Adaptive Multi-Meshes in Finite Element Approximation of Optimal Control

W. B. Liu

Abstract. We present some recent developments and issues on adaptive multi-mesh finite element method, and its applications to computational optimal control. Particularly, we discuss several difficult issues on finite element schemes with adaptive multi-set meshes for parabolic optimal control.

1. Introduction

Solutions of practical problems may involve several components which have different natures or singularities. For example for the 2D Navier-Stokes equations in vorticity and streamfunction formulation, the stream-function is in general smoother than the vorticity. Another example is the following elliptic system:

$$\begin{aligned} -\nabla\left(A^0\nabla u^0\right)+\sum_{i=0}^{1}c_i^0u^i &= f^0 \\ -\nabla\left(A^1\nabla u^1\right)+\sum_{i=0}^{1}c_i^1u^i &= f^1 \end{aligned} \tag{1.1}$$

on a two dimensional domain Ω with homogeneous Dirichlet boundary condition:

$$u^i\big|_{\partial\Omega}=0, \qquad i=0,1, \tag{1.2}$$

where the coefficient matrices A^0 and A^1 are symmetric positive definite but have different locations of singularities. Figures 1-2 show an example of such cases, where meshes adapted for two solutions are quite different, see [**10**] for the details.

Multi-mesh strategies proved particularly useful in computational optimal control, where the regularities of the optimal control and the state are often very different. Furthermore the locations of their singularities are very different. This indicates that the current all-in-one mesh strategy may be inefficient. In fact, it was shown in [**7**] that adaptive multi-meshes; that is, separate adaptive meshes which are adjusted according to different error indicators, are often necessary, and can indeed greatly reduce computational work in solving optimal control problems. Figures 3-5 show a typical case of constrained optimal control, where the locations and natures of their singularities are very different, see [**7**] for the details. It is clear

Key words and phrases. optimal control problems, finite element approximation, adaptive finite element methods, a posteriori error estimates, multi-meshes.

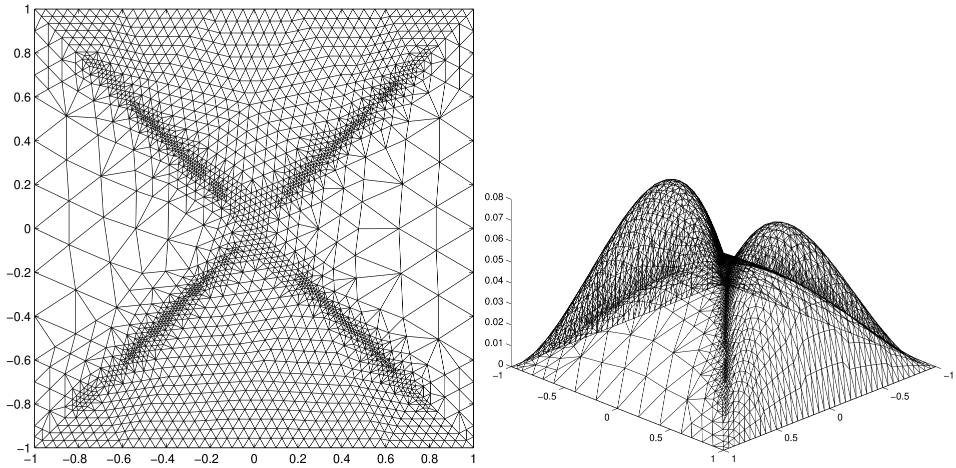

FIGURE 1. Weakly coupled elliptical system: mesh (left) and solution (right) for u_h^0.

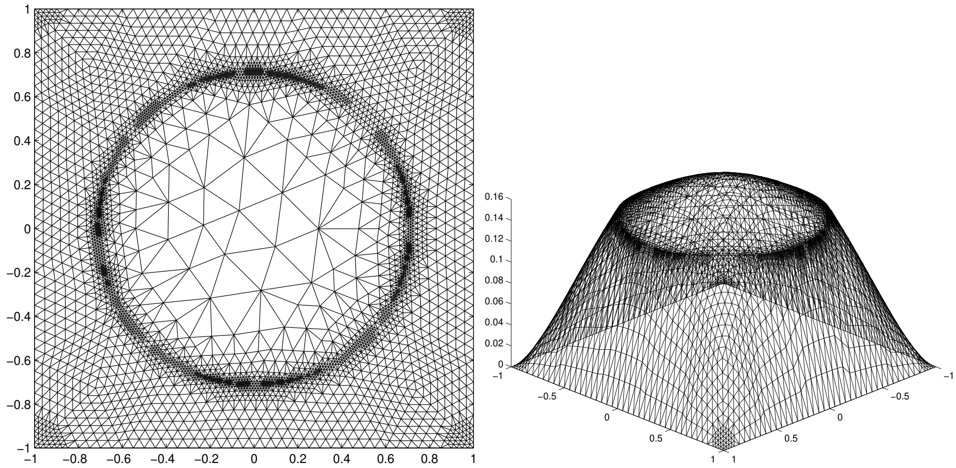

FIGURE 2. Weakly coupled elliptical system: mesh (left) and surface (right) for u_h^1.

that significant computational efficiency may be lost if a single adaptive mesh is used.

Most of the optimal control problems that we are interested in can be symbolically written in the following form: (OCP)

$$\min_{u \in U_{ad}} J(y, u)$$

s.t. (state equation)

$$A(y, u) = 0,$$

where J is the objective functional; y is the state variable in a state space Y; u is the control variable in a control space U; U_{ad} is the admissible set of the control (design); and A is a suitable operator on $Y \times U_{ad}$. Here the state equation

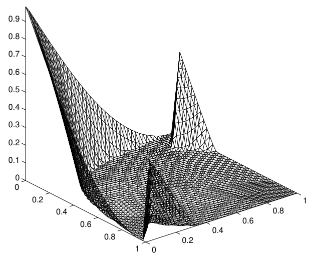

FIGURE 3. Profile of the optimal control u

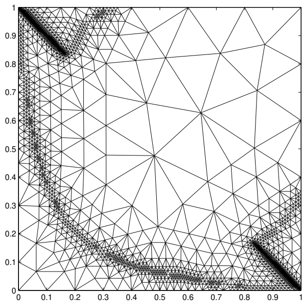

FIGURE 4. Adaptive Mesh Suit Control u

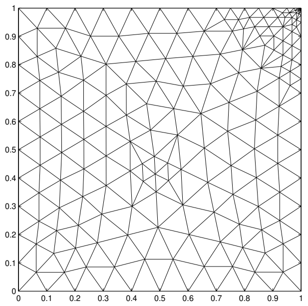

FIGURE 5. Adaptive Mesh Suit States y and p

$A(y,u) = 0$ can be a PDE, or a variational inequality, etc., and may incorporate state constraints. Sometimes, the control variable is called design variable, the control space is referred to as design space, and the state variable may be referred to as system variable. The literature on (OCP) is too huge to be mentioned here even briefly.

Let Ω (Ω_U) be a bounded open set in R^n $(n \leq 3)$ with a Lipschitz boundary $\partial\Omega$ $(\partial\Omega_U)$. We shall take the state space $V = H_0^1(\Omega)$ and the control space $U = L^2(\Omega_U)$. Let the observation space $Y = L^2(\Omega)$, $H = L^2(\Omega)$.

Let B be a linear continuous operator from U to H, let K be a convex subset of U. Assume that g and h are strictly convex functionals which are differentiable on $H = L^2(\Omega)$, $h(u) \to +\infty$ as $\|u\|_U \to \infty$, and that $g(\cdot)$ is bounded below. We take the following typical control problem to proceed.

