드래그 감소를 위한 유체의 최적 엑티브 제어 및 최적화 알고리즘의 개발(1)
-대용량, 비선형 유체의 최적화를 위한 알고리즘 및 테크닉의 개발

Optimal Active-Control & Development of Optimization Algorithm for Reduction of Drag in Flow Problems(1)
-Development of Optimization Algorithm and Techniques for Large-Scale and Highly Nonlinear Flow Problem

박 재 형†
Bark, Jaihyeong
(논문접수일 : 2007년 7월 20일 ; 심사종료일 : 2007년 8월 16일)

요 지

바람에 저항하는 초고층 건물, 비행기나 자동차, 물에 저항하는 선박 등은 동일한 거동을 보여준다. 즉, 유속이 빨라 질 경우, 건물 혹은 비행기, 자동차, 선박 뒤편에는 마이너스 압력과 와류가 발생하게 되는데 이로 인해 건물에서는 변위가 크게 발생하게 되고, 비행기나 자동차, 선박 등에서는 속력이 저하된다. 본 연구에서는 흡입과 방출이라는 기법을 이용하여 유체의 흐름을 우리가 원하는대로 적극적으로 제어하고자 한다. 그렇게 할 수만 있다면 초고층 건물에서의 변위를 대폭 줄일 수 있을 것이고, 자동차나 비행기 선박 등은 더 빠른 속도로 달릴 수 있을 것이다. 그렇다면 문제는 유체를 제어하기 위한 최적의 흡입 혹은 방출량을 구하는 것이고, 이 최적의 양들을 어떤 방법으로 구하는 것이냐 하는 것이다.

본 연구는 최적화 기법을 사용하여 Navier-Stokes 유체를 받는 물체의 표면에서 최적의 흡입, 그리고 방출량을 결정하려는 시도에서 출발하였다. 그러나 이 문제는 큰 Reynols Number 상태에서는 높은 비선형성으로 인하여 직접 한번에 Navier-Stokes 유체의 해석조차 불가능하였고, 더군다나 너무나 많은 변수로 인하여 기존의 방법으로는 최적화는 도저히 불가능 하였다. 본 연구에서는 이를 해결하기 위한 최적화 알고리즘을 제안하고, 또한 수렴속도도 대폭 증가시키기 위한 매우 효율적인 몇 가지 방법들을 제안하였다.

핵심용어 : 최적 제어, Navier-Stokes 유체, 흡입, 방출, SQP기법, 유사뉴턴법, 민감도 해석, 수렴판정치의 제어

Abstract

Ever since the Prandtl's experiment in 1934 and X-21 airjet test in 1950 both attempting to reduce drag, it was found that controlling the velocities of surface for extremely fast-moving object in the air through suction or injection was highly effective and active method. To obtain the right amount of suction or injection, however, repetitive trial-and error parameter test has been still used up to now.

This study started from an attempt to decide optimal amount of suction and injection of incompressible Navier-Stokes by employing optimization techniques. However, optimization with traditional methods are very limited, especially when Reynolds number gets high and many unexpected variables emerges. In earlier study, we have proposed an algorithm to solve this problem by using step by step method in analysis and introducing SQP method in optimization.

In this study, we propose more effective and robust algorithm and techniques in solving flow optimization problem.

Keywords : *optimal control, Navier-Stokes flow, suction, injection, SQP method, quasi-Newton method, sensitivity analysis, control of convergency criteria*

† 책임저자, 정회원 · 목원대학교 건축학부 교수
Tel: 016-464-6250 ; Fax: 042-822-5260
E-mail: jhbark@mokwon.ac.kr

• 이 논문에 대한 토론을 2007년 12월 31일까지 본 학회에 보내주시면 2008년 2월호에 그 결과를 게재하겠습니다.

1. Introduction

Highrise buildings acting against the wind shows same behavior with vehicles(e.g. airplanes, automobiles, ships) under the influence of wind or water. That is, as the speed of wind or water gets faster, minus pressure and vortex occur in the backside of the buildings and vehicles. In turn, it causes large displacement in highrise buildings and reduced speed in vehicles. To reduce such effects, attempts to partly change the flow by altering shape or adding attachment(e.g. rear wing of automobile) had been made. However, the outcome of these passive efforts was not so powerful.

