Transitional Flow Modeling and Application to High-Lift Multi-Element Airfoil Configurations

Andreas Krumbein∗

Deutsche Forschungsanstalt für Luft- und Raumfahrt, German Aerospace Center, D-38108 Brunswick, Germany

To enhance its capabilities to handle flows with transition, a Reynolds averaged Navier–Stokes solver has been extended with regard to the modeling of transitional flow regions based on transition length models and the intermittency function. Because the full coupling of the solver to an e^N-method that predicts the locations of transition onset has not yet been completed, the points of laminar separation are supposed to represent the transition locations in a first step. A method and an algorithm for detecting the laminar separation points are derived, and the intermittency function and two transition length models are implemented and validated for a selected high-lift multi-element test case. The background of the implementation work and the testing of the functionalities of the algorithms are focused on. Details of the implementation, which are consequences of an underlying transition prediction strategy, are outlined. The testing is described and then documented.

Nomenclature

c_d	=	drag coefficient
c_f	=	skin-friction coefficient
c_l	=	lift coefficient
c_p	=	pressure coefficient
FLG_{lt}	=	laminar–turbulent flag for the code internal eddy viscosity μ_t^{code}
k	=	turbulent kinetic energy
l_{tr}	=	transition length
N_e	=	maximum number of the geometry elements
n_e	=	number of the current geometry element
P_F	=	field point of the computational grid
P_S	=	surface point of the computational grid
$P^{(1)}$	=	point of the computational grid on the first grid line
Re_l	=	Reynolds number based on the length l, that is, $U_e l / \nu_e$
$\boldsymbol{r}$	=	position vector
s	=	arc length starting at the stagnation point
U	=	tangential flow velocity
U_e	=	tangential flow velocity at the boundary-layer edge
$\boldsymbol{v}_t$	=	tangential velocity vector
x	=	longitudinal coordinate of the configuration
γ	=	intermittency function, $\gamma(\zeta) = 1 - \exp(-0.412\xi^2)$
δ	=	boundary layer thickness, $\delta(\zeta)$
$\delta*$	=	displacement thickness,

$$\delta*(\zeta) = \int_0^{\delta(\zeta)} \left(1 - \frac{(\rho U)(y)}{\rho_e U_e}\right) dy$$

ζ	=	length coordinate along the streamline of the boundary-layer edge
λ	=	constant of the intermittency function
μ_t	=	eddy viscosity
ν_e	=	kinematic viscosity at the boundary-layer edge
ξ	=	variable of the intermittency function, $\xi = (\zeta - \zeta_{tr}^{beg})/\lambda$
ρ	=	density
ρ_e	=	density at the boundary-layer edge

ω	=	specific turbulent dissipation rate

Subscripts

beg	=	beginning of the transitional flow region
code	=	inside the code
comp	=	computational
e	=	element or edge of the boundary layer
elem	=	element, that is, slat, main, flap
end	=	end of the transitional flow region
exp	=	experimental
F	=	field
ft	=	fully turbulent
i	=	counter of the surface points
init	=	initial
low	=	lower side
max	=	maximum
nose	=	nose of the airfoil
nst	=	nearest
nu	=	nonunique
old	=	old
q	=	placemarker for upper side or lower side
S	=	surface
sep	=	at the separation point
stag	=	at the stagnation point
t	=	turbulent
tr	=	at transition onset
trail	=	trailing edge
upp	=	upper side

I. Introduction

THE modeling of laminar–turbulent transition in Reynolds averaged Navier–Stokes (RANS) solvers is a necessary requirement for the computation of flows over airfoils and wings in the aerospace industry because it is not possible to obtain quantitatively correct results if the laminar–turbulent transition is not taken into account. For the design process of wings in industry, there exists the demand for a RANS-based computational fluid dynamics tool that is able to handle flows automatically and autonomously with laminar–turbulent transition.

The first steps toward the setup of such a tool were taken in Ref. 1, where a RANS solver and an e^N-method[2,3] based on linear stability theory and the parallel flow assumption were applied and in Ref. 4, where a RANS solver, a laminar boundary-layer method,[5] and an e^N-method were coupled. There, the boundary-layer method was used to produce highly accurate laminar, viscous layer data to be analyzed by a linear stability code. Hence, the very expensive grid

adaptation necessary to produce accurate viscous layer data directly from the Navier–Stokes grid was avoided. The use of an e^N-database method[6] results in a coupled program system that is able to handle automatically transition prediction. Alternative approaches using a transition closure model or a transition/turbulence model directly incorporated into the RANS solver are documented in Refs. 7–9.

At the Deutsche Forschungsanstalt für Luft- und Raumfahrt, German Aerospace Center, the structured RANS code FLOWer[10] is used together with the laminar boundary-layer method of Ref. 5 and the e^N-database method of Ref. 6. The laminar boundary-layer method and the e^N-database method form a so called transition prediction module that is coupled to the RANS solver and that interacts with the RANS solver during the computation.[11,12] Presently, the transition prediction module of FLOWer can be applied to two-dimensional one-element configurations.

The description of transitional flow regions in FLOWer is done by the application of point transition, which means that turbulence quantities, which are suppressed in the laminar part of the flow, suddenly become active at the location of transition onset. This procedure results in a sudden change of the flow quantities in this area. Because of the effects of numerical dissipation, a small transitionallike flow region is generated artificially in a computation without physical transition modeling. However, with the present procedure, the sudden change of the flow quantities is often strong enough to prevent the convergence of the iterative transition prediction process.[13] In addition, the application of point transition generates a strong upstream influence so that the transitionallike flow region starts considerably upstream of the transition location. In two-dimensional airfoil flows, an upstream influence up to 10% of the chord length of the airfoil can be observed.

The extension of the FLOWer code to overcome these two limitations, the restricted application of the transition prediction module to two-dimensional one-element configurations and the application of point transition instead of the physical modeling of transitional flow, is currently under way. The coupled program system is extended to two-dimensional multi-element configurations, and physical models for the computation of transitional flow regions are introduced. The extended code is applied to two-dimensional high-lift systems. The extensions are performed in two steps. First, a generalized infrastructure in the FLOWer code with respect to the transition prediction module is built up and tested, that is, the code is changed in such a way that the transition prediction module can be activated in the future for arbitrary multi-element configurations independent of the block topology and the grid structure. Second, the transition prediction module is coupled to the generalized infrastructure. In the framework of this paper, the first extension step is documented. Thus, this paper has the character of a progress report.

At the time that this paper was written, the transition prediction module had not been coupled to the extended FLOWer code; the locations of laminar separation determined by the FLOWer code are supposed to represent the laminar–turbulent transition locations in a first step. In many cases, this assumption leads to a good approximation of the real transition point, particularly for low Reynolds number airfoil flows, when transition does not occur before the laminar boundary layer separates. Because the term transition prediction in this restricted context is not strictly accurate, transition determination is used for the handling of points where transition is fixed throughout this paper. The main objective of the performed work presented is to supply a reliably working infrastructure in a RANS code so that the RANS code together with the transition prediction module described earlier can be used in the future for the computation of two-dimensional multi-element high-lift systems of aircraft including transitional flow regions. To achieve these objectives in the FLOWer code, the steps that have to be taken are 1) implement the capability to fix transition at the point of laminar separation in the RANS computation and 2) implement the capability to compute transitional flow regions. These two issues are the subjects of this paper.

