut_lineid = com_num(p,1,2) ! shank/airfoil cut plane edge
! lines at cutting plane are oriented like PS-TE-SS-LE
lineid = com_num(p,1,1) ! line at the platform
! lines at cutting plane are oriented like PS-LE-SS-TE

! selecting entities
lsel,s,,,com_num(p,1,1)
lsel,a,,,com_num(p,1,2)
asel,a,,,pl_no

! creating compound fillets
*DO, i, 1, div(1,p),1

! defining radius parameters
R1 = chromosome (1,div(i+1,p),p)
R2 = chromosome (2,div(i+1,p),p)
H = chromosome (3,div(i+1,p),p)
! using table so all between given positions are interpolated

! checking if model is prepared for the 1st time
! switching to modified previously radii values
*IF, REPEAT_R2, GE,1, THEN
  R2 = radius_check(i,p,2)
  R1 = radius_check(i,p,1)
*ENDIF

! setting starting numbers for entities
kpmax = KPINQR(0,14)
lmaxd = LSINQR(0,14)
NUMSTR, LINE, lmaxd+1
NUMSTR, KP, kpmax+1

```

bopt,keep,yes

```
! offsetting WP to parametric pos. of spline and creation of loc. csys
LWPLAN,,lineid,div(i+1,p)
CSWPLA,10*p+i,0,
k,kpmax+1, 0, 0, 0
k,kpmax+2, 0, 0, 1
! csys100 - coordinate sys with X axis normal to pl_no at the center
csys,100
kgen,2,kpmax+1,,,1,0,0,2
cskp,10*p+i,0,kpmax+1,kpmax+3,kpmax+2
CLOCAL,10*p+i,0,0,0,0,,crot(p,1),,
!!! upper fillet creation
! arc endpoint creation (on Y axis)
REPEAT_R1 = 0
csys,10*p+i
! delta - offset along Y local csys axis due to non-radial offset
! of lower and upper profile boundary
Hmax2 = 0.5978 ! radial distance between platform and cutting plane
y_upper = ly(cut_lineid,1-div(i+1,p))! analogical point on the upper line
z_upper = lz(cut_lineid,1-div(i+1,p))
delta_y = (y_upper/Hmax2)*H
delta_z = (z_upper/Hmax2)*H
delta_H = -R1*sin(atan(y_upper/Hmax2))
delta_yy = -delta_H*(y_upper/Hmax2)
delta_Hz = 0
*IF, p, NE, 2, THEN
  *IF, p, EQ, 1, AND, i, EQ, 1, THEN
    delta_Hz = 1*sin(atan(z_upper/Hmax))
  *ENDIF
  *IF, p, EQ, 3, AND, i, EQ, 8, THEN
    delta_Hz = 1*sin(atan(z_upper/Hmax))
  *ENDIF
  *IF, p, EQ, 4, THEN
    delta_Hz = 1*sin(atan(z_upper/Hmax))
  *ENDIF
*ENDIF

:R1_BEGIN ! modification of R1 value if too small and after rot.
*IF,REPEAT_R1, GE,1,THEN ! arc endpoint is above platform
  ldele,lmaxd+1,,,1
  NUMSTR, LINE, lmaxd+1
  kdele, kpmax+4, kpmax+6
  R1 = 1.10*R1
  /COM, REP. NO %REPEAT_R1% LOOP R1_BEGIN PART %p% NDIV %i%, R1=%R1%
*ENDIF

rot_ang = -atan(3*H/R1) ! rotation angle
kpmax = KPINQR(0,14)
NUMSTR, KP, kpmax+1
k,kpmax+1, H,0+delta_y,0+delta_z ! creating upper fillet first point
! points defining the upper fillet axis
k,kpmax+2, H+delta_H, R1+delta_y-delta_yy,0
k,kpmax+3, H+delta_H-delta_Hz, R1+delta_y-delta_yy, 1
lrotat,kpmax+1,,,,,kpmax+2,kpmax+3,rot_ang,1 ! line lmaxd+1 generated
REPEAT_R1 = REPEAT_R1+1
*IF, KX(kpmax+4), GT, -DIST_PLTF, :R1_BEGIN
!*IF, KY(kpmax+4), LT, delta_y, :R1_BEGIN
```

```

LINA, lmaxd+1, pl_no ! intersecting line created by rotation with platform
*IF, KPINQR(0,14), NE, kpmx+5, :R1_BEGIN

! deleting kps defining the rotation axis
kdele,kpmx+2
kdele,kpmx+3
kdele,kpmx+5
ldele,lmaxd+2,,,1
!creating points
KGEN, 2, kpmx+1,,,5,0,0 ! kp id is kpmx+2
L, kpmx+1, kpmx+2, ! horiz. line from upper fillet start point kpmx+1 is lmaxd+2
ADRAG, lmaxd+1,,,,,lmaxd+2 ! upper fillet arc dragged along horizontal line

/NERR,-1,1.0E+9,
/UIS, MSGPOP, 4
*DO, jj, 0.0001, DIST_MIN*2, 0.0002
  btol, jj! changing tolerance for areas breaking - problems
  aina,pl_no,pl_no+1
  *IF, LSINQR(0,14), EQ, lmaxd+6, THEN
    *EXIT ! line created, id lmaxd+6
  *ENDIF
*ENDDO
/UIS, MSGPOP, 3

btol,0.01
! breaking lines - desired upper line part id is lmaxd+7
! lmaxd+1 is unchanged after operation
lsbl,lmaxd+1,lmaxd+6
btol,defa !!!
! cleaning
adele,pl_no+1,,,1
ldele,lmaxd+1,lmaxd+2,1,1
ldele,lmaxd+8,,1,1
NUMSTR, LINE, lmaxd+1