"Can we control the flow as we want?"

Active control of flow by suction and injection came up as an answer to the question. This can greatly reduce minus pressure and vortex, and in turn leads to decreased displacement in buildings and increased speed in vehicles. Then, questions such as optimal amount of suction or injection needed to best control the flow and a way to obtain the amount remain to be answered.

Ever since the Prandtl's experiment in 1934 and X-21 airjet test in 1950 both attempting to reduce drag, it was found that controlling the velocities of surface for extremely fast-moving object in the air through suction or injection was highly effective and active method. To obtain the right amount of suction or injection, however, repetitive trial-and error parameter test has been still used up to now.

This study started from an attempt to decide optimal amount of suction and injection of incompressible Navier-Stokes by employing optimization techniques. However, optimization with traditional methods is very limited, especially when Reynolds number gets high and many unexpected variables emerges. In earlier study, we have proposed an algorithm to solve this problem by using step by step method in analysis and introducing SQP method in optimization(Bark, 2002).

In this study, we propose more effective and robust algorithm and techniques in solving flow optimization problem.

2. Analysis for the incompressible steady-state Navier-Stokes flow

Here, we discuss the numerical approximation and solution of the governing flow equations, namely the incompressible Navier-Stokes equations.

2.1 Finite element approximation

Incompressible steady-state Navier-Stokes equations can be expressed as:

$$-\mu\Delta\mathbf{u}+\rho(\mathbf{u}\cdot\nabla)\mathbf{u}+\nabla p=0 \qquad (1)$$

$$\nabla\cdot\mathbf{u}=0 \qquad (2)$$

where μ is the dynamic viscosity, ρ is the density, p is the pressure, and $\mathbf{u}$ is the flow velocity.

The programming complexity associated with Ladyzhenskaya-Babuska-Brezzi(LBB)-satisfying elements leads us to consider a different form of the equations, whose finite element approximation is not subject to the the LBB condition. This form is established by replacing the conservation of mass equation (2) with the approximation

$$\nabla\cdot\mathbf{u}=-\epsilon p \qquad (3)$$

This allows us to solve for pressure in terms of velocity, and we can thus eliminate pressure from the governing equations, resulting equation (1) and (2) can be reduced by (4)

$$-\mu\Delta\mathbf{u}+\rho(\mathbf{u}\cdot\nabla)\mathbf{u}-\frac{1}{\epsilon}\nabla(\nabla\cdot\mathbf{u})=0 \qquad (4)$$

Thus, the number of unknowns is reduced significantly since pressure is no longer a variable. Most importantly, the LBB condition is no longer applicable, since equation (4) is not mixed form. This simplifies the programming techniques and data structures. Of course, in the limit as penalty parameter $\epsilon\to 0$, we regain the original problem. $10^{-4}\sim$

10^{-7} was a reasonable range for the penalty parameter(ϵ).

The finite element formulation can be obtained from the variational, or weak formulation of equations (4) by using the Galerkin method.

A code has been written that solves the incompressible Navier-Stokes equations. It employs isoparametric biquadratic 9-nodes rectangular elements. For Gauss-Legendre numerical integration, basically a 3×3 scheme is used. However, under-integration, i.e. 2×2, is necessary for the pressure terms to insure singularity of the corresponding contribution to the Jacobian matrix, and hence avoid "locking" of the approximate solution. The pressure terms are the ones that involve the penalty parameter(ϵ).

2.2 Solution methods for discrete Navier-Stokes equations

The discrete form of the Navier-Stokes equations is a system of nonlinear algebraic equation, i.e. $\mathbf{h}(\mathbf{u})=0$, where $\mathbf{h}$ represents the residuals and $\mathbf{u}$ represents the vector of unknown velocities. A very effective method for solving these equations is Newton's method. It is well-known that this method is locally quadratically convergent, that is that close to the solution, the error is squared between subsequent iterations, i.e. the number of correct digit is doubled. Solution of equation (4) by Newton's method requires the Jacobian matrix of $\mathbf{h}$. These Jacobian matrices $\mathbf{J}$ are analogous to stiffness matrices for linear finite element problems.