A method and an algorithm for detecting the laminar separation points are derived and implemented into the FLOWer code. For each element of a high-lift configuration on the upper and lower side, the laminar separation point will be detected, and the transition fixed there. The intermittency function and two transition length models are implemented and validated in a variety of test computations for a selected high-lift multi-element test case. This paper focuses on the background of the implementation work and the testing of the functionalities of the algorithms. Details of the implementation, which are consequences of the underlying transition prediction strategy[4–6,11,12] are outlined. The testing is described and documented by the results of the transition determination procedure and of the transition length models.

II. Implementation

FLOWer is a three-dimensional, compressible RANS code for steady or unsteady flow problems and uses structured body-fitted multiblock meshes. The code is based on a finite volume method and a cell-vertex spatial discretization scheme and uses an explicit Runge–Kutta time integration scheme with multigrid acceleration. The influence of turbulence is taken into account by eddy viscosity turbulence models according to the Boussinesq approximation. The transition handling is independent of the block topology of the computational grid and of the grid structure (structured, unstructured, or hybrid grid).[11] The implementation consists of three thematically different areas: the handling of the surface points of the configuration in such a way as to build up a method that is independent of the block topology and the grid structure, the detection of laminar separation, and the generation of transitional flow regions in the code.

A. Handling of Surface Points

The complete coupled program system that will be used for transition prediction with the RANS solver FLOWer consists of the RANS solver itself,[10] a laminar boundary-layer method for swept, tapered wings,[5] and a transition prediction method, which is provided with all necessary data, for example, boundary-layer parameters, by the laminar boundary-layer method. Besides a number of empirical transition criteria, the most general transition prediction method that is available in the FLOWer transition prediction module is an e^N-database method.[6]

The RANS solver communicates the surface pressure distribution of the configuration as input data to the laminar boundary-layer method, the laminar boundary-layer method computes all of the boundary-layer parameters that are needed for the transition prediction method, the transition prediction method determines new transition locations that are given back to the RANS solver. This coupled structure results in an iteration procedure for the transition locations within the iteration of the RANS equations. Because a boundary-layer method is an essential part of the coupled program system, there are a number of conditions that must be fulfilled by the input of the surface points of the configuration to the boundary-layer method during the iteration process.

1) Each element of a multi-element configuration must be divided into an upper and a lower side. The point that defines the division is the stagnation point on the airfoil surface.

2) The surface points on the upper and lower sides must be ordered. The sequences of points start at the stagnation point and end at the trailing-edge points of the upper or lower side, respectively.

3) The ordered sequences of points must not contain topologically singular surface grid points. This may happen in the case that the surface of an airfoil is contained in more than one block. The surface points on the block cuts are topologically nonunique. Each physical surface point must exist only once in the ordered sequence of points.

To fulfill these conditions the following steps can be taken.

First, the identification values of the surface points of each element are stored in an additional array. Because these identification values are integer values, the i index, j index, and k index and the block number in case of a structured solver or the point number in case of an unstructured solver, integer arrays are sufficient for this purpose. All of the following steps are done using these new arrays. The identification values in the new arrays are used to address the surface grid points and to have access to the flow variables at the

surface points. How grid points are assigned to specific elements of a multi-element surface is described in detail in Ref. 11.

Second, together with the additional arrays for the identification values, one needs another integer array for individual information about each surface point of each element. This array contains information about the single surface point and the ordered sequence of the surface points. This array is called the upper–lower flag (ULFLG) because it contains the upper–lower status of the surface point. Let $P_S(n_e)$ be a surface point of element n_e, $1 \leq n_e \leq N_e$, where N_e is the maximum number of configuration elements and ULFLG $[P_S(n_e)]$ its upper–lower flag.

First, the ULFLG for each surface point is initialized

$$\text{ULFLG}[P_S(n_e)]^{\text{init}} = 7 \tag{1}$$

Where the value 7 is just an arbitrary value for the initialization.

Third, when the run of the RANS code starts, during the initialization phase, an a priori division of each element into an upper and lower side is performed. The division is defined by the geometric nose point of each element and by the rearmost trailing-edge point of the element. The rearmost trailing-edge point is the one that has the greatest coordinate value with respect to the chordwise direction of the element. The geometric nose point is the surface point that has the greatest distance from the rearmost trailing-edge point.

The division into upper and lower sides is easy for airfoils with two trailing edges. In this case, the user can enter the coordinates of two points that define a straight line that divides the airfoil into an upper and lower side. One point may be any point on a line between upper and lower trailing edge, and the other point may be near the geometric nose point, for example, or the geometric nose point itself. Because this way of partitioning is not applicable to airfoils with one trailing edge, the division should be done automatically. In this case, it is necessary to approximate the mean line of the airfoil. As a very weak condition for the quality of the approximation of the mean line, it must be ensured that all of the points of the polygonial line that defines the mean line are located within the airfoil contour.

The geometric nose point belongs to both sides. For this point,

$$\text{ULFLG}[P_{S,\text{nose}}(n_e)] = 0 \tag{2}$$

A surface point on the upper side of the airfoil gets

$$\text{ULFLG}[P_{S,\text{upp}}(n_e)] = 1 \tag{3}$$

A surface point on the lower side of the airfoil gets

$$\text{ULFLG}[P_{S,\text{low}}(n_e)] = -1 \tag{4}$$

After this step there exists a geometrical division of each element into upper and lower sides.

Fourth, all pairs of points that are topologically nonunique are identified. One point of each pair and all trailing-edge points get

$$\text{ULFLG}[P_{S,\text{nu}}(n_e)] = 2 \times \text{sgn}\{\text{ULFLG}[P_{S,\text{nu}}(n_e)]\} \tag{5}$$

During the transition determination procedure surface points with

$$|\text{ULFLG}[P_S(n_e)]| = 2 \tag{6}$$

are not taken into account.

Fifth, for each element, the surface points first on the upper side and then on the lower side are ordered according to their Euclidean distance from the geometric nose point. The ordering is performed within the array ULFLG(n_e) and within the new arrays for the identification values. After this step, the surface points of upper and lower sides of each element are in an ordered sequence along the airfoil contour from a geometrical point of view.

Sixth, during the transient phase of the RANS computation the stagnation point is determined. The stagnation point defines the aerodynamical division of each airfoil into upper and lower sides, and its location changes the number of points that belong either to the upper or lower side.

Each time when the stagnation point is determined the corresponding surface point gets

$$\text{ULFLG}[P_{S,\text{stag}}(n_e)] = 0 \tag{7}$$

The stagnation point has moved either into the area of the former upper side or into the area of the former lower side. For all of the surface points that are located between the two points that are marked with ULFLG$[P_S(n_e)] = 0$, the algebraic sign must be reversed and the ULFLG of the old stagnation point must be given the correct value. In the case that the new stagnation point has moved into the area of the former lower side

$$\text{ULFLG}[P_{S,\text{stag}}(n_e)]^{\text{old}} = 1 \tag{8}$$

is set, in the case that the new stagnation point has moved into the area of the former upper side

$$\text{ULFLG}[P_{S,\text{stag}}(n_e)]^{\text{old}} = -1 \tag{9}$$

is set.