! checking for maximum fillet radius value, reducing if exceeded
*GET,kp5,LINE,lmaxd+6,KP,2
*GET,kp4,LINE,lmaxd+7,KP,2
*IF,kp5,EQ,kp4,THEN
  *GET,kp5,LINE,lmaxd+6,KP,1
*ENDIF
*GET,kp6,LINE,lmaxd+7,KP,1
LEN1=abs(DISTKP(kp4,kp5))
LEN2=abs(DISTKP(kp4,kp6))
LMIN=LEN1
*IF,LEN2,LT,LEN1,THEN
  LMIN=LEN2
*ENDIF
phi=ANGLEK(kp4,kp6,kp5)
FILMAX = LMIN*tan(phi/2) ! Calculate maximum fillet radius
*IF, R2, GE, FILMAX, THEN
  R2 = FILMAX*0.95 ! reducing R2 to 95% of max value if exceeded
*ENDIF

/NERR,-1,1.0E+9,
/UIS, MSGPOP, 4
R20 = R2

```

```

iii = 0
! lower fillet creation - line id is lmaxd+1
*IF, R20, GT, R2_min, THEN
  *DO, ri, R2, R2_min, -(R2-R2_min)/20 ! loop for FILMAX
    iii = iii+1
    /COM, %iii%th EXECUTION OF LFILLT FOR PART %p% SEC %i%
    lfillt,lmaxd+6,lmaxd+7,ri
    *IF, LSEL(lmaxd+1), EQ, 1, THEN
      /COM, REDUCING R2=%R20% to R2=%ri%
      *EXIT
    *ENDIF
  *ENDDO
*ELSEIF, R20, LE, R2_min
  lfillt,lmaxd+6,lmaxd+7,R2_min
  /COM, %iii%th EXECUTION OF LFILLT FOR PART %p% SEC %i%, R2_min used
*ENDIF
/UIS, MSGPOP, 3

! storing ending and starting kps of upper arc
lreverse,lmaxd+7
kp1=kpmax+1
*GET,kp2,LINE,lmaxd+7,KP,2
*GET,kp3,LINE,lmaxd+1,KP,1
!creating points on shank/airfoil division lines
kpmax = KPINQR (0,14)
! creating kps on shank/airfoil cut line
kl, cut_lineid,1-div(i+1,p),kpmax+1
kp5 = kpmax+1
l,kp1,kp5 ! desired line id is lmaxd+2
lcomb,lmaxd+7,lmaxd+2,0 !adding upper fillet and straight lines
! desired upper fillet line id is lmaxd+2

! adding lines of upper and lower fillet to divide them at constant
! height above the platform
lcomb, lmaxd+2, lmaxd+1, 0 ! desired line id is lmaxd+1
! what should be parametric position on the line to divide it
! at hdiv = 0.134 ? For now a division at 30% length will be done

WPLANE,,25.2236,0.293782,316.489, 26.2236, 0.293782, 316.489, 25.2236, 1.293782, 316.489
WPROT,0,0,90
LSBW,lmaxd+1,,DELETE

*IF,LSEL(lmaxd+4),EQ,1,THEN
  lcomb, lmaxd+4, lmaxd+3, 0 !!! in case of breaking the upper line in two
*ENDIF

! storing kp numbers in an array
ii=0
*IF,p,EQ,2,OR,p,EQ,4,THEN
  ii=1
*ENDIF
! starting fillet point before merginig
*GET,ent_no(i+ii,1,p),LINE,lmaxd+3,KP,2
! intermediate fillet point
ent_no(i+ii,2,p)=kp1
! bottom fillet point
ent_no(i+ii,3,p)=kp3 ! very rare error - number of bottom fillet endpoint

```



```

*IF, KSEL(kp3), EQ, 0, THEN
  ent_no(i+ii,3,p)=kp3+1
*ENDIF
! storing number of fillet lines for each section
! upper fillet line
ent_no(i+ii,5,p)=lmaxd+3
! lower fillet line
ent_no(i+ii,6,p)=lmaxd+2
! deleting bottom line on the platform
ldele,lmaxd+5,lmaxd+7,,1
! storing value of R which could be changed
radius_check(i,p,1) = R1
radius_check(i,p,2) = R2
*ENDDO

! cleaning and number substitution
allsel,all
nummrg,kp,,,HIGH
boptn,keep,no
! storing numbers- upper arc points after merging
*GET,ent_no(1+ii,1,p),LINE,ent_no(1+ii,5,p),KP,2
*GET,ent_no(div(1,p)+ii,1,p),LINE,ent_no(div(1,p)+ii,5,p),KP,2
! breaking shank/airfoil cut line
*DO,i,1,div(1,p),1
  lsbl,cut_lineid,ent_no(i+ii,5,p),,keep,keep
*ENDDO
*DO,i,1,max_div,1
  *DO,jj,1,part_no,1
    *IF, LSEL(ent_no(i,5,jj)), EQ, 1, THEN
      lsel,u,,ent_no(i,5,jj)
    *ENDIF
    *IF, LSEL(ent_no(i,6,jj)), EQ, 1, THEN
      lsel,u,,ent_no(i,6,jj)
    *ENDIF
  *ENDDO
*ENDDO
lovlap,ALL
*ENDDO