EXAMPLE 1.1. *Consider*

$$\min_{u\in K}\{g(y)+h(u)\},$$

$$-div(A\nabla y) = f + Bu, \quad in\ \Omega,\ y|_{\partial\Omega} = 0.$$

The weak formula: Let

$$a(y,w) = \int_\Omega (A\nabla y)\cdot\nabla w, \quad \forall y,w\in V,$$

$$(u,v)_U = \int_{\Omega_U} uv, \quad \forall (u,v)\in U\times U.$$

Therefore the control problem can be restated as (QCP):

$$\min_{u\in K}\{g(y)+h(u)\}\ , \tag{1.3}$$

$$a(y(u),w) = (f+Bu,w), \quad \forall w\in H_0^1(\Omega), \tag{1.4}$$

(y^*,u^*) *is the solution iff there is a co-state* $p^*\in V$ *such that* (y^*,p^*,u^*) *satisfies the following optimality conditions: (OPT)*

$$a(y^*,w) = (f+Bu^*,w) \qquad \forall w\in V = H_0^1(\Omega), \tag{1.5}$$

$$a(q,p^*) = (g'(y^*),q) \qquad \forall q\in V = H_0^1(\Omega), \tag{1.6}$$

$$(h'(u^*)+B^*p^*, v-u^*)_U \geq 0 \qquad \forall v\in K\subset U = L^2(\Omega_U), \tag{1.7}$$

It is clear that for constrained control problems, the control singularity is due to that of initial data, or/and a level set change of the co-state. For instance, for a typical quadratic control problem with $g(y)=\int_\Omega (y-y_0)^2/2,\ h(u)=\int_{\Omega_U} u^2/2$, $B=I$ and $K=\{u: u\geq 0\}$, it follows from (QCP-OPT) that

$$u^* = \max(0,-p^*)$$

So u^* may have gradient jumps along the free boundary of the level set of the co-state $p^*=0$, though the state or co-state usually are quite regular. Consequently even p^*, y^* are very smooth, ∇u^* may have discontinuity. The locations of computational difficulties for the control and for the state are in general very different in **constrained** control problems.

These can be clearly seen from Figs (3-5). Thus different adaptive meshes should be used for the control and the states in order to produce most efficient computational meshes. Using different adaptive meshes for the control and the state allows very coarse meshes to be used in solving the state and co-state equations. Thus much computational work can be saved since one of the major computational loads is to solve the state and co-state equations repeatedly. This can be clearly seen from our numerical experiments, see [**7**] for the details.

There are several key ingredients in realizing the above idea. Firstly sharp error indicators have to be derived allowing different computational meshes to be used for the control and the states. In [**11**], sharp error indicators were derived for constrained linear elliptic control problems, and implemented in adaptive finite element schemes with impressive initial numerical results. However in the indicators derived in [**11**], one needs to compute the term $\nabla(h'(u_h)+B^*p_h)|_{\tau_U}$, and it is further assumed to be a polynomial on τ_U (Theorem 3.3). These conditions are easy to meet if only one set of meshes is to be used, but become too restrictive for a multi-mesh scheme. In [**7**], we derived different a posteriori error estimators for optimal control governed by elliptic systems , which are equivalent to the discretised error and do not require these conditions. In [**15**] and [**16**], sharp a posteriori error estimates were derived for optimal control of stokes equations and parabolic equations on separate meshes. However there are still some outstanding issues in this area, which will be discussed in Section 3.

Although the idea of using multi-meshes seems to be very natural, it has been considered very expensive to implement in practical computations. A good implementation has to be very efficient as otherwise the benefits brought by this strategy can easily be consumed by the extra work introduced. Particularly one has to compute numerical integrals over different (maybe overlapped) domains, and this can easily consume much computational work. We have built a C++ adaptive finite element Lib: AFEPACK, which can efficiently generate multi-meshes, base functions, and stiff matrices, see Section 3 for more details.

The above issues become even more complex when the control problems are time-dependent. In computing numerical solutions for a parabolic equation, time matching schemes like the backward Euler scheme are mostly used. The main advantage of using the time matching scheme in solving parabolic equations is that one can then solve them step by step in time. However that also means that it is impossible to adjust meshes locally in x-t space. Consequently an adaptive time matching scheme may not be able to properly handle the singularities that have complicated geometric distribution in x-t space. This is not so serious in solving a parabolic equation as the singularity of its solution normally has the evolutional nature. However this is not the case for parabolic control problem. Due to the global property in time of the control problem there may be less important to use time matching discretization. Thus we propose a new scheme which approximates the state and the control by the finite element method with an isotropic treatment of the variables x and t. In our numerical tests, we found that for a given error tolerance, the computational work on solving the full linear systems generated from the new scheme is much smaller than that on solving the backward Euler systems, unless the optimal control is smooth. Thus the above approaches are efficient for their own territories- either very smooth cases or nonsmooth cases. The ideal combination seems to apply finite element approximation to the control u both in space and time with an isotropic treatment of the variables x and t, but use the usual time-stepping method for the time discretization of the state equation, which is governed by the parabolic equation. This again essentially involves multi-mesh issues, and its implementation is still an open question.

2. Numerical Solutions of Optimal Control

To solve the control problems numerically, one needs to approximate this infinite dimensional optimisation problem with a sequence of finite dimensional ones. Because of the complex geometric configuration of engineering problems, finite element discretization of the state equation is widely applied. In such discretization, the feasible domains are first to be subdivided into sub-domains (or finite elements). Then the state variable y and control variable u are to be approximated by some piecewise polynomials y_h and u_h on the domains, where h is maximum size of the finite elements, see Section 3 for the details.

Then one obtains the following finite dimensional optimisation problem: $(OCP)_h$

$$\min_{u_h \in U_{ad}^h} J_h(y_h, u_h)$$

s.t.

$$A_h(y_h, u_h) = 0.$$

This is a finite dimensional optimization problem and may be solved by existing mathematical programming methods such as the steepest descent method, conjugate gradient method, trust domain method, and SQP, see [**1**], [**4**], [**6**], [**8**], and [**9**], etc. The standard FEM approximation and a priori error estimation were much studied in 70-80's, typically with optimization algorithms of gradient type. In fact, these algorithms are still widely used in engineering design work as they are easy to combine with sensitivity analysis and existing solvers.

There are (at least) three key steps in solving an optimal control problem:

$$Discretisation\ \ Of\ \ The\ \ Optimal\ \ Control$$

$$State\ \ Equation - A_h(y_h, u_h) = 0$$

$$Mathematical\ \ Programming - minJ(y_h, u_h)$$

The second and third steps have been the focus points of intensive studies.

There are two main approaches in solving $(OCP)_h$:

a) Black-box approach - studying $J(y_h(u_h), u_h)$, while y_h is computed from the state equation. With an algorithm of gradient type, it utilises existing packages easily. In fact, it is still widely used in engineering computing. Engineers use two packages: one for state-costate equations (for computing gradient) and one for optimization.

b) All in one approach - viewing (u, y) to be independent variables in the mathematical programming problems. Thus the state equation $A_h(y_h, u_h) = 0$ may not be always solved accurately. For instance, one can directly discretise the optimality conditions, and solve them with semi-smooth Newton's method.

For many problems, this approach leads to the same numerical schemes as a). However this is not always true. Also there are other approaches like Least Square Method.

3. Adaptive Approximations for steady Control

As seen in Section 1, for constrained control problems the states are often smooth. The control u^* typically has jumps in gradient, which are determined by level sets of the co-state. Thus an efficient computational mesh for the control should have a higher density of nodes around the free boundary.