Finally, for each element, the surface points between the old and the new stagnation points are ordered according to their arc length along the airfoil contour measured from the new stagnation point. The ordering is again performed within the array ULFLG(n_e) and within the new arrays for the identification values. After this step, the surface points of the upper and lower sides of each element are in an ordered sequence along the airfoil contour from an aerodynamic point of view.

The described handling of surface points using new arrays for the identification values and the additional array ULFLG(n_e), decouples the point-ordering issues and the characterization of the surface points from the code internal data structure of the computational grid. The algorithm can be looked upon as a module whose inputs are the data stored at the surface grid points of the computational grid. Inside the module exists a pure structured single-zone data structure for each side of an airfoil. When the algorithm has ended, all necessary information is contained in the additional integer arrays. When these arrays are used, every surface point of the computational grid with its corresponding flow quantities can be addressed in the order that is stored in the additional arrays. Thus, the algorithm is independent of the grid structure and the grid topology.

B. Detection of Laminar Separation

The algorithm for the detection of laminar separation consists of two parts, the determination of the stagnation point and the determination of separation points. The algorithm itself does not make a distinction between laminar and nonlaminar separation points, it simply detects separation points. However, from the global strategy for the transition determination iteration, it is clear that when the algorithm interprets a separation point as a transition point it must always be a laminar separation point. This is the case only when a separation point is located upstream of the actual transition point, that is, the transition point that is currently dividing the airfoil side into a laminar and a nonlaminar part at this stage of the transition location iteration. The stagnation point is defined as the surface grid point where the maximum c_p value is found,

$$c_p[P_{S,\text{stag}}(n_e)] = \max_{P_S}\{c_p[P_S(n_e)]\} \tag{10}$$

After the division of the current airfoil into upper and lower side has been done, the algorithm looks for a separation point, first on the upper side, then on the lower side. The search starts at the stagnation point for each side of the airfoil and ends at the corresponding trailing-edge point. The existence of a separation point $\boldsymbol{r}_{\text{sep}}$ is defined by the following condition, with the position counter i, that counts the surface points from the stagnation point to the trailing-edge point, the tangential velocity vector $\boldsymbol{v}_t$ at the first grid point $P^{(1)}$ apart from the solid wall of the airfoil, the direction vector $\Delta\boldsymbol{r}_{i+1,i}$ pointing from point $P_{i+1}^{(1)}$ to point $P_i^{(1)}$, $\Delta\boldsymbol{r}_{i+1,i} = \boldsymbol{r}(P_{i+1}^{(1)}) - \boldsymbol{r}(P_i^{(1)})$, and . indicating the scalar product between vectors:

$$\text{sgn}[\boldsymbol{v}_t(P_i^{(1)}) \cdot \Delta\boldsymbol{r}_{i+1,i}] = \text{sgn}[\boldsymbol{v}_t(P_{i+1}^{(1)}) \cdot \Delta\boldsymbol{r}_{i+1,i}] \tag{11}$$

which implies that there does not exist a separation point r_{sep} in the closed interval between the surface points $P_{S,i}$ and $P_{S,i+1}$. Also,

$$\operatorname{sgn}\left[\nu_t\left(P_i^{(1)}\right)\right] \cdot \Delta r_{i+1,i} \neq \operatorname{sgn}\left[\nu_t\left(P_{i+1}^{(1)}\right)\right] \cdot \Delta r_{i+1,i} \quad (12)$$

which implies that there exists a separation point r_{sep} in the closed interval between the surface points $P_{S,i}$ and $P_{S,i+1}$. In the case that the detected separation point r_{sep} is located upstream of the transition point currently used, r_{sep} is a laminar separation point and surface point $P_{S,i}$ is used as new transition location on the corresponding side of the airfoil.

C.   Generation of Transitional Flow Regions

In the case that a new transition location has been determined, the laminar, transitional, and turbulent flow regions must be generated anew within the computational grid. The generation of the different regions is done by the setting of a real value flag FLG_{lt} at each point of the computational grid that is multiplied with the value of the eddy viscosity μ_t, which is computed for every point in the flow field. FLG_{lt} is applied in the following way for all of the points on solid walls of the configuration:

$$\mu_t^{code}(P_S) = FLG_{lt}(P_S)\mu_t(P_S) \quad (13)$$

with $FLG_{lt}(P_S) = 0.0$ for a laminar surface point, $FLG_{lt}(P_S) = 1.0$ for a turbulent surface point, and $FLG_{lt}(P_S) = \gamma(P_S)$ for a transitional surface point, where $\gamma(P_S)$ is the value of the intermittency function γ at the surface point P_S.

The laminar length on upper or lower sides of an airfoil is defined by the interval between the stagnation point and the transition point on the side q, with $q = $ upp, low indicating either the upper or the lower side of the airfoil, $0 \leq s_q \leq s_{q,tr}^{beg}$, where s_q is the arc length on the side q starting at the stagnation point. The turbulent length is defined by the interval between the ending point of the transitional region and the trailing-edge point on side q, $s_{q,tr}^{end} \leq s_q \leq s_q^{trail}$, and the transitional length is the interval between the transition point and the ending point of the transitional region, $s_{q,tr}^{beg} < s_q < s_{q,tr}^{end}$. The different intervals are shown in Fig. 1.

Here γ is expressed as

$$\gamma(x) = 1 - \exp(-0.412\xi^2) \quad (14)$$

with

$$\xi = \left(x - x_{tr}^{beg}\right)/\lambda \quad (15)$$

according to Ref. 14, where x is the longitudinal coordinate of a flat plate with its origin located in the upstream end of the plate and λ is a measure of the extent of the transitional region. According to Ref. 13, the ending point of the transitional region x_{tr}^{end} can be defined as

$$x_{tr}^{end} = x(\gamma = 0.99) \quad (16)$$

which yields

$$\lambda = \left(\frac{x_{tr}^{end} - x_{tr}^{beg}}{3.36}\right) \quad (17)$$

For the determination of the extent of the transitional region, the transition length $l_{tr} = x_{tr}^{end} - x_{tr}^{beg}$, formulas from Ref. 15,

$$Re_{l_{tr}} = 5.2\left(Re_{x_{tr}^{beg}}\right)^{\frac{3}{4}} \quad (18)$$
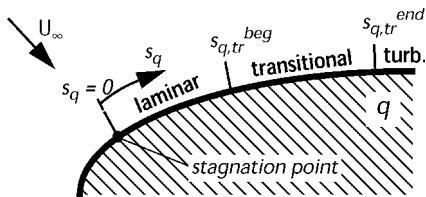
for flows without pressure gradient, and

$$Re_{l_{tr}} = 2.3\left(Re_{\delta_{tr}^{* beg}}\right)^{\frac{3}{2}} \quad (19)$$

for flows with pressure gradient, are applied, as is recommended in Ref. 13 for flows in which transition does not occur before laminar separation, which is the case for all computations whose results are presented in this paper due to the underlying method of determining the transition points by fixing transition at the locations of laminar separation. Here

$$\delta*(x) = \int_0^{\delta(x)} \left[1 - \frac{(\rho U)(y)}{\rho_e U_e}\right] dy \quad (20)$$

The thickness of the laminar boundary layer δ is evaluated according to a procedure described in Ref. 4. Thus, $\delta*$ and ρ_e and U_e, the values of the density and the tangential flow velocity at the boundary-layer edge, can be determined.