! storing additional numbers for LE and TE
*DO,p,2,part_no,2
  *IF,p,EQ,2,THEN
    *DO,i,1,6,1
      ent_no(1,i,p)=ent_no(div(1,p-1),i,p-1)
      ent_no(div(1,p)+2,i,p)=ent_no(1,i,p+1)
    *ENDDO
  *ELSEIF,p,EQ,4
    *DO,i,1,6,1
      ent_no(1,i,p)=ent_no(div(1,p-1),i,p-1)
      ent_no(div(1,p)+2,i,p)=ent_no(1,i,1)
    *ENDDO
  *ENDIF
*ENDDO
! storing numbers of lines connecting points along profile circ.
! usage: n_store, col_lines, col_kpoints
! col_lines - col. of ent_no to store lines numbers,
! col_kpoints - col. of ent_no with points to divide the line,

```

```
! here I store lines connecting points at shank/airfoil division
n_store,7,1
```

```
! creating splines along profile circumference at intermediate and bot. level
csys,1
```

```
*DO, i, 0, 1, 1
! number of sections
flst, 2, tot_div-div(1,4), 3
! PS
*DO, j, 1, div(1,1), 1
  fitem, 2, ent_no(j, 2+i, 1)
*ENDDO
! LE
*DO, j, 1, div(1,2), 1
  fitem, 2, ent_no(j+1, 2+i, 2)
*ENDDO
! SS
*DO, j, 1, div(1,3), 1
  fitem, 2, ent_no(j, 2+i, 3)
*ENDDO
spline, p51x
n_store, 8+i, 2+i
! TE patch creation
lmaxd = LSINQR(0, 14)
k1 = KPINQR(0, 14)+1
NUMSTR, KP, k1
NUMSTR, LINE, lmaxd+1
csys, 10*4+1
XV1 = LX(ent_no(div(1,3)-1, 8+i, 3), 0.98)-KX(ent_no(div(1,3), 2+i, 3))
YV1 = LY(ent_no(div(1,3)-1, 8+i, 3), 0.98)-KY(ent_no(div(1,3), 2+i, 3))
ZV1 = LZ(ent_no(div(1,3)-1, 8+i, 3), 0.98)-KZ(ent_no(div(1,3), 2+i, 3))
XV2 = LX(ent_no(1, 8+i, 1), 0.02)-KX(ent_no(1, 2+i, 1))
YV2 = LY(ent_no(1, 8+i, 1), 0.02)-KY(ent_no(1, 2+i, 1))
ZV2 = LZ(ent_no(1, 8+i, 1), 0.02)-KZ(ent_no(1, 2+i, 1))
spline, ent_no(div(1,3), 2+i, 3), ent_no(2, 2+i, 4), ent_no(1, 2+i, 1),, ,, XV1, YV1, ZV1, XV2, YV2, ZV2
ent_no(1, 8+i, 4)=lmaxd+1
ent_no(2, 8+i, 4)=lmaxd+2
NUMSTR, LINE, lmaxd+4
*ENDDO
```

```
alls
```

```
csys, 100 ! selecting coord with X axis normal to the platform
DIST_CHECK = 0 ! DIST CHECK
! creating kps on line to check distance of TE curve from platform edge before projection
kpmax = KPINQR(0, 14)
KL, COM_NUM(1, 1, 3), 0.11, kpmax+1 ! point in the middle of platform edge
*DO, i, 0, 1, 0.02
  KL, ent_no(2, 9, 4), i, kpmax+2
  DIST_EDGE = KY(kpmax+1) - KY(kpmax+2)
  *IF, DIST_EDGE, LT, DIST_MIN, THEN
    *IF, i, LE, 0.5, THEN
      radius_check(1, 4, 2) = 0.95*radius_check(1, 4, 2)
      radius_check(1, 4, 1) = 1.05*radius_check(1, 4, 1)
    *IF, radius_check(1, 4, 2), LE, R2_min, THEN
      radius_check(1, 4, 2) = R2_min
    *ENDIF
  *ENDIF
*ENDDO
```

```

*IF, radius_check(1,4,1), GT, R1_max, THEN
  radius_check(1,4,1) = R1_max
*ENDIF
*ELSEIF,i,GT,0.5,THEN
  radius_check(1,1,2)= 0.95*radius_check(1,1,2)
  radius_check(1,1,1)= 1.05*radius_check(1,1,1)
  *IF,radius_check(1,1,2),LE,R2_min,THEN
    radius_check(1,1,2) = R2_min
  *ENDIF
  *IF, radius_check(1,1,1), GT, R1_max, THEN
    radius_check(1,1,1) = R1_max
  *ENDIF
*ENDIF
DIST_CHECK = 1
*EXIT
*ENDIF
*ENDDO
kdele, kpmx+1, kpmx+2 ! cleaning