It has been recently found that suitable adaptive meshes can greatly reduce discretisation errors, see [**2**], [**3**], [**7**], and [**11**]. If the computational meshes are not properly generated, then there may be large error around the singularities of the control, which cannot be removed later on. Adaptive finite element approximation uses *a posteriori* error indicator to guide the mesh refinement procedure. Only the area where the error indicator is larger will be refined so that a higher density of nodes is distributed over the area where the solution is difficult to approximate. In this sense efficiency and reliability of adaptive finite element approximation rely very much on the error indicator used. Thus the first key question is how to obtain correct error estimators.

3.1. A posteriori error estimators. To fix the idea, we examine several existing approaches using the same example (QCP):

$$\min_{u\in K}\{g(y)+h(u)\}\ , \tag{3.1}$$

$$a(y(u),w)=(f+Bu,w), \qquad \forall w\in V=H_0^1(\Omega). \tag{3.2}$$

As in [**5**], let Ω^h be a polygonal approximation to Ω with a boundary $\partial\Omega^h$. For simplicity, we assume that $\Omega^h=\Omega$ in this paper. Let T^h be a partitioning of Ω^h into disjoint regular n-simplices τ, so that $\bar\Omega^h=\bigcup_{\tau\in T^h}\bar\tau$. Each element has at most one face on $\partial\Omega^h$, and $\bar\tau$ and $\bar\tau'$ have either only one common vertex or a whole edge or face if τ and $\tau'\in T^h$.

Associated with T^h is a finite dimensional subspace S^h of $C(\bar\Omega^h)$, such that $\chi|_\tau$ are polynomials of m-order $(m\ge 1)$ $\forall\chi\in S^h$ and $\tau\in T^h$. Let $V^h=S^h\cap H_0^1(\Omega)$. It is easy to see that $V^h\subset V$.

Let T_U^h be a partitioning of Ω_U^h into disjoint regular n-simplices τ_U, so that $\bar\Omega_U^h=\bigcup_{\tau_U\in T_U^h}\bar\tau_U$. Assume that $\Omega_U^h=\Omega_U$, and $\bar\tau_U$ and $\bar\tau_U'$ have either only one common vertex or a whole face or are disjoint if τ_U and $\tau_U'\in T_U^h$.

Associated with T_U^h is another finite dimensional subspace W_U^h of $L^2(\Omega_U^h)$, such that $\chi|_{\tau_U}$ are polynomials of m-order $(m\ge 0)$ $\forall\chi\in W_U^h$ and $\tau_U\in T_U^h$. An optimal control of a constrained problem normally has lower regularity so that we shall use discontinuous base functions to approximate the control. Hence there is no requirement for continuity of the functions in W_U^h. Let $U^h=W_U^h$. Let h_τ (h_{τ_U}) denote the maximum diameter of the element τ (τ_U) in T^h (T_U^h).

Then a possible finite element approximation of (QCP), which we shall label $(QCP)^h$, is as follows:

$$(QCP)^h:\ \begin{cases}\min_{u_h\in K^h\subset U^h}\{g(y_h)+h(u_h)\}\ ,\\[2ex] a(y_h,w_h)=(f+Bu_h,w_h)\qquad \forall w_h\in V^h,\end{cases} \tag{3.3}$$

where K^h is a closed convex set in U^h. It follows that the control problem $(QCP)^h$ has a solution (y_h^*,u_h^*) and that if a pair $(y_h^*,u_h^*)\in V^h\times U^h$ is the solution of $(QCP)^h$, then there is a co-state $p_h^*\in V^h$ such that the triplet (y_h^*,p_h^*,u_h^*) satisfies the following optimality conditions, which we shall label $(QCP-OPT)^h$:

$$(QCP-OPT)^h:\ \begin{cases}a(y_h^*,w_h)=(f+Bu_h^*,w_h)\qquad \forall w_h\in V^h\subset V=H_0^1(\Omega),\\[2ex] a(q_h,p_h^*)=(g'(y_h^*),q_h)\qquad \forall q_h\in V^h\subset V=H_0^1(\Omega),\\[2ex] (h'(u_h^*)+B^*p_h^*,v_h-u_h^*)_U\ge 0\quad \forall v_h\in K^h\subset U=L^2(\Omega_U).\end{cases} \tag{3.4}$$

Until late 90s', the most widely used approach in engineering is to use the decoupled (for a fixed u_h) the state equations to derive a posteriori error estimators from the error equations

$$Ay=f+Bu,\quad A_hy_h=f_h+B_hu_h$$

in the mesh refinement. Clearly this approach may be very inefficient, as mentioned before, the state and the control may have very different natures and properties. Taking a boundary control problem as an example, it is easy to understand this strategy could seriously over-refine the interior of the domain, since the objective functional is only defined on the boundary. Even for a distributed quadratic control problem with very smooth state, the control may have singularity - as seen before. Thus the error indicators from approximation error of the state often failed to detect the singularity .

However it is not straightforward to derive suitable *a posteriori* error estimators for general optimal control problems. It is clear that the optimality conditions of the above control problem are two equations coupled with a variational inequality. Only in late 90s, some a posteriori error indicators have been derived for optimal control or objective functional by, e.g., Becker and Rannacher, and Liu and Yan.

Let us briefly discuss some approaches currently used, again using the above simple control problem as example. Let us first restate the optimality conditions: (OPT): there is a co-state $p \in V$ such that the triplet (y, p, u) satisfies

$$a(y, w) = (f + Bu, w) \qquad \forall w \in V = H_0^1(\Omega), \tag{3.5}$$

$$a(q, p) = (g'(y), q) \qquad \forall q \in V = H_0^1(\Omega), \tag{3.6}$$

$$(h'(u) + B^*p, v - u)_U \geq 0 \qquad \forall v \in K \subset U = L^2(\Omega_U). \tag{3.7}$$

(A) Goal orientated approach: When $U_{ad} = K = U$, one has $h'(u) + B^*p = 0$ and then the above coupled system becomes an elliptic system. Then the optimal control problem is to compute:

$$\min J(y, p) : \quad D(y, p) = 0.$$

Then one can apply the ajoint approach by utilising the adjoint equation $D^*(y, p) = 0$. The weighted a posteriori error estimation technique are applicable - see [**2**] and [**3**] to obtain a posteriori estimates for objective functional approximation error. It has the distinctive feature that the error indicators are derived for the objective functional.

(B) Residual type error indicators for convex control: In this case some techniques have been been developed by us to derive a posteriori error estimators for the control and the states. The main idea is to first obtain a posteriori error estimates for control: $\|u - u_h\| \leq C(\eta_1^2 + \|p(u_h) - p_h\|)$, where $p(u_h)$ is an intermediated variable, and then to estimate $\|p(u_h) - p_h\|$. A posteriori error estimates have been obtained for the control and state of distributed and boundary control problems, -see [**15**]-[**16**]. This approach is also applicable for evolutional control.