For the implementation in the RANS solver, the x coordinate in the formulas is replaced by the arc length s. Here $s_{q,tr}^{beg}$ is given by the location of a laminar separation point and $s_{q,tr}^{end}$ is determined by formula a, based on Eq. (19), or formula b, based on Eq. (18), respectively. Formula a reads

$$s_{q,tr}^{end(a)} = 2.3\sqrt{(U_e/\nu_e)_{s_{q,tr}^{beg}}}\left(\delta*_{s_{q,tr}^{beg}}\right)^{\frac{3}{2}} + s_{q,tr}^{beg} \quad (21)$$

Formula b reads

$$s_{q,tr}^{end(b)} = 5.2\left[(U_e/\nu_e)_{s_{q,tr}^{beg}}\right]^{-\frac{1}{4}}\left(s_{q,tr}^{beg}\right)^{\frac{3}{4}} + s_{q,tr}^{beg} \quad (22)$$

and the intermittency function γ is applied in the form

$$\gamma(s_q) = 1 - \exp\left[-0.412\left(3.36\frac{s_q - s_{q,tr}^{beg}}{s_{q,tr}^{end} - s_{q,tr}^{beg}}\right)^2\right] \quad (23)$$

The computational tests will show that only formula b, based on flat plate theory, will yield results that compare well enough with experimental findings.

After all of the surface points on upper and lower side of an airfoil have been assigned to either the corresponding laminar, turbulent, or transitional interval, the field points, all points apart from the solid walls, are treated in the following way.[11]

Within a limiting wall normal distance that can be adjusted by the user of the code, every field point P_F assumes the flag value of the surface point P_S^{nst} that is located nearest to P_F,

$$FLG_{lt}(P_F) = FLG_{lt}\left[P_S^{nst}(P_F)\right] \quad (24)$$

By this treatment, a laminar and a transitional zone for the current element is generated within the turbulent remainder of the computational grid. A partitioning into a pure laminar zone within the turbulent remainder of the flow domain (point transition) is shown in Fig. 2. All of the steps of this procedure must be applied to all elements n_e, $1 \leq n_e \leq N_e$, of the configuration for which transition determination is performed. The order of the elements within the procedure is irrelevant.
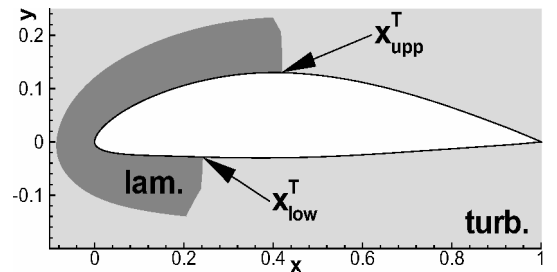

Fig. 1   Laminar, transitional, and turbulent flow regions at surface side q.

Fig. 2   Laminar zone in a turbulent remainder of the flow domain, point transition.

III. Computations

A. Test Case

The test case used to investigate the functionality of the algorithms is the two-dimensional Airbus A310 three-element landing configuration consisting of slat, main airfoil, and flap.[16–18] The turbulence model used is the Spalart–Allmaras one-equation transport model with Edwards and Chandra modification,[19] and the following different computational cases have been performed: 1) fully turbulent; 2) prescribed (fixed) point transition; 3) determined transition, point transition; 4a) determined transition, transition length; and 4b) determined transition, transition length.

The aerodynamic parameters used in the computations are $M_\infty = 0.22$, $Re_\infty = 4.1 \times 10^6$, and $\alpha = 22.4$ deg.

B. Results

The computations of all cases result in a flow structure that is typical for an aircraft multi-element high-lift configuration before maximum lift. The flow is fully attached on upper and lower sides of the slat and the main airfoil and on the lower side of the flap. On the upper side of the flap, there exists a small separation bubble at the trailing edge. In the cove of the main airfoil and of the slat, separation bubbles are located. The separation bubble in the slat cove is significantly smaller in the cases with transition than in the fully turbulent case.

Figure 3 shows the convergence history of the computations of cases 2, 3, and 4a. For all computations with transition, a three-level multigrid method was applied, and the fully turbulent computation had to be run in single-grid mode. All computations with transition converge satisfactorily fast. For the runs with transition determination, the flowfield was initialized with the solution of the fully turbulent computation after 15,000 RANS cycles. The fully turbulent computation needs about 70,000 cycles to converge. An attempt at a preconditioned computation did not succeed because the density residual leveled out at an order of magnitude of 10^{-2}. In this context, a computation is considered to be converged when the value of the lift coefficient c_l does not change anymore above the fourth decimal digit and the value of the drag coefficient c_d does not change anymore above the sixth decimal digit.

Figures 4 and 5 show the convergence history of the transition locations when applying point transition and formula a. For all three elements of the configuration, the longitudinal coordinates of the separation locations are plotted vs the RANS iteration cycles. Laminar and nonlaminar separation locations were plotted. The laminar separation locations are marked with symbols; the nonlaminar separation locations are not marked. All laminar separation points have been set as transition locations on the upper sides of the elements.

The procedure starts with transition locations initially set at the trailing edges of all elements, so that in the beginning a fully laminar flow is computed. Thus, it is ensured that laminar separation will occur in any case and that the largest possible extent of laminar flow, excluding laminar separation bubbles, is obtained. During the
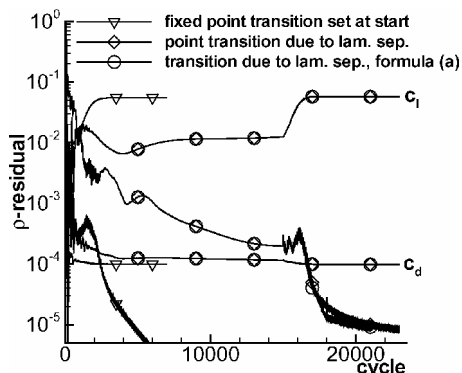

Fig. 3   Convergence history of the RANS computations with transition for the basic test case.
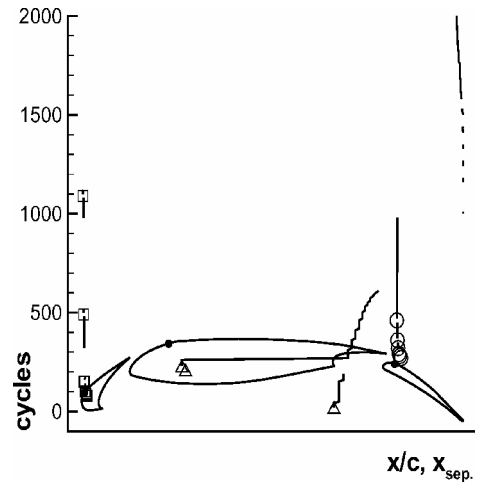