Following is the summary of the steps of Newton's method for Analysis

1) update $\mathbf{h}_k$ and $\mathbf{J}_k$
2) check convergence criterion : if $\|\mathbf{h}_k\| \le \gamma$, then terminate; otherwise go to step 3
3) solve $\mathbf{J}_k\mathbf{p} = -\mathbf{h}_k$
4) $\mathbf{u}_{k+1} = \mathbf{u}_k + \mathbf{p}$
5) go to step 1

where $\mathbf{h}_k$ and $\mathbf{J}_k$ indicate evaluation of $\mathbf{h}$ and $\mathbf{J}$ at $\mathbf{u}_k$, k is the k-th iterate point. In general we use $\gamma = 10^{-7}$.

Step 3 is a linear system of the form:

$$\begin{bmatrix} J_{xx} & J_{xy} & J_{xz} \\ J_{yx} & J_{yy} & J_{yz} \\ J_{zx} & J_{zy} & J_{zz} \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} -h_x \\ -h_y \\ -h_z \end{bmatrix} \tag{5}$$

Because of the large dimensions involved, we must use an efficient method to solve this system of equations. It will be seen that sensitivity analysis requires the repeated solution of linear systems having the same coefficient matrix, but different right-hand sides, each corresponding to a different control variables. This, as well as the fact that $\mathbf{J}$ is unsymmetric, favors sparse direct methods for solution. Perhaps the most efficient code for factorization of sparse unsymmetric matrices is the unsymmetric-pattern multifrontal sparse *LU* factorization code UMFPACK(Davis, 1993), and we use this code for solution of the linear system (5) arising at each step of Newton's method.

Despite its excellent convergence rate, Newton's method is only locally convergent. In particular for Navier-Stokes equations, an upper bound on the diameter of the convergence "ball" for Newton's method varies as $1/Re$(Gunzburger, 1989). The consequence is that as the Reynolds number increases, one needs better initial guesses to guarantee convergence to a solution; otherwise, divergence may occur. Thus, we solve a sequence of problems leading to the Reynolds number of interest, as follows:

1) solve the problem with low Reynolds number.
2) increment Reynolds number.
3) solve the problem using the results of previous step as initial guesses.
4) repeat above steps until final Reynolds number is reached.

Density(ρ) is used as a parameter to increase Reynolds number. Within each step, we iterate until convergence is achieved; then the converged solution is used as the initial guesses for the next step. However we will discuss a more sophisticated continuation strategy that makes use of sensitivity information.

3. Optimization

The continuous optimization problem is defined as:

$$\text{minimize } 2\mu\int_{\Omega}[\mathbf{D}:\mathbf{D}]\,d\Omega \tag{6}$$

subject to

$$-\mu\Delta\mathbf{u}+\rho(\mathbf{u}\cdot\nabla)\mathbf{u}-\frac{1}{\epsilon}\nabla(\nabla\cdot\mathbf{u})=0 \tag{7}$$

where $\mathbf{D}=\mathbf{D}(\mathbf{u})=(\nabla\mathbf{u}+\nabla\mathbf{u}^T)/2$, and the symbol : represents the scalar product of two tensors.

3.1 Formulation of the discrete optimization problem

Using the finite element approximation defined in section 2, we arrive at a discretized form of the optimal control problem, which in symbolic form is:

$$\textit{minimize} \quad \Phi(\mathbf{u},\mathbf{b}) \tag{8}$$

$$\textit{subject to} \quad \mathbf{h}(\mathbf{u},\mathbf{b})=0 \tag{9}$$

Here, the constraints $\mathbf{h}=0$ are the discrete form of the Navier-Stokes equations. We have partitioned the velocities $\mathbf{u}$(i.e. the velocities at all nodes other than those that lie on suction/injection holes), and the control variables $\mathbf{b}$(i.e. the velocities of nodes where suction/injection is applied). This objective function Φ is related to velocities by

$$\Phi=\frac{1}{2}\mathbf{u}^T\mathbf{J}_L\mathbf{u} \tag{10}$$

and again depends on both state velocities $\mathbf{u}$ and control velocities $\mathbf{b}$. Here $\mathbf{J}_L$ is the portion of the Jacobian matrix that depends on viscosity. We may also choose to augmented the constraints by bounds on the control variables. The problem is then one in nonlinear-constrained smooth optimization.