In the follows we illustrate some essential ideas to derive error indicators of residual type for $(QCP) - (QCP)^h$. In order to derive sharp a posteriori error estimates, we divide Ω_U into some subsets:

$$\Omega_d^- = \{x \in \Omega_U : \ (B^*p_h^*)(x) \leq -h'(d)\},$$

$$\Omega_d = \{x \in \Omega_U : \ (B^*p_h^*)(x) > -h'(d), \ u_h^* = d\},$$

$$\Omega_d^+ = \{x \in \Omega_U : \ (B^*p_h^*)(x) > -h'(d), \ u_h^* > d\}.$$

Then, it is easy to see that above three subsets are not intersected each other, and

$$\bar{\Omega}_U = \bar{\Omega}_d^- \cup \bar{\Omega}_d \cup \bar{\Omega}_d^+.$$

Now let us have an intuitive analysis on the approximation error for the control. On Ω_d, asymptotically we can assume that

$$0 < B^*p_h^*(x) + h'(u_h^*) \to B^*p^*(x) + h'(u^*).$$

Hence it follows from the optimality conditions that $u^* = u_h^* = d$ on Ω_d. Thus the error on Ω_d may be negligible. We should only need to estimate the error on

$$\Omega_U \setminus \Omega_d = \Omega_d^- \cup \Omega_d^+.$$

in order to avoid over-estimate. Further analysis shows that the error over Ω_d^+ can actually be bounded by that over Ω_d^-. Therefore we only need to estimate the error on Ω_d^-, and this leads to the following results (for the details, see [**7**]):

THEOREM 3.1. *Let (y^*, u^*) and (y_h^*, u_h^*) be the solutions of* (QCP) *and* $(QCP)^h$. *Let p^* and p_h^* be the solutions of the co-state equation and the discretised co-state equation respectively. Moreover, assume that g' and h' are locally Lipschitz continuous in a neighborhood of y^* and u^*. Then,*

$$\|u^* - u_h^*\|^2_{L^2(\Omega_U)} + \|y^* - y_h^*\|^2_{H^1(\Omega)} + \|p^* - p_h^*\|^2_{H^1(\Omega)} \leq C \sum_{i=1}^{3} \underline{\eta}_i^2, \tag{3.8}$$

where

$$\underline{\eta}_1^2 = \int_{\Omega_d^-} |h'(u_h^*) + B^* p_h^*|^2,$$

$$\underline{\eta}_2^2 = \sum_{\tau \in T^h} \int_\tau h_\tau^2 (div(A^* \nabla p_h^*) + g'(y_h^*))^2 + \sum_{l \cap \partial\Omega = \emptyset} \int_l h_l [(A^* \nabla p_h^* \cdot n)]^2,$$

$$\underline{\eta}_3^2 = \sum_{\tau \in T^h} \int_\tau h_\tau^2 (f + Bu_h^* + div(A \nabla y_h^*))^2 + \sum_{l \cap \partial\Omega = \emptyset} \int_l h_l [(A \nabla y_h^* \cdot n)]^2,$$

where l is a face of an element τ, $[(A^* \nabla p_h^* \cdot n)]$ and $[(A \nabla y_h^* \cdot n)]$ are the A-normal derivative jumps over the interior face l, defined by

$$[(A^* \nabla p_h^* \cdot n)]_l = (A^* \nabla p_h^*|_{\tau_l^1} - A^* \nabla p_h^*|_{\tau_l^2}) \cdot n,$$

$$[(A \nabla y_h^* \cdot n)]_l = (A \nabla y_h^*|_{\tau_l^1} - A \nabla y_h^*|_{\tau_l^2}) \cdot n,$$

where n is the unit normal vector on $l = \bar{\tau}_l^1 \cap \bar{\tau}_l^2$ outwards τ_l^1, h_l is the maximum diameter of the face l.

(C) Recovery type error indicators for convex control. In [**14**], we provide a posteriori error estimates of recovery type for both the control and the state approximation, which are generally equivalent. Under some stronger assumptions, they are further shown to be asymptotically exact.

For all the above approaches, it is possible to derive a posteriori error estimates using different meshes for the state and the control. In fact, our error estimators are always presented in this setting. Some computational benefits can be seen from [**7**].

The following tables describe a typical computational example, which is a quadratic control problem. The details can be found in [**7**]. In the first adaptive finite element approximation, the same mesh is used for the control and the states with the error indicator $\underline{\eta}^2 = \underline{\eta}_1^2 + \underline{\eta}_2^2 + \underline{\eta}_3^2$. The mesh information and numerical results are presented in Table 2. As the free boundary is rather complicated in this example, even adaptive meshes may not greatly reduce the numbers of u-freedoms. This however forces large numbers of y-p-freedoms to be used, as they share the same mesh, as seen in Table 2. Furthermore we compute this example on adaptive multi-meshes which are produced separately for the control and for the states. It was found that these adaptive meshes can further reduce more than 10 times of the

		y,p piecewise linear			y,p piecewise quadratic		
		u	y	p	u	y	p
mesh info	# nodes	8097	8097	8097	8097	8097	8097
	# sides	23968	23968	23968	23968	23968	23968
	# elements	15872	15872	15872	15872	15872	15872
	# dofs	47616	8097	8097	47616	32065	32065
L^2 error		3.5511e-2	4.3790e-4	5.9413e-6	3.5511e-2	4.8599e-5	2.0869e-6

TABLE 1. Uniform mesh, u is piecewise linear discritized

		y,p piecewise linear			y,p piecewise quadratic		
		u	y	p	u	y	p
mesh info	# nodes	4176	4176	4176	4176	4176	4176
	# sides	12055	12055	12055	12055	12055	12055
	# elements	7880	7880	7880	7880	7880	7880
	# dofs	23640	4176	4176	23640	16539	16539
L^2 error		3.5736e-2	4.0268e-2	7.4660e-4	3.5735e-2	5.4475e-4	4.3194e-6

TABLE 2. u, y and p on the same adaptive mesh, u is piecewise linear discritized

		y,p piecewise linear			y,p piecewise quadratic		
		u	y	p	u	y	p
mesh info	# nodes	4473	341	341	4486	132	132
	# sides	13006	933	933	13046	340	340
	# elements	8534	593	593	8561	209	209
	# dofs	25602	341	341	25683	497	497
L^2 error		3.5534e-2	1.7884e-2	1.9891e-4	3.5533e-2	3.3854e-3	2.0201e-5

TABLE 3. (y,p) and u on separate adaptive meshes, u is piecewise linear discritized

dofs in all the state variables to produce a given L^2 control error reduction as shown in Table 3. Thus adaptive multi-meshes can indeed save substantial computational work.

To derive a posteriori error estimates of residual type, the system $(QCP-OPT)^h$ first has to have solutions. When y_h, p_h use the same meshes, this existence comes from the equivalence of $(QCP)^h$ and $(QCP-OPT)^h$. However the equivalence breaks down if they use different meshes. Let us consider the case where different meshes T_Y^h and T_P^h with the mesh sizes h_Y and h_P are used for the finite element approximations of the state y^* and the co-state p^*, so that we have the different finite element spaces V_Y^h and V_P^h for y^* and p^* respectively. Then, we may approximate the optimality conditions (OCP-OPT) by

$$a(y_h^*, w_h) = (f + Bu_h^*, w_h) \qquad \forall w_h \in V_Y^h \subset V = H_0^1(\Omega), \tag{3.9}$$

$$a(q_h, p_h^*) = (g'(y_h^*), q_h) \qquad \forall q_h \in V_P^h \subset V = H_0^1(\Omega), \tag{3.10}$$

$$(h'(u_h^*) + B^*p_h^*, v_h - u_h^*)_U \geq 0 \quad \forall v_h \in K^h \subset U = L^2(\Omega_U). \tag{3.11}$$

As long as the system (3.9)-(3.11) has a triple solution (y_h^*, p_h^*, u_h^*), we can derive similar error estimates to Theorem 3.1. However this existence is still an open problem. Similar difficulties are met in flow control, where different discretisations for (y_h,p_h) are needed for stabilizing.