! creating kps on line to check distance of SS curve from platform edge before projection
kpmx = KPINQR(0,14)
KL, COM_NUM(3,1,3), 0.5, kpmx+1 ! point in the middle of platform edge
*DO, j, 3,4,1
  *DO, i, 0, 1, 0.02
    KL, ent_no(j,9,3), i, kpmx+2
    DIST_EDGE = KY(kpmx+1) - KY(kpmx+2)
    rof = 0! radius no. offset to make spline convex along 3-4-5 points
    *IF, DIST_EDGE, GT, -DIST_MIN, THEN
      *IF,i, LE, 0.5, THEN
        *IF, j, EQ, 4, AND, KY(ent_no(j+1,3,3)), LT, 0.95*KY(ent_no(j,3,3)),THEN
          rof = 1
        *ENDIF
        radius_check(j+rof,3,2)= 0.95*radius_check(j+rof,3,2)
        radius_check(j+rof,3,1)= 1.05*radius_check(j+rof,3,1)
        *IF,radius_check(j+rof,3,2),LE,R2_min,THEN
          radius_check(j+rof,3,2) = R2_min
        *ENDIF
        *IF, radius_check(j+rof,3,1), GT, R1_max, THEN
          radius_check(j+rof,3,1) = R1_max
        *ENDIF
      *ELSEIF,i,GT,0.5,THEN
        *IF, j, EQ, 3, AND, KY(ent_no(j,3,3)), LT, 0.95*KY(ent_no(j-1,3,3)),THEN
          rof = -1
        *ENDIF
        radius_check(j+1+rof,3,2)= 0.95*radius_check(j+1+rof,3,2)
        radius_check(j+1+rof,3,1)= 1.05*radius_check(j+1+rof,3,1)
        *IF,radius_check(j+1,3,2),LE,R2_min,THEN
          radius_check(j+1,3,2) = R2_min
        *ENDIF
        *IF, radius_check(j+1+rof,3,1), GT, R1_max, THEN
          radius_check(j+1+rof,3,1) = R1_max
        *ENDIF
      *ENDIF
    *ENDIF
    DIST_CHECK = 1
  *EXIT
*ENDIF
*ENDDO

```

```

*ENDDDO
kdele, kpmx+1, kpmx+2 ! cleaning

! creating kps on line to check distance of LE curve from platform edge before projection
kpmx = KPINQR(0,14)
KL, COM_NUM(2,1,3), 0.5, kpmx+1 ! point in the middle of platform edge
*DO, i, 0, 1, 0.02
  KL, ent_no(2,9,2), i, kpmx+2
  DIST_EDGE = KZ(kpmx+1) - KZ(kpmx+2)
  *IF, DIST_EDGE, GT, -DIST_MIN, THEN
    *IF, i, LE, 0.5, THEN
      radius_check(1,2,2)= 0.95*radius_check(1,2,2)
      radius_check(1,2,1)= 1.05*radius_check(1,2,1)
      *IF, radius_check(1,2,2), LE, R2_min, THEN
        radius_check(1,2,2) = R2_min
      *ENDIF
      *IF, radius_check(1,2,1), GT, R1_max, THEN
        radius_check(1,2,1) = R1_max
      *ENDIF
    *ELSEIF, i, GT, 0.5, THEN
      radius_check(1,3,2)= 0.95*radius_check(1,3,2)
      radius_check(1,3,1)= 1.05*radius_check(1,3,1)
      *IF, radius_check(1,3,2), LE, R2_min, THEN
        radius_check(1,3,2) = R2_min
      *ENDIF
      *IF, radius_check(1,3,1), GT, R1_max, THEN
        radius_check(1,3,1) = R1_max
      *ENDIF
    *ENDIF
  DIST_CHECK = 1
  *EXIT
*ENDIF
*ENDDDO
kdele, kpmx+1, kpmx+2 ! cleaning

! looping model generator if R2 was changed or edges too close
*IF, DIST_CHECK, EQ, 1, THEN
  alls
  adele,all,,1
  ldele,all,,1
  kdele,all
  REPEAT_R2=REPEAT_R2+1 ! setting model repetition counter
  /COM, REPETITION NO %REPEAT_R2% LOOP R2_BEGIN CALLED BY DIST_CHECK
*ENDIF
*IF, DIST_CHECK, EQ, 1, :R2_BEGIN

*DO, i, 1, part_no, 1
  LDELETE, COM_NUM(i,1,1),,,1 ! deleting spline
*ENDDDO

/NERR, -1, 1.0E+9,
/UIS, MSGPOP, 4
alls
kdele, all
! projecting bottom spline lines onto the platform
kpmx = KPINQR (0,14)
lmaxd = LSINQR (0,14)

```

```

NUMSTR,LINE,Imaxd+1
k,kpmax+1,0.3,0,0
kgen,2,kpmax+1,,,-0.6,0,0,1
l,kpmax+1,kpmax+2 ! creating line to define drag path, id Imaxd+1
boptn,keep,no
*DO, p,1,part_no,1 ! loop for all segments
  ii=-1
  *IF,p,EQ,2,OR,p,EQ,4,THEN
    ii=1
  *ENDIF
*DO, j,1,div(1,p)+ii,1
  lineid = ent_no(j,9,p) ! selecting line
  *GET, leng, LINE, lineid, LENG ! getting line length
  *GET, K1, LINE, lineid, KP, 1
  *GET, K2, LINE, lineid, KP, 2
  lsel,s,,lineid,,
  ! generating line lmaxdnew+1 0.1 inch above line lineid
  lmaxdnew = LSINQR (0,14)
  NUMSTR,LINE,lmaxdnew+1
  lgen,2,lineid,,0.2,0,0,
  amaxd = ARINQR (0,14)
  NUMSTR,AREA,amaxd+1
  ! copying area pl_no to store it before ainp operation
  asel,a,,pl_no
  agen,2,pl_no,,0,0,0
  asel,u,,pl_no
  ! creating area to cut pl_no and define line lying at the pl_no
  adrag,lmaxdnew+1,,,,,lmaxd+1
  lmax = LSINQR(0,14)
  NUMSTR,LINE,lmax+1
  btol, defa
  ainp,amaxd+1,amaxd+2 ! if not executed, alternative process will be launched