In general it is not straightforward to apply SQP methods to flow optimization problems of the form (8) and (9), since the constraint sets produced are very large and nonlinear. Therefore we pursue a decomposition of the problem into the state space and the control space, as follows: solve at each iteration the discrete Navier-Stokes equations($\mathbf{h}(\mathbf{u},\mathbf{b})=0$) for the state variables($\mathbf{u}$) given values of the control variables($\mathbf{b}$). Thus, we have eliminated the state equations from the constraint set, and we have eliminated the state variables from the set of optimization variables. As a result of this decomposition, the state variables become an implicit function of the control variables, the implicit function being the flow solution itself. Thus, we can write the optimization problem as the unconstrained optimization problem:

$$\textit{minimize} \quad \Phi(\mathbf{u}(\mathbf{b}),\mathbf{b}) \tag{11}$$

The dimension of the optimization problem is now greatly reduced, and the constraints are eliminated. However, the implicit dependence of state variables on control variables requires a special approach, called sensitivity analysis, to find the derivatives of the objective function with respect to the control variables.

3.2 Sensitivity analysis

Here, we will show how to obtain the gradient of objective function and constraints with respect to the control variables, taking into account the implicit dependence of the state variables on the controls through the discrete Navier-Stokes equation. SQP requires this information to solve the optimization problem. Furthermore, this sensitivity analysis also gives us a good way to obtain better initial guesses for state variables for the analysis problem at each control iteration, as will be shown in section 5.1.

Let F denote either the objective or a constraint function. Here, F depends explicitly on the control variables $\mathbf{b}$ as well as implicitly on them through the state variables $\mathbf{u}$:

$$F = F(\mathbf{u}(\mathbf{b}), \mathbf{b}) \tag{12}$$

The relationship between $\mathbf{u}$ and $\mathbf{b}$ is dictated by the discrete form of the Navier-Stokes equations, which we refer to as the state equations:

$$\mathbf{h}(\mathbf{u}(\mathbf{b}), \mathbf{b}) = 0 \tag{13}$$

The total derivative of the objective with respect to the control variables can be found by applying the chain rule,

$$\frac{DF}{D\mathbf{b}} = \frac{\partial F}{\partial \mathbf{b}} + \frac{\partial F}{\partial \mathbf{u}} \cdot \frac{d\mathbf{u}}{d\mathbf{b}} \tag{14}$$

Here $\partial F/\partial \mathbf{b}$ and $\partial F/\partial \mathbf{u}$ can be readily found from the expression for the discrete form of the objective function. The only unknown is $d\mathbf{u}/d\mathbf{b}$, but this vector can be found by applying the implicit function theorem for the state equations, yielding

$$\frac{D\mathbf{h}}{D\mathbf{b}} = \frac{\partial \mathbf{h}}{\partial \mathbf{b}} + \frac{\partial \mathbf{h}}{\partial \mathbf{u}} \cdot \frac{d\mathbf{u}}{d\mathbf{b}} = 0 \tag{15}$$

Thus, the sensitivity of the velocity field with respect to control variables can be found by solving the linear system

$$\frac{\partial \mathbf{h}}{\partial \mathbf{u}} \cdot \frac{d\mathbf{u}}{d\mathbf{b}} = -\frac{\partial \mathbf{h}}{\partial \mathbf{b}} \tag{16}$$

This is a linear system with coefficient matrix that is just the Jacobian of the state equations with respect to the state variables, and thus is a component of any Newton solver. Therefore, the same linear solver that is at the heart of the flow simulation is used for sensitivity analysis, with the exception of the right-hand side vectors. The matrix of these vectors, $\partial \mathbf{h}/\partial \mathbf{b}$, can be readily found analytically. It is seen that this first-order sensitivity analysis requires the solution of a system of equations for as many right-hand sides as there are control variables.