3.2. Multi meshes strategy. As seen above, successful implementation of the adaptive mesh strategy can enhance the accuracy of the numerical approximations and also reduce the computational cost. Over the past two decades, several very efficient adaptive techniques for the numerical solution of partial differential equations have been developed. The existing adaptive libraries include ALBERT, deal.II and UG. These libraries are widely used. It is noted that the libraries mentioned above have only considered mesh adaptation on a single mesh. As mentioned before although the idea of using multi-meshes seems to be very natural, it has been considered very expensive to implement in practical computations. A good implementation has to be very efficient as otherwise the benefits brought by this strategy can easily be consumed by the extra work introduced. However there seems no the best way to implement adaptive multi meshes.

The implementation issues have been discussed in [**10**]. The follows give a summary: "The key problem in implementing adaptive multi-meshes efficiently is to build-up the relationship among the different meshes used in order to compute the quantities like numerical integrations involving functions on different meshes. There seem two possible ways to do this: The first one is to build up an inter-relationship data structure beforehand and the second one is to integrate all the quantities to a much finer reference mesh. If both meshes are almost uniform, or one of the two meshes is extremely coarse, both ways will be feasible. Unfortunately, this is not the case for our problem. A satisfactory mesh to approximate a function with a local singularity will generally locate most of its grid nodes to a small part of the domain. In particular, the size of the largest element is often hundreds of times bigger than that of the smallest one. Such irregular mesh structure makes it extremely costly to establish useful inter-mesh relationship. The main reason is that almost every element of one mesh has common parts with almost every element of the other mesh. As a result, we cannot save much by using such data structure. The second method above is also expensive either, since the reference mesh is generally a very fine uniform one which is finer than both solution meshes at every part of the domain."

Thus it is very costly for arbitrary two meshes to build up useful relationship between them. To construct efficient adaptive multi-mesh algorithms, the meshes must have some known relationship. If the relationship is too close or too loose, we either lose flexility of multi-mesh, or fail to implement it efficiently. Our choice is the *hierarchy geometry tree* or hierarchical grids. By hierarchical we mean that the structures of the grid are described hierarchically. The hierarchical data structure serves the simplicity of programming and allows fast algorithms for grid refinements. The details can be found in [**10**]. We have developed a software package AFEPack to do this job. This package is freely available at our website.

4. Control by Parabolic Equations

In rest of the paper, we consider the control problem governed by a parabolic equation:

$$\min_{u(t)\in K}\left\{\frac{1}{2}\int_0^T g(y)+h(u)dt\right\}\ ,$$

subject to

$$(y_t(u),w)+a(y(u),w)=(f+Bu,w),\quad t\in(0,T],\ \forall w\in V=H_0^1(\Omega),$$

$$y(u)(x,0) = y_0(x),$$

where g, h, B satisfy similar conditions as before. It can be shown that a pair (y^*, u^*) is the solution iff there is a co-state $p^* \in H^1(0,T;H) \cap W$ such that the triplet (y^*, p^*, u^*) satisfies the following optimality conditions for $t \in (0,T]$: (OPTE)

$$(y_t^*, w) + a(y^*, w) = (f + Bu^*, w), \forall w \in V = H_0^1(\Omega), y^*(x,0) = y_0(x), \tag{4.1}$$

$$-(p_t^*, q) + a(q, p^*) = (g'(y^*), q), \forall q \in V = H_0^1(\Omega),\ p^*(x,T) = 0, \tag{4.2}$$

$$(h'(u^*) + B^* p^*, v - u^*)_U \geq 0, \quad u^*(t) \in K,\ \forall v \in K \subset U, \tag{4.3}$$

where B^* is the adjoint operators of B. Clear it is no longer possible to solve the above system by time-matching.

Of course we still can solve the state equations by time-matching iterations. Thus we first consider the backward Euler scheme for the control problem.

4.1. Backward Euler Scheme in Time. Let $0 = t_0 < t_1 < \cdots < t_{N-1} < t_N = T$, $k_i = t_i - t_{i-1}$, $i = 1, 2, \cdots, N$, $k = \max_{i \in [1,N]}\{k_i\}$. For $i = 1, 2, \cdots, N$, construct the finite element spaces $V_i^h \in H_0^1(\Omega)$ with the mesh T_i^h.

Similarly, construct the finite element spaces $U_i^h \in L^2(\Omega_U)$ with the mesh $(T_U^h)_i$. Let h_{τ^i} $(h_{\tau_U^i})$ denote the maximum diameter of the element τ^i (τ_U^i) in T_i^h $((T_U^h)_i)$. Let $K_i^h \subset U_i^h \cap K$. The fully discrete approximation scheme $(QCP)^{hk}$ is to find $(y_h^i, u_h^i) \in V_i^h \times K_i^h$, $i = 1, 2, \cdots, N$, such that

$$\min_{u_h^i \in K_i^h} \left\{ \sum_{i=1}^{N} k_i (g(y_h^i) + h(u_h^i)) \right\},$$

$$\left(\frac{y_h^i - y_h^{i-1}}{k_i}, w_h \right) + a(y_h^i, w_h) = (f(x, t_i) + Bu_h^i, w_h),$$

$$\forall w_h \in V_i^h \subset V = H_0^1(\Omega), \ \ i = 1, \cdots, N,\ y_h^0(x) = y_0^h(x), \ \ x \in \Omega.$$

$(Y_h^i, U_h^i) \in V_i^h \times K_i^h$, $i = 1, 2, \cdots, N$, is the solution iff there is a co-state $P_h^{i-1} \in V_i^h$, $i = 1, 2, \cdots, N$, such that $(Y_h^i, P_h^{i-1}, U_h^i)$, satisfies the following optimality conditions: $(QCP - OPT)^{hk}$

$$\left(\frac{Y_h^i - Y_h^{i-1}}{k_i}, w_h \right) + a(Y_h^i, w_h) = (f(x, t_i) + BU_h^i, w_h), \tag{4.4}$$

$$\forall w_h \in V_i^h \subset V = H_0^1(\Omega), \ \ i = 1, \cdots, N,$$

$$Y_h^0(x) = y_0^h(x), \ \ x \in \Omega,$$

$$\left(\frac{P_h^{i-1} - P_h^i}{k_i}, q_h \right) + a(q_h, P_h^{i-1}) = (g'(Y_h^i), q_h), \tag{4.5}$$

$$\forall q_h \in V_i^h \subset V = H_0^1(\Omega), \ \ i = N, \cdots, 1,$$

$$P_h^N(x) = 0, \ \ x \in \Omega,$$

$$(h'(U_h^i) + B^* P_h^{i-1}, v_h - U_h^i)_U \geq 0 \quad U_h^i \in K_i^h,\ \forall v_h \in K_i^h \subset U_i^h \cap K. \tag{4.6}$$

This is again a finite dimensional optimisation problem, and may be solved by existing mathematical programming methods. such as Steepest Descent Method, Conjugate Gradient Method, Trust Domain Method, and SQP, etc. For $i = 1, 2, .., N$, let

$$\begin{aligned} Y_h|_{(t_{i-1},t_i]} &= ((t_i - t)Y_h^{i-1} + (t - t_{i-1})Y_h^i)/k_i, \\ P_h|_{(t_{i-1},t_i]} &= ((t_i - t)P_h^{i-1} + (t - t_{i-1})P_h^i)/k_i, \\ U_h|_{(t_{i-1},t_i]} &= U_h^i. \end{aligned}$$

It is much more complicated to apply adaptive finite element methods to parabolic control. Again the first step is to derive error indicators.