Fig. 4   Convergence history of the transition locations, point transition.
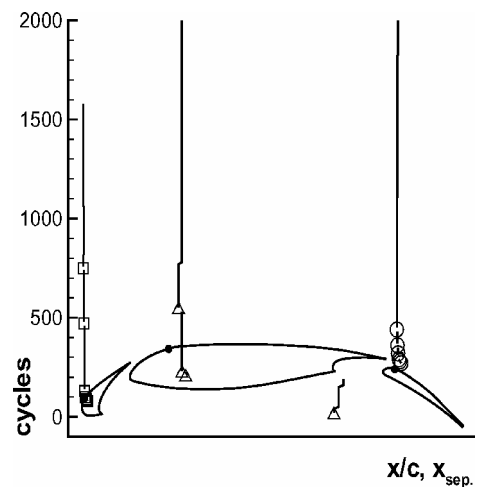

Fig. 5   Convergence history of the transition locations, transition lengths, formula a.

computation the laminar separation points move from the trailing edges toward the noses of each element. Because of the high angle of attack, this happens on the upper sides only.

In both cases, the laminar separation stops near the element's nose on all three elements. In the case of point transition, the nonlaminar separation vanishes on the slat, and the main airfoil and moves back to the trailing edge of the flap. In the case of formula a, a nonlaminar separation point remains directly downstream of the transition point of the main airfoil and the flap. On the slat, the nonlaminar separation vanishes after about 1600 RANS cycles. The transition locations that existed during the experimental measuring of the configuration are plotted as black circular symbols on the surface of the elements (Figs. 4 and 5). The transition points on slat and flap have been determined using an approximate computational transition prediction method. The transition point on the main airfoil is the position of a transition band that tripped the boundary layer in the experiment. The differences $\Delta x_{tr,elem}$ between the experimental and the computed values of the transition locations are given as

$$\Delta x_{tr,elem} = \left(x_{tr,elem}^{comp} - x_{tr,elem}^{exp} \right) / x_{tr,elem}^{exp} \qquad (25)$$

in Table 1. For the locations of free transition on slat and flap, very good results were obtained for both cases.

The convergence histories of the transition locations are shown in a blowup of the main airfoil (Figs. 6 and 7) and the flap (Figs. 8 and 9) with the corresponding surface friction distributions c_f. The regions on the upper sides of the elements that are marked with a thick line indicate the domains in which the separation locations, laminar and nonlaminar, are moving to and fro. In Figs. 7 and 9, the

Table 1 Δ of transition locations

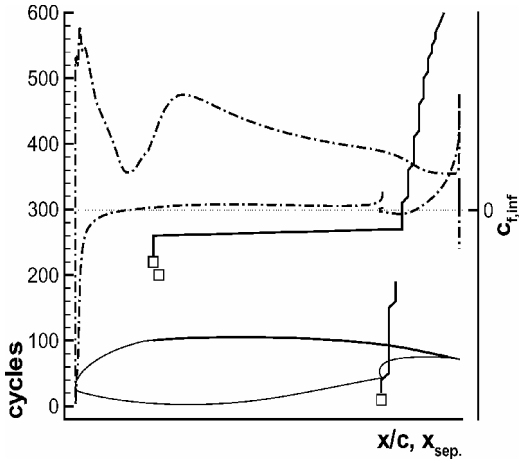| Element	$\Delta x_{\mathrm{tr,elem}}$	
	Formula a,%	Formula b,%
Slat	1.7	3.4
Main airfoil	24.5	18
Flap	0.79	0.78

Fig. 6   Convergence history of the transition locations and c_f distribution of the main airfoil, point transition.
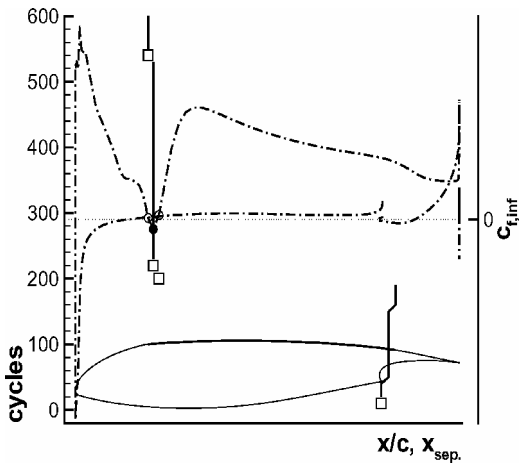

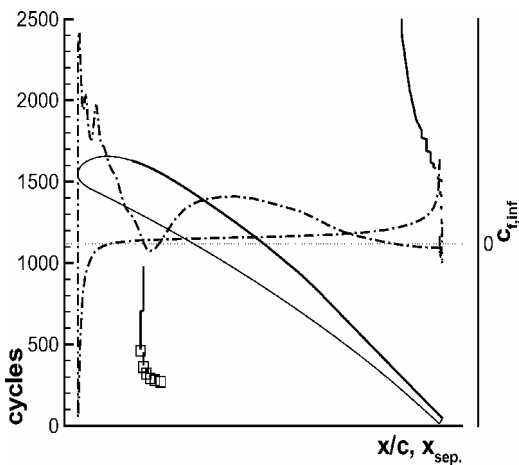Fig. 7   Convergence history of the transition locations and c_f distribution of the main airfoil, transition lengths, formula a.

Fig. 8   Convergence history of the transition locations and c_f distribution of the flap, point transition.
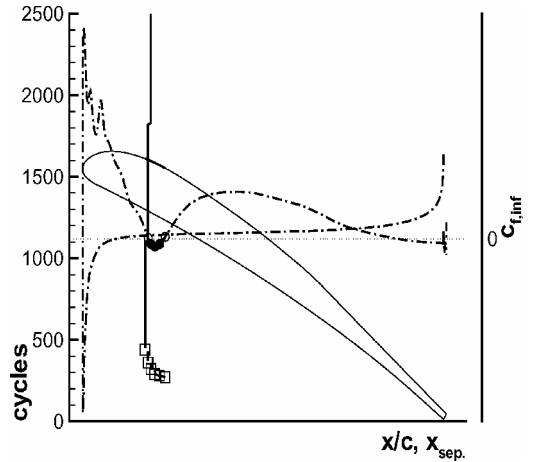
Fig. 9   Convergence history of the transition locations and c_f distribution of the flap, transition lengths, formula a.
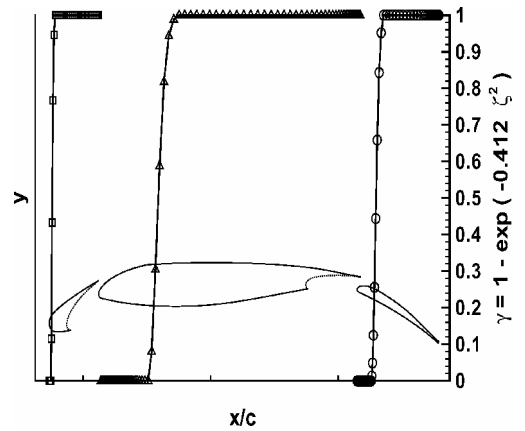