3.3 SQP methods for solving the optimization problem

The basic SQP iteration can be viewed as(Bark, 2002):

1) do analysis (obtain $\mathbf{u}_k$ knowing $\mathbf{b}_k$ by solving equation (13))
2) do sensitivity analysis (obtain $DF/D\mathbf{b}_k$ from equation (14))
3) construct approximate Hessian matrix $\mathbf{B}_k$ from $DF/D\mathbf{b}_k$
4) check convergence criterion : if $\|DF/D\mathbf{b}_k\| \leq \gamma$, then terminate; otherwise go to step 5
5) find $\mathbf{p}_k$ by solving

$$minimize \ \frac{1}{2}\mathbf{p}_k^T \mathbf{B}_k \mathbf{p}_k + \mathbf{p}_k^T \frac{DF}{D\mathbf{b}}_k$$

6) $\mathbf{b}_{k+1} = \mathbf{b}_k + \alpha \mathbf{p}_k \ \ \alpha \in (0,1]$
7) go to step 1

As mentioned before, the continuation technique is used on the analysis problem. The straightforward way to integrate this continuation technique into an optimization method is as follows(see Figure 1).

1) use continuation to solve the analysis problem for the given Reynolds number.
2) update the control variables by doing one optimization iteration.
3) repeat the above for each optimization iteration until the optimum is reached.

Actually we proposed this continuation technique in the previous paper(Bark, 2002), which we will

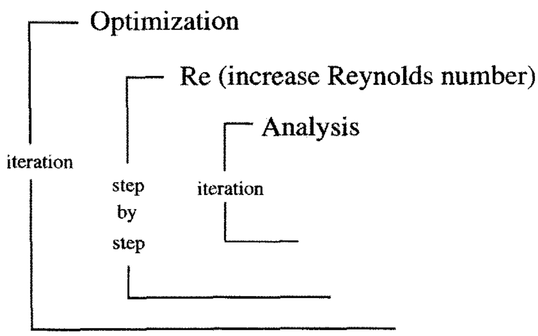

Figure 1 Algorithm of OA-SQP

refer to as *OA-SQP*(old algorithm).

This algorithm is superior to the steepest descent method, because it uses (approximate) curvature information in solving the optimization problem. However this algorithm still entails much work. We perform analysis to get the value of state variables, which are used for the objective function and the sensitivity analysis. Usually the number of state variables is much larger than that of control variables. Therefore most of cost will be spent on getting the values of the state variables. One analysis step itself requires a lot of work; furthermore we need to perform several analysis steps at each optimization iteration. Therefore solving analysis with only continuation at each control iteration requires a great deal of work especially for higher Reynolds number problem. Therefore we want to propose a new algorithm and a number of techniques to overcome the computational complexity of *OA-SQP*.

4. Newly developed algorithm

Steepest descent and *OA-SQP* method entail a lot of work, and sometimes they may not converge at all. Here, we will propose new algorithm to increase the efficiency of the optimization process.

We have used continuation techniques for the analysis problem in *OA-SQP*, but a more promising idea is to integrate continuation with optimization, which we refer to as *NA-SQP*(newly developed algorithm). See Figure 2.

1) solve the optimization problem with a given Reynolds number (Initially a low Reynolds number is used). At each optimization iteration, we only need to do one analysis step, and the initial guesses for analysis are the values of previous optimization iteration.
2) increment Reynolds number. Initial values of analysis quantities(state variables) and optimization quantities(control values, Hessian matrix and Lagrangian multipliers) of the current step are taken as the converged value of the previous step.
3) repeat above until target Reynolds number is reached.

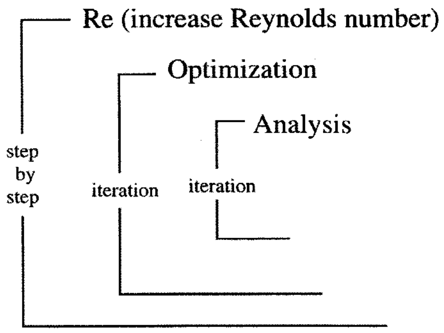

Figure 2 Algorithm of NA-SQP

In this method, initial values of analysis and optimization quantities at the beginning of each Reynolds step are taken as those of previous Reynolds step, and initial values of analysis variables at each optimization iteration are taken as those of the previous optimization iteration.