A Posteriori Error Estimators. We assume that the constraint on the control is an obstacle such that

$$K = \{v \in L^2(0, T; L^2(\Omega_U)) : \ v \geq \alpha, \ \ \text{a.e. in } \Omega_U \times (0, T]\},$$

where α is a constant. Define the contact set $\Omega_U^-(t)$ and the non-contact set $\Omega_U^+(t)$ as follows:

$$\Omega_U^-(t) := \{x \in \Omega_U : u(x, t) = \alpha\}, \qquad \Omega_U^+(t) := \{x \in \Omega_U : u(x, t) > \alpha\}.$$

Let

$$\begin{aligned} K_i^h &= \{v \in U_i^h : \ v \geq \alpha \text{ in } \Omega_U \times (t_{i-1}, t_i]\}, \\ K^h &= \{v : \ v|_{(t_{i-1},t_i]} \in K_i^h\}. \end{aligned}$$

Here we assume that

$$h(u) = \int_{\Omega_U} j(u),$$

where $j(\cdot)$ is a convex continuously differentiable function on R. To obtain sharp estimators, again we divide Ω_U into the following three subsets:

$$\begin{aligned} \Omega_-^i &= \{x \in \Omega_U : \ (B^* P_h)(x, t_{i-1}) \leq -j'(\alpha)\}, \\ \Omega_0^i &= \{x \in \Omega_U : \ (B^* P_h)(x, t_{i-1}) > -j'(\alpha), \ U_h^i = \alpha\}, \\ \Omega_+^i &= \{x \in \Omega_U : \ (B^* P_h)(x, t_{i-1}) > -j'(\alpha), \ U_h^i > \alpha\}. \end{aligned}$$

Similarly to the case of steady control, only error on Ω_-^i is important.

THEOREM 4.1. *Let (y, p, u) and (Y_h, P_h, U_h) be the solutions of the control problem and its approximation via the backward Euler scheme respectively. Moreover, assume that $j'(\cdot)$ and $g'(\cdot)$ are locally Lipschitz continuous. Then we have that*

$$\|u - U_h\|^2_{L^2(0,T;L^2(\Omega_U))} + \|y - Y_h\|^2_{L^2(Q)} + \|p - P_h\|^2_{L^2(Q)} \leq C(\eta_1^2 + \eta_2^2 + \eta_3^2) \tag{4.7}$$

where $Q = \Omega \times (0, T)$,

$$\eta_1^2 = \sum_{i=1}^N \int_{t_{i-1}}^{t_i} \int_{\Omega_-^i} (j'(U_h) + B^* \tilde{P}_h)^2,$$

and $\eta_2^2 + \eta_3^2$ consists of the standard residual a posteriori error indicators for the state and co-state equations.

See [**12**] and [**17**] for the details and the proofs. However it is clear that with the backward Euler scheme it is impossible to adjust meshes locally in x-t space. Often the time-refinement is more or less uniform. Soon computer' memory ran out.

4.2. A Global Scheme. Due to the global property in time and space of the control problem there seems less convenient to use step by step discretization in time. Thus we here approximate the state and control by the finite element method with an isotropic treatment of the variables x and t.

We first give the discretization of the state. Let Ω^h be a polygonal approximation to Ω with the boundary $\partial\Omega^h$, $Q = \Omega \times (0,T)$, and $Q^h = \Omega^h \times (0,T)$. We consider conforming $(n+1)$-simplex Lagrange elements on Q^h. Let T^h be a partitioning of Q^h into disjoint regular $(n+1)$-simplex τ, so that $\bar{Q}^h = \cup_{\tau \in T^h} \bar{\tau}$.

Associated with T^h is a finite dimensional subspace S^h of $C(\bar{Q}^h)$, such that $w|_\tau$ are polynomials of m-order ($m \geq 1$) for all $w \in W^h$ and $\tau \in T^h$. Let $W^h = S^h \cap L^2(0,T;H_0^1(\Omega))$ and h_τ denote the maximum diameter of the element τ in T^h.

The partitioning of the domain $Q_U = \Omega_U \times (0,T)$ is the same but with the notations $T_U^h, \tau_U, Q_U^h = \Omega_U^h \times (0,T), \partial Q_U^h$, and h_{τ_U}. Associated with T_U^h is a finite dimensional subspace X^h of $L^2(Q_U^h)$, such that $v|_{\tau_U}$ are polynomials of m-order ($m \geq 0$) for all $v \in X^h$ and $\tau_U \in T_U^h$. Here there is no requirement for the continuity.

Let h_{τ_U} denote the maximum diameter of the element τ_U in T_U^h. Let $K^h \subset (X^h \cap K)$. The fully discrete approximation scheme is to find (y_h, u_h) such that

$$\min_{u_h \in K^h} \int_0^T g(y_h) + h(u_h),$$

$$\int_0^T ((\partial_t y_h, w) + a(y_h, w))\,dt + (y_h(0) - y_0, w(0)) = \int_0^T (f + Bu_h, w)\,dt,$$
$$\forall w \in W^h.$$

with $y_0^h \in W^h|_{t=0}$ being an approximation to y_0. We have similar optimality conditions.

This scheme needs to solve full linear systems, which are neither symmetric or positive. In our computation, we used some AMG solvers.

A Posteriori Error Estimators. Let

$$K^h = \{v \in X^h:\ v \geq \alpha \text{ a.e. in } Q_U\} \subset K.$$

Similarly we divide Q_U into the following three subsets:

$$\begin{aligned}
Q_\alpha^- &= \{(x,t) \in Q_U:\ (B^* p_h)(x,t) \leq -j'(\alpha)\},\\
Q_\alpha^0 &= \{(x,t) \in Q_U:\ (B^* p_h)(x,t) > -j'(\alpha),\ u_h = \alpha\},\\
Q_\alpha^+ &= \{(x,t) \in Q_U:\ (B^* p_h)(x,t) > -j'(\alpha),\ u_h > \alpha\}.
\end{aligned}$$

Then, it is easy to see that above three subsets are not overlapped each other, and

$$\bar{Q}_U = \bar{Q}_\alpha^- \cup \bar{Q}_\alpha^0 \cup \bar{Q}_\alpha^+.$$

Again only error on Ω_α^- is important.

THEOREM 4.2. *Let (y,p,u) and (y_h, p_h, u_h) be the solutions of the control problem and its approximation via the global scheme respectively. Assume K^h is defined as above. Moreover, assume that $j'(\cdot)$ and $g'(\cdot)$ are locally Lipschitz continuous. Then*

$$\|u_h - u\|^2_{L^2(0,T;L^2(\Omega_U)} + \|y - y_h\|^2_{L^2(Q)} + \|p - p_h\|^2_{L^2(Q)} \leq C(\hat{\eta}_1^2 + \hat{\eta}_2^2 + \hat{\eta}_3^2),$$

where

$$\hat{\eta}_1^2 = \|j'(u_h) + B^* p_h\|^2_{0,Q_\alpha^-},$$

and $\hat{\eta}_2^2 + \hat{\eta}_3^2$ *consists of the standard residual a posteriori error indicators for the state and co-state equations.*

For the details of the proofs, the readers are referred to [**12**]. This scheme can produce perfect meshes to resolve complicated shapes of singularities. However it does need to solve full linear systems, which are neither symmetric or positive, though the overall computational work is often less when u is singular.