*IF,LSEL(lmax+1), NE, 1, THEN ! starting alternative process of line projection
  asel,all
  adele, amaxd+1, amaxd+2,,1
  ldele,lmaxdnew+1,lmaxdnew+7,,1 ! in case adrag is incomplete
  step = 0.001
  bopt,keep,yes
  lmax2 = lsinqr(0,14)
  FLST, 5, NINT((leng/step)-1) , 3,, ! preparation of kp list for bspline command
  kpmaxold = KPINQR (0,14)
  *DO, i, step/leng, 1.0, step/leng
    NUMSTR, line, lmax2+1
    kpmax1 = KPINQR (0,14)
    NUMSTR, kp, kpmax1+1
    kl, lineid, i, kpmax1+1
    KMODIF,kpmax1+1,kx(kpmax1+1)+0.3,ky(kpmax1+1),kz(kpmax1+1)
    ! moving up kp on line being projected (problems at TE)
    kgen, 2, kpmax1+1,,,-0.6,0,0,1
    l, kpmax1+1, kpmax1+2 ! line intersecting pl_no to project point
    LINA, lmax2+1, pl_no ! intersection of line with area, kp is kpmax1+3
    LDELE, lmax2+1,, 0 ! cleaning
    *IF, DISTKP(kpmax1+1, kpmax1+3), GT, step, THEN
      kdele,kpmax1+1, kpmax1+2
      fitem, 5, kpmax1+3 ! list refresh
    *ELSE

```

```

    kdele,kpmax1+2, kpmax1+3
    fitem, 5, kpmax1+1 ! list refresh
*ENDIF
NUMSTR, line, lmax2+1
*ENDDO
ksel,s,,p51x
cm, kp_temp, kp
cmsel, s, kp_temp
LSEL, NONE
nk1 = NINT(kpinqr(0,13)/5)
nk2 = NINT((kpinqr(0,13)-nk1)/4)
nk3 = NINT((kpinqr(0,13)-nk1-nk2)/3)
nk4 = NINT((kpinqr(0,13)-nk1-nk2-nk3)/2)
nk5 = kpinqr(0,13)-nk1-nk2-nk3-nk4
knum = kpNext(0)
knum0 = knum
ksel,a,,k1
NUMSTR,LINE,lmax+1
FLST, 3, nk1+1 , 3,, ! preparation of kp list for bspline command
fitem, 3, K1
*DO,jj,1,nk1,1
    fitem, 3, knum
    knum = kpNext(knum)
*ENDDO
bsplin,,p51x
knum = knum-3
FLST, 3, nk2+1 , 3,, ! preparation of kp list for bspline command
*DO,jj,1,nk2+1,1
    fitem, 3, knum
    knum = kpNext(knum)
*ENDDO
bsplin,,p51x
knum = knum-3
FLST, 3, nk3+1 , 3,, ! preparation of kp list for bspline command
*DO,jj,1,nk3+1,1
    fitem, 3, knum
    knum = kpNext(knum)
*ENDDO
bsplin,,p51x
knum = knum-3
FLST, 3, nk4+1 , 3,, ! preparation of kp list for bspline command
*DO,jj,1,nk4+1,1
    fitem, 3, knum
    knum = kpNext(knum)
*ENDDO
bsplin,,p51x
knum = knum-3
FLST, 3, nk5+2 , 3,, ! preparation of kp list for bspline command
*DO,jj,1,nk5+2,1
    fitem, 3, knum
    knum = kpNext(knum)
*ENDDO
ksel,a,,k2
fitem, 3, k2
bsplin,,p51x
lssel, a,, lmax+1, lmax+4
lcomb, lmax+4, lmax+5,0

```

```

lcomb, lmax+3, lmax+4, 0
lcomb, lmax+2, lmax+3, 0
lcomb, lmax+1, lmax+2, 0
bopt, keep, no
kdele, all
/COM, Repetition of line proj. for p.%p%, seg %j%
*ENDIF
lsel, a,,,lineid
ldele, lineid,,,0
NUMSTR,LINE,lineid
lgen, 2,lmax+1
ldele, lmax+1, ,,1
bopt, keep, yes
alls
LINA, lineid, pl_no lintersecting line with area - helps to make ASBL then
bopt, keep, no
btol, defa
nummrg, kp, 0.012, 0.012, ,low
*ENDDO
*ENDDO