Usually this method needs more optimization iteration than *OA-SQP* dose. However this method reduces the number of analysis iterations significantly, and most of cost to solve this kind of problem is spent on doing analysis. Therefore if we want to solve an Reynolds number(Re)=500 problem with step size Re=50, *OA-SQP* needs one optimization cycle but n analysis step for each optimization iteration, which *NA-SQP* needs n optimization cycles but only one analysis step for each optimization iteration. Our results demonstrate that *NA-SQP* not only improves robustness(converges in cases where *OA-SQP* doesn't), but also increase the efficiency of the optimization process. For example, for a problem with a target Reynolds number of 400 with step size 50, it reduced CPU time by a factor of 5 over the *OA-SQP*. For Reynolds numbers over 500, *OA-SQP* did not converge with step size 50.

5. Proposed techniques

In previous section, we proposed *NA-SQP*. Here, we will propose several techniques to further increase the efficiency of the optimization process. They all also include the feature of *NA-SQP*.

5.1 Technique-1

Since methods of Newton form are basically stationary iterative processes, and since our analysis solver and optimizer are both methods of Newton form, it is important to provide then with good initial guesses. One technique that can be used to provide a good initial guesses is available from sensitivity information at essentially no cost.

At each optimization iteration, initial guesses for analysis are the results of previous iteration's analysis. Therefore,

$$\mathbf{u}_{initial} = \mathbf{u}_{previous} \tag{17}$$

However we can approximate new values of the velocities at each optimization iteration, using a first-order approximation of their change:

$$\mathbf{u}_{new} \approx \mathbf{u}_{previous} + \frac{d\mathbf{u}}{d\mathbf{b}} \cdot \triangle\mathbf{b} \tag{18}$$

Here, $\mathbf{u}_{previous}$ and $\triangle\mathbf{b}$ are known. From sensitivity analysis, $d\mathbf{u}/d\mathbf{b}$ is available easily.

$$\frac{d\mathbf{u}}{d\mathbf{b}} = -\left(\frac{\partial\mathbf{h}}{\partial\mathbf{u}}\right)^{-1} \cdot \frac{\partial\mathbf{h}}{\partial\mathbf{b}} \tag{19}$$

Therefore we can use this $\mathbf{u}_{new}$ to initiate the Newton solution at each control iteration.

$$\mathbf{u}_{initial} = \mathbf{u}_{new} \tag{20}$$

We refer this technique as *T1-NA-SQP*. It also includes the feature of *NA-SQP*.

This idea can be extended to include second-order information. Although we did not implement it, we will sketch the basic idea here.

We can approximate $\mathbf{u}_{initial}$ using one more term in the Taylor series expression:

$$\mathbf{u}_{initial} \approx \mathbf{u}_{previous} + \frac{d\mathbf{u}}{d\mathbf{b}} \cdot \triangle\mathbf{b} + \frac{1}{2}\triangle\mathbf{b}^T \frac{d^2\mathbf{u}}{d\mathbf{b}^2}\triangle\mathbf{b} \tag{21}$$

Even with these improvements, one can ask several questions:

1) At each iteration, an analysis is performed instead of embedding the state equations as equality constraints. This analysis provides the data for the objective function and derivative of the objective function. Bur are full analyses needed when one is far from the optimal solution?
2) Continuation between optimization problems is used to generate good initial guesses for the final step. What then is a good initial guess? How good should the initial guesses be?

Each question motivates a different modification to our SQP method.

5.2 Technique-2

Motivated by the simultaneous method(Bark, 1996), we try to address the first question mentioned above. In the simultaneous method, the state equations are embedded as equality constraints and the optimization problem is solved directly in the form of equation (8) and (9). Such an approach will not be feasible here, since the discrete Navier-Stokes equations will number in tens of thousands, and solving the QP subproblem is extremely difficult. However, it is instructive to observe the behavior of this so-called simultaneous method, for it suggests a remedy even if we eliminate the flow equations at each control iteration. The philosophy of the simultaneous method is that the constraints (i.e. the flow equations) need not be satisfied when we are far from the optimum. This suggests the following idea: Far from the optimum, there is no need to perform a full analysis. The convergence rate is only linear away from the neighborhood of the optimum, so provided convergence is not compromised, and it's convergence rate will not deteriorate. This forms the basic of Technique-2, which also includes the features of *T1-NA-SQP*. We refer this technique as *T2-NA-SQP*.