Fig. 10   Extent and shape of the transitional flow domains, transition lengths, formula a.
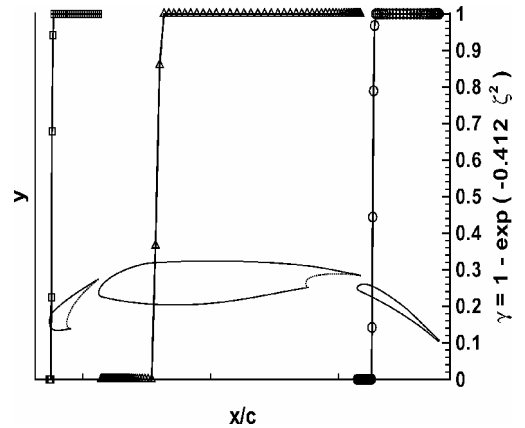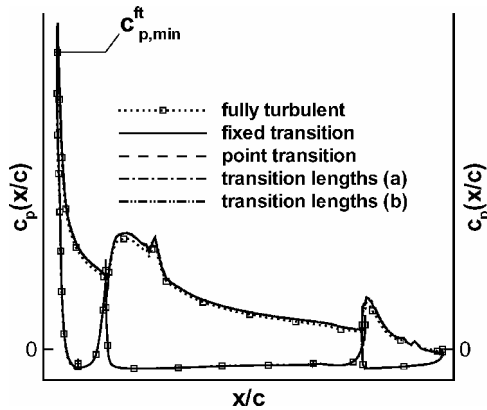

Fig. 11   Extent and shape of the transitional flow domains, transition lengths, formula b.

nonlaminar separation regions downstream of the transition points are clearly visible on the main airfoil and flap.

Figures 10 and 11 show the distributions of FLG_{lt} along the upper sides of the surface contours of all elements and, thus, the computed transitions lengths, $l_{\mathrm{tr,elem}}$, resulting from formulas a and b. Formula a yields values between 12 and 15% of the corresponding element's chord length, as given in Table 2, which are much greater than experimental findings in which values between 3 and 5% are usual. Formula b, however, yields the expected magnitude of the values of the transition lengths and prevents the transitional

Table 2   Values of chord lengths and transition lengths

Element	c_{elem}	Formula a,%	Formula b,%
		$l_{tr,elem}$	
Slat	0.2047	4.64 c_{slat}	4.9 c_{slat}
Main airfoil	0.82053	11.85 c_{main}	4.75 c_{main}
Flap	0.2998	15.19 c_{flap}	4.7 c_{flap}
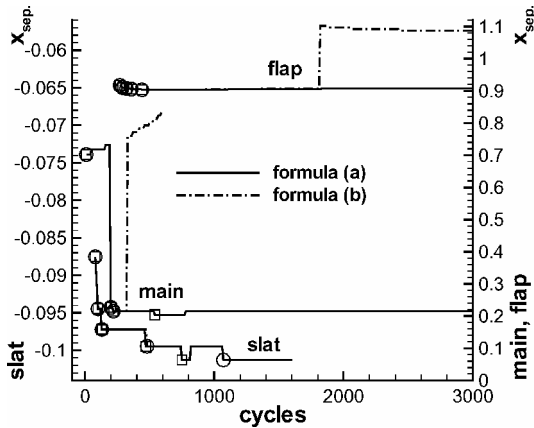
Fig. 12   Convergence history of the transition locations, formula a vs formula b.

Fig. 13   Distributions of c_p.

Table 3   Force coefficients

Case	c_l	c_d, counts
Fully turbulent (FT)	c_l^{ft}	c_d^{ft}
Fixed point transition (FPT)	$c_l^{ft} + 0.1953$	$c_d^{ft} - 130.5$
Determined point transition (DPT)	$c_l^{ft} + 0.2131$	$c_d^{ft} - 139.2$
Determined transition, formula a (DTA)	$c_l^{ft} + 0.2191$	$c_d^{ft} - 140.2$
Determined transition, formula b (DTB)	$c_l^{ft} + 0.2174$	$c_d^{ft} - 140.6$

Fig. 14   Distributions of c_f, fully turbulent, prescribed and determined point transition, determined transition with formula a.

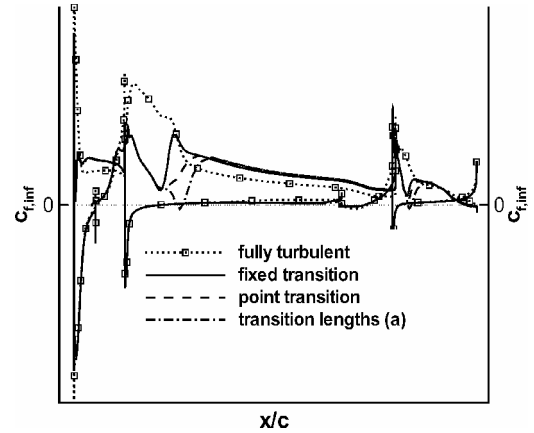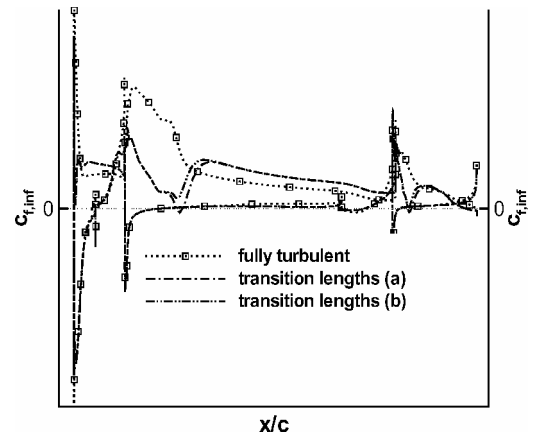**Fig. 15   Distributions of c_f, determined transition, formula a vs formula b.**

separations directly downstream of the transition locations (Table 2).

Figure 12 shows a comparison of the convergence behavior of the determination procedure applying the two different formulas and shows that the final values of the transition locations are the same. Figure 13 shows the c_p distributions for all cases, Fig. 14 shows the c_f-distributions for cases 1–3 and 4a, and Fig. 15 shows a comparison of the c_f distributions for cases 4a and 4b. As expected, the pressure distributions between the fully turbulent case on the one hand and the cases with transition on the other hand show a clearly visible discrepancy on the upper sides of each element of the configuration. The size of the discrepancy is greatest in the suction zones near the noses of the elements. The pressure distributions of all of the cases with transition can not be distinguished in this representation.

Comparing the c_f distributions of the main airfoil of cases 2 and 3 in Fig. 14, one can clearly see the very strong upstream influence of case 3. The location of the local minimum friction value, which marks the end of the laminar boundary layer, is almost the same as in case 2, although in case 3 the location of transition onset is much farther downstream than in case 2. The difference between the numerically simulated transitional region in case 3 (upstream influence) and the physically simulated transitional region in case 4 is remarkable. The numerically simulated transitional re-

gion is almost 30% longer than the physically simulated transitional region.