btol, defa
LSEL,S,,,ent_no(1,9,1) , ent_no(2,9,4),1
ASEL,S,,, pl_no
/NERR,-1,1.0E+9,
/UIS, MSGPOP, 4
*DO, i, 0.001, DIST_MIN, 0.0002 ! loop for range of btol - tolerance problems
  *IF, ARINQR(0,13), GT, 1, THEN
    /COM, ASBL executed with tolerance %i%
    *EXIT
  *ENDIF
  btol, i
  /COM, %((DIST_MIN)-i)/0.0002+1%th EXECUTION OF ASBL
  ASBL, pl_no, ALL ! breaking platform with bottom airfoil circ. lines
*ENDDO
*DO, p, 1, part_no,1
  *DO, i,1,div(1,p)+1,1
  *GET, ent_no(i,3,p), LINE, ent_no(i,6,p), KP, 2
  *ENDDO
*ENDDO
n_store, 9, 3
KSEL,s,,,ent_no(1,3,1)
LSLK
LSEL,u,,,ent_no(1,6,1)
LSEL,u,,,ent_no(1,9,1) ! in case number of bottom LE 2nd line is changed
LSEL,u,,,ent_no(1,9,4)
*GET,ent_no(2,9,4),LINE,0,NUM,MAX
KSLI
LSLK
LSEL,u,,,ent_no(1,6,1)
LSEL,u,,,ent_no(1,9,1) ! in case number of bottom LE 2nd line is changed
LSEL,u,,,ent_no(2,9,4)
*GET,ent_no(1,9,4),LINE,0,NUM,MAX

ALLS
nummrg,kp,0.005,0.005
btol, defa

```

```

pl_no = ARINQR(0,14)! indexing platform area
adele, pl_no-1 ,,0 ! deleting inner area
! storing numbers of spline segments at lying at the platform
AESIZE,all,0.1 ! first esize on all areas, some will be redefined then

```

```

! loop for upper and lower area band
*DO, j, 0, 1, 1
! creating areas around airfoil, loop for PS and SS
*DO, p,1,part_no-1,2
  i_loop = 1
  *DO, i,1,div(1,p)-1,1
    first_line=ent_no(i,5+j,p)
    second_line=ent_no(i+1,5+j,p)
    lower_line=ent_no(i,7+j,p)
    upper_line=ent_no(i,8+j,p)
    amaxd = ARINQR (0,14)
    NUMSTR, AREA, amaxd+1
    AL,first_line,lower_line,second_line,upper_line
    *IF, _RETURN, EQ, 0, THEN
    *ENDIF
    AESIZE,amaxd+1,(1-j)*0.1+j*0.035
    ent_no(i_loop,10+j,p) = amaxd+1
    i_loop = i_loop+1
  *ENDDO
*ENDDO
*DO, p,2,part_no,2
  i_loop = 1
  *DO, i,1,div(1,p)+1,1
    first_line=ent_no(i,5+j,p)
    second_line=ent_no(i+1,5+j,p)
    lower_line=ent_no(i,7+j,p)
    upper_line=ent_no(i,8+j,p)
    amaxd = ARINQR (0,14)
    NUMSTR, AREA, amaxd+1
    AL,first_line,lower_line,second_line,upper_line
    AESIZE,amaxd+1,0.02
    ent_no(i_loop,10+j,p) = amaxd+1
    i_loop = i_loop+1
  *ENDDO
*ENDDO
*ENDDO
! preparing components for pressure mapping
asel,none
asel,a,,ent_no(2,10,4)
asel,a,,ent_no(1,10,1),ent_no(6,10,1)
asel,a,,ent_no(1,10,2)
cm,cfdpres_area_ps,area
asel,none
asel,a,,ent_no(2,10,2)
asel,a,,ent_no(1,10,3),ent_no(7,10,3)
asel,a,,ent_no(1,10,4)
cm,cfdpres_area_ss,area
! preparing areas for stres evaluation
asel,none
asel,s,,ent_no(1,11,1),ent_no(6,11,1)
asel,a,,ent_no(1,10,1),ent_no(6,10,1)
cm,ps_areas,area

```



```

asel,s,,,ent_no(1,11,2),ent_no(2,11,2)
asel,a,,,ent_no(1,10,2),ent_no(2,10,2)
cm,le_areas,area
asel,s,,,ent_no(1,11,3),ent_no(7,11,3)
asel,a,,,ent_no(1,10,3),ent_no(7,10,3)
cm,ss_areas,area
asel,s,,,ent_no(1,11,4),ent_no(2,11,4)
asel,a,,,ent_no(1,10,4),ent_no(2,10,4)
cm,te_areas,area
! reading rest of model into database
alls

! model sizing
LESIZE, ent_no(1,8,4),,, 2 ! divisions of TE lines set to 2 to avoid
LESIZE, ent_no(2,8,4),,, 2 ! stress peaks
CDREAD,COMB,'lpt3a_interface','cdb',' ', , , ! READING INTERFACE
alls
NUMMRG,KP,0.01,0.01,,low
NUMSTR, ELEM, 30000
NUMSTR, NODE, 30000
va,all ! volume created
VATT, 2, , 1, 0
SHPP, OFF, ALL
VMESH,1 ! meshing volume
esel,s, type,,1
*IF, elmiqr(0,13), EQ, 0, THEN
  SMRT,6
  MSHAPE,1,3D
  MSHKEY,0
  VMESH, 1
*ENDIF
alls
NOFFSET, ELEM
NOFFSET, NODE
CDREAD,DB,'lpt3p3a_coarse_all_model_red1','cdb',' ', , , !reading rest of the model
alls
! PARAMETRIC MODEL LOADING
csys,1
! selecting components and mapping pressures
cmsel,s,cfdpres_area_ps
ALLSEL,BELOW,AREA
NSLA,S,1
mapper,'CFDPRES_PS'
! calling macro
cmsel,s,cfdpres_area_ss
ALLSEL,BELOW,AREA
NSLA,S,1
mapper,'CFDPRES_SS'
! mapping temperatures
alls, below, volu
asel,s,,,14
asel,a,,,360
nsla,s,1
esln,s
cm, elem_to_reject, elem