5. Numerical Results: Time Matching via Global Solvers

In our numerical tests, we used the following simple black-box algorithm (see [**12**]):

$$(u^{k+1/2}, v_h)_{L^2} = (u^k, v_h) + \rho_k(J'(u^k), v_h),$$

$$u^{k+1} \geq 0 : (u^{k+1} - u^{k+1/2}, v_h - u^{k+1}) \geq 0,\ \forall v_h \geq 0,$$

with

$$(J'(u^k), v_h) = \int_0^T (h'(u^k) + B^* p^k, v_h), v_h \in U^h$$

$$y_t^k + a(y^k, w_h) = (f + Bu^k, w_h), \quad w_h \in V^h,$$

$$-p_t^k + a(q_h, p^k) = (g(y^k), q_h), \quad q_h \in V^h.$$

The main computational work is to solve the coupled state-costate equations repeatedly - using iterations $u_k \to y_k \to p_k \to u_{k+1}$. Using the backward Euler scheme, one enjoys time matching solvers for the two equations. However, mesh refinement in time has to be global. Thus when the solution has singularity crossing $[0,T] \times \Omega$, the resulting time-mesh refinement is often just a uniform one.

Using the global scheme, mesh refinement is carried out locally in x-t space and parallelization is easier. But one has to solve two equations globally with respect to x-t. For control problems, the global scheme normally results in much better mesh-refinement, and thus much smaller equations, unless the control is smooth.

For the second scheme, we apply the standard mesh refinement technique mentioned above on the whole space-time domain. While for the first scheme, the temporal direction is divided uniformly and we didn't apply any adaptation in this direction, since it is clear that adaptive refinement in temporal direction for the back-Euler scheme will only lead to uniform like meshes for our examples. For the spatial variables, we apply the mesh adaptation independently on every time step. We obtain different meshes on different time steps, and those meshes are adapted according to their own indicators. We shall use η_1 ($\hat{\eta}$) as the control mesh refinement indicator, and $\eta_2 + \eta_3$ ($\hat{\eta}_2 + \hat{\eta}_3$) as the state's and co-state's for the backward Euler scheme (the global scheme). The details of the implementation can be found in [**7**] and [**12**]. In all our experiments, the numerical examples are the following type of optimal control problems:

$$\begin{aligned} &\min \quad \int_0^1 (g(y) + h(u))dt \\ &\text{s.t.} \quad \partial_t y - \Delta y + \phi(y) = u + f, \quad u \geq 0. \end{aligned} \tag{5.1}$$

in which

$$g(y) = \int_\Omega (y-y_0)^2\,dx$$
$$h(u) = \int_\Omega (u-u_0)^2\,dx$$

and the spatial domain Ω here is set as the unit square $[0,1]\times[0,1]$. Let Ω^h and Ω_U^h be partitioned into T^h and T_U^h as described. Piecewise linear finite elements have been used for both the schemes and the examples. We may use different meshes for the approximation of the state and the control.

Example 1. For the first example, the data are as follows:

$$\begin{aligned} \mu(x) &= \sin\pi x_1 \sin\pi x_2 \\ \nu(t) &= \sin\pi t \\ p(x,t) &= \mu(x)\nu(t) \\ z(x,t) &= \begin{cases} 1/2, & \text{if } x_1+x_2>1.0 \\ 0, & \text{otherwise} \end{cases} \\ u_0(x,t) &= 1-\sin\frac{\pi x_1}{2}-\sin\frac{\pi x_2}{2}+z \\ u(x,t) &= \max(u_0-p,\ 0) \\ y_0(x,t) &= 0 \\ y(x,t) &= -\partial_t p-\Delta p+y_0 \\ f(x,t) &= \partial_t y-\Delta y-u \\ \phi(s) &= 0. \end{aligned} \tag{5.2}$$

The optimal control has a strong jump (discontinuity), introduced by u_0. In Table (4) numerical results are presented for the first scheme with 41 time steps, while results for the second scheme are in Table (5). It can be seen that for both schemes, adaptive meshes can substantially save computational work, since the main computing work required in the projection algorithm is to solve the state and co-state equations repeatedly. It is clear that the global scheme can achieve better approximation using much fewer y,p dofs. However, the linear system from the backward Euler scheme can be solved via time-matching, while this is impossible for the linear system from the global scheme. It was found that for the same total dofs the backward Euler system can be solved faster, though it yields less accuracy. To increase the approximation accuracy, one has to further refine the space grids. This leads to a larger backward Euler linear system so that both the schemes have a similar computational load for this example. This is mainly because the jump only expand or contract, and thus is relatively simpler in the x-t space.

Example 2. In this example, the state equation is replaced by the following nonlinear equation:

$$\partial_t y-\Delta y+y^3=u+f \tag{5.3}$$

and the dual equation now is

$$-\partial_t p-\Delta p+3y^2p=y-y_0. \tag{5.4}$$

		on uniform mesh			on adaptive mesh		
		u	y	p	u	y	p
mesh info	# nodes	82600	84665	84665	28718	16678	16678
	# sides	241280	247312	247312	74652	46857	46857
	# elements	158720	162688	162688	45974	30220	30220
	# dofs	476160	84665	84665	137922	16678	16678
$\sum_i \lvert u(t_i) - u_h(t_i)\rvert_{L^2}$		1.530e-02	4.777e-02	5.173e-02	1.148e-02	6.668e-02	5.154e-02
Total L^2 error		1.594e-02	1.685e-01	5.233e-02	1.232e-02	1.746e-01	5.214e-02

TABLE 4. Example 1 via backward Euler with time step 41

		on uniform mesh			on adaptive mesh		
		u	y	p	u	y	p
mesh info	# nodes	56853	56853	56853	29633	9339	9339
	# edges	383764	383764	383764	176988	59472	59472
	# faces	646912	646912	646912	279910	97547	97547
	# elements	320000	320000	320000	132554	47413	47413
	# dofs	1280000	56853	56853	530216	9339	9339
Total L^2 error		7.0833e-03	1.6213e-02	1.9210e-03	6.1029e-03	5.6538e-02	5.5697e-03

TABLE 5. Example 1 via global discretization

The problem is still well-posed with such a special nonlinear term. The data under testing are as follows:

$$\begin{aligned}
\mu(x) &= \sin 2\pi x_1 \sin 2\pi x_2 \\
\nu(t) &= \sin 2\pi t \\
p(x,t) &= 0 \\
u_0(x,t) &= \max(4\pi^2 \mu(x)\nu(t), 0) \\
u(x,t) &= u_0 \\
y_0(x,t) &= \mu(x)\nu(t) \\
y(x,t) &= y_0 \\
f(x,t) &= \partial_t y - \Delta y + \phi(y) - u \\
\phi(s) &= s^3.
\end{aligned} \tag{5.5}$$

Some numerical results are presented in Table (6) - Table (8). For the backward Euler scheme one has to uniformly refine the time-meshes to reduce approximation errors, since the total L^2 error in the space variables at each time step is already of higher order comparing with the total approximation L^2 error. Indeed we doubled the time grids, and reduced the total error by a half. However it is clear that to achieve the same error tolerance as the global scheme, the total dofs have to be much more than those of the global scheme. In fact, they will be of order difference. Thus the global scheme is much more efficient than the backward Euler scheme in this example.