In Fig. 15, one can clearly see the differences caused by the different transition length models. For formula b, the transitional separations on the main airfoil and flap directly downstream of the transition points that existed for formula a do not appear. In Table 3, a summary of the computed force coefficients is given.

The differences between two force coefficients $c_{k,1}$ and $c_{k,2}$ were determined according to

$$\Delta c_k = (c_{k,2} - c_{k,1})/c_{k,1}, \qquad k = l, d \qquad (26)$$

and give an impression of the integral effect of the different modeling levels (Table 4). The values $c_{k,1}$ and $c_{k,2}$ are taken from two consecutive lines of Table 4, where the index 1 indicates the upper line and the index 2 the lower line.

C.   Initialization

The basic idea for the transition determination process is to start the flow computation with a flowfield initialized with freestream values and transition locations set very far downstream, for example, at the trailing edges, so that, effectively, a fully laminar flow

Table 4 Δ of force coefficients

| Case	Δc_l, %	Δc_d, %
(FPT–FT)/FT	4.5	_10
(DPT–FPT)/FPT	0.4	_0.7
(DTA–DPT)/DPT	0.13	_0.08
(DTB–DTA)/DTA	_0.04	_0.03

is computed in the beginning of the flow computation. During the transient phase of the computation, the successively detected laminar separation points are then used as transition locations until a converged state of the transition locations has been reached. However, the strategy to start the computation with freestream values fails for the following reason. For the detection of separation points on either the upper or the lower side of an airfoil, the location of the stagnation point on the airfoil must be known because it divides the upper from the lower side from the aerodynamical point of view. The stagnation point is the point of $c_{p,\max}$ at the airfoil surface. Additionally, it is a point where the tangential velocity vectors of two differentially neighboring points have different algebraic signs. In a steady flow, these two facts are valid for the same physical point in space. A separation point is characterized by two differentially neighboring points whose tangential velocity vectors have different algebraic signs, too.

In the first tests with airfoil flows under high angles of attack, it turned out that the two conditions that define the stagnation point do not indicate the same point in space during the transient phase of the computation. Usually, the point with $c_{p,\max}$ is very near the real stagnation point from the very beginning throughout the transient phase until the RANS computation has converged to a steady solution. The point whose neighbors have different signs of their tangential velocity vectors is, at first, located very near to the geometric nose point of the airfoil. During the transient phase it moves downstream from the geometric nose point toward the point with $c_{p,\max}$. When the RANS computation has converged, these two points have become one single point, the stagnation point. Thus, during the transient phase, it is not possible to use these two conditions to detect the stagnation point. Only the $c_{p,\max}$ criterion is usable to find the stagnation point.

Because a separation point is also characterized by two differentially neighboring points whose tangential velocity vectors have different algebraic signs the first detected separation point is the "wrong stagnation point" that is moving toward the point with $c_{p,\max}$. As a consequence, it does not seem possible to start the computation using freestream values as initial values. To overcome this problem, the computation may be initialized with a converged steady solution of a fully turbulent flowfield. In this case, the two conditions that define the stagnation point mark the same physical point in space. All computations initialized with a fully turbulent flowfield yielded very good results.

Figures 16 and 17 suggest the computations that were initialized with freestream values. They illustrate the unfavorable behavior of the RANS computation in the case where the transition point is set at the wrong position on the main airfoil due to the erroneous determination of the laminar separation point. When the separation points were determined for the first time, after about 120 RANS cycles, the wrong stagnation point on the main airfoil is detected as a laminar separation point, located very near to the nose of the main airfoil, and is set as transition point. Figures 16 and 17 show that the flow is strongly affected on all three elements both for the application of point transition and for the modeling of transitional flow using formula a. The computations result in a strongly detached flow over the main airfoil and flap. Also in the cove of the slat, the flow is fully detached. There is no recognizable tendency for the flow to reattach again if the computation were continued.

This behavior is noteworthy because the flow over the main airfoil is fully turbulent on the complete upper side of the airfoil, starting at the wrong stagnation point, running from the nose to the upper trailing edge. In quite a number of cases, such a situation leads to a more stable evolution of the computation compared to a situation where a small region of laminar flow exists. On the other hand, this behavior is known from experimental and numerical investiga-
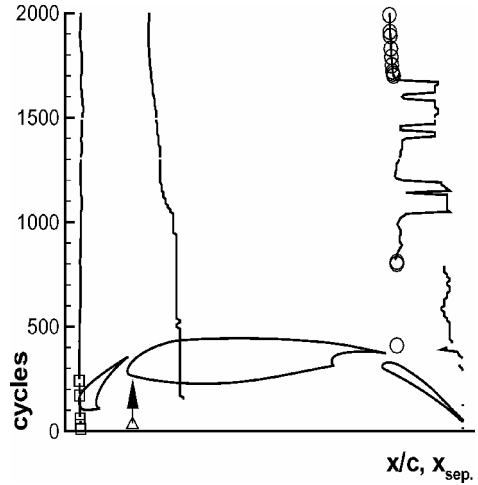
Fig. 16 Convergence history of the transition locations; point transition, without fully turbulent initialization of the flowfield.
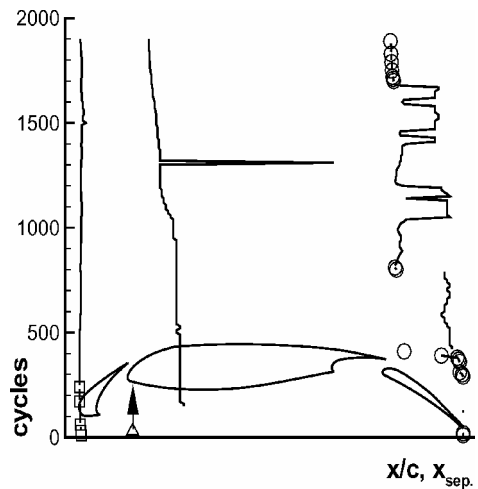
Fig. 17 Convergence history of the transition locations; transition lengths, formula a, without fully turbulent initialization of the flowfield.

tions of flows over single element airfoils near maximum lift with Reynolds numbers in the range at around $Re = 1 \times 10^6$–10×10^6, when the laminar–turbulent tripping of the boundary layer was set too close to the geometrical nose point. In the Figs. 16 and 17, the position of the wrong stagnation point on the main airfoil is marked by an arrow. From this point on, the flow is fully turbulent on the upper side and fully laminar on the lower side. The graphs show again the separation points plotted vs the RANS iteration cycles.

D. Other Turbulence Models

In addition to the application of the Spalart–Allmaras model with Edwards and Chandra modification,[19] other turbulence models were applied to test the strategy of the transition determination procedure for the selected test case. The selected models are the algebraic Baldwin–Lomax model,[20] the standard Wilcox k–ω model,[21] and the linear explicit algebraic (LEA) k–ω model.[22] The transition determination procedure worked successfully in all cases, although the case using the Baldwin–Lomax turbulence model did not converge with respect to the RANS iteration, which was not expected. All determined values for the transition locations and the transition lengths are of the expected magnitude.