alls, below, volu

```

```

cmsel, u, elem_to_reject
nsle, s
cm, shank_nodes, node ! saving component for further results request
cm, shank_elems, elem ! saving component for further results request
! saving nodal components for stress evaluation
cmsel,s,ps_areas
ALLSEL,BELOW,AREA
NSLA,S,1
cm,ps_nodes, node
cmsel,s,le_areas
ALLSEL,BELOW,AREA
NSLA,S,1
cm,le_nodes, node
cmsel,s,ss_areas
ALLSEL,BELOW,AREA
NSLA,S,1
cm, ss_nodes, node
ALLSEL,BELOW,AREA
cmsel,s,te_areas
NSLA,S,1
cm,te_nodes, node

```

```

ALLS
ALLS, below, volu
nnum = NDINQR (0,13)
*GET,nmax,node,,num,max
*GET,nmin,node,,num,min
n = nmin
avg_temp=0
*ABSET,Blade_Temp_Application_Progress,BOTH
*DO,i,1,nnum,1
  BF,n,TEMP,temp_R_blade(NX(n))
  n=NDNEXT(n)
  *ABCHECK,(i/nnum)*100,
*ENDDO
*ABFINI
! END OF PARAMETRIC MODEL LOADING

```

```

!setting analysis
alls
/SOLU
/TITLE,LMS100 LPT3P3A TOTAL@3600RPM CHROM %u% GENE %G%
TIME,3
EQLV,SPARSE ! sparse solver
OUTRES, ALL, NONE
OUTRES, STRS, LAST, shank_elems ! setting output request - only stress
OUTRES, NLOAD, LAST ! setting output request - only nforces
IRLF, -1 ! pre-calculate mass
SOLVE
IRLIST ! mass properties retrieve
*GET,mass,elem,,mtot,x ! mass stored as "mass" parameter
mass = mass - 0.0194659 !!! all model - mass of unparam. parts (red1)

```

```

! postprocessing
/POST1

```

```

! max stress

```

```

cmsel,s,ps_nodes
cmsel,a,le_nodes
cmsel,a,ss_nodes
cmsel,a,te_nodes
NSORT, S, 1, 1, 0,
*GET, max_1st_princ, SORT, 0, MAX ! max first principal

alls
*IF, max_1st_princ, EQ, 0, THEN
  /COPY, model_G%G%_CH%u%,err,,failed_CH%u%_G%G%, err
  mass = 1
  max_1st_princ = 100000
*ENDIF

/PREP7
! substitution of original chromosome with modified one
*DO, p, 1, part_no, 1
  j_loop=1
  *DO, i, 1, div(1,p), 1
    *IF, key(i,p), EQ, 1, THEN
      chromosome (1,j_loop,p)=radius_check(i,p,1)
      chromosome (2,j_loop,p)=radius_check(i,p,2)
      *IF, p, EQ, 2, OR, p, EQ, 4, THEN
        chromosome(1,2,p)=radius_check(i,p,1) ! values of R1
        chromosome(2,2,p)=radius_check(i,p,2)! val.of R2
      *ENDIF
    j_loop=j_loop+1
  *ENDIF
*ENDDO
*ENDDO
FINI
! cleaning
AMAXD =
BVY =
BVZ =
CUT_LINEID=
EVY=
EVZ=
FILMAX =
FIRST_LINE=
H =
I =
II =
I_LOOP =
J =
J_LOOP =
KP1 =
KP2 =
KP3 =
KP5 =
KPMAX =
LINEID =
LMAXD =
LMAXDNEW=
LNUM_NEW=
LOWER_LINE=
N =

```

```
NMAX =
NMIN =
NNUM =
NPRES =
NTEMP =
P =
PL_NO =
R1 =
R2 =
REPEAT =
ROT_ANG =
SECOND_LINE=
UPPER_LINE=
Z =
cfdpres_ps =
cfdpres_ss =
cfdxyz =
temp_r_blade =
temp_r_disc =
presval =
presval_ =
_L3=
_L4=
_L5=
_L6=
```

```
parsav,all,temp_params,par
FINISH
/CLEAR,START
```

```
parres,,temp_params,par
/SYS, del *.err
/SYS, del *.log
/SYS, del *.emat
/SYS, del *.esav
/SYS, del *.mntr
/SYS, del *.stat
/SYS, del *.full
/SYS, del *.PCS
/SYS, del *.par
/SYS, del *.rst
```

3. Procedura MAPPER