5.1. Mix time-matching with global strategy. For this scheme, we use the backward Euler scheme in time discretization, and the finite element method in space discretization for the state and co-state equations. We however approximate the control by the finite element method with an isotropic treatment of the variables x and t. The motivation is that typically it is the control that has complicated singularities.

		on uniform mesh			on adaptive mesh		
		u	y	p	u	y	p
mesh info	# nodes	82600	84665	84665	93339	8097	8097
	# sides	241280	247312	247312	263569	21773	21773
	# elements	158720	162688	162688	170270	13717	13717
	# dofs	476160	84665	84665	510810	8097	8097
$\sum_i \|u(t_i) - u_h(t_i)\|_{L^2}$		6.8699e-02	7.9276e-03	1.4275e-04	3.4834e-02	2.1368e-03	1.0385e-04
Total L^2 error		8.9763e-01	3.3280e-02	*	8.9567e-01	3.8533e-02	*

TABLE 6. Example 2 via backward Euler with time step 41

		on uniform mesh			on adaptive mesh		
		u	y	p	u	y	p
mesh info	# nodes	165200	167265	167265	188219	16315	16315
	# sides	482560	488592	488592	531166	43366	43366
	# elements	317440	321408	321408	343027	27312	27312
	# dofs	952320	167265	167265	1029081	16315	16315
$\sum_i \|u(t_i) - u_h(t_i)\|_{L^2}$		6.8699e-02	4.4839e-03	7.9721e-04	3.4517e-02	1.9077e-03	2.5011e-04
Total L^2 error		4.5277e-01	1.6704e-02	*	4.4886e-01	2.4874e-02	*

TABLE 7. Example 2 via backward Euler with time step 81

		on uniform mesh			on adaptive mesh		
		u	y	p	u	y	p
mesh info	# nodes	56853	56853	56853	47866	7536	7536
	# edges	383764	383764	383764	288927	49057	49057
	# faces	646912	646912	646912	460374	81298	81298
	# elements	320000	320000	320000	219312	39776	39776
	# dofs	1280000	56853	56853	877248	7536	7536
Total L^2 error		6.6950e-02	4.7943e-03	7.9111e-05	5.6611e-02	1.8451e-03	3.0143e-05

TABLE 8. Example 2 via global discretization

For any function $w \in C(0,T;L^2(\Omega))$, let $\hat{w}(x,t)|_{t\in(t_{k-1},t_k]} = w(x,t_k)$, $\bar{w}(x,t)|_{t\in(t_{k-1},t_k]} = \int_{t_{k-1}}^{t_k} w(x,t)dt/\Delta t_k$. Then the fully discrete approximation scheme is to find (y_h, u_h) such that

$$\min_{u_h\in K^h} \int_0^T (g(\hat{y}_h) + h(u_h)), \tag{5.6}$$

$$(\frac{\partial y_h}{\partial t}, w_h) + a(\hat{y}_h, w_h) = (\bar{f} + B\bar{u}_h, w_h), \tag{5.7}$$

$$\forall w_h \in V_i^h \subset V = H_0^1(\Omega), \ \ t \in (t_{i-1}, t_i], \ \ i = 1, 2, \cdots, N,$$

$$y_h(x,0) = y_0^h(x), \ \ x \in \Omega.$$

We can derive a posteriori error estimates. Again we divide Q_U into three subsets:

$$Q_\alpha^- = \{(x,t) \in Q_U : \ (B^*\tilde{p}_h)(x,t) \le -j'(\alpha)\},$$
$$Q_\alpha^0 = \{(x,t) \in Q_U : \ (B^*\tilde{p}_h)(x,t) > -j'(\alpha), \ u_h = \alpha\},$$
$$Q_\alpha^+ = \{(x,t) \in Q_U : \ (B^*\tilde{p}_h)(x,t) > -j'(\alpha), \ u_h > \alpha\}.$$

Then, it is easy to see that above three subsets are not overlapped each other, and

$$\bar{Q}_U = \bar{Q}_\alpha^- \cup \bar{Q}_\alpha^0 \cup \bar{Q}_\alpha^+.$$

THEOREM 5.1. *Let (y, p, u) and (y_h, p_h, u_h) be the solutions of the control problem and its approximation via the mixed scheme respectively. Assume that $j'(\cdot)$ and $g'(\cdot)$ are locally Lipschitz continuous. Then*

$$\|u_h - u\|^2_{L^2(0,T;L^2(\Omega_U)} + \|y - y_h\|^2_{L^2(Q)} + \|p - p_h\|^2_{L^2(Q)} \leq C(\underline{\hat{\eta}}^2_1 + \underline{\hat{\eta}}^2_2 + \underline{\hat{\eta}}^2_3),$$

where

$$\underline{\hat{\eta}}^2_1 = \|j'(u_h) + B^*\tilde{p}_h\|^2_{0,Q^-_\alpha},$$

and $\underline{\hat{\eta}}^2_2 + \underline{\hat{\eta}}^2_3$ consists of the standard residual a posteriori error indicators for the state and co-state equations.

The readers are referred to [**13**] for the details. However currently we were not able to efficiently implement the scheme yet. The reason is that in our previous implementation, we had to assume some relationships between the two set of meshes, but these relationships can no longer be assumed for this mixed scheme.

References

1. N. Arada, J.-P. Raymond, and F. Troltzsch, *On an augmented Lagrangian SQP method for a class of optimal control problems in Banach spaces*, Computational Optimization and Applications **22** (2002), 369–398.
2. R. Becker, H. Kapp, and R. Rannacher, *Adaptive finite element methods for optimal control of partial differential equations: Basic concept*, SIAM J. Control Optimization **39** (2000), 113–132.
3. R. Becker and R. Rannacher, *An optimal control approach to a-posteriori error estimation*, Acta Numerica (A. Iserles, ed.), Cambridge University Press, 2001, pp. 1–102.
4. M. Bergounioux, M. Haddou, M. Hintermuller, and K. Kunisch, *A comparison of a Moreau-Yosida based active set strategy and interior point methods for constrained optimal control problems*, SIAM J. Optim. **11** (2000), 495–521.
5. P.G. Ciarlet, *The finite element method for elliptic problems*, Amsterdam, North-Holland, 1978.
6. M. Hintermuller, K. Ito, and K. Kunisch, *The primal-dual active set strategy as semi-smooth Newton method*, Tech. Report 214, Technische Universitat Graz, 2001.
7. Y. Huang, R. Li, W.B. Liu, and N.N. Yan, *Adaptive multi-mesh finite element approximation for constrained optimal control problems*, submitted.
8. K. Ito and K. Kunisch, *The augmented Lagrangian method for parameter estimation in elliptic systems*, SIAM J. Control. Optim. **28** (1990), 113–136.
9. C.T. Kelley and E.W. Sachs, *A trust region method for parablic boundary control problems*, SIAM J. Optim. **9** (1999), 1064–1091.
10. R. Li, *On multi-mesh h-adaptive methods*, J. Sci. Comput., accepted, 2004.
11. R. Li, W. B. Liu, H.P. Ma, and T. Tang, *Adaptive finite element approximation of elliptic optimal control*, SIAM J. Control. Optim. **41** (2002), 1321–1349.
12. R. Li, W.B. Liu, H.P. Ma, and N.N. Yan, *Adaptive finite element approximation for distributed optimal control governed by parabolic equations*, submitted.
13. ———, *Adaptive finite element approximation for distributed optimal control governed by parabolic equations -II*, in preparation.
14. R. Li, W.B. Liu, and N.N. Yan, *A posteriori error estimates of recovery type for convex optimal control*, submitted.
15. W. B. Liu, H.P. Ma, T. Tang, and N.N. Yan, *A posteriori error estimates of DG time-stepping method for optimal control problems governed by parabolic equations*, SIAM J. Numer. Anal **42** (2004), 1032–1061.
16. W.B. Liu and N.N. Yan, *A posteriori error estimates for optimal control of Stokes flows*, SIAM J. Numer. Anal. **40** (2003), 1805–1869.
17. ———, *A posteriori error estimates for optimal control problems governed by parabolic equations*, Numer. Math. **93** (2003), 497–521.

Institute of Mathematics and Statistics, University of Kent, Canterbury, CT2 7PE, UK

E-mail address: W.B.Liu@ukc.ac.uk