IV. Conclusions

The algorithm for the detection of laminar separation was applied to a two-dimensional multi-element configuration on a structured multiblock mesh. In principle, its application is independent of the grid structure and the grid topology. The basic test cases on which

the functionality of the algorithms and the transition determination strategy were tested used the Spalart–Allmaras turbulence model with Edwards and Chandra modification. The separation points were successfully set as transition points, and the determination procedure converged without significant influence on the convergence behavior of the RANS computation.

The determination procedure works successfully, applying point transition and two different transition length models, the one based on boundary-layer properties, formula a, and the other one based on flat plate theory, formula b. Formula a yields transition lengths that are much greater than experimental findings and small transitional separation regions on some of the elements of the configuration directly downstream of the transition locations. Formula b yields transition lengths that have a value of about 5% of the chord length of the corresponding element, which is a value that fits the experimental results. The transitional separation regions do not appear.

In addition to the Spalart–Allmaras model with Edwards and Chandra modification, the algebraic Baldwin–Lomax model, the standard Wilcox k–ω model, and the LEA k–ω model have been applied. All determined values for the transition locations and the transition lengths are of very similar magnitude to the values obtained for the basic test case.

The next steps are the coupling of the extended FLOWer code to the transition prediction module and the comparison of the results with experimental data for the selected test case.

Acknowledgments

References

[1]Radespiel, R., Graage, K., and Brodersen, O., "Transition Predictions Using Reynolds-Averaged Navier–Stokes and Linear Stability Analysis Methods," AIAA Paper 91-1641, June 1991.

[2]Smith, A. M. O., and Gamberoni, N., "Transition, Pressure Gradient and Stability Theory," Rept. ES 26388, Douglas Aircraft Co., Long Beach, CA, Aug. 1956.

[3]van Ingen, J. L., "A suggested Semi-Empirical Method for the Calculation of the Boundary Layer Transition Region," Dept. of Aerospace Engineering, Rept. VTH-74, Univ. of Delft, Delft, The Netherlands, Oct. 1956.

[4]Stock, H. W., and Haase, W., "A Feasibility Study of e^N Transition Prediction in Navier–Stokes Methods for Airfoils," AIAA Journal, Vol. 37, No. 10, 1999, pp. 1187–1196.

[5]Horton, H. P., and Stock, H. W., "Computation of Compressible, Laminar Boundary Layers on Swept, Tapered Wings," Journal of Aircraft, Vol. 32, No. 6, 1995, pp. 1402–1405.

[6]Stock, H. W., and Degenhardt, E., "A Simplified e^N Method for Transition Prediction in Two-Dimensional, Incompressible Boundary Layers," Zeitung für Flugwissenschaft und Weltraumforschung, Vol. 13, Heft 1, 1989, pp. 16–30.

[7]Warren, E. S., and Hassan, H. A., "Transition Closure Model for Predicting Transition Onset," Journal of Aircraft, Vol. 35, No. 5, 1998, pp. 769–775.

[8]Czerwiec, R., Edwards, J. R., Rumsey, C. L., Bertelrud, A., and Hassan, H. A., "Study of High-Lift Configurations Using k–ζ Transition/Turbulence Model," Journal of Aircraft, Vol. 37, No. 6, 2000, pp. 1008–1016.

[9]Edwards, J. R., Roy, C. J., Blottner, F. G., and Hassan, H. A., "Development of a One-Equation Transition/Turbulence Model," AIAA Journal, Vol. 39, No. 9, 2001, pp. 1691–1698.

[10]Kroll, N., Rossow, C. C., Becker, K., and Thiele, F., "The MEGAFLOW Project," Aerospace Science and Technology, Vol. 4, No. 4, 2000, pp. 223–237.

[11]Krumbein, A., and Stock, H. W., "Laminar–turbulent Transition Modeling in Navier–Stokes Solvers Using Engineering Methods," ECCOMAS 2000 [CD-ROM], International Center for Numerical Methods in Engineering, Barcelona, Depósito Legal: B-37139-2000, 2000.

[12]Krumbein, A., "Coupling of the DLR Navier-Stokes Solver FLOWer with an e^N-Database Method for Laminar–Turbulent Transition Prediction on Airfoils," New Results in Numerical and Experimental Fluid Mechanics III, Notes on Numerical Fluid Mechanics, Vol. 77, Springer-Verlag, Berlin, 2002, pp. 92–99.

[13]Stock, H. W., and Haase, W., "Navier–Stokes Airfoil Computations with e^N Transition Prediction Including Transitional Flow Regions," AIAA Journal, Vol. 38, No. 11, 2000, pp. 2059–2066.

[14]Dhawan, S., and Narasimha, R., "Some Properties of Boundary Layer Flow During the Transition from Laminar to Turbulent Motion," Journal of Fluid Mechanics, Vol. 3, 1958, pp. 418–436.

[15]Walker, G. J., "Transitional Flow on Axial Turbomachine Blading," AIAA Journal, Vol. 27, No. 5, 1989, pp. 595–602.

[16]Dargel, G., and Schnieder, H., "GARTEUR AD (AG08) Final Report," GARTEUR High Lift Action Group, Rept. AD (AG08), TP043, MBB Transport- und Verkehrsflugzeuge, Bremen, Germany, Nov. 1989.

[17]Thibert, J. J., "The GARTEUR High Lift Research Programme," High-Lift System Aerodynamics, CP-515, AGARD, 1993, pp. 16-1–16-21.

[18]Brodersen, O., Ronzheimer, A., Ziegler, R., Kunert, T., Wild, J., and Hepperle, M., "Aerodynamic Applications Using MegaCads," Proceedings of 6th International Conference on Numerical Grid Generation in Computational Field Simulations, edited by M. Cross, P. Eiseman, J. Häeuser, B. K. Soni, and J. F. Thompson, International Society of Grid Generation, National Science Foundation Engineering Research Center for Computational Field Simulation, Mississippi State Univ., Mississippi State, Mississippi 39762, U.S.A. 1998, pp. 793–802.

[19]Edwards, J. R., and Chandra, S., "Comparison of Eddy Viscosity–Transport Turbulence Models for Three-Dimensional, Shock-Separated Flowfields," AIAA Journal, Vol. 34, No. 4, 1996, pp. 756–763.

[20]Baldwin, B. S., and Lomax, H., "Thin-Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper 78-257, Jan. 1978.

[21]Wilcox, D. C., "Reassessment of the Scale-Determining Equation for Advanced Turbulence Models," AIAA Journal, Vol. 26, No. 11, 1988, pp. 1299–1310.

[22]Rung, T., Lübcke, H., Franke, M., Xue, L., Thiele, F., and Fu, S., "Assessment of Explicit Algebraic Stress Models in Transonic Flows," Proceedings of the 4th International Symposium on Engineering Turbulence Modelling and Measurements, edited by W. Rodi, and D. Laurence, Elsevier, Amsterdam, 1999, pp. 659–668.