Therefore, we begin with a coarse(say 10^{-1}) convergence criterion for the discrete Navier-Stokes equations, and decrease it in proportion to the optimality condition of the optimization problem. Thus, for points far from the optimal solution, where the optimality condition is far from zero, a roughly-converged flow solution is acceptable. As the optimum approaches, the optimality condition decrease, and with it so does the convergence criterion for the flow problem, until the optimality termination condition 10^{-7} is reached. Figure 3 shows the nesting of iteration of this technique. The steps are summarized below:

1) given Reynolds number, solve optimization problem as follows:
   (a) use coarse convergence criterion for optimality condition (γ) and residual of analysis problem(ϵ) : $\gamma = \epsilon = \delta$(say 10^{-1})
   (b) solve optimization problem
   (c) if both final output value of KT(Kuhn-Tucker optimality criterion) and $\|\mathbf{h}\|$ are less than target convergence criterion(say 10^{-7}), then terminate (Go to step 2); otherwise decrease δ : for example, $\delta = 10^{-1} \times$max(final output values of KT, $\|\mathbf{h}\|$ of the previous step) set $\gamma = \epsilon = \delta$(d) go to (b)
2) Increment Reynolds number.
3) Repeat above until target Reynolds number is reached.

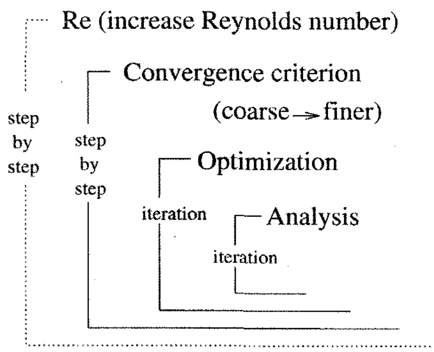

Figure 3 Algorithm of T2-NA-SQP

This technique looks like another continuation technique on *convergence condition*. In continuation on Reynolds number, the increment of Reynolds number was given in advance, but in continuation on convergence condition, the current convergence criterion depends on the final output values of KT and $\|\mathbf{h}\|$ of the previous convergence step(not previous step's convergence criterion because final output values of KT and $\|\mathbf{h}\|$ are usually less than convergence criterion).

5.3 Technique-3

T2-NA-SQP use a continuation technique on the optimization iteration. Essentially this technique is used to avoid the need to completely converge flow equations far from the optimum. The same idea can be used to avoid complete solution of optimization problem that arises at each Reynolds step. This suggests following idea : Each of the sequence of optimization problem may not have to be solved to exact optimality far from the target Reynolds number, since the function of each is simply to provide a goos initial guess for the next in the sequence.

Therefore, in Technique-3 we begin with a coarse convergence criterion(say 10^{-1}) for the optimization problem as well as for the discrete Navier-Stokes equations, except for the final step. This convergence criterion(say 10^{-1}) provides quite good results, but it reduces the number of control iteration and the analysis iteration significantly, especially for higher Reynolds number problems. Only in the final step, we use a fine(say 10^{-7}) convergence criterion. We refer this technique as *T3-NA-SQP*. Note that *T3-NA-SQP* also incorporates the features of *T2-NA-SQP*.

With these four improvements, we have been able to reduce by a factor of more than 15 comparing with time consumed by *OA-SQP* for the *Re*=400 problem with step size *Re*=50.

We will show these results in the following papers.

6. Conclusion

Up to now, analysis and optimization to flow problem has been regarded as very difficult matter. This study attempted to employ optimization technique to flow problem. For the purpose, we developed a new algorithm to improve convergency of optimization and three other techniques to increase the effectiveness of the algorithm.

First of all, step by step method in optimization process was employed to improve the convergency. In addition, following techniques were used to improve convergence rate, i.e., techniques of furnishing good initial guesses for analysis using sensitivity information acquired from optimization iteration, and of manipulating optimal convergency criterion motivated from simultaneous technique.