```
/nopr
/NERR,-1,2E+31
```

```
presval =
presval_ =
*set,presval_,arg1
```

```
csys,1
pmapxyz=
pmapval=
pmapxyz2=
pmapval2=
nlis =
nmask =
```

```

*get,nnum,node,,count
*get,nmax,node,,num,max
*dim,pmapxyz,array,nnum,3
*dim,pmapxyz2,array,nmax,3
*dim,pmapval,array,nnum,1
*dim,pmapval2,array,nmax,1
*dim,nlis,array,nmax,1
*dim,nmask,array,nmax,1
*dim,presval,array,
*vget,nmask(1),node,1,nsel
*vfill,nlis(1,1),ramp,1,1

nd=0
*DO,vv,1,nnum,1
nd=ndnext(nd)
!pmapxyz2(nd,1) = nx(nd)
!pmapxyz2(nd,2) = ny(nd)
pmapxyz2(nd,3) = nz(nd)
*ENDDO
*vget,pmapxyz2(1,1),node,,loc,x
*vget,pmapxyz2(1,2),node,,loc,y

*vmask,nmask(1)
*vfun,pmapxyz(1,1),comp,pmapxyz2(1,1)
*vmask,nmask(1)
*vfun,pmapxyz(1,2),comp,pmapxyz2(1,2)
*vmask,nmask(1)
*vfun,pmapxyz(1,3),comp,pmapxyz2(1,3)

*moper,pmapval(1),pmapxyz(1,1),MAP,presval_,cfdxyz(1,1),2,.5,0

esln
*vmask,nmask(1)
*vfun,nlis(1),comp,nlis(1)
*voper,pmapval2(1),pmapval(1),SCAT,nlis(1)
esel,r,type,,1
sffun,pres,pmapval2(1)
sf,all,pres
sffun

!cleaning
pmapxyz =
pmapxyz2 =
pmapval =
pmapval2 =
nlis=
nmask=
/go

```

4. Procedura NSTORE

```

! macro to store numbers of splines along profile section
! macro works for 4 divisions (PS, LE, SS, TE)
! matrix div(,) and ent_no(,) must be defined before
col_to_store=arg1
point_col=arg2

```

```

! storing upper connecting line
*DO,p,1,4,1
  ksel,s,,ent_no(1,point_col,p)
  ksel,a,,ent_no(2,point_col,p)
  LSLK,s,1
  *GET,ent_no(1,col_to_store,p),LINE,0,NUM,MIN
  *DO,z,2,div(1,p)-2,1
    ksel,s,,ent_no(z,point_col,p)
    ksel,a,,ent_no(z+1,point_col,p)
    LSLK,s,1
    *GET,ent_no(z,col_to_store,p),LINE,0,NUM,MAX
  *ENDDO
  ksel,s,,ent_no(z+1,point_col,p)
  LSLK,s,0
  lsel,u,,ent_no(z,col_to_store,p)
  *GET,ent_no(z+1,col_to_store,p),LINE,0,NUM,MAX
  *IF,p,EQ,4,THEN
    ksel,s,,ent_no(z+1,point_col,p)
    LSLK,s,0
    *GET,ent_no(z,col_to_store,p),LINE,0,NUM,MAX
    ent_no(z+1,col_to_store,p)=0
  *ELSEIF,p,EQ,2,THEN
    ksel,s,,ent_no(z+1,point_col,p)
    LSLK,s,0
    lsel,u,,ent_no(z+1,5,p)
    lsel,u,,ent_no(z+1,6,p)
    *GET,ent_no(z,col_to_store,p),LINE,0,NUM,MIN
    ent_no(z+1,col_to_store,p)=0
  *ENDIF
*ENDDO

col_to_store=
point_col=
line_col=

```

DODATEK 2 - STRUKTURA PLIKÓW WEJŚCIA-WYJŚCIA PROGRAMU OPTYMALIZACYJNEGO

Poniższa tabela zawiera spis plików wykorzystywanych w trakcie działania programu optymalizacyjnego.

Nazwa pliku	Typ	Opis
master.mac	skrypt <i>APDL</i>	Program zarządzający procesem optymalizacji
mgen.mac	skrypt <i>APDL</i>	Program przeprowadzający generację modelu <i>MES</i> i analizę
mapper.mac	skrypt <i>APDL</i>	Program przeprowadzający mapowanie ciśnień na modelu <i>MES</i>
nstore.mac	skrypt <i>APDL</i>	Program pomocniczy do przygotowania modelu geometrycznego
shank_iges_62.iges	plik tekstowy <i>IGES</i>	Plik zawierający dane wejściowe do generacji modelu geometrycznego fragmentu łopatkki
lptp3a_interface.cdb	archiwum tekstowe <i>Ansys</i>	Plik tekstowy zawierający siatkę <i>MES</i> na powierzchniach wspólnych części optymalizowanej i pozostałej części modelu <i>MES</i>
lpt3p3a_coarse_model.cdb	archiwum tekstowe <i>Ansys</i>	Plik zawierający model <i>MES</i> rotora turbiny oprócz części optymalizowanej
generation_N.001	plik tekstowy	Plik tekstowy zawierający chromosomy pokolenia <i>N</i> przed oceną funkcji przystosowania
generation_N.003	plik tekstowy	Plik tekstowy zawierający chromosomy pokolenia <i>N</i> , wartości masy i maksymalnego naprężenia głównego na optymalizowanej części modelu

Poniższy schemat blokowy przedstawia operacje wejścia – wyjścia wykonywane przez program Ansys w trakcie przeprowadzania procesu optymalizacji.