It is anticipated that new algorithm and techniques introduced in this paper will enable optimization to be utilized in solving two and even three-dimensional problems. Effectiveness of this model will be verified in the next series of papers.

References

박재형, 이리형 (1996) 구조체의 위상학적 최적화를 위한 비선형 프로그래밍, 한국전산구조공학회논문집, 9(3), pp.169~178.

박재형, 홍순조 (2002) Navier-Stokes 유체의 최적 제어, 한국전산구조공학회논문집, 15(4), pp.661~674.

박재형, 홍순조 (2002), Navier-Stokes 유체의 최적제어를 위한 SQP기법의 개발, 한국전산구조공학회논문집, 15(4), pp.675~691.

박재형, 홍순조, 이리형 (1996) Multiple-loading condition을 고려한 구조체의 위상학적 최적화, 한국전산구조공학회논문집, 9(3), pp.179~186.

Batchelor, G.K. (1967) *An introduction to Fluid Dynamics*, Cambridge Univ. Press, Cambridge, UK.

Biegler, L.T., Nocedal, J., Schmid, C. (1995) A reduced Hessian method for large-scale constrained optimization, *SIAM. J. Optim.* 5, 314.

Cramer, E.J., Dennis, J.E., Frank, P.D., Lewis, R.M., Shubin, G.R. (1994) Problem formulation for multidisciplinary optimization, *SIAM J. Optim.* 4, 754.

Davis, T.A. (1993) *Users' Guide for the Unsymmetric Pattern Multifrontal Package (UMFPACK)*, Tech. Rep. TR-93-020, CIS Dept., University of Florida, Gainesville, FL.

Davis, T.A., Duff, I.S. (1993) *An Unsymmetic Pattern Multifrontal Method for Sparse LU Factorization*, Tech. Rep. TR-93-018, CIS Dep., University of Florida, Gainesville, FL.

Farthat, C., Chen, P.S. (1994) Tailoring domain decomposition methods for efficient parallel coarse grid solution and for system with many right hand sides, in Domain Decomposition Methods in Science and Engineering, *Contemporary Mathematics*, 180, *American Mathematical Society, Providence*, RI.

Gabay, D. (1982) Reduced *Quasi-Newton methods with feasibility improvement for nonlinearly constrained optimization*, Math, Programming Study 16, 18.

Gad-el-Hak, M. (1989) Flow control, *Appl. Mech.* Rev. 42, 261.

Ghattas, O., Orozco, C.E. (1997) A parallel reduced Hessian SQP method for shape optimization, in Multidisciplinary Design Optimization, *State-of-the-Art, edited by N. Alexandrov and M. Hussaini*, p.133, SIAM, Philadelphia.

Gill, P.E., Murray, W., Wright, M.H. (1981) *Practical Optimization*, Academic Press, New York

Gunzburger, M.D. (1989) *Finite Element Methods for Viscous Incompressible Flow*, Academic Press, San Diego.

Gunzburger, M.D. (1995) Flow Control, *IMA Volumes in Mathematics and Its Applications*, Vol. 68, Springer-Verlag, Berlin/New York.

Gunzburger, M.D., Hou, L.S., Svobodny, T.P. (1993) *Optimal control and optimization of viscous, incompressible flows, in Incompressible Computational Fluid Dynamics*, edited by M.D. Gunzburger and R.A, Nicolaides, Cambridge Univ. Press, Cambridge, UK.

Haftka, R.T., Gurdal, Z., Kamat, M.P. (1990) *Elements of Structural Optimization*, Kluwer Academic, Dordercht/Norwall, MA.

Moin, P., Bewley, T. (1994) Feedback control of turbulence, *App. Mech. Rev.* 47,s3.

Prandtl, L., Tietjens,, O.G. (1934) *Applied Hydro-and Aeromechanics*, Dover, New York, p.81.

Xie, Y. (1991) Reduced Hessian Algorithms for solving Large-Scale Equality Constrained Optimization Problem, Ph. D. thesis, University of Cololado, Boulder, Department of Computer Science.